

UNIVERSIDAD DE SEVILLA

ESCUELA TECNICA SUPERIOR DE INGENIEROS INDUSTRIALES

T. 1  
-----  
32

\* \* \*

T E S I S

ANALISIS Y DISEÑO ELASTOPLASTICO DE ESTRUCTURAS

PLANAS FORMADAS POR BARRAS PRISMATICAS

PRESENTADA POR

JOSE CAÑAS DELGADO

PARA LA OBTENCION DEL GRADO  
DE DOCTOR INGENIERO INDUSTRIAL

T O M O II

SEVILLA , MARZO 1.986

---

TESIS DOCTORAL

ANALISIS Y DISEÑO ELASTOPLASTICO DE  
ESTRUCTURAS PLANAS FORMADAS POR BARRAS PRISMATICAS

Por : JOSE CAÑAS DELGADO

Director : Dr. RAFAEL PICON CARRIZOSA

Sevilla, Marzo de 1986

## T O M O I I

|  |    |
|--|----|
| ANEJO ASPECTOS COMPUTACIONALES         | 1  |
| 1 INTRODUCCION                         | 2  |
| 2 DESCRIPCION DEL PROGRAMA             | 3  |
| 2.1. Programas Principales             | 3  |
| 2.2. Segmentos                         | 3  |
| 2.3. Subrutinas y Funciones            | 6  |
| APENDICE I "ARQUITECTURA DEL PROGRAMA" | 12 |
| APENDICE II "MANUAL DE USUARIO"        | 33 |
| APENDICE III "LISTADO DE ORDENADOR"    | 54 |

\* \* \*

ASPECTOS COMPUTACIONALES



## 1. INTRODUCCION

En el presente Anejo, se analiza y se describe el programa de cálculo desarrollado para el análisis y diseño límite de estructuras planas, formadas por barras prismáticas.

El programa se ha realizado en lenguaje FORTRAN IV y se encuentra implementado en un miniordenador HP-21Mx del Centro de Cálculo de la E.T.S.I.I. de Sevilla.

Dos son los objetivos que se han perseguido en la confección de los programas. Por una parte se ha optimizado la programación en el sentido de utilización de la memoria central que el programa utiliza (28 K). El otro objetivo perseguido, ha sido por una parte, conseguir una cómoda y fácil entrada de datos que permitan al posible usuario del programa una rápida codificación del problema. Asimismo, se ha intentado que la salida de resultados impresa, sea lo más clara y corta posible, evitando de esta forma posibles confusiones en la interpretación de los mismos.

Con objeto de futuras ampliaciones, la confección de los programas, se ha realizado usando técnicas de programación que permiten el acople de nuevas subrutinas y segmentos al programa ya existente sin modificación sustancial del mismo.

Con el fin de evitar errores numéricos derivados de malos condicionamientos de las matrices, las ecuaciones manejadas por el programa y desarrolladas en los capítulos II, III y IV han sido adimensionalizadas.

## 2. DESCRIPCION DEL PROGRAMA

El programa de cálculo desarrollado consta de:

- 2 Programas Principales
  - PLASI
  - PLADI
- 31 Segmentos
- 77 Subrutinas
- 1 Función

La Arquitectura básica del programa se detalla en los diagramas de bloques del Apendice I.

### 2.1. Programas Principales

La misión de los dos programas principales, es la de reservar el área COMMON necesaria.

- PLASI.- Programa principal encargado del control de los problemas de análisis y diseño sin restricciones.
- PLADI.- Programa principal encargado del control de los problemas de diseño bajo restricciones de servicio y de carga de colapso.

### 2.2. Segmentos

- LEDAT.- Segmento para la lectura y escritura de los datos geométricos y mecánicos de la estructura.
- LECAR.- Segmento encargado de la lectura y escritura de los datos de carga.

- **FORMA.**- Es el encargado de la formación de la matriz restrictiva y cálculo de las soluciones independientes.
- **MECAN.**- Formación de las ecuaciones de equilibrio.
- **TECAN.**- Determina el vector o vectores de cargas externas asociado a cada ecuación de equilibrio.
- **ITERA.**- Forma las ecuaciones de equilibrio asociadas a barras cargadas.
- **COLAP.**- Realiza el Análisis Estático. Calcula la carga de Colapso.
- **INFOR.**- Extrae la información necesaria del Análisis Estático y forma las Ecuaciones de Compatibilidad.
- **ROTAC.**- Determina la solución de momentos y rotaciones plásticas. Detección de pasivas.
- **MOVIM.**- Cálculo de los desplazamientos. Detección final. Modifica las ecuaciones si procede.
- **PASO1.**- Prepara y determina las matrices necesarias para el análisis paso a paso.
- **PAS1B.**- Determina los vectores de carga generalizados para el análisis paso a paso.
- **PASO2.**- Determina los incrementos de rotación plástica de cada etapa, comprueba las descargas y chequea el colapso o pseudo.

- PASO3.- Calcula los incrementos de momentos de una etapa.
- PAS3B.- Obtiene el incremento de carga seguro de la etapa. Actualiza si procede las ecuaciones asociadas a nudos variables.
- PAS3C.- Cálcula la carga de la etapa.
- PASO4.- Obtención de los desplazamientos y escritura de la etapa.
- PASO5.- Obtención de los Axiles y modificación del vector de carga. Análisis paso a paso 2º Orden.
- SHDO1.- Obtención de la Envolvente máxima y mínima de los momentos elásticos.
- SHDO2.- Determinación de la carga de Shake-Down y escritura de resultados.
- DISEN.- Formación de la matriz de restricciones en un problema de diseño límite. Resolución.
- DISEP.- Elección del perfil más adecuado. Diseño práctico.
- RESUL.- Escritura de los resultados.
- COMUN.- Determina las matrices fijas en el problema de diseño con restricciones. Pasa el control al programa principal PLADI.
- DRSU1.- Recoge los datos transmitidos por el Segmento COMUN
- DRSU2.- Obtención de la solución inicial de diseño.

- DRSU3.- Linealiza las restricciones. Comprueba la violación de las restricciones.
- DRSU4.- Formación de la matriz de restricciones. Resolución del problema de P.L.
- DRSU5.- Compara si la solución obtenida es la correcta. Determina el peso de la estructura.
- DRSU6.- Obtención de las restricciones de colapso. Análisis de colapso.
- DRSU7.- Escritura de las soluciones finales.

### 2.3. Subrutinas y Funciones

A continuación, se describen brevemente, las subrutinas y función empleadas.

- APCUN.- Calcula las deformaciones debidas a cargas uniformes
- CAGE.- Cálcula el vector de carga generalizado.
- CALIN.- Obtención de los incrementos de momentos flectores.
- CAMBI.- Determinación del Momento máximo y posición.
- CAMZ.- Calcula el valor actual de las variables en función de la posición de los nudos variables.
- CAXIL.- Determina los esfuerzos Axiles.

- COLAI.- Algoritmo S.V.A.
- CSHDO.- Calcula la carga de Shakedown.
- DACAF.- Lectura y escritura de las cargas fijas.
- DACU.- Obtención del vector de cargas uniformes.
- DACAV.- Lectura y escritura de las cargas variables.
- DATOS.- Lectura y escritura de los datos geométricos y mecánicos de la estructura.
- DECET.- Determina el incremento de carga seguro de la etapa.
- DEFO.- Calcula el vector de deformaciones.
- DELMO.- Modifica el desplazamiento relativo en barras cargadas
- DELTA.- Determina el vector de desplazamientos relativos.
- DEMAX.- Calcula la envolvente de momentos elásticos.
- DEPSE.- Chequea los pseudomecanismos.
- DESCA.- Algoritmo de descargas.
- DISCO.- Lectura-escritura en ficheros.
- ECUA.- Determina la distribución de momentos y rotaciones plásticas.
- ELEC.- Elección del perfil adecuado.

- ELECO.- Obtención de las propiedades en función del momento plástico.
- ELETE.- Obtención de las constantes para las características mecánicas.
- ESCRM.- Escritura de una matriz.
- ESSHD.- Escritura de los resultados de Shakedown.
- FLEX.- Obtención de los coeficientes de estabilidad.
- FOMHI.- Formación de  $\underline{b}_H^T \underline{A} \underline{b}_H$ .
- FOMTI.- Formación de  $\underline{b}_H^T \underline{A} \underline{b}_C$ .
- FORBF.- Determina la respuesta elástica en el caso de cargas no proporcionales o variables.
- FORBL.- Determina la respuesta elástica en el caso de cargas proporcionales.
- FORCA.- Formación del vector de Trabajo.
- FORCU.- Corrección del vector de trabajo en caso de cargas uniformes.
- FORDI.- Formación de la matriz de restricciones en un problema de diseño.
- FORM.- Formación de la matriz restrictiva.
- FORVI.- Formación de los virtuales.

- 
- FORY.- Formación de la matriz Y.
  - FORZ.- Formación de la matriz Z.
  - GEREH.- Cálculo del Jacobiano.
  - IMPRE.- Escritura de una Etapa.
  - INVER.- Inversión de una matriz.
  - ITER.- Corrección de las ecuaciones de equilibrio con cargas uniformes.
  - LEEDI.- Lectura secuencial de fichero.
  - LEIND.- Lectura indirecta de fichero.
  - LOPI.- Cálculo de la carga de colapso.
  - MECA.- Formación de ecuaciones de equilibrio.
  - MECNU.- Formación de las ecuaciones de nudo.
  - METRA.- Formación de las ecuaciones traslacionales.
  - MODIF.- Altera el vector de cargas externas. Reducción  $M_p$ .
  - MOVI.- Cálculo de los movimientos generalizados.
  - MOVIT.- Cálculo de los desplazamientos globales.
  - OBTCO.- Obtención de la carga de la etapa.

