

2 1978

REPUBLICA ARGENTINA
COMISION NACIONAL DE ENERGIA ATOMICA

GERENCIA DE INVESTIGACIONES
DEPARTAMENTO DE FISICA

GUIA DE MODELOS
PARA EL USO DE TARJETAS PERFORADAS
EN LENGUAJE IBM DE CONTROL DE TRABAJO
EN SISTEMA OPERATIVO

ALICIA M. DIAZ ROMERO

BUENOS AIRES
1978

REPUBLICA ARGENTINA
COMISION NACIONAL DE ENERGIA ATOMICA

GERENCIA DE INVESTIGACIONES
DEPARTAMENTO DE FISICA

GUIA DE MODELOS
PARA EL USO DE TARJETAS PERFORADAS
EN LENGUAJE IBM DE CONTROL DE TRABAJO
EN SISTEMA OPERATIVO

ALICIA M. DIAZ ROMERO

BUENOS AIRES
1978

I N D I C E I.

PLAN GENERAL DE LA GUIA	III
INTRODUCCION	V
INDICE II: TEMARIO DE CADA CAPITULO	423
INDICE III: DESCRIPCION DE LOS MODELOS	431
INDICE IV: UBICACION DE LOS CAPITULOS Y SUS PUNTOS EN EL INDICE III	473
AGRADECIMIENTOS	475
BIBLIOGRAFIA	477

USO DE LA GUIA

Consultar primero el PLAN GENERAL DE LA GUIA y elegir el CAPITULO que corresponda; luego el INDICE II para seleccionar el PUNTO adecuado dentro de ese capítulo; finalmente el INDICE III conducirá al modelo necesario o similar.

El INDICE IV tiene por objeto facilitar la ubicación de los CAPITULOS y sus respectivos PUNTOS dentro del INDICE III que es muy extenso.

PLAN GENERAL DE LA GUIA

CAPITULO I.

Modelos para operar con programas perforados en tarjetas que no utilizan bibliotecas catalogadas.....	1

CAPITULO II.

Modelos para el uso de una biblioteca particionada y catalogada de programas en lenguaje simbólico que reside en una unidad de discos	33

CAPITULO III.

Modelos para la ejecución de programas en FORTRAN que se presentan en distintos soportes y utilizan una biblioteca particionada y catalogada de programas en lenguaje simbólico como archivos de entrada y salida de datos.....	87

CAPITULO IV.

Modelos para el uso de una biblioteca particionada y catalogada de módulos de carga que reside en una unidad de discos.....	143

CAPITULO V.

Modelos para el uso simultáneo de dos bibliotecas particionadas: una de programas en lenguaje simbólico y otra de módulos de carga.	173

CAPITULO VI.

Modelos para la creación y mantenimiento de bibliotecas particionadas y catalogadas.....	241

CAPITULO VII.

Modelos para listado, grabación y transferencia de archivos en cintas magnéticas.....	263

CAPITULO VIII.

Modelos para la ejecución de programas grabados en cintas magnéticas.....	297

CAPITULO IX.

Modelos para la ejecución de programas que utilizan cintas magnéticas como archivos de entrada de datos.....	325

CAPITULO X.

Modelos para la ejecución de programas que utilizan cintas magnéticas como archivos de salida de datos.....	339

CAPITULO XI.

Modelos para la ejecución de programas que utilizan cintas magnéticas como archivos de entrada y salida de datos simultáneamente..	367

CAPITULO XII.

Modelos para la utilización de cintas magnéticas como archivos de módulos de programas o de bibliotecas particionadas cuando no se dispone de espacio propio en una unidad de discos.....	405

I N T R O D U C C I O N

La presente colección de modelos está destinada a los científicos y tecnólogos para quienes la computación es simplemente una herramienta a cuyo acabado dominio no le pueden dedicar ni demasiado tiempo ni demasiado esfuerzo. El Lenguaje IBM de Control de Trabajo es en si mismo un lenguaje simbólico de alto nivel que aparece detalladamente explicado en los manuales editados por la firma IBM y en folletos cuyos autores procuran transmitir al profano, aclarado y simplificado, su propio conocimiento de dicho lenguaje. No he tenido la intención de escribir un nuevo manual; antes bien, considero excelentes los que se encuentran habitualmente en los laboratorios y encarezco su consulta cuando se desee ampliar cualquier punto. El objeto de esta publicación es ofrecer, ya preparado y ensayado, el modelo de programa que el profesional necesita en un momento determinado. La conservación de los modelos en tarjetas perforadas y su archivo por temas, fué utilizada con mucho éxito en el sector Programación del Departamento de Física de la CNEA y de ahí surgió la idea de verter esta experiencia acumulada a lo largo de seis años, por escrito, para facilitar su ordenamiento y difusión. Los 500 modelos reunidos aquí, no representan en ninguna forma la totalidad de los numerosísimos ejemplos posibles, sino mas bien una selección de aquéllos de uso mas corriente en la CNEA.

Un constante punto de mira rige el planeamiento de esta Guía: presentar cada modelo completo, con todos sus pasos y sus notas, aun a costa de que el conjunto aparezca reiterativo; he preferido esta crítica, a la que me anticipo, antes que fatigar al usuario con la continua búsqueda de citas de páginas, programas, pasos o notas anteriores. También, dada la suficiente diferencia que en caracteres de máquina de escribir existe entre el cero y la letra O mayúscula, he evitado expresar el uso de la Ø para identificar a esta última, por considerarla incómoda para el lector.

Para estructurar los modelos en los cuales el programa principal está codifi-

cado en un lenguaje y las rutinas o funciones que invoca lo están en otro, he combinado el FORTRAN y el ASSEMBLER, cada uno en sus variedades (simbólico y absoluto) con los distintos soportes (tarjetas, cintas magnéticas, discos auxiliares y bibliotecas particionadas). Los modelos en PL/1 han sido considerados como codificados siempre en un solo lenguaje (simbólico o absoluto), por saberlo tan poderoso que difícilmente requiera, en la práctica, un modelo mixto. A los efectos de que las variantes disponibles no eleven el número de modelos a muchos centenares, cuando un programa lee datos de entrada o graba resultados en distintos soportes, he empleado únicamente el FORTRAN y en el caso de salida impresa he supuesto, para todos los lenguajes, que ésta se realiza desde el programa principal solamente. De todas maneras, el FORTRAN es el que aparece con mas frecuencia, luego el ASSEMBLER y finalmente el PL/1, por ser ese el orden en que se usan en el Departamento de Física de la CNEA.

He obviado los anglicismos de la jerga que se habla en los centros de cómputos, basándome en que esgrachear, printear, punchear, lodear y otros, tienen su correspondiente y correcta palabra en el idioma nacional. Por la misma razón, entre las posibles traducciones de la expresión inglesa "linkage editor" he elegido como la mas conveniente "editor de vínculos", de manera que, de un procedimiento FORTRAN tal como FTG1CLG, se dirá que compila, vincula y ejecuta.

Por último, cada modelo no es necesariamente el ideal en cuanto a concepción, número de sentencias de control y tiempo de máquina; tampoco es único pues un mismo problema en general se puede encarar y resolver de varias maneras diferentes; en todo caso su mérito consiste principalmente en haber sido, sin excepción, probados en la práctica.

Buenos Aires, Octubre de 1978

CAPITULO I

MODELOS PARA OPERAR CON PROGRAMAS PERFORADOS EN
TARJETAS QUE NO UTILIZAN BIBLIOTECAS CATALOGADAS

PUNTO 1.

Ejecución de programas en tarjetas perforadas, codificados totalmente en un mismo lenguaje.

Modelo 001 . Ejecución de un programa en lenguaje simbólico FORTRAN.

```
//..... JOB ....
//PASO1 EXEC PROC=FTG1CLG
//FORT.SYSIN DD *
  Aquí tarjetas perforadas en FORTRAN
/*
//GO.SYSIN DD *
  Aquí tarjetas perforadas de DATOS
/*
//
```

Modelo 002 . Ejecución de un programa en lenguaje simbólico ASSEMBLER.

```
//..... JOB ....
//PASO1 EXEC PROC=ASMFCLG
//ASM.SYSIN DD *
  Aquí tarjetas perforadas en ASSEMBLER
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: En el programa en ASSEMBLER se ha abierto un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

Modelo 003 . Ejecución de un programa en lenguaje simbólico PL/1.

```
//..... JOB ....
//PASO1 EXEC PROC=PLIXCLG,PARM.PLI='CS(60) '
//PLI.SYSIN DD *
  Aquí tarjetas perforadas en PL/1
/*
//GO.SYSIN DD *
  Aquí tarjetas perforadas de DATOS
/*
//
```


Modelo 004 . Ejecución de un programa en lenguaje absoluto de FORTRAN.

```
//..... JOB ....
//PAS01 EXEC PROC=FTG1LG
//LKED.SYSIN DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas en absoluto de FORTRAN
/*
//GO.SYSIN DD *
  Aquí tarjetas perforadas de DATOS
/*
//
```

Modelo 005 . Ejecución de un programa en lenguaje absoluto de ASSEMBLER.

```
//..... JOB ....
//PAS01 EXEC PROC=LKEDG
//LKED.SYSIN DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas en absoluto de ASSEMBLER
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: En el programa ASSEMBLER que dió origen al lote de tarjetas en absoluto, se abrió un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

Modelo 006 . Ejecución de un programa en lenguaje absoluto de PL/1.

```
//..... JOB ....
//PAS01 EXEC PROC=PLIXLG
//LKED.SYSIN DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas en absoluto de PL/1
/*
//GO.SYSIN DD *
  Aquí tarjetas perforadas de DATOS
/*
//
```

PUNTO 2.

Obtención de programas objeto en tarjetas perforadas, a partir de programas en lenguaje simbólico, que también se presentan en tarjetas perforadas.

Modelo 007 . Obtención de un programa objeto en lenguaje absoluto de FORTRAN.

```
//..... JOB ....
//PAS01 EXEC PROC=FTG1C,PARM.FORT=DECK
//FORT.SYSPUNCH DD SYSOUT=B
//FORT.SYSIN DD *
  Aquí tarjetas perforadas en FORTRAN
/*
//
```

Modelo 008 . Obtención de un programa objeto en lenguaje absoluto de ASSEMBLER.

```
//..... JOB ....
//PAS01 EXEC PROC=ASMFC,PARM.ASM=DECK
//ASM.SYSPUNCH DD SYSOUT=B
//ASM.SYSIN DD *
  Aquí tarjetas perforadas en ASSEMBLER
/*
//
```

Modelo 009 . Obtención de un programa objeto en lenguaje absoluto de PL/1

```
//..... JOB ....
//PAS01 EXEC PROC=PLIXC,PARM.PLI='DECK,CS(60)'
//PLI.SYSPUNCH DD SYSOUT=B
//PLI.SYSIN DD *
  Aquí tarjetas perforadas en PL/1
/*
//
```

PUNTO 3.

Obtención de programas objeto grabados en cintas magnéticas, a partir de programas en lenguaje simbólico, que se presentan en tarjetas perforadas.

Modelo 010 . Obtención de un programa objeto en lenguaje absoluto de FORTRAN y su grabación, con el nombre ABFORPEG, en el rótulo 3 de una cinta magnética que lleva el número de carrete 13854 y tiene etiquetas modelo IBM.

```
//..... JOB ....
//PAS01 EXEC PROC=FTG1C,PARM.FORT=DECK
//FORT.SYSPUNCH DD DSN=ABFORPEG,UNIT=TAPE,VOL=SER=T13854,LABEL=(3,SL),
// DISP=(NEW,KEEP),DCB=BLKSIZE=80
//FORT.SYSIN DD *
Aquí tarjetas perforadas en FORTRAN
/*
//
```

Modelo 011 . Obtención de un programa objeto en lenguaje absoluto de ASSEMBLER y su grabación, con el nombre ABASSPEG, en el rótulo 7 de una cinta magnética que lleva el número de carrete 46621 y tiene etiquetas modelo IBM.

```
//..... JOB ....
//PAS01 EXEC PROC=ASMFC,PARM.ASM=DECK
//ASM.SYSPUNCH DD DSN=ABASSPEG,UNIT=TAPE,VOL=SER=T46621,LABEL=(7,SL),
// DISP=(NEW,KEEP),DCB=BLKSIZE=80
//ASM.SYSIN DD *
Aquí tarjetas perforadas en ASSEMBLER
/*
//
```

Modelo 012 . Obtención de un programa objeto en lenguaje absoluto de PL/1 y su grabación, con el nombre ABPL1PEG, en el rótulo 14 de una cinta magnética que lleva el número de carrete 07910 y tiene etiquetas modelo IBM.

```
//..... JOB ....
//PAS01 EXEC PROC=PLIXC,PARM.PLI='DECK,CS(60)'
//PLI.SYSPUNCH DD DSN=ABPL1PEG,UNIT=TAPE,VOL=SER=T07910,LABEL=(14,SL),
// DISP=(NEW,KEEP),DCB=BLKSIZE=80
//PLI.SYSIN DD *
Aquí tarjetas perforadas en PL/1
/*
//
```

PUNTO 4.

Listado de tarjetas perforadas.

Modelo 013 . Obtención del listado de programas, rutinas, funciones o datos numéricos perforados en tarjetas.

```
//..... JOB ....  
//PAS01 EXEC PGM=IEBPTPCH  
//SYSPRINT DD SYSOUT=A  
//SYSUT2 DD SYSOUT=A  
//SYSUT1 DD *  
  Aquí tarjetas perforadas  
/*  
//SYSIN DD *  
  PRINT MAXFLDS=1  
  RECORD FIELD=(80)  
/*  
//
```

PUNTO 5.

Ejecución de programas que invocan rutinas y funciones, codificadas en otro lenguaje, que se presentan en tarjetas perforadas.

Modelo 014 . Ejecución de un programa en lenguaje simbólico FORTRAN, que invoca rutinas y funciones en lenguaje absoluto de FORTRAN o de ASSEMBLER.

```
//..... JOB ....
//PAS01 EXEC PROC=FTG1CLG
//FORT.SYSIN DD *
  Aquí tarjetas perforadas en FORTRAN
/*
//LKED.SYSIN DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas en absoluto de FORTRAN o de ASSEMBLER
/*
//GO.SYSIN DD *
  Aquí tarjetas perforadas de DATOS
/*
//
```

Modelo 015 . Ejecución de un programa en lenguaje simbólico FORTRAN, que invoca rutinas y funciones en lenguaje simbólico ASSEMBLER. El problema se puede resolver en dos pasos: en el PAS01 el procedimiento ASMFC compila las rutinas y funciones en ASSEMBLER, cargando el módulo resultante en un disco auxiliar, con el nombre OTROLENG; en el PAS02 el procedimiento FTG1CLG compila el programa en FORTRAN, lo vincula con el módulo OTROLENG y ejecuta.

```
//..... JOB ....
//PAS01 EXEC PROC=ASMFC,PARM.ASM=DECK
//ASM.SYSPUNCH DD DSN=OTROLENG,DISP=(,PASS),UNIT=WORKDISK,
// SPACE=(CYL,(1,1)),DCB=BLKSIZE=80
//ASM.SYSIN DD *
  Aquí tarjetas perforadas en ASSEMBLER
/*
//PAS02 EXEC PROC=FTG1CLG
//FORT.SYSIN DD *
  Aquí tarjetas perforadas en FORTRAN
/*
//LKED.SYSIN DD DSN=OTROLENG,UNIT=WORKDISK,DISP=(OLD,DELETE)
/*
//GO.SYSIN DD *
  Aquí tarjetas perforadas de DATOS
/*
//
```

Modelo 016 . Ejecución de un programa en lenguaje absoluto de FORTRAN, que invo-

ca rutinas y funciones en lenguaje simbólico FORTRAN.

```
//..... JOB ....
//PASO1 EXEC PROC=FTG1CLG
//FORT.SYSIN DD *
  Aquí tarjetas perforadas en FORTRAN
/*
//LKED.SYSIN DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas en absoluto de FORTRAN
/*
//GO.SYSIN DD *
  Aquí tarjetas perforadas de DATOS
/*
//
```

Modelo 017 . Ejecución de un programa en lenguaje absoluto de FORTRAN, que invoca rutinas y funciones en lenguaje simbólico ASSEMBLER. El problema se puede resolver en dos pasos: en el PASO1 el procedimiento ASMFC, compila las rutinas y funciones en ASSEMBLER, cargando el módulo resultante en un disco auxiliar, con el nombre SASEGUN; en el PASO2 el procedimiento FTG1LG vincula el programa en absoluto de FORTRAN con el módulo SASEGUN y ejecuta.

```
//..... JOB ....
//PASO1 EXEC PROC=ASMFC,PARM.ASM=DECK
//ASM.SYSPUNCH DD DSN=SASEGUN,DISP=(,PASS),UNIT=WORKDISK,
//  SPACE=(CYL,(1,1)),DCB=BLKSIZE=80
//ASM.SYSIN DD *
  Aquí tarjetas perforadas en ASSEMBLER
/*
//PASO2 EXEC PROC=FTG1LG
//LKED.SYSIN DD DSN=SASEGUN,UNIT=WORKDISK,DISP=(OLD,DELETE)
//          DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas en absoluto de FORTRAN
/*
//GO.SYSIN DD *
  Aquí tarjetas perforadas de DATOS
/*
//
```

Modelo 018 . Ejecución de un programa en lenguaje absoluto de FORTRAN, que invoca rutinas y funciones en lenguaje absoluto de ASSEMBLER.

```
//..... JOB ....
//PASO1 EXEC PROC=FTG1LG
//LKED.SYSIN DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas en absoluto de FORTRAN
  Aquí tarjetas perforadas en absoluto de ASSEMBLER
/*
```

```
//GO.SYSIN DD *
Aquí tarjetas perforadas de DATOS
/*
//
```

Modelo 019 . Ejecución de un programa en lenguaje simbólico ASSEMBLER, que invoca rutinas y funciones en lenguaje absoluto de ASSEMBLER.

```
//..... JOB ....
//PAS01 EXEC PROC=ASMFCLG
//ASM.SYSIN DD *
Aquí tarjetas perforadas en ASSEMBLER
//LKED.SYSIN DD *,DCB=BLKSIZE=80
Aquí tarjetas perforadas en absoluto de ASSEMBLER
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: En el programa principal ASSEMBLER se ha abierto un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

Modelo 020 . Ejecución de un programa en lenguaje simbólico ASSEMBLER, que invoca rutinas y funciones en lenguaje simbólico FORTRAN. El problema se puede resolver en tres pasos: en el PAS01 el procedimiento ASMFCL compila y vincula el programa ASSEMBLER, cargando el módulo resultante, con el nombre FUREXMON, en una biblioteca temporaria particionada que se crea en ese momento y reside en un disco auxiliar; en el PAS02 el procedimiento FTG1CL compila y vincula las rutinas y funciones en FORTRAN, cargando el módulo resultante, con el nombre MONO, en la misma biblioteca temporaria creada en el paso anterior; en el PAS03 el procedimiento FTG1LG vincula los módulos FUREXMON y MONO y ejecuta.

```
//..... JOB ....
//PAS01 EXEC PROC=ASMFCL
//ASM.SYSIN DD *
Aquí tarjetas perforadas en ASSEMBLER
//LKED.SYSLMOD DD DSN=TEMPOR.ADR(FUREXMON),DISP=(NEW,PASS),
// UNIT=WORKDISK,SPACE=(CYL,(1,1,1)),
// DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
/*
//PAS02 EXEC PROC=FTG1CL
```

```
//FORT.SYSIN DD *
  Aquí tarjetas perforadas en FORTRAN
//LKED.SYSLMOD DD DSN=TEMPOR.ADR(MONO),DISP=OLD,
//  VOL=REF=*.PAS01.LKED.SYSLMOD
/*
//PAS03 EXEC PROC=FTG1LG
//LKED.OTRALIB DD DSN=TEMPOR.ADR,DISP=(OLD,DELETE)
//LKED.SYSIN DD *,DCB=BLKSIZE=80
      INCLUDE OTRALIB(FUREXMON)
      INCLUDE OTRALIB(MONO)
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: En el PAS01 se menciona el DCB con el cual se grabará el disco auxiliar. En el programa principal ASSEMBLER se ha abierto un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

Modelo 021 . Ejecución de un programa en lenguaje simbólico ASSEMBLER, que invoca rutinas y funciones en lenguaje absoluto de FORTRAN. El problema se puede resolver en tres pasos: en el PAS01 el procedimiento ASMFCL compila y vincula el programa en ASSEMBLER, cargando el módulo resultante, con el nombre FURMONCD, en una biblioteca temporaria particionada que se crea en ese momento y reside en un disco auxiliar; en el PAS02 el procedimiento LKED vincula las rutinas y funciones FORTRAN, cargando el módulo resultante, con el nombre MONOFOR, en la misma biblioteca temporaria creada en el paso anterior; en el PAS03 el procedimiento FTG1LG vincula los módulos FURMONCD y MONOFOR que obtiene de la biblioteca temporaria y ejecuta.

```
//..... JOB ....
//PAS01 EXEC PROC=ASMFCL
//ASM.SYSIN DD *
  Aquí tarjetas perforadas en ASSEMBLER
//LKED.SYSLMOD DD DSN=TEMPOR.ADR(FURMONCD),DISP=(NEW,PASS),
//  UNIT=WORKDISK,SPACE=(CYL,(1,1,1)),
//  DCB=(BLKSIZE=800,RECFM=FB,LRECL=80)
/*
//PAS02 EXEC PROC=LKED
//LKED.SYSLMOD DD DSN=TEMPOR.ADR(MONOFOR),DISP=OLD,
//  VOL=REF=*.PAS01.LKED.SYSLMOD
//LKED.SYSIN DD *,DCB=BLKSIZE=80
```



```

Aquí tarjetas perforadas en absoluto de FORTRAN
/*
//PAS03 EXEC PROC=FTG1LG
//LKED.OTRALIB DD DSN=TEMPOR.ADR,DISP=(OLD,DELETE)
//LKED.SYSIN DD *,DCB=BLKSIZE=80
        INCLUDE OTRALIB(FURMONCD)
        INCLUDE OTRALIB(MONOFOR)
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
Aquí tarjetas perforadas de DATOS
/*
//

```

NOTA: En el PAS01 se menciona el DCB con el cual se grabará el disco auxiliar. En el programa principal ASSEMBLER se ha abierto un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

Modelo 022. Ejecución de un programa en lenguaje absoluto de ASSEMBLER, que invoca rutinas y funciones en lenguaje simbólico ASSEMBLER. El problema se puede resolver en dos pasos: en el PAS01 el procedimiento ASMFCL compila y vincula las rutinas y funciones en ASSEMBLER, cargando el módulo resultante, con el nombre MADEREF, en una biblioteca temporaria particionada que se crea en ese momento y reside en un disco auxiliar; en el PAS02 el procedimiento LKEDG vincula el programa en absoluto de ASSEMBLER con el módulo MADEREF que obtiene de la biblioteca temporaria y ejecuta.

```

//..... JOB ....
//PAS01 EXEC PROC=ASMFCL
//ASM.SYSIN DD *
Aquí tarjetas perforadas en ASSEMBLER
//LKED.SYSLMOD DD DSN=TEMPOR.ADR(MADEREF),DISP=(NEW,PASS),
// UNIT=WORKDISK,SPACE=(CYL,(1,1,1)),
// DCB=(BLKSIZE=1600,RECFM=FB,LRECL=80)
/*
//PAS02 EXEC PROC=LKEDG
//LKED.OTRALIB DD DSN=TEMPOR.ADR,DISP=(OLD,DELETE)
//LKED.SYSIN DD *,DCB=BLKSIZE=80
Aquí tarjetas perforadas en absoluto de ASSEMBLER
        INCLUDE OTRALIB(MADEREF)
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
Aquí tarjetas perforadas de DATOS
/*
//

```

NOTA: En el PASO1 se menciona el DCB con el cual se grabará el disco auxiliar. En el programa principal ASSEMBLER que dió origen al lote de tarjetas en absoluto, se abrió un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

Modelo 023. Ejecución de un programa en lenguaje absoluto de ASSEMBLER que invoca rutinas y funciones en lenguaje simbólico FORTRAN. El problema se puede resolver en tres pasos: en el PASO1 el procedimiento LKED vincula el programa en absoluto de ASSEMBLER, cargando el módulo resultante, con el nombre PRIMO, en una biblioteca temporaria particionada que se crea en ese momento y reside en un disco auxiliar; en el PASO2 el procedimiento FTG1CL compila y vincula las rutinas y funciones en FORTRAN, cargando el módulo resultante, con el nombre NOTEM, en la misma biblioteca temporaria creada en el paso anterior; en el PASO3 el procedimiento FTG1LG vincula los módulos PRIMO y NOTEM, que obtiene de la biblioteca temporaria y ejecuta.

```
//..... JOB ....
//PASO1 EXEC PROC=LKED
//LKED.SYSLMOD DD DSN=TEMPOR.ADR(PRIMO),DISP=(NEW,PASS),
// UNIT=WORKDISK,SPACE=(CYL,(1,1,1)),
// DCB=(BLKSIZE=800,RECFM=FB,LRECL=80)
//LKED.SYSIN DD *,DCB=BLKSIZE=80
Aquí tarjetas perforadas en absoluto de ASSEMBLER
/*
//PASO2 EXEC PROC=FTG1CL
//FORT.SYSIN DD *
Aquí tarjetas perforadas en FORTRAN
//LKED.SYSLMOD DD DSN=TEMPOR.ADR(NOTEM),DISP=OLD,
// VOL=REF=*.PASO1.LKED.SYSLMOD
/*
//PASO3 EXEC PROC=FTG1LG
//LKED.OTRALIB DD DSN=TEMPOR.ADR,DISP=(OLD,DELETE)
//LKED.SYSIN DD *,DCB=BLKSIZE=80
INCLUDE OTRALIB(PRIMO)
INCLUDE OTRALIB(NOTEM)
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: En el PASO1 se menciona el DCB con el cual se grabará el disco auxiliar. En

el programa principal ASSEMBLER que dió origen al lote de tarjetas en absoluto, se abrió un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

Modelo 024 . Ejecución de un programa en lenguaje absoluto de ASSEMBLER, que invoca rutinas y funciones en lenguaje absoluto de FORTRAN.

```
//..... JOB ....
//PAS01 EXEC PROC=FTG1LG
//LKED.SYSIN DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas en absoluto de ASSEMBLER
  Aquí tarjetas perforadas en absoluto de FORTRAN
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: En el programa principal ASSEMBLER que dió origen al lote de tarjetas en absoluto, se abrió un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

Modelo 025 . Ejecución de un programa en lenguaje simbólico ASSEMBLER, que invoca rutinas y funciones en lenguaje simbólico FORTRAN y ASSEMBLER. El problema se puede resolver en cuatro pasos: en el PAS01 el procedimiento ASMFCL compila y vincula el programa en ASSEMBLER, cargando el módulo resultante, con el nombre PARSEC, en una biblioteca temporaria particionada que se crea en ese momento y reside en un disco auxiliar; en el PAS02 el procedimiento FTG1CL compila y vincula las rutinas y funciones en FORTRAN, cargando el módulo resultante, con el nombre STAR1, en la misma biblioteca temporaria; en el PAS03 el procedimiento ASMFCL compila y vincula las rutinas y funciones en ASSEMBLER, cargando el módulo resultante, con el nombre STAR2, en la biblioteca temporaria; en el PAS04 el procedimiento FTG1LG vincula los módulos PARSEC, STAR1 y STAR2 y ejecuta.

```
//..... JOB ....
//PAS01 EXEC PROC=ASMFCL
//ASM.SYSIN DD *
  Aquí tarjetas perforadas del programa en ASSEMBLER
```

```

//LKED.SYSLMOD DD DSN=TEMPOR.ADR(PARSEC),SPACE=(CYL,(1,1,1)),
// UNIT=WORKDISK,DISP=(,PASS),DCB=(BLKSIZE=2400,LRECL=80,RECFM=FB)
/*
//PASO2 EXEC PROC=FTG1CL
//FORT.SYSIN DD *
  Aquí tarjetas perforadas en FORTRAN
//LKED.SYSLMOD DD DSN=TEMPOR.ADR(STAR1),DISP=OLD,
// VOL=REF=*.PASO1.LKED.SYSLMOD
/*
//PASO3 EXEC PROC=ASMFCL
//ASM.SYSIN DD *
  Aquí tarjetas perforadas de las rutinas y funciones en ASSEMBLER
//LKED.SYSLMOD DD DSN=TEMPOR.ADR(STAR2),DISP=OLD,
// VOL=REF=*.PASO1.LKED.SYSLMOD
/*
//PASO4 EXEC PROC=FTG1LG
//LKED.OTRALIB DD DSN=TEMPOR.ADR,DISP=(OLD,DELETE)
//LKED.SYSIN DD *,DCB=BLKSIZE=80
      INCLUDE OTRALIB(PARSEC)
      INCLUDE OTRALIB(STAR1)
      INCLUDE OTRALIB(STAR2)
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas de DATOS
/*
//

```

NOTA: En el PASO1 se menciona el DCB con el cual se grabará el disco auxiliar. En el programa principal ASSEMBLER se ha abierto un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

PUNTO 6.

Ejecución de programas que invocan rutinas y funciones codificadas en otro lenguaje, que se presentan grabadas en cintas magnéticas.

Modelo 026. Ejecución de un programa en lenguaje simbólico FORTRAN, que invoca rutinas y funciones en lenguaje absoluto de FORTRAN o de ASSEMBLER, grabadas con el nombre JUNCAL en el rótulo 3 de una cinta magnética, que lleva el número de carrete 21823 y tiene etiquetas modelo IBM.

```
//..... JOB ....
//PAS01 EXEC PROC=FTG1CLG
//FORT.SYSIN DD *
Aquí tarjetas perforadas en FORTRAN
/*
//LKED.SYSIN DD DSN=JUNCAL,UNIT=TAPE,VOL=SER=T21823,LABEL=(3,SL),
// DISP=(OLD,KEEP)
/*
//GO.SYSIN DD *
Aquí tarjetas perforadas de DATOS
/*
//
```

Modelo 027. Ejecución de un programa en lenguaje simbólico FORTRAN, que invoca rutinas y funciones en lenguaje simbólico ASSEMBLER, grabadas con el nombre BATEC en el rótulo 10 de una cinta magnética que lleva el número de carrete 01697 y tiene etiquetas modelo IBM. El problema se puede resolver en dos pasos: en el PAS01 el procedimiento ASMFC compila las rutinas y funciones en ASSEMBLER, cargando el módulo resultante en un disco auxiliar; en el PAS02 el procedimiento FTG1CLG compila el programa en FORTRAN, lo vincula con el módulo BATEC que obtiene del disco auxiliar y ejecuta.

```
//..... JOB ....
//PAS01 EXEC PROC=ASMFC,PARM.ASM=DECK
//ASM.SYSPUNCH DD DSN=BATEC,DISP=(,PASS),UNIT=WORKDISK,
// SPACE=(CYL,(1,1)),DCB=BLKSIZE=80
//ASM.SYSIN DD DSN=BATEC,UNIT=TAPE,VOL=SER=T01697,LABEL=(10,SL),
// DISP=(OLD,KEEP)
/*
//PAS02 EXEC PROC=FTG1CLG
//FORT.SYSIN DD *
Aquí tarjetas perforadas en FORTRAN
/*
//LKED.SYSIN DD DSN=BATEC,UNIT=WORKDISK,DISP=(OLD,DELETE)
```

```

/*
//GO.SYSIN DD *
Aquí tarjetas perforadas de DATOS
/*
//

```

Modelo 028 . Ejecución de un programa en lenguaje absoluto de FORTRAN, que invoca rutinas y funciones en lenguaje simbólico FORTRAN, grabadas con el nombre STATIC en el rótulo 11 de una cinta magnética, que lleva el número de carrete 26281 y tiene etiquetas modelo IBM.

```

//..... JOB ....
//PAS01 EXEC PROC=FTG1CLG
//FORT.SYSIN DD DSN=STATIC,UNIT=TAPE,VOL=SER=T26281,LABEL=(11,SL),
// DISP=(OLD,KEEP)
/*
//LKED.SYSIN DD *,DCB=BLKSIZE=80
Aquí tarjetas perforadas en absoluto de FORTRAN
/*
//GO.SYSIN DD *
Aquí tarjetas perforadas de DATOS
/*
//

```

Modelo 029 . Ejecución de un programa en lenguaje absoluto de FORTRAN, que invoca rutinas y funciones en lenguaje simbólico ASSEMBLER, grabadas con el nombre TRIAXIAL en el rótulo 18 de una cinta magnética que lleva el número de carrete 55713 y tiene etiquetas modelo IBM. El problema se puede resolver en dos pasos: en el PAS01 el procedimiento ASMFC compila las rutinas y funciones en ASSEMBLER, cargándolas en un disco auxiliar; en el PAS02 el procedimiento FTG1LG vincula el programa en absoluto de FORTRAN con el módulo TRIAXIAL y ejecuta.

```

//..... JOB ....
//PAS01 EXEC PROC=ASMFC,PARM.ASM=DECK
//ASM.SYSPUNCH DD DSN=TRIAxIAL,DISP=(,PASS),UNIT=WORKDISK,
// SPACE=(CYL,(1,1)),DCB=BLKSIZE=80
//ASM.SYSIN DD DSN=TRIAxIAL,UNIT=TAPE,VOL=SER=T55713,LABEL=(18,SL),
// DISP=(OLD,KEEP)
/*
//PAS02 EXEC PROC=FTG1LG
//LKED.SYSIN DD DSN=TRIAxIAL,UNIT=WORKDISK,DISP=(OLD,DELETE)
// DD *,DCB=BLKSIZE=80
Aquí tarjetas perforadas en absoluto de FORTRAN
/*
. //GO.SYSIN DD *

```

```

Aquí tarjetas perforadas de DATOS
/*
//

```

Modelo 030 . Ejecución de un programa en lenguaje absoluto de FORTRAN, que invoca rutinas y funciones en lenguaje absoluto de FORTRAN o de ASSEMBLER, grabadas con el nombre REFIL en el rótulo 29 de una cinta magnética, que lleva el número de carrete 97184 y tiene etiquetas modelo IBM.

```

//..... JOB ....
//PAS01 EXEC PROC=FTGILG
//LKED.SYSIN DD DSN=REFIL,UNIT=TAPE,VOL=SER=T97184,LABEL=(29,SL),
// DISP=(OLD,KEEP)
//          DD *,DCB=BLKSIZE=80
Aquí tarjetas perforadas en absoluto de FORTRAN o de ASSEMBLER
/*
//GO.SYSIN DD *
Aquí tarjetas perforadas de DATOS
/*
//

```

Modelo 031 . Ejecución de un programa en lenguaje simbólico ASSEMBLER, que invoca rutinas y funciones en lenguaje absoluto de ASSEMBLER, grabadas con el nombre FACTOR25 en el rótulo 21 de una cinta magnética, que lleva el número de carrete 22566 y tiene etiquetas modelo IBM.

```

//..... JOB ....
//PAS01 EXEC PROC=ASMFCLG
//ASM.SYSIN DD *
Aquí tarjetas perforadas en ASSEMBLER
/*
//LKED.SYSIN DD DSN=FACTOR25,UNIT=TAPE,VOL=SER=T22566,LABEL=(21,SL),
// DISP=(OLD,KEEP)
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
Aquí tarjetas perforadas de DATOS
/*
//

```

NOTA: En el programa principal ASSEMBLER se ha abierto un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

Modelo 032 . Ejecución de un programa en lenguaje simbólico ASSEMBLER, que invoca rutinas y funciones en lenguaje simbólico FORTRAN, grabadas con el nombre

ASUNTO8, en el rótulo 14 de una cinta magnética que lleva el número de carrete 16288 y tiene etiquetas modelo IBM . El problema se puede resolver en tres pasos: en el PASO1 el procedimiento ASMFCL compila y vincula el programa en ASSEMBLER, cargando el módulo resultante, con el nombre ASUNTOS, en una biblioteca temporaria particionada que se crea en ese momento y reside en un disco auxiliar; en el PASO2 el procedimiento FTG1CL compila y vincula las rutinas y funciones en FORTRAN, cargando el módulo resultante en la misma biblioteca temporaria creada en el paso anterior; en el PASO3 el procedimiento FTG1LG vincula los módulos ASUNTOS y ASUNTO8, que obtiene de la biblioteca temporaria y ejecuta.

```
//..... JOB ....
//PASO1 EXEC PROC=ASMFCL
//ASM.SYSIN DD *
  Aquí tarjetas perforadas en ASSEMBLER
//LKED.SYSLMOD DD DSN=TEMPOR.ADR(ASUNTOS),DISP=(NEW,PASS),
//  UNIT=WORKDISK,SPACE=(CYL,(1,1,1)),
//  DCB=(BLKSIZE=2400,LRECL=80,RECFM=FB)
/*
//PASO2 EXEC PROC=FTG1CL
//FORT.SYSIN DD DSN=ASUNTO8,UNIT=TAPE,VOL=SER=T16288,
//  LABEL=(14,SL),DISP=(OLD,KEEP)
//LKED.SYSLMOD DD DSN=TEMPOR.ADR(ASUNTO8),DISP=OLD,
//  VOL=REF=*.PASO1.LKED.SYSLMOD
/*
//PASO3 EXEC PROC=FTG1LG
//LKED.OTRALIB DD DSN=TEMPOR.ADR,DISP=(OLD,DELETE)
//LKED.SYSIN DD *,DCB=BLKSIZE=80
      INCLUDE OTRALIB(ASUNTOS)
      INCLUDE OTRALIB(ASUNTO8)
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80,
  Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: En el PASO1 se menciona el DCB con el cual se grabará el disco auxiliar. En el programa principal ASSEMBLER se ha abierto un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

Modelo 033 . Ejecución de un programa en lenguaje simbólico ASSEMBLER, que invoca rutinas y funciones en lenguaje absoluto de FORTRAN, grabadas con el nombre FASTO en el rótulo 17 de una cinta magnética, que lleva el número de carrete

38021 y tiene etiquetas modelo IBM. El problema se puede resolver en tres pasos: en el PASO1 el procedimiento ASMFCL compila y vincula el programa en ASSEMBLER, cargando el módulo resultante, con el nombre FASTOPG, en una biblioteca temporaria particionada que se crea en ese momento y reside en un disco auxiliar; en el PASO2 el procedimiento LKED vincula las rutinas y funciones en absoluto de FORTRAN, cargando el módulo resultante en la misma biblioteca temporaria creada en el paso anterior; en el PASO3 el procedimiento FTG1LG vincula los módulos FASTOPG y FASTO, que obtiene de la biblioteca temporaria y ejecuta.

```
//..... JOB ....
//PASO1 EXEC PROC=ASMFCL
//ASM.SYSIN DD *
  Aquí tarjetas perforadas en ASSEMBLER
//LKED.SYSLMOD DD DSN=TEMPOR.ADR(FASTOPG),DISP=(NEW,PASS),
//  UNIT=WORKDISK,SPACE=(CYL,(1,1,1)),
//  DCB=(BLKSIZE=1600,LRECL=80,RECFM=FB)
/*
//PASO2 EXEC PROC=LKED
//LKED.SYSLMOD DD DSN=TEMPOR.ADR(FASTO),DISP=OLD,
//  VOL=REF=*.PASO1.LKED.SYSLMOD
//LKED.SYSIN DD DSN=FASTO,UNIT=TAPE,VOL=SER=T38021,LABEL=(17,SL),
//  DISP=(OLD,KEEP)
/*
//PASO3 EXEC PROC=FTG1LG
//LKED.OTRALIB DD DSN=TEMPOR.ADR,DISP=(OLD,DELETE)
//LKED.SYSIN DD *,DCB=BLKSIZE=80
      INCLUDE OTRALIB(FASTOPG)
      INCLUDE OTRALIB(FASTO)
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas de DATOS
/*
//
```

NCTA: En el PASO1 se menciona el DCB con el cual se grabará el disco auxiliar. En el programa principal ASSEMBLER se ha abierto un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

Modelo 034 . Ejecución de un programa en lenguaje absoluto de ASSEMBLER, que invoca rutinas y funciones en lenguaje simbólico ASSEMBLER, grabadas con el nombre DYNAMIC en el rótulo 21 de una cinta magnética, que lleva el número de carrete 32195 y tiene etiquetas modelo IBM. El problema se puede resolver en dos pasos:

en el PAS01 el procedimiento ASMFCL compila y vincula las rutinas y funciones en ASSEMBLER, cargando el módulo resultante en una biblioteca temporaria particionada que se crea en ese momento y reside en un disco auxiliar; en el PAS02 el procedimiento LKEDG vincula el programa en absoluto de ASSEMBLER con el módulo DYNAMIC, que obtiene de la biblioteca temporaria y ejecuta.

```
//..... JOB ....
//PAS01 EXEC PROC=ASMFCL
//ASM.SYSIN DD DSN=DYNAMIC,UNIT=TAPE,VOL=SER=T32195,LABEL=(21,SL),
// DISP=(OLD,KEEP)
//LKED.SYSLMOD DD DSN=TEMPOR.ADR(DYNAMIC),DISP=(NEW,PASS),
// UNIT=WORKDISK,SPACE=(CYL,(1,1,1)),
// DCB=(BLKSIZE=2400,LRECL=80,RECFM=FB)
/*
//PAS02 EXEC PROC=LKEDG
//LKED.OTRALIB DD DSN=TEMPOR.ADR,DISP=(OLD,DELETE)
//LKED.SYSIN DD *,DCB=BLKSIZE=80
Aquí tarjetas perforadas en absoluto de ASSEMBLER
      INCLUDE OTRALIB(DYNAMIC)
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: En el PAS01 se menciona el DCB con el cual se grabará el disco auxiliar. En el programa principal ASSEMBLER que dió origen al lote de tarjetas en absoluto, se abrió un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

Modelo 035 . Ejecución de un programa en lenguaje absoluto de ASSEMBLER, que invoca rutinas y funciones en lenguaje simbólico FORTRAN, grabadas con el nombre CERNRUT en el rótulo 5 de una cinta magnética, que lleva el número de carrete 63304 y tiene etiquetas modelo IBM. El problema se puede resolver en tres pasos: en el PAS01 el procedimiento LKED vincula el programa en absoluto de ASSEMBLER, cargando el módulo resultante, con el nombre PGM CERN, en una biblioteca temporaria particionada que se crea en ese momento y reside en un disco auxiliar; en el PAS02 el procedimiento FTG1CL compila y vincula las rutinas y funciones en FORTRAN, cargando el módulo resultante en la misma biblioteca temporaria creada

en el paso anterior; en el PASO3 el procedimiento FTG1LG vincula los módulos PGM CERN y CERNRUT, que obtiene de la biblioteca temporaria y ejecuta.

```
//..... JOB ....
//PASO1 EXEC PROC=LKED
//LKED.SYSLMOD DD DSN=TEMPOR.ADR(PGM CERN),DISP=(NEW,PASS),
// UNIT=WORKDISK,SPACE=(CYL,(1,1,1)),
// DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//LKED.SYSIN DD *,DCB=BLKSIZE=80
Aquí tarjetas perforadas en absoluto de ASSEMBLER
/*
//PASO2 EXEC PROC=FTG1CL
//FORT.SYSIN DD DSN=CERNRUT,UNIT=TAPE,VOL=SER=T63304,LABEL=(5,SL),
// DISP=(OLD,KEEP)
//LKED.SYSLMOD DD DSN=TEMPOR.ADR(CERNRUT),DISP=OLD,
// VOL=REF=*.PASO1.LKED.SYSLMOD
/*
//PASO3 EXEC PROC=FTG1LG
//LKED.OTRALIB DD DSN=TEMPOR.ADR,DISP=(OLD,DELETE)
//LKED.SYSIN DD *,DCB=BLKSIZE=80
INCLUDE OTRALIB(PGM CERN)
INCLUDE OTRALIB(CERNRUT)
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: En el PASO1 se menciona el DCB con el cual se grabará el disco auxiliar. En el programa principal ASSEMBLER que dió origen al lote de tarjetas en absoluto, se abrió un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

Modelo 036 .Ejecución de un programa en lenguaje absoluto de ASSEMBLER, que invoca rutinas y funciones en lenguaje absoluto de ASSEMBLER, grabadas con el nombre SUBFINAL en el rótulo 24 de una cinta magnética, que lleva el número de carrete 72327 y tiene etiquetas modelo IBM. El problema se puede resolver en dos pasos: en el PASO1 el procedimiento LKED vincula las rutinas y funciones en absoluto de ASSEMBLER, cargando el módulo resultante en una biblioteca temporaria particionada que se crea en ese momento y reside en un disco auxiliar; en el PASO2 el procedimiento LKEDG vincula el programa en absoluto de ASSEMBLER con el módulo SUBFINAL, que obtiene de la biblioteca temporaria creada en el paso anterior y e-

jecuta.

```
//..... JOB ....
//PAS01 EXEC PROC=LKED
//LKED.SYSLMOD DD DSN=TEMPOR.ADR(SUBFINAL),DISP=(NEW,PASS),
// UNIT=WORKDISK,SPACE=(CYL,(1,1,1)),
// DCB=(BLKSIZE=800,LRECL=80,RECFM=FB)
//LKED.SYSIN DD DSN=SUBFINAL,UNIT=TAPE,VOL=SER=T72327,LABEL=(24,SL),
// DISP=(OLD,KEEP)
/*
//PAS02 EXEC PROC=LKEDG
//LKED.OTRALIB DD DSN=TEMPOR.ADR,DISP=(OLD,DELETE)
//LKED.SYSIN DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas en absoluto de ASSEMBLER
    INCLUDE OTRALIB(SUBFINAL)
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: En el PAS01 se menciona el DCB con el cual se grabará el disco auxiliar. En el programa principal ASSEMBLER que dió origen al lote de tarjetas en absoluto, se abrió un archivo de entrada de datos de nombre TARJETA y un archivo de salida de nombre IMPRESO.

Modelo 037. Ejecución de un programa en lenguaje absoluto de ASSEMBLER, que invoca rutinas y funciones en lenguaje absoluto de FORTRAN, grabadas con el nombre OPTICAL1 en el rótulo 16 de una cinta magnética, que lleva el número de carrete 10001 y tiene etiquetas modelo IBM. El problema se puede resolver en tres pasos: en el PAS01 el procedimiento LKED vincula el programa en absoluto de ASSEMBLER, cargando el módulo resultante, con el nombre OPTIC en una biblioteca temporaria particionada que se crea en ese momento y reside en un disco auxiliar; en el PAS02 el mismo procedimiento vincula las rutinas y funciones en absoluto de FORTRAN, cargando el módulo resultante en la misma biblioteca temporaria creada en el paso anterior; en el PAS03 el procedimiento FTGILG vincula los módulos OPTIC y OPTICAL1, que obtiene de la biblioteca temporaria y ejecuta.

```
//..... JOB ....
//PAS01 EXEC PROC=LKED
```

```
//LKED.SYSLMOD DD DSN=TEMPOR.ADR(OPTIC),DISP=(NEW,PASS),
// UNIT=WORKDISK,SPACE=(CYL,(1,1,1)),
// DCB=(BLKSIZE=1600,LRECL=80,RECFM=FB)
//LKED.SYSIN DD *,DCB=BLKSIZE=80
Aquí tarjetas perforadas en absoluto de ASSEMBLER
/*
//PASO2 EXEC PROC=LKED
//LKED.SYSLMOD DD DSN=TEMPOR.ADR(OPTICAL1),DISP=OLD,
// VOL=REF=*.PASO1.LKED.SYSLMOD
//LKED.SYSIN DD DSN=OPTICAL1,UNIT=TAPE,VOL=SER=T10001,LABEL=(16,SL),
// DISP=(OLD,KEEP)
/*
//PASO3 EXEC PROC=FTG1LG
//LKED.OTRALIB DD DSN=TEMPOR.ADR,DISP=(OLD,DELETE)
//LKED.SYSIN DD *,DCB=BLKSIZE=80
INCLUDE OTRALIB(OPTIC)
INCLUDE OTRALIB(OPTICAL1)
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: En el PASO1 se menciona el DCB con el cual se grabará el disco auxiliar. En el programa principal ASSEMBLER que dió origen al lote de tarjetas en absoluto, se abrió un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

Modelo 038 . Ejecución de un programa en lenguaje simbólico ASSEMBLER, que invoca rutinas y funciones en lenguaje simbólico FORTRAN y ASSEMBLER, que están grabadas con los nombres DERIVADA e INTEGRAL, en los rótulos 18 y 25, respectivamente, de una cinta magnética que lleva el número de carrete 80316 y tiene etiquetas modelo IBM. El problema se puede resolver en cuatro pasos: en el PASO1 el procedimiento ASMFCL compila y vincula el programa en ASSEMBLER, cargando el módulo resultante, con el nombre ECUACION, en una biblioteca temporaria particionada que se crea en ese momento y reside en un disco auxiliar; en el PASO2 el procedimiento FTG1CL compila y vincula las rutinas y funciones en FORTRAN, cargando el módulo resultante en la misma biblioteca temporaria; en el PASO3 el procedimiento ASMFCL compila y vincula las rutinas y funciones en ASSEMBLER, cargando el módulo resultante en la biblioteca temporaria; en el PASO4 el procedimiento FTG1LG vin-

cula los módulos ECUACION, DERIVADA e INTEGRAL y ejecuta.

```
//..... JOB ....
//PASO1 EXEC PROC=ASMFCL
//ASM.SYSIN DD *
  Aquí tarjetas perforadas del programa en ASSEMBLER
//LKED.SYSLMOD DD DSN=TEMPOR.ADR(ECUACION),SPACE=(CYL,(1,1,1),
//  UNIT=WORKDISK,DISP=(,PASS),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
/*
//PASO2 EXEC PROC=FTG1CL
//FORT.SYSIN DD DSN=DERIVADA,UNIT=TAPE,VOL=SER=T80316,LABEL=(18,SL),
//  DISP=(OLD,PASS)
//LKED.SYSLMOD DD DSN=TEMPOR.ADR(DERIVADA),DISP=OLD,
//  VOL=REF=*.PASO1.LKED.SYSLMOD
/*
//PASO3 EXEC PROC=ASMFCL
//ASM.SYSIN DD DSN=INTEGRAL,UNIT=TAPE,VOL=SER=T80316,LABEL=(25,SL),
//  DISP=(OLD,KEEP)
//LKED.SYSLMOD DD DSN=TEMPOR.ADR(INTEGRAL),DISP=OLD,
//  VOL=REF=*.PASO1.LKED.SYSLMOD
/*
//PASO4 EXEC PROC=FTG1LG
//LKED.OTRALIB DD DSN=TEMPOR.ADR,DISP=(OLD,DELETE)
//LKED.SYSIN DD *,DCB=BLKSIZE=80
      INCLUDE OTRALIB(ECUACION)
      INCLUDE OTRALIB(DERIVADA)
      INCLUDE OTRALIB(INTEGRAL)
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: En el PASO1 se menciona el DCB con el cual se grabará el disco auxiliar. En el programa principal ASSEMBLER se ha abierto un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

Modelo 039 . Ejecución de un programa en lenguaje absoluto de ASSEMBLER, que invoca rutinas y funciones en lenguaje simbólico FORTRAN y ASSEMBLER, que están grabadas, con los nombres RUTINA1 y RUTINA2, en los rótulos 11 y 14 respectivamente, de una cinta magnética que lleva el número de carrete 35525 y tiene etiquetas modelo IBM. El problema se puede resolver en cuatro pasos: en el PASO1 el procedimiento LKED vincula el programa en absoluto de ASSEMBLER, cargando el módulo resultante, con el nombre ASSMAIN, en una biblioteca temporaria particionada que se crea en ese momento y reside en un disco auxiliar; en el PASO2 el procedimiento

FTG1CL compila y vincula las rutinas y funciones en FORTRAN, cargando el módulo resultante en la misma biblioteca temporaria; en el PASO3 el procedimiento ASMFCL compila y vincula las rutinas y funciones en ASSEMBLER, cargando el módulo resultante en la biblioteca temporaria; en el PASO4 el procedimiento FTG1LG vincula los módulos ASSMAIN, RUTINA1 y RUTINA2 y ejecuta.

```
//..... JOB ....
//PASO1 EXEC PROC=LKED
//LKED.SYSLMOD DD DSN=TEMPOR.ADR(ASSMAIN),SPACE=(CYL,(1,1,1)),
// UNIT=WORKDISK,DISP=(,PASS),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//LKED.SYSIN DD *,DCB=BLKSIZE=80
Aquí tarjetas perforadas en absoluto de ASSEMBLER
/*
//PASO2 EXEC PROC=FTG1CL
//FORT.SYSIN DD DSN=RUTINA1,UNIT=TAPE,VOL=SER=T35525,LABEL=(11,SL),
// DISP=(OLD,PASS)
//LKED.SYSLMOD DD DSN=TEMPOR.ADR(RUTINA1),DISP=OLD,
// VOL=REF=*.PASO1.LKED.SYSLMOD
/*
//PASO3 EXEC PROC=ASMFCL
//ASM.SYSIN DD DSN=RUTINA2,UNIT=TAPE,VOL=SER=T35525,LABEL=(14,SL),
// DISP=(OLD,KEEP)
//LKED.SYSLMOD DD DSN=TEMPOR.ADR(RUTINA2),DISP=OLD,
// VOL=REF=*.PASO1.LKED.SYSLMOD
/*
//PASO4 EXEC PROC=FTG1LG
//LKED.OTRALIB DD TEMPOR.ADR,DISP=(OLD,DELETE)
//LKED.SYSIN DD *,DCB=BLKSIZE=80
        INCLUDE OTRALIB(ASSMAIN)
        INCLUDE OTRALIB(RUTINA1)
        INCLUDE OTRALIB(RUTINA2)
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: En el PASO1 se menciona el DCB con el cual se grabará el disco auxiliar. En el programa principal ASSEMBLER que dió origen al lote de tarjetas en absoluto, se abrió un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

PUNTO 7.

Uso de la Referencia hacia Atrás al programa vinculado en un paso anterior de la misma tarea, para ejecutar dos o mas juegos de datos distintos, en los pasos siguientes.

Modelo 040 . Ejecución de un programa en lenguaje simbólico FORTRAN, compilado y vinculado en el PASO1 durante el cual resuelve el juego de DATOS1; en el PASO2 se hace referencia al mismo programa para resolver el juego de DATOS2.

```
//..... JOB ....
//PASO1 EXEC PROC=FTG1CLG
//FORT.SYSIN DD *
  Aquí tarjetas perforadas en FORTRAN
/*
//GO.SYSIN DD *
  Aquí tarjetas perforadas de DATOS1
/*
//PASO2 EXEC PGM=*.PASO1.LKED.SYSLMOD
//FT06F001 DD SYSOUT=A
//FT05F001 DD *
  Aquí tarjetas perforadas de DATOS2
/*
//
```

Modelo 041 . Ejecución de un programa en lenguaje absoluto de FORTRAN, vinculado en el PASO1 durante el cual resuelve el juego de DATOS1; en los PASOS 2 y 3 se hace referencia al mismo programa para resolver los juegos de DATOS2 y DATOS3.

```
//..... JOB ....
//PASO1 EXEC PROC=FTG1LG
//LKED.SYSIN DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas en absoluto de FORTRAN
/*
//GO.SYSIN DD *
  Aquí tarjetas perforadas de DATOS1
/*
//PASO2 EXEC PGM=*.PASO1.LKED.SYSLMOD
//FT06F001 DD SYSOUT=A
//FT05F001 DD *
  Aquí tarjetas perforadas de DATOS2
/*
//PASO3 EXEC PGM=*.PASO1.LKED.SYSLMOD
//FT06F001 DD SYSOUT=A
//FT05F001 DD *
  Aquí tarjetas perforadas de DATOS3
/*
//
```


Modelo 042 . Ejecución de un programa en lenguaje simbólico ASSEMBLER, compilado y vinculado en el PAS01 durante el cual resuelve el juego de DATOS1; en los PASOS 2 y 3 se hace referencia al mismo programa para resolver los juegos de DATOS2 y DATOS3.

```
//..... JOB ....
//PAS01 EXEC PROC=ASMFCLG
//ASM.SYSIN DD *
  Aquí tarjetas perforadas en ASSEMBLER
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas de DATOS1
/*
//PAS02 EXEC PGM=*.PAS01.LKED.SYSLMOD
//IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//TARJETA DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas de DATOS2
/*
//PAS03 EXEC PGM=*.PAS01.LKED.SYSLMOD
//IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//TARJETA DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas de DATOS3
/*
//
```

NOTA: En el programa ASSEMBLER se ha abierto un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

Modelo 043 . Ejecución de un programa en lenguaje absoluto de ASSEMBLER, vinculado en el PAS01 durante el cual resuelve el juego de DATOS1; en el PAS02 se hace referencia al mismo programa para resolver el juego de DATOS2.

```
//..... JOB ....
//PAS01 EXEC PROC=LKEDG
//LKED.SYSIN DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas en absoluto de ASSEMBLER
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas de DATOS1
/*
//PAS02 EXEC PGM=*.PAS01.LKED.SYSLMOD
//IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//TARJETA DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas de DATOS2
/*
//
```

NOTA: En el programa ASSEMBLER que dió origen al lote de tarjetas en absoluto, se abrió un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

Modelo 044. Ejecución de un programa en lenguaje simbólico PL/1, compilado y vinculado en el PASO1 durante el cual resuelve el juego de DATOS1; en el PASO2 se hace referencia al mismo programa para resolver el juego de DATOS2.

```
//..... JOB ....
//PASO1 EXEC PROC=PLIXCLG,PARM.PLI='CS(60)'
//PLI.SYSIN DD *
Aquí tarjetas perforadas en PL/1
/*
//GO.SYSIN DD *
Aquí tarjetas perforadas de DATOS1
/*
//PASO2 EXEC PGM=*.PASO1.LKED.SYSLMOD
//SYSPRINT DD SYSOUT=A
//SYSIN DD *
Aquí tarjetas perforadas de DATOS2
/*
//
```

Modelo 045. Ejecución de un programa en lenguaje absoluto de PL/1, vinculado en el PASO1 durante el cual resuelve el juego de DATOS1; en el PASO2 se hace referencia al mismo programa para resolver el juego de DATOS2.

```
//..... JOB ....
//PASO1 EXEC PROC=PLIXLG
//LKED.SYSIN DD *,DCB=BLKSIZE=80
Aquí tarjetas perforadas en absoluto de PL/1
/*
//GO.SYSIN DD *
Aquí tarjetas perforadas de DATOS1
/*
//PASO2 EXEC PGM=*.PASO1.LKED.SYSLMOD
//SYSPRINT DD SYSOUT=A
//SYSIN DD *
Aquí tarjetas perforadas de DATOS2
/*
//
```

PUNTO 8.

Uso de la Referencia hacia Atrás a un juego de datos ejecutado en un paso anterior de la misma tarea, para ser procesado por dos o mas programas distintos, en los pasos siguientes.

Modelo 046 . Ejecución de un programa en lenguaje simbólico FORTRAN, compilado y vinculado en el PASO1, en el cual resuelve el juego de DATOS1 y de otro programa en lenguaje simbólico FORTRAN, compilado y vinculado en el PASO2, en el cual se hace referencia al mismo juego de DATOS1.

```
//..... JOB ....
//PASO1 EXEC PROC=FTG1CLG
//FORT.SYSIN DD *
Aquí tarjetas perforadas del primer programa en FORTRAN
/*
//GO.SYSIN DD *
Aquí tarjetas perforadas de DATOS1
/*
//PASO2 EXEC PROC=FTG1CLG
//FORT.SYSIN DD *
Aquí tarjetas perforadas del segundo programa en FORTRAN
/*
//GO.SYSIN DD DSNAME=*.PASO1.GO.FT05F001,DISP=(OLD,PASS)
/*
//
```

Modelo 047 . Ejecución de un programa en lenguaje absoluto de FORTRAN, vinculado en el PASO1, en el cual resuelve el juego de DATOS1 y de otro programa en lenguaje absoluto de FORTRAN, vinculado en el PASO2, en el cual se hace referencia al mismo juego de DATOS1.

```
//..... JOB ....
//PASO1 EXEC PROC=FTG1LG
//LKED.SYSIN DD *,DCB=BLKSIZE=80
Aquí tarjetas perforadas del primer programa en absoluto de FORTRAN
/*
//GO.SYSIN DD *
Aquí tarjetas perforadas de DATOS1
/*
//PASO2 EXEC PROC=FTG1LG
//LKED.SYSIN DD *,DCB=BLKSIZE=80
Aquí tarjetas perforadas del segundo programa en absoluto de FORTRAN
/*
//GO.SYSIN DD DSNAME=*.PASO1.GO.FT05F001,DISP=(OLD,PASS)
/*
//
```

Modelo 048. Ejecución de un programa en lenguaje simbólico ASSEMBLER, compilado y vinculado en el PAS01, en el cual resuelve el juego de DATOS1 y de otro programa en lenguaje simbólico ASSEMBLER, compilado y vinculado en el PASO2, en el cual se hace referencia al mismo juego de DATOS1.

```
//..... JOB ....
//PAS01 EXEC PROC=ASMFCLG
//ASM.SYSIN DD *
  Aquí tarjetas perforadas del primer programa en ASSEMBLER
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas de DATOS1
/*
//PASO2 EXEC PROC=ASMFCLG
//ASM.SYSIN DD *
  Aquí tarjetas perforadas del segundo programa en ASSEMBLER
/*
//GO.IMPRIMIR DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.LEER DD DSNAME=*.PAS01.GO.TARJETA,DISP=(OLD,PASS)
/*
//
```

NOTA: En el programa ASSEMBLER usado en el PAS01, se ha abierto un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO. En el programa ASSEMBLER usado en el PASO2 se ha abierto un archivo de entrada de nombre LEER y un archivo de salida de nombre IMPRIMIR.

Modelo 049. Ejecución de un programa en lenguaje absoluto de ASSEMBLER, vinculado en el PAS01, en el cual resuelve el juego de DATOS1 y de otro programa en lenguaje absoluto de ASSEMBLER, vinculado en el PASO2, en el cual se hace referencia al mismo juego de DATOS1.

```
//..... JOB ....
//PAS01 EXEC PROC=LKEDG
//LKED.SYSIN DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas del primer programa en absoluto de ASSEMBLER
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas de DATOS1
/*
//PASO2 EXEC PROC=LKEDG
//LKED.SYSIN DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas del segundo programa en absoluto de ASSEMBLER
```

```

/*
//GO.IMPRIMIR DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.LEER DD DSN=*.PAS01.GC.TARJETA,DISP=(OLD,PASS)
/*
//

```

NOTA: En el programa ASSEMBLER que dió origen al lote de tarjetas en absoluto usado en el PAS01, se abrió un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO. En el programa ASSEMBLER que dió origen al lote de tarjetas en absoluto usado en el PAS02, se abrió un archivo de entrada de datos de nombre LEER y un archivo de salida de datos de nombre IMPRIMIR.

Modelo 050 . Ejecución de un programa en lenguaje simbólico PL/1, compilado y vinculado en el PAS01, en el cual resuelve el juego de DATOS1 y de otro programa en lenguaje simbólico PL/1, compilado y vinculado en el PAS02, en el cual se hace referencia al mismo juego de DATOS1.

```

//..... JOB ....
//PAS01 EXEC PROC=PLIXCLG,PARM.PLI='CS(60)'
//PLI.SYSIN DD *
Aquí tarjetas perforadas del primer programa en PL/1
/*
//GO.SYSIN DD *
Aquí tarjetas perforadas de DATOS1
/*
//PAS02 EXEC PROC=PLIXCLG,PARM.PLI='CS(60)'
//PLI.SYSIN DD *
Aquí tarjetas perforadas del segundo programa en PL/1
/*
//GO.SYSIN DD DSN=*.PAS01.GC.SYSIN,DISP=(OLD,PASS)
/*
//

```

Modelo 051 . Ejecución de un programa en lenguaje absoluto de PL/1, vinculado en el PAS01, en el cual resuelve el juego de DATOS1 y de otro programa en lenguaje absoluto de PL/1, vinculado en el PAS02, en el cual se hace referencia al mismo juego de DATOS1.

```

//..... JOB ....
//PAS01 EXEC PROC=PLIXLG
//LKED.SYSIN DD *,DCB=BLKSIZE=80
Aquí tarjetas perforadas del primer programa en absoluto de PL/1
/*

```

```
//GO.SYSIN DD *  
  Aquí tarjetas perforadas de DATOS1  
/*  
//PASO2 EXEC PROC=PLIXLG  
//LKED.SYSIN DD *,DCB=BLKSIZE=80  
  Aquí tarjetas perforadas del segundo programa en absoluto de PL/1  
/*  
//GO.SYSIN DD DSN=*.PASO1.GO.SYSIN,DISP=(OLD,PASS)  
/*  
//
```


CAPITULO II

MODELOS PARA EL USO DE UNA BIBLIOTECA PARTICIONADA
Y CATALOGADA DE PROGRAMAS EN LENGUAJE SIMBOLICO
QUE RESIDE EN UNA UNIDAD DE DISCOS

PUNTO 1.

Obtención de listados y lotes de tarjetas perforadas de uno o mas miembros de una biblioteca de nombre SIMBOLIC.CNEA.

Modelo 052 . Listado de la totalidad de la biblioteca.

```
//..... JOB ....
//PAS01 EXEC PGM=IEBPTPCH
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA,DISP=SHR
//SYSUT2 DD SYSOUT=A
//SYSIN DD *
PRINT TYPORG=PO
/*
//
```

Modelo 053 . Listado de un miembro de la biblioteca y obtención del lote de tarjetas perforadas de otro miembro. El programa consta de dos pasos: en el PAS01 el utilitario IEBPTPCH lista el miembro de nombre DEIMOS y en el PAS02, el mismo utilitario obtiene el lote del miembro FOBOS.

```
//..... JOB ....
//PAS01 EXEC PGM=IEBPTPCH
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA,DISP=SHR
//SYSUT2 DD SYSOUT=A
//SYSIN DD *
PRINT TYPORG=PO,MAXNAME=1,MAXFLDS=1
MEMBER NAME=DEIMOS
RECORD FIELD=(80)
/*
//PAS02 EXEC PGM=IEBPTPCH
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA,DISP=SHR
//SYSUT2 DD SYSOUT=B
//SYSIN DD *
PUNCH TYPORG=PO,MAXNAME=1,MAXFLDS=1
MEMBER NAME=FOBOS
RECORD FIELD=(80)
/*
//
```

Modelo 054 . Listado de cinco miembros de la biblioteca y obtención de los lotes de tarjetas perforadas de otros tres miembros. El programa consta de dos pasos:

en el PAS01 el utilitario IEBPTPCH lista los miembros de nombres NINE, HARPA, MAIN3, COUP y TRANSFE; en el PAS02, el mismo utilitario obtiene los lotes de los miembros de nombres FERMION, REGISTRO y VEA.

```
//..... JOB ....
//PAS01 EXEC PGM=IEBPTPCH
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA,DISP=SHR
//SYSUT2 DD SYSOUT=A
//SYSIN DD *
PRINT TYPORG=PO,MAXNAME=5,MAXFLDS=5
MEMBER NAME=NINE
RECORD FIELD=(80)
MEMBER NAME=HARPA
RECORD FIELD=(80)
MEMBER NAME=MAIN3
RECORD FIELD=(80)
MEMBER NAME=COUP
RECORD FIELD=(80)
MEMBER NAME=TRANSFE
RECORD FIELD=(80)
/*
//PAS02 EXEC PGM=IEBPTPCH
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA,DISP=SHR
//SYSUT2 DD SYSOUT=B
//SYSIN DD *
PUNCH TYPORG=PO,MAXNAME=3,MAXFLDS=3
MEMBER NAME=FERMION
RECORD FIELD=(80)
MEMBER NAME=REGISTRO
RECORD FIELD=(80)
MEMBER NAME=VEA
RECORD FIELD=(80)
/*
//
```

PUNTO 2.

Ejecución de programas de una biblioteca de nombre SIMBOLIC.CNEA.

Modelo 055 . Ejecución de un programa en FORTRAN, grabado en la biblioteca con el nombre CUATNIV.

```
//..... JOB ....
//PAS01 EXEC PROC=FTG1CLG
//FORT.SYSIN DD DSN=SIMBOLIC.CNEA(CUATNIV),DISP=SHR
/*
//GO.SYSIN DD *
  Aquí tarjetas perforadas de DATOS
/*
//
```

Modelo 056 . Ejecución de un programa en ASSEMBLER, grabado en la biblioteca con el nombre ASEMAT.

```
//..... JOB ....
//PAS01 EXEC PROC=ASMFCLG
//ASM.SYSIN DD DSN=SIMBOLIC.CNEA(ASEMAT),DISP=SHR
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: En el programa se ha abierto un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

Modelo 057 . Ejecución de un programa en PL/1, grabado en la biblioteca con el nombre PULNIT.

```
//..... JOB ....
//PAS01 EXEC PROC=PLIXCLG,PARM.PLI='CS(60) '
//PLI.SYSIN DD DSN=SIMBOLIC.CNEA(PULNIT),DISP=SHR
/*
//GO.SYSIN DD *
  Aquí tarjetas perforadas de DATOS
/*
//
```

PUNTO 3.

Ejecución de programas de una biblioteca de nombre SIMBOLIC.CNEA, que invocan rutinas y funciones residentes en la propia biblioteca, como miembros independientes.

Modelo 058. Ejecución de un programa en FORTRAN, grabado en la biblioteca con el nombre CALCULO3, que invoca rutinas y funciones en FORTRAN, residentes en la propia biblioteca como miembros independientes, con los nombres ALFA, SIXJ y GAMMA.

```
//..... JOB ....
//PAS01 EXEC PROC=FTG1CLG
//FORT.SYSIN DD DSN=SIMBOLIC.CNEA(CALCULO3),DISP=SHR
//          DD DSN=SIMBOLIC.CNEA(ALFA),DISP=SHR
//          DD DSN=SIMBOLIC.CNEA(SIXJ),DISP=SHR
//          DD DSN=SIMBOLIC.CNEA(GAMMA),DISP=SHR
/*
//GO.SYSIN DD *
Aquí tarjetas perforadas de DATOS
/*
//
```

Modelo 059. Ejecución de un programa en FORTRAN, grabado en la biblioteca con el nombre SALIN, que invoca rutinas y funciones en ASSEMBLER, residentes en la propia biblioteca como un miembro independiente, con el nombre ASMDR. El problema se puede resolver en dos pasos: en el PAS01 el procedimiento ASMFC compila las rutinas y funciones en ASSEMBLER, cargando el módulo resultante en un disco auxiliar; en el PAS02 el procedimiento FTG1CLG compila el programa SALIN, lo vincula con el módulo ASMDR, que obtiene del disco auxiliar y ejecuta.

```
//..... JOB ....
//PAS01 EXEC PROC=ASMFC,PARM.ASM=DECK
//ASM.SYSPUNCH DD DSN=ASMDR,DISP=(,PASS),UNIT=WORKDISK,
// SPACE=(CYL,(1,1)),DCB=BLKSIZE=80
//ASM.SYSIN DD DSN=SIMBOLIC.CNEA(ASMDR),DISP=SHR
/*
//PAS02 EXEC PROC=FTG1CLG
//FORT.SYSIN DD DSN=SIMBOLIC.CNEA(SALIN),DISP=SHR
/*
//LKED.SYSIN DD DSN=ASMDR,UNIT=WORKDISK,DISP=(OLD,DELETE)
/*
//GO.SYSIN DD *
Aquí tarjetas perforadas de DATOS
/*
//
```

Modelo 060 . Ejecución de un programa en ASSEMBLER, grabado en la biblioteca con el nombre COTASUP, que invoca rutinas y funciones en ASSEMBLER, residentes en la propia biblioteca como un miembro independiente, con el nombre Tsuma. El problema se puede resolver en dos pasos: en el PASO1 el procedimiento ASMFCL compila y vincula las rutinas y funciones en ASSEMBLER, cargando el módulo resultante en una biblioteca temporaria particionada que se crea en ese momento y reside en un disco auxiliar; en el PASO2 el procedimiento ASMFCLG compila el programa COTASUP y lo vincula con el módulo Tsuma, que obtiene de la biblioteca temporaria y ejecuta.

```
//..... JOB ....
//PASO1 EXEC PROC=ASMFCL
//ASM.SYSIN DD DSN=SIMBOLIC.CNEA(Tsuma),DISP=SHR
//LKED.SYSMOD DD DSN=TEMPOR.ADR(Tsuma),DISP=(,PASS),
// UNIT=WORKDISK,SPACE=(CYL,(1,1,1)),
// DCB=(BLKSIZE=800,RECFM=FB,LRECL=80)
/*
//PASO2 EXEC PROC=ASMFCLG
//ASM.SYSIN DD DSN=SIMBOLIC.CNEA(COTASUP),DISP=SHR
/*
//LKED.OTRALIB DD DSN=TEMPOR.ADR,DISP=(OLD,DELETE)
//LKED.SYSIN DD *,DCB=BLKSIZE=80
        INCLUDE OTRALIB(Tsuma)
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: En el PASO1 se menciona el DCB con el cual se graba el disco auxiliar. En el programa principal ASSEMBLER se ha abierto un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

Modelo 061 . Ejecución de un programa en ASSEMBLER, grabado en la biblioteca con el nombre VINTOB, que invoca rutinas y funciones en FORTRAN, residentes en la propia biblioteca como miembros independientes, con los nombres PLAN01 y PLAN02. El problema se puede resolver en tres pasos: en el PASO1 el procedimiento ASMFCL compila y vincula el programa VINTOB, cargando el módulo resultante en una biblioteca temporaria particionada que se crea en ese momento y reside en un disco

auxiliar; en el PASO2 el procedimiento FTG1CL compila y vincula las rutinas y funciones en FORTRAN cargando el módulo resultante, con el nombre PLANTOT, en la misma biblioteca temporaria creada en el paso anterior; en el PASO3 el procedimiento FTG1LG vincula los módulos VINTOB y PLANTOT, que obtiene de la biblioteca temporaria y ejecuta.

```
//..... JOB ....
//PASO1 EXEC PROC=ASMFCL
//ASM.SYSIN DD DSN=SIMBOLIC.CNEA(VINTOB),DISP=SHR
//LKED.SYSLMOD DD DSN=TEMPOR.ADR(VINTOB),DISP=(,PASS),
//  UNIT=WORKDISK,SPACE=(CYL,(1,1,1)),
//  DCB=(BLKSIZE=1600,RECFM=FB,LRECL=80)
/*
//PASO2 EXEC PROC=FTG1CL
//FORT.SYSIN DD DSN=SIMBOLIC.CNEA(PLANO1),DISP=SHR
//          DD DSN=SIMBOLIC.CNEA(PLANO2),DISP=SHR
//LKED.SYSLMOD DD DSN=TEMPOR.ADR(PLANTOT),DISP=OLD,
//  VOL=REF=*.PASO1.LKED.SYSLMOD
/*
//PASO3 EXEC PROC=FTG1LG
//LKED.OTRALIB DD DSN=TEMPOR.ADR,DISP=(OLD,DELETE)
//LKED.SYSIN DD *,DCB=BLKSIZE=80
//          INCLUDE OTRALIB(VINTOB)
//          INCLUDE OTRALIB(PLANTOT)
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
//  Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: En el PASO1 se menciona el DCB con el cual se grabará el disco auxiliar. En el programa principal ASSEMBLER se ha abierto un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

PUNTO 4.

Ejecución de programas de una biblioteca de nombre SIMBOLIC.CNEA, que invocan rutinas y funciones que se presentan en tarjetas perforadas.

Modelo 062 . Ejecución de un programa en FORTRAN, grabado en la biblioteca con el nombre TOTALIT, que invoca rutinas y funciones que se presentan en tarjetas perforadas en lenguaje simbólico FORTRAN.

```
//..... JOB ....
//PAS01 EXEC PROC=FTG1CLG
//FORT.SYSIN DD DSN=SIMBOLIC.CNEA(TOTALIT),DISP=SHR
//          DD *
Aquí tarjetas perforadas en FORTRAN
/*
//GO.SYSIN DD *
Aquí tarjetas perforadas de DATOS
/*
//
```

Modelo 063 . Ejecución de un programa en FORTRAN, grabado en la biblioteca con el nombre SULTAN, que invoca rutinas y funciones que se presentan en tarjetas perforadas en lenguaje absoluto de FORTRAN o de ASSEMBLER.

```
//..... JOB ....
//PAS01 EXEC PROC=FTG1CLG
//FORT.SYSIN DD DSN=SIMBOLIC.CNEA(SULTAN),DISP=SHR
/*
//LKED.SYSIN DD *,DCB=BLKSIZE=80
Aquí tarjetas perforadas en absoluto de FORTRAN o de ASSEMBLER
/*
//GO.SYSIN DD *
Aquí tarjetas perforadas de DATOS
/*
//
```

Modelo 064 . Ejecución de un programa en FORTRAN, grabado en la biblioteca con el nombre ALCORIN, que invoca rutinas y funciones que se presentan en tarjetas perforadas en lenguaje simbólico ASSEMBLER. El problema se puede resolver en dos pasos: en el PAS01 el procedimiento ASMFC compila las rutinas y funciones en ASSEMBLER, cargando el módulo resultante en un disco auxiliar con el nombre SAMEN; en el PAS02 el procedimiento FTG1CLG compila el programa ALCORIN, lo vincula con el módulo SAMEN, que obtiene del disco auxiliar y ejecuta.

```
//..... JOB ....
//PASO1 EXEC PROC=ASMFC,PARM.ASM=DECK
//ASM.SYSPUNCH DD DSN=SAMEN,DISP=(,PASS),UNIT=WORKDISK,
// SPACE=(CYL,(1,1)),DCB=BLKSIZE=80
//ASM.SYSIN DD *
Aquí tarjetas perforadas en ASSEMBLER
/*
//PASO2 EXEC PROC=FTG1CLG
//FORT.SYSIN DD DSN=SIMBOLIC.CNEA(ALCORIN),DISP=SHR
/*
//LKED.SYSIN DD DSN=SAMEN,UNIT=WORKDISK,DISP=(OLD,DELETE)
/*
//GO.SYSIN DD *
Aquí tarjetas perforadas de DATOS
/*
//
```

Modelo 065 . Ejecución de un programa en ASSEMBLER, grabado en la biblioteca con el nombre VALORES, que invoca rutinas y funciones que se presentan en tarjetas perforadas en lenguaje simbólico ASSEMBLER. El problema se puede resolver en dos pasos: en el PASO1 el procedimiento ASMFC compila las rutinas y funciones en ASSEMBLER, cargando el módulo resultante en un disco auxiliar con el nombre SOLEM; en el PASO2 el procedimiento ASMFCLG compila el programa VALORES, lo vincula con el módulo SOLEM que obtiene del disco auxiliar y ejecuta.

```
//..... JOB ....
//PASO1 EXEC PROC=ASMFC,PARM.ASM=DECK
//ASM.SYSPUNCH DD DSN=SOLEM,DISP=(,PASS),UNIT=WORKDISK,
// SPACE=(CYL,(1,1)),DCB=BLKSIZE=80
//ASM.SYSIN DD *
Aquí tarjetas perforadas en ASSEMBLER
/*
//PASO2 EXEC PROC=ASMFCLG
//ASM.SYSIN DD DSN=SIMBOLIC.CNEA(VALORES),DISP=SHR
//LKED.SYSIN DD DSN=SOLEM,UNIT=WORKDISK,DISP=(OLD,DELETE)
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCE=BLKSIZE=80
Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: En el programa principal ASSEMBLER se ha abierto un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

Modelo 066 . Ejecución de un programa en ASSEMBLER, grabado en la biblioteca con el nombre RESUMEN, que invoca rutinas y funciones que se presentan en tarjetas

perforadas en lenguaje absoluto de ASSEMBLER.

```
//..... JOB ....
//PAS01 EXEC PROC=ASMFCLG
//ASM.SYSIN DD DSN=SIMBOLIC.CNEA(RESUMEN),DISP=SHR
/*
//LKED.SYSIN DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas en absoluto de ASSEMBLER
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: En el programa principal ASSEMBLER se ha abierto un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

Modelo 067 . Ejecución de un programa en ASSEMBLER, grabado en la biblioteca con el nombre ASUNTOS2, que invoca rutinas y funciones que se presentan en tarjetas perforadas en lenguaje simbólico FORTRAN. El problema se puede resolver en tres pasos: en el PAS01 el procedimiento ASMFCL compila y vincula el programa ASUNTOS2, cargando el módulo resultante en una biblioteca temporaria particionada que se crea en ese momento y reside en un disco auxiliar; en el PAS02 el procedimiento FTG1CL compila y vincula las rutinas y funciones en FORTRAN, cargando el módulo resultante, con el nombre ASUNTOS3, en la misma biblioteca temporaria creada en el paso anterior; en el PAS03 el procedimiento FTG1LG vincula los módulos ASUNTOS2 y ASUNTOS3 que obtiene de la biblioteca temporaria y ejecuta.

```
//..... JOB ....
//PAS01 EXEC PROC=ASMFCL
//ASM.SYSIN DD DSN=SIMBOLIC.CNEA(ASUNTOS2),DISP=SHR
//LKED.SYSLMOD DD DSN=TEMPOR.ADR(ASUNTOS2),DISP=(,PASS),
// UNIT=WORKDISK,SPACE=(CYL,(1,1,1)),
// DCB=(BLKSIZE=1600,RECFM=FB,LRECL=80)
/*
//PAS02 EXEC PROC=FTG1CL
//FORT.SYSIN DD *
  Aquí tarjetas perforadas en FORTRAN
//LKED.SYSLMOD DD DSN=TEMPOR.ADR(ASUNTOS3),DISP=OLD,
// VOL=REF=*.PAS01.LKED.SYSLMOD
/*
//PAS03 EXEC PROC=FTG1LG
//LKED.OTRALIB DD DSN=TEMPOR.ADR,DISP=(OLD,DELETE)
//LKED.SYSIN DD *,DCB=BLKSIZE=80
```

```

INCLUDE OTRALIB(ASUNTOS2)
INCLUDE OTRALIB(ASUNTOS3)

```

```

/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas de DATOS
/*
//

```

NOTA: En el PASO1 se menciona el DCB con el cual se grabará el disco auxiliar. En el programa principal ASSEMBLER se ha abierto un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

Modelo 068 . Ejecución de un programa en ASSEMBLER, grabado en la biblioteca con el nombre TEMA12, que invoca rutinas y funciones que se presentan en tarjetas perforadas en lenguaje absoluto de FORTRAN. El problema se puede resolver en tres pasos: en el PASO1 el procedimiento ASMFCL compila y vincula el programa TEMA12, cargando el módulo resultante en una biblioteca temporaria particionada que se crea en ese momento y reside en un disco auxiliar; en el PASO2 el procedimiento LKED vincula rutinas y funciones en absoluto de FORTRAN, cargando el módulo resultante, con el nombre TEMA16, en la misma biblioteca temporaria creada en el paso anterior; en el PASO3 el procedimiento FTG1LG vincula los módulos TEMA12 y TEMA16, que obtiene de la biblioteca temporaria y ejecuta.

```

//..... JOB ....
//PASO1 EXEC PROC=ASMFCL
//ASM.SYSIN DD DSN=SIMBOLIC.CNEA(TEMA12),DISP=SHR
//LKED.SYSLMOD DD DSN=TEMPOR.ADR(TEMA12),DISP=(NEW,PASS),
//  UNIT=WORKDISK,SPACE=(CYL,(1,1,1)),
//  DCB=(BLKSIZE=3200,RECFM=FB,LRECL=80)
/*
//PASO2 EXEC PROC=LKED
//LKED.SYSLMOD DD DSN=TEMPOR.ADR(TEMA16),DISP=OLD,
//  VOL=REF=*.PASO1.LKED.SYSLMOD
//LKED.SYSIN DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas en absoluto de FORTRAN
/*
//PASO3 EXEC PROC=FTG1LG
//LKED.OTRALIB DD DSN=TEMPOR.ADR,DISP=(OLD,DELETE)
//LKED.SYSIN DD *,DCB=BLKSIZE=80
      INCLUDE OTRALIB(TEMA12)
      INCLUDE OTRALIB(TEMA16)
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)

```

```
//GO.TARJETA DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: Se menciona el DCB con el cual se grabará el disco auxiliar. En el programa principal ASSEMBLER se ha abierto un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

Modelo 069. Ejecución de un programa en ASSEMBLER, grabado en la biblioteca con el nombre LOCAL, que invoca rutinas y funciones que se presentan en tarjetas perforadas en lenguaje simbólico FORTRAN y ASSEMBLER. El problema se puede resolver en cuatro pasos: en el PASO1 el procedimiento ASMFCL compila y vincula el programa LOCAL, cargando el módulo resultante en una biblioteca temporaria particionada que se crea en ese momento y reside en un disco auxiliar; en el PASO2 el procedimiento FTG1CL, compila y vincula las rutinas en FORTRAN, cargando el módulo resultante, con el nombre RUTFORT, en la misma biblioteca temporaria; en el PASO3 el procedimiento ASMFCL compila y vincula las rutinas y funciones en ASSEMBLER, cargando el módulo resultante, con el nombre RUTASSEM, en la biblioteca temporaria; en el PASO4 el procedimiento FTG1LG vincula los módulos LOCAL, RUTFORT y RUTASSEM y ejecuta.

```
//..... JOB ....
//PASO1 EXEC PROC=ASMFCL
//ASM.SYSIN DD DSN=SIMBOLIC.CNEA(LOCAL),DISP=SHR
//LKED.SYSLMOD DD DSN=TEMPOR.ADR(LOCAL),SPACE=(CYL,(1,1,1)),
// UNIT=WORKDISK,DISP=(,PASS),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
/*
//PASO2 EXEC PROC=FTG1CL
//FORT.SYSIN DD *
  Aquí tarjetas perforadas en FORTRAN
//LKED.SYSLMOD DD DSN=TEMPOR.ADR(RUTFORT),DISP=OLD,
// VOL=REF=*.PASO1.LKED.SYSLMOD
/*
//PASO3 EXEC PROC=ASMFCL
//ASM.SYSIN DD *
  Aquí tarjetas perforadas en ASSEMBLER
//LKED.SYSLMOD DD DSN=TEMPOR.ADR(RUTASSEM),DISP=OLD,
// VOL=REF=*.PASO1.LKED.SYSLMOD
/*
//PASO4 EXEC PROC=FTG1LG
//LKED.OTRALIB DD DSN=TEMPOR.ADR,DISP=(OLD,DELETE)
```

```
//LKED.SYSIN DD *,DCB=BLKSIZE=80
      INCLUDE OTRALIB(LOCAL)
      INCLUDE OTRALIB(RUTFORT)
      INCLUDE OTRALIB(RUTASSEM)
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
      Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: En el PAS01 se menciona el DCB con el cual se grabará el disco auxiliar. En el programa principal ASSEMBLER se ha abierto un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

PUNTO 5.

Ejecución de programas de una biblioteca de nombre SIMBOLIC.CNEA, que invocan rutinas y funciones grabadas en cintas magnéticas.

Modelo 070 . Ejecución de un programa en FORTRAN, residente en la biblioteca con el nombre LLAMAR, que invoca rutinas y funciones en lenguaje simbólico FORTRAN, grabadas con el nombre TYRIL, en el rótulo 19 de una cinta magnética que lleva el número de carrete 35680 y tiene etiquetas modelo IBM.

```
//..... JOB ....
//PAS01 EXEC PROC=FTG1CLG
//FORT.SYSIN DD DSN=SIMBOLIC.CNEA(LLAMAR),DISP=SHR
//          DD DSN=TYRIL,UNIT=TAPE,VOL=SER=T35680,LABEL=(19,SL),
//  DISP=(OLD,KEEP)
/*
//GO.SYSIN DD *
Aquí tarjetas perforadas de DATOS
/*
//
```

Modelo 071 . Ejecución de un programa en FORTRAN, residente en la biblioteca con el nombre ZURIR, que invoca rutinas y funciones en lenguaje absoluto de FORTRAN o de ASSEMBLER, grabadas con el nombre ZUTAP, en el rótulo 12 de una cinta magnética que lleva el número de carrete 16791 y tiene etiquetas modelo IBM.

```
//..... JOB ....
//PAS01 EXEC PROC=FTG1CLG
//FORT.SYSIN DD DSN=SIMBOLIC.CNEA(ZURIR),DISP=SHR
/*
//LKED.SYSIN DD DSN=ZUTAP,UNIT=TAPE,VOL=SER=T16791,LABEL=(12,SL),
//  DISP=(OLD,KEEP)
/*
//GO.SYSIN DD *
Aquí tarjetas perforadas de DATOS
/*
//
```

Modelo 072 . Ejecución de un programa en FORTRAN, residente en la biblioteca con el nombre ASISTEN, que invoca rutinas y funciones en lenguaje simbólico ASSEMBLER, grabadas con el nombre PUNTUAL, en el rótulo 8 de una cinta magnética que lleva el número de carrete 71294 y tiene etiquetas modelo IBM. El problema se puede resolver en dos pasos: en el PAS01 el procedimiento ASMFC compila las rutinas y

funciones en ASSEMBLER, cargando el módulo resultante en un disco auxiliar; en el PASO2 el procedimiento FTG1CLG compila el programa ASISTEN, lo vincula con el módulo PUNTUAL que obtiene del disco auxiliar y ejecuta.

```
//..... JOB ....
//PASO1 EXEC PROC=ASMFC,PARM.ASM=DECK
//ASM.SYSPUNCH DD DSN=PUNTUAL,DISP=(,PASS),UNIT=WORKDISK,
//  SPACE=(CYL,(1,1)),DCB=BLKSIZE=80
//ASM.SYSIN DD DSN=PUNTUAL,UNIT=TAPE,VOL=SER=T71294,LABEL=(8,SL),
//  DISP=(OLD,KEEP)
/*
//PASO2 EXEC PROC=FTG1CLG
//FORT.SYSIN DD DSN=SIMBOLIC.CNEA(ASISTEN),DISP=SHR
/*
//LKED.SYSIN DD DSN=PUNTUAL,UNIT=WORKDISK,DISP=(OLD,DELETE)
/*
//GO.SYSIN DD *
Aquí tarjetas perforadas de DATOS
/*
//
```

Modelo 073 . Ejecución de un programa en ASSEMBLER, residente en la biblioteca con el nombre EIGENVAL, que invoca rutinas y funciones en lenguaje simbólico ASSEMBLER, grabadas con el nombre ASSEMDOS, en el rótulo 21 de una cinta magnética que lleva el número de carrete 13138 y tiene etiquetas modelo IBM. El problema se puede resolver en dos pasos: en el PASO1 el procedimiento ASMFC compila las rutinas y funciones en ASSEMBLER, cargando el módulo resultante en un disco auxiliar; en el PASO2 el procedimiento ASMFCLG compila el programa EIGENVAL, lo vincula con el módulo ASSEMDOS, que obtiene del disco auxiliar y ejecuta.

```
//..... JOB ....
//PASO1 EXEC PROC=ASMFC,PARM.ASM=DECK
//ASM.SYSPUNCH DD DSN=ASSEMDOS,DISP=(,PASS),UNIT=WORKDISK,
//  SPACE=(CYL,(1,1)),DCB=BLKSIZE=80
//ASM.SYSIN DD DSN=ASSEMDOS,UNIT=TAPE,VOL=SER=T13138,LABEL=(21,SL),
//  DISP=(OLD,KEEP)
/*
//PASO2 EXEC PROC=ASMFCLG
//ASM.SYSIN DD DSN=SIMBOLIC.CNEA(EIGENVAL),DISP=SHR
//LKED.SYSIN DD DSN=ASSEMDOS,UNIT=WORKDISK,DISP=(OLD,DELETE)
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: En el programa principal ASSEMBLER se ha abierto un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

Modelo 074 . Ejecución de un programa en ASSEMBLER, residente en la biblioteca con el nombre RACE, que invoca rutinas y funciones en lenguaje absoluto de ASSEMBLER, grabadas con el nombre INVERSA, en el rótulo 12 de una cinta magnética que lleva el número de carrete 56752 y tiene etiquetas modelo IBM.

```
//..... JOB ....
//PAS01 EXEC PROC=ASMFCLG
//ASM.SYSIN DD DSN=SIMBOLIC.CNEA(RACE),DISP=SHR
/*
//LKED.SYSIN DD DSN=INVERSA,UNIT=TAPE,VOL=SER=T56752,LABEL=(12,SL),
// DISP=(OLD,KEEP)
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: En el programa principal ASSEMBLER se ha abierto un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

Modelo 075 . Ejecución de un programa en ASSEMBLER, residente en la biblioteca con el nombre ONDACHOQ, que invoca rutinas y funciones en lenguaje simbólico FORTRAN, grabadas con el nombre SOLIDO, en el rótulo 41 de una cinta magnética que lleva el número de carrete 21235 y tiene etiquetas modelo IBM. El problema se puede resolver en tres pasos: en el PAS01 el procedimiento ASMFCL compila y vincula el programa ONDACHOQ, cargando el módulo resultante en una biblioteca temporaria particionada que se crea en ese momento y reside en un disco auxiliar; en el PAS02 el procedimiento FTG1CL compila y vincula las rutinas y funciones en FORTRAN, cargando el módulo resultante en la misma biblioteca temporaria creada en el paso anterior; en el PAS03 el procedimiento FTG1LG vincula los módulos ONDACHOQ y SOLIDO, que obtiene de la biblioteca temporaria y ejecuta.

```
//..... JOB ....
//PAS01 EXEC PROC=ASMFCL
//ASM.SYSIN DD DSN=SIMBOLIC.CNEA(ONDACHOQ),DISP=SHR
```

```
//LKED.SYSLMOD DD DSN=TEMPOR.ADR(ONDACHOQ),DISP=(NEW,PASS),
// UNIT=WORKDISK,SPACE=(CYL,(1,1,1)),
// DCB=(BLKSIZE=3200,LRECL=80,RECFM=FB)
/*
//PASO2 EXEC PROC=FTG1CL
//FORT.SYSIN DD DSN=SOLIDO,UNIT=TAPE,VOL=SER=T21235,LABEL=(41,SL),
// DISP=(OLD,KEEP)
//LKED.SYSLMOD DD DSN=TEMPOR.ADR(SOLIDO),DISP=OLD,
// VOL=REF=*.PASO1.LKED.SYSLMOD
/*
//PASO3 EXEC PROC=FTG1LG
//LKED.OTRALIB DD DSN=TEMPOR.ADR,DISP=(OLD,DELETE)
//LKED.SYSIN DD *,DCB=BLKSIZE=80
        INCLUDE OTRALIB(ONDACHOQ)
        INCLUDE OTRALIB(SOLIDO)
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: En el PASO1 se menciona el DCB con el cual se grabará el disco auxiliar. En el programa principal ASSEMBLER se ha abierto un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

Modelo 076 .Ejecución de un programa en ASSEMBLER, residente en la biblioteca con el nombre ISOTOPOS, que invoca rutinas y funciones en lenguaje absoluto de FORTRAN, que están grabadas con el nombre DETECTOR en el rótulo 29 de una cinta magnética que lleva el número de carrete 70007 y tiene etiquetas modelo IBM. El problema se puede resolver en tres pasos: en el PASO1 el procedimiento ASMFCL compila y vincula el programa en ASSEMBLER, cargando el módulo resultante en una biblioteca temporaria particionada que se crea en ese momento y reside en un disco auxiliar; en el PASO2 el procedimiento LKED vincula las rutinas y funciones en absoluto de FORTRAN, cargando el módulo resultante en la misma biblioteca temporaria creada en el paso anterior; en el PASO3 el procedimiento FTG1LG vincula los módulos ISOTOPOS y DETECTOR, que obtiene de la biblioteca temporaria y ejecuta.

```
//..... JOB ....
//PASO1 EXEC PROC=ASMFCL
//ASM.SYSIN DD DSN=SIMBOLIC.CNEA(ISOTOPOS),DISP=SHR
//LKED.SYSLMOD DD DSN=TEMPOR.ADR(ISOTOPOS),DISP=(NEW,PASS),
// UNIT=WORKDISK,SPACE=(CYL,(1,1,1)),
```



```
// DCB=(BLKSIZE=800,LRECL=80,RECFM=FB)
/*
//PASO2 EXEC PROC=LKED
//LKED.SYSLMOD DD DSN=TEMPOR.ADR(DETECTOR),DISP=OLD,
// VOL=REF=*.PASO1.LKED.SYSLMOD
//LKED.SYSIN DD DSN=DETECTOR,UNIT=TAPE,VOL=SER=T70007,LABEL=(29,SL),
// DISP=(OLD,KEEP)
/*
//PASO3 EXEC PROC=FTG1LG
//LKED.OTRALIB DD DSN=TEMPOR.ADR,DISP=(OLD,DELETE)
//LKED.SYSIN DD *,DCB=BLKSIZE=80
        INCLUDE OTRALIB(ISOTOPOS)
        INCLUDE OTRALIB(DETECTOR)
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: En el PASO1 se menciona el DCB con el cual se grabará el disco auxiliar. En el programa principal ASSEMBLER se ha abierto un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

Modelo 077 . Ejecución de un programa en ASSEMBLER, residente en la biblioteca con el nombre LIE, que invoca rutinas y funciones en lenguaje simbólico FORTRAN y ASSEMBLER, grabadas con los nombres LIERUTF y LIERUTAS, en los rótulos 14 y 19 respectivamente, de una cinta magnética que lleva el número de carrete 42271 y tiene etiquetas modelo IBM. El problema se puede resolver en cuatro pasos: en el PASO1 el procedimiento ASMFCL compila y vincula el programa en ASSEMBLER, cargando el módulo resultante en una biblioteca temporaria particionada que se crea en ese momento y reside en un disco auxiliar; en el PASO2 el procedimiento FTG1CL compila y vincula las rutinas y funciones en FORTRAN, cargando el módulo resultante en la misma biblioteca temporaria; en el PASO3 el procedimiento ASMFCL compila y vincula las rutinas y funciones en ASSEMBLER, cargando el módulo resultante en la biblioteca temporaria; en el PASO4 el procedimiento FTG1LG vincula los módulos LIE, LIERUTF y LIERUTAS y ejecuta.

```
//..... JOB ....
//PASO1 EXEC PROC=ASMFCL
//ASM.SYSIN DD DSN=SIMBOLIC.CNEA(LIE),DISP=SHR
```

```

//LKED.SYSLMOD DD DSN=TEMPOR.ADR(LIE),SPACE=(CYL,(1,1,1)),
//  UNIT=WORKDISK,DISP=(,PASS),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
/*
//PASO2 EXEC PROC=FTG1CL
//FORT.SYSIN DD DSN=LIERUTF,UNIT=TAPE,VOL=SER=T42271,LABEL=(14,SL),
//  DISP=(OLD,PASS)
//LKED.SYSLMOD DD DSN=TEMPOR.ADR(LIERUTF),DISP=OLD,
//  VOL=REF=*.PASO1.LKED.SYSLMOD
/*
//PASO3 EXEC PROC=ASMFCL
//ASM.SYSIN DD DSN=LIERUTAS,UNIT=TAPE,VOL=SER=T42271,LABEL=(19,SL),
//  DISP=(OLD,KEEP)
//LKED.SYSLMOD DD DSN=TEMPOR.ADR(LIERUTAS),DISP=OLD,
//  VOL=REF=*.PASO1.LKED.SYSLMOD
/*
//PASO4 EXEC PROC=FTG1LG
//LKED.OTRALIB DD DSN=TEMPOR.ADR,DISP=(OLD,DELETE)
//LKED.SYSIN DD *,DCB=BLKSIZE=80
//      INCLUDE OTRALIB(LIE)
//      INCLUDE OTRALIB(LIERUTF)
//      INCLUDE OTRALIB(LIERUTAS)
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
//  Aquí tarjetas perforadas de DATOS
/*
//

```

NOTA: En el PASO1 se menciona el DCB con el cual se grabará el disco auxiliar. En el programa principal ASSEMBLER se ha abierto un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

PUNTO 6.

Uso de la Referencia hacia Atrás a un programa de una biblioteca de nombre SIMBOLIC.CNEA, compilado y vinculado en un paso anterior de la misma tarea, para ejecutar dos o mas juegos de datos distintos, en los pasos siguientes.

Modelo 078. Ejecución de un programa en FORTRAN, grabado en la biblioteca con el nombre CASIMIL, compilado y vinculado en el PASO1 durante el cual resuelve el juego de DATOS1; en el PASO2 se hace referencia al mismo programa para resolver el juego de DATOS2.

```
//..... JOB ....
//PASO1 EXEC PROC=FTG1CLG
//FORT.SYSIN DD DSN=SIMBOLIC.CNEA(CASIMIL),DISP=SHR
/*
//GO.SYSIN DD *
  Aquí tarjetas perforadas de DATOS1
/*
//PASO2 EXEC PGM=*.PASO1.LKED.SYSLMOD
//FT06F001 DD SYSOUT=A
//FT05F001 DD *
  Aquí tarjetas perforadas de DATOS2
/*
//
```

Modelo 079. Ejecución de un programa en ASSEMBLER, grabado en la biblioteca con el nombre NEFTAL, compilado y vinculado en el PASO1 durante el cual resuelve el juego de DATOS1; en los PASOS 2 y 3 se hace referencia al mismo programa para resolver los juegos de DATOS2 y DATOS3.

```
//..... JOB ....
//PASO1 EXEC PROC=ASMFC LG
//ASM.SYSIN DD DSN=SIMBOLIC.CNEA(NEFTAL),DISP=SHR
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas de DATOS1
/*
//PASO2 EXEC PGM=*.PASO1.LKED.SYSLMOD
//IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//TARJETA DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas de DATOS2
/*
//PASO3 EXEC PGM=*.PASO1.LKED.SYSLMOD
//IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//TARJETA DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas de DATOS3
```

```
/*
//
```

NOTA: En el programa ASSEMBLER se ha abierto un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

Modelo 080. Ejecución de un programa en PL/1, grabado en la biblioteca con el nombre ENERGIA, compilado y vinculado en el PASO1 durante el cual resuelve el juego de DATOS1; en el PASO2 se hace referencia al mismo programa para resolver el juego de DATOS2.

```
//..... JOB ....
//PASO1 EXEC PROC=PLIXCLG,PARM.PLI='CS(60)'
//PLI.SYSIN DD DSN=SIMBOLIC.CNEA(ENERGIA),DISP=SHR
/*
//GO.SYSIN DD *
Aquí tarjetas perforadas de DATOS1
/*
//PASO2 EXEC PGM=*.PASO1.LKED.SYSLMOD
//SYSPRINT DD SYSOUT=A
/*
//SYSIN DD *
Aquí tarjetas perforadas de DATOS2
/*
//
```

PUNTO 7.

Uso de la Referencia hacia Atrás a un juego de datos ejecutado en un paso anterior de la misma tarea, para ser procesado en los pasos siguientes, por dos o mas programas distintos, de una biblioteca de nombre SIMBOLIC.CNEA.

Modelo 081 . Ejecución de un programa en FORTRAN, grabado en la biblioteca con el nombre SIMILAR1, compilado y vinculado en el PASO1 en el cual resuelve el juego de DATOS1 y de otro programa en FORTRAN, grabado en la biblioteca con el nombre SIMILAR2, compilado y vinculado en el PASO2, en el cual se hace referencia al mismo juego de DATOS1.

```
//..... JOB ....
//PASO1 EXEC PROC=FTG1CLG
//FORT.SYSIN DD DSN=SIMBOLIC.CNEA(SIMILAR1),DISP=SHR
/*
//GO.SYSIN DD *
  Aquí tarjetas perforadas de DATOS1
/*
//PASO2 EXEC PROC=FTG1CLG
//FORT.SYSIN DD DSN=SIMBOLIC.CNEA(SIMILAR2),DISP=SHR
/*
//GO.SYSIN DD DSNAME=*.PASO1.GO.FT05F001,DISP=(OLD,PASS)
/*
//
```

Modelo 082 . Ejecución de un programa en ASSEMBLER grabado en la biblioteca con el nombre LAM17A, compilado y vinculado en el PASO1 en el cual resuelve el juego de DATOS1 y de otro programa en ASSEMBLER, grabado en la biblioteca con el nombre LAM17B, compilado y vinculado en el PASO2, en el cual se hace referencia al mismo juego de DATOS1.

```
//..... JOB ....
//PASO1 EXEC PROC=ASMFCLG
//ASM.SYSIN DD DSN=SIMBOLIC.CNEA(LAM17A),DISP=SHR
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas de DATOS1
/*
//PASO2 EXEC PROC=ASMFCLG
//ASM.SYSIN DD DSN=SIMBOLIC.CNEA(LAM17B),DISP=SHR
/*
```

```
//GO.IMPRIMIR DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.LEER DD DSNAME=*.PAS01.GO.TARJETA,DISP=(OLD,PASS)
/*
//
```

NOTA: En el programa LAM17A se ha abierto un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO. Correlativamente, en el programa LAM17B se ha abierto un archivo de entrada de datos de nombre LEER y un archivo de salida de datos de nombre IMPRIMIR.

Modelo 083. Ejecución de un programa en PL/1, grabado en la biblioteca con el nombre RED1, compilado y vinculado en el PAS01 en el cual resuelve el juego de DATOS1 y de otro programa en PL/1, grabado en la biblioteca con el nombre RED2, compilado y vinculado en el PAS02, en el cual se hace referencia al mismo juego de DATOS1.

```
//..... JOB ....
//PAS01 EXEC PROC=PLIXCLG,PARM.PLI='CS(60)'
//PLI.SYSIN DD DSN=SIMBOLIC.CNEA(RED1),DISP=SHR
/*
//GO.SYSIN DD *
Aquí tarjetas perforadas de DATOS1
/*
//PAS02 EXEC PROC=PLIXCLG,PARM.PLI='CS(60)'
//PLI.SYSIN DD DSN=SIMBOLIC.CNEA(RED2),DISP=SHR
/*
//GO.SYSIN DD DSNAME=*.PAS01.GO.SYSIN,DISP=(OLD,PASS)
/*
//
```

PUNTO 8.

Ejecución de programas en tarjetas perforadas, que invocan rutinas y funciones residentes, como miembros independientes, en una biblioteca de nombre SIMBOLIC.CNEA.

Modelo 084 . Ejecución de un programa en tarjetas perforadas en lenguaje simbólico FORTRAN, que invoca rutinas y funciones en FORTRAN, residentes en la biblioteca como miembros independientes, con los nombres DIAG1, NINEJ y SUMA.

```
//..... JOB ....
//PAS01 EXEC PROC=FTG1CLG
//FORT.SYSIN DD DSN=SIMBOLIC.CNEA(DIAG1),DISP=SHR
//          DD DSN=SIMBOLIC.CNEA(NINEJ),DISP=SHR
//          DD DSN=SIMBOLIC.CNEA(SUMA),DISP=SHR
//          DD *
  Aquí tarjetas perforadas en FORTRAN
/*
//GO.SYSIN DD *
  Aquí tarjetas perforadas de DATOS
/*
//
```

Modelo 085 . Ejecución de un programa en tarjetas perforadas en lenguaje absoluto de FORTRAN, que invoca rutinas y funciones en FORTRAN, residentes en la biblioteca como miembros independientes, con los nombres POPULAT, ROTATION, BANDS y PIL.

```
//..... JOB ....
//PAS01 EXEC PROC=FTG1CLG
//FORT.SYSIN DD DSN=SIMBOLIC.CNEA(POPULAT),DISP=SHR
//          DD DSN=SIMBOLIC.CNEA(ROTATION),DISP=SHR
//          DD DSN=SIMBOLIC.CNEA(BANDS),DISP=SHR
//          DD DSN=SIMBOLIC.CNEA(PIL),DISP=SHR
/*
//LKED.SYSIN DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas en absoluto de FORTRAN
/*
//GO.SYSIN DD *
  Aquí tarjetas perforadas de DATOS
/*
//
```

Modelo 086 . Ejecución de un programa en tarjetas perforadas en lenguaje simbólico FORTRAN, que invoca rutinas y funciones en ASSEMBLER, residentes en la biblioteca, como un miembro independiente, con el nombre SIMPAT. El problema se

puede resolver en dos pasos: en el PASO1 el procedimiento ASMFC compila las rutinas y funciones en ASSEMBLER, cargando el módulo resultante en un disco auxiliar; en el PASO2 el procedimiento FTG1CLG compila el programa en FORTRAN, lo vincula con el módulo SIMPAT, que obtiene del disco auxiliar y ejecuta.

```
//..... JOB ....
//PASO1 EXEC PROC=ASMFC,PARM.ASM=DECK
//ASM.SYSPUNCH DD DSN=SIMPAT,DISP=(,PASS),UNIT=WORKDISK,
// SPACE=(CYL,(1,1)),DCB=BLKSIZE=80
//ASM.SYSIN DD DSN=SIMBOLIC.CNEA(SIMPAT),DISP=SHR
/*
//PASO2 EXEC PROC=FTG1CLG
//FORT.SYSIN DD *
Aquí tarjetas perforadas en FORTRAN
/*
//LKED.SYSIN DD DSN=SIMPAT,UNIT=WORKDISK,DISP=(OLD,DELETE)
/*
//GO.SYSIN DD *
Aquí tarjetas perforadas de DATOS
/*
//
```

Modelo 087. Ejecución de un programa en tarjetas perforadas en lenguaje absoluto de FORTRAN, que invoca rutinas y funciones en ASSEMBLER, residentes en la biblioteca, como un miembro independiente, con el nombre FOLDING. El problema se puede resolver en dos pasos: en el PASO1 el procedimiento ASMFC, compila las rutinas y funciones en ASSEMBLER, cargando el módulo resultante en un disco auxiliar; en el PASO2 el procedimiento FTG1LG vincula el programa en absoluto de FORTRAN con el módulo FOLDING y ejecuta.

```
//..... JOB ....
//PASO1 EXEC PROC=ASMFC,PARM.ASM=DECK
//ASM.SYSPUNCH DD DSN=FOLDING,DISP=(,PASS),UNIT=WORKDISK,
// SPACE=(CYL,(1,1)),DCB=BLKSIZE=80
//ASM.SYSIN DD DSN=SIMBOLIC.CNEA(FOLDING),DISP=SHR
/*
//PASO2 EXEC PROC=FTG1LG
//LKED.SYSIN DD DSN=FOLDING,UNIT=WORKDISK,DISP=(OLD,DELETE)
// DD *,DCB=BLKSIZE=80
Aquí tarjetas perforadas en absoluto de FORTRAN
/*
//GO.SYSIN DD *
Aquí tarjetas perforadas de DATOS
/*
//
```

Modelo 088. Ejecución de un programa en tarjetas perforadas en lenguaje simbó-

lico ASSEMBLER, que invoca rutinas y funciones en ASSEMBLER residentes en la biblioteca como un miembro independiente, con el nombre LIMITE. El problema se puede resolver en dos pasos: en el PASO1 el procedimiento ASMFCL compila y vincula las rutinas y funciones en ASSEMBLER, cargando el módulo resultante en una biblioteca temporaria particionada que se crea en ese momento y reside en un disco auxiliar; en el PASO2 el procedimiento ASMFCLG compila el programa en ASSEMBLER, lo vincula con el módulo LIMITE, que obtiene de la biblioteca temporaria y ejecuta.

```
//..... JOB ....
//PASO1 EXEC PROC=ASMFCL
//ASM.SYSIN DD DSN=SIMBOLIC.CNEA(LIMITE),DISP=SHR
//LKED.SYSLMOD DD DSN=TEMPOR.ADR(LIMITE),DISP=(NEW,PASS),
// UNIT=WORKDISK,SPACE=(CYL,(1,1,1)),
// DCB=(BLKSIZE=2400,RECFM=FB,LRECL=80)
/*
//PASO2 EXEC PROC=ASMFCLG
//ASM.SYSIN DD *
  Aquí tarjetas perforadas en ASSEMBLER
/*
//LKED.OTRALIB DD DSN=TEMPOR.ADR,DISP(OLD,DELETE)
//LKED.SYSIN DD *,DCB=BLKSIZE=80
      INCLUDE OTRALIB(LIMITE)
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: En el PASO1 se menciona el DCB con el cual se grabará el disco auxiliar. En el programa principal ASSEMBLER se ha abierto un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

Modelo 089 . Ejecución de un programa en tarjetas perforadas en lenguaje absoluto de ASSEMBLER, que invoca rutinas y funciones en ASSEMBLER, residentes en la biblioteca, como un miembro independiente, con el nombre DECAY. El problema se puede resolver en dos pasos: en el PASO1 el procedimiento ASMFCL compila y vincula las rutinas y funciones en ASSEMBLER, cargando el módulo resultante en una biblioteca temporaria particionada que se crea en ese momento y reside en un disco

auxiliar; en el PASO2 el procedimiento LKEDG vincula el programa en absoluto de ASSEMBLER con el módulo DECAY, que obtiene de la biblioteca temporaria creada en el paso anterior y ejecuta.

```
//..... JOB ....
//PASO1 EXEC PROC=ASMFCL
//ASM.SYSIN DD DSN=SIMBOLIC.CNEA(DECAY),DISP=SHR
//LKED.SYSLMOD DD DSN=TEMPOR.ADR(DECAY),DISP=(NEW,PASS),
// UNIT=WORKDISK,SPACE=(CYL,(1,1,1)),
// DCB=(BLKSIZE=800,RECFM=FB,LRECL=80)
/*
//PASO2 EXEC PROC=LKEDG
//LKED.OTRALIB DD DSN=TEMPOR.ADR,DISP=(OLD,DELETE)
//LKED.SYSIN DD *,DCB=BLKSIZE=80
Aquí tarjetas perforadas en absoluto de ASSEMBLER
      INCLUDE OTRALIB(DECAY)
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: En el PASO1 se menciona el DCB con el cual se graba el disco auxiliar. En el programa principal ASSEMBLER que dió origen al lote de tarjetas en absoluto, se abrió un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

Modelo 090 . Ejecución de un programa en tarjetas perforadas en lenguaje simbólico ASSEMBLER, que invoca rutinas y funciones en FORTRAN, residentes en la biblioteca como miembros independientes, con los nombres PROPERT, PARTICLE y STRUCT. El problema se puede resolver en tres pasos: en el PASO1 el procedimiento ASMFCL compila y vincula el programa en ASSEMBLER, cargando el módulo resultante, con el nombre PROPAR, en una biblioteca temporaria particionada que se crea en ese momento y reside en un disco auxiliar; en el PASO2 el procedimiento FTG1CL compila y vincula las rutinas y funciones en FORTRAN, cargando el módulo resultante, con el nombre TRIPLE, en la misma biblioteca temporaria creada en el paso anterior; en el PASO3 el procedimiento FTG1LG vincula los módulos PROPAR y TRIPLE, que obtiene de la biblioteca temporaria y ejecuta.

```

//..... JOB ....
//PAS01 EXEC PROC=ASMFCL
//ASM.SYSIN DD *
  Aquí tarjetas perforadas en ASSEMBLER
//LKED.SYSLMOD DD DSN=TEMPOR.ADR(PPROPAR),DISP=(NEW,PASS),
//  UNIT=WORKDISK,SPACE=(CYL,(1,1,1)),
//  DCB=(BLKSIZE=1200,RECFM=FB,LRECL=80)
/*
//PAS02 EXEC PROC=FTG1CL
//FORT.SYSIN DD DSN=SIMBOLIC.CNEA(PPROPERT),DISP=SHR
//          DD DSN=SIMBOLIC.CNEA(PARTICLE),DISP=SHR
//          DD DSN=SIMBOLIC.CNEA(STRUCT),DISP=SHR
//LKED.SYSLMOD DD DSN=TEMPOR.ADR(TRIPLE),DISP=OLD,
//  VOL=REF=*.PAS01.LKED.SYSLMOD
/*
//PAS03 EXEC PROC=FTG1LG
//LKED.OTRALIB DD DSN=TEMPOR.ADR,DISP=(OLD,DELETE)
//LKED.SYSIN DD *,DCB=BLKSIZE=80
//          INCLUDE OTRALIB(PPROPAR)
//          INCLUDE OTRALIB(TRIPLE)
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas de DATOS
/*
//

```

NOTA: En el PAS01 se menciona el DCB con el cual se grabará el disco auxiliar. En el programa principal ASSEMBLER se ha abierto un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

Modelo 091 . Ejecución de un programa en tarjetas perforadas en lenguaje absoluto de ASSEMBLER, que invoca rutinas y funciones en ASSEMBLER, residentes en la biblioteca, como un miembro independiente, con el nombre TECNICA. El problema se puede resolver en dos pasos: en el PAS01 el procedimiento ASMFCL compila y vincula las rutinas y funciones en ASSEMBLER, cargando el módulo resultante en una biblioteca temporaria particionada que se crea en ese momento y reside en un disco auxiliar; en el PAS02 el procedimiento LKEDG vincula el programa en absoluto de ASSEMBLER con el módulo TECNICA, que obtiene de la biblioteca temporaria creada en el paso anterior y ejecuta.

```

//..... JOB ....
//PAS01 EXEC PROC=ASMFCL
//ASM.SYSIN DD DSN=SIMBOLIC.CNEA(TECNICA),DISP=SHR

```

```
//LKED.SYSLMOD DD DSN=TEMPOR.ADR(TECNICA),DISP=(NEW,PASS),
// UNIT=WORKDISK,SPACE=(CYL,(1,1,1)),
// DCB=(BLKSIZE=1600,RECFM=FB,LRECL=80)
/*
//PASO2 EXEC PROC=LKEDG
//LKED.OTRALIB DD DSN=TEMPOR.ADR,DISP=(OLD,DELETE)
//LKED.SYSIN DD *,DCB=BLKSIZE=80
Aquí tarjetas perforadas en absoluto de ASSEMBLER
    INCLUDE OTRALIB(TECNICA)
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: En el PASO1 se menciona el DCB con el cual se grabará el disco auxiliar. En el programa principal ASSEMBLER que dió origen al lote de tarjetas en absoluto, se abrió un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

Modelo 092. Ejecución de un programa en tarjetas perforadas en lenguaje absoluto de ASSEMBLER, que invoca rutinas y funciones en FORTRAN, residentes en la biblioteca como miembros independientes, con los nombres POTENTIA, CLUSTER y STATES. El problema se puede resolver en tres pasos: en el PASO1 el procedimiento LKED vincula el programa en absoluto de ASSEMBLER, cargando el módulo resultante, con el nombre LIGHTNCL, en una biblioteca temporaria particionada que se crea en ese momento y reside en un disco auxiliar; en el PASO2 el procedimiento FTG1CL compila y vincula las rutinas y funciones en FORTRAN, cargando el módulo resultante, con el nombre FORTRUT, en la misma biblioteca temporaria creada en el paso anterior; en el PASO3 el procedimiento FTG1LG vincula los módulos LIGHTNCL y FORTRUT, que obtiene de la biblioteca temporaria y ejecuta.

```
//..... JOB ....
//PASO1 EXEC PROC=LKED
//LKED.SYSLMOD DD DSN=TEMPOR.ADR(LIGHTNCL),DISP=(NEW,PASS),
// UNIT=WORKDISK,SPACE=(CYL,(1,1,1)),
// DCB=(BLKSIZE=1200,RECFM=FB,LRECL=80)
//LKED.SYSIN DD *,DCB=BLKSIZE=80
Aquí tarjetas perforadas en absoluto de ASSEMBLER
/*
//PASO2 EXEC PROC=FTG1CL
```

```

//FORT.SYSIN DD DSN=SIMBOLIC.CNEA(POTENTIA),DISP=SHR
//          DD DSN=SIMBOLIC.CNEA(CLUSTER),DISP=SHR
//          DD DSN=SIMBOLIC.CNEA(STATES),DISP=SHR
//LKED.SYSLMOD DD DSN=TEMPOR.ADR(FORTRUT),DISP=OLD,
//  VOL=REF=*.PAS01.LKED.SYSLMOD
/*
//PAS03 EXEC PROC=FTG1LG
//LKED.OTRALIB DD DSN=TEMPOR.ADR,DISP=(OLD,DELETE)
//LKED.SYSIN DD *,DCB=BLKSIZE=80
//          INCLUDE OTRALIB(LIGHTNCL)
//          INCLUDE OTRALIB(FORTRUT)
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
//  Aquí tarjetas perforadas de DATOS
/*
//

```

NOTA: En el PAS01 se menciona el DCB con el cual se grabará el disco auxiliar. En el programa principal ASSEMBLER que dió origen al lote de tarjetas en absoluto, se abrió un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

PUNTO 9.

Ejecución de programas grabados en cintas magnéticas, que invocan rutinas y funciones residentes, como miembros independientes, en una biblioteca de nombre SIMBOLIC.CNEA.

Modelo 093 . Ejecución de un programa en lenguaje simbólico FORTRAN grabado en una cinta magnética, que invoca rutinas y funciones en FORTRAN residentes en la biblioteca, como miembros independientes, con los nombres NORM, CONTROL y FIGURA. El programa, de nombre PORTO, está grabado en el rótulo 16 de una cinta que tiene etiquetas modelo IBM y cuyo número de carrete es 38771.

```
//..... JOB ....
//PAS01 EXEC PROC=FTG1CLG
//FORT.SYSIN DD DSN=PORTO,UNIT=TAPE,VOL=SER=T38771,LABEL=(16,SL),
// DISP=(OLD,KEEP)
//          DD DSN=SIMBOLIC.CNEA(NORM),DISP=SHR
//          DD DSN=SIMBOLIC.CNEA(CONTROL),DISP=SHR
//          DD DSN=SIMBOLIC.CNEA(FIGURA),DISP=SHR
/*
//GO.SYSIN DD *
Aquí tarjetas perforadas de DATOS
/*
//
```

Modelo 094 . Ejecución de un programa en lenguaje absoluto de FORTRAN grabado en una cinta magnética, que invoca rutinas y funciones en FORTRAN residentes en la biblioteca, como miembros independientes, con los nombres PHASE,DIAGRAM, SPIN y FLUID. El programa, de nombre MIXTURE, está grabado en el rótulo 9 de una cinta que tiene etiquetas modelo IBM y cuyo número de carrete es 62924.

```
//..... JOB ....
//PAS01 EXEC PROC=FTG1CLG
//FORT.SYSIN DD DSN=SIMBOLIC.CNEA(PHASE),DISP=SHR
//          DD DSN=SIMBOLIC.CNEA(DIAGRAM),DISP=SHR
//          DD DSN=SIMBOLIC.CNEA(SPIN),DISP=SHR
/*
//LKED.SYSIN DD DSN=MIXTURE,UNIT=TAPE,VOL=SER=T62924,LABEL=(9,SL),
// DISP=(OLD,KEEP)
/*
//GO.SYSIN DD *
Aquí tarjetas perforadas de DATOS
```

```
/*
//
```

Modelo 095. Ejecución de un programa en lenguaje simbólico FORTRAN grabado en una cinta magnética, que invoca rutinas y funciones en ASSEMBLER residentes en la biblioteca, como un miembro independiente, con el nombre TORSIL. El programa, de nombre CURISIM, está grabado en el rótulo 22 de una cinta que tiene etiquetas modelo IBM y lleva el número de carrete 02158. El problema se puede resolver en dos pasos: en el PASO1 el procedimiento ASMFC compila las rutinas y funciones en ASSEMBLER, cargando el módulo resultante en un disco auxiliar; en el PASO2 el procedimiento FTG1CLG compila el programa CURISIM, lo vincula con el módulo TORSIL, que obtiene del disco auxiliar y ejecuta.

```
//..... JOB ....
//PASO1 EXEC PROC=ASMFC,PARM.ASM=DECK
//ASM.SYSPUNCH DD DSN=TORSIL,UNIT=WORKDISK,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1)),DCB=BLKSIZE=80
//ASM.SYSIN DD DSN=SIMBOLIC.CNEA(TORSIL),DISP=SHR
/*
//PASO2 EXEC PROC=FTG1CLG
//FORT.SYSIN DD DSN=CURISIM,UNIT=TAPE,VOL=SER=TO2153,LABEL=(22,SL),
// DISP=(OLD,KEEP)
//LKED.SYSIN DD DSN=TORSIL,UNIT=WORKDISK,DISP=(OLD,DELETE)
/*
//GO.SYSIN DD *
Aquí tarjetas perforadas de DATOS
/*
//
```

Modelo 096. Ejecución de un programa en lenguaje absoluto de FORTRAN, grabado en una cinta magnética, que invoca rutinas y funciones en ASSEMBLER residentes en la biblioteca, como un miembro independiente, con el nombre UNEQUAL. El programa, de nombre KERNELS, está grabado en el rótulo 33 de una cinta que lleva el número de carrete 86192 y tiene etiquetas modelo IBM. El problema se puede resolver en dos pasos: en el PASO1 el procedimiento ASMFC compila las rutinas y funciones en ASSEMBLER, cargando el módulo resultante en un disco auxiliar; en el PASO2 el procedimiento FTG1LG vincula el programa KERNELS con el módulo UNEQUAL, que obtiene del disco auxiliar y ejecuta.

```
//..... JOB ....
//PAS01 EXEC PROC=ASMFC,PARM.ASM=DECK
//ASM.SYSPUNCH DD DSN=UNEQUAL,UNIT=WORKDISK,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1)),DCB=BLKSIZE=80
//ASM.SYSIN DD DSN=SIMBOLIC.CNEA(UNEQUAL),DISP=SHR
/*
//PAS02 EXEC PROC=FTG1LG
//LKED.SYSIN DD DSN=KERNELS,UNIT=TAPE,VOL=SER=T86192,LABEL=(33,SL),
// DISP=(OLD,KEEP)
// DD DSN=UNEQUAL,UNIT=WORKDISK,DISP=(OLD,DELETE)
/*
//GO.SYSIN DD *
Aquí tarjetas perforadas de DATOS
/*
//
```

Modelo 097. Ejecución de un programa en lenguaje simbólico ASSEMBLER grabado en una cinta magnética, que invoca rutinas y funciones en ASSEMBLER residentes en la biblioteca, como un miembro independiente, con el nombre NUEVAFUN. El programa, de nombre SOLAM, está grabado en el rótulo 4 de una cinta que lleva el número de carrete 19324 y tiene etiquetas modelo IBM. El problema se puede resolver en dos pasos: en el PAS01 el procedimiento ASMFCL compila y vincula las rutinas y funciones en ASSEMBLER, cargando el módulo resultante en una biblioteca temporaria particionada que se crea en ese momento y reside en un disco auxiliar; en el PAS02 el procedimiento ASMFCLG compila el programa SOLAM, lo vincula con el módulo NUEVAFUN, que obtiene de la biblioteca temporaria creada en el paso anterior y ejecuta.

```
//..... JOB ....
//PAS01 EXEC PROC=ASMFCL
//ASM.SYSIN DD DSN=SIMBOLIC.CNEA(NUEVAFUN),DISP=SHR
//LKED.SYSLMOD DD DSN=TEMPOR.ADR(NUEVAFUN),DISP=(NEW,PASS),
// UNIT=WORKDISK,SPACE=(CYL,(1,1,1)),
// DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
/*
//PAS02 EXEC PROC=ASMFCLG
//ASM.SYSIN DD DSN=SOLAM,UNIT=TAPE,VOL=SER=T19324,LABEL=(4,SL),
// DISP=(OLD,KEEP)
/*
//LKED.OTRALIB DD DSN=TEMPOR.ADR,DISP=(OLD,DELETE)
//LKED.SYSIN DD *,DCB=BLKSIZE=80
INCLUDE OTRALIB(NUEVAFUN)
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
Aquí tarjetas perforadas de DATOS
```



```
/*
//
```

NOTA: En el PAS01 se menciona el DCB con el cual se grabará el disco auxiliar. En el programa principal ASSEMBLER se ha abierto un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

Modelo 098. Ejecución de un programa en lenguaje absoluto de ASSEMBLER grabado en una cinta magnética, que invoca rutinas y funciones en ASSEMBLER residentes en la biblioteca, como un miembro independiente, con el nombre ANALYTIC. El programa, de nombre GENERAT, está grabado en el rótulo 14 de una cinta que tiene etiquetas modelo IBM y lleva el número de carrete 42599. El problema se puede resolver en tres pasos: en el PAS01 el procedimiento LKED vincula el programa GENERAT, cargando el módulo resultante en una biblioteca temporaria particionada que se crea en ese momento y reside en un disco auxiliar; en el PAS02 el procedimiento ASMFCL compila y vincula las rutinas y funciones ANALYTIC, cargando el módulo resultante en la misma biblioteca temporaria creada en el paso anterior; en el PAS03 el procedimiento LKEDG vincula los módulos GENERAT y ANALYTIC, que obtiene de la biblioteca temporaria y ejecuta.

```
//..... JOB ....
//PAS01 EXEC PROC=LKED
//LKED.SYSLMOD DD DSN=TEMPOR.ADR(GENERAT),DISP=(NEW,PASS),
// UNIT=WORKDISK,SPACE=(CYL,(1,1,1)),
// DCB=(BLKSIZE=1200,RECFM=FB,LRECL=80)
//LKED.SYSIN DD DSN=GENERAT,UNIT=TAPE,VOL=SER=T42599,LABEL=(14,SL),
// DISP=(OLD,KEEP)
/*
//PAS02 EXEC PROC=ASMFCL
//ASM.SYSIN DD DSN=SIMBOLIC.CNEA(ANALYTIC),DISP=SHR
//LKED.SYSLMOD DD DSN=TEMPOR.ADR(ANALYTIC),DISP=OLD,
// VOL=REF=*.PAS01.LKED.SYSLMOD
/*
//PAS03 EXEC PROC=LKEDG
//LKED.OTRALIB DD DSN=TEMPOR.ADR,DISP=(OLD,DELETE)
//LKED.SYSIN DD *,DCB=BLKSIZE=80
//          INCLUDE OTRALIB(GENERAT)
/*          INCLUDE OTRALIB(ANALYTIC)
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
Aquí tarjetas perforadas de DATOS
```

```
/*
//
```

NOTA: En el PAS01 se menciona el DCB con el cual se grabará el disco auxiliar. En el programa principal ASSEMBLER que dió origen al módulo grabado en la cinta, se abrió un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

Modelo 099. Ejecución de un programa en lenguaje simbólico ASSEMBLER grabado en una cinta magnética, que invoca rutinas y funciones en FORTRAN residentes en la biblioteca, como miembros independientes, con los nombres UNITARY, AMPLITUD y SPIN. El programa, de nombre PARTICLE, está grabado en el rótulo 26 de una cinta que lleva el número de carrete 33312 y tiene etiquetas modelo IBM. El problema se puede resolver en tres pasos: en el PAS01 el procedimiento ASMFCL compila y vincula el programa ASSEMBLER, cargando el módulo resultante en una biblioteca temporaria particionada que se crea en ese momento y reside en un disco auxiliar; en el PAS02 el procedimiento FTG1CL compila y vincula las rutinas y funciones en FORTRAN, cargando el módulo resultante, con el nombre RUTITRES, en la misma biblioteca temporaria creada en el paso anterior; en el PAS03 el procedimiento FTG1LG vincula los módulos PARTICLE y RUTITRES, que obtiene de la biblioteca temporaria y ejecuta.

```
//..... JOB ....
//PAS01 EXEC PROC=ASMFCL
//ASM.SYSIN DD DSN=PARTICLE,UNIT=TAPE,VOL=SER=T33312,LABEL=(26,SL),
//  DISP=(OLD,KEEP)
//LKED.SYSLMOD DD DSN=TEMPOR.ADR(PARTICLE),DISP=(NEW,PASS),
//  UNIT=WORKDISK,SPACE=(CYL,(1,1,1)),
//  DCB=(BLKSIZE=800,RECFM=FB,LRECL=80)
/*
//PAS02 EXEC PROC=FTG1CL
//FORT.SYSIN DD DSN=SIMBOLIC.CNEA(UNITARY),DISP=SHR
//          DD DSN=SIMBOLIC.CNEA(AMPLITUD),DISP=SHR
//          DD DSN=SIMBOLIC.CNEA(SPIN),DISP=SHR
//LKED.SYSLMOD DD DSN=TEMPOR.ADR(RUTITRES),DISP=OLD,
//  VOL=REF=*,PAS01.LKED.SYSLMOD
/*
//PAS03 EXEC PROC=FTG1LG
//LKED.OTRALIB DD DSN=TEMPOR.ADR,DISP=(OLD,DELETE)
//LKED.SYSIN DD *,DCB=BLKSIZE=80
//          INCLUDE OTRALIB(PARTICLE)
```

```

                INCLUDE OTRALIB(RUTITRES)
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
    Aquí tarjetas perforadas de DATOS
/*
//

```

NOTA: En el PASO1 se menciona el DCB con el cual se grabará el disco auxiliar. En el programa principal ASSEMBLER se ha abierto un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

Modelo 100. Ejecución de un programa en lenguaje absoluto de ASSEMBLER grabado en una cinta magnética, que invoca rutinas y funciones en ASSEMBLER residentes en la biblioteca, como un miembro independiente, con el nombre CAUSALITY. El programa, de nombre MOMENTUM, está grabado en el rótulo 22 de una cinta que lleva el número de carrete 17226 y tiene etiquetas modelo IBM. El problema se puede resolver en tres pasos: en el PASO1 el procedimiento LKED vincula el programa MOMENTUM, cargando el módulo resultante en una biblioteca temporaria particionada que se crea en ese momento y reside en un disco auxiliar; en el PASO2 el procedimiento ASMFCL compila y vincula las rutinas y funciones en ASSEMBLER, cargando el módulo resultante en la misma biblioteca temporaria creada en el paso anterior; en el PASO3 el procedimiento LKEDG vincula los módulos MOMENTUM y CAUSALITY, que obtiene de la biblioteca temporaria y ejecuta.

```

//..... JOB ....
//PASO1 EXEC PROC=LKED
//LKED.SYSLMOD DD DSN=TEMPOR.ADR(MOMENTUM),DISP=(NEW,PASS),
//  UNIT=WORKDISK,SPACE=(CYL,(1,1,1)),
//  DCB=(BLKSIZE=1200,RECFM=FB,LRECL=80)
//LKED.SYSIN DD DSN=MOMENTUM,UNIT=TAPE,VOL=SER=T17226,LABEL=(22,SL),
//  DISP=(OLD,KEEP)
/*
//PASO2 EXEC PROC=ASMFCL
//ASM.SYSIN DD DSN=SIMBOLIC.CNEA(CAUSALITY),DISP=SHR
//LKED.SYSLMOD DD DSN=TEMPOR.ADR(CAUSALITY),DISP=OLD,
//  VOL=REF=*.PASO1.LKED.SYSLMOD
/*
//PASO3 EXEC PROC=LKEDG
//LKED.OTRALIB DD DSN=TEMPOR.ADR,DISP=(OLD,DELETE)
//LKED.SYSIN DD *,DCB=BLKSIZE=80
                INCLUDE OTRALIB(MOMENTUM)

```

```
INCLUDE OTRALIB(CAUSALITY)
```

```
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: En el PASO1 se menciona el DCB con el cual se grabará el disco auxiliar. En el programa principal ASSEMBLER que dió origen al módulo grabado en la cinta, se abrió un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

Modelo 101 . Ejecución de un programa en lenguaje absoluto de ASSEMBLER grabado en una cinta magnética, que invoca rutinas y funciones en FORTRAN residentes en la biblioteca, como miembros independientes, con los nombres NEUTRAL, CURRENT e INDUCED. El programa, de nombre GARGAM, está grabado en el rótulo 13 de una cinta que lleva el número de carrete 73201 y tiene etiquetas modelo IBM. El problema se puede resolver en tres pasos: en el PASO1 el procedimiento LKED vincula el programa GARGAM, cargando el módulo resultante en una biblioteca temporaria particionada que se crea en ese momento y reside en un disco auxiliar; en el PASO2 el procedimiento FTG1CL compila y vincula las rutinas y funciones en FORTRAN, cargando el módulo resultante, con el nombre BACKGRND, en la misma biblioteca temporaria creada en el paso anterior; en el PASO3 el procedimiento FTG1LG vincula los módulos GARGAM y BACKGRND, que obtiene de la biblioteca temporaria y ejecuta.

```
//..... JOB ....
//PASO1 EXEC PROC=LKED
//LKED.SYSLMOD DD DSN=TEMPOR.ADR(GRAGAM),DISP=(NEW,PASS),
// UNIT=WORKDISK,SPACE=(CYL,(1,1,1)),
// DCB=(BLKSIZE=1200,RECFM=FB,LRECL=80)
//LKED.SYSIN DD DSN=GARGAM,UNIT=TAPE,VOL=SER=T73201,LABEL=(13,SL),
// DISP=(OLD,KEEP)
/*
//PASO2 EXEC PROC=FTG1CL
//FORT.SYSIN DD DSN=SIMBOLIC.CNEA(NEUTRAL),DISP=SHR
//          DD DSN=SIMBOLIC.CNEA(CURRENT),DISP=SHR
//          DD DSN=SIMBOLIC.CNEA(INDUCED),DISP=SHR
//LKED.SYSLMOD DD DSN=TEMPOR.ADR(BACKGRND),DISP=OLD,
// VOL=REF=*.PASO1.LKED.SYSLMOD
/*
```

```

//PASO3 EXEC PROC=FTGILG
//LKED.OTRALIB DD DSN=TEMPOR.ADR,DISP=(OLD,DELETE)
//LKED.SYSIN DD *,DCB=BLKSIZE=80
        INCLUDE OTRALIB(GARGAM)
        INCLUDE OTRALIB(BACKGRND)
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
    Aquí tarjetas perforadas de DATOS
/*
//

```

NOTA: En el PASO1 se menciona el DCB con el cual se grabará el disco auxiliar. En el programa principal ASSEMBLER que dió origen al módulo grabado en la cinta, se abrió un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

PUNTO 10.

Grabación de programas que se presentan en tarjetas perforadas. Agrupación y duplicación de programas ya existentes en la biblioteca.

Modelo 102. Grabación en la biblioteca de un programa en lenguaje simbólico, de nombre CILINDRO. El programa se presenta en tarjetas que ya llevan, en las columnas 73 y siguientes, perforados los números que se desean adjudicar a los registros.

```
//..... JOB ....
//PAS01 EXEC PGM=IEBUPDTE,PARM=NEW
//SYSPRINT DD SYSOUT=A
//SYSUT2 DD DSN=SIMBOLIC.CNEA,DISP= OLD
//SYSIN DD *
./  ADD  NAME=CILINDRO
  Aquí tarjetas perforadas en lenguaje simbólico
./  ENDUP
/*
//
```

Modelo 103 . Grabación en la biblioteca de un programa en lenguaje simbólico, de nombre CAUTEL. El programa se presenta en tarjetas que no llevan numeración perforada en las columnas 73 y siguientes. La máquina numerará los registros de 100 en 100 a partir del número 20.

```
//..... JOB ....
//PAS01 EXEC PGM=IEBUPDTE,PARM=NEW
//SYSPRINT DD SYSOUT=A
//SYSUT2 DD DSN=SIMBOLIC.CNEA,DISP= OLD
//SYSIN DD *
./  ADD  NAME=CAUTEL
./  NUMBER NEW1=20,INCR=100
  Aquí tarjetas perforadas en lenguaje simbólico
./  ENDUP
/*
//
```

Modelo 104 . Grabación en la biblioteca de un programa en lenguaje simbólico, de nombre SORPRES, usando el parámetro SEQFLD. El programa se presenta en tarjetas que no llevan numeración perforada en las columnas 73 y siguientes. La máquina numerará los registros de 50 en 50 a partir del número 10. El parámetro SEQFLD permite elegir la posición que ocupará dicha numeración en el registro; en el mo-

delo, SEQFLD=763 significa: numerar desde la columna 76 y usar solamente 3 cifras. Si ese parámetro no se codifica, la máquina supone la opción tipo del sistema que es SEQFLD=738, es decir: numerar desde la columna 73 y usar 8 cifras; si las cifras significativas son menos de 8, la máquina rellenará con ceros a la izquierda de las mismas, hasta completar las 8 columnas (entre la 73 y la 80).

```
//..... JOB ....
//PAS01 EXEC PGM=IEBUPDTE,PARM=NEW
//SYSPRINT DD SYSOUT=A
//SYSUT2 DD DSN=SIMBOLIC.CNEA,DISP=OLD
//SYSIN DD *
./  ADD  NAME=SORPRES,SEQFLD=763
./  NUMBER NEW1=10,INCR=50
  Aquí tarjetas perforadas en lenguaje simbólico
./  ENDUP
/*
//
```

NOTA: Cuando se codifica en una sentencia ADD una opción SEQFLD distinta de la que el sistema supone por omisión, deberá repetirse cada vez que se actualiza el archivo, es decir, tiene que figurar el parámetro con el mismo valor original, en la sentencia CHANGE correspondiente.

Modelo 105 . Grabación simultánea en la biblioteca, como miembros independientes, de programas, rutinas, funciones y datos que se presentan en tarjetas que no llevan numeración perforada en las columnas 73 y siguientes. La máquina numerará los registros de la rutina de nombre LECTURA, de 20 en 20 a partir del número 100; los registros del programa JUNIN, de 100 en 100 a partir del número 150; los registros de la función de nombre AJUSTE, de 1 en 1 a partir del número 25 y finalmente el juego de datos de nombre DATPRUEB, se grabará omitiendo la numeración de sus registros.

```
//..... JOB ....
//PAS01 EXEC PGM=IEBUPDTE,PARM=NEW
//SYSPRINT DD SYSOUT=A
//SYSUT2 DD DSN=SIMBOLIC.CNEA,DISP=OLD
//SYSIN DD *
./  ADD  NAME=LECTURA
./  NUMBER NEW1=100,INCR=20
  Aquí tarjetas perforadas en lenguaje simbólico
```

```

./ ADD NAME=JUNIN
./ NUMBER NEW1=150,INCR=100
Aquí tarjetas perforadas en lenguaje simbólico
./ ADD NAME=AJUSTE
./ NUMBER NEW1=25,INCR=1
Aquí tarjetas perforadas en lenguaje simbólico
./ ADD NAME=DATPRUEB
Aquí tarjetas perforadas de DATOS
./ ENDUP
/*
//

```

Modelo 106. Grabación bajo un solo nombre de varios miembros independientes, ya existentes en la biblioteca. Se desean regrabar los miembros de nombres FOTON, ESPECTRO, DISPERS y TULIP como un solo miembro con el nombre RUTINAS. El problema se puede resolver en dos pasos: en el PASO1 el utilitario IEBGENER transfiere los miembros de la biblioteca a un disco auxiliar grabándolos con el nombre RUTINAS; en el PASO2 el utilitario IEBUPDTE transfiere el miembro RUTINAS del disco auxiliar a la biblioteca SIMBOLIC.CNEA.

```

//..... JOB ....
//PASO1 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSIN DD DUMMY
//SYSUT1 DD DSN=SIMBOLIC.CNEA(FOTON),DISP=SHR
//          DD DSN=SIMBOLIC.CNEA(ESPECTRO),DISP=SHR
//          DD DSN=SIMBOLIC.CNEA(DISPERS),DISP=SHR
//          DD DSN=SIMBOLIC.CNEA(TULIP),DISP=SHR
//SYSUT2 DD DSN=RUTINAS,UNIT=WORKDISK,DISP=(NEW,PASS),
//          SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
/*
//PASO2 EXEC PGM=IEBUPDTE,PARM=NEW
//SYSPRINT DD SYSOUT=A
//SYSUT2 DD DSN=SIMBOLIC.CNEA,DISP=OLD
//SYSIN DD *
./ ADD NAME=RUTINAS
//          DD DSN=RUTINAS,DISP=(OLD,DELETE)
./ ENDUP
/*
//

```

NOTA: En el PASO1 se menciona el DCB con el cual se grabará el disco auxiliar.

Modelo 107. Grabación en la biblioteca de un miembro actualizado o mas conveniente, que reemplaza a otro del mismo nombre. El miembro de nombre SUMATRES, ya existente en la biblioteca, es remplazado con ese mismo nombre, por un programa

en lenguaje simbólico que se presenta en tarjetas, que no llevan numeración perforada en las columnas 73 y siguientes. La máquina numerará los registros del nuevo programa, de 100 en 100 a partir del número 10.

```
//..... JOB ....
//PASO1 EXEC PGM=IEBUPDTE,PARM=MOD
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA,DISP=SHR
//SYSUT2 DD DSN=SIMBOLIC.CNEA,DISP=OLD
//SYSIN DD *
./  REPL  NAME=SUMATRES
./  NUMBER NEW1=10,INCR=100
  Aquí tarjetas perforadas en lenguaje simbólico
./  ENDUP
/*
//
```

Modelo 108. Duplicación de un miembro de la biblioteca que se grabará de nuevo con otro nombre. El objeto de esta maniobra es conservar uno de los miembros inalterado como original y ensayar modificaciones en el duplicado. El problema se puede resolver en dos pasos: en el PASO1 el utilitario IEBGENER transfiere el programa de nombre NIVEL25, que es el que se desea duplicar, de la biblioteca a un disco auxiliar; en el PASO2 el utilitario IEBUPDTE transfiere el programa del disco auxiliar a la biblioteca, con el nombre NIVEL26, sin cambiar la numeración que ya tenían sus registros.

```
//..... JOB ....
//PASO1 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA(NIVEL25),DISP=SHR
//SYSUT2 DD DSN=NIVEL25,DISP=(NEW,PASS),UNIT=WORKDISK,
//  SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//SYSIN DD DUMMY
/*
//PASO2 EXEC PGM=IEBUPDTE,PARM=NEW
//SYSPRINT DD SYSOUT=A
//SYSUT2 DD DSN=SIMBOLIC.CNEA,DISP=OLD
//SYSIN DD *
./  ADD  NAME=NIVEL26
//      DD DSN=NIVEL25,DISP=(OLD,DELETE)
./  ENDUP
/*
//
```

PUNTO 11.

Grabación de programas, residentes en cintas magnéticas, en una biblioteca de nombre SIMBOLIC.CNEA.

Modelo 109 . Grabación en la biblioteca de un programa en lenguaje simbólico, de nombre TWOPAR. El programa reside en el rótulo 9 de una cinta magnética que lleva el número de carrete 23375 y tiene etiquetas modelo IBM.

```
//..... JOB ....
//PAS01 EXEC PGM=IEBUPDTE,PARM=MOD
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=TWOPAR,UNIT=TAPE,VOL=SER=T23375,LABEL=(9,SL),
// DISP=(OLD,KEEP)
//SYSUT2 DD DSN=SIMBOLIC.CNEA,DISP=OLD
//SYSIN DD *
./ REPRO NEW=PO,MEMBER=TWOPAR,LIST=ALL
./ ENDUP
/*
//
```

Modelo 110 . Grabación en la biblioteca de un miembro actualizado o mas conveniente, que reemplaza a otro del mismo nombre. El miembro de nombre FUSS14, ya existente en la biblioteca, es reemplazado con ese mismo nombre, por un programa en lenguaje simbólico que reside en el rótulo 28 de una cinta magnética que lleva el número de carrete 82210 y tiene etiquetas modelo IBM. La máquina numerará los registros del nuevo programa de 50 en 50 a partir del número 100. El problema se puede resolver en dos pasos: en el PAS01 el utilitario IEBGENER transfiere el programa FUSS14 de la cinta en que reside a un disco auxiliar; en el PAS02 el utilitario IEBUPDTE transfiere el programa FUSS14 del disco a la biblioteca SIMBOLIC.CNEA, reemplazando con ese mismo nombre a la versión anterior.

```
//..... JOB ....
//PAS01 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=FUSS14,UNIT=TAPE,VOL=SER=T82210,LABEL=(28,SL),
// DISP=(OLD,KEEP)
//SYSUT2 DD DSN=FUSS14,UNIT=WORKDISK,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//SYSIN DD DUMMY
/*
//PAS02 EXEC PGM=IEBUPDTE,PARM=MOD
//SYSPRINT DD SYSOUT=A
```

```
//SYSUT1 DD DSN=SIMBOLIC.CNEA,DISP=SHR
//SYSUT2 DD DSN=SIMBOLIC.CNEA,DISP=OLD
//SYSIN DD *
./  REPL  NAME=FUSS14
./  NUMBER NEW1=100,INCR=50
//      DD DSN=FUSS14,DISP=(OLD,DELETE)
./  ENDUP
/*
//
```

NOTA: En el PASO1 se menciona el DCB con el cual se grabará el disco auxiliar.

Modelo 111. Grabación de un programa en FORTRAN que reside, junto con otros, en el mismo rótulo de una cinta magnética, formando un archivo secuencial. En el rótulo 2 de la cinta que lleva el número de carrete 98675 y tiene etiquetas modelo IBM, se ha grabado con el nombre PROGRAMS un conjunto de 25 programas y rutinas. Se desea seleccionar solamente el programa de nombre QUALITE que ocupa el noveno lugar en ese archivo y grabarlo en la biblioteca. El problema se puede resolver en dos pasos: en el PASO1 el utilitario IEBGENER identifica en la cinta, reconociendo la sentencia FORTRAN 'END', los nueve primeros programas y rutinas, creando con ellos una biblioteca temporaria particionada que reside en un disco auxiliar; en el PASO2 el utilitario IEBCOPY selecciona el programa QUALITE de la biblioteca temporaria creada en el paso anterior y lo transfiere a la biblioteca SIMBOLIC.CNEA.

```
//..... JOB ....
//PASO1 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD UNIT=TAPE,VOL=SER=T98675,LABEL=(2,SL),DISP=(OLD,KEEP),
//  DSN=PROGRAMS
//SYSUT2 DD UNIT=WORKDISK,DSN=TEMPOR.ADR,DISP=(NEW,PASS),
//  SPACE=(CYL,(1,1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//SYSIN DD *
      GENERATE MAXNAME=9,MAXGPS=9
      MEMBER NAME=UNO
GROUP1 RECORD IDENT=(3,'END',7)
      MEMBER NAME=DOS
GROUP2 RECORD IDENT=(3,'END',7)
      MEMBER NAME=TRES
GROUP3 RECORD IDENT=(3,'END',7)
      MEMBER NAME=CUATRO
GROUP4 RECORD IDENT=(3,'END',7)
      MEMBER NAME=CINCO
GROUP5 RECORD IDENT=(3,'END',7)
```

```

      MEMBER NAME=SEIS
GROUP6 RECORD IDENT=(3,'END',7)
      MEMBER NAME=SIETE
GROUP7 RECORD IDENT=(3,'END',7)
      MEMBER NAME=OCHO
GROUP8 RECORD IDENT=(3,'END',7)
      MEMBER NAME=QUALITE
GROUP9 RECORD IDENT=(3,'END',7)
/*
//PASO2 EXEC PGM=IEBCOPY
//SYSPRINT DD SYSOUT=A
//ENTRADA DD VOL=REF=*.PASO1.SYSUT2,DSN=TEMPOR.ADR,
//  DISP=(OLD,DELETE,KEEP)
//SALIDA DD DSN=SIMBOLIC.CNEA,DISP=OLD
//SYSIN DD *
  C O=SALIDA,I=ENTRADA
  S M=((QUALITE,,R))
/*
//

```

NOTA: El parámetro IDENT=(3,'END',7) del PASO1 significa: identificar una tira de 3 caracteres, igual a la que aparece entre comillas, que está colocada de la columna 7 en adelante, en la imagen de tarjeta grabada en la cinta. Se menciona el DCB con el cual se grabará el disco auxiliar.

Modelo 112 . Grabación simultánea en la biblioteca de programas, rutinas, funciones y datos que residen aisladamente en rótulos consecutivos o no consecutivos de una cinta magnética. En la cinta que lleva el número de carrete 90314 y tiene etiquetas modelo IBM, están grabados: el programa de nombre POLEMIC en el rótulo 27, la rutina de nombre SUPCOND en el rótulo 28, la función de nombre INVERS en el rótulo 31 y los datos de nombre ENTRADA en el rótulo 35. El problema se puede resolver en cuatro pasos: en todos ellos el utilitario IEBUPDTE transfiere, rótulo por rótulo, los miembros de la cinta a la biblioteca SIMBOLIC.CNEA.

```

//..... JOB ....
//PASO1 EXEC PGM=IEBUPDTE,PARM=MOD
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=POLEMIC,UNIT=TAPE,VOL=SER=T90314,LABEL=(27,SL),
//  DISP=(OLD,PASS)
//SYSUT2 DD DSN=SIMBOLIC.CNEA,DISP=OLD
//SYSIN DD *
./  REPRO NEW=PO,MEMBER=POLEMIC,LIST=ALL
./  ENDUP
/*
//PASO2 EXEC PGM=IEBUPDTE,PARM=MOD

```

```

//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SUPCOND,UNIT=TAPE,VOL=REF=*.PAS01.SYSUT1,
//  LABEL=(28,SL)
//SYSUT2 DD DSN=SIMBOLIC.CNEA,DISP=OLD
//SYSIN DD *
./  REPRO NEW=PO,MEMBER=SUPCOND,LIST=ALL
./  ENDUP
/*
//PAS03 EXEC PGM=IEBUPDTE,PARM=MOD
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=INVERS,UNIT=TAPE,VOL=REF=*.PAS01.SYSUT1,
//  LABEL=(31,SL)
//SYSUT2 DD DSN=SIMBOLIC.CNEA,DISP=OLD
//SYSIN DD *
./  REPRO NEW=PO,MEMBER=INVERS,LIST=ALL
./  ENDUP
/*
//PAS04 EXEC PGM=IEBUPDTE,PARM=MOD
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=ENTRADA,UNIT=TAPE,VOL=REF=*.PAS01.SYSUT1,
//  DISP=(OLD,KEEP),LABEL=(35,SL)
//SYSUT2 DD DSN=SIMBOLIC.CNEA,DISP=OLD
//SYSIN DD *
./  REPRO NEW=PO,MEMBER=ENTRADA,LIST=ALL
./  ENDUP
/*
//

```

NOTA: En el PAS04 al finalizar la transferencia, se codifica DISP=(OLD,KEEP) para que la cinta sea rebobinada.

Modelo 113. Grabación de varios programas en FORTRAN que residen, junto con otros, en el mismo rótulo de una cinta magnética, formando un archivo secuencial. En el rótulo 7 de la cinta que lleva el número de carrete 03291 y tiene etiquetas modelo IBM, se ha grabado con el nombre PGMFISIC un conjunto de 30 programas y rutinas. Se desean seleccionar solamente los programas de nombres PARAMET, PARTIC y SECTOR que ocupan el segundo, quinto y octavo lugar, respectivamente, en ese archivo y grabarlos en la biblioteca como miembros independientes. El problema se puede resolver en dos pasos: en el PAS01 el utilitario IEBGENER identifica en la cinta, reconociendo la sentencia FORTRAN 'END', los ocho primeros programas y rutinas, creando con ellos una biblioteca temporaria particionada que reside en un

disco auxiliar; en el PASO2 el utilitario IEBCOPY selecciona los miembros
PARAMET, PARTIC y SECTOR de la biblioteca temporaria creada en el paso anterior y
los transfiere a la biblioteca SIMBOLIC.CNEA.

```
//..... JOB ....
//PASO1 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=PGMFISIC,UNIT=TAPE,VOL=SER=T03291,LABEL=(7,SL),
// DISP=(OLD,KEEP)
//SYSUT2 DD DSN=TEMPOR.ADR,UNIT=WORKDISK,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//SYSIN DD *
    GENERATE MAXNAME=8,MAXGPS=8
    MEMBER NAME=UNO
GROUP1 RECORD IDENT=(3,'END',7)
    MEMBER NAME=PARAMET
GROUP2 RECORD IDENT=(3,'END',7)
    MEMBER NAME=TRES
GROUP3 RECORD IDENT=(3,'END',7)
    MEMBER NAME=CUATRO
GROUP4 RECORD IDENT=(3,'END',7)
    MEMBER NAME=PARTIC
GROUP5 RECORD IDENT=(3,'END',7)
    MEMBER NAME=SEIS
GROUP6 RECORD IDENT=(3,'END',7)
    MEMBER NAME=SIETE
GROUP7 RECORD IDENT=(3,'END',7)
    MEMBER NAME=SECTOR
GROUP8 RECORD IDENT=(3,'END',7)
/*
//PASO2 EXEC PGM=IEBCOPY
//SYSPRINT DD SYSOUT=A
//ENTRADA DD VOL=REF=*.PASO1.SYSUT2,DSN=TEMPOR.ADR,
// DISP=(OLD,DELETE,KEEP)
//SALIDA DD DSN=SIMBOLIC.CNEA,DISP=OLD
//SYSIN DD *
    C O=SALIDA,I=ENTRADA
    S M=((PARAMET,,R),(PARTIC,,R),(SECTOR,,R))
/*
//
```

NOTA: El parámetro IDENT=(3,'END',7) del PASO1 significa: identificar una tira de
3 caracteres, igual a la que aparece entre comillas, que está colocada de la co-
lumna 7 en adelante, en la imagen de tarjeta grabada en la cinta. Se menciona el
DCB con el cual se grabará el disco auxiliar.

PUNTO 12.

Corrección de programas grabados en una biblioteca de nombre SIMBOLIC.CNEA.

Modelo 114. Corrección, inserción y supresión de registros de un programa en FORTRAN, grabado en la biblioteca con el nombre ESPECTRO. El utilitario IEBUPDTE realiza la actualización que indican las tarjetas perforadas incluídas entre las sentencias CHANGE y ENDUP. Dicha actualización es la siguiente: a) se suprimen los registros 39288 a 39300 inclusive; b) se sustituye el registro 39420 ya existente, por la sentencia FORTRAN que figura en la tarjeta perforada; c) se suprime el registro 40025; d) entre los registros 41627 y 41638 ya existentes, se insertan las sentencias FORTRAN 41630, 41633 y 41635 que figuran en las tarjetas perforadas.

```
//..... JOB ....
//PAS01 EXEC PGM=IEBUPDTE,PARM=MOD
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA,DISP=SHR
//SYSUT2 DD DSN=SIMBOLIC.CNEA,DISP=OLD
//SYSIN DD *
./  CHANGE NAME=ESPECTRO,LIST=ALL
./  DELETE SEQ1=39288,SEQ2=39300
      1 CONTINUE                                00039420
./  DELETE SEQ1=40025,SEQ2=40025
      100 FORMAT(25X,4(F10.5,I5),'RESULTADOS',/)    00041630
          VECTOR=VECTOR*COS(A+B/C)                00041633
          DIMENSION A1(30,5),A2(2,3,3),A3(5,2,4),B(15) 00041635
./  ENDUP
/*
//
```

Modelo 115. Corrección, inserción y supresión de registros de un programa en FORTRAN, residente en la biblioteca con el nombre CUSTOM, que en su oportunidad fué grabado codificando el parámetro SEQFLD=754 en la sentencia ADD correspondiente. El utilitario IEBUPDTE realiza la actualización que indican las tarjetas perforadas incluídas entre las sentencias CHANGE y ENDUP. Dicha actualización es la siguiente: a) se suprimen los registros 4223 a 4250 inclusive; b) se sustituye el registro 4384 ya existente, por la sentencia FORTRAN que figura en la tarjeta perforada; c) se suprime el registro 5109; d) entre los registros 6020 y 6028 ya

existentes, se insertan las sentencias FORTRAN: 6022, 6024 y 6026 que figuran en las tarjetas perforadas

```
//..... JOB ....
//PASO1 EXEC PGM=IEBUPDTE,PARM=MOD
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA,DISP=SHR
//SYSUT2 DD DSN=SIMBOLIC.CNEA,DISP=OLD
//SYSIN DD *
./  CHANGE NAME=CUSTOM,LIST=ALL,SEQFLD=754
./  DELETE SEQ1=4223,SEQ2=4250
      VAR(I)=SQRT(Z(I,J)+Z(I,3))                                4384
./  DELETE SEQ1=5109,SEQ2=5109
      25 IF(A1-C1) 10,10,20                                      6022
      A(K,L)=SQRT(VAR(I)+BART(J))                                6024
      100 FORMAT(1H1)                                           6026
./  ENDUP
/*
//
```

Modelo 116. Corrección simultánea de tres miembros grabados en FORTRAN en la biblioteca, cuyos nombres son: FUNCION1, FUNCION2 y FUNCION3. El utilitario IEBUPDTE realiza la actualización que indican las tarjetas perforadas incluidas a continuación de cada sentencia CHANGE y consiste en lo siguiente: en FUNCION1 se suprimen los registros 99295 a 99300 inclusive; en FUNCION2 se insertan dos sentencias FORTRAN y se suprime el registro 100223; en FUNCION3 el comentario que se perfora en la tarjeta, reemplaza a una sentencia ya existente correspondiente al registro 5511.

```
//..... JOB ....
//PASO1 EXEC PGM=IEBUPDTE,PARM=MOD
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA,DISP=SHR
//SYSUT2 DD DSN=SIMBOLIC.CNEA,DISP=OLD
//SYSIN DD *
./  CHANGE NAME=FUNCION1,LIST=ALL
./  DELETE SEQ1=99295,SEQ2=99300
./  CHANGE NAME=FUNCION2,LIST=ALL
      INDICE=INDICE-2                                           00084220
      SUMA=GRT(II)-VEC(K)                                       00099476
./  DELETE SEQ1=100223,SEQ2=100223
./  CHANGE NAME=FUNCION3,LIST=ALL
C    EL LIMITE SUPERIOR DE MAXIN ES 250                        00005511
./  ENDUP
/*
//
```


Modelo 117. Corrección y ejecución de programas grabados en la biblioteca, codificando el parámetro COND que impide la ejecución de la versión anterior del programa, en caso de fallar la corrección. El programa en FORTRAN, de nombre AJUSTE, se corrige en el PASO1 con el utilitario IEBUPDTE y se ejecuta en el PASO2 con el procedimiento FTG1CLG; en la sentencia JOB del encabezamiento se ha codificado el parámetro COND de tal manera que el programa se ejecuta solamente en el caso en que la corrección haya resultado exitosa (todos los códigos de retorno iguales a cero). La corrección que se introduce en el programa es la que indican las tarjetas perforadas incluídas entre las sentencias CHANGE y ENDUP y consiste en lo siguiente: suprimir el registro 7325 e insertar la sentencia FORTRAN cuyo número de registro es 7848.

```
//..... JOB .....,
// COND=(0,NE)
//PASO1 EXEC PGM=IEBUPDTE,PARM=MOD
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA,DISP=SHR
//SYSUT2 DD DSN=SIMBOLIC.CNEA,DISP=OLD
//SYSIN DD *
./ CHANGE NAME=AJUSTE
./ DELETE SEQ1=7325,SEQ2=7325
    39 ARC=B+A*C**4
./ ENDUP
/*
//PASO2 EXEC PROC=FTG1CLG
//FORT.SYSIN DD DSN=SIMBOLIC.CNEA(AJUSTE),DISP=SHR
/*
//GO.SYSIN DD *
    Aquí tarjetas perforadas de DATOS
/*
//
```

00007848

NOTA: Como el PASO2 produce el listado del programa fuente durante su compilación, se ha suprimido la codificación del parámetro LIST en la sentencia CHANGE del PASO1.

PUNTO 13.

Agregado, inserción y renumeración total de registros de programas ya grabados en una biblioteca de nombre SIMBOLIC.CNEA.

Modelo 118. Agregado de rutinas y funciones, que se presentan en tarjetas, a continuación del último registro de un programa grabado en la biblioteca con el nombre SIMULESP; las tarjetas no llevan numeración perforada en las columnas 73 y siguientes. En la sentencia NUMBER el parámetro SEQ1 indica que el último registro del programa SIMULESP lleva el número 36500; el parámetro NEW1 indica que la máquina adjudicará al primer registro de la parte nueva el número 36600 y a partir de ahí numerará con incremento igual a 100.

```
//..... JOB ....
//PASO1 EXEC PGM=IEBUPDTE,PARM=MOD
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA,DISP=SHR
//SYSUT2 DD DSN=SIMBOLIC.CNEA,DISP=OLD
//SYSIN DD *
./  CHANGE NAME=SIMULESP,LIST=ALL
./  NUMBER SEQ1=36500,NEW1=36600,INCR=100,INSERT=YES
Aquí tarjetas perforadas en lenguaje simbólico, que se agregan
./  ENDUP
/*
//
```

Modelo 119. Inserción de un número muy grande de nuevos registros, que se presentan en tarjetas, entre dos registros consecutivos de un programa grabado en la biblioteca con el nombre PRUEBA32; las tarjetas no llevan numeración perforada en las columnas 73 y siguientes. Si se insertan 100 registros nuevos entre los de números 46520 y 46530, es evidente que la parte del programa que sigue a la inserción deberá experimentar la renumeración de sus registros. En la sentencia NUMBER el parámetro SEQ1 indica que la intercalación se hace a partir del registro 46520; el parámetro NEW1 indica que la máquina adjudicará al primer registro insertado el número 46600 y a partir de ahí numerará con incremento 100, tanto los registros intercalados como los que siguen a ellos, hasta el fin del programa.

```
//..... JOB ....
//PAS01 EXEC PGM=IEBUPDTE,PARM=MOD
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA,DISP=SHR
//SYSUT2 DD DSN=SIMBOLIC.CNEA,DISP=OLD
//SYSIN DD *
./  CHANGE NAME=PRUEBA32,LIST=ALL
./  NUMBER SEQ1=46520,NEW1=46600,INCR=100,INSERT=YES
  Aquí tarjetas perforadas en lenguaje simbólico, que se intercalan
./  ENDUP
/*
//
```

Modelo 120. Renumeración total de un programa que reside en la biblioteca con el nombre DENSESPC, a partir del número 200, con incremento 10.

```
//..... JOB ....
//PAS01 EXEC PGM=IEBUPDTE,PARM=MOD
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA,DISP=SHR
//SYSUT2 DD DSN=SIMBOLIC.CNEA,DISP=OLD
//SYSIN DD *
./  CHANGE NAME=DENSESPC,LIST=ALL
./  NUMBER SEQ1=ALL,NEW1=200,INCR=10
./  ENDUP
/*
//
```

PUNTO 14.

Transferencia a cinta magnética de uno o mas miembros de una biblioteca de nombre SIMBOLIC.CNEA.

Modelo 121 . Transferencia de un miembro de la biblioteca, de nombre TANDEM, al rótulo 13 de una cinta magnética que lleva el número de carrete 45760 y tiene etiquetas modelo IBM.

```
//..... JOB ....
//PAS01 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA(TANDEM),DISP=SHR
//SYSUT2 DD DSN=TANDEM,UNIT=TAPE,VOL=SER=T45760,LABEL=(13,SL),
//  DISP=(NEW,KEEP),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//SYSIN DD DUMMY
/*
//
```

NOTA: Se debe codificar el DCB con el cual se grabará la cinta.

Modelo 122 . Transferencia de miembros de la biblioteca a rótulos consecutivos de una misma cinta magnética. Los miembros de nombres APPLEMAN,FORDAP y SFLS se graban en los rótulos 7, 8 y 9, respectivamente, de una cinta cuyo carrete lleva el número 82222 y tiene etiquetas modelo IBM. El problema se puede resolver en tres pasos con el utilitario IEBGENER, a razón de un paso por cada rótulo.

```
//..... JOB ....
//PAS01 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA(APPLEMAN),DISP=SHR
//SYSUT2 DD DSN=APPLEMAN,UNIT=TAPE,VOL=SER=T82222,LABEL=(7,SL),
//  DISP=(NEW,PASS),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//SYSIN DD DUMMY
/*
//PAS02 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA(FORDAP),DISP=SHR
//SYSUT2 DD DSN=FORDAP,UNIT=TAPE,VOL=REF=*.PAS01.SYSUT2,
//  LABEL=(8,SL)
//SYSIN DD DUMMY
/*
//PAS03 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA(SFLS),DISP=SHR
//SYSUT2 DD DSN=SFLS,UNIT=TAPE,VOL=REF=*.PAS01.SYSUT2,
//  LABEL=(9,SL),DISP=(NEW,KEEP)
```

```
//SYSIN DD DUMMY
/*
//
```

NOTA: Se debe codificar el DCB con el cual se grabará la cinta. En el PAS03 se codifica DISP=(NEW,KEEP), para que la cinta sea rebobinada.

Modelo 123 . Transferencia de miembros de la biblioteca a un mismo rótulo de una cinta magnética. Los miembros de nombres CARDUP, MINNEW y MARS se graban, con el nombre GRUPO3, en el rótulo 5 de la cinta que lleva el número de carrete 80002 y tiene etiquetas modelo IBM.

```
//..... JOB ....
//PAS01 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA(CARDUP),DISP=SHR
//          DD DSN=SIMBOLIC.CNEA(MINNEW),DISP=SHR
//          DD DSN=SIMBOLIC.CNEA(MARS),DISP=SHR
//SYSUT2 DD DSN=GRUPO3,UNIT=TAPE,VOL=SER=T80002,LABEL=(5,SL),
//  DISP=(NEW,KEEP),DCB=(BLKSIZE=7200,LRECL=80,RECFM=FB)
//SYSIN DD DUMMY
/*
//
```

NOTA: Se debe codificar el DCB con el cual se grabará la cinta.

CAPITULO III

MODELOS PARA LA EJECUCION DE PROGRAMAS EN FORTRAN
QUE SE PRESENTAN EN DISTINTOS SOPORTES Y UTILIZAN UNA
BIBLIOTECA PARTICIONADA Y CATALOGADA DE PROGRAMAS
EN LENGUAJE SIMBOLICO COMO ARCHIVOS DE ENTRADA Y
SALIDA DE DATOS

PUNTO 1.

Ejecución de programas en FORTRAN que utilizan archivos de entrada de datos residentes en una biblioteca particionada y catalogada de programas en lenguaje simbólico, de nombre SIMBOLIC.CNEA.

Modelo 124 . Ejecución de un programa en tarjetas perforadas en lenguaje simbólico FORTRAN, cuyos datos de entrada constituyen un archivo que reside en la biblioteca, como un miembro independiente, con el nombre DATDR. El problema se puede resolver en dos pasos: en el PASO1 el utilitario IEBGENER transfiere el archivo de la biblioteca a un disco auxiliar; en el PASO2 el procedimiento FTG1CLG ejecuta el programa obteniendo los datos de entrada del disco auxiliar.

```
//..... JOB ....
//PASO1 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA(DATDR),DISP=SHR
//SYSUT2 DD DSN=DATDR,UNIT=WORKDISK,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//SYSIN DD DUMMY
/*
//PASO2 EXEC PROC=FTG1CLG
//FORT.SYSIN DD *
  Aquí tarjetas perforadas en FORTRAN
/*
//GO.FT20F001 DD DSN=DATDR,UNIT=WORKDISK,DISP=(OLD,DELETE)
/*
//
```

NOTA: El archivo de entrada de datos está definido en la sentencia GO del PASO2, en donde se le asigna el número de referencia 20; por lo tanto en el programa fuente cuando se lean datos de ese archivo, se codificará READ20. Se menciona el DCB con el cual se grabará el disco auxiliar.

Modelo 125 . Ejecución de un programa en tarjetas perforadas en lenguaje absoluto de FORTRAN, cuyos datos de entrada constituyen un archivo que reside en la biblioteca, como un miembro independiente, con el nombre OCTODAT. El problema se puede resolver en dos pasos: en el PASO1 el utilitario IEBGENER transfiere el archivo de la biblioteca a un disco auxiliar; en el PASO2 el procedimiento FTG1LG

ejecuta el programa obteniendo el archivo de entrada de datos del disco auxiliar.

```
//..... JOB ....
//PAS01 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA(OCTODAT),DISP=SHR
//SYSUT2 DD DSN=OCTODAT,UNIT=WORKDISK,DISP=(NEW,PASS),
//  SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//SYSIN DD DUMMY
/*
//PAS02 EXEC PROC=FTG1LG
//LKED.SYSIN DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas en absoluto de FORTRAN
/*
//GO.FT18F001 DD DSN=OCTODAT,UNIT=WORKDISK,DISP=(OLD,DELETE)
/*
//
```

NOTA: El archivo de entrada de datos está definido en la sentencia GO del PAS02, en donde se le asigna el número de referencia 18; por lo tanto en el programa fuente que dió origen al lote de tarjetas perforadas en absoluto de FORTRAN, se codificó READ18 para leer datos de ese archivo. Se menciona el DCB con el cual se grabará el disco auxiliar.

Modelo 126 . Ejecución de un programa en lenguaje simbólico FORTRAN de nombre PASERIN grabado en el rótulo 16 de una cinta magnética, cuyos datos de entrada constituyen un archivo que reside en la biblioteca, como un miembro independiente, con el nombre PASERDAT. La cinta lleva el número de carrete 16655 y tiene etiquetas modelo IBM. El problema se puede resolver en dos pasos: en el PAS01 el utilitario IEBGENER transfiere el archivo de la biblioteca a un disco auxiliar; en el PAS02 el procedimiento FTG1CLG ejecuta el programa PASERIN obteniendo del disco auxiliar el archivo de entrada de datos.

```
//..... JOB ....
//PAS01 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA(PASERDAT),DISP=SHR
//SYSUT2 DD DSN=PASERDAT,UNIT=WORKDISK,DISP=(NEW,PASS),
//  SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//SYSIN DD DUMMY
/*
//PAS02 EXEC PROC=FTG1CLG
//FORT.SYSIN DD DSN=PASERIN,UNIT=TAPE,VOL=SER=T16655,LABEL=(16,SL),
```

```
// DISP=(OLD,KEEP)
/*
//GO.FT03F001 DD DSN=PASERDAT,UNIT=WORKDISK,DISP=(OLD,DELETE)
/*
//
```

NOTA: El archivo de entrada de datos está definido en la sentencia GO del PAS02, en donde se le asigna el número de referencia 3; por lo tanto en el programa fuente cuando se lean datos de ese archivo, se codificará READ3. Se menciona el DCB con el cual se grabará el disco auxiliar.

Modelo 127. Ejecución de un programa en lenguaje absoluto de FORTRAN de nombre COLS, grabado en el rótulo 7 de una cinta magnética, cuyos datos de entrada constituyen un archivo que reside en la biblioteca, como un miembro independiente, con el nombre COLDATOS. La cinta lleva el número de carrete 03824 y tiene etiquetas modelo IBM. El problema se puede resolver en dos pasos: en el PAS01 el utilitario IEBGENER transfiere el archivo de la biblioteca a un disco auxiliar; en el PAS02 el procedimiento FTG1LG ejecuta el programa COLS obteniendo del disco auxiliar el archivo de entrada de datos.

```
//..... JOB ....
//PAS01 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA(COLDATOS),DISP=SHR
//SYSUT2 DD DSN=COLDATOS,UNIT=WORKDISK,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//SYSIN DD DUMMY
/*
//PAS02 EXEC PROC=FTG1LG
//LKED.SYSIN DD DSN=COLS,UNIT=TAPE,VOL=SER=T03824,LABEL=(7,SL),
// DISP=(OLD,KEEP)
/*
//GO.FT15F001 DD DSN=COLDATOS,UNIT=WORKDISK,DISP=(OLD,DELETE)
/*
//
```

NOTA: El archivo de entrada de datos está definido en la sentencia GO del PAS02, en donde se le asigna el número de referencia 15; por lo tanto en el programa fuente que dió origen al módulo grabado en la cinta, se codificó READ15 para leer datos de ese archivo. Se menciona el DCB con el cual se grabará el disco auxi-

liar.

Modelo 128 . Ejecución de un programa grabado en la biblioteca con el nombre JAR, cuyos datos de entrada constituyen un archivo que reside en la propia biblioteca, como un miembro independiente, con el nombre JARDATOS. El problema se puede resolver en dos pasos: en el PASO1 el utilitario IEBGENER transfiere el archivo de la biblioteca a un disco auxiliar; en el PASO2 el procedimiento FTG1CLG ejecuta el programa JAR obteniendo del disco auxiliar el archivo de entrada de datos.

```
//..... JOB ....
//PASO1 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA(JARDATOS),DISP=SHR
//SYSUT2 DD DSN=JARDATOS,UNIT=WORKDISK,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//SYSIN DD DUMMY
/*
//PASO2 EXEC PROC=FTG1CLG
//FORT.SYSIN DD DSN=SIMBOLIC.CNEA(JAR),DISP=SHR
/*
//GO.FT08F001 DD DSN=JARDATOS,UNIT=WORKDISK,DISP=(OLD,DELETE)
/*
//
```

NOTA: El archivo de entrada de datos está definido en la sentencia GO del PASO2, en donde se le asigna el número de referencia 8; por lo tanto en el programa fuente cuando se lean datos de ese archivo, se codificará READ8. Se menciona el DCB con el cual se grabará el disco auxiliar.

Modelo 129 . Ejecución de un programa en tarjetas perforadas en lenguaje simbólico FORTRAN, cuyos datos de entrada son tres archivos distintos que residen en la biblioteca, como miembros independientes, con los nombres DATOSA, DATOSB y DATOSC. El problema se puede resolver en cuatro pasos: en los tres primeros (PASO1, PASO2 y PASO3) el utilitario IEBGENER transfiere, de a uno, los tres archivos al mismo disco auxiliar; en el PASO4 el procedimiento FTG1CLG ejecuta el programa obteniendo del disco auxiliar los archivos de entrada de datos.

```
//..... JOB ....
```

```

//PAS01 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA(DATOSA),DISP=SHR
//SYSUT2 DD UNIT=WORKDISK,DISP=(NEW,PASS),SPACE=(CYL,(1,1)),
//   DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB),DSN=DATOSA
//SYSIN DD DUMMY
/*
//PAS02 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA(DATOSB),DISP=SHR
//SYSUT2 DD VOL=REF=*.PAS01.SYSUT2,DISP=(NEW,PASS),
//   SPACE=(CYL,(1,1)),DSN=DATOSB
//SYSIN DD DUMMY
/*
//PAS03 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA(DATOSC),DISP=SHR
//SYSUT2 DD VOL=REF=*.PAS01.SYSUT2,DISP=(NEW,PASS),
//   SPACE=(CYL,(1,1)),DSN=DATOSC
//SYSIN DD DUMMY
/*
//PAS04 EXEC PROC=FTG1CLG
//FORT.SYSIN DD *
  Aquí tarjetas perforadas en FORTRAN
/*
//GO.FT04F001 DD DSN=DATOSA,UNIT=WORKDISK,DISP=(OLD,DELETE)
//GO.FT08F001 DD DSN=DATOSB,UNIT=WORKDISK,DISP=(OLD,DELETE)
//GO.FT11F001 DD DSN=DATOSC,UNIT=WORKDISK,DISP=(OLD,DELETE)
/*
//

```

NOTA: Los archivos de entrada de datos están definidos en las sentencias GO del PAS04, en donde se les asignan los números de referencia 4, 8 y 11; por lo tanto en el programa fuente, cuando se lean datos de esos archivos, se codificará READ4, READ8 y READ11 respectivamente. Se menciona el DCB con el cual se grabará el disco auxiliar.

Modelo 130 . Ejecución de un programa en tarjetas perforadas en lenguaje absoluto de FORTRAN, cuyos datos de entrada son tres archivos distintos que residen en la biblioteca, como miembros independientes, con los nombres ARCHIVO7, ARCHIVO8 y ARCHIVO9. El problema se puede resolver en cuatro pasos: en los tres primeros (PAS01, PAS02 y PAS03) el utilitario IEBGENER transfiere, de a uno, los tres archivos al mismo disco auxiliar; en el PAS04 el procedimiento FTG1LG ejecuta el programa obteniendo del disco auxiliar los archivos de entrada de datos.

```

//..... JOB ....

```

```

//PAS01 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA(ARCHIVO7),DISP=SHR
//SYSUT2 DD UNIT=WORKDISK,DISP=(NEW,PASS),SPACE=(CYL,(1,1)),
//  DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB),DSN=ARCHIVO7
//SYSIN DD DUMMY
/*
//PAS02 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA(ARCHIVO8),DISP=SHR
//SYSUT2 DD VOL=REF=*.PAS01.SYSUT2,DISP=(NEW,PASS),
//  SPACE=(CYL,(1,1)),DSN=ARCHIVO8
//SYSIN DD DUMMY
/*
//PAS03 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA(ARCHIVO9),DISP=SHR
//SYSUT2 DD VOL=REF=*.PAS01.SYSUT2,DISP=(NEW,PASS),
//  SPACE=(CYL,(1,1)),DSN=ARCHIVO9
//SYSIN DD DUMMY
/*
//PAS04 EXEC PROC=FTG1LG
//LKED.SYSIN DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas en absoluto de FORTRAN
/*
//GO.FT13F001 DD DSN=ARCHIVO7,UNIT=WORKDISK,DISP=(OLD,DELETE)
//GO.FT02F001 DD DSN=ARCHIVO8,UNIT=WORKDISK,DISP=(OLD,DELETE)
//GO.FT14F001 DD DSN=ARCHIVO9,UNIT=WORKDISK,DISP=(OLD,DELETE)
/*
//

```

NOTA: Los archivos de entrada de datos están definidos en las sentencias GO del PAS04, en donde se les asignan los números de referencia 13, 2 y 14; por lo tanto en el programa fuente que dió origen al lote de tarjetas perforadas en absoluto de FORTRAN, se codificó READ13, READ2 y READ14 respectivamente, para leer datos de esos archivos. Se menciona el DCB con el cual se grabará el disco auxiliar.

Modelo 131 . Ejecución de un programa en lenguaje simbólico FORTRAN de nombre FUR, grabado en el rótulo 12 de una cinta magnética, cuyos datos de entrada son dos archivos distintos que residen en la biblioteca, como miembros independientes, con los nombres FURDAT25 y FURDAT26. La cinta lleva el número de carrete 90009 y tiene etiquetas modelo IBM. El problema se pueda resolver en tres pasos: en los dos primeros (PAS01 y PAS02) el utilitario IEBGENER transfiere, de a uno, los dos archivos al mismo disco auxiliar; en el PAS03 el procedimiento FTG1CLG ejecuta el programa FUR obteniendo del disco auxiliar los archivos de entrada de

datos.

```
//..... JOB ....
//PAS01 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA(FURDAT25),DISP=SHR
//SYSUT2 DD UNIT=WORKDISK,DISP=(NEW,PASS),SPACE=(CYL,(1,1)),
//   DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB),DSN=FURDAT25
//SYSIN DD DUMMY
/*
//PAS02 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA(FURDAT26),DISP=SHR
//SYSUT2 DD VOL=REF=*.PAS01.SYSUT2,DISP=(NEW,PASS),
//   SPACE=(CYL,(1,1)),DSN=FURDAT26
//SYSIN DD DUMMY
/*
//PAS03 EXEC PROC=FTG1CLG
//FORT.SYSIN DD DSN=FUR,UNIT=TAPE,VOL=SER=T90009,LABEL=(12,SL),
//   DISP=(OLD,KEEP)
/*
//GO.FT11F001 DD DSN=FURDAT25,UNIT=WORKDISK,DISP=(OLD,DELETE)
//GO.FT15F001 DD DSN=FURDAT26,UNIT=WORKDISK,DISP=(OLD,DELETE)
/*
//
```

NOTA: Los archivos de entrada de datos están definidos en las sentencias GO del PASO3, en donde se les asignan los números de referencia 11 y 15; por lo tanto en el programa fuente, cuando se lean datos de esos archivos se codificará READ11 y READ15 respectivamente. Se menciona el DCB con el cual se grabará el disco auxiliar.

Modelo 132 . Ejecución de un programa en lenguaje absoluto de FORTRAN de nombre WAT, grabado en el rótulo 5 de una cinta magnética, cuyos datos de entrada son tres archivos distintos que residen en la biblioteca, como miembros independientes, con los nombres WATARCH1, WATARCH2 y WATARCH3. La cinta lleva el número de carrete 62041 y tiene etiquetas modelo IBM. El problema se puede resolver en cuatro pasos: en los tres primeros (PAS01, PASO2 y PASO3) el utilitario IEBGENER transfiere, de a uno, los tres archivos al mismo disco auxiliar; en el PASO4 el procedimiento FTG1LG ejecuta el programa WAT obteniendo del disco auxiliar los archivos de entrada de datos.

```
//..... JOB ....
```

```

//PAS01 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA(WATARCH1),DISP=SHR
//SYSUT2 DD UNIT=WORKDISK,DISP=(NEW,PASS),SPACE=(CYL,(1,1)),
//   DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB),DSN=WATARCH1
//SYSIN DD DUMMY
/*
//PAS02 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA(WATARCH2),DISP=SHR
//SYSUT2 DD VOL=REF=*.PAS01.SYSUT2,DISP=(NEW,PASS),
//   SPACE=(CYL,(1,1)),DSN=WATARCH2
//SYSIN DD DUMMY
/*
//PAS03 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA(WATARCH3),DISP=SHR
//SYSUT2 DD VOL=REF=*.PAS01.SYSUT2,DISP=(NEW,PASS),
//   SPACE=(CYL,(1,1)),DSN=WATARCH3
//SYSIN DD DUMMY
/*
//PAS04 EXEC PROC=FTG1LG
//LKED.SYSIN DD DSN=WAT,UNIT=TAPE,VOL=SER=T62041,LABEL=(5,SL),
//   DISP=(OLD,KEEP)
/*
//GO.FT13F001 DD DSN=WATARCH1,UNIT=WORKDISK,DISP=(OLD,DELETE)
//GO.FT15F001 DD DSN=WATARCH2,UNIT=WORKDISK,DISP=(OLD,DELETE)
//GO.FT11F001 DD DSN=WATARCH3,UNIT=WORKDISK,DISP=(OLD,DELETE)
/*
//

```

NOTA: Los archivos de entrada de datos están definidos en las sentencias GO del PAS04, en donde se les asignan los números de referencia 13, 15 y 11; por lo tanto en el programa fuente que dió origen al módulo grabado en la cinta, se codificó READ13, READ15 y READ11 respectivamente, para leer datos de esos archivos. Se menciona el DCB con el cual se grabará el disco auxiliar.

Modelo 133. Ejecución de un programa grabado en la biblioteca con el nombre MACK, cuyos datos de entrada son dos archivos distintos, que residen en la propia biblioteca, como miembros independientes, con los nombres DATMACKA y DATMACKB. El problema se puede resolver en tres pasos: en los dos primeros (PAS01 y PAS02) el utilitario IEBGENER transfiere, de a uno, los dos archivos al mismo disco auxiliar; en el PAS03 el procedimiento FTG1CLG ejecuta el programa MACK obteniendo del disco auxiliar los archivos de entrada de datos.

```

//..... JOB ....
//PAS01 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA(DATMACKA),DISP=SHR
//SYSUT2 DD UNIT=WORKDISK,DISP=(NEW,PASS),SPACE=(CYL,(1,1)),
//   DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB),DSN=DATMACKA
//SYSIN DD DUMMY
/*
//PAS02 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA(DATMACKB),DISP=SHR
//SYSUT2 DD VOL=REF=*.PAS01.SYSUT2,DISP=(NEW,PASS),
//   SPACE=(CYL,(1,1)),DSN=DATMACKB
//SYSIN DD DUMMY
/*
//PAS03 EXEC PROC=FTG1CLG
//FORT.SYSIN DD DSN=SIMBOLIC.CNEA(MACK),DISP=SHR
/*
//GO.FT16F001 DD DSN=DATMACKA,UNIT=WORKDISK,DISP=(OLD,DELETE)
//GO.FT15F001 DD DSN=DATMACKB,UNIT=WORKDISK,DISP=(OLD,DELETE)
/*
//

```

NOTA: Los archivos de entrada de datos están definidos en las sentencias GO del PAS03, en donde se les asignan los números de referencia 16 y 15; por lo tanto en el programa fuente, cuando se lean datos de esos archivos, se codificará READ16 y READ15 respectivamente. Se menciona el DCB con el cual se grabará el disco auxiliar.

PUNTO 2.

Ejecución de programas en FORTRAN que graban archivos de salida de datos en una biblioteca particionada y catalogada de programas en lenguaje simbólico, de nombre SIMBOLIC.CNEA.

Modelo 134 . Ejecución de un programa en tarjetas perforadas en lenguaje simbólico FORTRAN, cuyos resultados constituyen un archivo de salida que se graba en la biblioteca como un miembro independiente. El problema se puede resolver en dos pasos: en el PASO1 el procedimiento FTG1CLG ejecuta el programa grabando los resultados, con el nombre RESULDAT, en un disco auxiliar; en el PASO2 el utilitario IEBUPDTE transfiere ese archivo del disco auxiliar a la biblioteca SIMBOLIC.CNEA.

```
//..... JOB ....
//PASO1 EXEC PROC=FTG1CLG
//FORT.SYSIN DD *
  Aquí tarjetas perforadas en FORTRAN
/*
//GO.FT17FO01 DD DSN=RESULDAT,UNIT=WORKDISK,DISP=(NEW,PASS),
//  SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//GO.SYSIN DD *
  Aquí tarjetas perforadas de DATOS
/*
//PASO2 EXEC PGM=IEBUPDTE,PARM=NEW
//SYSPRINT DD SYSOUT=A
//SYSUT2 DD DSN=SIMBOLIC.CNEA,DISP=OLD
//SYSIN DD *
./  ADD  NAME=RESULDAT
//      DD DSN=RESULDAT,DISP=(OLD,DELETE)
./  ENDUP
/*
//
```

NOTA: El archivo de salida de datos está definido en la primera sentencia GO del PASO1, en donde se le asigna el número de referencia 17; por lo tanto en el programa fuente, cuando se graben resultados en ese archivo, se codificará WRITE17. Se menciona el DCB con el cual se grabará el disco auxiliar.

Modelo 135 . Ejecución de un programa en tarjetas perforadas en lenguaje absoluto de FORTRAN, cuyos resultados constituyen un archivo de salida que se graba en la biblioteca como un miembro independiente. El problema se puede resolver en dos

pasos: en el PAS01 el procedimiento FTG1LG ejecuta el programa grabando los resultados, con el nombre CALCULO4, en un disco auxiliar; en el PAS02 el utilitario IEBUPDTE transfiere ese archivo del disco auxiliar a la biblioteca SIMBOLIC.CNEA.

```
//..... JOB ....
//PAS01 EXEC PROC=FTG1LG
//LKED.SYSIN DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas en absoluto de FORTRAN
/*
//GO.FT12F001 DD DSN=CALCULO4,UNIT=WORKDISK,DISP=(NEW,PASS),
//  SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//GO.SYSIN DD *
  Aquí tarjetas perforadas de DATOS
/*
//PAS02 EXEC PGM=IEBUPDTE,PARM=NEW
//SYSPRINT DD SYSOUT=A
//SYSUT2 DD DSN=SIMBOLIC.CNEA,DISP=OLD
//SYSIN DD *
./  ADD  NAME=CALCULO4
//      DD  DSN=CALCULO4,DISP=(OLD,DELETE)
./  ENDUP
/*
//
```

NOTA: El archivo de salida de datos está definido en la primera sentencia GO del PAS01, en donde se le asigna el número de referencia 12; por lo tanto en el programa fuente que dió origen al lote de tarjetas en absoluto de FORTRAN, se codificó WRITE12 para grabar resultados en ese archivo. Se menciona el DCB con el cual se grabará el disco auxiliar.

Modelo 136 . Ejecución de un programa en lenguaje simbólico FORTRAN de nombre COLATER, residente en el rótulo 8 de una cinta magnética, cuyos resultados constituyen un archivo de salida que se graba en la biblioteca, como un miembro independiente. La cinta lleva el número de carrete 21164 y tiene etiquetas modelo IBM. El problema se puede resolver en dos pasos: en el PAS01 el procedimiento FTG1CLG ejecuta el programa grabando los resultados, con el nombre RESULCOL, en un disco auxiliar; en el PAS02 el utilitario IEBUPDTE transfiere ese archivo del disco auxiliar a la biblioteca SIMBOLIC.CNEA.

```
//..... JOB ....
```

```
//PAS01 EXEC PROC=FTG1CLG
//FORT.SYSIN DD DSN=COLATER,UNIT=TAPE,VOL=SER=T21164,LABEL=(8,SL),
//  DISP=(OLD,KEEP)
/*
//GO.FT11F001 DD DSN=RESULCOL,UNIT=WORKDISK,DISP=(NEW,PASS),
//  SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//GO.SYSIN DD *
  Aquí tarjetas perforadas de DATOS
/*
//PAS02 EXEC PGM=IEBUPDTE,PARM=NEW
//SYSPRINT DD SYSOUT=A
//SYSUT2 DD DSN=SIMBOLIC.CNEA,DISP=OLD
//SYSIN DD *
./  ADD  NAME=RESULCOL
//      DD DSN=RESULCOL,DISP=(OLD,DELETE)
./  ENDUP
/*
//
```

NOTA: El archivo de salida de datos está definido en la primera sentencia GO del PAS01, en donde se le asigna el número de referencia 11; por lo tanto en el programa fuente, cuando se graben resultados en ese archivo, se codificará WRITE11. Se menciona el DCB con el cual se grabará el disco auxiliar.

Modelo 137 . Ejecución de un programa en lenguaje absoluto de FORTRAN de nombre DIVERGEN, residente en el rótulo 6 de una cinta magnética, cuyos resultados constituyen un archivo de salida que se graba en la biblioteca, como un miembro independiente. La cinta lleva el número de carrete 38750 y tiene etiquetas modelo IBM. El problema se puede resolver en dos pasos: en el PAS01 el procedimiento FTG1LG ejecuta el programa grabando los resultados, con el nombre DIVERRES, en un disco auxiliar; en el PAS02 el utilitario IEBUPDTE transfiere ese archivo del disco auxiliar a la biblioteca SIMBOLIC.CNEA.

```
//..... JOB ....
//PAS01 EXEC PROC=FTG1LG
//LKED.SYSIN DD DSN=DIVERGEN,UNIT=TAPE,VOL=SER=T38750,LABEL=(6,SL),
//  DISP=(OLD,KEEP)
/*
//GO.FT18F001 DD DSN=DIVERRES,UNIT=WORKDISK,DISP=(NEW,PASS),
//  SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//GO.SYSIN DD *
  Aquí tarjetas perforadas de DATOS
/*
//PAS02 EXEC PGM=IEBUPDTE,PARM=NEW
```

```
//SYSPRINT DD SYSOUT=A
//SYSUT2 DD DSN=SIMBOLIC.CNEA,DISP=OLD
//SYSIN DD *
./ ADD NAME=DIVERRES
// DD DSN=DIVERRES,DISP=(OLD,DELETE)
./ ENDUP
/*
//
```

NOTA: El archivo de salida de datos está definido en la primera sentencia GO del PAS01, en donde se le asigna el número de referencia 18; por lo tanto en el programa fuente que dió origen al módulo grabado en la cinta, se codificó WRITE18 para grabar resultados en ese archivo. Se menciona el DCB con el cual se grabará el disco auxiliar.

Modelo 138 . Ejecución de un programa residente en la biblioteca con el nombre CATASIN, cuyos resultados constituyen un archivo de salida que se graba en la propia biblioteca, como un miembro independiente. El problema se puede resolver en dos pasos: en el PAS01 el procedimiento FTG1CLG ejecuta el programa grabando los resultados, con el nombre CATARES, en un disco auxiliar; en el PAS02 el utilitario IEBUPDTE transfiere ese archivo del disco auxiliar a la biblioteca SIMBOLIC.CNEA.

```
//..... JOB ....
//PAS01 EXEC PROC=FTG1CLG
//FORT.SYSIN DD DSN=SIMBOLIC.CNEA(CATASIN),DISP=SHR
/*
//GO.FT11F001 DD DSN=CATARES,UNIT=WORKDISK,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//GO.SYSIN DD *
Aquí tarjetas perforadas de DATOS
/*
//PAS02 EXEC PGM=IEBUPDTE,PARM=NEW
//SYSPRINT DD SYSOUT=A
//SYSUT2 DD DSN=SIMBOLIC.CNEA,DISP=OLD
//SYSIN DD *
./ ADD NAME=CATARES
// DD DSN=CATARES,DISP=(OLD,DELETE)
./ ENDUP
/*
//
```

NOTA: El archivo de salida de datos está definido en la primera sentencia GO del

PAS01, donde se le asigna el número de referencia 11; por lo tanto en el programa fuente, cuando se graben resultados en ese archivo, se codificará WRITE11. Se menciona el DCB con el cual se grabará el disco auxiliar.

Modelo 139 . Ejecución de un programa en tarjetas perforadas en lenguaje simbólico FORTRAN, cuyos resultados son tres archivos de salida distintos, que se graban en la biblioteca como miembros independientes. El problema se puede resolver en dos pasos: en el PAS01 el procedimiento FTG1CLG ejecuta el programa grabando los resultados, con los nombres RESULTA1, RESULTA2 y RESULTA3, en un disco auxiliar; en el PAS02 el utilitario IEBUPDTE transfiere esos archivos del disco auxiliar a la biblioteca SIMBOLIC.CNEA.

```
//..... JOB ....
//PAS01 EXEC PROC=FTG1CLG
//FORT.SYSIN DD *
  Aquí tarjetas perforadas en FORTRAN
/*
//GO.FT15F001 DD DSN=RESULTA1,UNIT=WORKDISK,DISP=(NEW,PASS),
//  SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//GO.FT12F001 DD DSN=RESULTA2,VOL=REF=*.GO.FT15F001,DISP=(NEW,PASS),
//  SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//GO.FT13F001 DD DSN=RESULTA3,VOL=REF=*.GO.FT15F001,DISP=(NEW,PASS),
//  SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//GO.SYSIN DD *
  Aquí tarjetas perforadas de DATOS
/*
//PAS02 EXEC PGM=IEBUPDTE,PARM=MOD
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA,DISP=SHR
//SYSUT2 DD DSN=SIMBOLIC.CNEA,DISP=OLD
//SYSIN DD *
./  ADD NAME=RESULTA1
/*
//      DD DSN=RESULTA1,DISP=(OLD,DELETE)
//      DD *
./  ADD NAME=RESULTA2
/*
//      DD DSN=RESULTA2,DISP=(OLD,DELETE)
//      DD *
./  ADD NAME=RESULTA3
/*
//      DD DSN=RESULTA3,DISP=(OLD,DELETE)
//      DD *
./  ENDUP
/*
//
```

NOTA: Los archivos de salida de datos están definidos en las tres primeras sentencias GO del PAS01, en donde se les asignan los números de referencia 15, 12 y 13; por lo tanto en el programa fuente, cuando se graben resultados en esos archivos, se codificará WRITE15, WRITE12 y WRITE13 respectivamente. Se menciona el DCB con el cual se grabará el disco auxiliar.

Modelo 140 . Ejecución de un programa en tarjetas perforadas en lenguaje absoluto de FORTRAN, cuyos resultados son cuatro archivos de salida distintos, que se graban en la biblioteca como miembros independientes. El problema se puede resolver en dos pasos: en el PAS01 el procedimiento FTG1LG ejecuta el programa grabando los resultados, con los nombres ARCHIDR1, ARCHIDR2, ARCHIDR3 y ARCHIDR4, en un disco auxiliar; en el PAS02 el utilitario IEBUPDTE transfiere esos archivos del disco auxiliar a la biblioteca SIMBOLIC.CNEA.

```
//..... JOB ....
//PAS01 EXEC PROC=FTG1LG
//LKED.SYSIN DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas en absoluto de FORTRAN
/*
//GO.FT09F001 DD DSN=ARCHIDR1,UNIT=WORKDISK,DISP=(NEW,PASS),
//  SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//GO.FT12F001 DD DSN=ARCHIDR2,VOL=REF=*.GO.FT09F001,DISP=(NEW,PASS),
//  SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//GO.FT04F001 DD DSN=ARCHIDR3,VOL=REF=*.GO.FT09F001,DISP=(NEW,PASS),
//  SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//GO.FT20F001 DD DSN=ARCHIDR4,VOL=REF=*.GO.FT09F001,DISP=(NEW,PASS),
//  SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//GO.SYSIN DD *
  Aquí tarjetas perforadas de DATOS
/*
//PAS02 EXEC PGM=IEBUPDTE,PARM=MOD
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA,DISP=SHR
//SYSUT2 DD DSN=SIMBOLIC.CNEA,DISP=OLD
//SYSIN DD *
./  ADD  NAME=ARCHIDR1
/*
//      DD DSN=ARCHIDR1,DISP=(OLD,DELETE)
//      DD *
./  ADD  NAME=ARCHIDR2
/*
//      DD DSN=ARCHIDR2,DISP=(OLD,DELETE)
//      DD *
./  ADD  NAME=ARCHIDR3
/*
//      DD DSN=ARCHIDR3
```

```
//      DD *
./ ADD  NAME=ARCHIDR4
/*
//      DD DSN=ARCHIDR4,DISP=(OLD,DELETE)
//      DD *
./ ENDUP
/*
//
```

NOTA: Los archivos de salida de datos están definidos en las cuatro primeras sentencias GO del PAS01, en donde se les asignan los números de referencia 9, 12, 4 y 20; por lo tanto en el programa fuente que dió origen al lote de tarjetas en absoluto de FORTRAN, se codificó WRITE9, WRITE12, WRITE4 y WRITE20 respectivamente, para grabar resultados en esos archivos. Se menciona el DCB con el cual se grabará el disco auxiliar.

Modelo 141 . Ejecución de un programa en lenguaje simbólico FORTRAN de nombre ANALOGIA, residente en el rótulo 46 de una cinta magnética, cuyos resultados son tres archivos de salida distintos, que se graban en la biblioteca como miembros independientes. La cinta lleva el número de carrete 80044 y tiene etiquetas modelo IBM. El problema se puede resolver en dos pasos: en el PAS01 el procedimiento FTG1CLG ejecuta el programa grabando los resultados, con los nombres ANALOG65, ANALOG66 y ANALOG73, en un disco auxiliar; en el PAS02 el utilitario IEBUPDTE, transfiere esos archivos del disco auxiliar a la biblioteca SIMBOLIC.CNEA.

```
//..... JOB ....
//PAS01 EXEC PROC=FTG1CLG
//FORT.SYSIN DD DSN=ANALOGIA,UNIT=TAPE,VOL=SER=T80044,LABEL=(46,SL),
// DISP=(OLD,KEEP)
/*
//GO.FT15F001 DD DSN=ANALOG65,UNIT=WORKDISK,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//GO.FT11F001 DD DSN=ANALOG66,VOL=REF=*.GO.FT15F001,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//GO.FT14F001 DD DSN=ANALOG73,VOL=REF=*.GO.FT15F001,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//GO.SYSIN DD *
Aquí tarjetas perforadas de DATOS
/*
//PAS02 EXEC PGM=IEBUPDTE,PARM=MOD
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA,DISP=SHR
//SYSUT2 DD DSN=SIMBOLIC.CNEA,DISP=OLD
```

```
//SYSIN DD *
./ ADD NAME=ANALOG65
/*
// DD DSN=ANALOG65,DISP=(OLD,DELETE)
// DD *
./ ADD NAME=ANALOG66
/*
// DD DSN=ANALOG66,DISP=(OLD,DELETE)
// DD *
./ ADD NAME=ANALOG73
/*
// DD DSN=ANALOG73,DISP=(OLD,DELETE)
// DD *
./ ENDUP
/*
//
```

NOTA: Los archivos de salida de datos están definidos en las tres primeras sentencias GO del PAS01, en donde se les asignan los números de referencia 15, 11 y 14; por lo tanto en el programa fuente, cuando se graben resultados en esos archivos, se codificará WRITE15, WRITE11 y WRITE14 respectivamente. Se menciona el DCB con el cual se grabará el disco auxiliar.

Modelo 142 . Ejecución de un programa en lenguaje absoluto de FORTRAN de nombre MEDIDAS, residente en el rótulo 2 de una cinta magnética, cuyos resultados son dos archivos de salida distintos, que se graban en la biblioteca como miembros independientes. La cinta lleva el número de carrete 32121 y tiene etiquetas modelo IBM. El problema se puede resolver en dos pasos: en el PAS01 el procedimiento FTG1LG ejecuta el programa grabando los resultados, con los nombres RESMED28 y RESMED29, en un disco auxiliar; en el PAS02 el utilitario IEBUPDTE transfiere esos archivos del disco auxiliar a la biblioteca SIMBOLIC.CNEA.

```
//..... JOB ....
//PAS01 EXEC PROC=FTG1LG
//LKED.SYSIN DD DSN=MEDIDAS,UNIT=TAPE,VOL=SER=T32121,LABEL=(2,SL),
// DISP=(OLD,KEEP)
/*
//GO.FT12F001 DD DSN=RESMED28,UNIT=WORKDISK,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//GO.FT15F001 DD DSN=RESMED29,VOL=REF=*.GO.FT12F001,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//GO.SYSIN DD *
Aquí tarjetas perforadas de DATOS
```



```

/*
//PASO2 EXEC PGM=IEBUPDTE,PARM=MOD
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA,DISP=SHR
//SYSUT2 DD DSN=SIMBOLIC.CNEA,DISP=OLD
//SYSIN DD *
./ ADD NAME=RESMED28
/*
// DD DSN=RESMED28,DISP=(OLD,DELETE)
// DD *
./ ADD NAME=RESMED29
/*
// DD DSN=RESMED29,DISP=(OLD,DELETE)
// DD *
./ ENDUP
/*
//

```

NOTA: Los archivos de salida de datos están definidos en las dos primeras sentencias GO del PASO1, en donde se les asignan los números de referencia 12 y 15; por lo tanto en el programa fuente que dió origen al módulo grabado en la cinta, se codificó WRITE12 y WRITE15 respectivamente, para grabar resultados en esos archivos. Se menciona el DCB con el cual se grabará el disco auxiliar.

Modelo 143 . Ejecución de un programa residente en la biblioteca con el nombre GEOM, cuyos resultados son cuatro archivos de salida distintos que se graban en la propia biblioteca, como miembros independientes. El problema se puede resolver en dos pasos: en el PASO1 el procedimiento FTG1CLG ejecuta el programa grabando los resultados, con los nombres GEOMRESA, GEOMRESB, GEOMRESC y GEOMRESD, en un disco auxiliar; en el PASO2 el utilitario IEBUPDTE transfiere esos archivos del disco auxiliar a la biblioteca.

```

//..... JOB ....
//PASO1 EXEC PROC=FTG1CLG
//FORT.SYSIN DD DSN=SIMBOLIC.CNEA(GEOM),DISP=SHR
/*
//GO.FT17F001 DD DSN=GEOMRESA,UNIT=WORKDISK,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//GO.FT12F001 DD DSN=GEOMRESB,VOL=REF=*.GO.FT17F001,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//GO.FT15F001 DD DSN=GEOMRESC,VOL=REF=*.GO.FT17F001,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//GO.FT16F001 DD DSN=GEOMRESD,VOL=REF=*.GO.FT17F001,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//GO.SYSIN DD *

```

```

Aquí tarjetas perforadas de DATOS
/*
//PASO2 EXEC PGM=IEBUPDTE,PARM=MOD
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA,DISP=SHR
//SYSUT2 DD DSN=SIMBOLIC.CNEA,DISP=OLD
//SYSIN DD *
./  ADD  NAME=GEOMRESA
/*
//      DD DSN=GEOMRESA,DISP=(OLD,DELETE)
//      DD *
./  ADD  NAME=GEOMRESB
/*
//      DD DSN=GEOMRESB,DISP=(OLD,DELETE)
//      DD *
./  ADD  NAME=GEOMRESC
/*
//      DD DSN=GEOMRESC,DISP=(OLD,DELETE)
//      DD *
./  ADD  NAME=GEOMRES D
/*
//      DD DSN=GEOMRES D,DISP=(OLD,DELETE)
//      DD *
./  ENDUP
/*
//

```

NOTA: Los archivos de salida de datos están definidos en las cuatro primeras sentencias GO del PASO1, en donde se les asignan los números de referencia 17, 12, 15 y 16; por lo tanto en el programa fuente, cuando se graben resultados en esos archivos, se codificará WRITE17, WRITE12, WRITE15 y WRITE16 respectivamente. Se menciona el DCB con el cual se grabará el disco auxiliar.

PUNTO 3.

Ejecución de programas en FORTRAN que utilizan archivos de entrada de datos residentes en r tulos consecutivos o no consecutivos de una cinta magn tica y simult neamente graban archivos de salida de datos en una biblioteca particionada y catalogada de programas en lenguaje simb lico, de nombre SIMBOLIC.CNEA.

Modelo 144 . Ejecuci n de un programa en tarjetas perforadas en lenguaje simb lico FORTRAN, cuyos datos de entrada constituyen un archivo que reside, con el nombre DATOS615, en el r tulo 37 de una cinta magn tica y cuyos resultados se graban en la biblioteca, como un miembro independiente. La cinta lleva el n mero de carrete 12208 y tiene etiquetas modelo IBM. El problema se puede resolver en dos pasos: en el PAS01 el procedimiento FTG1CLG ejecuta el programa obteniendo los datos de entrada de la cinta y grabando los resultados, con el nombre RESULT28, en un disco auxiliar; en el PAS02 el utilitario IEBUPDTE transfiere este  ltimo archivo del disco a la biblioteca.

```
//..... JOB ....
//PAS01 EXEC PROC=FTG1CLG
//FORT.SYSIN DD *
  Aqu  tarjetas perforadas en FORTRAN
/*
//GO.FT10F001 DD DSN=DATOS615,UNIT=TAPE,VOL=SER=T12208,LABEL=(37,SL),
//  DISP=(OLD,KEEP)
//GO.FT19F001 DD DSN=RESULT28,UNIT=WORKDISK,DISP=(NEW,PASS),
//  SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
/*
//PAS02 EXEC PGM=IEBUPDTE,PARM=NEW
//SYSPRINT DD SYSOUT=A
//SYSUT2 DD DSN=SIMBOLIC.CNEA,DISP=OLD
//SYSIN DD *
./  ADD  NAME=RESULT28
//      DD DSN=RESULT28,DISP=(OLD,DELETE)
./  ENDUP
/*
//
```

NOTA: El archivo de entrada de datos est  definido en la primera sentencia GO del PAS01, en donde se le asigna el n mero de referencia 10 y el archivo de salida de datos est  definido en la segunda sentencia GO, en donde se le asigna el n mero

de referencia 19. Por lo tanto en el programa fuente, cuando se lean datos del primer archivo se codificará READ10 y cuando se graben resultados en el segundo archivo se codificará WRITE19. Se menciona el DCB con el cual se grabará el disco auxiliar.

Modelo 145 . Ejecución de un programa en tarjetas perforadas en lenguaje absoluto de FORTRAN, cuyos datos de entrada constituyen un archivo que reside, con el nombre DATINPUT en el rótulo 4 de una cinta magnética y cuyos resultados se graban en la biblioteca, como un miembro independiente. La cinta lleva el número de carrete 53697 y tiene etiquetas modelo IBM. El problema se puede resolver en dos pasos: en el PASO1 el procedimiento FTG1LG ejecuta el programa obteniendo los datos de entrada de la cinta y grabando los resultados, con el nombre RESULOUT, en un disco auxiliar; en el PASO2 el utilitario IEBUPDTE transfiere este último archivo del disco a la biblioteca SIMBOLIC.CNEA.

```
//..... JOB ....
//PASO1 EXEC PROC=FTG1LG
//LKED.SYSIN DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas en absoluto de FORTRAN
/*
//GO.FT02F001 DD DSN=DATINPUT,UNIT=TAPE,VOL=SER=T53697,LABEL=(4,SL),
//  DISP=(OLD,KEEP)
//GO.FT09F001 DD DSN=RESULOUT,UNIT=WORKDISK,DISP=(NEW,PASS),
//  SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
/*
//PASO2 EXEC PGM=IEBUPDTE,PARM=NEW
//SYSPRINT DD SYSOUT=A
//SYSUT2 DD DSN=SIMBOLIC.CNEA,DISP=OLD
//SYSIN DD *
./  ADD  NAME=RESULOUT
./      DD DSN=RESULOUT,DISP=(OLD,DELETE)
./  ENDUP
/*
//
```

NOTA: El archivo de entrada de datos está definido en la primera sentencia GO del PASO1, en donde se le asigna el número de referencia 2 y el archivo de salida de datos está definido en la segunda sentencia GO, en donde se le asigna el número de referencia 9. Por lo tanto en el programa fuente que dió origen al lote de

tarjetas en absoluto de FORTRAN, se codificó READ2 para leer datos del primer archivo y WRITE9 para grabar resultados en el segundo archivo. Se menciona el DCB con el cual se grabará el disco auxiliar.

Modelo 146 . Ejecución de un programa en lenguaje simbólico FORTRAN de nombre QUATY, grabado en el rótulo 15 de una cinta magnética, cuyos datos de entrada constituyen un archivo que reside, con el nombre DATOSQUA, en el rótulo 28 de otra cinta y cuyos resultados se graban en la biblioteca, como un miembro independiente. Las cintas llevan respectivamente los números de carrete 32661 y 32662 y tienen etiquetas modelo IBM. El problema se puede resolver en dos pasos: en el PASO1 el procedimiento FTG1CLG ejecuta el programa QUATY obteniendo los datos de entrada de la cinta 32662 y grabando los resultados, con el nombre SALIDA15, en un disco auxiliar; en el PASO2 el utilitario IEBUPDTE transfiere este último archivo del disco a la biblioteca SIMBOLIC.CNEA.

```
//..... JOB ....
//PASO1 EXEC PROC=FTG1CLG
//FORT.SYSIN DD DSN=QUATY,UNIT=TAPE,VOL=SER=T32661,LABEL=(15,SL),
//  DISP=(OLD,KEEP)
/*
//GO.FT01F001 DD DSN=DATOSQUA,UNIT=TAPE,VOL=SER=T32662,LABEL=(28,SL),
//  DISP=(OLD,KEEP)
//GO.FT08F001 DD DSN=SALIDA15,UNIT=WORKDISK,DISP=(NEW,PASS),
//  SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
/*
//PASO2 EXEC PGM=IEBUPDTE,PARM=NEW
//SYSPRINT DD SYSOUT=A
//SYSUT2 DD DSN=SIMBOLIC.CNEA,DISP=OLD
//SYSIN DD *
./  ADD  NAME=SALIDA15
//      DD DSN=SALIDA15
./  ENDUP
/*
//
```

NOTA: El archivo de entrada de datos está definido en la primera sentencia GO del PASO1, en donde se le asigna el número de referencia 1 y el archivo de salida de datos está definido en la segunda sentencia GO, en donde se le asigna el número de referencia 8. Por lo tanto en el programa fuente, cuando se lean datos del primer archivo se codificará READ1 y cuando se graben resultados en el segundo

archivo se codificará WRITE8. Se menciona el DCB con el cual se grabará el disco auxiliar.

Modelo 147 . Ejecución de un programa en lenguaje absoluto de FORTRAN de nombre FULLEN, grabado en el rótulo 21 de una cinta magnética, cuyos datos de entrada constituyen un archivo que reside, con el nombre DATOFULL, en el rótulo 14 de otra cinta y cuyos resultados se graban en la biblioteca, como un miembro independiente. Las cintas llevan respectivamente los números de carrete 01003 y 01005 y tienen etiquetas modelo IBM. El problema se puede resolver en dos pasos: en el PASO1 el procedimiento FTG1LG ejecuta el programa FULLEN obteniendo los datos de entrada de la cinta 01005 y grabando los resultados, con el nombre RESFULL, en un disco auxiliar; en el PASO2 el utilitario IEBUPDTE transfiere este último archivo del disco a la biblioteca SIMBOLIC.CNEA.

```
//..... JOB ....
//PASO1 EXEC PROC=FTG1LG
//LKED.SYSIN DD DSN=FULLEN,UNIT=TAPE,VOL=SER=T01003,LABEL=(21,SL),
// DISP=(OLD,KEEP)
/*
//GO.FT15F001 DD DSN=DATOFULL,UNIT=TAPE,VOL=SER=T01005,LABEL=(14,SL),
// DISP=(OLD,KEEP)
//GO.FT12F001 DD DSN=RESFULL,UNIT=WORKDISK,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
/*
//PASO2 EXEC PGM=IEBUPDTE,PARM=NEW
//SYSPRINT DD SYSOUT=A
//SYSUT2 DD DSN=SIMBOLIC.CNEA,DISP=OLD
//SYSIN DD *
./ ADD NAME=RESFULL
// DD DSN=RESFULL,DISP=(OLD,DELETE)
./ ENDUP
/*
//
```

NOTA: El archivo de entrada de datos está definido en la primera sentencia GO del PASO1, en donde se le asigna el número de referencia 15 y el archivo de salida de datos está definido en la segunda sentencia GO, en donde se le asigna el número de referencia 12. Por lo tanto en el programa fuente que dió origen al módulo grabado en la cinta, se codificó READ15 para leer datos del primer archivo y

WRITE12 para grabar resultados en el segundo archivo. Se menciona el DCB con el cual se grabará el disco auxiliar.

Modelo 148 . Ejecución de un programa grabado en la biblioteca con el nombre FOULAR, cuyos datos de entrada constituyen un archivo que reside, con el nombre DATOFOUL, en el rótulo 21 de una cinta magnética y cuyos resultados se graban en la propia biblioteca, como un miembro independiente. La cinta lleva el número de carrete 30033 y tiene etiquetas modelo IBM. El problema se puede resolver en dos pasos: en el PASO1 el procedimiento FTG1CLG ejecuta el programa obteniendo los datos de entrada de la cinta y grabando los resultados, con el nombre FOULOUT, en un disco auxiliar; en el PASO2 el utilitario IEBUPDTE transfiere este último archivo del disco a la biblioteca SIMBOLIC.CNEA.

```
//..... JOB ....
//PASO1 EXEC PROC=FTG1CLG
//FORT.SYSIN DD DSN=SIMBOLIC.CNEA(FOULAR),DISP=SHR
/*
//GO.FT19F001 DD DSN=DATOFOUL,UNIT=TAPE,VOL=SER=T30033,LABEL=(21,SL),
// DISP=(OLD,KEEP)
//GO.FT13F001 DD DSN=FOULOUT,UNIT=WORKDISK,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
/*
//PASO2 EXEC PGM=IEBUPDTE,PARM=NEW
//SYSPRINT DD SYSOUT=A
//SYSUT2 DD DSN=SIMBOLIC.CNEA,DISP=OLD
//SYSIN DD *
./ ADD NAME=FOULOUT
// DD DSN=FOULOUT,DISP=(OLD,DELETE)
./ ENDUP
/*
//
```

NOTA: El archivo de entrada de datos está definido en la primera sentencia GO del PASO1, en donde se le asigna el número de referencia 19 y el archivo de salida de datos está definido en la segunda sentencia GO, en donde se le asigna el número de referencia 13. Por lo tanto en el programa fuente, cuando se lean datos del primer archivo se codificará READ19 y cuando se graben resultados en el segundo archivo se codificará WRITE13. Se menciona el DCB con el cual se grabará el disco auxiliar.

Modelo 149 . Ejecución de un programa en tarjetas perforadas en lenguaje simbólico FORTRAN, cuyos datos de entrada son tres archivos distintos que residen, con los nombres DAT125, DAT126 y DAT127 en los r  tulos 21, 23 y 28 de una cinta magn  tica y cuyos resultados son dos archivos de salida distintos que se graban en la biblioteca, como miembros independientes. La cinta lleva el n  mero de carrete 71315 y tiene etiquetas modelo IBM. El problema se puede resolver en dos pasos: en el PAS01 el procedimiento FTG1CLG ejecuta el programa obteniendo los datos de entrada de la cinta y grabando los resultados, con los nombres SALIDA11 y SALIDA12, en un mismo disco auxiliar; en el PAS02 el utilitario IEBUPDTE transfiere estos   ltimos archivos del disco a la biblioteca SIMBOLIC.CNEA.

```
//..... JOB ....
//PAS01 EXEC PROC=FTG1CLG
//FORT.SYSIN DD *
  Aqu   tarjetas perforadas en FORTRAN
/*
//GO.FT16F001 DD DSN=DAT125,UNIT=TAPE,VOL=SER=T71315,LABEL=(21,SL),
// DISP=(OLD,PASS)
//GO.FT16F002 DD DSN=DAT126,UNIT=TAPE,VOL=REF=*.GO.FT16F001,
// LABEL=(23,SL)
//GO.FT16F003 DD DSN=DAT127,UNIT=TAPE,VOL=REF=*.GO.FT16F001,
// LABEL=(28,SL),DISP=(OLD,KEEP)
//GO.FT18F001 DD DSN=SALIDA11,UNIT=WORKDISK,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//GO.FT20F001 DD DSN=SALIDA12,VOL=REF=*.GO.FT18F001,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
/*
//PAS02 EXEC PGM=IEBUPDTE,PARM=MOD
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA,DISP=SHR
//SYSUT2 DD DSN=SIMBOLIC.CNEA,DISP=OLD
//SYSIN DD *
./ ADD NAME=SALIDA11
/*
// DD DSN=SALIDA11,DISP=(OLD,DELETE)
// DD *
./ ADD NAME=SALIDA12
/*
// DD DSN=SALIDA12,DISP=(OLD,DELETE)
// DD *
./ ENDUP
/*
//
```

NOTA: Los archivos de entrada de datos est  n definidos en las tres primeras sentencias GO del PAS01, en donde se les asigna el mismo n  mero de referencia 16 y

los números de secuencia 1, 2 y 3 respectivamente. Por lo tanto en el programa fuente, cuando se lean datos de esos archivos se codificará una sentencia READ de este tipo: READ(16,nº de formato,END=dirección), en donde la "dirección" significa la marca de la sentencia a la cual deberá saltar el programa para continuar la ejecución, cada vez que llega al final de uno de los archivos. En la sentencia DD en que se define el último archivo se codifica DISP=(OLD,KEEP) para que la cinta sea rebobinada. A su vez los archivos de salida están definidos en la cuarta y quinta sentencia GO, en donde se les asignan los números de referencia 18 y 20. Por consiguiente en el programa fuente, cuando se graben resultados en esos archivos se codificará WRITE18 y WRITE20 respectivamente. Se menciona el DCB con el cual se grabará el disco auxiliar.

Modelo 150. Ejecución de un programa en tarjetas perforadas en lenguaje absoluto de FORTRAN, cuyos datos de entrada son dos archivos distintos que residen, con los nombres INPUT42A e INPUT48A en los rótulos 3 y 15 de una cinta magnética y cuyos resultados son 3 archivos de salida distintos que se graban en la biblioteca, como miembros independientes. La cinta lleva el número de carrete 60228 y tiene etiquetas modelo IBM. El problema se puede resolver en dos pasos: en el PASO1 el procedimiento FTGILG ejecuta el programa obteniendo los datos de entrada de la cinta y grabando los resultados, con los nombres OUTPUT89, OUTPUT97 y OUTPUT99, en un mismo disco auxiliar; en el PASO2 el utilitario IEBUPDTE transfiere estos últimos archivos del disco a la biblioteca SIMBOLIC.CNEA.

```
//..... JOB ....
//PASO1 EXEC PROC=FTGILG
//LKED.SYSIN DD *,DCB=BLKSIZE=80
Aquí tarjetas perforadas en absoluto de FORTRAN
/*
//GO.FT14F001 DD DSN=INPUT42A,UNIT=TAPE,VOL=SER=T60228,LABEL=(3,SL),
// DISP=(OLD,PASS)
//GO.FT14F002 DD DSN=INPUT48A,UNIT=TAPE,VOL=REF=*.GO.FT14F001,
// LABEL=(15,SL),DISP=(OLD,KEEP)
//GO.FT11F001 DD DSN=OUTPUT89,UNIT=WORKDISK,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//GO.FT03F001 DD DSN=OUTPUT97,VOL=REF=*.GO.FT11F001,DISP=(,PASS),
// SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
```

```

//GO.FT08F001 DD DSN=OUTPUT99,VOL=REF=*.GO.FT11F001,DISP=(,PASS),
// SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
/*
//PASO2 EXEC PGM=IEBUPDTE,PARM=MOD
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA,DISP=SHR
//SYSUT2 DD DSN=SIMBOLIC.CNEA,DISP=OLD
//SYSIN DD *
./ ADD NAME=OUTPUT89
/*
// DD DSN=OUTPUT89,DISP=(OLD,DELETE)
// DD *
./ ADD NAME=OUTPUT97
/*
// DD DSN=OUTPUT97,DISP=(OLD,DELETE)
// DD *
./ ADD NAME=OUTPUT99
/*
// DD DSN=OUTOUT99,DISP=(OLD,DELETE)
// DD *
./ ENDUP
/*
//

```

NOTA: Los archivos de entrada de datos están definidos en las dos primeras sentencias GO del PASO1, en donde se les asigna el mismo número de referencia 14 y los números de secuencia 1 y 2 respectivamente. Por lo tanto en el programa fuente que dió origen al lote de tarjetas en absoluto de FORTRAN, para leer datos de esos archivos se codificó una sentencia READ de este tipo:

READ(14,n° de formato,END=dirección), en donde la "dirección" significa la marca de la sentencia a la cual deberá saltar el programa para continuar la ejecución, cada vez que llega al final de uno de los archivos. En la sentencia DD en que se define el último archivo se codifica DISP=(OLD,KEEP) para que la cinta sea rebobinada. A su vez los archivos de salida están definidos en la tercera, cuarta y quinta sentencia GO, en donde se les asignan los números de referencia 11, 3 y 8. Por consiguiente en el programa fuente original, se codificó WRITE11, WRITE3 y WRITE8 respectivamente, para grabar resultados en esos archivos. Se menciona el DCB con el cual se grabará el disco auxiliar.

Modelo 151. Ejecución de un programa en lenguaje simbólico FORTRAN de nombre CARTAL, grabado en el rótulo 44 de una cinta magnética, cuyos datos de entrada

son tres archivos distintos que residen, con los nombres DATOCAR1, DATOCAR2 y DATOCAR3, en los r tulos 23, 31 y 40 de otra cinta y cuyos resultados son tres archivos de salida distintos, que se graban en la biblioteca, como miembros independientes. Las cintas llevan respectivamente los n meros de carrete 22154 y 22157 y tienen etiquetas modelo IBM. El problema se puede resolver en dos pasos: en el PASO1 el procedimiento FTG1CLG ejecuta el programa CARTAL obteniendo los datos de entrada de la cinta 22157 y grabando los resultados con los nombres RESCAR1, RESCAR2 y RESCAR3, en un mismo disco auxiliar; en el PASO2 el utilitario IEBUPDTE transfiere estos  ltimos archivos del disco a la biblioteca SIMBOLIC. CNEA.

```
//..... JOB ....
//PASO1 EXEC PROC=FTG1CLG
//FORT.SYSIN DD DSN=CARTAL,UNIT=TAPE,VOL=SER=T22154,LABEL=(44,SL),
// DISP=(OLD,KEEP)
/*
//GO.FT12F001 DD DSN=DATOCAR1,UNIT=TAPE,VOL=SER=T22157,LABEL=(23,SL),
// DISP=(OLD,PASS)
//GO.FT12F002 DD DSN=DATOCAR2,UNIT=TAPE,VOL=REF=*.GO.FT12F001,
// LABEL=(31,SL)
//GO.FT12F003 DD DSN=DATOCAR3,UNIT=TAPE,VOL=REF=*.GO.FT12F001,
// DISP=(OLD,KEEP),LABEL=(40,SL)
//GO.FT05F001 DD DSN=RESCAR1,UNIT=WORKDISK,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//GO.FT04F001 DD DSN=RESCAR2,VOL=REF=*.GO.FT05F001,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//GO.FT13F001 DD DSN=RESCAR3,VOL=REF=*.GO.FT05F001,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
/*
//PASO2 EXEC PGM=IEBUPDTE,PARM=MOD
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA,DISP=SHR
//SYSUT2 DD DSN=SIMBOLIC.CNEA,DISP=OLD
//SYSIN DD *
./ ADD NAME=RESCAR1
/*
// DD DSN=RESCAR1,DISP=(OLD,DELETE)
// DD *
./ ADD NAME=RESCAR2
/*
// DD DSN=RESCAR2,DISP=(OLD,DELETE)
// DD *
./ ADD NAME=RESCAR3
/*
// DD DSN=RESCAR3,DISP=(OLD,DELETE)
// DD *
./ ENDUP
```

```
/*
//
```

NOTA: Los archivos de entrada de datos están definidos en las tres primeras sentencias GO del PAS01, en donde se les asigna el mismo número de referencia 12 y los números de secuencia 1, 2 y 3 respectivamente. Por lo tanto en el programa fuente, cuando se lean datos de esos archivos se codificará una sentencia READ de este tipo: READ(12,n° de formato,END=dirección), en donde la dirección significa la marca de la sentencia a la cual deberá saltar el programa para continuar la ejecución, cada vez que llega al final de uno de los archivos. En la sentencia DD en que se define el último archivo se codifica DISP=(OLD,KEEP) para que la cinta sea rebobinada. A su vez los archivos de salida están definidos en la cuarta, quinta y sexta sentencia GO en donde se les asignan los números de referencia 5, 4 y 13. Por consiguiente en el programa fuente, cuando se graben resultados en esos archivos se codificará WRITE5, WRITE4 y WRITE13 respectivamente. Se menciona el DCB con el cual se grabará el disco auxiliar.

Modelo 152. Ejecución de un programa en lenguaje absoluto de FORTRAN de nombre TEMA, grabado en el rótulo 17 de una cinta magnética, cuyos datos de entrada son dos archivos distintos que residen, con los nombres DATEMA1 y DATEMA2, en los rótulos 15 y 18 de otra cinta y cuyos resultados son dos archivos de salida distintos, que se graban en la biblioteca, como miembros independientes. Las cintas llevan respectivamente los números de carrete 02022 y 02030 y tienen etiquetas modelo IBM. El problema se puede resolver en dos pasos: en el PAS01 el procedimiento FTG1LG ejecuta el programa TEMA obteniendo los datos de entrada de la cinta 02030 y grabando los resultados, con los nombres RESTEM1 y RESTEM2, en un mismo disco auxiliar; en el PAS02 el utilitario IEBUPDTE transfiere estos últimos archivos del disco a la biblioteca SIMBOLIC.CNEA.

```
//..... JOB ....
//PAS01 EXEC PROC=FTG1LG
//LKED.SYSIN DD DSN=TEMA,UNIT=TAPE,VOL=SER=T02022,LABEL=(17,SL),
// DISP=(OLD,KEEP)
```

```

/*
//GO.FT20F001 DD DSN=DATEMA1,UNIT=TAPE,VOL=SER=T02030,LABEL=(15,SL),
// DISP=(OLD,PASS)
//GO.FT20F002 DD DSN=DATEMA2,UNIT=TAPE,VOL=REF=*.GO.FT20F001,
// LABEL=(18,SL),DISP=(OLD,KEEP)
//GO.FT13F001 DD DSN=RESTEM1,UNIT=WORKDISK,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//GO.FT19F001 DD DSN=RESTEM2,VOL=REF=*.GO.FT13F001,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
/*
//PAS02 EXEC PGM=IEBUPDTE,PARM=MOD
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA,DISP=SHR
//SYSUT2 DD DSN=SIMBOLIC.CNEA,DISP=OLD
//SYSIN DD *
./ ADD NAME=RESTEM1
/*
// DD DSN=RESTEM1,DISP=(OLD,DELETE)
// DD *
./ ADD NAME=RESTEM2
/*
// DD DSN=RESTEM2,DISP=(OLD,DELETE)
// DD *
./ ENDUP
/*
//

```

NOTA: Los archivos de entrada de datos están definidos en las dos primeras sentencias GO del PAS01, en donde se les asigna el mismo número de referencia 20 y los números de secuencia 1 y 2 respectivamente. Por lo tanto en el programa fuente que dió origen al módulo grabado en la cinta, para leer datos de esos archivos se codificó una sentencia READ de este tipo: READ(20,n°de formato,END=dirección), en donde la "dirección" significa la marca de la sentencia a la cual deberá saltar el programa para continuar la ejecución, cada vez que llega al final de uno de los archivos. En la sentencia DD en que se define el último archivo se codifica DISP=(OLD,KEEP) para que la cinta sea rebobinada. A su vez los archivos de salida están definidos en la tercera y cuarta sentencia GO, en donde se les asignan los números de referencia 13 y 19. Por consiguiente en el programa fuente original, se codificó WRITE13 y WRITE19 respectivamente, para grabar resultados en esos archivos. Se menciona el DCB con el cual se grabará el disco auxiliar.

Modelo 153 . Ejecución de un programa grabado en la biblioteca con el nombre NIVEL4, cuyos datos de entrada son tres archivos distintos que residen, con los

nombres DATNIV41, DATNIV42 y DATNIV43 en los rótulos 11,15 y 18 de una cinta magnética y cuyos resultados son dos archivos de salida distintos que se graban en la propia biblioteca, como miembros independientes. La cinta lleva el número de carrete 43310 y tiene etiquetas modelo IBM. El problema se puede resolver en dos pasos: en el PASO1 el procedimiento FTG1CLG ejecuta el programa obteniendo los datos de entrada de la cinta y grabando los resultados, con los nombres RESNIV67 y RESNIV69, en un mismo disco auxiliar; en el PASO2 el utilitario IEBUPDTE transfiere estos últimos archivos del disco a la biblioteca SIMBOLIC.CNEA.

```
//..... JOB ....
//PASO1 EXEC PROC=FTG1CLG
//FORT.SYSIN DD DSN=SIMBOLIC.CNEA(NIVEL4),DISP=SHR
/*
//GO.FT02F001 DD DSN=DATNIV41,UNIT=TAPE,VOL=SER=T43310,LABEL=(11,SL),
// DISP=(OLD,PASS)
//GO.FT02F002 DD DSN=DATNIV42,UNIT=TAPE,VOL=REF=*.GO.FT02F001,
// LABEL=(15,SL)
//GO.FT02F003 DD DSN=DATNIV43,UNIT=TAPE,VOL=REF=*.GO.FT02F001,
// LABEL=(18,SL),DISP=(OLD,KEEP)
//GO.FT01F001 DD DSN=RESNIV67,UNIT=WORKDISK,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//GO.FT09F001 DD DSN=RESNIV69,VOL=REF=*.GO.FT01F001,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
/*
//PASO2 EXEC PGM=IEBUPDTE,PARM=MOD
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA,DISP=SHR
//SYSUT2 DD DSN=SIMBOLIC.CNEA,DISP=OLD
//SYSIN DD *
./ ADD NAME=RESNIV67
/*
// DD DSN=RESNIV67,DISP=(OLD,DELETE)
// DD *
./ ADD NAME=RESNIV69
/*
// DD DSN=RESNIV69,DISP=(OLD,DELETE)
// DD *
./ ENDUP
/*
//
```

NOTA: Los archivos de entrada de datos están definidos en las tres primeras sentencias GO del PASO1, en donde se les asigna el mismo número de referencia 2 y los números de secuencia 1, 2 y 3 respectivamente. Por lo tanto en el programa

fuelle, cuando se lean datos de esos archivos se codificará una sentencia READ de este tipo: READ(2,n°de formato,END=dirección), en donde la "dirección" significa la marca de la sentencia a la cual deberá saltar el programa para continuar la ejecución, cada vez que llega al final de uno de los archivos. En la sentencia DD en que se define el último archivo se codifica DISP=(OLD,KEEP) para que la cinta sea rebobinada. A su vez los archivos de salida están definidos en la cuarta y quinta sentencia GO, en donde se les asignan los números de referencia 1 y 9. Por consiguiente en el programa fuente, cuando se graben resultados en esos archivos se codificará WRITE1 y WRITE9 respectivamente. Se menciona el DCB con el cual se grabará el disco auxiliar.

PUNTO 4.

Ejecución de programas en FORTRAN que utilizan archivos de entrada de datos residentes en r tulos consecutivos o no consecutivos de sendas cintas magn ticas y simult neamente graban archivos de salida de datos en una biblioteca particionada y catalogada de programas en lenguaje simb lico, de nombre SIMBOLIC.CNEA.

Modelo 154 . Ejecuci n de un programa en tarjetas perforadas en lenguaje simb lico FORTRAN, cuyos datos de entrada residen en sendas cintas magn ticas en cada una de las cuales lee un archivo y cuyos resultados se graban en la biblioteca, como un miembro independiente. En el r tulo 2 de la cinta que lleva el n mero de carrete 15963 est  grabado, con el nombre DATPIR1, el primer archivo de entrada y en el r tulo 15 de la cinta que lleva el n mero de carrete 06063 est  grabado, con el nombre DATPIR2, el segundo archivo de entrada. Ambas cintas tienen etiquetas modelo IBM. El problema se puede resolver en dos pasos: en el PASO1 el procedimiento FTG1CLG ejecuta el programa obteniendo los datos de entrada de las cintas y grabando los resultados, con el nombre SALIDA30, en un disco auxiliar; en el PASO2 el utilitario IEBUPDTE transfiere este  ltimo archivo del disco a la biblioteca SIMBOLIC.CNEA.

```
//..... JOB ....
//PASO1 EXEC PROC=FTG1CLG
//FORT.SYSIN DD *
  Aqu  tarjetas perforadas en FORTRAN
/*
//GO.FT11F001 DD DSN=DATPIR1,UNIT=TAPE,VOL=SER=T15963,LABEL=(2,SL),
//  DISP=(OLD,KEEP)
//GO.FT14F001 DD DSN=DATPIR2,UNIT=TAPE,VOL=SER=T06063,LABEL=(15,SL),
//  DISP=(OLD,KEEP)
//GO.FT15F001 DD DSN=SALIDA30,UNIT=WORKDISK,DISP=(NEW,PASS),
//  SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
/*
//PASO2 EXEC PGM=IEBUPDTE,PARM=NEW
//SYSPRINT DD SYSOUT=A
//SYSUT2 DD DSN=SIMBOLIC.CNEA,DISP=OLD
//SYSIN DD *
./  ADD  NAME=SALIDA30
//      DD DSN=SALIDA30,DISP=(OLD,DELETE)
./  ENDUP
/*
//
```


NOTA: Los archivos de entrada de datos están definidos en las dos primeras sentencias GO del PASO1, en donde se les asignan los números de referencia 11 y 14 y el archivo de salida de datos está definido en la tercera sentencia GO, en donde se le asigna el número de referencia 15. Por lo tanto en el programa fuente, cuando se lean datos de esos archivos se codificará READ11 y READ14 respectivamente y cuando se graben resultados en el último archivo se codificará WRITE15. Se menciona el DCB con el cual se grabará el disco auxiliar.

Modelo 155. Ejecución de un programa en tarjetas perforadas en lenguaje absoluto de FORTRAN, cuyos datos de entrada residen en sendas cintas magnéticas en cada una de las cuales lee un archivo y cuyos resultados se graban en la biblioteca, como un miembro independiente. En el rótulo 31 de la cinta que lleva el número de carrete 53089 está grabado, con el nombre DATOSAA, el primer archivo de entrada y en el rótulo 20 de la cinta que lleva el número de carrete 27164 está grabado, con el nombre DATOSBB, el segundo archivo de entrada. Ambas cintas tienen etiquetas modelo IBM. El problema se puede resolver en dos pasos: en el PASO1 el procedimiento FTG1LG ejecuta el programa obteniendo los datos de entrada de las cintas y grabando los resultados, con el nombre SALIDAAB, en un disco auxiliar; en el PASO2 el utilitario IEBUPDTE transfiere este último archivo del disco a la biblioteca SIMBOLIC.CNEA.

```
//..... JOB ....
//PASO1 EXEC PROC=FTG1LG
//LKED.SYSIN DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas en absoluto de FORTRAN
/*
//GO.FT12F001 DD DSN=DATOSAA,UNIT=TAPE,VOL=SER=T53089,LABEL=(31,SL),
//  DISP=(OLD,KEEP)
//GO.FT16F001 DD DSN=DATOSBB,UNIT=TAPE,VOL=SER=T27164,LABEL=(20,SL),
//  DISP=(OLD,KEEP)
//GO.FT03F001 DD DSN=SALIDAAB,UNIT=WORKDISK,DISP=(NEW,PASS),
//  SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
/*
//PASO2 EXEC PGM=IEBUPDTE,PARM=NEW
//SYSPRINT DD SYSOUT=A
//SYSUT2 DD DSN=SIMBOLIC.CNEA,DISP=OLD
//SYSIN DD *
./ ADD NAME=SALIDAAB
```

```
//      DD DSN=SALIDAAB
./  ENDUP
/*
//
```

NOTA: Los archivos de entrada de datos están definidos en las dos primeras sentencias GO del PASO1, en donde se les asignan los números de referencia 12 y 16 y el archivo de salida de datos está definido en la tercera sentencia GO, en donde se le asigna el número de referencia 3. Por lo tanto en el programa fuente que dió origen al lote de tarjetas en absoluto de FORTRAN, se codificó READ12 y READ16 respectivamente, para leer datos de esos archivos y WRITE3 para grabar resultados en el último archivo. Se menciona el DCB con el cual se grabará el disco auxiliar.

Modelo 156. Ejecución de un programa grabado en la biblioteca con el nombre ARCOS, cuyos datos de entrada residen en sendas cintas magnéticas en cada una de las cuales lee un archivo y cuyos resultados se graban en la propia biblioteca, como un miembro independiente. En el rótulo 19 de la cinta que lleva el número de carrete 60006 está grabado, con el nombre DATARCO1, el primer archivo de entrada y en el rótulo 7 de la cinta que lleva el número de carrete 17778 está grabado, con el nombre DATARCO2, el segundo archivo de entrada. Ambas cintas tienen etiquetas modelo IBM. El problema se puede resolver en dos pasos: en el PASO1 el procedimiento FTG1CLG ejecuta el programa obteniendo los datos de entrada de las cintas y grabando los resultados, con el nombre RESARCOS, en un disco auxiliar; en el PASO2 el utilitario IEBUPDTE transfiere este último archivo del disco a la biblioteca SIMBOLIC.CNEA.

```
//..... JOB ....
//PASO1 EXEC PROC=FTG1CLG
//FORT.SYSIN DD DSN=SIMBOLIC.CNEA(ARCOS),DISP=SHR
/*
//GO.FT18F001 DD DSN=DATARCO1,UNIT=TAPE,VOL=SER=T60006,LABEL=(19,SL),
// DISP=(OLD,KEEP)
//GO.FT19F001 DD DSN=DATARCO2,UNIT=TAPE,VOL=SER=T17778,LABEL=(7,SL),
// DISP=(OLD,KEEP)
//GO.FT04F001 DD DSN=RESARCOS,UNIT=WORKDISK,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
```

```

/*
//PASO2 EXEC PGM=IEBUPDTE,PARM=NEW
//SYSPRINT DD SYSOUT=A
//SYSUT2 DD DSN=SIMBOLIC.CNEA,DISP=OLD
//SYSIN DD *
./  ADD  NAME=RESARCOS
//      DD DSN=RESARCOS,DISP=(OLD,DELETE)
./  ENDUP
/*
//

```

NOTA: Los archivos de entrada de datos están definidos en las dos primeras sentencias GO del PASO1, en donde se les asignan los números de referencia 18 y 19 y el archivo de salida de datos está definido en la tercera sentencia GO, en donde se le asigna el número de referencia 4. Por lo tanto en el programa fuente, cuando se lean datos de esos archivos se codificará READ18 y READ19 respectivamente y cuando se graben resultados en el último archivo se codificará WRITE4. Se menciona el DCB con el cual se grabará el disco auxiliar.

Modelo 157. Ejecución de un programa en tarjetas perforadas en lenguaje simbólico FORTRAN, cuyos datos de entrada residen en sendas cintas magnéticas en cada una de las cuales lee varios archivos y cuyos resultados son cuatro archivos de salida distintos que se graban en la biblioteca, como miembros independientes. En la cinta que lleva el número de carrete 92267 están grabados en los rótulos 8 y 10 los archivos de entrada de nombres DATINP1A y DATINP1B y en la cinta que lleva el número de carrete 10063 están grabados en los rótulos 2, 6 y 7 los archivos de entrada de nombres DATINP2X, DATINP2Y y DATINP2Z. Ambas cintas tienen etiquetas modelo IBM. El problema se puede resolver en dos pasos: en el PASO1 el procedimiento FTG1CLG ejecuta el programa obteniendo los datos de entrada de las cintas y grabando los resultados, con los nombres SALIDAP1, SALIDAP2, SALIDAP3 y SALIDA4, en un mismo disco auxiliar; en el PASO2 el utilitario IEBUPDTE transfiere estos últimos archivos del disco auxiliar a la biblioteca SIMBOLIC.CNEA.

```

//..... JOB ....
//PASO1 EXEC PROC=FTG1CLG
//FORT.SYSIN DD *
Aquí tarjetas perforadas en FORTRAN

```

```

/*
//GO.FT04F001 DD DSN=DATINP1A,UNIT=TAPE,VOL=SER=T92267,LABEL=(8,SL),
// DISP=(OLD,PASS)
//GO.FT04F002 DD DSN=DATINP1B,UNIT=TAPE,VOL=REF=*.GO.FT04F001,
// LABEL=(10,SL),DISP=(OLD,KEEP)
//GO.FT02F001 DD DSN=DATINP2X,UNIT=TAPE,VOL=SER=T10063,LABEL=(2,SL),
// DISP=(OLD,PASS)
//GO.FT02F002 DD DSN=DATINP2Y,UNIT=TAPE,VOL=REF=*.GO.FT02F001,
// LABEL=(6,SL)
//GO.FT02F003 DD DSN=DATINP2Z,UNIT=TAPE,VOL=REF=*.GO.FT02F001,
// LABEL=(7,SL),DISP=(OLD,KEEP)
//GO.FT12F001 DD DSN=SALIDAP1,UNIT=WORKDISK,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//GO.FT15F001 DD DSN=SALIDAP2,VOL=REF=*.GO.FT12F001,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//GO.FT16F001 DD DSN=SALIDAP3,VOL=REF=*.GO.FT12F001,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//GO.FT18F001 DD DSN=SALIDAP4,VOL=REF=*.GO.FT12F001,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
/*
//PASO2 EXEC PGM=IEBUPDTE,PARM=MOD
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA,DISP=SHR
//SYSUT2 DD DSN=SIMBOLIC.CNEA,DISP=OLD
//SYSIN DD *
./ ADD NAME=SALIDAP1
/*
// DD DSN=SALIDAP1
// DD *
./ ADD NAME=SALIDAP2
/*
// DD DSN=SALIDAP2
// DD *
./ ADD NAME=SALIDAP3
/*
// DD DSN=SALIDAP3
// DD *
./ ADD NAME=SALIDAP4
/*
// DD DSN=SALIDAP4
// DD *
./ ENDUP
/*
//

```

NOTA: Los archivos de entrada de datos correspondientes a la cinta número 92267, están definidos en las dos primeras sentencias GO del PASO1, en donde se les asigna el mismo número de referencia 4 y los números de secuencia 1 y 2; por lo tanto en el programa fuente, cuando se lean datos de esa cinta, se codificará una sentencia READ de este tipo: READ(4,n°de formato,END=dirección), en donde la dirección significa la marca de la sentencia a la cual deberá saltar el programa

para continuar la ejecución, cada vez que llega al final de uno de los archivos. En la sentencia DD en que se define el último archivo de esa cinta, se codifica `DISP=(OLD,KEEP)` para que ésta sea rebobinada. Los archivos de entrada de datos correspondientes a la cinta número 10063, están definidos en la tercera, cuarta y quinta sentencia GO, en donde se les asigna el mismo número de referencia 2 y los números de secuencia 1, 2 y 3 ; por consiguiente en el programa fuente, cuando se lean datos de esa cinta, se codificará una sentencia READ del tipo:

`READ(2,n° de formato,END=dirección)`, tal como se explicó mas arriba. En la sentencia DD en que se define el último archivo de esa cinta, se codifica `DISP=(OLD,KEEP)` para que ésta sea rebobinada. A su vez los archivos de salida de datos están definidos en la sexta, séptima, octava y novena sentencia GO, en donde se les asignan los números de referencia 12, 15, 16 y 18; se sigue de esto que en el programa fuente, cuando se graben resultados en esos archivos, se codificará `WRITE12`, `WRITE15`, `WRITE16` y `WRITE18` respectivamente. Se menciona el DCB con el cual se grabará el disco auxiliar.

Modelo 158. Ejecución de un programa en tarjetas perforadas en lenguaje absoluto de FORTRAN, cuyos datos de entrada residen en sendas cintas magnéticas en cada una de las cuales lee varios archivos y cuyos resultados son tres archivos de salida distintos que se graban en la biblioteca, como miembros independientes. En la cinta que lleva el número de carrete 32189, están grabados en los rótulos 20, 21, 33 y 37, los archivos de entrada `ACANDAT1`, `ACANDAT2`, `ACANDAT3` y `ACANDAT4` y en y en la cinta que lleva el número de carrete 00037, están grabados en los rótulos 15 y 17, los archivos de entrada de nombres `DATOS77` y `DATOS87`. Ambas cintas tienen etiquetas modelo IBM. El problema se puede resolver en dos pasos: en el PASO1 el procedimiento FTGILG ejecuta el programa obteniendo los datos de entrada de las cintas y grabando los resultados con los nombres `ACANRES1`, `ACANRES2` y `ACANRES3`, en un mismo disco auxiliar; en el PASO2 el utilitario IEBUPDTE transfiere estos últimos archivos del disco a la biblioteca `SIMBOLIC.CNEA`.

```

//..... JOB ....
//PAS01 EXEC PROC=FTGILG
//LKED.SYSIN DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas en absoluto de FORTRAN
/*
//GO.FT03F001 DD DSN=ACANDAT1,UNIT=TAPE,VOL=SER=T32189,LABEL=(20,SL),
//  DISP=(OLD,PASS)
//GO.FT03F002 DD DSN=ACANDAT2,UNIT=TAPE,VOL=REF=*.GO.FT03F001,
//  LABEL=(21,SL)
//GO.FT03F003 DD DSN=ACANDAT3,UNIT=TAPE,VOL=REF=*.GO.FT03F001,
//  LABEL=(33,SL)
//GO.FT03F004 DD DSN=ACANDAT4,UNIT=TAPE,VOL=REF=*.GO.FT03F001,
//  LABEL=(37,SL),DISP=(OLD,KEEP)
//GO.FT09F001 DD DSN=DATOS77,UNIT=TAPE,VOL=SER=T00037,LABEL=(15,SL),
//  DISP=(OLD,PASS)
//GO.FT09F002 DD DSN=DATOS87,UNIT=TAPE,VOL=REF=*.GO.FT09F001,
//  LABEL=(17,SL),DISP=(OLD,KEEP)
//GO.FT18F001 DD DSN=ACANRES1,UNIT=WORKDISK,DISP=(NEW,PASS),
//  SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//GO.FT20F001 DD DSN=ACANRES2,VOL=REF=*.GO.FT18F001,DISP=(NEW,PASS),
//  SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//GO.FT12F001 DD DSN=ACANRES3,VOL=REF=*.GO.FT18F001,DISP=(NEW,PASS),
//  SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
/*
//PAS02 EXEC PGM=IEBUPDTE,PARM=MOD
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA,DISP=SHR
//SYSUT2 DD DSN=SIMBOLIC.CNEA,DISP=OLD
//SYSIN DD *
./  ADD  NAME=ACANRES1
/*
//      DD DSN=ACANRES1,DISP=(OLD,DELETE)
//      DD *
./  ADD  NAME=ACANRES2
/*
//      DD DSN=ACANRES2,DISP=(OLD,DELETE)
//      DD *
./  ADD  NAME=ACANRES3
/*
//      DD DSN=ACANRES3,DISP=(OLD,DELETE)
//      DD *
./  ENDUP
/*
//

```

NOTA: Los archivos de entrada de datos correspondientes a la cinta número 32189, están definidos en las cuatro primeras sentencias GO del PAS01, en donde se les asigna el mismo número de referencia 3 y los números de secuencia 1, 2, 3 y 4; por lo tanto en el programa fuente que dió origen al lote de tarjetas en absoluto de FORTRAN, para leer datos de esa cinta se codificó una sentencia READ del tipo: READ(3,n° de formato,END=dirección), en donde la "dirección" significa la marca de la sentencia a la cual deberá saltar el programa para continuar la ejecu-

ción, cada vez que llega al final de uno de los archivos. En la sentencia DD en que se define el último archivo de esa cinta, se codifica DISP=(OLD,KEEP) para que la cinta sea rebobinada. Los archivos de entrada de datos correspondientes a la cinta número 00037, están definidos en la quinta y sexta sentencia GO, en donde se les asigna el mismo número de referencia 9 y los números de secuencia 1 y 2; por consiguiente en el programa fuente original para leer datos de esa cinta, se codificó una sentencia READ del tipo READ(9,n° de formato,END=dirección), tal como se explicó mas arriba. En la sentencia DD en que se define el último archivo de esa cinta, se codifica DISP=(OLD,KEEP) para que la cinta sea rebobinada. A su vez los archivos de salida de datos están definidos en la séptima, octava y novena sentencia GO, en donde se les asignan los números de referencia 18, 20 y 12; se sigue de esto que en el programa fuente original, para grabar resultados en esos archivos, se codificó WRITE18, WRITE20 y WRITE12 respectivamente. Se menciona el DCB con el cual se grabará el disco auxiliar.

Modelo 159 . Ejecución de un programa grabado en la biblioteca con el nombre METRICA, cuyos datos de entrada residen en sendas cintas magnéticas en cada una de las cuales lee varios archivos y cuyos resultados son tres archivos de salida distintos que se graban en la propia biblioteca, como miembros independientes. En la cinta que lleva el número de carrete 00300 están grabados en los rótulos 5, 6 y 18 los archivos de entrada de nombres DATASIM1, DATASIM2 y DATASIM3 y en la cinta que lleva el número de carrete 50512 están grabados en los rótulos 20, 25 y 26 los archivos de entrada de nombres DATOS41, DATOS42 y DATOS43. Ambas cintas tienen etiquetas modelo IBM. El problema se puede resolver en dos pasos: en el PASO1 el procedimiento FTG1CLG ejecuta el programa obteniendo los datos de entrada de las cintas y grabando los resultados, con los nombres SALRES1, SALRES2 y SALRES3, en un mismo disco auxiliar; en el PASO2 el utilitario IEBUPDTE transfiere estos últimos archivos del disco a la biblioteca SIMBOLIC.CNEA.

```
//..... JOB ....
//PASO1 EXEC PROC=FTG1CLG
```

```

//FORT.SYSIN DD DSN=SIMBOLIC.CNEA(METRICA),DISP=SHR
/*
//GO.FT13F001 DD DSN=DATASIM1,UNIT=TAPE,VOL=SER=T00300,LABEL=(5,SL),
// DISP=(OLD,PASS)
//GO.FT13F002 DD DSN=DATASIM2,UNIT=TAPE,VOL=REF=*.GO.FT13F001,
// LABEL=(6,SL)
//GO.FT13F003 DD DSN=DATASIM3,UNIT=TAPE,VOL=REF=*.GO.FT13F001,
// LABEL=(18,SL),DISP=(OLD,KEEP)
//GO.FT15F001 DD DSN=DATOS41,UNIT=TAPE,VOL=SER=T50512,LABEL=(20,SL),
// DISP=(OLD,PASS)
//GO.FT15F002 DD DSN=DATOS42,UNIT=TAPE,VOL=REF=*.GO.FT15F001,
// LABEL=(25,SL)
//GO.FT15F003 DD DSN=DATOS43,UNIT=TAPE,VOL=REF=*.GO.FT15F001,
// LABEL=(26,SL),DISP=(OLD,KEEP)
//GO.FT01F001 DD DSN=SALRES1,UNIT=WORKDISK,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//GO.FT03F001 DD DSN=SALRES2,VOL=REF=*.GO.FT01F001,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//GO.FT04F001 DD DSN=SALRES3,VOL=REF=*.GO.FT01F001,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
/*
//PASO2 EXEC PGM=IEBUPDTE,PARM.MOD
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA,DISP=SHR
//SYSUT2 DD DSN=SIMBOLIC.CNEA,DISP=OLD
//SYSIN DD *
./ ADD NAME=SALRES1
/*
// DD DSN=SALRES1,DISP=(OLD,DELETE)
// DD *
./ ADD NAME=SALRES2
/*
// DD DSN=SALRES2,DISP=(OLD,DELETE)
// DD *
./ ADD NAME=SALRES3
/*
// DD DSN=SALRES3,DISP=(OLD,DELETE)
// DD *
./ ENDUP
/*
//

```

NOTA: Los archivos de entrada de datos correspondientes a la cinta número 00300, están definidos en las tres primeras sentencias GO del PASO1, en donde se les asigna el mismo número de referencia 13 y los números de secuencia 1, 2 y 3; por lo tanto en el programa fuente, cuando se lean datos de esa cinta, se codificará una sentencia READ del tipo: READ(13,nº de formato,END=dirección), en donde la dirección significa la marca de la sentencia a la cual deberá saltar el programa para continuar la ejecución, cada vez que llega al final de uno de los archivos.

En la sentencia DD en que se define el último archivo de esa cinta, se codifica `DISP=(OLD,KEEP)` para que la cinta sea rebobinada. Los archivos de entrada de datos correspondientes a la cinta número 50512, están definidos en la cuarta, quinta y sexta sentencia GO, en donde se les asigna el mismo número de referencia 15 y los números de secuencia 1, 2 y 3; por consiguiente en el programa fuente, cuando se lean datos de esa cinta, se codificará una sentencia READ del tipo: `READ(15,n°de formato,END=dirección)`, tal como se explicó mas arriba. En la sentencia DD en que se define el último archivo de esa cinta, se codifica `DISP=(OLD,KEEP)` para que la cinta sea rebobinada. A su vez los archivos de salida de datos están definidos en la séptima, octava y novena sentencia GO, en donde se les asignan los números de referencia 1, 3 y 4; se sigue de esto que en el programa fuente, cuando se graben resultados en esos archivos, se codificará `WRITE1`, `WRITE3` y `WRITE4` respectivamente. Se menciona el DCB con el cual se grabará el disco auxiliar.

PUNTO 5.

Ejecución de programas FORTRAN que utilizan archivos de entrada de datos residentes en una biblioteca particionada y catalogada de programas en lenguaje simbólico, de nombre SIMBOLIC.CNEA y simultáneamente graban archivos de salida de datos en esa misma biblioteca.