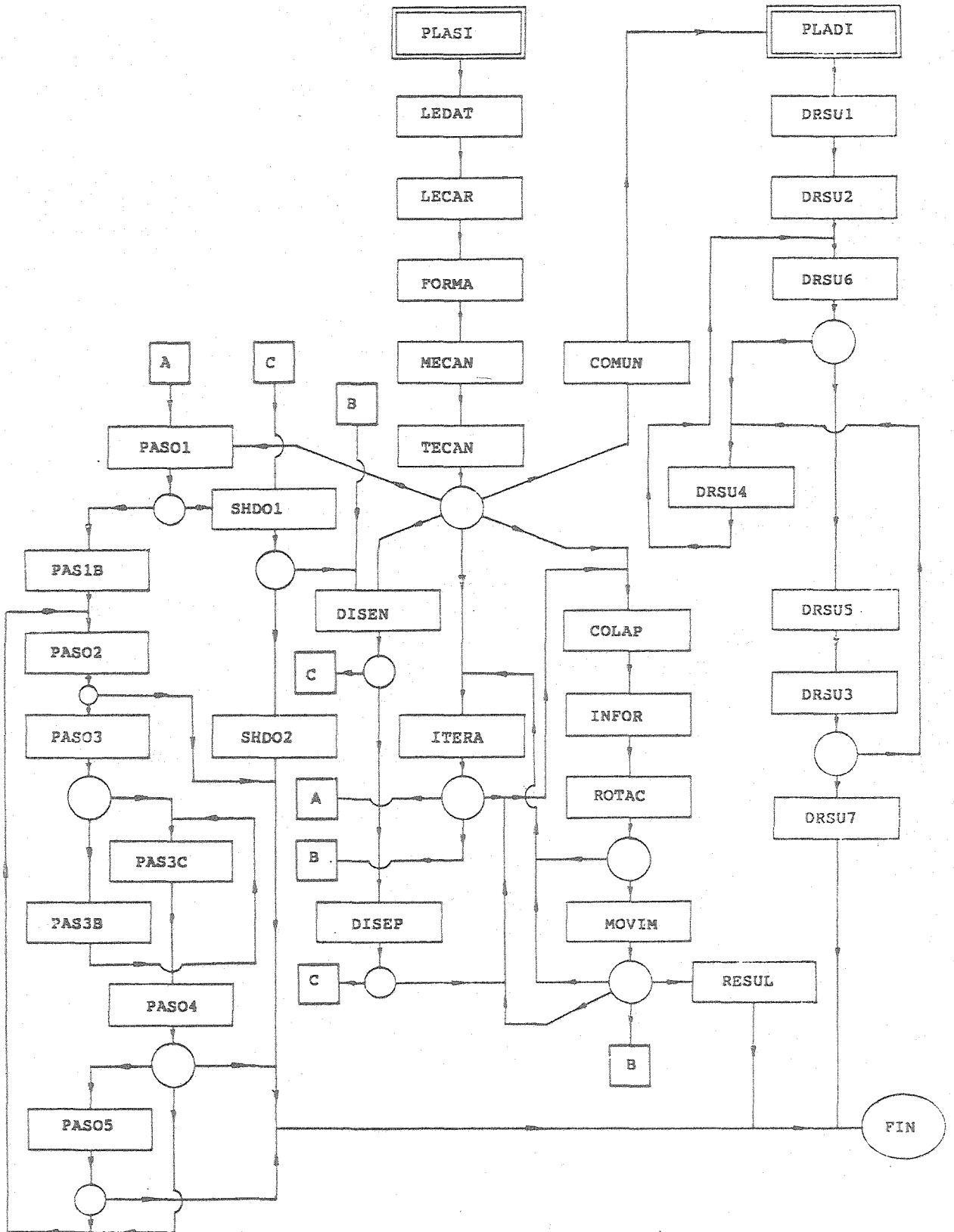


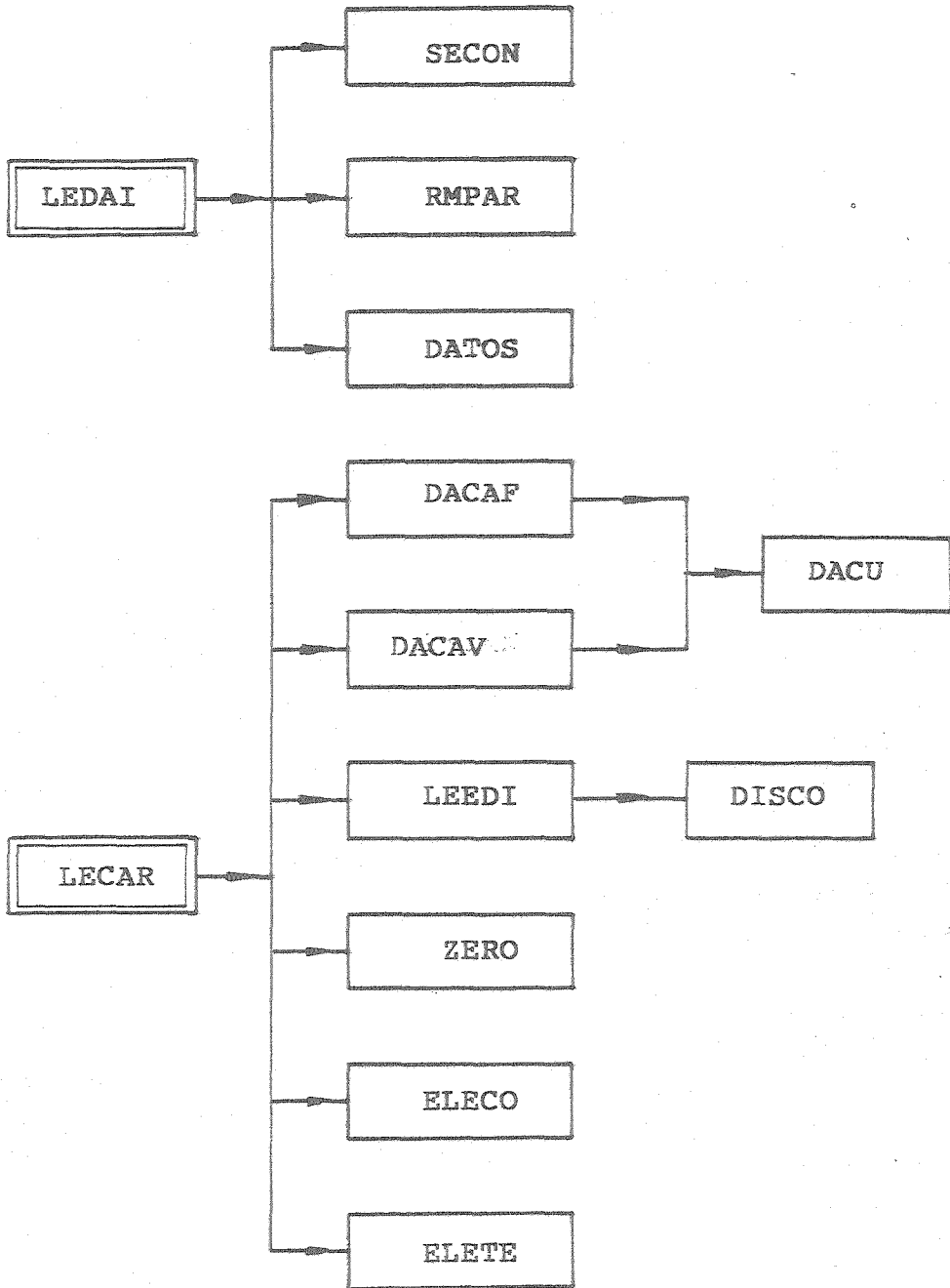
- OBTER.- Determina la matriz de equilibrio reducida.
- OBTFI.- Obtención del incremento de rotación plástica.
- OBTMO.- Obtención de los movimientos de la etapa.
- PASOI.- Tratamiento de una etapa genérica.
- PIV.- Pivotamiento de una columna genérica.
- PIV1.- Elección de pivote.
- POS.- Reordenación interna.
- PRE.- Preparación de la matriz del S.V.A.
- PROMA.- Multiplica dos matrices.
- PRTI.- Elección adecuada de los pivotes en caso de cargas uniformes.
- RESOL.- Cálculo de las soluciones independientes.
- RESUT.- Escritura de resultados.
- RMPAR.- Obtención de las unidades entrada/salida.
- SECON.- Obtención del Tiempo de C.P.U. de una Sesión.
- SIMP.- Algoritmo Simplex.
- SLNPD.- Resolución de un sistema de Ecuaciones.
- SOELA.- Determinación de la respuesta elástica.

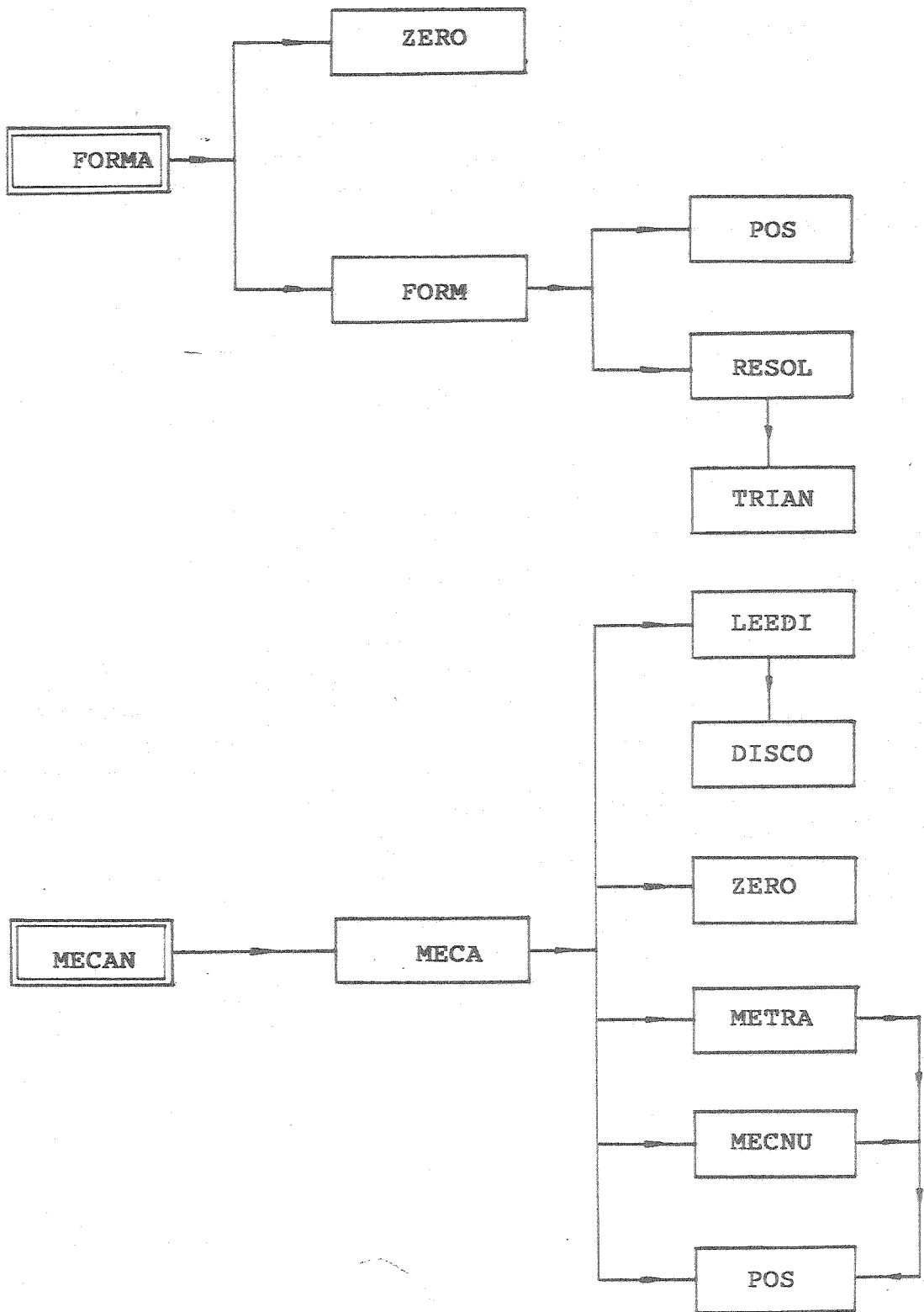
- TOLE.- Función de Truncamiento.
- TRCUM.- Tratamiento de las C.U. en el análisis paso a paso.  
Actualización de variables.
- TRIAN.- Triangulación de una matriz.
- TRABA.- Cálculo de los términos de trabajo.
- TRAB2.- Corrige los términos de trabajo.
- VARIA.- Determinar los desplazamientos en nudos variables.
- VIRTI.- Generación de las virtuales.
- VIRTU.- Generación de códigos de ordenación.
- ZERO.- Puesta a cero de una matriz.

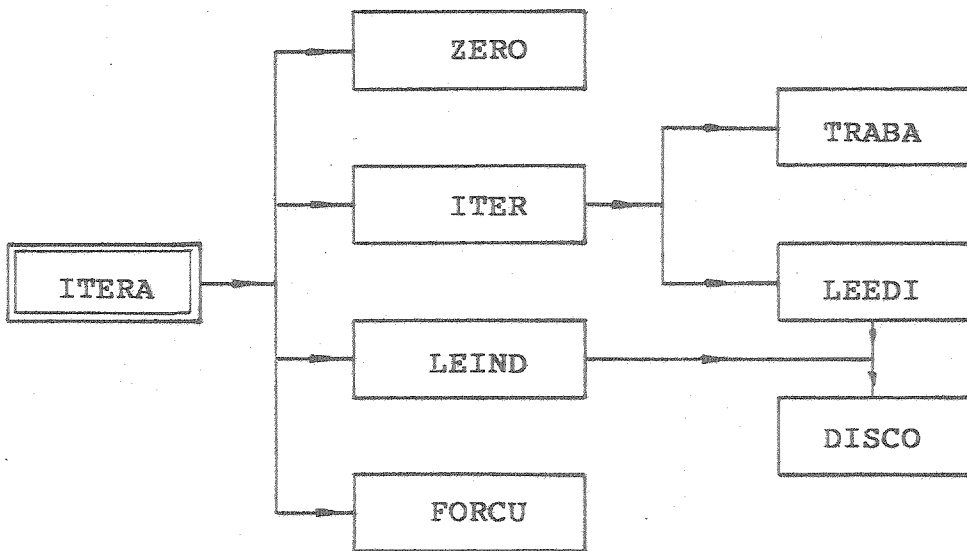
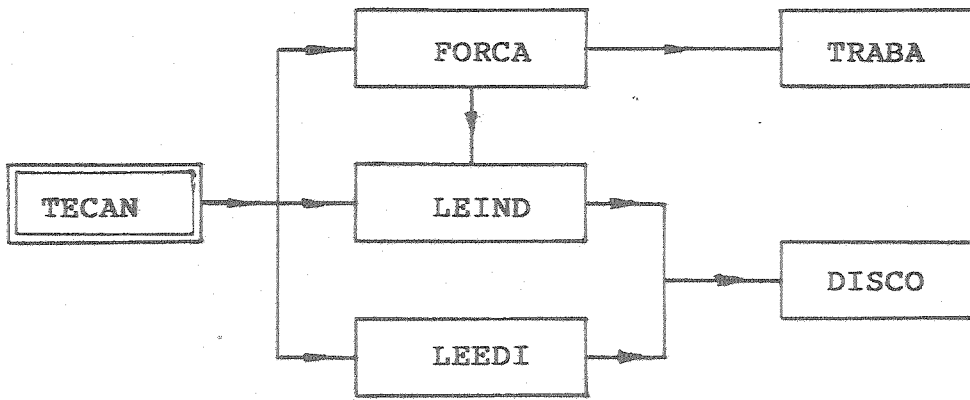
APENDICE 1

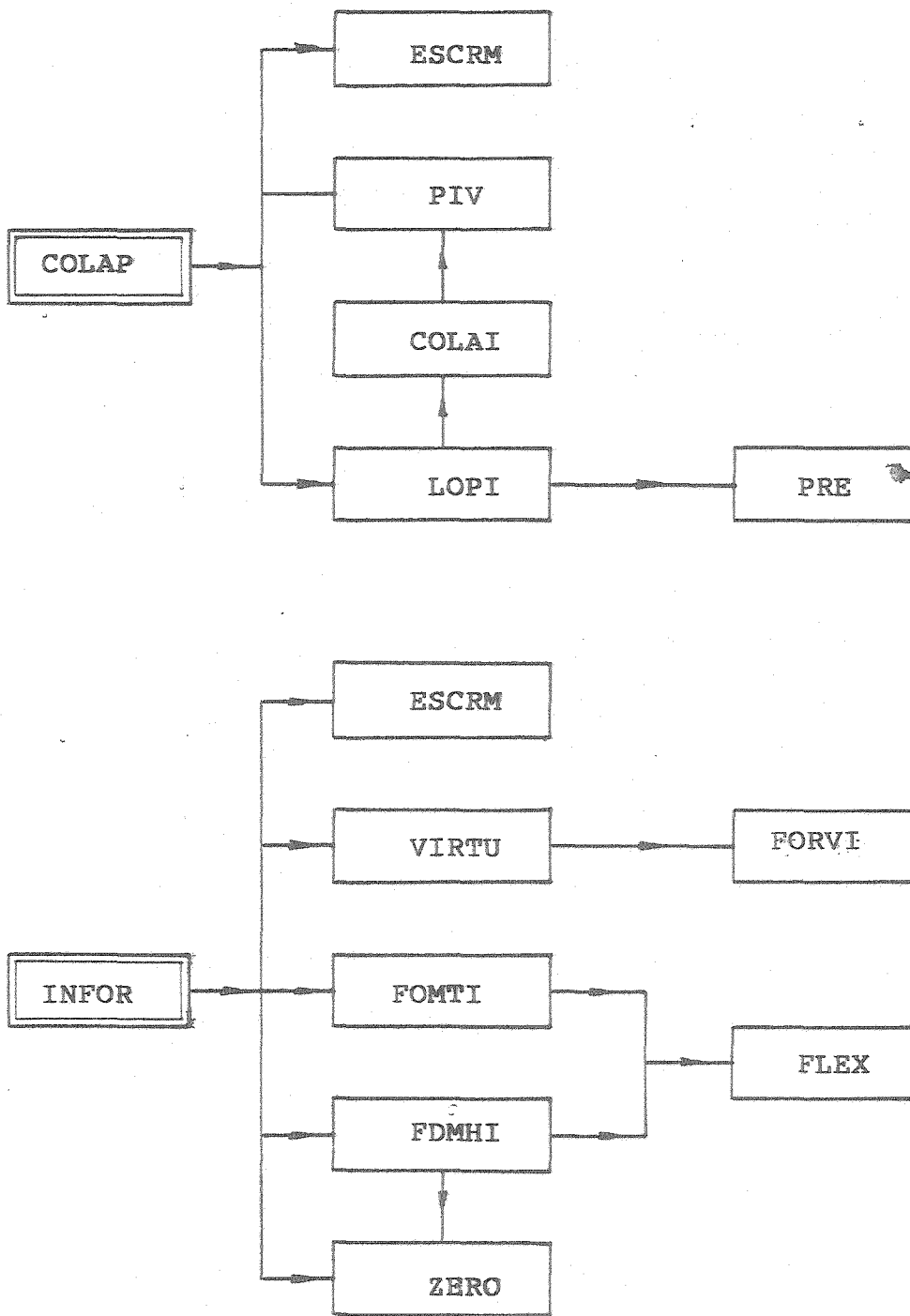
"ARQUITECTURA DEL PROGRAMA"



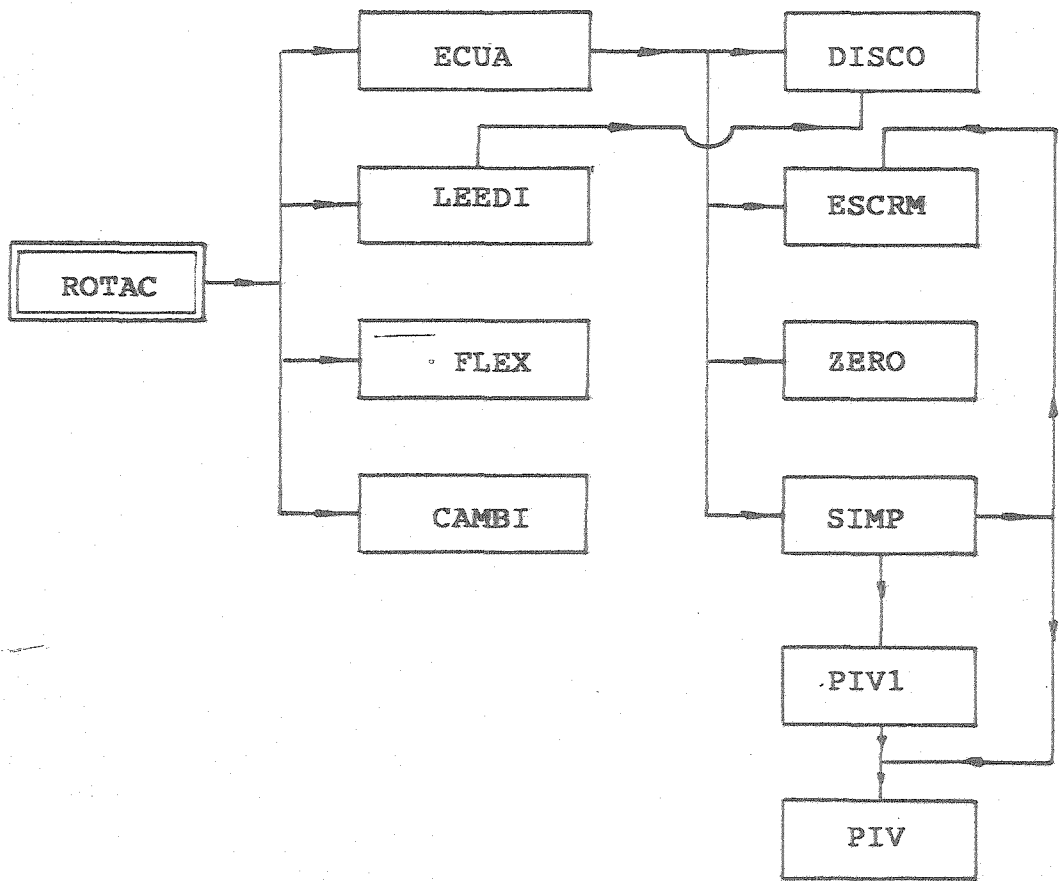


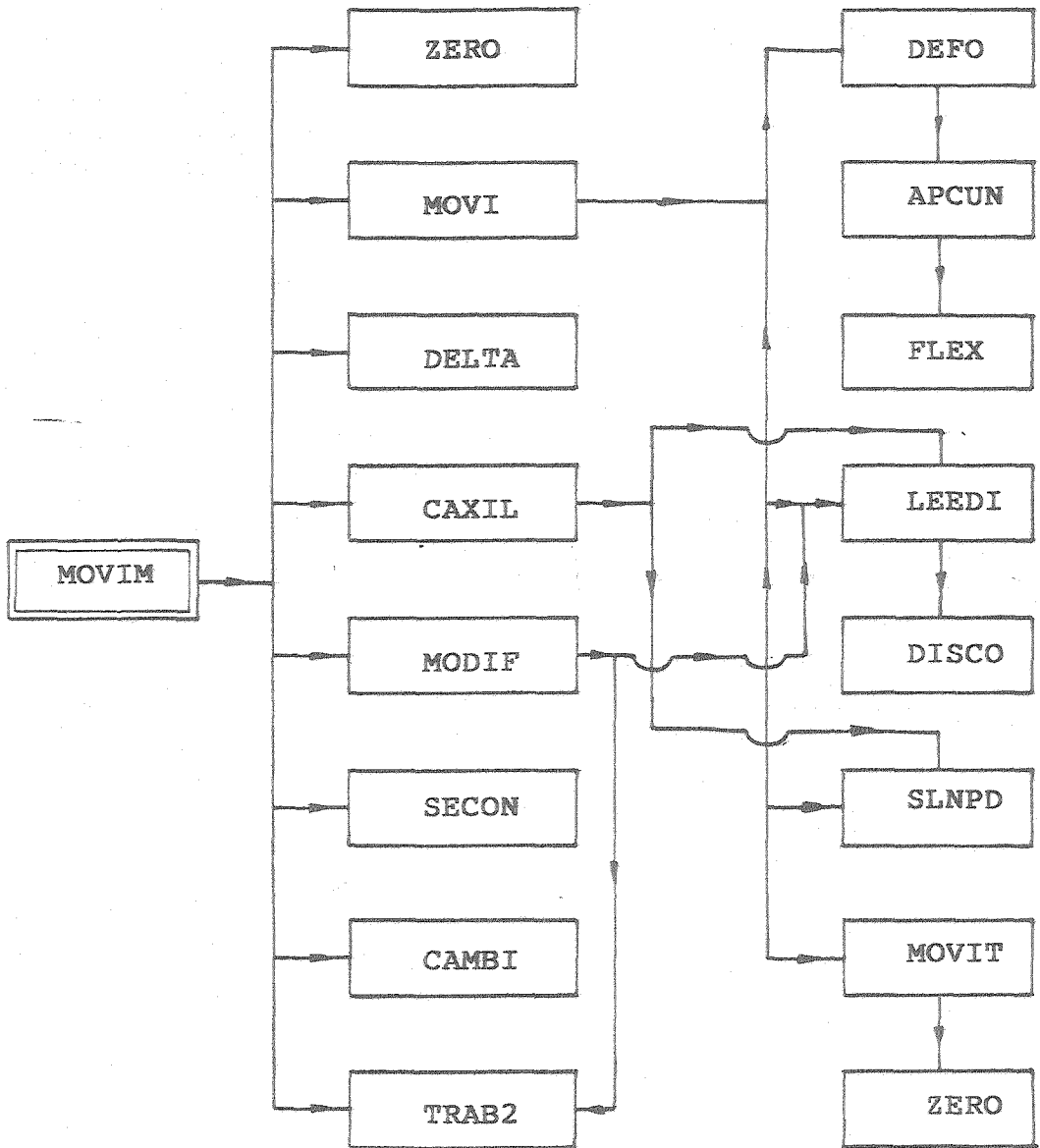


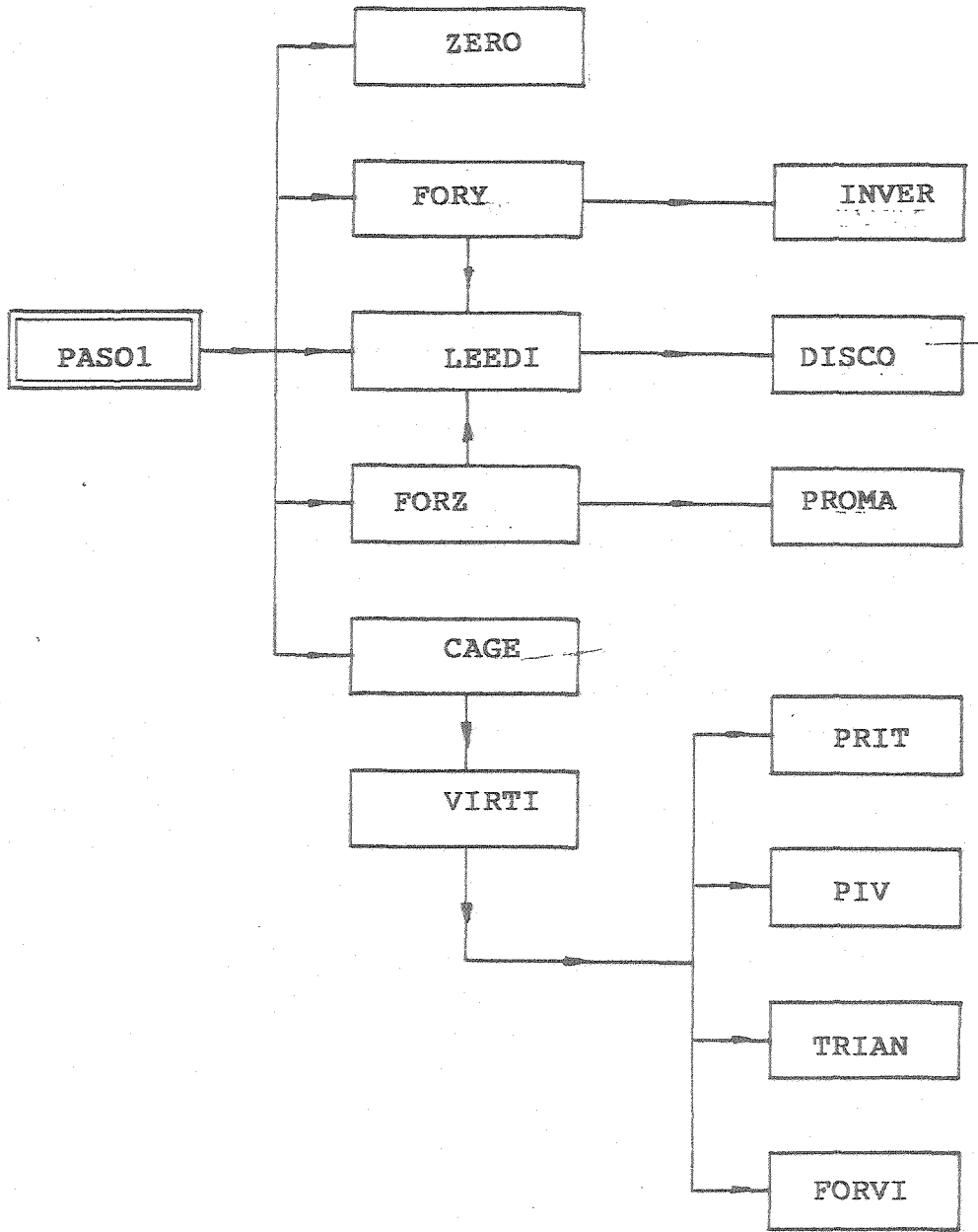