Modelo 160 . Ejecución de un programa en tarjetas perforadas en lenguaje simbólico FORTRAN cuyos datos de entrada constituyen un archivo que reside en la biblioteca, como un miembro independiente y cuyos resultados se graban en esa misma biblioteca, también como un miembro independiente. El problema se puede resolver en tres pasos: en el PASO1 el utilitario IEBGENER transfiere el archivo de nombre DATOS17 de la biblioteca a un disco auxiliar; en el PASO2 el procedimiento FTG1CLG ejecuta el programa obteniendo los datos de entrada del disco auxiliar y grabando los resultados, con el nombre RESUL17, también en un disco auxiliar; en el PASO3 el utilitario IEBUPDTE transfiere este último archivo del disco a la biblioteca SIMBOLIC.CNEA.

```
//..... JOB ....
//PASO1 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA(DATOS17),DISP=SHR
//SYSUT2 DD DSN=DATOS17,UNIT=WORKDISK,DISP=(NEW,PASS),
//  SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//SYSIN DD DUMMY
/*
//PASO2 EXEC PROC=FTG1CLG
//FORT.SYSIN DD *
Aquí tarjetas perforadas en FORTRAN
/*
//GO.FT08F001 DD DSN=DATOS17,UNIT=WORKDISK,DISP=(OLD,DELETE)
//GO.FT09F001 DD DSN=RESUL17,UNIT=WORKDISK,DISP=(NEW,PASS),
//  SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
/*
//PASO3 EXEC PGM=IEBUPDTE,PARM=NEW
//SYSPRINT DD SYSOUT=A
//SYSUT2 DD DSN=SIMBOLIC.CNEA,DISP=OLD
//SYSIN DD *
./  ADD  NAME=RESUL17
./      DD DSN=RESUL17,DISP=(OLD,DELETE)
./  ENDUP
```

```
/*
//
```

NOTA: El archivo de entrada de datos está definido en la primera sentencia GO del PASO1, en donde se le asigna el número de referencia 8 y el archivo de salida de datos está definido en la segunda sentencia GO, en donde se le asigna el número de referencia 9. Por lo tanto en el programa fuente, cuando se lean datos del primer archivo se codificará READ8 y cuando se graben resultados en el segundo archivo, se codificará WRITE9. Se menciona el DCB con el cual se grabará el disco auxiliar.

Modelo 161 . Ejecución de un programa en tarjetas perforadas en lenguaje absoluto de FORTRAN, cuyos datos de entrada constituyen un archivo que reside en la biblioteca, como un miembro independiente y cuyos resultados se graban en esa misma biblioteca, también como un miembro independiente. El problema se puede resolver en tres pasos: en el PASO1 el utilitario IEBGENER transfiere el archivo de nombre DATOPLUS de la biblioteca a un disco auxiliar; en el PASO2 el procedimiento FTGLLG ejecuta el programa obteniendo los datos de entrada del disco auxiliar y grabando los resultados, con el nombre RESULPLU , también en un disco auxiliar; en el PASO3 el utilitario IEBUPDTE transfiere este último archivo del disco a la biblioteca SIMBOLIC.CNEA.

```
//..... JOB ....
//PASO1 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA(DATOPLUS),DISP=SHR
//SYSUT2 DD DSN=DATOPLUS,UNIT=WORKDISK,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//SYSIN DD DUMMY
/*
//PASO2 EXEC PROC=FTGLLG
//LKED.SYSIN DD *,DCB=BLKSIZE=80
Aquí tarjetas perforadas en absoluto de FORTRAN
/*
//GO.FT01F001 DD DSN=DATOPLUS,UNIT=WORKDISK,DISP=(OLD,DELETE)
//GO.FT04F001 DD DSN=RESULPLU,UNIT=WORKDISK,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
/*
//PASO3 EXEC PGM=IEBUPDTE,PARM=NEW
```

```
//SYSPRINT DD SYSOUT=A
//SYSUT2 DD DSN=SIMBOLIC.CNEA,DISP=OLD
//SYSIN DD *
./ ADD NAME=RESULPLU
// DD DSN=RESULPLU,DISP=(OLD,DELETE)
./ ENDUP
/*
//
```

NOTA: El archivo de entrada de datos está definido en la primera sentencia GO del PASO2, en donde se le asigna el número de referencia 1 y el archivo de salida de datos está definido en la segunda sentencia GO, en donde se le asigna el número de referencia 4. Por lo tanto en el programa fuente que dió origen al lote de tarjetas en absoluto de FORTRAN, se codificó READ1 para leer datos del primer archivo y WRITE4 para grabar resultados en el segundo archivo. Se menciona el DCB con el cual se grabará el disco auxiliar.

Modelo 162 . Ejecución de un programa en lenguaje simbólico FORTRAN de nombre SIMETRIA, grabado en el rótulo 12 de una cinta magnética, cuyos datos de entrada constituyen un archivo que reside en la biblioteca, como un miembro independiente y cuyos resultados se graban en esa misma biblioteca, también como un miembro independiente. La cinta lleva el número de carrete 07751 y tiene etiquetas modelo IBM. El problema se puede resolver en tres pasos: en el PASO1 el utilitario IEBGENER transfiere el archivo de nombre DATOSMAQ de la biblioteca a un disco auxiliar; en el PASO2 el procedimiento FTG1CLG ejecuta el programa obteniendo los datos de entrada del disco auxiliar y grabando los resultados, con el nombre RESULMAQ, también en un disco auxiliar; en el PASO3 el utilitario IEBUPDTE transfiere este último archivo del disco a la biblioteca SIMBOLIC.CNEA.

```
//..... JOB ....
//PASO1 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA(DATOSMAQ),DISP=SHR
//SYSUT2 DD DSN=DATOSMAQ,UNIT=WORKDISK,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//SYSIN DD DUMMY
/*
//PASO2 EXEC PROC=FTG1CLG
//FORT.SYSIN DD DSN=SIMETRIA,UNIT=TAPE,VOL=SER=T07751,LABEL=(12,SL),
```

```
// DISP=(OLD,KEEP)
/*
//GO.FT03F001 DD DSN=DATOSMAQ,UNIT=WORKDISK,DISP=(OLD,DELETE)
//GO.FT10F001 DD DSN=RESULMAQ,UNIT=WORKDISK,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
/*
//PASO3 EXEC PGM=IEBUPDTE,PARM=NEW
//SYSPRINT DD SYSOUT=A
//SYSUT2 DD DSN=SIMBOLIC.CNEA,DISP=OLD
//SYSIN DD *
./ ADD NAME=RESULMAQ
// DD DSN=RESULMAQ,DISP=(OLD,DELETE)
./ ENDUP
/*
//
```

NOTA: El archivo de entrada de datos está definido en la primera sentencia GO del PASO2, en donde se le asigna el número de referencia 3 y el archivo de salida de datos está definido en la segunda sentencia GO, en donde se le asigna el número de referencia 10. Por lo tanto en el programa fuente, cuando se lean datos del primer archivo se codificará READ3 y cuando se graben resultados en el segundo archivo se codificará WRITE10. Se menciona el DCB con el cual se grabará el disco auxiliar.

Modelo 163. Ejecución de un programa en lenguaje absoluto de FORTRAN de nombre SEMICON, grabado en el rótulo 33 de una cinta magnética, cuyos datos de entrada constituyen un archivo que reside en la biblioteca, como un miembro independiente y cuyos resultados se graban en esa misma biblioteca, también como un miembro independiente. La cinta lleva el número de carrete 68243 y tiene etiquetas modelo IBM. El problema se puede resolver en tres pasos: en el PASO1 el utilitario IEBGENER transfiere el archivo de nombre DATOSSEM de la biblioteca a un disco auxiliar; en el PASO2 el procedimiento FTGLLG ejecuta el programa obteniendo los datos de entrada del disco auxiliar y grabando los resultados, con el nombre RESULSEM, también en un disco; en el PASO3 el utilitario IEBUPDTE transfiere este último archivo del disco a la biblioteca SIMBOLIC.CNEA.

```
//..... JOB ....
//PASO1 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
```

```

//SYSUT1 DD DSN=SIMBOLIC.CNEA(DATOSSEM),DISP=SHR
//SYSUT2 DD DSN=DATOSSEM,UNIT=WORKDISK,DISP=(NEW,PASS),
//  SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//SYSIN DD DUMMY
/*
//PASO2 EXEC PROC=FTG1LG
//LKED.SYSIN DD DSN=SEMICON,UNIT=TAPE,VOL=SER=T68243,LABEL=(33,SL),
//  DISP=(OLD,KEEP)
/*
//GO.FT08F001 DD DSN=DATOSSEM,UNIT=WORKDISK,DISP=(OLD,DELETE)
//GO.FT12F001 DD DSN=RESULSEM,UNIT=WORKDISK,DISP=(NEW,PASS),
//  SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
/*
//PASO3 EXEC PGM=IEBUPDTE,PARM=NEW
//SYSPRINT DD SYSOUT=A
//SYSUT2 DD DSN=SIMBOLIC.CNEA,DISP=OLD
//SYSIN DD *
./  ADD  NAME=RESULSEM
//      DD DSN=RESULSEM,DISP=(OLD,DELETE)
./  ENDUP
/*
//

```

NOTA: El archivo de entrada de datos está definido en la primera sentencia GO del PASO2, en donde se le asigna el número de referencia 8 y el archivo de salida de datos está definido en la segunda sentencia GO, en donde se le asigna el número de referencia 12. Por lo tanto en el programa fuente que dió origen al módulo grabado en la cinta, se codificó READ8 para leer datos del primer archivo y WRITE12 para grabar resultados en el segundo archivo. Se menciona el DCB con el cual se grabará el disco auxiliar.

Modelo 164 . Ejecución de un programa grabado en la biblioteca con el nombre FOCATIP, cuyos datos de entrada constituyen un archivo que reside en la propia biblioteca, como un miembro independiente y cuyos resultados se graban en esa misma biblioteca, también como un miembro independiente. El problema se puede resolver en tres pasos: en el PASO1 el utilitario IEBGENER transfiere el archivo de nombre FOCADAT de la biblioteca a un disco auxiliar; en el PASO2 el procedimiento FTG1CLG ejecuta el programa obteniendo los datos de entrada del disco auxiliar y grabando los resultados, con el nombre FOCARES, también en un disco; en el PASO3 el utilitario IEBUPDTE transfiere este último archivo del disco a la biblioteca SIMBOLIC.CNEA.

```
//..... JOB ....
//PAS01 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA(FOCADAT),DISP=SHR
//SYSUT2 DD DSN=FOCADAT,UNIT=WORKDISK,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//SYSIN DD DUMMY
/*
//PAS02 EXEC PROC=FTG1CLG
//FORT.SYSIN DD DSN=SIMBOLIC.CNEA(FOCATIP),DISP=SHR
/*
//GO.FT10F001 DD DSN=FOCADAT,UNIT=WORKDISK,DISP=(OLD,DELETE)
//GO.FT13F001 DD DSN=FOCARES,UNIT=WORKDISK,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
/*
//PAS03 EXEC PGM=IEBUPDTE,PARM=NEW
//SYSPRINT DD SYSOUT=A
//SYSUT2 DD DSN=SIMBOLIC.CNEA,DISP=OLD
//SYSIN DD *
./ ADD NAME=FOCARES
// DD DSN=FOCARES,DISP=(OLD,DELETE)
./ ENDUP
/*
//
```

NOTA: El archivo de entrada de datos está definido en la primera sentencia GO del PAS02, en donde se le asigna el número de referencia 10 y el archivo de salida de datos está definido en la segunda sentencia GO, en donde se le asigna el número de referencia 13; por lo tanto en el programa fuente cuando se lean datos del primer archivo, se codificará READ10 y cuando se graben resultados en el segundo archivo, se codificará WRITE13. Se menciona el DCB con el cual se grabará el disco auxiliar.

Modelo 165. Ejecución de un programa en tarjetas perforadas en lenguaje simbólico FORTRAN, cuyos datos de entrada son tres archivos distintos que residen en la biblioteca, como miembros independientes y cuyos resultados son dos archivos de salida distintos que se graban en esa misma biblioteca, también como miembros independientes. El problema se puede resolver en cinco pasos: en los tres primeros (PAS01, PAS02 y PAS03) el utilitario IEBGENER transfiere, de a uno, los archivos de nombres DATOSX, DATOSY y DATOSZ de la biblioteca a un mismo disco auxiliar; en el PAS04 el procedimiento FTG1CLG ejecuta el programa obteniendo los da-

tos de entrada del disco auxiliar y grabando los resultados, con los nombres PARAMA y PARAMB, en un mismo disco; en el PASO5 el utilitario IEBURDTE transfiere estos últimos archivos del disco a la biblioteca SIMBOLIC.CNEA.

```
//..... JOB ....
//PASO1 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA(DATOSX),DISP=SHR
//SYSUT2 DD DSN=DATOSX,UNIT=WORKDISK,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//SYSIN DD DUMMY
/*
//PASO2 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA(DATOSY),DISP=SHR
//SYSUT2 DD DSN=DATOSY,VOL=REF=*.PASO1.SYSUT2,DISP=(,PASS),
// SPACE=(CYL,(1,1))
//SYSIN DD DUMMY
/*
//PASO3 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA(DATOSZ),DISP=SHR
//SYSUT2 DD DSN=DATOSZ,VOL=REF=*.PASO1.SYSUT2,DISP=(,PASS),
// SPACE=(CYL,(1,1))
//SYSIN DD DUMMY
/*
//PASO4 EXEC PROC=FTG1CLG
//FORT.SYSIN DD *
  Aquí tarjetas perforadas en FORTRAN
/*
//GO.FT10F001 DD DSN=DATOSX,UNIT=WORKDISK,DISP=(OLD,DELETE)
//GO.FT18F001 DD DSN=DATOSY,UNIT=WORKDISK,DISP=(OLD,DELETE)
//GO.FT19F001 DD DSN=DATOSZ,UNIT=WORKDISK,DISP=(OLD,DELETE)
//GO.FT17F001 DD DSN=PARAMA,UNIT=WORKDISK,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//GO.FT08F001 DD DSN=PARAMB,VOL=REF=*.GO.FT17F001,DISP=(,PASS),
// SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
/*
//PASO5 EXEC PGM=IEBUPDTE,PARM=MOD
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA,DISP=SHR
//SYSUT2 DD DSN=SIMBOLIC.CNEA,DISP=OLD
//SYSIN DD *
./ ADD NAME=PARAMA
/*
// DD DSN=PARAMA,DISP=(OLD,DELETE)
// DD *
./ ADD NAME=PARAMB
/*
// DD DSN=PARAMB,DISP=(OLD,DELETE)
// DD *
./ ENDUP
/*
//
```


NOTA: Los archivos de entrada de datos están definidos en las tres primeras sentencias GO del PAS04, en donde se les asignan los números de referencia 10, 18 y 19; por lo tanto en el programa fuente, cuando se lean datos de esos archivos, se codificará READ10, READ18 y READ19 respectivamente. A su vez los archivos de salida de datos están definidos en la cuarta y quinta sentencia GO, en donde se les asignan los números de referencia 17 y 8; por consiguiente en el programa fuente, cuando se graben resultados en esos archivos, se codificará WRITE17 y WRITE8 respectivamente. Se menciona el DCB con el cual se grabará el disco auxiliar.

Modelo 166 . Ejecución de un programa en tarjetas perforadas en lenguaje absoluto de FORTRAN, cuyos datos de entrada son dos archivos distintos que residen en la biblioteca, como miembros independientes y cuyos resultados son tres archivos de salida distintos que se graban en esa misma biblioteca, también como miembros independientes. El problema se puede resolver en cuatro pasos: en los dos primeros (PAS01 y PAS02) el utilitario IEBGENER transfiere, de a uno, los archivos de nombres DATOSPER y DATOSFEL de la biblioteca a un mismo disco auxiliar; en el PAS03 el procedimiento FTG1LG ejecuta el programa obteniendo los datos de entrada del disco auxiliar y grabando los resultados, con los nombres RESUL30, RESUL31 y RESUL32, en un mismo disco; en el PAS04 el utilitario IEBUPDTE transfiere estos últimos archivos del disco a la biblioteca SIMBOLIC.CNEA.

```
//..... JOB ....
//PAS01 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA(DATOSPER),DISP=SHR
//SYSUT2 DD DSN=DATOSPER,UNIT=WORKDISK,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//SYSIN DD DUMMY
/*
//PAS02 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA(DATOSFEL),DISP=SHR
//SYSUT2 DD DSN=DATOSFEL,VOL=REF=*.PAS01.SYSUT2,DISP=(,PASS),
// SPACE=(CYL,(1,1))
//SYSIN DD DUMMY
/*
//PAS03 EXEC PROC=FTG1LG
//LKED.SYSIN DD *,DCB=BLKSIZE=80
```

```

Aquí tarjetas perforadas en absoluto de FORTRAN
/*
//GO.FT12F001 DD DSN=DATOSPER,UNIT=WORKDISK,DISP=(OLD,DELETE)
//GO.FT14F001 DD DSN=DATOSFEL,UNIT=WORKDISK,DISP=(OLD,DELETE)
//GO.FT13F001 DD DSN=RESUL30,UNIT=WORKDISK,DISP=(NEW,PASS),
//  SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//GO.FT15F001 DD DSN=RESUL31,VOL=REF=*.GO.FT13F001,DISP=(,PASS),
//  SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//GO.FT16F001 DD DSN=RESUL32,VOL=REF=*.GO.FT13F001,DISP=(,PASS),
//  SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
/*
//PASO4 EXEC PGM=IEBUPDTE,PARM=MOD
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA,DISP=SHR
//SYSUT2 DD DSN=SIMBOLIC.CNEA,DISP=OLD
//SYSIN DD *
./  ADD  NAME=RESUL30
/*
//      DD DSN=RESUL30,DISP=(OLD,DELETE)
//      DD *
./  ADD  NAME=RESUL31
/*
//      DD DSN=RESUL31,DISP=(OLD,DELETE)
//      DD *
./  ADD  NAME=RESUL32
/*
//      DD DSN=RESUL32,DISP=(OLD,DELETE)
//      DD *
./  ENDUP
/*
//

```

NOTA: Los archivos de entrada de datos están definidos en las dos primeras sentencias GO del PASO3, en donde se les asignan los números de referencia 12 y 14; por lo tanto en el programa fuente que dió origen al lote de tarjetas en absoluto de FORTRAN, se codificó READ12 y READ14 respectivamente, para leer datos de esos archivos. A su vez los archivos de salida de datos están definidos en la tercera, cuarta y quinta sentencia GO, en donde se les asignan los números de referencia 13, 15 y 16; por consiguiente en el programa fuente se codificó WRITE13, WRITE15 y WRITE16 respectivamente, para grabar resultados en esos archivos. Se menciona el DCB con el cual se grabará el disco auxiliar.

Modelo 167 . Ejecución de un programa en lenguaje simbólico FORTRAN de nombre CURVATUR, grabado en el rótulo 22 de una cinta magnética, cuyos datos de entrada son tres archivos distintos que residen en la biblioteca, como miembros indepen-

dientes y cuyos resultados son tres archivos de salida distintos que se graban en esa misma biblioteca, también como miembros independientes. La cinta lleva el número de carrete 33781 y tiene etiquetas modelo IBM. El problema se puede resolver en cinco pasos: en los tres primeros (PAS01, PAS02 y PAS03) el utilitario IEBGENER transfiere, de a uno, los archivos de nombres DATOTUR1, DATOTUR2 y DATOTUR3 de la biblioteca a un mismo disco auxiliar; en el PAS04 el procedimiento FTG1CLG ejecuta el programa obteniendo los datos de entrada del disco auxiliar y grabando los resultados con los nombres RESUTUR1, RESUTUR2 y RESUTUR3, en un mismo disco; en el PAS05 el utilitario IEBUPDTE transfiere estos últimos archivos del disco a la biblioteca SIMBOLIC.CNEA.

```
//..... JOB ....
//PAS01 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA(DATOTUR1),DISP=SHR
//SYSUT2 DD DSN=DATOTUR1,UNIT=WORKDISK,DISP=(NEW,PASS),
//  SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//SYSIN DD DUMMY
/*
//PAS02 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA(DATOTUR2),DISP=SHR
//SYSUT2 DD DSN=DATOTUR2,VOL=REF=*.PAS01.SYSUT2,DISP=(,PASS),
//  SPACE=(CYL,(1,1))
//SYSIN DD DUMMY
/*
//PAS03 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA(DATOTUR3),DISP=SHR
//SYSUT2 DD DSN=DATOTUR3,VOL=REF=*.PAS01.SYSUT2,DISP=(,PASS),
//  SPACE=(CYL,(1,1))
//SYSIN DD DUMMY
/*
//PAS04 EXEC PROC=FTG1CLG
//FORT.SYSIN DD DSN=CURVATUR,UNIT=TAPE,VOL=SER=T33781,LABEL=(22,SL),
//  DISP=(OLD,KEEP)
/*
//GO.FT03F001 DD DSN=DATOTUR1,UNIT=WORKDISK,DISP=(OLD,DELETE)
//GO.FT20F001 DD DSN=DATOTUR2,UNIT=WORKDISK,DISP=(OLD,DELETE)
//GO.FT11F001 DD DSN=DATOTUR3,UNIT=WORKDISK,DISP=(OLD,DELETE)
//GO.FT04F001 DD DSN=RESUTUR1,UNIT=WORKDISK,DISP=(NEW,PASS),
//  SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//GO.FT02F001 DD DSN=RESUTUR2,VOL=REF=*.GO.FT04F001,DISP=(,PASS),
//  SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//GO.FT09F001 DD DSN=RESUTUR3,VOL=REF=*.GO.FT04F001,DISP=(,PASS),
//  SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
/*
//PAS05 EXEC PGM=IEBUPDTE,PARM=MOD
```

```

//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA,DISP=SHR
//SYSUT2 DD DSN=SIMBOLIC.CNEA,DISP=OLD
//SYSIN DD *
./  ADD  NAME=RESUTUR1
/*
//      DD DSN=RESUTUR1,DISP=(OLD,DELETE)
//      DD *
./  ADD  NAME=RESUTUR2
/*
//      DD DSN=RESUTUR2,DISP=(OLD,DELETE)
//      DD *
./  ADD  NAME=RESUTUR3
/*
//      DD DSN=RESUTUR3,DISP=(OLD,DELETE)
//      DD *
./  ENDUP
/*
//

```

NOTA: Los archivos de entrada de datos están definidos en las tres primeras sentencias GO del PASO4, en donde se les asignan los números de referencia 3, 20 y 11; por lo tanto en el programa fuente, cuando se lean datos de esos archivos, se codificará READ3, READ20 y READ11 respectivamente. A su vez los archivos de salida de datos están definidos en la cuarta, quinta y sexta sentencia GO, en donde se les asignan los números de referencia 4, 2 y 9; por consiguiente en el programa fuente, cuando se graben resultados en esos archivos, se codificará WRITE4, WRITE2 y WRITE9 respectivamente. Se menciona el DCB con el cual se grabará el disco auxiliar.

Modelo 168 . Ejecución de un programa en lenguaje absoluto de FORTRAN de nombre INVERS, grabado en el rótulo 17 de una cinta magnética, cuyos datos de entrada son dos archivos distintos que residen en la biblioteca, como miembros independientes y cuyos resultados son dos archivos de salida distintos que se graban en esa misma biblioteca, también como miembros independientes. La cinta lleva el número de carrete 90254 y tiene etiquetas modelo IBM. El problema se puede resolver en cuatro pasos: en los dos primeros (PASO1 y PASO2) el utilitario IEBGENER transfiere, de a uno, los archivos de nombre DATINV26 y DATINV42 de la biblioteca a un mismo disco auxiliar; en el PASO3 el procedimiento FTGLLG ejecuta el pro-

grama obteniendo los datos de entrada del disco auxiliar y grabando los resultados, con los nombres RESINV1 y RESINV2, en un mismo disco; en el PASO4 el utilitario IEBUPDTE transfiere estos últimos archivos del disco a la biblioteca

SIMBOLIC.CNEA.

```
//..... JOB ....
//PASO1 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA(DATINV26),DISP=SHR
//SYSUT2 DD DSN=DATINV26,UNIT=WORKDISK,DISP=(NEW,PASS),
//  SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//SYSIN DD DUMMY
/*
//PASO2 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA(DATINV42),DISP=SHR
//SYSUT2 DD DSN=DATINV42,VOL=REF=*.PASO1.SYSUT2,DISP=(NEW,PASS),
//  SPACE=(CYL,(1,1))
//SYSIN DD DUMMY
/*
//PASO3 EXEC PROC=FTG1LG
//LKED.SYSIN DD DSN=INVERS,UNIT=TAPE,VOL=SER=T90254,LABEL=(17,SL),
//  DISP=(OLD,KEEP)
/*
//GO.FT03F001 DD DSN=DATINV26,UNIT=WORKDISK,DISP=(OLD,DELETE)
//GO.FT18F001 DD DSN=DATINV42,UNIT=WORKDISK,DISP=(OLD,DELETE)
//GO.FT01F001 DD DSN=RESINV1,UNIT=WORKDISK,DISP=(NEW,PASS),
//  SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//GO.FT20F001 DD DSN=RESINV2,VOL=REF=*.GO.FT01F001,DISP=(,PASS),
//  SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
/*
//PASO4 EXEC PGM=IEBUPDTE,PARM=MOD
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA,DISP=SHR
//SYSUT2 DD DSN=SIMBOLIC.CNEA,DISP=OLD
//SYSIN DD *
./  ADD  NAME=RESINV1
/*
//      DD DSN=RESINV1,DISP=(OLD,DELETE)
//      DD *
./  ADD  NAME=RESINV2
/*
//      DD DSN=RESINV2,DISP=(OLD,DELETE)
//      DD *
./  ENDUP
/*
//
```

NOTA: Los archivos de entrada de datos están definidos en las dos primeras sentencias GO del PASO3, en donde se les asignan los números de referencia 3 y 18; por lo tanto en el programa fuente que dió origen al módulo grabado en la cinta,

se codificó READ3 y READ18 respectivamente, para leer datos de esos archivos. A su vez los archivos de salida de datos están definidos en la tercera y cuarta sentencia GO, en donde se les asignan los números de referencia 1 y 20; por consiguiente en el programa fuente se codificó WRITE1 y WRITE20 respectivamente, para grabar resultados en esos archivos. Se menciona el DCB con el cual se grabará el disco auxiliar.

Modelo 169 . Ejecución de un programa grabado en la biblioteca con el nombre GRANLI, cuyos datos de entrada son dos archivos distintos que residen en la biblioteca como miembros independientes y cuyos resultados son tres archivos de salida distintos que se graban en la propia biblioteca, también como miembros independientes. El problema se puede resolver en cuatro pasos: en los dos primeros (PASO1 y PASO2) el utilitario IEBGENER transfiere, de a uno, los archivos de nombres GRANDATA y GRANDATB de la biblioteca a un mismo disco auxiliar; en el PASO3 el procedimiento FTG1CLG ejecuta el programa obteniendo los datos de entrada del disco auxiliar y grabando los resultados, con los nombres GRANRES1, GRANRES2 y GRANRES3, en un mismo disco; en el PASO4 el utilitario IEBUPDTE transfiere estos últimos archivos del disco a la biblioteca SIMBOLIC.CNEA.

```
//..... JOB ....
//PASO1 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA(GRANDATA),DISP=SHR
//SYSUT2 DD DSN=GRANDATA,UNIT=WORKDISK,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//SYSIN DD DUMMY
/*
//PASO2 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA(GRANDATB),DISP=SHR
//SYSUT2 DD DSN=GRANDATB,VOL=REF=*.PASO1.SYSUT2,DISP=(,PASS),
// SPACE=(CYL,(1,1))
//SYSIN DD DUMMY
/*
//PASO3 EXEC PROC=FTG1CLG
//FORT.SYSIN DD DSN=SIMBOLIC.CNEA(GRANLI),DISP=SHR
/*
//GO.FT13F001 DD DSN=GRANDATA,UNIT=WORKDISK,DISP=(OLD,DELETE)
//GO.FT15F001 DD DSN=GRANDATB,UNIT=WORKDISK,DISP=(OLD,DELETE)
//GO.FT01F001 DD DSN=GRANRES1,UNIT=WORKDISK,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
```

```

//GO.FT03F001 DD DSN=GRANRES2,VOL=REF=*.GO.FT01F001,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//GO.FT04F001 DD DSN=GRANRES3,VOL=REF=*.GO.FT01F001,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
/*
//PAS04 EXEC PGM=IEBUPDTE,PARM=MOD
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA,DISP=SHR
//SYSUT2 DD DSN=SIMBOLIC.CNEA,DISP=OLD
//SYSIN DD *
./ ADD NAME=GRANRES1
/*
// DD DSN=GRANRES1,DISP=(OLD,DELETE)
// DD *
./ ADD NAME=GRANRES2
/*
// DD DSN=GRANRES2,DISP=(OLD,DELETE)
// DD *
./ ADD NAME=GRANRES3
/*
// DD DSN=GRANRES3,DISP=(OLD,DELETE)
// DD *
./ ENDUP
/*
//

```

NOTA: Los archivos de entrada de datos están definidos en las dos primeras sentencias GO del PAS03, en donde se les asignan los números de referencia 13 y 15; por lo tanto en el programa fuente, cuando se lean datos de esos archivos se codificará READ13 y READ15 respectivamente. A su vez los archivos de salida de datos están definidos en la tercera, cuarta y quinta sentencia GO, en donde se les asignan los números de referencia 1, 3 y 4; por consiguiente en el programa fuente, cuando se graben resultados en esos archivos, se codificará WRITE1, WRITE3 y WRITE4 respectivamente. Se menciona el DCB con el cual se grabará el disco auxiliar.

CAPITULO IV

MODELOS PARA EL USO DE UNA BIBLIOTECA PARTICIONADA
Y CATALOGADA DE MODULOS DE CARGA QUE RESIDE EN
UNA UNIDAD DE DISCOS

PUNTO 1.

Ejecución de módulos de carga que configuran un programa completo, grabados en una biblioteca de nombre REALES.CNEA.

Modelo 170. Ejecución de un módulo de carga procedente de FORTRAN, grabado en la biblioteca con el nombre PASATIM.

```
//..... JOB ....
//PAS01 EXEC PGM=PASATIM
//STEPLIB DD DSN=REALES.CNEA,DISP=SHR
/*
//FT06F001 DD SYSOUT=A
//FT05F001 DD *
Aquí tarjetas perforadas de DATOS
/*
//
```

Modelo 171. Ejecución de un módulo de carga procedente de ASSEMBLER, grabado en la biblioteca con el nombre NIVEL17.

```
//..... JOB ....
//PAS01 EXEC PGM=NIVEL17
//STEPLIB DD DSN=REALES.CNEA,DISP=SHR
/*
//IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//TARJETA DD *,DCB=BLKSIZE=80
Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: En el programa fuente en ASSEMBLER que dió origen al módulo de carga, se abrió un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

Modelo 172. Ejecución de un módulo de carga procedente de PL/1 grabado en la biblioteca con el nombre TANGENT.

```
//..... JOB ....
//PAS01 EXEC PGM=TANGENT
//STEPLIB DD DSN=REALES.CNEA,DISP=SHR
/*
//SYSPRINT DD SYSOUT=A
//SYSIN DD *
Aquí tarjetas perforadas de DATOS
/*
//
```

PUNTO 2.

Ejecución de programas que son módulos de carga grabados en una biblioteca de nombre REALES.CNEA e invocan rutinas y funciones residentes en la propia biblioteca, como módulos independientes.

Modelo 173 . Ejecución de un programa que es un módulo procedente de FORTRAN, reside en la biblioteca con el nombre FRAT e invoca rutinas y funciones que son módulos procedentes de FORTRAN o de ASSEMBLER, grabados en la propia biblioteca con los nombres FUNCDP1, FUNCDP2 y CATALISI.

```
//..... JOB ....
//PAS01 EXEC PROC=FTGILG
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
//LKED.SYSIN DD *,DCB=BLKSIZE=80
        INCLUDE OTRALIB(FRAT)
        INCLUDE OTRALIB(FUNCDP1)
        INCLUDE OTRALIB(FUNCDP2)
        INCLUDE OTRALIB(CATALISI)
/*
//GO.SYSIN DD *
Aquí tarjetas perforadas de DATOS
/*
//
```

Modelo 174 . Ejecución de un programa que es un módulo procedente de ASSEMBLER, reside en la biblioteca con el nombre RADIAT e invoca rutinas y funciones que son módulos procedentes de ASSEMBLER, grabados en la propia biblioteca con los nombres GLASS y ELECTRON.

```
//..... JOB ....
//PAS01 EXEC PROC=LKEDG
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
//LKED.SYSIN DD *,DCB=BLKSIZE=80
        INCLUDE OTRALIB(RADIAT)
        INCLUDE OTRALIB(GLASS)
        INCLUDE OTRALIB(ELECTRON)
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: En el programa principal ASSEMBLER que dió origen al módulo RADIAT, se abrió un archivo de entrada de datos de nombre TARJETA y un archivo de salida de

datos de nombre IMPRESO.

Modelo 175 . Ejecución de un programa que es un módulo procedente de ASSEMBLER, reside en la biblioteca con el nombre DENSITY e invoca rutinas y funciones que son módulos procedentes de FORTRAN, grabados en la propia biblioteca con los nombres INTERAC1, INTERAC2 y DEPEND.

```
//..... JOB ....
//PASO1 EXEC PROC=FTG1LG
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
//LKED.SYSIN DD *,DCB=BLKSIZE=80
        INCLUDE OTRALIB(DENSITY)
        INCLUDE OTRALIB(INTERAC1)
        INCLUDE OTRALIB(INTERAC2)
        INCLUDE OTRALIB(DEPEND)

/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
    Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: En el programa principal ASSEMBLER que dió origen al módulo DENSITY, se abrió un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

Modelo 176 . Ejecución de un programa que es un módulo procedente de PL/1, reside en la biblioteca con el nombre IMAN e invoca rutinas y funciones que son módulos procedentes de PL/1, grabados en la propia biblioteca con los nombres IMANRUT1 e IMANRUT2.

```
//..... JOB ....
//PASO1 EXEC PROC=PL1XLG
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
//LKED.SYSIN DD *,DCB=BLKSIZE=80
        INCLUDE OTRALIB(IMAN)
        INCLUDE OTRALIB(IMANRUT1)
        INCLUDE OTRALIB(IMANRUT2)

/*
//GO.SYSIN DD *
    Aquí tarjetas perforadas de DATOS
/*
//
```

PUNTO 3.

Ejecución de programas que son módulos de carga de una biblioteca de nombre REALES.CNEA e invocan rutinas y funciones en tarjetas perforadas.

Modelo 177 . Ejecución de un programa que es un módulo procedente de FORTRAN, reside en la biblioteca con el nombre MULTIPAR e invoca rutinas y funciones que se presentan en tarjetas perforadas en lenguaje simbólico FORTRAN. El problema se puede resolver en dos pasos: en el PASO1 el procedimiento FTG1CL compila y vincula las rutinas y funciones en FORTRAN cargando el módulo resultante, con el nombre RUTINAS, en la biblioteca REALES.CNEA; en el PASO2 el procedimiento FTG1LG vincula el módulo MULTIPAR con el módulo RUTINAS y ejecuta.

```
//..... JOB ....
//PASO1 EXEC PROC=FTG1CL
//FORT.SYSIN DD *
  Aquí tarjetas perforadas en FORTRAN
//LKED.SYSMOD DD DSN=REALES.CNEA(RUTINAS),SPACE=,DISP=OLD
/*
//PASO2 EXEC PROC=FTG1LG
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
//LKED.SYSIN DD *,DCB=BLKSIZE=80
      INCLUDE OTRALIB(MULTIPAR)
      INCLUDE OTRALIB(RUTINAS)
/*
//GO.SYSIN DD *
  Aquí tarjetas perforadas de DATOS
/*
//
```

Modelo 178 . Ejecución de un programa que es un módulo procedente de FORTRAN, reside en la biblioteca con el nombre JUSTIN e invoca rutinas y funciones que se presentan en tarjetas perforadas en lenguaje simbólico ASSEMBLER. El problema se puede resolver en dos pasos: en el PASO1 el procedimiento ASMFCL compila y vincula las rutinas y funciones en ASSEMBLER cargando el módulo resultante, con el nombre VARIAS, en la biblioteca REALES.CNEA; en el PASO2 el procedimiento FTG1LG vincula los módulos JUSTIN y VARIAS y ejecuta.

```
//..... JOB ....
//PASO1 EXEC PROC=ASMFCL
//ASM.SYSIN DD *
```

```

Aquí tarjetas perforadas en ASSEMBLER
//LKED.SYSLMOD DD DSN=REALES.CNEA(VARIAS),SPACE=,DISP=OLD
/*
//PASO2 EXEC PROC=FTG1LG
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
//LKED.SYSIN DD *,DCB=BLKSIZE=80
        INCLUDE OTRALIB(JUSTIN)
        INCLUDE OTRALIB(VARIAS)
/*
//GO.SYSIN DD *
Aquí tarjetas perforadas de DATOS
/*
//

```

Modelo 179 . Ejecución de un programa que es un módulo procedente de FORTRAN, reside en la biblioteca con el nombre RESTAURA e invoca rutinas y funciones que se presentan en tarjetas perforadas en absoluto de FORTRAN o de ASSEMBLER.

```

//..... JOB ....
//PASO1 EXEC PROC=FTG1LG
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
//LKED.SYSIN DD *,DCB=BLKSIZE=80
        INCLUDE OTRALIB(RESTAURA)
Aquí tarjetas perforadas en absoluto de FORTRAN o de ASSEMBLER
/*
//GO.SYSIN DD *
Aquí tarjetas perforadas de DATOS
/*
//

```

Modelo 180 . Ejecución de un programa que es un módulo procedente de ASSEMBLER, reside en la biblioteca con el nombre METRICA e invoca rutinas y funciones que se presentan en tarjetas perforadas en lenguaje simbólico ASSEMBLER. El problema se puede resolver en dos pasos: en el PASO1 el procedimiento ASMFCL compila y vincula las rutinas y funciones en ASSEMBLER cargando el módulo resultante, con el nombre EFECTO, en la biblioteca REALES.CNEA; en el PASO2 el procedimiento LKEDG vincula los módulos METRICA y EFECTO y ejecuta.

```

//..... JOB ....
//PASO1 EXEC PROC=ASMFCL
//ASM.SYSIN DD *
Aquí tarjetas perforadas en ASSEMBLER
//LKED.SYSLMOD DD DSN=REALES.CNEA(EFECTO),SPACE=,DISP=OLD
/*
//PASO2 EXEC PROC=LKEDG
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR

```

```
//LKED.SYSIN DD *,DCB=BLKSIZE=80
      INCLUDE OTRALIB(METRICA)
      INCLUDE OTRALIB(EFECTO)
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
      Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: En el programa principal ASSEMBLER que dió origen al módulo METRICA, se abrió un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

Modelo 181 . Ejecución de un programa que es un módulo procedente de ASSEMBLER, reside en la biblioteca con el nombre PERSISTE e invoca rutinas y funciones que se presentan en tarjetas perforadas en lenguaje simbólico FORTRAN. El problema se puede resolver en dos pasos: en el PASO1 el procedimiento FTG1CL compila y vincula las rutinas y funciones en FORTRAN cargando el módulo resultante, con el nombre FUMIGAR, en la biblioteca REALES.CNEA; en el PASO2 el procedimiento FTG1LG vincula los módulos PERSISTE y FUMIGAR y ejecuta.

```
//..... JOB ....
//PASO1 EXEC PROC=FTG1CL
//FORT.SYSIN DD *
      Aquí tarjetas perforadas en FORTRAN
//LKED.SYSLMOD DD DSN=REALES.CNEA(FUMIGAR),SPACE=,DISP=OLD
/*
//PASO2 EXEC PROC=FTG1LG
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
//LKED.SYSIN DD *,DCB=BLKSIZE=80
      INCLUDE OTRALIB(PERSISTE)
      INCLUDE OTRALIB(FUMIGAR)
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
      Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: En el programa principal ASSEMBLER que dió origen al módulo PERSISTE, se abrió un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

Modelo 182 . Ejecución de un programa que es un módulo procedente de ASSEMBLER,

reside en la biblioteca con el nombre STOCHAS e invoca rutinas y funciones que se presentan en tarjetas perforadas en absoluto de ASSEMBLER.

```
//..... JOB ....
//PAS01 EXEC PROC=LKEDG
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
//LKED.SYSIN DD *,DCB=BLKSIZE=80
        INCLUDE OTRALIB(STOCHAS)
Aquí tarjetas perforadas en absoluto de ASSEMBLER
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=30
Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: En el programa principal ASSEMBLER que dió origen al módulo STOCHAS, se abrió un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

Modelo 183 . Ejecución de un programa que es un módulo procedente de ASSEMBLER, reside en la biblioteca con el nombre FREE e invoca rutinas y funciones que se presentan en tarjetas perforadas en absoluto de FORTRAN.

```
//..... JOB ....
//PAS01 EXEC PROC=FTG1LG
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
//LKED.SYSIN DD *,DCB=BLKSIZE=80
        INCLUDE OTRALIB(FREE)
Aquí tarjetas perforadas en absoluto de FORTRAN
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: En el programa principal ASSEMBLER que dió origen al módulo FREE, se abrió un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

Modelo 184 . Ejecución de un programa que es un módulo procedente de PL/1, reside en la biblioteca con el nombre MODULOS e invoca una rutina que se presenta en tarjetas perforadas en lenguaje simbólico PL/1. El problema se puede resolver en

dos pasos: en el PASO1 el procedimiento PLIXCL compila y vincula la rutina cargando el módulo resultante, con el nombre NODRUTIN, en la biblioteca REALES.CNEA; en el PASO2 el procedimiento PLIXLG vincula el módulo NODULOS con el módulo NODRUTIN y ejecuta.

```
//..... JOB ....
//PASO1 EXEC PROC=PLIXCL,PARM.PLI='CS(60)'
//PLI.SYSIN DD *
  Aquí tarjetas perforadas en PL/1
//LKED.SYSLMOD DD DSN=REALES.CNEA(NODRUTIN),SPACE=,DISP=OLD
/*
//PASO2 EXEC PROC=PLIXLG
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
//LKED.SYSIN DD *,DCB=BLKSIZE=80
      INCLUDE OTRALIB(NODULOS)
      INCLUDE OTRALIB(NODRUTIN)
/*
//GO.SYSIN DD *
  Aquí tarjetas perforadas de DATOS
/*
//
```


PUNTO 4.

Ejecución de programas que son módulos de carga de una biblioteca de nombre REALES.CNEA e invocan rutinas y funciones grabadas en cintas magnéticas.

Modelo 185 . Ejecución de un programa que es un módulo procedente de FORTRAN, reside en la biblioteca con el nombre TALISMAN e invoca rutinas y funciones en lenguaje simbólico FORTRAN, grabadas con el nombre SUCUMB en el rótulo 25 de una cinta magnética que lleva el número de carrete 39988 y tiene etiquetas modelo IBM. El problema se puede resolver en dos pasos: en el PASO1 el procedimiento FTG1CL compila y vincula las rutinas y funciones en FORTRAN, cargando el módulo resultante en la biblioteca REALES.CNEA; en el PASO2 el procedimiento FTG1LG vincula los módulos TALISMAN y SUCUMB y ejecuta.

```
//..... JOB ....
//PASO1 EXEC PROC=FTG1CL
//FORT.SYSIN DD UNIT=TAPE,VOL=SER=T39988,DSN=SUCUMB,LABEL=(25,SL),
// DISP=(OLD,KEEP)
//LKED.SYSLMOD DD DSN=REALES.CNEA(SUCUMB),SPACE=,DISP=OLD
/*
//PASO2 EXEC PROC=FTG1LG
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
//LKED.SYSIN DD *,DCB=BLKSIZE=80
        INCLUDE OTRALIB(TALISMAN)
        INCLUDE OTRALIB(SUCUMB)
/*
//GO.SYSIN DD *
Aquí tarjetas perforadas de DATOS
/*
//
```

Modelo 186 . Ejecución de un programa que es un módulo procedente de FORTRAN, reside en la biblioteca con el nombre CLIMAT e invoca rutinas y funciones en lenguaje simbólico ASSEMBLER, grabadas con el nombre DIAGONAL en el rótulo 12 de una cinta magnética que lleva el número de carrete 05466 y tiene etiquetas modelo IBM. El problema se puede resolver en dos pasos: en el PASO1 el procedimiento ASMFCL compila y vincula las rutinas y funciones en ASSEMBLER, cargando el módulo resultante en la biblioteca REALES.CNEA; en el PASO2 el procedimiento FTG1LG vincula los módulos CLIMAT y DIAGONAL y ejecuta.

```
//..... JOB ....
//PAS01 EXEC PROC=ASMFCL
//ASM.SYSIN DD UNIT=TAPE,VOL=SER=T05466,DSN=DIAGONAL,LABEL=(12,SL),
// DISP=(OLD,KEEP)
//LKED.SYSLMOD DD DSN=REALES.CNEA(DIAGONAL),SPACE=,DISP=OLD
/*
//PAS02 EXEC PROC=FTG1LG
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
//LKED.SYSIN DD *,DCB=BLKSIZE=80
        INCLUDE OTRALIB(CLIMAT)
        INCLUDE OTRALIB(DIAGONAL)
/*
//GO.SYSIN DD *
Aquí tarjetas perforadas de DATOS
/*
//
```

Modelo 187 . Ejecución de un programa que es un módulo procedente de FORTRAN, reside en la biblioteca con el nombre GATEF e invoca rutinas y funciones en lenguaje absoluto de FORTRAN o de ASSEMBLER, grabadas con el nombre GATERUT en el rótulo 30 de una cinta magnética que lleva el número de carrete 02555 y tiene etiquetas modelo IBM. El problema se puede resolver en dos pasos: en el PAS01 el procedimiento LKED vincula las rutinas y funciones en lenguaje absoluto, cargando el módulo resultante en una biblioteca temporaria particionada que se crea en ese momento y reside en un disco auxiliar; en el PAS02 el procedimiento FTG1LG vincula los módulos GATEF y GATERUT y ejecuta.

```
//..... JOB ....
//PAS01 EXEC PROC=LKED
//LKED.SYSLMOD DD DSN=TEMPOR.ADR(GATERUT),UNIT=WORKDISK,
// DISP=(NEW,PASS),DCB=BLKSIZE=80,SPACE=(CYL,(1,1,1))
//LKED.SYSIN DD DSN=GATERUT,UNIT=TAPE,VOL=SER=T02555,LABEL=(30,SL),
// DISP=(OLD,KEEP)
/*
//PAS02 EXEC PROC=FTG1LG
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
//          DD DSN=TEMPOR.ADR,DISP=(OLD,DELETE)
//LKED.SYSIN DD *,DCB=BLKSIZE=80
        INCLUDE OTRALIB(GATEF)
        INCLUDE OTRALIB(GATERUT)
/*
//GO.SYSIN DD *
Aquí tarjetas perforadas de DATOS
/*
//
```

Modelo 188 . Ejecución de un programa que es un módulo procedente de ASSEMBLER, reside en la biblioteca con el nombre PRASEOD e invoca rutinas y funciones en

lenguaje simbólico ASSEMBLER, grabadas con el nombre CLOSEPAC en el rótulo 17 de una cinta magnética que lleva el número de carrete 10247 y tiene etiquetas modelo IBM. El problema se puede resolver en dos pasos: en el PASO1 el procedimiento ASMFCL compila y vincula las rutinas y funciones en ASSEMBLER, cargando el módulo resultante en la biblioteca REALES.CNEA; en el PASO2 el procedimiento LKEDG vincula el módulo PRASEOD con el módulo CLOSEPAC y ejecuta.

```
//..... JOB ....
//PASO1 EXEC PROC=ASMFCL
//ASM.SYSIN DD DSN=CLOSEPAC,UNIT=TAPE,VOL=SER=T10247,LABEL=(17,SL),
//  DISP=(OLD,KEEP)
//LKED.SYSLMOD LD DSN=REALES.CNEA(CLOSEPAC),SPACE=,DISP=OLD
/*
//PASO2 EXEC PROC=LKEDG
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
//LKED.SYSIN DD *,DCB=BLKSIZE=80
        INCLUDE OTRALIB(PRASEOD)
        INCLUDE OTRALIB(CLOSEPAC)
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
    Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: En el programa principal ASSEMBLER que dió origen al módulo PRASEOD, se abrió un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

Modelo 189 . Ejecución de un programa que es un módulo procedente de ASSEMBLER, reside en la biblioteca con el nombre DOUBLET e invoca rutinas y funciones en lenguaje simbólico FORTRAN, grabadas con el nombre EXCITAT en el rótulo 29 de una cinta magnética que lleva el número de carrete 63780 y tiene etiquetas modelo IBM. El problema se puede resolver en dos pasos: en el PASO1 el procedimiento FTG1CL compila y vincula las rutinas y funciones en FORTRAN, cargando el módulo resultante en la biblioteca REALES.CNEA; en el PASO2 el procedimiento FTG1LG vincula el módulo DOUBLFT con el módulo EXCITAT y ejecuta.

```
//..... JOB ....
//PASO1 EXEC PROC=FTG1CL
```

```
//FORT.SYSIN DD DSN=EXCITAT,UNIT=TAPE,VOL=SER=T63780,LABEL=(29,SL),
// DISP=(OLD,KEEP)
//LKED.SYSLMOD DD DSN=REALES.CNEA(EXCITAT),SPACE=,DISP=OLD
/*
//PASO2 EXEC PROC=FTGILG
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
//LKED.SYSIN DD *,DCB=BLKSIZE=80
        INCLUDE OTRALIB(DOUBLET)
        INCLUDE OTRALIB(EXCITAT)
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: En el programa principal ASSEMBLER que dió origen al módulo DOUBLET, se abrió un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

Modelo 190 . Ejecución de un programa que es un módulo procedente de ASSEMBLER, reside en la biblioteca con el nombre PARAMAGN e invoca rutinas y funciones en lenguaje absoluto de ASSEMBLER, grabadas con el nombre SINGLET en el rótulo 19 de una cinta magnética que lleva el número de carrete 64221 y tiene etiquetas modelo IBM. El problema se puede resolver en dos pasos: en el PASO1 el procedimiento LKED vincula las rutinas y funciones en absoluto de ASSEMBLER, cargando el módulo resultante en una biblioteca temporaria particionada que se crea en ese momento y reside en un disco auxiliar; en el PASO2 el procedimiento LKEDG vincula el módulo PARAMAGN con el módulo SINGLET y ejecuta.

```
//..... JOB ....
//PASO1 EXEC PROC=LKED
//LKED.SYSLMOD DD DSN=TEMPOR.ADR(SINGLET),UNIT=WORKDISK,
// DISP=(NEW,PASS),DCB=BLKSIZE=80,SPACE=(CYL,(1,1,1))
//LKED.SYSIN DD DSN=SINGLET,UNIT=TAPE,VOL=SER=T64221,LABEL=(19,SL),
// DISP=(OLD,KEEP)
/*
//PASO2 EXEC PROC=LKEDG
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
//          DD DSN=TEMPOR.ADR,DISP=(OLD,DELETE)
//LKED.SYSIN DD *,DCB=BLKSIZE=80
        INCLUDE OTRALIB(PARAMAGN)
        INCLUDE OTRALIB(SINGLET)
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
```

```
//GO.TARJETA DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: En el programa principal ASSEMBLER que dió origen al módulo PARAMAGN, se abrió un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

Modelo 191. Ejecución de un programa que es un módulo procedente de ASSEMBLER, reside en la biblioteca con el nombre COUPLING e invoca rutinas y funciones en lenguaje absoluto de FORTRAN, grabadas con el nombre WEAKMODL en el rótulo 38 de una cinta que lleva el número de carrete 81923 y tiene etiquetas modelo IBM. El problema se puede resolver en dos pasos: previamente se define un procedimiento de nombre DRLKDFTR; en el PASO1 se invoca dicho procedimiento que vincula las rutinas y funciones en absoluto de FORTRAN, cargando el módulo resultante en una biblioteca temporaria particionada que se crea en ese momento y reside en un disco auxiliar; en el PASO2 el procedimiento FTG1LG vincula el módulo COUPLING con el módulo WEAKMODL y ejecuta.

```
//..... JOB ....
//DRLKDFTR PROC
//LKED EXEC PGM=IEWL,PARM=(XREF,LIST,LET)
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=&SYSUT1,SPACE=(3520,(80,80),,,ROUND),UNIT=SYSDA
//SYSLMOD DD DSN=&&LOD(X),UNIT=SYSDA,DISP=(,PASS),
// SPACE=(CYL,(1,1,1)),DCB=(BLKSIZE=13030,LRECL=1024,RECFM=U)
//SYSLIB DD DSNNAME=SYS1.FORTLIB,DISP=SHR
//SYSLIN DD DDNAME=SYSIN,DCB=BLKSIZE=80
// PEND
//PASO1 EXEC PROC=DRLKDFTR
//LKED.SYSLMOD DD DSN=TEMPOR.ADR(WEAKMODL),UNIT=WORKDISK,
// DISP=(NEW,PASS),DCB=BLKSIZE=80,SPACE=(CYL,(1,1,1))
//LKED.SYSIN DD DSN=WEAKMODL,UNIT=TAPE,VOL=SER=T81923,
// DISP=(OLD,KEEP),LABEL=(38,SL)
/*
//PASO2 EXEC PROC=FTG1LG
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
// DD DSN=TEMPOR.ADR,DISP=(OLD,DELETE)
//LKED.SYSIN DD *,DCB=BLKSIZE=80
// INCLUDE OTRALIB(COUPLING)
// INCLUDE OTRALIB(WEAKMODL)
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
```

```
//GO.TARJETA DD *,DCB=BLKSIZE=80
Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: En el programa principal ASSEMBLER que dió origen al módulo COUPLING, se abrió un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

Modelo 192. Ejecución de un programa que es un módulo procedente de PL/1, reside en la biblioteca con el nombre DIODO e invoca una rutina en lenguaje simbólico PL/1, grabada con el nombre DIODRUT en el rótulo 13 de una cinta magnética que lleva el número de carrete 00210 y tiene etiquetas modelo IBM. El problema se puede resolver en dos pasos: en el PASO1 el procedimiento PLIXCL compila y vincula la rutina cargando el módulo resultante en la biblioteca REALES.CNEA; en el PASO2 el procedimiento PLIXLG vincula los módulos DIODO y DIODRUT y ejecuta.

```
//..... JOB ....
//PASO1 EXEC PROC=PLIXCL,PARM.PLI='CS(60)'
//PLI.SYSIN DD UNIT=TAPE,VOL=SER=T00210,DSN=DIODRUT,LABEL=(13,SL),
// DISP=(OLD,KEEP)
//LKED.SYSMOD DD DSN=REALES.CNEA(DIODO),SPACE=,DISP=OLD
/*
//PASO2 EXEC PROC=PLIXLG
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
//LKED.SYSIN DD *,DCB=BLKSIZE=80
        INCLUDE OTRALIB(DIODO)
        INCLUDE OTRALIB(DIODRUT)
/*
//GO.SYSIN DE *
Aquí tarjetas perforadas de DATOS
/*
//
```

PUNTO 5.

Uso de la Referencia hacia Atrás a un juego de datos ejecutado en un paso anterior de la misma tarea, para ser procesado en los pasos siguientes, por dos o mas módulos distintos de una biblioteca de nombre REALES.CNEA.

Modelo 193 . Ejecución de un módulo procedente de FORTRAN, grabado en la biblioteca con el nombre CURS016, que en el PAS01 resuelve el juego de DATOS1 y de otro módulo procedente de FORTRAN, grabado en la biblioteca con el nombre CURS017 que en el PAS02 hace referencia al mismo juego de DATOS1.

```
//..... JOB ....
//PAS01 EXEC PGM=CURS016
//STEPLIB DD DSN=REALES.CNEA,DISP=SHR
/*
//FT06F001 DD SYSOUT=A
//FT05F001 DD *
  Aquí tarjetas perforadas de DATOS1
/*
//PAS02 EXEC PGM=CURS017
//STEPLIB DD DSN=REALES.CNEA,DISP=SHR
/*
//FT06F001 DD SYSOUT=A
//FT05F001 DD DSN=DSNAME=*.PAS01.FT05F001,DISP=(OLD,PASS)
/*
//
```

Modelo 194 . Ejecución de un módulo procedente de ASSEMBLER, grabado en la biblioteca con el nombre GRADIENX, que en el PAS01 resuelve el juego de DATOS1 y de otro módulo procedente de ASSEMBLER, grabado en la biblioteca con el nombre GRADIENZ que en el PAS02 hace referencia al mismo juego de DATOS1.

```
//..... JOB ....
//PAS01 EXEC PGM=GRADIENX
//STEPLIB DD DSN=REALES.CNEA,DISP=SHR
/*
//IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//TARJETA DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas de DATOS1
/*
//PAS02 EXEC PGM=GRADIENZ
//STEPLIB DD DSN=REALES.CNEA,DISP=SHR
/*
//IMPRIMIR DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//LEER DD DSNAME=*.PAS01.TARJETA,DISP=(OLD,PASS)
/*
```

//

NOTA: En el programa principal ASSEMBLER que dió origen al módulo GRADIENX, se abrió un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO. En el programa principal ASSEMBLER que dió origen al módulo GRADIENZ se abrió un archivo de entrada de datos de nombre LEER y un archivo de salida de datos de nombre IMPRIMIR.

Modelo 195 . Ejecución de un módulo procedente de PL/1, grabado en la biblioteca con el nombre CARMAR1, que en el PASO1 resuelve el juego de DATOS1 y de otro módulo procedente de PL/1, grabado en la biblioteca con el nombre CARMAR2, que en el PASO2 hace referencia al mismo juego de DATOS1.

```
//..... JOB ....
//PASO1 EXEC PGM=CARMAR1
//STEPLIB DD DSN=REALES.CNEA,DISP=SHR
/*
//SYSPRINT DD SYSOUT=A
//SYSIN DD *
  Aquí tarjetas perforadas de DATOS1
/*
//PASO2 EXEC PGM=CARMAR2
//STEPLIB DD DSN=REALES.CNEA,DISP=SHR
/*
//SYSPRINT DD SYSOUT=A
//SYSIN DD DSNAME=*.PASO1.SYSIN,DISP=(OLD,PASS)
/*
//
```


PUNTO 6.

Ejecución de programas en tarjetas perforadas, que invocan rutinas y funciones residentes, como módulos de carga, en una biblioteca de nombre REALES.CNEA.

Modelo 196. Ejecución de un programa en tarjetas perforadas en lenguaje simbólico FORTRAN que invoca rutinas y funciones procedentes de FORTRAN o de ASSEMBLER, grabadas en la biblioteca como módulos independientes, con los nombres DEFLECT y COULOMB.

```
//..... JOB ....
//PAS01 EXEC PROC=FTG1CLG
//FORT.SYSIN DD *
  Aquí tarjetas perforadas en FORTRAN
/*
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
//LKED.SYSIN DD *,DCB=BLKSIZE=80
      INCLUDE OTRALIB(DEFLECT)
      INCLUDE OTRALIB(COULOMB)
/*
//GO.SYSIN DD *
  Aquí tarjetas perforadas de DATOS
/*
//
```

Modelo 197. Ejecución de un programa de tarjetas perforadas en lenguaje absoluto de FORTRAN que invoca rutinas y funciones procedentes de FORTRAN o de ASSEMBLER, grabadas en la biblioteca como módulos independientes, con los nombres FRACCION, ENTRADA, POTEN y FUNCLIB.

```
//..... JOB ....
//PAS01 EXEC PROC=FTG1LG
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
//LKED.SYSIN DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas en absoluto de FORTRAN
      INCLUDE OTRALIB(FRACCION)
      INCLUDE OTRALIB(ENTRADA)
      INCLUDE OTRALIB(POTEN)
      INCLUDE OTRALIB(FUNCLIB)
/*
//GO.SYSIN DD *
  Aquí tarjetas perforadas de DATOS
/*
//
```

Modelo 198. Ejecución de un programa en tarjetas perforadas en lenguaje simbólico

lico ASSEMBLER que invoca rutinas y funciones procedentes de ASSEMBLER o de FORTRAN, grabadas en la biblioteca como módulos independientes, con los nombres SPUTTERI, NIOBIO y PROTONS.

```
//..... JOB ....
//PAS01 EXEC PROC=ASMFCLC
//ASM.SYSIN DD *
  Aquí tarjetas perforadas en ASSEMBLER
/*
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
//LKED.SYSIN DD *,DCB=BLKSIZE=80
      INCLUDE OTRALIB(SPUTTERI)
      INCLUDE OTRALIB(NIOBIO)
      INCLUDE OTRALIB(PROTONS)
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: En el programa principal ASSEMBLER se ha abierto un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

Modelo 199 . Ejecución de un programa en tarjetas perforadas en lenguaje absoluto de ASSEMBLER que invoca rutinas y funciones procedentes de ASSEMBLER o de FORTRAN, grabadas en la biblioteca como módulos independientes, con los nombres SEPTUM, DIPOLE y MAGNET.

```
//..... JOB ....
//PAS01 EXEC PROC=LKEDG
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
//LKED.SYSIN DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas en absoluto de ASSEMBLER
      INCLUDE OTRALIB(SEPTUM)
      INCLUDE OTRALIB(DIPOLE)
      INCLUDE OTRALIB(MAGNET)
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: En el programa principal ASSEMBLER que dió origen al lote de tarjetas en absoluto, se abrió un archivo de entrada de datos de nombre TARJETA y un archivo

de salida de datos de nombre IMPRESO.

Modelo 200. Ejecución de un programa en tarjetas perforadas en lenguaje simbólico PL/1 que invoca rutinas y funciones procedentes de PL/1, grabadas en la biblioteca como módulos independientes, con los nombres MEDIA, PROMED y BISECT.

```
//..... JOB ....
//PASO1 EXEC PROC=PLIXCLG,PARM.PLI='CS(60)'
//PLI.SYSIN DD *
  Aquí tarjetas perforadas en PL/1
/*
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
//LKED.SYSIN DD *,DCB=BLKSIZE=80
      INCLUDE OTRALIB(MEDIA)
      INCLUDE OTRALIB(PROMED)
      INCLUDE OTRALIB(BISECT)
/*
//GO.SYSIN DD *
  Aquí tarjetas perforadas de DATOS
/*
//
```

PUNTO 7.

Ejecución de programas residentes en cintas magnéticas, que invocan rutinas y funciones grabadas como módulos de carga en una biblioteca de nombre REALES.CNEA.

Modelo 201 . Ejecución de un programa en lenguaje simbólico FORTRAN residente en una cinta magnética, que invoca rutinas y funciones procedentes de FORTRAN o de ASSEMBLER, grabadas en la biblioteca como módulos independientes, con los nombres MESONPI, LEPTON y GAMMA. El programa, de nombre PARTIC, está grabado en el rótulo 3 de una cinta que tiene etiquetas modelo IBM y lleva el número de carrete 52329.

```
//..... JOB ....
//PAS01 EXEC PROC=FTG1CLG
//FORT.SYSIN DD DSN=PARTIC,UNIT=TAPE,VOL=SER=T52329,LABEL=(3,SL),
// DISP=(OLD,KEEP)
/*
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
//LKED.SYSIN DD *,DCB=BLKSIZE=80
        INCLUDE OTRALIB(MESONPI)
        INCLUDE OTRALIB(LEPTON)
        INCLUDE OTRALIB(GAMMA)
/*
//GO.SYSIN DD *
Aquí tarjetas perforadas de DATOS
/*
//
```

Modelo 202 . Ejecución de un programa en lenguaje absoluto de FORTRAN residente en una cinta magnética, que invoca rutinas y funciones procedentes de FORTRAN o de ASSEMBLER, grabadas en la biblioteca como módulos independientes, con los nombres TABLES y RATES. El programa, de nombre THERMNUC, está grabado en el rótulo 14 de una cinta que tiene etiquetas modelo IBM y lleva el número de carrete 60022. El problema se puede resolver en dos pasos: en el PAS01 el procedimiento LKED vincula el programa THERMNUC grabando el módulo resultante en una biblioteca temporaria particionada que se crea en ese momento y reside en un disco auxiliar; en el PAS02 el procedimiento FTG1LG vincula el módulo THERMNUC con los módulos TABLES y RATES y ejecuta.

```
//..... JOB ....
//PAS01 EXEC PROC=LKED
```

```
//LKED.SYSLMOD DD DSN=TEMPOR.ADR(THERMNUC),UNIT=WORKDISK,
//  DISP=(NEW,PASS),DCB=BLKSIZE=80,SPACE=(CYL,(1,1,1))
//LKED.SYSIN DD DSN=THERMNUC,UNIT=TAPE,VOL=SER=T60022,LABEL=(14,SL),
//  DISP=(OLD,KEEP)
/*
//PASO2 EXEC PROC=FTG1LG
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
//          DD DSN=TEMPOR.ADR,DISP=(OLD,DELETE)
//LKED.SYSIN DD *,DCB=BLKSIZE=80
//          INCLUDE OTRALIB(THERMNUC)
//          INCLUDE OTRALIB(TABLES)
//          INCLUDE OTRALIB(RATES)
/*
//GO.SYSIN DD *
//  Aquí tarjetas perforadas de DATOS
/*
//
```

Modelo 203 . Ejecución de un programa en lenguaje simbólico ASSEMBLER residente en una cinta magnética, que invoca rutinas y funciones procedentes de FORTRAN o de ASSEMBLER, grabadas en la biblioteca como módulos independientes, con los nombres INTENSIT, MASS y SHIFT. El programa, de nombre SYMMETRY, está grabado en el rótulo 18 de una cinta magnética que tiene etiquetas modelo IBM y cuyo número de carrete es 45346.

```
//..... JOB ....
//PASO1 EXEC PROC=ASMFLG
//ASM.SYSIN DD DSN=SYMMETRY,UNIT=TAPE,VOL=SER=T45346,LABEL=(18,SL),
//  DISP=(OLD,KEEP)
/*
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
//LKED.SYSIN DD *,DCB=BLKSIZE=80
//          INCLUDE OTRALIB(INTENSIT)
//          INCLUDE OTRALIB(MASS)
//          INCLUDE OTRALIB(SHIFT)
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
//  Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: En el programa principal ASSEMBLER se ha abierto un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

Modelo 204 . Ejecución de un programa en lenguaje absoluto de ASSEMBLER residente en una cinta magnética, que invoca rutinas y funciones procedentes de

ASSEMBLER, grabadas en la biblioteca como módulos independientes, con los nombres COMPARO1 y COMPARO2. El programa, de nombre DRCOMPAR, está grabado en el rótulo 50 de una cinta magnética que tiene etiquetas modelo IBM y lleva el número de carrete 84782. El problema se puede resolver en dos pasos. en el PASO1 el procedimiento LKED vincula el programa DRCOMPAR cargando el módulo resultante en una biblioteca temporaria particionada que se crea en ese momento y reside en un disco auxiliar; en el PASO2 el procedimiento LKEDG vincula los módulos DRCOMPAR, COMPARO1 y COMPARO2 y ejecuta.

```
//..... JOB ....
//PASO1 EXEC PROC=LKED
//LKED.SYSLMOD DD DSN=TEMPOR.ADR(DRCOMPAR),UNIT=WORKDISK,
//  DISP=(NEW,PASS),DCB=BLKSIZE=80,SPACE=(CYL,(1,1,1))
//LKED.SYSIN DD DSN=DRCOMPAR,UNIT=TAPE,VOL=SER=T84782,LABEL=(50,SL),
//  DISP=(OLD,KEEP)
/*
//PASO2 EXEC PROC=LKEDG
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
//          DD DSN=TEMPOR.ADR,DISP=(OLD,DELETE)
//LKED.SYSIN DD *,DCB=BLKSIZE=80
//          INCLUDE OTRALIB(DRCOMPAR)
//          INCLUDE OTRALIB(COMPARO1)
//          INCLUDE OTRALIB(COMPARO2)
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
//  Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: En el programa principal ASSEMBLER que dió origen al módulo DRCOMPAR grabado en la cinta, se abrió un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

Modelo 205 . Ejecución de un programa en lenguaje absoluto de ASSEMBLER residente en una cinta magnética, que invoca rutinas y funciones procedentes de FORTRAN, grabadas en la biblioteca como módulos independientes, con los nombres PERT2 y PERT3. El programa, de nombre CAMICRIT, está grabado en el rótulo 22 de una cinta que tiene etiquetas modelo IBM y lleva el número de carrete 71020. El problema se puede resolver en dos pasos: en el PASO1 el procedimiento LKED vincula el programa CAMICRIT cargando el módulo resultante en una biblioteca temporaria particio-

nada que se crea en ese momento y reside en un disco auxiliar; en el PASO2 el procedimiento FTG1LG vincula los módulos CAMICRIT, PERT2 y PART3 y ejecuta.

```
//..... JOB ....
//PASO1 EXEC PROC=LKED
//LKED.SYSLMOD DD DSN=TEMPOR.ADR(CAMICRIT),UNIT=WORKDISK,
// DISP=(NEW,PASS),DCB=BLKSIZE=80,SPACE=(CYL,(1,1,1))
//LKED.SYSIN DD DSN=CAMICRIT,UNIT=TAPE,VOL=SER=T71020,DISP=(OLD,KEEP),
// LABEL=(22,SL)
/*
//PASO2 EXEC PROC=FTG1LG
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
//          DD DSN=TEMPOR.ADR,DISP=(OLD,DELETE)
//LKED.SYSIN DD *,DCB=BLKSIZE=80
//          INCLUDE OTRALIB(CAMICRIT)
//          INCLUDE OTRALIB(PERT2)
//          INCLUDE OTRALIB(PERT3)
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
//Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: En el programa principal ASSEMBLER que dió origen al módulo CAMICRIT grabado en la cinta, se abrió un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

Modelo 206 . Ejecución de un programa en lenguaje simbólico PL/1 residente en una cinta magnética, que invoca rutinas y funciones procedentes de PL/1, grabadas en la biblioteca como módulos independientes, con los nombres MIRRELAT, MIRABSOL y MIRTOTAL. El programa, de nombre MIR, está grabado en el rótulo 14 de una cinta que tiene etiquetas modelo IBM y cuyo número de carrete es 41037.

```
//..... JOB ....
//PASO1 EXEC PROC=PLIXCLG,PARM.PLI='CS(60)'
//PLI.SYSIN DD DSN=MIR,UNIT=TAPE,VOL=SER=T41037,LABEL=(14,SL),
// DISP=(OLD,KEEP)
/*
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
//LKED.SYSIN DD *,DCB=BLKSIZE=80
//          INCLUDE OTRALIB(MIRRELAT)
//          INCLUDE OTRALIB(MIRABSOL)
//          INCLUDE OTRALIB(MIRTOTAL)
/*
//GO.SYSIN DD *
//Aquí tarjetas perforadas de DATOS
/*
//
```

PUNTO 8.

Grabación de programas, rutinas o funciones, que se presentan en tarjetas perforadas en lenguajes simbólicos, en una biblioteca de módulos de carga de nombre REALES.CNEA.

Modelo 207 . Grabación en la biblioteca, como módulos independientes, de un programa completo con el nombre KUNZ, de una rutina con el nombre KATER y de una función con el nombre RAICES. Las tarjetas están perforadas en lenguaje simbólico FORTRAN. El problema se puede resolver en tres pasos con el procedimiento FTG1CL.

```
//..... JOB ....
//PASO1 EXEC PROC=FTG1CL
//FORT.SYSIN DD *
  Aquí tarjetas perforadas en FORTRAN
//LKED.SYSLMOD DD DSN=REALES.CNEA(KUNZ),SPACE=,DISP=OLD
/*
//PASO2 EXEC PROC=FTG1CL
//FORT.SYSIN DD *
  Aquí tarjetas perforadas en FORTRAN
//LKED.SYSLMOD DD DSN=REALES.CNEA(KATER),SPACE=,DISP=OLD
/*
//PASO3 EXEC PROC=FTG1CL
//FORT.SYSIN DD *
  Aquí tarjetas perforadas en FORTRAN
//LKED.SYSLMOD DD DSN=REALES.CNEA(RAICES),SPACE=,DISP=OLD
/*
//
```

Modelo 208 .Grabación en la biblioteca, como módulos independientes, de un programa completo con el nombre LISTA, de una rutina con el nombre MC03AS y de una función con el nombre TOTALASS. Las tarjetas están perforadas en lenguaje simbólico ASSEMBLER. El problema se puede resolver en tres pasos con el procedimiento ASMFCL.

```
//..... JOB ....
//PASO1 EXEC PROC=ASMFCL
//ASM.SYSIN DD *
  Aquí tarjetas perforadas en ASSEMBLER
//LKED.SYSLMOD DD DSN=REALES.CNEA(LISTA),SPACE=,DISP=OLD
/*
//PASO2 EXEC PROC=ASMFCL
//ASM.SYSIN DD *
  Aquí tarjetas perforadas en ASSEMBLER
//LKED.SYSLMOD DD DSN=REALES.CNEA(MC03AS),SPACE=,DISP=OLD
```



```

/*
//PASO3 EXEC PROC=ASMFCL
//ASM.SYSIN DD *
  Aquí tarjetas perforadas en ASSEMBLER
//LKED.SYSLMOD DD DSN=REALES.CNEA(TOTALASS),SPACE=,DISP=OLD
/*
//

```

Modelo 209 . Grabación en la biblioteca, como módulos independientes, de un programa completo con el nombre POLAR, de una rutina con el nombre POLARIT y de una función con el nombre MATRIX. Las tarjetas están perforadas en lenguaje simbólico PL/1. El problema se puede resolver en tres pasos con el procedimiento PLIXCL.

```

//..... JOB ....
//PASO1 EXEC PROC=PLIXCL,PARM.PLI='CS(60)'
//PLI.SYSIN DD *
  Aquí tarjetas perforadas en PL/1
//LKED.SYSLMOD DD DSN=REALES.CNEA(POLAR),SPACE=,DISP=OLD
/*
//PASO2 EXEC PROC=PLIXCL,PARM.PLI='CS(60)'
//PLI.SYSIN DD *
  Aquí tarjetas perforadas en PL/1
//LKED.SYSLMOD DD DSN=REALES.CNEA(POLARIT),SPACE=,DISP=OLD
/*
//PASO3 EXEC PROC=PLIXCL,PARM.PLI='CS(60)'
//PLI.SYSIN DD *
  Aquí tarjetas perforadas en PL/1
//LKED.SYSLMOD DD DSN=REALES.CNEA(MATRIX),SPACE=,DISP=OLD
/*
//

```

PUNTO 9.

Grabación de programas, rutinas o funciones, que están grabadas en cintas magnéticas en lenguajes simbólicos, en una biblioteca de módulos de carga de nombre REALES.CNEA.

Modelo 210. Grabación en la biblioteca, como módulos independientes, de un programa completo, de una rutina y de una función residentes, en lenguaje simbólico FORTRAN, en una cinta magnética que lleva el número de carrete 16158 y tiene etiquetas modelo IBM. El programa está grabado en el rótulo 3 con el nombre CAPACITY, la rutina en el rótulo 7 con el nombre ORTOGON y la función en el rótulo 24 con el nombre POLOS.

```
//..... JOB ....
//PAS01 EXEC PROC=FTG1CL
//FORT.SYSIN DD DSN=CAPACITY,UNIT=TAPE,VOL=SER=T16158,LABEL=(3,SL),
//  DISP=(OLD,PASS)
//LKED.SYSLMOD DD DSN=REALES.CNEA(CAPACITY),SPACE=,DISP=OLD
/*
//PAS02 EXEC PROC=FTG1CL
//FORT.SYSIN DD DSN=ORTOGON,UNIT=TAPE,VOL=SER=T16158,LABEL=(7,SL),
//  DISP=(OLD,PASS)
//LKED.SYSLMOD DD DSN=REALES.CNEA(ORTOGON),SPACE=,DISP=OLD
/*
//PAS03 EXEC PROC=FTG1CL
//FORT.SYSIN DD DSN=POLOS,UNIT=TAPE,VOL=SER=T16158,LABEL=(24,SL),
//  DISP=(OLD,KEEP)
//LKED.SYSLMOD DD DSN=REALES.CNEA(POLOS),SPACE=,DISP=OLD
/*
//
```

Modelo 211. Grabación en la biblioteca, como módulos independientes de un programa completo, de una rutina y de una función residentes, en lenguaje simbólico ASSEMBLER, en una cinta magnética que lleva el número de carrete 70001 y tiene etiquetas modelo IBM. El programa está grabado en el rótulo 18 con el nombre FORTIN, la rutina en el rótulo 25 con el nombre GRUPO3 y la función en el rótulo 26 con el nombre AMAT.

```
//..... JOB ....
//PAS01 EXEC PROC=ASMFCL
//ASM.SYSIN DD DSN=FORTIN,UNIT=TAPE,VOL=SER=T70001,LABEL=(18,SL),
//  DISP=(OLD,PASS)
```

```
//LKED.SYSLMOD DD DSN=REALES.CNEA(FORTIN),SPACE=,DISP=OLD
/*
//PASO2 EXEC PROC=ASMFCL
//ASM.SYSIN DD DSN=GRUPO3,UNIT=TAPE,VOL=SER=T70001,LABEL=(25,SL),
// DISP=(OLD,PASS)
//LKED.SYSLMOD DD DSN=REALES.CNEA(GRUPO3),SPACE=,DISP=OLD
/*
//PASO3 EXEC PROC=ASMFCL
//ASM.SYSIN DD DSN=AMAT,UNIT=TAPE,VOL=SER=T70001,LABEL=(26,SL),
// DISP=(OLD,KEEP)
//LKED.SYSLMOD DD DSN=REALES.CNEA(AMAT),SPACE=,DISP=OLD
/*
//
```

Modelo 212 . Grabación en la biblioteca, como módulos independientes, de un programa completo, de una rutina y de una función residentes, en lenguaje simbólico PL/1, en una cinta magnética que lleva el número de carrete 63271 y tiene etiquetas modelo IBM. El programa está grabado en el rótulo 12 con el nombre RUBIDIO, la rutina en el rótulo 27 con el nombre SOMALI y la función en el rótulo 30 con el nombre FUNC25.

```
//..... JOB ....
//PASO1 EXEC PROC=PLIXCL,PARM.PLI='CS(60)'
//PLI.SYSIN DD DSN=RUBIDIO,UNIT=TAPE,VOL=SER=T63271,LABEL=(12,SL),
// DISP=(OLD,PASS)
//LKED.SYSLMOD DD DSN=REALES.CNEA(RUBIDIO),SPACE=,DISP=OLD
/*
//PASO2 EXEC PROC=PLIXCL,PARM.PLI='CS(60)'
//PLI.SYSIN DD DSN=SOMALI,UNIT=TAPE,VOL=SER=T63271,LABEL=(27,SL),
// DISP=(OLD,PASS)
//LKED.SYSLMOD DD DSN=REALES.CNEA(SOMALI),SPACE=,DISP=OLD
/*
//PASO3 EXEC PROC=PLIXCL,PARM.PLI='CS(60)'
//PLI.SYSIN DD DSN=FUNC25,UNIT=TAPE,VOL=SER=T63271,LABEL=(30,SL),
// DISP=(OLD,KEEP)
//LKED.SYSLMOD DD DSN=REALES.CNEA(FUNC25),SPACE=,DISP=OLD
/*
//
```

PUNTO 10.

Corrección de módulos de carga grabados en una biblioteca de nombre REALES.

CNEA, cuyos programas fuente en lenguaje simbólico, se presentan en tarjetas perforadas.

Modelo 213 . Corrección de un módulo de carga, procedente de FORTRAN, grabado en la biblioteca con el nombre SUSTRAE. La corrección se realiza en el lote de tarjetas perforadas en lenguaje simbólico, que dió origen al módulo. El procedimiento FTG1CL carga el nuevo módulo en la biblioteca, sustituyendo al anterior con el mismo nombre.

```
//..... JOB ....
//PAS01 EXEC PROC=FTG1CL
//FORT.SYSIN DD *
  Aquí tarjetas perforadas en FORTRAN
//LKED.SYSLMOD DD DSN=REALES.CNEA(SUSTRAE),SPACE=,DISP=OLD
/*
//
```

Modelo 214 . Corrección de un módulo de carga, procedente de ASSEMBLER, grabado en la biblioteca con el nombre MODELO35. La corrección se realiza en el lote de tarjetas perforadas en lenguaje simbólico, que dió origen al módulo. El procedimiento ASMFCL carga el nuevo módulo en la biblioteca, sustituyendo al anterior con el mismo nombre.

```
//..... JOB ....
//PAS01 EXEC PROC=ASMFCL
//ASM.SYSIN DD *
  Aquí tarjetas perforadas en ASSEMBLER
//LKED.SYSLMOD DD DSN=REALES.CNEA(MODELO35),SPACE=,DISP=OLD
/*
//
```

Modelo 215 . Corrección de un módulo de carga, procedente de PL/1, grabado en la biblioteca con el nombre ATINA. La corrección se realiza en el lote de tarjetas perforadas en lenguaje simbólico, que dió origen al módulo. El procedimiento PLIXCL carga el nuevo módulo en la biblioteca, sustituyendo al anterior con el mismo nombre.

```
//..... JOB ....
```

```
//PASO1 EXEC PROC=PLIXCL,PARM.PLI='CS(60)'  
//PLI.SYSIN DD *  
    Aquí tarjetas perforadas en PL/1  
//LKED.SYSLMOD DD DSN=REALES.CNEA(ATINA),SPACE=,DISP=SHR  
/*  
//
```


CAPITULO V

MODELOS PARA EL USO SIMULTANEO DE DOS
BIBLIOTECAS PARTICIONADAS, UNA DE PROGRAMAS EN
LENGUAJE SIMBOLICO Y OTRA DE MODULOS DE CARGA

PUNTO 1.

Ejecución de programas en lenguaje simbólico, que están grabados en una biblioteca de nombre SIMBOLIC.CNEA e invocan rutinas y funciones residentes en una biblioteca de módulos de carga de nombre REALES.CNEA.

Modelo 216. Ejecución de un programa en lenguaje FORTRAN residente en la biblioteca de simbólicos con el nombre CLUB, que invoca rutinas y funciones procedentes de FORTRAN o de ASSEMBLER, grabadas en la biblioteca de módulos de carga con los nombres FORAN2, FORAN3 y FONON.

```
//..... JOB ....
//PAS01 EXEC PROC=FTG1CLG
//FORT.SYSIN DD DSN=SIMBOLIC.CNEA(CLUB),DISP=SHR
/*
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
//LKED.SYSIN DD *,DCB=BLKSIZE=80
        INCLUDE OTRALIB(FORAN2)
        INCLUDE OTRALIB(FORAN3)
        INCLUDE OTRALIB(FONON)
/*
//GO.SYSIN DD *
Aquí tarjetas perforadas de DATOS
/*
//
```

Modelo 217. Ejecución de un programa en lenguaje ASSEMBLER residente en la biblioteca de simbólicos con el nombre CAPDENEV, que invoca rutinas y funciones procedentes de FORTRAN o de ASSEMBLER, grabadas en la biblioteca de módulos de carga con los nombres GAMMA y VALENCIA.

```
//..... JOB ....
//PAS01 EXEC PROC=ASMFC LG
//ASM.SYSIN DD DSN=SIMBOLIC.CNEA(CAPDENEV),DISP=SHR
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
//LKED.SYSIN DD *,DCB=BLKSIZE=80
        INCLUDE OTRALIB(GAMMA)
        INCLUDE OTRALIB(VALENCIA)
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=153,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: En el programa principal ASSEMBLER se ha abierto un archivo de entrada de

datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

Modelo 218. Ejecución de un programa en lenguaje PL/1 residente en la biblioteca de simbólicos con el nombre GRISET, que invoca rutinas y funciones procedentes de PL/1, grabadas en la biblioteca de módulos de carga con los nombres UNCAR, DOLMEN, BASEN y TIR.

```
//..... JOB ....
//PASO1 EXEC PROC=PLIXCLG,PARM.PLI='CS(60)'
//PLI.SYSIN DD DSN=SIMBOLIC.CNEA(GRISET),DISP=SHR
/*
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
//LKED.SYSIN DD *,DCB=BLKSIZE=80
        INCLUDE OTRALIB(UNCAR)
        INCLUDE OTRALIB(DOLMEN)
        INCLUDE OTRALIB(BASEN)
        INCLUDE OTRALIB(TIR)
/*
//GO.SYSIN DD *
Aquí tarjetas perforadas de DATOS
/*
//
```

PUNTO 2.

Ejecución de programas en lenguaje simbólico, que están grabados en una biblioteca de nombre SIMBOLIC.CNEA e invocan rutinas y funciones residentes en una biblioteca de módulos de carga de nombre REALES.CNEA, a mas de otras rutinas y funciones que se presentan en tarjetas perforadas.

Modelo 219. Ejecución de un programa en lenguaje FORTRAN residente en la biblioteca de simbólicos con el nombre KOLAPS, que invoca rutinas y funciones procedentes de FORTRAN o de ASSEMBLER, grabadas en la biblioteca de módulos de carga con los nombres MODUL75 y MODUL76, a mas de otras rutinas y funciones que se presentan en tarjetas perforadas en lenguaje simbólico FORTRAN.

```
//..... JOB ....
//PASO1 EXEC PROC=FTG1CLG
//FORT.SYSIN DD DSN=SIMBOLIC.CNEA(KOLAPS),DISP=SHR
//          DD *
    Aquí tarjetas perforadas en FORTRAN
/*
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
//LKED.SYSIN DD *,DCB=BLKSIZE=80
           INCLUDE OTRALIB(MODUL75)
           INCLUDE OTRALIB(MODUL76)
/*
//GO.SYSIN DD *
    Aquí tarjetas perforadas de DATOS
/*
//
```

Modelo 220. Ejecución de un programa en lenguaje FORTRAN residente en la biblioteca de simbólicos con el nombre IMAGEN, que invoca rutinas y funciones procedentes de FORTRAN o de ASSEMBLER, grabadas en la biblioteca de módulos de carga con los nombres UTILIN y SAGITAR, a mas de otras rutinas y funciones que se presentan en tarjetas perforadas en lenguaje simbólico ASSEMBLER. El problema se puede resolver en dos pasos: en el PASO1 el procedimiento ASMFCL compila y vincula las rutinas y funciones en ASSEMBLER, grabando el módulo resultante, con el nombre ASEMRUT, en la biblioteca REALES.CNEA; en el PASO2 el procedimiento FTG1CLG ejecuta el programa IMAGEN vinculándolo con los módulos UTILIN, SAGITAR y ASEMRUT.

```
//..... JOB ....
```

```
//PAS01 EXEC PROC=ASMFCL
//ASM.SYSIN DD *
  Aquí tarjetas perforadas en ASSEMBLER
//LKED.SYSLMOD DD DSN=REALES.CNEA(ASEMRUT),SPACE=,DISP=OLD
/*
//PAS02 EXEC PROC=FTGICLG
//FORT.SYSIN DD DSN=SIMBOLIC.CNEA(IMAGEN),DISP=SHR
/*
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
//LKED.SYSIN DD *,DCB=BLKSIZE=80
      INCLUDE OTRALIB(UTILIN)
      INCLUDE OTRALIB(SAGITAR)
      INCLUDE OTRALIB(ASEMRUT)
/*
//GO.SYSIN DD *
  Aquí tarjetas perforadas de DATOS
/*
//
```

Modelo 221 . Ejecución de un programa en lenguaje FORTRAN residente en la biblioteca de simbólicos con el nombre SUTUR, que invoca rutinas y funciones procedentes de FORTRAN o de ASSEMBLER, grabadas en la biblioteca de módulos de carga con los nombres VIDAMED, NIVENERG, BORO y FLUOR, a mas de otras rutinas y funciones que se presentan en tarjetas perforadas en lenguaje absoluto de FORTRAN o de ASSEMBLER.

```
//..... JOB ....
//PAS01 EXEC PROC=FTGICLG
//FORT.SYSIN DD DSN=SIMBOLIC.CNEA(SUTUR),DISP=SHR
/*
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
//LKED.SYSIN DD *,DCB=BLKSIZE=80
      INCLUDE OTRALIB(VIDAMED)
      INCLUDE OTRALIB(NIVENERG)
      INCLUDE OTRALIB(BORO)
      INCLUDE OTRALIB(FLUOR)
  Aquí tarjetas perforadas en absoluto de FORTRAN o de ASSEMBLER
/*
//GO.SYSIN DD *
  Aquí tarjetas perforadas de DATOS
/*
//
```

Modelo 222 . Ejecución de un programa en lenguaje ASSEMBLER residente en la biblioteca de simbólicos con el nombre SCATT, que invoca rutinas y funciones procedentes de FORTRAN o de ASSEMBLER, grabadas en la biblioteca de módulos de carga con los nombres ELASTIC e INELAST, a mas de otras rutinas y funciones que se pre-

sentan en tarjetas perforadas en lenguaje simbólico ASSEMBLER. El problema se puede resolver en dos pasos: en el PASO1 el procedimiento ASMFCL compila y vincula las rutinas y funciones en ASSEMBLER, grabando el módulo resultante, con el nombre SUSPEN, en la biblioteca REALES.CNEA; en el PASO2 el procedimiento ASMFCLG ejecuta el programa SCATT vinculándolo con los módulos ELASTIC, INELAST y SUSPEN.

```
//..... JOB ....
//PASO1 EXEC PROC=ASMFCL
//ASM.SYSIN DD *
  Aquí tarjetas perforadas en ASSEMBLER
//LKED.SYSLMOD DD DSN=REALES.CNEA(SUSPEN),SPACE=,DISP=OLD
/*
//PASO2 EXEC PROC=ASMFCLG
//ASM.SYSIN DD DSN=SIMBOLIC.CNEA(SCATT),DISP=SHR
/*
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
//LKED.SYSIN DD *,DCB=BLKSIZE=80
      INCLUDE OTRALIB(ELASTIC)
      INCLUDE OTRALIB(INELAST)
      INCLUDE OTRALIB(SUSPEN)
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: En el programa principal ASSEMBLER se ha abierto un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

Modelo 223. Ejecución de un programa en lenguaje ASSEMBLER, residente en la biblioteca de simbólicos con el nombre STERN, que invoca rutinas y funciones procedentes de FORTRAN o de ASSEMBLER, grabadas en la biblioteca de módulos de carga con los nombres KLUG y WILD, a mas de otras rutinas y funciones que se presentan en tarjetas perforadas en lenguaje simbólico FORTRAN. El problema se puede resolver en dos pasos: en el PASO1 el procedimiento FTG1CL compila y vincula las rutinas y funciones en FORTRAN, grabando el módulo resultante, con el nombre LANG, en la biblioteca REALES.CNEA; en el PASO2 el procedimiento ASMFCLG ejecuta el programa STERN vinculándolo con los módulos KLUG, WILD y LANG.

```
//..... JOB ....
//PASO1 EXEC PROC=FTG1CL
```

```
//FORT.SYSIN DD *
  Aquí tarjetas perforadas en FORTRAN
//LKED.SYSLMOD DD DSN=REALES.CNEA(LANG),SPACE=,DISP=OLD
/*
//PASO2 EXEC PROC=ASMFCLG
//ASM.SYSIN DD DSN=SIMBOLIC.CNEA(STERM),DISP=SHR
/*
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
//LKED.SYSIN DD *,DCB=BLKSIZE=80
      INCLUDE OTRALIB(KLUG)
      INCLUDE OTRALIB(WILD)
      INCLUDE OTRALIB(LANG)
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas de DATOS
/*
/'
```

NOTA: En el programa principal ASSEMBLER se ha abierto un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

Modelo 224. Ejecución de un programa en lenguaje ASSEMBLER residente en la biblioteca de simbólicos con el nombre FIELD, que invoca rutinas y funciones procedentes de FORTRAN o de ASSEMBLER, grabadas en la biblioteca de módulos de carga con los nombres GAUGE, INVAR y OPERATOR, a mas de otras rutinas y funciones que se presentan en tarjetas perforadas en absoluto de ASSEMBLER.

```
//..... JOB ....
//PASO1 EXEC PROC=ASMFCLG
//ASM.SYSIN DD DSN=SIMBOLIC.CNEA(FIELD),DISP=SHR
/*
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
//LKED.SYSIN DD *,DCB=BLKSIZE=80
      INCLUDE OTRALIB(GAUGE)
      INCLUDE OTRALIB(INVAR)
      INCLUDE OTRALIB(OPERATOR)
  Aquí tarjetas perforadas en absoluto de ASSEMBLER
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: En el programa principal ASSEMBLER se ha abierto un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

PUNTO 3.

Ejecución de programas en lenguaje simbólico, que están grabados en una biblioteca de nombre SIMBOLIC.CNEA e invocan rutinas y funciones residentes en una biblioteca de módulos de carga de nombre REALES.CNEA, a mas de otras rutinas y funciones grabadas en cintas magnéticas.

Modelo 225 . Ejecución de un programa en lenguaje FORTRAN residente en la biblioteca de simbólicos con el nombre CRIPTO, que invoca rutinas y funciones procedentes de FORTRAN o de ASSEMBLER, grabadas en la biblioteca de módulos de carga con los nombres SALGAN, OFUSC y DARDAN, a mas de otras rutinas y funciones en lenguaje simbólico FORTRAN residentes en cinta magnética. Estas últimas están grabadas, con el nombre SOLBAR, en el rótulo 2 de una cinta que lleva el número de carrete 62751 y tiene etiquetas modelo IBM.

```
//..... JOB ....
//PASO1 EXEC PROC=FTG1CLG
//FORT.SYSIN DD DSN=SIMBOLIC.CNEA(CRIPTO),DISP=SHR
//          DD DSN=SOLBAR,UNIT=TAPE,VOL=SER=T62751,LABEL=(2,SL),
//  DISP=(OLD,KEEP)
/*
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
//LKED.SYSIN DD *,DCB=BLKSIZE=80
           INCLUDE OTRALIB(SALGAN)
           INCLUDE OTRALIB(OFUSC)
           INCLUDE OTRALIB(DARDAN)
/*
//GO.SYSIN DD *
Aquí tarjetas perforadas de DATOS
/*
//
```

Modelo 226 . Ejecución de un programa en lenguaje FORTRAN residente en la biblioteca de simbólicos con el nombre COMBUST, que invoca rutinas y funciones procedentes de FORTRAN o de ASSEMBLER, grabadas en la biblioteca de módulos de carga con los nombres RENOVAC y VAC, a mas de otras rutinas y funciones en lenguaje absoluto de FORTRAN o de ASSEMBLER residentes en cinta magnética. Estas últimas están grabadas con el nombre POLITAR, en el rótulo 45 de una cinta que lleva el número de carrete 15476 y tiene etiquetas modelo IBM. El problema se puede resolver

en dos pasos: en el PAS01 el procedimiento LKED vincula las rutinas y funciones en lenguaje absoluto, cargando el módulo resultante en una biblioteca temporaria particionada que se crea en ese momento y reside en un disco auxiliar; en el PAS02 el procedimiento FTG1CLG ejecuta el programa COMBUST vinculándolo con los módulos RENOVAC, VAC y POLITAR.

```
//..... JOB ....
//PAS01 EXEC PROC=LKED
//LKED.SYSLMOD DD DSN=TEMPOR.ADR(POLITAR),UNIT=WORKDISK,
//  DISP=(NEW,PASS),DCB=BLKSIZE=80,SPACE=(CYL,(1,1,1))
//LKED.SYSIN DD DSN=POLITAR,UNIT=TAPE,VOL=SER=T15476,LABEL=(45,SL),
//  DISP=(OLD,KEEP)
/*
//PAS02 EXEC PROC=FTG1CLG
//FORT.SYSIN DD DSN=SIMBOLIC.CNEA(COMBUST),DISP=SHR
/*
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
//          DD DSN=TEMPOR.ADR,DISP=(OLD,DELETE)
//LKED.SYSIN DD *,DCB=BLKSIZE=80
//          INCLUDE OTRALIB(REN OVAC),
//          INCLUDE OTRALIB(VAC)
//          INCLUDE OTRALIB(POLITAR)
/*
//GO.SYSIN DD *
//  Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: Se menciona el DCB con el cual se grabará el disco auxiliar.

Modelo 227. Ejecución de un programa en lenguaje FORTRAN residente en la biblioteca de simbólicos con el nombre TURBINA, que invoca rutinas y funciones procedentes de FORTRAN o de ASSEMBLER, grabadas en la biblioteca de módulos de carga con los nombres REACTOR, VANTE y DOBLET, a mas de otras rutinas y funciones en lenguaje simbólico ASSEMBLER residentes en cinta magnética. Estas últimas están grabadas, con el nombre TRIPLET, en el rótulo 15 de una cinta que lleva el número de carrete 20165 y tiene etiquetas modelo IBM. El problema se puede resolver en dos pasos: en el PAS01 el procedimiento ASMFCL compila y vincula las rutinas y funciones en ASSEMBLER de la cinta magnética, cargando el módulo resultante en la biblioteca REALES.CNEA: en el PAS02 el procedimiento FTG1CLG ejecuta el programa TURBINA vinculándolo con los módulos REACTOR, VANTE, DOBLET y TRIPLET.

```
//..... JOB ....
//PASO1 EXEC PROC=ASMFCL
//ASM.SYSIN DD DSN=TRIPLET,UNIT=TAPE,VOL=SER=T20165,
// LABEL=(15,SL),DISP=(OLD,KEEP)
//LKED.SYSLMOD DD DSN=REALES.CNEA(TRIPLET),SPACE=,DISP=OLD
/*
//PASO2 EXEC PROC=FTG1CLG
//FORT.SYSIN DD DSN=SIMBOLIC.CNEA(TURBINA),DISP=SHR
/*
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
//LKED.SYSIN DD *,DCB=BLKSIZE=80
        INCLUDE OTRALIB(REACTOR)
        INCLUDE OTRALIB(VANTE)
        INCLUDE OTRALIB(DOBLLET)
        INCLUDE OTRALIB(TRIPLET)
/*
//GO.SYSIN DD *
    Aquí tarjetas perforadas de DATOS
/*
//
```

Modelo 228. Ejecución de un programa en lenguaje ASSEMBLER residente en la biblioteca de simbólicos con el nombre NUCLMATT, que invoca rutinas y funciones procedentes de FORTRAN o de ASSEMBLER, grabadas en la biblioteca de módulos de carga con los nombres MIXING, PARITY y WEAK, a mas de otras rutinas y funciones en lenguaje simbólico ASSEMBLER residentes en cinta magnética. Estas últimas están grabadas, con el nombre NUCLINT, en el rótulo 18 de una cinta que lleva el número de carrete 52183 y tiene etiquetas modelo IBM. El problema se puede resolver en dos pasos: en el PASO1 el procedimiento ASMFCL compila y vincula las rutinas y funciones en ASSEMBLER que residen en la cinta, grabando el módulo resultante en la biblioteca REALES.CNEA; en el PASO2 el procedimiento ASMFCLG ejecuta el programa NUCLMATT vinculándolo con los módulos NUCLINT, MIXING, PARITY y WEAK.

```
//..... JOB ....
//PASO1 EXEC PROC=ASMFCL
//ASM.SYSIN DD DSN=NUCLINT,UNIT=TAPE,VOL=SER=T52183,LABEL=(18,SL),
// DISP=(OLD,KEEP)
//LKED.SYSLMOD DD DSN=REALES.CNEA(NUCLINT),SPACE=,DISP=OLD
/*
//PASO2 EXEC PROC=ASMFCLG
//ASM.SYSIN DD DSN=SIMBOLIC.CNEA(NUCLMATT),DISP=SHR
/*
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
//LKED.SYSIN DD *,DCB=BLKSIZE=80
        INCLUDE OTRALIB(NUCLINT)
        INCLUDE OTRALIB(MIXING)
```



```

INCLUDE OTRALIB(PARITY)
INCLUDE OTRALIB(WEEK)

```

```

/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
Aquí tarjetas perforadas de DATOS
/*
//

```

NOTA: En el programa principal ASSEMBLER se ha abierto un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

Modelo 229 . Ejecución de un programa en lenguaje ASSEMBLER residente en la biblioteca de simbólicos con el nombre FUERZAS, que invoca rutinas y funciones procedentes de FORTRAN o de ASSEMBLER, grabadas en la biblioteca de módulos de carga con los nombres COMPONEN, RESULTAN y PALANCA, a mas de otras rutinas y funciones en lenguaje absoluto de ASSEMBLER, residentes en cinta magnética. Estas últimas están grabadas, con el nombre EQUILIB, en el rótulo 31 de una cinta que lleva el número de carrete 21235 y tiene etiquetas modelo IBM. El problema se puede resolver en dos pasos: en el PASO1 el procedimiento LKED vincula las rutinas en absoluto de ASSEMBLER, cargando el módulo resultante en una biblioteca temporaria particionada que se crea en ese momento y reside en un disco auxiliar; en el PASO2 el procedimiento ASMFCLG ejecuta el programa FUERZAS vinculándolo con los módulos COMPONEN, RESULTAN, PALANCA y EQUILIB.

```

//..... JOB ....
//PASO1 EXEC PROC=LKED
//LKED.SYSMOD DD DSN=TEMPOR.ADR(EQUILIB),UNIT=WORKDISK,DISP=(,PASS),
//  DCB=BLKSIZE=80,SPACE=(CYL,(1,1,1))
//LKED.SYSIN DD DSN=EQUILIB,UNIT=TAPE,VOL=SER=T21235,LABEL=(31,SL),
//  DISP=(OLD,KEEP)
/*
//PASO2 EXEC PROC=ASMFCLG
//ASM.SYSIN DD DSN=SIMBOLIC.CNEA(FUERZAS),DISP=SHR
/*
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
//          DD DSN=TEMPOR.ADR,DISP=(OLD,DELETE)
//LKED.SYSIN DD *,DCB=BLKSIZE=80
        INCLUDE OTRALIB(COMPONEN)
        INCLUDE OTRALIB(RESULTAN)
        INCLUDE OTRALIB(PALANCA)
        INCLUDE OTRALIB(EQUILIB)
/*

```

```
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: Se menciona el DCB con el cual se grabará el disco auxiliar. En el programa principal ASSEMBLER se ha abierto un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

Modelo 230. Ejecución de un programa en lenguaje ASSEMBLER residente en la biblioteca de simbólicos con el nombre TOTCROSS, que invoca rutinas y funciones procedentes de FORTRAN o de ASSEMBLER, grabadas en la biblioteca de módulos de carga con los nombres CROSSECT, DEUTERON y PHOTON, a mas de otras rutinas y funciones en lenguaje simbólico FORTRAN residentes en cinta magnética. Estas últimas están grabadas, con el nombre SHADOW, en el rótulo 10 de una cinta que lleva el número de carrete 15532 y tiene etiquetas modelo IBM. El problema se puede resolver en dos pasos: en el PASO1 el procedimiento FTG1CL compila y vincula las rutinas y funciones FORTRAN cargando el módulo resultante en la biblioteca REALES. CNEA; en el PASO2 el procedimiento ASMFCLG ejecuta el programa TOTCROSS y lo vincula con los módulos CROSSECT, DEUTERON, PHOTON y SHADOW.

```
//..... JOB ....
//PASO1 EXEC PROC=FTG1CL
//FORT.SYSIN DD DSN=SHADOW,UNIT=TAPE,VOL=SER=T15532,LABEL=(10,SL),
// DISP=(OLD,KEEP)
//LKED.SYSLMOD DD DSN=REALES.CNEA(SHADOW),SPACE=,DISP=OLD
/*
//PASO2 EXEC PROC=ASMFCLG
//ASM.SYSIN DD DSN=SIMBOLIC.CNEA(TOTCROSS),DISP=SHR
/*
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
//LKED.SYSIN DD *,DCB=BLKSIZE=80
      INCLUDE OTRALIB(CROSSECT)
      INCLUDE OTRALIB(DEUTERON)
      INCLUDE OTRALIB(PHOTON)
      INCLUDE OTRALIB(SHADOW)
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: En el programa principal ASSEMBLER se ha abierto un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

Modelo 231. Ejecución de un programa en lenguaje ASSEMBLER residente en la biblioteca de simbólicos con el nombre SUPERT, que invoca rutinas y funciones procedentes de FORTRAN o de ASSEMBLER, grabadas en la biblioteca de módulos de carga con los nombres ASINTOTA y CALCULO8, a mas de otras rutinas y funciones en lenguaje absoluto de FORTRAN residentes en cinta magnética. Estas últimas están grabadas, con el nombre GENERAL, en el rótulo 17 de una cinta que lleva el número de carrete 16523 y tiene etiquetas modelo IBM. El problema se puede resolver en dos pasos: Previamente se define un procedimiento de nombre DRLKDFTR; en el PASO1 se invoca dicho procedimiento que vincula las rutinas y funciones en absoluto de FORTRAN, cargando el módulo resultante en una biblioteca temporaria particionada que se crea en ese momento y reside en un disco auxiliar; en el PASO2 el procedimiento ASMFCLG ejecuta el programa SUPERT vinculándolo con los módulos ASINTOTA, CALCULO8 y GENERAL.

```
//..... JOB ....
//DRLKDFTR PROC
//LKED EXEC PGM=IEWL,PARM=(XREF,LIST,LET)
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=&SYSUT1,SPACE=(3520,(80,80),,,ROUND),UNIT=SYSDA
//SYSLMOD DD DSN=&&LOD(X),UNIT=SYSDA,DISP=(,PASS),
// SPACE=(CYL,(1,1,1)),DCB=(BLKSIZE=13030,LRECL=1024,RECFM=U)
//SYSLIB DD DSN=SYS1.FORTLIB,DISP=SHR
//SYSLIN DD DDNAME=SYSIN,DCB=BLKSIZE=80
// PEND
//PASO1 EXEC PROC=DRLKDFTR
//LKED SYSLMOD DD DSN=TEMPOR.ADR(GENERAL),UNIT=WORKDISK,
// DISP=(,PASS),DCB=BLKSIZE=80,SPACE=(CYL,(1,1,1))
//LKED.SYSIN DD DSN=GENERAL,UNIT=TAPE,VOL=SER=T16523,LABEL=(17,SL),
// DISP=(OLD,KEEP)
/*
//PASO2 EXEC PROC=ASMFCLG
//ASM.SYSIN DD DSN=SIMBOLIC.CNEA(SUPERT),DISP=SHR
/*
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
// DD DSN=TEMPOR.ADR,DISP=(OLD,DELETE)
//LKED.SYSIN DD *,DCB=BLKSIZE=80
INCLUDE OTRALIB(ASINTOTA)
INCLUDE OTRALIB(CALCULO8)
INCLUDE OTRALIB(GENERAL)
/*
```

```
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: Se menciona el DCB con el cual se grabará el disco auxiliar. En el programa principal ASSEMBLER se ha abierto un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

PUNTO 4.

Ejecución de programas que son módulos de carga de una biblioteca de nombre REALES.CNEA e invocan rutinas y funciones en lenguaje simbólico, grabadas como miembros independientes en una biblioteca de nombre SIMBOLIC.CNEA.

Modelo 232 . Ejecución de un módulo de carga procedente de FORTRAN, grabado en la biblioteca con el nombre MEASUREM, que invoca rutinas y funciones en lenguaje FORTRAN, residentes en la biblioteca de simbólicos con los nombres MASSES, NUCLEI, SHELL y BETADEC. El problema se puede resolver en dos pasos: en el PASO1 el procedimiento FTG1CL compila y vincula las rutinas y funciones en FORTRAN, cargando el módulo resultante, con el nombre RUTIN45, en la biblioteca REALES.CNEA; en el PASO2 el procedimiento FTG1LG vincula los módulos MEASUREM y RUTIN45 y ejecuta.

```
//..... JOB ....
//PASO1 EXEC PROC=FTG1CL
//FORT.SYSIN DD DSN=SIMBOLIC.CNEA(MASSES),DISP=SHR
//          DD DSN=SIMBOLIC.CNEA(NUCLEI),DISP=SHR
//          DD DSN=SIMBOLIC.CNEA(SHELL),DISP=SHR
//          DD DSN=SIMBOLIC.CNEA(BETADEC),DISP=SHR
//LKED.SYSMOD DD DSN=REALES.CNEA(RUTIN45),SPACE=,DISP=OLD
/*
//PASO2 EXEC PROC=FTG1LG
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
//LKED.SYSIN DD *,DCB=BLKSIZE=80
              INCLUDE OTRALIB(MEASUREM)
              INCLUDE OTRALIB(RUTIN45)
/*
//GO.SYSIN DD *
Aquí tarjetas perforadas de DATOS
/*
//
```

Modelo 233 . Ejecución de un módulo de carga procedente de FORTRAN, grabado en la biblioteca con el nombre PRUEBTC, que invoca rutinas y funciones en lenguaje ASSEMBLER que residen, con el nombre VARIAS, en la biblioteca de simbólicos. El problema se puede resolver en dos pasos: en el PASO1 el procedimiento ASMFCL compila y vincula las rutinas y funciones en ASSEMBLER cargando el módulo resultante en la biblioteca REALES.CNEA; en el PASO2 el procedimiento FTG1LG vincula los mó-

dulos PRUEBTC y VARIAS y ejecuta.

```
//..... JOB ....
//PASO1 EXEC PROC=ASMFCL
//ASM.SYSIN DD DSN=SIMBOLIC.CNEA(VARIAS),DISP=SHR
//LKED.SYSLMOD DD DSN=REALES.CNEA(VARIAS),SPACE=,DISP=OLD
/*
//PASO2 EXEC PROC=FTG1LG
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
//LKED.SYSIN DD *,DCB=BLKSIZE=80
        INCLUDE OTRALIB(PRUEBTC)
        INCLUDE OTRALIB(VARIAS)
/*
//GO.SYSIN DD *
Aquí tarjetas perforadas de DATOS
/*
//
```

Modelo 234 . Ejecución de un módulo de carga procedente de ASSEMBLER, grabado en la biblioteca con el nombre IMPACTO, que invoca rutinas y funciones en lenguaje ASSEMBLER que residen en la biblioteca de simbólicos, con el nombre NEUTRAL. El problema se puede resolver en dos pasos: en el PASO1 el procedimiento ASMFCL compila y vincula las rutinas y funciones en ASSEMBLER, cargando el módulo resultante en la biblioteca REALES.CNEA; en el PASO2 el procedimiento LKEDG vincula el módulo IMPACTO con el módulo NEUTRAL y ejecuta.

```
//..... JOB ....
//PASO1 EXEC PROC=ASMFCL
//ASM.SYSIN DD DSN=SIMBOLIC.CNEA(NEUTRAL),DISP=SHR
//LKED.SYSLMOD DD DSN=REALES.CNEA(NEUTRAL),SPACE=,DISP=OLD
/*
//PASO2 EXEC PROC=LKEDG
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
//LKED.SYSIN DD *,DCB=BLKSIZE=80
        INCLUDE OTRALIB(IMPACTO)
        INCLUDE OTRALIB(NEUTRAL)
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: En el programa principal ASSEMBLER que dió origen al módulo IMPACTO, se abrió un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

Modelo 235 . Ejecución de un módulo de carga procedente de ASSEMBLER, grabado en la biblioteca con el nombre CHANNEL, que invoca rutinas y funciones en lenguaje FORTRAN, que residen en la biblioteca de simbólicos con los nombres TWOBODY, AMPLITUD, UNITARY y SCATTER. El problema se puede resolver en dos pasos: en el PASO1 el procedimiento FTG1CL compila y vincula las rutinas y funciones en FORTRAN cargando el módulo resultante, con el nombre FUMIGAR, en la biblioteca REALES.CNEA; en el PASO2 el procedimiento LKEDG vincula los módulos CHANNEL y FUMIGAR y ejecuta.

```
//..... JOB ....
//PASO1 EXEC PROC=FTG1CL
//FORT.SYSIN DD DSN=SIMBOLIC.CNEA(TWOBODY),DISP=SHR
//          DD DSN=SIMBOLIC.CNEA(AMPLITUD),DISP=SHR
//          DD DSN=SIMBOLIC.CNEA(UNITARY),DISP=SHR
//          DD DSN=SIMBOLIC.CNEA(SCATTER),DISP=SHR
//LKED.SYSLMOD DD DSN=REALES.CNEA(FUMIGAR),SPACE=,DISP=OLD
/*
//PASO2 EXEC PROC=LKEDG
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
//LKED.SYSIN DD *,DCB=BLKSIZE=80
           INCLUDE OTRALIB(CHANNEL)
           INCLUDE OTRALIB(FUMIGAR)
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: En el programa principal ASSEMBLER que dió origen al módulo CHANNEL, se abrió un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

PUNTO 5.

Ejecución de programas que son módulos de carga de una biblioteca de nombre REALES.CNEA e invocan rutinas y funciones en lenguaje simbólico, grabadas como miembros independientes en una biblioteca de nombre SIMBOLIC.CNEA, a mas de otras rutinas y funciones que se presentan en tarjetas perforadas.

Modelo 236 . Ejecución de un módulo de carga procedente de FORTRAN, que está grabado en la biblioteca con el nombre ALFIL e invoca rutinas y funciones en lenguaje FORTRAN, residentes en la biblioteca de simbólicos con los nombres PARTIM, CUSTOM y SIGNAL, a mas de otras rutinas y funciones que se presentan en tarjetas perforadas en lenguaje simbólico FORTRAN. El problema se puede resolver en dos pasos: en el PASO1 el procedimiento FTG1CL compila y vincula las rutinas y funciones en FORTRAN de la biblioteca SIMBOLIC.CNEA, mas las tarjetas perforadas en ese mismo lenguaje simbólico, cargando el módulo resultante, con el nombre ALFILRUT, en la biblioteca REALES.CNEA; en el PASO2 el procedimiento FTG1LG vincula los módulos ALFIL y ALFILRUT y ejecuta.

```
//..... JOB ....
//PASO1 EXEC PROC=FTG1CL
//FORT.SYSIN DD DSN=SIMBOLIC.CNEA(PARTIM),DISP=SHR
//          DD DSN=SIMBOLIC.CNEA(CUSTOM),DISP=SHR
//          DD DSN=SIMBOLIC.CNEA(SIGNAL),DISP=SHR
//          DD *
    Aquí tarjetas perforadas en FORTRAN
//LKED.SYSMOD DD DSN=REALES.CNEA(ALFILRUT),SPACE=,DISP=OLD
/*
//PASO2 EXEC PROC=FTG1LG
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
//LKED.SYSIN DD *,DCB=BLKSIZE=80
            INCLUDE OTRALIB(ALFIL)
            INCLUDE OTRALIB(ALFILRUT)
/*
//GO.SYSIN DD *
    Aquí tarjetas perforadas de DATOS
/*
//
```

Modelo 237 . Ejecución de un módulo de carga procedente de FORTRAN, que está grabado en la biblioteca con el nombre PROGRAM9 e invoca rutinas y funciones en lenguaje FORTRAN, residentes en la biblioteca de simbólicos con los nombres

RUTINA26, RUTINA27 y RUTINA28, a mas de otras rutinas y funciones que se presentan en tarjetas perforadas en lenguaje absoluto de FORTRAN o de ASSEMBLER. El problema se puede resolver en dos pasos: en el PASO1 el procedimiento FTG1CL compila y vincula las rutinas y funciones en FORTRAN de la biblioteca SIMBOLIC.CNEA, cargando el módulo resultante, con el nombre RUTINAS, en la biblioteca REALES.CNEA; en el PASO2 el procedimiento FTG1LG vincula los módulos PROGRAM9 y RUTINAS con las tarjetas perforadas en absoluto y ejecuta.

```
//..... JOB ....
//PASO1 EXEC PROC=FTG1CL
//FORK.SYSIN DD DSN=SIMBOLIC.CNEA(RUTINA26),DISP=SHR
//          DD DSN=SIMBOLIC.CNEA(RUTINA27),DISP=SHR
//          DD DSN=SIMBOLIC.CNEA(RUTINA28),DISP=SHR
//LKED.SYSLMOD DD DSN=REALES.CNEA(RUTINAS),SPACE=,DISP=OLD
/*
//PASO2 EXEC PROC=FTG1LG
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
//LKED.SYSIN DD *,DCB=BLKSIZE=80
//          INCLUDE OTRALIB(PROGRAM9)
//          INCLUDE OTRALIB(RUTINAS)
Aquí tarjetas perforadas en absoluto de FORTRAN o de ASSEMBLER
/*
//GO.SYSIN DD *
Aquí tarjetas perforadas de DATOS
/*
//
```

Modelo 238 . Ejecución de un módulo de carga procedente de FORTRAN, que está grabado en la biblioteca con el nombre RESONAN e invoca rutinas y funciones en lenguaje FORTRAN, residentes en la biblioteca de simbólicos con los nombres DIFFEREN, THREEDIM y MAGNFLD, a mas de otras rutinas y funciones que se presentan en tarjetas perforadas en lenguaje simbólico ASSEMBLER. El problema se puede resolver en tres pasos: en el PASO1 el procedimiento FTG1CL compila y vincula las rutinas y funciones en FORTRAN de la biblioteca SIMBOLIC.CNEA, cargando el módulo resultante, con el nombre SATRAP en la biblioteca REALES.CNEA; en el PASO2 el procedimiento ASMFCL compila y vincula las rutinas y funciones en ASSEMBLER en tarjetas perforadas, cargando el módulo resultante, con el nombre SOCOLIC, también en la biblioteca de módulos; en el PASO3 el procedimiento FTG1LG vincula los módulos RESONAN, SATRAP y SOCOLIC y ejecuta.

```
//..... JOB ....
//PASO1 EXEC PROC=FTG1CL
//FORT.SYSIN DD DSN=SIMBOLIC.CNEA(DIFFEREN),DISP=SHR
//          DD DSN=SIMBOLIC.CNEA(THREEDIM),DISP=SHR
//          DD DSN=SIMBOLIC.CNEA(MAGNFLD),DISP=SHR
//LKED.SYSLMOD DD DSN=REALES.CNEA(SATRAP),SPACE=,DISP=OLD
/*
//PASO2 EXEC PROC=ASMFCL
//ASM.SYSIN DD *
  Aquí tarjetas perforadas en ASSEMBLER
//LKED.SYSLMOD DD DSN=REALES.CNEA(SOCOLIC),SPACE=,DISP=OLD
/*
//PASO3 EXEC PROC=FTG1LG
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
//LKED.SYSIN DD *,DCB=BLKSIZE=80
      INCLUDE OTRALIB(RESONAN)
      INCLUDE OTRALIB(SATRAP)
      INCLUDE OTRALIB(SOCOLIC)
/*
//GO.SYSIN DD *
  Aquí tarjetas perforadas de DATOS
/*
//
```

Modelo 239 . Ejecución de un módulo de carga procedente de FORTRAN, que está grabado en la biblioteca con el nombre FOTOION e invoca rutinas y funciones en lenguaje ASSEMBLER, residentes en la biblioteca de simbólicos con el nombre HELIO, a mas de otras rutinas y funciones que se presentan en tarjetas perforadas en lenguaje simbólico FORTRAN. El problema se puede resolver en tres pasos: en el PASO1 el procedimiento ASMFCL compila y vincula las rutinas y funciones en ASSEMBLER de la biblioteca SIMBOLIC.CNEA, cargando el módulo resultante en la biblioteca REALES.CNEA; en el PASO2 el procedimiento FTG1CL compila y vincula las rutinas y funciones en FORTRAN en tarjetas perforadas, cargando el módulo resultante, con el nombre FASCINA, también en la biblioteca de módulos; en el PASO3 el procedimiento FTG1LG vincula los módulos FOTOION, HELIO y FASCINA y ejecuta.

```
//..... JOB ....
//PASO1 EXEC PROC=ASMFCL
//ASM.SYSIN DD DSN=SIMBOLIC.CNEA(HELIO),DISP=SHR
//LKED.SYSLMOD DD DSN=REALES.CNEA(HELIO),SPACE=,DISP=OLD
/*
//PASO2 EXEC PROC=FTG1CL
//FORT.SYSIN DD *
  Aquí tarjetas perforadas en FORTRAN
//LKED.SYSLMOD DD DSN=REALES.CNEA(FASCINA),SPACE=,DISP=OLD
```

```

/*
//PASO3 EXEC PROC=FTG1LG
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
//LKED.SYSIN DD *,DCB=BLKSIZE=80
        INCLUDE OTRALIB(FOTOION)
        INCLUDE OTRALIB(HELIO)
        INCLUDE OTRALIB(FASCINA)
/*
//GO.SYSIN DD *
Aquí tarjetas perforadas de DATOS
/*
//

```

Modelo 240 . Ejecución de un módulo de carga procedente de FORTRAN, que está grabado en la biblioteca con el nombre ACTINIDE e invoca rutinas y funciones en lenguaje ASSEMBLER, residentes en la biblioteca de simbólicos con el nombre FOTOFIS, a mas de otras rutinas y funciones que se presentan en tarjetas perforadas en lenguaje absoluto de FORTRAN o de ASSEMBLER. El problema se puede resolver en dos pasos: en el PASO1 el procedimiento ASMFCL compila y vincula las rutinas y funciones en ASSEMBLER de la biblioteca SIMBOLIC.CNEA, cargando el módulo resultante en la biblioteca REALES.CNEA; en el PASO2 el procedimiento FTG1LG vincula los módulos ACTINIDE y FOTOFIS con las tarjetas perforadas en absoluto y ejecuta.

```

//..... JOB ....
//PASO1 EXEC PROC=ASMFCL
//ASM.SYSIN DD DSN=SIMBOLIC.CNEA(FOTOFIS),DISP=SHR
//LKED.SYSLMOD DD DSN=REALES.CNEA(FOTOFIS),SPACE=,DISP=OLD
/*
//PASO2 EXEC PROC=FTG1LG
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
//LKED.SYSIN DD *,DCB=BLKSIZE=80
        INCLUDE OTRALIB(ACTINIDE)
        INCLUDE OTRALIB(FOTOFIS)
Aquí tarjetas perforadas en absoluto de FORTRAN o de ASSEMBLER
/*
//GO.SYSIN DD *
Aquí tarjetas perforadas de DATOS
/*
//

```

Modelo 241 . Ejecución de un módulo de carga procedente de FORTRAN, que está grabado en la biblioteca con el nombre COUPLING e invoca rutinas y funciones en lenguaje ASSEMBLER, residentes en la biblioteca de simbólicos con el nombre MODELAPP, a mas de otras rutinas y funciones que se presentan en tarjetas perfo-

radas en lenguaje simbólico ASSEMBLER. El problema se puede resolver en tres pasos: en el PASO1 el procedimiento ASMFCL compila y vincula las rutinas y funciones en ASSEMBLER de la biblioteca SIMBOLIC.CNEA, cargando el módulo resultante en la biblioteca REALES.CNEA; en el PASO2 el procedimiento ASMFCL compila y vincula las rutinas y funciones en ASSEMBLER en tarjetas perforadas, cargando el módulo resultante, con el nombre BROKEN, también en la biblioteca de módulos; en el PASO3 el procedimiento FTG1LG vincula los módulos COUPLING, MODELAPP y BROKEN y ejecuta.

```
//..... JOB ....
//PASO1 EXEC PROC=ASMFCL
//ASM.SYSIN DD DSN=SIMBOLIC.CNEA(MODELAPP),DISP=SHR
//LKED.SYSLMOD DD DSN=REALES.CNEA(MODELAPP),SPACE=,DISP=OLD
/*
//PASO2 EXEC PROC=ASMFCL
//ASM.SYSIN DD *
  Aquí tarjetas perforadas en ASSEMBLER
//LKED.SYSLMOD DD DSN=REALES.CNEA(BROKEN),SPACE=,DISP=OLD
/*
//PASO3 EXEC PROC=FTG1LG
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
//LKED.SYSIN DD *,DCB=BLKSIZE=80
      INCLUDE OTRALIB(COUPLING)
      INCLUDE OTRALIB(MODELAPP)
      INCLUDE OTRALIB(BROKEN)
/*
//GO.SYSIN DD *
  Aquí tarjetas perforadas de DATOS
/*
//
```

Modelo 242 . Ejecución de un módulo de carga procedente de ASSEMBLER, que está grabado en la biblioteca con el nombre GALAXIA e invoca rutinas y funciones en lenguaje ASSEMBLER, residentes en la biblioteca de simbólicos, con el nombre LUMINOS, a mas de otras rutinas y funciones que se presentan en tarjetas perforadas en lenguaje simbólico ASSEMBLER. El problema se puede resolver en tres pasos: en el PASO1 el procedimiento ASMFCL compila y vincula las rutinas y funciones en ASSEMBLER de la biblioteca SIMBOLIC.CNEA, cargando el módulo resultante en la biblioteca REALES.CNEA; en el PASO2 el mismo procedimiento compila y vincula las rutinas y funciones en tarjetas perforadas en ASSEMBLER, cargando el módulo resul-

tante, con el nombre SOLARUT, también en la biblioteca de módulos; en el PASO3 el procedimiento LKEDG vincula los módulos GALAXIA, LUMINOS y SOLARUT y ejecuta.

```
//..... JOB ....
//PASO1 EXEC PROC=ASMFCL
//ASM.SYSIN DD DSN=SIMBOLIC.CNEA(LUMINOS),DISP=SHR
//LKED.SYSLMOD DD DSN=REALES.CNEA(LUMINOS),SPACE=,DISP=OLD
/*
//PASO2 EXEC PROC=ASMFCL
//ASM.SYSIN DD *
  Aquí tarjetas perforadas en ASSEMBLER
//LKED.SYSLMOD DD DSN=REALES.CNEA(SOLARUT),SPACE=,DISP=OLD
/*
//PASO3 EXEC PROC=LKEDG
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
//LKED.SYSIN DD *,DCB=BLKSIZE=80
      INCLUDE OTRALIB(GALAXIA)
      INCLUDE OTRALIB(LUMINOS)
      INCLUDE OTRALIB(SOLARUT)
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas de DATOS.
/*
//
```

NOTA: En el programa principal ASSEMBLER que dió origen al módulo GALAXIA, se abrió un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

Modelo 243 . Ejecución de un módulo de carga procedente de ASSEMBLER, que está grabado en la biblioteca con el nombre COMPOSIT e invoca rutinas y funciones en lenguaje ASSEMBLER, residentes en la biblioteca de simbólicos con el nombre DYNAMICA, a mas de otras rutinas y funciones que se presentan en tarjetas perforadas en lenguaje simbólico FORTRAN. El problema se puede resolver en tres pasos: en el PASO1 el procedimiento ASMFCL compila y vincula las rutinas y funciones en ASSEMBLER de la biblioteca SIMBOLIC.CNEA, cargando el módulo resultante en la biblioteca REALES.CNEA; en el PASO2 el procedimiento FTG1CL compila y vincula las rutinas y funciones en tarjetas perforadas en FORTRAN, cargando el módulo resultante, con el nombre TURB, también en la biblioteca de módulos; en el PASO3 el procedimiento LKEDG vincula los módulos COMPOSIT, DYNAMICA y TURB y ejecuta.

```

//..... JOB ....
//PASO1 EXEC PROC=ASMFCL
//ASM.SYSIN DD DSN=SIMBOLIC.CNEA(DYNAMICA),DISP=SHR
//LKED.SYSLMOD DD DSN=REALES.CNEA(DYNAMICA),SPACE=,DISP=OLD
/*
//PASO2 EXEC PROC=FTG1CL
//FORT.SYSIN DD *
  Aquí tarjetas perforadas en FORTRAN
//LKED.SYSLMOD DD DSN=REALES.CNEA(TURB),SPACE=,DISP=OLD
/*
//PASO3 EXEC PROC=LKEDG
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
//LKED.SYSIN DD *,DCB=BLKSIZE=80
      INCLUDE OTRALIB(COMPOSIT)
      INCLUDE OTRALIB(DYNAMICA)
      INCLUDE OTRALIB(TURB)
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas de DATOS
/*
//

```

NOTA: En el programa principal ASSEMBLER que dió origen al módulo COMPOSIT, se abrió un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

Modelo 244 . Ejecución de un módulo de carga procedente de ASSEMBLER, que está grabado en la biblioteca con el nombre LUNAR e invoca rutinas y funciones en lenguaje ASSEMBLER, residentes en la biblioteca de simbólicos con el nombre ORBIT, a mas de otras rutinas y funciones que se presentan en tarjetas perforadas en lenguaje absoluto de FORTRAN o de ASSEMBLER. El problema se puede resolver en dos pasos: en el PASO1 el procedimiento ASMFCL compila y vincula las rutinas y funciones en ASSEMBLER de la biblioteca SIMBOLIC.CNEA, cargando el módulo resultante en la biblioteca REALES.CNEA; en el PASO2 el procedimiento FTG1LG vincula los módulos LUNAR y ORBIT con las tarjetas perforadas en absoluto de FORTRAN y ejecuta.