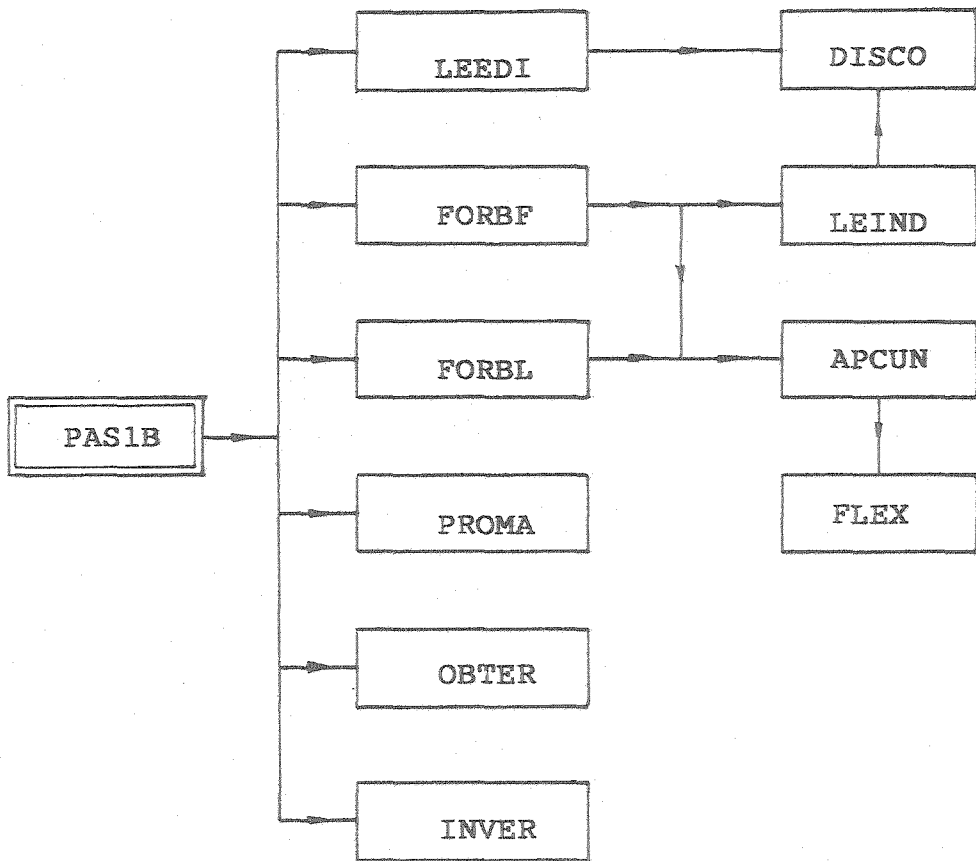




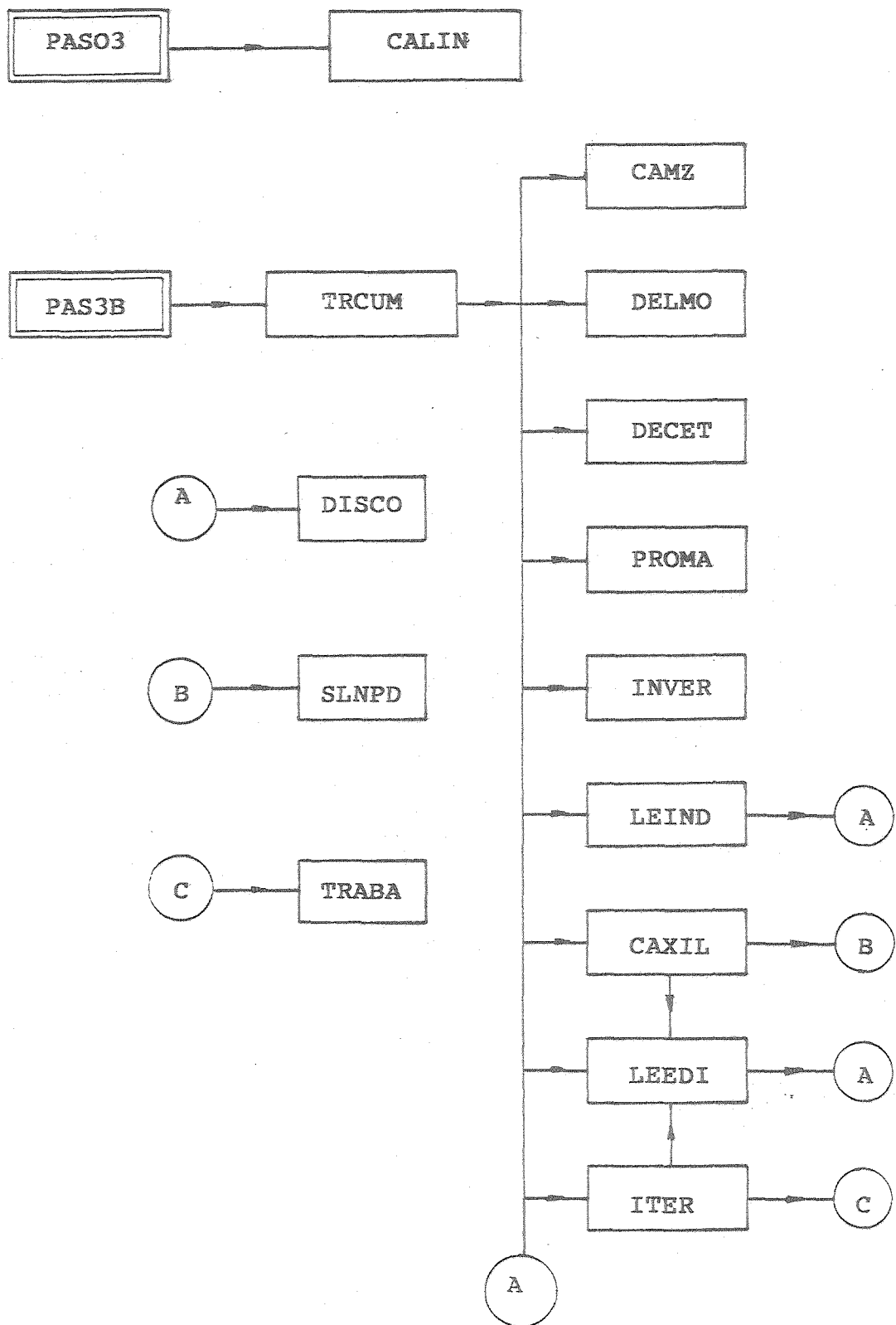


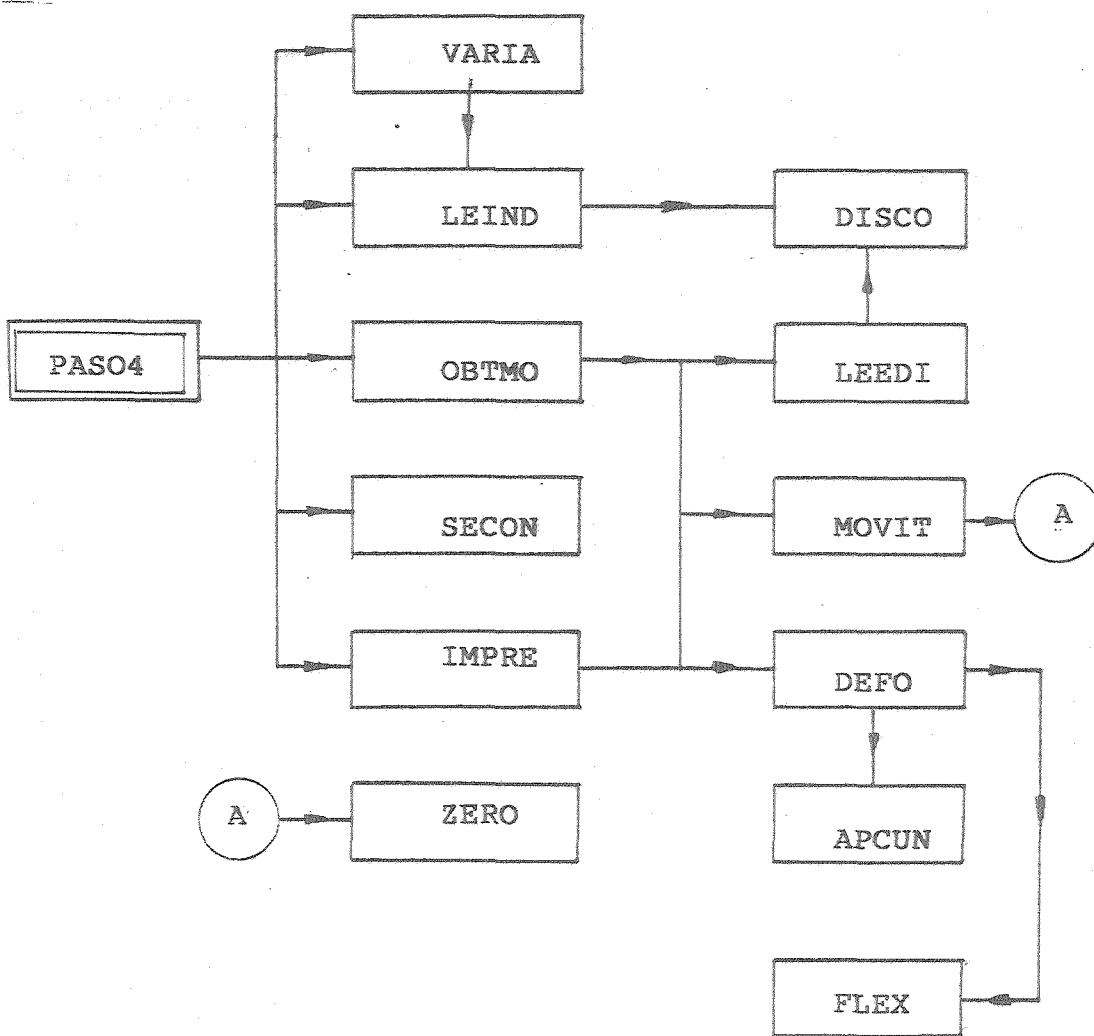
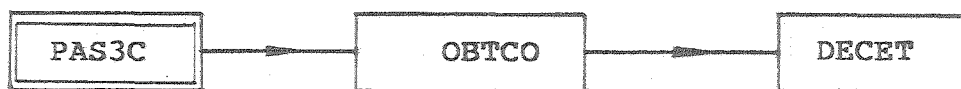