```

//..... JOB ....
//PASO1 EXEC PROC=ASMFCL
//ASM.SYSIN DD DSN=SIMBOLIC.CNEA(ORBIT),DISP=SHR
//LKED.SYSLMOD DD DSN=REALES.CNEA(ORBIT),SPACE=,DISP=OLD
/*
//PASO2 EXEC PROC=FTG1LG

```

```
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
//LKED.SYSIN DD *,DCB=BLKSIZE=80
        INCLUDE OTRALIB(LUNAR)
        INCLUDE OTRALIB(ORBIT)
Aquí tarjetas perforadas en absoluto de FORTRAN o de ASSEMBLER
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: En el programa principal ASSEMBLER que dió origen al módulo LUNAR, se abrió un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

Modelo 245. Ejecución de un módulo de carga procedente de ASSEMBLER, que está grabado en la biblioteca con el nombre KEPLER e invoca rutinas y funciones en lenguaje FORTRAN, residentes en la biblioteca de simbólicos con los nombres SYSTEM, SPINORB y MODEL, a mas de otras rutinas y funciones que se presentan en tarjetas perforadas en lenguaje simbólico ASSEMBLER. El problema se puede resolver en tres pasos: en el PASO1 el procedimiento FTG1CL compila y vincula las rutinas y funciones en FORTRAN de la biblioteca SIMBOLIC.CNEA, cargando el módulo resultante, con el nombre KARPER, en la biblioteca REALES.CNEA; en el PASO2 el procedimiento ASMFCL compila y vincula las rutinas y funciones en ASSEMBLER en tarjetas perforadas, cargando el módulo resultante, con el nombre FIELD, también en la biblioteca de módulos; en el PASO3 el procedimiento LKEDG vincula los módulos KEPLER, KARPER y FIELD y ejecuta.

```
//..... JOB ....
//PASO1 EXEC PROC=FTG1CL
//FORT.SYSIN DD DSN=SIMBOLIC.CNEA(SYSTEM),DISP=SHR
//          DD DSN=SIMBOLIC.CNEA(SPINORB),DISP=SHR
//          DD DSN=SIMBOLIC.CNEA(MODEL),DISP=SHR
//LKED.SYSLMOD DD DSN=REALES.CNEA(KARPER),SPACE=,DISP=OLD
/*
//PASO2 EXEC PROC=ASMFCL
//ASM.SYSIN DD *
Aquí tarjetas perforadas en ASSEMBLER
//LKED.SYSLMOD DD DSN=REALES.CNEA(FIELD),SPACE=,DISP=OLD
/*
//PASO3 EXEC PROC=LKEDG
```

```
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
//LKED.SYSIN DD *,DCB=BLKSIZE=80
        INCLUDE OTRALIB(KEPLER)
        INCLUDE OTRALIB(KARPER)
        INCLUDE OTRALIB(FIELD)
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
    Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: En el programa principal ASSEMBLER que dió origen al módulo KEPLER, se abrió un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

Modelo 246 . Ejecución de un módulo de carga procedente de ASSEMBLER, que está grabado en la biblioteca con el nombre LOCAL e invoca rutinas y funciones en lenguaje FORTRAN, residentes en la biblioteca de simbólicos con los nombres GRUPO, ACCION y ALGEBRA, a mas de otras rutinas y funciones que se presentan en tarjetas perforadas en lenguaje simbólico FORTRAN. El problema se puede resolver en dos pasos: en el PASO1 el procedimiento FTG1CL compila y vincula las rutinas y funciones en FORTRAN de la biblioteca SIMBOLIC.CNEA, mas las tarjetas perforadas en ese mismo lenguaje simbólico, cargando el módulo resultante, con el nombre FORM12, en la biblioteca REALES.CNEA; en el PASO2 el procedimiento LKEDG vincula los módulos LOCAL y FORM12 y ejecuta.

```
//..... JOB ....
//PASO1 EXEC PROC=FTG1CL
//FORT.SYSIN DD DSN=SIMBOLIC.CNEA(GRUPO),DISP=SHR
//          DD DSN=SIMBOLIC.CNEA(ACCION),DISP=SHR
//          DD DSN=SIMBOLIC.CNEA(ALGEBRA),DISP=SHR
//          DD *
    Aquí tarjetas perforadas en FORTRAN
//LKED.SYSLMOD DD DSN=REALES.CNEA(FORM12),SPACE=,DISP=OLD
/*
//PASO2 EXEC PROC=LKEDG
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
//LKED.SYSIN DD *,DCB=BLKSIZE=80
        INCLUDE OTRALIB(LOCAL)
        INCLUDE OTRALIB(FORM12)
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
```



```

Aquí tarjetas perforadas de DATOS
/*
//

```

NOTA: En el programa principal ASSEMBLER que dió origen al módulo LOCAL se abrió un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

Modelo 247 . Ejecución de un módulo de carga procedente de ASSEMBLER, que está grabado en la biblioteca con el nombre APPROXIM e invoca rutinas y funciones en lenguaje FORTRAN, residentes en la biblioteca de simbólicos con los nombres LATE, SLAGE, ARRAY y RODS, a mas de otras rutinas y funciones que se presentan en tarjetas perforadas en lenguaje absoluto de FORTRAN o de ASSEMBLER. El problema se puede resolver en dos pasos: en el PASO1 el procedimiento FTG1CL compila y vincula las rutinas y funciones en FORTRAN de la biblioteca SIMBOLIC.CNEA, cargando el módulo resultante, con el nombre VISCOUS, en la biblioteca REALES.CNEA; en el PASO2 el procedimiento FTG1LG vincula los módulos APPROXIM y VISCOUS con las tarjetas perforadas en absoluto y ejecuta.

```

//..... JOB ....
//PASO1 EXEC PROC=FTG1CL
//FORT.SYSIN DD DSN=SIMBOLIC.CNEA(LATE),DISP=SHR
//          DD DSN=SIMBOLIC.CNEA(SLAGE),DISP=SHR
//          DD DSN=SIMBOLIC.CNEA(ARRAY),DISP=SHR
//          DD DSN=SIMBOLIC.CNEA(RODS),DISP=SHR
//LKED.SYSMOD DD DSN=REALES.CNEA(VISCOUS),SPACE=,DISP=OLD
/*
//PASO2 EXEC PROC=FTG1LG
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
//LKED.SYSIN DD *,DCB=BLKSIZE=80
//          INCLUDE OTRALIB(APPROXIM)
//          INCLUDE OTRALIB(VISCOUS)
Aquí tarjetas perforadas en absoluto de FORTRAN o de ASSEMBLER
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
Aquí tarjetas perforadas de DATOS
/*
//

```

NOTA: En el programa principal ASSEMBLER que dió origen al módulo APPROXIM se abrió un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

PUNTO 6.

Ejecución de programas que son módulos de carga de una biblioteca de nombre REALES.CNEA e invocan rutinas y funciones en lenguaje simbólico que residen, como miembros independientes, en una biblioteca de nombre SIMBOLIC.CNEA, a mas de otras rutinas y funciones grabadas en cintas magnéticas.

Modelo 248 . Ejecución de un módulo de carga procedente de FORTRAN, que está grabado en la biblioteca con el nombre MOMENTO e invoca rutinas y funciones en lenguaje FORTRAN, residentes en la biblioteca de simbólicos con los nombres INERCIA, PARIDAD y VIBRAC; a mas de otras rutinas y funciones grabadas en lenguaje simbólico FORTRAN, con el nombre ESTADOS, en el rótulo 16 de una cinta magnética. La cinta lleva el número de carrete 68201 y tiene etiquetas modelo IBM. El problema se puede resolver en dos pasos: en el PASO1 el procedimiento FTG1CL compila y vincula las rutinas y funciones en FORTRAN de la biblioteca SIMBOLIC.CNEA, mas las rutinas y funciones grabadas en la cinta, cargando el módulo resultante, con el nombre ANOMALIA, en la biblioteca REALES.CNEA; en el PASO2 el procedimiento FTG1LG vincula los módulos MOMENTO y ANOMALIA y ejecuta.

```
//..... JOB ....
//PASO1 EXEC PROC=FTG1CL
//FORT.SYSIN DD DSN=SIMBOLIC.CNEA(INERCIA),DISP=SHR
//          DD DSN=SIMBOLIC.CNEA(PARIDAD),DISP=SHR
//          DD DSN=SIMBOLIC.CNEA(VIBRAC),DISP=SHR
//          DD DSN=ESTADOS,UNIT=TAPE,VOL=SER=T68021,LABEL=(16,SL),
// DISP=(OLD,KEEP)
//LKED.SYSLMOD DD DSN=REALES.CNEA(ANOMALIA),SPACE=,DISP=OLD
/*
//PASO2 EXEC PROC=FTG1LG
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
//LKED.SYSIN DD *,DCB=BLKSIZE=80
//          INCLUDE OTRALIB(MOMENTO)
//          INCLUDE OTRALIB(ANOMALIA)
/*
//GO.SYSIN DD *
//Aquí tarjetas perforadas de DATOS
/*
//
```

Modelo 249 . Ejecución de un módulo de carga procedente de FORTRAN, que está grabado en la biblioteca con el nombre HEAVY e invoca rutinas y funciones en len-

guaje FORTRAN, residentes en la biblioteca de simbólicos con los nombres RECOIL y SECOND; a mas de otras rutinas y funciones grabadas en lenguaje absoluto de FORTRAN o de ASSEMBLER, con el nombre ORDER, en el rótulo 31 de una cinta magnética. La cinta lleva el número de carrete 00901 y tiene etiquetas modelo IBM. El problema se puede resolver en tres pasos: en el PASO1 el procedimiento FTG1CL compila y vincula las rutinas y funciones en FORTRAN de la biblioteca SIMBOLIC.CNEA, cargando el módulo resultante, con el nombre SERIES, en la biblioteca REALES.CNEA; en el PASO2 el procedimiento LKED vincula las rutinas y funciones en lenguaje absoluto, cargando el módulo resultante en una biblioteca temporaria particionada que se crea en ese momento y reside en un disco auxiliar; en el PASO3 el procedimiento FTG1LG vincula los módulos HEAVY, SERIES y ORDER y ejecuta.

```
//..... JOB ....
//PASO1 EXEC PROC=FTG1CL
//FORT.SYSIN DD DSN=SIMBOLIC.CNEA(RECOIL),DISP=SHR
//          DD DSN=SIMBOLIC.CNEA(SECOND),DISP=SHR
//LKED.SYSLMOD DD DSN=REALES.CNEA(SERIES),SPACE=,DISP=OLD
/*
//PASO2 EXEC PROC=LKED
//LKED.SYSLMOD DD DSN=TEMPOR.ADR(ORDER),UNIT=WORKDISK,
//  DISP=(NEW,PASS),DCB=BLKSIZE=80,SPACE=(CYL,(1,1,1))
//LKED.SYSIN DD DSN=ORDER,UNIT=TAPE,VOL=SER=T00901,LABEL=(31,SL),
//  DISP=(OLD,KEEP)
/*
//PASO3 EXEC PROC=FTG1LG
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
//          DD DSN=TEMPOR.ADR,DISP=(OLD,DELETE)
//LKED.SYSIN DD *,DCB=BLKSIZE=80
//          INCLUDE OTRALIB(HEAVY)
//          INCLUDE OTRALIB(SERIES)
//          INCLUDE OTRALIB(ORDER)
/*
//GO.SYSIN DD *
Aquí tarjetas perforadas de DATOS
/*
//
```

Modelo 250. Ejecución de un módulo de carga procedente de FORTRAN, que está grabado en la biblioteca con el nombre MATTER e invoca rutinas y funciones en lenguaje FORTRAN, residentes en la biblioteca de simbólicos con los nombres MASSACH1, MASSACH2 y MASSACH3, a mas de otras rutinas y funciones grabadas en

lenguaje simbólico ASSEMBLER, con el nombre DISTRIB, en el rótulo 7 de una cinta magnética. La cinta lleva el número de carrete 63321 y tiene etiquetas modelo IBM. El problema se puede resolver en tres pasos: en el PASO1 el procedimiento FTG1CL compila y vincula las rutinas y funciones en FORTRAN de la biblioteca SIMBOLIC.CNEA, cargando el módulo resultante, con el nombre NUCLTHEO, en la biblioteca REALES.CNEA; en el PASO2 el procedimiento ASMFCL compila y vincula las rutinas y funciones en ASSEMBLER grabadas en la cinta, cargando el módulo resultante también en la biblioteca de módulos; en el PASO3 el procedimiento FTG1LG vincula los módulos MATTER, NUCLTHEO y DISTRIB y ejecuta.

```
//..... JOB ....
//PASO1 EXEC PROC=FTG1CL
//FORT.SYSIN DD DSN=SIMBOLIC.CNEA(MASSACH1),DISP=SHR
//          DD DSN=SIMBOLIC.CNEA(MASSACH2),DISP=SHR
//          DD DSN=SIMBOLIC.CNEA(MASSACH3),DISP=SHR
//LKED.SYSMOD DD DSN=REALES.CNEA(NUCLTHEO),SPACE=,DISP=OLD
/*
//PASO2 EXEC PROC=ASMFCL
//ASM.SYSIN DD DSN=DISTRIB,UNIT=TAPE,VOL=SER=T63321,LABEL=(7,SL),
//  DISP=(OLD,KEEP)
//LKED.SYSMOD DD DSN=REALES.CNEA(DISTRIB),SPACE=,DISP=OLD
/*
//PASO3 EXEC PROC=FTG1LG
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
//LKED.SYSIN DD *,DCB=BLKSIZE=80
//          INCLUDE OTRALIB(MATTER)
//          INCLUDE OTRALIB(NUCLTHEO)
//          INCLUDE OTRALIB(DISTRIB)
/*
//GO.SYSIN DD *
//  Aquí tarjetas perforadas de DATOS
/*
//
```

Modelo 251. Ejecución de un módulo de carga procedente de FORTRAN, que está grabado en la biblioteca con el nombre SELECCI e invoca rutinas y funciones en lenguaje ASSEMBLER, residentes en la biblioteca de simbólicos con el nombre CRITERIO, a mas de otras rutinas y funciones grabadas en lenguaje simbólico FORTRAN, con el nombre SIMILAR, en el rótulo 27 de una cinta magnética. La cinta lleva el número de carrete 68097 y tiene etiquetas modelo IBM. El problema se puede resolver en tres pasos: en el PASO1 el procedimiento ASMFCL compila y vin-

cula las rutinas y funciones en ASSEMBLER de la biblioteca SIMBOLIC.CNEA, cargando el módulo resultante en la biblioteca REALES.CNEA, en el PASO2 el procedimiento FTG1CL compila y vincula las rutinas y funciones en FORTRAN grabadas en la cinta, cargando el módulo resultante también en la biblioteca de módulos; en el PASO3 el procedimiento FTG1LG vincula los módulos SELECCI, CRITERIO y SIMILAR y ejecuta.

```
//..... JOB ....
//PASO1 EXEC PROC=ASMFCL
//ASM.SYSIN DD DSN=SIMBOLIC.CNEA(CRITERIO),DISP=SHR
//LKED.SYSLMOD DD DSN=REALES.CNEA(CRITERIO),SPACE=,DISP=OLD
/*
//PASO2 EXEC PROC=FTG1CL
//FORT.SYSIN DD DSN=SIMILAR,UNIT=TAPE,VOL=SER=T68097,LABEL=(27,SL),
// DISP=(OLD,KEEP)
//LKED.SYSLMOD DD DSN=REALES.CNEA(SIMILAR),SPACE=,DISP=OLD
/*
//PASO3 EXEC PROC=FTG1LG
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
//LKED.SYSIN DD *,DCB=BLKSIZE=80
        INCLUDE OTRALIB(SELECCI)
        INCLUDE OTRALIB(CRITERIO)
        INCLUDE OTRALIB(SIMILAR)
/*
//GO.SYSIN DD *
Aquí tarjetas perforadas de DATOS
/*
//
```

Modelo 252 . Ejecución de un módulo de carga procedente de FORTRAN, que está grabado en la biblioteca con el nombre CURVA e invoca rutinas y funciones en lenguaje ASSEMBLER, residentes en la biblioteca de simbólicos con el nombre ESPECTRO, a mas de otras rutinas y funciones grabadas en lenguaje absoluto de FORTRAN o de ASSEMBLER, con el nombre DOMINANT, en el rótulo 35 de una cinta magnética. La cinta lleva el número de carrete 22107 y tiene etiquetas modelo IBM. El problema se puede resolver en tres pasos: en el PASO1 el procedimiento ASMFCL compila y vincula las rutinas y funciones en ASSEMBLER de la biblioteca SIMBOLIC.CNEA, cargando el módulo resultante en la biblioteca REALES.CNEA; en el PASO2 el procedimiento LKED vincula las rutinas y funciones en lenguaje absoluto, cargando el módulo resultante en una biblioteca temporaria particionada que se crea en ese

momento y reside en un disco auxiliar, en el PASO3 el procedimiento FTG1LG vincula los módulos CURVA, ESPECTRO y DOMINANT y ejecuta.

```
//..... JOB ....
//PAS01 EXEC PROC=ASMFCL
//ASM.SYSIN DD DSN=SIMBOLIC.CNEA(ESPECTRO),DISP=SHR
//LKED.SYSLMOD DD DSN=REALES.CNEA(ESPECTRO),SPACE=,DISP=OLD
/*
//PAS02 EXEC PROC=LKED
//LKED.SYSLMOD DD DSN=TEMPOR.ADR(DOMINANT),UNIT=WCRDISK,DISP=(,PASS),
//  DCB=BLKSIZE=80,SPACE=(CYL,(1,1,1))
//LKED.SYSIN DD DSN=DOMINANT,UNIT=TAPE,VOL=SER=T22107,LABEL=(35,SL),
//  DISP=(OLD,KEEP)
/*
//PAS03 EXEC PROC=FTG1LG
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
//          DD DSN=TEMPOR.ADR,DISP=(OLD,DELETE)
//LKED.SYSIN DD *,DCB=BLKSIZE=80
//          INCLUDE OTRALIB(CURVA)
//          INCLUDE OTRALIB(ESPECTRO)
//          INCLUDE OTRALIB(DOMINANT)
/*
//GO.SYSIN DE *
//  Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: Se menciona el DCB con el cual se grabará el disco auxiliar.

Modelo 253 . Ejecución de un módulo de carga procedente de FORTRAN, que está grabado en la biblioteca con el nombre PERIFERI e invoca rutinas y funciones en lenguaje ASSEMBLER, residentes en la biblioteca de simbólicos con el nombre COEFIC, a mas de otras rutinas y funciones grabadas en lenguaje simbólico ASSEMBLER, con el nombre DIAGRAMA, en el rótulo 6 de una cinta magnética. La cinta lleva el número de carrete 53555 y tiene etiquetas modelo IBM. El problema se puede resolver en tres pasos: en el PASO1 el procedimiento ASMFCL compila y vincula las rutinas y funciones en ASSEMBLER de la biblioteca SIMBOLIC.CNEA, cargando el módulo resultante en la biblioteca REALES.CNEA; en el PASO2 el mismo procedimiento compila y vincula las rutinas y funciones grabadas en la cinta, cargando el módulo resultante también en la biblioteca de módulos; en el PASO3 el procedimiento FTG1LG vincula los módulos PERIFERI, COEFIC y DIAGRAMA y ejecuta.

```
//..... JOB ....
```

```

//PAS01 EXEC PROC=ASMFCL
//ASM.SYSIN DD DSN=SIMBOLIC.CNEA(COEFIC),DISP=SHR
//LKED.SYSLMOD DD DSN=REALES.CNEA(COEFIC),SPACE=,DISP=OLD
/*
//PAS02 EXEC PROC=ASMFCL
//ASM.SYSIN DD DSN=DIAGRAMA,UNIT=TAPE,VOL=SER=T53555,LABEL=(6,SL),
//  DISP=(OLD,KEEP)
//LKED.SYSLMOD DD DSN=REALES.CNEA(DIAGRAMA),SPACE=,DISP=OLD
/*
//PAS03 EXEC PROC=FTGILG
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
//LKED.SYSIN DD *,DCB=BLKSIZE=80
        INCLUDE OTRALIB(PERIFERI)
        INCLUDE OTRALIB(COEFIC)
        INCLUDE OTRALIB(DIAGRAMA)
/*
//GO.SYSIN DD *
Aquí tarjetas perforadas de DATOS
/*
//

```

Modelo 254 . Ejecución de un módulo de carga procedente de ASSEMBLER, que está grabado en la biblioteca con el nombre ESTFINAL e invoca rutinas y funciones en lenguaje ASSEMBLER, residentes en la biblioteca de simbólicos con el nombre COLISION, a mas de otras rutinas y funciones grabadas en lenguaje simbólico ASSEMBLER, con el nombre NUCLEON, en el rótulo 2 de una cinta magnética. La cinta lleva el número de carrete 00193 y tiene etiquetas modelo IBM. El problema se puede resolver en tres pasos: en el PAS01 el procedimiento ASMFCL compila y vincula las rutinas y funciones en ASSEMBLER de la biblioteca SIMBOLIC.CNEA, cargando el módulo resultante en la biblioteca REALES.CNEA; en el PAS02 el mismo procedimiento compila y vincula las rutinas y funciones en ASSEMBLER grabadas en la cinta, cargando el módulo resultante también en la biblioteca de módulos. En el PAS03 el procedimiento LKEDG vincula los módulos ESTFINAL, COLISION y NUCLEON y ejecuta.

```

//..... JOB ....
//PAS01 EXEC PROC=ASMFCL
//ASM.SYSIN DD DSN=SIMBOLIC.CNEA(COLISION),DISP=SHR
//LKED.SYSLMOD DD DSN=REALES.CNEA(COLISION),SPACE=,DISP=OLD
/*
//PAS02 EXEC PROC=ASMFCL
//ASM.SYSIN DD DSN=NUCLEON,UNIT=TAPE,VOL=SER=T00193,LABEL=(2,SL),
//  DISP=(OLD,KEEP)
//LKED.SYSLMOD DD DSN=REALES.CNEA(NUCLEON),SPACE=,DISP=OLD

```

```

/*
//PASO3 EXEC PROC=LKEDG
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
//LKED.SYSIN DD *,DCB=BLKSIZE=80
        INCLUDE OTRALIB(ESTFINAL)
        INCLUDE OTRAJIB(COLISION)
        INCLUDE OTRALIB(NUCLEON)
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
    Aquí tarjetas perforadas de DATOS
/*
//

```

NOTA: En el programa principal ASSEMBLER que dió origen al módulo ESTFINAL, se abrió un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

Modelo 255. Ejecución de un módulo de carga procedente de ASSEMBLER, que está grabado en la biblioteca con el nombre VELOCITY e invoca rutinas y funciones en lenguaje ASSEMBLER, residentes en la biblioteca de simbólicos con el nombre POPULAT, a mas de otras rutinas y funciones grabadas en lenguaje simbólico FORTRAN, con el nombre ATOMSTAT, en el rótulo 13 de una cinta magnética. La cinta lleva el número de carrete 21371 y tiene etiquetas modelo IBM. El problema se puede resolver en tres pasos: en el PASO1 el procedimiento ASMFCL compila y vincula las rutinas y funciones en ASSEMBLER de la biblioteca SIMBOLIC.CNEA, cargando el módulo resultante en la biblioteca REALES.CNEA; en el PASO2 el procedimiento FTG1CL compila y vincula las rutinas y funciones en FORTRAN grabadas en la cinta, cargando el módulo resultante también en la biblioteca de módulos; en el PASO3 el procedimiento LKEDG vincula los módulos VELOCITY, POPULAT y ATOMSTAT y ejecuta.

```

//..... JOB ....
//PASO1 EXEC PROC=ASMFCL
//ASM.SYSIN DD DSN=SIMBOLIC.CNEA(POPULAT),DISP=SHR
//LKED.SYSLMOD DD DSN=REALES.CNEA(POPULAT),SPACE=,DISP=OLD
/*
//PASO2 EXEC PROC=FTG1CL
//FORT.SYSIN DD DSN=ATOMSTAT,UNIT=TAPE,VOL=SER=T21371,LABEL=(13,SL),
// DISP=(OLD,KEEP)
//LKED.SYSLMOD DD DSN=REALES.CNEA(ATOMSTAT),SPACE=,DISP=OLD
/*

```



```
//PAS03 EXEC PROC=LKEDG
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
//LKED.SYSIN DD *,DCB=BLKSIZE=80
        INCLUDE OTRALIB(VELOCITY)
        INCLUDE OTRALIB(POPULAT)
        INCLUDE OTRALIB(ATOMSTAT)
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: En el programa principal ASSEMBLER que dió origen al módulo VELOCITY, se abrió un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

Modelo 256. Ejecución de un módulo de carga procedente de ASSEMBLER, que está grabado en la biblioteca con el nombre BETARAY e invoca rutinas y funciones en lenguaje ASSEMBLER, residentes en la biblioteca de simbólicos con el nombre BRANCH, a mas de otras rutinas y funciones grabadas en lenguaje absoluto de FORTRAN o de ASSEMBLER, con el nombre HALFLIV, en el rótulo 37 de una cinta magnética. La cinta lleva el número de carrete 60459 y tiene etiquetas modelo IBM. El problema se puede resolver en tres pasos: en el PASO1 el procedimiento ASMFCL compila y vincula las rutinas y funciones en ASSEMBLER de la biblioteca SIMBOLIC.CNEA, cargando el módulo resultante en la biblioteca REALES.CNEA; en el PASO2 el procedimiento LKED vincula las rutinas y funciones en lenguaje absoluto, cargando el módulo resultante en una biblioteca temporaria particionada que se crea en ese momento y reside en un disco auxiliar; en el PASO3 el procedimiento FTGLG vincula los módulos BETARAY, BRANCH y HALFLIV y ejecuta.

```
//..... JOB ....
//PAS01 EXEC PROC=ASMFCL
//ASM.SYSIN DD DSN=SIMBOLIC.CNEA(BRANCH),DISP=SHR
//LKED.SYSLMOD DD DSN=REALES.CNEA(BRANCH),SPACE=,DISP=OLD
/*
//PASO2 EXEC PROC=LKED
//LKED.SYSLMOD DD DSN=TEMPOR.ADR(HALFLIV),UNIT=WORKDISK,DISP=(,PASS),
// DCB=BLKSIZE=80,SPACE=(CYL,(1,1,1))
//LKED.SYSIN DD DSN=HALFLIV,UNIT=TAPE,VOL=SER=T60459,LABEL=(37,SL),
// DISP=(OLD,KEEP)
```

```

/*
//PAS03 EXEC PROC=FTG1LG
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
//          DD DSN=TEMPOR.ADR,DISP=(OLD,DELETE)
//LKED.SYSIN DD *,DCB=BLKSIZE=80
//          INCLUDE OTRALIB(BETARAY)
//          INCLUDE OTRALIB(BRANCH)
//          INCLUDE OTRALIB(HALFLIV)
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(ELKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
//Aquí tarjetas perforadas de DATOS
/*
//

```

NOTA: Se menciona el DCB con el cual se grabará el disco auxiliar. En el programa principal ASSEMBLER que dió origen al módulo BETARAY, se abrió un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

Modelo 257 . Ejecución de un módulo de carga procedente de ASSEMBLER, que está grabado en la biblioteca con el nombre ENDEPEND e invoca rutinas y funciones en lenguaje FORTRAN, residentes en la biblioteca de simbólicos con los nombres HEAVYION, OPTICAL y POTNTIAL, a mas de otras rutinas y funciones grabadas en lenguaje simbólico ASSEMBLER, con el nombre ELASTIC, en el rótulo 9 de una cinta magnética. La cinta lleva el número de carrete 77000 y tiene etiquetas modelo IBM. El problema se puede resolver en tres pasos: en el PAS01 el procedimiento FTG1CL compila y vincula las rutinas y funciones en FORTRAN de la biblioteca SIMBOLIC.CNEA, cargando el módulo resultante, con el nombre SCATT, en la biblioteca REALES.CNEA; en el PAS02 el procedimiento ASMFCL compila y vincula las rutinas y funciones en ASSEMBLER grabadas en la cinta, cargando el módulo resultante también en la biblioteca de módulos; en el PAS03 el procedimiento LKEDG vincula los módulos ENDEPEND, SCATT y ELASTIC y ejecuta.

```

//..... JOB ....
//PAS01 EXEC PROC=FTG1CL
//FORT.SYSIN DD DSN=SIMBOLIC.CNEA(HEAVYION),DISP=SHR
//          DD DSN=SIMBOLIC.CNEA(OPTICAL),DISP=SHR
//          DD DSN=SIMBOLIC.CNEA(POTNTIAL),DISP=SHR
//LKED.SYSMOD DD DSN=REALES.CNEA(SCATT),SPACE=,DISP=OLD

```

```

/*
//PAS02 EXEC PROC=ASMFCL
//ASM.SYSIN DD DSN=ELASTIC,UNIT=TAPE,VOL=SER=T77000,LABEL=(9,SL),
//  DISP=(OLD,KEEP)
//LKED.SYSMOD DD DSN=REALES.CNEA(ELASTIC),SPACE=,DISP=OLD
/*
//PAS03 EXEC PROC=LKEDG
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
//LKED.SYSIN DD *,DCB=BLKSIZE=80
        INCLUDE OTRALIB(ENDEPEND)
        INCLUDE OTRALIB(SCATT)
        INCLUDE OTRALIB(ELASTIC)
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
    Aquí tarjetas perforadas de DATOS
/*
//

```

NOTA: En el programa principal ASSEMBLER que dió origen al módulo ENDEPEND se abrió un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

Modelo 258 . Ejecución de un módulo de carga procedente de ASSEMBLER, que está grabado en la biblioteca con el nombre COLDEPI e invoca rutinas y funciones en lenguaje FORTRAN, residentes en la biblioteca de simbólicos con los nombres MASA, ISOBARO, TRANSFOR y FACTOR, a mas de otras rutinas y funciones grabadas en lenguaje simbólico FORTRAN, con el nombre RESULEXP, en el rótulo 43 de una cinta magnética. La cinta lleva el número de carrete 88814 y tiene etiquetas modelo IBM. El problema se puede resolver en dos pasos: en el PAS01 el procedimiento FTG1CL compila y vincula las rutinas y funciones en lenguaje simbólico FORTRAN de la biblioteca SIMBOLIC.CNEA, mas las grabadas en ese mismo lenguaje en la cinta, cargando el módulo resultante, con el nombre PARTEFOR, en la biblioteca REALES.CNEA; en el PAS02 el procedimiento LKEDG vincula los módulos COLDEPI y PARTEFOR y ejecuta.

```

//..... JOB ....
//PAS01 EXEC PROC=FTG1CL
//FORT.SYSIN DD DSN=SIMBOLIC.CNEA(MASA),DISP=SHR
//          DD DSN=SIMBOLIC.CNEA(ISOBARO),DISP=SHR
//          DD DSN=SIMBOLIC.CNEA(TRANSFOR),DISP=SHR
//          DD DSN=SIMBOLIC.CNEA(FACTOR),DISP=SHR

```

```
//          DD DSN=RESULEXP,UNIT=TAPE,VOL=SER=T88814,LABEL=(43,SL),
//  DISP=(OLD,KEEP)
//LKED.SYSLMOD DD DSN=REALES.CNEA(PARTEFOR),SPACE=,DISP=CLD
/*
//PASO2 EXEC PROC=LKEDG
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
//LKED.SYSIN DD *,DCB=BLKSIZE=80
//          INCLUDE OTRALIB(COLDEPI)
//          INCLUDE OTRALIB(PARTEFOR)
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
//          Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: En el programa principal ASSEMBLER que dió origen al módulo COLDEPI, se abrió un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

Modelo 259. Ejecución de un módulo de carga procedente de ASSEMBLER, que está grabado en la biblioteca con el nombre HAZ e invoca rutinas y funciones en lenguaje FORTRAN, residentes en la biblioteca de simbólicos con los nombres PORCENT, ANGESPAC y MOMENTO, a mas de otras rutinas y funciones grabadas en lenguaje absoluto de FORTRAN o de ASSEMBLER, con el nombre LONGTRAZ, en el rótulo 16 de una cinta magnética. La cinta lleva el número de carrete 37195 y tiene etiquetas modelo IBM. El problema se puede resolver en tres pasos: en el PASO1 el procedimiento FTG1CL compila y vincula las rutinas y funciones en FORTRAN de la biblioteca SIMBOLIC.CNEA, cargando el módulo resultante, con el nombre RUTIFORT, en la biblioteca REALES.CNEA; en el PASO2 el procedimiento LKED vincula las rutinas y funciones en lenguaje absoluto, cargando el módulo resultante en una biblioteca temporaria particionada que se crea en ese momento y reside en un disco auxiliar; en el PASO3 el procedimiento LKEDG vincula los módulos HAZ, RUTIFORT y LONGTRAZ y ejecuta.

```
//..... JOB ....
//PASO1 EXEC PROC=FTG1CL
//FORT.SYSIN DD DSN=SIMBOLIC.CNEA(PORCENT),DISP=SHR
//          DD DSN=SIMBOLIC.CNEA(ANGESPAC),DISP=SHR
//          DD DSN=SIMBOLIC.CNEA(MOMENTO),DISP=SHR
```

```

//LKED.SYSLMOD DD DSN=REALES.CNEA(RUTIFORT),SPACE=,DISP=OLD
/*
//PASO2 EXEC PROC=LKED
//LKED.SYSLMOD DD DSN=TEMPOR.ADR(LONGTRAZ),UNIT=WORKDISK,
//  DISP=(,PASS),DCB=BLKSIZE=80,SPACE=(CYL,(1,1,1))
//LKED.SYSIN DD DSN=LONGTRAZ,UNIT=TAPE,VOL=SER=T37195,LABEL=(16,SL),
//  DISP=(OLD,KEEP)
/*
//PASO3 EXEC PROC=LKEDG
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
//      DD DSN=TEMPOR.ADR,DISP=(OLD,DELETE)
//LKED.SYSIN DD *,DCB=BLKSIZE=80
//      INCLUDE OTRALIB(HAZ)
//      INCLUDE OTRALIB(RUTIFORT)
//      INCLUDE OTRALIB(LONGTRAZ)
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
//      Aquí tarjetas perforadas de DATOS
/*
//

```

NOTA: En el programa principal ASSEMBLER que dió origen al módulo HAZ, se abrió un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

PUNTO 7.

Ejecución de programas en tarjetas perforadas, que invocan rutinas y funciones en lenguaje simbólico residentes en una biblioteca de nombre SIMBOLIC.CNEA, a mas de otras rutinas y funciones que son módulos de carga de una biblioteca de nombre REALES.CNEA.

Modelo 260 . Ejecución de un programa en tarjetas perforadas en lenguaje simbólico FORTRAN, que invoca rutinas y funciones en lenguaje FORTRAN residentes en la biblioteca de simbólicos con los nombres INPUT2, CARGA y PLISAM, a mas de otras rutinas y funciones grabadas en la biblioteca de módulos de carga con los nombres PATIN, INPUT1 y SALDAM.

```
//..... JOB ....
//PAS01 EXEC PROC=FTG1CLG
//FORT.SYSIN DD DSN=SIMBOLIC.CNEA(INPUT2),DISP=SHR
//          DD DSN=SIMBOLIC.CNEA(CARGA),DISP=SHR
//          DD DSN=SIMBOLIC.CNEA(PLISAM),DISP=SHR
//          DD *
  Aquí tarjetas perforadas en FORTRAN
/*
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
//LKED.SYSIN DD *,DCB=BLKSIZE=80
              INCLUDE OTRALIB(PATIN)
              INCLUDE OTRALIB(INPUT1)
              INCLUDE OTRALIB(SALDAM)
/*
//GO.SYSIN DD *
  Aquí tarjetas perforadas de DATOS
/*
//
```

Modelo 261 . Ejecución de un programa en tarjetas perforadas en lenguaje absoluto de FORTRAN, que invoca rutinas y funciones en lenguaje FORTRAN residentes en la biblioteca de simbólicos con los nombres PALTEN, SIERRAP y URANIO, a mas de otras rutinas y funciones grabadas en la biblioteca de módulos de carga con los nombres ISOTOP01, ISOTOP03 y PESOATOM.

```
//..... JOB ....
//PAS01 EXEC PROC=FTG1CLG
//FORT.SYSIN DD DSN=SIMBOLIC.CNEA(PALTEN),DISP=SHR
//          DD DSN=SIMBOLIC.CNEA(SIERRAP),DISP=SHR
//          DD DSN=SIMBOLIC.CNEA(URANIO),DISP=SHR
```

```

/*
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
//LKED.SYSIN DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas en absoluto de FORTRAN
      INCLUDE OTRALIB(ISOTCP01)
      INCLUDE OTRALIB(ISOTOPO3)
      INCLUDE OTRALIB(PESOATOM)
/*
//GO.SYSIN DD *
  Aquí tarjetas perforadas de DATOS
/*
//

```

Modelo 262. Ejecución de un programa en tarjetas perforadas en lenguaje simbólico FORTRAN, que invoca rutinas y funciones en lenguaje ASSEMBLER residentes en la biblioteca de simbólicos con el nombre ELECTIMP, a mas de otras rutinas y funciones grabadas en la biblioteca de módulos con los nombres HOTPLSM, NEUTRHE y QUANMECH. El problema se puede resolver en dos pasos: en el PASO1 el procedimiento ASMFCL compila y vincula las rutinas y funciones en ASSEMBLER de la biblioteca SIMBOLIC.CNEA, cargando el módulo resultante en la biblioteca REALES.CNFA; en el PASO2 el procedimiento FTG1CLG compila y vincula el programa en FORTRAN, lo vincula con los módulos ELECTIMP, HOTPLSM, NEUTRHE y QUANMECH y ejecuta.

```

//..... JOB ....
//PASO1 EXEC PROC=ASMFCL
//ASM.SYSIN DD DSN=SIMBOLIC.CNEA(ELECTIMP),DISP=SHR
//LKED.SYSMOD DD DSN=REALES.CNEA(ELECTIMP),SPACE=,DISP=OLD
/*
//PASO2 EXEC PROC=FTG1CLG
//FORT.SYSIN DD *
  Aquí tarjetas perforadas en FORTRAN
/*
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
//LKED.SYSIN DD *,DCB=BLKSIZE=80
      INCLUDE OTRALIB(ELECTIMP)
      INCLUDE OTRALIB(HOTPLSM)
      INCLUDE OTRALIB(NEUTRHE)
      INCLUDE OTRALIB(QUANMECH)
/*
//GO.SYSIN DD *
  Aquí tarjetas perforadas de DATOS
/*
//

```

Modelo 263. Ejecución de un programa en tarjetas perforadas en lenguaje absoluto de FORTRAN, que invoca rutinas y funciones en lenguaje ASSEMBLER residentes en

la biblioteca de simbólicos con el nombre AMPLITUD, a mas de otras rutinas y funciones grabadas en la biblioteca de módulos de carga con los nombres UNITARIO y ANALITIC. El problema se puede resolver en dos pasos: en el PASO1 el procedimiento ASMFCL compila y vincula las rutinas y funciones en ASSEMBLER de la biblioteca SIMBOLIC.CNEA, cargando el módulo resultante en la biblioteca REALES.CNEA; en el PASO2 el procedimiento FTG1LG vincula el programa en absoluto de FORTRAN con los módulos AMPLITUD, UNITARIO y ANALITIC y ejecuta.

```
//..... JOB ....
//PASO1 EXEC PROC=ASMFCL
//ASM.SYSIN DD DSN=SIMBOLIC.CNEA(AMPLITUD),DISP=SHR
//LKED.SYSLMOD DD DSN=REALES.CNEA(AMPLITUD),SPACE=,DISP=OLD
/*
//PASO2 EXEC PROC=FTG1LG
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
//LKED.SYSIN DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas en absoluto de FORTRAN
      INCLUDE OTRALIB(AMPLITUD)
      INCLUDE OTFALIB(UNITARIO)
      INCLUDE OTRALIB(ANALITIC)
/*
//GO.SYSIN DD *
  Aquí tarjetas perforadas de DATOS
/*
//
```

Modelo 264 . Ejecución de un programa en tarjetas perforadas en lenguaje simbólico ASSEMBLER, que invoca rutinas y funciones en lenguaje ASSEMBLER residentes en la biblioteca de simbólicos con el nombre YBANDS, a mas de otras rutinas y funciones grabadas en la biblioteca de módulos de carga con los nombres ENERGIES y SURFACES. El problema se puede resolver en dos pasos: en el PASO1 el procedimiento ASMFCL compila y vincula las rutinas y funciones en ASSEMBLER de la biblioteca SIMBOLIC.CNEA, cargando el módulo resultante en la biblioteca REALES.CNEA; en el PASO2 el procedimiento ASMFCLG compila y vincula el programa en ASSEMBLER, lo vincula con los módulos ENERGIES y SURFACES y ejecuta.

```
//..... JOB ....
//PASO1 EXEC PROC=ASMFCL
//ASM.SYSIN DD DSN=SIMBOLIC.CNEA(YBANDS),DISP=SHR
//LKED.SYSLMOD DD DSN=REALES.CNEA(YBANDS),SPACE=,DISP=OLD
/*
```



```
//PAS02 EXEC PROC=ASMFCLG
//ASM.SYSIN DD *
  Aquí tarjetas perforadas en ASSEMBLER
/*
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
//LKED.SYSIN DD *,DCB=BLKSIZE=80
      INCLUDE OTRALIB(YBANDS)
      INCLUDE OTRALIB(ENERGIES)
      INCLUDE OTRALIB(SURFACES)
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: En el programa principal ASSEMBLER en tarjetas perforadas, se ha abierto un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

Modelo 265 . Ejecución de un programa en tarjetas perforadas en lenguaje absoluto de ASSEMBLER, que invoca rutinas y funciones en lenguaje ASSEMBLER residentes en la biblioteca de simbólicos con el nombre RANDOM a mas de otras rutinas y funciones grabadas en la biblioteca de módulos de carga con los nombres BEHAVI, CRITICAL y SYSTEMS. El problema se puede resolver en dos pasos: en el PAS01 el procedimiento ASMFCL compila y vincula las rutinas y funciones en ASSEMBLER de la biblioteca SIMBOLIC.CNEA, cargando el módulo resultante en la biblioteca REALES.CNEA; en el PAS02 el procedimiento LKEDG vincula el programa en absoluto de ASSEMBLER con los módulos RANDOM, BEHAVI, CRITICAL y SYSTEMS y ejecuta.

```
//..... JOB ....
//PAS01 EXEC PROC=ASMFCL
//ASM.SYSIN DD DSN=SIMBOLIC.CNEA(RANDOM),DISP=SHR
//LKED.SYSMOD DD DSN=REALES.CNEA(RANDOM),SPACE=,DISP=OLD
/*
//PAS02 EXEC PROC=LKEDG
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
//LKED.SYSIN DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas en absoluto de ASSEMBLER
      INCLUDE OTRALIB(RANDOM)
      INCLUDE OTRALIB(BEHAVI)
      INCLUDE OTRALIB(CRITICAL)
      INCLUDE OTRALIB(SYSTEMS)
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
```

```
//GO.TARJETA DD *,DCB=BLKSIZE=80
Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: En el programa principal ASSEMBLER que dió origen al lote de tarjetas en absoluto, se abrió un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

Modelo 266. Ejecución de un programa en tarjetas perforadas en lenguaje simbólico ASSEMBLER, que invoca rutinas y funciones en lenguaje FORTRAN residentes en la biblioteca de simbólicos con los nombres QUADRPOL, MAGNET y SYNCHROT, a mas de otras rutinas y funciones grabadas en la biblioteca de módulos de carga con los nombres POWER y SUPPLIES. El problema se puede resolver en dos pasos: en el PASO1 el procedimiento FTG1CL compila y vincula las rutinas y funciones en FORTRAN de la biblioteca SIMBOLIC.CNEA, cargando el módulo resultante, con el nombre NINA, en la biblioteca REALES.CNEA; en el PASO2 el procedimiento ASMFCLG compila y vincula el programa en ASSEMBLER con los módulos NINA, POWER y SUPPLIES y ejecuta.

```
//..... JOB ....
//PASO1 EXEC PROC=FTG1CL
//FORT.SYSIN DD DSN=SIMBOLIC.CNEA(QUADRPOL),DISP=SHR
//          DD DSN=SIMBOLIC.CNEA(MAGNET),DISP=SHR
//          DD DSN=SIMBOLIC.CNEA(SYNCHROT),DISP=SHR
//LKED.SYSIMOD DD DSN=REALES.CNEA(NINA),SPACE=,DISP=OLD
/*
//PASO2 EXEC PROC=ASMFCLG
//ASM.SYSIN DD *
Aquí tarjetas perforadas en ASSEMBLER
/*
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
//LKED.SYSIN DD *,DCB=BLKSIZE=80
          INCLUDE OTRALIB(NINA)
          INCLUDE OTRALIB(POWER)
          INCLUDE OTRALIB(SUPPLIES)
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: En el programa principal ASSEMBLER en tarjetas perforadas, se ha abierto un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

Modelo 267. Ejecución de un programa en tarjetas perforadas en lenguaje absoluto de ASSEMBLER, que invoca rutinas y funciones en lenguaje FORTRAN residentes en la biblioteca de simbólicos con los nombres PARAMET, CUADMIN y COEFIC, a mas de otras rutinas y funciones grabadas en la biblioteca de módulos de carga con los nombres PINTI55, PINTI60 y SAGASTI. El problema se puede resolver en dos pasos: en el PASO1 el procedimiento FTG1CL compila y vincula las rutinas y funciones en FORTRAN de la biblioteca SIMBOLIC.CNEA, cargando el módulo resultante, con el nombre AJUSTIN, en la biblioteca REALES.CNEA; en el PASO2 el procedimiento LKEDG vincula el programa en absoluto de ASSEMBLER con los módulos AJUSTIN, PINTI55, PINTI60 y SAGASTI y ejecuta.

```
//..... JOB ....
//PASO1 EXEC PROC=FTG1CL
//FORT.SYSIN DD DSN=SIMBOLIC.CNEA(PARAMET),DISP=SHR
//          DD DSN=SIMBOLIC.CNEA(CUADMIN),DISP=SHR
//          DD DSN=SIMBOLIC.CNEA(COEFIC),DISP=SHR
//LKED.SYSLMOD DD DSN=REALES.CNEA(AJUSTIN),SPACE=,DISP=OLD
/*
//PASO2 EXEC PROC=LKEDG
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
//LKED.SYSIN DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas en absoluto de ASSEMBLER
      INCLUDE OTRALIB(AJUSTIN)
      INCLUDE OTRALIB(PINTI55)
      INCLUDE OTRALIB(PINTI60)
      INCLUDE OTRALIB(SAGASTI)
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: En el programa principal ASSEMBLER que dió origen al lote de tarjetas en absoluto se abrió un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

PUNTO 8.

Ejecución de programas grabados en cintas magnéticas, que invocan rutinas y funciones en lenguaje simbólico residentes en una biblioteca de nombre SIMBOLIC.CNEA, a mas de otras rutinas y funciones que son módulos de carga de una biblioteca de nombre REALES.CNEA.

Modelo 268. Ejecución de un programa en lenguaje simbólico FORTRAN que está grabado, con el nombre FASE, en el rótulo 23 de una cinta magnética e invoca rutinas y funciones en lenguaje FORTRAN residentes en la biblioteca de simbólicos con los nombres EJES, FLUJO y RED, a mas de otras rutinas y funciones cargadas en la biblioteca de módulos con los nombres TEMPERAT y MAGNETIS. La cinta lleva el número de carrete 52336 y tiene etiquetas modelo IBM.

```
//..... JOB ....
//PAS01 EXEC PROC=FTG1CLG
//FORT.SYSIN DD DSN=FASE,UNIT=TAPE,VOL=SER=T52336,LABEL=(23,SL),
//  DISP=(OLD,KEEP)
//          DD DSN=SIMBOLIC.CNEA(EJES),DISP=SHR
//          DD DSN=SIMBOLIC.CNEA(FLUJO),DISP=SHR
//          DD DSN=SIMBOLIC.CNEA(RED),DISP=SHR
/*
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
//LKED.SYSIN DD *,DCB=BLKSIZE=80
//          INCLUDE OTRALIB(TEMPERAT)
//          INCLUDE OTRALIB(MAGNETIS)
/*
//GO.SYSIN DD *
//Aquí tarjetas perforadas de DATOS
/*
//
```

Modelo 269. Ejecución de un programa en lenguaje absoluto de FORTRAN que está grabado, con el nombre NEWSHAD, en el rótulo 18 de una cinta magnética e invoca rutinas y funciones en lenguaje FORTRAN residentes en la biblioteca de simbólicos con los nombres FOTON, DEUTERON y SECCEFIC, a mas de otras rutinas y funciones cargadas en la biblioteca de módulos con los nombres CORREC, SHADOW y TOTAL. La cinta lleva el número de carrete 01102 y tiene etiquetas modelo IBM. El problema se puede resolver en tres pasos: en el PAS01 el procedimiento LKED vincula el programa en absoluto de FORTRAN, cargando el módulo resultante en una biblioteca

temporaria particionada que se crea en ese momento y reside en un disco auxiliar; en el PASO2 el procedimiento FTG1CL compila y vincula las rutinas y funciones en FORTRAN de la biblioteca SIMBOLIC.CNEA, cargando el módulo resultante, con el nombre FOTOSEC, en la biblioteca REALES.CNEA; en el PASO3 el procedimiento FTG1LG vincula todos los módulos y ejecuta.

```
//..... JOB ....
//PASO1 EXEC PROC=LKED
//LKED.SYSLMOD DD DSN=TEMPOR.ADR(NEWSHAD),UNIT=WORKDISK,
//  DISP=(,PASS),DCB=BLKSIZE=80,SPACE=(CYL,(1,1,1))
//LKED.SYSIN DD DSN=NEWSHAD,UNIT=TAPE,VOL=SER=T01102,LABEL=(18,SL),
//  DISP=(OLD,KEEP)
/*
//PASO2 EXEC PROC=FTG1CL
//FORT.SYSIN DD DSN=SIMBOLIC.CNEA(FOTON),DISP=SHR
//          DD DSN=SIMBOLIC.CNEA(DEUTERON),DISP=SHR
//          DD DSN=SIMBOLIC.CNEA(SECCEPIC),DISP=SHR
//LKED.SYSLMOD DD DSN=REALES.CNEA(FOTOSEC),SPACE=,DISP=OLD
/*
//PASO3 EXEC PROC=FTG1LG
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
//          DD DSN=TEMPOR.ADR,DISP=(OLD,DELETE)
//LKEL.SYSIN DD *,DCB=BLKSIZE=80
//          INCLUDE OTRALIB(NEWSHAD)
//          INCLUDE OTRALIB(FOTOSEC)
//          INCLUDE OTRALIB(CORREC)
//          INCLUDE OTRALIB(SHADOW)
//          INCLUDE OTRALIB(TOTAL)
/*
//GO.SYSIN DD *
Aquí tarjetas perforadas de DATOS
/*
//
```

Modelo 270. Ejecución de un programa en lenguaje absoluto de FORTRAN, que está grabado con el nombre CHAR, en el rótulo 14 de una cinta magnética e invoca rutinas y funciones en lenguaje ASSEMBLER residentes en la biblioteca de simbólicos con el nombre ACELERAD, a mas de otras rutinas y funciones cargadas en la biblioteca de módulos con los nombres MECANICA y NUEVO. La cinta lleva el número de carrete 37002 y tiene etiquetas modelo IBM. El problema se puede resolver en tres pasos: en el PASO1 el procedimiento LKED vincula el programa en absoluto de FORTRAN, cargando el módulo resultante en una biblioteca temporaria particionada que se crea en ese momento y reside en un disco auxiliar; en el PASO2 el procedi-

miento ASMFCL compila y vincula las rutinas y funciones en ASSEMBLER de la biblioteca SIMBOLIC.CNEA, cargando el módulo resultante en la biblioteca REALES.CNEA; en el PASO3 el procedimiento FTG1LG vincula todos los módulos y ejecuta.

```
//..... JOB ....
//PASO1 EXEC PROC=LKED
//LKED.SYSLMOD DD DSN=TEMPOR.ADR(CHAR),UNIT=WORKDISK,
//  DISP=(NEW,PASS),DCB=BLKSIZE=80,SPACE=(CYL,(1,1,1))
//LKED.SYSIN DD DSN=CHAR,UNIT=TAPE,VOL=SER=T37002,LABEL=(14,SL),
//  DISP=(OLD,KEEP)
/*
//PASO2 EXEC PROC=ASMFCL
//ASM.SYSIN DD DSN=SIMBOLIC.CNEA(ACELERAD),DISP=SHR
//LKED.SYSLMOD DD DSN=REALES.CNEA(ACELERAD),SPACE=,DISP=OLD
/*
//PASO3 EXEC PROC=FTG1LG
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
//          DD DSN=TEMPOR.ADR,DISP=(OLD,DELETE)
//LKED.SYSIN DD *,DCB=BLKSIZE=80
//          INCLUDE OTRALIB(CHAR)
//          INCLUDE OTRALIB(ACELERAD)
//          INCLUDE OTRALIB(MECANICA)
//          INCLUDE OTRALIB(NUEVO)
/*
//GO.SYSIN DD *
//  Aquí tarjetas perforadas de DATOS
/*
//
```

Modelo 271 . Ejecución de un programa en lenguaje simbólico ASSEMBLER, que está grabado con el nombre TLUMINIS, en el rótulo 38 de una cinta magnética e invoca rutinas y funciones en lenguaje ASSEMBLER, residentes en la biblioteca de simbólicos con el nombre CRISTAL, a mas de otras rutinas y funciones cargadas en la biblioteca de módulos con los nombres IMPUREZA, TEMPERAT y RANGOS. La cinta lleva el número de carrete 53408 y tiene etiquetas modelo IBM. El problema se puede resolver en dos pasos: en el PASO1 el procedimiento ASMFCL compila y vincula las rutinas y funciones en ASSEMBLER de la biblioteca SIMBOLIC.CNEA, cargando el módulo resultante en la biblioteca REALES.CNEA; en el PASO2 el procedimiento ASMFCLG compila y vincula el programa en ASSEMBLER de la cinta, vincula todos los módulos y ejecuta.

```
//..... JOB ....
//PASO1 EXEC PROC=ASMFCL
```

```
//ASM.SYSIN DD DSN=SIMBOLIC.CNEA(CRISTAL),DISP=SHR
//LKED.SYSLMOD DD DSN=REALES.CNEA(CRISTAL),SPACE=,DISP=OLD
/*
//PASO2 EXEC PROC=ASMFCLG
//ASM.SYSIN DD DSN=TLUMINIS,UNIT=TAPE,VOL=SER=T53408,LABEL=(38,SL),
// DISP=(OLD,KEEP)
/*
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
//LKED.SYSIN DD *,DCB=BLKSIZE=80
        INCLUDE OTRALIB(CRISTAL)
        INCLUDE OTRALIB(IMPUREZA)
        INCLUDE OTRALIB(TEMPERAT)
        INCLUDE OTRALIB(RANGOS)
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: En el programa principal ASSEMBLER de la cinta, se ha abierto un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

Modelo 272 . Ejecución de un programa en lenguaje absoluto de ASSEMBLER, que está grabado con el nombre TIERRARS, en el rótulo 22 de una cinta magnética e invoca rutinas y funciones en lenguaje ASSEMBLER, residentes en la biblioteca de simbólicos con el nombre ALCALI, a mas de otras rutinas y funciones cargadas en la biblioteca de módulos con los nombres BRILLO y DOSIMET. La cinta lleva el número de carrete 19902 y tiene etiquetas modelo IBM. El problema se puede resolver en tres pasos: en el PASO1 el procedimiento ASMFCL compila y vincula las rutinas y funciones en ASSEMBLER de la biblioteca SIMBOLIC.CNEA, cargando el módulo resultante en la biblioteca REALES.CNEA; en el PASO2 el procedimiento LKED vincula el programa en absoluto de ASSEMBLER, cargando el módulo resultante en una biblioteca temporaria particionada que se crea en ese momento y reside en un disco auxiliar; en el PASO3 el procedimiento LKEDG vincula todos los módulos y ejecuta.

```
//..... JOB ....
//PASO1 EXEC PROC=ASMFCL
//ASM.SYSIN DD DSN=SIMBOLIC.CNEA(ALCALI),DISP=SHR
//LKED.SYSLMOD DD DSN=REALES.CNEA(ALCALI),SPACE=,DISP=OLD
/*
```

```

//PASO2 EXEC PROC=LKED
//LKED.SYSLMOD DD DSN=TEMPOR.ADR(TIERRARS),UNIT=WORKDISK,
//  DISP=(NEW,PASS),DCB=BLKSIZE=80,SPACE=(CYL,(1,1,1))
//LKED.SYSIN DD DSN=TIERRARS,UNIT=TAPE,VOL=SER=119902,LABEL=(22,SL),
//  DISP=(OLD,KEEP)
/*
//PASO3 EXEC PROC=LKEDG
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
//          DD DSN=TEMPOR.ADR,DISP=(OLD,DELETE)
//LKED.SYSIN DD *,DCB=BLKSIZE=80
//          INCLUDE OTRALIB(TIERRARS)
//          INCLUDE OTRALIB(ALCALI)
//          INCLUDE OTRALIB(BRILLO)
//          INCLUDE OTRALIB(DOSIMET)
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
//  Aquí tarjetas perforadas de DATOS
/*
//

```

NOTA: En el programa principal ASSEMBLER que dió origen al módulo TIERRARS de la cinta, se abrió un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

Modelo 273 Ejecución de un programa en lenguaje simbólico ASSEMBLER, que está grabado con el nombre PERFIL, en el rótulo 16 de una cinta magnética e invoca rutinas y funciones en lenguaje FORTRAN residentes con los nombres PLANES y PERIODIC en la biblioteca de simbólicos, a mas de otras rutinas y funciones cargadas en la biblioteca de módulos con los nombres GRADO, PRESION y DENSIDAD. La cinta lleva el número de carrete 00600 y tiene etiquetas modelo IBM. El problema se puede resolver en dos pasos: en el PASO1 el procedimiento FTG1CL compila y vincula las rutinas y funciones en FORTRAN de la biblioteca SIMBOLIC.CNEA, cargando el módulo resultante, con el nombre PERIPLAN, en la biblioteca REALES.CNEA; en el PASO2 el procedimiento ASMFCLG compila y vincula el programa en ASSEMBLER de la cinta, vincula todos los módulos y ejecuta.

```

//..... JOB ....
//PASO1 EXEC PROC=FTG1CL
//FORT.SYSIN DD DSN=SIMBOLIC.CNEA(PANES),DISP=SHR
//          DD DSN=SIMBOLIC.CNEA(PERIODIC),DISP=SHR
//LKED.SYSLMOD DD DSN=REALES.CNEA(PERIPLAN),SPACE=,DISP=OLD
/*

```



```
//PASO2 EXEC PROC=ASMFCGLG
//ASM.SYSIN DD DSN=PERFIL,UNIT=TAPE,VOL=SER=T00600,LABEL=(16,SL),
// DISP=(OLD,KEEP)
/*
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
//LKED.SYSIN DD *,DCB=BLKSIZE=80
        INCLUDE OTRALIB(PERIPLAN)
        INCLUDE OTRALIB(GRADO)
        INCLUDE OTRALIB(PRESION)
        INCLUDE OTRALIB(DENSIDAD)
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: En el programa principal ASSEMBLER de la cinta se ha abierto un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

Modelo 274 Ejecución de un programa en lenguaje absoluto de ASSEMBLER, que está grabado, con el nombre HIDRODIN, en el rótulo 47 de una cinta magnética e invoca rutinas y funciones en lenguaje FORTRAN, residentes en la biblioteca de simbólicos con los nombres EXPANS, METODO1 y METODO2 a mas de otras rutinas y funciones cargadas en la biblioteca de módulos con los nombres FERROMAG y PICOS. La cinta lleva el número de carrete 91807 y tiene etiquetas modelo IBM. El problema se puede resolver en tres pasos: en el PASO1 el procedimiento LKED vincula el programa en absoluto de ASSEMBLER, cargando el módulo resultante en una biblioteca temporaria particionada que se crea en ese momento y reside en un disco auxiliar; en el PASO2 el procedimiento FTGLCL compila y vincula las rutinas y funciones en FORTRAN de la biblioteca SIMBOLIC.CNEA, cargando el módulo resultante, con el nombre METODOS, en la biblioteca REALES.CNEA; en el PASO3 el procedimiento FTGLLG vincula todos los módulos y ejecuta.

```
//..... JOB ....
//PASO1 EXEC PROC=LKED
//LKED.SYSLMOD DD DSN=TEMPOR.ADR(HIDRODIN),UNIT=WORKDISK,
// DISP=(NEW,PASS),DCB=BLKSIZE=80,SPACE=(CYL,(1,1,1))
//LKED.SYSIN DD DSN=HIDRODIN,UNIT=TAPE,VOL=SER=T91807,LABEL=(47,SL),
// DISP=(OLD,KEEP)
/*
```

```

//PASO2 EXEC PROC=FTG1CL
//FORT.SYSIN DD DSN=SIMBOLIC.CNEA(EXPANS),DISP=SHR
//          DD DSN=SIMBOLIC.CNEA(METODO1),DISP=SHR
//          DD DSN=SIMBOLIC.CNEA(METODO2),DISP=SHR
//LKED.SYSLMOD DD DSN=REALES.CNEA(METODOS),SPACE=,DISP=OLD
/*
//PASO3 EXEC PROC=LKEDG
//LKED.OTRALIB DD DSN=REALES.CNEA,DISP=SHR
//          DD DSN=TEMPOR.ADR,DISP=(OLD,DELETE)
//LKED.SYSIN DD *,DCB=BLKSIZE=80
//          INCLUDE OTRALIB(HIDRODIN)
//          INCLUDE OTRALIB(METODOS)
//          INCLUDE OTRALIB(FERROMAG)
//          INCLUDE OTRALIB(PICOS)
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
//          Aqui tarjetas perforadas de DATOS
/*
//

```

NOTA: En el programa principal ASSEMBLER que dió origen al módulo HIDRODIN grabado en la cinta, se abrió un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

PUNTO 9.

Ejecución de módulos de carga procedentes de FORTRAN, de la biblioteca REALES.CNEA, que utilizan archivos de entrada de datos residentes en una biblioteca particionada y catalogada de programas en lenguaje simbólico, de nombre SIMBOLIC.CNEA.

Modelo 275 . Ejecución de un módulo de carga procedente de FORTRAN, grabado en la biblioteca REALES.CNEA con el nombre ADD, cuyos datos de entrada constituyen un archivo que reside en la biblioteca SIMBOLIC.CNEA, como un miembro independiente, con el nombre MATRDAT. El problema se puede resolver en dos pasos: en el PASO1 el utilitario IEBGENER transfiere el archivo de la biblioteca de simbólicos a un disco auxiliar; en el PASO2 se ejecuta el programa ADD obteniendo del disco auxiliar el archivo de entrada de datos.

```
//..... JOB ....
//PASO1 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA(MATRDAT),DISP=SHR
//SYSUT2 DD DSN=MATRDAT,UNIT=WORKDISK,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//SYSIN DD DUMMY
/*
//PASO2 EXEC PGM=ADD
//STEPLIB DD DSN=REALES.CNEA,DISP=SHR
/*
//FT16F001 DD DSN=MATRDAT,UNIT=WORKDISK,DISP=(OLD,DELETE)
//FT06F001 DD SYSOUT=A
/*
//
```

NOTA: El archivo de entrada de datos está definido en la sentencia FT16F001 del PASO2, en donde se le asigna el número de referencia 16; por lo tanto en el programa fuente que dió origen al módulo ADD, se codificó READ16 para leer datos de ese archivo. Se menciona el DCB con el cual se grabará el disco auxiliar.

Modelo 276 . Ejecución de un módulo de carga procedente de FORTRAN, grabado en la biblioteca REALES.CNEA con el nombre SUPERTAB, cuyos datos de entrada son tres archivos distintos que residen en la biblioteca SIMBOLIC.CNEA, como miembros independientes, con los nombres DATTAB1, DATTAB2 y DATTAB3. El problema se puede

resolver en cuatro pasos: en los tres primeros (PAS01, PAS02 y PAS03) el utilitario IEBGENER transfiere, de a uno, los tres archivos de la biblioteca de símbolos al mismo disco auxiliar; en el PAS04 se ejecuta el programa SUPERTAB obteniendo los archivos de entrada de datos del disco auxiliar.

```
//..... JOB ....
//PAS01 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA(DATTAB1),DISP=SHR
//SYSUT2 DD UNIT=WORKDISK,DISP=(NEW,PASS),SPACE=(CYL,(1,1)),
//   DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB),DSN=DATTAB1
//SYSIN DD DUMMY
/*
//PAS02 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA(DATTAB2),DISP=SHR
//SYSUT2 DD VOL=REF=*.PAS01.SYSUT2,DISP=(NEW,PASS),
//   SPACE=(CYL,(1,1)),DSN=DATTAB2
//SYSIN DD DUMMY
/*
//PAS03 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA(DATTAB3),DISP=SHR
//SYSUT2 DD VOL=REF=*.PAS01.SYSUT2,DISP=(NEW,PASS),
//   SPACE=(CYL,(1,1)),DSN=DATTAB3
//SYSIN DD DUMMY
/*
//PAS04 EXEC PGM=SUPERTAB
//STEPLIB DD DSN=REALES.CNEA,DISP=SHR
/*
//FT08F001 DD DSN=DATTAB1,UNIT=WORKDISK,DISP=(OLD,DELETE)
//FT04F001 DD DSN=DATTAB2,UNIT=WORKDISK,DISP=(OLD,DELETE)
//FT11F001 DD DSN=DATTAB3,UNIT=WORKDISK,DISP=(OLD,DELETE)
//FT06F001 DD SYSOUT=A
/*
//
```

NOTA: Los archivos de entrada de datos están definidos en la sentencia FT08F001 y siguientes del PAS04, en donde se les asignan los números de referencia 8, 4 y 11; por lo tanto en el programa fuente que dió origen al módulo SUPERTAB, se codificó READ8, READ4 y READ11 respectivamente, para leer datos de esos archivos. Se menciona el DCB con el cual se grabará el disco auxiliar.

PUNTO 10.

Ejecución de módulos de carga procedentes de FORTRAN de la biblioteca de nombre REALES.CNEA, que graban archivos de salida de datos en una biblioteca particionada y catalogada de programas en lenguaje simbólico, de nombre SIMBOLIC.CNEA.

Modelo 277 . Ejecución de un módulo de carga procedente de FORTRAN, grabado en la biblioteca REALES.CNEA con el nombre FORMULA9, cuyos resultados constituyen un archivo de salida que se graba en la biblioteca SIMBOLIC.CNEA, como un miembro independiente. El problema se puede resolver en dos pasos: en el PAS01 se ejecuta el programa FORMULA9, grabándose los resultados, con el nombre SALIDFOR, en un disco auxiliar; en el PAS02 el utilitario IEBUPDTE transfiere ese archivo del disco auxiliar a la biblioteca de simbólicos.

```
//..... JOB ....
//PAS01 EXEC PGM=FORMULA9
//STEPLIB DD DSN=REALES.CNEA,DISP=SHR
/*
//FT15F001 DD DSN=SALIDFOR,UNIT=WORKDISK,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,IRECL=80,RECFM=FB)
//FT06F001 DD SYSOUT=A
//FT05F001 DD *
  Aquí tarjetas perforadas de DATOS
/*
//PAS02 EXEC PGM=IEBUPDTE,PARM=NEW
//SYSPRINT DD SYSOUT=A
//SYSUT2 DD DSN=SIMBOLIC.CNEA,DISP=OLD
//SYSIN DD *
./ ADD NAME=SALIDFOR
// DD DSN=SALIDFOR,DISP=(OLD,DELETE)
./ ENDUP
/*
//
```

NOTA: El archivo de salida de datos está definido en la sentencia FT15F001 del PAS01, en donde se le asigna el número de referencia 15; por lo tanto en el programa fuente que dió origen al módulo FORMULA9, se codificó WRITE15 para grabar resultados en ese archivo. Se menciona el DCB con el cual se grabará el disco auxiliar.

Modelo 278 . Ejecución de un módulo de carga procedente de FORTRAN, grabado en

la biblioteca REALES.CNEA con el nombre PGMGRABA, cuyos resultados son tres archivos de salida distintos, que se graban en la biblioteca SIMBOLIC.CNEA, como miembros independientes. El problema se puede resolver en dos pasos: en el PASO1 se ejecuta el programa PGMGRABA, grabándose los resultados, con los nombres RESGRAB1, RESGRAB2 y RESGRAB3, en un disco auxiliar; en el PASO2 el utilitario IEBUPDTE transfiere esos archivos del disco auxiliar a la biblioteca de simbólicos.

```
//..... JOB ....
//PASO1 EXEC PGM=PGMGRABA
//STEPLIB DD DSN=REALES.CNEA,DISP=SHR
/*
//FT13F001 DD DSN=RESGRAB1,UNIT=WORKDISK,DISP=(NEW,PASS),
//  SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//FT12F001 DD DSN=RESGRAB2,VOL=REF=*.FT13F001,DISP=(,PASS),
//  SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//FT10F001 DD DSN=RESGRAB3,VOL=REF=*.FT13F001,DISP=(,PASS),
//  SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//FT06F001 DD SYSOUT=A
//FT05F001 DD *
  Aquí tarjetas perforadas de DATOS
/*
//PASO2 EXEC PGM=IEBUPDTE,PARM=MOD
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA,DISP=SHR
//SYSUT2 DD DSN=SIMBOLIC.CNEA,DISP=OLD
//SYSIN DD *
./  ADD  NAME=RESGRAB1
/*
//      DD DSN=RESGRAB1,DISP=(OLD,DELETE)
//      DD *
./  ADD  NAME=RESGRAB2
/*
//      DD DSN=RESGRAB2,DISP=(OLD,DELETE)
//      DD *
./  ADD  NAME=RESGRAB3
/*
//      DD DSN=RESGRAB3,DISP=(OLD,DELETE)
//      DD *
./  ENDUP
/*
//
```

NOTA: Los archivos de salida de datos están definidos en la sentencia FT13F001 y siguientes del PASO1, en donde se les asignan los números de referencia 13, 12 y 10; por lo tanto en el programa fuente que dió origen al módulo PGMGRABA, se co-

dificó WRITE13, WRITE12 y WRITE10 respectivamente, para grabar resultados en esos archivos. Se menciona el DCB con el cual se grabará el disco auxiliar.

PUNTO 11.

Ejecución de módulos de carga procedentes de FORTRAN de la biblioteca de nombre REALES.CNEA, que utilizan archivos de entrada de datos residentes en una biblioteca particionada y catalogada de programas en lenguaje simbólico, de nombre SIMBOLIC.CNEA y simultáneamente graban archivos de salida de datos en esa misma biblioteca.

Modelo 279 . Ejecución de un módulo de carga procedente de FORTRAN, grabado en la biblioteca REALES.CNEA con el nombre LEEGRABA, cuyos datos de entrada constituyen un archivo que reside en la biblioteca SIMBOLIC.CNEA, como un miembro independiente y cuyos resultados se graban en esa misma biblioteca, también como un miembro independiente. El problema se puede resolver en tres pasos: en el PASO1 el utilitario IEBGENER transfiere el archivo de nombre DATOS14 de la biblioteca de simbólicos a un disco auxiliar; en el PASO2 se ejecuta el programa LEEGRABA obteniéndose los datos de entrada del disco auxiliar y grabando los resultados, con el nombre RESUL14, también en un disco auxiliar; en el PASO3 el utilitario IEBUPDTE transfiere este último archivo del disco a la biblioteca SIMBOLIC.CNEA.

```
//..... JOB ....
//PASO1 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA(DATOS14),DISP=SHR
//SYSUT2 DD DSN=DATOS14,UNIT=WORKDISK,DISP=(,PASS)
// SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//SYSIN DD DUMMY
/*
//PASO2 EXEC PGM=LEEGRABA
//STEPLIB DD DSN=REALES.CNEA,DISP=SHR
/*
//FT03F001 DD DSN=DATOS14,UNIT=WORKDISK,DISP=(OLD,DELETE)
//FT08F001 DD DSN=RESUL14,UNIT=WORKDISK,DISP=(,PASS),
// SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//FT06F001 SYSOUT=A
/*
//PASO3 EXEC PGM=IEBUPDTE,PARM=NEW
//SYSPRINT DD SYSOUT=A
//SYSUT2 DD DSN=SIMBOLIC.CNEA,DISP=OLD
//SYSIN DD *
./ ADD NAME=RESUL14
// DL DSN=RESUL14,DISP=(OLD,DELETE)
./ ENDUP
/*
```


//

NOTA: El archivo de entrada de datos está definido en la sentencia FT03F001 del PAS02, en donde se le asigna el número de referencia 3 y el archivo de salida de datos está definido en la sentencia FT08F001, en donde se le asigna el número de referencia 8. Por lo tanto en el programa fuente que dió origen al módulo LEEGRABA, se codificó READ3 para leer datos del primer archivo y WRITE8 para grabar resultados en el segundo archivo. Se menciona el DCB con el cual se grabará el disco auxiliar.

Modelo 280. Ejecución de un módulo de carga procedente de FORTRAN grabado en la biblioteca REALES.CNEA con el nombre LEE3GRB3, cuyos datos de entrada son tres archivos distintos que residen en la biblioteca SIMBOLIC.CNEA, como miembros independientes y cuyos resultados son tres archivos de salida distintos que se graban en esa misma biblioteca, también como miembros independientes. El problema se puede resolver en cinco pasos: en los tres primeros (PAS01, PAS02 y PAS03) el utilitario IEBGENER transfiere, de a uno, los archivos de nombre DATIN6, DATIN7 y DATIN8 de la biblioteca de simbólicos a un mismo disco auxiliar; en el PAS04 se ejecuta el programa LEE3GRB3 obteniéndose los datos de entrada del disco auxiliar y grabando los resultados, con los nombres SALIDA6, SALIDA7 y SALIDA8, en un mismo disco; en el PAS05 el utilitario IEBUPDTE transfiere estos últimos archivos del disco a la biblioteca SIMBOLIC.CNEA

```
//..... JOB ....
//PAS01 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA(DATIN6),DISP=SHR
//SYSUT2 DD DSN=DATIN6,UNIT=WORKDISK,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//SYSIN DD DUMMY
/*
//PAS02 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA(DATIN7),DISP=SHR
//SYSUT2 DD DSN=DATIN7,VOL=REF=*.PAS01.SYSUT2,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1))
//SYSIN DD DUMMY
/*
```

```

//PASO3 EXEC PGM=IEBCENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA(DATIN8),DISP=SHR
//SYSUT2 DD DSN=DATIN8,VOL=REF=*.PASO1.SYSUT2,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1))
//SYSIN DD DUMMY
/*
//PASO4 EXEC PGM=LEE3GRB3
//STEPLIB DD DSN=REALES.CNEA,DISP=SHR
/*
//FT17F001 DD DSN=DATIN6,UNIT=WORKDISK,DISP=(OLD,DELETE)
//FT18F001 DD DSN=DATIN7,UNIT=WORKDISK,DISP=(OLD,DELETE)
//FT20F001 DD DSN=DATIN8,UNIT=WORKDISK,DISP=(OLD,DELETE)
//FT10F001 DD DSN=SALIDA6,UNIT=WORKDISK,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//FT12F001 DD DSN=SALIDA7,VOL=REF=*.GO.FT10F001,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//FT14F001 DD DSN=SALIDA8,VOL=REF=*.GO.FT10F001,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//FT06F001 DD SYSOUT=A
/*
//PASO5 EXEC PGM=IEBUPDTE,PARM=MOD
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA,DISP=SHR
//SYSUT2 DD DSN=SIMBOLIC.CNEA,DISP=OLD
//SYSIN DD *
./ ADD NAME=SALIDA6
/*
// DD DSN=SALIDA6,DISP=(OLD,DELETE)
// DD *
./ ADD NAME=SALIDA7
/*
// DD DSN=SALIDA7,DISP=(OLD,DELETE)
// DD *
./ ADD NAME=SALIDA8
/*
// DD DSN=SALIDA8,DISP=(OLD,DELETE)
// DD *
./ ENDUP
/*
//

```

NOTA: Los archivos de entrada de datos están definidos en las sentencias FT17F001, FT18F001 y FT20F001 del PASO4, en donde se les asignan los números de referencia 17, 18 y 20; por lo tanto en el programa fuente que dió origen al módulo LEE3GRB3, se codificó READ17, READ18 y READ20 respectivamente, para leer datos de esos archivos. A su vez los archivos de salida de datos están definidos en las sentencias FT10F001, FT12F001 y FT14F001, en donde se les asignan los números de referencia 10, 12 y 14 ; por consiguiente en el programa fuente se codificó

WRITE10, WRITE12 y WRITE14 respectivamente, para grabar resultados en esos archivos. Se menciona el DCB con el cual se grabará el disco auxiliar.

PUNTO 12.

Corrección de módulos de carga grabados en una biblioteca de nombre REALES. CNEA, cuyos programas fuente en lenguaje simbólico, son miembros de la biblioteca SIMBOLIC.CNEA.

Modelo 281. Corrección de un módulo de carga, procedente de FORTRAN, grabado en la biblioteca con el nombre TRAYECTO. La corrección se realiza previamente en la biblioteca SIMBOLIC.CNEA, en la cual está grabado el programa fuente que dió origen al módulo. El problema se puede resolver en dos pasos: En el PASO1 el utilitario IEBUPDTE realiza la actualización que indican las tarjetas perforadas incluidas entre las sentencias CHANGE y ENDUP, que consiste en lo siguiente: a) Se suprime el registro 5319. b) Se insertan las sentencias FORTRAN 600322 y 600327. En el PASO2 el procedimiento FTG1CL carga en la biblioteca el nuevo módulo, sustituyendo al anterior con el mismo nombre.

```
//..... JOB ....
//PASO1 EXEC PGM=IEBUPDTE,PARM=MOD
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA,DISP=SHR
//SYSUT2 DD DSN=SIMBOLIC.CNEA,DISP=OLD
//SYSIN DD *
./  CHANGE  NAME=TRAYECTO
./  DELETE SEQ1=5319,SEQ2=5319
      READ(5,2007) NUM,INDICE,VOL,CUAD          00600322
      READ(5,2008) (INI(K),K=1,NUM)             00600327
./  ENDUP
/*
//PASO2 EXEC PROC=FTG1CL
//FORT.SYSIN DD DSN=SIMBOLIC.CNEA(TRAYECTO),DISP=SHR
//LKED.SYSLMOD DD DSN=REALES.CNEA(TRAYECTO),SPACE=,DISP=SHR
/*
//
```

NOTA: Como el PASO2 produce automáticamente el listado del programa fuente, se ha suprimido la codificación del parámetro LIST en la sentencia CHANGE del PASO1.

Modelo 282. Corrección de un módulo de carga, procedente de ASSEMBLER, grabado en la biblioteca con el nombre ANALISIS. La corrección se realiza previamente en la biblioteca SIMBOLIC.CNEA, en la cual está grabado el programa fuente que dió origen al módulo. El problema se puede resolver en dos pasos: En el PASO1 el uti-

utilitario IEBUPDTE realiza la actualización que indican las tarjetas perforadas incluidas entre las sentencias CHANGE y ENDUP, que consiste en lo siguiente: a) Se insertan las sentencias ASSEMBLER 4001, 4011, 4021 y 4022. b) Se suprimen los registros 5328 a 5390 inclusive. En el PASO2 el procedimiento ASMFCL carga en la biblioteca el nuevo módulo, sustituyendo al anterior con el mismo nombre.

```
//..... JOB ....
//PASO1 EXEC PGM=IEBUPDTE,PARM=MOD
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA,DISP=SHR
//SYSUT2 DD DSN=SIMBOLIC.CNEA,DISP=OLD
//SYSIN DD *
./  CHANGE  NAME=ANALISIS
          MVI   BC,X'8B'                00004001
          OPEN  (PRIN,(OUTPUT))         00004011
          LA    14,VUELTA                00004021
          BR    15                      00004022
./  DELETE  SEQ1=5328,SEQ2=5390
./  ENDUP
/*
//PASO2 EXEC PROC=ASMFCL
//ASM.SYSIN DD DSN=SIMBOLIC.CNEA(ANALISIS),DISP=SHR
//LKED.SYSLMOD DD DSN=REALES.CNEA(ANALISIS),SPACE=,DISP=SHR
/*
//
```

NOTA: Como el PASO2 produce automáticamente el listado del programa fuente, se ha suprimido la codificación del parámetro LIST en la sentencia CHANGE del PASO1.

Modelo 283. Corrección de un módulo de carga, procedente de PL/1, grabado en la biblioteca con el nombre BALISTIC. La corrección se realiza previamente en la biblioteca SIMBOLIC.CNEA, en la cual está grabado el programa fuente que dió origen al módulo. El problema se puede resolver en dos pasos. En el PASO1 el utilitario IEBUPDTE realiza la actualización que indican las tarjetas perforadas incluidas entre las sentencias CHANGE y ENDUP, que consiste en lo siguiente: a) Se insertan las sentencias PL/1 84001, 84961 y 90772; b) Se suprimen los registros 100452 a 100499 inclusive; c) Se suprime el registro 125346. En el PASO2 el procedimiento PLIXCL carga en la biblioteca el nuevo módulo, sustituyendo al anterior con el mismo nombre.

```
//..... JOB ....
```

```

//PAS01 EXEC PGM=IEBUPDTE,PARM=MOD
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA,DISP=SHR
//SYSUT2 DD DSN=SIMBOLIC.CNEA,DISP=OLD
//SYSIN DD *
./  CHANGE  NAME=BALISTIC
        DECLARE (SUMA,VALOR,I) FIXED(5);
        PUT LIST (VAL,XIN(N),INDICE);
        DO I=GAMMA TO GAMAX BY INDICE WHILE VER>GAR;
./  DELETE  SEQ1=100452,SEQ2=100499
./  DELETE  SEQ1=125346,SEQ2=125346
./  ENDUP
/*
//PAS02 EXEC PROC=PLIXCL,PARM.PLI='CS(60)'
//PLI.SYSIN DD DSN=SIMBOLIC.CNEA(BALISTIC),DISP=SHR
//LKED.SYSLMOD DD DSN=REALES.CNEA(BALISTIC),SPACE=,DISP=OLD
/*
//

```

NOTA: Como el PAS02 produce automáticamente el listado del programa fuente, se ha suprimido la codificación del parámetro LIST en la sentencia CHANGE del PAS01.

PUNTO 13.

Transferencia de programas de una biblioteca de simbólicos de nombre SIMBOLIC.CNEA a una biblioteca de módulos de carga de nombre REALES.CNEA.

Modelo 284 . Transferencia de un programa en lenguaje FORTRAN, grabado en la biblioteca de simbólicos con el nombre TERIC, a una biblioteca de módulos de carga.

```
//..... JOB ....
//PAS01 EXEC PROC=FTG1CL
//FORT.SYSIN DD DSN=SIMBOLIC.CNEA(TERIC),DISP=SHR
//LKED.SYSLMOD DD DSN=REALES.CNEA(TERIC),SPACE=,DISP=OLD
/*
//
```

Modelo 285 . Transferencia de un programa en lenguaje ASSEMBLER, grabado en la biblioteca de simbólicos con el nombre ARTAL, a una biblioteca de módulos de carga.

```
//..... JOB ....
//PAS01 EXEC PROC=ASMFCL
//ASM.SYSIN DD DSN=SIMBOLIC.CNEA(ARTAL),DISP=SHR
//LKED.SYSLMOD DD DSN=REALES.CNEA(ARTAL),SPACE=,DISP=OLD
/*
//
```

Modelo 286 . Transferencia de un programa en lenguaje PL/1, grabado en la biblioteca de simbólicos con el nombre ASTER, a una biblioteca de módulos de carga.

```
//..... JOB ....
//PAS01 EXEC PROC=PLIXCL,PARM.PLI='CS(60)'
//PLI.SYSIN DD DSN=SIMBOLIC.CNEA(ASTER),DISP=SHR
//LKED.SYSLMOD DD DSN=REALES.CNEA(ASTER),SPACE=,DISP=OLD
/*
//
```

PUNTO 1.-

Transferencia simultánea de programas que se presentan en tarjetas perforadas o en cintas magnéticas, a dos bibliotecas particionadas: una de programas en lenguaje simbólico de nombre SIMBOLIC.CNEA y otra de módulos de carga de nombre REALES.CNEA.

Modelo 287 . Transferencia de un programa en lenguaje simbólico FORTRAN, de nombre METALES, que se presenta en tarjetas perforadas, primero a la biblioteca de simbólicos y de ahí a la biblioteca de módulos de carga. El problema se puede resolver en dos pasos: en el PASO1 el utilitario IEBUPDTE carga el programa METALES, numerando sus registros de 100 en 100 a partir del número 10, en la biblioteca SIMBOLIC.CNEA; en el PASO2 el procedimiento FTG1CL compila y vincula el programa en FORTRAN, cargando el módulo resultante en la biblioteca REALES.CNEA.

```
//..... JOB....
//PASO1 EXEC PGM=IEBUPDTE,PARM=NEW
//SYSPRINT DD SYSOUT=A
//SYSUT2 DD DSN=SIMBOLIC.CNEA,DISP=OLD
//SYSIN DD *
./  ADD  NAME=METALES
./  NUMBER NEW1=10,INCR=100
  Aquí tarjetas perforadas en FORTRAN
./  ENDUP
/*
//PASO2 EXEC PROC=FTG1CL
//FORT.SYSIN DD DSN=SIMBOLIC.CNEA(METALES),DISP=SHR
//LKED.SYSMOD DD DSN=REALES.CNEA(METALES),SPACE=,DISP=OLD
/*
//
```

Modelo 288 . Transferencia de un programa en lenguaje simbólico ASSEMBLER, de nombre COLIMAD, que se presenta en tarjetas perforadas, primero a la biblioteca de simbólicos y de ahí a la biblioteca de módulos de carga. El problema se puede resolver en dos pasos: en el PASO1 el utilitario IEBUPDTE carga el programa COLIMAD, numerando sus registros de 100 en 100 a partir de 10, en la biblioteca SIMBOLIC.CNEA; en el PASO2 el procedimiento ASMFCL compila y vincula el programa en ASSEMBLER, cargando el módulo resultante en la biblioteca REALES.CNEA.

```
//..... JOB ....
```



```
//PASO1 EXEC PGM=IEBUPDTE,PARM=NEW
//SYSPRINT DD SYSOUT=A
//SYSUT2 DD DSN=SIMBOLIC.CNEA,DISP=OLD
//SYSIN DD *
./  ADD  NAME=COLIMAD
./  NUMBER NEWI=10,INCR=100
Aquí tarjetas perforadas en ASSEMBLER
./  ENDUP
/*
//PASO2 EXEC PROC=ASMFCL
//ASM.SYSIN DD DSN=SIMBOLIC.CNEA(COLIMAD),DISP=SHR
//LKED.SYSLMOD DD DSN=REALES.CNEA(COLIMAD),SPACE=,DISP=OLD
/*
//
```

Modelo 289 . Transferencia de un programa en lenguaje simbólico PL/1, de nombre DEFLECT, que se presenta en tarjetas perforadas, primero a la biblioteca de simbólicos y de ahí a la biblioteca de módulos de carga. El problema se puede resolver en dos pasos: en el PASO1 el utilitario IEBUPDTE carga el programa DEFLECT en la biblioteca SIMBOLIC.CNEA; en el PASO2 el procedimiento PLIXCL compila y vincula el programa en PL/1, cargando el módulo resultante en la biblioteca REALES.CNEA.

```
//..... JOB ....
//PASO1 EXEC PGM=IEBUPDTE,PARM=NEW
//SYSPRINT DD SYSOUT=A
//SYSUT2 DD DSN=SIMBOLIC.CNEA,DISP=OLD
//SYSIN DD *
./  ADD  NAME=DEFLECT
Aquí tarjetas perforadas en PL/1
./  ENDUP
/*
//PASO2 EXEC PROC=PLIXCL,PARM.PLI='CS(60)'
//PLI.SYSIN DD DSN=SIMBOLIC.CNEA(DEFLECT),DISP=SHR
//LKED.SYSLMOD DD DSN=REALES.CNEA(DEFLECT),SPACE=,DISP=OLD
/*
//
```

Modelo 290 . Transferencia de un programa en lenguaje simbólico FORTRAN, grabado con el nombre AFAN en el rótulo 26 de una cinta magnética, primero a la biblioteca de simbólicos y de ahí a la biblioteca de módulos de carga. La cinta lleva el número de carrete 11278 y tiene etiquetas modelo IBM. El problema se puede resolver en dos pasos: en el PASO1 el utilitario IEBUPDTE transfiere el programa AFAN de la cinta a la biblioteca SIMBOLIC.CNEA; en el PASO2 el procedimiento

FTG1CL compila y vincula el programa en FORTRAN, cargando el módulo resultante en la biblioteca REALES.CNEA.

```
//..... JOB ....
//PASO1 EXEC PGM=IEBUPDTE,PARM=MOD
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=AFAN,UNIT=TAPE,VOL=SER=T11278,LABEL=(26,SL),
// DISP=(OLD,KEEP)
//SYSUT2 DD DSN=SIMBOLIC.CNEA,DISP=OLD
//SYSIN DD *
./ REPRO NEW=PO, MEMBER=AFAN
./ ENDUP
/*
//PASO2 EXEC PROC=FTG1CL
//FORT.SYSIN DD DSN=SIMBOLIC.CNEA(AFAN),DISP=SHR
//LKED.SYSLMOD DD DSN=REALES.CNEA(AFAN),SPACE=,DISP=OLD
/*
//
```

Modelo 291 . Transferencia de un programa en lenguaje simbólico ASSEMBLER, grabado con el nombre FURIN en el rótulo 12 de una cinta magnética, primero en la biblioteca de simbólicos y de ahí a la biblioteca de módulos de carga. La cinta lleva el número de carrete 61350 y tiene etiquetas modelo IBM. El problema se puede resolver en dos pasos: en el PASO1 el utilitario IEBUPDTE transfiere el programa FURIN de la cinta a la biblioteca SIMBOLIC.CNEA; en el PASO2 el procedimiento ASMFCL compila y vincula el programa en ASSEMBLER, cargando el módulo resultante en la biblioteca REALES.CNEA.

```
//..... JOB ....
//PASO1 EXEC PGM=IEBUPDTE,PARM=MOD
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=FURIN,UNIT=TAPE,VOL=SER=T61350,LABEL=(12,SL),
// DISP=(OLD,KEEP)
//SYSUT2 DD DSN=SIMBOLIC.CNEA,DISP=OLD
//SYSIN DD *
./ REPRO NEW=PO, MEMBER=FURIN
./ ENDUP
/*
//PASO2 EXEC PROC=ASMFCL
//ASM.SYSIN DD DSN=SIMBOLIC.CNEA(FURIN),DISP=SHR
//LKED.SYSLMOD DD DSN=REALES.CNEA(FURIN),SPACE=,DISP=OLD
/*
//
```

Modelo 292 . Transferencia de un programa en lenguaje simbólico PL/1, grabado con el nombre SPOT, en el rótulo 31 de una cinta magnética, primero a la biblio-

teca de simbólicos y de ahí a la biblioteca de módulos de carga. La cinta lleva el número de carrete 00200 y tiene etiquetas modelo IBM. El problema se puede resolver en dos pasos: en el PASO1 el utilitario IEBUPDTE transfiere el programa SPOT de la cinta a la biblioteca SIMBOLIC.CNEA; en el PASO2 el procedimiento PLIXCL compila y vincula el programa en PL/1, cargando el módulo resultante en la biblioteca REALES.CNEA.

```
//..... JOB ....
//PASO1 EXEC PGM=IEBUPDTE,PARM=MOD
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SPOT,UNIT=TAPE,VOL=SER=T00200,LABEL=(31,SL),
//  DISP=(CLD,KEEP)
//SYSUT2 DD DSN=SIMBOLIC.CNEA,DISP=OLD
//SYSIN DD *
./  REPRO NEW=PO,MEMBER=SPOT
./  ENDUP
/*
//PASO2 EXEC PROC=PLIXCL,PARM.PLI='CS(60)'
//PLI.SYSIN DD DSN=SIMBOLIC.CNEA(SPOT),DISP=SHR
//LKED.SYSLMOD DD DSN=REALES.CNEA(SPOT),SPACE=,DISP=OLD
/*
//
```

CAPITULO VI

MODELOS PARA LA CREACION Y MANTENIMIENTO DE BIBLIOTECAS PARTICIONADAS Y CATALOGADAS

PUNTO 1.

Creación de nuevas bibliotecas particionadas y catalogadas de programas en lenguaje simbólico.

Modelo 293 . Creación de una biblioteca de nombre SIMBOLIC.FISICA y su carga con programas que se presentan en tarjetas perforadas en lenguaje simbólico. La biblioteca residirá en una unidad de discos con el número de serie del volumen CNEA02, con una capacidad de 10 cilindros para sus miembros y 15 bloques para el directorio. El problema se puede resolver en dos pasos: Previamente se define un procedimiento de nombre BIBDESIM cuyos parámetros simbólicos serán sustituidos en cada oportunidad por valores concretos; en el PASO1 se invoca dicho procedimiento y se lo ejecuta con los valores específicos que tomarán los parámetros simbólicos en esa corrida y que se mencionan ahí mismo; en el PASO2 el utilitario IEBUPDTE carga los programas en tarjetas en la biblioteca recién creada, con los nombres METODOS, RANGO, FERMI, PROTON, ESTADOF, ALFA, DENSIDAD y FISION. En cada miembro la máquina numerará los registros de 100 en 100 a partir del número 50.

```
//..... JOB ....
//BIBDESIM PROC
//IEFPROC EXEC PGM=IEFBRI4
//DD1 DD DSN=SIMBOLIC.&D,UNIT=SYSDA,VOL=SER=&V,
// DISP=(NEW,CATLG),LABEL=EXPDT=99360,SPACE=(CYL,(&C)),
// DCB=(BLKSIZE=3200,LRECL=80,RECFM=FB)
// PEND
//PASO1 EXEC PROC=BIBDESIM,D='FISICA',V='CNEA02',C='10,,15'
/*
//PASO2 EXEC PGM=IEBUPDTE,PARM=NEW
//SYSPRINT DD SYSOUT=A
//SYSUT2 DD DSN=SIMBOLIC.FISICA,DISP=OLD
//SYSIN DD *
./ ADD NAME=METODOS
./ NUMBER NEW1=50,INCR=100
Aquí tarjetas perforadas en lenguaje simbólico
./ ADD NAME=RANGO
./ NUMBER NEW1=50,INCR=100
Aquí tarjetas perforadas en lenguaje simbólico
./ ADD NAME=FERMI
./ NUMBER NEW1=50,INCR=100
Aquí tarjetas perforadas en lenguaje simbólico
./ ADD NAME=PROTON
./ NUMBER NEW1=50,INCR=100
Aquí tarjetas perforadas en lenguaje simbólico
```

```

./ ADD NAME=ESTADOF
./ NUMBER NEW1=50,INCR=100
Aquí tarjetas perforadas en lenguaje simbólico
./ ADD NAME=ALFA
./ NUMBER NEW1=50,INCR=100
Aquí tarjetas perforadas en lenguaje simbólico
./ ADD NAME=DENSIDAD
./ NUMBER NEW1=50,INCR=100
Aquí tarjetas perforadas en lenguaje simbólico
./ ADD NAME=FISION
./ NUMBER NEW1=50,INCR=100
Aquí tarjetas perforadas en lenguaje simbólico
./ ENDUP
/*
//

```

NOTA: En el procedimiento BIBDESIM se codifica el parámetro LABEL con la fecha de expiración de la biblioteca y el parámetro DCB que se usará para grabar los miembros.

Modelo 294 . Creación de una biblioteca de nombre SIMBOLIC.QUIMICA y su carga con programas grabados en rótulos consecutivos o no consecutivos de una cinta magnética. La biblioteca residirá en una unidad de discos con el número de serie del volumen CNEA03, con una capacidad de 15 cilindros para sus miembros y 20 bloques para el directorio. Los programas, de nombres FONON, DIFUSION y DECAI están grabados en los rótulos 3, 8 y 9 de la cinta que lleva el número de carrete 51537 y tiene etiquetas modelo IBM. El problema se puede resolver en cuatro pasos: Previamente se define un procedimiento de nombre BIBDESIM cuyos parámetros simbólicos serán sustituidos en cada oportunidad por valores concretos; en el PASO1 se invoca dicho procedimiento y se lo ejecuta con los valores específicos que tomarán los parámetros simbólicos en esa corrida y que se mencionan ahí mismo; en el PASO2 y siguientes el utilitario IEBUPDTE transfiere, rótulo por rótulo, los programas de la cinta a la biblioteca recién creada.

```

//..... JOB ....
//BIBDESIM PROC
//IEFPROC EXEC PGM=IEFBR14
//DD1 DD DSN=SIMBOLIC.&D,UNIT=SYSDA,VOL=SER=&V,
// DISP=(NEW,CATLG),LABEL=EXPDT=99360,SPACE=(CYL,(&C)),
// DCB=(BLKSIZE=3200,LRECL=80,RECFM=FB)
// PEND

```

```

//PASO1 EXEC PROC=BIBDESIM,D='QUIMICA',V='CNEA03',C='15,,20'
/*
//PASO2 EXEC PGM=IEBUPDTE,PARM=MOD
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=FONON,UNIT=TAPE,VOL=SER=T51537,LABEL=(3,SL),
//  DISP=(OLD,PASS)
//SYSUT2 DD DSN=SIMBOLIC.QUIMICA,DISP=OLD
//SYSIN DD *
./  REPRO NEW=PO,MEMBER=FONON,LIST=ALL
./  ENDUP
/*
//PASO3 EXEC PGM=IEBUPDTE,PARM=MOD
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=DIFUSION,UNIT=TAPE,VOL=REF=*.PASO2.SYSUT1,
//  LABEL=(8,SL)
//SYSUT2 DD DSN=SIMBOLIC.QUIMICA,DISP=OLD
//SYSIN DD *
./  REPRO NEW=PO,MEMBER=DIFUSION,LIST=ALL
./  ENDUP
/*
//PASO4 EXEC PGM=IEBUPDTE,PARM=MOD
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=DECAI,UNIT=TAPE,VOL=REF=*.PASO2.SYSUT1,
//  LABEL=(9,SL),DISP=(OLD,KEEP)
//SYSUT2 DD DSN=SIMBOLIC.QUIMICA,DISP=OLD
//SYSIN DD *
./  REPRO NEW=PO,MEMBER=DECAI,LIST=ALL
./  ENDUP
/*
//

```

NOTA: En el procedimiento BIBDESIM se codifica el parámetro LABEL con la fecha de expiración de la biblioteca y el parámetro DCB que se usará para grabar los miembros.

Modelo 295 . Creación de una biblioteca de nombre SIMBOLIC.RADIO y su carga con programas procedentes de otra biblioteca particionada que fué grabada en su totalidad, incluido el directorio, en una cinta magnética con el programa IEBCOPY. La nueva biblioteca residirá en una unidad de discos con el número de serie del volumen CNEA01, con una capacidad de 16 cilindros para sus miembros y 40 bloques para el directorio. La cinta, que lleva el número de corete 32107 y tiene etiquetas modelo IBM, contiene la biblioteca anterior grabada en el rótulo 2, con el nombre SIMBOLIC.ANT. El problema se puede resolver en dos pasos: Previamente se define un procedimiento de nombre BIBDESIM cuyos parámetros simbólicos serán sustituidos en cada oportunidad por valores concretos; en el PASO1 se invoca dicho procedi-

miento y se lo ejecuta con los valores específicos que tomarán los parámetros simbólicos en esa corrida y que se mencionan ahí mismo; en el PASO2 el utilitario IEBCOPY transfiere la biblioteca de la cinta a la unidad de acceso directo que le corresponde, seleccionando solamente los miembros de nombres PAR, IMPAR, NOTA, PERT, ION, NEUTRON, POLAR y COULOMB. El miembro NOTA y el miembro POLAR cambiarán sus nombres, al ingresar a la nueva biblioteca, por los de NOTA33 y RUTINPOL.

```
//..... JOB ....
//BIBDESIM PROC
//IEFPROC EXEC PGM=IEFBR14
//DD1 DD DSN=SIMBOLIC.&D,UNIT=SYSDA,VOL=SER=&V,
// DISP=(NEW,CATLG),LABEL=EXPDT=99360,SPACE=(CYL,(&C)),
// DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
// PEND
//PASO1 EXEC PROC=BIBDESIM,D='RADIO',V='CNEA01',C='16',40'
/*
//PASO2 EXEC PGM=IEBCOPY
//SYSPRINT DD SYSOUT=A
//SYSUT3 DD UNIT=SYSDA,SPACE=(CYL,(5,2))
//SYSUT4 DD UNIT=SYSDA,SPACE=(CYL,(5,2))
//ENTRADA DD DSN=SIMBOLIC.ANT,UNIT=TAPE,VOL=SER=T32107,
// LABEL=(2,SL),DISP=(OLD,KEEP)
//SALIDA DD DSN=SIMBOLIC.RADIO,DISP=OLD
//SYSIN DD *
  C O=SALIDA,I=ENTRADA
  S M=((PAR,,R),(IMPAR,,R),(NOTA,NOTA33,R),(PERT,,R),(ION,,R),
      (NEUTRON,,R),(POLAR,RUTINPOL,R),(COULOMB,,R))
/*
//
```

NOTA: En el procedimiento BIBDESIM se codifica el parámetro LABEL con la fecha de expiración de la biblioteca y el parámetro DCB que se usará para grabar los miembros. En la sentencia de selección de miembros del PASO2 se pueden codificar hasta ocho nombres. Se indica la existencia de una tarjeta continuación con un carácter cualquiera, distinto de blanco, en la columna 72; la tarjeta que sigue debe empezar en la columna 16.

Modelo 296 . Creación de una biblioteca de nombre SIMBOLIC.MATEMAT y su carga con programas procedentes de otra biblioteca catalogada similar, ya existente en la misma instalación con el nombre PROGRAM.MAT. La nueva biblioteca residirá en una unidad de discos con el número de serie del volumen CNEA04 y una capacidad de

5 cilindros para sus miembros y 8 bloques para el directorio. El problema se puede resolver en dos pasos: Previamente se define un procedimiento de nombre BIBDESIM cuyos parámetros simbólicos serán sustituidos en cada oportunidad por valores concretos; en el PASO1 se invoca dicho procedimiento y se lo ejecuta con los valores específicos que tomarán los parámetros simbólicos en esa corrida y que se mencionan ahí mismo; en el PASO2 el utilitario IEBCOPY transfiere todos los miembros de la biblioteca PROGRAM.MAT a la biblioteca recién creada, con exclusión de los programas ACTINIO, MECANICA, COMBAT y HEAVYION, que están fuera de uso.

```
//..... JOB ....
//BIBDESIM PROC
//IEFPROC EXEC PGM=IEFBR14
//DD1 DD DSN=SIMBOLIC.&D,UNIT=SYSDA,VOL=SER=&V,
// DISP=(NEW,CATLG),LABEL=EXPDT=99360,SPACE=(CYL,(2C)),
// DCB=(BLKSIZE=2400,LRECL=80,RECFM=FB)
// PEND
//PASO1 EXEC PROC=BIBDESIM,D='MATEMAT',V='CNEA04',C='5,,8'
/*
//PASO2 EXEC PGM=IEBCOPY
//SYSPRINT DD SYSOUT=A
//SYSUT3 DD UNIT=SYSDA,SPACE=(CYL,(2,1))
//SYSUT4 DD UNIT=SYSDA,SPACE=(CYL,(2,1))
//ENTRADA DD DSN=PROGRAM.MAT,DISP=SHR
//SALIDA DD DSN=SIMBOLIC.MATEMAT,DISP=OLD
//SYSIN DD *
  C O=SALIDA,I=ENTRADA
  EXCLUDE MEMBER=ACTINIO,MECANICA,COMBAT,HEAVYION
/*
//
```

NOTA: En el procedimiento BIBDESIM se codifica el parámetro LABEL con la fecha de expiración de la biblioteca y el parámetro DCB que se usará para grabar los miembros.

PUNTO 2.

Creación de nuevas bibliotecas particionadas y catalogadas de módulos de carga.

Modelo 297 . Creación de una biblioteca de nombre REALES.RADIO y su carga con módulos procedentes de otra biblioteca particionada que fué grabada en su totalidad, incluido el directorio, en una cinta magnética con el programa IEBCOPY. La nueva biblioteca residirá en una unidad de discos con el número de serie del volumen CNEA03 y una capacidad de 16 cilindros para sus módulos y 40 bloques para el directorio. La cinta, que lleva el número de carrete 31899 y tiene etiquetas modelo IBM, contiene la biblioteca anterior grabada en el rótulo 1, con el nombre REALES.ANT. El problema se puede resolver en dos pasos: Previamente se define un procedimiento de nombre BIBREAL cuyos parámetros simbólicos serán sustituidos en cada oportunidad por valores concretos; en el PASO1 se invoca dicho procedimiento y se lo ejecuta con los valores específicos que tomarán los parámetros simbólicos en esa corrida y que se mencionan ahí mismo; en el PASO2 el utilitario IEBCOPY transfiere la biblioteca de la cinta a la unidad de acceso directo que le corresponde, seleccionando solamente los módulos de nombres CONFIAR, MATAC, ANGULAR, CANAL, MATTER, PAR,ORDEN y FOTOELEC. El módulo MATAC y el módulo CANAL cambiarán sus nombres al ingresar a la nueva biblioteca, por los de MATPAS y CANAL4.

```
//..... JOB ....
//BIBREAL PROC
//IEFPROC EXEC PGM=IEFBR14
//DD1 DD DSN=REALES.&D,UNIT=SYSDA,VOL=SER=&V,
// DISP=(NEW,CATLG),LABEL=EXPDT=99360,SPACE=(CYL,(&C)),
// DCB=(BLKSIZE=13030,LRECL=1024,RECFM=U)
// PEND
//PASO1 EXEC PROC=BIBREAL,D='RADIO',V='CNEA03',C='16,,40'
/*
//PASO2 EXEC PGM=IEBCOPY
//SYSPRINT DD SYSOUT=A
//SYSUT3 DD UNIT=SYSDA,SPACE=(CYL,(5,2))
//SYSUT4 DD UNIT=SYSDA,SPACE=(CYL,(5,2))
//ENTRADA DD DSN=REALES.ANT,UNIT=TAPE,VOL=SER=T31899,
// LABEL=(1,SL),DISP=(OLD,KEEP)
//SALIDA DD DSN=REALES.RADIO,DISP=OLD
//SYSIN DD *
C O=SALIDA,I=ENTRADA
S M=((CONFIAR,,R),(MATAC,MATPAS,R),(ANGULAR,,R),(CANAL,CANAL4,R),
      (MATTER,,R),(PAR,,R),(ORDEN,,R),(FOTOELEC,,R))
```

/*
//

NOTA: En el procedimiento BIBREAL se codifica el parámetro LABEL con la fecha de expiración de la biblioteca y el parámetro DCB que se usará para grabar los módulos. En la sentencia de selección de miembros del PASO2 se pueden codificar hasta ocho nombres. Se indica la existencia de una tarjeta continuación con un carácter cualquiera, distinto de blanco, en la columna 72; la tarjeta que sigue debe empezar en la columna 16.

Modelo 298 . Creación de una biblioteca de nombre REALES.MATEMAT y su carga con módulos procedentes de otra biblioteca catalogada similar, ya existente en la misma instalación con el nombre MODULOS.MAT. La nueva biblioteca residirá en una unidad de discos con el número de serie del volumen CNEA02 y una capacidad de 5 cilindros para sus módulos y 8 bloques para el directorio. El problema se puede resolver en dos pasos: Previamente se define un procedimiento de nombre BIBREAL cuyos parámetros simbólicos serán sustituidos en cada oportunidad por valores concretos; en el PASO1 se invoca dicho procedimiento y se lo ejecuta con los valores específicos que tomarán los parámetros simbólicos en esa corrida y que se mencionan ahí mismo; en el PASO2 el utilitario IEBCOPY transfiere todos los módulos de la biblioteca MODULOS.MAT a la biblioteca recién creada, con exclusión de los módulos ISOTOPO, COLOR, DEUTERON y BANDA, que están fuera de uso.

```
//..... JOB ....
//BIBREAL PROC
//IEFPROC EXEC PGM=IEFBR14
//DD1 DD DSN=REALES.&D,UNIT=SYSDA,VOL=SER=&V,
// DISP=(NEW,CATLG),LABEL=EXPDT=99360,SPACE=(CYL,(&C)),
// DCB=(BLKSIZE=13030,LRECL=1024,RECFM=U)
// PEND
//PASO1 EXEC PROC=BIBREAL,D='MATEMAT',V='CNEA02',C='5,,8'
/*
//PASO2 EXEC PGM=IEBCOPY
//SYSPRINT DD SYSOUT=A
//SYSUT3 DD UNIT=SYSDA,SPACE=(CYL,(2,1))
//SYSUT4 DD UNIT=SYSDA,SPACE=(CYL,(2,1))
//ENTRADA DD DSN=MODULOS.MAT,DISP=SHR
//SALIDA DD DSN=REALES.MATEMAT,DISP=OLD
//SYSIN DD *
```

```
C O=SALIDA,I=ENTRADA
EXCLUDE MEMBER=ISOTOPO,COLOR,DEUTERON,BANDA
/*
//
```

NOTA: En el procedimiento BIBREAL se codifica el parámetro LABEL con la fecha de expiración de la biblioteca y el parámetro DCB que se usará para grabar los módulos.

PUNTO 3.

Reposición en su totalidad o en parte de una biblioteca de programas en lenguaje simbólico, de nombre LIB3262.SIMBFIS, deteriorada accidentalmente, a partir de una cinta magnética de resguardo que fué grabada con el utilitario IEBCOPY. La biblioteca reside en la unidad de discos 3350, con el número de serie del volumen CNEA02.

Modelo 299 . Reposición total de la biblioteca copiándola de la cinta magnética de resguardo. La biblioteca completa, incluido su directorio, está grabada con el nombre SALVASII, en el rótulo 1 de una cinta que lleva el número de carrete 80001 y tiene etiquetas modelo IBM. El problema se puede resolver en tres pasos: en el PASO1 el utilitario IEHPROGM borra la biblioteca deteriorada, inclusive su directorio; en el PASO2 el utilitario IEBCOPY transfiere toda la biblioteca de la cinta a un disco auxiliar, con el nombre BIBLIOT; en el PASO3 el mismo utilitario transfiere la biblioteca del disco auxiliar a la unidad de acceso directo que le corresponde.

```
//..... JOB ....
//PASO1 EXEC PGM=IEHPROGM
//SYSPRINT DD SYSOUT=A
//DD1 DD UNIT=3350,VOL=SER=CNEA02,DISP=OLD
//SYSIN DD *
  SCRATCH DSN=LIB3262.SIMBFIS,VOL=3350=CNEA02,PURGE
/*
//PASO2 EXEC PGM=IEBCOPY
//SYSPRINT DD SYSOUT=A
//SYSUT3 DD UNIT=WORKDISK,SPACE=(CYL,(5,2))
//SYSUT4 DD UNIT=WORKDISK,SPACE=(CYL,(5,2))
//ENTRADA DD DSN=SALVASII,UNIT=TAPE,VOL=SER=T80001,DISP=OLD
//SALIDA DD DSN=BIBLIOT,UNIT=WORKDISK,DISP=(NEW,PASS),
//  SPACE=(CYL,(15,,20)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//SYSIN DD *
  C 0=SALIDA,I=ENTRADA
/*
//PASO3 EXEC PGM=IEBCOPY
//SYSPRINT DD SYSOUT=A
//SYSUT3 DD UNIT=WORKDISK,SPACE=(CYL,(5,2))
//SYSUT4 DD UNIT=WORKDISK,SPACE=(CYL,(5,2))
//ENTRADA DD DSN=BIBLIOT,UNIT=WORKDISK,DISP=(OLD,DELETE)
//SALIDA DD DSN=LIB3262.SIMBFIS,DISP=OLD
```

```
//SYSIN DD *
C O=SALIDA,I=ENTRADA
/*
//
```

NOTA: Se menciona el DCB con el cual se grabará el disco auxiliar. Cuando la biblioteca deteriorada se borra en el PAS01, no por eso resulta decatalogada, es decir, el nombre LIB3262.SIMBFIS continua figurando en el catálogo del sistema. Por lo tanto no es necesario crearla otra vez, sino simplemente llenarla: si se pretendiera crear la biblioteca mediante el procedimiento adecuado, el sistema acusará el error "DIRECT ACCESS-DUPLICATE NAME ON VOLUME", (en el volumen CNEA02 el nombre LIB3262.SIMBFIS está duplicado).

Modelo 300 . Reposición de un miembro deteriorado de la biblioteca obteniéndolo de la cinta de resguardo. La biblioteca completa, incluido su directorio, está grabada con el nombre SALVASI2, en el rótulo 1 de una cinta que tiene etiquetas modelo IBM y lleva el número de carrete 80001. El miembro que se va a reponer se llama HAFOBOS. El problema se puede resolver en dos pasos: en el PAS01 el utilitario IEBCOPY transfiere toda la biblioteca de la cinta a un disco auxiliar, con el nombre BIBLIOT; en el PAS02 el mismo utilitario selecciona del disco auxiliar solamente el miembro HAFOBOS y lo graba en la biblioteca LIB3262.SIMBFIS.

```
//..... JOB ....
//PAS01 EXEC PGM=IEBCOPY
//SYSPRINT DD SYSOUT=A
//SYSUT3 DD UNIT=WORKDISK,SPACE=(CYL,(5,2))
//SYSUT4 DD UNIT=WORKDISK,SPACE=(CYL,(5,2))
//ENTRADA DD DSN=SALVASI2,UNIT=TAPE,VOL=SER=T80001,DISP=OLD
//SALIDA DD DSN=BIBLIOT,UNIT=WORKDISK,DISP=(,PASS),
// SPACE=(CYL,(15,,20)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//SYSIN DD *
C O=SALIDA,I=ENTRADA
/*
//PAS02 EXEC PGM=IEBCOPY
//SYSPRINT DD SYSOUT=A
//SYSUT3 DD UNIT=WORKDISK,SPACE=(CYL,(5,2))
//SYSUT4 DD UNIT=WORKDISK,SPACE=(CYL,(5,2))
//ENTRADA DD DSN=BIBLIOT,UNIT=WORKDISK,DISP=(OLD,DELETE)
//SALIDA DD DSN=LIB3262.SIMBFIS,DISP=OLD
//SYSIN DD *
C O=SALIDA,I=ENTRADA
```

```

S M=((HAFOBOS,,R))
/*
//

```

NOTA: El DCB codificado en la sentencia SALIDA del PASO1, para grabar el disco auxiliar, es propio de la biblioteca LIB3262.SIMBFIS.

Modelo 301 . Reposición de varios miembros deteriorados de la biblioteca, obteniéndolos de la cinta magnética de resguardo. La biblioteca completa, incluido su directorio, está grabada con el nombre SALVASI2, en el rótulo 1 de una cinta que tiene etiquetas modelo IBM y lleva el número de carrete 80001. Los miembros que se van a reponer se llaman ZLEVEL, COORDEN y ATOM35; además, al miembro COORDEN se le cambiará el nombre por el de ORDYAB. El problema se puede resolver en dos pasos: en el PASO1 el utilitario IEBCOPY transfiere toda la biblioteca de la cinta a un disco auxiliar, con el nombre REPOSIC; en el PASO2 el mismo utilitario selecciona del disco auxiliar solamente los miembros solicitados y los graba en la biblioteca LIB3262.SIMBFIS. Al miembro COORDEN le cambia el nombre antes de grabarlo.

```

//..... JOB ....
//PASO1 EXEC PGM=IEBCOPY
//SYSPRINT DD SYSOUT=A
//SYSUT3 DD UNIT=WORKDISK,SPACE=(CYL,(5,2))
//SYSUT4 DD UNIT=WORKDISK,SPACE=(CYL,(5,2))
//ENTRADA DD DSN=SALVASI2,UNIT=TAPE,VOL=SER=T80001,DISP=OLD
//SALIDA DD DSN=REPOSIC,UNIT=WORKDISK,DISP=(,PASS),
// SPACE=(CYL,(15,,20)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//SYSIN DD *
C O=SALIDA,I=ENTRADA
/*
//PASO2 EXEC PGM=IEBCOPY
//SYSPRINT DD SYSOUT=A
//SYSUT3 DD UNIT=WORKDISK,SPACE=(CYL,(5,2))
//SYSUT4 DD UNIT=WORKDISK,SPACE=(CYL,(5,2))
//ENTRADA DD DSN=REPOSIC,UNIT=WORKDISK,DISP=(OLD,DELETE)
//SALIDA DD DSN=LIB3262.SIMBFIS,DISP=OLD
//SYSIN DD *
C O=SALIDA,I=ENTRADA
S M=((ZLEVEL,,R),(COORDEN,ORDYAB,R),(ATOM35,,R))
/*
//

```

NOTA: La sentencia S del PASO2 admite la mención de hasta ocho miembros. El DCB

codificado en la sentencia SALIDA del PAS01, para grabar el disco auxiliar, es propio de la biblioteca LIB0262.SIMBFIS.

PUNTO 4.

Reposición en su totalidad o en parte de una biblioteca de módulos de carga, de nombre L06201.REALFIS, deteriorada accidentalmente, a partir de una cinta magnética de resguardo que fué grabada con el utilitario IEBCOPY. La biblioteca reside en la unidad de discos 3350, con el número de serie del volumen CNEA02.

Modelo 302 . Reposición total de la biblioteca copiándola de la cinta magnética de resguardo. La biblioteca completa, incluido su directorio, está grabada con el nombre SALVARE2, en el rótulo 1 de una cinta que lleva el número de carrete 13321 y tiene etiquetas modelo IBM. El problema se puede resolver en tres pasos: en el PAS01 el utilitario IEHPRGM borra la biblioteca deteriorada, inclusive su directorio; en el PAS02 el utilitario IEBCOPY transfiere toda la biblioteca de la cinta a un disco auxiliar, con el nombre REALBIBL; en el PAS03 el mismo utilitario transfiere la biblioteca del disco auxiliar a la unidad de acceso directo que le corresponde.

```
//..... JOB ....
//PAS01 EXEC PGM=IEHPRGM
//SYSPRINT DD SYSOUT=A
//DD1 DD UNIT=3350,VOL=SER=CNEA02,DISP=OLD
//SYSIN DD *
  SCRATCH DSNAME=L06201.REALFIS,VOL=3350=CNEA02,PURGE
/*
//PAS02 EXEC PGM=IEBCOPY
//SYSPRINT DD SYSOUT=A
//SYSUT3 DD UNIT=WORKDISK,SPACE=(CYL,(5,2))
//SYSUT4 DD UNIT=WORKDISK,SPACE=(CYL,(5,2))
//ENTRADA DD DSN=SALVARE2,UNIT=TAPE,VOL=SER=T13321,DISP=OLD
//SALIDA DD DSN=REALBIBL,UNIT=WORKDISK,DISP=(NEW,PASS),
//  SPACE=(CYL,(15,,20)),DCB=(BLKSIZE=13030,LRECL=1024,RECFM=U)
//SYSIN DD *
  C O=SALIDA,I=ENTRADA
/*
//PAS03 EXEC PGM=IEBCOPY
//SYSPRINT DD SYSOUT=A
//SYSUT3 DD UNIT=WORKDISK,SPACE=(CYL,(5,2))
//SYSUT4 DD UNIT=WORKDISK,SPACE=(CYL,(5,2))
//ENTRADA DD DSN=REALBIBL,UNIT=WORKDISK,DISP=(OLD,DELETE)
//SALIDA DD DSN=L06201.REALFIS,DISP=OLD
```

```
//SYSIN DD *
C O=SALIDA,I=ENTRADA
/*
//
```

NOTA: Se menciona el DCB con el cual se grabará el disco auxiliar. Cuando la biblioteca deteriorada se borra en el PAS01, no por eso resulta decatalogada, es decir, el nombre L06201.REALFIS continua figurando en el catálogo del sistema. Por lo tanto no es necesario crearla otra vez, sino simplemente llenarla: si se pretendiera crear la biblioteca mediante el procedimiento adecuado, el sistema acusará el error "DIRECT ACCESS-DUPLICATE NAME ON VOLUME", (en el volumen CNEA02 el nombre L06201.REALFIS está duplicado).

Modelo 303. Reposición de un módulo deteriorado de la biblioteca, obteniéndolo de la cinta magnética de resguardo. La biblioteca completa, incluido su directorio, está grabada con el nombre SALVARE3, en el rótulo 1 de una cinta que lleva el número de carrete 13321 y tiene etiquetas modelo IBM. El módulo que se va a reponer se llama MERCURIO. El problema se puede resolver en dos pasos: en el PAS01 el utilitario IEBCOPY transfiere toda la biblioteca de la cinta a un disco auxiliar, con el nombre REALBIBL; en el PAS02 el mismo utilitario selecciona del disco auxiliar, sólo el módulo MERCURIO y lo graba en la biblioteca L06201.REALFIS.

```
//..... JOB ....
//PAS01 EXEC PGM=IEBCOPY
//SYSPRINT DD SYSOUT=A
//SYSUT3 DD UNIT=WORKDISK,SPACE=(CYL,(5,2))
//SYSUT4 DD UNIT=WORKDISK,SPACE=(CYL,(5,2))
//ENTRADA DD DSN=SALVARE3,UNIT=TAPE,VOL=SER=T13321,DISP=OLD
//SALIDA DD DSN=REALBIBL,UNIT=WORKDISK,DISP=(,PASS),
// SPACE=(CYL,(15,,20)),DCB=(BLKSIZE=13030,LRECL=1024,RECFM=U)
//SYSIN DD *
C O=SALIDA,I=ENTRADA
/*
//PAS02 EXEC PGM=IEBCOPY
//SYSPRINT DD SYSOUT=A
//SYSUT3 DD UNIT=WORKDISK,SPACE=(CYL,(5,2))
//SYSUT4 DD UNIT=WORKDISK,SPACE=(CYL,(5,2))
//ENTRADA DD DSN=REALBIBL,UNIT=WORKDISK,DISP=(OLD,DELETE)
//SALIDA DD DSN=L06201.REALFIS,DISP=OLD
//SYSIN DD *
```

```

C  O=SALIDA,I=ENTRADA
S  M=((MERCURIO,,R))
/*
//

```

NOTA: El DCB codificado en la sentencia SALIDA del PASO1, para grabar el disco auxiliar, es propio de la biblioteca L06201.REALFIS.

Modelo 304 . Reposición de varios módulos deteriorados de la biblioteca, obteniéndolos de la cinta magnética de resguardo. La biblioteca completa, incluido su directorio, está grabada con el nombre SALVARE4, en el rótulo 1 de una cinta que lleva el número de carrete 73299 y tiene etiquetas modelo IBM. Los módulos que se van a reponer se llaman PATOM, FALENCIA, TOTALID y FRECUEN; además, al módulo FALENCIA se le cambiará el nombre por el de FALACIA y al módulo TOTALID se le cambiará el nombre por el de PARCIAL. El problema se puede resolver en dos pasos: en el PASO1 el utilitario IEBCOPY transfiere toda la biblioteca de la cinta a un disco auxiliar, con el nombre REALTEMP; en el PASO2 el mismo utilitario selecciona del disco auxiliar solamente los módulos de carga solicitados y los graba en la biblioteca L06201.REALFIS. A los módulos FALENCIA y TOTALID les cambia el nombre antes de grabarlos.

```

//..... JOB ....
//PASO1 EXEC PGM=IEBCOPY
//SYSPRINT DD SYSOUT=A
//SYSUT3 DD UNIT=WORKDISK,SPACE=(CYL,(5,2))
//SYSUT4 DD UNIT=WORKDISK,SPACE=(CYL,(5,2))
//ENTRADA DD DSN=SALVARE4,UNIT=TAPE,VOL=SER=T73299,DISP=OLD
//SALIDA DD DSN=REALTEMP,UNIT=WORKDISK,DISP=(NEW,PASS),
//  SPACE=(CYL,(15,,20)),DCB=(BLKSIZE=13030,LRECL=1024,RECFM=U)
//SYSIN DD *
C  O=SALIDA,I=ENTRADA
/*
//PASO2 EXEC PGM=IEBCOPY
//SYSPRINT DD SYSOUT=A
//SYSUT3 DD UNIT=WORKDISK,SPACE=(CYL,(5,2))
//SYSUT4 DD UNIT=WORKDISK,SPACE=(CYL,(5,2))
//ENTRADA DD DSN=REALTEMP,UNIT=WORKDISK,DISP=(OLD,DELETE)
//SALIDA DD DSN=L06201.REALFIS,DISP=OLD
//SYSIN DD *
C  O=SALIDA,I=ENTRADA
S  M=((PATOM,,R),(FALENCIA,FALACIA,R),(TOTALID,PARCIAL,R),
      (FRECUEN,,R))
/*
//

```

NOTA: La sentencia S del PASO2 admite la mención de hasta ocho miembros. Se indica que la sentencia S tiene continuación codificando un carácter cualquiera, distinto de blanco, en la columna 72 . La tarjeta continuación tiene que empezar en la columna 16. El DCB codificado en la sentencia SALIDA del PASO1, para grabar el disco auxiliar, es propio de la biblioteca L06201.REALNIS.

PUNTO 5.

Resguardo, compresión y listado del directorio con información sobre el número de bloques del mismo que todavía no han sido utilizados, de una biblioteca de nombre PARTCNDS.FISICA. La biblioteca reside en la unidad de discos 3350, con el número de serie del volumen CNEA02.

Modelo 305 . Resguardo de la biblioteca en previsión de deterioros accidentales. Con el nombre SALVASII el utilitario IEBCOPY copia la totalidad de la biblioteca, incluido el directorio, en el rótulo 1 de una cinta magnética que lleva el número de carrete 80000 y tiene etiquetas modelo IBM.

```
//..... JOB ....
//PAS01 EXEC PGM=IEBCOPY
//SYSPRINT DD SYSOUT=A
//SYSUT3 DD UNIT=SYSDA,SPACE=(CYL,(2,1))
//SYSUT4 DD UNIT=SYSDA,SPACE=(CYL,(2,1))
//IN DD DSN=PARTCNDS.FISICA,DISP=SHR
//OUT DD UNIT=TAPE,VOL=SER=T80000,LABEL=(1,SL),DSN=SALVASII,
// DISP=(NEW,KEEP)
//SYSIN DD *
C O=OUT,I=IN
/*
//
```

NOTA: Para que el resguardo de una biblioteca de este tipo sea efectivo, se debe realizar por lo menos en dos cintas distintas, alternadamente, una vez por semana.

Modelo 306 . Compresión de la biblioteca para eliminar los espacios vacíos.

```
//..... JOB ....
//PAS01 EXEC PGM=IEBCOPY
//SYSPRINT DD SYSOUT=A
//SYSUT3 DD UNIT=SYSDA,SPACE=(TRK,(1))
//SYSUT4 DD UNIT=SYSDA,SPACE=(TRK,(1))
//INOUT5 DD DSN=PARTCNDS.FISICA,DISP=OLD
//SYSIN DD *
COPYOPER COPY OUTDD=INOUT5,INDD=INOUT5
/*
//
```

Modelo 307 . Listado del directorio de la biblioteca e información sobre el número de bloques del mismo que todavía no han sido utilizados.

```
//..... JOB ....  
//PAS01 EXEC PGM=IEHLIST  
//SYSPRINT DD SYSOUT=A  
//DD2 DD UNIT=3350,VOL=SER=CNEA02,DISP=OLD  
//SYSIN DD *  
    LISTPDS DSNAME=PARTONDS.FISICA,VOL=3350=CNEA02  
/*  
//
```

PUNTO 6.

Supresión y cambio de nombre de uno o mas miembros de una biblioteca de nombre PARTCNDS.FISICA. La biblioteca reside en la unidad de discos 3350, con el número de serie del volumen CNEA02.

Modelo 308 . Supresión de varios miembros de la biblioteca. Se van a borrar los miembros de nombres SPGRP, SYM y SMI. Se codifica la sentencia SCRATCH para cada miembro.

```
//..... JOB ....
//PAS01 EXEC PGM=IEHPRGM
//SYSPRINT DD SYSOUT=A
//DD1 DD VOL=SER=CNEA02,DISP=OLD,UNIT=3350
//SYSIN DD *
  SCRATCH DSNAME=PARTCNDS.FISICA,VOL=3350=CNEA02,PURGE,MEMBER=SPGRP
  SCRATCH DSNAME=PARTCNDS.FISICA,VOL=3350=CNEA02,PURGE,MEMBER=SYM
  SCRATCH DSNAME=PARTCNDS.FISICA,VOL=3350=CNEA02,PURGE,MEMBER=SMI
/*
//
```

Nota: Es conveniente codificar cada sentencia SCRATCH en una sola tarjeta perforada; si se codifican tarjetas continuación, cualquier error puede provocar la supresión de toda la biblioteca.

Modelo 309 . Cambio de nombre de varios miembros de la biblioteca. El miembro XGUN se llamará XTANA; el miembro DOSSUB se llamará UNASUB y el miembro REFEREN se llamará REFER1. Se codifica una sentencia RENAME para cada uno de los cambios de nombre.

```
//..... JOB ....
//PAS01 EXEC PGM=IEHPRGM
//SYSPRINT DD SYSOUT=A
//DD1 DD VOL=SER=CNEA02,UNIT=3350,DISP=OLD
//SYSIN DD *
  RENAME VOL=3350=CNEA02,DSNAME=PARTCNDS.FISICA,NEWNAME=XTANA,      C
          MEMBER=XGUN
  RENAME VOL=3350=CNEA02,DSNAME=PARTCNDS.FISICA,NEWNAME=UNASUB,      C
          MEMBER=DOSSUB
  RENAME VOL=3350=CNEA02,DSNAME=PARTCNDS.FISICA,NEWNAME=REFER1,      C
          MEMBER=REFEREN
/*
//
```

NOTA: Se indica que la sentencia RENAME tiene continuación codificando un carácter

ter cualquiera, distinto de blanco, en la columna 72. La tarjeta continuación tiene que empezar en la columna 16.

PUNTO 7.

Borrado y decatalogación de bibliotecas fuera de uso. Cambio del nombre de la biblioteca. Obtención del catálogo de bibliotecas contenidas en el volumen.

Modelo 310 . Borrado y decatalogación de una biblioteca fuera de uso de nombre PARTCNDS.FISICA, que reside en la unidad de discos 3350, con el número de serie del volumen CNEA03. El catálogo reside en la misma unidad de discos, con el número de serie del volumen CNEA01.

```
//..... JOB ....
//PAS01 EXEC PGM=IEHPRGM
//SYSPRINT DD SYSOUT=A
//DD1 DD VOL=SER=CNEA01,UNIT=3350,DISP=OLD
//DD2 DD VOL=SER=CNEA03,UNIT=3350,DISP=OLD
//SYSIN DD *
  SCRATCH DSNAME=PARTCNDS.FISICA,VOL=3350=CNEA03,PURGE
  UNCATLG DSNAME=PARTCNDS.FISICA
/*
//
```

Modelo 311 . Cambio de nombre de la biblioteca PARTCNDS.FISICA por el de PROGRAM.FISTEOR con catalogación del nuevo nombre y decatalogación del anterior. La biblioteca reside en la unidad de discos 3350, con el número de serie del volumen CNEA03. El catálogo reside en la misma unidad de discos con el número de serie del volumen CNEA01.

```
//..... JOB ....
//PAS01 EXEC PGM=IEHPRGM
//SYSPRINT DD SYSOUT=A
//DD1 DD UNIT=3350,VOL=SER=CNEA01,DISP=OLD
//DD2 DD UNIT=3350,VOL=SER=CNEA03,DISP=OLD
//SYSIN DD *
  RENAME VOL=3350=CNEA03,DSNAME=PARTCNDS.FISICA,NEWNAME=PROGRAM.FISTEOR
  CATLG VOL=3350=CNEA03,DSNAME=PROGRAM.FISTEOR
  UNCATLG DSNAME=PARTCNDS.FISICA
/*
//
```

Modelo 312 . Obtención del catálogo de las bibliotecas que residen en la unidad de discos 3350, con el número de serie del volumen CNEA02 e información sobre el número de cilindros y pistas libres en cada una de las bibliotecas y en el volumen.

```
//..... JOB ....
//PAS01 EXEC PGM=IEHLIST
//SYSPRINT DD SYSOUT=A
//DD2 DD UNIT=3350,VOL=SER=CNEA02,DISP=OLD
//SYSIN DD *
    LISTVTOC FORMAT,VOL=3350=CNEA02
/*
//
```

Modelo 313 . Obtención del catálogo de las bibliotecas que residen en la unidad de discos 3350, con los números de serie de los volúmenes CNEA01, CNEA02 y CNEA03. Información sobre el número de cilindros y pistas libres en cada una de las bibliotecas y en cada uno de los volúmenes.

```
//..... JOB ....
//PAS01 EXEC PGM=IEHLIST
//SYSPRINT DD SYSOUT=A
//DD1 DD UNIT=3350,VOL=SER=CNEA01,DISP=OLD
//DD2 DD UNIT=3350,VOL=SER=CNEA02,DISP=OLD
//DD3 DD UNIT=3350,VOL=SER=CNEA03,DISP=OLD
//SYSIN DD *
    LISTVTOC FORMAT,VOL=3350=CNEA01
    LISTVTOC FORMAT,VOL=3350=CNEA02
    LISTVTOC FORMAT,VOL=3350=CNEA03
/*
//
```

CAPITULO VII

MODELOS PARA LISTADO, GRABACION Y TRANSFERENCIA DE ARCHIVOS EN CINTAS MAGNETICAS

PUNTO 1.

Listado de cintas magnéticas

Modelo 314. Listado de los rótulos 5, 7, 21 y 38 de una cinta magnética que tiene etiquetas modelo IBM, lleva el número de carrete 53319 y fué grabada con el mismo DCB en todos sus rótulos.

```
//..... JOB ....
//PASO1 EXEC PGM=IEBPTPCH
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=PROGRM16,UNIT=TAPE,VOL=SER=T53319,
// LABEL=(5,SL),DISP=(OLD,PASS)
//SYSUT2 DD SYSOUT=A
//SYSIN DD *
PRINT MAXFLDS=1
RECORD FIELD=(80)
/*
//PASO2 EXEC PGM=IEBPTPCH
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=TURBAB,UNIT=TAPE,VOL=REF=*.PASO1.SYSUT1,LABEL=(7,SL)
//SYSUT2 DD SYSOUT=A
//SYSIN DD *
PRINT MAXFLDS=1
RECORD FIELD=(80)
/*
//PASO3 EXEC PGM=IEBPTPCH
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=ALFA12,UNIT=TAPE,VOL=REF=*.PASO1.SYSUT1,LABEL=(21,SL)
//SYSUT2 DD SYSOUT=A
//SYSIN DD *
PRINT MAXFLDS=1
RECORD FIELD=(80)
/*
//PASO4 EXEC PGM=IEBPTPCH
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=PROGRM17,UNIT=TAPE,VOL=REF=*.PASO1.SYSUT1,LABEL=(38,SL),
// DISP=(OLD,KEEP)
//SYSUT2 DD SYSOUT=A
//SYSIN DD *
PRINT MAXFLDS=1
RECORD FIELD=(80)
/*
//
```

NOTA: La Referencia hacia Atrás empleada en la sentencia SYSUT1 aligera la perforación de tarjetas en un programa de muchos pasos.

Modelo 315. Listado de los rótulos 3, 17 y 20 de una cinta magnética que tiene etiquetas modelo IBM, lleva el número de carrete 10055 y fué grabada con distintos DCB en dos o mas de sus rótulos.

```

//..... JOB ....
//PAS01 EXEC PGM=IEBPTPCH
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=VALOR37,UNIT=TAPE,VOL=SER=T10055,LABEL=(3,SL),
// DISP=(OLD,PASS)
//SYSUT2 DD SYSOUT=A
//SYSIN DD *
PRINT MAXFLDS=1
RECORD FIELD=(80)
/*
//PAS02 EXEC PGM=IEBPTPCH
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=TURBAC,UNIT=TAPE,VOL=SER=T10055,LABEL=(17,SL),
// DISP=(OLD,PASS)
//SYSUT2 DD SYSOUT=A
//SYSIN DD *
PRINT MAXFLDS=1
RECORD FIELD=(80)
/*
//PAS03 EXEC PGM=IEBPTPCH
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=FELAP,UNIT=TAPE,VOL=SER=T10055,LABEL=(20,SL),
// DISP=(OLD,KEEP)
//SYSUT2 DD SYSOUT=A
//SYSIN DD *
PRINT MAXFLDS=1
RECORD FIELD=(80)
/*
//

```

Modelo 316 . Listado de los rótulos 2, 4, 15 y 18 de una cinta magnética que no tiene etiquetas y de la cual se conoce el DCB con que fué grabado cada rótulo. La cinta lleva el número de carrete 22806 y está grabada con DEN=3 (1600 bpi) que es la densidad que el sistema supone por omisión.

```

//..... JOB ....
//PAS01 EXEC PGM=IEBPTPCH
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD UNIT=TAPE,VOL=SER=T22806,LABEL=(2,NL),DISP=(OLD,PASS),
// DCB=(BLKSIZE=800,LRECL=80,RECFM=FB)
//SYSUT2 DD SYSOUT=A
//SYSIN DD *
PRINT MAXFLDS=1
RECORD FIELD=(80)
/*
//PAS02 EXEC PGM=IEBPTPCH
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD UNIT=TAPE,VOL=SER=T22806,LABEL=(4,NL),DISP=(OLD,PASS),
// DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//SYSUT2 DD SYSOUT=A
//SYSIN DD *
PRINT MAXFLDS=1

```

```

RECORD FIELD=(80)
/*
//PASO3 EXEC PGM=IEBPTPCH
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD UNIT=TAPE,VOL=SER=T22806,LABEL=(15,NL),DISP=(OLD,PASS),
//  DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//SYSUT2 DD SYSOUT=A
//SYSIN DD *
PRINT MAXFLDS=1
RECORD FIELD=(80)
/*
//PASO4 EXEC PGM=IEBPTPCH
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD UNIT=TAPE,VOL=SER=T22806,LABEL=(18,NL),DISP=(OLD,PASS),
//  DCB=(BLKSIZE=2400,LRECL=80,RECFM=FB)
//SYSUT2 DD SYSOUT=A
//SYSIN DD *
PRINT MAXFLDS=1
RECORD FIELD=(80)
/*
//

```

Modelo 317 . Listado de los rótulos 3, 6 y 20 de una cinta magnética que no tiene etiquetas y de la cual se desconoce el DCB con el cual fué grabado cada rótulo. La cinta lleva el número de carrete 74193 y está grabada con DEN=2 (800 bpi).

```

//..... JOB ....
//PASO1 EXEC PROC=OSLISTA
//SYSUT1 DD UNIT=TAPE,DISP=(OLD,PASS),VOL=SER=T74193,LABEL=(3,NL),
//  DCB=(BLKSIZE=8000,RECFM=U,DEN=2)
PRINT FL=C,LR=100
/*
//PASO2 EXEC PROC=OSLISTA
//SYSUT1 DD UNIT=TAPE,DISP=(OLD,PASS),VOL=SER=T74193,LABEL=(6,NL),
//  DCB=(BLKSIZE=8000,RECFM=U,DEN=2)
PRINT FL=C,LR=450
/*
//PASO3 EXEC PROC=OSLISTA
//SYSUT1 DD UNIT=TAPE,DISP=(OLD,PASS),VOL=SER=T74193,LABEL=(20,NL),
//  DCB=(BLKSIZE=8000,RECFM=U,DEN=2)
PRINT FL=C
/*
//

```

NOTA: El parámetro LR de la sentencia PRINT indica el número correlativo del último registro lógico de entrada a procesar en cada rótulo; por consiguiente en el PASO1 se leerán solamente 100 registros y en el PASO2 solamente 450. En el PASO3, por no tener codificado el parámetro LR el sistema supone infinito, es decir, se leerán todos los registros que hubiera en el rótulo 20.

Modelo 318 . Listado de un programa en FORTRAN que reside, junto con otros, en el mismo rótulo de una cinta magnética, formando un archivo secuencial. En el rótulo 12 de la cinta que lleva el número de carrete 56673 y tiene etiquetas modelo IBM, se ha grabado con el nombre VARIOSL1 un conjunto de 20 programas y rutinas. Se desea el listado del programa ROMBERG solamente, que ocupa el cuarto lugar en ese archivo secuencial. El problema se puede resolver en dos pasos: en el PASO1 el utilitario IEBGENER identifica en la cinta, reconociendo la sentencia FORTRAN 'END' , los cuatro primeros programas y rutinas, creando con ellos una biblioteca temporaria particionada que se graba en un disco auxiliar; en el PASO2 el utilitario IEBPTPCH obtiene de la biblioteca temporaria el miembro ROMBERG y produce su listado.

```
//..... JOB ....
//PASO1 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD UNIT=TAPE,VOL=SER=T56673,DSN=VARIOSL1,LABEL=(12,SL),
//  DISP=(OLD,KEEP)
//SYSUT2 DD UNIT=WORKDISK,DSN=TEMPOR.ADR(VARIOSL1),DISP=(NEW,PASS),
//  SPACE=(CYL,(1,1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//SYSIN DD *
      GENERATE MAXNAME=4,MAXGPS=4
      MEMBER NAME=FORTIT
GROUP1 RECORD IDENT=(3,'END',7)
      MEMBER NAME=MONO65
GROUP2 RECORD IDENT=(3,'END',7)
      MEMBER NAME=MADO1B
GROUP3 RECORD IDENT=(3,'END',7)
      MEMBER NAME=ROMBERG
GROUP4 RECORD IDENT=(3,'END',7)
/*
//PASO2 EXEC PGM=IEBPTPCH
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=TEMPOR.ADR,UNIT=WORKDISK,DISP=(OLD,DELETE)
//SYSUT2 DD SYSOUT=A
//SYSIN DD *
      PRINT TYPOFC=PO,MAXNAME=1,MAXFLDS=1
      MEMBER NAME=ROMBERG
      RECORD FIELD=(80)
/*
//
```

NOTA: El parámetro IDENT=(3,'END',7) del PASO1 significa: identificar una tira de 3 caracteres, igual a la que aparece entre comillas, que está colocada de la columna 7 en adelante, en la imagen de tarjeta grabada en la cinta.

Modelo 319 . Listado de varios programas en FORTRAN que residen, junto con otros, en el mismo rótulo de una cinta magnética formando un archivo secuencial. En el rótulo 7 de la cinta que lleva el número de carrete 56670 y tiene etiquetas modelo IBM, se ha grabado con el nombre PROGR978 un conjunto de 15 programas y rutinas. Se desean los listados de los programas ANCORA y TWOQUAPA solamente, que ocupan el quinto y octavo lugar respectivamente, en ese archivo secuencial. El problema se puede resolver en dos pasos: en el PASO1 el utilitario IEBGENER identifica en la cinta, reconociendo la sentencia FORTRAN 'END', los ocho primeros programas y rutinas, creando con ellos una biblioteca temporaria particionada que se graba en un disco auxiliar; en el PASO2 el utilitario IEBPTPCH obtiene de la biblioteca temporaria los miembros ANCORA y TWOQUAPA y produce sus listados.

```
//..... JOB ....
//PASO1 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD UNIT=TAPE,VOL=SER=T56670,LABEL=(7,SL),DSN=PROGR978,
// DISP=(OLD,KEEP)
//SYSUT2 DD UNIT=WORKDISK,DSN=TEMPOR.ADR(PROGR978),DISP=(NEW,PASS),
// SPACE=(CYL,(1,1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//SYSIN DD *
    GENERATE MAXNAME=8,MAXGPS=8
    MEMBER NAME=BAR35
GROUP1 RECORD IDENT=(3,'END',7)
    MEMBER NAME=CORRELAC
GROUP2 RECORD IDENT=(3,'END',7)
    MEMBER NAME=FURSAN
GROUP3 RECORD IDENT=(3,'END',7)
    MEMBER NAME=DOSCUERP
GROUP4 RECORD IDENT=(3,'END',7)
    MEMBER NAME=ANCORA
GROUP5 RECORD IDENT=(3,'END',7)
    MEMBER NAME=SUSTAL
GROUP6 RECORD IDENT=(3,'END',7)
    MEMBER NAME=SUISSE
GROUP7 RECORD IDENT=(3,'END',7)
    MEMBER NAME=TWOQUAPA
GROUP8 RECORD IDENT=(3,'END',7)
/*
//PASO2 EXEC PGM=IEBPTPCH
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=TEMPOR.ADR,UNIT=WORKDISK,DISP=(OLD,DELETE)
//SYSUT2 DD SYSOUT=A
//SYSIN DD *
    PRINT TYPORG=PO,MAXNAME=2,MAXFLDS=2
    MEMBER NAME=ANCORA
    RECORD FIELD=(80)
    MEMBER NAME=TWOQUAPA
```



```

RECORD FIELD=(80)
/*
//

```

NOTA: El parámetro IDENT=(3,'END',7) del PAS01 significa: identificar una tira de 3 caracteres, igual a la que aparece entre comillas, que está colocada de la columna 7 en adelante, en la imagen de tarjeta grabada en la cinta.

Modelo 320 . Listado de una rutina en lenguaje simbólico que reside, junto con otras, en el mismo rótulo de una cinta magnética y está marcada en cada uno de sus registros con una tira de caracteres. En el rótulo 22 de una cinta que lleva el número de carrete 19117 y tiene etiquetas modelo IBM, está grabado, con el nombre PROGRFIS un conjunto de 200 programas, rutinas y funciones cada uno de los cuales está marcado en la columna 73 y siguientes con una abreviatura de su nombre. Se desea listar solamente la rutina POTENTIA identificando la tira de caracteres POTNT.

```

//..... JOB ....
//PAS01 EXEC PROC=OSLISTA
//SYSUT1 DD DSN=PROGRFIS,UNIT=TAPE,VOL=SER=T19117,LABEL=(22,SL),
// DISP=(OLD,KEEP)
//SYSIN DD *
RCIN 73,,'POTNT'
PRINT FL=C
/*
//

```

Modelo 321 . Listado de un programa, una función y una rutina en lenguaje simbólico que residen, junto con otros, en el mismo rótulo de una cinta magnética y están marcados en cada uno de sus registros con una tira de caracteres. En el rótulo 34 de una cinta que lleva el número de carrete 28300 y tiene etiquetas modelo IBM, está grabado, con el nombre PGMBORR un conjunto de 50 programas, rutinas y funciones cada uno de los cuales está marcado en la columna 73 y siguientes con una abreviatura de su nombre. Se desean listar solamente el programa COLLECDB, la función ERRORCCT y la rutina POLARID, identificando las tiras de caracteres CLLEC, ERRCT y POLR respectivamente.

```

//..... JOB ....

```

```
//PAS01 EXEC PROC=OSLISTA
//SYSUT1 DD DSN=PGMBOHR,UNIT=TAPE,VOL=SER=T28300,LABEL=(34,SL),
// DISP=(OLD,KEEP)
//SYSIN DD *
RCIN 73,, 'CLLEC'
RCIN 73,, 'ERRCT'
RCIN 73,, 'POLR'
PRINT FL=C
/*
//
```

Modelo 322 . Listado de los nombres solamente, de un conjunto de rutinas y funciones en lenguaje simbólico FORTRAN, grabadas secuencialmente en el mismo rótulo de una cinta magnética. En el rótulo 7 de la cinta que lleva el número de carrete 05102 y tiene etiquetas modelo IBM está grabado, con el nombre BIBLADR, un conjunto de 150 programas, rutinas y funciones; se desea la lista de los nombres de las rutinas y de las funciones..El problema se puede resolver en dos pasos: en el PAS01 el procedimiento OSLISTA identifica la tira de caracteres 'SUBROUTINE' a partir de la columna 7 y en el PAS02 el mismo procedimiento identifica la tira de caracteres 'FUNCTION' también a partir de la columna 7.

```
//..... JOB .....
//PAS01 EXEC PROC=OSLISTA
//SYSUT1 DD DSN=BIBLADR,UNIT=TAPE,VOL=SER=T05102,LABEL=(7,SL),
// DISP=(OLD,KEEP)
//SYSIN DD *
RCIN 7,, 'SUBROUTINE'
PRINT FL=C
/*
//PAS02 EXEC PROC=OSLISTA
//SYSUT1 DD DSN=BIBLADR,UNIT=TAPE,VOL=SER=T05102,LABEL=(7,SL),
// DISP=(OLD,KEEP)
//SYSIN DD *
RCIN 7,, 'FUNCTION'
PRINT FL=C
/*
//
```

Modelo 323 . Listado de un miembro de una biblioteca particionada de programas en lenguaje simbólico, que fué grabada en su totalidad (incluido el directorio), con el nombre COPIABIB, en el rótulo 1 de una cinta magnética, utilizando el utilitario IEBCOPY. La cinta lleva el número de carrete 61123 y tiene etiquetas modelo IBM. El miembro que se desea listar tiene el nombre TWOPAR. El problema se

puede resolver en dos pasos: en el PAS01 el utilitario IEBCOPY transfiere el contenido del rótulo 1 a un disco auxiliar, creando una biblioteca temporaria particionada; en el PAS02 el utilitario IEBPTPCH produce el listado del programa TWOPAR.

```
//..... JOB ....
//PAS01 EXEC PGM=IEBCOPY
//SYSPRINT DD SYSOUT=A
//SYSUT3 DD UNIT=WORKDISK,SPACE=(CYL,(5,2))
//SYSUT4 DD UNIT=WORKDISK,SPACE=(CYL,(5,2))
//ENTRADA DD DSN=COPIABIB,UNIT=TAPE,VOL=SER=T61123,DISP=(OLD,KEEP)
//SALIDA DD DSN=TEMPOR.COPIA,UNIT=WORKDISK,DISP=(NEW,PASS),
// SPACE=(CYL,(15,,20),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//SYSIN DD *
C O=SALIDA,I=ENTRADA
/*
//PAS02 EXEC PGM=IEBPTPCH
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=TEMPOR.COPIA,UNIT=WORKDISK,DISP=(OLD,DELETE)
//SYSUT2 DD SYSOUT=A
//SYSIN DD *
PRINT TYPORG=PO,MAXFLDS=1,MAXNAME=1
MEMBER NAME=TWOPAR
RECORD FIELD=(80)
/*
//
```

Modelo 324 . Listado de varios miembros de una biblioteca particionada de programas en lenguaje simbólico, que fué grabada en su totalidad (incluido el directorio), con el nombre BIBDEFIS, en el rótulo 1 de una cinta magnética, utilizando el utilitario IEBCOPY. La cinta lleva el número de carrete 20550 y tiene etiquetas modelo IBM. Los miembros que se desean listar tienen los nombres CARGA y MALEK. El problema se puede resolver en dos pasos: en el PAS01 el utilitario IEBCOPY transfiere el contenido del rótulo 1 a un disco auxiliar, creando una biblioteca temporaria particionada; en el PAS02 el utilitario IEBPTPCH produce el listado de los miembros CARGA y MALEK.

```
//..... JOB ....
//PAS01 EXEC PGM=IEBCOPY
//SYSPRINT DD SYSOUT=A
//SYSUT3 DD UNIT=WORKDISK,SPACE=(CYL,(5,2))
//SYSUT4 DD UNIT=WORKDISK,SPACE=(CYL,(5,2))
//ENTRADA DD DSN=BIBDEFIS,UNIT=TAPE,VOL=SER=T20550,DISP=(OLD,KEEP)
//SALIDA DD DSN=TEMPOR.FISICA,UNIT=WORKDISK,DISP=(NEW,PASS),
```

```
// SPACE=(CYL,(15,,20)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//SYSIN DD *
C O=SALIDA,I=ENTRADA
/*
//PASO2 EXEC PGM=IEBPTPCH
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=TEMPOR.FISICA,UNIT=WORKDISK,DISP=(OLD,DELETE)
//SYSUT2 DD SYSOUT=A
//SYSIN DD *
PRINT TYPORG=PO,MAXFLDS=2,MAXNAME=2
MEMBER NAME=CARGA
RECORD FIELD=(80)
MEMBER NAME=MALEK
RECORD FIELD=(80)
/*
//
```

Modelo 325 . Listado de los n primeros registros de un rótulo cualquiera de una cinta magnética que tiene etiquetas modelo IBM. Se desea el listado de los 5 primeros registros del rótulo 20, que fué grabado con el nombre ELEMENT en la cinta que lleva el número de carrete 90190.

```
//..... JOB ....
//PASO1 EXEC PGM=IEBPTPCH
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=ELEMENT,UNIT=TAPE,VOL=SER=T90190,LABEL=(20,SL),
// DISP=(OLD,KEEP)
//SYSUT2 DD SYSOUT=A
//SYSIN DD *
PRINT STOPAFT=5,MAXFLDS=1
RECORD FIELD=(80)
/*
//
```

Modelo 326 . Listado de n registros intermedios de un rótulo cualquiera de una cinta magnética que tiene etiquetas modelo IBM. Se desea el listado de los registros del rótulo 16, desde el registro de orden 25 inclusive hasta el registro de orden 61 inclusive (37 registros en total). El rótulo fué grabado con el nombre RUTENIO en la cinta que lleva el número de carrete 58273.

```
//..... JOB ....
//PASO1 EXEC PGM=IEBPTPCH
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=RUTENIO,UNIT=TAPE,VOL=SER=T58273,LABEL=(16,SL),
// DISP=(OLD,KEEP)
//SYSUT2 DD SYSOUT=A
//SYSIN DD *
```

```

PRINT STRTAFT=24,STOPAFT=37,MAXFLDS=1
RECORD FIELD=(80)
/*
//

```

Modelo 327 . Listado de los n primeros registros de un rótulo cualquiera de una cinta magnética que no tiene etiquetas y lleva el número de carrete 56671. Se desea el listado de los 3 primeros registros del rótulo 11.

```

//..... JOB ....
//PAS01 EXEC PGM=IEBPTPCH
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD UNIT=TAPE,LABEL=(11,NL),DISP=OLD,VOL=SER=T.6671,
// DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//SYSUT2 DD SYSOUT=A
//SYSIN DD *
PRINT STOPAFT=3,MAXFLDS=1
RECORD FIELD=(80)
/*
//

```

NOTA; El DCB con el cual se lee la cinta no es necesariamente aquél con el cual fué grabada.

Modelo 328 . Listado de n registros intermedios de un rótulo cualquiera de una cinta magnética que no tiene etiquetas y lleva el número de carrete 01923. Se desea el listado de los registros del rótulo 28, desde el registro de orden 81 inclusive hasta el registro 125 inclusive (45 registros en total).

```

//..... JOB ....
//PAS01 EXEC PGM=IEBPTPCH
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD UNIT=TAPE,LABEL=(28,NL),DISP=OLD,VOL=SER=001923,
// DCB=(BLKSIZE=2400,LRECL=80,RECFM=FB)
//SYSUT2 DD SYSOUT=A
//SYSIN DD *
PRINT STRTAFT=80,STOPAFT=45,MAXFLDS=1
RECORD FIELD=(80)
/*
//

```

NOTA: El DCB con el cual se lee la cinta no es necesariamente aquél con el cual fué grabada.

PUNTO 2.

Obtención de tarjetas perforadas a partir de cintas magnéticas.

Modelo 329 . Obtención de las tarjetas perforadas de los rótulos 8 y 14 de una cinta magnética que tiene etiquetas modelo IBM y lleva el número de carrete 64207.

```
//..... JOB ....
//PAS01 EXEC PGM=IEBTPCH
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=BARM,UNIT=TAPE,VOL=SER=T64207,LABEL=(8,SL),
// DISP=(OLD,PASS)
//SYSUT2 DD SYSOUT=B
//SYSIN DD *
  PUNCH MAXFLDS=1
  RECORD FIELD=(80)
/*
//PAS02 EXEC PGM=IEBTPCH
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=BARM13,UNIT=TAPE,VOL=SER=T64207,LABEL=(14,SL),
// DISP=(OLD,KEEP)
//SYSUT2 DD SYSOUT=B
//SYSIN DD *
  PUNCH MAXFLDS=1
  RECORD FIELD=(80)
/*
//
```

NOTA: Si no se menciona el DCB, la perforación se realiza con el DCB con que fué grabada la cinta.

Modelo 330 . Obtención de las tarjetas perforadas de los rótulos 25 y 43 de una cinta magnética que no tiene etiquetas y lleva el número de carrete 85923.

```
//..... JOB ....
//PAS01 EXEC PGM=IEBTPCH
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD UNIT=TAPE,VOL=SER=T85923,LABEL=(25,NL),DISP=(OLD,PASS),
// DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//SYSUT2 DD SYSOUT=B
//SYSIN DD *
  PUNCH MAXFLDS=1
  RECORD FIELD=(80)
/*
//PAS02 EXEC PGM=IEBTPCH
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD UNIT=TAPE,VOL=SER=T85923,LABEL=(43,NL),DISP=(OLD,PASS),
// DCB=(BLKSIZE=2000,LRECL=80,RECFM=FB)
//SYSUT2 DD SYSOUT=B
//SYSIN DD *
  PUNCH MAXFLDS=1
```

```
RECORD FIELD=(80)
```

```
/*  
//
```

NOTA: Se debe mencionar en cada paso el DCB con el cual se efectuará la perforación de las tarjetas. No es necesariamente el DCB con el cual fué grabada la cinta.

Modelo 331 . Obtención de las tarjetas perforadas de un programa en FORTRAN que reside, junto con otros, en el mismo rótulo de una cinta magnética, formando un archivo secuencial. En el rótulo 5 de la cinta que lleva el número de carrete 00092 y tiene etiquetas modelo IBM, se ha grabado con el nombre ARCHSEC un conjunto de 30 programas y rutinas. Se desean las tarjetas perforadas del programa ASEMFORZ solamente, que ocupa el quinto lugar en ese archivo secuencial. El problema se puede resolver en dos pasos: en el PASO1 el utilitario IEBGENER identifica en la cinta, reconociendo la sentencia FORTRAN 'END', los cinco primeros programas y rutinas, creando con ellos una biblioteca temporaria particionada que se graba en un disco auxiliar; en el PASO2 el utilitario IEBPTPCH obtiene de la biblioteca temporaria el miembro ASEMFORZ y produce el lote de tarjetas perforadas.

```
//..... JOB ....  
//PASO1 EXEC PGM=IEBGENER  
//SYSPRINT DD SYSOUT=A  
//SYSUT1 DD UNIT=TAPE,VOL=SER=T00092,LABEL=(5,SL),DSN=ARCHSEC,  
// DISP=(OLD,KEEP)  
//SYSUT2 DD UNIT=WORKDISK,DSN=TEMPOR.ADR(ARCHSEC),DISP=(,PASS),  
// SPACE=(CYL,(1,1,1)),DCB=(BLKSIZE=1600,LRECL=80,RECFM=FB)  
//SYSIN DD *  
    GENERATE MAXNAME=5,MAXGPS=5  
    MEMBER NAME=SOLIT  
GROUP1 RECORD IDENT=(3 'END',7)  
    MEMBER NAME=UNIT  
GROUP2 RECORD IDENT=(3 'END',7)  
    MEMBER NAME=PARAM  
GROUP3 RECORD IDENT=(3 'END',7)  
    MEMBER NAME=COORDEN  
GROUP4 RECORD IDENT=(3 'END',7)  
    MEMBER NAME=ASEMFORZ  
GROUP5 RECORD IDENT=(3 'END',7)  
/*  
//PASO2 EXEC PGM=IEBPTPCH
```

```
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=TEMPOR.ADR,UNIT=WORKDISK,DISP=(OLD,DELETE)
//SYSUT2 DD SYSOUT=B
//SYSIN DD *
  PUNCH TYPORG=PC,MAXNAME=1,MAXFLDS=1
  MEMBER NAME=ASEMFURZ
  RECORD FIELD=(80)
/*
//
```

NOTA: El parámetro IDENT=(3,'END',7) del PAS01 significa: identificar una tira de 3 caracteres, igual a la que aparece entre comillas, que está colocada de la columna 7 en adelante, en la imagen de tarjeta grabada en la cinta.

Modelo 332 . Obtención de las tarjetas perforadas de varios programas en FORTRAN que residen, junto con otros, en el mismo rótulo de una cinta magnética formando un archivo secuencial. En el rótulo 18 de la cinta que lleva el número de carrete 61824 y tiene etiquetas modelo IBM, se ha grabado con el nombre BIBLRUTI un conjunto de 40 programas y rutinas. Se desean las tarjetas perforadas de los programas TABLA6 y TABLA7 solamente, que ocupan el tercero y décimo lugar respectivamente, en ese archivo secuencial. El problema se puede resolver en dos pasos: en el PAS01 el utilitario IEBGENER identifica en la cinta, reconociendo la sentencia FORTRAN 'END', los diez primeros programas y rutinas, creando con ellos una biblioteca temporaria particionada que se graba en un disco auxiliar; en el PAS02 el utilitario IEBPTPCH obtiene de la biblioteca temporaria los miembros TABLA6 y TABLA7 y produce los correspondientes lotes de tarjetas perforadas.

```
//..... JOB ....
//PAS01 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD UNIT=TAPE,VOL=SER=T61824,LABEL=(18,SL),DSN=BIBLRUTI,
//  DISP=(OLD,KEEP)
//SYSUT2 DD UNIT=WORKDISK,DSN=TEMPOR.ADR(BIBLRUTI),DISP=(,PASS),
//  SPACE=(CYL,(1,1,1)),DCB=(BLKSIZE=800,LRECL=80,RECFM=FB)
//SYSIN DD *
  GENERATE MAXNAME=10,MAXGPS=10
  MEMBER NAME=DUPLIC
GROUP1 RECORD IDENT=(3,'END',7)
  MEMBER NAME=ASISTEN
GROUP2 RECORD IDENT=(3,'END',7)
  MEMBER NAME=TABLA6
GROUP3 RECORD IDENT=(3,'END',7)
```



```

        MEMBER NAME=ASINTOT
GROUP4 RECORD IDENT=(3,'END',7)
        MEMBER NAME=SECTOR
GROUP5 RECORD IDENT=(3,'END',7)
        MEMBER NAME=CONVERS
GROUP6 RECORD IDENT=(3,'END',7)
        MEMBER NAME=HIPOTEN
GROUP7 RECORD IDENT=(3,'END',7)
        MEMBER NAME=CUADMIN
GROUP8 RECORD IDENT=(3,'END',7)
        MEMBER NAME=AJUSTE
GROUP9 RECORD IDENT=(3,'END',7)
        MEMBER NAME=TABLA7
GROUP10 RECORD IDENT=(3,'END',7)
/*
//PASO2 EXEC PGM=IEBPTPCH
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=TEMPOR.ADR,UNIT=WORKDISK,DISP=(OLD,DELETE)
//SYSUT2 DD SYSOUT=B
//SYSIN DD *
PUNCH TYPCRG=PO,MAXNAME=2,MAXFLDS=2
MEMBER NAME=TABLA6
RECORD FIELD=(80)
MEMBER NAME=TABLA7
RECORD FIELD=(80)
/*
//

```

NOTA: El parámetro IDENT=(3,'END',7) del PASO1 significa: identificar una tira de 3 caracteres, igual a la que aparece entre comillas, que está colocada de la columna 7 en adelante, en la imagen de tarjeta grabada en la cinta.

Modelo 333 . Obtención de las tarjetas perforadas de un miembro de una biblioteca particionada de programas en lenguaje simbólico, que fué grabada en su totalidad (incluido el directorio), con el nombre BIBLFISI, en el rótulo 1 de una cinta magnética, utilizando el utilitario IEBCOPY. La cinta lleva el número de carrete 10005 y tiene etiquetas modelo IBM. Se desean las tarjetas perforadas del programa SUTIL solamente. El problema se puede resolver en dos pasos: en el PASO1 el utilitario IEBCOPY transfiere el contenido del rótulo 1 a un disco auxiliar creando una biblioteca temporaria particionada; en el PASO2 el utilitario IEBPTPCH produce el lote de tarjetas perforadas del programa SUTIL.

```

//..... JOB ....
//PASO1 EXEC PGM=IEBCOPY
//SYSPRINT DD SYSOUT=A

```

```
//SYSUT3 DD UNIT=WORKDISK,SPACE=(CYL,(5,2))
//SYSUT4 DD UNIT=WORKDISK,SPACE=(CYL,(5,2))
//ENTRADA DD DSN=BIBLFISI,UNIT=TAPE,VOL=SER=T10005,DISP=(OLD,KEEP)
//SALIDA DD DSN=TEMPOR.ADR,UNIT=WORKDISK,DISP=(NEW,PASS),
// SPACE=(CYL,(15,,20)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//SYSIN DD *
  C O=SALIDA,I=ENTRADA
/*
//PASO2 EXEC PGM=IEBPTPCH
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=TEMPOR.ADR,UNIT=WORKDISK,DISP=(OLD,DELETE)
//SYSUT2 DD SYSOUT=B
//SYSIN DD *
  PUNCH TYPORG=PO,MAXFLDS=1,MAXNAME=1
  MEMBER NAME=SUTIL
  RECORD FIELD=(80)
/*
//
```

Modelo 334 . Obtención de las tarjetas perforadas de varios miembros de una biblioteca particionada de programas en lenguaje simbólico, que fué grabada en su totalidad (incluido el directorio), con el nombre FISICNUC, en el rótulo 1 de una cinta magnética, utilizando el utilitario IEBCOPY. La cinta tiene etiquetas modelo IBM y lleva el número de carrete 15158. Se desean las tarjetas perforadas de los miembros de nombres COLTAR y COLCIN. El problema se puede resolver en dos pasos: en el PASO1 el utilitario IEBCOPY transfiere el contenido del rótulo 1 a un disco auxiliar, creando una biblioteca temporaria particionada; en el PASO2 el utilitario IEBPTPCH produce los lotes de tarjetas correspondientes a los miembros COLTAR y COLCIN.

```
//..... JOB ....
//PASO1 EXEC PGM=IEBCOPY
//SYSPRINT DD SYSOUT=A
//SYSUT3 DD UNIT=WORKDISK,SPACE=(CYL,(5,2))
//SYSUT4 DD UNIT=WORKDISK,SPACE=(CYL,(5,2))
//ENTRADA DD DSN=FISICNUC,UNIT=TAPE,VOL=SER=T15158,DISP=(OLD,KEEP)
//SALIDA DD DSN=TEMPOR.NUCLEAR,UNIT=WORKDISK,DISP=(NEW,PASS),
// SPACE=(CYL,(15,,20)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//SYSIN DD *
  C O=SALIDA,I=ENTRADA
/*
//PASO2 EXEC PGM=IEBPTPCH
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=TEMPOR.NUCLEAR,UNIT=WORKDISK,DISP=(OLD,DELETE)
//SYSUT2 DD SYSOUT=B
//SYSIN DD *
```

```

PUNCH TYPORG=PO,MAXFLDS=2,MAXNAME=2
MEMBER NAME=COLTAR
RECORD FIELD=(80)
MEMBER NAME=COLCIN
RECORD FIELD=(80)
/*
//

```

Modelo 335 . Obtención de las tarjetas perforadas de los n primeros registros de un rótulo cualquiera de una cinta magnética que tiene etiquetas modelo IBM. Se cesea el lote de tarjetas de los 90 primeros registros del rótulo 32, que fué grabado con el nombre PLUTONIO en la cinta que lleva el número de carrete 21572.

```

//..... JOB ....
//PAS01 EXEC PGM=IEBPTPCH
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=PLUTONIO,UNIT=TAPE,VOL=SER=T21572,LABEL=(32,SL),
// DISP=(OLD,KEEP)
//SYSUT2 DD SYSOUT=B
//SYSIN DD *
PUNCH STOPAFT=90,MAXFLDS=1
RECORD FIELD=(80)
/*
//

```

Modelo 336 . Obtención de las tarjetas perforadas de n registros intermedios de un rótulo cualquiera de una cinta magnética que tiene etiquetas modelo IBM. Se desea el lote de tarjetas de los registros del rótulo 25, desde el registro de orden 64 inclusive hasta el registro de orden 500 inclusive (437 registros en total). El rótulo fué grabado con el nombre TABLAQ en la cinta que lleva el número de carrete 90000.

```

//..... JOB ....
//PAS01 EXEC PGM=IEBPTPCH
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=TABLAQ,UNIT=TAPE,VOL=SER=T90000,LABEL=(25,SL),
// DISP=(OLD,KEEP)
//SYSUT2 DD SYSOUT=B
//SYSIN DD *
PUNCH STRAFT=63,STOPAFT=437,MAXFLDS=1
RECORD FIELD=(80)
/*
//

```

Modelo 337 . Obtención de las tarjetas perforadas de los n primeros registros de un rótulo cualquiera de una cinta magnética que no tiene etiquetas de cabecera y

lleva el número de carrete 02507. Se desea el lote de tarjetas de los 120 primeros registros del rótulo 16.

```
//..... JOB ....
//PASO1 EXEC PGM=IEBTPCH
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD UNIT=TAPE,VOL=SER=T02507,LABEL=(16,NL),DISP=OLD,
//  DCB=(BLKSIZE=800,LRECL=80,RECFM=FB)
//SYSUT2 DD SYSOUT=B
//SYSIN DD *
PUNCH STOPAFT=120,MAXFLDS=1
RECORD FIELD=(80)
/*
//
```

Modelo 338. Obtención de las tarjetas perforadas de n registros intermedios de un rótulo cualquiera de una cinta magnética que no tiene etiquetas de cabecera y lleva el número de carrete 47856. Se desea el lote de tarjetas de los registros del rótulo 47, desde el registro de orden 90 inclusive hasta el registro de orden 613 inclusive (524 registros en total).

```
//..... JOB ....
//PASO1 EXEC PGM=IEBTPCH
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD UNIT=TAPE,VOL=SER=T47856,LABEL=(47,NL),DISP=OLD,
//  DCB=(BLKSIZE=1600,LRECL=80,RECFM=FB)
//SYSUT2 DD SYSOUT=B
//SYSIN DD *
PUNCH STRTAFT=89,STOPAFT=524,MAXFLDS=1
RECORD FIELD=(80)
/*
//
```

PUNTO 3.

Grabación en cintas magnéticas de programas que se presentan en tarjetas perforadas.

Modelo 339 . Grabación de programas que se presentan en tarjetas perforadas en lenguaje simbólico y en lenguaje absoluto, en varios rótulos consecutivos de una cinta magnética que tiene etiquetas modelo IBM. Se graban los programas en FORTRAN de nombres MULTIPL y FOCAL en los rótulos 8 y 9; el programa en ASSEMBLER de nombre RUTAS en el rótulo 10; los programas en absoluto de FORTRAN y de ASSEMBLER de nombres ABSOKAL y ABALONG en los rótulos 11 y 12. La cinta lleva el número de carrete 38523. Los programas en lenguaje simbólico se cargan con factor de bloqueo igual a 40; los programas en lenguaje absoluto se cargan desbloqueados.

```
//..... JOB ....
//PASO1 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSIN DD DUMMY
//SYSUT2 DD DSN=MULTIPL,UNIT=TAPE,VOL=SER=T38523,LABEL=(8,SL),
// DISP=(NEW,PASS),DCB=(BLKSIZE=3200,LRECL=80,RECFM=FB)
//SYSUT1 DD *
  Aquí tarjetas perforadas del programa MULTIPL
/*
//PASO2 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSIN DD DUMMY
//SYSUT2 DD DSN=FOCAL,UNIT=TAPE,VOL=REF=*.PASO1.SYSUT2,LABEL=(9,SL)
//SYSUT1 DD *
  Aquí tarjetas perforadas del programa FOCAL
/*
//PASO3 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSIN DD DUMMY
//SYSUT2 DD DSN=RUTAS,UNIT=TAPE,VOL=REF=*.PASO1.SYSUT2,LABEL=(10,SL)
//SYSUT1 DD *
  Aquí tarjetas perforadas del programa RUTAS
/*
//PASO4 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSIN DD DUMMY
//SYSUT2 DD DSN=ABSOKAL,UNIT=TAPE,VOL=SER=T38523,LABEL=(11,SL),
// DISP=(NEW,PASS),DCB=BLKSIZE=80
//SYSUT1 DD *
  Aquí tarjetas perforadas del programa ABSOKAL
/*
```

```
//PAS05 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSIN DD DUMMY
//SYSUT2 DD DSN=ABALONG,UNIT=TAPE,VOL=REF=*.PAS04.SYSUT2,LABEL=(12,SL),
// DISP=(NEW,KEEP)
//SYSUT1 DD *
  Aquí tarjetas perforadas del programa ABALONG
/*
//
```

Modelo 340. Grabación de programas que se presentan en tarjetas perforadas en lenguaje simbólico y en lenguaje absoluto, en varios rótulos consecutivos de una cinta magnética que no tiene etiquetas de cabecera. Se graban dos programas en absoluto de FORTRAN y uno en absoluto de ASSEMBLER en los rótulos 25, 26 y 27; un programa en ASSEMBLER en el rótulo 28 y un programa en FORTRAN en el rótulo 29. La cinta lleva el número de carrete 17604. Los programas en lenguaje absoluto se cargan desbloqueados; los programas en lenguaje simbólico se cargan con factor de bloqueo igual a 30.

```
//..... JOB ....
//PAS01 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSIN DD DUMMY
//SYSUT2 DD UNIT=TAPE,VOL=SER=T17604,LABEL=(25,NL),DISP=(NEW,PASS),
// DCB=BLKSIZE=80
//SYSUT1 DD *
  Aquí tarjetas perforadas del primer programa en lenguaje absoluto
/*
//PAS02 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSIN DD DUMMY
//SYSUT2 DD UNIT=TAPE,VOL=REF=*.PAS01.SYSUT2,LABEL=(26,NL)
//SYSUT1 DD *
  Aquí tarjetas perforadas del segundo programa en lenguaje absoluto
/*
//PAS03 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSIN DD DUMMY
//SYSUT2 DD UNIT=TAPE,VOL=REF=*.PAS01.SYSUT2,LABEL=(27,NL)
//SYSUT1 DD *
  Aquí tarjetas perforadas del tercer programa en lenguaje absoluto
/*
//PAS04 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSIN DD DUMMY
//SYSUT2 DD UNIT=TAPE,VOL=SER=T17604,LABEL=(28,NL),DISP=(NEW,PASS),
// DCB=(BLKSIZE=2400,LRECL=80,RECFM=FB)
//SYSUT1 DD *
```

```

Aquí tarjetas perforadas del primer programa en lenguaje simbólico
/*
//PAS05 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSIN DD DUMMY
//SYSUT2 DD UNIT=TAPE,VOL=REF=*.PAS04.SYSUT2,LABEL=(2,NL)
//SYSUT1 DD *
Aquí tarjetas perforadas del segundo programa en lenguaje simbólico
/*
//

```

Modelo 341 . Grabación de programas que se presentan en tarjetas perforadas en lenguaje simbólico y en lenguaje absoluto, en varios rotulos consecutivos de una cinta magnética que tiene etiquetas de cabecera desconocidas o deterioradas que se desean borrar. Se graban dos programas en FORTRAN en los rótulos 1 y 2; tres programas en absoluto de ASSEMBLER y de FORTRAN en los rótulos 3, 4 y 5. La cinta lleva el número de carrete 20578. Los programas en lenguaje simbólico se cargan con factor de bloqueo igual a 25; los programas en lenguaje absoluto se cargan desbloqueados.

```

//..... JOB ....
//PAS01 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSIN DD DUMMY
//SYSUT2 DD UNIT=TAPE,VOL=SER=T20578,LABEL=(1,BLP),DISP=(NEW,PASS),
// DCB=(BLKSIZE=2000,LRECL=80,RECFM=FB)
//SYSUT1 DD *
Aquí tarjetas perforadas del primer programa en lenguaje simbólico
/*
//PAS02 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSIN DD DUMMY
//SYSUT2 DD UNIT=TAPE,VOL=REF=*.PAS01.SYSUT2,LABEL=(2,NL)
//SYSUT1 DD *
Aquí tarjetas perforadas del segundo programa en lenguaje simbólico
/*
//PAS03 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSIN DD DUMMY
//SYSUT2 DD UNIT=TAPE,VOL=SER=T20578,LABEL=(3,NL),DISP=(NEW,PASS),
// DCB=BLKSIZE=80
//SYSUT1 DD *
Aquí tarjetas perforadas del primer programa en lenguaje absoluto
/*
//PAS04 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSIN DD DUMMY
//SYSUT2 DD UNIT=TAPE,VOL=REF=*.PAS03.SYSUT2,LABEL=(4,NL)
//SYSUT1 DD *

```

```
Aquí tarjetas perforadas del segundo programa en lenguaje absoluto
/*
//PAS05 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSIN DD DUMMY
//SYSUT2 DD UNIT=TAPE,VOL=REF=*.PAS03.SYSUT2,LABEL=(5,NL)
//SYSUT1 DD *
Aquí tarjetas perforadas del tercer programa en lenguaje absoluto
/*
//
```

NOTA: El parámetro LABEL=(1,BLP) codificado en el PAS01 borra las etiquetas de cabecera si las hubiera y convierte la cinta en NO LABEL (NL).

PUNTO 4.

Grabación y listado de etiquetas modelo IBM.

Modelo 342 . Inicialización de una cinta magnética con etiquetas modelo IBM, que no será utilizada acto seguido para grabar (será desmontada).

```
//..... JOB ....
//PASO1 EXEC PGM=IEHINITT
//SYSPRINT DD SYSOUT=A
//LABEL DD DCB=DEN=3,UNIT=(2400,1,DEFER)
//SYSIN DD *
LABEL INITT SER=T60008,OWNER='G.LOPEZ',NUMTAPE=1
/*
//
```

NOTA: La grabación de las etiquetas puede ser realizada indistintamente: en una cinta nueva; en una cinta que tiene etiquetas desconocidas que se desean sustituir; en una cinta cuyas etiquetas están deterioradas en su totalidad o en parte; en una cinta que tiene etiquetas modelo IBM en buen estado pero que se desean cambiar. El parámetro NUMTAPE se refiere al número de cintas que va a inicializar el programa.

Modelo 343 . Inicialización de una cinta magnética con etiquetas modelo IBM, que será utilizada acto seguido para grabar.

```
//..... JOB ....
//PASO1 EXEC PGM=IEHINITT
//SYSPRINT DD SYSOUT=A
//LABEL DD DCB=DEN=3,UNIT=(2400,1,DEFER)
//SYSIN DD *
LABEL INITT SER=T60009,OWNER='G.R.GARCIA',NUMTAPE=1,DISP=REWIND
/*
//
```

NOTA: La grabación de las etiquetas puede ser realizada indistintamente: en una cinta nueva; en una cinta que tiene etiquetas desconocidas que se desean sustituir; en una cinta cuyas etiquetas están deterioradas en su totalidad o en parte; en una cinta que tiene etiquetas modelo IBM en buen estado pero que se desean cambiar. El parámetro NUMTAPE se refiere al número de cintas que va a inicializar el programa.

Modelo 344 . Listado de las etiquetas de cabecera del rótulo 1 de una cinta magnética que tiene etiquetas modelo IBM y lleva el número de carrete 60109.

```
//..... JOB ....
//PAS01 EXEC PGM=IEBPTPCH
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD UNIT=TAPE,LABEL=(1,BLP),DISP=(OLD,KEEP),
// VOL=SER=T60109,DCB=BLKSIZE=80
//SYSUT2 DD SYSOUT=A
//SYSIN DD *
PRINT MAXFLDS=1
RECORD FIELD=(80)
/*
//
```

Modelo 345 . Listado de las etiquetas de cabecera de rótulos subsiguientes del 1 de una cinta magnética que tiene etiquetas modelo IBM. Se desean los listados de las etiquetas de los rótulos 2, 3 y 7 de la cinta que lleva el número de carrete 75011.

```
//..... JOB ....
//PAS01 EXEC PGM=IEBPTPCH
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD UNIT=TAPE,VOL=SER=T75011,LABEL=(4,BLP),
// DISP=(OLD,PASS),DCB=BLKSIZE=80
//SYSUT2 DD SYSOUT=A
//SYSIN DD *
PRINT MAXFLDS =1
RECORD FIELD=(80)
/*
//PAS02 EXEC PGM=IEBPTPCH
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD UNIT=TAPE,VOL=SER=T75011,LABEL=(7,BLP),
// DISP=(OLD,PASS),DCB=BLKSIZE=80
//SYSUT2 DD SYSOUT=A
//SYSIN DD *
PRINT MAXFLDS=1
RECORD FIELD=(80)
/*
//PAS03 EXEC PGM=IEBPTPCH
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD UNIT=TAPE,VOL=SER=75011,LABEL=(19,BLP),
// DISP=(OLD,KEEP),DCB=BLKSIZE=80
//SYSUT2 DD SYSOUT=A
//SYSIN DD *
PRINT MAXFLDS=1
RECORD FIELD=(80)
/*
//
```

PUNTO 5.

Transferencia de archivos de cinta a otra cinta; de cinta a la misma cinta a través de disco y de disco a disco a través de cinta.

Modelo 346 . Transferencia de los archivos de varios rótulos consecutivos o no, de una cinta magnética a otra nueva, con supresión de archivos, cambio de nombre, de densidad de grabación y de factor de bloqueo. La cinta original lleva el número de carrete 77178, fué grabada con densidad 6250 bits por pulgada y factor de bloqueo 44; la cinta nueva lleva el número de carrete 92800, será grabada con densidad 1600 bits por pulgada y factor de bloqueo 30; ambas cintas tienen etiquetas modelo IBM. Los archivos de nombres BETA6, ARCCS, HIPERBOL, CATALIS y URANIO, grabados en los rótulos 6, 15, 16, 24 y 30 de la primera cinta, se transfieren a los rótulos 1, 2, 3, 4 y 5 de la nueva cinta conservando sus nombres, salvo el archivo CATALIS que se denominará NUEVCTLS. Los archivos correspondientes a los rótulos 1 a 5, 7 a 14, 17 a 23 y 25 a 29 de la cinta original resultan suprimidos en la nueva cinta.

```
//..... JOB ....
//PAS01 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSIN DD DUMMY
//SYSUT1 DD DSN=BETA6,UNIT=TAPE,VOL=SER=T77178,LABEL=(6,SL),
// DISP=(OLD,PASS)
//SYSUT2 DD DSN=BETA6,UNIT=TAPE,VOL=SER=T92800,LABEL=(1,SL),
// DISP=(NEW,PASS),DCB=(BLKSIZE=2400,LRECL=80,RECFM=FB,DEN=3)
/*
//PAS02 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSIN DD DUMMY
//SYSUT1 DD DSN=ARCCS,UNIT=TAPE,VOL=REF=*.PAS01.SYSUT1,LABEL=(15,SL)
//SYSUT2 DD DSN=ARCCS,UNIT=TAPE,VOL=REF=*.PAS01.SYSUT1,LABEL=(2,SL)
/*
//PAS03 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSIN DD DUMMY
//SYSUT1 DD DSN=HIPERBOL,UNIT=TAPE,VOL=REF=*.PAS01.SYSUT1,LABEL=(16,SL)
//SYSUT2 DD DSN=HIPERBOL,UNIT=TAPE,VOL=REF=*.PAS01.SYSUT2,LABEL=(3,SL)
/*
//PAS04 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSIN DD DUMMY
//SYSUT1 DD DSN=CATALIS,UNIT=TAPE,VOL=REF=*.PAS01.SYSUT1,LABEL=(24,SL)
//SYSUT2 DD DSN=NUEVCTLS,UNIT=TAPE,VOL=REF=*.PAS01.SYSUT2,LABEL=(4,SL)
```

```

/*
//PAS05 EXEC PGM=IEBGENER
//SYSPRINT DD SYSCUT=A
//SYSIN DD DUMMY
//SYSUT1 DD DSN=URANIO,UNIT=TAPE,VOL=REF=*.PAS01.SYSUT1,LABEL=(30,SL),
//  DISP=(OLD,KEEP)
//SYSUT2 DD DSN=URANIO,UNIT=TAPE,VOL=REF=*.PAS01.SYSUT2,LABEL=(5,SL),
//  DISP=(NEW,KEEP)
/*
//

```

Modelo 347 . Transferencia de los archivos de varios r tulos consecutivos o no, de una cinta magn tica a un disco auxiliar y de  ste a la misma cinta, con el objeto de suprimir archivos fuera de uso, cambiar nombres, variar la densidad de grabaci n y el factor de bloqueo. La cinta, que lleva el n mero de carrete 20987 y tiene etiquetas modelo IBM, fu  grabada con densidad DEN=4 y longitud m xima de bloque BLKSIZE=2400; la nueva grabaci n se har  con densidad DEN=3 y longitud de bloque BLKSIZE=3520. Los archivos de nombres SOLOM, TACO y FACTOR residentes en los r tulos 3, 21 y 28 se grabar n en la misma cinta en los r tulos 1, 2 y 3, con lo cual los archivos correspondientes a los r tulos 1 y 2, 4 a 20 y 22 a 27 resultan suprimidos. El archivo TACO tomar  el nombre COTA53. El problema se puede resolver en seis pasos: en los tres primeros (PAS01, PAS02 y PAS03) se realiza la transferencia de cinta a disco auxiliar y en los tres pasos restantes (PAS04, PAS05 y PAS06) se realiza la transferencia de disco a la misma cinta.

```

//..... JOB ....
//PAS01 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSIN DD DUMMY
//SYSUT1 DD DSN=SOLOM,UNIT=TAPE,VOL=SER=T20987,LABEL=(3,SL),
//  DISP=OLD
//SYSUT2 DD DSN=SOLODISK,UNIT=WORKDISK,DISP=(NEW,PASS),
//  DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB),SPACE=(CYL,(1,1))
/*
//PAS02 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSIN DD DUMMY
//SYSUT1 DD DSN=TACO,UNIT=TAPE,VOL=SER=T20987,LABEL=(21,SL),DISP=OLD
//SYSUT2 DD DSN=TACODISK,VOL=REF=*.PAS01.SYSUT2,DISP=(NEW,PASS),
//  SPACE=(CYL,(1,1))
/*
//PAS03 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSIN DD DUMMY

```

```

//SYSUT1 DD DSN=FACTOR,UNIT=TAPE,VOL=SER=T20987,LABEL=(28,SL),
// DISP=OLD
//SYSUT2 DD DSN=FACTDISK,VOL=REF=*.PAS01.SYSUT2,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1))
/*
//PAS04 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSIN DD DUMMY
//SYSUT1 DD DSN=SOLODISK,UNIT=WORKDISK,DISP=(OLD,DELETE,KEEP)
//SYSUT2 DD DSN=SOLOM,UNIT=TAPE,VOL=SER=T20987,LABEL=(1,SL),
// DISP=(NEW,PASS),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB,DEN=3)
/*
//PAS05 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSIN DD DUMMY
//SYSUT1 DD DSN=TACODISK,UNIT=WORKDISK,DISP=(OLD,DELETE,KEEP)
//SYSUT2 DD DSN=COTA53,UNIT=TAPE,VOL=REF=*.PAS04.SYSUT2,LABEL=(2,SL)
/*
//PAS06 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSIN DD DUMMY
//SYSUT1 DD DSN=FACTDISK,UNIT=WORKDISK,DISP=(OLD,DELETE,KEEP)
//SYSUT2 DD DSN=FACTOR,UNIT=TAPE,VOL=REF=*.PAS04.SYSUT2,LABEL=(3,SL),
// DISP=(NEW,KEEP)
/*
//

```

NOTA: Se menciona el DCB con el cual se grabará el disco auxiliar y también el que se utilizará para grabar la cinta.

Modelo 348 . Transferencia total de una biblioteca particionada y catalogada de programas en lenguaje simbólico o de módulos de carga indistintamente, desde la unidad de discos en que reside a la unidad que le está destinada en otro centro de cómputos. Se excluirán los miembros que estén fuera de uso, utilizando una cinta magnética como soporte físico de la información, la cual será grabada con la densidad DEN=3. Se desea trasladar la biblioteca de nombre FISICA.CUP a una biblioteca similar ya existente, de nombre FISICA.CDEC. Los miembros que se van a excluir al hacer la transferencia llevan los nombres BARRA, DOBLESIG, ALAN, MARMORAL y TENSION. El problema se puede resolver en dos etapas: en la primera etapa (Programa A) el utilitario IEBCOPY transfiere la biblioteca FISICA.CUP con su directorio y excluyendo los miembros mencionados en la sentencia EXCLUDE, con el nombre FISICA.CINTA y densidad 1600 bits por pulgada, al rótulo 1 de la cinta

que lleva el número de carrete 06963 y tiene etiquetas modelo IBM. En la segunda etapa (Programa B) el utilitario IEBCOPY transfiere la biblioteca FISICA.CINTA de la cinta a la biblioteca FISICA.CDEC.

Programa A. (Para el primer centro de cómputos)

```
//..... JOB ....
//PASO1 EXEC PGM=IEBCOPY
//SYSPRINT DD SYSOUT=A
//SYSUT3 DD UNIT=SYSDA,SPACE=(CYL,(5,2))
//SYSUT4 DD UNIT=SYSDA,SPACE=(CYL,(5,2))
//ENTRADA DD DSN=FISICA.CUP,DISP=SHR
//SALIDA DD DSN=FISICA.CINTA,UNIT=TAPE,VOL=SER=T06968,
//  DCB=DEN=3,DISP=(NEW,KEEP)
//SYSIN DD *
  C O=SALIDA,I=ENTRADA
  EXCLUDE MEMBER=BARRA,DOBLESIG,ALAN,MARMORAL,TENSION
/*
//
```

Programa B. (Para el segundo centro de cómputos)

```
//..... JOB ....
//PASO1 EXEC PGM=IEBCOPY
//SYSPRINT DD SYSOUT=A
//SYSUT3 DD UNIT=SYSDA,SPACE=(CYL,(5,2))
//SYSUT4 DD UNIT=SYSDA,SPACE=(CYL,(5,2))
//ENTRADA DD DSN=FISICA.CINTA,UNIT=TAPE,VOL=SER=T06968,DISP=(OLD,KEEP)
//SALIDA DD DSN=FISICA.CDEC,DISP=OLD
//SYSIN DD *
  C O=SALIDA,I=ENTRADA
/*
//
```

Modelo 349 . Transferencia de varios miembros de una biblioteca particionada y catalogada de programas en lenguaje simbólico o de módulos de carga, indistintamente, desde la unidad de discos en que reside a la unidad que le está destinada en otro centro de cómputos. Se seleccionarán solamente los miembros de nombres FIZZ, FOCO, IMAN y POLO, utilizando una cinta magnética como soporte físico de la información, la cual será grabada con la densidad DEN=3. La biblioteca anterior se llama QUIMICQ.CUP y la nueva, ya existente y similar a aquélla, se llama QUIMICA.CDEC. Al miembro IMAN se le cambiará el nombre por el de MAGNET al trasladarlo. La cinta lleva el número de carrete 88169 y tiene etiquetas modelo IBM. El problema se puede resolver en dos etapas: en la primera etapa (Programa A) el

utilitario IEBCOPY selecciona los miembros FIZZ, FOCO, IMAN y POLO de la biblioteca QUIMICA.CUP, le cambia el nombre al miembro IMAN y crea una biblioteca temporaria particionada de nombre QUIMICA.CINTA que se graba en el rótulo 19 de la cinta; en la segunda etapa (Programa B) el mismo utilitario transfiere la biblioteca temporaria de la cinta a la biblioteca QUIMICA.CDEC.

Programa A. (Para el primer centro de cómputos)

```
//..... JOB ....
//PAS01 EXEC PGM=IEBCOPY
//SYSPRINT DD SYSOUT=A
//SYSUT3 DD UNIT=SYSDA,SPACE=(CYL,(2,1))
//SYSUT4 DD UNIT=SYSDA,SPACE=(CYL,(2,1))
//ENTRADA DD DSN=QUIMICA.CUP,DISP=SHR
//SALIDA DD DSN=QUIMICA.CINTA,UNIT=TAPE,VOL=SER=T88169,LABEL=(19,SL),
// DISP=(NEW,KEEP),DCB=DEN=3
//SYSIN DD *
C O=SALIDA,I=ENTRADA
S M=((FIZZ,,R),(FOCO,,R),(IMAN,MAGNET,R),(POLO,,R))
/*
//
```

Programa B. (Para el segundo centro de cómputos)

```
//..... JOB ....
//PAS01 EXEC PGM=IEBCOPY
//SYSPRINT DD SYSOUT=A
//SYSUT3 DD UNIT=SYSDA,SPACE=(CYL,(2,1))
//SYSUT4 DD UNIT=SYSDA,SPACE=(CYL,(2,1))
//ENTRADA DD DSN=QUIMICA.CINTA,UNIT=TAPE,VOL=SER=T88169,LABEL=(19,SL),
// DISP=(OLD,KEEP)
//SALIDA DD DSN=QUIMICA.CDEC,DISP=OLD
//SYSIN DD *
C O=SALIDA,I=ENTRADA
/*
//
```

PUNTO 6.

Transferencia selectiva de programas, rutinas y funciones grabadas secuencialmente en el mismo rótulo de una cinta magnética, mediante la identificación de tiras de caracteres, a otros soportes de información.

Modelo 350. Transferencia de un programa en lenguaje simbólico que reside, junto con otros, en el mismo rótulo de una cinta magnética y está marcado en cada uno de sus registros con una tira de caracteres a una biblioteca particionada de programas en lenguaje simbólico, de nombre SIMBOLIC.CNEA, como un miembro independiente. En el rótulo 32 de una cinta que lleva el número de carrete 36215 y tiene etiquetas modelo IBM, está grabado, con el nombre PGMNUPH, un conjunto de 80 programas, rutinas y funciones, cada uno de los cuales está marcado en la columna 73 y siguientes con una abreviatura de su nombre. Se desea transferir a la biblioteca SIMBOLIC.CNEA solamente el programa SPECTRA identificando la tira de caracteres SPTR.

```
//..... JOB ....
//PAS01 EXEC PROC=OSGENER
//SYSUT1 DD DSN=PGMNUPH,UNIT=TAPE,VOL=SER=T36215,LABEL=(32,SL),
// DISP=(OLD,KEEP)
//SYSUT2 DD DSN=SIMBOLIC.CNEA(SPECTRA),DISP=OLD
//SYSIN DD *
RCIN 73,,'SPTR'
COPY
/*
//
```

Modelo 351. Transferencia de varias rutinas en lenguaje simbólico que residen, junto con otras, en el mismo rótulo de una cinta magnética y están marcadas en cada uno de sus registros con una tira de caracteres, a una biblioteca particionada de programas en lenguaje simbólico, de nombre SIMBOLIC.CNEA, como un miembro independiente. En el rótulo 7 de una cinta que lleva el número de carrete 36222 y tiene etiquetas modelo IBM, está grabado, con el nombre PROGRAMS, un conjunto de 65 programas, rutinas y funciones, cada uno de los cuales está marcado en la columna 73 y siguientes con una abreviatura de su nombre. Se desean transferir a la

biblioteca de simbólicos, con el nombre SEISRUT, solamente las rutinas SAVEPOLO, OSCILAT, CUP1276, DIAGFASE, POTENCIA y CUADRADO identificando, respectivamente, las tiras de caracteres SAVE, OSC, CUP, DIAG, POT y CUAD.

```
//..... JOB ....
//PASO1 EXEC PROC=OSGENER
//SYSUT1 DD DSN=PROGRAMS,UNIT=TAPE,VOL=SER=T36222,LABEL=(7,SL),
// DISP=(OLD,KEEP)
//SYSUT2 DD DSN=SIMBOLIC.CNEA(SEISRUT),DISP=OLD
//SYSIN DE *
RCIN 73,, 'SAVE'
RCIN 73,, 'OSC'
RCIN 73,, 'CUP'
RCIN 73,, 'DIAG'
RCIN 73,, 'POT'
RCIN 73,, 'CUAD'
COPY
/*
//
```

Modelo 352 . Transferencia de un programa, una rutina y una función en lenguaje simbólico que residen junto con otros, en el mismo rótulo de una cinta magnética y están marcados en cada uno de sus registros con una tira de caracteres, a una biblioteca particionada de programas en lenguaje simbólico, de nombre SIMBOLIC.CNEA, como miembros independientes. En el rótulo 14 de una cinta que lleva el número de carrete 03311 y tiene etiquetas modelo IBM, está grabado, con el nombre ARCHIVO3, un conjunto de 90 programas, rutinas y funciones, cada uno de los cuales está marcado en la columna 73 y siguientes con una abreviatura de su nombre. Se desean transferir a la biblioteca de simbólicos: el programa PALMUD, la rutina DENSIDAD y la función RAICES5 identificando, respectivamente, las tiras de caracteres PAL, DEN y RAI. El problema se puede resolver en tres pasos a razón de un paso por cada miembro independiente de la biblioteca.

```
//..... JOB ....
//PASO1 EXEC PROC=OSGENER
//SYSUT1 DD DSN=ARCHIVO3,UNIT=TAPE,VOL=SER=T03311,LABEL=(14,SL),
// DISP=(OLD,KEEP)
//SYSUT2 DD DSN=SIMBOLIC.CNEA(PALMUD),DISP=OLD
//SYSIN DD *
RCIN 73,, 'PAL'
COPY
/*
```

```

//PASO2 EXEC PROC=OSGENER
//SYSUT1 DD DSN=ARCHIVO3,UNIT=TAPE,VOL=SER=T03311,LABEL=(14,SL),
// DISP=(OLD,KEEP)
//SYSUT2 DD DSN=SIMBOLIC.CNEA(DENSIDAD),DISP=OLD
//SYSIN DD *
RCIN 73,, 'DEN'
COPY
/*
//PASO3 EXEC PROC=OSGENER
//SYSUT1 DD DSN=ARCHIVO3,UNIT=TAPE,VOL=SER=T03311,LABEL=(14,SL),
// DISP=(OLD,KEEP)
//SYSUT2 DD DSN=SIMBOLIC.CNEA(RAICES5),DISP=OLD
//SYSIN DD *
RCIN 73,, 'RAI'
COPY
/*
//

```

Modelo 353 . Transferencia de un programa en lenguaje simbólico FORTRAN que reside, junto con otros, en el mismo rótulo de una cinta magnética y está marcado en cada uno de sus registros con una tira de caracteres, a una biblioteca particionada de módulos de carga, de nombre REALES.CNEA, como un módulo independiente. En el rótulo 21 de una cinta que lleva el número de carrete 33066 y tiene etiquetas modelo IBM, está grabado, con el nombre EVANS, un conjunto de 60 programas, rutinas y funciones cada uno de los cuales está marcado en la columna 73 y siguientes con una abreviatura de su nombre. Se desea transferir a la biblioteca REALES.CNEA solamente el programa BOHR5521, identificando la tira de caracteres BHR1. El problema se puede resolver en dos pasos: en el PASO1 el procedimiento OSGENER selecciona el programa BOHR5521 y lo graba en una biblioteca temporaria particionada que se crea en ese momento y reside en un disco auxiliar; en el PASO2 el procedimiento FTG1CL compila y vincula el programa cargando el módulo resultante en la biblioteca REALES.CNEA.

```

//..... JOB ....
//PASO1 EXEC PROC=OSGENER
//SYSUT1 DD DSN=EVANS,UNIT=TAPE,VOL=SER=T33066,LABEL=(21,SL),
// DISP=(OLD,KEEP)
//SYSUT2 DD DSN=TEMPOR.ADR(BOHR5521),UNIT=WORKDISK,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1,1)),DCB=(BLKSIZE=3200,LRECL=80,RECFM=FB)
//SYSIN DD *
RCIN 73,, 'BHR1'
COPY
/*
//PASO2 EXEC PROC=FTG1CL

```

```
//FORT.SYSIN DD DSN=TEMPOR.ADR(BOHR5521),UNIT=WORKDISK,
// DISP=(OLD,DELETE)
//LKED.SYSLMOD DD DSN=REALES.CNEA(BOHR5521),SPACE=,DISP=OLD
/*
//
```

NOTA: Se menciona el DCB con el cual se grabará el disco auxiliar.

Modelo 354 . Transferencia de un programa en lenguaje simbólico ASSEMBLER que reside, junto con otros, en el mismo rótulo de una cinta magnética y está marcado en cada uno de sus registros con una tira de caracteres, a una biblioteca de módulos de carga, de nombre REALES.CNEA, como un módulo independiente. En el rótulo 33 de una cinta que lleva el número de carrete 42198 y tiene etiquetas modelo IBM, está grabado, con el nombre PGMNUCL, un conjunto de 100 programas, rutinas y funciones cada uno de los cuales está marcado en la columna 73 y siguientes con una abreviatura de su nombre. Se desea transferir a la biblioteca de módulos, solamente el programa ASSUNA, identificando la tira de caracteres ASS. El problema se puede resolver en dos pasos: en el PASO1 el procedimiento OSGENER selecciona el programa ASSUNA y lo graba en una biblioteca temporaria particionada que se crea en ese momento y reside en un disco auxiliar; en el PASO2 el procedimiento ASMFCL compila y vincula el programa cargando el módulo resultante en la biblioteca REALES.CNEA.

```
//..... JOB ....
//PASO1 EXEC PROC=OSGENER
//SYSUT1 DD DSN=PGMNUCL,UNIT=TAPE,VOL=SER=T42198,LABEL=(33,SL),
// DISP=(OLD,KEEP)
//SYSUT2 DD DSN=TEMPOR.ADR(ASSUNA),UNIT=WORKDISK,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1,1)),DCB=(BLKSIZE=2400,LRECL=80,RECFM=FB)
//SYSIN DD *
RCIN 73,,'ASS'
COPY
/*
//PASO2 EXEC PROC=ASMFCL
//ASM.SYSIN DD DSN=TEMPOR.ADR(ASSUNA),UNIT=WORKDISK,
// DISP=(OLD,DELETE)
//LKED.SYSLMOD DD DSN=REALES.CNEA(ASSUNA),SPACE=,DISP=OLD
/*
//
```

NOTA: Se menciona el DCB con el cual se grabará el disco auxiliar.

Modelo 355 . Transferencia de un programa en lenguaje simbólico PL/1 que reside, junto con otros, en el mismo rótulo de una cinta magnética y está marcado en cada uno de sus registros con una tira de caracteres, a una biblioteca particionada de módulos de carga, de nombre REALES.CNEA, como un módulo independiente. En el rótulo 8 de una cinta que lleva el número de carrete 06163 y tiene etiquetas modelo IBM, está grabado, con el nombre PGM27MEV, un conjunto de 200 programas, rutinas y funciones cada uno de los cuales está marcado en la columna 73 y siguientes con una abreviatura de su nombre. Se desea transferir a la biblioteca de módulos solamente el programa DRPLCDAT, identificando la tira de caracteres DRDT. El problema se puede resolver en dos pasos: en el PASO1 el procedimiento OSGENER selecciona el programa DRPLCDAT y lo graba en una biblioteca temporaria particionada que se crea en ese momento y reside en un disco auxiliar; en el PASO2 el procedimiento PLIXCL compila y vincula el programa cargando el módulo resultante en la biblioteca REALES.CNEA.

```
//..... JOB ....
//PASO1 EXEC PROC=OSGENER
//SYSUT1 DD DSN=PGM27MEV,UNIT=TAPE,VOL=SER=T06163,LABEL=(8,SL),
// DISP=(OLD,KEEP)
//SYSUT2 DD DSN=TEMPOR.ADR(DRPLCDAT),UNIT=WORKDISK,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1,1)),DCB=(BLKSIZE=3200,LRECL=80,RECFM=FB)
//SYSIN DD *
RCIN 73,, 'DRDT'
COPY
/*
//PASO2 EXEC PROC=PLIXCL,PARM.PLI='CS(60)'
//PLI.SYSIN DD DSN=TEMPOR.ADR(DRPLCDAT),UNIT=WORKDISK,
// DISP=(OLD,DELETE)
//LKED.SYSLMOD DD DSN=REALES.CNEA(DRPLCDAT),SPACE=,DISP=OLD
/*
//
```

NOTA: Se menciona el DCB con el cual se grabará el disco auxiliar.

Modelo 356 . Transferencia de varias rutinas en lenguaje simbólico que residen, junto con otras, en el mismo rótulo de una cinta magnética y están marcadas en cada uno de sus registros con una tira de caracteres, a rótulos consecutivos de otra cinta. En el rótulo 18 de una cinta que lleva el número de carrete 12397 es-

tá grabado, con el nombre FILE25, un conjunto de 100 programas, rutinas y funciones, cada uno de los cuales está marcado en la columna 73 y siguientes con una abreviatura de su nombre. Se desean transferir a los rótulos 20, 21 y 22, respectivamente, de la cinta que lleva el número de carrete 11400, las rutinas PARTICLE, EVENTS y CHARGES, identificando las tiras de caracteres PAR, EV y CHAR con que están marcadas. Ambas cintas tienen etiquetas modelo IBM. El problema se puede resolver en tres pasos a razón de un paso por cada rótulo de la nueva cinta.

```
//..... JOB ....
//PAS01 EXEC PROC=OSGENER
//SYSUT1 DD DSN=FILE25,UNIT=TAPE,VOL=SER=T12397,LABEL=(18,SL),
//  DISP=(OLD,KEEP)
//SYSUT2 DD DSN=PARTICLE,UNIT=TAPE,VOL=SER=T12400,LABEL=(20,SL),
//  DISP=(NEW,PASS),DCB=(BLKSIZE=1600,LRECL=80,RECFM=FB)
//SYSIN DD *
RCIN 73,, 'PAR'
COPY
/*
//PAS02 EXEC PROC=OSGENER
//SYSUT1 DD DSN=FILE25,UNIT=TAPE,VOL=SER=T12397,LABEL=(18,SL),
//  DISP=(OLD,KEEP)
//SYSUT2 DD DSN=EVENTS,UNIT=TAPE,VOL=SER=T12400,LABEL=(21,SL),
//  DISP=(NEW,PASS),DCB=(BLKSIZE=1600,LRECL=80,RECFM=FB)
//SYSIN DD *
RCIN 73,, 'EV'
COPY
/*
//PAS03 EXEC PROC=OSGENER
//SYSUT1 DD DSN=FILE25,UNIT=TAPE,VOL=SER=T12397,LABEL=(18,SL),
//  DISP=(OLD,KEEP)
//SYSUT2 DD DSN=CHARGES,UNIT=TAPE,VOL=SER=T12400,LABEL=(22,SL),
//  DISP=(NEW,KEEP),DCB=(BLKSIZE=1600,LRECL=80,RECFM=FB)
//SYSIN DD *
RCIN 73,, 'CHAR'
COPY
/*
//
```

NOTA: Se menciona el DCB con el cual se grabará la cinta.

Modelo 355 . Transferencia de un programa en lenguaje simbólico PL/1 que reside, junto con otros, en el mismo rótulo de una cinta magnética y está marcado en cada uno de sus registros con una tira de caracteres, a una biblioteca particionada de módulos de carga, de nombre REALES.CNEA, como un módulo independiente. En el rótulo 8 de una cinta que lleva el número de carrete 06163 y tiene etiquetas modelo IBM, está grabado, con el nombre PGM27MEV, un conjunto de 200 programas, rutinas y funciones cada uno de los cuales está marcado en la columna 73 y siguientes con una abreviatura de su nombre. Se desea transferir a la biblioteca de módulos solamente el programa DRPLCDAT, identificando la tira de caracteres DRDT. El problema se puede resolver en dos pasos: en el PASO1 el procedimiento OSGENER selecciona el programa DRPLCDAT y lo graba en una biblioteca temporaria particionada que se crea en ese momento y reside en un disco auxiliar; en el PASO2 el procedimiento PLIXCL compila y vincula el programa cargando el módulo resultante en la biblioteca REALES.CNEA.

```
//..... JOB ....
//PASO1 EXEC PROC=OSGENER
//SYSUT1 DD DSN=PGM27MEV,UNIT=TAPE,VOL=SER=T06163,LABEL=(8,SL),
// DISP=(OLD,KEEP)
//SYSUT2 DD DSN=TEMPOR.ADR(DRPLCDAT),UNIT=WORKDISK,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1,1)),DCB=(BLKSIZE=3200,LRECL=80,RECFM=FB)
//SYSIN DD *
RCIN 73,, 'DRDT'
COPY
/*
//PASO2 EXEC PROC=PLIXCL,PARM.PLI='CS(60)'
//PLI.SYSIN DD DSN=TEMPOR.ADR(DRPLCDAT),UNIT=WORKDISK,
// DISP=(OLD,DELETE)
//LKED.SYSLMOD DD DSN=REALES.CNEA(DRPLCDAT),SPACE=,DISP=OLD
/*
//
```

NOTA: Se menciona el DCB con el cual se grabará el disco auxiliar.

Modelo 356 . Transferencia de varias rutinas en lenguaje simbólico que residen, junto con otras, en el mismo rótulo de una cinta magnética y están marcadas en cada uno de sus registros con una tira de caracteres, a rótulos consecutivos de otra cinta. En el rótulo 18 de una cinta que lleva el número de carrete 12397 es-

tá grabado, con el nombre FILE25, un conjunto de 100 programas, rutinas y funciones, cada uno de los cuales está marcado en la columna 73 y siguientes con una abreviatura de su nombre. Se desean transferir a los rótulos 20, 21 y 22, respectivamente, de la cinta que lleva el número de carrete 12400, las rutinas PARTICLE, EVENTS y CHARGES, identificando las tiras de caracteres PAR, EV y CHAR con que están marcadas. Ambas cintas tienen etiquetas modelo IBM. El problema se puede resolver en tres pasos a razón de un paso por cada rótulo de la nueva cinta.

```
//..... JOB ....
//PAS01 EXEC PROC=OSGENER
//SYSUT1 DD DSN=FILE25,UNIT=TAPE,VOL=SER=T12397,LABEL=(18,SL),
//  DISP=(OLD,KEEP)
//SYSUT2 DD DSN=PARTICLE,UNIT=TAPE,VOL=SER=T12400,LABEL=(20,SL),
//  DISP=(NEW,PASS),DCB=(BLKSIZE=1600,LRECL=80,RECFM=FB)
//SYSIN DD *
RCIN 73,, 'PAR'
COPY
/*
//PAS02 EXEC PROC=OSGENER
//SYSUT1 DD DSN=FILE25,UNIT=TAPE,VOL=SER=T12397,LABEL=(18,SL),
//  DISP=(OLD,KEEP)
//SYSUT2 DD DSN=EVENTS,UNIT=TAPE,VOL=SER=T12400,LABEL=(21,SL),
//  DISP=(NEW,PASS),DCB=(BLKSIZE=1600,LRECL=80,RECFM=FB)
//SYSIN DD *
RCIN 73,, 'EV'
COPY
/*
//PAS03 EXEC PROC=OSGENER
//SYSUT1 DD DSN=FILE25,UNIT=TAPE,VOL=SER=T12397,LABEL=(18,SL),
//  DISP=(OLD,KEEP)
//SYSUT2 DD DSN=CHARGES,UNIT=TAPE,VOL=SER=T12400,LABEL=(22,SL),
//  DISP=(NEW,KEEP),DCB=(BLKSIZE=1600,LRECL=80,RECFM=FB)
//SYSIN DD *
RCIN 73,, 'CHAR'
COPY
/*
//
```

NOTA: Se menciona el DCB con el cual se grabará la cinta.

CAPITULO VIII

MODELOS PARA LA EJECUCION DE PROGRAMAS GRABADOS EN CINTAS MAGNETICAS

PUNTO 1.

Ejecución de programas grabados en cintas magnéticas cuyos datos se presentan en tarjetas perforadas.

Modelo 357. Ejecución de un programa en lenguaje simbólico FORTRAN grabado completo, con el nombre GERMANIO, en el rótulo 14 de una cinta magnética que lleva el número de carrete 40022 y tiene etiquetas modelo IBM.

```
//..... JOB ....
//PAS01 EXEC PROC=FTG1CLG
//FORT.SYSIN DD DSN=GERMANIO,UNIT=TAPE,VOL=SER=T40022,LABEL=(14,SL),
// DISP=(OLD,KEEP)
/*
//GO.SYSIN DD *
Aquí tarjetas perforadas de DATOS
/*
//
```

Modelo 358. Ejecución de un programa en lenguaje absoluto de FORTRAN grabado completo, con el nombre TORIO, en el rótulo 3 de una cinta que lleva el número de carrete 31586 y tiene etiquetas modelo IBM.

```
//..... JOB ....
//PAS01 EXEC PROC=FTG1LG
//LKED.SYSIN DD DSN=TORIO,UNIT=TAPE,VOL=SER=T31586,LABEL=(3,SL),
// DISP=(OLD,KEEP)
/*
//GO.SYSIN DD *
Aquí tarjetas perforadas de DATOS
/*
//
```

Modelo 359. Ejecución de un programa en lenguaje simbólico FORTRAN del cual una parte está grabada, con el nombre SOLAR, en el rótulo 19 de una cinta magnética y el resto se presenta en tarjetas perforadas. La cinta lleva el número de carrete 21235 y tiene etiquetas modelo IBM.

```
//..... JOB ....
//PAS01 EXEC PROC=FTG1CLG
//FORT.SYSIN DD DSN=SOLAR,UNIT=TAPE,VOL=SER=T21235,LABEL=(19,SL),
// DISP=(OLD,KEEP)
// DD *
Aquí tarjetas perforadas en FORTRAN
/*
```

```
//GO.SYSIN DD *
Aquí tarjetas perforadas de DATOS
/*
//
```

Modelo 360. Ejecución de un programa en lenguaje absoluto de FORTRAN del cual una parte está grabada, con el nombre STARS, en el rótulo 33 de una cinta magnética y el resto se presenta en tarjetas perforadas. La cinta lleva el número de carrete 60844 y tiene etiquetas modelo IBM.

```
//..... JOB ....
//PASOI EXEC PROC=FTG1LG
//LKED.SYSIN DD DSN=STARS,UNIT=TAPE,VOL=SER=T60844,LABEL=(33,SL),
// DISP=(OLD,KEEP)
//          DD *,DCB=BLKSIZE=80
Aquí tarjetas perforadas en absoluto de FORTRAN
/*
//GO.SYSIN DD *
Aquí tarjetas perforadas de DATOS
/*
//
```

Modelo 361. Ejecución de un programa en lenguaje simbólico FORTRAN del cual una parte está grabada, con el nombre HYPERFIN, en el rótulo 6 de una cinta magnética y el resto está grabado, con el nombre STRUCTUR, en el rótulo 10 de la misma cinta. Esta lleva el número de carrete 01338 y tiene etiquetas modelo IBM.

```
//..... JOB ....
//PASOI EXEC PROC=FTG1CLG
//FORT.SYSIN DD DSN=HYPERFIN,UNIT=TAPE,VOL=SER=T01338,LABEL=(6,SL),
// DISP=(OLD,PASS)
//          DD DSN=STRUCTUR,UNIT=TAPE,VOL=REF=*,FORT.SYSIN,
// LABEL=(10,SL),DISP=(OLD,KEEP)
/*
//GO.SYSIN DD *
Aquí tarjetas perforadas de DATOS
/*
//
```

Modelo 362. Ejecución de un programa en lenguaje absoluto de FORTRAN del cual una parte está grabada, con el nombre CLORO, en el rótulo 2 de una cinta magnética y el resto está grabado, con el nombre TARGET, en el rótulo 14 de la misma cinta. Esta lleva el número de carrete 22716 y tiene etiquetas modelo IBM.

```
//..... JOB ....
```

```
//PASO1 EXEC PROC=FTG1LG
//LKED.SYSIN DD DSN=CLORO,UNIT=TAPE,VOL=SER=T22716,LABEL=(2,SL),
//  DISP=(OLD,PASS)
//          DD DSN=TARGET,UNIT=TAPE,VOL=SER=T22716,LABEL=(14,SL),
//  DISP=(OLD,KEEP)
/*
//GO.SYSIN DD *
Aquí tarjetas perforadas de DATOS
/*
//
```

Modelo 363. Ejecución de un programa en lenguaje simbólico FORTRAN del cual una parte está grabada, con el nombre OPTICAL, en el rótulo 28 de una cinta magnética y el resto está grabado, con el nombre ARTE, en el rótulo 11 de otra cinta. Las cintas llevan los números de carrete 61122 y 61133 respectivamente y tienen etiquetas modelo IBM.

```
//..... JOB ....
//PASO1 EXEC PROC=FTG1CLG
//FORT.SYSIN DD DSN=OPTICAL,UNIT=TAPE,VOL=SER=T61122,LABEL=(28,SL),
//  DISP=(OLD,KEEP)
//          DD DSN=ARTE,UNIT=TAPE,VOL=SER=T61133,LABEL=(11,SL),
//  DISP=(OLD,KEEP)
/*
//GO.SYSIN DD *
Aquí tarjetas perforadas de DATOS
/*
//
```

Modelo 364. Ejecución de un programa en lenguaje absoluto de FORTRAN del cual una parte está grabada, con el nombre PARCIAL, en el rótulo 15 de una cinta magnética y el resto está grabado, con el nombre FINAL en el rótulo 5 de otra cinta. Las cintas llevan los números de carrete 06314 y 10011 respectivamente y tienen etiquetas modelo IBM.

```
//..... JOB ....
//PASO1 EXEC PROC=FTG1LG
//LKED.SYSIN DD DSN=PARCIAL,UNIT=TAPE,VOL=SER=T06314,LABEL=(15,SL),
//  DISP=(OLD,KEEP)
//          DD DSN=FINAL,UNIT=TAPE,VOL=SER=T10011,LABEL=(5,SL),
//  DISP=(OLD,KEEP)
/*
//GO.SYSIN DD *
Aquí tarjetas perforadas de DATOS
/*
//
```

Modelo 265. Ejecución de un programa en lenguaje simbólico FORTRAN que reside, junto con otros, en el mismo rótulo de una cinta magnética, formando un archivo secuencial. En el rótulo 12 de la cinta que lleva el número de carrete 57639 y tiene etiquetas modelo IBM, se ha grabado con el nombre PGMDEADR un conjunto de 28 programas y rutinas. Se desea ejecutar solamente el programa PARSEC y sus rutinas CUAD y DIAG que ocupan el tercero, quinto y séptimo lugar, respectivamente en ese archivo. El problema se puede resolver en dos pasos: en el PASO1 el utilitario IEBGENER identifica en la cinta, reconociendo la sentencia FORTRAN 'END' los siete primeros programas y rutinas, creando con ellos una biblioteca temporaria particionada que reside en un disco auxiliar; en el PASO2 el procedimiento FTG1CLG compila, vincula y ejecuta el programa PARSEC con sus rutinas, obteniendo los tres miembros de la biblioteca temporaria.

```
//..... JOB ....
//PASO1 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=PGMDEADR,UNIT=TAPE,VOL=SER=T57689,LABEL=(12,SL),
// DISP=(OLD,KEEP)
//SYSUT2 DD DSN=TEMPOR.ADR,UNIT=WORKDISK,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1,1)),DCB=(BLKSIZE=2400,LRECL=80,RECFM=FB)
//SYSIN DD *
    GENERATE MAXNAME=7,MAXGPS=7
    MEMBER NAME=UNO
GROUP1 RECORD IDENT=(3,'END',7)
    MEMBER NAME=DOS
GROUP2 RECORD IDENT=(3,'END',7)
    MEMBER NAME=PARSEC
GROUP3 RECORD IDENT=(3,'END',7)
    MEMBER NAME=CUATRO
GROUP4 RECORD IDENT=(3,'END',7)
    MEMBER NAME=CUAD
GROUP5 RECORD IDENT=(3,'END',7)
    MEMBER NAME=SEIS
GROUP6 RECORD IDENT=(3,'END',7)
    MEMBER NAME=DIAG
GROUP7 RECORD IDENT=(3,'END',7)
/*
//PASO2 EXEC PROC=FTG1CLG
//FORT.SYSIN DD DSN=TEMPOR.ADR(PARSEC),VOL=REF=*.PASO1.SYSUT2,
// DISP=OLD
// DD DSN=TEMPOR.ADR(CUAD),VOL=REF=*.PASO1.SYSUT2,
// DISP=OLD
// DD DSN=TEMPOR.ADR(DIAG),VOL=REF=*.PASO1.SYSUT2,
// DISP=OLD
/*
```

```
//GO.SYSIN DD *
Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: El parámetro IDENT=(3,'END',7) del PASO1 significa: identificar una tira de tres caracteres, igual a la que aparece entre comillas, que está colocada de la columna 7 en adelante, en la imagen de tarjeta grabada en la cinta. Se menciona el DCB con el cual se grabará el disco auxiliar.

Modelo 366. Ejecución de un programa en lenguaje simbólico FORTRAN que reside, junto con otros, en el mismo rótulo de una cinta magnética y está marcado en cada uno de sus registros con una tira de caracteres. En el rótulo 24 de una cinta que lleva el número de carrete 10882 y tiene etiquetas modelo IBM, está grabado, con el nombre ARCHIVPH, un conjunto de 150 programas, cada uno de los cuales está marcado en la columna 73 y siguientes con una abreviatura de su nombre. Se desea ejecutar solamente el programa MULTIPAR identificando la tira de caracteres MULT. El problema se puede resolver en dos pasos: en el PASO1 el procedimiento OSGENER selecciona el programa MULTIPAR y lo graba en una biblioteca temporaria particionada que se crea en ese momento y reside en un disco auxiliar; en el PASO2 el procedimiento FTG1CLG ejecuta el programa obteniéndolo del disco auxiliar.

```
//..... JOB ....
//PASO1 EXEC PROC=OSGENER
//SYSUT1 DD DSN=ARCHIVPH,UNIT=TAPE,VOL=SER=T10882,LABEL=(24,SL),
// DISP=(OLD,KEEP)
//SYSUT2 DD DSN=TEMPOR.ADR(MULTIPAR),UNIT=WORKDISK,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//SYSIN DD *
RCIN 73,, 'MULT'
COPY
/*
//PASO2 EXEC PROC=FTG1CLG
//FORT.SYSIN DD DSN=TEMPOR.ADR(MULTIPAR),VOL=REF=*.PASO1.SYSUT2,
// DISP=(OLD,DELETE)
/*
//GO.SYSIN DD *
Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: Se menciona el DCB con el cual se grabará el disco auxiliar.

Modelo 367 . Ejecución de un programa en lenguaje simbólico FORTRAN de nombre PARTICLE, que pertenece a la biblioteca particionada BIBLIOT.FISICA. Esta biblioteca fué grabada en su totalidad, incluido el directorio y utilizando el utilitario IEBCOPY, en el rótulo 8 de la cinta magnética que lleva el número de carrete 13307 y tiene etiquetas modelo IBM. Este problema se puede resolver en dos pasos: en el PASO1 el utilitario IEBCOPY selecciona el programa PARTICLE y lo graba en una biblioteca temporaria particionada que se crea en ese momento y reside en un disco auxiliar. En el PASO2 el procedimiento FTG1CLG ejecuta el programa obteniéndolo de la biblioteca temporaria.

```
//..... JOB ....
//PASO1 EXEC PGM=IEBCOPY
//SYSPRINT DD SYSOUT=A
//SYSUT3 DD UNIT=WORKDISK,SPACE=(CYL,(2,1))
//SYSUT4 DD UNIT=WORKDISK,SPACE=(CYL,(2,1))
//ENTRADA DD DSN=BIBLIOT.FISICA,UNIT=TAPE,VOL=SER=T13307,LABEL=(8,SL),
// DISP=(OLD,KEEP)
//SALIDA DD DSN=TEMPOR.ADR,UNIT=WORKDISK,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1,1)),DCB=(BLKSIZE=3200,LRECL=80,RECFM=FB)
//SYSIN DD *
C O=SALIDA,I=ENTRADA
S M=((PARTICLE,,R))
/*
//PASO2 EXEC PROC=FTG1CLG
//FORT.SYSIN DD DSN=TEMPOR.ADR(PARTICLE),DISP=(OLD,DELETE),
// VOL=REF=*.PASO1.SALIDA
/*
//GO.SYSIN DD *
Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: Se menciona el DCB con el cual se grabará el disco auxiliar.

Modelo 368 . Ejecución de un programa en lenguaje simbólico ASSEMBLER grabado solo, con el nombre PAQUETE, en el rótulo 23 de una cinta magnética que lleva el número de carrete 51508 y tiene etiquetas modelo IBM.

```
//..... JOB ....
//PASO1 EXEC PROC=ASMFCLG
//ASM.SYSIN DD DSN=PAQUETE,UNIT=TAPE,VOL=SER=T51508,LABEL=(23,SL),
// DISP=(OLD,KEEP)
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
```

```
//GO.TARJETA DD *,DCB=BLKSIZE=80
Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: En el programa fuente ASSEMBLER se ha abierto un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

Modelo 369 . Ejecución de un programa en lenguaje absoluto de ASSEMBLER grabado solo, con el nombre GEN, en el rótulo 17 de una cinta magnética que lleva el número de carrete 47029 y tiene etiquetas modelo IBM.

```
//..... JOB ....
//PAS01 EXEC PROC=LKEDG
//LKED.SYSIN DD DSN=GEN,UNIT=TAPE,VOL=SER=T47029,LABEL=(17,SL),
// DISP=(OLD,KEEP)
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: En el programa fuente ASSEMBLER que dió origen al módulo grabado en la cinta, se abrió un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

Modelo 370 . Ejecución de un programa en lenguaje simbólico ASSEMBLER que reside, junto con otros, en el mismo rótulo de una cinta magnética y está marcado en cada uno de sus registros con una tira de caracteres. En el rótulo 12 de una cinta que lleva el número de carrete 73694 y tiene etiquetas modelo IBM, está grabado, con el nombre PGMQUIM un conjunto de 90 programas, cada uno de los cuales está marcado en la columna 73 y siguientes con una abreviatura de su nombre. Se desea ejecutar solamente el programa ASSEMPGM identificando la tira de caracteres ASPG. El problema se puede resolver en dos pasos: en el PAS01 el procedimiento OSGENER selecciona el programa ASSEMPGM y lo graba en una biblioteca temporaria particionada que se crea en ese momento y reside en un disco auxiliar; en el PAS02 el procedimiento ASMFCLG ejecuta el programa obteniéndolo del disco auxiliar.

```
//..... JOB ....
//PAS01 EXEC PROC=OSGENER
//SYSUT1 DD DSN=PGMQUIM,UNIT=TAPE,VOL=SER=T73694,LABEL=(12,SL),
// DISP=(OLD,KEEP)
//SYSUT2 DD DSN=TEMPOR.ADR(ASSEMPGM),UNIT=WORKDISK,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1,1)),DCB=(BLKSIZE=1600,LRECL=80,RECFM=FB)
//SYSIN DD *
RCIN 73,, 'ASPG'
COPY
/*
//PAS02 EXEC PROC=ASMFCLG
//ASM.SYSIN DD DSN=TEMPOR.ADR(ASSEMPGM),VOL=REF=*.PAS01.SYSUT2,
// DISP=(OLD,DELETE)
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: Se menciona el DCB con el cual se grabará el disco auxiliar. En el programa ASSEMBLER se ha abierto un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

Modelo 371. Ejecución de un programa en lenguaje simbólico ASSEMBLER de nombre SUFISA, que pertenece a la biblioteca particionada BIB.TEORICA. Esta biblioteca fué grabada en su totalidad, incluido el directorio y utilizando el utilitario IEBCOPY, en el rótulo 23 de una cinta magnética que lleva el número de carrete 21219 y tiene etiquetas modelo IBM. Este problema se puede resolver en dos pasos: en el PAS01 el utilitario IEBCOPY selecciona el programa SUFISA y lo graba en una biblioteca temporaria particionada que se crea en ese momento y reside en un disco auxiliar. En el PAS02 el procedimiento ASMFCLG ejecuta el programa obteniéndolo de la biblioteca temporaria.

```
//..... JOB ....
//PAS01 EXEC PGM=IEBCOPY
//SYSPRINT DD SYSOUT=A
//SYSUT3 DD UNIT=WORKDISK,SPACE=(CYL,(2,1))
//SYSUT4 DD UNIT=WORKDISK,SPACE=(CYL,(2,1))
//ENTRADA DD DSN=BIB.TEORICA,UNIT=TAPE,VOL=SER=T21219,LABEL=(23,SL),
// DISP=(OLD,KEEP)
//SALIDA DD DSN=TEMPOR.ADR,UNIT=WORKDISK,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1,1)),DCB=(BLKSIZE=2400,LRECL=80,RECFM=FB)
//SYSIN DD *
C O=SALIDA,I=ENTRADA
```



```

S  M=((SUFISA,,R))
/*
//PASO2 EXEC PROC=ASMFCLG
//ASM.SYSIN DD DSN=TEMPOR.ADR(SUFISA),DISP=(OLD,DELETE),
//  VOL=REF=*.PASO1.SALIDA
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas de DATOS
/*
//

```

NOTA: Se menciona el DCB con el cual se grabará el disco auxiliar. En el programa ASSEMBLER se ha abierto un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

Modelo 372. Ejecución de un programa en lenguaje simbólico PL/1 grabado solo, con el nombre ORDER, en el rótulo 27 de una cinta magnética que lleva el número de carrete 26810 y tiene etiquetas modelo IBM.

```

//..... JOB ....
//PASO1 EXEC PROC=PLIXCLG,PARM.PLI='CS(60)'
//PLI.SYSIN DD DSN=ORDER,UNIT=TAPE,VOL=SER=T26810,LABEL=(27,SL),
//  DISP=(OLD,KEEP)
/*
//GO.SYSIN DD *
  Aquí tarjetas perforadas de DATOS
/*
//

```

Modelo 373. Ejecución de un programa en lenguaje absoluto de PL/1 grabado solo, con el nombre ASIN, en el rótulo 36 de una cinta magnética que lleva el número de carrete 01123 y tiene etiquetas modelo IBM.

```

//..... JOB ....
//PASO1 EXEC PROC PLIXLG
//LKED.SYSIN DD DSN=ASIN,UNIT=TAPE,VOL=SER=T01123,LABEL=(36,SL),
//  DISP=(OLD,KEEP)
/*
//GO.SYSIN DD *
  Aquí tarjetas perforadas de DATOS
/*
//

```

Modelo 374. Ejecución de un programa en lenguaje simbólico PL/1 que reside, junto con otros, en el mismo rótulo de una cinta magnética y está marcado en cada

uno de sus registros con una tira de caracteres. En el rótulo 29 de una cinta que lleva el número de carrete 03791 y tiene etiquetas modelo IBM, está grabado, con el nombre ARCHIV78, un conjunto de 70 programas, cada uno de los cuales está marcado en la columna 73 y siguientes con una abreviatura de su nombre. Se desea ejecutar solamente el programa MATRIX identificando la tira de caracteres MAT. El problema se puede resolver en dos pasos: en el PASO1 el procedimiento OSGENER selecciona el programa MATRIX y lo graba en una biblioteca temporaria particionada que se crea en ese momento y reside en un disco auxiliar; en el PASO2 el procedimiento PLIXCLG ejecuta el programa obteniéndolo del disco auxiliar.

```
//..... JOB ....
//PASO1 EXEC PROC=OSGENER
//SYSUT1 DD DSN=ARCHIV78,UNIT=TAPE,VOL=SER=T03791,LABEL=(29,SL),
//  DISP=(OLD,KEEP)
//SYSUT2 DD DSN=TEMPOR.ADR(MATRIX),UNIT=WORKDISK,DISP=(NEW,PASS),
//  SPACE=(CYL,(1,1,1)),DCB=(BLKSIZE=2400,LRECL=80,RECFM=FB)
//SYSIN DD *
RCIN 73,, 'MAT'
COPY
/*
//PASO2 EXEC PROC=PLIXCLG,PARM.PLI='CS(60)'
//PLI.SYSIN DD DSN=TEMPOR.ADR(MATRIX),VOL=REF=*.PASO1.SYSUT2,
//  DISP=(OLD,DELETE)
/*
//GO.SYSIN DD *
Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: Se menciona el DCB con el cual se grabará el disco auxiliar.

Modelo 375 . Ejecución de un programa en lenguaje simbólico PL/1 de nombre ESTRUCT, que pertenece a la biblioteca BIBLIOT.FISEXP. Esta biblioteca fué grabada en su totalidad, incluido el directorio y utilizando el utilitario IEBCOPY, en el rótulo 40 de una cinta magnética que lleva el número de carrete 61238 y tiene etiquetas modelo IBM. El problema se puede resolver en dos pasos: en el PASO1 el utilitario IEBCOPY selecciona el programa ESTRUCT y lo graba en una biblioteca temporaria particionada que se crea en ese momento y reside en un disco auxiliar. En el PASO2 el procedimiento PLIXCLG ejecuta el programa obteniéndolo de la biblioteca temporaria.

```

//..... JOB ....
//PASO1 EXEC PGM=IEBCOPY
//SYSPRINT DD SYSOUT=A
//SYSUT3 DD UNIT=WORKDISK,SPACE=(CYL,(2,1))
//SYSUT4 DD UNIT=WORKDISK,SPACE=(CYL,(2,1))
//ENTRADA DD UNIT=TAPE,VOL=SER=T61238,LABEL=(40,SL),DSN=BIBLIOT.FISEXP,
//  DISP=(OLD,KEEP)
//SALIDA DD DSN=TEMPOR.ADR,UNIT=WORKDISK,DISP=(NEW,PASS),
//  SPACE=(CYL,(1,1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//SYSIN DD *
  C  O=SALIDA,I=ENTRADA
  S  M=((ESTRUCT,,R))
/*
//PASO2 EXEC PROC=PLIXCLG,PARM.PLI='CS(60)'
//PLI.SYSIN DD DSN=TEMPOR.ADR(ESTRUCT),VOL=REF=*.PASO1.SALIDA,
//  DISP=(OLD,DELETE)
/*
//GO.SYSIN DD *
  Aquí tarjetas perforadas de DATOS
/*
//

```

NOTA: Se menciona el DCB con el cual se grabará el disco auxiliar.

PUNTO 2.

Uso de la Referencia hacia Atrás a un programa grabado en cinta magnética y vinculado en un paso anterior de la misma tarea, para ejecutar dos o mas juegos de datos distintos, en los pasos siguientes.

Modelo 376 . Ejecución de un programa en lenguaje simbólico FORTRAN grabado en cinta magnética, compilado y vinculado en el PASO1 durante el cual resuelve el juego de DATOS1; en el PASO2 se hace referencia al mismo programa para resolver el juego de DATOS2. El programa reside, con el nombre PGFORSIM, en el rótulo 11 de una cinta que lleva el número de carrete 61002 y tiene etiquetas modelo IBM.

```
//..... JOB ....
//PASO1 EXEC PROC=FTG1CLG
//FORT.SYSIN DD DSN=PGFORSIM,UNIT=TAPE,VOL=SER=T61002,LABEL=(11,SL),
//  DISP=(OLD,KEEP)
/*
//GO.SYSIN DD *
  Aquí tarjetas perforadas de DATOS1
/*
//PASO2 EXEC PGM=*.PASO1.LKED.SYSLMOD
//FT06F001 DD SYSOUT=A
//FT05F001 DD *
  Aquí tarjetas perforadas de DATOS2
/*
//
```

Modelo 377 . Ejecución de un programa en lenguaje absoluto de FORTRAN grabado en cinta magnética, vinculado en el PASO1 en el cual resuelve el juego de DATOS1; en los PASOS 2 y 3 se hace referencia al mismo programa para resolver los juegos de DATOS2 y DATOS3. El programa reside, con el nombre PGFORABS, en el rótulo 6 de una cinta que lleva el número de carrete 15422 y tiene etiquetas modelo IBM.

```
//..... JOB ....
//PASO1 EXEC PROC=FTG1LG
//LKED.SYSIN DD DSN=PGFORABS,UNIT=TAPE,VOL=SER=T15422,LABEL=(6,SL),
//  DISP=(OLD,KEEP)
/*
//GO.SYSIN DD *
  Aquí tarjetas perforadas de DATOS1
/*
//PASO2 EXEC PGM=*.PASO1.LKED.SYSLMOD
//FT06F001 DD SYSOUT=A
//FT05F001 DD *
```

```

    Aquí tarjetas perforadas de DATOS2
/*
//PASO3 EXEC PGM=*.PASO1.LKED.SYSLMOD
//FT06F001 DD SYSOUT=A
//FT05F001 DD *
    Aquí tarjetas perforadas de DATOS3
/*
//

```

Modelo 378. Ejecución de un programa en lenguaje simbólico ASSEMBLER grabado en cinta magnética, compilado y vinculado en el PASO1 durante el cual resuelve el juego de DATOS1; en los PASOS 2 y 3 se hace referencia al mismo programa para resolver los juegos de DATOS2 y DATOS3. El programa reside, con el nombre PGASESIM, en el rótulo 13 de una cinta que lleva el número de carrete 37980 y tiene etiquetas modelo IBM.

```

//..... JOB ....
//PASO1 EXEC PROC=ASMFCLG
//ASM.SYSIN DD DSN=PGASESIM,UNIT=TAPE,VOL=SER=T37980,LABEL=(13,SL),
// DISP=(OLD,KEEP)
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
    Aquí tarjetas perforadas de DATOS1
/*
//PASO2 EXEC PGM=*.PASO1.LKED.SYSLMOD
//IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//TARJETA DD *,DCB=BLKSIZE=80
    Aquí tarjetas perforadas de DATOS2
/*
//PASO3 EXEC PGM=*.PASO1.LKED.SYSLMOD
//IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//TARJETA DD *,DCB=BLKSIZE=80
    Aquí tarjetas perforadas de DATOS3
/*
//

```

NOTA: En el programa ASSEMBLER se ha abierto un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

Modelo 379. Ejecución de un programa en lenguaje absoluto de ASSEMBLER grabado en cinta magnética, vinculado en el PASO1 durante el cual resuelve el juego de DATOS1; en el PASO2 se hace referencia al mismo programa para resolver el juego de DATOS2. El programa reside, con el nombre PGASEABS, en el rótulo 10 de una cinta que lleva el número de carrete 71018 y tiene etiquetas modelo IBM.

```
//..... JOB ....
//PAS01 EXEC PROC=LKEDG
//LKED.SYSIN DD DSN=PGASEABS,UNIT=TAPE,VOL=SER=T71018,LABEL=(10,SL),
// DISP=(OLD,KEEP)
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas de DATOS1
/*
//PAS02 EXEC PGM=*.PAS01.LKED.SYSLMOD
//IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//TARJETA DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas de DATOS2
/*
//
~//
```

NOTA: En el programa fuente ASSEMBLER que dió origen al módulo grabado en la cinta, se abrió un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

Modelo 380 . Ejecución de un programa en lenguaje simbólico PL/I grabado en cinta magnética, compilado y vinculado en el PAS01 durante el cual resuelve el juego de DATOS1; en el PAS02 se hace referencia al mismo programa para resolver el juego de DATOS2. El programa reside, con el nombre PGPL1SIM, en el rótulo 22 de una cinta que lleva el número de carrete 37381 y tiene etiquetas modelo IBM.

```
//..... JOB ....
//PAS01 EXEC PROC=PLIXCLG,PARM.PLI='CS(60)'
//PLI.SYSIN DD DSN=PGPL1SIM,UNIT=TAPE,VOL=SER=T37381,LABEL=(22,SL),
// DISP=(OLD,KEEP)
/*
//GO.SYSIN DD *
  Aquí tarjetas perforadas de DATOS1
/*
//PAS02 EXEC PGM=*.PAS01.LKED.SYSLMOD
//SYSPRINT DD SYSOUT=A
//SYSIN DD *
  Aquí tarjetas perforadas de DATOS2
/*
//
```

Modelo 381 . Ejecución de un programa en lenguaje absoluto de PL/I grabado en cinta magnética, vinculado en el PAS01 durante el cual resuelve el juego de DATOS1; en el PAS02 se hace referencia al mismo programa para resolver el juego de DATOS2. El programa reside, con el nombre PGPL1ABS, en el rótulo 7 de una cinta que lleva el número de carrete 92203 y tiene etiquetas modelo IBM.

```
//..... JOB ....  
//PAS01 EXEC PROC=PLIXLG  
//LKED.SYSIN DD DSN=PGPL1ABS,UNIT=TAPE,VOL=SER=T92203,LABEL=(7,SL),  
// DISP=(OLD,KEEP)  
/*  
//GO.SYSIN DD *  
  Aquí tarjetas perforadas de DATOS1  
/*  
//PAS02 EXEC PGM=*.PAS01.LKED.SYSLMOD  
//SYSPRINT DD SYSOUT=A  
//SYSIN DD *  
  Aquí tarjetas perforadas de DATOS2  
/*  
//
```

PUNTO 3.

Uso de la Referencia hacia Atrás a un juego de datos ejecutado en un paso anterior de la misma tarea, para ser procesado por dos o mas programas grabados en cintas magnéticas, en los pasos siguientes.

Modelo 382 . Ejecución de un programa en lenguaje simbólico FORTRAN grabado en cinta magnética, compilado y vinculado en el PAS01, en el cual resuelve el juego de DATOS1 y de otro programa en lenguaje simbólico FORTRAN que se presenta en tarjetas perforadas, compilado y vinculado en el PAS02, en el cual se hace referencia al mismo juego de DATOS1. El primer programa reside, con el nombre PGFORCIN, en el rótulo 4 de una cinta que lleva el número de carrete 16105 y tiene etiquetas modelo IBM.

```
//..... JOB ....
//PAS01 EXEC PROC=FTG1CLG
//FORT.SYSIN DD DSN=PGFORCIN,UNIT=TAPE,VOL=SER=T16105,LABEL=(4,SL),
//  DISP=(OLD,KEEP)
/*
//GO.SYSIN DD *
  Aquí tarjetas perforadas de DATOS1
/*
//PAS02 EXEC PROC=FTG1CLG
//FORT.SYSIN DD *
  Aquí tarjetas perforadas en FORTRAN
/*
//GO.SYSIN DD DSNAME=*.PAS01.GO.FT05FO01,DISP=(OLD,PASS)
/*
//
```

Modelo 383 . Ejecución de un programa en lenguaje absoluto de FORTRAN grabado en cinta magnética, vinculado en el PAS01, en el cual resuelve el juego de DATOS1 y de otro programa en lenguaje absoluto de FORTRAN que se presenta en tarjetas perforadas, vinculado en el PAS02, en el cual se hace referencia al mismo juego de DATOS1. El primer programa reside, con el nombre ABFORCIN, en el rótulo 9 de una cinta que lleva el número de carrete 20901 y tiene etiquetas modelo IBM.

```
//..... JOB ....
//PAS01 EXEC PROC=FTG1LG
//LKED.SYSIN DD DSN=ABFORCIN,UNIT=TAPE,VOL=SER=T20901,LABEL=(9,SL),
//  DISP=(OLD,KEEP)
/*
```



```
//GO.SYSIN DD *
Aquí tarjetas perforadas de DATOS1
/*
//PASO2 EXEC PROC=FTG1LG
//LKED.SYSIN DD *,DCB=BLKSIZE=80
Aquí tarjetas perforadas en absoluto de FORTRAN
/*
//GO.SYSIN DD DSN=*.PASO1.GO.FT05F001,DISP=(OLD,PASS)
/*
//
```

Modelo 384 . Ejecución de un programa en lenguaje simbólico FORTRAN grabado en cinta magnética, compilado y vinculado en el PASO1, en el cual resuelve el juego de DATOS1 y de otro programa en lenguaje simbólico FORTRAN grabado en otro rótulo de la misma cinta, compilado y vinculado en el PASO2, en el cual se hace referencia al mismo juego de DATOS1. El primer programa reside, con el nombre PROGRM1, en el rótulo 7 de una cinta que lleva el número de carrete 51016 y tiene etiquetas delo IBM: el segundo programa reside, con el nombre PROGRM2, en el rótulo 16 de esa misma cinta.

```
//..... JOB ....
//PASO1 EXEC PROC=FTG1CLG
//FORT.SYSIN DD DSN=PROGRM1,UNIT=TAPE,VOL=SER=T51016,LABEL=(7,SL),
// DISP=(OLD,PASS)
/*
//GO.SYSIN DD *
Aquí tarjetas perforadas de DATOS1
/*
//PASO2 EXEC PROC=FTG1CLG
//FORT.SYSIN DD DSN=PROGRM2,UNIT=TAPE,VOL=REF=*.PASO1.FORT.SYSIN,
// LABEL=(16,SL),DISP=(OLD,KEEP)
/*
//GO.SYSIN DD DSN=*.PASO1.GO.FT05F001,DISP=(OLD,PASS)
/*
//
```

Modelo 385 . Ejecución de un programa en lenguaje absoluto de FORTRAN grabado en cinta magnética, vinculado en el PASO1, en el cual resuelve el juego de DATOS1 y de otro programa en lenguaje absoluto de FORTRAN grabado en otro rótulo de la misma cinta, vinculado en el PASO2, en el cual se hace referencia al mismo juego de DATOS1. El primer programa reside, con el nombre ABSFORT1, en el rótulo 10 de una cinta que lleva el número de carrete 65655 y tiene etiquetas modelo IEM; el segundo programa reside, con el nombre ABSFORT2, en el rótulo 18 de esa misma

cinta.

```
//..... JOB ....
//PAS01 EXEC PROC=FTG1LG
//LKED.SYSIN DD DSN=ABSFORT1,UNIT=TAPE,VOL=SER=T65655,LABEL=(10,SL),
// DISP=(OLD,PASS)
/*
//GO.SYSIN DD *
  Aquí tarjetas perforadas de DATOS1
/*
//PAS02 EXEC PROC=FTG1LG
//LKED.SYSIN DD DSN=ABSFORT2,UNIT=TAPE,VOL=SER=T65655,LABEL=(18,SL),
// DISP=(OLD,KEEP)
/*
//GO.SYSIN DD DSNAME=*.PAS01.GO.FT05F001,DISP=(OLD,PASS)
/*
//
```

Modelo 386 . Ejecución de un programa en lenguaje simbólico FORTRAN grabado en cinta magnética, compilado y vinculado en el PAS01, en el cual resuelve el juego de DATOS1 y de otro programa en lenguaje simbólico FORTRAN grabado en otra cinta, compilado y vinculado en el PAS02, en el cual se hace referencia al mismo juego de DATOS1. El primer programa reside, con el nombre PRUEBA1, en el rótulo 13 de una cinta que lleva el número de carrete 60211 y el segundo programa reside, con el nombre PRUEBA2, en el rótulo 8 de otra cinta que lleva el número de carrete 77122. Ambas cintas tienen etiquetas modelo IBM.

```
//..... JOB ....
//PAS01 EXEC PROC=FTG1CLG
//FORT.SYSIN DD DSN=PRUEBA1,UNIT=TAPE,VOL=SER=T60211,LABEL=(13,SL),
// DISP=(OLD,KEEP)
/*
//GO.SYSIN DD *
  Aquí tarjetas perforadas de DATOS1
/*
//PAS02 EXEC PROC=FTG1CLG
//FORT.SYSIN DD DSN=PRUEBA2,UNIT=TAPE,VOL=SER=T77122,LABEL=(8,SL),
// DISP=(OLD,KEEP)
/*
//GO.SYSIN DD DSNAME=*.PAS01.GO.FT05F001,DISP=(OLD,PASS)
/*
//
```

Modelo 387 . Ejecución de un programa en lenguaje absoluto de FORTRAN grabado en cinta magnética, vinculado en el PAS01, en el cual resuelve el juego de DATOS1 y

de otro programa en lenguaje absoluto de FORTRAN grabado en otra cinta, vinculado en el PAS02, en el cual se hace referencia al mismo juego de DATOS1. El primer programa reside, con el nombre DRFORT1, en el rótulo 11 de una cinta que lleva el número de carrete 53890 y el segundo programa reside, con el nombre DRFORT2, en el rótulo 9 de otra cinta que lleva el número de carrete 50001. Ambas cintas tienen etiquetas modelo IBM.

```
//..... JOB ....
//PAS01 EXEC PROC=FTG1LG
//LKED.SYSIN DD DSN=DRFORT1,UNIT=TAPE,VOL=SER=T53890,LABEL=(11,SL),
// DISP=(OLD,KEEP)
/*
//GO.SYSIN DD *
  Aquí tarjetas perforadas de DATOS1
/*
//PAS02 EXEC PROC=FTG1LG
//LKED.SYSIN DD DSN=DRFORT2,UNIT=TAPE,VOL=SER=T50001,LABEL=(9,SL),
// DISP=(OLD,KEEP)
/*
//GO.SYSIN DD DSNAME=*.PAS01.GO.FT05F001,DISP=(OLD,PASS)
/*
//
```

Modelo 388 . Ejecución de un programa en lenguaje simbólico ASSEMBLER grabado en cinta magnética, compilado y vinculado en el PAS01, en el cual resuelve el juego de DATOS1 y de otro programa en lenguaje simbólico ASSEMBLER que se presenta en tarjetas perforadas, compilado y vinculado en el PAS02, en el cual se hace referencia al mismo juego de DATOS1. El primer programa reside con el nombre PGASSCIN, en el rótulo 18 de una cinta que lleva el número de carrete 30446 y tiene etiquetas modelo IBM.

```
//..... JOB ....
//PAS01 EXEC PROC=ASMFCLG
//ASM.SYSIN DD DSN=PGASSCIN,UNIT=TAPE,VOL=SER=T30446,LABEL=(18,SL),
// DISP=(OLD,KEEP)
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas de DATOS1
/*
//PAS02 EXEC PROC=ASMFCLG
//ASM.SYSIN DD *
  Aquí tarjetas perforadas en ASSEMBLER
/*
```

```
//GO.IMPRIMIR DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.LEER DD DSN=*.PAS01.GO.TARJETA,DISP=(OLD,PASS)
/*
//
```

NOTA: En el programa ASSEMBLER usado en el PAS01, se ha abierto un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO y en el ASSEMBLER usado en el PAS02 se ha abierto un archivo de entrada de nombre LEER y un archivo de salida de nombre IMPRIMIR.

Modelo 389. Ejecución de un programa en lenguaje absoluto de ASSEMBLER grabado en cinta magnética, vinculado en el PAS01, en el cual resuelve el juego de DATOS1 y de otro programa en lenguaje absoluto de ASSEMBLER que se presenta en tarjetas perforadas, vinculado en el PAS02, en el cual se hace referencia al mismo juego de DATOS1. El primer programa reside, con el nombre ABASSCIN, en el rótulo 19 de una cinta que lleva el número de carrete 08334 y tiene etiquetas modelo IBM.

```
//..... JOB ....
//PAS01 EXEC PROC=LKEDG
//LKED.SYSIN DD DSN=ABASSCIN,UNIT=TAPE,VOL=SER=T08334,LABEL=(19,SL),
// DISP=(OLD,KEEP)
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
Aquí tarjetas perforadas de DATOS1
/*
//PAS02 EXEC PROC=LKEDG
//LKED.SYSIN DD *,DCB=BLKSIZE=80
Aquí tarjetas perforadas en absoluto de ASSEMBLER
/*
//GO.IMPRIMIR DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.LEER DD DSN=*.PAS01.GO.TARJETA,DISP=(OLD,PASS)
/*
//
```

NOTA: En el programa fuente ASSEMBLER que dió origen al módulo usado en el PAS01, se abrió un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO. En el programa que dió origen al módulo usado en el PAS02 se abrió un archivo de entrada de nombre LEER y uno de salida IMPRIMIR.

Modelo 390. Ejecución de un programa en lenguaje simbólico ASSEMBLER grabado en cinta magnética, compilado y vinculado en el PAS01, en el cual resuelve el juego

de DATOS1 y de otro programa en lenguaje simbólico ASSEMBLER grabado en otro rótulo de la misma cinta, compilado y vinculado en el PASO2, en el cual se hace referencia al mismo juego de DATOS1. El primer programa reside con el nombre FORMULA1, en el rótulo 30 de una cinta que lleva el número de carrete 32888 y tiene etiquetas modelo IBM; el segundo programa reside, con el nombre FORMULA2, en el rótulo 33 de esa misma cinta.

```
//..... JOB ....
//PASO1 EXEC PROC=ASMFCLG
//ASM.SYSIN DD DSN=FORMULA1,UNIT=TAPE,VOL=SER=T32888,LABEL=(30,SL),
// DISP=(OLD,PASS)
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas de DATOS1
/*
//PASO2 EXEC PROC=ASMFCLG
//ASM.SYSIN DD DSN=FORMULA2,UNIT=TAPE,VOL=SER=T32888,LABEL=(33,SL),
// DISP=(OLD,KEEP)
/*
//CO.IMPRIMIR DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.LEER DD DSNAME=*.PASO1.CO.TARJETA,DISP=(OLD,PASS)
/*
//
```

NOTA: En el programa ASSEMBLER usado en el PASO1, se ha abierto un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO. En el programa ASSEMBLER usado en el PASO2 se ha abierto un archivo de entrada de datos de nombre LEER y un archivo de salida de datos de nombre IMPRIMIR.

Modelo 391 . Ejecución de un programa en lenguaje absoluto de ASSEMBLER grabado en cinta magnética, vinculado en el PASO1, en el cual resuelve el juego de DATOS1 y de otro programa en lenguaje absoluto de ASSEMBLER grabado en otro rótulo de la misma cinta, vinculado en el PASO2, en el cual se hace referencia al mismo juego de DATOS1. El primer programa reside, con el nombre TEMA1, en el rótulo 20 de una cinta que lleva el número de carrete 22022 y tiene etiquetas modelo IBM; el segundo programa reside, con el nombre TEMA2, en el rótulo 31 de la misma cinta.

```
//..... JOB ....
```

```
//PAS01 EXEC PROC=LKEDG
//LKED.SYSIN DD DSN=TEMA1,UNIT=TAPE,VOL=SER=T22022,LABEL=(20,SL),
//  DISP=(OLD,PASS)
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas de DATOS1
/*
//PAS02 EXEC PROC=LKEDG
//LKED.SYSIN DD DSN=TEMA2,UNIT=TAPE,VOL=SER=T22022,LABEL=(31,SL),
//  DISP=(OLD,KEEP)
/*
//GO.IMPRIMIR DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.LEER DD DSN=*.PAS01.GO.TARJETA,DISP=(OLD,PASS)
/*
//
```

NOTA: En el programa fuente ASSEMBLER que dió origen al módulo usado en el PAS01, se abrió un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO. En el programa fuente ASSEMBLER que dió origen al módulo usado en el PAS02 se abrió un archivo de entrada de datos de nombre LEER y un archivo de salida de datos de nombre IMPRIMIR.

Modelo 392 . Ejecución de un programa en lenguaje simbólico ASSEMBLER grabado en cinta magnética, compilado y vinculado en el PAS01, en el cual resuelve el juego de DATOS1 y de otro programa en lenguaje simbólico ASSEMBLER grabado en otra cinta, compilado y vinculado en el PAS02, en el cual se hace referencia al mismo juego de DATOS1. El primer programa reside, con el nombre ALGORIT1, en el rótulo 17 de una cinta que lleva el número de carrete 03321 y el segundo programa reside, con el nombre LGORIT2, en el rótulo 23 de otra cinta que lleva el número de carrete 04421. Ambas cintas tienen etiquetas modelo IBM.

```
//..... JOB ....
//PAS01 EXEC PROC=ASMFCLG
//ASM.SYSIN DD DSN=ALGORIT1,UNIT=TAPE,VOL=SER=T03321,LABEL=(17,SL),
//  DISP=(OLD,KEEP)
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas de DATOS1
/*
//PAS02 EXEC PROC=ASMFCLG
//ASM.SYSIN DD DSN=ALGORIT2,UNIT=TAPE,VOL=SER=T04421,LABEL=(23,SL),
//  DISP=(OLD,KEEP)
```

```

/*
//GO.IMPRIMIR DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.LEER DD DSN=*.PAS01.GO.TARJETA,DISP=(OLD,PASS)
/*
//

```

NOTA: En el programa ASSEMBLER usado en el PAS01, se ha abierto un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO. En el programa ASSEMBLER usado en el PAS02 se ha abierto un archivo de entrada de datos de nombre LEER y un archivo de salida de datos de nombre IMPRIMIR.

Modelo 393 . Ejecución de un programa en lenguaje absoluto de ASSEMBLER grabado en cinta magnética, vinculado en el PAS01, en el cual resuelve el juego de DATOS1 y de otro programa en lenguaje absoluto de ASSEMBLER grabado en otra cinta, vinculado en el PAS02, en el cual se hace referencia al mismo juego de DATOS1. El primer programa reside, con el nombre DRPGMAS1, en el rótulo 16 de una cinta que lleva el número de carrete 71161 y el segundo programa reside, con el nombre DRPGMAS2, en el rótulo 6 de otra cinta que lleva el número de carrete 00033. Ambas cintas tienen etiquetas modelo IBM.

```

//..... JOB ....
//PAS01 EXEC PROC=LKEDG
//LKED.SYSIN DD DSN=DRPGMAS1,UNIT=TAPE,VOL=SER=T71161,LABEL=(16,SL),
// DISP=(OLD,KEEP)
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
Aquí tarjetas perforadas de DATOS1
/*
//PAS02 EXEC PROC=LKEDG
//LKED.SYSIN DD DSN=DRPGMAS2,UNIT=TAPE,VOL=SER=T00033,LABEL=(6,SL),
// DISP=(OLD,KEEP)
/*
//GO.IMPRIMIR DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.LEER DD DSN=*.PAS01.GO.TARJETA,DISP=(OLD,PASS)
/*
//

```

NOTA: En el programa fuente ASSEMBLER que dió origen al módulo usado en el PAS01, se abrió un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO. En el programa ASSEMBLER que dió origen al módulo u-

sado en el PAS02, se abrió un archivo de entrada de datos de nombre LEER y un archivo de salida de datos de nombre IMPRIMIR.

Modelo 394 . Ejecución de un programa en lenguaje simbólico PL/1 grabado en cinta magnética, compilado y vinculado en el PAS01, en el cual resuelve el juego de DATOS1 y de otro programa en lenguaje simbólico PL/1 que se presenta en tarjetas perforadas, compilado y vinculado en el PAS02, en el cual se hace referencia al mismo juego de DATOS1. El primer programa reside, con el nombre PGPL1CIN, en el rótulo 14 de una cinta que lleva el número de carrete 40261 y tiene etiquetas modelo IBM.