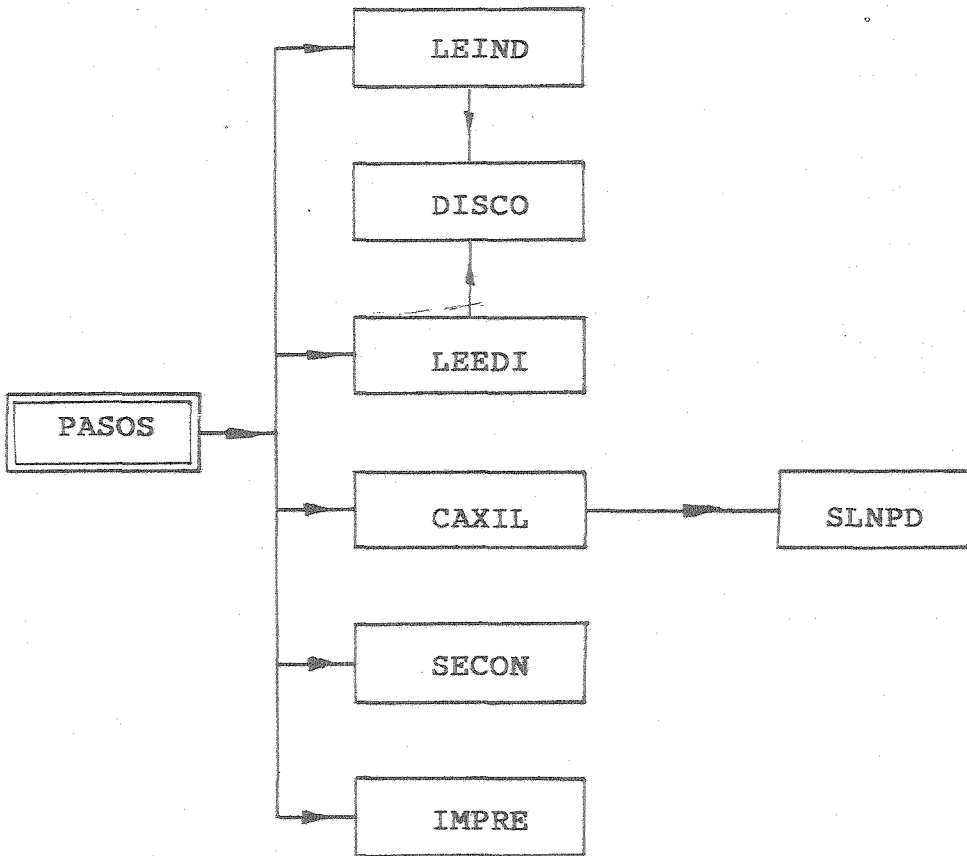




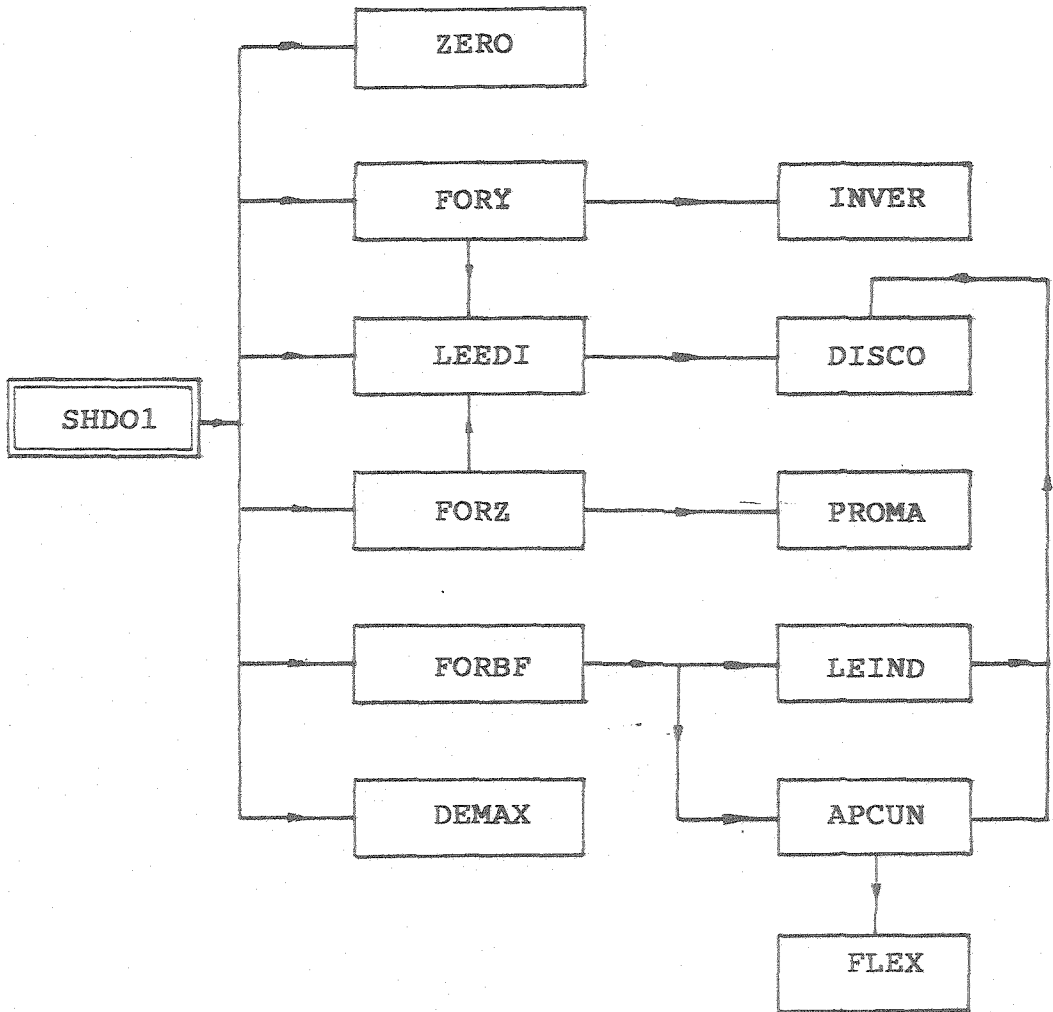


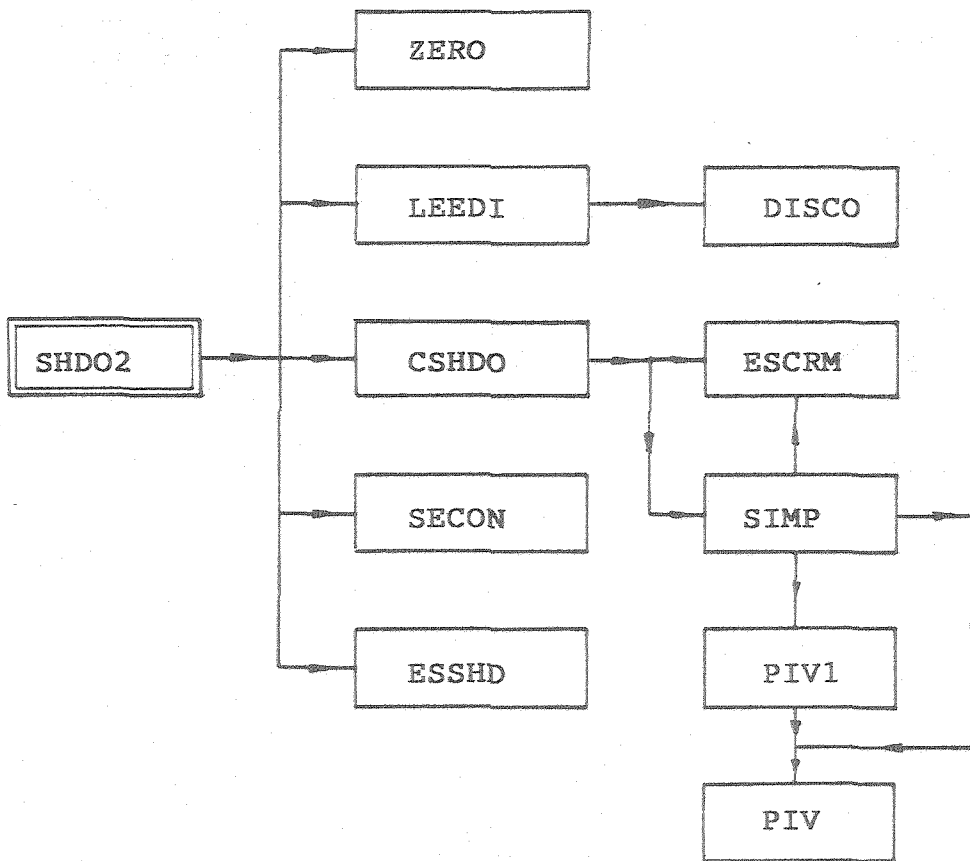


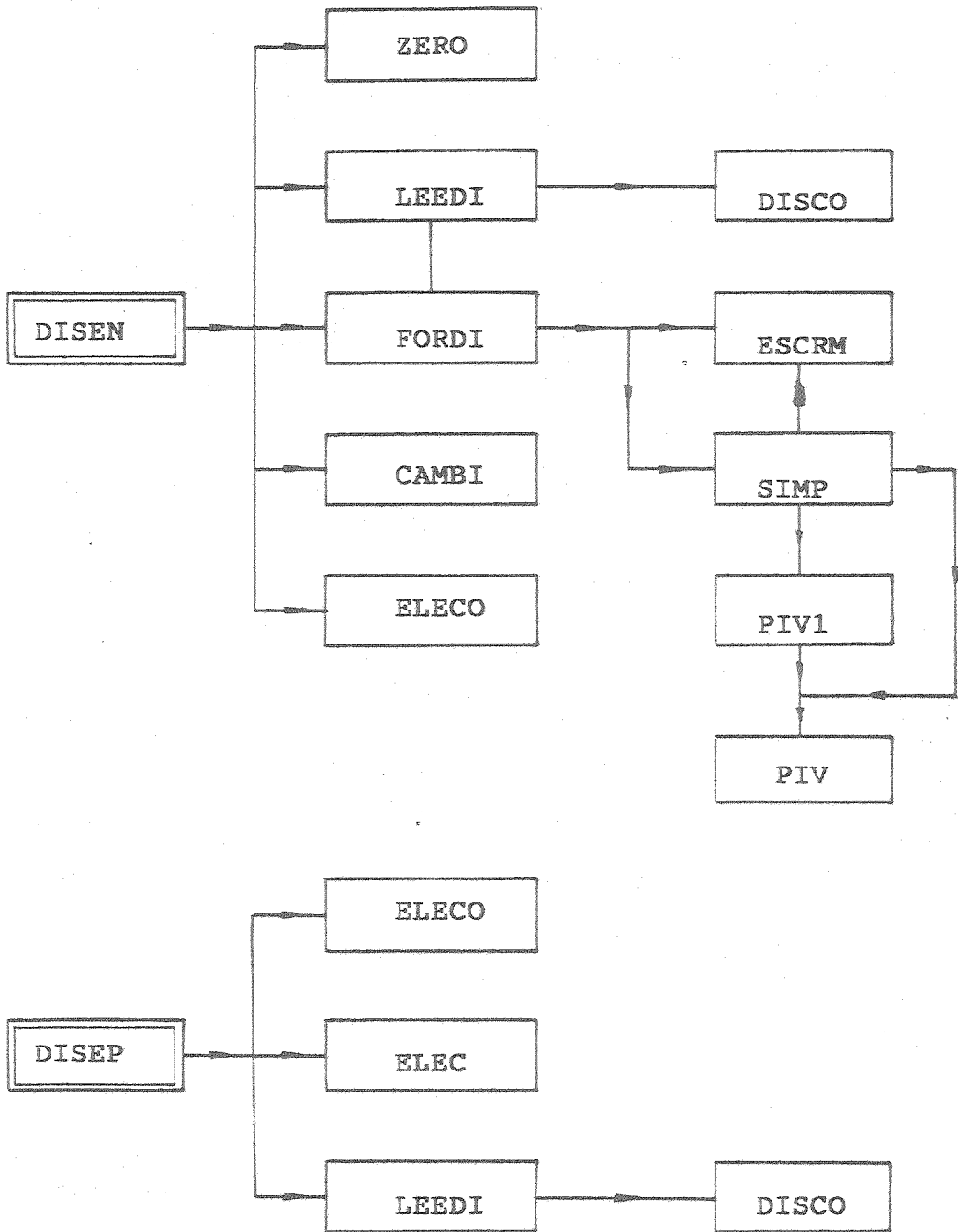


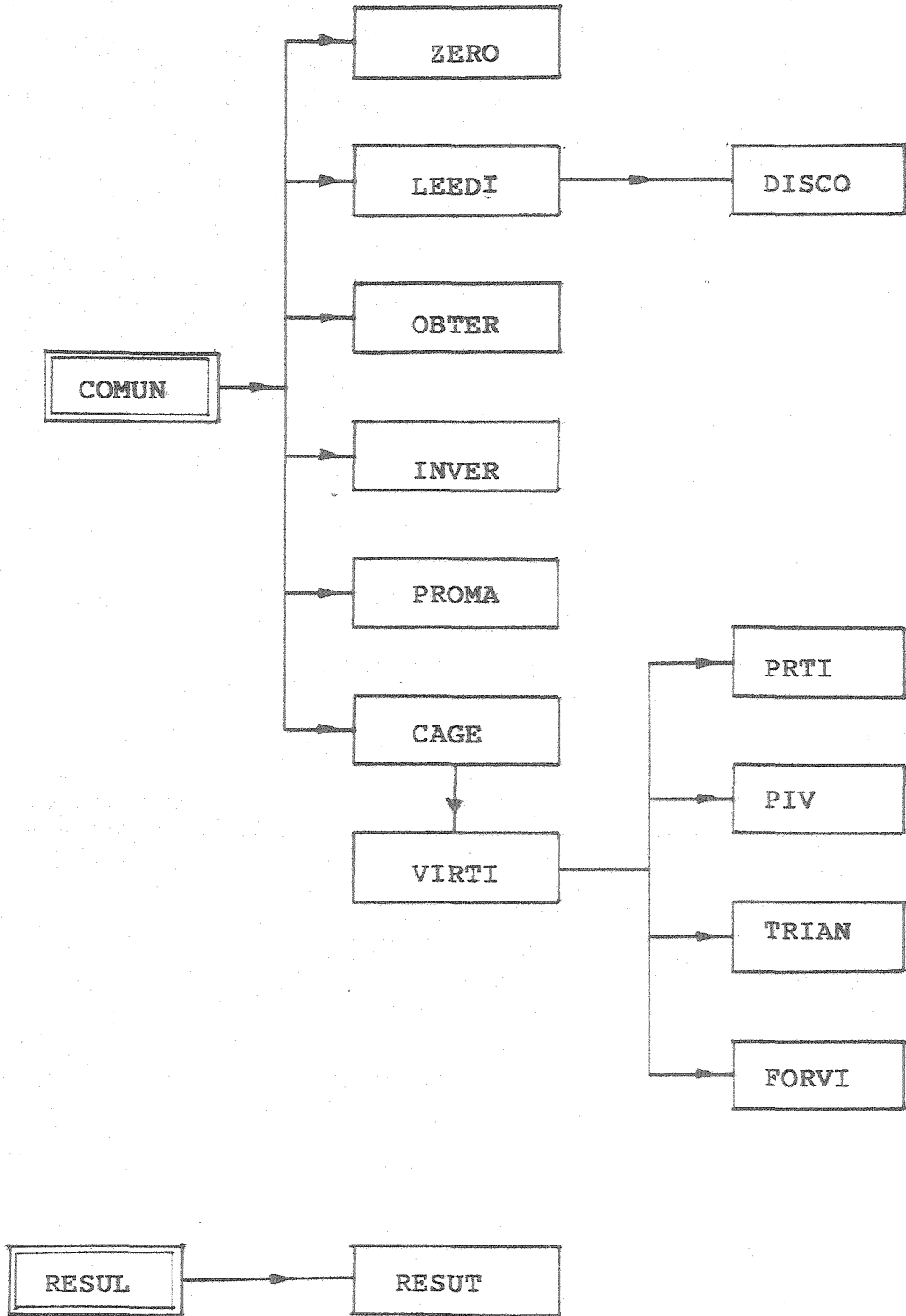


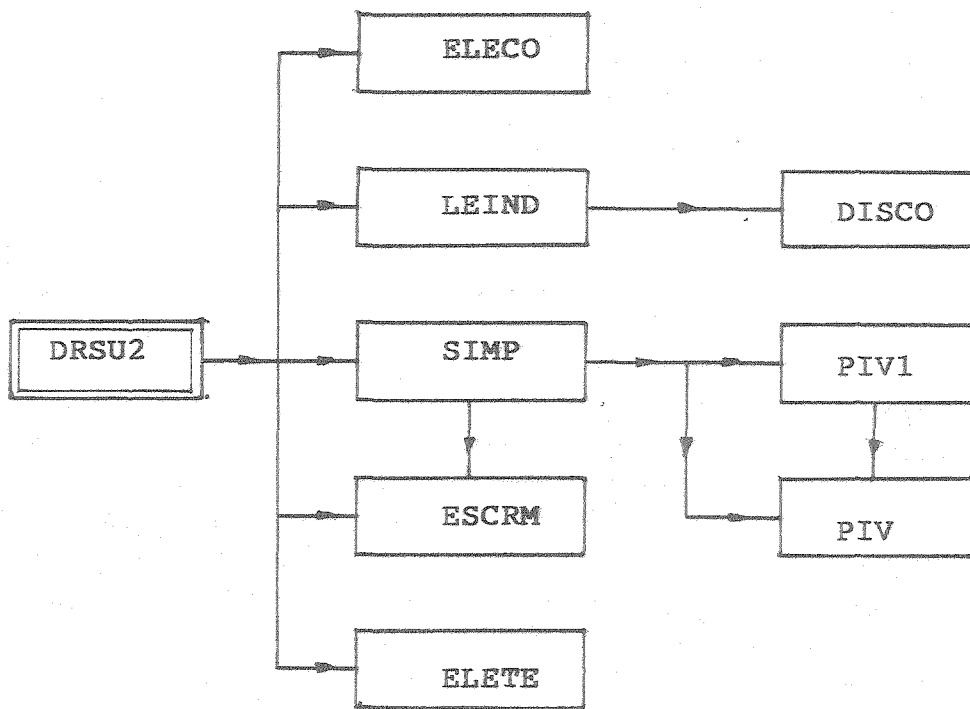
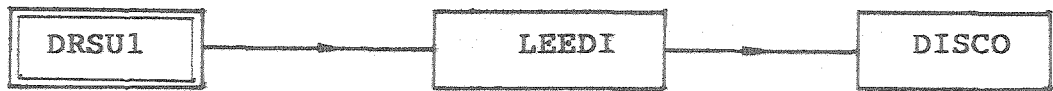


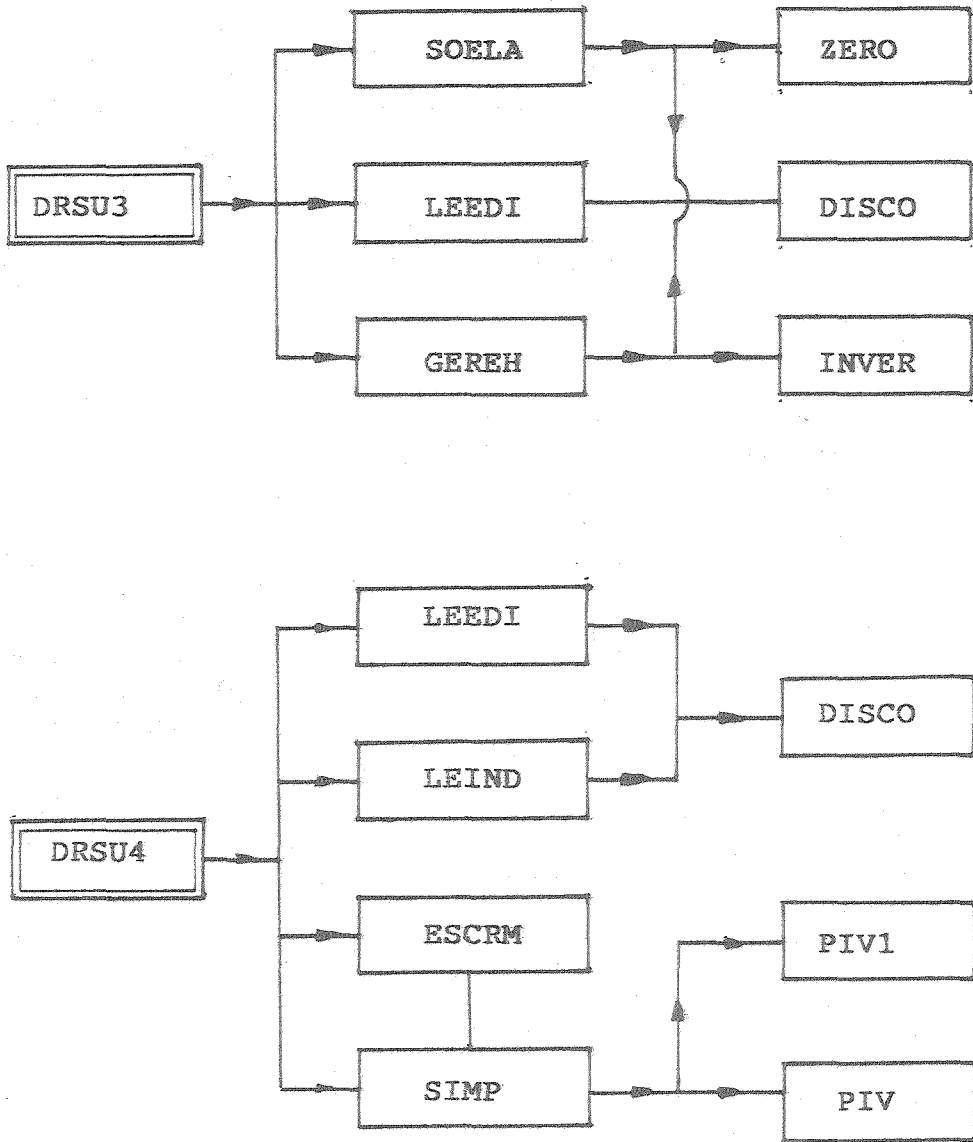


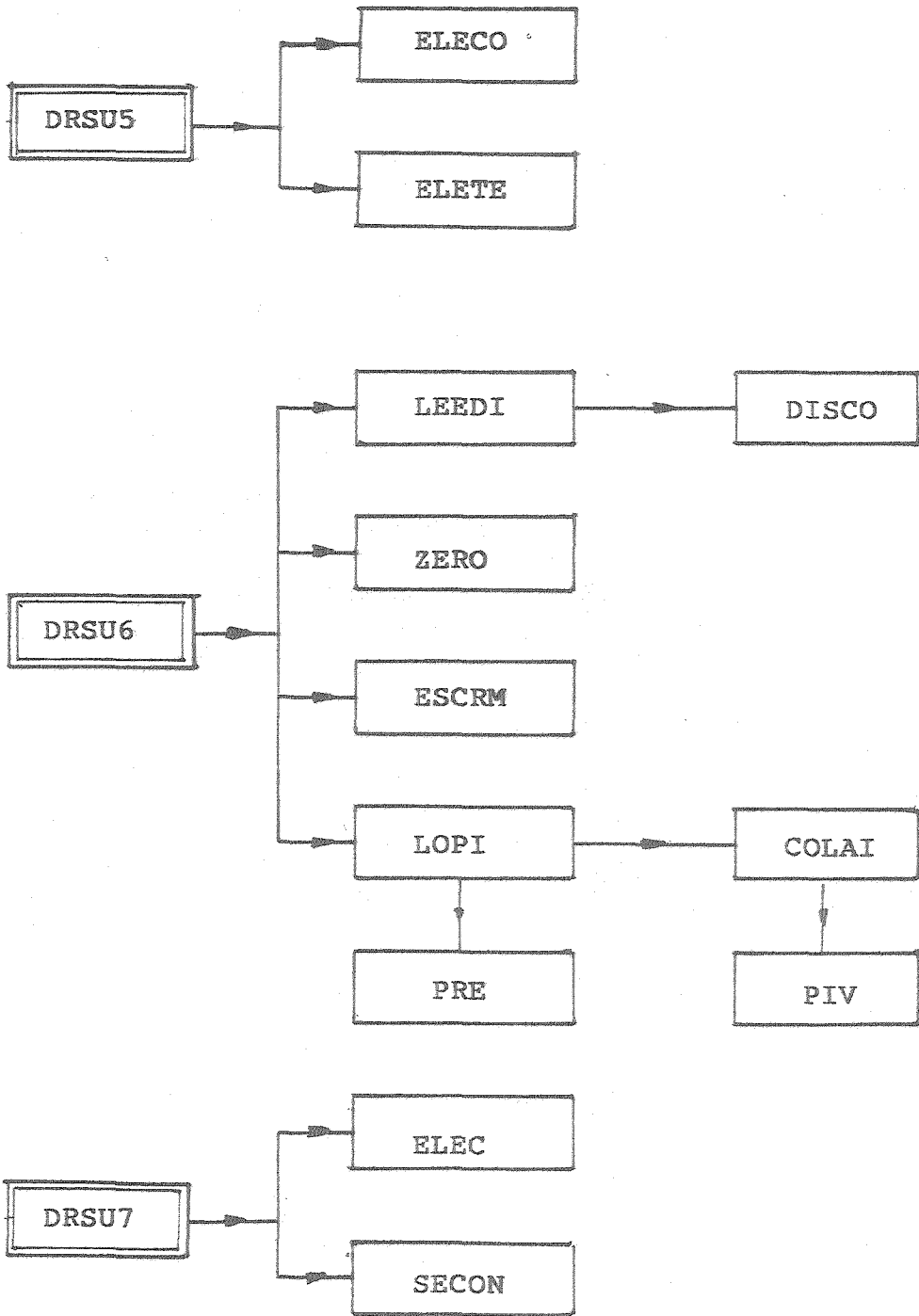












APENDICE 2

"MANUAL DE USUARIO"



## CARACTERISTICAS GENERALES

El programa PLASI permite el análisis y diseño límite de estructuras planas, formadas por barras prismáticas con cargas en su plano.

Se admiten tanto estructuras ortogonales como de barras inclinadas.

En el proceso de cálculo, se ignoran las deformaciones originadas por el esfuerzo Axil y Cortante.

Se toman dos sistemas de coordenadas. El sistema global se define tomando como eje horizontal el eje  $x$ , definido como positivo de izquierda a derecha. El resto del sistema se define de forma que sea dextrógiro. Al sistema global, se referirán las coordenadas de los nudos y los desplazamientos de los mismos.

El sistema local se define con el eje  $x$  local coincidente con el eje de la barra y dirección la de la barra (de nudo origen a extremo). El resto del sistema se definirá de forma que sea dextrógiro. Los resultados de esfuerzos, (momentos flectores y axiles) vendrán referidos a este sistema según el convenio positivo de la Figura 1.

Todos los datos que definan la estructura deberán estar en unidades coherentes. Si el problema objeto de estudio es de diseño, los datos se darán en las siguientes unidades:

- Coordenadas de nudos..... cm.
- Fuerzas (Momentos)..... T o T.cm.
- Módulo de Elasticidad y Limite elástico... T/cm<sup>2</sup>

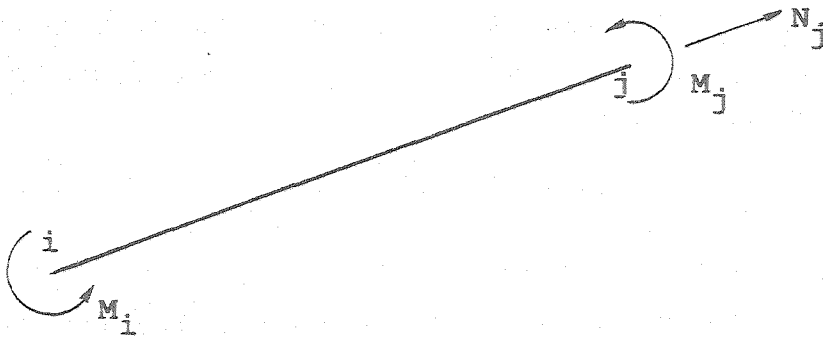


Figura 1. Convenio de signos.

Los resultados que se obtengan vendrán en unidades acordes con los datos de entrada.

#### TIPOS DE PROBLEMAS

Mediante el programa PLASI, pueden ser tratados los siguientes problemas:

##### 1. Análisis Directo 1º Orden.

Permite conocer la carga de colapso, el mecanismo de colapso, la distribución de momentos flectores y los desplazamientos de una estructura en el instante del colapso, bajo las hipótesis de una Teoría de 1º Orden y estado de carga proporcional (cargas puntuales y/o uniformes).

##### 2. Análisis Paso a Paso 1º Orden.

Permite obtener la evolución histórica de la estructura hasta el colapso. Proporciona la carga a la que se va forman-

do las sucesivas rótulas plásticas. Se consideran estados de carga proporcional y Teoría de Primer Orden (cargas puntuales y/o uniformes).

### 3. Análisis Paso a Paso 1º Orden C.N.P.

Igual que el tipo anterior pero con estados de carga no proporcional (cargas puntuales y/o uniformes).

### 4. Análisis de Shakedown 1º Orden.

Realiza el cálculo de la carga de Shakedown, se obtiene además, la envolvente de momentos elásticos máxima y mínima, así como la distribución de momentos residuales. Los estados de cargas son variables, pudiendo además tratarse juntamente cargas fijas proporcionales (cargas puntuales).

### 5. Análisis Paso a Paso 2º Orden.

Proporciona los mismos resultados que el problema 2 pero bajo la Teoría de Segundo Orden.

### 6. Análisis Directo 2º Orden.

Proporciona los mismos resultados que el problema 1 pero bajo la Teoría de Segundo Orden.

### 7. Análisis Paso a Paso 2ª Orden C.N.P.

Proporciona los mismos resultados que el problema 3 pero bajo la Teoría de Segundo Orden.

#### 8. Diseño a Mínimo Peso 1º Orden S.R.

Permite conocer la estructura que es segura frente al colapso. El usuario puede definir el coeficiente de seguridad que considere adecuado (cargas puntuales y/o uniformes). A partir del diseño obtenido, el programa proporciona un diseño práctico y realiza un análisis de ese diseño.

#### 9. Diseño a Mínimo Peso 1º Orden C.R.

Permite diseñar una estructura bajo restricciones de servicio y de colapso. Se consideran los datos de carga referidos al estado de servicio. La seguridad frente al colapso, vendrá definida por el coeficiente de seguridad que el usuario proporcione (cargas puntuales).

#### 10. Diseño a Mínimo Peso Bajo Cargas Variables.

Permite obtener una estructura segura frente a la carga de Shakedown. A partir del diseño obtenido, el programa proporciona un diseño práctico y realiza un análisis de ese diseño. (cargas puntuales)

#### 11. Diseño a Mínimo Peso 2º Orden S.R.

Igual que el problema 8 bajo la Teoría de Segundo Orden.

## DATOS DE ENTRADA.

La lectura de todos los datos de entrada, se realiza en formato libre, excepto la tarjeta de título. Las características del formato libre son:

- a) Los datos pueden ser enteros, reales o en notación exponencial.
- b) Para separar un dato del siguiente basta dejar un espacio en blanco o separarlos por comas.
- c) En el caso de que todos los datos a perforar en una sola tarjeta no quepan en ésta, o que se desee emplear más de una, puede hacerse colocando una barra (/) detrás del último dato, indicando así continuación.

### 1. Tarjeta de Título (30A2)

De la columna 1 a la 60 un comentario alfanumérico que identifique el problema objeto de estudio.

### 2. Tarjeta de Tipo de Problema

Llevará los siguientes datos en este orden:

- a) Tipo de problema (NCL). Un número indicativo del tipo de problema a estudiar.

| <u>NCL</u> | <u>TIPO DE PROBLEMA</u>                                |
|------------|--|
| 10         | ..... ANALISIS DIRECTO 1 ORDEN                         |
| 11         | ..... ANALISIS PASO-PASO 1 ORDEN                       |
| 12         | ..... ANALISIS PASO-PASO 1 ORDEN CARGA NO PROPORCIONAL |
| 13         | ..... ANALISIS DE SHAKE-DOWN 1 ORDEN                   |
| 14         | ..... ANALISIS PASO-PASO 2 ORDEN                       |
| 15         | ..... ANALISIS DIRECTO 2 ORDEN                         |
| 16         | ..... ANALISIS PASO-PASO 2 ORDEN CARGA NO PROPORCIONAL |
| 20         | ..... DISEÑO A MINIMO PESO 1 ORDEN SIN RESTRICCIONES   |
| 21         | ..... DISEÑO A MINIMO PESO 1 ORDEN CON RESTRICCIONES   |
| 22         | ..... DISEÑO A MINIMO PESO 1 ORDEN CARGA VARIABLE      |
| 23         | ..... DISEÑO A MINIMO PESO 2 ORDEN                     |

b) Tres valores ( $N_1$ ,  $N_2$ ,  $N_3$ ) en caso de Análisis o Diseño en Segundo Orden, indicativo de las simplificaciones que se desee realizar:

$N_1 = 1(0)$  ..... No (Si) se considera el efecto secundario del Momento flector.

$N_2 = 1(0)$  ..... No (Si) se considera el efecto secundario del esfuerzo Axil en vigas.

$N_3 = 1(0)$  ..... No (Si) se considera reducción del Momento plástico.

### 3. Tarjeta de Constantes del Diseño (Sólo si $NCL \geq 20$ )

Llevarán los siguientes datos en este orden:

- Grupos de barras. Variables de diseño (NGD)
- Grupos distintos de perfiles (NGP)
- Coefficiente de Seguridad (CADI)

#### 4. Tarjeta de Constantes Generales

Se incluirán los siguientes datos:

- a) Número de Barras (NB)
- b) Número de Nudos (ND)
- c) Número de Soportes (NS)
- d) Número de Libertades (NLI)
- e) Número de Apoyos Inclinados (NAI)
- f) Número de barras con carga uniforme\* (NBCU)
- g) Módulo de Elasticidad (E). Si el problema es de diseño se dará en unas unidades determinadas
- h) Límite elástico (SIGMA)
- i) Número de etapas de carga no proporcional (NECNP)  
Sólo en caso de carga no proporcional.

#### 5. Tarjetas de Nudos

Existirán tantas tarjetas como nudos (ND) se hayan definido, debiendo estar ordenadas en sentido creciente.

Los datos de cada tarjeta serán:

- a) Número del Nudo
- b) Tipo del Nudo (IT)
- c) Coordenada
- d) Coordenada
- e) Angulo del apoyo inclinado (Z). Sólo si el nudo es tipo 5. El ángulo se referirá al sistema global.

Nota: \* Siempre será un número par.

El tipo de nudo lo definirá un número entre 0 y 8 en función de los grados de libertad que en nudo tenga.

| <u>TIPO</u> |       | <u>LIBERTAD (X,Y,Z)</u> |                        |
|-------------|-------|-------------------------|------------------------|
| 0           | ..... | X, Y, Z                 | (Libre) ó (rígido)     |
| 1           | ..... | -, Y, Z                 | (Articulación móvil)   |
| 2           | ..... | X, -, Z                 | (Articulación móvil)   |
| 3           | ..... | -, -, Z                 | (Articulación fija)    |
| 4           | ..... | -, -, -                 | (Empotramiento)        |
| 5           | ..... | X, -, Z                 | (Apoyo inclinado)      |
| 6           | ..... | X, Y, Z                 | (Nudo variable)        |
| 7           | ..... | -, Y, -                 | (Pseudo empotramiento) |
| 8           | ..... | X, -, -                 | (Pseudo empotramiento) |

## 6. Tarjetas de Barras

Existirán tantas como barras, y ordenadas en sentido creciente según el número de éstas.

El formato de las tarjetas de barras será función del tipo de problema.

### 6.1. Análisis (NCL < 20)

En cada tarjeta irá:

- a) Número de la barra
- b) Nudo origen (NI)
- c) Nudo extremo (NJ)
- d) Momento Plástico (MP)
- e) Inercia de la barra (INER)
- f) Area de la barra (AREA). Si el análisis es de primer orden no hace falta incluirla. En caso de que haya-



mos definido  $N_2 = 1$  y la barra sea una viga, se deberá dar un valor del Area mayor que 100000.

## 6.2. Diseño ( $NCL \geq 20$ )

En cada tarjeta irá:

- a) Número de la barra
- b) Nudo Origen (NI)
- c) Nudo extremo (NJ)
- d) Grupo al que pertenece (NGB). Si sólo hay un grupo, ignorar este dato e incluir en su posición el siguiente.
- e) Tipo de Perfil (NTP). Un número del 0 al 3.

| <u>NTP</u> | <u>TIPO DE PERFIL</u> |
|------------|-----------------------|
| 0          | A DECIDIR             |
| 1          | IPN                   |
| 2          | IPE                   |
| 3          | HEB                   |

Tanto en las tarjetas de Análisis como de Diseño las dos barras cargadas que componen la barra cargada deberán tener numeración consecutiva y el nudo variable (Tipo 6) será un nudo extremo para ambas barras.

## 7. Libertades en Barras (Sólo si $NLI \neq 0$ )

Tantas tarjetas como libertades. En cada una irán los siguientes datos:

- a) Número de la libertad
- b) Barra
- e) Nudo
- f) Carga en la libertad (Momento Flector actuante )

## 8. Tarjetas de Datos de Carga

Los datos de cargas serán función del tipo de problema objeto de estudio.

### 8.1. Carga Proporcional (NCL $\neq$ 12, 16, 13, 22)

#### 8.1.1. Tarjetas de Cargas en Nudos

Habrán tantas tarjetas como nudos cargados en cada tarjeta irán los siguientes datos:

- a) Nudo cargado
- b) Fuerza X (D1). Fuerza actuando según el eje X global.
- c) Fuerza Y (D2). Fuerza actuando según el eje Y global
- d) Fuerza Z (D3). Momento actuando según el eje Z global

Como tarjeta final de cargas en nudos se colocará un cero, como número del nudo cargado.

#### 8.1.2. Tarjetas de Cargas Repartidas

Habrán tantas tarjetas, como barras con carga uniforme se hallan definido (NBCU). Sólo se admiten cargas repartidas uniformemente, parcial o totalmente sobre la barra. Cada tarjeta llevará los siguientes datos:

- a) Número de la barra cargada
- b) Tipo de carga
- c) Dato 1 (DA1). Indicando la longitud desde el nudo origen al punto de comienzo de la carga repartida
- d) Dato 2 (DA2). Valor de la carga (en unidades de Fuerza/longitud).