```
//..... JOB ....
//PAS01 EXEC PROC=PLIXCLG,PARM.PLI='CS(60)'
//PLI.SYSIN DD DSN=PGPL1CIN,UNIT=TAPE,VOL=SER=T40261,LABEL=(14,SL),
// DISP=(OLD,KEEP)
/*
//GO.SYSIN DD *
Aquí tarjetas perforadas de DATOS1
/*
//PAS02 EXEC PROC=PLIXCLG,PARM.PLI='CS(60)'
//PLI.SYSIN DD *
Aquí tarjetas perforadas en PL/1
/*
//GO.SYSIN DD DSNAME= .PAS01.GO.SYSIN,DISP=(OLD,PASS)
/*
//
```

Modelo 395 . Ejecución de un programa en lenguaje absoluto de PL/1 grabado en cinta magnética, vinculado en el PAS01, en el cual resuelve el juego de DATOS1 y de otro programa en lenguaje absoluto de PL/1 que se presenta en tarjetas perforadas, vinculado en el PAS02, en el cual se hace referencia al mismo juego de DATOS1. El primer programa reside, con el nombre ABPL1CIN, en el rótulo 30 de una cinta que lleva el número de carrete 11533 y tiene etiquetas modelo IBM.

```
//..... JOB ....
//PAS01 EXEC PROC=PLIXLG
//LKED.SYSIN DD DSN=ABPL1CIN,UNIT=TAPE,VOL=SER=T11533,LABEL=(30,SL),
// DISP=(OLD,KEEP)
/*
//GO.SYSIN DD *
Aquí tarjetas perforadas de DATOS1
/*
```



```
//PASO2 EXEC PROC=PLIXLG
//LKED.SYSIN DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas en absoluto de PL/1
/*
//GO.SYSIN DD DSN=*.PASO1.GO.SYSIN,DISP=(OLD,PASS)
/*
//
```

Modelo 396 . Ejecución de un programa en lenguaje simbólico PL/1 grabado en cinta magnética, compilado y vinculado en el PASO1, en el cual resuelve el juego de DATOS1 y de otro programa en lenguaje simbólico PL/1 grabado en otro rótulo de la misma cinta, compilado y vinculado en el PASO2, en el cual se hace referencia al mismo juego de DATOS1. El primer programa reside, con el nombre PRGRPL11, en el rótulo 17 de una cinta que lleva el número de carrete 31555 y tiene etiquetas modelo IBM; el segundo programa reside, con el nombre PRGRPL22, en el rótulo 26 de esa misma cinta.

```
//..... JOB ....
//PASO1 EXEC PROC=PLIXCLG,PARM.PLI='CS(60)'
//PLI.SYSIN DD DSN=PRGRPL11,UNIT=TAPE,VOL=SER=T31555,LABEL=(17,SL),
// DISP=(OLD,PASS)
/*
//GO.SYSIN DD *
  Aquí tarjetas perforadas de DATOS1
/*
//PASO2 EXEC PROC=PLIXCLG,PARM.PLI='CS(60)'
//PLI.SYSIN DD DSN=PRGRPL22,UNIT=TAPE,VOL=REF=*.PASO1.PLI.SYSIN,
// LABEL=(26,SL),DISP=(OLD,KEEP)
/*
//GO.SYSIN DD DSN=*.PASO1.GO.SYSIN,DISP=(OLD,PASS)
/*
//
```

Modelo 397 . Ejecución de un programa en lenguaje absoluto de PL/1 grabado en cinta magnética, vinculado en el PASO1, en el cual resuelve el juego de DATOS1 y de otro programa en lenguaje absoluto de PL/1 grabado en otro rótulo de la misma cinta, vinculado en el PASO2, en el cual se hace referencia al mismo juego de DATOS1. El primer programa reside, con el nombre ABSPL11, en el rótulo 20 de una cinta que lleva el número de carrete 12312 y tiene etiquetas modelo IBM; el segundo programa reside, con el nombre ABSPL12, en el rótulo 23 de esa misma cinta.

```
//..... JOB ....
```

```
//PAS01 EXEC PROC=PLIXLG
//LKED.SYSIN DD DSN=ABSPL11,UNIT=TAPE,VOL=SER=T12312,LABEL=(20,SL),
// DISP=(OLD,PASS)
/*
//GO.SYSIN DD *
Aquí tarjetas perforadas de DATOS1
/*
//PAS02 EXEC PROC=PLIXLG
//LKED.SYSIN DD DSN=ABSPL12,UNIT=TAPE,VOL=SER=T12312,LABEL=(23,SL),
// DISP=(OLD,KEEP)
/*
//GO.SYSIN DD DSNAME=*.PAS01.GO.SYSIN,DISP=(OLD,PASS)
/*
//
```

Modelo 398 . Ejecución de un programa en lenguaje simbólico PL/1 grabado en cinta magnética, compilado y vinculado en el PAS01, en el cual resuelve el juego de DATOS1 y de otro programa en lenguaje simbólico PL/1 grabado en otra cinta, compilado y vinculado en el PAS02, en el cual se hace referencia al mismo juego de DATOS1. El primer programa reside, con el nombre ENSAY01, en el rótulo 3 de una cinta que lleva el número de carrete 33190 y el segundo programa reside, con el nombre ENSAY02, en el rótulo 15 de otra cinta que lleva el número de carrete 33197. Ambas cintas tienen etiquetas modelo IBM.

```
//..... JOB ....
//PAS01 EXEC PROC=PLIXCLG,PARM.PLI='CS(60)'
//PLI.SYSIN DD DSN=ENSAY01,UNIT=TAPE,VOL=SER=T33190,LABEL=(3,SL),
// DISP=(OLD,KEEP)
/*
//GO.SYSIN DD *
Aquí tarjetas perforadas de DATOS1
/*
//PAS02 EXEC PROC=PLIXCLG,PARM.PLI='CS(60)'
//PLI.SYSIN DD DSN=ENSAY02,UNIT=TAPE,VOL=SER=T33197,LABEL=(15,SL),
// DISP=(OLD,KEEP)
/*
//GO.SYSIN DD DSNAME=*.PAS01.GO.SYSIN,DISP=(OLD,PASS)
/*
//
```

Modelo 399 . Ejecución de un programa en lenguaje absoluto de PL/1 grabado en cinta magnética, vinculado en el PAS01, en el cual resuelve el juego de DATOS1 y de otro programa en lenguaje absoluto de PL/1 grabado en otra cinta, vinculado en el PAS02, en el cual se hace referencia al mismo juego de DATOS1. El primer programa reside, con el nombre DRPL11, en el rótulo 12 de una cinta que lleva el nú-

mero de carrete 60060 y el segundo programa reside, con el nombre DRPL12, en el r tulo 3 de otra cinta que lleva el n mero de carrete 70008. Ambas cintas tienen etiquetas modelo IBM.

```
//..... JOB ....
//PASO1 EXEC PROC=PLIXLG
//LKED.SYSIN DD DSN=DRPL11,UNIT=TAPE,VOL=SER=T60060,LABEL=(12,SL),
//  DISP=(OLD,KEEP)
/*
//GO.SYSIN DD *
  Aqu  tarjetas perforadas de DATOS1
/*
//PASO2 EXEC PROC=PLIXLG
//LKED.SYSIN DD DSN=DRPL12,UNIT=TAPE,VOL=SER=T70008,LABEL=(3,SL),
//  DISP=(OLD,KEEP)
/*
//GO.SYSIN DD DSN=*.PASO1.GO.SYSIN,DISP=(OLD,PASS)
/*
//
```


CAPITULO IX

MODELOS PARA LA EJECUCION DE PROGRAMAS QUE
UTILIZAN CINTAS MAGNETICAS COMO ARCHIVOS DE
ENTRADA DE DATOS

PUNTO 1.

Ejecución de programas en FORTRAN que se presentan en tarjetas perforadas y utilizan cintas magnéticas como archivos de entrada de datos.

Modelo 400 . Ejecución de un programa en tarjetas perforadas en lenguaje simbólico FORTRAN, cuyos datos de entrada constituyen un archivo que reside, con el nombre DATOS, en el rótulo 5 de una cinta magnética. La cinta lleva el número de carrete 03303 y tiene etiquetas modelo IBM.

```
//..... JOB ....
//PAS01 EXEC PROC=FTG1CLG
//FORT.SYSIN DD *
  Aquí tarjetas perforadas en FORTRAN
/*
//GO.FT14F001 DD DSN=DATOS,UNIT=TAPE,VOL=SER=T03303,LABEL=(5,SL),
// DISP=(OLD,KEEP)
/*
//
```

NOTA: El archivo de entrada de datos está definido en la sentencia GO, en donde se le asigna el número de referencia 14; por lo tanto en el programa fuente, cuando se lean datos de ese archivo, se codificará READ14.

Modelo 401 . Ejecución de un programa en tarjetas perforadas en lenguaje absoluto de FORTRAN, cuyos datos de entrada constituyen un archivo que reside, con el nombre FILE en el rótulo 20 de una cinta magnética. La cinta lleva el número de carrete 51320 y tiene etiquetas modelo IBM.

```
//..... JOB ....
//PAS01 EXEC PROC=FTG1LG
//LKED.SYSIN DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas en absoluto de FORTRAN
/*
//GO.FT18F001 DD DSN=FILE,UNIT=TAPE,VOL=SER=T51320,LABEL=(20,SL),
// DISP=(OLD,KEEP)
/*
//
```

NOTA: El archivo de entrada de datos está definido en la sentencia GO, en donde se le asigna el número de referencia 18; por lo tanto en el programa fuente que dió origen al lote de tarjetas en absoluto, para leer datos de ese archivo se co-

dificó READ18.

Modelo 402 . Ejecución de un programa en tarjetas perforadas en lenguaje simbólico FORTRAN, cuyos datos de entrada son tres archivos distintos que residen en rótulos consecutivos o no consecutivos de una cinta magnética. Los archivos, de nombres DATOSAL, DATOSNS y DATOSMR, están grabados en los rótulos 19, 30 y 31 de la cinta que lleva el número de carrete 16556 y tiene etiquetas modelo IBM.

```
//..... JOB ....
//PAS01 EXEC PROC=FTG1CLG
//FORT.SYSIN DD *
  Aquí tarjetas perforadas en FORTRAN
/*
//GO.FT13F001 DD DSN=DATOSAL,UNIT=TAPE,VOL=SER=T16556,LABEL=(19,SL),
// DISP=(OLD,PASS)
//GO.FT13F002 DD DSN=DATOSNS,UNIT=TAPE,VOL=REF=*.GO.FT13F001,
// LABEL=(30,SL)
//GO.FT13F003 DD DSN=DATOSMR,UNIT=TAPE,VOL=REF=*.GO.FT13F001,
// LABEL=(31,SL),DISP=(OLD,KEEP)
/*
//
```

NOTA: Los archivos de entrada de datos están definidos en las sentencias GO, en donde se les asigna el mismo número de referencia 13 y los números de secuencia 1, 2 y 3 respectivamente. Por lo tanto en el programa fuente, cuando se lean datos de esos archivos se codificará una sentencia READ de este tipo:

READ(13,n° de formato,END=dirección). La "dirección" es la marca de la sentencia a la cual deberá saltar el programa para continuar la ejecución, cada vez que llega al final de uno de los archivos.

Modelo 403 . Ejecución de un programa en tarjetas perforadas en lenguaje absoluto de FORTRAN, cuyos datos de entrada son cuatro archivos distintos que residen en rótulos consecutivos o no consecutivos de la misma cinta magnética. Los archivos, de nombres ARCHIVO1, ARCHIVO2, ARCHIVO3 y ARCHIVO4, están grabados en los rótulos 6, 17, 18 y 23 de la cinta que lleva el número de carrete 67722 y tiene etiquetas modelo IBM.

```
//..... JOB ....
//PAS01 EXEC PROC=FTG1LG
```

```
//LKED.SYSIN DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas en absoluto de FORTRAN
/*
//GO.FT17F001 DD DSN=ARCHIVO1,UNIT=TAPE,VOL=SER=T67722,LABEL=(6,SL),
//  DISP=(OLD,PASS)
//GO.FT17F002 DD DSN=ARCHIVO2,UNIT=TAPE,VOL=REF=*.GO.FT17F001,
//  LABEL=(17,SL)
//GO.FT17F003 DD DSN=ARCHIVO3,UNIT=TAPE,VOL=REF=*.GO.FT17F001,
//  LABEL=(18,SL)
//GO.FT17F004 DD DSN=ARCHIVO4,UNIT=TAPE,VOL=REF=*.GO.FT17F001,
//  LABEL=(23,SL),DISP=(OLD,KEEP)
/*
//
```

NOTA: Los archivos de entrada de datos está definidos en las sentencias GO, en donde se les asigna el mismo número de referencia 17 y los números de secuencia 1, 2, 3 y 4 respectivamente. Por lo tanto en el programa fuente que dió origen al lote de tarjetas perforadas en absoluto, para leer datos de esos archivos se codificó una sentencia READ de este tipo: READ(17,n°de formato,END=dirección). La "dirección" es la marca de la sentencia a la cual deberá saltar el programa para continuar la ejecución, cada vez que llega al final de uno de los archivos.

Modelo 404 . Ejecución de un programa en tarjetas perforadas en lenguaje simbólico FORTRAN, cuyos datos de entrada son dos archivos que residen en dos cintas magnéticas distintas. En el rótulo 28 de la cinta que lleva el número de carrete 25107 están grabados, con el nombre INPUT40, los datos de entrada del primer archivo y en el rótulo 7 de la cinta que lleva el número de carrete 88661 están grabados, con el nombre INPUT55, los datos de entrada del segundo archivo. Ambas cintas tienen etiquetas modelo IBM.

```
//..... JOB ....
//PAS01 EXEC PROC=FTG1CLG
//FORT.SYSIN DD *
  Aquí tarjetas perforadas en FORTRAN
/*
//GO.FT13F001 DD DSN=INPUT40,VOL=SER=T25107,LABEL=(28,SL),UNIT=TAPE,
//  DISP=(OLD,KEEP)
//GO.FT18F001 DD DSN=INPUT55,VOL=SER=T88661,LABEL=(7,SL),UNIT=TAPE,
//  DISP=(OLD,KEEP)
/*
//
```

NOTA: Los archivos de entrada de datos están definidos en las sentencias GO, en

donde se les asignan los números de referencia 13 y 18; por lo tanto en el programa fuente, cuando se lean datos de esos archivos se codificará READ13 y READ18 respectivamente.

Modelo 405 . Ejecución de un programa en tarjetas perforadas en lenguaje absoluto de FORTRAN, cuyos datos de entrada son dos archivos que residen en dos cintas magnéticas distintas. En el rótulo 36 de la cinta que lleva el número de carrete 03349 están grabados, con el nombre DATOS10, los datos de entrada del primer archivo y en el rótulo 21 de la cinta que lleva el número de carrete 10608 están grabados, con el nombre DATOS13 los datos de entrada del segundo archivo. Ambas cintas tienen etiquetas modelo IBM.

```
//..... JOB ....
//PAS01 EXEC PROC=FTG1LG
//LKED.SYSIN DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas en absoluto de FORTRAN
/*
//GO.FT15F001 DD DSN=DATOS10,VOL=SER=T03349,LABEL=(36,SL),UNIT=TAPE,
//  DISP=(OLD,KEEP)
//GO.FT04F001 DD DSN=DATOS13,VOL=SER=T10608,LABEL=(21,SL),UNIT=TAPE,
//  DISP=(OLD,KEEP)
/*
//
```

NOTA: Los archivos de entrada de datos están definidos en las sentencias GO, en donde se les asignan los números de referencia 15 y 4; por lo tanto en el programa fuente que dió origen el lote de tarjetas perforadas en absoluto, para leer datos de esos archivos se codificó READ 15 y READ 4 respectivamente.

PUNTO 2.

Ejecución de programas en FORTRAN, residentes en una biblioteca particionada y catalogada de programas en lenguaje simbólico, de nombre SIMBOLIC.CNEA, que utilizan cintas magnéticas como archivos de entrada de datos.

Modelo 406 . Ejecución de un programa grabado en la biblioteca con el nombre SUPATIP, cuyos datos de entrada constituyen un archivo que reside, con el nombre SUPADAT, en el rótulo 14 de una cinta magnética que lleva el número de carrete 12110 y tiene etiquetas modelo IBM.

```
//..... JOB ....
//PAS01 EXEC PROC=FTG1CLG
//FORT.SYSIN DD DSN=SIMBOLIC.CNEA(SUPATIP),DISP=SHR
/*
//GO.FT13F001 DD DSN=SUPADAT,UNIT=TAPE,VOL=SER=T12110,DISP=(OLD,KEEP),
// LABEL=(14,SL)
/*
//
```

NOTA: El archivo de entrada de datos está definido en la sentencia GO, en donde se le asigna el número de referencia 13; por lo tanto en el programa fuente, cuando se lean datos de ese archivo, se codificará READ 13.

Modelo 407 . Ejecución de un programa grabado en la biblioteca con el nombre DISTANG, cuyos datos de entrada son cuatro archivos distintos que residen en rótulos consecutivos o no consecutivos de una cinta magnética. Los archivos, de nombres DISDAT1A, DISDAT1B, DISDAT1C y DISDAT1D, están grabados en los rótulos 4, 7, 16 y 17 de la cinta que lleva el número de carrete 13370 y tiene etiquetas modelo IBM.

```
//..... JOB ....
//PAS01 EXEC PROC=FTG1CLG
//FORT.SYSIN DD DSN=SIMBOLIC.CNEA(DISTANG),DISP=SHR
/*
//GO.FT15F001 DD DSN=DISDAT1A,UNIT=TAPE,VOL=SER=T13370,LABEL=(4,SL),
// DISP=(OLD,PASS)
//GO.FT15F002 DD DSN=DISDAT1B,UNIT=TAPE,VOL=REF=*.GO.FT15F001,
// LABEL=(7,SL)
//GO.FT15F003 DD DSN=DISDAT1C,UNIT=TAPE,VOL=REF=*.GO.FT15F001,
// LABEL=(16,SL)
//GO.FT15F004 DD DSN=DISDAT1D,UNIT=TAPE,VOL=REF=*.GO.FT15F001,
```

```
// LABEL=(17,SL),DISP=(OLD,KEEP)
/*
//
```

NOTA: Los archivos de entrada de datos están definidos en las sentencias GO, en donde se les asigna el mismo número de referencia 15 y los números de secuencia 1, 2 3 y 4 respectivamente. Por lo tanto en el programa fuente, cuando se lean datos de esos archivos se codificará una sentencia READ de este tipo:

READ(15,n°de formato,END=dirección). La "dirección" es la marca de la sentencia a la cual deberá saltar el programa para continuar la ejecución, cada vez que llega al final de uno de los archivos.

Modelo 408 . Ejecución de un programa grabado en la biblioteca con el nombre SUTIL181, cuyos datos de entrada son dos archivos que residen en dos cintas magnéticas distintas. En el rótulo 47 de la cinta que lleva el número de carrete 65500 están grabados, con el nombre SUTIDAT1, los datos de entrada del primer archivo y en el rótulo 35 de la cinta que lleva el número de carrete 08971 están grabados, con el nombre SUTIDAT2, los datos de entrada del segundo archivo. Ambas cintas tienen etiquetas modelo IBM.

```
//..... JOB ....
//PAS01 EXEC PROC=FTG1CLG
//FORT.SYSIN DD DSN=SIMBOLIC.CNEA(SUTIL181),DISP=SHR
/*
//GO.FT13F001 DD DSN=SUTIDAT1,UNIT=TAPE,VOL=SER=T65500,LABEL=(47,SL),
// DISP=(OLD,KEEP)
//GO.FT11F001 DD DSN=SUTIDAT2,UNIT=TAPE,VOL=SER=T08971,LABEL=(35,SL),
// DISP=(OLD,KEEP)
/*
//
```

NOTA: Los archivos de entrada de datos están definidos en las sentencias GO, en donde se les asignan los números de referencia 13 y 11; por lo tanto en el programa fuente, cuando se lean datos de esos archivos, se codificará READ13 y READ11 respectivamente.

PUNTO 3.

Ejecución de módulos procedentes de FORTRAN, grabados en una biblioteca particionada y catalogada de módulos de carga, de nombre REALES.CNEA, que utilizan cintas magnéticas como archivos de entrada de datos.

Modelo 409 . Ejecución de un módulo de carga grabado en la biblioteca con el nombre HELP, cuyos datos de entrada constituyen un archivo que reside, con el nombre DATHELP, en el rótulo 25 de una cinta magnética que tiene etiquetas modelo IBM y lleva el número de carrete 53281.

```
//..... JOB ....
//PASO1 EXEC PGM=HELP
//STEPLIB DD DSN=REALES.CNEA,DISP=SHR
/*
//FT06F001 DD SYSOUT=A
//FT08F001 DD UNIT=TAPE,DSN=DATHELP,VOL=SER=T53281,LABEL=(25,SL),
// DISP=(OLD,KEEP)
/*
//
```

NOTA: El archivo de entrada de datos está definido en la sentencia FT08F001, en donde se le asigna el número de referencia 8; por lo tanto en el programa fuente que dió origen al módulo, se codificó READ8 para leer datos de ese archivo.

Modelo 410 . Ejecución de un módulo de carga grabado en la biblioteca con el nombre GRATIN, cuyos datos de entrada son tres archivos distintos que residen en rótulos consecutivos o no consecutivos de una cinta magnética. Los archivos de nombres GRADAT1, GRADAT2 y GRADAT3 están grabados en los rótulos 9, 10 y 21 de la cinta que lleva el número de carrete 71326 y tiene etiquetas modelo IBM.

```
//..... JOB ....
//PASO1 EXEC PGM=GRATIN
//STEPLIB DD DSN=REALES.CNEA,DISP=SHR
/*
//FT06F001 DD SYSOUT=A
//FT12F001 DD DSN=GRADAT1,UNIT=TAPE,VOL=SER=T71326,LABEL=(9,SL),
// DISP=(OLD,PASS)
//FT12F002 DD DSN=GRADAT2,UNIT=TAPE,VOL=REF=*.FT12F001,LABEL=(10,SL)
//FT12F003 DD DSN=GRADAT3,UNIT=TAPE,VOL=REF=*.FT12F001,LABEL=(21,SL),
// DISP=(OLD,KEEP)
/*
//
```

NOTA: Los archivos de entrada de datos están definidos en la sentencia FT12F001 y siguientes, en donde se les asigna el mismo número de referencia 12 y los números de secuencia 1, 2 y 3 respectivamente. Por lo tanto en el programa fuente que dió origen al módulo, para leer datos de esos archivos se codificó una sentencia READ de este tipo: READ(12,nº de formato,END=dirección). La "dirección" es la marca de la sentencia a la cual deberá saltar el programa para continuar la ejecución, cada vez que llega al final de uno de los archivos.

Modelo 411 . Ejecución de un módulo de carga grabado en la biblioteca con el nombre ONMAS, cuyos datos de entrada son dos archivos que residen en dos cintas magnéticas distintas. En el rótulo 16 de la cinta que lleva el número de carrete 75963, está grabado el primer archivo con el nombre DATOSON1 y en el rótulo 7 de la cinta que lleva el número de carrete 36060, está grabado el segundo archivo con el nombre DATOSON2. Ambas cintas tienen etiquetas modelo IBM.

```
//..... JOB ....
//PAS01 EXEC PGM=ONMAS
//STEPLIB DD DSN=REALES.CNEA,DISP=SHR
/*
//FT06F001 DD SYSOUT=A
//FT12F001 DD UNIT=TAPE,DSN=DATOSON1,VOL=SER=T75963,LABEL=(16,SL),
//  DISP=(OLD,KEEP)
//FT13F001 DD UNIT=TAPE,DSN=DATOSON2,VOL=SER=T36060,LABEL=(7,SL),
//  DISP=(OLD,KEEP)
/*
//
```

NOTA: Los archivos de entrada de datos están definidos en las sentencias FT12F001 y FT13F001, en donde se les asignan los números de referencia 12 y 13; por lo tanto en el programa fuente que dió origen al módulo, se codificó READ12 y READ13 respectivamente, para leer datos de esos archivos.

PUNTO 4.

Ejecución de programas en FORTRAN grabados en cintas magnéticas, que utilizan cintas como archivos de entrada de datos.

Modelo 412 . Ejecución de un programa en lenguaje simbólico FORTRAN grabado, con el nombre SUPERFIC, en el rótulo 3 de una cinta magnética y cuyos datos de entrada constituyen un archivo que reside, con el nombre SUPERDAT, en el rótulo 28 de esa misma cinta. Esta lleva el número de carrete 66156 y tiene etiquetas modelo IBM.

```
//..... JOB ....
//PAS01 EXEC PROC=FTG1CLG
//FORT.SYSIN DD DSN=SUPERFIC,UNIT=TAPE,VOL=SER=T66156,LABEL=(3,SL),
// DISP=(OLD,PASS)
/*
//GO.FT09F001 DD DSN=SUPERDAT,UNIT=TAPE,VOL=SER=T66156,LABEL=(28,SL),
// DISP=(OLD,KEEP)
/*
//
```

NOTA; El archivo de entrada de datos está definido en la sentencia GO, en donde se le asigna el número de referencia 9; por lo tanto en el programa fuente, cuando se lean datos de ese archivo, se codificará READ9.

Modelo 413 . Ejecución de un programa en lenguaje absoluto de FORTRAN grabado, con el nombre PLAN, en el rótulo 20 de una cinta magnética y cuyos datos de entrada constituyen un archivo que reside, con el nombre DATOPLAN, en el rótulo 25 de esa misma cinta. Esta lleva el número de carrete 03317 y tiene etiquetas modelo IBM.

```
//..... JOB ....
//PAS01 EXEC PROC=FTG1LG
//LKED.SYSIN DD DSN=PLAN,UNIT=TAPE,VOL=SER=T03317,LABEL=(20,SL),
// DISP=(OLD,PASS)
/*
//GO.FT08F001 DD DSN=DATOPLAN,UNIT=TAPE,VOL=SER=T03317,LABEL=(25,SL),
// DISP=(OLD,KEEP)
/*
//
```

NOTA: El archivo de entrada de datos está definido en la sentencia GO, en donde se le asigna el número de referencia 8; por lo tanto en el programa fuente que

dió origen al módulo PLAN grabado en la cinta, para leer datos de ese archivo se codificó READ8.

Modelo 414 . Ejecución de un programa en lenguaje simbólico FORTRAN grabado, con el nombre PGMCINTA, en el rótulo 6 de una cinta magnética que lleva el número de carrete 00288 y cuyos datos de entrada constituyen un archivo que reside, con el nombre DATCINTA, en el rótulo 13 de otra cinta que lleva el número de carrete 00300. Ambas cintas tienen etiquetas modelo IBM.

```
//..... JOB ....
//PAS01 EXEC PROC=FTG1CLG
//FORT.SYSIN DD DSN=PGMCINTA,UNIT=TAPE,VOL=SER=T00288,LABEL=(6,SL),
// DISP=(OLD,KEEP)
/*
//GO.FT18F001 DD DSN=DATCINTA,UNIT=TAPE,VOL=SER=T00300,LABEL=(13,SL),
// DISP=(OLD,KEEP)
/*
//
```

NOTA: El archivo de entrada de datos está definido en la sentencia GO, en donde se le asigna el número de referencia 18; por lo tanto en el programa fuente, cuando se lean datos de ese archivo, se codificará READ18.

Modelo 415 . Ejecución de un programa en lenguaje absoluto de FORTRAN grabado, con el nombre ABSCINTA, en el rótulo 16 de una cinta magnética que lleva el número de carrete 99143 y cuyos datos de entrada constituyen un archivo que reside, con el nombre ARCHCIN, en el rótulo 7 de otra cinta que lleva el número de carrete 99500. Ambas cintas tienen etiquetas modelo IBM.

```
//..... JOB ....
//PAS01 EXEC PROC=FTG1LG
//LKED.SYSIN DD DSN=ABSCINTA,UNIT=TAPE,VOL=SER=T99143,LABEL=(16,SL),
// DISP=(OLD,KEEP)
/*
//GO.FT17F001 DD DSN=ARCHCIN,UNIT=TAPE,VOL=SER=T99500,LABEL=(7,SL),
// DISP=(OLD,KEEP)
/*
//
```

NOTA: El archivo de entrada de datos está definido en la sentencia GO, en donde se le asigna el número de referencia 17; por lo tanto en el programa fuente que

dió origen al módulo grabado en la cinta, para leer datos de ese archivo se codificó READ17.

Modelo 416 . Ejecución de un programa en lenguaje simbólico FORTRAN grabado, con el nombre FORSEC en el rótulo 14 de una cinta magnética y cuyos datos de entrada son tres archivos distintos que residen en rótulos consecutivos o no consecutivos de esa misma cinta. Los archivos de nombres DATTAPE1, DATTAPE2 y DATTAPE3 están grabados en los rótulos 21, 24 y 25 de la cinta que lleva el número de carrete 16016 y tiene etiquetas modelo IBM.

```
//..... JOB ....
//PAS01 EXEC PROC=FTG1CLG
//FORT.SYSIN DD DSN=FORSEC,UNIT=TAPE,VOL=SER=T16016,LABEL=(14,SL),
// DISP=(OLD,PASS)
/*
//GO.FT16F001 DD DSN=DATTAPE1,UNIT=TAPE,VOL=SER=T16016,LABEL=(21,SL),
// DISP=(OLD,PASS)
//GO.FT16F002 DD DSN=DATTAPE2,UNIT=TAPE,VOL=REF=*.GO.FT16F001,
// LABEL=(24,SL)
//GO.FT16F003 DD DSN=DATTAPE3,UNIT=TAPE,VOL=REF=*.GO.FT16F001,
// LABEL=(25,SL),DISP=(OLD,KEEP)
/*
//
```

NOTA: Los archivos de entrada de datos están definidos en las sentencias GO, en donde se les asigna el mismo número de referencia 16 y los números de secuencia 1, 2 y 3 respectivamente. Por lo tanto en el programa fuente, cuando se lean datos de esos archivos se codificará una sentencia READ de este tipo:

READ(16,n°de formato,END=dirección). La "dirección" es la marca de la sentencia a la cual deberá saltar el programa para continuar la ejecución, cada vez que llega al final de uno de los archivos.

Modelo 417 . Ejecución de un programa en lenguaje absoluto de FORTRAN grabado, con el nombre METHOD02 en el rótulo 16 de una cinta magnética y cuyos datos de entrada son cuatro archivos distintos que residen en rótulos consecutivos o no consecutivos de esa misma cinta. Los archivos, de nombres METODAT1, METODAT2, METODAT3 y METODAT4, están grabados en los rótulos 19, 20, 25 y 28 de la cinta que lleva el número de carrete 17722 y tiene etiquetas modelo IBM.


```
//..... JOB ....
//PAS01 EXEC PROC=FTG1LG
//LKED.SYSIN DD DSN=METODO2,UNIT=TAPE,VOL=SER=T17722,LABEL=(16,SL),
// DISP=(OLD,PASS)
/*
//GO.FT12F001 DD DSN=METODAT1,UNIT=TAPE,VOL=SER=T17722,LABEL=(19,SL),
// DISP=(OLD,PASS)
//GO.FT12F002 DD DSN=METODAT2,UNIT=TAPE,VOL=REF=*.GO.FT12F001,
// LABEL=(20,SL)
//GO.FT12F003 DD DSN=METODAT3,UNIT=TAPE,VOL=REF=*.GO.FT12F001,
// LABEL=(25,SL)
//GO.FT12F004 DD DSN=METODAT4,UNIT=TAPE,VOL=REF=*.GO.FT12F001,
// LABEL=(28,SL),DISP=(OLD,KEEP)
/*
//
```

NOTA: Los archivos de entrada de datos están definidos en las sentencias GO, en donde se les asigna el mismo número de referencia 12 y los números de secuencia 1, 2, 3 y 4 respectivamente. Por lo tanto en el programa fuente que dió origen al módulo grabado en la cinta, para leer datos de esos archivos se codificó una sentencia READ de este tipo: READ(12,nº formato,END=dirección). La "dirección" es la marca de la sentencia a la cual deberá saltar el programa para continuar la ejecución, cada vez que llega al final de uno de los archivos.

Modelo 418 . Ejecución de un programa en lenguaje simbólico FORTRAN grabado, con el nombre ICAN, en el rótulo 8 de una cinta magnética que lleva el número de carrete 07177 y cuyos datos de entrada son cuatro archivos distintos que residen en rótulos consecutivos o no consecutivos de otra cinta. Los archivos, de nombres ICANDAT1, ICANDAT2, ICANDAT3 e ICANDAT4, están grabados en los rótulos 13, 14, 19 y 23 de la cinta que lleva el número de carrete 08188. Ambas cintas tienen etiquetas modelo IBM.

```
//..... JOB ....
//PAS01 EXEC PROC=FTG1CLG
//FORT.SYSIN DD DSN=ICAN,UNIT=TAPE,VOL=SER=T07177,LABEL=(8,SL),
// DISP=(OLD,KEEP)
/*
//GO.FT03F001 DD DSN=ICANDAT1,UNIT=TAPE,VOL=SER=T08188,LABEL=(13,SL),
// DISP=(OLD,PASS)
//GO.FT03F002 DD DSN=ICANDAT2,UNIT=TAPE,VOL=REF=*.GO.FT03F001,
// LABEL=(14,SL)
//GO.FT03F003 DD DSN=ICANDAT3,UNIT=TAPE,VOL=REF=*.GO.FT03F001,
// LABEL=(19,SL)
```

```
//GO.FT03F004 DD DSN=ICANDAT4,UNIT=TAPE,VOL=REF=*.GO.FT03F001,
// LABEL=(23,SL),DISP=(OLD,KEEP)
/*
//
```

NOTA: Los archivos de entrada de datos están definidos en las sentencias GO, en donde se les asigna el mismo número de referencia 3 y los números de secuencia 1, 2, 3 y 4 respectivamente. Por lo tanto en el programa fuente, cuando se lean datos de esos archivos se codificará una sentencia READ de este tipo:

READ(3,nº de formato,END=dirección). La "dirección" es la marca de la sentencia a la cual deberá saltar el programa para continuar la ejecución, cada vez que llega al final de uno de los archivos.

Modelo 419 . Ejecución de un programa en lenguaje absoluto de FORTRAN grabado, con el nombre POLIMERO, en el rótulo 16 de una cinta magnética que lleva el número de carrete 21121 y cuyos datos de entrada son cuatro archivos distintos que residen en rótulos consecutivos o no consecutivos de otra cinta. Los archivos, de nombres POLIDAT1, POLIDAT2, POLIDAT3 y POLIDAT4 están grabados en los rótulos 11, 17, 31 y 32 de la cinta que lleva el número de carrete 30007. Ambas cintas tienen etiquetas modelo IBM.

```
//..... JOB ....
//PAS01 EXEC PROC=FTG1LG
//LKED.SYSIN DD DSN=POLIMERO,UNIT=TAPE,VOL=SER=T21121,LABEL=(16,SL),
// DISP=(OLD,KEEP)
/*
//GO.FT18F001 DD DSN=POLIDAT1,UNIT=TAPE,VOL=SER=T30007,LABEL=(11,SL),
// DISP=(OLD,PASS)
//GO.FT18F002 DD DSN=POLIDAT2,UNIT=TAPE,VOL=REF=*.GO.FT18F001,
// LABEL=(17,SL)
//GO.FT18F003 DD DSN=POLIDAT3,UNIT=TAPE,VOL=REF=*.GO.FT18F001,
// LABEL=(31,SL)
//GO.FT18F004 DD DSN=POLIDAT4,UNIT=TAPE,VOL=REF=*.GO.FT18F001,
// LABEL=(32,SL),DISP=(OLD,KEEP)
/*
//
```

NOTA: Los archivos de entrada de datos están definidos en la sentencia GO, en donde se les asigna el mismo número de referencia 18 y los números de secuencia 1, 2, 3 y 4 respectivamente. Por lo tanto en el programa fuente que dió origen al

módulo grabado en la cinta, para leer datos de esos archivos se codificó una sentencia READ de este tipo: READ(18,nº de formato,END=dirección). La "dirección" es la marca de la sentencia a la cual deberá saltar el programa para continuar la ejecución, cada vez que llega al final de uno de los archivos.

CAPITULO X

MODELOS PARA LA EJECUCION DE PROGRAMAS QUE
UTILIZAN CINTAS MAGNETICAS COMO ARCHIVOS DE
SALIDA DE DATOS

PUNTO 1.

Ejecución de programas en FORTRAN que se presentan en tarjetas perforadas y utilizan cintas magnéticas como archivos de salida de datos.

Modelo 420 . Ejecución de un programa en tarjetas perforadas en lenguaje simbólico FORTRAN, cuyos resultados constituyen un archivo de salida que se graba, con el nombre SALIDA25, en el rótulo 40 de una cinta magnética que lleva el número de carrete 16606 y tiene etiquetas modelo IBM.

```
//..... JOB ....
//PAS01 EXEC PROC=FTG1CLG
//FORT.SYSIN DD *
  Aquí tarjetas perforadas en FORTRAN
/*
//GO.FT12F001 DD DSN=SALIDA25,UNIT=TAPE,VOL=SER=T16606,LABEL=(40,SL),
//  DISP=(NEW,KEEP),DCB=(BLKSIZE=2000,LRECL=80,RECFM=FB)
//GO.SYSIN DD *
  Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: El archivo de entrada de datos está definido en la primera sentencia GO, en donde se le asigna el número de referencia 12; por consiguiente en el programa fuente, cuando se graben resultados en ese archivo, se codificará WRITE12. Se menciona el DCB con el cual se grabará la cinta.

Modelo 421 . Ejecución de un programa en tarjetas perforadas en lenguaje absoluto de FORTRAN, cuyos resultados constituyen un archivo de salida que se graba, con el nombre RESUL, en el rótulo 29 de una cinta magnética que lleva el número de carrete 35192 y tiene etiquetas modelo IBM.

```
//..... JOB ....
//PAS01 EXEC PROC=FTG1LG
//LKED.SYSIN DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas en absoluto de FORTRAN
/*
//GO.FT20F001 DD DSN=RESUL,UNIT=TAPE,VOL=SER=T35192,DISP=(NEW,KEEP),
//  LABEL=(29,SL),DCB=(BLKSIZE=1600,LRECL=80,RECFM=FB)
//GO.SYSIN DD *
  Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: El archivo de entrada de datos está definido en la primera sentencia GO, en donde se le asigna el número de referencia 20; por consiguiente en el programa fuente que dió origen al lote de tarjetas perforadas en absoluto, para grabar resultados en ese archivo se codificó WRITE20. Se menciona el DCB con el cual se grabará la cinta.

Modelo 422 . Ejecución de un programa en tarjetas perforadas en lenguaje simbólico FORTRAN, cuyos resultados son cuatro archivos de salida distintos que se graban con los nombres COMPRES1, COMPRES2, COMPRES3 y COMPRES4, en rótulos consecutivos de una cinta magnética. La grabación se realiza en los rótulos 7, 8, 9 y 10 de la cinta que lleva el número de carrete 53321 y tiene etiquetas modelo IBM.

```
//..... JOB ....
//PAS01 EXEC PROC=FTG1CLG
//FORT.SYSIN DD *
  Aquí tarjetas perforadas en FORTRAN
/*
//GO.FT12F001 DD DSN=COMPRES1,UNIT=TAPE,VOL=SER=T53321,LABEL=(7,SL),
//  DISP=(NEW,PASS),DCB=(BLKSIZE=800,LRECL=80,RECFM=FB)
//GO.FT12F002 DD DSN=COMPRES2,UNIT=TAPE,VOL=REF=*.GO.FT12F001,
//  LABEL=(8,SL)
//GO.FT12F003 DD DSN=COMPRES3,UNIT=TAPE,VOL=REF=*.GO.FT12F001,
//  LABEL=(9,SL)
//GO.FT12F004 DD DSN=COMPRES4,UNIT=TAPE,VOL=REF=*.GO.FT12F001,
//  LABEL=(10,SL),DISP=(NEW,KEEP)
//GO.SYSIN DD *
  Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: Los archivos de salida de datos están definidos en las cuatro primeras sentencias GO, en donde se les asigna el mismo número de referencia 12 y los números de secuencia 1, 2, 3 y 4 respectivamente. Por lo tanto en el programa fuente, cuando se graben resultados en esos archivos se codificará una sentencia WRITE12, además de la sentencia ENDFILE12 para señalar la finalización de cada uno de ellos. Se menciona el DCB con el cual se grabará la cinta.

Modelo 423 . Ejecución de un programa en tarjetas perforadas en lenguaje absoluto de FORTRAN, cuyos resultados son tres archivos de salida distintos que se graban, con los nombres ABSORES1, ABSORES2 y ABSORES3, en rótulos consecutivos de una

cinta magnética. La grabación se realiza en los r tulos 13, 14 y 15 de la cinta que lleva el n mero de carrete 28003 y tiene etiquetas modelo IBM.

```
//..... JOB ....
//PAS01 EXEC PROC=FTG1LG
//LKED.SYSIN DD *,DCB=BLKSIZE=80
  Aqu  tarjetas perforadas en absoluto de FORTRAN
/*
//GO.FT14F001 DD DSN=ABSORES1,UNIT=TAPE,VOL=SER=T28003,LABEL=(13,SL),
//  DISP=(NEW,PASS),DCB=(BLKSIZE=1600,LRECL=80,RECFM=FB)
//GO.FT14F002 DD DSN=ABSORES2,UNIT=TAPE,VOL=REF=*.GO.FT14F001,
//  LABEL=(14,SL)
//GO.FT14F003 DD DSN=ABSORES3,UNIT=TAPE,VOL=REF=*.GO.FT14F001,
//  LABEL=(15,SL),DISP=(NEW,KEEP)
//GO.SYSIN DD *
  Aqu  tarjetas perforadas de DATOS
/*
//
```

NOTA: Los archivos de salida de datos est n definidos en las tres primeras sentencias GO, en donde se les asigna el mismo n mero de referencia 14 y los n meros de secuencia 1, 2 y 3 respectivamente. Por lo tanto en el programa fuente que di  origen al lote de tarjetas perforadas en absoluto, para grabar resultados en esos archivos se codific  una sentencia WRITE14, adem s de la sentencia ENDFILE14 para se alar la finalizaci n de cada uno de ellos. Se menciona el DCB con el cual se grabar  la cinta.

Modelo 424 . Ejecuci n de un programa en tarjetas perforadas en lenguaje simb lico FORTRAN, cuyos resultados son dos archivos de salida distintos que se graban en sendas cintas magn ticas. El primer archivo se graba, con el nombre VALOR1, en el r tulo 7 de una cinta que lleva el n mero de carrete 11125 y el segundo, con el nombre VALOR2, en el r tulo 16 de otra cinta que lleva el n mero de carrete 11133. Ambas cintas tienen etiquetas modelo IBM.

```
//..... JOB ....
//PAS01 EXEC PROC=FTG1CLG
//FORT.SYSIN DD *
  Aqu  tarjetas perforadas en FORTRAN
/*
//GO.FT20F001 DD DSN=VALOR1,UNIT=TAPE,VOL=SER=T11125,LABEL=(7,SL),
//  DISP=(NEW,KEEP),DCB=(BLKSIZE=800,LRECL=80,RECFM=FB)
//GO.FT18F001 DD DSN=VALOR2,UNIT=TAPE,VOL=SER=T11133,LABEL=(16,SL),
```

```
// DISP=(NEW,KEEP),DCB=(BLKSIZE=800,LRECL=80,RECFM=FB)
//GO.SYSIN DD *
Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: Los archivos de salida de datos están definidos en las dos primeras sentencias GO, en donde se les asignan los números de referencia 20 y 18; por lo tanto en el programa fuente, cuando se graben resultados en esos archivos, se codificará WRITE20 y WRITE18 respectivamente. Se mencionan los DCB con los cuales se grabarán las cintas.

Modelo 425 . Ejecución de un programa en tarjetas perforadas en lenguaje absoluto de FORTRAN, cuyos resultados son dos archivos de salida distintos que se graban en sendas cintas magnéticas. El primer archivo se graba, con el nombre SOLAR3 en el rótulo 30 de una cinta que lleva el número de carrete 07800 y el segundo, con el nombre PANTALLA, en el rótulo 13 de otra cinta que lleva el número de carrete 07866. Ambas cintas tienen etiquetas modelo IBM.

```
//..... JOB ....
//PAS01 EXEC PROC=FTG1LG
//LKED.SYSIN DD *,DCB=BLKSIZE=80
Aquí tarjetas perforadas en absoluto de FORTRAN
/*
//GO.FT12F001 DD DSN=SOLAR3,UNIT=TAPE,VOL=SER=T07800,LABEL=(30,SL),
// DISP=(NEW,KEEP),DCB=(BLKSIZE=2400,LRECL=80,RECFM=FB)
//GO.FT15F001 DD DSN=PANTALLA,UNIT=TAPE,VOL=SER=T07866,LABEL=(13,SL),
// DISP=(NEW,KEEP),DCB=(BLKSIZE=2000,LRECL=80,RECFM=FB)
//GO.SYSIN DD *
Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: Los archivos de salida de datos están definidos en las dos primeras sentencias GO, en donde se les asignan los números de referencia 12 y 15; por lo tanto en el programa fuente que dió origen al lote de tarjetas perforadas en absoluto, para grabar resultados en esos archivos, se codificó WRITE12 y WRITE15 respectivamente. Se mencionan los DCB con los cuales se grabarán las cintas.

PUNTO 2.

Ejecución de programas en FORTRAN que se presentan en tarjetas perforadas, cuyos datos de entrada residen en una biblioteca particionada y catalogada de programas en lenguaje simbólico, de nombre SIMBOLIC.CNEA y cuyos resultados se graban en cintas magnéticas.

Modelo 426. Ejecución de un programa en tarjetas perforadas en lenguaje simbólico FORTRAN, cuyos datos de entrada constituyen un archivo que reside, con el nombre DATOTEK, en la biblioteca SIMBOLIC.CNEA y cuyos resultados configuran un archivo de salida que se graba en una cinta magnética. El problema se puede resolver en dos pasos: en el PASO1 el utilitario IEBGENER transfiere el archivo de entrada de datos de la biblioteca a un disco auxiliar; en el PASO2 el procedimiento FTG1CLG ejecuta el programa obteniendo los datos del disco auxiliar y grabando los resultados, con el nombre RESULTEK en el rótulo 19 de una cinta que lleva el número de carrete 06405 y tiene etiquetas modelo IBM.

```
//..... JOB ....
//PASO1 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA(DATOTEK),DISP=SHR
//SYSUT2 DD DSN=DATOTEK,UNIT=WORKDISK,DISP=(NEW,PASS),
//  SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//SYSIN DD DUMMY
/*
//PASO2 EXEC PROC=FTG1CLG
//FORT.SYSIN DD *
  Aquí tarjetas perforadas en FORTRAN
/*
//GO.FT08F001 DD DSN=DATOTEK,UNIT=WORKDISK,DISP=(OLD,DELETE)
//GO.FT13F001 DD DSN=RESULTEK,UNIT=TAPE,VOL=SER=T06405,LABEL=(19,SL),
//  DISP=(NEW,KEEP),DCB=(BLKSIZE=800,LRECL=80,RECFM=FB)
/*
//
```

NOTA: El archivo de entrada de datos está definido en la primera sentencia GO, en donde se le asigna el número de referencia 8 y el archivo de salida de datos está definido en la segunda sentencia GO, en donde se le asigna el número de referencia 13. Por lo tanto en el programa fuente, cuando se lean datos del primer archivo se codificará READ8 y cuando se graben resultados en el segundo archivo se

codificará WRITE13. Se mencionan los DCB con los cuales se grabarán el disco auxiliar y la cinta.

Modelo 427 . Ejecución de un programa en tarjetas perforadas en lenguaje absoluto de FORTRAN, cuyos datos de entrada constituyen un archivo que reside, con el nombre DATOMEGA, en la biblioteca SIMBOLIC.CNEA y cuyos resultados configuran un archivo de salida que se graba en una cinta magnética. El problema se puede resolver en dos pasos: en el PASO1 el utilitario IEBGENER transfiere el archivo de entrada de datos de la biblioteca a un disco auxiliar; en el PASO2 el procedimiento FTG1LG ejecuta el programa obteniendo los datos del disco auxiliar y grabando los resultados, con el nombre RESULOME en el rótulo 26 de una cinta que lleva el número de carrete 13782 y tiene etiquetas modelo IUM.

```
//..... JOB ....
//PASO1 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA(DATOMEGA),DISP=SHR
//SYSUT2 DD DSN=DATOMEGA,UNIT=WORKDISK,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//SYSIN DD DUMMY
/*
//PASO2 EXEC PROC=FTG1LG
//LKED.SYSIN DD *,DCB=BLKSIZE=80
Aquí tarjetas perforadas en absoluto de FORTRAN
/*
//GO.FT20F001 DD DSN=DATOMEGA,UNIT=WORKDISK,DISP=(OLD,DELETE)
//GO.FT12F001 DD DSN=RESULOME,UNIT=TAPE,VOL=SER=T13782,LABEL=(26,SL),
// DISP=(NEW,KEEP),DCB=(BLKSIZE=2000,LRECL=80,RECFM=FB)
/*
//
```

NOTA: El archivo de entrada de datos está definido en la primera sentencia GO, en donde se le asigna el número de referencia 20 y el archivo de salida de datos está definido en la segunda sentencia GO, en donde se le asigna el número de referencia 12. Por lo tanto en el programa fuente que dió origen al lote de tarjetas perforadas en absoluto, se codificó READ20 para leer datos del primer archivo y WRITE12 para grabar resultados en el segundo archivo. Se mencionan los DCB con los cuales se grabarán el disco auxiliar y la cinta.

Modelo 428 . Ejecución de un programa en tarjetas perforadas en lenguaje simbó-

lico FORTRAN, cuyos datos de entrada son dos archivos distintos que residen en la biblioteca SIMBOLIC.CNEA, como miembros independientes, con los nombres JULIDAT1 y JULIDAT2 y cuyos resultados son tres archivos distintos que se graban en r tulos consecutivos de una cinta magn tica. Con los nombres JULIRES1, JULIRES2 y JULIRES3, los archivos de salida se graban en los r tulos 3, 4 y 5 de la cinta que lleva el n mero de carrete 51732 y tiene etiquetas modelo IBM. El problema se puede resolver en tres pasos: en los dos primeros (PASO1 y PASO2) el utilitario IEBGENER transfiere, de a uno, los dos archivos de entrada de datos de la biblioteca al mismo disco auxiliar; en el PASO3 el procedimiento FTG1CLG ejecuta el programa obteniendo los datos de entrada del disco auxiliar y grabando los resultados en la cinta.

```
//..... JOB ....
//PASO1 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA(JULIDAT1),DISP=SHR
//SYSUT2 DD UNIT=WORKDISK,DISP=(NEW,PASS),SPACE=(CYL,(1,1)),
// DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB),DSN=JULIDAT1
//SYSIN DD DUMMY
/*
//PASO2 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA(JULIDAT2),DISP=SHR
//SYSUT2 DD VOL=REF=*.PASO1.SYSUT2,DISP=(NEW,PASS),
// DSN=JULIDAT2,SPACE=(CYL,(1,1))
//SYSIN DD DUMMY
/*
//PASO3 EXEC PROC=FTG1CLG
//FORT.SYSIN DD *
Aqu  tarjetas perforadas en FORTRAN
/*
//GO.FT09F001 DD UNIT=WORKDISK,DSN=JULIDAT1,DISP=(OLD,DELETE)
//GO.FT10F001 DD UNIT=WORKDISK,DSN=JULIDAT2,DISP=(OLD,DELETE)
//GO.FT12F001 DD DSN=JULIRES1,UNIT=TAPE,VOL=SER=T51732,LABEL=(3,SL),
// DISP=(NEW,PASS),DCB=(BLKSIZE=800,LRECL=80,RECFM=FB)
//GO.FT12F002 DD DSN=JULIRES2,UNIT=TAPE,VOL=REF=*.GO.FT12F001,
// LABEL=(4,SL)
//GO.FT12F003 DD DSN=JULIRES3,UNIT=TAPE,VOL=REF=*.GO.FT12F001,
// LABEL=(5,SL),DISP=(NEW,KEEP)
/*
//
```

NOTA: Los archivos de entrada de datos est n definidos en las dos primeras sentencias GO, en donde se les asignan los n meros de referencia 9 y 10; por lo tanto

en el programa fuente, cuando se lean datos de esos archivos se codificará READ9 y READ10 respectivamente. A su vez los archivos de salida de datos están definidos en la tercera sentencia GO y siguientes, en donde se les asigna el mismo número de referencia 12 y los números de secuencia 1, 2 y 3 respectivamente. Por consiguiente en el programa fuente, cuando se graben resultados en esos archivos se codificará WRITE12, además de la sentencia ENDFILE12 para señalar la finalización de cada uno de ellos. Se mencionan los DCB con los cuales se grabará el disco auxiliar y la cinta.

Modelo 429 . Ejecución de un programa en tarjetas perforadas en lenguaje absoluto de FORTRAN, cuyos datos de entrada son dos archivos que residen en la biblioteca SIMBOLIC.CNEA, como miembros independientes, con los nombres SPINDAT1 y SPINDAT2 y cuyos resultados son cuatro archivos distintos que se graban en rótulos consecutivos de una cinta magnética. Con los nombres SPINRES1, SPINRES2, SPINRES3 y SPINRES4, los archivos de salida se graban en los rótulos 23, 24, 25 y 26 de la cinta que lleva el número de carrete 15166 y tiene etiquetas modelo IBM. El problema se puede resolver en tres pasos: en los dos primeros (PASO1 y PASO2) el utilitario IEBGENER transfiere, de a uno, los dos archivos de entrada de la biblioteca al mismo disco auxiliar; en el PASO3 el procedimiento FTG1LG ejecuta el programa obteniendo los datos de entrada del disco auxiliar y grabando los resultados en la cinta.

```
//..... JOB ..
//PASO1 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA(SPINDAT1),DISP=SHR
//SYSUT2 DD DSN=SPINDAT1,UNIT=WORKDISK,DISP=(NEW,PASS),
//  SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//SYSIN DD DUMMY
/*
//PASO2 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA(SPINDAT2),DISP=SHR
//SYSUT2 DD DSN=SPINDAT2,VOL=REF=*.PASO1.SYSUT2,DISP=(NEW,PASS),
//  SPACE=(CYL,(1,1))
//SYSIN DD DUMMY
/*
```

```
//PAS03 EXEC PROC=FTGILG
//LKED.SYSIN DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas en absoluto de FORTRAN
/*
//GO.FT03F001 DD UNIT=WORKDISK,DSN=SPINDAT1,DISP=(OLD,DELETE)
//GO.FT02F001 DD UNIT=WORKDISK,DSN=SPINDAT2,DISP=(OLD,DELETE)
//GO.FT19F001 DD DSN=SPINRES1,UNIT=TAPE,VOL=SER=T15166,LABEL=(23,SL),
//  DISP=(NEW,PASS),DCB=(BLKSIZE=1600,LRECL=80,RECFM=FB)
//GO.FT19F002 DD DSN=SPINRES2,UNIT=TAPE,VOL=REF=*.GO.FT19F001,
//  LABEL=(24,SL)
//GO.FT19F003 DD DSN=SPINRES3,UNIT=TAPE,VOL=REF=*.GO.FT19F001,
//  LABEL=(25,SL)
//GO.FT19F004 DD DSN=SPINRES4,UNIT=TAPE,VOL=REF=*.GO.FT19F001,
//  LABEL=(26,SL),DISP=(NEW,KEEP)
/*
//
```

NOTA: Los archivos de entrada de datos están definidos en las dos primeras sentencias GO, en donde se les asignan los números de referencia 3 y 2 respectivamente; por lo tanto en el programa fuente que dió origen al lote de tarjetas en absoluto, para leer datos de esos archivos se codificó READ3 y READ2. A su vez los archivos de salida de datos están definidos en la tercera sentencia GO y siguientes, en donde se les asigna el mismo número de referencia 19 y los números de secuencia 1, 2, 3 y 4 respectivamente. Por consiguiente en el programa fuente, para grabar resultados en esos archivos se codificó WRITE19, además de la sentencia ENDFILE19 para señalar la finalización de cada uno de ellos. Se mencionan los DCB con los cuales se grabarán el disco auxiliar y la cinta.

Modelo 430 . Ejecución de un programa en tarjetas perforadas en lenguaje simbólico FORTRAN, cuyos datos de entrada son tres archivos distintos que residen en la biblioteca SIMULIC.CNEA, como miembros independientes, con los nombres GRUPO25, GRUPO26 y GRUPO27 y cuyos resultados son dos archivos de salida que se graban en sendas cintas magnéticas. En el rótulo 15 de la cinta que lleva el número de carrete 62000, se graban los resultados del primer archivo con el nombre RESULT12 y en el rótulo 5 de la cinta que lleva el número de carrete 20202, se graban los resultados del segundo archivo con el nombre RESULT13. Ambas cintas tienen etiquetas modelo IBM. El problema se puede resolver en cuatro pasos: en los tres primeros (PAS01, PAS02 y PAS03) el utilitario IEBGENER transfiere, de a

uno, los tres archivos de entrada de la biblioteca al mismo disco auxiliar; en el PAS04 el procedimiento FTG1CLG ejecuta el programa, obteniendo los datos de entrada del disco auxiliar y grabando los resultados en las cintas.

```
//..... JOB ....
//PAS01 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA(GRUPO25),DISP=SHR
//SYSUT2 DD UNIT=WORKDISK,DISP=(NEW,PASS),DSN=GRUPO25,
//  SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//SYSIN DD DUMMY
/*
//PAS02 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA(GRUPO26),DISP=SHR
//SYSUT2 DD VOL=REF=*.PAS01.SYSUT2,DISP=(NEW,PASS),DSN=GRUPO26,
//  SPACE=(CYL,(1,1))
//SYSIN DD DUMMY
/*
//PAS03 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA(GRUPO27),DISP=SHR
//SYSUT2 DD VOL=REF=*.PAS01.SYSUT2,DISP=(NEW,PASS),DSN=GRUPO27,
//  SPACE=(CYL,(1,1))
//SYSIN DD DUMMY
/*
//PAS04 EXEC PROC=FTG1CLG
//FORT.SYSIN DD *
  Aquí tarjetas perforadas en FORTRAN
/*
//GO.FT12F001 DD UNIT=WORKDISK,DSN=GRUPO25,DISP=(OLD,DELETE)
//GO.FT13F001 DD UNIT=WORKDISK,DSN=GRUPO26,DISP=(OLD,DELETE)
//GO.FT11F001 DD UNIT=WORKDISK,DSN=GRUPO27,DISP=(OLD,DELETE)
//GO.FT14F001 DD DSN=RESULT12,UNIT=TAPE,VOL=SER=T62000,LABEL=(15,SL),
//  DISP=(NEW,KEEP),DCB=(BLKSIZE=3200,LRECL=80,RECFM=FB)
//GO.FT18F001 DD DSN=RESULT13,UNIT=TAPE,VOL=SER=T20202,LABEL=(5,SL),
//  DISP=(NEW,KEEP),DCB=(BLKSIZE=1200,LRECL=80,RECFM=FB)
/*
//
```

NOTA: Los archivos de entrada de datos están definidos en las tres primeras sentencias GO, en donde se les asignan los números de referencia 12, 13 y 11; por lo tanto en el programa fuente, cuando se lean datos de esos archivos, se codificará READ12, READ13 y READ11 respectivamente. A su vez los archivos de salida de datos están definidos en la cuarta y quinta sentencia GO, en donde se les asignan los números de referencia 14 y 18; por consiguiente en el programa fuente, cuando se graben resultados en esos archivos, se codificará WRITE14 y WRITE18 respectiva-

mente. Se mencionan los DCB con los cuales se grabarán el disco auxiliar y las cintas.

Modelo 431 . Ejecución de un programa en tarjetas perforadas en lenguaje absoluto de FORTRAN, cuyos datos de entrada son dos archivos distintos que residen en la biblioteca SIMBOLIC.CNEA, como miembros independientes, con los nombres DATO55 y DATO66 y cuyos resultados son dos archivos de salida que se graban en sendas cintas magnéticas. En el rótulo 25 de la cinta que lleva el número de carrete 77713, se graban los resultados del primer archivo con el nombre SALIDA1 y en el rótulo 6 de la cinta que lleva el número de carrete 88815, se graban los resultados del segundo archivo con el nombre SALIDA2. Ambas cintas tienen etiquetas modelo IBM. El problema se puede resolver en tres pasos: en los dos primeros (PASO1 y PASO2) el utilitario IEBGENER transfiere, de a uno, los dos archivos de entrada de la biblioteca al mismo disco auxiliar; en el PASO3 el procedimiento FTG1LG ejecuta el programa, obteniendo los datos de entrada del disco auxiliar y grabando los resultados en las cintas.

```
//..... JOB ....
//PASO1 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA(DATO55),DISP=SHR
//SYSUT2 DD DSN=DATO55,UNIT=WORKDISK,DISP=(NEW,PASS),
//  SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//SYSIN DD DUMMY
/*
//PASO2 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA(DATO66),DISP=SHR
//SYSUT2 DD DSN=DATO66,VOL=REF=*.PASO1.SYSUT2,DISP=(NEW,PASS),
//  SPACE=(CYL,(1,1))
//SYSIN DD DUMMY
/*
//PASO3 EXEC PROC=FTG1LG
//LKED.SYSIN DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas en absoluto de FORTRAN
/*
//GO.FT14FOO1 DD UNIT=WORKDISK,DSN=DATO55,DISP=(OLD,DELETE)
//GO.FT18FOO1 DD UNIT=WORKDISK,DSN=DATO66,DISP=(OLD,DELETE)
//GO.FT03FOO1 DD DSN=SALIDA1,UNIT=TAPE,VOL=SER=T77713,LABEL=(25,SL),
//  DISP=(NEW,KEEP),DCB=(BLKSIZE=1600,LRECL=80,RECFM=FB)
//GO.FT16FOO1 DD DSN=SALIDA2,UNIT=TAPE,VOL=SER=T88815,LABEL=(6,SL),
//  DISP=(NEW,KEEP),DCB=(BLKSIZE=1600,LRECL=80,RECFM=FB)
```

```
/*  
//
```

NOTA: Los archivos de entrada de datos están definidos en las dos primeras sentencias GO, en donde se les asignan los números de referencia 14 y 18; por lo tanto en el programa fuente que dió origen al lote de tarjetas perforadas en absoluto, para leer datos de esos archivos se codificó READ14 y READ18 respectivamente. A su vez los archivos de salida de datos están definidos en la tercera y cuarta sentencia GO, en donde se les asignan los números de referencia 3 y 16; por consiguiente en el programa fuente para grabar resultados en esos archivos se codificó WRITE3 y WRITE16 respectivamente. Se mencionan los DCB con los cuales se grabarán el disco auxiliar y las cintas.

PUNTO 3.

Ejecución de programas en FORTRAN, residentes en una biblioteca particionada y catalogada de programas en lenguaje simbólico, de nombre SIMBOLIC.CNEA, que utilizan cintas magnéticas como archivos de salida de datos.

Modelo 432 . Ejecución de un programa residente en la biblioteca con el nombre JUNTA, cuyos resultados constituyen un archivo de salida de datos que se graba, con el nombre JUNTARES, en el rótulo 12 de una cinta magnética que lleva el número de carrete 04443 y tiene etiquetas modelo IBM.

```
//..... JOB ....
//PAS01 EXEC PROC=FTG1CLG
//FORT.SYSIN DD DSN=SIMBOLIC.CNEA(JUNTA),DISP=SHR
/*
//GO.FT08F001 DD DSN=JUNTARES,UNIT=TAPE,VOL=SER=T04443,LABEL=(12,SL),
// DISP=(NEW,KEEP),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//GO.SYSIN DD *
Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: El archivo de salida de datos está definido en la primera sentencia GO, en donde se le asigna el número de referencia 8; por lo tanto en el programa fuente, cuando se graben los resultados en ese archivo, se codificará WRITE8. Se menciona el DCB con el cual se grabará la cinta.

Modelo 433 . Ejecución de un programa residente en la biblioteca con el nombre GRABADOR, cuyos resultados son cinco archivos de salida distintos que se graban, con los nombres CALCULO1, CALCULO2, CALCULO3, CALCULO4 y CALCULO5 en rótulos consecutivos de una cinta magnética. La grabación se realiza en los rótulos 20, 21, 22, 23 y 24 de la cinta que lleva el número de carrete 85166 y tiene etiquetas modelo IBM.

```
//..... JOB ....
//PAS01 EXEC PROC=FTG1CLG
//FORT.SYSIN DD DSN=SIMBOLIC.CNEA(GRABADOR),DISP=SHR
/*
//GO.FT12F001 DD DSN=CALCULO1,UNIT=TAPE,LABEL=(20,SL),VOL=SER=T85166,
// DISP=(NEW,PASS),DCB=(BLKSIZE=1200,LRECL=80,RECFM=FB)
//GO.FT12F002 DD DSN=CALCULO2,UNIT=TAPE,VOL=REF=*.GO.FT12F001,
```

```
// LABEL=(21,SL)
//GO.FT12F003 DD DSN=CALCULO3,UNIT=TAPE,VOL=REF=*.GO.FT12F001,
// LABEL=(22,SL)
//GO.FT12F004 DD DSN=CALCULO4,UNIT=TAPE,VOL=REF=*.GO.FT12F001,
// LABEL=(23,SL)
//GO.FT12F005 DD DSN=CALCULO5,UNIT=TAPE,VOL=REF=*.GO.FT12F001,
// LABEL=(24,SL),DISP=(NEW,KEEP)
//GO.SYSIN DD *
  Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: Los archivos de salida de datos están definidos en las cinco primeras sentencias GO, en donde se les asigna el mismo número de referencia 12 y los números de secuencia 1, 2, 3, 4 y 5 respectivamente. Por lo tanto en el programa fuente, cuando se graben datos en esos archivos se codificará WRITE 12, además de la sentencia ENDFILE12 cada vez que se concluya la grabación de alguno de los archivos. Se menciona el DCB con el cual se grabará la cinta.

Modelo 434 . Ejecución de un programa residente en la biblioteca con el nombre MITRE121, cuyos resultados son dos archivos de salida distintos, que se graban en sendas cintas magnéticas. El primer archivo se graba, con el nombre RESMIT1, en el rótulo 16 de una cinta que lleva el número de carrete 11118 y el segundo, con el nombre RESMIT2, en el rótulo 5 de otra cinta que lleva el número de carrete 63052. Ambas cintas tienen etiquetas modelo IBM.

```
//..... JOB ....
//PAS01 EXEC PROC=FTGICLG
//FORT.SYSIN DD DSN=SIMBOLIC.CNEA(MITRE121),DISP=SHR
/*
//GO.FT11F001 DD DSN=RESMIT1,UNIT=TAPE,VOL=SER=T11118,LABEL=(16,SL),
// DISP=(NEW,KEEP),DCB=(BLKSIZE=1600,LRECL=80,RECFM=FB)
//GO.FT12F001 DD DSN=RESMIT2,UNIT=TAPE,VOL=SER=T63052,LABEL=(5,SL),
// DISP=(NEW,KEEP),DCB=(BLKSIZE=2400,LRECL=80,RECFM=FB)
//GO.SYSIN DD *
  Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: Los archivos de salida de datos están definidos en las dos primeras sentencias GO, en donde se les asignan los números de referencia 11 y 12; por lo tanto en el programa fuente, cuando se graben resultados en esos archivos, se codifica-

r  WRITE11 y WRITE12 respectivamente. Se mencionan los DCB con los cuales se grabar n las cintas.

PUNTO 4.

Ejecución de programas en FORTRAN, residentes en una biblioteca particionada y catalogada de programas en lenguaje simbólico, de nombre SIMBOLIC.CNEA, cuyos datos de entrada residen en la propia biblioteca y cuyos resultados se graban en cintas magnéticas.

Modelo 435 . Ejecución de un programa grabado en la biblioteca con el nombre INERCIAL, cuyos datos de entrada constituyen un archivo que reside, con el nombre DATOINER, en la propia biblioteca como un miembro independiente y cuyos resultados configuran un archivo de salida que se graba en una cinta magnética. El problema se puede resolver en dos pasos: en el PASO1 el utilitario IEBCGENER transfiere el archivo de entrada de datos de la biblioteca a un disco auxiliar; en el PASO2 el procedimiento FTG1CLG ejecuta el programa obteniendo los datos del disco auxiliar y grabando los resultados, con el nombre SALIDA, en el rótulo 6 de la cinta que lleva el número de carrete 15251 y tiene etiquetas modelo IBM.

```
//..... JOB ....
//PASO1 EXEC PGM=IEBCGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA(DATOINER),DISP=SHR
//SYSUT2 DD DSN=DATOINER,UNIT=WORKDISK,DISP=(NEW,PASS),
//  SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//SYSIN DD DUMMY
/*
//PASO2 EXEC PROC=FTG1CLG
//FORT.SYSIN DD DSN=SIMBOLIC.CNEA(INERCIAL),DISP=SHR
/*
//GO.FT18F001 DD DSN=DATOINER,UNIT=WORKDISK,DISP=(OLD,DELETE)
//GO.FT03F001 DD DSN=SALIDA,UNIT=TAPE,VOL=SER=T15251,DISP=(NEW,KEEP),
//  LABEL=(6,SL),DCB=(BLKSIZE=800,LRECL=80,RECFM=FB)
/*
//
```

NOTA: El archivo de entrada de datos está definido en la primera sentencia GO, en donde se le asigna el número de referencia 18 y el archivo de salida de datos está definido en la segunda sentencia GO, en donde se le asigna el número de referencia 3. Por lo tanto en el programa fuente, cuando se lean datos del primer archivo se codificará READ18 y cuando se graben resultados en el segundo, se codi-

ficará WRITE 3. Se mencionan los DCB con los cuales se grabarán el disco auxiliar y la cinta.

Modelo 436 . Ejecución de un programa grabado en la biblioteca con el nombre BATITU, cuyos datos de entrada son tres archivos distintos que residen en la propia biblioteca, como miembros independientes, con los nombres BATIDATA, BATIDATB y BATIDATC y cuyos resultados son cuatro archivos distintos que se graban en rótulos consecutivos de una cinta magnética. Con los nombres BATIRES1, BATIRES2, BATIRES3 y BATIRES4, los archivos de salida se graban en los rótulos 11, 12, 13 y 14 de la cinta que lleva el número de carrete 17715 y tiene etiquetas modelo IBM. El problema se puede resolver en cuatro pasos: en los tres primeros (PASO1, PASO2 y PASO3) el utilitario IEBGENER transfiere, de a uno, los tres archivos de entrada de datos de la biblioteca a un disco auxiliar; en el PASO4 el procedimiento FTG1CLG ejecuta el programa obteniendo los datos de entrada del disco auxiliar y grabando los resultados en la cinta.

```
//..... JOB ....
//PASO1 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA(BATIDATA),DISP=SHR
//SYSUT2 DD DSN=BATIDATA,UNIT=WORKDISK,DISP=(NEW,PASS),
//  SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//SYSIN DD DUMMY
/*
//PASO2 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA(BATIDATB),DISP=SHR
//SYSUT2 DD DSN=BATIDATB,VOL=REF=*.PASO1.SYSUT2,DISP=(NEW,PASS),
//  SPACE=(CYL,(1,1))
//SYSIN DD DUMMY
/*
//PASO3 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA(BATIDATC),DISP=SHR
//SYSUT2 DD DSN=BATIDATC,VOL=REF=*.PASO1.SYSUT2,DISP=(NEW,PASS),
//  SPACE=(CYL,(1,1))
//SYSIN DD DUMMY
/*
//PASO4 EXEC PROC=FTG1CLG
//FORT.SYSIN DD DSN=SIMBOLIC.CNEA(BATITU),DISP=SHR
/*
//GO.FT13F001 DD UNIT=WORKDISK,DSN=BATIDATA,DISP=(OLD,DELETE)
//GO.FT11F001 DD UNIT=WORKDISK,DSN=BATIDATB,DISP=(OLD,DELETE)
//GO.FT10F001 DD UNIT=WORKDISK,DSN=BATIDATC,DISP=(OLD,DELETE)
```

```
//GO.FT08F001 DD DSN=BATIRES1,UNIT=TAPE,VOL=SER=T17715,LABEL=(11,SL),
// DISP=(NEW,PASS),DCB=(BLKSIZE=2000,LRECL=80,RECFM=FB)
//GO.FT08F002 DD DSN=BATIRES2,UNIT=TAPE,VOL=REF=*.GO.FT08F001,
// LABEL=(12,SL)
//GO.FT08F003 DD DSN=BATIRES3,UNIT=TAPE,VOL=REF=*.GO.FT08F001,
// LABEL=(13,SL)
//GO.FT08F004 DD DSN=BATIRES4,UNIT=TAPE,VOL=REF=*.GO.FT08F001,
// LABEL=(14,SL),DISP=(NEW,KEEP)
/*
//
```

NOTA: Los archivos de entrada de datos están definidos en las tres primeras sentencias GO, en donde se les asignan los números de referencia 13, 11 y 10; por lo tanto en el programa fuente, cuando se lean datos de esos archivos se codificará READ13, READ11 y READ10 respectivamente. A su vez los archivos de salida de datos están definidos en la cuarta sentencia GO y siguientes, en donde se les asigna el mismo número de referencia 8 y los números de secuencia 1, 2, 3 y 4 respectivamente. Por consiguiente en el programa fuente, cuando se graben resultados en esos archivos se codificará WRITE8, además de la sentencia ENDFILE8 para señalar la finalización de cada uno de ellos. Se mencionan los DCB con los cuales se grabarán el disco auxiliar y la cinta.

Modelo 437. Ejecución de un programa grabado en la biblioteca con el nombre NUEVOCAL, cuyos datos de entrada son dos archivos distintos que residen en la propia biblioteca, como miembros independientes, con los nombres NUEVDAT7 y NUEVDAT8 y cuyos resultados son dos archivos de salida que se graban en sendas cintas magnéticas. En el rótulo 12 de la cinta que lleva el número de carrete 99166, se graban los resultados del primer archivo con el nombre NUEVRESA y en el rótulo 3 de la cinta que lleva el número de carrete 00800, se graban los resultados del segundo archivo con el nombre NUEVRESB. Ambas cintas tienen etiquetas modelo IBM. El problema se puede resolver en tres pasos: en los dos primeros (PAS01 y PAS02) el utilitario IEBGENER transfiere, de a uno, los dos archivos de entrada de la biblioteca al mismo disco auxiliar; en el PAS03 el procedimiento FTG1CLG ejecuta el programa obteniendo los datos de entrada del disco auxiliar y grabando los resultados en las cintas.

```

//..... JOB ....
//PASO1 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA(NUEVDAT7),DISP=SHR
//SYSUT2 DD UNIT=WORKDISK,DISP=(NEW,PASS),DSN=NUEVDAT7,
//  SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//SYSIN DD DUMMY
/*
//PASO2 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.CNEA(NUEVDAT8),DISP=SHR
//SYSUT2 DD VOL=REF=*.PASO1.SYSUT2,DISP=(NEW,PASS),DSN=NUEVDAT8,
//  SPACE=(CYL,(1,1))
//SYSIN DD DUMMY
/*
//PASO3 EXEC PROC=FTG1CLG
//FORT.SYSIN DD DSN=SIMBOLIC.CNEA(NUEVOCAL),DISP=SHR
/*
//GO.FT01F001 DD UNIT=WORKDISK,DSN=NUEVDAT7,DISP=(OLD,DELETE)
//GO.FT02F001 DD UNIT=WORKDISK,DSN=NUEVDAT8,DISP=(OLD,DELETE)
//GO.FT03F001 DD UNIT=TAPE,VOL=SER=T99166,DISP=(NEW,KEEP),DSN=NUEVRESA,
//  LABEL=(12,SL),DCB=(BLKSIZE=1200,LRECL=80,RECFM=FB)
//GO.FT08F001 DD UNIT=TAPE,VOL=SER=T00800,DISP=(NEW,KEEP),DSN=NUEVRESB,
//  LABEL=(3,SL),DCB=(BLKSIZE=2400,LRECL=80,RECFM=FB)
/*
//

```

NOTA: Los archivos de entrada de datos están definidos en las dos primeras sentencias GO, en donde se les asignan los números de referencia 1 y 2; por lo tanto en el programa fuente, cuando se lean datos de esos archivos, se codificará READ1 y READ2 respectivamente. A su vez los archivos de salida de datos están definidos en la tercera y cuarta sentencia GO, en donde se les asignan los números de referencia 3 y 8; por consiguiente en el programa fuente, cuando se graben resultados en esos archivos, se codificará WRITE3 y WRITE8 respectivamente. Se mencionan los DCB con los cuales se grabarán el disco auxiliar y las cintas.

PUNTO 5.

Ejecución de módulos procedentes de FORTRAN, grabados en una biblioteca particionada y catalogada de módulos de carga, de nombre REALES.CNEA, que utilizan cintas magnéticas como archivos de salida de datos.

Modelo 438 . Ejecución de un módulo de carga que reside en la biblioteca con el nombre GEMA, cuyos resultados constituyen un archivo de salida que se graba, con el nombre GEMRESUL, en el rótulo 13 de una cinta magnética que lleva el número de carrete 90122 y tiene etiquetas modelo IBM.

```
//..... JOB ....
//PAS01 EXEC PGM=GEMA
//STEPLIB DD DSN=REALES.CNEA,DISP=SHR
/*
//FT06F001 DD SYSOUT=A
//FT02F001 DD DSN=GEMRESUL,UNIT=TAPE,VOL=SER=T90122,LABEL=(13,SL),
//  DISP=(NEW,KEEP),DCB=(BLKSIZE=720,LRECL=80,RECFM=FB)
//FT05F001 DD *
  Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: El archivo de salida de datos está definido en la sentencia FT02F001, en donde se le asigna el número de referencia 2; por lo tanto en el programa fuente que dió origen al módulo, se codificó WRITE2 para grabar resultados en ese archivo. Se menciona el DCB con el cual se grabará la cinta.

Modelo 439 . Ejecución de un módulo de carga, grabado en la biblioteca con el nombre FERRITA, cuyos resultados son cinco archivos de salida distintos que se graban, con los nombres FERRES1, FERRES2, FERRES3, FERRES4 y FERRES5, en los rótulos 18, 19, 20, 21 y 22 de una cinta magnética que lleva el número de carrete 66158 y tiene etiquetas modelo IBM.

```
//..... JOB ....
//PAS01 EXEC PGM=FERRITA
//STEPLIB DD DSN=REALES.CNEA,DISP=SHR
/*
//FT06F001 DD SYSOUT=A
//FT08F001 DD DSN=FERRES1,UNIT=TAPE,LABEL=(18,SL),DISP=(NEW,PASS),
//  VOL=SER=T66158,DCB=(BLKSIZE=1200,LRECL=80,RECFM=FB)
//FT08F002 DD DSN=FERRES2,UNIT=TAPE,VOL=REF=*.FT08F001,LABEL=(19,SL)
```



```
//FT08F003 DD DSN=FERRES3,UNIT=TAPE,VOL=REF=*.FT08F001,LABEL=(20,SL)
//FT08F004 DD DSN=FERRES4,UNIT=TAPE,VOL=REF=*.FT08F001,LABEL=(21,SL)
//FT08F005 DD DSN=FERRES5,UNIT=TAPE,VOL=REF=*.FT08F001,LABEL=(22,SL),
// DISP=(NEW,KEEP)
//FT05F001 DD *
  Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: Los archivos de salida de datos están definidos en las sentencias

FT08F001, FT08F002, FT08F003, FT08F004 y FT08F005, en donde se les asigna el mismo número de referencia 8 y los números de secuencia 1, 2, 3, 4 y 5 respectivamente. Por lo tanto en el programa fuente que dió origen al módulo, se codificó WRITE8 para grabar resultados en esos archivos, además de la sentencia ENDFILE8 para señalar la finalización de cada uno de ellos. Se menciona el DCB con el cual se grabará la cinta.

Modelo 440 . Ejecución de un módulo de carga, grabado en la biblioteca con el nombre TANDEM, cuyos resultados son dos archivos de salida de datos que se graban en sendas cintas magnéticas: el primero, con el nombre TANRES1, en el rótulo 3 de la cinta que lleva el número de carrete 81111 y el segundo, con el nombre TANRES2, en el rótulo 7 de la cinta que lleva el número de carrete 25036. Ambas cintas tienen etiquetas modelo IBM.

```
//..... JOB ....
//PAS01 EXEC PGM=TANDEM
//STEPLIB DD DSN=REALES.CNEA,DISP=SHR
/*
//FT06F001 DD SYSOUT=A
//FT03F001 DD DSN=TANRES1,UNIT=TAPE,VOL=SER=T81111,LABEL=(3,SL),
// DISP=(NEW,KEEP),DCB=(BLKSIZE=1600,LRECL=80,RECFM=FB)
//FT02F001 DD DSN=TANRES2,UNIT=TAPE,VOL=SER=T25036,LABEL=(7,SL),
// DISP=(NEW,KEEP),DCB=(BLKSIZE=2400,LRECL=80,RECFM=FB)
//FT05F001 DD *
  Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: Los archivos de salida de datos están definidos en las sentencias FT03F001 y FT02F001, en donde se les asignan los números de referencia 3 y 2; por lo tanto en el programa fuente que dió origen al módulo, se codificó WRITE3 y WRITE2 res-

pectivamente, para grabar los resultados en esos archivos. Se mencionan los DCB con los cuales se grabarán las cintas.

PUNTO 6.

Ejecución de programas en FORTRAN grabados en cintas magnéticas, que utilizan cintas como archivos de salida de datos.

Modelo 441 . Ejecución de un programa en lenguaje simbólico FORTRAN que reside, con el nombre FUSTE, en el rótulo 5 de una cinta magnética y cuyos resultados constituyen un archivo de salida que se graba; con el nombre FUSTESAL, en el rótulo 28 de esa misma cinta. Esta lleva el número de carrete 02847 y tiene etiquetas modelo IBM.

```
//..... JOB ....
//PAS01 EXEC PROC=FTG1CLG
//FORT.SYSIN DD DSN=FUSTE,UNIT=TAPE,VOL=SER=T02847,LABEL=(5,SL),
// DISP=(OLD,PASS)
/*
//GO.FT13F001 DD DSN=FUSTESAL,UNIT=TAPE,VOL=SER=T02847,LABEL=(28,SL),
// DISP=(NEW,KEEP),DCB=(BLKSIZE=2000,LRECL=80,RECFM=FB)
//GO.SYSIN DD *
  Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: El archivo de salida de datos está definido en la primera sentencia GO, en donde se le asigna el número de referencia 13; por consiguiente en el programa fuente, cuando se graben resultados en ese archivo, se codificará WRITE13. Se menciona el DCB con el cual se grabará la cinta.

Modelo 442 . Ejecución de un programa en lenguaje absoluto de FORTRAN que reside, con el nombre RAYOSX, en el rótulo 30 de una cinta magnética y cuyos resultados constituyen un archivo de salida que se graba, con el nombre RESULX, en el rótulo 42 de esa misma cinta. Esta lleva el número de carrete 10576 y tiene etiquetas modelo IBM.

```
//..... JOB ....
//PAS01 EXEC PROC=FTG1LG
//LKED.SYSIN DD DSN=RAYOSX,UNIT=TAPE,VOL=SER=T10576,LABEL=(30,SL),
// DISP=(OLD,PASS)
/*
//GO.FT04F001 DD DSN=RESULX,UNIT=TAPE,VOL=SER=T10576,LABEL=(42,SL),
// DISP=(NEW,KEEP),DCB=(BLKSIZE=1600,LRECL=80,RECFM=FB)
//GO.SYSIN DD *
```

```

Aquí tarjetas perforadas de DATOS
/*
//

```

NOTA: El archivo de salida de datos está definido en la primera sentencia GO, en donde se le asigna el número de referencia 4; por lo tanto en el programa fuente que dió origen al módulo grabado en la cinta, para grabar resultados en ese archivo se codificó WRITE4. Se menciona el DCB con el cual se grabará la cinta.

Modelo 443 . Ejecución de un programa en lenguaje simbólico FORTRAN que reside, con el nombre BETA, en el rótulo 15 de una cinta magnética que lleva el número de carrete 11025 y cuyos resultados constituyen un archivo de salida que se graba, con el nombre OUTPUT21, en el rótulo 23 de otra cinta que lleva el número de carrete 26268. Ambas cintas tienen etiquetas modelo IBM.

```

//..... JOB ....
//PASO1 EXEC PROC=FTG1CLG
//FORT.SYSIN DD DSN=BETA,UNIT=TAPE,VOL=SER=T11025,LABEL=(15,SL),
// DISP=(OLD,KEEP)
/*
//GO.FT09F001 DD DSN=OUTPUT21,UNIT=TAPE,VOL=SER=T26268,LABEL=(23,SL),
// DISP=(NEW,KEEP),DCB=(BLKSIZE=1600,LRECL=80,RECFM=FB)
//GO.SYSIN DD *
Aquí tarjetas perforadas de DATOS
/*
//

```

NOTA: El archivo de salida de datos está definido en la primera sentencia GO, en donde se le asigna el número de referencia 9; por lo tanto en el programa fuente, cuando se graben resultados en ese archivo se codificará WRITE9. Se menciona el DCB con el cual se grabará la cinta.

Modelo 444 . Ejecución de un programa en lenguaje absoluto de FORTRAN que reside, con el nombre ROTOR, en el rótulo 17 de una cinta magnética que lleva el número de carrete 13303 y cuyos resultados constituyen un archivo de salida que se graba, con el nombre ROTRESUL, en el rótulo 11 de otra cinta que lleva el número de carrete 10008. Ambas cintas tienen etiquetas modelo IBM.

```

//..... JOB ....
//PASO1 EXEC PROC=FTG1LG

```

```
//LKED.SYSIN DD DSN=ROTOR,UNIT=TAPE,VOL=SER=T13303,LABEL=(17,SL),
// DISP=(OLD,KEEP)
/*
//GO.FT14F001 DD DSN=ROTRESUL,UNIT=TAPE,VOL=SER=T10008,LABEL=(11,SL),
// DISP=(NEW,KEEP),DCB=(BLKSIZE=3200,LRECL=80,RECFM=FB)
//GO.SYSIN DD *
Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: El archivo de salida de datos está definido en la primera sentencia GO, en donde se le asigna el número de referencia 14; por lo tanto en el programa fuente que dió origen al módulo grabado en la cinta, para grabar resultados en ese archivo se codificó WRITE14. Se menciona el DCB con el cual se grabará la cinta.

Modelo 445. Ejecución de un programa en lenguaje simbólico FORTRAN que reside, con el nombre PULSO, en el rótulo 9 de una cinta magnética y cuyos resultados son tres archivos de salida distintos que se graban en esa misma cinta. Los archivos de nombres RESPULS1, RESPULS2 y RESPULS3, se graban en los rótulos 32, 33 y 34. La cinta lleva el número de carrete 12828 y tiene etiquetas modelo IBM.

```
//..... JOB ....
//PAS01 EXEC PROC=FTG1CLG
//FORT.SYSIN DD DSN=PULSO,UNIT=TAPE,VOL=SER=T12828,LABEL=(9,SL),
// DISP=(OLD,PASS)
/*
//GO.FT04F001 DD DSN=RESPULS1,UNIT=TAPE,VOL=SER=T12828,LABEL=(32,SL),
// DISP=(NEW,PASS),DCB=(BLKSIZE=2400,LRECL=80,RECFM=FB)
//GO.FT04F002 DD DSN=RESPULS2,UNIT=TAPE,VOL=REF=*.GO.FT04F001,
// LABEL=(33,SL)
//GO.FT04F003 DD DSN=RESPULS3,UNIT=TAPE,VOL=REF=*.GO.FT04F001,
// LABEL=(34,SL),DISP=(NEW,KEEP)
//GO.SYSIN DD *
Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: Los archivos de salida de datos están definidos en las tres primeras sentencias GO, en donde se les asigna el mismo número de referencia 4 y los números de secuencia 1, 2 y 3 respectivamente. Por lo tanto en el programa fuente, cuando se graben resultados en esos archivos se codificará WRITE4, además de la sentencia ENDFILE4 para señalar la finalización de cada uno de ellos. Se menciona el DCB con el cual se grabará la cinta.

Modelo 446 . Ejecución de un programa en lenguaje absoluto de FORTRAN que reside, con el nombre ORBITAL, en el rótulo 31 de una cinta magnética y cuyos resultados son cuatro archivos de salida que se graban en esa misma cinta. Los archivos de nombres SALIDA1, SALIDA2, SALIDA3 y SALIDA4 se graban en los r tulos 45, 46, 47 y 48. La cinta lleva el n mero de carrete 12321 y tiene etiquetas modelo IBM.

```
//..... JOB ....
//PASO1 EXEC PROC=FTG1LG
//LKED.SYSIN DD DSN=ORBITAL,UNIT=TAPE,VOL=SER=T12321,LABEL=(31,SL),
// DISP=(OLD,PASS)
/*
//GO.FT20F001 DD DSN=SALIDA1,UNIT=TAPE,VOL=SER=T12321,LABEL=(45,SL),
// DISP=(NEW,PASS),DCB=(BLKSIZE=1600,LRECL=80,RECFM=FB)
//GO.FT20F002 DD DSN=SALIDA2,UNIT=TAPE,VOL=REF=*.GO.FT20F001,
// LABEL=(46,SL)
//GO.FT20F003 DD DSN=SALIDA3,UNIT=TAPE,VOL=REF=*.GO.FT20F001,
// LABEL=(47,SL)
//GO.FT20F004 DD DSN=SALIDA4,UNIT=TAPE,VOL=REF=*.GO.FT20F001,
// LABEL=(48,SL),DISP=(NEW,KEEP)
//GO.SYSIN DD *
Aqu  tarjetas perforadas de DATOS
/*
//
```

NOTA: Los archivos de salida de datos est n definidos en las cuatro primeras sentencias GO, en donde se les asigna el mismo n mero de referencia 20 y los n meros de secuencia 1, 2, 3 y 4 respectivamente. Por lo tanto en el programa fuente que di  origen al m dulo grabado en la cinta, para grabar resultados en esos archivos se codific  WRITE20, adem s de la sentencia ENDFILE20, para se alar la finalizaci n de cada uno de ellos. Se menciona el DCB con el cual se grabar  la cinta.

Modelo 447 . Ejecuci n de un programa en lenguaje simb lico FORTRAN que reside, con el nombre HAZ, en el r tulo 2 de una cinta magn tica que lleva el n mero de carrete 61020 y cuyos resultados son cuatro archivos de salida que se graban en otra cinta. Los archivos de nombres SALIDA11, SALIDA12, SALIDA13 y SALIDA14 se graban en los r tulos 15, 16, 17 y 18 de la cinta que lleva el n mero de carrete 61030. Ambas cintas tienen etiquetas modelo IBM.

```
//..... JOB ....
```

```
//PAS01 EXEC PROC=FTG1CLG
//FORT.SYSIN DD DSN=HAZ,UNIT=TAPE,VOL=SER=T61020,LABEL=(2,SL),
// DISP=(OLD,KEEP)
/*
//GO.FT19F001 DD DSN=SALIDA11,UNIT=TAPE,VOL=SER=T61030,LABEL=(15,SL),
// DISP=(NEW,PASS),DCB=(BLKSIZE=3200,LRECL=80,RECFM=FB)
//GO.FT19F002 DD DSN=SALIDA12,UNIT=TAPE,VOL=REF=*.GO.FT19F001,
// LABEL=(16,SL)
//GO.FT19F003 DD DSN=SALIDA13,UNIT=TAPE,VOL=REF=*.GO.FT19F001,
// LABEL=(17,SL)
//GO.FT19F004 DD DSN=SALIDA14,UNIT=TAPE,VOL=REF=*.GO.FT19F001,
// LABEL=(18,SL),DISP=(NEW,KEEP)
//GO.SYSIN DD *
Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: Los archivos de salida de datos están definidos en las cuatro primeras sentencias GO, en donde se les asigna el mismo número de referencia 19 y los números de secuencia 1, 2, 3 y 4 respectivamente. Por lo tanto en el programa fuente, cuando se graben resultados en esos archivos se codificará WRITE19, además de la sentencia ENDFILE19 para señalar la finalización de cada uno de ellos. Se menciona el DCB con el cual se grabará la cinta.

Modelo 448 . Ejecución de un programa en lenguaje absoluto de FORTRAN que reside, con el nombre MESON, en el rótulo 25 de una cinta magnética que lleva el número de carrete 54321 y cuyos resultados son 4 archivos de salida que se graban en otra cinta. Los archivos, de nombres MESORES1, MESORES2, MESORES3 y MESORES4, se graban en los rótulos 5, 6, 7 y 8 de la cinta que lleva el número de carrete 00100. Ambas cintas tienen etiquetas modelo IBM.

```
//..... JOB ....
//PAS01 EXEC PROC=FTG1LG
//LKED.SYSIN DD DSN=MESON,UNIT=TAPE,VOL=SER=T54321,LABEL=(25,SL),
// DISP=(OLD,KEEP)
/*
//GO.FT18F001 DD DSN=MESORES1,UNIT=TAPE,VOL=SER=T00100,LABEL=(5,SL),
// DISP=(NEW,PASS),DCB=(BLKSIZE=800,LRECL=80,RECFM=FB)
//GO.FT18F002 DD DSN=MESORES2,UNIT=TAPE,VOL=REF=*.GO.FT18F001,
// LABEL=(6,SL)
//GO.FT18F003 DD DSN=MESORES3,UNIT=TAPE,VOL=REF=*.GO.FT18F001,
// LABEL=(7,SL)
//GO.FT18F004 DD DSN=MESORES4,UNIT=TAPE,VOL=REF=*.GO.FT18F001,
// LABEL=(8,SL),DISP=(NEW,KEEP)
//GO.SYSIN DD *
```

```
Aquí tarjetas perforadas de DATOS  
/*  
//
```

NOTA: Los archivos de salida de datos están definidos en las tres primeras sentencias GO, en donde se les asigna el mismo número de referencia 18 y los números de secuencia 1, 2, 3 y 4 respectivamente. Por lo tanto en el programa fuente que dió origen al módulo grabado en la cinta, para grabar resultados en esos archivos se codificó WRITE18, además de la sentencia ENDFILE18, para señalar la finalización de cada uno de ellos. Se menciona el DCB con el cual se grabará la cinta.

CAPITULO XI

MODELOS PARA LA EJECUCION DE PROGRAMAS QUE
UTILIZAN CINTAS MAGNETICAS COMO ARCHIVOS DE
ENTRADA Y SALIDA DE DATOS SIMULTANEAMENTE

PUNTO 1.

Ejecución de programas en FORTRAN que se presentan en tarjetas perforadas y utilizan cintas magnéticas como archivos de entrada y salida de datos, simultáneamente.

Modelo 449 . Ejecución de un programa en tarjetas perforadas en lenguaje simbólico FORTRAN, cuyos datos de entrada constituyen un archivo que reside en una cinta magnética y cuyos resultados configuran un archivo de salida que se graba en esa misma cinta. El archivo de entrada reside, con el nombre DATFENOM, en el rótulo 2 de la cinta que lleva el número de carrete 06517 y tiene etiquetas modelo IBM; el archivo de salida se graba, con el nombre RESFENOM, en el rótulo 6 de esa misma cinta. El problema se puede resolver en dos pasos: en el PASO1 el utilitario IEBGENER transfiere el juego de datos DATFENOM de la cinta a un disco auxiliar; en el PASO2 el procedimiento FTG1CLG ejecuta el programa obteniendo los datos de entrada del disco auxiliar y grabando los resultados en la cinta.

```
//..... JOB ....
//PASO1 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSIN DD DUMMY
//SYSUT1 DD DSN=DATFENOM,UNIT=TAPE,VOL=SER=TO6517,LABEL=(2,SL),
// DISP=(OLD,PASS)
//SYSUT2 DD DSN=DATFENOM,UNIT=WORKDISK,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
/*
//PASO2 EXEC PROC=FTG1CLG
//FORT.SYSIN DD *
Aquí tarjetas perforadas en FORTRAN
/*
//GO.FT11F001 DD DSN=DATFENOM,UNIT=WORKDISK,DISP=(OLD,DELETE)
//GO.FT15F001 DD DSN=RESFENOM,UNIT=TAPE,VOL=SER=TO6517,LABEL=(6,SL),
// DISP=(NEW,KEEP),DCB=(BLKSIZE=800,LRECL=80,RECFM=FB)
/*
//
```

NOTA: El archivo de entrada de datos está definido en la primera sentencia GO, en donde se le asigna el número de referencia 11 y el archivo de salida está definido en la segunda sentencia GO, en donde se le asigna el número de referencia 15; por lo tanto en el programa fuente, se codificará READ11 y WRITE15 respectivamente.

te, para leer y grabar datos en esos archivos. Se mencionan los DCB con los cuales se grabarán el disco auxiliar y la cinta.

Modelo 450 . Ejecución de un programa en tarjetas perforadas en lenguaje absoluto de FORTRAN, cuyos datos de entrada constituyen un archivo que reside en una cinta magnética y cuyos resultados configuran un archivo de salida que se graba en esa misma cinta. El archivo de entrada reside, con el nombre DATOIN, en el rótulo 7 de la cinta que lleva el número de carrete 55177 y tiene etiquetas modelo IBM; el archivo de salida se graba, con el nombre DATOUT, en el rótulo 21 de la misma cinta. El problema se puede resolver en dos pasos: en el PASO1 el utilitario IEBGENER transfiere el juego de datos DATOIN de la cinta a un disco auxiliar; en el PASO2 el procedimiento FTG1LG ejecuta el programa obteniendo los datos de entrada del disco auxiliar y grabando los resultados en la cinta.

```
//..... JOB ....
//PASO1 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSIN DD DUMMY
//SYSUT1 DD DSN=DATOIN,UNIT=TAPE,VOL=SER=T55177,LABEL=(7,SL),
// DISP=(OLD,PASS)
//SYSUT2 DD DSN=DATOIN,UNIT=WORKDISK,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1)),DCB=(BLKSIZE=1200,LRECL=80,RECFM=FB)
/*
//PASO2 EXEC PROC=FTG1LG
//LKED.SYSIN DD *,DCB=BLKSIZE=80
Aquí tarjetas perforadas en absoluto de FORTRAN
/*
//GO.FT17F001 DD DSN=DATOIN,UNIT=WORKDISK,DISP=(OLD,DELETE)
//GO.FT15F001 DD DSN=DATOUT,UNIT=TAPE,VOL=SER=T55177,LABEL=(21,SL),
// DISP=(NEW,KEEP),DCB=(BLKSIZE=800,LRECL=80,RECFM=FB)
/*
//
```

NOTA: El archivo de entrada de datos está definido en la primera sentencia GO, en donde se le asigna el número de referencia 17 y el archivo de salida está definido en la segunda sentencia GO, en donde se le asigna el número de referencia 15; por lo tanto en el programa fuente que dió origen al lote de tarjetas perforadas en absoluto, se codificó READ17 y WRITE15 respectivamente, para leer y grabar datos en esos archivos. Se mencionan los DCB con los cuales se grabarán el disco

auxiliar y la cinta.

Modelo 451 . Ejecución de un programa en tarjetas perforadas en lenguaje simbólico FORTRAN, cuyos datos de entrada constituyen un archivo que reside en una cinta magnética y cuyos resultados configuran un archivo de salida que se graba en otra cinta. El archivo de entrada reside, con el nombre DATOS133, en el rótulo 7 de una cinta que lleva el número de carrete 15500; el archivo de salida se graba, con el nombre RESUL133 en el rótulo 5 de otra cinta que lleva el número de carrete 16600. Ambas cintas tienen etiquetas modelo IBM.

```
//..... JOB ....
//PAS01 EXEC PROC=FTG1CLG
//FORT.SYSIN DD *
  Aquí tarjetas perforadas en FORTRAN
/*
//GO.FT16F001 DD DSN=DATOS133,UNIT=TAPE,VOL=SER=T15500,LABEL=(7,SL),
//  DISP=(OLD,KEEP)
//GO.FT12F001 DD DSN=RESUL133,UNIT=TAPE,VOL=SER=T16600,LABEL=(5,SL),
//  DISP=(NEW,KEEP),DCB=(BLKSIZE=2400,LRECL=80,RECFM=FB)
/*
//
```

NOTA: El archivo de entrada de datos está definido en la primera sentencia GO, en donde se le asigna el número de referencia 16 y el archivo de salida está definido en la segunda sentencia GO, en donde se le asigna el número de referencia 12; por lo tanto en el programa fuente, se codificará READ16 y WRITE12 respectivamente, para leer y grabar datos en esos archivos. Se menciona el DCB con el cual se grabará la cinta.

Modelo 452 . Ejecución de un programa en tarjetas perforadas en lenguaje absoluto de FORTRAN, cuyos datos de entrada constituyen un archivo que reside en una cinta magnética y cuyos resultados configuran un archivo de salida que se graba en otra cinta. El archivo de entrada reside, con el nombre DATOPICK, en el rótulo 11 de una cinta de carrete número 10606; el archivo de salida se graba, con el nombre RESULPIC, en el rótulo 3 de otra cinta de carrete número 77575. Ambas cintas tienen etiquetas modelo IBM.

```
//..... JOB ....
//PAS01 EXEC PROC=FTG1LG
```

```
//LKED.SYSIN DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas en absoluto de FORTRAN
/*
//GO.FT14F001 DD DSN=DATOPICK,UNIT=TAPE,VOL=SER=T10606,LABEL=(11,SL),
//  DISP=(OLD,KEEP)
//GO.FT18F001 DD DSN=RESULPIC,UNIT=TAPE,VOL=SER=T77575,LABEL=(3,SL),
//  DISP=(NEW,KEEP),DCB=(BLKSIZE=800,LRECL=80,RECFM=FB)
/*
//
```

NOTA: El archivo de entrada de datos está definido en la primera sentencia GO, en donde se le asigna el número de referencia 14 y el archivo de salida está definido en la segunda sentencia GO, en donde se le asigna el número de referencia 18; por lo tanto en el programa fuente que dió origen al lote de tarjetas perforadas en absoluto, se codificó READ14 y WRITE18 respectivamente, para leer y grabar datos en esos archivos. Se menciona el DCB con el cual se grabará la cinta.

Modelo 453. Ejecución de un programa en tarjetas perforadas en lenguaje simbólico FORTRAN, cuyos datos de entrada son cuatro archivos distintos que residen en rótulos consecutivos o no consecutivos de una cinta magnética y cuyos resultados son tres archivos de salida que se graban en esa misma cinta. Los archivos de entrada de nombres DATALFA, DATBETA, DATGAMMA y DATDELTA, están grabados en los rótulos 18, 25, 26 y 35 de la cinta que lleva el número de carrete 35533 y tiene etiquetas modelo IBM. Con los nombres RESUL1, RESUL2 y RESUL3, los resultados se graban en los rótulos 40, 41 y 42 de esa misma cinta. El problema se puede resolver en cinco pasos: en los cuatro primeros (PASO1, PASO2, PASO3 y PASO4) el utilitario IEBGENER transfiere, de a uno, los cuatro archivos de entrada de datos de la cinta al mismo disco auxiliar; en el PASO5 el procedimiento FTG1CLG ejecuta el programa obteniendo los datos de entrada del disco auxiliar y grabando los resultados en la cinta.

```
//..... JOB ....
//PASO1 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=DATALFA,UNIT=TAPE,VOL=SER=T35533,LABEL=(18,SL),
//  DISP=(OLD,PASS)
//SYSUT2 DD DSN=DATALFA,UNIT=WORKDISK,DISP=(NEW,PASS),
//  SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
```

```

//SYSIN DD DUMMY
/*
//PASO2 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=DATBETA,UNIT=TAPE,VOL=REF=*.PASO1.SYSUT1,
// LABEL=(25,SL)
//SYSUT2 DD DSN=DATBETA,VOL=REF=*.PASO1.SYSUT2,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1))
//SYSIN DD DUMMY
/*
//PASO3 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=DATGAMMA,UNIT=TAPE,VOL=REF=*.PASO1.SYSUT1,
// LABEL=(26,SL)
//SYSUT2 DD DSN=DATGAMMA,VOL=REF=*.PASO1.SYSUT2,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1))
//SYSIN DD DUMMY
/*
//PASO4 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=DATDELTA,UNIT=TAPE,VOL=REF=*.PASO1.SYSUT1,
// LABEL=(35,SL)
//SYSUT2 DD DSN=DATDELTA,VOL=REF=*.PASO1.SYSUT2,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1))
//SYSIN DD DUMMY
/*
//PASO5 EXEC PROC=FTG1CLG
//FORT.SYSIN DD *
  Aquí tarjetas perforadas en FORTRAN
/*
//GO.FT17F001 DD UNIT=WORKDISK,DSN=DATA1FA,DISP=(OLD,DELETE)
//GO.FT20F001 DD UNIT=WORKDISK,DSN=DATBETA,DISP=(OLD,DELETE)
//GO.FT19F001 DD UNIT=WORKDISK,DSN=DATGAMMA,DISP=(OLD,DELETE)
//GO.FT18F001 DD UNIT=WORKDISK,DSN=DATDELTA,DISP=(OLD,DELETE)
//GO.FT09F001 DD DSN=RESUL1,UNIT=TAPE,VOL=SER=T35533,LABEL=(40,SL),
// DISP=(NEW,PASS),DCB=(BLKSIZE=1600,LRECL=80,RECFM=FB)
//GO.FT09F002 DD DSN=RESUL2,UNIT=TAPE,VOL=REF=*.GO.FT09F001,
// LABEL=(41,SL)
//GO.FT09F003 DD DSN=RESUL3,UNIT=TAPE,VOL=REF=*.GO.FT09F001,
// LABEL=(42,SL),DISP=(NEW,KEEP)
/*
//

```

NOTA: Los archivos de entrada de datos están definidos en las cuatro primeras sentencias GO, en donde se les asignan los números de referencia 17, 20, 19 y 18. Por lo tanto en el programa fuente, cuando se lean datos de esos archivos se codificará READ17, READ20, READ19 y READ18 respectivamente. A su vez los archivos de salida de datos están definidos en la quinta sentencia GO y siguientes, en donde se les asigna el mismo número de referencia 9 y los números de secuencia 1, 2 y 3 respectivamente. Por consiguiente en el programa fuente, cuando se graben

resultados en esos archivos se codificará una sentencia WRITE9, además de la sentencia ENDFILE9, para señalar la finalización de cada uno de ellos. Se mencionan los DCB con los cuales se grabarán el disco auxiliar y la cinta.

Modelo 454 . Ejecución de un programa en tarjetas perforadas en lenguaje absoluto de FORTRAN, cuyos datos de entrada son tres archivos distintos que residen en rótulos consecutivos o no consecutivos de una cinta magnética y cuyos resultados son dos archivos de salida que se graban en esa misma cinta. Los archivos de entrada, de nombres IKADAT1, IKADAT2 e IKADAT3, están grabados en los rótulos 4, 5 y 7 de la cinta que lleva el número de carrete 86103 y tiene etiquetas modelo IBM. Con los nombres RESIKA1 y RESIKA2, los resultados se graban en los rótulos 38 y 39 de esa misma cinta. El problema se puede resolver en cuatro pasos: en los tres primeros (PASO1, PASO2 y PASO3) el utilitario IEBGENER transfiere, de a uno, los tres archivos de entrada de datos de la cinta al mismo disco auxiliar; en el PASO4 el procedimiento FTGILG ejecuta el programa obteniendo los datos de entrada del disco auxiliar y grabando los resultados en la cinta.

```
//.....JOB ....
//PASO1 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=IKADAT1,UNIT=TAPE,VOL=SER=T86103,LABEL=(4,SL),
//  DISP=(OLD,PASS)
//SYSUT2 DD DSN=IKADAT1,UNIT=WORKDISK,DISP=(NEW,PASS),
//  SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//SYSIN DD DUMMY
/*
//PASO2 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=IKADAT2,UNIT=TAPE,VOL=REF=*.PASO1.SYSUT1,
//  LABEL=(5,SL)
//SYSUT2 DD DSN=IKADAT2,VOL=REF=*.PASO1.SYSUT2,DISP=(NEW,PASS),
//  SPACE=(CYL,(1,1))
//SYSIN DD DUMMY
/*
//PASO3 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=IKADAT3,UNIT=TAPE,VOL=REF=*.PASO1.SYSUT1,
//  LABEL=(7,SL)
//SYSUT2 DD DSN=IKADAT3,VOL=REF=*.PASO1.SYSUT2,DISP=(NEW,PASS),
//  SPACE=(CYL,(1,1))
//SYSIN DD DUMMY
/*
```

```
//PASO4 EXEC PROC=FTG1LG
//LKED.SYSIN DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas en absoluto de FORTRAN
/*
//GO.FT02F001 DD UNIT=WORKDISK,DSN=IKADAT1,DISP=(OLD,DELETE)
//GO.FT04F001 DD UNIT=WORKDISK,DSN=IKADAT2,DISP=(OLD,DELETE)
//GO.FT08F001 DD UNIT=WORKDISK,DSN=IKADAT3,DISP=(OLD,DELETE)
//GO.FT10F001 DD DSN=RESIKA1,UNIT=TAPE,VOL=SER=T86103,LABEL=(38,SL),
//  DISP=(NEW,PASS),DCB=(BLKSIZE=800,LRECL=80,RECFM=FB)
//GO.FT10F002 DD DSN=RESIKA2,UNIT=TAPE,VOL=REF=*.GO.FT10F001,
//  LABEL=(39,SL),DISP=(NEW,KEEP)
/*
//
```

NOTA: Los archivos de entrada de datos están definidos en las tres primeras sentencias GO, en donde se les asignan los números de referencia 2, 4 y 8. Por lo tanto en el programa fuente que dió origen al lote de tarjetas en absoluto, para leer datos de esos archivos se codificó READ2, READ4 y READ8 respectivamente. A su vez los archivos de salida de datos están definidos en la cuarta y quinta sentencia GO, en donde se les asigna el mismo número de referencia 10 y los números de secuencia 1 y 2 respectivamente. Por consiguiente en el programa fuente, para grabar resultados en esos archivos se codificó una sentencia WRITE10, además de la sentencia ENDFILE10, para señalar la finalización de cada uno de ellos. Se mencionan los DCB con los cuales se grabarán el disco auxiliar y la cinta.

Modelo 455 . Ejecución de un programa en tarjetas perforadas en lenguaje simbólico FORTRAN, cuyos datos de entrada son tres archivos distintos que residen en rótulos consecutivos o no consecutivos de una cinta magnética y cuyos resultados son tres archivos de salida que se graban en otra cinta. Los archivos de entrada, de nombres POLIDAT1, POLIDAT2 y POLIDAT3, están grabados en los rótulos 1, 10 y 11 de la cinta que lleva el número de carrete 10021 y con los nombres POLIRES1, POLIRES2 y POLIRES3, los resultados se graban en los rótulos 2, 3 y 4 de otra cinta que lleva el número de carrete 10033. Ambas cintas tienen etiquetas modelo IBM.

```
//..... JOB ....
//PASO1 EXEC PROC=FTG1CLG
//FORT.SYSIN DD *
```


Aquí tarjetas perforadas en FORTRAN

```

/*
//GO.FT04F001 DD DSN=POLIDAT1,UNIT=TAPE,VOL=SER=T10021,LABEL=(1,SL),
// DISP=(OLD,PASS)
//GO.FT04F002 DD DSN=POLIDAT2,UNIT=TAPE,VOL=REF=*.GO.FT04F001,
// LABEL=(10,SL)
//GO.FT04F003 DD DSN=POLIDAT3,UNIT=TAPE,VOL=REF=*.GO.FT04F001,
// LABEL=(11,SL),DISP=(OLD,KEEP)
//GO.FT12F001 DD DSN=POLIRES1,UNIT=TAPE,VOL=SER=T10033,LABEL=(2,SL),
// DISP=(NEW,PASS),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//GO.FT12F002 DD DSN=POLIRES2,UNIT=TAPE,VOL=REF=*.GO.FT12F001,
// LABEL=(3,SL)
//GO.FT12F003 DD DSN=POLIRES3,UNIT=TAPE,VOL=REF=*.GO.FT12F001,
// LABEL=(4,SL),DISP=(NEW,KEEP)
/*
//

```

NOTA: Los archivos de entrada de datos están definidos en las tres primeras sentencias GO, en donde se les asigna el mismo número de referencia 4 y los números de secuencia 1, 2 y 3 respectivamente. Por lo tanto en el programa fuente, cuando se lean datos de esos archivos se codificará una sentencia READ de este tipo:

READ(4,nº de formato,END=dirección). La dirección es la marca de la sentencia a la cual deberá saltar el programa para continuar la ejecución, cada vez que llega al final de uno de los archivos. A su vez los archivos de salida de datos están definidos en la cuarta sentencia GO y siguientes, en donde se les asigna el mismo número de referencia 12 y los números de secuencia 1, 2 y 3 respectivamente. Por consiguiente en el programa fuente, cuando se graben resultados en esos archivos se codificará WRITE12, además de la sentencia ENDFILE12, para señalar la finalización de cada uno de ellos. Se menciona el DCB con el cual se grabará la cinta.

Modelo 456 . Ejecución de un programa en tarjetas perforadas en lenguaje absoluto de FORTRAN, cuyos datos de entrada son tres archivos distintos que residen en rótulos consecutivos o no consecutivos de una cinta magnética y cuyos resultados son cuatro archivos de salida que se graban en otra cinta. Los archivos de entrada, de nombres IN25, IN26 e IN27, están grabados en los rótulos 17, 18 y 28 de la cinta que lleva el número de carrete 00222 y con los nombres OUT1A, OUT2A, OUT3A y OUT4A, los resultados se graban en los rótulos 9, 10, 11 y 12 de otra cinta que lleva el número de carrete 01010. Ambas cintas tienen etiquetas modelo IBM.

```
//..... JOB ....
//PAS01 EXEC PROC=FTG1LG
//LKED.SYSIN DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas en absoluto de FORTRAN
/*
//GO.FT20F001 DD DSN=IN25,UNIT=TAPE,VOL=SER=T00222,LABEL=(17,SL),
//  DISP=(OLD,PASS)
//GO.FT20F002 DD DSN=IN26,UNIT=TAPE,VOL=REF=*.GO.FT20F001,
//  LABEL=(18,SL)
//GO.FT20F003 DD DSN=IN27,UNIT=TAPE,VOL=REF=*.GO.FT20F001,
//  LABEL=(28,SL),DISP=(OLD,KEEP)
//GO.FT11F001 DD DSN=OUT1A,UNIT=TAPE,VOL=SER=T01010,LABEL=(9,SL),
//  DISP=(NEW,PASS),DCB=(BLKSIZE=2000,LRECL=80,RECFM=FB)
//GO.FT11F002 DD DSN=OUT2A,UNIT=TAPE,VOL=REF=*.GO.FT11F001,
//  LABEL=(10,SL)
//GO.FT11F003 DD DSN=OUT3A,UNIT=TAPE,VOL=REF=*.GO.FT11F001,
//  LABEL=(11,SL)
//GO.FT11F004 DD DSN=OUT4A,UNIT=TAPE,VOL=REF=*.GO.FT11F001,
//  LABEL=(12,SL),DISP=(NEW,KEEP)
/*
//
```

NOTA: Los archivos de entrada de datos están definidos en las tres primeras sentencias GO, en donde se les asigna el mismo número de referencia 20 y los números de secuencia 1, 2 y 3. Por lo tanto en el programa fuente que dió origen al lote de tarjetas en absoluto, para leer datos de esos archivos se codificó una sentencia READ de este tipo: READ(20,nº de formato,END=dirección). La "dirección" es la marca de la sentencia a la cual deberá saltar el programa para continuar la ejecución, cada vez que llega al final de uno de los archivos. A su vez los archivos de salida de datos están definidos en la cuarta sentencia GO y siguientes, en donde se les asigna el mismo número de referencia 11 y los números de secuencia 1, 2, 3 y 4 respectivamente. Por consiguiente en el programa fuente, para grabar resultados en esos archivos se codificó una sentencia WRITE11, además de la sentencia ENDFILE11 para señalar la finalización de cada uno de ellos. Se menciona el DCB con el cual se grabará la cinta.

PUNTO 2.

Ejecución de programas en FORTRAN, residentes en una biblioteca particionada y catalogada de programas en lenguaje simbólico, de nombre SIMBOLIC.CNEA, que utilizan cintas magnéticas como archivos de entrada y salida de datos, simultáneamente.

Modelo 457 . Ejecución de un programa grabado en la biblioteca con el nombre QUANTO, cuyos datos de entrada constituyen un archivo que reside en una cinta magnética y cuyos resultados configuran un archivo de salida que se graba en esa misma cinta. El archivo de entrada reside, con el nombre QUANTDAT, en el rótulo 2 de la cinta que lleva el número de carrete 16288 y tiene etiquetas modelo IBM: el archivo de salida se graba, con el nombre QUANTRES, en el rótulo 11 de esa misma cinta. El problema se puede resolver en dos pasos: En el PASO1 el utilitario IEBGENER transfiere el juego de datos QUANTDAT de la cinta a un disco auxiliar; en el PASO2 el procedimiento FTG1CLG ejecuta el programa obteniendo los datos de entrada del disco auxiliar y grabando los resultados en la cinta.

```
//..... JOB ....
//PASO1 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSIN DD DUMMY
//SYSUT1 DD DSN=QUANTDAT,UNIT=TAPE,VOL=SER=T16288,LABEL=(2,SL),
// DISP=(OLD,PASS)
//SYSUT2 DD DSN=QUANTDAT,UNIT=WORKDISK,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1)),DCB=(BLKSIZE=1600,LRECL=80,RECFM=FB)
/*
//PASO2 EXEC PROC=FTG1CLG
//FORT.SYSIN DD DSN=SIMBOLIC.CNEA(QUANTO),DISP=SHR
/*
//GO.FT01FO01 DD DSN=QUANTDAT,UNIT=WORKDISK,DISP=(OLD,DELETE)
//GO.FT15FO01 DD DSN=QUANTRES,UNIT=TAPE,VOL=SER=T16288,LABEL=(11,SL),
// DISP=(NEW,KEEP),DCB=(BLKSIZE=1600,LRECL=80,RECFM=FB)
/*
//
```

NOTA: El archivo de entrada de datos está definido en la primera sentencia GO, en donde se le asigna el número de referencia 1 y el archivo de salida de datos está definido en la segunda sentencia GO, en donde se le asigna el número de referencia 15; por lo tanto en el programa fuente, cuando se lean datos del primer ar-

chivo se codificará READ1 y cuando se graben resultados en el segundo archivo se codificará WRITE15. Se mencionan los DCB con los cuales se grabarán el disco auxiliar y la cinta.

Modelo 458. Ejecución de un programa grabado en la biblioteca con el nombre CURSAR, cuyos datos de entrada constituyen un archivo que reside en una cinta magnética y cuyos resultados configuran un archivo de salida que se graba en otra cinta. El archivo de entrada reside, con el nombre DATOPHYS, en el rótulo 33 de una cinta que lleva el número de carrete 67076; el archivo de salida se graba, con el nombre RESPHYS, en el rótulo 21 de otra cinta que lleva el número de carrete 15000. Ambas cintas tienen etiquetas modelo IBM.

```
//..... JOB ....
//PASO1 EXEC PROC=FTG1CLG
//FORT.SYSIN DD DSN=SIMBOLIC.CNEA(CURSAR),DISP=SHR
/*
//GO.FT02F001 DD DSN=DATOPHYS,UNIT=TAPE,VOL=SER=T67076,LABEL=(33,SL),
// DISP=(OLD,KEEP)
//GO.FT08F001 DD DSN=RESPHYS,UNIT=TAPE,VOL=SER=T15000,LABEL=(21,SL),
// DISP=(NEW,KEEP),DCB=(BLKSIZE=800,LRECL=80,RECFM=FB)
/*
//
```

NOTA: El archivo de entrada de datos está definido en la primera sentencia GO, en donde se le asigna el número de referencia 2 y el archivo de salida de datos está definido en la segunda sentencia GO, en donde se le asigna el número de referencia 8; por lo tanto en el programa fuente, cuando se lean datos del primer archivo se codificará READ2 y cuando se graben resultados en el segundo archivo se codificará WRITE8. Se menciona el DCB con el cual se grabará la cinta.

Modelo 459. Ejecución de un programa grabado en la biblioteca con el nombre AMORTIG, cuyos datos de entrada son cuatro archivos distintos que residen en rótulos consecutivos o no consecutivos de una cinta magnética y cuyos resultados son tres archivos de salida que se graban en esa misma cinta. Los archivos de entrada de nombres AMRTDAT1, AMRTDAT2, AMRTDAT3 y AMRTDAT4 están grabados en los rótulos 11, 12, 20 y 32 de la cinta que lleva el número de carrete 15116 y tiene

etiquetas modelo IBM. Con los nombres SALIDA71, SALIDA72 y SALIDA73, los resultados se graban en los r tulos 35, 36 y 37 de esa misma cinta. El problema se puede resolver en cinco pasos: en los cuatro primeros (PASO1, PASO2, PASO3 y PASO4) el utilitario IEBGENER transfiere, de a uno, los cuatro archivos de entrada de datos de la cinta al mismo disco auxiliar; en el PASO5 el procedimiento FTG1CLG ejecuta el programa obteniendo los datos de entrada del disco auxiliar y grabando los resultados en la cinta.

```
//..... JOB ....
//PASO1 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=AMRTDAT1,UNIT=TAPE,VOL=SER=T15116,LABEL=(11,SL),
// DISP=(OLD,PASS)
//SYSUT2 DD DSN=AMRTDAT1,UNIT=WORKDISK,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1)),DCB=(BLKSIZE=1200,LRECL=80,RECFM=FB)
//SYSIN DD DUMMY
/*
//PASO2 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=AMRTDAT2,UNIT=TAPE,VOL=REF=*.PASO1.SYSUT1,
// LABEL=(12,SL)
//SYSUT2 DD DSN=AMRTDAT2,VOL=REF=*.PASO1.SYSUT2,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1))
//SYSIN DD DUMMY
/*
//PASO3 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=AMRTDAT3,UNIT=TAPE,VOL=REF=*.PASO1.SYSUT1,
// LABEL=(20,SL)
//SYSUT2 DD DSN=AMRTDAT3,VOL=REF=*.PASO1.SYSUT2,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1))
//SYSIN DD DUMMY
/*
//PASO4 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=AMRTDAT4,UNIT=TAPE,VOL=REF=*.PASO1.SYSUT1,
// LABEL=(32,SL)
//SYSUT2 DD DSN=AMRTDAT4,VOL=REF=*.PASO1.SYSUT2,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1))
//SYSIN DD DUMMY
/*
//PASO5 EXEC PROC=FTG1CLG
//FORT.SYSIN DD DSN=SIMBOLIC.CNEA(AMORTIG),DISP=SHR
/*
//GO.FT10F001 DD UNIT=WORKDISK,DSN=AMRTDAT1,DISP=(OLD,DELETE)
//GO.FT13F001 DD UNIT=WORKDISK,DSN=AMRTDAT2,DISP=(OLD,DELETE)
//GO.FT18F001 DD UNIT=WORKDISK,DSN=AMRTDAT3,DISP=(OLD,DELETE)
//GO.FT20F001 DD UNIT=WORKDISK,DSN=AMRTDAT4,DISP=(OLD,DELETE)
//GO.FT14F001 DD DSN=SALIDA71,UNIT=TAPE,VOL=SER=T15116,LABEL=(35,SL),
// DISP=(NEW,PASS),DCB=(BLKSIZE=1200,LRECL=80,RECFM=FB)
```

```
//GO.FT14F002 DD DSN=SALIDA72,UNIT=TAPE,VOL=REF=*.GO.FT14F001,
// LABEL=(36,SL)
//GO.FT14F003 DD DSN=SALIDA73,UNIT=TAPE,VOL=REF=*.GO.FT14F001,
// LABEL=(37,SL),DISP=(NEW,KEEP)
/*
//
```

NOTA: Los archivos de entrada de datos están definidos en las cuatro primeras sentencias GO, en donde se les asignan los números de referencia 10, 13, 18 y 20. Por lo tanto en el programa fuente, cuando se lean datos de esos archivos se codificará READ10, READ13, READ18 y READ20 respectivamente. A su vez los archivos de salida de datos están definidos en la quinta sentencia GO y siguientes, en donde se les asigna el mismo número de referencia 14 y los números de secuencia 1, 2 y 3 respectivamente. Por consiguiente en el programa fuente, cuando se graben resultados en esos archivos se codificará una sentencia WRITE14, además de la sentencia ENDFILE14, para señalar la finalización de cada uno de ellos. Se mencionan los DCB con los cuales se grabarán el disco auxiliar y la cinta.

Modelo 460 . Ejecución de un programa grabado en la biblioteca con el nombre RATIO, cuyos datos de entrada son 4 archivos distintos que residen en rótulos consecutivos o no consecutivos de una cinta magnética y cuyos resultados son 3 archivos de salida que se graban en otra cinta. Los archivos de entrada, de nombres RATIDAT1, RATIDAT2, RATIDAT3 y RATIDAT4, están grabados en los rótulos 2, 3, 6 y 15 de la cinta que lleva el número de carrete 33129 y con los nombres RATIRES1, RATIRES2 y RATIRES3, los resultados se graban en los rótulos 15, 16 y 17 de otra cinta que lleva el número de carrete 13781. Ambas cintas tienen etiquetas modelo IBM.

```
//..... JOB ....
//PAS01 EXEC PROC=FTG1CLG
//FORT.SYSIN DD DSN=SIMBOLIC.CNEA(RATIO),DISP=SHR
/*
//GO.FT11F001 DD DSN=RATIDAT1,UNIT=TAPE,VOL=SER=T33129,LABEL=(2,SL),
// DISP=(OLD,PASS)
//GO.FT11F002 DD DSN=RATIDAT2,UNIT=TAPE,VOL=REF=*.GO.FT11F001,
// LABEL=(3,SL)
//GO.FT11F003 DD DSN=RATIDAT3,UNIT=TAPE,VOL=REF=*.GO.FT11F001,
// LABEL=(6,SL)
```

```
//GO.FT11F004 DD DSN=RATIDAT4,UNIT=TAPE,VOL=REF=*.GO.FT11F001,
// LABEL=(15,SL),DISP=(OLD,KEEP)
//GO.FT13F001 DD DSN=RATIRES1,UNIT=TAPE,VOL=SER=T13781,LABEL=(15,SL),
// DISP=(NEW,PASS),DCB=(BLKSIZE=1600,LRECL=80,RECFM=FB)
//GO.FT13F002 DD DSN=RATIRES2,UNIT=TAPE,VOL=REF=*.GO.FT13F001,
// LABEL=(16,SL)
//GO.FT13F003 DD DSN=RATIRES3,UNIT=TAPE,VOL=REF=*.GO.FT13F001,
// LABEL=(17,SL),DISP=(NEW,KEEP)
/*
//
```

NOTA: Los archivos de entrada de datos están definidos en las cuatro primeras sentencias GO, en donde se les asigna el mismo número de referencia 11 y los números de secuencia 1, 2, 3 y 4; por lo tanto en el programa fuente, cuando se lean datos de esos archivos, se codificará una sentencia READ de este tipo:

READ(11,n°de formato,END=dirección). La "dirección" significa la marca de la sentencia a la cual deberá saltar el programa para continuar la ejecución, cada vez que llega al final de uno de los archivos. A su vez los archivos de salida de datos están definidos en la quinta sentencia GO y siguientes en donde se les asigna el mismo número de referencia 13 y los números de secuencia 1, 2 y 3; por consiguiente en el programa fuente, cuando se graben resultados en esos archivos, se codificará WRITE13, además de la sentencia ENDFILE 13, para señalar la finalización de cada uno de ellos. Se menciona el DCB con el cual se grabará la cinta.

PUNTO 3.

Ejecución de módulos procedentes de FORTRAN, grabados en una biblioteca particionada y catalogada de módulos de carga de nombre REALES.CNEA, que utilizan cintas magnéticas como archivos de entrada y salida de datos, simultáneamente.

Modelo 461. Ejecución de un módulo de carga grabado en la biblioteca con el nombre MATEC, cuyos datos de entrada constituyen un archivo que reside en una cinta magnética y cuyos resultados configuran un archivo de salida que se graba en esa misma cinta. El archivo de entrada reside, con el nombre MATECDAT, en el rótulo 1 de la cinta que lleva el número de carrete 62625 y tiene etiquetas modelo IBM; el archivo de salida se graba, con el nombre MATECRES, en el rótulo 13 de esa misma cinta. El problema se puede resolver en dos pasos: en el PASO1 el utilitario IEBGENER transfiere el juego de datos MATECDAT de la cinta a un disco auxiliar; en el PASO2 se ejecuta el programa MATEC obteniéndose los datos de entrada del disco auxiliar y grabando los resultados en la cinta.

```
//..... JOB ....
//PASO1 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSIN DD DUMMY
//SYSUT1 DD UNIT=TAPE,DSN=MATECDAT,VOL=SER=T62625,LABEL=(1,SL),
// DISP=(OLD,PASS)
//SYSUT2 DD UNIT=WORKDISK,DISP=(NEW,PASS),DSN=MATECDAT,
// SPACE=(CYL,(1,1)),DCB=(BLKSIZE=1200,LRECL=80,RECFM=FB)
/*
//PASO2 EXEC PGM=MATEC
//STEPLIB DD DSN=REALES.CNEA,DISP=SHR
/*
//FT06F001 DD SYSOUT=A
//FT09F001 DD UNIT=WORKDISK,DSN=MATECDAT,DISP=(OLD,DELETE)
//FT11F001 DD UNIT=TAPE,DSN=MATECRES,VOL=SER=T62625,LABEL=(13,SL),
// DISP=(NEW,KEEP),DCB=(BLKSIZE=1200,LRECL=80,RECFM=FB)
/*
//
```

NOTA: El archivo de entrada de datos está definido en la sentencia FT09F001, en donde se le asigna el número de referencia 9 y el archivo de salida de datos está definido en la sentencia FT11F001, en donde se le asigna el número de referencia 11; por lo tanto en el programa fuente que dió origen al módulo se codificó

READ9 para leer datos del primer archivo y WRITE11 para grabar resultados en el segundo archivo. Se mencionan los DCB con los cuales se grabarán el disco auxiliar y la cinta.

Modelo 462. Ejecución de un módulo de carga grabado en la biblioteca con el nombre CEMASIT, cuyos datos de entrada constituyen un archivo que reside, con el nombre MASIDAT, en el rótulo 14 de una cinta magnética que lleva el número de carrete 76285. Con el nombre MASIRES, los resultados se graban en el rótulo 22 de otra cinta que lleva el número de carrete 00331. Ambas cintas tienen etiquetas modelo IBM.

```
//..... JOB ....
//PAS01 EXEC PGM=CEMASIT
//STEPLIB DD DSN=REALES.CNEA,DISP=SHR
/*
//FT06F001 DD SYSOUT=A
//FT03F001 DD UNIT=TAPE,DSN=MASIDAT,VOL=SER=T76285,DISP=(OLD,KEEP),
// LABEL=(14,SL)
//FT04F001 DD UNIT=TAPE,DSN=MASIRES,VOL=SER=T00331,DISP=(NEW,KEEP),
// LABEL=(22,SL),DCB=(BLKSIZE=3200,LRECL=80,RECFM=FB)
/*
//
```

NOTA: El archivo de entrada de datos está definido en la sentencia FT03F001, en donde se le asigna el número de referencia 3 y el archivo de salida de datos está definido en la sentencia FT04F001, en donde se le asigna el número de referencia 4; por lo tanto en el programa fuente que dió origen al módulo, se codificó READ3 para leer datos del primer archivo y WRITE4 para grabar resultados en el segundo archivo. Se menciona el DCB con el cual se grabará la cinta.

Modelo 463. Ejecución de un módulo de carga grabado en la biblioteca con el nombre POLITE, cuyos datos de entrada son tres archivos distintos que residen en rótulos consecutivos o no consecutivos de una cinta magnética y cuyos resultados son dos archivos de salida que se graban en esa misma cinta. Los archivos de entrada de nombres POLITEX, POLITEY y POLITEZ están grabados en los rótulos 12, 13 y 18 de la cinta que lleva el número de carrete 23881 y tiene etiquetas modelo IBM. Con los nombres POLIRES1 y POLIRES2 los resultados se graban en los rótulos

26 y 27 de esa misma cinta. El problema se puede resolver en cuatro pasos: en los tres primeros (PAS01, PAS02 y PAS03) el utilitario IEBGENER transfiere, de a uno, los tres archivos de entrada de datos de la cinta al mismo disco auxiliar; en el PAS04 se ejecuta el programa POLITE obteniéndose los datos de entrada del disco auxiliar y grabando los resultados en la cinta.

```
//..... JOB ....
//PAS01 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=POLITEX,UNIT=TAPE,VOL=SER=T23881,LABEL=(12,SL),
//  DISP=(OLD,PASS)
//SYSUT2 DD DSN=POLITEX,UNIT=WORKDISK,DISP=(NEW,PASS),
//  SPACE=(CYL,(1,1)),DCB=(BLKSIZE=1200,LRECL=80,RECFM=FB)
//SYSIN DD DUMMY
/*
//PAS02 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=POLITEY,UNIT=TAPE,VOL=REF=*.PAS01.SYSUT1,
//  LABEL=(13,SL)
//SYSUT2 DD DSN=POLITEY,VOL=REF=*.PAS01.SYSUT2,DISP=(NEW,PASS),
//  SPACE=(CYL,(1,1))
//SYSIN DD DUMMY
/*
//PAS03 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=POLITEZ,UNIT=TAPE,VOL=REF=*.PAS01.SYSUT1,
//  LABEL=(18,SL)
//SYSUT2 DD DSN=POLITEZ,VOL=REF=*.PAS01.SYSUT2,DISP=(NEW,PASS),
//  SPACE=(CYL,(1,1))
//SYSIN DD DUMMY
/*
//PAS04 EXEC PGM=POLITE
//STEPLIB DD DSN=REALES.CNEA,DISP=SHR
/*
//FT06F001 DD SYSOUT=A
//FT10F001 DD DSN=POLITEX,UNIT=WORKDISK,DISP=(OLD,DELETE)
//FT11F001 DD DSN=POLITEY,UNIT=WORKDISK,DISP=(OLD,DELETE)
//FT15F001 DD DSN=POLITEZ,UNIT=WORKDISK,DISP=(OLD,DELETE)
//FT21F001 DD DSN=POLIRES1,UNIT=TAPE,VOL=SER=T23881,LABEL=(26,SL),
//  DISP=(NEW,PASS),DCB=(BLKSIZE=2000,LRECL=80,RECFM=FB)
//FT21F002 DD DSN=POLIRES2,UNIT=TAPE,VOL=REF=*.FT21F001,LABEL=(27,SL),
//  DISP=(NEW,KEEP)
/*
//
```

NOTA: Los archivos de entrada de datos están definidos en las sentencias FT10F001, FT11F001 y FT15F001, en donde se les asignan los números de referencia 10, 11 y 15; por lo tanto en el programa fuente que dió origen al módulo, se codificó READ10, READ11 y READ15 respectivamente, para leer datos de esos archivos.

A su vez los archivos de salida de datos están definidos en las sentencias FT21F001 y FT21F002, en donde se les asigna el mismo número de referencia 21 y los números de secuencia 1 y 2; por consiguiente en el programa fuente se codificó WRITE21 para grabar resultados en esos archivos, además de la sentencia ENDFILE21 para señalar la finalización de cada uno de ellos. Se mencionan los DCB con los cuales se grabarán el disco auxiliar y la cinta.

Modelo 464. Ejecución de un módulo de carga grabado en la biblioteca con el nombre ARMIN, cuyos datos de entrada son cuatro archivos distintos que residen en rótulos consecutivos o no consecutivos de una cinta magnética y cuyos resultados son tres archivos de salida que se graban en otra cinta. Los archivos de entrada de nombres ARMIDAT1, ARMIDAT2, ARMIDAT3 y ARMIDAT4, están grabados en los rótulos 16, 17, 28 y 39 de la cinta que lleva el número de carrete 12900 y con los nombres ARMIRES1, ARMIRES2 y ARMIRES3, los resultados se graban en los rótulos 20, 21 y 22 de otra cinta que lleva el número de carrete 01069. Ambas cintas tienen etiquetas modelo IBM.

```
//..... JOB ....
//PAS01 EXEC PGM=ARMIN
//STEPLIB DD DSN=REALES.CNEA,DISP=SHR
/*
//FT06F001 DD SYSOUT=A
//FT03F001 DD DSN=ARMIDAT1,UNIT=TAPE,VOL=SER=T12900,LABEL=(16,SL),
// DISP=(OLD,PASS)
//FT03F002 DD DSN=ARMIDAT2,UNIT=TAPE,VOL=REF=*.FT03F001,LABEL=(17,SL)
//FT03F003 DD DSN=ARMIDAT3,UNIT=TAPE,VOL=REF=*.FT03F001,LABEL=(28,SL)
//FT03F004 DD DSN=ARMIDAT4,UNIT=TAPE,VOL=REF=*.FT03F001,LABEL=(39,SL),
// DISP=(OLD,KEEP)
//FT09F001 DD DSN=ARMIRES1,UNIT=TAPE,VOL=SER=T01069,LABEL=(20,SL),
// DISP=(NEW,PASS),DCB=(BLKSIZE=2800,LRECL=80,RECFM=FB)
//FT09F002 DD DSN=ARMIRES2,UNIT=TAPE,VOL=REF=*.FT09F001,LABEL=(21,SL)
//FT09F003 DD DSN=ARMIRES3,UNIT=TAPE,VOL=REF=*.FT09F001,LABEL=(22,SL),
// DISP=(NEW,KEEP)
/*
//
```

NOTA: Los archivos de entrada de datos están definidos en las sentencias

FT03F001, FT03F002, FT03F003 y FT03F004, en donde se les asigna el mismo número de referencia 3 y los números de secuencia 1, 2, 3 y 4; por lo tanto en el pro-

grama fuente que dió origen al módulo se codificó una sentencia READ de este tipo: READ(3,n°de formato,END=dirección), para leer datos de esos archivos. La "dirección" es la marca de la sentencia a la cual deberá saltar el programa para continuar la ejecución, cada vez que llega al final de uno de los archivos. A su vez los archivos de salida de datos están definidos en las sentencias FT09F001, FT09F002 y FT09F003, en donde se les asigna el mismo número de referencia 9 y los números de secuencia 1, 2 y 3; por consiguiente en el programa fuente se codificó WRITE9 para grabar resultados en esos archivos, además de la sentencia ENDFILE9 para señalar la finalización de cada uno de ellos. Se menciona el DCB con el cual se grabará la cinta.

PUNTO 4.

Ejecución de programas en FORTRAN grabados en cintas magnéticas, que utilizan cintas como archivos de entrada y salida de datos, simultáneamente.

Modelo 465. Ejecución de un programa en lenguaje simbólico FORTRAN grabado, con el nombre MARINE, en el rótulo 18 de una cinta magnética que lleva el número de carrete 08844, cuyos datos de entrada constituyen un archivo que reside en esa misma cinta y cuyos resultados configuran un archivo de salida que se graba también en esa cinta. El archivo de entrada reside, con el nombre MARINDAT, en el rótulo 7 y el archivo de salida se graba, con el nombre MARINRES, en el rótulo 31. La cinta tiene etiquetas modelo IBM. El problema se puede resolver en dos pasos: en el PASO1 el utilitario IEBGENER transfiere el juego de datos MARINDAT de la cinta a un disco auxiliar; en el PASO2 el procedimiento FTG1CLG ejecuta el programa obteniendo los datos de entrada del disco auxiliar y grabando los resultados en la cinta.

```
//..... JOB ....
//PASO1 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSIN DD DUMMY
//SYSUT1 DD DSN=MARINDAT,UNIT=TAPE,VOL=SER=T08844,LABEL=(7,SL),
//  DISP=(OLD,PASS)
//SYSUT2 DD DSN=MARINDAT,UNIT=WORKDISK,DISP=(NEW,PASS),
//  SPACE=(CYL,(1,1)),DCB=(BLKSIZE=2000,LRECL=80,RECFM=FB)
/*
//PASO2 EXEC PROC=FTG1CLG
//FORT.SYSIN DD DSN=MARINE,UNIT=TAPE,VOL=SER=T08844,LABEL=(18,SL),
//  DISP=(OLD,PASS)
/*
//GO.FT10F001 DD DSN=MARINDAT,UNIT=WORKDISK,DISP=(OLD,DELETE)
//GO.FT08F001 DD DSN=MARINRES,UNIT=TAPE,VOL=SER=T08844,LABEL=(31,SL),
//  DISP=(NEW,KEEP),DCB=(BLKSIZE=800,LRECL=80,RECFM=FB)
/*
//
```

NOTA: El archivo de entrada de datos está definido en la primera sentencia GO, en donde se le asigna el número de referencia 10 y el archivo de salida está definido en la segunda sentencia GO, en donde se le asigna el número de referencia 8; por lo tanto en el programa fuente se codificará READ10 y WRITE8 respectivamente,

para leer y grabar datos en esos archivos. Se mencionan los DCB con los cuales se grabarán el disco auxiliar y la cinta.

Modelo 466. Ejecución de un programa en lenguaje absoluto de FORTRAN grabado, con el nombre PLUG, en el rótulo 13 de una cinta magnética que lleva el número de carrete 53771, cuyos datos de entrada constituyen un archivo que reside en esa misma cinta y cuyos resultados configuran un archivo de salida que se graba también en esa cinta. El archivo de entrada reside, con el nombre PLUGDAT, en el rótulo 6 y el archivo de salida se graba, con el nombre PLUGRES en el rótulo 25. La cinta tiene etiquetas modelo IBM. El problema se puede resolver en dos pasos: en el PASO1 el utilitario IEBGENER transfiere el juego de datos PLUGDAT de la cinta a un disco auxiliar; en el PASO2 el procedimiento FTG1LG ejecuta el programa obteniendo los datos de entrada del disco auxiliar y grabando los resultados en la cinta.

```
//..... JOB ....
//PASO1 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSIN DD DUMMY
//SYSUT1 DD DSN=PLUGDAT,UNIT=TAPE,VOL=SER=T53771,LABEL=(6,SL),
//  DISP=(OLD,PASS)
//SYSUT2 DD DSN=PLUGDAT,UNIT=WORKDISK,DISP=(NEW,PASS),
//  SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3200,LRECL=80,RECFM=FB)
/*
//PASO2 EXEC PROC=FTG1LG
//LKED.SYSIN DD DSN=PLUG,UNIT=TAPE,VOL=SER=T53771,LABEL=(13,SL),
//  DISP=(OLD,PASS)
/*
//GO.FT09F001 DD DSN=PLUGDAT,UNIT=WORKDISK,DISP=(OLD,DELETE)
//GO.FT19F001 DD DSN=PLUGRES,UNIT=TAPE,VOL=SER=T53771,LABEL=(25,SL),
//  DISP=(NEW,KEEP),DCB=(BLKSIZE=1600,LRECL=80,RECFM=FB)
/*
//
```

NOTA: El archivo de entrada de datos está definido en la primera sentencia GO, en donde se le asigna el número de referencia 9 y el archivo de salida está definido en la segunda sentencia GO, en donde se le asigna el número de referencia 19; por lo tanto en el programa fuente que dió origen al módulo grabado en la cinta, se codificó READ9 y WRITE19 respectivamente, para leer y grabar datos en esos archi-

vos. Se mencionan los DCB con los cuales se grabarán el disco auxiliar y la cinta.

Modelo 467 . Ejecución de un programa en lenguaje simbólico FORTRAN grabado, con el nombre PIXE, en el rótulo 4 de una cinta magnética que lleva el número de carrete 02166, cuyos datos de entrada constituyen un archivo que reside en esa misma cinta y cuyos resultados configuran un archivo de salida que se graba en otra cinta. El archivo de entrada reside, con el nombre PIXEDAT, en el rótulo 17 de la cinta 02166 y el archivo de salida se graba, con el nombre PIXERES en el rótulo 3 de la cinta que lleva el número de carrete 02177. Ambas cintas tienen etiquetas modelo IBM.

```
//..... JOB ....
//PAS01 EXEC PROC=FTG1CLG
//FORT.SYSIN DD DSN=PIXE,UNIT=TAPE,VOL=SER=T02166,LABEL=(4,SL),
// DISP=(OLD,PASS)
/*
//GO.FT04F001 DD DSN=PIXEDAT,UNIT=TAPE,VOL=SER=T02166,LABEL=(17,SL),
// DISP=(OLD,KEEP)
//GO.FT20F001 DD DSN=PIXERES,UNIT=TAPE,VOL=SER=T02177,LABEL=(3,SL),
// DISP=(NEW,KEEP),DCB=(BLKSIZE=1200,LRECL=80,RECFM=FB)
/*
//
```

NOTA: El archivo de entrada de datos está definido en la primera sentencia GO, en donde se le asigna el número de referencia 4 y el archivo de salida está definido en la segunda sentencia GO, en donde se le asigna el número de referencia 20; por lo tanto en el programa fuente se codificará READ4 y WRITE20 respectivamente, para leer y grabar datos en esos archivos. Se menciona el DCB con el cual se grabará la cinta.

Modelo 468 . Ejecución de un programa en lenguaje absoluto de FORTRAN grabado, con el nombre MASTER, en el rótulo 8 de una cinta magnética que lleva el número de carrete 90909, cuyos datos de entrada constituyen un archivo que reside en esa misma cinta y cuyos resultados configuran un archivo de salida que se graba en otra cinta. El archivo de entrada reside, con el nombre MASTDAT, en el rótulo 20 de la cinta 90909 y el archivo de salida se graba, con el nombre MASTRES en el

rótulo 16 de la cinta que lleva el número de carrete 11122 . Ambas cintas tienen etiquetas modelo IBM.

```
//..... JOB ....
//PAS01 EXEC PROC=FTG1LG
//LKED.SYSIN DD DSN=MASTER,UNIT=TAPE,VOL=SER=T90909,LABEL=(8,SL),
//  DISP=(OLD,PASS)
/*
//GO.FT20F001 DD DSN=MASTDAT,UNIT=TAPE,VOL=SER=T90909,LABEL=(20,SL),
//  DISP=(OLD,KEEP)
//GO.FT03F001 DD DSN=MASTRES,UNIT=TAPE,VOL=SER=T11122,LABEL=(16,SL),
//  DISP=(NEW,KEEP),DCB=(BLKSIZE=2400,LRECL=80,RECFM=FB)
/*
//
```

NOTA: El archivo de entrada de datos está definido en la primera sentencia GO, en donde se le asigna el número de referencia 20 y el archivo de salida está definido en la segunda sentencia GO, en donde se le asigna el número de referencia 3; por lo tanto en el programa fuente que dió origen al módulo grabado en la cinta, se codificó READ20 y WRITE3 respectivamente, para leer y grabar datos en esos archivos. Se menciona el DCB con el cual se grabará la cinta.

Modelo 469 . Ejecución de un programa en lenguaje simbólico FORTRAN grabado, con el nombre MATEM, en el rótulo 35 de una cinta magnética que lleva el número de carrete 15051. Los datos de entrada constituyen un archivo que reside, con el nombre MATEMDAT, en el rótulo 4 de una cinta que lleva el número de carrete 22227 y los resultados configuran un archivo de salida que se graba, con el nombre MATEMRES, en el rótulo 11 de esta última. Ambas cintas tienen etiquetas modelo IBM. El problema se puede resolver en dos pasos: en el PAS01 el utilitario IEBGENER transfiere el juego de datos MATEMDAT de la cinta 22227 a un disco auxiliar; en el PAS02 el procedimiento FTG1CLG ejecuta el programa obteniendo los datos de entrada del disco auxiliar y grabando los resultados en la cinta 22227.

```
//..... JOB ....
//PAS01 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSIN DD DUMMY
//SYSUT1 DD DSN=MATEMDAT,UNIT=TAPE,VOL=SER=T22227,LABEL=(4,SL),
//  DISP=(OLD,PASS)
//SYSUT2 DD DSN=MATEMDAT,UNIT=WORKDISK,DISP=(NEW,PASS),
//  SPACE=(CYL,(1,1)),DCB=(BLKSIZE=800,LRECL=80,RECFM=FB)
```



```

/*
//PASO2 EXEC PROC=FTG1CLG
//FORT.SYSIN DD DSN=MATEM,UNIT=TAPE,VOL=SER=T15051,LABEL=(35,SL),
// DISP=(OLD,KEEP)
/*
//GO.FT18F001 DD DSN=MATEMDAT,UNIT=WORKDISK,DISP=(OLD,DELETE)
//GO.FT20F001 DD DSN=MATEMRES,UNIT=TAPE,VOL=SER=T22227,LABEL=(11,SL),
// DISP=(NEW,KEEP),DCB=(BLKSIZE=1200,LRECL=80,RECFM=FB)
/*
//

```

NOTA: El archivo de entrada de datos está definido en la primera sentencia GO, en donde se le asigna el número de referencia 18 y el archivo de salida está definido en la segunda sentencia GO, en donde se le asigna el número de referencia 20; por lo tanto en el programa fuente, se codificará READ18 y WRITE20 respectivamente, para leer y grabar datos en esos archivos. Se mencionan los DCB con los cuales se grabarán el disco auxiliar y la cinta.

Modelo 470 . Ejecución de un programa en lenguaje absoluto de FORTRAN, grabado, con el nombre CODIGO, en el rótulo 14 de una cinta magnética que lleva el número de carrete 36666. Los datos de entrada constituyen un archivo que reside, con el nombre CDGDATOS, en el rótulo 7 de una cinta que lleva el número de carrete 80888 y los resultados configuran un archivo de salida que se graba, con el nombre CDGRESUL, en el rótulo 28 de esta última. Ambas cintas tienen etiquetas modelo IBM. El problema se puede resolver en dos pasos: en el PASO1 el utilitario IEBGENER transfiere el juego de datos CDGDATOS de la cinta 80888 a un disco auxiliar; en el PASO2 el procedimiento FTG1LG ejecuta el programa obteniendo los datos de entrada del disco auxiliar y grabando resultados también en la cinta 80888.

```

//..... JOB ....
//PASO1 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSIN DD DUMMY
//SYSUT1 DD DSN=CDGDATOS,UNIT=TAPE,VOL=SER=T80888,LABEL=(7,SL),
// DISP=(OLD,PASS)
//SYSUT2 DD DSN=CDGDATOS,UNIT=WORKDISK,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1)),DCB=(BLKSIZE=1600,LRECL=80,RECFM=FB)
/*
//PASO2 EXEC PROC=FTG1LG

```

```
//LKED.SYSIN DD DSN=CODIGO,UNIT=TAPE,VOL=SER=T36666,LABEL=(14,SL),
// DISP=(OLD,KEEP)
/*
//GO.FT17F001 DD DSN=CDG DATOS,UNIT=WORKDISK,DISP=(OLD,DELETE)
//GO.FT19F001 DD DSN=CDGRESUL,UNIT=TAPE,VOL=SER=T80888,LABEL=(28,SL),
// DISP=(NEW,KEEP),DCB=(BLKSIZE=800,LRECL=80,RECFM=FB)
/*
//
```

NOTA: El archivo de entrada de datos está definido en la primera sentencia GO, en donde se le asigna el número de referencia 17 y el archivo de salida está definido en la segunda sentencia GO, en donde se le asigna el número de referencia 19; por lo tanto en el programa fuente que dió origen al módulo grabado en la cinta, se codificó READ17 y WRITE19 respectivamente, para leer y grabar datos en esos archivos. Se mencionan los DCB con los cuales se grabarán el disco auxiliar y la cinta.

Modelo 471 . Ejecución de un programa en lenguaje simbólico FORTRAN grabado, con el nombre BATERIA, en el rótulo 15 de una cinta magnética que lleva el número de carrete 32233. Los datos de entrada constituyen un archivo que reside, con el nombre BATEDAT, en el rótulo 2 de una cinta que lleva el número de carrete 16789 y los resultados configuran un archivo de salida que se graba, con el nombre BATERES, en el rótulo 40 de la cinta 32233 en que está grabado el programa. Ambas cintas tienen etiquetas modelo IBM.

```
//..... JOB ....
//PAS01 EXEC PROC=FTG1CLG
//FORT.SYSIN DD DSN=BATERIA,UNIT=TAPE,VOL=SER=T32233,LABEL=(15,SL),
// DISP=(OLD,PASS)
/*
//GO.FT18F001 DD DSN=BATEDAT,UNIT=TAPE,VOL=SER=T16789,LABEL=(2,SL),
// DISP=(OLD,KEEP)
//GO.FT12F001 DD DSN=BATERES,UNIT=TAPE,VOL=SER=T32233,LABEL=(40,SL),
// DISP=(NEW,KEEP),DCB=(BLKSIZE=2000,LRECL=80,RECFM=FB)
/*
//
```

NOTA: El archivo de entrada de datos está definido en la primera sentencia GO, en donde se le asigna el número de referencia 18 y el archivo de salida está definido en la segunda sentencia GO, en donde se le asigna el número de referencia 12; por

lo tanto en el programa fuente, se codificará READ18 y WRITE12 respectivamente, para leer y grabar datos en esos archivos. Se menciona el DCB con el cual se grabará la cinta.

Modelo 472 . Ejecución de un programa en lenguaje absoluto de FORTRAN grabado, con el nombre SELF, en el rótulo 25 de una cinta magnética que lleva el número de carrete 32101. Los datos de entrada constituyen un archivo que reside, con el nombre SELFDAT, en el rótulo 12 de una cinta que lleva el número de carrete 37777 y los resultados configuran un archivo de salida que se graba, con el nombre SELFRES, en el rótulo 43 de la cinta 32101 en que está grabado el programa. Ambas cintas tienen etiquetas modelo IBM.

```
//..... JOB ....
//PAS01 EXEC PROC=FTG1LG
//LKED.SYSIN DD DSN=SELF,UNIT=TAPE,VOL=SER=T32101,LABEL=(25,SL),
// DISP=(OLD,PASS)
/*
//GO.FT04F001 DD DSN=SELFDAT,UNIT=TAPE,VOL=SER=T37777,LABEL=(12,SL),
// DISP=(OLD,KEEP)
//GO.FT09F001 DD DSN=SELFRES,UNIT=TAPE,VOL=SER=T32101,LABEL=(43,SL),
// DISP=(NEW,KEEP),DCB=(BLKSIZE=2000,LRECL=80,RECFM=FB)
/*
//
```

NOTA: El archivo de entrada de datos está definido en la primera sentencia GO, en donde se le asigna el número de referencia 4 y el archivo de salida está definido en la segunda sentencia GO, en donde se le asigna el número de referencia 9; por lo tanto en el programa fuente que dió origen al módulo grabado en la cinta, se codificó READ4 y WRITE9 respectivamente, para leer y grabar datos en esos archivos. Se menciona el DCB con el cual se grabará la cinta.

Modelo 473 . Ejecución de un programa en lenguaje simbólico FORTRAN grabado, con el nombre IMPACTO, en el rótulo 15 de una cinta magnética que lleva el número de carrete 11221, cuyos datos de entrada son tres archivos distintos que residen en rótulos consecutivos o no consecutivos de esa misma cinta y cuyos resultados son cuatro archivos de salida que se graban también en esa cinta. Los archivos de entrada residen, con los nombres PACDAT1, PACDAT2 y PACDAT3, en los rótulos 2, 8 y

9 y los archivos de salida se graban, con los nombres PACRES1, PACRES2, PACRES3 y PACRES4 en los r tulos 38, 39, 40 y 41. La cinta tiene etiquetas modelo IBM. El problema se puede resolver en cuatro pasos: en los tres primeros (PASO1, PASO2 y PASO3) el utilitario IEBGENER transfiere, de a uno, los tres archivos de entrada de datos de la cinta al mismo disco auxiliar; en el PASO4 el procedimiento FTG1CLG ejecuta el programa obteniendo los datos de entrada del disco auxiliar y grabando los resultados en la cinta.

```
//..... JOB ....
//PASO1 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=PACDAT1,UNIT=TAPE,VOL=SER=T11221,LABEL=(2,SL);
// DISP=(OLD,PASS)
//SYSUT2 DD DSN=PACDAT1,UNIT=WORKDISK,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1)),DCB=(BLKSIZE=2800,LRECL=80,RECFM=FB)
//SYSIN DD DUMMY
/*
//PASO2 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=PACDAT2,UNIT=TAPE,VOL=REF=*.PASO1.SYSUT1,
// LABEL=(3,SL)
//SYSUT2 DD DSN=PACDAT2,VOL=REF=*.PASO1.SYSUT2,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1))
//SYSIN DD DUMMY
/*
//PASO3 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=PACDAT3,UNIT=TAPE,VOL=REF=*.PASO1.SYSUT1,
// LABEL=(9,SL)
//SYSUT2 DD DSN=PACDAT3,VOL=REF=*.PASO1.SYSUT2,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1))
//SYSIN DD DUMMY
/*
//PASO4 EXEC PROC=FTG1CLG
//FORT.SYSIN DD DSN=IMPACTO,UNIT=TAPE,VOL=SER=T11221,LABEL=(15,SL),
// DISP=(OLD,PASS)
/*
//GO.FT10F001 DD DSN=PACDAT1,UNIT=WORKDISK,DISP=(OLD,DELETE)
//GO.FT19F001 DD DSN=PACDAT2,UNIT=WORKDISK,DISP=(OLD,DELETE)
//GO.FT20F001 DD DSN=PACDAT3,UNIT=WORKDISK,DISP=(OLD,DELETE)
//GO.FT16F001 DD DSN=PACRES1,UNIT=TAPE,VOL=SER=T11221,LABEL=(38,SL),
// DISP=(NEW,PASS),DCB=(BLKSIZE=2400,LRECL=80,RECFM=FB)
//GO.FT16F002 DD DSN=PACRES2,UNIT=TAPE,VOL=REF=*.GO.FT16F001,
// LABEL=(39,SL)
//GO.FT16F003 DD DSN=PACRES3,UNIT=TAPE,VOL=REF=*.GO.FT16F001,
// LABEL=(40,SL)
//GO.FT16F004 DD DSN=PACRES4,UNIT=TAPE,VOL=REF=*.GO.FT16F001,
// LABEL=(41,SL),DISP=(NEW,KEEP)
/*
//
```

NOTA: Los archivos de entrada de datos están definidos en las tres primeras sentencias GO, en donde se les asignan los números de referencia 10, 19 y 20. Por lo tanto en el programa fuente, cuando se lean datos de esos archivos se codificará READ10, READ19 y READ20 respectivamente. A su vez los archivos de salida de datos están definidos en la cuarta sentencia GO y siguientes, en donde se les asigna el mismo número de referencia 16 y los números de secuencia 1, 2, 3 y 4 respectivamente. Por consiguiente en el programa fuente, cuando se graben resultados en esos archivos se codificará una sentencia WRITE16, además de la sentencia ENDFILE16, para señalar la finalización de cada uno de ellos. Se mencionan los DCB con los cuales se grabarán el disco auxiliar y la cinta.

Modelo 474 . Ejecución de un programa en lenguaje absoluto de FORTRAN grabado, con el nombre COLISION, en el rótulo 26 de una cinta magnética que lleva el número de carrete 65641, cuyos datos de entrada son tres archivos que residen en róticos consecutivos o no consecutivos de esa misma cinta y cuyos resultados son tres archivos de salida que se graban también en esa cinta. Los archivos de entrada residen, con los nombres COLIDAT1, COLIDAT2 y COLIDAT3, en los róticos 16, 17 y 21 y los archivos de salida se graban, con los nombres COLIRES1, COLIRES2 y COLIRES3, en los róticos 30, 31 y 32. La cinta tiene etiquetas modelo IBM. El problema se puede resolver en cuatro pasos: en los tres primeros (PASO1, PASO2 y PASO3) el utilitario IEBGENER transfiere, de a uno, los tres archivos de entrada de datos de la cinta al mismo disco auxiliar; en el PASO4 el procedimiento FTGLLG ejecuta el programa obteniendo los datos de entrada del disco auxiliar y grabando los resultados en la cinta.

```
//..... JOB ....
//PASO1 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=COLIDAT1,UNIT=TAPE,VOL=SER=T65641,LABEL=(16,SL),
// DISP=(OLD,PASS)
//SYSUT2 DD DSN=COLIDAT1,UNIT=WORKDISK,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1)),DCB=(BLKSIZE=3200,LRECL=80,RECFM=FB)
//SYSIN DD DUMMY
/*
```

```

//PASO2 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=COLIDAT2,UNIT=TAPE,VOL=REF=*.PASO1.SYSUT1,
// LABEL=(17,SL)
//SYSUT2 DD DSN=COLIDAT2,VOL=REF=*.PASO1.SYSUT2,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1))
//SYSIN DD DUMMY
/*
//PASO3 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=COLIDAT3,UNIT=TAPE,VOL=REF=*.PASO1.SYSUT1,
// LABEL=(21,SL)
//SYSUT2 DD DSN=COLIDAT3,VOL=REF=*.PASO1.SYSUT2,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1))
//SYSIN DD DUMMY
/*
//PASO4 EXEC PROC=FTG1LG
//LKED.SYSIN DD DSN=COLISION,UNIT=TAPE,VOL=SER=T65641,LABEL=(26,SL),
// DISP=(OLD,PASS)
/*
//GO.FT18F001 DD DSN=COLIDAT1,UNIT=WORKDISK,DISP=(OLD,DELETE)
//GO.FT13F001 DD DSN=COLIDAT2,UNIT=WORKDISK,DISP=(OLD,DELETE)
//GO.FT11F001 DD DSN=COLIDAT3,UNIT=WORKDISK,DISP=(OLD,DELETE)
//GO.FT17F001 DD DSN=COLIRES1,UNIT=TAPE,VOL=SER=T65641,LABEL=(30,SL),
// DISP=(NEW,PASS),DCB=(BLKSIZE=2000,LRECL=80,RECFM=FB)
//GO.FT17F002 DD DSN=COLIRES2,UNIT=TAPE,VOL=REF=*.GO.FT17F001,
// LABEL=(31,SL)
//GO.FT17F003 DD DSN=COLIRES3,UNIT=TAPE,VOL=REF=*.GO.FT17F001,
// LABEL=(32,SL), DISP=(NEW,KEEP)
/*
//

```

NOTA: Los archivos de entrada de datos están definidos en las tres primeras sentencias GO, en donde se les asignan los números de referencia 18, 13 y 11. Por lo tanto en el programa fuente que dió origen al módulo grabado en la cinta, para leer datos de esos archivos se codificó READ18, READ13 y READ11 respectivamente. A su vez los archivos de salida de datos están definidos en la cuarta sentencia GO y siguientes, en donde se les asigna el mismo número de referencia 17 y los números de secuencia 1, 2 y 3 respectivamente. Por consiguiente en el programa fuente, para grabar resultados en esos archivos se codificó una sentencia WRITE17, además de la sentencia ENDFILE17, para señalar la finalización de cada uno de ellos. Se mencionan los DCB con los cuales se grabarán el disco auxiliar y la cinta.

Modelo 475. Ejecución de un programa en lenguaje simbólico FORTRAN grabado, con

el nombre ECUACION, en el rótulo 19 de una cinta magnética que lleva el número de carrete 30903, cuyos datos de entrada son cuatro archivos distintos que residen en rótulos consecutivos o no consecutivos de esa misma cinta y cuyos resultados son tres archivos de salida que se graban en otra cinta. Los archivos de entrada residen, con los nombres INPUT1, INPUT2, INPUT3 e INPUT4 en los rótulos 42, 43, 51 y 54 de la cinta 30903 y los archivos de salida se graban, con los nombres OUTPUT1, OUTPUT2 y OUTPUT3, en los rótulos 61, 62 y 63 de la cinta que lleva el número de carrete 55855. Ambas cintas tienen etiquetas modelo IBM.

```
//..... JOB ....
//PASO1 EXEC PROC=FTG1CLG
//FORT.SYSIN DD DSN=ECUACION,UNIT=TAPE,VOL=SER=T30903,LABEL=(19,SL),
// DISP=(OLD,PASS)
/*
//GO.FT10F001 DD DSN=INPUT1,UNIT=TAPE,VOL=SER=T30903,LABEL=(42,SL),
// DISP=(OLD,PASS)
//GO.FT10F002 DD DSN=INPUT2,UNIT=TAPE,VOL=REF=*.GO.FT10F001,
// LABEL=(43,SL)
//GO.FT10F003 DD DSN=INPUT3,UNIT=TAPE,VOL=REF=*.GO.FT10F001,
// LABEL=(51,SL)
//GO.FT10F004 DD DSN=INPUT4,UNIT=TAPE,VOL=REF=*.GO.FT10F001,
// LABEL=(54,SL),DISP=(OLD,KEEP)
//GO.FT18F001 DD DSN=OUTPUT1,UNIT=TAPE,VOL=SER=T55855,LABEL=(61,SL),
// DISP=(NEW,PASS),DCB=(BLKSIZE=1600,LRECL=80,RECFM=FB)
//GO.FT18F002 DD DSN=OUTPUT2,UNIT=TAPE,VOL=REF=*.GO.FT18F001,
// LABEL=(62,SL)
//GO.FT18F003 DD DSN=OUTPUT3,UNIT=TAPE,VOL=REF=*.GO.FT18F001,
// LABEL=(63,SL),DISP=(NEW,KEEP)
/*
//
```

NOTA: Los archivos de entrada de datos están definidos en las cuatro primeras sentencias GO, en donde se les asigna el mismo número de referencia 10 y los números de secuencia 1, 2, 3 y 4 respectivamente. Por lo tanto en el programa fuente, cuando se lean datos de esos archivos se codificará una sentencia READ de este tipo: READ(10,n°de formato,END=dirección). La "dirección" es la marca de la sentencia a la cual deberá saltar el programa para continuar la ejecución, cada vez que llega al final de uno de los archivos. A su vez los archivos de salida de datos están definidos en la quinta sentencia GO y siguientes, en donde se les asigna el mismo número de referencia 18 y los números de secuencia 1, 2 y 3 res-

pectivamente. Por consiguiente en el programa fuente, cuando se graben resultados en esos archivos, se codificará una sentencia WRITE18, además de la sentencia ENDFILE18, para señalar la finalización de cada uno de ellos. Se menciona el DCB con el cual se grabará la cinta.

Modelo 476 . Ejecución de un programa en lenguaje absoluto de FORTRAN grabado, con el nombre CLASE, en el rótulo 8 de una cinta magnética que lleva el número de carrete 76677, cuyos datos de entrada son tres archivos distintos que residen en rótulos consecutivos o no consecutivos de esa misma cinta y cuyos resultados son cuatro archivos de salida que se graban en otra cinta. Los archivos de entrada residen, con los nombres CLASDAT1, CLASDAT2 y CLASDAT3, en los rótulos 21, 25 y 26 de la cinta 76677 y los archivos de salida se graban, con los nombres CLASRES1, CLASRES2, CLASRES3 y CLASRES4, en los rótulos 2, 3, 4 y 5 de la cinta que lleva el número de carrete 10001. Ambas cintas tienen etiquetas modelo IBM.

```
//..... JOB ....
//PAS01 EXEC PROC=FTG1LG
//LKED.SYSIN DD DSN=CLASE,UNIT=TAPE,VOL=SER=T76677,LABEL=(8,SL),
// DISP=(OLD,PASS)
/*
//GO.FT11F001 DD DSN=CLASDAT1,UNIT=TAPE,VOL=SER=T76677,LABEL=(21,SL),
// DISP=(OLD,PASS)
//GO.FT11F002 DD DSN=CLASDAT2,UNIT=TAPE,VOL=REF=*.GO.FT11F001,
// LABEL=(25,SL)
//GO.FT11F003 DD DSN=CLASDAT3,UNIT=TAPE,VOL=REF=*.GO.FT11F001,
// LABEL=(26,SL),DISP=(OLD,KEEP)
//GO.FT09F001 DD DSN=CLASRES1,UNIT=TAPE,VOL=SER=T10001,LABEL=(2,SL),
// DISP=(NEW,PASS),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//GO.FT09F002 DD DSN=CLASRES2,UNIT=TAPE,VOL=REF=*.GO.FT09F001,
// LABEL=(3,SL)
//GO.FT09F003 DD DSN=CLASRES3,UNIT=TAPE,VOL=REF=*.GO.FT09F001,
// LABEL=(4,SL)
//GO.FT09F004 DD DSN=CLASRES4,UNIT=TAPE,VOL=REF=*.GO.FT09F001,
// LABEL=(5,SL),DISP=(NEW,KEEP)
/*
//
```

NOTA: Los archivos de entrada de datos están definidos en las tres primeras sentencias GO, en donde se les asigna el mismo número de referencia 11 y los números de secuencia 1, 2 y 3 respectivamente. Por lo tanto en el programa fuente que dió origen al módulo grabado en la cinta, para leer datos de esos archivos se codifi-

có una sentencia READ de este tipo: READ(11,n°de formato,END=dirección). La dirección es la marca de la sentencia a la cual deberá saltar el programa para continuar la ejecución, cada vez que llega al final de uno de los archivos. A su vez los archivos de salida de datos están definidos en la cuarta sentencia GO y siguientes, en donde se les asigna el mismo número de referencia 9 y los números de secuencia 1, 2, 3 y 4 respectivamente. Por consiguiente en el programa fuente, para grabar resultados en esos archivos se codificó una sentencia WRITE9, además de la sentencia ENDFILE9, para señalar la finalización de cada uno de ellos. Se menciona el DCB con el cual se grabará la cinta.

Modelo 477 . Ejecución de un programa en lenguaje simbólico FORTRAN grabado, con el nombre ESTADO, en el rótulo 5 de una cinta magnética que lleva el número de carrete 00044, cuyos datos de entrada son tres archivos distintos que residen en rótulos consecutivos o no consecutivos de otra cinta y cuyos resultados son cuatro archivos de salida que se graban en esta última cinta. Los archivos de entrada residen, con los nombres DATEST1, DATEST2 y DATEST3, en los rótulos 2, 3 y 18 de una cinta que lleva el número de carrete 55522 y los archivos de salida se graban, con los nombres RESEST1, RESEST2, RESEST3 y RESEST4, en los rótulos 25, 26, 27 y 28 de esta última. Ambas cintas tienen etiquetas modelo IBM. El problema se puede resolver en cuatro pasos: en los tres primeros (PASO1, PASO2 y PASO3) el utilitario IEBGENER transfiere, de a uno, los tres archivos de entrada de datos de la cinta al mismo disco auxiliar; en el PASO4 el procedimiento FTG1CLG ejecuta el programa obteniendo los datos de entrada del disco auxiliar y grabando los resultados en la cinta 55522.

```
//..... JOB ....
//PASO1 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=DATEST1,UNIT=TAPE,VOL=SER=T55522,LABEL=(2,SL),
// DISP=(OLD,PASS)
//SYSUT2 DD DSN=DATEST1,UNIT=WORKDISK,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1)),DCB=(BLKSIZE=1200,LRECL=80,RECFM=FB)
//SYSIN DD DUMMY
/*
```

```

//PASO2 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=DATEST2,UNIT=TAPE,VOL=REF=*.PASO1.SYSUT1,
// LABEL=(3,SL)
//SYSUT2 DD DSN=DATEST2,VOL=REF=*.PASO1.SYSUT2,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1))
//SYSIN DD DUMMY
/*
//PASO3 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=DATEST3,UNIT=TAPE,VOL=REF=*.PASO1.SYSUT1,
// LABEL=(18,SL)
//SYSUT2 DD DSN=DATEST3,VOL=REF=*.PASO1.SYSUT2,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1))
//SYSIN DD DUMMY
/*
//PASO4 EXEC PROC=FTG1CLG
//FORT.SYSIN DD DSN=ESTADO,UNIT=TAPE,VOL=SER=T00044,LABEL=(5,SL),
// DISP=(OLD,KEEP)
/*
//GO.FT01F001 DD DSN=DATEST1,UNIT=WORKDISK,DISP=(OLD,DELETE)
//GO.FT03F001 DD DSN=DATEST2,UNIT=WORKDISK,DISP=(OLD,DELETE)
//GO.FT08F001 DD DSN=DATEST3,UNIT=WORKDISK,DISP=(OLD,DELETE)
//GO.FT15F001 DD DSN=RESEST1,UNIT=TAPE,VOL=SER=T55522,LABEL=(25,SL),
// DISP=(NEW,PASS),DCB=(BLKSIZE=1600,LRECL=80,RECFM=FB)
//GO.FT15F002 DD DSN=RESEST2,UNIT=TAPE,VOL=REF=*.GO.FT15F001,
// LABEL=(26,SL)
//GO.FT15F003 DD DSN=RESEST3,UNIT=TAPE,VOL=REF=*.GO.FT15F001,
// LABEL=(27,SL)
//GO.FT15F004 DD DSN=RESEST4,UNIT=TAPE,VOL=REF=*.GO.FT15F001,
// LABEL=(28,SL),DISP=(NEW,KEEP)
/*
//

```

NOTA: Los archivos de entrada de datos están definidos en las tres primeras sentencias GO, en donde se les asignan los números de referencia 1, 3 y 8. Por lo tanto en el programa fuente, cuando se lean datos de esos archivos se codificará READ1, READ3 y READ8 respectivamente. A su vez los archivos de salida de datos están definidos en la cuarta sentencia GO y siguientes, en donde se les asigna el mismo número de referencia 15 y los números de secuencia 1, 2, 3 y 4 respectivamente. Por consiguiente en el programa fuente, cuando se graben resultados en esos archivos, se codificará una sentencia WRITE15, además de la sentencia ENDFILE15, para señalar la finalización de cada uno de ellos. Se mencionan los DCB con los cuales se grabarán el disco auxiliar y la cinta.

Modelo 478 . Ejecución de un programa en lenguaje absoluto de FORTRAN grabado,

con el nombre EXAMEN, en el rótulo 21 de una cinta magnética que lleva el número de carrete 39393, cuyos datos de entrada son tres archivos distintos que residen en rótulos consecutivos o no consecutivos de otra cinta y cuyos resultados son tres archivos de salida que se graban en esta última cinta. Los archivos de entrada residen, con los nombres EXADAT1, EXADAT2 y EXADAT3, en los rótulos 6, 7 y 16 de una cinta que lleva el número de carrete 60001 y los archivos de salida se graban, con los nombres EXARES1, EXARES2 y EXARES3, en los rótulos 32, 33 y 34 de esta última. Ambas cintas tienen etiquetas modelo IBM. El problema se puede resolver en cuatro pasos: en los tres primeros (PASO1, PASO2 y PASO3) el utilitario IEBGENER transfiere, de a uno, los tres archivos de entrada de datos de la cinta al mismo disco auxiliar; en el PASO4 el procedimiento FTG1LG ejecuta el programa obteniendo los datos de entrada del disco auxiliar y grabando los resultados en la cinta 60001.

```
//..... JOB ....
//PASO1 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=EXADAT1,UNIT=TAPE,VOL=SER=T60001,LABEL=(6,SL),
//  DISP=(OLD,PASS)
//SYSUT2 DD DSN=EXADAT1,UNIT=WORKDISK,DISP=(NEW,PASS),
//  SPACE=(CYL,(1,1)),DCB=(BLKSIZE=800,LRECL=80,RECFM=FB)
//SYSIN DD DUMMY
/*
//PASO2 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=EXADAT2,UNIT=TAPE,VOL=REF=*.PASO1.SYSUT1,
//  LABEL=(7,SL)
//SYSUT2 DD DSN=EXADAT2,VOL=REF=*.PASO1.SYSUT2,DISP=(NEW,PASS),
//  SPACE=(CYL,(1,1))
//SYSIN DD DUMMY
/*
//PASO3 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=EXADAT3,UNIT=TAPE,VOL=REF=*.PASO1.SYSUT1,
//  LABEL=(16,SL)
//SYSUT2 DD DSN=EXADAT3,VOL=REF=*.PASO1.SYSUT2,DISP=(NEW,PASS),
//  SPACE=(CYL,(1,1))
//SYSIN DD DUMMY
/*
//PASO4 EXEC PROC=FTG1LG
//LKED.SYSIN DD DSN=EXAMEN,UNIT=TAPE,VOL=SER=T39393,LABEL=(21,SL),
//  DISP=(OLD,KEEP)
/*
//GO.FT20F001 DD DSN=EXADAT1,UNIT=WORKDISK,DISP=(OLD,DELETE)
```

```
//GO.FT14F001 DD DSN=EXADAT2,UNIT=WORKDISK,DISP=(OLD,DELETE)
//GO.FT18F001 DD DSN=EXADAT3,UNIT=WORKDISK,DISP=(OLD,DELETE)
//GO.FT17F001 DD DSN=EXARES1,UNIT=TAPE,VOL=SER=T60001,LABEL=(32,SL),
//  DISP=(NEW,PASS),DCB=(BLKSIZE=800,LRECL=80,RECFM=FB)
//GO.FT17F002 DD DSN=EXARES2,UNIT=TAPE,VOL=REF=*.GO.FT17F001,
//  LABEL=(33,SL)
//GO.FT17F003 DD DSN=EXARES3,UNIT=TAPE,VOL=REF=*.GO.FT17F001,
//  LABEL=(34,SL),DISP=(NEW,KEEP)
/*
//
```

NOTA: Los archivos de entrada de datos están definidos en las tres primeras sentencias GO, en donde se les asignan los números de referencia 20, 14 y 18. Por lo tanto en el programa fuente que dió origen al módulo grabado en la cinta, para leer datos de esos archivos se codificó READ20, READ14 y READ18 respectivamente. A su vez los archivos de salida de datos están definidos en la cuarta sentencia GO y siguientes, en donde se les asigna el mismo número de referencia 17 y los números de secuencia 1, 2 y 3 respectivamente. Por consiguiente en el programa fuente, para grabar resultados en esos archivos se codificó una sentencia WRITE17, además de la sentencia ENDFILE17, para señalar la finalización de cada uno de ellos. Se mencionan los DCB con los cuales se grabarán el disco auxiliar y la cinta.

Modelo 479 . Ejecución de un programa en lenguaje simbólico FORTRAN grabado, con el nombre APERTURA, en el rótulo 10 de una cinta magnética que lleva el número de carrete 28082, cuyos datos de entrada son cuatro archivos que residen en rótulos consecutivos o no consecutivos de otra cinta y cuyos resultados son dos archivos de salida que se graban en la cinta 28082 en que está el programa. Los archivos de entrada residen, con los nombres APERDAT1, APERDAT2, APERDAT3 y APERDAT4, en los rótulos 9, 11, 12 y 23 de una cinta que lleva el número de carrete 76543 y los archivos de salida se graban, con los nombres RESULT1 y RESULT2, en los rótulos 15 y 16 de la cinta 28082. Ambas cintas tienen etiquetas modelo IBM.

```
//..... JOB ....
//PAS01 EXEC PROC=FTG1CLG
//FORT.SYSIN DD DSN=APERTURA,UNIT=TAPE,VOL=SER=T28082,LABEL=(10,SL),
//  DISP=(OLD,PASS)
```

```

/*
//GO.FT14F001 DD DSN=APERDAT1,UNIT=TAPE,VOL=SER=T76543,LABEL=(9,SL),
// DISP=(OLD,PASS)
//GO.FT14F002 DD DSN=APERDAT2,UNIT=TAPE,VOL=REF=*.GO.FT14F001,
// LABEL=(11,SL)
//GO.FT14F003 DD DSN=APERDAT3,UNIT=TAPE,VOL=REF=*.GO.FT14F001,
// LABEL=(12,SL)
//GO.FT14F004 DD DSN=APERDAT4,UNIT=TAPE,VOL=REF=*.GO.FT14F001,
// LABEL=(23,SL),DISP=(OLD,KEEP)
//GO.FT08F001 DD DSN=RESULT1,UNIT=TAPE,VOL=SER=T28082,LABEL=(15,SL),
// DISP=(NEW,PASS),DCB=(BLKSIZE=2000,LRECL=80,RECFM=FB)
//GO.FT08F002 DD DSN=RESULT2,UNIT=TAPE,VOL=REF=*.GO.FT08F001,
// LABEL=(16,SL),DISP=(NEW,KEEP)
/*
//

```

NOTA: Los archivos de entrada de datos están definidos en las cuatro primeras sentencias GO, en donde se les asigna el mismo número de referencia 14 y los números de secuencia 1, 2, 3 y 4 respectivamente. Por lo tanto en el programa fuente, cuando se lean datos de esos archivos, se codificará una sentencia READ de este tipo: READ(14,nº de formato,END=dirección). La "dirección" es la marca de la sentencia a la cual deberá saltar el programa para continuar la ejecución, cada vez que llega al final de uno de los archivos. A su vez los archivos de salida de datos están definidos en la quinta y sexta sentencia GO, en donde se les asigna el mismo número de referencia 8 y los números de secuencia 1 y 2 respectivamente. Por consiguiente en el programa fuente, cuando se graben resultados en esos archivos, se codificará una sentencia WRITE8, además de la sentencia ENDFILE8, para señalar la finalización de cada uno de ellos. Se menciona el DCB con el cual se grabará la cinta.

Modelo 480 . Ejecución de un programa en lenguaje absoluto de FORTRAN grabado, con el nombre FLAG, en el rótulo 32 de una cinta magnética que lleva el número de carrete 76500, cuyos datos de entrada son tres archivos que residen en rótulos consecutivos o no consecutivos de otra cinta y cuyos resultados son tres archivos de salida que se graban en la cinta 76500 en que está el programa. Los archivos de entrada residen, con los nombres DATFLAG1, DATFLAG2 y DATFLAG3, en los rótulos 5, 6 y 12 de una cinta que lleva el número de carrete 40044 y los archivos de sa-

lida se graban, con los nombres FLAGRES1, FLAGRES2 y FLAGRES3, en los r tulos 43, 44 y 45 de la cinta 76500. Ambas cintas tienen etiquetas modelo IBM.

```
//..... JOB ....
//PASO1 EXEC PROC=FTG1LG
//LKED.SYSIN DD DSN=FLAG,UNIT=TAPE,VOL=SER=T76500,LABEL=(32,SL),
// DISP=(OLD,PASS)
/*
//GO.FT09F001 DD DSN=DATFLAG1,UNIT=TAPE,VOL=SER=T40044,LABEL=(5,SL),
// DISP=(OLD,PASS)
//GO.FT09F002 DD DSN=DATFLAG2,UNIT=TAPE,VOL=REF=*.GO.FT09F001,
// LABEL=(6,SL)
//GO.FT09F003 DD DSN=DATFLAG3,UNIT=TAPE,VOL=REF=*.GO.FT09F001,
// LABEL=(12,SL),DISP=(OLD,KEEP)
//GO.FT13F001 DD DSN=FLACRES1,UNIT=TAPE,VOL=SER=T76500,LABEL=(43,SL),
// DISP=(NEW,PASS),DCB=(BLKSIZE=3200,LRECL=80,RECFM=FB)
//GO.FT13F002 DD DSN=FLAGRES2,UNIT=TAPE,VOL=REF=*.GO.FT13F001,
// LABEL=(44,SL)
//GO.FT13F003 DD DSN=FLAGRES3,UNIT=TAPE,VOL=REF=*.GO.FT13F001,
// LABEL=(45,SL),DISP=(NEW,KEEP)
/*
//
```

NOTA: Los archivos de entrada de datos est n definidos en las tres primeras sentencias GO, en donde se les asigna el mismo n mero de referencia 9 y los n meros de secuencia 1, 2 y 3 respectivamente. Por lo tanto en el programa fuente que di  origen al m dulo grabado en la cinta, para leer datos de esos archivos se codific  una sentencia READ de este tipo: READ(9,n  de formato,END=direcci n). La "direcci n" es la marca de la sentencia a la cual deber  saltar el programa para continuar la ejecuci n, cada vez que llega al final de uno de los archivos. A su vez los archivos de salida est n definidos en la cuarta sentencia GO y siguientes, en donde se les asigna el mismo n mero de referencia 13 y los n meros de secuencia 1, 2 y 3 respectivamente. Por consiguiente en el programa fuente, para grabar resultados en esos archivos se codific  una sentencia WRITE13, adem s de la sentencia ENDFILE13, para se alar la finalizaci n de cada uno de ellos. Se menciona el DCB con el cual se grabar  la cinta.

CAPITULO XII

MODELOS PARA LA UTILIZACION DE CINTAS MAGNETICAS
COMO ARCHIVOS DE MODULOS DE PROGRAMAS O DE
BIBLIOTECAS PARTICIONADAS CUANDO NO SE DISPONE
DE ESPACIO PROPIO EN UNA UNIDAD DE DISCOS

PUNTO 1.

Utilización de cintas magnéticas como archivo del módulo de un programa escrito en dos lenguajes simbólicos distintos.

Modelo 481 . Obtención del módulo, archivo del mismo en cinta magnética y su ulterior ejecución, de un programa en FORTRAN que se presenta en tarjetas perforadas e invoca una rutina en ASSEMBLER, también en tarjetas perforadas. Se dispone de la cinta que lleva el número de carrete 13325 y tiene etiquetas modelo IBM, para grabar el módulo en el rótulo 16. El problema se puede resolver en tres pasos: en el PASO1 el procedimiento FTG1C compila el programa en FORTRAN y graba el módulo en la cinta, con el nombre RESONAN; en el PASO2 el procedimiento ASMFC compila la rutina en ASSEMBLER y graba el nuevo módulo, como una ampliación del archivo anterior, con el mismo nombre y en el mismo rótulo; en el PASO3 el procedimiento FTG1LG ejecuta el módulo completo, obteniéndolo de la cinta.

```
//..... JOB ....
//PASO1 EXEC PROC=FTG1C,PARM.FORT=DECK
//FORT.SYSPUNCH DD DSN=RESONAN,UNIT=TAPE,VOL=SER=T13325,LABEL=(16,SL),
// DISP=(NEW,PASS),DCB=BLKSIZE=80
//FORT.SYSIN DD *
  Aquí tarjetas perforadas en FORTRAN
/*
//PASO2 EXEC PROC=ASMFC,PARM.ASM=DECK
//ASM.SYSPUNCH DD DSN=RESONAN,UNIT=TAPE,VOL=SER=T13325,LABEL=(16,SL),
// DISP=(MOD,KEEP),DCB=BLKSIZE=80
//ASM.SYSIN DD *
  Aquí tarjetas perforadas en ASSEMBLER
/*
//PASO3 EXEC PROC=FTG1LG
//LKED.SYSIN DD DSN=RESONAN,UNIT=TAPE,VOL=SER=T13325,LABEL=(16,SL),
// DISP=(OLD,KEEP)
/*
//GO.SYSIN DD *
  Aquí tarjetas perforadas de DATOS
/*
//
```

Modelo 482 . Obtención del módulo, archivo del mismo en cinta magnética y su ulterior ejecución, de un programa en FORTRAN que se presenta en tarjetas perforadas e invoca una rutina en ASSEMBLER que reside, con el nombre FORMAL, en el rótulo 18 de una cinta que lleva el número de carrete 05500. Se dispone además de

otra cinta que lleva el número de carrete 07700, para grabar el módulo en el rótulo 5. Ambas cintas tienen etiquetas modelo IBM. El problema se puede resolver en tres pasos: en el PASO1 el procedimiento FTG1C compila el programa en FORTRAN y graba el módulo, con el nombre STEP, en la cinta 07700; en el PASO2 el procedimiento ASMFC compila la rutina en ASSEMBLER y graba el nuevo módulo, como una ampliación del archivo anterior, con el mismo nombre y en el mismo rótulo; en el PASO3 el procedimiento FTG1LG ejecuta el módulo completo, obteniéndolo de la cinta.

```
//..... JOB ....
//PASO1 EXEC PROC=FTG1C,PARM.FORT=DECK
//FORT.SYSPUNCH DD DSN=STEP,UNIT=TAPE,VOL=SER=T07700,LABEL=(5,SL),
//  DISP=(NEW,PASS),DCB=BLKSIZE=80
//FORT.SYSIN DD *
Aquí tarjetas perforadas en FORTRAN
/*
//PASO2 EXEC PROC=ASMFC,PARM.ASM=DECK
//ASM.SYSPUNCH DD DSN=STEP,UNIT=TAPE,VOL=SER=T07700,LABEL=(5,SL),
//  DISP=(MOD,KEEP),DCB=BLKSIZE=80
//ASM.SYSIN DD DSN=FORMAL,UNIT=TAPE,VOL=SER=T05500,LABEL=(18,SL),
//  DISP=(OLD,KEEP)
/*
//PASO3 EXEC PROC=FTG1LG
//LKED.SYSIN DD DSN=STEP,UNIT=TAPE,VOL=SER=T07700,LABEL=(5,SL),
//  DISP=(OLD,KEEP)
/*
//GO.SYSIN DD *
Aquí tarjetas perforadas de DATOS
/*
//
```

Modelo 483. Obtención del módulo, archivo del mismo en cinta magnética y su ulterior ejecución, de un programa en FORTRAN que reside, con el nombre POWER, en el rótulo 10 de una cinta que lleva el número de carrete 12221 e invoca una rutina en ASSEMBLER que se presenta en tarjetas perforadas. Se dispone además de otra cinta que lleva el número de carrete 30003, para grabar el módulo en el rótulo 28. Ambas cintas tienen etiquetas modelo IBM. El problema se puede resolver en tres pasos: en el PASO1 el procedimiento FTG1C compila el programa en FORTRAN y graba el módulo, con el nombre PGRMPOW, en la cinta 30003; en el PASO2 el procedimiento ASMFC compila la rutina en ASSEMBLER y graba el nuevo módulo, como una

ampliación del archivo anterior, con el mismo nombre y en el mismo rótulo; en el PASO3 el procedimiento FTG1LG ejecuta el módulo completo, obteniéndolo de la cinta.

```
//..... JOB ....
//PASO1 EXEC PROC=FTG1C,PARM.FORT=DECK
//FORT.SYSPUNCH DD DSN=PGRMPOW,UNIT=TAPE,VOL=SER=T30003,LABEL=(28,SL),
//  DISP=(NEW,PASS),DCB=BLKSIZE=80
//FORT.SYSIN DD DSN=POWER,UNIT=TAPE,VOL=SER=T12221,LABEL=(10,SL),
//  DISP=(OLD,KEEP)
/*
//PASO2 EXEC PROC=ASMFC,PARM.ASM=DECK
//ASM.SYSPUNCH DD DSN=PGRMPOW,UNIT=TAPE,VOL=SER=T30003,LABEL=(28,SL),
//  DISP=(MOD,KEEP),DCB=BLKSIZE=80
//ASM.SYSIN DD *
  Aquí tarjetas perforadas en ASSEMBLER
/*
//PASO3 EXEC PROC=FTG1LG
//LKED.SYSIN DD DSN=PGRMPOW,UNIT=TAPE,VOL=SER=T30003,LABEL=(28,SL),
//  DISP=(OLD,KEEP)
/*
//GO.SYSIN DD *
  Aquí tarjetas perforadas de DATOS
/*
//
```

Modelo 484 . Obtención del módulo, archivo del mismo en cinta magnética y su ulterior ejecución, de un programa en ASSEMBLER que se presenta en tarjetas perforadas e invoca una rutina en FORTRAN, también en tarjetas perforadas. Se dispone de la cinta que lleva el número de carrete 65432 y tiene etiquetas modelo IBM, para grabar el módulo en el rótulo 11. El problema se puede resolver en tres pasos: en el PASO1 el procedimiento ASMFC compila el programa en ASSEMBLER y graba el módulo en la cinta, con el nombre PERSONAL; en el PASO2 el procedimiento FTG1C compila la rutina en FORTRAN y graba el nuevo módulo, como una ampliación del archivo anterior, con el mismo nombre y en el mismo rótulo. A continuación se define un procedimiento de nombre GOLKDFTR y en el PASO3 se invoca dicho procedimiento que ejecuta el módulo completo, obteniéndolo de la cinta.

```
//..... JOB ....
//PASO1 EXEC PROC=ASMFC,PARM.ASM=DECK
//ASM.SYSPUNCH DD DSN=PERSONAL,UNIT=TAPE,VOL=SER=T65432,LABEL=(11,SL),
//  DISP=(NEW,PASS),DCB=BLKSIZE=80
//ASM.SYSIN DD *
  Aquí tarjetas perforadas en ASSEMBLER
```

```

/*
//PAS02 EXEC PROC=FTG1C,PARM.FORT=DECK
//FORT.SYSPUNCH DD DSN=PERSONAL,UNIT=TAPE,VOL=SER=T65432,LABEL=(11,SL),
//  DISP=(MOD,KEEP),DCB=BLKSIZE=80
//FORT.SYSIN DD *
  Aquí tarjetas perforadas en FORTRAN
/*
//GOLKDFTR PROC
//LKED EXEC PGM=IEWL,REGION=96K,PARM=(XREF,LET,LIST)
//SYSLIB DD DSN=SYS1.FORTLIB,DISP=SHR
//SYSLMOD DD DSN=&GOSET(MAIN),DISP=(NEW,PASS),UNIT=SYSDA,
//  SPACE=(1024,(20,10,1),RLSE),DCB=BLKSIZE=1024
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD UNIT=SYSDA,SPACE=(1024,(100,10),RLSE),DCB=BLKSIZE=1024,
//  DSN=&SYSUT1
//SYSLIN DD DDNAME=SYSIN,DCB=BLKSIZE=80
//GO EXEC PGM=*.LKED.SYSLMOD,COND=(4,LT,LKED)
//FT05F001 DD DDNAME=SYSIN
//FT06F001 DD SYSOUT=A
//  PEND
//PAS03 EXEC PROC=GOLKDFTR
//LKED.SYSIN DD DSN=PERSONAL,UNIT=TAPE,VOL=SER=T65432,LABEL=(11,SL),
//  DISP=(OLD,KEEP)
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas de DATOS
/*
//

```

NOTA: En el programa principal ASSEMBLER se ha abierto un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

Modelo 485 . Obtención del módulo, archivo del mismo en cinta magnética y su ulterior ejecución, de un programa en ASSEMBLER que se presenta en tarjetas perforadas e invoca una rutina en FORTRAN que reside, con el nombre BAJAENER, en el rótulo 22 de una cinta que lleva el número de carrete 30033. Se dispone además de otra cinta que lleva el número de carrete 66666, para grabar el módulo en el rótulo 9. Ambas cintas tienen etiquetas modelo IBM. El problema se puede resolver en tres pasos: en el PAS01 el procedimiento ASMFC compila el programa en ASSEMBLER y graba el módulo, con el nombre ENERGIA, en la cinta 66666; en el PAS02 el procedimiento FTG1C compila la rutina en FORTRAN y graba el nuevo módulo, como una ampliación del archivo anterior, con el mismo nombre y en el mismo rótulo. A continuación se define un procedimiento de nombre GOLKDFTR y en el PAS03 se invoca

dicho procedimiento que ejecuta el módulo completo, obteniéndolo de la cinta.

```
//..... JOB ....
//PAS01 EXEC PROC=ASMFC,PARM.ASM=DECK
//ASM.SYSPUNCH DD DSN=ENERGIA,UNIT=TAPE,VOL=SER=T66666,LABEL=(9,SL),
// DISP=(NEW,PASS),DCB=BLKSIZE=80
//ASM.SYSIN DD *
  Aquí tarjetas perforadas en ASSEMBLER
/*
//PAS02 EXEC PROC=FTGIC,PARM.FORT=DECK
//FORT.SYSPUNCH DD DSN=ENERGIA,UNIT=TAPE,VOL=SER=T66666,LABEL=(9,SL),
// DISP=(MOD,KEEP),DCB=BLKSIZE=80
//FORT.SYSIN DD DSN=BAJAENER,UNIT=TAPE,VOL=SER=T30033,LABEL=(22,SL),
// DISP=(OLD,KEEP)
/*
//GOLKDFTR PROC
//LKED EXEC PGM=IEWL,REGION=96K,PARM=(XREF,LET,LIST)
//SYSLIB DD DSN=SYS1.FORTLIB,DISP=SHR
//SYSLMOD DD DSN=&GOSET(MAIN),DISP=(NEW,PASS),UNIT=SYSDA,
// SPACE=(1024,(20,10,1),RLSE),DCB=BLKSIZE=1024
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD UNIT=SYSDA,SPACE=(1024,(100,10),RLSE),DCB=BLKSIZE=1024,
// DSN=&SYSUT1
//SYSLIN DD DDNAME=SYSIN,DCB=BLKSIZE=80
//GO EXEC PGM=*.LKED.SYSLMOD,COND=(4,LT,LKED)
//FT05F001 DD DDNAME=SYSIN
//FT06F001 DD SYSOUT=A
// PEND
//PAS03 EXEC PROC=GOLKDFTR
//LKED.SYSIN DD DSN=ENERGIA,UNIT=TAPE,VOL=SER=T66666,LABEL=(9,SL),
// DISP=(OLD,KEEP)
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: En el programa principal ASSEMBLER se ha abierto un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

Modelo 486. Obtención del módulo, archivo del mismo en cinta magnética y su ulterior ejecución, de un programa en ASSEMBLER que reside, con el nombre YRAST, en el rótulo 24 de una cinta que lleva el número de carrete 00400 e invoca una rutina en FORTRAN que se presenta en tarjetas perforadas. Se dispone además de otra cinta que lleva el número de carrete 53332, para grabar el módulo en el rótulo 50. Ambas cintas tienen etiquetas modelo IBM. El problema se puede resolver en tres pasos: en el PAS01 el procedimiento ASMFC compila el programa en ASSEMBLER y

graba el módulo, con el nombre LIFETIME, en la cinta 53332; en el PASO2 el procedimiento FTG1C compila la rutina en FORTRAN y graba el nuevo módulo, como una ampliación del archivo anterior, con el mismo nombre y en el mismo rótulo. A continuación se define un procedimiento de nombre GOLKDFTR y en el PASO3 se invoca dicho procedimiento que ejecuta el módulo completo, obteniéndolo de la cinta.

```
//..... JOB ....
//PASO1 EXEC PROC=ASMFC,PARM.ASM=DECK
//ASM.SYSPUNCH DD DSN=LIFETIME,UNIT=TAPE,VOL=SER=T53332,LABEL=(50,SL),
//  DISP=(NEW,PASS),DCB=BLKSIZE=80
//ASM.SYSIN DD DSN=YRAST,UNIT=TAPE,VOL=SER=T00400,LABEL=(24,SL),
//  DISP=(OLD,KEEP)
/*
//PASO2 EXEC PROC=FTG1C,PARM.FORT=DECK
//FORT.SYSPUNCH DD DSN=LIFETIME,UNIT=TAPE,VOL=SER=T53332,LABEL=(50,SL),
//  DISP=(MOD,KEEP),DCB=BLKSIZE=80
//FORT.SYSIN DD *
  Aquí tarjetas perforadas en FORTRAN
/*
//GOLKDFTR PROC
//LKED EXEC PGM=IEWL,REGION=96K,PARM=(XREF,LET,LIST)
//SYSLIB DD DSNAME=SYS1.FORTLIB,DISP=SHR
//SYSLMOD DD DSNAME=&COSET(MAIN),DISP=(NEW,PASS),UNIT=SYSDA,
//  SPACE=(1024,(20,10,1),RLSE),DCB=BLKSIZE=1024
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD UNIT=SYSDA,SPACE=(1024,(100,10),RLSE),DCB=BLKSIZE=1024,
//  DSNAME=&SYSUT1
//SYSLIN DD DDNAME=SYSIN,DCB=BLKSIZE=80
//GO EXEC PGM=*.LKED.SYSLMOD,COND=(4,LT,LKED)
//FT05F001 DD DDNAME=SYSIN
//FT06F001 DD SYSOUT=A
//  PEND
//PASO3 EXEC PROC=GOLKDFTR
//LKED.SYSIN DD DSN=LIFETIME,UNIT=TAPE,VOL=SER=T53332,LABEL=(50,SL),
//  DISP=(OLD,KEEP)
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
  Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: En el programa principal ASSEMBLER se ha abierto un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO.

PUNTO 2.

Creación y uso de una biblioteca particionada de programas en lenguaje simbólico, de nombre SIMBOLIC.TAPE y su archivo en el rótulo 1 de una cinta magnética que lleva el número de carrete 56600 y tiene etiquetas modelo IBM.

Modelo 487. Creación de la biblioteca y su carga con programas, rutinas y funciones de nombres FLASH, ALFA, ARGON, CASO27, FINITO y COLECC, que se presentan en tarjetas perforadas. El problema se puede resolver en dos pasos: en el PASO1 el utilitario IEBUPDTE crea una biblioteca temporaria particionada que reside en un disco auxiliar; carga los programas que obtiene de las tarjetas perforadas y numera los registros de 100 en 100 a partir del número 50. En el PASO2 el utilitario IEBCOPY transfiere la biblioteca completa, incluido su directorio, del disco auxiliar a la cinta dispuesta para su archivo.

```
//..... JOB ....
//PASO1 EXEC PGM=IEBUPDTE,PARM=NEW
//SYSPRINT DD SYSOUT=A
//SYSUT2 DD DSN=SIMBOLIC.TAPE,UNIT=WORKDISK,DISP=(NEW,PASS),
// SPACE=(CYL,(5,2,3)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//SYSIN DD *
./ ADD NAME=FLASH
./ NUMBER NEW1=50,INCR=100
Aquí tarjetas perforadas en lenguaje simbólico
./ ADD NAME=ALFA
./ NUMBER NEW1=50,INCR=100
Aquí tarjetas perforadas en lenguaje simbólico
./ ADD NAME=ARGON
./ NUMBER NEW1=50,INCR=100
Aquí tarjetas perforadas en lenguaje simbólico
./ ADD NAME=CASO27
./ NUMBER NEW1=50,INCR=100
Aquí tarjetas perforadas en lenguaje simbólico
./ ADD NAME=FINITO
./ NUMBER NEW1=50,INCR=100
Aquí tarjetas perforadas en lenguaje simbólico
./ ADD NAME=COLECC
./ NUMBER NEW1=50,INCR=100
Aquí tarjetas perforadas en lenguaje simbólico
./ ENDUP
/*
//PASO2 EXEC PGM=IEBCOPY
//SYSPRINT DD SYSOUT=A
//ENTRADA DD DSN=SIMBOLIC.TAPE,UNIT=WORKDISK,DISP=(OLD,DELETE)
//SALIDA DD DSN=SIMBOLIC.TAPE,UNIT=TAPE,VOL=SER=T56600,LABEL=(1,SL)
//SYSIN DD *
```

```

C  O=SALIDA,I=ENTRADA
/*
//

```

NOTA: Se menciona el DCB con el cual se grabará el disco auxiliar.

Modelo 488 . Creación de la biblioteca y su carga con programas, rutinas y funciones que están grabadas, con los nombres GAUGE, PRUEBA7, LOCAL y SIMPLE en los rótulos 7, 24, 25 y 31 de la cinta que lleva el número de carrete 10088 y tiene etiquetas modelo IBM. El problema se puede resolver en cinco pasos: en el PASO1 el utilitario IEBUPDTE crea una biblioteca temporaria particionada que reside en un disco auxiliar y carga el primer programa; en los pasos siguientes (PASO2, PASO3 y PASO4) el mismo utilitario carga, de a uno, el resto de los programas en la biblioteca recién creada. En el PASO5 el utilitario IEBCOPY transfiere la biblioteca completa, incluido su directorio, del disco auxiliar a la cinta dispuesta para su archivo.

```

//..... JOB ....
//PASO1 EXEC PGM=IEBUPDTE,PARM=MOD
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=GAUGE,UNIT=TAPE,VOL=SER=T10088,LABEL=(7,SL),
//  DISP=(OLD,PASS)
//SYSUT2 DD DSN=SIMBOLIC.TAPE,UNIT=WORKDISK,DISP=(NEW,PASS),
//  SPACE=(CYL,(5,2,3)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//SYSIN DD *
./  REPRO NEW=PO,MEMBER=GAUGE,LIST=ALL
./  ENDUP
/*
//PASO2 EXEC PGM=IEBUPDTE,PARM=MOD
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=PRUEBA7,UNIT=TAPE,VOL=REF=*.PASO1.SYSUT1,
//  LABEL=(24,SL)
//SYSUT2 DD DSN=SIMBOLIC.TAPE,UNIT=WORKDISK,DISP=(OLD,PASS)
//SYSIN DD *
./  REPRO NEW=PO,MEMBER=PRUEBA7,LIST=ALL
./  ENDUP
/*
//PASO3 EXEC PGM=IEBUPDTE,PARM=MOD
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=LOCAL,UNIT=TAPE,VOL=REF=*.PASO1.SYSUT1,LABEL=(25,SL)
//SYSUT2 DD DSN=SIMBOLIC.TAPE,UNIT=WORKDISK,DISP=(OLD,PASS)
//SYSIN DD *
./  REPRO NEW=PO,MEMBER=LOCAL,LIST=ALL
./  ENDUP
/*
//PASO4 EXEC PGM=IEBUPDTE,PARM=MOD

```



```
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMPLE,UNIT=TAPE,VOL=REF=*.PAS01.SYSUT1,LABEL=(31,SL),
// DISP=(OLD,KEEP)
//SYSUT2 DD DSN=SIMBOLIC.TAPE,UNIT=WORKDISK,DISP=(OLD,PASS)
//SYSIN DD *
./ REPRO NEW=PO,MEMBER=SIMPLE,LIST=ALL
./ ENDUP
/*
//PAS05 EXEC PGM=IEBCOPY
//SYSPRINT DD SYSOUT=A
//ENTRADA DD DSN=SIMBOLIC.TAPE,UNIT=WORKDISK,DISP=(OLD,DELETE)
//SALIDA DD DSN=SIMBOLIC.TAPE,UNIT=TAPE,VOL=SER=T56600,LABEL=(1,SL)
//SYSIN DD *
C O=SALIDA,I=ENTRADA
/*
//
```

NOTA: Se menciona el DCB con el cual se grabará el disco auxiliar.

Modelo 489 . Grabación en la biblioteca, ya creada y conservada en cinta magnética, de nuevos programas, rutinas y funciones que se presentan en tarjetas perforadas. El problema se puede resolver en tres pasos: en el PAS01 el utilitario IEBCOPY transfiere la biblioteca completa, incluido su directorio, de la cinta a un disco auxiliar. En el PAS02 el utilitario IEBUPDTE graba los programas, con los nombres HOMO, CONTROL y VALIDITY, numerando los registros de 100 en 100 a partir del número 50, en la biblioteca que está en el disco auxiliar. En el PAS03 el utilitario IEBCOPY transfiere la biblioteca modificada, del disco auxiliar a la cinta dispuesta para su archivo.

```
//..... JOB ....
//PAS01 EXEC PGM=IEBCOPY
//SYSPRINT DD SYSOUT=A
//ENTRADA DD DSN=SIMBOLIC.TAPE,UNIT=TAPE,VOL=SER=T56600,LABEL=(1,SL),
// DISP=(OLD,KEEP)
//SALIDA DD DSN=SIMBOLIC.TAPE,UNIT=WORKDISK,DISP=(NEW,PASS),
// SPACE=(CYL,(5,2,3)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//SYSIN DD *
C O=SALIDA,I=ENTRADA
/*
//PAS02 EXEC PGM=IEBUPDTE,PARM=NEW
//SYSPRINT DD SYSOUT=A
//SYSUT2 DD DSN=SIMBOLIC.TAPE,UNIT=WORKDISK,DISP=(OLD,PASS)
//SYSIN DD *
./ ADD NAME=HOMO
./ NUMBER NEW1=50,INCR=100
Aquí tarjetas perforadas en lenguaje simbólico
./ ADD NAME=CONTROL
```

```

./ NUMBER NEW1=50,INCR=100
Aquí tarjetas perforadas en lenguaje simbólico
./ ADD NAME=VALIDITY
./ NUMBER NEW1=50,INCR=100
Aquí tarjetas perforadas en lenguaje simbólico
./ ENDUP
/*
//PASO3 EXEC PGM=IEBCOPY
//SYSPRINT DD SYSOUT=A
//ENTRADA DD DSN=SIMBOLIC.TAPE,UNIT=WORKDISK,DISP=(OLD,DELETE)
//SALIDA DD DSN=SIMBOLIC.TAPE,UNIT=TAPE,VOL=SER=T56600,LABEL=(1,SL)
//SYSIN DD *
C O=SALIDA,I=ENTRADA
/*
//

```

NOTA: Se menciona el DCB con el cual se grabará el disco auxiliar.

Modelo 490 . Grabación en la biblioteca, ya creada y conservada en cinta magnética, de nuevos programas rutinas y funciones que se presentan grabados, con los nombres ISOMERO, FISION y METODO35, en los rótulos 3, 15 y 16 de una cinta que lleva el número de carrete 01221 y tiene etiquetas modelo IBM. El problema se puede resolver en cinco pasos: en el PASO1 el utilitario IEBCOPY transfiere la biblioteca completa, incluido su directorio, de la cinta a un disco auxiliar; en los pasos siguientes (PASO2, PASO3 y PASO4) el utilitario IEBUPDTE graba, de a uno, los programas en la biblioteca que está en el disco auxiliar. En el PASO5 el utilitario IEBCOPY transfiere la biblioteca modificada, del disco auxiliar a la cinta dispuesta para su archivo.

```

//..... JOB ....
//PASO1 EXEC PGM=IEBCOPY
//SYSPRINT DD SYSOUT=A
//ENTRADA DD DSN=SIMBOLIC.TAPE,UNIT=TAPE,VOL=SER=T56600,LABEL=(1,SL),
// DISP=(OLD,KEEP)
//SALIDA DD DSN=SIMBOLIC.TAPE,UNIT=WORKDISK,DISP=(NEW,PASS),
// SPACE=(CYL,(5,2,3)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//SYSIN DD *
C O=SALIDA,I=ENTRADA
/*
//PASO2 EXEC PGM=IEBUPDTE,PARM=MOD
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=ISOMERO,UNIT=TAPE,VOL=SER=T01221,LABEL=(3,SL),
// DISP=(OLD,PASS)
//SYSUT2 DD DSN=SIMBOLIC.TAPE,UNIT=WORKDISK,DISP=(OLD,PASS)
//SYSIN DD *
./ REPRO NEW=PO,MEMBER=ISOMERO,LIST=ALL

```

```

./  ENDUP
/*
//PASO3 EXEC PGM=IEBUPDTE,PARM=MOD
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=FISION,UNIT=TAPE,VOL=REF=*.PASO2.SYSUT1,LABEL=(15,SL)
//SYSUT2 DD DSN=SIMBOLIC.TAPE,UNIT=WORKDISK,DISP=(OLD,PASS)
//SYSIN DD *
./  REPRO NEW=PO,MEMBER=FISION,LIST=ALL
./  ENDUP
/*
//PASO4 EXEC PGM=IEBUPDTE,PARM=MOD
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=METODO35,UNIT=TAPE,VOL=REF=*.PASO2.SYSUT1,
//  LABEL=(16,SL),DISP=(OLD,KEEP)
//SYSUT2 DD DSN=SIMBOLIC.TAPE,UNIT=WORKDISK,DISP=(OLD,PASS)
//SYSIN DD *
./  REPRO NEW=PO,MEMBER=METODO35,LIST=ALL
./  ENDUP
/*
//PASO5 EXEC PGM=IEBCOPY
//SYSPRINT DD SYSOUT=A
//ENTRADA DD DSN=SIMBOLIC.TAPE,UNIT=WORKDISK,DISP=(OLD,DELETE)
//SALIDA DD DSN=SIMBOLIC.TAPE,UNIT=TAPE,VOL=SER=T56600,LABEL=(1,SL)
//SYSIN DD *
  C  O=SALIDA,I=ENTRADA
/*
//

```

NOTA: Se menciona el DCB con el cual se grabará el disco auxiliar

Modelo 491. Listado de la totalidad de la biblioteca ya creada y conservada en cinta magnética. El problema se puede resolver en dos pasos: en el PASO1 el utilitario IEBCOPY transfiere la biblioteca completa, incluido su directorio, de la cinta a un disco auxiliar. En el PASO2 el utilitario IEBPTPCH produce el listado de todos los miembros de la biblioteca obteniéndolos del disco auxiliar.

```

//..... JOB ....
//PASO1 EXEC PGM=IEBCOPY
//SYSPRINT DD SYSOUT=A
//ENTRADA DD DSN=SIMBOLIC.TAPE,UNIT=TAPE,VOL=SER=T56600,LABEL=(1,SL),
//  DISP(OLD,KEEP)
//SALIDA DD DSN=SIMBOLIC.TAPE,UNIT=WORKDISK,DISP=(NEW,PASS),
//  SPACE=(CYL,(5,2,3)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//SYSIN DD *
  C  O=SALIDA,I=ENTRADA
/*
//PASO2 EXEC PGM=IEBPTPCH
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.TAPE,UNIT=WORKDISK,DISP=(OLD,DELETE)
//SYSUT2 DD SYSOUT=A
//SYSIN DD *

```

```

PRINT TYPORG=PO
/*
//

```

NOTA: Se menciona el DCB con el cual se grabará el disco auxiliar.

Modelo 492. Listado de un miembro de la biblioteca ya creada y conservada en cinta magnética. Se desea el listado del miembro SOFTW. El problema se puede resolver en dos pasos: en el PASO1 el utilitario IEBCOPY transfiere la biblioteca, con el miembro requerido solamente, de la cinta a un disco auxiliar; en el PASO2 el utilitario IEBTPCH produce el listado del programa SOFTW obteniéndolo del disco auxiliar.

```

//..... JOB ....
//PASO1 EXEC PGM=IEBCOPY
//SYSPRINT DD SYSOUT=A
//ENTRADA DD DSN=SIMBOLIC.TAPE,UNIT=TAPE,VOL=SER=T56600,LABEL=(1,SL),
//  DISP=(OLD,KEEP)
//SALIDA DD DSN=SIMBOLIC.TAPE,UNIT=WORKDISK,DISP=(NEW,PASS),
//  SPACE=(CYL,(1,1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//SYSIN DD *
  C O=SALIDA,I=ENTRADA
  S M=((SOFTW,,R))
/*
//PASO2 EXEC PGM=IEBTPCH
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.TAPE,UNIT=WORKDISK,DISP=(OLD,DELETE)
//SYSUT2 DD SYSOUT=A
//SYSIN DD *
PRINT TYPORG=PO,MAXNAME=1,MAXFLDS=1
MEMBER NAME=SOFTW
RECORD FIELD=(80)
/*
//

```

NOTA: Se menciona el DCB con el cual se grabará el disco auxiliar.

Modelo 493. Listado de varios miembros de la biblioteca ya creada y conservada en cinta magnética. Se desean los listados de los miembros SIZE, MODELO18, SOURCE y SHIP. El problema se puede resolver en dos pasos: en el PASO1 el utilitario IEBCOPY transfiere la biblioteca, con los miembros requeridos solamente, de la cinta a un disco auxiliar; en el PASO2 el utilitario IEBTPCH produce los listados de los programas obteniéndolos del disco auxiliar.

```

//..... JOB ....
//PASO1 EXEC PGM=IEBCOPY
//SYSPRINT DD SYSOUT=A
//ENTRADA DD DSN=SIMBOLIC.TAPE,UNIT=TAPE,VOL=SER=T56600,LABEL=(1,SL),
// DISP=(OLD,KEEP)
//SALIDA DD DSN=SIMBOLIC.TAPE,UNIT=WORKDISK,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//SYSIN DD *
C O=SALIDA,I=ENTRADA
S M=((SIZE,,R),(MODELO18,,R),(SOURCE,,R),(SHIP,,R))
/*
//PASO2 EXEC PGM=IEBPTPCH
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.TAPE,UNIT=WORKDISK,DISP=(OLD,DELETE)
//SYSUT2 DD SYSOUT=A
//SYSIN DD *
PRINT TYPORG=PO,MAXNAME=4,MAXFLDS=4
MEMBER NAME=SIZE
RECORD FIELD=(80)
MEMBER NAME=MODELO18
RECORD FIELD=(80)
MEMBER NAME=SOURCE
RECORD FIELD=(80)
MEMBER NAME=SHIP
RECORD FIELD=(80)
/*
//

```

NOTA: Se menciona el DCB con el cual se grabará el disco auxiliar.

Modelo 494. Ejecución de un programa en FORTRAN de nombre PROGRESO, grabado en la biblioteca ya creada y conservada en cinta magnética. El problema se puede resolver en dos pasos: en el PASO1 el utilitario IEBCOPY transfiere la biblioteca, con el miembro PROGRESO solamente, de la cinta a un disco auxiliar; en el PASO2 el procedimiento FTG1CLG ejecuta el programa obteniéndolo del disco auxiliar.

```

//..... JOB ....
//PASO1 EXEC PGM=IEBCOPY
//SYSPRINT DD SYSOUT=A
//ENTRADA DD DSN=SIMBOLIC.TAPE,UNIT=TAPE,VOL=SER=T56600,LABEL=(1,SL),
// DISP=(OLD,KEEP)
//SALIDA DD DSN=SIMBOLIC.TAPE,UNIT=WORKDISK,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//SYSIN DD *
C O=SALIDA,I=ENTRADA
S M=((PROGRESO,,R))
/*
//PASO2 EXEC PROC=FTG1CLG
//FORT.SYSIN DD DSN=SIMBOLIC.TAPE(PROGRESO),UNIT=WORKDISK,
// DISP=(OLD,DELETE)
/*
//GO.SYSIN DD *
Aquí tarjetas perforadas de DATOS

```

```
/*
//
```

NOTA: Se menciona el DCB con el cual se grabará el disco auxiliar.

Modelo 495 . Ejecución de un programa en ASSEMBLER, de nombre BARRERA, grabado en la biblioteca ya creada y conservada en cinta magnética. El problema se puede resolver en dos pasos: en el PASO1 el utilitario IEBCOPY transfiere la biblioteca, con el miembro BARRERA solamente, de la cinta a un disco auxiliar; en el PASO2 el procedimiento ASMFCLG ejecuta el programa obteniéndolo del disco auxiliar.

```
//..... JOB ....
//PASO1 EXEC PGM=IEBCOPY
//SYSPRINT DD SYSOUT=A
//ENTRADA DD DSN=SIMBOLIC.TAPE,UNIT=TAPE,VOL=SER=T56600,LABEL=(1,SL),
// DISP=(OLD,KEEP)
//SALIDA DD DSN=SIMBOLIC.TAPE,UNIT=WORKDISK,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//SYSIN DD *
C O=SALIDA,I=ENTRADA
S M=((BARRERA,,R))
/*
//PASO2 EXEC PROC=ASMFCLG
//ASM.SYSIN DD DSN=SIMBOLIC.TAPE(BARRERA),UNIT=WORKDISK,
// DISP=(OLD,DELETE)
/*
//GO.IMPRESO DD SYSOUT=A,DCB=(BLKSIZE=3990,LRECL=133,RECFM=FBM)
//GO.TARJETA DD *,DCB=BLKSIZE=80
Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA. En el programa en ASSEMBLER se ha abierto un archivo de entrada de datos de nombre TARJETA y un archivo de salida de datos de nombre IMPRESO. Se menciona el DCB con el cual se grabará el disco auxiliar.

Modelo 496 . Ejecución de un programa en PL/1, de nombre LIMITE, grabado en la biblioteca ya creada y conservada en cinta magnética. El problema se puede resolver en dos pasos: en el PASO1 el utilitario IEBCOPY transfiere la biblioteca, con el miembro LIMITE solamente, de la cinta a un disco auxiliar; en el PASO2 el procedimiento PLIXCLG ejecuta el programa obteniéndolo del disco auxiliar.

```
//..... JOB ....
```

```
//PASO1 EXEC PGM=IEBCOPY
//SYSPRINT DD SYSOUT=A
//ENTRADA DD DSN=SIMBOLIC.TAPE,UNIT=TAPE,VOL=SER=T56600,LABEL=(1,SL),
// DISP=(OLD,KEEP)
//SALIDA DD DSN=SIMBOLIC.TAPE,UNIT=WORKDISK,DISP=(NEW,PASS),
// SPACE=(CYL,(1,1,1)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//SYSIN DD *
  C O=SALIDA,I=ENTRADA
  S M=((LIMITE,,R))
/*
//PASO2 EXEC PROC=PLIXCLG,PARM.PLI='CS(60)'
//PLI.SYSIN DD DSN=SIMBOLIC.TAPE(LIMITE),UNIT=WORKDISK,
// DISP=(OLD,DELETE)
/*
//GO.SYSIN DD *
  Aquí tarjetas perforadas de DATOS
/*
//
```

NOTA: Se menciona el DCB con el cual se grabará el disco auxiliar.

Modelo 497 . Corrección, inserción y supresión de registros de un programa en FORTRAN, de nombre TOL, grabado en la biblioteca ya creada y conservada en cinta magnética. El problema se puede resolver en tres pasos: en el PASO1 el utilitario IEBCOPY transfiere la biblioteca completa, incluido su directorio, de la cinta a un disco auxiliar; en el PASO2 el utilitario IEBUPDTE realiza la actualización que indican las tarjetas perforadas incluidas entre las sentencias CHANGE y ENDUP, en la biblioteca que está en el disco auxiliar; en el PASO3 el utilitario IEBCOPY transfiere la biblioteca modificada, del disco auxiliar a la cinta dispuesta para su conservación.

```
//..... JOB ....
//PASO1 EXEC PGM=IEBCOPY
//SYSPRINT DD SYSOUT=A
//ENTRADA DD DSN=SIMBOLIC.TAPE,VOL=SER=T56600,LABEL=(1,SL),
// DISP=(OLD,KEEP)
//SALIDA DD DSN=SIMBOLIC.TAPE,UNIT=WORKDISK,DISP=(NEW,PASS),
// SPACE=(CYL,(5,2,3)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//SYSIN DD *
  C O=SALIDA,I=ENTRADA
/*
//PASO2 EXEC PGM=IEBUPDTE,PARM=MOD
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.TAPE,VOL=REF=*.PASO1.SALIDA,DISP=OLD
//SYSUT2 DD DSN=SIMBOLIC.TAPE,VOL=REF=*.PASO1.SALIDA,DISP=OLD
//SYSIN DD *
./ CHANGE NAME=TOL,LIST=ALL
```

```

        DIMENSION A(10),B(10),C(15),D(15)
125  FORMAT(3F10.2,5I3)
./  DELETE SEQ1=77060,SEQ2=77060
    92  CONTINUE
./  DELETE SEQ1=93500,SEQ2=93580
./  ENDUP
/*
//PASO3 EXEC PGM=IEBCOPY
//SYSPRINT DD SYSOUT=A
//ENTRADA DD DSN=SIMBOLIC.TAPE,UNIT=WORKDISK,DISP=(OLD,DELETE)
//SALIDA DD DSN=SIMBOLIC.TAPE,UNIT=TAPE,VOL=SER=T56600,LABEL=(1,SL)
//SYSIN DD *
    C  O=SALIDA,I=ENTRADA
/*
//

```

00076780
00077000
00080200

NOTA: Se menciona el DCB con el cual se grabará el disco auxiliar.

Modelo 498. Renumeración total de un programa, de nombre TWOBODY, grabado en la biblioteca ya creada y conservada en cinta magnética. El problema se puede resolver en tres pasos: en el PASO1 el utilitario IEBCOPY transfiere la biblioteca completa, incluido su directorio, de la cinta a un disco auxiliar; en el PASO2 el utilitario IEBUPDTE realiza la renumeración total a partir del número 100 con incremento 50; en el PASO3 el utilitario IEBCOPY transfiere la biblioteca modificada, del disco auxiliar a la cinta dispuesta para su conservación.

```

//..... JOB ....
//PASO1 EXEC PGM=IEBCOPY
//SYSPRINT DD SYSOUT=A
//ENTRADA DD DSN=SIMBOLIC.TAPE,UNIT=TAPE,VOL=SER=T56600,LABEL=(1,SL),
//  DISP=(OLD,KEEP)
//SALIDA DD DSN=SIMBOLIC.TAPE,UNIT=WORKDISK,DISP=(NEW,PASS),
//  SPACE=(CYL,(5,2,3)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//SYSIN DD *
    C  O=SALIDA,I=ENTRADA
/*
//PASO2 EXEC PGM=IEBUPDTE,PARM=MOD
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DSN=SIMBOLIC.TAPE,VOL=REF=*.PASO1.SALIDA,DISP=OLD
//SYSUT2 DD DSN=SIMBOLIC.TAPE,VOL=REF=*.PASO1.SALIDA,DISP=OLD
//SYSIN DD *
./  CHANGE NAME=TWOBODY,LIST=ALL
./  NUMBER  SEQ1=ALL,NEW1=100,INCR=50
./  ENDUP
/*
//PASO3 EXEC PGM=IEBCOPY
//SYSPRINT DD SYSOUT=A
//ENTRADA DD DSN=SIMBOLIC.TAPE,UNIT=WORKDISK,DISP=(OLD,DELETE)
//SALIDA DD DSN=SIMBOLIC.TAPE,UNIT=TAPE,VOL=SER=T56600,LABEL=(1,SL)

```



```
//SYSIN DD *
C O=SALIDA,I=ENTRADA
/*
//
```

NOTA: Se menciona el DCB con el cual se grabará el disco auxiliar.

Modelo 499 . Supresión de varios miembros de la biblioteca ya creada y conservada en cinta magnética. Se van a borrar los miembros SEEK, AJUSTE, COFCG y MARS. El problema se puede resolver en dos pasos: en el PASO1 el utilitario IEBCOPY transfiere la biblioteca, incluido su directorio y con exclusión de los miembros que figuran en la sentencia EXCLUDE, de la cinta a un disco auxiliar; en el PASO2 el mismo utilitario transfiere la biblioteca modificada, del disco auxiliar a la cinta dispuesta para su archivo.

```
//..... JOB ....
//PASO1 EXEC PGM=IEBCOPY
//SYSPRINT DD SYSOUT=A
//ENTRADA DD DSN=SIMBOLIC.TAPE,UNIT=TAPE,VOL=SER=T56600,LABEL=(1,SL),
// DISP=(OLD,KEEP)
//SALIDA DD DSN=SIMBOLIC.TAPE,UNIT=WORKDISK,DISP=(NEW,PASS),
// SPACE=(CYL,(5,2,3)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//SYSIN DD *
C O=SALIDA,I=ENTRADA
EXCLUDE MEMBER=SEEK,AJUSTE,COFCG,MARS
/*
//PASO2 EXEC PGM=IEBCOPY
//SYSPRINT DD SYSOUT=A
//ENTRADA DD DSN=SIMBOLIC.TAPE,UNIT=WORKDISK,DISP=(OLD,DELETE)
//SALIDA DD DSN=SIMBOLIC.TAPE,UNIT=TAPE,VOL=SER=T56600,LABEL=(1,SL)
//SYSIN DD *
C O=SALIDA,I=ENTRADA
/*
//
```

NOTA: Se menciona el DCB con el cual se grabará el disco auxiliar.

Modelo 500 . Resguardo de la biblioteca en previsión de deterioros accidentales de la cinta magnética número 56600 dispuesta para su archivo. El problema se puede resolver en dos pasos: en el PASO1 el utilitario IEBCOPY transfiere la biblioteca completa, incluido su directorio, de la cinta a un disco auxiliar; en el PASO2 el mismo utilitario, con el nombre SIMBOLIC.COPIA, transfiere la biblioteca con su directorio, del disco auxiliar al rótulo 1 de otra cinta que lleva el nú-

mero de carrete 00711 y tiene etiquetas modelo IBM.