Los tipos de cargas que se pueden considerar son:

#### TIPO

- |   |       |                                      |
|---|-------|--------------------------------------|
| 1 | ..... | Carga actuando según el eje Y global |
| 2 | ..... | Carga actuando según el eje X global |
| 3 | ..... | Carga actuando según el eje x local  |
| 4 | ..... | Carga actuando según el eje y local  |

### 8.2. Carga no Proporcional (NCL = 12 ó 16)

Habrán tantos grupos de tarjetas como los definidos en 8.1.1 y 8.1.2 como etapas no proporcionales de carga (NECNP) se hallan definidos. Antes de cada grupo 8.1.1 irá una tarjeta en la que se incluirá:

- a) Número de la etapa de carga
- b) Valor del factor de carga (COTA).

### 8.3. Cargas Variables (NCL = 13 ó 22)

Habrán inicialmente una tarjeta en la que se incluirá:

- a) Número de nudos con cargas puntuales fijas (NCFP)
- b) Número de nudos con cargas puntuales variables (NCPV)

Si NCFP es igual a cero pasar a 8.3.1, en caso contrario incluir el bloque de tarjetas 8.1.1. Habrá tantas tarjetas como número indique NCFP.

### 8.3.1. Datos de Cargas Variables

Habrá tantas tarjetas como número indique NCPV. En cada tarjeta irá:

- a) Nudo cargado
- b) Fuerza X según el eje X global
- c) Fuerza Y según el eje Y global
- d) Momento Z según el eje Z global
- e) Límite superior de la Fuerza X
- f) Límite inferior de la Fuerza X
- g) Límite superior de la Fuerza Y
- h) Límite inferior de la Fuerza Y
- i) Límite superior del Momento Z
- j) Límite inferior del Momento Z

### 9. TARJETAS DE LIMITACION DE DESPLAZAMIENTOS

Sólo en el caso de diseño bajo restricciones de servicio (NCL = 21). En cada tarjetá irá:

- a) Nudo limitado
- b) Cota de desplazamiento horizontal (X)
- c) Cota de desplazamiento vertical (Y).

Como tarjeta final, se perfora como nudo limitado un cero.

Las cotas en desplazamientos, se darán en valor absoluto.

**RESTRICCIONES**

Las restricciones en cuanto a las dimensiones de problema a tratar dependerán en general del tipo de estudio que se quiera realizar, y de la capacidad del ordenador. Para un mini ordenador MP-21Mx y con 28 Kbytes de memoria central las restricciones son:

- Máximo número de barras (NB)..... 25
- Máximo número de nudos (ND)..... 25
- Máximo número de apoyos inclinados..... 5
- Máximo número de cargas variables..... 10
- Máximo número de etapas de carga (NECNP).. 10

Además de las generales, según el tipo de problema:

**Análisis Shakedown**

- $2(2 \cdot NB + NMI) + 1$  ..... 47
- $2(4 \cdot NB + NMI) + 1$  ..... 147

NMI = Número de mecanismos independientes.

**Diseño sin restricciones de servicio**

- $NMI + 2 \cdot NB + 2 \cdot NGD + 1$  ..... 62
- $4 \cdot NB + 2 \cdot NGD$  ..... 101

NGD = Grupos de barras de características diferentes.

**Diseño con restricciones de servicio**

Será función del número de restricciones activas. Como mínimo

2(NRS+NGD)+1..... <= 90

2(NRS+NGD)+NGD..... <= 92

NRS = Número de restricciones de desplazamientos.

### PARADAS DEL PROGRAMA

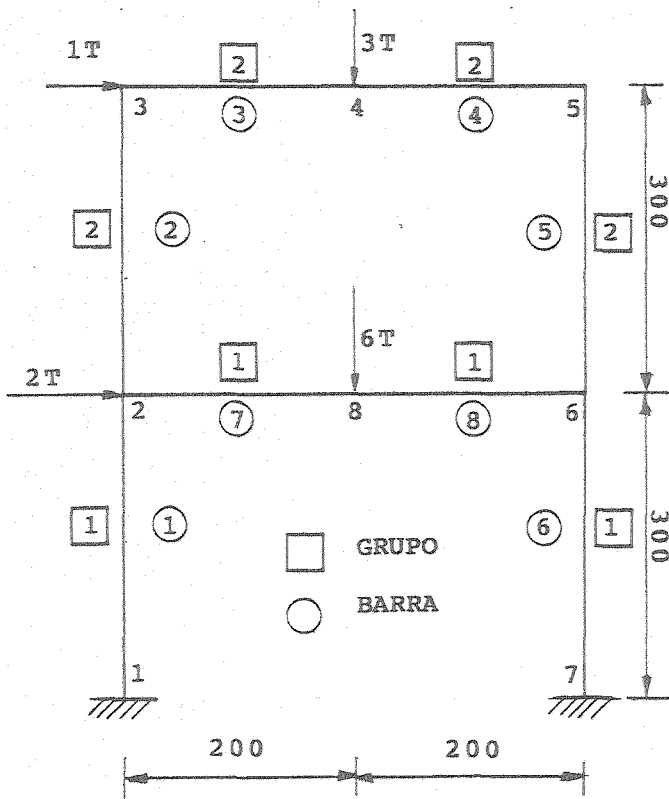
El programa finaliza cuando por la consola principal aparece STOP 0000; sin embargo, el programa puede finalizar por errores de diversa índole.

Si se produce una finalización por algún error se imprimirá un mensaje de STOP xxxx. El significado de cada finalización del programa es la siguiente:

| <u>STOP</u> | <u>DESCRIPCION</u>                  |
|-------------|-------------------------------------|
| 0000        | Problema terminado                  |
| 1111        | Dimensiones no admisibles           |
| 2222        | Problema no conocido                |
| 3330        | No existe pivote $\neq$ 0 en S.V.A. |
| 3331        | Pivote = 0 en el S.V.A.             |
| 3332        | No existe solución en el Simplex    |
| 4440        | Error IN/IO en la rutina DISCO      |
| 4441        | Error IN/IO en la rutina LEEDI      |
| 4442        | Error IN/IO en la rutina LEIND      |
| 5550        | Matriz Singular (INVER)             |
| 5551        | Matriz Singular (SLNPD)             |

CODIFICACION Y SALIDA DE RESULTADOS

A modo de ejemplo, se realiza la codificación y se presentan los resultados obtenidos para el caso de diseño limite en teoría de Primer Orden del pórtico de la Figura:



DATOS:

$$\lambda_{sg} = 1$$

$$E = 2110.09 \text{ T/cm}^2$$

$$\sigma_E = 2.4 \text{ T/cm}^2$$

| <u>barras</u> | <u>GRUPO</u> | <u>TIPO</u> |
|---------------|--------------|-------------|
| 1, 6, 7, 8    | 1            | IPE         |
| 2, 3, 4, 5    | 2            | IPE         |

CODIFICACION

EJEMPLO D 1 1 a

20

|   |   |     |     |   |                     |
|---|---|-----|-----|---|---------------------|
| 2 | 1 | 1   |     |   |                     |
| 8 | 8 | 2   | 0   | 0 | 0 2110.0917 2.40061 |
| 1 | 4 |     |     |   |                     |
| 2 | 0 | 0   | 300 |   |                     |
| 3 | 0 | 0   | 600 |   |                     |
| 4 | 0 | 200 | 600 |   |                     |
| 5 | 0 | 400 | 600 |   |                     |
| 6 | 0 | 400 | 300 |   |                     |
| 7 | 4 | 400 | 0   |   |                     |
| 8 | 0 | 200 | 300 |   |                     |
| 1 | 1 | 2   | 1   | 2 |                     |
| 2 | 2 | 3   | 2   | 2 |                     |
| 3 | 3 | 4   | 2   | 2 |                     |
| 4 | 4 | 5   | 2   | 2 |                     |
| 5 | 5 | 6   | 2   | 2 |                     |
| 6 | 6 | 7   | 1   | 2 |                     |
| 7 | 2 | 8   | 1   | 2 |                     |
| 8 | 8 | 6   | 1   | 2 |                     |
| 2 | 2 |     |     |   |                     |
| 3 | 1 |     |     |   |                     |
| 4 | 0 | -3  |     |   |                     |
| 8 | 0 | -6  |     |   |                     |
| 0 |   |     |     |   |                     |



EJEMPLO D-1-1-a

\*\*\*\*\*

CONSTANTES DE LA ESTRUCTURA

---

NUMERO DE BARRAS 8  
 NUMERO DE NUDOS 8  
 NUMERO DE SOPORTES 2  
 NUMERO DE LIBERTADES 0  
 NUMERO DE APOYOS INCLINADOS 0  
 NUMERO DE BARRAS CON CARG. UNIF. 0

MODULO DE ELASTICIDAD .211009E+04

TIPO DE PROBLEMA : DISEÑO A MINIMO PESO  
 TENSION DE FLUENCIA : .240061E+01 T/CM^2  
 CARGA DE DISEÑO : .100000E+01  
 GRUPOS DISTINTO DE BARRAS : 2

HIPOTESIS DE CALCULO :

1.- NO SE CONSIDERA DEFORMACION DEBIDA AL AXIL NI AL CORTANTE

DATOS DE NUDOS

---

| NUDO | TIPO | COORD. X | COORD. Y | ANG. APOYO INC. |
|------|------|----------|----------|-----------------|
| 1    | 4    | 0.000    | 0.000    | 0.000           |
| 2    | 0    | 0.000    | 300.000  | 0.000           |
| 3    | 0    | 0.000    | 600.000  | 0.000           |
| 4    | 0    | 200.000  | 600.000  | 0.000           |
| 5    | 0    | 400.000  | 600.000  | 0.000           |
| 6    | 0    | 400.000  | 300.000  | 0.000           |
| 7    | 4    | 400.000  | 0.000    | 0.000           |
| 8    | 0    | 200.000  | 300.000  | 0.000           |

TABLA DE CONECTIVIDAD Y PROP. GEOMETRICAS DE BARRAS

---

| BARRA | NUDO1 | NUDO2 | LONGITUD | ANGULO   |
|-------|-------|-------|----------|----------|
| 1     | 1     | 2     | 300.0000 | 90.0000  |
| 2     | 2     | 3     | 300.0000 | 90.0000  |
| 3     | 3     | 4     | 200.0000 | 0.0000   |
| 4     | 4     | 5     | 200.0000 | 0.0000   |
| 5     | 5     | 6     | 300.0000 | -90.0000 |
| 6     | 6     | 7     | 300.0000 | -90.0000 |
| 7     | 2     | 8     | 200.0000 | 0.0000   |
| 8     | 8     | 6     | 200.0000 | 0.0000   |

---

TABLA CON LAS DIFERENTES CLASES DE BARRAS

---

| BARRA | GRUPO | TIPO PERFIL |
|-------|-------|-------------|
| 1     | 1     | IPE         |
| 2     | 2     | IPE         |
| 3     | 2     | IPE         |
| 4     | 2     | IPE         |
| 5     | 2     | IPE         |
| 6     | 1     | IPE         |
| 7     | 1     | IPE         |
| 8     | 1     | IPE         |

---

DATOS DE CARGAS EN NUDOS

---

| NUDO | FUERZA X | FUERZA Y | MOME Z |
|------|----------|----------|--------|
| 2    | 2.000    | 0.000    | 0.000  |
| 3    | 1.000    | 0.000    | 0.000  |
| 4    | 0.000    | -3.000   | 0.000  |
| 8    | 0.000    | -6.000   | 0.000  |

## EJEMPLO D-1-1-a

\*\*\*\*\*

CARGA DE COLAPSO : .13392E+01

PESO DE LA ESTRUCTURA : .35300E+03 KG.

## DISTRIBUCION DE MOMENTOS Y SOLUCION DE DISEÑO :

| BARRA | NUDO | MOMENTO     | M.PLAST E  | M.PLAST P  | TIPO-PERFIL |
|-------|------|-------------|------------|------------|-------------|
| 1     | 1    | .52813E+03  | .40000E+03 | .52813E+03 | IPE-200     |
|       | 2    | -.16689E+03 |            |            |             |
| 2     | 2    | .14423E+03  | .15000E+03 | .21221E+03 | IPE-140     |
|       | 3    | -.16689E+03 |            |            |             |
| 3     | 3    | .16689E+03  | .15000E+03 | .21221E+03 | IPE-140     |
|       | 4    | .21221E+03  |            |            |             |
| 4     | 4    | -.21221E+03 | .15000E+03 | .21221E+03 | IPE-140     |
|       | 5    | -.21221E+03 |            |            |             |
| 5     | 5    | .21221E+03  | .15000E+03 | .21221E+03 | IPE-140     |
|       | 6    | .21221E+03  |            |            |             |
| 6     | 6    | .31592E+03  | .40000E+03 | .52813E+03 | IPE-200     |
|       | 7    | .52813E+03  |            |            |             |
| 7     | 2    | .22662E+02  | .40000E+03 | .52813E+03 | IPE-200     |
|       | 8    | .52813E+03  |            |            |             |
| 8     | 8    | -.52813E+03 | .40000E+03 | .52813E+03 | IPE-200     |
|       | 6    | -.52813E+03 |            |            |             |

## MOVIMIENTOS DE LOS NUDOS EN EL INSTANTE DEL COLAPSO :

| NUDO | MOV. X   | MOV. Y    | GIRO    |
|------|----------|-----------|---------|
| 1    | 0.00000  | 0.00000   | 0.00000 |
| 2    | 4.48199  | 0.00000   | -.02547 |
| 3    | 18.10552 | 0.00000   | -.06635 |
| 4    | 18.10552 | -13.97964 | .06370  |
| 5    | 18.10552 | 0.00000   | .06370  |
| 6    | 4.48199  | 0.00000   | -.01367 |
| 7    | 0.00000  | 0.00000   | 0.00000 |
| 8    | 4.48199  | -4.30723  | -.01312 |

## ROTACIONES PLASTICAS :

---

| BARRA | NUDO | ROTACION    |
|-------|------|-------------|
| 1     | 1    | .00000E+00  |
|       | 2    | .00000E+00  |
| 2     | 2    | .00000E+00  |
|       | 3    | .00000E+00  |
| 3     | 3    | .00000E+00  |
|       | 4    | .12608E+00  |
| 4     | 4    | .00000E+00  |
|       | 5    | .00000E+00  |
| 5     | 5    | .99818E-01  |
|       | 6    | .22444E-01  |
| 6     | 6    | .00000E+00  |
|       | 7    | .58971E-02  |
| 7     | 2    | .00000E+00  |
|       | 8    | .00000E+00  |
| 8     | 8    | -.30355E-01 |
|       | 6    | -.30909E-01 |

\*\*\* TIEMPO C.P.U EMPLEADO..... 14.560 SG.