```
//..... JOB ....
//PASO1 EXEC PGM=IEBCOPY
//SYSPRINT DD SYSOUT=A
//ENTRADA DD DSN=SIMBOLIC.TAPE,UNIT=TAPE,VOL=SER=T56600,LABEL=(1,SL),
// DISP=(OLD,KEEP)
//SALIDA DD DSN=SIMBOLIC.TAPE,UNIT=WORKDISK,DISP=(NEW,PASS),
// SPACE=(CYL,(5,2,3)),DCB=(BLKSIZE=3520,LRECL=80,RECFM=FB)
//SYSIN DD *
C O=SALIDA,I=ENTRADA
/*
//PASO2 EXEC PGM=IEBCOPY
//SYSPRINT DD SYSOUT=A
//ENTRADA DD DSN=SIMBOLIC.TAPE,UNIT=WORKDISK,DISP=(OLD,DELETE)
//SALIDA DD DSN=SIMBOLIC.COPIA,UNIT=TAPE,VOL=SER=T00711,LABEL=(1,SL)
//SYSIN DD *
C O=SALIDA,I=ENTRADA
/*
//
```

NOTA: Se menciona el DCB con el cual se grabará el disco auxiliar.

I N D I C E II.TEMARIO DE CADA CAPITULOCAPITULO I.

	Modelos para operar con programas perforados en tarjetas que no utilizan bibliotecas catalogadas.....	1
<u>PUNTO 1.</u>	Ejecución de programas en tarjetas perforadas, codificados totalmente en un mismo lenguaje.....	1
<u>PUNTO 2.</u>	Obtención de programas objeto en tarjetas perforadas, a partir de programas en lenguaje simbólico, que también se presentan en tarjetas perforadas.....	3
<u>PUNTO 3.</u>	Obtención de programas objeto grabados en cintas magnéticas, a partir de programas en lenguaje simbólico, que se presentan en tarjetas perforadas.....	4
<u>PUNTO 4.</u>	Listado de tarjetas perforadas.....	5
<u>PUNTO 5.</u>	Ejecución de programas que invocan rutinas y funciones, codificadas en otro lenguaje, que se presentan en tarjetas perforadas.....	6
<u>PUNTO 6.</u>	Ejecución de programas que invocan rutinas y funciones, codificadas en otro lenguaje, que se presentan grabadas en cintas magnéticas.....	14
<u>PUNTO 7.</u>	Uso de la Referencia hacia Atrás al programa vinculado en un paso anterior de la misma tarea, para ejecutar dos o mas juegos de datos distintos, en los pasos siguientes.....	25
<u>PUNTO 8.</u>	Uso de la Referencia hacia Atrás a un juego de datos ejecutado en un paso anterior de la misma tarea, para ser procesado por dos o mas programas distintos, en los pasos siguientes.....	28

CAPITULO II.

	Modelos para el uso de una biblioteca particionada y catalogada de programas en lenguaje simbólico que reside en una unidad de discos	33
<u>PUNTO 1.</u>	Obtención de listados y lotes de tarjetas perforadas de uno o mas miembros de una biblioteca de nombre SIMBOLIC.CNEA.....	33
<u>PUNTO 2.</u>	Ejecución de programas de una biblioteca de nombre SIMBOLIC.CNEA..	35
<u>PUNTO 3.</u>	Ejecución de programas de una biblioteca de nombre SIMBOLIC.CNEA, que invocan rutinas y funciones residentes en la propia biblioteca como miembros independientes.....	36
<u>PUNTO 4.</u>	Ejecución de programas de una biblioteca de nombre SIMBOLIC.CNEA, que invocan rutinas y funciones que se presentan en tarjetas perforadas.....	39
<u>PUNTO 5.</u>	Ejecución de programas de una biblioteca de nombre SIMBOLIC.CNEA, que invocan rutinas y funciones grabadas en cintas magnéticas.....	45
<u>PUNTO 6.</u>	Uso de la Referencia hacia Atrás a un programa de una biblioteca de nombre SIMBOLIC.CNEA, compilado y vinculado en un paso anterior de la misma tarea, para ejecutar dos o mas juegos de datos distintos, en los pasos siguientes.....	51
<u>PUNTO 7.</u>	Uso de la Referencia hacia Atrás a un juego de datos ejecutado en un paso anterior de la misma tarea, para ser procesado en los pasos siguientes, por dos o mas programas distintos, de una biblioteca de nombre SIMBOLIC.CNEA.....	53

<u>PUNTO 8.</u>	Ejecución de programas en tarjetas perforadas, que invocan rutinas y funciones residentes, como miembros independientes, en una biblioteca de nombre SIMBOLIC.CNEA.....	55
<u>PUNTO 9.</u>	Ejecución de programas grabados en cintas magnéticas, que invocan rutinas y funciones residentes, como miembros independientes, en una biblioteca de nombre SIMBOLIC.CNEA.....	62
<u>PUNTO 10.</u>	Grabación de programas que se presentan en tarjetas perforadas. Agrupación y duplicación de programas ya existentes en la biblioteca.	70
<u>PUNTO 11.</u>	Grabación de programas, residentes en cintas magnéticas, en una biblioteca de nombre SIMBOLIC.CNEA.....	74
<u>PUNTO 12.</u>	Corrección de programas grabados en una biblioteca de nombre SIMBOLIC.CNEA.....	79
<u>PUNTO 13.</u>	Agregado, inserción y reenumeración total de registros de programas grabados en una biblioteca de nombre SIMBOLIC.CNEA.....	82
<u>PUNTO 14.</u>	Transferencia a cinta magnética de uno o mas miembros de una biblioteca de nombre SIMBOLIC.CNEA.....	84

CAPITULO III.

Modelos para la ejecución de programas en FORTRAN que se presentan en distintos soportes y utilizan una biblioteca particionada y catalogada de programas en lenguaje simbólico como archivos de entrada y salida de datos..... 87

<u>PUNTO 1.</u>	Ejecución de programas en FORTRAN que utilizan archivos de entrada de datos residentes en una biblioteca particionada y catalogada de programas en lenguaje simbólico, de nombre SIMBOLIC.CNEA.....	87
<u>PUNTO 2.</u>	Ejecución de programas en FORTRAN que graban archivos de salida de datos en una biblioteca particionada y catalogada de programas en lenguaje simbólico, de nombre SIMBOLIC.CNEA.....	96
<u>PUNTO 3.</u>	Ejecución de programas en FORTRAN que utilizan archivos de entrada de datos residentes en rótulos consecutivos o no consecutivos de una cinta magnética y simultáneamente graban archivos de salida de datos en una biblioteca particionada y catalogada de programas en lenguaje simbólico, de nombre SIMBOLIC.CNEA.....	106
<u>PUNTO 4.</u>	Ejecución de programas en FORTRAN que utilizan archivos de entrada de datos residentes en rótulos consecutivos o no consecutivos de sendas cintas magnéticas y simultáneamente graban archivos de salida de datos en una biblioteca particionada y catalogada de programas en lenguaje simbólico, de nombre SIMBOLIC.CNEA.....	119
<u>PUNTO 5.</u>	Ejecución de programas en FORTRAN que utilizan archivos de entrada de datos residentes en una biblioteca particionada y catalogada de programas en lenguaje simbólico, de nombre SIMBOLIC.CNEA y simultáneamente graban archivos de salida de datos en esa misma biblioteca.....	129

CAPITULO IV.

Modelos para el uso de una biblioteca particionada y catalogada de módulos de carga que reside en una unidad de discos..... 143

<u>PUNTO 1.</u>	Ejecución de módulos de carga que configuran un programa completo,

	grabados en una biblioteca de nombre REALES.CNEA.....	143
<u>PUNTO 2.</u>	Ejecución de programas que son módulos de carga grabados en una biblioteca de nombre REALES.CNEA e invocan rutinas y funciones residentes en la propia biblioteca, como módulos independientes.....	144
<u>PUNTO 3.</u>	Ejecución de programas que son módulos de carga de una biblioteca de nombre REALES.CNEA e invocan rutinas y funciones en tarjetas perforadas.....	146
<u>PUNTO 4.</u>	Ejecución de programas que son módulos de carga de una biblioteca de nombre REALES.CNEA e invocan rutinas y funciones grabadas en cintas magnéticas.....	151
<u>PUNTO 5.</u>	Uso de la Referencia hacia Atrás a un juego de datos ejecutado en un paso anterior de la misma tarea, para ser procesado en los pasos siguientes, por dos o mas módulos distintos de una biblioteca de nombre REALES.CNEA.....	157
<u>PUNTO 6.</u>	Ejecución de programas en tarjetas perforadas, que invocan rutinas y funciones residentes, como módulos de carga, en una biblioteca de nombre REALES.CNEA.....	159
<u>PUNTO 7.</u>	Ejecución de programas residentes en cintas magnéticas, que invocan rutinas y funciones grabadas como módulos de carga en una biblioteca de nombre REALES.CNEA.....	162
<u>PUNTO 8.</u>	Grabación de programas, rutinas o funciones, que se presentan en tarjetas perforadas en lenguajes simbólicos, en una biblioteca de módulos de carga de nombre REALES.CNEA.....	166
<u>PUNTO 9.</u>	Grabación de programas, rutinas o funciones, que están grabadas en cintas magnéticas en lenguajes simbólicos, en una biblioteca de módulos de carga de nombre REALES.CNEA.....	168
<u>PUNTO 10.</u>	Corrección de módulos de carga grabados en una biblioteca de nombre REALES.CNEA, cuyos programas fuente en lenguaje simbólico, se presentan en tarjetas perforadas.....	170

CAPITULO V.

	Modelos para el uso simultáneo de dos bibliotecas particionadas: una de programas en lenguaje simbólico y otra de módulos de carga.	173
<u>PUNTO 1.</u>	Ejecución de programas en lenguaje simbólico, que están grabados en una biblioteca de nombre SIMBOLIC.CNEA e invocan rutinas y funciones residentes en una biblioteca de módulos de carga de nombre REALES.CNEA.....	173
<u>PUNTO 2.</u>	Ejecución de programas en lenguaje simbólico, que están grabados en una biblioteca de nombre SIMBOLIC.CNEA e invocan rutinas y funciones residentes en una biblioteca de módulos de carga de nombre REALES.CNEA, a mas de otras rutinas y funciones que se presentan en tarjetas perforadas.....	175
<u>PUNTO 3.</u>	Ejecución de programas en lenguaje simbólico, que están grabados en una biblioteca de nombre SIMBOLIC.CNEA e invocan rutinas y funciones residentes en una biblioteca de módulos de carga de nombre REALES.CNEA, a mas de otras rutinas y funciones grabadas en cintas magnéticas.....	179
<u>PUNTO 4.</u>	Ejecución de programas que son módulos de carga de una biblioteca de nombre REALES.CNEA e invocan rutinas y funciones en lenguaje simbólico, grabadas como miembros independientes en una biblioteca de nombre SIMBOLIC.CNEA.....	186
<u>PUNTO 5.</u>	Ejecución de programas que son módulos de carga de una biblioteca	

	de nombre REALES.CNEA e invocan rutinas y funciones en lenguaje simbólico, grabadas como miembros independientes en una biblioteca de nombre SIMBOLIC.CNEA, a mas de otras rutinas y funciones que se presentan en tarjetas perforadas.....	189
<u>PUNTO 6.</u>	Ejecución de programas que son módulos de carga de una biblioteca de nombre REALES.CNEA e invocan rutinas y funciones en lenguaje simbólico que residen, como miembros independientes, en una biblioteca de nombre SIMBOLIC.CNEA, a mas de otras rutinas y funciones grabadas en cintas magnéticas.....	199
<u>PUNTO 7.</u>	Ejecución de programas en tarjetas perforadas, que invocan rutinas y funciones en lenguaje simbólico residentes en una biblioteca de nombre SIMBOLIC.CNEA, a mas de otras rutinas y funciones que son módulos de carga de una biblioteca de nombre REALES.CNEA.....	211
<u>PUNTO 8.</u>	Ejecución de programas grabados en cintas magnéticas, que invocan rutinas y funciones en lenguaje simbólico residentes en una biblioteca de nombre SIMBOLIC.CNEA, a mas de otras rutinas y funciones que son módulos de carga de una biblioteca de nombre REALES.CNEA.....	217
<u>PUNTO 9.</u>	Ejecución de módulos de carga procedentes de FORTRAN, de la biblioteca REALES.CNEA, que utilizan archivos de entrada de datos residentes en una biblioteca particionada y catalogada de programas en lenguaje simbólico, de nombre SIMBOLIC.CNEA.....	224
<u>PUNTO 10.</u>	Ejecución de módulos de carga procedentes de FORTRAN de la biblioteca de nombre REALES.CNEA, que graban archivos de salida de datos en una biblioteca particionada y catalogada de programas en lenguaje simbólico, de nombre SIMBOLIC.CNEA.....	226
<u>PUNTO 11.</u>	Ejecución de módulos de carga procedentes de FORTRAN de la biblioteca de nombre REALES.CNEA, que utilizan archivos de entrada de datos residentes en una biblioteca particionada y catalogada de programas en lenguaje simbólico, de nombre SIMBOLIC.CNEA y simultáneamente graban archivos de salida de datos en esa misma biblioteca.....	229
<u>PUNTO 12.</u>	Corrección de módulos de carga grabados en una biblioteca de nombre REALES.CNEA, cuyos programas fuente en lenguaje simbólico, son miembros de la biblioteca SIMBOLIC.CNEA.....	233
<u>PUNTO 13.</u>	Transferencia de programas de una biblioteca de simbólicos de nombre SIMBOLIC.CNEA a una biblioteca de módulos de carga de nombre REALES.CNEA.....	236
<u>PUNTO 14.</u>	Transferencia simultánea de programas que se presentan en tarjetas perforadas o en cintas magnéticas, a dos bibliotecas particionadas: una de programas en lenguaje simbólico de nombre SIMBOLIC.CNEA y otra de módulos de carga de nombre REALES.CNEA.....	237

CAPITULO VI.

	Modelos para la creación y mantenimiento de bibliotecas particionadas y catalogadas.....	241
<u>PUNTO 1.</u>	Creación de nuevas bibliotecas particionadas y catalogadas de programas en lenguaje simbólico.....	241
<u>PUNTO 2.</u>	Creación de nuevas bibliotecas particionadas y catalogadas de módulos de carga.....	246
<u>PUNTO 3.</u>	Reposición en su totalidad o en parte de una biblioteca de programas en lenguaje simbólico, deteriorada accidentalmente, a partir de una cinta magnética de resguardo que fué grabada con el utili-	

	tario IEBCOPY. La biblioteca reside en la unidad de discos 3350, con el número de serie del volumen CNEA02.....	249
<u>PUNTO 4.</u>	Reposición en su totalidad o en parte de una biblioteca de módulos de carga, deteriorada accidentalmente, a partir de una cinta magnética de resguardo que fué grabada con el utilitario IEBCOPY. La biblioteca reside en la unidad de discos 3350, con el número de serie del volumen CNEA02.....	253
<u>PUNTO 5.</u>	Resguardo, compresión y listado del directorio con información sobre el número de bloques del mismo que todavía no han sido utilizados. La biblioteca reside en la unidad de discos 3350, con el número de serie del volumen CNEA02.....	257
<u>PUNTO 6.</u>	Supresión y cambio de nombre de uno o mas miembros de una biblioteca de nombre PARTCNDS.FISICA, que reside en la unidad de discos 3350 con el número de serie del volumen CNEA02.....	259
<u>PUNTO 7.</u>	Borrado y decatalogación de bibliotecas fuera de uso. Cambio del nombre de la biblioteca. Obtención del catálogo de bibliotecas contenidas en el volumen.....	261

CAPITULO VII.

	Modelos para listado, grabación y transferencia de archivos en cintas magnéticas.....	263
<u>PUNTO 1.</u>	Listado de cintas magnéticas.....	263
<u>PUNTO 2.</u>	Obtención de tarjetas perforadas a partir de cintas.....	273
<u>PUNTO 3.</u>	Grabación en cintas magnéticas de programas que se presentan en tarjetas perforadas.....	280
<u>PUNTO 4.</u>	Grabación y listado de etiquetas modelo IBM.....	284
<u>PUNTO 5.</u>	Transferencia de archivos de cinta a otra cinta; de cinta a la misma cinta a través de disco y de disco a disco a través de cinta	286
<u>PUNTO 6.</u>	Transferencia selectiva de programas, rutinas y funciones grabadas secuencialmente en el mismo rótulo de una cinta magnética, mediante la identificación de tiras de caracteres, a otros soportes de información.....	291

CAPITULO VIII.

	Modelos para la ejecución de programas grabados en cintas magnéticas.....	297
<u>PUNTO 1.</u>	Ejecución de programas grabados en cintas magnéticas cuyos datos se presentan en tarjetas perforadas.....	297
<u>PUNTO 2.</u>	Uso de la Referencia hacia Atrás a un programa grabado en cinta magnética y vinculado en un paso anterior de la misma tarea, para ejecutar dos o mas juegos de datos distintos, en los pasos siguientes.....	308
<u>PUNTO 3.</u>	Uso de la Referencia hacia Atrás a un juego de datos ejecutado en un paso anterior de la misma tarea, para ser procesado por dos o mas programas grabados en cintas magnéticas, en los pasos siguientes.....	312

CAPITULO IX.

Modelos para la ejecución de programas que utilizan cintas magnéticas como archivos de entrada de datos.....	325

<u>PUNTO 1.</u> Ejecución de programas en FORTRAN que se presentan en tarjetas perforadas y utilizan cintas magnéticas como archivos de entrada de datos.....	325
<u>PUNTO 2.</u> Ejecución de programas en FORTRAN, residentes en una biblioteca particionada y catalogada de programas en lenguaje simbólico, de nombre SIMBOLIC.CNEA, que utilizan cintas magnéticas como archivos de entrada de datos.....	329
<u>PUNTO 3.</u> Ejecución de módulos procedentes de FORTRAN, grabados en una biblioteca particionada y catalogada de módulos de carga, de nombre REALES.CNEA, que utilizan cintas magnéticas como archivos de entrada de datos.....	331
<u>PUNTO 4.</u> Ejecución de programas en FORTRAN grabados en cintas magnéticas, que utilizan cintas como archivos de entrada de datos.....	333

CAPITULO X.

Modelos para la ejecución de programas que utilizan cintas magnéticas como archivos de salida de datos.....	339

<u>PUNTO 1.</u> Ejecución de programas en FORTRAN que se presentan en tarjetas perforadas y utilizan cintas magnéticas como archivos de salida de datos.....	339
<u>PUNTO 2.</u> Ejecución de programas en FORTRAN que se presentan en tarjetas perforadas, cuyos datos de entrada residen en una biblioteca particionada y catalogada de programas en lenguaje simbólico, de nombre SIMBOLIC.CNEA y cuyos resultados se graban en cintas magnéticas.....	343
<u>PUNTO 3.</u> Ejecución de programas en FORTRAN, residentes en una biblioteca particionada y catalogada de programas en lenguaje simbólico, de nombre SIMBOLIC.CNEA, que utilizan cintas magnéticas como archivos de salida de datos.....	351
<u>PUNTO 4.</u> Ejecución de programas en FORTRAN, residentes en una biblioteca particionada y catalogada de programas en lenguaje simbólico, de nombre SIMBOLIC.CNEA, cuyos datos de entrada residen en la propia biblioteca y cuyos resultados se graban en cintas magnéticas.....	354
<u>PUNTO 5.</u> Ejecución de módulos procedentes de FORTRAN, grabados en una biblioteca particionada y catalogada de módulos de carga, de nombre REALES.CNEA, que utilizan cintas magnéticas como archivos de salida de datos.....	358
<u>PUNTO 6.</u> Ejecución de programas en FORTRAN grabados en cintas magnéticas, que utilizan cintas como archivos de salida de datos.....	361

CAPITULO XI.

Modelos para la ejecución de programas que utilizan cintas magnéticas como archivos de entrada y salida de datos simultáneamente..	367

<u>PUNTO 1.</u> Ejecución de programas en FORTRAN que se presentan en tarjetas perforadas y utilizan cintas magnéticas como archivos de entrada y	

	salida de datos, simultáneamente.....	367
<u>PUNTO 2.</u>	Ejecución de programas en FORTRAN, residentes en una biblioteca particionada y catalogada de programas en lenguaje simbólico, de nombre SIMBOLIC.CNEA, que utilizan cintas magnéticas como archivos de entrada y salida de datos, simultáneamente.....	376
<u>PUNTO 3.</u>	Ejecución de módulos procedentes de FORTRAN, grabados en una biblioteca particionada y catalogada de módulos de carga de nombre REALES.CNEA, que utilizan cintas magnéticas como archivos de entrada y salida de datos, simultáneamente.....	381
<u>PUNTO 4.</u>	Ejecución de programas en FORTRAN grabados en cintas magnéticas, que utilizan cintas como archivos de entrada y salida de datos, simultáneamente.....	386

CAPITULO XII.

	Modelos para la utilización de cintas magnéticas como archivos de módulos de programas o de bibliotecas particionadas cuando no se dispone de espacio propio en una unidad de discos.....	405
<u>PUNTO 1.</u>	Utilización de cintas magnéticas como archivo del módulo de un programa escrito en dos lenguajes simbólicos distintos.....	405
<u>PUNTO 2.</u>	Creación y uso de una biblioteca particionada de programas en lenguaje simbólico, de nombre SIMBOLIC.TAPE y su archivo en el rótulo 1 de una cinta magnética que lleva el número de carrete 56600 y tiene etiquetas modelo IBM.....	411

I N D I C E III.DESCRIPCION DE LOS MODELOS

<u>CAPITULO I.</u> Modelos para operar con programas perforados en tarjetas que no utilizan bibliotecas catalogadas.....	1
<u>PUNTO 1.</u> Ejecución de programas en tarjetas perforadas, codificados totalmente en un mismo lenguaje.....	1
<u>Modelo 001.</u> Ejecución de un programa en lenguaje simbólico FORTRAN.....	1
<u>Modelo 002.</u> Ejecución de un programa en lenguaje simbólico ASSEMBLER.....	1
<u>Modelo 003.</u> Ejecución de un programa en lenguaje simbólico PL/1.....	1
<u>Modelo 004.</u> Ejecución de un programa en lenguaje absoluto de FORTRAN.....	2
<u>Modelo 005.</u> Ejecución de un programa en lenguaje absoluto de ASSEMBLER....	2
<u>Modelo 006.</u> Ejecución de un programa en lenguaje absoluto de PL/1.....	2
<u>PUNTO 2.</u> Obtención de programas objeto en tarjetas perforadas, a partir de programas en lenguaje simbólico, que se presentan en tarjetas perforadas....	3
<u>Modelo 007.</u> Obtención de un programa objeto en lenguaje absoluto de FORTRAN.....	3
<u>Modelo 008.</u> Obtención de un programa objeto en lenguaje absoluto de ASSEMBLER.....	3
<u>Modelo 009.</u> Obtención de un programa objeto en lenguaje absoluto de PL/1..	3
<u>PUNTO 3.</u> Obtención de programas objeto grabados en cintas magnéticas, a partir de programas en lenguaje simbólico, que se presentan en tarjetas perforadas.....	4
<u>Modelo 010.</u> Obtención de un programa objeto en lenguaje absoluto de FORTRAN y su grabación en una cinta magnética.....	4
<u>Modelo 011.</u> Obtención de un programa objeto en lenguaje absoluto de ASSEMBLER y su grabación en una cinta magnética.....	4
<u>Modelo 012.</u> Obtención de un programa objeto en lenguaje absoluto de PL/1 y su grabación en una cinta magnética.....	4
<u>PUNTO 4.</u> Listado de tarjetas perforadas.....	5
<u>Modelo 013.</u> Obtención del listado de programas, rutinas, funciones o datos numéricos perforados en tarjetas.....	5
<u>PUNTO 5.</u> Ejecución de programas que invocan rutinas y funciones, codificadas en otro lenguaje, que se presentan en tarjetas perforadas.....	6
<u>Modelo 014.</u> Ejecución de un programa en lenguaje simbólico FORTRAN, que invoca rutinas y funciones en lenguaje absoluto de FORTRAN o de ASSEMBLER.	6
<u>Modelo 015.</u> Ejecución de un programa en lenguaje simbólico FORTRAN, que invoca rutinas y funciones en lenguaje simbólico ASSEMBLER.....	6
<u>Modelo 016.</u> Ejecución de un programa en lenguaje absoluto de FORTRAN, que invoca rutinas y funciones en lenguaje simbólico FORTRAN.....	6
<u>Modelo 017.</u> Ejecución de un programa en lenguaje absoluto de FORTRAN, que invoca rutinas y funciones en lenguaje simbólico ASSEMBLER.....	7
<u>Modelo 018.</u> Ejecución de un programa en lenguaje absoluto de FORTRAN, que invoca rutinas y funciones en lenguaje absoluto de ASSEMBLER.....	7
<u>Modelo 019.</u> Ejecución de un programa en lenguaje simbólico ASSEMBLER, que invoca rutinas y funciones en lenguaje absoluto de ASSEMBLER.....	8
<u>Modelo 020.</u> Ejecución de un programa en lenguaje simbólico ASSEMBLER, que	

invoca rutinas y funciones en lenguaje simbólico FORTRAN.....	8
<u>Modelo 021.</u> Ejecución de un programa en lenguaje simbólico ASSEMBLER, que invoca rutinas y funciones en lenguaje absoluto de FORTRAN.....	9
<u>Modelo 022.</u> Ejecución de un programa en lenguaje absoluto de ASSEMBLER, que invoca rutinas y funciones en lenguaje simbólico ASSEMBLER.....	10
<u>Modelo 023.</u> Ejecución de un programa en lenguaje absoluto de ASSEMBLER, que invoca rutinas y funciones en lenguaje simbólico FORTRAN.....	11
<u>Modelo 024.</u> Ejecución de un programa en lenguaje absoluto de ASSEMBLER, que invoca rutinas y funciones en lenguaje absoluto de FORTRAN.....	12
<u>Modelo 025.</u> Ejecución de un programa en lenguaje simbólico ASSEMBLER, que invoca rutinas y funciones en lenguaje simbólico FORTRAN y ASSEMBLER.....	12
 <u>PUNTO 6.</u> Ejecución de programas que invocan rutinas y funciones codificadas en otro lenguaje, que se presentan grabadas en cintas magnéticas.	14
 <u>Modelo 026.</u> Ejecución de un programa en lenguaje simbólico FORTRAN, que invoca rutinas y funciones en lenguaje absoluto de FORTRAN o de ASSEMBLER, grabadas en cinta magnética.....	14
<u>Modelo 027.</u> Ejecución de un programa en lenguaje simbólico FORTRAN, que invoca rutinas y funciones en lenguaje simbólico ASSEMBLER, grabadas en cinta magnética.....	14
<u>Modelo 028.</u> Ejecución de un programa en lenguaje absoluto de FORTRAN, que invoca rutinas y funciones en lenguaje simbólico FORTRAN, grabadas en cinta magnética.....	15
<u>Modelo 029.</u> Ejecución de un programa en lenguaje absoluto de FORTRAN, que invoca rutinas y funciones en lenguaje simbólico ASSEMBLER, grabadas en cinta magnética.....	15
<u>Modelo 030.</u> Ejecución de un programa en lenguaje absoluto de FORTRAN, que invoca rutinas y funciones en lenguaje absoluto de FORTRAN o de ASSEMBLER, grabadas en cinta magnética.....	16
<u>Modelo 031.</u> Ejecución de un programa en lenguaje simbólico ASSEMBLER, que invoca rutinas y funciones en lenguaje absoluto de ASSEMBLER, grabadas en cinta magnética.....	16
<u>Modelo 032.</u> Ejecución de un programa en lenguaje simbólico ASSEMBLER, que invoca rutinas y funciones en lenguaje simbólico FORTRAN, grabadas en cinta magnética.....	16
<u>Modelo 033.</u> Ejecución de un programa en lenguaje simbólico ASSEMBLER, que invoca rutinas y funciones en lenguaje absoluto de FORTRAN, grabadas en cinta magnética.....	17
<u>Modelo 034.</u> Ejecución de un programa en lenguaje absoluto de ASSEMBLER, que invoca rutinas y funciones en lenguaje simbólico ASSEMBLER, grabadas en cinta magnética.....	18
<u>Modelo 035.</u> Ejecución de un programa en lenguaje absoluto de ASSEMBLER, que invoca rutinas y funciones en lenguaje simbólico FORTRAN, grabadas en cinta magnética.....	19
<u>Modelo 036.</u> Ejecución de un programa en lenguaje absoluto de ASSEMBLER, que invoca rutinas y funciones en lenguaje absoluto de ASSEMBLER, grabadas en cinta magnética.....	20
<u>Modelo 037.</u> Ejecución de un programa en lenguaje absoluto de ASSEMBLER, que invoca rutinas y funciones en lenguaje absoluto de FORTRAN, grabadas en cinta magnética.....	21
<u>Modelo 038.</u> Ejecución de un programa en lenguaje simbólico ASSEMBLER, que invoca rutinas y funciones en lenguaje simbólico FORTRAN y ASSEMBLER, grabadas en cinta magnética.....	22
<u>Modelo 039.</u> Ejecución de un programa en lenguaje absoluto de ASSEMBLER, que invoca rutinas y funciones en lenguaje simbólico FORTRAN y ASSEMBLER, grabadas en cinta magnética.....	23

PUNTO 7. Uso de la Referencia hacia Atrás al programa vinculado en un paso anterior de la misma tarea, para ejecutar dos o mas juegos de datos distintos, en los pasos siguientes.....	25
Modelo 040. Ejecución de un programa en lenguaje simbólico FORTRAN, compilado y vinculado en el PASO1 durante el cual resuelve el juego de DATOS1; en el PASO2 se hace referencia al mismo programa para resolver el juego de DATOS2.....	25
Modelo 041. Ejecución de un programa en lenguaje absoluto de FORTRAN, vinculado en el PASO1 durante el cual resuelve el juego de DATOS1; en los PASOS 2 y 3 se hace referencia al mismo programa para resolver los juegos de DATOS2 y DATOS3.....	25
Modelo 042. Ejecución de un programa en lenguaje simbólico ASSEMBLER, compilado y vinculado en el PASO1 durante el cual resuelve el juego de DATOS1; en los PASOS 2 y 3 se hace referencia al mismo programa para resolver los juegos de DATOS2 y DATOS3.....	26
Modelo 043. Ejecución de un programa en lenguaje absoluto de ASSEMBLER, vinculado en el PASO1 durante el cual resuelve el juego de DATOS1; en el PASO2 se hace referencia al mismo programa para resolver el juego de DATOS2.....	26
Modelo 044. Ejecución de un programa en lenguaje simbólico PL/1, compilado y vinculado en el PASO1 durante el cual resuelve el juego de DATOS1; en el PASO2 se hace referencia al mismo programa para resolver el juego de DATOS2.....	27
Modelo 045. Ejecución de un programa en lenguaje absoluto de PL/1, vinculado en el PASO1 durante el cual resuelve el juego de DATOS1; en el PASO2 se hace referencia al mismo programa para resolver el juego de DATOS2.....	27
PUNTO 8. Uso de la Referencia hacia Atrás a un juego de datos ejecutado en un paso anterior de la misma tarea, para ser procesado por dos o mas programas distintos, en los pasos siguientes.....	28
Modelo 046. Ejecución de un programa en lenguaje simbólico FORTRAN, compilado y vinculado en el PASO1, en el cual resuelve el juego de DATOS1 y de otro programa en lenguaje simbólico FORTRAN, compilado y vinculado en el PASO2, en el cual se hace referencia al mismo juego de DATOS1.....	28
Modelo 047. Ejecución de un programa en lenguaje absoluto de FORTRAN, vinculado en el PASO1, en el cual resuelve el juego de DATOS1 y de otro programa en lenguaje absoluto de FORTRAN, vinculado en el PASO2, en el cual se hace referencia al mismo juego de DATOS1.....	28
Modelo 048. Ejecución de un programa en lenguaje simbólico ASSEMBLER, compilado y vinculado en el PASO1 en el cual resuelve el juego de DATOS1 y de otro programa en lenguaje simbólico ASSEMBLER, compilado y vinculado en el PASO2, en el cual se hace referencia al mismo juego de DATOS1.....	29
Modelo 049. Ejecución de un programa en lenguaje absoluto de ASSEMBLER, vinculado en el PASO1, en el cual resuelve el juego de DATOS1 y de otro programa en lenguaje absoluto de ASSEMBLER, vinculado en el PASO2, en el cual se hace referencia al mismo juego de DATOS1.....	29
Modelo 050. Ejecución de un programa en lenguaje simbólico PL/1, compilado y vinculado en el PASO1, en el cual resuelve el juego de DATOS1 y de otro programa en lenguaje simbólico PL/1, compilado y vinculado en el PASO2, en el cual se hace referencia al mismo juego de DATOS1.....	30
Modelo 051. Ejecución de un programa en lenguaje absoluto de PL/1, vinculado en el PASO1, en el cual resuelve el juego de DATOS1 y de otro programa en lenguaje absoluto de PL/1, vinculado en el PASO2, en el cual se hace referencia al mismo juego de DATOS1.....	30

<u>CAPITULO II. Modelos para el uso de una biblioteca particionada y catalogada de programas en lenguaje simbólico que reside en una unidad de discos.....</u>	33
<u>PUNTO 1. Obtención de listados y lotes de tarjetas perforadas de uno o mas miembros de una biblioteca de nombre SIMBOLIC.CNEA.....</u>	33
<u>Modelo 052. Listado de la totalidad de la biblioteca.....</u>	33
<u>Modelo 053. Listado de un miembro de la biblioteca y obtención del lote de tarjetas perforadas de otro miembro.....</u>	33
<u>Modelo 054. Listado de cinco miembros de la biblioteca y obtención de los lotes de tarjetas perforadas de otros tres miembros.....</u>	33
<u>PUNTO 2. Ejecución de programas de una biblioteca de nombre SIMBOLIC.CNEA..</u>	35
<u>Modelo 055. Ejecución de un programa en FORTRAN.....</u>	35
<u>Modelo 056. Ejecución de un programa en ASSEMBLER.....</u>	35
<u>Modelo 057. Ejecución de un programa en PL/1.....</u>	35
<u>PUNTO 3. Ejecución de programas de una biblioteca de nombre SIMBOLIC.CNEA, que invocan rutinas y funciones residentes en la propia biblioteca, como miembros independientes.....</u>	36
<u>Modelo 058. Ejecución de un programa en FORTRAN que invoca rutinas y funciones en FORTRAN residentes en la propia biblioteca.....</u>	36
<u>Modelo 059. Ejecución de un programa en FORTRAN que invoca rutinas y funciones en ASSEMBLER residentes en la propia biblioteca.....</u>	36
<u>Modelo 060. Ejecución de un programa en ASSEMBLER que invoca rutinas y funciones en ASSEMBLER residentes en la propia biblioteca.....</u>	37
<u>Modelo 061. Ejecución de un programa en ASSEMBLER que invoca rutinas y funciones en FORTRAN residentes en la propia biblioteca.....</u>	37
<u>PUNTO 4. Ejecución de programas de una biblioteca de nombre SIMBOLIC.CNEA, que invocan rutinas y funciones que se presentan en tarjetas perforadas.....</u>	39
<u>Modelo 062. Ejecución de un programa en FORTRAN que invoca rutinas y funciones que se presentan en tarjetas perforadas en lenguaje simbólico FORTRAN.....</u>	39
<u>Modelo 063. Ejecución de un programa en FORTRAN que invoca rutinas y funciones que se presentan en tarjetas perforadas en lenguaje absoluto de FORTRAN o de ASSEMBLER.....</u>	39
<u>Modelo 064. Ejecución de un programa en FORTRAN que invoca rutinas y funciones que se presentan en tarjetas perforadas en lenguaje simbólico ASSEMBLER.....</u>	39
<u>Modelo 065. Ejecución de un programa en ASSEMBLER que invoca rutinas y funciones que se presentan en tarjetas perforadas en lenguaje simbólico ASSEMBLER.....</u>	40
<u>Modelo 066. Ejecución de un programa en ASSEMBLER que invoca rutinas y funciones que se presentan en tarjetas perforadas en lenguaje absoluto de ASSEMBLER.....</u>	40
<u>Modelo 067. Ejecución de un programa en ASSEMBLER que invoca rutinas y funciones que se presentan en tarjetas perforadas en lenguaje simbólico FORTRAN.....</u>	41
<u>Modelo 068. Ejecución de un programa en ASSEMBLER que invoca rutinas y funciones que se presentan en tarjetas perforadas en lenguaje absoluto de FORTRAN.....</u>	42
<u>Modelo 069. Ejecución de un programa en ASSEMBLER que invoca rutinas y funciones que se presentan en tarjetas perforadas en lenguaje simbólico FORTRAN y ASSEMBLER.....</u>	43

<u>PUNTO 5.</u> Ejecución de programas de una biblioteca de nombre SIMBOLIC.CNEA, que invocan rutinas y funciones grabadas en cintas magnéticas.....	45
<u>Modelo 070.</u> Ejecución de un programa en FORTRAN que invoca rutinas y funciones en lenguaje simbólico FORTRAN grabadas en cinta magnética.....	45
<u>Modelo 071.</u> Ejecución de un programa en FORTRAN que invoca rutinas y funciones en lenguaje absoluto de FORTRAN o de ASSEMBLER grabadas en cinta magnética.....	45
<u>Modelo 072.</u> Ejecución de un programa en FORTRAN que invoca rutinas y funciones en lenguaje simbólico ASSEMBLER grabadas en cinta magnética.....	45
<u>Modelo 073.</u> Ejecución de un programa en ASSEMBLER que invoca rutinas y funciones en lenguaje simbólico ASSEMBLER grabadas en cinta magnética.....	46
<u>Modelo 074.</u> Ejecución de un programa en ASSEMBLER que invoca rutinas y funciones en lenguaje absoluto de ASSEMBLER grabadas en cinta magnética...	47
<u>Modelo 075.</u> Ejecución de un programa en ASSEMBLER que invoca rutinas y funciones en lenguaje simbólico FORTRAN grabadas en cinta magnética.....	47
<u>Modelo 076.</u> Ejecución de un programa en ASSEMBLER que invoca rutinas y funciones en lenguaje absoluto de FORTRAN grabadas en cinta magnética.....	48
<u>Modelo 077.</u> Ejecución de un programa en ASSEMBLER que invoca rutinas y funciones en lenguaje simbólico FORTRAN y ASSEMBLER grabadas en cinta magnética.....	49
 <u>PUNTO 6.</u> Uso de la Referencia hacia Atrás a un programa de una biblioteca de nombre SIMBOLIC.CNEA, compilado y vinculado en un paso anterior de la misma tarea, para ejecutar dos o mas juegos de datos distintos, en los pasos siguientes.....	51
<u>Modelo 078.</u> Ejecución de un programa en FORTRAN compilado y vinculado en el PASO1 durante el cual resuelve el juego de DATOS1; en el PASO2 se hace referencia al mismo programa para resolver el juego de DATOS2.....	51
<u>Modelo 079.</u> Ejecución de un programa en ASSEMBLER compilado y vinculado en el PASO1 durante el cual resuelve el juego de DATOS1; en los PASOS 2 y 3 se hace referencia al mismo programa para resolver los juegos de DATOS2 y DATOS3.....	51
<u>Modelo 080.</u> Ejecución de un programa en PL/1 compilado y vinculado en el PASO1 durante el cual resuelve el juego de DATOS1; en el PASO2 se hace referencia al mismo programa para resolver el juego de DATOS2.....	52
 <u>PUNTO 7.</u> Uso de la Referencia hacia Atrás a un juego de datos ejecutado en un paso anterior de la misma tarea, para ser procesado en los pasos siguientes, por dos o mas programas distintos, de una biblioteca de nombre SIMBOLIC.CNEA.....	53
<u>Modelo 081.</u> Ejecución de un programa en FORTRAN compilado y vinculado en el PASO1 en el cual resuelve el juego de DATOS1 y de otro programa en FORTRAN de la biblioteca, compilado y vinculado en el PASO2, en el cual se hace referencia al mismo juego de DATOS1.....	53
<u>Modelo 082.</u> Ejecución de un programa en ASSEMBLER compilado y vinculado en el PASO1 en el cual resuelve el juego de DATOS1 y de otro programa en ASSEMBLER de la biblioteca, compilado y vinculado en el PASO2, en el cual se hace referencia al mismo juego de DATOS1.....	53
<u>Modelo 083.</u> Ejecución de un programa en PL/1 compilado y vinculado en el PASO1 en el cual resuelve el juego de DATOS1 y de otro programa en PL/1 de la biblioteca, compilado y vinculado en el PASO2, en el cual se hace referencia al mismo juego de DATOS1.....	54
 <u>PUNTO 8.</u> Ejecución de programas en tarjetas perforadas, que invocan rutinas y funciones residentes, como miembros independientes, en una biblioteca de nombre SIMBOLIC.CNEA.....	55

<u>Modelo 084.</u> Ejecución de un programa en tarjetas perforadas en lenguaje simbólico FORTRAN, que invoca rutinas y funciones en FORTRAN, residentes en la biblioteca como miembros independientes.....	55
<u>Modelo 085.</u> Ejecución de un programa en tarjetas perforadas en lenguaje absoluto de FORTRAN, que invoca rutinas y funciones en FORTRAN, residentes en la biblioteca como miembros independientes.....	55
<u>Modelo 086.</u> Ejecución de un programa en tarjetas perforadas en lenguaje simbólico FORTRAN, que invoca rutinas y funciones en ASSEMBLER, residentes en la biblioteca como un miembro independiente.....	55
<u>Modelo 087.</u> Ejecución de un programa en tarjetas perforadas en lenguaje absoluto de FORTRAN, que invoca rutinas y funciones en ASSEMBLER, residentes en la biblioteca como un miembro independiente.....	56
<u>Modelo 088.</u> Ejecución de un programa en tarjetas perforadas en lenguaje simbólico ASSEMBLER, que invoca rutinas y funciones en ASSEMBLER, residentes en la biblioteca como un miembro independiente.....	56
<u>Modelo 089.</u> Ejecución de un programa en tarjetas perforadas en lenguaje absoluto de ASSEMBLER, que invoca rutinas y funciones en ASSEMBLER, residentes en la biblioteca como un miembro independiente.....	57
<u>Modelo 090.</u> Ejecución de un programa en tarjetas perforadas en lenguaje simbólico ASSEMBLER, que invoca rutinas y funciones en FORTRAN, residentes en la biblioteca como miembros independientes.....	58
<u>Modelo 091.</u> Ejecución de un programa en tarjetas perforadas en lenguaje absoluto de ASSEMBLER, que invoca rutinas y funciones en ASSEMBLER, residentes en la biblioteca como un miembro independiente.....	59
<u>Modelo 092.</u> Ejecución de un programa en tarjetas perforadas en lenguaje absoluto de ASSEMBLER, que invoca rutinas y funciones en FORTRAN, residentes en la biblioteca como miembros independientes.....	60
 <u>PUNTO 9.</u> Ejecución de programas grabados en cintas magnéticas, que invocan rutinas y funciones residentes, como miembros independientes, en una biblioteca de nombre SIMBOLIC.CNEA.....	62
 <u>Modelo 093.</u> Ejecución de un programa en lenguaje simbólico FORTRAN grabado en una cinta magnética, que invoca rutinas y funciones en FORTRAN residentes en la biblioteca, como miembros independientes.....	62
<u>Modelo 094.</u> Ejecución de un programa en lenguaje absoluto de FORTRAN grabado en una cinta magnética, que invoca rutinas y funciones en FORTRAN residentes en la biblioteca, como miembros independientes.....	62
<u>Modelo 095.</u> Ejecución de un programa en lenguaje simbólico FORTRAN grabado en una cinta magnética, que invoca rutinas y funciones en ASSEMBLER residentes en la biblioteca, como un miembro independiente.....	63
<u>Modelo 096.</u> Ejecución de un programa en lenguaje absoluto de FORTRAN grabado en una cinta magnética, que invoca rutinas y funciones en ASSEMBLER residentes en la biblioteca, como un miembro independiente.....	63
<u>Modelo 097.</u> Ejecución de un programa en lenguaje simbólico ASSEMBLER grabado en una cinta magnética, que invoca rutinas y funciones en ASSEMBLER residentes en la biblioteca, como un miembro independiente.....	64
<u>Modelo 098.</u> Ejecución de un programa en lenguaje absoluto de ASSEMBLER grabado en una cinta magnética, que invoca rutinas y funciones en ASSEMBLER residentes en la biblioteca, como un miembro independiente.....	65
<u>Modelo 099.</u> Ejecución de un programa en lenguaje simbólico ASSEMBLER grabado en una cinta magnética, que invoca rutinas y funciones en FORTRAN residentes en la biblioteca, como miembros independientes.....	66
<u>Modelo 100.</u> Ejecución de un programa en lenguaje absoluto de ASSEMBLER grabado en una cinta magnética, que invoca rutinas y funciones en ASSEMBLER residentes en la biblioteca, como un miembro independiente.....	67
<u>Modelo 101.</u> Ejecución de un programa en lenguaje absoluto de ASSEMBLER	

grabado en una cinta magnética, que invoca rutinas y funciones en FORTRAN residentes en la biblioteca, como miembros independientes.....	68
<u>PUNTO 10.</u> Grabación de programas que se presentan en tarjetas perforadas. Agrupación y duplicación de programas ya existentes en la biblioteca.....	70
<u>Modelo 102.</u> Grabación en la biblioteca de un programa en lenguaje simbólico. El programa se presenta en tarjetas que ya llevan, en las columnas 73 y siguientes, perforados los números que se desean adjudicar a los registros.....	70
<u>Modelo 103.</u> Grabación en la biblioteca de un programa en lenguaje simbólico. El programa se presenta en tarjetas que no llevan numeración perforada en las columnas 73 y siguientes.....	70
<u>Modelo 104.</u> Grabación en la biblioteca de un programa en lenguaje simbólico, usando el parámetro SEQFLD.....	70
<u>Modelo 105.</u> Grabación simultánea en la biblioteca, como miembros independientes, de programas, rutinas, funciones y datos que se presentan en tarjetas que no llevan numeración perforada en las columnas 73 y siguientes.....	71
<u>Modelo 106.</u> Grabación bajo un solo nombre de varios miembros independientes, ya existentes en la biblioteca.....	72
<u>Modelo 107.</u> Grabación en la biblioteca de un miembro actualizado o mas conveniente, que reemplaza a otro del mismo nombre.....	72
<u>Modelo 108.</u> Duplicación de un miembro de la biblioteca que se grabará de nuevo con otro nombre.....	73
<u>PUNTO 11.</u> Grabación de programas, residentes en cintas magnéticas, en una biblioteca de nombre SIMBOLIC.CNEA.....	74
<u>Modelo 109.</u> Grabación en la biblioteca de un programa en lenguaje simbólico.....	74
<u>Modelo 110.</u> Grabación en la biblioteca de un miembro actualizado o mas conveniente, que reemplaza a otro del mismo nombre.....	74
<u>Modelo 111.</u> Grabación de un programa en FORTRAN que reside, junto con otros, en el mismo rótulo de una cinta magnética, formando un archivo secuencial.....	75
<u>Modelo 112.</u> Grabación simultánea en la biblioteca de programas, rutinas, funciones y datos que residen aisladamente en rótulos consecutivos o no consecutivos de una cinta magnética.....	76
<u>Modelo 113.</u> Grabación de varios programas en FORTRAN que residen, junto con otros, en el mismo rótulo de una cinta magnética, formando un archivo secuencial.....	77
<u>PUNTO 12.</u> Corrección de programas grabados en una biblioteca de nombre SIMBOLIC.CNEA.....	79
<u>Modelo 114.</u> Corrección, inserción y supresión de registros de un programa en FORTRAN, grabado en la biblioteca.....	79
<u>Modelo 115.</u> Corrección, inserción y supresión de registros de un programa en FORTRAN, residente en la biblioteca, que en su oportunidad fué grabado codificando el parámetro SEQFLD.....	79
<u>Modelo 116.</u> Corrección simultánea de tres miembros grabados en FORTRAN en la biblioteca.....	80
<u>Modelo 117.</u> Corrección y ejecución de programas grabados en la biblioteca, codificando el parámetro COND que impide la ejecución de la versión anterior del programa, en caso de fallar la corrección.....	81
<u>PUNTO 13.</u> Agregado, inserción y renumeración total de registros de programas	

ya grabados en una biblioteca de nombre SIMBOLIC.CNEA.....	82
<u>Modelo 118.</u> Agregado de rutinas y funciones, que se presentan en tarjetas, a continuación del último registro de un programa grabado en la biblioteca	82
<u>Modelo 119.</u> Inserción de un número muy grande de nuevos registros, que se presentan en tarjetas, entre dos registros consecutivos de un programa grabado en la biblioteca.....	82
<u>Modelo 120.</u> Renumeración total de un programa que reside en la biblioteca.	83
 <u>PUNTO 14.</u> Transferencia a cinta magnética de uno o mas miembros de una biblioteca de nombre SIMBOLIC.CNEA.....	84
<u>Modelo 121.</u> Transferencia de un miembro de la biblioteca a cinta magnética	84
<u>Modelo 122.</u> Transferencia de miembros de la biblioteca a rótulos consecutivos de una misma cinta magnética.....	84
<u>Modelo 123.</u> Transferencia de miembros de la biblioteca a un mismo rótulo de una cinta magnética.....	85

<u>CAPITULO III. Modelos para la ejecución de programas en FORTRAN que se presentan en distintos soportes y utilizan una biblioteca particionada y catalogada de programas en lenguaje simbólico como archivos de entrada y salida de datos.....</u>	87
<u>PUNTO 1. Ejecución de programas en FORTRAN que utilizan archivos de entrada de datos residentes en una biblioteca particionada y catalogada de programas en lenguaje simbólico, de nombre SIMBOLIC.CNEA.....</u>	87
<u>Modelo 124. Ejecución de un programa en tarjetas perforadas en lenguaje simbólico FORTRAN, cuyos datos de entrada constituyen un archivo que reside en la biblioteca, como un miembro independiente.....</u>	87
<u>Modelo 125. Ejecución de un programa en tarjetas perforadas en lenguaje absoluto de FORTRAN, cuyos datos de entrada constituyen un archivo que reside en la biblioteca, como un miembro independiente.....</u>	87
<u>Modelo 126. Ejecución de un programa en lenguaje simbólico FORTRAN grabado en una cinta magnética, cuyos datos de entrada constituyen un archivo que reside en la biblioteca, como un miembro independiente.....</u>	88
<u>Modelo 127. Ejecución de un programa en lenguaje absoluto de FORTRAN grabado en una cinta magnética, cuyos datos de entrada constituyen un archivo que reside en la biblioteca, como un miembro independiente.....</u>	89
<u>Modelo 128. Ejecución de un programa grabado en la biblioteca, cuyos datos de entrada constituyen un archivo que reside en la propia biblioteca, como un miembro independiente.....</u>	90
<u>Modelo 129. Ejecución de un programa en tarjetas perforadas en lenguaje simbólico FORTRAN, cuyos datos de entrada son tres archivos distintos que residen en la biblioteca, como miembros independientes.....</u>	90
<u>Modelo 130. Ejecución de un programa en tarjetas perforadas en lenguaje absoluto de FORTRAN, cuyos datos de entrada son tres archivos distintos que residen en la biblioteca, como miembros independientes.....</u>	91
<u>Modelo 131. Ejecución de un programa en lenguaje simbólico FORTRAN grabado en una cinta magnética, cuyos datos de entrada son dos archivos distintos que residen en la biblioteca, como miembros independientes.....</u>	92
<u>Modelo 132. Ejecución de un programa en lenguaje absoluto de FORTRAN grabado en una cinta magnética, cuyos datos de entrada son tres archivos distintos que residen en la biblioteca, como miembros independientes.....</u>	93
<u>Modelo 133. Ejecución de un programa grabado en la biblioteca, cuyos datos de entrada son dos archivos distintos que residen en la propia biblioteca, como miembros independientes.....</u>	94
<u>PUNTO 2. Ejecución de programas en FORTRAN que graban archivos de salida de datos en una biblioteca particionada y catalogada de programas en lenguaje simbólico, de nombre SIMBOLIC.CNEA.....</u>	96
<u>Modelo 134. Ejecución de un programa en tarjetas perforadas en lenguaje simbólico FORTRAN, cuyos resultados constituyen un archivo de salida que se graba en la biblioteca como un miembro independiente.....</u>	96
<u>Modelo 135. Ejecución de un programa en tarjetas perforadas en lenguaje absoluto de FORTRAN, cuyos resultados constituyen un archivo de salida que se graba en la biblioteca como un miembro independiente.....</u>	96
<u>Modelo 136. Ejecución de un programa en lenguaje simbólico FORTRAN grabado en una cinta magnética, cuyos resultados constituyen un archivo de salida que se graba en la biblioteca como un miembro independiente.....</u>	97
<u>Modelo 137. Ejecución de un programa en lenguaje absoluto de FORTRAN grabado en una cinta magnética, cuyos resultados constituyen un archivo de salida que se graba en la biblioteca como un miembro independiente.....</u>	98
<u>Modelo 138. Ejecución de un programa residente en la biblioteca, cuyos resultados constituyen un archivo de salida que se graba en la propia bi-</u>	

biblioteca, como un miembro independiente.....	99
<u>Modelo 139.</u> Ejecución de un programa en tarjetas perforadas en lenguaje simbólico FORTRAN, cuyos resultados son tres archivos de salida distintos, que se graban en la biblioteca como miembros independientes.....	100
<u>Modelo 140.</u> Ejecución de un programa en tarjetas perforadas en lenguaje absoluto de FORTRAN, cuyos resultados son cuatro archivos de salida distintos, que se graban en la biblioteca como miembros independientes.....	101
<u>Modelo 141.</u> Ejecución de un programa en lenguaje simbólico FORTRAN grabado en una cinta magnética, cuyos resultados son tres archivos de salida distintos, que se graban en la biblioteca como miembros independientes.....	102
<u>Modelo 142.</u> Ejecución de un programa en lenguaje absoluto de FORTRAN grabado en una cinta magnética, cuyos resultados son dos archivos de salida distintos, que se graban en la biblioteca como miembros independientes....	103
<u>Modelo 143.</u> Ejecución de un programa residente en la biblioteca, cuyos resultados son cuatro archivos de salida distintos que se graban en la propia biblioteca, como miembros independientes.....	104
 <u>PUNTO 3.</u> Ejecución de programas en FORTRAN que utilizan archivos de entrada de datos residentes en rótulos consecutivos o no consecutivos de una cinta magnética y simultáneamente graban archivos de salida de datos en una biblioteca particionada y catalogada de programas en lenguaje simbólico , de nombre SIMBOLIC.CNEA.....	106
<u>Modelo 144.</u> Ejecución de un programa en tarjetas perforadas en lenguaje simbólico FORTRAN, cuyos datos de entrada constituyen un archivo que reside en una cinta magnética y cuyos resultados se graban en la biblioteca, como un miembro independiente.....	106
<u>Modelo 145.</u> Ejecución de un programa en tarjetas perforadas en lenguaje absoluto de FORTRAN, cuyos datos de entrada constituyen un archivo que reside en una cinta magnética y cuyos resultados se graban en la biblioteca, como un miembro independiente.....	107
<u>Modelo 146.</u> Ejecución de un programa en lenguaje simbólico FORTRAN grabado en una cinta magnética, cuyos datos de entrada constituyen un archivo que reside en otra cinta magnética y cuyos resultados se graban en la biblioteca, como un miembro independiente.....	108
<u>Modelo 147.</u> Ejecución de un programa en lenguaje absoluto de FORTRAN grabado en una cinta magnética, cuyos datos de entrada constituyen un archivo que reside en otra cinta magnética y cuyos resultados se graban en la biblioteca, como un miembro independiente.....	109
<u>Modelo 148.</u> Ejecución de un programa grabado en la biblioteca, cuyos datos de entrada constituyen un archivo que reside en una cinta magnética y cuyos resultados se graban en la propia biblioteca, como un miembro independiente.....	110
<u>Modelo 149.</u> Ejecución de un programa en tarjetas perforadas en lenguaje simbólico FORTRAN, cuyos datos de entrada son tres archivos distintos que residen en una cinta magnética y cuyos resultados son dos archivos de salida distintos que se graban en la biblioteca, como miembros independientes.....	111
<u>Modelo 150.</u> Ejecución de un programa en tarjetas perforadas en lenguaje absoluto de FORTRAN, cuyos datos de entrada son dos archivos distintos que residen en una cinta magnética y cuyos resultados son tres archivos de salida distintos que se graban en la biblioteca, como miembros independientes.....	112
<u>Modelo 151.</u> Ejecución de un programa en lenguaje simbólico FORTRAN grabado en una cinta magnética, cuyos datos de entrada son tres archivos distintos que residen en otra cinta magnética y cuyos resultados son tres archivos de salida distintos, que se graban en la biblioteca como miembros indepen-	

dientes.....	113
<u>Modelo 152.</u> Ejecución de un programa en lenguaje absoluto de FORTRAN grabado en una cinta magnética, cuyos datos de entrada son dos archivos distintos que residen en otra cinta y cuyos resultados son dos archivos de salida, que se graban en la biblioteca como miembros independientes.....	115
<u>Modelo 153.</u> Ejecución de un programa grabado en la biblioteca, cuyos datos de entrada son tres archivos distintos que residen en una cinta magnética y cuyos resultados son dos archivos de salida distintos que se graban en la propia biblioteca, como miembros independientes.....	116
<u>PUNTO 4.</u> Ejecución de programas en FORTRAN que utilizan archivos de entrada de datos residentes en rótulos consecutivos o no consecutivos de sendas cintas magnéticas y simultáneamente graban archivos de salida de datos en una biblioteca particionada y catalogada de programas en lenguaje simbólico, de nombre SIMBOLIC.CNEA.....	119
<u>Modelo 154.</u> Ejecución de un programa en tarjetas perforadas en lenguaje simbólico FORTRAN, cuyos datos de entrada residen en sendas cintas magnéticas en cada una de las cuales lee un archivo y cuyos resultados se graban en la biblioteca, como un miembro independiente.....	119
<u>Modelo 155.</u> Ejecución de un programa en tarjetas perforadas en lenguaje absoluto de FORTRAN, cuyos datos de entrada residen en sendas cintas magnéticas en cada una de las cuales lee un archivo y cuyos resultados se graban en la biblioteca, como un miembro independiente.....	120
<u>Modelo 156.</u> Ejecución de un programa grabado en la biblioteca, cuyos datos de entrada residen en sendas cintas magnéticas en cada una de las cuales lee un archivo y cuyos resultados se graban en la propia biblioteca, como un miembro independiente.....	121
<u>Modelo 157.</u> Ejecución de un programa en tarjetas perforadas en lenguaje simbólico FORTRAN, cuyos datos de entrada residen en sendas cintas magnéticas en cada una de las cuales lee varios archivos y cuyos resultados son varios archivos de salida distintos que se graban en la biblioteca, como miembros independientes.....	122
<u>Modelo 158.</u> Ejecución de un programa en tarjetas perforadas en lenguaje absoluto de FORTRAN, cuyos datos de entrada residen en sendas cintas magnéticas en cada una de las cuales lee varios archivos y cuyos resultados son varios archivos de salida distintos que se graban en la biblioteca, como miembros independientes.....	124
<u>Modelo 159.</u> Ejecución de un programa grabado en la biblioteca, cuyos datos de entrada residen en sendas cintas magnéticas en cada una de las cuales lee varios archivos y cuyos resultados son varios archivos de salida distintos que se graban en la propia biblioteca, como miembros independientes	126
<u>PUNTO 5.</u> Ejecución de programas en FORTRAN que utilizan archivos de entrada de datos residentes en una biblioteca particionada y catalogada de programas en lenguaje simbólico, de nombre SIMBOLIC.CNEA y simultáneamente graban archivos de salida de datos en esa misma biblioteca.....	129
<u>Modelo 160.</u> Ejecución de un programa en tarjetas perforadas en lenguaje simbólico FORTRAN cuyos datos de entrada constituyen un archivo que reside en la biblioteca, como un miembro independiente y cuyos resultados se graban en esa misma biblioteca, también como un miembro independiente.....	129
<u>Modelo 161.</u> Ejecución de un programa en tarjetas perforadas en lenguaje absoluto de FORTRAN, cuyos datos de entrada constituyen 1 archivo que reside en la biblioteca, como un miembro independiente y cuyos resultados se graban en esa misma biblioteca, también como un miembro independiente.....	130
<u>Modelo 162.</u> Ejecución de un programa en lenguaje simbólico FORTRAN grabado en una cinta magnética, cuyos datos de entrada constituyen un archivo que	

reside en la biblioteca, como un miembro independiente y cuyos resultados se graban en esa misma biblioteca, también como un miembro independiente...	131
<u>Modelo 163.</u> Ejecución de un programa en lenguaje absoluto de FORTRAN grabado en una cinta magnética, cuyos datos de entrada constituyen un archivo que reside en la biblioteca, como un miembro independiente y cuyos resultados se graban en esa misma biblioteca, también como un miembro independiente.....	132
<u>Modelo 164.</u> Ejecución de un programa grabado en la biblioteca, cuyos datos de entrada constituyen un archivo que reside en la propia biblioteca, como un miembro independiente y cuyos resultados se graban en esa misma biblioteca, también como un miembro independiente.....	133
<u>Modelo 165.</u> Ejecución de un programa en tarjetas perforadas en lenguaje simbólico FORTRAN, cuyos datos de entrada son tres archivos distintos que residen en la biblioteca, como miembros independientes y cuyos resultados son dos archivos de salida distintos que se graban en esa misma biblioteca, también como miembros independientes.....	134
<u>Modelo 166.</u> Ejecución de un programa en tarjetas perforadas en lenguaje absoluto de FORTRAN, cuyos datos de entrada son dos archivos distintos que residen en la biblioteca, como miembros independientes y cuyos resultados son tres archivos de salida distintos que se graban en esa misma biblioteca, también como miembros independientes.....	136
<u>Modelo 167.</u> Ejecución de un programa en lenguaje simbólico FORTRAN grabado en una cinta magnética, cuyos datos de entrada son tres archivos distintos que residen en la biblioteca, como miembros independientes y cuyos resultados son tres archivos de salida distintos que se graban en esa misma biblioteca, también como miembros independientes.....	137
<u>Modelo 168.</u> Ejecución de un programa en lenguaje absoluto de FORTRAN grabado en una cinta magnética, cuyos datos de entrada son dos archivos distintos que residen en la biblioteca, como miembros independientes y cuyos resultados son dos archivos de salida distintos que se graban en esa misma biblioteca, también como miembros independientes.....	139
<u>Modelo 169.</u> Ejecución de un programa grabado en la biblioteca, cuyos datos de entrada son dos archivos distintos que residen en la biblioteca como miembros independientes y cuyos resultados son tres archivos de salida distintos que se graban en la propia biblioteca, también como miembros independientes.....	141

<u>CAPITULO IV.</u> Modelos para el uso de una biblioteca particionada y catalogada de módulos de carga que reside en una unidad de discos.....	143
<u>PUNTO 1.</u> Ejecución de módulos de carga que configuran un programa completo, grabados en una biblioteca de nombre REALES.CNEA.....	143
<u>Modelo 170.</u> Ejecución de un módulo de carga procedente de FORTRAN.....	143
<u>Modelo 171.</u> Ejecución de un módulo de carga procedente de ASSEMBLER.....	143
<u>Modelo 172.</u> Ejecución de un módulo de carga procedente de PL/1.....	143
<u>PUNTO 2.</u> Ejecución de programas que son módulos de carga grabados en una biblioteca de nombre REALES.CNEA e invocan rutinas y funciones residentes en la propia biblioteca, como módulos independientes.....	144
<u>Modelo 173.</u> Ejecución de un programa que es un módulo procedente de FORTRAN e invoca rutinas y funciones que son módulos procedentes de FORTRAN o de ASSEMBLER grabados en la propia biblioteca.....	144
<u>Modelo 174.</u> Ejecución de un programa que es un módulo procedente de ASSEMBLER e invoca rutinas y funciones que son módulos procedentes de ASSEMBLER grabados en la propia biblioteca.....	144
<u>Modelo 175.</u> Ejecución de un programa que es un módulo procedente de ASSEMBLER e invoca rutinas y funciones que son módulos procedentes de FORTRAN grabados en la propia biblioteca.....	145
<u>Modelo 176.</u> Ejecución de un programa que es un módulo procedente de PL/1 e invoca rutinas y funciones que son módulos procedentes de PL/1 grabados en la propia biblioteca.....	145
<u>PUNTO 3.</u> Ejecución de programas que son módulos de carga de una biblioteca de nombre REALES.CNEA e invocan rutinas y funciones en tarjetas perforadas..	146
<u>Modelo 177.</u> Ejecución de un programa que es un módulo procedente de FORTRAN e invoca rutinas y funciones que se presentan en tarjetas perforadas en lenguaje simbólico FORTRAN.....	146
<u>Modelo 178.</u> Ejecución de un programa que es un módulo procedente de FORTRAN e invoca rutinas y funciones que se presentan en tarjetas perforadas en lenguaje simbólico ASSEMBLER.....	146
<u>Modelo 179.</u> Ejecución de un programa que es un módulo procedente de FORTRAN e invoca rutinas y funciones que se presentan en tarjetas perforadas en lenguaje absoluto de FORTRAN o de ASSEMBLER.....	147
<u>Modelo 180.</u> Ejecución de un programa que es un módulo procedente de ASSEMBLER e invoca rutinas y funciones que se presentan en tarjetas perforadas en lenguaje simbólico ASSEMBLER.....	147
<u>Modelo 181.</u> Ejecución de un programa que es un módulo procedente de ASSEMBLER e invoca rutinas y funciones que se presentan en tarjetas perforadas en lenguaje simbólico FORTRAN.....	148
<u>Modelo 182.</u> Ejecución de un programa que es un módulo procedente de ASSEMBLER e invoca rutinas y funciones que se presentan en tarjetas perforadas en absoluto de ASSEMBLER.....	148
<u>Modelo 183.</u> Ejecución de un programa que es un módulo procedente de ASSEMBLER e invoca rutinas y funciones que se presentan en tarjetas perforadas en absoluto de FORTRAN.....	149
<u>Modelo 184.</u> Ejecución de un programa que es un módulo procedente de PL/1 e invoca una rutina o una función que se presenta en tarjetas perforadas en lenguaje simbólico PL/1.....	149
<u>PUNTO 4.</u> Ejecución de programas que son módulos de carga de una biblioteca de nombre REALES.CNEA e invocan rutinas y funciones grabadas en cintas magnéticas.....	151

<u>Modelo 185.</u> Ejecución de un programa que es un módulo procedente de FORTRAN e invoca rutinas y funciones en lenguaje simbólico FORTRAN grabadas en cinta magnética.....	151
<u>Modelo 186.</u> Ejecución de un programa que es un módulo procedente de FORTRAN e invoca rutinas y funciones en lenguaje simbólico ASSEMBLER grabadas en cinta magnética.....	151
<u>Modelo 187.</u> Ejecución de un programa que es un módulo procedente de FORTRAN e invoca rutinas y funciones en lenguaje absoluto de FORTRAN o de ASSEMBLER grabadas en cinta magnética.....	152
<u>Modelo 188.</u> Ejecución de un programa que es un módulo procedente de ASSEMBLER e invoca rutinas y funciones en lenguaje simbólico ASSEMBLER grabadas en cinta magnética.....	152
<u>Modelo 189.</u> Ejecución de un programa que es un módulo procedente de ASSEMBLER e invoca rutinas y funciones en lenguaje simbólico FORTRAN grabadas en cinta magnética.....	153
<u>Modelo 190.</u> Ejecución de un programa que es un módulo procedente de ASSEMBLER e invoca rutinas y funciones en lenguaje absoluto de ASSEMBLER grabadas en cinta magnética.....	154
<u>Modelo 191.</u> Ejecución de un programa que es un módulo procedente de ASSEMBLER e invoca rutinas y funciones en lenguaje absoluto de FORTRAN grabadas en cinta magnética.....	155
<u>Modelo 192.</u> Ejecución de un programa que es un módulo procedente de PL/1 e invoca una rutina en lenguaje simbólico PL/1 grabada en cinta magnética...	156
 <u>PUNTO 5.</u> Uso de la Referencia hacia Atrás a un juego de datos ejecutado en un paso anterior de la misma tarea, para ser procesado en los pasos siguientes, por dos o mas módulos distintos de una biblioteca de nombre REALES. CNEA.....	157
<u>Modelo 193.</u> Ejecución de un módulo procedente de FORTRAN que en el PASO1 resuelve el juego de DATOS1 y de otro módulo procedente de FORTRAN que en el PASO2 hace referencia al mismo juego de DATOS1.....	157
<u>Modelo 194.</u> Ejecución de un módulo procedente de ASSEMBLER que en el PASO1 resuelve el juego de DATOS1 y de otro módulo procedente de ASSEMBLER que en el PASO2 hace referencia al mismo juego de DATOS1.....	157
<u>Modelo 195.</u> Ejecución de un módulo procedente de PL/1 que en el PASO1 resuelve el juego de DATOS1 y de otro módulo procedente de PL/1 que en el PASO2 hace referencia al mismo juego de DATOS1.....	158
 <u>PUNTO 6.</u> Ejecución de programas en tarjetas perforadas, que invocan rutinas y funciones residentes, como módulos de carga, en una biblioteca de nombre REALES.CNEA.....	159
<u>Modelo 196.</u> Ejecución de un programa en tarjetas perforadas en lenguaje simbólico FORTRAN que invoca rutinas y funciones procedentes de FORTRAN o de ASSEMBLER grabadas en la biblioteca como módulos independientes.....	159
<u>Modelo 197.</u> Ejecución de un programa en tarjetas perforadas en lenguaje absoluto de FORTRAN que invoca rutinas y funciones procedentes de FORTRAN o de ASSEMBLER grabadas en la biblioteca como módulos independientes.....	159
<u>Modelo 198.</u> Ejecución de un programa en tarjetas perforadas en lenguaje simbólico ASSEMBLER que invoca rutinas y funciones procedentes de FORTRAN o de ASSEMBLER grabadas en la biblioteca como módulos independientes.....	159
<u>Modelo 199.</u> Ejecución de un programa en tarjetas perforadas en lenguaje absoluto de ASSEMBLER que invoca rutinas y funciones procedentes de ASSEMBLER o de FORTRAN grabadas en la biblioteca como módulos independientes.....	160
<u>Modelo 200.</u> Ejecución de un programa en tarjetas perforadas en lenguaje	

simbólico PL/1 que invoca rutinas y funciones procedentes de PL/1 grabadas en la biblioteca como módulos independientes.....	161
PUNTO 7. Ejecución de programas residentes en cintas magnéticas, que invocan rutinas y funciones grabadas como módulos de carga en una biblioteca de nombre REALES.CNEA.....	162
Modelo 201. Ejecución de un programa en lenguaje simbólico FORTRAN residente en una cinta magnética, que invoca rutinas y funciones procedentes de FORTRAN o de ASSEMBLER, grabadas en la biblioteca como módulos independientes.....	162
Modelo 202. Ejecución de un programa en lenguaje absoluto de FORTRAN residente en una cinta magnética, que invoca rutinas y funciones procedentes de FORTRAN o de ASSEMBLER grabadas en la biblioteca como módulos independientes.....	162
Modelo 203. Ejecución de un programa en lenguaje simbólico ASSEMBLER residente en una cinta magnética, que invoca rutinas y funciones procedentes de FORTRAN o de ASSEMBLER, grabadas en la biblioteca como módulos independientes.....	163
Modelo 204. Ejecución de un programa en lenguaje absoluto de ASSEMBLER residente en una cinta magnética, que invoca rutinas y funciones procedentes de ASSEMBLER, grabadas en la biblioteca como módulos independientes.....	163
Modelo 205. Ejecución de un programa en lenguaje absoluto de ASSEMBLER residente en una cinta magnética, que invoca rutinas y funciones procedentes de FORTRAN, grabadas en la biblioteca como módulos independientes.....	164
Modelo 206. Ejecución de un programa en lenguaje simbólico PL/1 residente en una cinta magnética, que invoca rutinas y funciones procedentes de PL/1 grabadas en la biblioteca como módulos independientes.....	165
PUNTO 8. Grabación de programas, rutinas o funciones, que se presentan en tarjetas perforadas en lenguajes simbólicos, en una biblioteca de módulos de carga de nombre REALES.CNEA.....	166
Modelo 207. Grabación en la biblioteca, como módulos independientes, de un programa completo, una rutina y una función que se presentan en tarjetas perforadas en lenguaje simbólico FORTRAN.....	166
Modelo 208. Grabación en la biblioteca, como módulos independientes, de un programa completo, una rutina y una función que se presentan en tarjetas perforadas en lenguaje simbólico ASSEMBLER.....	166
Modelo 209. Grabación en la biblioteca, como módulos independientes, de un programa completo, una rutina y una función que se presentan en tarjetas perforadas en lenguaje simbólico PL/1.....	167
PUNTO 9. Grabación de programas, rutinas o funciones que están grabadas en cintas magnéticas en lenguajes simbólicos, en una biblioteca de módulos de carga de nombre REALES.CNEA.....	168
Modelo 210. Grabación en la biblioteca, como módulos independientes, de un programa completo, una rutina y una función que están grabadas en una cinta magnética, en lenguaje simbólico FORTRAN.....	168
Modelo 211. Grabación en la biblioteca, como módulos independientes, de un programa completo, una rutina y una función que están grabadas en una cinta magnética, en lenguaje simbólico ASSEMBLER.....	168
Modelo 212. Grabación en la biblioteca, como módulos independientes, de un programa completo, una rutina y una función que están grabadas en una cinta magnética, en lenguaje simbólico PL/1.....	169
PUNTO 10. Corrección de módulos de carga grabados en una biblioteca de nombre REALES.CNEA, cuyos programas fuente en lenguaje simbólico, se presentan	

en tarjetas perforadas.....	170
<u>Modelo 213.</u> Corrección de un módulo de carga, procedente de FORTRAN, grabado en la biblioteca. La corrección se realiza en el lote de tarjetas perforadas en lenguaje simbólico, que dió origen al módulo. El procedimiento FTG1CL carga el nuevo módulo en la biblioteca, sustituyendo al anterior con el mismo nombre.....	170
<u>Modelo 214.</u> Corrección de un módulo de carga, procedente de ASSEMBLER, grabado en la biblioteca. La corrección se realiza en el lote de tarjetas perforadas en lenguaje simbólico, que dió origen al módulo. El procedimiento ASMFCL carga el nuevo módulo en la biblioteca, sustituyendo al anterior con el mismo nombre.....	170
<u>Modelo 215.</u> Corrección de un módulo de carga, procedente de PL/1, grabado en la biblioteca. La corrección se realiza en el lote de tarjetas perforadas en lenguaje simbólico, que dió origen al módulo. El procedimiento PLIXCL carga el nuevo módulo en la biblioteca, sustituyendo al anterior con el mismo nombre.....	170

CAPITULO V. Modelos para el uso simultáneo de dos bibliotecas particionadas:
una de programas en lenguaje simbólico y otra de módulos de carga..... 173

PUNTO 1. Ejecución de programas en lenguaje simbólico, que están grabados
en una biblioteca de nombre SIMBOLIC.CNEA e invocan rutinas y funciones re-
sidentes en una biblioteca de módulos de carga de nombre REALES.CNEA..... 173

Modelo 216. Ejecución de un programa en lenguaje FORTRAN residente en la
biblioteca de simbólicos e invoca rutinas y funciones procedentes de
FORTRAN o de ASSEMBLER, grabadas en la biblioteca de módulos de carga..... 173

Modelo 217. Ejecución de un programa en lenguaje ASSEMBLER residente en la
biblioteca de simbólicos e invoca rutinas y funciones procedentes de
ASSEMBLER o de FORTRAN, grabadas en la biblioteca de módulos de carga..... 173

Modelo 218. Ejecución de un programa en lenguaje PL/1 residente en la bi-
blioteca de simbólicos e invoca rutinas y funciones procedentes de PL/1,
grabadas en la biblioteca de módulos de carga..... 174

PUNTO 2. Ejecución de programas en lenguaje simbólico, que están grabados
en una biblioteca de nombre SIMBOLIC.CNEA e invocan rutinas y funciones re-
sidentes en una biblioteca de módulos de carga de nombre REALES.CNEA, a mas
de otras rutinas y funciones que se presentan en tarjetas perforadas..... 175

Modelo 219. Ejecución de un programa en lenguaje FORTRAN residente en la
biblioteca de simbólicos, e invoca rutinas y funciones procedentes de
FORTRAN o de ASSEMBLER, grabadas en la biblioteca de módulos de carga, a
mas de otras rutinas y funciones que se presentan en tarjetas perforadas
en lenguaje simbólico FORTRAN..... 175

Modelo 220. Ejecución de un programa en lenguaje FORTRAN residente en la
biblioteca de simbólicos, e invoca rutinas y funciones precedentes de
FORTRAN o de ASSEMBLER, grabadas en la biblioteca de módulos de carga, a
mas de otras rutinas y funciones que se presentan en tarjetas perforadas
en lenguaje simbólico ASSEMBLER..... 175

Modelo 221. Ejecución de un programa en lenguaje FORTRAN residente en la
biblioteca de simbólicos, e invoca rutinas y funciones procedentes de
FORTRAN o de ASSEMBLER, grabadas en la biblioteca de módulos de carga, a
mas de otras rutinas y funciones que se presentan en tarjetas perforadas
en lenguaje absoluto de FORTRAN o de ASSEMBLER..... 176

Modelo 222. Ejecución de un programa en lenguaje ASSEMBLER residente en la
biblioteca de simbólicos, e invoca rutinas y funciones procedentes de
FORTRAN o de ASSEMBLER, grabadas en la biblioteca de módulos de carga, a
mas de otras rutinas y funciones que se presentan en tarjetas perforadas
en lenguaje simbólico ASSEMBLER..... 176

Modelo 223. Ejecución de un programa en lenguaje ASSEMBLER residente en la
biblioteca de simbólicos, e invoca rutinas y funciones procedentes de
FORTRAN o de ASSEMBLER, grabadas en la biblioteca de módulos de carga, a
mas de otras rutinas y funciones que se presentan en tarjetas perforadas
en lenguaje simbólico FORTRAN..... 177

Modelo 224. Ejecución de un programa en lenguaje ASSEMBLER residente en la
biblioteca de simbólicos, e invoca rutinas y funciones procedentes de
FORTRAN o de ASSEMBLER, grabadas en la biblioteca de módulos de carga, a
mas de otras rutinas y funciones que se presentan en tarjetas perforadas
en lenguaje absoluto de ASSEMBLER..... 178

PUNTO 3. Ejecución de programas en lenguaje simbólico, que están grabados
en una biblioteca de nombre SIMBOLIC.CNEA e invocan rutinas y funciones re-
sidentes en una biblioteca de módulos de carga de nombre REALES.CNEA. a mas
de otras rutinas y funciones grabadas en cintas magnéticas..... 179

<u>Modelo 225.</u> Ejecución de un programa en lenguaje FORTRAN residente en la biblioteca de simbólicos, que invoca rutinas y funciones procedentes de FORTRAN o de ASSEMBLER, grabadas en la biblioteca de módulos de carga, a mas de otras rutinas y funciones grabadas en lenguaje simbólico FORTRAN en una cinta magnética.....	179
<u>Modelo 226.</u> Ejecución de un programa en lenguaje FORTRAN residente en la biblioteca de simbólicos, que invoca rutinas y funciones procedentes de FORTRAN o de ASSEMBLER, grabadas en la biblioteca de módulos de carga, a mas de otras rutinas y funciones grabadas en lenguaje absoluto de FORTRAN o de ASSEMBLER en una cinta magnética.....	179
<u>Modelo 227.</u> Ejecución de un programa en lenguaje FORTRAN residente en la biblioteca de simbólicos, que invoca rutinas y funciones procedentes de FORTRAN o de ASSEMBLER, grabadas en la biblioteca de módulos de carga, a mas de otras rutinas y funciones grabadas en lenguaje simbólico ASSEMBLER en una cinta magnética.....	180
<u>Modelo 228.</u> Ejecución de un programa en lenguaje ASSEMBLER residente en la biblioteca de simbólicos, que invoca rutinas y funciones procedentes de FORTRAN o de ASSEMBLER, grabadas en la biblioteca de módulos de carga, a mas de otras rutinas y funciones grabadas en lenguaje simbólico ASSEMBLER en una cinta magnética.....	181
<u>Modelo 229.</u> Ejecución de un programa en lenguaje ASSEMBLER residente en la biblioteca de simbólicos, que invoca rutinas y funciones procedentes de FORTRAN o de ASSEMBLER, grabadas en la biblioteca de módulos de carga, a mas de otras rutinas y funciones grabadas en lenguaje absoluto de ASSEMBLER en una cinta magnética.....	182
<u>Modelo 230.</u> Ejecución de un programa en lenguaje ASSEMBLER residente en la biblioteca de simbólicos, que invoca rutinas y funciones procedentes de FORTRAN o de ASSEMBLER, grabadas en la biblioteca de módulos de carga, a mas de otras rutinas y funciones grabadas en lenguaje simbólico FORTRAN en una cinta magnética.....	183
<u>Modelo 231.</u> Ejecución de un programa en lenguaje ASSEMBLER residente en la biblioteca de simbólicos, que invoca rutinas y funciones procedentes de FORTRAN o de ASSEMBLER, grabadas en la biblioteca de módulos de carga, a mas de otras rutinas y funciones grabadas en lenguaje absoluto de FORTRAN en una cinta magnética.....	184
<u>PUNTO 4.</u> Ejecución de programas que son módulos de carga de una biblioteca de nombre REALES.CNEA e invocan rutinas y funciones en lenguaje simbólico, grabadas como miembros independientes en una biblioteca de nombre SIMBOLIC.CNEA.....	186
<u>Modelo 232.</u> Ejecución de un módulo de carga procedente de FORTRAN, que invoca rutinas y funciones en lenguaje FORTRAN residentes en la biblioteca de simbólicos.....	186
<u>Modelo 233.</u> Ejecución de un módulo de carga procedente de FORTRAN, que invoca rutinas y funciones en lenguaje ASSEMBLER residentes en la biblioteca de simbólicos.....	186
<u>Modelo 234.</u> Ejecución de un módulo de carga procedente de ASSEMBLER, que invoca rutinas y funciones en lenguaje ASSEMBLER residentes en la biblioteca de simbólicos.....	187
<u>Modelo 235.</u> Ejecución de un módulo de carga procedente de ASSEMBLER, que invoca rutinas y funciones en lenguaje FORTRAN residentes en la biblioteca de simbólicos.....	188
<u>PUNTO 5.</u> Ejecución de programas que son módulos de carga de una biblioteca de nombre REALES.CNEA e invocan rutinas y funciones en lenguaje simbólico grabadas como miembros independientes en una biblioteca de nombre SIMBOLIC.CNEA, a mas de otras rutinas y funciones que se presentan en tarjetas perfo-	

radas.....	189
<u>Modelo 236.</u> Ejecución de un módulo de carga procedente de FORTRAN, que invoca rutinas y funciones en lenguaje FORTRAN residentes en la biblioteca de simbólicos, a mas de otras rutinas y funciones que se presentan en tarjetas perforadas en lenguaje simbólico FORTRAN.....	189
<u>Modelo 237.</u> Ejecución de un módulo de carga procedente de FORTRAN, que invoca rutinas y funciones en lenguaje FORTRAN residentes en la biblioteca de simbólicos, a mas de otras rutinas y funciones que se presentan en tarjetas perforadas en lenguaje absoluto de FORTRAN o de ASSEMBLER.....	189
<u>Modelo 238.</u> Ejecución de un módulo de carga procedente de FORTRAN, que invoca rutinas y funciones en lenguaje FORTRAN residentes en la biblioteca de simbólicos, a mas de otras rutinas y funciones que se presentan en tarjetas perforadas en lenguaje simbólico ASSEMBLER.....	190
<u>Modelo 239.</u> Ejecución de un módulo de carga procedente de FORTRAN, que invoca rutinas y funciones en lenguaje ASSEMBLER residentes en la biblioteca de simbólicos, a mas de otras rutinas y funciones que se presentan en tarjetas perforadas en lenguaje simbólico FORTRAN.....	191
<u>Modelo 240.</u> Ejecución de un módulo de carga procedente de FORTRAN, que invoca rutinas y funciones en lenguaje ASSEMBLER residentes en la biblioteca de simbólicos, a mas de otras rutinas y funciones que se presentan en tarjetas perforadas en lenguaje absoluto de FORTRAN o de ASSEMBLER.....	192
<u>Modelo 241.</u> Ejecución de un módulo de carga procedente de FORTRAN, que invoca rutinas y funciones en lenguaje ASSEMBLER residentes en la biblioteca de simbólicos, a mas de otras rutinas y funciones que se presentan en tarjetas perforadas en lenguaje simbólico ASSEMBLER.....	192
<u>Modelo 242.</u> Ejecución de un módulo de carga procedente de ASSEMBLER, que invoca rutinas y funciones en lenguaje ASSEMBLER residentes en la biblioteca de simbólicos, a mas de otras rutinas y funciones que se presentan en tarjetas perforadas en lenguaje simbólico ASSEMBLER.....	193
<u>Modelo 243.</u> Ejecución de un módulo de carga procedente de ASSEMBLER, que invoca rutinas y funciones en lenguaje ASSEMBLER residentes en la biblioteca de simbólicos, a mas de otras rutinas y funciones que se presentan en tarjetas perforadas en lenguaje simbólico FORTRAN.....	194
<u>Modelo 244.</u> Ejecución de un módulo de carga procedente de ASSEMBLER, que invoca rutinas y funciones en lenguaje ASSEMBLER residentes en la biblioteca de simbólicos, a mas de otras rutinas y funciones que se presentan en tarjetas perforadas en lenguaje absoluto de FORTRAN o de ASSEMBLER.....	195
<u>Modelo 245.</u> Ejecución de un módulo de carga procedente de ASSEMBLER, que invoca rutinas y funciones en lenguaje FORTRAN residentes en la biblioteca de simbólicos, a mas de otras rutinas y funciones que se presentan en tarjetas perforadas en lenguaje simbólico ASSEMBLER.....	196
<u>Modelo 246.</u> Ejecución de un módulo de carga procedente de ASSEMBLER, que invoca rutinas y funciones en lenguaje FORTRAN residentes en la biblioteca de simbólicos, a mas de otras rutinas y funciones que se presentan en tarjetas perforadas en lenguaje simbólico FORTRAN.....	197
<u>Modelo 247.</u> Ejecución de un módulo de carga procedente de ASSEMBLER, que invoca rutinas y funciones en lenguaje FORTRAN residentes en la biblioteca de simbólicos, a mas de otras rutinas y funciones que se presentan en tarjetas perforadas en lenguaje absoluto de FORTRAN o de ASSEMBLER.....	198
 <u>PUNTO 6.</u> Ejecución de programas que son módulos de carga de una biblioteca de nombre REALES.CNEA e invocan rutinas y funciones en lenguaje simbólico que residen, como miembros independientes, en una biblioteca de nombre SIMBOLIC.CNEA, a mas de otras rutinas y funciones grabadas en cintas magnéticas.....	199

<u>Modelo 248.</u> Ejecución de un módulo de carga procedente de FORTRAN, que invoca rutinas y funciones en lenguaje FORTRAN residentes en la biblioteca de simbólicos, a mas de otras rutinas y funciones grabadas en lenguaje simbólico FORTRAN en una cinta magnética.....	199
<u>Modelo 249.</u> Ejecución de un módulo de carga procedente de FORTRAN, que invoca rutinas y funciones en lenguaje FORTRAN residentes en la biblioteca de simbólicos, a mas de otras rutinas y funciones grabadas en lenguaje absoluto de FORTRAN o de ASSEMBLER en una cinta magnética.....	199
<u>Modelo 250.</u> Ejecución de un módulo de carga procedente de FORTRAN, que invoca rutinas y funciones en lenguaje FORTRAN residentes en la biblioteca de simbólicos, a mas de otras rutinas y funciones grabadas en lenguaje simbólico ASSEMBLER en una cinta magnética.....	200
<u>Modelo 251.</u> Ejecución de un módulo de carga procedente de FORTRAN, que invoca rutinas y funciones en lenguaje ASSEMBLER residentes en la biblioteca de simbólicos, a mas de otras rutinas y funciones grabadas en lenguaje simbólico FORTRAN en una cinta magnética.....	201
<u>Modelo 252.</u> Ejecución de un módulo de carga procedente de FORTRAN, que invoca rutinas y funciones en lenguaje ASSEMBLER residentes en la biblioteca de simbólicos, a mas de otras rutinas y funciones grabadas en lenguaje absoluto de FORTRAN o de ASSEMBLER en una cinta magnética.....	202
<u>Modelo 253.</u> Ejecución de un módulo de carga procedente de FORTRAN, que invoca rutinas y funciones en lenguaje ASSEMBLER residentes en la biblioteca de simbólicos, a mas de otras rutinas y funciones grabadas en lenguaje simbólico ASSEMBLER en una cinta magnética.....	203
<u>Modelo 254.</u> Ejecución de un módulo de carga procedente de ASSEMBLER, que invoca rutinas y funciones en lenguaje ASSEMBLER residentes en la biblioteca de simbólicos, a mas de otras rutinas y funciones grabadas en lenguaje simbólico ASSEMBLER en una cinta magnética.....	204
<u>Modelo 255.</u> Ejecución de un módulo de carga procedente de ASSEMBLER, que invoca rutinas y funciones en lenguaje ASSEMBLER residentes en la biblioteca de simbólicos, a mas de otras rutinas y funciones grabadas en lenguaje simbólico FORTRAN en una cinta magnética.....	205
<u>Modelo 256.</u> Ejecución de un módulo de carga procedente de ASSEMBLER, que invoca rutinas y funciones en lenguaje ASSEMBLER residentes en la biblioteca de simbólicos, a mas de otras rutinas y funciones grabadas en lenguaje absoluto de FORTRAN o de ASSEMBLER en una cinta magnética.....	206
<u>Modelo 257.</u> Ejecución de un módulo de carga procedente de ASSEMBLER, que invoca rutinas y funciones en lenguaje FORTRAN residentes en la biblioteca de simbólicos, a mas de otras rutinas y funciones grabadas en lenguaje simbólico ASSEMBLER en una cinta magnética.....	207
<u>Modelo 258.</u> Ejecución de un módulo de carga procedente de ASSEMBLER, que invoca rutinas y funciones en lenguaje FORTRAN residentes en la biblioteca de simbólicos, a mas de otras rutinas y funciones grabadas en lenguaje simbólico FORTRAN en una cinta magnética.....	208
<u>Modelo 259.</u> Ejecución de un módulo de carga procedente de ASSEMBLER, que invoca rutinas y funciones en lenguaje FORTRAN residentes en la biblioteca de simbólicos, a mas de otras rutinas y funciones grabadas en lenguaje absoluto de FORTRAN o de ASSEMBLER en una cinta magnética.....	209

<u>PUNTO 7.</u> Ejecución de programas en tarjetas perforadas, que invocan rutinas y funciones en lenguaje simbólico residentes en una biblioteca de nombre SIMBOLIC.CNEA, a mas de otras rutinas y funciones que son módulos de carga de una biblioteca de nombre REALES.CNEA.....	211

Modelo 260. Ejecución de un programa en tarjetas perforadas en lenguaje simbólico FORTRAN, que invoca rutinas y funciones en lenguaje FORTRAN residentes en la biblioteca de simbólicos, a mas de otras rutinas y funcio-

nes grabadas en la biblioteca de módulos de carga.....	211
<u>Modelo 261.</u> Ejecución de un programa en tarjetas perforadas en lenguaje absoluto de FORTRAN, que invoca rutinas y funciones en lenguaje FORTRAN residentes en la biblioteca de simbólicos, a mas de otras rutinas y funciones grabadas en la biblioteca de módulos de carga.....	211
<u>Modelo 262.</u> Ejecución de un programa en tarjetas perforadas en lenguaje simbólico FORTRAN, que invoca rutinas y funciones en lenguaje ASSEMBLER residentes en la biblioteca de simbólicos, a mas de otras rutinas y funciones grabadas en la biblioteca de módulos de carga.....	212
<u>Modelo 263.</u> Ejecución de un programa en tarjetas perforadas en lenguaje absoluto de FORTRAN, que invoca rutinas y funciones en lenguaje ASSEMBLER residentes en la biblioteca de simbólicos, a mas de otras rutinas y funciones grabadas en la biblioteca de módulos de carga.....	212
<u>Modelo 264.</u> Ejecución de un programa en tarjetas perforadas en lenguaje simbólico ASSEMBLER, que invoca rutinas y funciones en lenguaje ASSEMBLER residentes en la biblioteca de simbólicos, a mas de otras rutinas y funciones grabadas en la biblioteca de módulos de carga.....	213
<u>Modelo 265.</u> Ejecución de un programa en tarjetas perforadas en lenguaje absoluto de ASSEMBLER, que invoca rutinas y funciones en lenguaje ASSEMBLER residentes en la biblioteca de simbólicos, a mas de otras rutinas y funciones grabadas en la biblioteca de módulos de carga.....	214
<u>Modelo 266.</u> Ejecución de un programa en tarjetas perforadas en lenguaje simbólico ASSEMBLER, que invoca rutinas y funciones en lenguaje FORTRAN residentes en la biblioteca de simbólicos, a mas de otras rutinas y funciones grabadas en la biblioteca de módulos de carga.....	215
<u>Modelo 267.</u> Ejecución de un programa en tarjetas perforadas en lenguaje absoluto de ASSEMBLER, que invoca rutinas y funciones en lenguaje FORTRAN residentes en la biblioteca de simbólicos, a mas de otras rutinas y funciones grabadas en la biblioteca de módulos de carga.....	216
 <u>PUNTO 8.</u> Ejecución de programas grabados en cintas magnéticas, que invocan rutinas y funciones en lenguaje simbólico residentes en una biblioteca de nombre SIMBOLIC.CNEA, a mas de otras rutinas y funciones que son módulos de carga de una biblioteca de nombre REALES.CNEA.....	217
<u>Modelo 268.</u> Ejecución de un programa en lenguaje simbólico FORTRAN grabado en una cinta magnética, que invoca rutinas y funciones en lenguaje FORTRAN residentes en la biblioteca de simbólicos, a mas de otras rutinas y funciones grabadas en la biblioteca de módulos de carga.....	217
<u>Modelo 269.</u> Ejecución de un programa en lenguaje absoluto de FORTRAN grabado en una cinta magnética, que invoca rutinas y funciones en lenguaje FORTRAN residentes en la biblioteca de simbólicos, a mas de otras rutinas y funciones grabadas en la biblioteca de módulos de carga.....	217
<u>Modelo 270.</u> Ejecución de un programa en lenguaje absoluto de FORTRAN grabado en una cinta magnética, que invoca rutinas y funciones en lenguaje ASSEMBLER residentes en la biblioteca de simbólicos, a mas de otras rutinas y funciones grabadas en la biblioteca de módulos de carga.....	218
<u>Modelo 271.</u> Ejecución de un programa en lenguaje simbólico ASSEMBLER grabado en una cinta magnética, que invoca rutinas y funciones en lenguaje ASSEMBLER residentes en la biblioteca de simbólicos, a mas de otras rutinas y funciones grabadas en la biblioteca de módulos de carga.....	219
<u>Modelo 272.</u> Ejecución de un programa en lenguaje absoluto de ASSEMBLER grabado en una cinta magnética, que invoca rutinas y funciones en lenguaje ASSEMBLER residentes en la biblioteca de simbólicos, a mas de otras rutinas y funciones grabadas en la biblioteca de módulos de carga.....	220
<u>Modelo 273.</u> Ejecución de un programa en lenguaje simbólico ASSEMBLER grabado en una cinta magnética, que invoca rutinas y funciones en lenguaje FORTRAN residentes en la biblioteca de simbólicos, a mas de otras rutinas	

y funciones grabadas en la biblioteca de módulos de carga.....	221
<u>Modelo 274.</u> Ejecución de un programa en lenguaje absoluto de ASSEMBLER grabado en una cinta magnética, que invoca rutinas y funciones en lenguaje FORTRAN residentes en la biblioteca de simbólicos, a mas de otras rutinas y funciones grabadas en la biblioteca de módulos de carga.....	222
<u>PUNTO 9.</u> Ejecución de módulos de carga procedentes de FORTRAN, de la biblioteca REALES.CNEA, que utilizan archivos de entrada de datos residentes en una biblioteca particionada y catalogada de programas en lenguaje simbólico, de nombre SIMBOLIC.CNEA.....	224
<u>Modelo 275.</u> Ejecución de un módulo de carga procedente de FORTRAN cuyos datos de entrada constituyen un archivo que reside en la biblioteca de simbólicos, como un miembro independiente.....	224
<u>Modelo 276.</u> Ejecución de un módulo de carga procedente de FORTRAN cuyos datos de entrada son tres archivos distintos que residen en la biblioteca de simbólicos, como miembros independientes.....	224
<u>PUNTO 10.</u> Ejecución de módulos de carga procedentes de FORTRAN, de la biblioteca REALES.CNEA, que graban archivos de salida de datos en una biblioteca particionada y catalogada de programas en lenguaje simbólico, de nombre SIMBOLIC.CNEA.....	226
<u>Modelo 277.</u> Ejecución de un módulo de carga procedente de FORTRAN cuyos resultados constituyen un archivo de salida que se graba en la biblioteca de simbólicos, como un miembro independiente.....	226
<u>Modelo 278.</u> Ejecución de un módulo de carga procedente de FORTRAN cuyos resultados son tres archivos de salida distintos que se graban en la biblioteca de simbólicos, como miembros independientes.....	226
<u>PUNTO 11.</u> Ejecución de módulos de carga procedentes de FORTRAN, de la biblioteca REALES.CNEA, que utilizan archivos de entrada de datos residentes en una biblioteca particionada y catalogada de programas en lenguaje simbólico, de nombre SIMBOLIC.CNEA y simultáneamente graban archivos de salida de datos en esa misma biblioteca.....	229
<u>Modelo 279.</u> Ejecución de un módulo de carga procedente de FORTRAN cuyos datos de entrada constituyen un archivo que reside en la biblioteca de simbólicos, como un miembro independiente y cuyos resultados se graban en esa misma biblioteca, también como un miembro independiente.....	229
<u>Modelo 280.</u> Ejecución de un módulo de carga procedente de FORTRAN cuyos datos de entrada son tres archivos distintos que residen en la biblioteca de simbólicos, como miembros independientes y cuyos resultados son tres archivos de salida distintos que se graban en esa misma biblioteca, también como miembros independientes.....	230
<u>PUNTO 12.</u> Corrección de módulos de carga grabados en una biblioteca de nombre REALES.CNEA, cuyos programas fuente en lenguaje simbólico, son miembros de la biblioteca SIMBOLIC.CNEA.....	233
<u>Modelo 281.</u> Corrección de un módulo de carga procedente de FORTRAN.....	233
<u>Modelo 282.</u> Corrección de un módulo de carga procedente de ASSEMBLER.....	233
<u>Modelo 283.</u> Corrección de un módulo de carga procedente de PL/1.....	234
<u>PUNTO 13.</u> Transferencia de programas de una biblioteca de simbólicos de nombre SIMBOLIC.CNEA a una biblioteca de módulos de carga de nombre REALES.CNEA	236
<u>Modelo 284.</u> Transferencia de un programa en lenguaje FORTRAN.....	236
<u>Modelo 285.</u> Transferencia de un programa en lenguaje ASSEMBLER.....	236

<u>Modelo 286.</u> Transferencia de un programa en lenguaje PL/1.....	236
<u>PUNTO 14.</u> Transferencia simultánea de programas que se presentan en tarjetas perforadas o en cintas magnéticas, a dos bibliotecas particionadas: una de programas en lenguaje simbólico de nombre SIMBOLIC.CNEA y otra de módulos de carga de nombre REALES.CNEA.....	237
<u>Modelo 287.</u> Transferencia de un programa en lenguaje simbólico FORTRAN que se presenta en tarjetas perforadas, primero a la biblioteca de simbólicos y de ahí a la de módulos de carga.....	237
<u>Modelo 288.</u> Transferencia de un programa en lenguaje simbólico ASSEMBLER que se presenta en tarjetas perforadas, primero a la biblioteca de simbólicos y de ahí a la de módulos de carga.....	237
<u>Modelo 289.</u> Transferencia de un programa en lenguaje simbólico PL/1 que se presenta en tarjetas perforadas, primero a la biblioteca de simbólicos y de ahí a la de módulos de carga.....	238
<u>Modelo 290.</u> Transferencia de un programa en lenguaje simbólico FORTRAN grabado en una cinta magnética, primero a la biblioteca de simbólicos y de ahí a la de módulos de carga.....	238
<u>Modelo 291.</u> Transferencia de un programa en lenguaje simbólico ASSEMBLER grabado en una cinta magnética, primero a la biblioteca de simbólicos y de ahí a la de módulos de carga.....	239
<u>Modelo 292.</u> Transferencia de un programa en lenguaje simbólico PL/1 grabado en una cinta magnética, primero a la biblioteca de simbólicos y de ahí a la de módulos de carga.....	239

<u>CAPITULO VI. Modelos para la creación y mantenimiento de bibliotecas particionadas y catalogadas.....</u>	241
<u>PUNTO 1. Creación de nuevas bibliotecas particionadas y catalogadas de programas en lenguaje simbólico.....</u>	241
<u>Modelo 293. Creación de una biblioteca y su carga con programas que se presentan en tarjetas perforadas.....</u>	241
<u>Modelo 294. Creación de una biblioteca y su carga con programas grabados en rótulos consecutivos o no consecutivos de una cinta magnética.....</u>	242
<u>Modelo 295. Creación de una biblioteca y su carga con programas procedentes de otra biblioteca particionada que fué grabada en su totalidad, incluido el directorio, en una cinta magnética con el utilitario IEBCOPY....</u>	243
<u>Modelo 296. Creación de una biblioteca y su carga con programas procedentes de otra biblioteca catalogada similar, ya existente en la misma instalación.....</u>	244
<u>PUNTO 2. Creación de nuevas bibliotecas particionadas y catalogadas de módulos de carga.....</u>	246
<u>Modelo 297. Creación de una biblioteca y su carga con módulos procedentes de otra biblioteca particionada que fué grabada en su totalidad, incluido el directorio, en una cinta magnética con el utilitario IEBCOPY.....</u>	246
<u>Modelo 298. Creación de una biblioteca y su carga con módulos procedentes de otra biblioteca catalogada similar, ya existente en la misma instalación.....</u>	247
<u>PUNTO 3. Reposición en su totalidad o en parte de una biblioteca de programas en lenguaje simbólico, deteriorada accidentalmente, a partir de una cinta magnética de resguardo, que fué grabada con el utilitario IEBCOPY</u>	249
<u>Modelo 299. Reposición total de la biblioteca copiándola de la cinta magnética de resguardo.....</u>	249
<u>Modelo 300. Reposición de un miembro deteriorado de la biblioteca obteniéndolo de la cinta magnética de resguardo.....</u>	250
<u>Modelo 301. Reposición de varios miembros deteriorados de la biblioteca, obteniéndolos de la cinta magnética de resguardo.....</u>	251
<u>PUNTO 4. Reposición en su totalidad o en parte de una biblioteca de módulos de carga, deteriorada accidentalmente, a partir de una cinta magnética de resguardo, que fué grabada con el utilitario IEBCOPY.....</u>	253
<u>Modelo 302. Reposición total de la biblioteca copiándola de la cinta magnética de resguardo.....</u>	253
<u>Modelo 303. Reposición de un módulo deteriorado de la biblioteca obteniéndolo de la cinta magnética de resguardo.....</u>	254
<u>Modelo 304. Reposición de varios módulos deteriorados de la biblioteca obteniéndolos de la cinta magnética de resguardo.....</u>	255
<u>PUNTO 5. Resguardo, compresión y listado del directorio con información sobre el número de bloques del mismo que todavía no han sido utilizados.....</u>	257
<u>Modelo 305. Resguardo de la biblioteca en previsión de deterioros accidentales, con el utilitario IEBCOPY.....</u>	257
<u>Modelo 306. Compresión de la biblioteca para eliminar los espacios vacíos.</u>	257
<u>Modelo 307. Listado del directorio de la biblioteca e información sobre el número de bloques del mismo que todavía no han sido utilizados.....</u>	257
<u>PUNTO 6. Supresión y cambio de nombre de uno o mas miembros de la biblioteca.....</u>	259

<u>Modelo 308.</u> Supresión de varios miembros de la biblioteca.....	259
<u>Modelo 309.</u> Cambio de nombre de varios miembros de la biblioteca.....	259
<u>PUNTO 7.</u> Borrado y decatalogación de bibliotecas fuera de uso. Cambio del nombre de la biblioteca. Obtención del catálogo de bibliotecas contenidas en el volumen.....	261
<u>Modelo 310.</u> Borrado y decatalogación de una biblioteca fuera de uso.....	261
<u>Modelo 311.</u> Cambio de nombre de la biblioteca con catalogación del nuevo nombre y decatalogación del anterior.....	261
<u>Modelo 312.</u> Obtención del catálogo de las bibliotecas que residen en un determinado volumen de la unidad de discos e información sobre el número de cilindros y pistas libres en cada una de las bibliotecas y en el volumen.....	261
<u>Modelo 313.</u> Obtención del catálogo de las bibliotecas que residen en varios volúmenes de la unidad de discos e información sobre el número de cilindros y pistas libres en cada una de las bibliotecas y en cada uno de los volúmenes.....	262

<u>CAPITULO VII. Modelos para listado, grabación y transferencia de archivos en cintas magnéticas.....</u>	263
<u>PUNTO 1. Listado de cintas magnéticas.....</u>	263
<u>Modelo 314. Listado de varios rótulos de una cinta magnética que tiene etiquetas modelo IBM y fué grabada con el mismo DCB en todos sus rótulos...</u>	263
<u>Modelo 315. Listado de varios rótulos de una cinta magnética que tiene etiquetas modelo IBM y fué grabada con distintos DCB en dos o mas de sus rótulos.....</u>	263
<u>Modelo 316. Listado de varios rótulos de una cinta magnética que no tiene etiquetas y de la cual se conoce el DCB con que fué grabado cada rótulo..</u>	264
<u>Modelo 317. Listado de varios rótulos de una cinta magnética que no tiene etiquetas y de la cual se desconoce el DCB con que fué grabado cada rótulo</u>	265
<u>Modelo 318. Listado de un programa en FORTRAN que reside, junto con otros, en el mismo rótulo de una cinta magnética, formando un archivo secuencial.</u>	266
<u>Modelo 319. Listado de varios programas en FORTRAN que residen, junto con otros, en el mismo rótulo de una cinta magnética formando un archivo secuencial.....</u>	267
<u>Modelo 320. Listado de una rutina en lenguaje simbólico que reside, junto con otras, en el mismo rótulo de una cinta magnética y está marcada en cada uno de sus registros con una tira de caracteres.....</u>	268
<u>Modelo 321. Listado de un programa, una función y una rutina en lenguaje simbólico que residen, junto con otros, en el mismo rótulo de una cinta magnética y están marcados en cada uno de sus registros con una tira de caracteres.....</u>	268
<u>Modelo 322. Listado de los nombres solamente, de un conjunto de rutinas y funciones en lenguaje simbólico FORTRAN, grabadas secuencialmente en el mismo rótulo de una cinta magnética.....</u>	269
<u>Modelo 323. Listado de un miembro de una biblioteca particionada de programas en lenguaje simbólico, que fué grabada en su totalidad (incluido el directorio), en un solo rótulo de una cinta magnética, utilizando el utilitario IEBCOPY.....</u>	269
<u>Modelo 324. Listado de varios miembros de una biblioteca particionada de programas en lenguaje simbólico, que fué grabada en su totalidad (incluido el directorio), en un solo rótulo de una cinta magnética, utilizando el utilitario IEBCOPY.....</u>	270
<u>Modelo 325. Listado de los n primeros registros de un rótulo cualquiera de una cinta magnética que tiene etiquetas modelo IBM.....</u>	271
<u>Modelo 326. Listado de n registros intermedios de un rótulo cualquiera de una cinta magnética que tiene etiquetas modelo IBM.....</u>	271
<u>Modelo 327. Listado de los n primeros registros de un rótulo cualquiera de una cinta magnética que no tiene etiquetas.....</u>	272
<u>Modelo 328. Listado de n registros intermedios de un rótulo cualquiera de una cinta magnética que no tiene etiquetas.....</u>	272
<u>PUNTO 2. Obtención de tarjetas perforadas a partir de cintas magnéticas....</u>	273
<u>Modelo 329. Obtención de las tarjetas perforadas de varios rótulos de una cinta magnética que tiene etiquetas modelo IBM.....</u>	273
<u>Modelo 330. Obtención de las tarjetas perforadas de varios rótulos de una cinta magnética que no tiene etiquetas.....</u>	273
<u>Modelo 331. Obtención de las tarjetas perforadas de un programa en FORTRAN que reside, junto con otros, en el mismo rótulo de una cinta magnética, formando un archivo secuencial.....</u>	274
<u>Modelo 332. Obtención de las tarjetas perforadas de varios programas en FORTRAN que residen, junto con otros, en el mismo rótulo de una cinta mag-</u>	

nética, formando un archivo secuencial.....	275
<u>Modelo 333.</u> Obtención de las tarjetas perforadas de un miembro de una biblioteca particionada de programas en lenguaje simbólico, que fué grabada en su totalidad (incluido el directorio), en un solo rótulo de una cinta magnética, utilizando el utilitario IEBCOPY.....	276
<u>Modelo 334.</u> Obtención de las tarjetas perforadas de varios miembros de una biblioteca particionada de programas en lenguaje simbólico, que fué grabada en su totalidad (incluido el directorio), en un solo rótulo de una cinta magnética, utilizando el utilitario IEBCOPY.....	277
<u>Modelo 335.</u> Obtención de las tarjetas perforadas de los n primeros registros de un rótulo cualquiera de una cinta magnética que tiene etiquetas modelo IBM.....	278
<u>Modelo 336.</u> Obtención de las tarjetas perforadas de n registros intermedios de un rótulo cualquiera de una cinta magnética que tiene etiquetas modelo IBM.....	278
<u>Modelo 337.</u> Obtención de las tarjetas perforadas de los n primeros registros de un rótulo cualquiera de una cinta magnética que no tiene etiquetas	278
<u>Modelo 338.</u> Obtención de las tarjetas perforadas de n registros intermedios de un rótulo cualquiera de una cinta magnética que no tiene etiquetas	279
 <u>PUNTO 3.</u> Grabación en cintas magnéticas de programas que se presentan en tarjetas perforadas.....	280
<u>Modelo 339.</u> Grabación de programas que se presentan en tarjetas perforadas en lenguaje simbólico y en lenguaje absoluto, en varios rótulos consecutivos de una cinta magnética que tiene etiquetas modelo IBM.....	280
<u>Modelo 340.</u> Grabación de programas que se presentan en tarjetas perforadas en lenguaje simbólico y en lenguaje absoluto, en varios rótulos consecutivos de una cinta que no tiene etiquetas.....	281
<u>Modelo 341.</u> Grabación de programas que se presentan en tarjetas perforadas en lenguaje simbólico y en lenguaje absoluto, en varios rótulos consecutivos de una cinta magnética que tiene etiquetas de cabecera desconocidas o deterioradas que se desean borrar.....	282
 <u>PUNTO 4.</u> Grabación y listado de etiquetas modelo IBM.....	284
<u>Modelo 342.</u> Inicialización de una cinta magnética con etiquetas modelo IBM, que no será utilizada acto seguido para grabar (será desmontada).....	284
<u>Modelo 343.</u> Inicialización de una cinta magnética con etiquetas modelo IBM, que será utilizada acto seguido para grabar.....	284
<u>Modelo 344.</u> Listado de las etiquetas de cabecera del rótulo l de una cinta magnética que tiene etiquetas modelo IBM.....	285
<u>Modelo 345.</u> Listado de las etiquetas de cabecera de rótulos subsiguientes del l, de una cinta magnética que tiene etiquetas modelo IBM.....	285
 <u>PUNTO 5.</u> Transferencia de archivos de cinta a otra cinta; de cinta a la misma cinta a través de disco y de disco a disco a través de cinta.....	286
<u>Modelo 346.</u> Transferencia de los archivos de varios rótulos consecutivos o no, de una cinta magnética a otra nueva, con supresión de archivos, cambio de nombre, de densidad de grabación y de factor de bloqueo.....	286
<u>Modelo 347.</u> Transferencia de los archivos de varios rótulos consecutivos o no, de una cinta magnética a un disco auxiliar y de éste a la misma cinta, con el objeto de suprimir archivos fuera de uso, cambiar nombres, variar la densidad de grabación y el factor de bloqueo.....	287
<u>Modelo 348.</u> Transferencia total de una biblioteca particionada y catalogada de programas en lenguaje simbólico o de módulos de carga, indistintamente, desde la unidad de discos en que reside a la unidad que le está	

destinada en otro centro de cómputos.....	288
<u>Modelo 349.</u> Transferencia de varios miembros de una biblioteca particiona- da y catalogada de programas en lenguaje simbólico o de módulos de carga, indistintamente, desde la unidad de discos en que reside a la unidad que le está destinada en otro centro de cómputos.....	289
<u>PUNTO 6.</u> Transferencia selectiva de programas, rutinas y funciones grabadas secuencialmente en el mismo rótulo de una cinta magnética, mediante la iden- tificación de tiras de caracteres, a otros soportes de información.....	291
<u>Modelo 350.</u> Transferencia de un programa en lenguaje simbólico que reside, junto con otros, en el mismo rótulo de una cinta magnética y está marcado en cada uno de sus registros con una tira de caracteres, a una biblioteca particionada de programas en lenguaje simbólico, de nombre SIMBOLIC.CNEA..	291
<u>Modelo 351.</u> Transferencia de varias rutinas en lenguaje simbólico que re- siden, junto con otras, en el mismo rótulo de una cinta magnética y están marcadas en cada uno de sus registros con una tira de caracteres, a una biblioteca particionada de programas en lenguaje simbólico, de nombre SIMBOLIC.CNEA, como un solo miembro independiente.....	291
<u>Modelo 352.</u> Transferencia de un programa, una rutina y una función en len- guaje simbólico que residen, junto con otros, en el mismo rótulo de una cinta magnética y están marcados en cada uno de sus registros con una tira de caracteres, a una biblioteca particionada de programas en lenguaje sim- bólico, de nombre SIMBOLIC.CNEA, cada uno como un miembro independiente...	292
<u>Modelo 353.</u> Transferencia de un programa en lenguaje simbólico FORTRAN que reside, junto con otros, en el mismo rótulo de una cinta magnética y está marcado en cada uno de sus registros con una tira de caracteres, a una bi- blioteca particionada de módulos de carga, de nombre REALES.CNEA.....	293
<u>Modelo 354.</u> Transferencia de un programa en lenguaje simbólico ASSEMBLER que reside, junto con otros, en el mismo rótulo de una cinta magnética y está marcado en cada uno de sus registros con una tira de caracteres, a una biblioteca de módulos de carga, de nombre REALES.CNEA.....	294
<u>Modelo 355.</u> Transferencia de un programa en lenguaje simbólico PL/1 que reside, junto con otros, en el mismo rótulo de una cinta magnética y está marcado en cada uno de sus registros con una tira de caracteres, a una bi- blioteca de módulos de carga, de nombre REALES.CNEA.....	295
<u>Modelo 356.</u> Transferencia de varias rutinas en lenguaje simbólico que re- siden, junto con otras, en el mismo rótulo de una cinta magnética y están marcadas en cada uno de sus registros con una tira de caracteres, a rōtu- los consecutivos de otra cinta.....	295

<u>CAPITULO VIII.</u> Modelos para la ejecución de programas grabados en cintas magnéticas.....	297
<u>PUNTO 1.</u> Ejecución de programas grabados en cintas magnéticas cuyos datos se presentan en tarjetas perforadas.....	297
<u>Modelo 357.</u> Ejecución de un programa en lenguaje simbólico FORTRAN grabado completo en un solo rótulo de una cinta magnética.....	297
<u>Modelo 358.</u> Ejecución de un programa en lenguaje absoluto de FORTRAN grabado completo en un solo rótulo de una cinta magnética.....	297
<u>Modelo 359.</u> Ejecución de un programa en lenguaje simbólico FORTRAN del cual una parte está grabada en un rótulo de una cinta magnética y el resto se presenta en tarjetas perforadas.....	297
<u>Modelo 360.</u> Ejecución de un programa en lenguaje absoluto de FORTRAN del cual una parte está grabada en un rótulo de una cinta magnética y el resto se presenta en tarjetas perforadas.....	298
<u>Modelo 361.</u> Ejecución de un programa en lenguaje simbólico FORTRAN del cual una parte está grabada en un rótulo de una cinta magnética y el resto en otro rótulo de la misma cinta.....	298
<u>Modelo 362.</u> Ejecución de un programa en lenguaje absoluto de FORTRAN del cual una parte está grabada en un rótulo de una cinta magnética y el resto en otro rótulo de la misma cinta.....	298
<u>Modelo 363.</u> Ejecución de un programa en lenguaje simbólico FORTRAN del cual una parte está grabada en un rótulo de una cinta magnética y el resto en un rótulo de otra cinta.....	299
<u>Modelo 364.</u> Ejecución de un programa en lenguaje absoluto de FORTRAN del cual una parte está grabado en un rótulo de una cinta magnética y el resto en un rótulo de otra cinta.....	299
<u>Modelo 365.</u> Ejecución de un programa en lenguaje simbólico FORTRAN que reside, junto con otros, en el mismo rótulo de una cinta magnética, formando un archivo secuencial.....	300
<u>Modelo 366.</u> Ejecución de un programa en lenguaje simbólico FORTRAN que reside, junto con otros, en el mismo rótulo de una cinta magnética y está marcado en cada uno de sus registros con una tira de caracteres.....	301
<u>Modelo 367.</u> Ejecución de un programa en lenguaje simbólico FORTRAN que pertenece a una biblioteca particionada que está grabada en su totalidad, incluido el directorio y utilizando el utilitario IEBCOPY, en un rótulo de una cinta magnética.....	302
<u>Modelo 368.</u> Ejecución de un programa en lenguaje simbólico ASSEMBLER grabado completo en un solo rótulo de una cinta magnética.....	302
<u>Modelo 369.</u> Ejecución de un programa en lenguaje absoluto de ASSEMBLER grabado completo en un solo rótulo de una cinta magnética.....	303
<u>Modelo 370.</u> Ejecución de un programa en lenguaje simbólico ASSEMBLER que reside, junto con otros, en el mismo rótulo de una cinta magnética y está marcado en cada uno de sus registros con una tira de caracteres.....	303
<u>Modelo 371.</u> Ejecución de un programa en lenguaje simbólico ASSEMBLER que pertenece a una biblioteca particionada que está grabada en su totalidad, incluido el directorio y utilizando el utilitario IEBCOPY, en un rótulo de una cinta magnética.....	304
<u>Modelo 372.</u> Ejecución de un programa en lenguaje simbólico PL/1 grabado completo en un solo rótulo de una cinta magnética.....	305
<u>Modelo 373.</u> Ejecución de un programa en lenguaje absoluto de PL/1 grabado completo en un solo rótulo de una cinta magnética.....	305
<u>Modelo 374.</u> Ejecución de un programa en lenguaje simbólico PL/1 que reside, junto con otros, en el mismo rótulo de una cinta magnética y está marcado en cada uno de sus registros con una tira de caracteres.....	305

<u>Modelo 375.</u> Ejecución de un programa en lenguaje simbólico PL/1 que pertenece a una biblioteca particionada que está grabada en su totalidad, incluido el directorio y utilizando el utilitario IEBCOPY, en un rótulo de una cinta magnética.....	306
<u>PUNTO 2.</u> Uso de la referencia hacia Atrás a un programa en cinta magnética y vinculado en un paso anterior de la misma tarea, para ejecutar dos o mas juegos de datos distintos, en los pasos siguientes.....	308
<u>Modelo 376.</u> Ejecución de un programa en lenguaje simbólico FORTRAN grabado en cinta magnética, compilado y vinculado en el PASO1 durante el cual resuelve el juego de DATOS1; en el PASO2 se hace referencia al mismo programa para resolver el juego de DATOS2.....	308
<u>Modelo 377.</u> Ejecución de un programa en lenguaje absoluto de FORTRAN grabado en cinta magnética, vinculado en el PASO1 durante el cual resuelve el juego de DATOS1; en los PASOS 2 y 3 se hace referencia al mismo programa para resolver los juegos de DATOS2 y DATOS3.....	308
<u>Modelo 378.</u> Ejecución de un programa en lenguaje simbólico ASSEMBLER grabado en cinta magnética, compilado y vinculado en el PASO1 durante el cual resuelve el juego de DATOS1; en los PASOS 2 y 3 se hace referencia al mismo programa para resolver los juegos de DATOS2 y DATOS3.....	309
<u>Modelo 379.</u> Ejecución de un programa en lenguaje absoluto de ASSEMBLER grabado en cinta magnética, vinculado en el PASO1 durante el cual resuelve el juego de DATOS1; en el PASO2 se hace referencia al mismo programa para resolver el juego de DATOS2.....	309
<u>Modelo 380.</u> Ejecución de un programa en lenguaje simbólico PL/1 grabado en cinta magnética, compilado y vinculado en el PASO1 durante el cual resuelve el juego de DATOS1; en el PASO2 se hace referencia al mismo programa para resolver el juego de DATOS2.....	310
<u>Modelo 381.</u> Ejecución de un programa en lenguaje absoluto de PL/1 grabado en cinta magnética, vinculado en el PASO1 durante el cual resuelve el juego de DATOS1; en el PASO2 se hace referencia al mismo programa para resolver el juego de DATOS2.....	310
<u>PUNTO 3.</u> Uso de la Referencia hacia Atrás a un juego de datos ejecutado en un paso anterior de la misma tarea, para ser procesado por dos o mas programas grabados en cintas magnéticas, en los pasos siguientes.....	312
<u>Modelo 382.</u> Ejecución de un programa en lenguaje simbólico FORTRAN grabado en cinta magnética, compilado y vinculado en el PASO1, en el cual resuelve el juego de DATOS1 y de otro programa en lenguaje simbólico FORTRAN que se presenta en tarjetas perforadas, compilado y vinculado en el PASO2, en el cual se hace referencia al mismo juego de DATOS1.....	312
<u>Modelo 383.</u> Ejecución de un programa en lenguaje absoluto de FORTRAN grabado en cinta magnética, vinculado en el PASO1, en el cual resuelve el juego de DATOS1 y de otro programa en lenguaje absoluto de FORTRAN que se presenta en tarjetas perforadas, vinculado en el PASO2, en el cual se hace referencia al mismo juego de DATOS1.....	312
<u>Modelo 384.</u> Ejecución de un programa en lenguaje simbólico FORTRAN grabado en cinta magnética, compilado y vinculado en el PASO1, en el cual resuelve el juego de DATOS1 y de otro programa en lenguaje simbólico FORTRAN grabado en otro rótulo de la misma cinta, compilado y vinculado en el PASO2, en el cual se hace referencia al mismo juego de DATOS1.....	313
<u>Modelo 385.</u> Ejecución de un programa en lenguaje absoluto de FORTRAN grabado en cinta magnética, vinculado en el PASO1 en el cual resuelve el juego de DATOS1 y de otro programa en lenguaje absoluto de FORTRAN grabado en otro rótulo de la misma cinta, vinculado en el PASO2, en el cual se hace referencia al mismo juego de DATOS1....	313

<u>Modelo 386.</u> Ejecución de un programa en lenguaje simbólico FORTRAN grabado en cinta magnética, compilado y vinculado en el PAS01, en el cual resuelve el juego de DATOS1 y de otro programa en lenguaje simbólico FORTRAN grabado en otra cinta, compilado y vinculado en el PAS02, en el cual se hace referencia al mismo juego de DATOS1.....	314
<u>Modelo 387.</u> Ejecución de un programa en lenguaje absoluto de FORTRAN grabado en cinta magnética, vinculado en el PAS01, en el cual resuelve el juego de DATOS1 y de otro programa en lenguaje absoluto de FORTRAN grabado en otra cinta, vinculado en el PAS02, en el cual se hace referencia al mismo juego de DATOS1.....	314
<u>Modelo 388.</u> Ejecución de un programa en lenguaje simbólico ASSEMBLER grabado en cinta magnética, compilado y vinculado en el PAS01, en el cual resuelve el juego de DATOS1 y de otro programa en lenguaje simbólico ASSEMBLER que se presenta en tarjetas perforadas, compilado y vinculado en el PAS02, en el cual se hace referencia al mismo juego de DATOS1.....	315
<u>Modelo 389.</u> Ejecución de un programa en lenguaje absoluto de ASSEMBLER grabado en cinta magnética, vinculado en el PAS01, en el cual resuelve el juego de DATOS1 y de otro programa en lenguaje absoluto de ASSEMBLER que se presenta en tarjetas perforadas, vinculado en el PAS02, en el cual se hace referencia al mismo juego de DATOS1.....	316
<u>Modelo 390.</u> Ejecución de un programa en lenguaje simbólico ASSEMBLER grabado en cinta magnética, compilado y vinculado en el PAS01, en el cual resuelve el juego de DATOS1 y de otro programa en lenguaje simbólico ASSEMBLER grabado en otro rótulo de la misma cinta, compilado y vinculado en el PAS02, en el cual se hace referencia al mismo juego de DATOS1.....	316
<u>Modelo 391.</u> Ejecución de un programa en lenguaje absoluto de ASSEMBLER grabado en cinta, vinculado en el PAS01, en el cual resuelve el juego de DATOS1 y de otro programa en lenguaje absoluto de ASSEMBLER grabado en otro rótulo de la misma cinta, vinculado en el PAS02, en el cual se hace referencia al mismo juego de DATOS1.....	317
<u>Modelo 392.</u> Ejecución de un programa en lenguaje simbólico ASSEMBLER grabado en cinta magnética, compilado y vinculado en el PAS01, en el cual resuelve el juego de DATOS1 y de otro programa en lenguaje simbólico ASSEMBLER grabado en otra cinta, compilado y vinculado en el PAS02, en el cual se hace referencia al mismo juego de DATOS1.....	318
<u>Modelo 393.</u> Ejecución de un programa en lenguaje absoluto de ASSEMBLER grabado en cinta magnética, vinculado en el PAS01, en el cual resuelve el juego de DATOS1 y de otro programa en lenguaje absoluto de ASSEMBLER grabado en otra cinta, vinculado en el PAS02, en el cual se hace referencia al mismo juego de DATOS1.....	319
<u>Modelo 394.</u> Ejecución de un programa en lenguaje simbólico PL/1 grabado en cinta magnética, compilado y vinculado en el PAS01, en el cual resuelve el juego de DATOS1 y de otro programa en lenguaje simbólico PL/1 que se presenta en tarjetas perforadas, compilado y vinculado en el PAS02, en el cual se hace referencia al mismo juego de DATOS1.....	320
<u>Modelo 395.</u> Ejecución de un programa en lenguaje absoluto de PL/1 grabado en cinta magnética, vinculado en el PAS01, en el cual resuelve el juego de DATOS1 y de otro programa en lenguaje absoluto de PL/1 que se presenta en tarjetas perforadas, vinculado en el PAS02, en el cual se hace referencia al mismo juego de DATOS1.....	320
<u>Modelo 396.</u> Ejecución de un programa en lenguaje simbólico PL/1 grabado en cinta magnética, compilado y vinculado en el PAS01, en el cual resuelve el juego de DATOS1 y de otro programa en lenguaje simbólico PL/1 grabado en otro rótulo de la misma cinta, compilado y vinculado en el PAS02, en el cual se hace referencia al mismo juego de DATOS1.....	321
<u>Modelo 397.</u> Ejecución de un programa en lenguaje absoluto de PL/1 grabado en cinta magnética, vinculado en el PAS01, en el cual resuelve el juego de	

DATOS1 y de otro programa en lenguaje absoluto de PL/1 grabado en otro rótulo de la misma cinta, vinculado en el PASO2, en el cual se hace referencia al mismo juego de DATOS1.....	321
<u>Modelo 398.</u> Ejecución de un programa en lenguaje simbólico PL/1 grabado en cinta magnética, compilado y vinculado en el PASO1, en el cual resuelve el juego de DATOS1 y de otro programa en lenguaje simbólico PL/1 grabado en otra cinta, compilado y vinculado en el PASO2, en el cual se hace referencia al mismo juego de DATOS1.....	322
<u>Modelo 399.</u> Ejecución de un programa en lenguaje absoluto de PL/1 grabado en cinta magnética, vinculado en el PASO1, en el cual resuelve el juego de DATOS1 y de otro programa en lenguaje absoluto de PL/1 grabado en otra cinta, vinculado en el PASO2, en el cual hace referencia al mismo juego de DATOS1.....	322

<u>CAPITULO IX. Modelos para la ejecución de programas que utilizan cintas magnéticas como archivos de entrada de datos.....</u>	325
<u>PUNTO 1. Ejecución de programas en FORTRAN que se presentan en tarjetas perforadas y utilizan cintas magnéticas como archivos de entrada de datos...</u>	325
<u>Modelo 400. Ejecución de un programa en tarjetas perforadas en lenguaje simbólico FORTRAN, cuyos datos de entrada constituyen un archivo grabado en cinta magnética.....</u>	325
<u>Modelo 401. Ejecución de un programa en tarjetas perforadas en lenguaje absoluto de FORTRAN, cuyos datos de entrada constituyen un archivo grabado en cinta magnética.....</u>	325
<u>Modelo 402. Ejecución de un programa en tarjetas perforadas en lenguaje simbólico FORTRAN, cuyos datos de entrada son varios archivos que residen en r�tulos consecutivos o no consecutivos de una cinta magn�tica.....</u>	326
<u>Modelo 403. Ejecuci�n de un programa en tarjetas perforadas en lenguaje absoluto de FORTRAN, cuyos datos de entrada son varios archivos que residen en r�tulos consecutivos o no consecutivos de una cinta magn�tica.....</u>	326
<u>Modelo 404. Ejecuci�n de un programa en tarjetas perforadas en lenguaje simb�lico FORTRAN, cuyos datos de entrada son dos archivos que residen en dos cintas magn�ticas.....</u>	327
<u>Modelo 405. Ejecuci�n de un programa en tarjetas perforadas en lenguaje absoluto de FORTRAN, cuyos datos de entrada son dos archivos que residen en dos cintas magn�ticas distintas.....</u>	328
<u>PUNTO 2. Ejecuci�n de programas en FORTRAN, residentes en una biblioteca particionada y catalogada de programas en lenguaje simb�lico, de nombre SIMBOLIC.CNEA, que utilizan cintas magn�ticas como archivos de entrada de datos.....</u>	329
<u>Modelo 406. Ejecuci�n de un programa grabado en la biblioteca, cuyos datos de entrada constituyen un archivo que reside en una cinta magn�tica.....</u>	329
<u>Modelo 407. Ejecuci�n de un programa grabado en la biblioteca, cuyos datos de entrada son varios archivos que residen en r�tulos consecutivos o no consecutivos de una cinta magn�tica.....</u>	329
<u>Modelo 408. Ejecuci�n de un programa grabado en la biblioteca, cuyos datos de entrada son dos archivos que residen en dos cintas magn�ticas distintas</u>	330
<u>PUNTO 3. Ejecuci�n de m�dulos procedentes de FORTRAN, grabados en una biblioteca particionada y catalogada de m�dulos de carga, de nombre REALES.CNEA, que utilizan cintas magn�ticas como archivos de entrada de datos.....</u>	331
<u>Modelo 409. Ejecuci�n de un m�dulo de carga grabado en la biblioteca, cuyos datos de entrada constituyen un archivo que reside en una cinta magn�tica.....</u>	331
<u>Modelo 410. Ejecuci�n de un m�dulo de carga grabado en la biblioteca, cuyos datos de entrada son varios archivos que residen en r�tulos consecutivos o no consecutivos de una cinta magn�tica.....</u>	331
<u>Modelo 411. Ejecuci�n de un m�dulo de carga grabado en la biblioteca, cuyos datos de entrada son dos archivos que residen en dos cintas magn�ticas distintas.....</u>	332
<u>PUNTO 4. Ejecuci�n de programas en FORTRAN grabados en cintas magn�ticas, que utilizan cintas como archivos de entrada de datos.....</u>	333
<u>Modelo 412. Ejecuci�n de un programa en lenguaje simb�lico FORTRAN grabado en una cinta magn�tica, cuyos datos de entrada constituyen un archivo que reside en otro r�tulo de esa misma cinta.....</u>	333

<u>Modelo 413.</u> Ejecución de un programa en lenguaje absoluto de FORTRAN grabado en una cinta magnética, cuyos datos de entrada constituyen un archivo que reside en otro rótulo de esa misma cinta.....	333
<u>Modelo 414.</u> Ejecución de un programa en lenguaje simbólico FORTRAN grabado en una cinta magnética, cuyos datos de entrada constituyen un archivo que reside en otra cinta.....	334
<u>Modelo 415.</u> Ejecución de un programa en lenguaje absoluto de FORTRAN grabado en una cinta magnética, cuyos datos de entrada constituyen un archivo que reside en otra cinta.....	334
<u>Modelo 416.</u> Ejecución de un programa en lenguaje simbólico FORTRAN grabado en una cinta magnética, cuyos datos de entrada son varios archivos que residen en rótulos consecutivos o no consecutivos de esa misma cinta.....	335
<u>Modelo 417.</u> Ejecución de un programa en lenguaje absoluto de FORTRAN grabado en una cinta magnética, cuyos datos de entrada son varios archivos que residen en rótulos consecutivos o no consecutivos de esa misma cinta..	335
<u>Modelo 418.</u> Ejecución de un programa en lenguaje simbólico FORTRAN grabado en una cinta magnética, cuyos datos de entrada son varios archivos que residen en rótulos consecutivos o no consecutivos de otra cinta.....	336
<u>Modelo 419.</u> Ejecución de un programa en lenguaje absoluto de FORTRAN grabado en una cinta magnética, cuyos datos de entrada son varios archivos que residen en rótulos consecutivos o no consecutivos de otra cinta.....	337

<u>CAPITULO X. Modelos para la ejecución de programas que utilizan cintas magnéticas como archivos de salida de datos.....</u>	339

<u>PUNTO 1. Ejecución de programas en FORTRAN que se presentan en tarjetas perforadas y utilizan cintas magnéticas como archivos de salida de datos....</u>	339

<u>Modelo 420. Ejecución de un programa en tarjetas perforadas en lenguaje simbólico FORTRAN, cuyos resultados constituyen un archivo de salida que se graba en una cinta magnética.....</u>	339

<u>Modelo 421. Ejecución de un programa en tarjetas perforadas en lenguaje absoluto de FORTRAN, cuyos resultados constituyen un archivo de salida que se graba en una cinta magnética.....</u>	339

<u>Modelo 422. Ejecución de un programa en tarjetas perforadas en lenguaje simbólico FORTRAN, cuyos resultados son varios archivos de salida que se graban en una cinta magnética.....</u>	340

<u>Modelo 423. Ejecución de un programa en tarjetas perforadas en lenguaje absoluto de FORTRAN, cuyos resultados son varios archivos de salida que se graban en una cinta magnética.....</u>	340

<u>Modelo 424. Ejecución de un programa en tarjetas perforadas en lenguaje simbólico FORTRAN, cuyos resultados son dos archivos de salida que se graban en sendas cintas magnéticas.....</u>	341

<u>Modelo 425. Ejecución de un programa en tarjetas perforadas en lenguaje absoluto de FORTRAN, cuyos resultados son dos archivos de salida que se graban en sendas cintas magnéticas.....</u>	342

<u>PUNTO 2. Ejecución de programas en FORTRAN que se presentan en tarjetas perforadas, cuyos datos de entrada residen en una biblioteca particionada y catalogada de programas en lenguaje simbólico, de nombre SIMBOLIC.CNEA y cuyos resultados se graban en cintas magnéticas.....</u>	343

<u>Modelo 426. Ejecución de un programa en tarjetas perforadas en lenguaje simbólico FORTRAN, cuyos datos de entrada constituyen un archivo que reside en la biblioteca SIMBOLIC.CNEA y cuyos resultados configuran un archivo que se graba en una cinta magnética.....</u>	343

<u>Modelo 427. Ejecución de un programa en tarjetas perforadas en lenguaje absoluto de FORTRAN, cuyos datos de entrada constituyen un archivo que reside en la biblioteca SIMBOLIC.CNEA y cuyos resultados configuran un archivo que se graba en una cinta magnética.....</u>	344

<u>Modelo 428. Ejecución de un programa en tarjetas perforadas en lenguaje simbólico FORTRAN, cuyos datos de entrada son varios archivos que residen en la biblioteca SIMBOLIC.CNEA, como miembros independientes y cuyos resultados son varios archivos que se graban en una cinta magnética.....</u>	344

<u>Modelo 429. Ejecución de un programa en tarjetas perforadas en lenguaje absoluto de FORTRAN, cuyos datos de entrada son varios archivos que residen en la biblioteca SIMBOLIC.CNEA, como miembros independientes y cuyos resultados son varios archivos que se graban en una cinta magnética.....</u>	346

<u>Modelo 430. Ejecución de un programa en tarjetas perforadas en lenguaje simbólico FORTRAN, cuyos datos de entrada son varios archivos que residen en la biblioteca SIMBOLIC.CNEA, como miembros independientes y cuyos resultados son dos archivos que se graban en sendas cintas magnéticas.....</u>	347

<u>Modelo 431. Ejecución de un programa en tarjetas perforadas en lenguaje absoluto de FORTRAN, cuyos datos de entrada son varios archivos que residen en la biblioteca SIMBOLIC.CNEA, como miembros independientes y cuyos resultados son dos archivos que se graban en sendas cintas magnéticas.....</u>	349

<u>PUNTO 3. Ejecución de programas en FORTRAN, residentes en una biblioteca particionada y catalogada de programas en lenguaje simbólico, de nombre</u>	

SIMBOLIC.CNEA, que utilizan cintas magnéticas como archivos de salida de datos.....	351
Modelo 432. Ejecución de un programa residente en la biblioteca, cuyos resultados constituyen un archivo que se graba en una cinta magnética.....	351
Modelo 433. Ejecución de un programa residente en la biblioteca, cuyos resultados son varios archivos que se graban en una cinta magnética.....	351
Modelo 434. Ejecución de un programa residente en la biblioteca, cuyos resultados son dos archivos que se graban en sendas cintas magnéticas.....	352
 PUNTO 4. Ejecución de programas en FORTRAN, residentes en una biblioteca particionada y catalogada de programas en lenguaje simbólico, de nombre SIMBOLIC.CNEA, cuyos datos de entrada residen en la propia biblioteca y cuyos resultados se graban en cintas magnéticas.....	354
Modelo 435. Ejecución de un programa grabado en la biblioteca cuyos datos de entrada constituyen un archivo que reside en la propia biblioteca, como un miembro independiente y cuyos resultados configuran un archivo que se graba en una cinta magnética.....	354
Modelo 436. Ejecución de un programa grabado en la biblioteca cuyos datos de entrada son varios archivos que residen en la propia biblioteca, como miembros independientes y cuyos resultados son varios archivos que se graban en una cinta magnética.....	355
Modelo 437. Ejecución de un programa grabado en la biblioteca cuyos datos de entrada son varios archivos que residen en la propia biblioteca, como miembros independientes y cuyos resultados son dos archivos que se graban en sendas cintas magnéticas.....	356
 PUNTO 5. Ejecución de módulos procedentes de FORTRAN, grabados en una biblioteca particionada y catalogada de módulos de carga, de nombre REALES.CNEA, que utilizan cintas magnéticas como archivos de salida de datos.....	358
Modelo 438. Ejecución de un módulo de carga que reside en la biblioteca, cuyos resultados constituyen un archivo que se graba en una cinta magnética.....	358
Modelo 439. Ejecución de un módulo de carga que reside en la biblioteca, cuyos resultados son varios archivos que se graban en una cinta magnética.....	358
Modelo 440. Ejecución de un módulo de carga que reside en la biblioteca, cuyos resultados son dos archivos que se graban en sendas cintas magnéticas.....	359
 PUNTO 6. Ejecución de programas en FORTRAN grabados en cintas magnéticas, que utilizan cintas como archivos de salida de datos.....	361
Modelo 441. Ejecución de un programa en lenguaje simbólico FORTRAN que reside en una cinta magnética y cuyos resultados constituyen un archivo que se graba en otro rótulo de esa misma cinta.....	361
Modelo 442. Ejecución de un programa en lenguaje absoluto de FORTRAN que reside en una cinta magnética y cuyos resultados constituyen un archivo que se graba en otro rótulo de esa misma cinta.....	361
Modelo 443. Ejecución de un programa en lenguaje simbólico FORTRAN que reside en una cinta magnética y cuyos resultados constituyen un archivo que se graba en otra cinta.....	362
Modelo 444. Ejecución de un programa en lenguaje absoluto de FORTRAN que reside en una cinta magnética y cuyos resultados constituyen un archivo que se graba en otra cinta.....	362
Modelo 445. Ejecución de un programa en lenguaje simbólico FORTRAN que reside en una cinta magnética y cuyos resultados son varios archivos que se graban en rótulos consecutivos de esa misma cinta.....	363

<u>Modelo 446.</u> Ejecución de un programa en lenguaje absoluto de FORTRAN que reside en una cinta magnética y cuyos resultados son varios archivos que se graban en rótulos consecutivos de esa misma cinta.....	364
<u>Modelo 447.</u> Ejecución de un programa en lenguaje simbólico FORTRAN que reside en una cinta magnética y cuyos resultados son varios archivos que se graban en rótulos consecutivos de otra cinta.....	364
<u>Modelo 448.</u> Ejecución de un programa en lenguaje absoluto de FORTRAN que reside en una cinta magnética y cuyos resultados son varios archivos que se graban en rótulos consecutivos de otra cinta.....	365

CAPITULO XI. Modelos para la ejecución de programas que utilizan cintas magnéticas como archivos de entrada y salida de datos simultáneamente..... 367

PUNTO 1. Ejecución de programas en FORTRAN que se presentan en tarjetas perforadas y utilizan cintas magnéticas como archivos de entrada y salida de datos, simultáneamente 367

Modelo 449. Ejecución de un programa en tarjetas perforadas en lenguaje simbólico FORTRAN, cuyos datos de entrada constituyen un archivo que reside en una cinta magnética y cuyos resultados configuran un archivo que se graba en esa misma cinta..... 367

Modelo 450. Ejecución de un programa en tarjetas perforadas en lenguaje absoluto de FORTRAN, cuyos datos de entrada constituyen un archivo que reside en una cinta magnética y cuyos resultados configuran un archivo que se graba en esa misma cinta..... 368

Modelo 451. Ejecución de un programa en tarjetas perforadas en lenguaje simbólico FORTRAN, cuyos datos de entrada constituyen un archivo que reside en una cinta magnética y cuyos resultados configuran un archivo que se graba en otra cinta..... 369

Modelo 452. Ejecución de un programa en tarjetas perforadas en lenguaje absoluto de FORTRAN, cuyos datos de entrada constituyen un archivo que reside en una cinta magnética y cuyos resultados configuran un archivo que se graba en otra cinta..... 369

Modelo 453. Ejecución de un programa en tarjetas perforadas en lenguaje simbólico FORTRAN, cuyos datos de entrada son varios archivos que residen en rótulos consecutivos o no consecutivos de una cinta magnética y cuyos resultados son varios archivos de salida que se graban en esa misma cinta. 370

Modelo 454. Ejecución de un programa en tarjetas perforadas en lenguaje absoluto de FORTRAN, cuyos datos de entrada son varios archivos que residen en rótulos consecutivos o no consecutivos de una cinta magnética y cuyos resultados son varios archivos de salida que se graban en esa misma cinta..... 372

Modelo 455. Ejecución de un programa en tarjetas perforadas en lenguaje simbólico FORTRAN, cuyos datos de entrada son varios archivos que residen en rótulos consecutivos o no consecutivos de una cinta magnética y cuyos resultados son varios archivos de salida que se graban en otra cinta..... 373

Modelo 456. Ejecución de un programa en tarjetas perforadas en lenguaje absoluto de FORTRAN, cuyos datos de entrada son varios archivos que residen en rótulos consecutivos o no consecutivos de una cinta magnética y cuyos resultados son varios archivos de salida que se graban en otra cinta.. 374

PUNTO 2. Ejecución de programas en FORTRAN, residentes en una biblioteca particionada y catalogada de programas en lenguaje simbólico, de nombre SIMBOLIC.CNEA, que utilizan cintas magnéticas como archivos de entrada y salida de datos, simultáneamente..... 376

Modelo 457. Ejecución de un programa grabado en la biblioteca, cuyos datos de entrada constituyen un archivo que reside en una cinta magnética y cuyos resultados configuran un archivo de salida que se graba en esa misma cinta..... 376

Modelo 458. Ejecución de un programa grabado en la biblioteca, cuyos datos de entrada constituyen un archivo que reside en una cinta magnética y cuyos resultados configuran un archivo de salida que se graba en otra cinta. 377

Modelo 459. Ejecución de un programa grabado en la biblioteca, cuyos datos de entrada son varios archivos que residen en una cinta magnética y cuyos resultados son varios archivos de salida que se graban en esa misma cinta. 377

Modelo 460. Ejecución de un programa grabado en la biblioteca, cuyos datos

de entrada son varios archivos que residen en rótulos consecutivos o no consecutivos de una cinta magnética y cuyos resultados son varios archivos de salida que se graban en otra cinta.....	379
PUNTO 3. Ejecución de módulos procedentes de FORTRAN, grabados en una biblioteca particionada y catalogada de módulos de carga de nombre REALES. CNEA, que utilizan cintas magnéticas como archivos de entrada y salida de datos, simultáneamente.....	381
<u>Modelo 461.</u> Ejecución de un módulo de carga grabado en la biblioteca, cuyos datos de entrada constituyen un archivo que reside en una cinta magnética y cuyos resultados configuran un archivo de salida que se graba en esa misma cinta.....	381
<u>Modelo 462.</u> Ejecución de un módulo de carga grabado en la biblioteca, cuyos datos de entrada constituyen un archivo que reside en una cinta magnética y cuyos resultados configuran un archivo de salida que se graba en otra cinta.....	382
<u>Modelo 463.</u> Ejecución de un módulo de carga grabado en la biblioteca, cuyos datos de entrada son varios archivos que residen en una cinta magnética y cuyos resultados son varios archivos de salida que se graban en esa misma cinta.....	382
<u>Modelo 464.</u> Ejecución de un módulo de carga grabado en la biblioteca, cuyos datos de entrada son varios archivos que residen en una cinta magnética y cuyos resultados son varios archivos de salida que se graban en otra cinta.....	384
PUNTO 4. Ejecución de programas en FORTRAN grabados en cintas magnéticas, que utilizan cintas como archivos de entrada y salida de datos, simultáneamente.....	386
<u>Modelo 465.</u> Ejecución de un programa en lenguaje simbólico FORTRAN grabado en cinta magnética, cuyos datos de entrada constituyen un archivo que reside en esa misma cinta y cuyos resultados configuran un archivo de salida que se graba también en esa cinta.....	386
<u>Modelo 466.</u> Ejecución de un programa en lenguaje absoluto de FORTRAN grabado en cinta magnética, cuyos datos de entrada constituyen un archivo que reside en esa misma cinta y cuyos resultados configuran un archivo de salida que se graba también en esa cinta.....	387
<u>Modelo 467.</u> Ejecución de un programa en lenguaje simbólico FORTRAN grabado en cinta magnética, cuyos datos de entrada constituyen un archivo que reside en esa misma cinta y cuyos resultados configuran un archivo de salida que se graba en otra cinta.....	388
<u>Modelo 468.</u> Ejecución de un programa en lenguaje absoluto de FORTRAN grabado en cinta magnética, cuyos datos de entrada constituyen un archivo que reside en esa misma cinta y cuyos resultados configuran un archivo de salida que se graba en otra cinta.....	388
<u>Modelo 469.</u> Ejecución de un programa en lenguaje simbólico FORTRAN grabado en cinta magnética, cuyos datos de entrada constituyen un archivo que reside en otra cinta magnética y cuyos resultados configuran un archivo de salida que se graba en esta última cinta.....	389
<u>Modelo 470.</u> Ejecución de un programa en lenguaje absoluto de FORTRAN grabado en cinta magnética, cuyos datos de entrada constituyen un archivo que reside en otra cinta magnética y cuyos resultados configuran un archivo de salida que se graba en esta última cinta.....	390
<u>Modelo 471.</u> Ejecución de un programa en lenguaje simbólico FORTRAN grabado en cinta magnética, cuyos datos de entrada constituyen un archivo que reside en otra cinta y cuyos resultados configuran un archivo de salida que se graba en aquella primera cinta en que está grabado el programa.....	391

<u>Modelo 472.</u> Ejecución de un programa en lenguaje absoluto de FORTRAN grabado en cinta magnética, cuyos datos de entrada constituyen un archivo que reside en otra cinta y cuyos resultados configuran un archivo de salida que se graba en aquella primera cinta en que está grabado el programa.....	392
<u>Modelo 473.</u> Ejecución de un programa en lenguaje simbólico FORTRAN grabado en cinta magnética, cuyos datos de entrada son varios archivos que residen en rótulos consecutivos o no consecutivos de esa misma cinta y cuyos resultados son varios archivos de salida que se graban también en esa cinta.	392
<u>Modelo 474.</u> Ejecución de un programa en lenguaje absoluto de FORTRAN grabado en cinta magnética, cuyos datos de entrada son varios archivos que residen en rótulos consecutivos o no consecutivos de esa misma cinta y cuyos resultados son varios archivos de salida que se graban también en esa cinta.....	394
<u>Modelo 475.</u> Ejecución de un programa en lenguaje simbólico FORTRAN grabado en cinta magnética, cuyos datos de entrada son varios archivos que residen en rótulos consecutivos o no consecutivos de esa misma cinta y cuyos resultados son varios archivos de salida que se graban en otra cinta.....	395
<u>Modelo 476.</u> Ejecución de un programa en lenguaje absoluto de FORTRAN grabado en cinta magnética, cuyos datos de entrada son varios archivos que residen en rótulos consecutivos o no consecutivos de esa misma cinta y cuyos resultados son varios archivos de salida que se graban en otra cinta..	397
<u>Modelo 477.</u> Ejecución de un programa en lenguaje simbólico FORTRAN grabado en cinta magnética, cuyos datos de entrada son varios archivos que residen en rótulos consecutivos o no consecutivos de otra cinta y cuyos resultados son varios archivos de salida que se graban en esta última cinta.....	398
<u>Modelo 478.</u> Ejecución de un programa en lenguaje absoluto de FORTRAN grabado en cinta magnética, cuyos datos de entrada son varios archivos que residen en rótulos consecutivos o no consecutivos de otra cinta y cuyos resultados son varios archivos de salida que se graban en esta última cinta.....	399
<u>Modelo 479.</u> Ejecución de un programa en lenguaje simbólico FORTRAN grabado en cinta magnética, cuyos datos de entrada son varios archivos que residen en rótulos consecutivos o no consecutivos de otra cinta y cuyos resultados son varios archivos de salida que se graban en aquella primera cinta en que está grabado el programa.....	401
<u>Modelo 480.</u> Ejecución de un programa en lenguaje absoluto de FORTRAN grabado en cinta magnética, cuyos datos de entrada son varios archivos que residen en rótulos consecutivos o no consecutivos de otra cinta y cuyos resultados son varios archivos de salida que se graban en aquella primera cinta en que está grabado el programa.....	402

<u>CAPITULO XII.</u> Modelos para la utilización de cintas magnéticas como archivos de módulos de programas o de bibliotecas particionadas cuando no se dispone de espacio propio en una unidad de discos.....	405

<u>PUNTO 1.</u> Utilización de cintas magnéticas como archivo del módulo de un programa escrito en dos lenguajes simbólicos distintos.....	405

<u>Modelo 481.</u> Obtención del módulo, archivo del mismo en cinta magnética y su ulterior ejecución, de un programa en FORTRAN que se presenta en tarjetas perforadas e invoca una rutina en ASSEMBLER, también en tarjetas perforadas.....	405

<u>Modelo 482.</u> Obtención del módulo, archivo del mismo en cinta magnética y su ulterior ejecución, de un programa en FORTRAN que se presenta en tarjetas perforadas e invoca una rutina en ASSEMBLER que reside en cinta magnética.....	405

<u>Modelo 483.</u> Obtención del módulo, archivo del mismo en cinta magnética y su ulterior ejecución, de un programa en FORTRAN que reside en cinta magnética e invoca una rutina en ASSEMBLER que se presenta en tarjetas perforadas.....	406

<u>Modelo 484.</u> Obtención del módulo, archivo del mismo en cinta magnética y su ulterior ejecución, de un programa en ASSEMBLER que se presenta en tarjetas perforadas e invoca una rutina en FORTRAN, también en tarjetas perforadas.....	407

<u>Modelo 485.</u> Obtención del módulo, archivo del mismo en cinta magnética y su ulterior ejecución, de un programa en ASSEMBLER que se presenta en tarjetas perforadas e invoca una rutina en FORTRAN que reside en cinta magnética.....	408

<u>Modelo 486.</u> Obtención del módulo, archivo del mismo en cinta magnética y su ulterior ejecución, de un programa en ASSEMBLER grabado en cinta magnética e invoca una rutina en FORTRAN que se presenta en tarjetas perforadas	409

<u>PUNTO 2.</u> Creación y uso de una biblioteca particionada de programas en lenguaje simbólico, de nombre SIMBOLIC.TAPE y su archivo en una cinta magnética dispuesta especialmente para su conservación.....	411

<u>Modelo 487.</u> Creación de la biblioteca y su carga con programas, rutinas y funciones que se presentan en tarjetas perforadas.....	411

<u>Modelo 488.</u> Creación de la biblioteca y su carga con programas, rutinas y funciones que están grabados en una cinta magnética.....	412

<u>Modelo 489.</u> Grabación en la biblioteca, ya creada y conservada en cinta magnética, de nuevos programas, rutinas y funciones que se presentan en tarjetas perforadas.....	413

<u>Modelo 490.</u> Grabación en la biblioteca, ya creada y conservada en cinta magnética, de nuevos programas, rutinas y funciones que están grabados en una cinta magnética.....	414

<u>Modelo 491.</u> Listado de la totalidad de la biblioteca ya creada y conservada en cinta magnética.....	415

<u>Modelo 492.</u> Listado de un miembro de la biblioteca ya creada y conservada en cinta magnética.....	416

<u>Modelo 493.</u> Listado de varios miembros de la biblioteca ya creada y conservada en cinta magnética.....	416

<u>Modelo 494.</u> Ejecución de un programa en FORTRAN grabado en la biblioteca ya creada y conservada en cinta magnética.....	417

<u>Modelo 495.</u> Ejecución de un programa en ASSEMBLER grabado en la biblioteca ya creada y conservada en cinta magnética.....	418

<u>Modelo 496.</u> Ejecución de un programa en PL/I grabado en la biblioteca ya creada y conservada en cinta magnética.....	418

<u>Modelo 497.</u> Corrección, inserción y supresión de registros de un programa en FORTRAN, grabado en la biblioteca ya creada y conservada en cinta magnética.....	419
<u>Modelo 498.</u> Renumeración total de un programa grabado en la biblioteca ya creada y conservada en cinta magnética.....	420
<u>Modelo 499.</u> Supresión de varios miembros de la biblioteca ya creada y conservada en cinta magnética.....	421
<u>Modelo 500.</u> Resguardo de la biblioteca en previsión de deterioros accidentales de la cinta magnética dispuesta especialmente para su conservación.....	421

I N D I C E IV. UBICACION DE LOS CAPITULOS Y SUS PUNTOS EN EL INDICE III.

CAP. I pág. 431

P 1, pág. 431; P 2, pág. 431; P 3, pág. 431; P 4, pág. 431; P 5, pág. 431;
P 6, pág. 432; P 7, pág. 433; P 8, pág. 433.

CAP. II pág. 434

P 1, pág. 434; P 2, pág. 434; P 3, pág. 434; P 4, pág. 434; P 5, pág. 435;
P 6, pág. 435; P 7, pág. 435; P 8, pág. 435; P 9, pág. 436; P 10, pág. 437;
P 11, pág. 437; P 12, pág. 437; P 13, pág. 437; P 14, pág. 438.

CAP. III pág. 439

P 1, pág. 439; P 2, pág. 439; P 3, pág. 440; P 4, pág. 441; P 5, pág. 441.

CAP. IV pág. 443

P 1, pág. 443; P 2, pág. 443; P 3, pág. 443; P 4, pág. 443; P 5, pág. 444;
P 6, pág. 444; P 7, pág. 445; P 8, pág. 445; P 9, pág. 445; P 10, pág. 445.

CAP. V pág. 447

P 1, pág. 447; P 2, pág. 447; P 3, pág. 447; P 4, pág. 448; P 5, pág. 448;
P 6, pág. 449; P 7, pág. 450; P 8, pág. 451; P 9, pág. 452; P 10, pág. 452;
P 11, pág. 452; P 12, pág. 452; P 13, pág. 452; P 14, pág. 453.

CAP. VI pág. 454

P 1, pág. 454; P 2, pág. 454; P 3, pág. 454; P 4, pág. 454; P 5, pág. 454;
P 6, pág. 454; P 7, pág. 455.

CAP. VII pág. 456

P 1, pág. 456; P 2, pág. 456; P 3, pág. 457; P 4, pág. 457; P 5, pág. 457;
P 6, pág. 458.

CAP. VIII pág. 459

P 1, pág. 459; P 2, pág. 460; P 3, pág. 460.

CAP. IX pág. 463

P 1, pág. 463; P 2, pág. 463; P 3, pág. 463; P 4, pág. 463.

CAP. X pág. 465

P 1, pág. 465; P 2, pág. 465; P 3, pág. 465; P 4, pág. 466; P 5, pág. 466;
P 6, pág. 466

CAP. XI pág. 468

P 1, pág. 468; P 2, pág. 468; P 3, pág. 469; P 4, pág. 469

CAP. XII pág. 471

P 1, pág. 471; P 2, pág. 471

A G R A D E C I M I E N T O S

Deseo manifestar mi reconocimiento a los Analistas de Sistemas de la División Investigación y Control de Gestión del Centro Unico de Procesamiento Electrónico de Datos, sin cuya permanente asistencia no hubiera podido concretar el presente trabajo. En especial menciono a los Señores Francisco J. Rindone, Eduardo García Segá, Rubén Sotera y Lazlo de Suto que me ayudaron a resolver los modelos mas complicados.

A los Doctores Ernesto E. Maqueda, Olga M. Dragún, Ricardo F. Baggio, Erhard O. Achterberg, Roberto P. Perazzo del Departamento de Física, al Doctor Héctor Schilman y al Señor Diego Murias del Departamento de Información Técnica de la CNEA, debo agradecer útiles discusiones e interesantes sugerencias.

El Doctor Osvaldo Gossman y la Computadora Científica Señora Elena Marengo pusieron a mi disposición su experiencia en computación y su vasto caudal de conocimientos en el tema, por lo cual les estoy muy agradecida.

A la eficiente intervención del Señor Aníbal Furze debo la preparación de programas en ASSEMBLER con los cuales se probaron los modelos de esta Guía en que aparece ese lenguaje.

Mi reconocimiento a las personas aquí nombradas no implica, en modo alguno, asignarles responsabilidad por los errores que puedan haberse deslizado en la presente Guía.

B I B L I O G R A F I A

1. IBM System/360
FORTRAN IV Language
2. IBM System/360 Operating System
FORTRAN IV (G and H) Programmer's Guide
3. IBM System/360 Operating System
Assembler Language
4. IBM System/360 Operating System
Assembler (F) Programmer's Guide
5. IBM System/360 Operating System
PL/I (F) Language Reference Manual
6. IBM System/360 Operating System
PL/I (F) Programmer's Guide
7. IBM System/360 Operating System
Job Control Language Reference
8. IBM Systems Reference Library
OS Utilities
9. Del Franco M.G. y Marengo E.C., Informe CNEA-AC21, Buenos Aires, 1975
10. Gossman O., IBM/360 Lenguaje de Control de Trabajo, Buenos Aires, 1976