APENDICE 3

"LISTADO DE ORDENADOR"

```
2 C
3 C DEFINICION DEL AREA COMMON
4 C
5 BLOCK DATA DOW
6 COMMON /VAR/ NVAR(89)
7 COMMON /CON/ NCON(15000)
8 COMMON /PAS/ NPAS(630)
9 COMMON /PAS2/ NPAS2(850)
10 END
```

FTN4X COMPILER: HP92834 REV.2130 (810716)

\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: (NONE) COMMON: (NONE)

```
BLOCK COMMON PAS2 SIZE: 850
BLOCK COMMON PAS SIZE: 630
BLOCK COMMON CON SIZE: 15000
BLOCK COMMON VAR SIZE: 89
```

```
11 PROGRAM PLASI
12 C
13 C
14 C *****
15 C *
16 C * PROGRAMA PARA EL ANALISIS Y DISEÑO ELASTOPLASTICO *
17 C * DE ESTRUCTURAS METALICAS PLANAS FORMADAS POR BARRAS *
18 C * PRISMATICAS. *
19 C *
20 C *****
21 C
22 C
23 C CATEDRA DE ELASTICIDAD Y RESISTENCIA DE MATERIALES.
24 C
25 C ESCUELA TECNICA SUPERIOR DE INGENIEROS INDUSTRIALES. SEVILLA
26 C
27 C
28 C
29 C SEVILLA, NOVIEMBRE 1.985
30 C
31 C
32 C
33 C
34 C PROGRAMA PRINCIPAL RESIDENTE EN MEMORIA
35 C
36 DIMENSION NAMI(3)
37 C
38 C LLAMADA AL SEGMENTO DE LECTURA DE DATOS
39 C
40 NAMI(1)=2HLE
41 NAMI(2)=2HDA
42 NAMI(3)=2HT
43 CALL EXEC(8,NAMI)
44 END
```

FTN4X COMPILER: HP92834 REV.2130 (810716)

\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: 28 COMMON: (NONE)

```
45 FUNCTION TOLE(F,TOL)
46 C
47 C FUNCION PARA ELIMINAR ERRORES DE TOLERANCIA
48 C
49 IF(ABS(F).LT.TOL) F=0.0
50 TOLE=F
51 RETURN
52 END
```

FTN4X COMPILER: HP92834 REV.2130 (810716)

\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: 28 COMMON: (NONE)



```

53 PROGRAM LEDAT(5)
54 C
55 C SEGMENTO PARA LECTURA Y ESCRITURA DE DATOS DE LA ESTRUCTURA
56 C
57 DIMENSION IP(5)
58 COMMON /VAR/ IN, IO, NOMB(30), ND, NB, INDT, NMI, NLI, NGL, NAI, NBCU, NMH,
59 @ NFI, NGD, NGP, NCL, LHF, LHV, NITER, NND, NH, NBB, NTCPV, NCFP,
60 @ NCPV, NMIN, NGH, NMCO, NMHCO, NECO, NTRA, LV, CADI, SUMF, E,
61 @ SUM, TOL, TOLS, SUM1, PESO, SIGMA, COP, COL, COI, COM, TCO
62 COMMON /CON/ IT(25), RMP(25), RIMON(50), RMOE(50), NI(25), NJ(25),
63 @ ALFA(25), RLON(25), RINER(25), RMP1(25), AXIL(25), XA(25),
64 @ YA(25), ZA(25), DCL(20), R(75), LBE(20), T(50), LICOL(50),
65 @ NCOL(50), NFILA(50), ICOL(50), DCBU(100), T1(50), Q(50),
66 @ NGB(25), NTP(25), LTP(25), AREA(25), TT(50), CFI(50),
67 @ CMH(50), CMV(50), FIFI(25), X(25), Y(25), Z(25),
68 @ RLCPV(10,2), TETA(50,50), RMAT(50,50), TCNP(50,10)
69 COMMON /PAS/ N1, N2, N3, N4, COLAA, COLAP, RMINC(50), RFIC(50), MC(50),
70 @ RLOF(25), NECNP, NENP, COTA(10)
71 COMMON /PAS2/ NN1, NN2, NN3, NN4
72 C
73 C
74 C SIGNIFICADO DE VARIABLES
75 C *****
76 C
77 C
78 C IN.....UNIDAD LOGICA DE ENTRADA DE DATOS
79 C IO.....UNIDAD LOGICA DE SALIDA DE RESULTADOS
80 C NOMB.....CONJUNTO ALFANUMERICO CON EL TITULO DEL PROBLEMA
81 C ND.....NUMERO DE NUDOS
82 C NB.....NUMERO DE NUDOS
83 C INDT.....NUMERO DE MECANISMOS TRASLACIONALES
84 C NMI.....NUMERO DE MECANISMOS INDEPENDIENTES
85 C NLI.....NUMERO DE LIBERTADES
86 C NRM.....NUMERO DE RESTRICCIONES EN MOVIMIENTO
87 C NGL.....NUMERO DE GRADOS DE LIBERTAD
88 C NAI.....NUMERO DE APOYOS INCLINADOS
89 C NBCU.....NUMERO DE BARRAS CON CARGA UNIFORME
90 C NMH.....NUMERO DE HIPERESTATICAS
91 C NFI.....PARAMETRO DE FIN DE ITERACIONES DE C.U
92 C NGD.....NUMERO DE GRUPOS DE BARRAS CON IDENTICAS CARCT.
93 C NGP.....NUMERO DE GRUPOS DE PERFILES
94 C NCL.....TIPOS DE PROBLEMAS
95 C NITER.....NUMERO DE ITERACIONES CON CARGA UNIFORME
96 C NTCPV.....NUMERO DE CARGAS VARIABLES
97 C NCFP.....NUMERO DE NUDOS CON CARGA FIJA
98 C NCPV.....NUMERO DE NUDOS CON CARGAS VARIABLES
99 C NMCO.....NUMERO DE MOMENTOS EN LA COTA
100 C NECO.....NUMERO DE ECUACIONES DE COLAPSO
101 C NTRA.....NUMERO TOTAL DE ROTULAS PASIVAS
102 C CADI.....CARGA DE DISEÑO
103 C E.....MODULO DE ELASTICIDAD
104 C SUM.....CARGA DE COLAPSO
105 C TOL.....TOLERANCIA DE LA OPERACION
106 C TOLS.....TOLERANCIA DE COMPARACION
107 C PESO.....PESO DE LA ESTRUCTURA

```

108 C SIGMA.....TENSION DE PLASTIFICACION  
 109 C COP.....VALOR MAXIMO DE CARGA APLICADA PARA ESCALADO  
 110 C COL.....VALOR MINIMO DE LONGITUD DE BARRA PARA ESCALADO  
 111 C COI.....VALOR MINIMO DE INERCIA DE BARRA PARA ESCALADO  
 112 C COM.....VALOR MINIMO DE MOMENTO PLASTICO PARA ESCALADO  
 113 C TCO.....TIEMPO DE COMIENZO DEL PROBLEMA  
 114 C COLAA.....CARGA ACUMULADA AL FINAL DE UNA ETAPA DE CARGA  
 115 C NO PROPORCIONAL  
 116 C COLAP.....INCREMENTO DE CARGA DE UNA ETAPA  
 117 C NECNF.....NUMERO DE ETAPAS DE CARGA NO PROPORCIONAL  
 118 C IT.....TIPOS DE NUDOS:

|       | TIPO | LIBERTAD (X,Y,Z)   |
|-------|------|--------------------|
|       | ---- | -----              |
| 120 C |      |                    |
| 121 C |      |                    |
| 122 C |      |                    |
| 123 C | 0    | X,Y,Z              |
| 124 C | 1    | -,Y,Z              |
| 125 C | 2    | X,-,Z              |
| 126 C | 3    | -, -,Z             |
| 127 C | 4    | -, -, -            |
| 128 C | 5    | x,-, - (INCLINADO) |
| 129 C | 6    | X,Y,Z (MOVIL)      |
| 130 C | 7    | -,Y,-              |
| 131 C | 8    | X,-, -             |

132 C

133 C RMP.....MOMENTO PLASTICO  
 134 C NI.....NUDO ORIGEN  
 135 C NJ.....NUDO EXTREMO  
 136 C ALFA.....ANGULO QUE FORMA LA BARRA CON LOS GLOBALES  
 137 C RLON.....LONGITUD DE LA BARRA  
 138 C RINER.....INERCIA DE LA BARRA  
 139 C LBE.....LIBERTADES DE BARRAS  
 140 C DCBU.....DATOS DE CARGAS EN BARRAS UNIFORMES  
 141 C CMH.....COTA MOVIMIENTO HORIZONTAL  
 142 C CMV.....COTA MOVIMIENTO VERTICAL  
 143 C RMINC.....INCREMENTO DE MOMENTO EN UNA ETAPA  
 144 C RFIC.....INCREMENTO DE ROTACION PASTICA EN UNA ETAPA  
 145 C TCNP.....TRABAJO DE CADA MECANISMO EN CADA ETAPA DE  
 146 C CARGA NO PROPORCIONAL  
 147 C FIFI.....TRABAJO EXTERNO DE CADA ECUACION ASOCIADA A  
 148 C UN ALARGAMIENTO UNIDAD DE UNA BARRA  
 149 C MC.....CONJUNTO EN EL QUE SE DENOTAN LAS ROTULAS  
 150 C FORMADAS EN UN ANALISIS PASO A PASO  
 151 C AREA.....SECCION DE LA BARRA  
 152 C AXIL.....ESFUERZO AXIL EN BARRAS  
 153 C RLCPV.....LIMITES DE CARGAS VARIABLES  
 154 C COTA.....COTAS DE LAS ETAPAS DE CARGA NO PROPORCIONAL  
 155 C NEC.....NUMERO DE ECUACIONES DE COLAPSO  
 156 C NRM.....NUMERO DE RESTRICCIONES EN DESPLAZAMIENTOS  
 157 C NRS.....NUMERO DE MOMENTOS DE SERVICIOS ACTIVOS

158 C

159 C

160 C TIPOS DE PROBLEMAS POSIBLES  
 161 C \*\*\*\*\*  
 162 C

163 C 10.....ANALISIS DIRECTO 1 ORDEN  
 164 C 11.....ANALISIS PASO-PASO 1 ORDEN  
 165 C 12.....ANALISIS PASO-PASO 1 ORDEN CARGA NO PROPORCIONAL  
 166 C 13.....ANALISIS DE SHAKE-DOWN 1 ORDEN  
 167 C 14.....ANALISIS PASO-PASO 2 ORDEN  
 168 C 15.....ANALISIS DIRECTO 2 ORDEN  
 169 C 16.....ANALISIS PASO-PASO 2 ORDEN CARGA NO PROPORCIONAL  
 170 C 20.....DISE#0 A MINIMO PESO 1 ORDEN SIN RESTRICCIONES.  
 171 C 21.....DISE#0 A MINIMO PESO 1 ORDEN CON RESTRICCIONES.  
 172 C 22.....DISE#0 A MINIMO PESO 1 ORDEN CARGA VARIABLE.  
 173 C 23.....DISE#0 A MINIMO PESO 2 ORDEN

174 C

175 C

176 C DIMENSIONES MAXIMAS.

177 C \*\*\*\*\*

178 C

179 C

180 C MAXIMO NUMERO DE BARRAS.....25

181 C MAXIMO NUMERO DE NUDOS.....25

182 C MAXIMO NUMERO DE APOYOS INCL.....5

183 C MAXIMO NUMERO DE CARGAS VAR.....10

184 C MAXIMO NUMERO DE ETAPAS DE C.N.P.10

185 C DIMENSIONES MAXIMAS DE LOS PROBLEMAS:

186 C

187 C

188 C

189 C

190 C

191 C

192 C

193 C

194 C

195 C

196 C

197 C

198 C

199 C

200 C

201 C

202 C

203 C

204 C

205 C

206 C

207 C

208 C

209 C

210 C

211 C

212 C

213 C

214 C

215 C

216 C

217 C

| Tipo        | Dimension | Cota     |
|-------------|-----------|----------|
| ----        | -----     | -----    |
| Analisis    | (L0,L0)   | (50,50)  |
| Anal. S-D   | (L1,L2)   | (47,147) |
| Dise#o      | (L3,L4)   | (62,101) |
| Dise#o(23)  | (L5,L6)   | (62,101) |
| Dise#o (21) | (L7,L8)   | (90,92)  |

L0..... 2\*N. DE BARRAS  
 L1.....  $2*(L0-NMI)+1$   
 L2.....  $4*L0-2*NMI+1$   
 L3.....  $NMI+L0+1$   
 L4.....  $2*L0+NGD$   
 L5.....  $L3+NGD$   
 L6.....  $2*L0+2*NGD$   
 L7.....  $NEC+NRS+2*NRM+2*NGD$   
 L8.....  $NEC+2*NRM+NGD+2*NGD$

210 C UNIDADES LOGICAS UTILIZADAS EN FICHEROS

211 C \*\*\*\*\*

212 C

213 C

214 C

215 C

216 C

217 C

51.....MATRIZ DE MECANISMO (Ut)  
 52.....MATRIZ  $D^*$   
 53.....MATRIZ J  
 54.....MATRIZ H  
 55.....MATRIZ DE EQUILIBRIO  
 56.....MATRIZ  $BH^* * Y$   
 57.....MATRIZ Y  
 58.....MATRIZ DE CARGAS UNIFORMES EN CASO DE C.N.P

218 C 59.....MATRIZ Z (EN PASO A PASO) Y MATRIZ DEL SISTEMA  
 219 C EN EL ANALIS DIRECTO  
 220 C 60.....MATRIZ DE EQUILIBRIO REDUCIDA  
 221 C 61.....MATRIZ PARA LA DETERMINACION DE LOS INC DE FI  
 222 C 62.....MATRIZ DE TRABAJO  
 223 C 63.....MATRIZ DE TRABAJO  
 224 C 49.....MATRIZ DE TRABAJO  
 225 C 48.....MATRIZ DE TRABAJO  
 226 C 50.....MATRIZ DE VIRTUALES

227 C

228 C

229 C OPCIONES DE SALIDA.

230 C \*\*\*\*\*

231 C

232 C ISSW

Descripcion

233 C

234 C 0

ENTRADA DEL S.V.A

235 C 1

SALIDA DEL S.V.A

236 C 2

ITERACIONES 2 ORDEN R-P

237 C 3

ITERACIONES 2 ORDEN E-P

238 C 4

ENTRADA SIMPLEX CALCULO DE FI

239 C 5

ENTRADA SIMPLEX ANALISIS S-D

240 C 6

ENTRADA SIMPLEX PROBLEMA DE DISE#0

241 C 7

ITERACION DISE#0 CON RESTRICCIONES

242 C 8

INFORMACION ADICIONAL

243 C 9

ITERACION CARGA UNIFORME

244 C 10

ITERACION DISE#0 CARGAS VARIABLES

245 C 11

ITERACION DISE#0 CON RESTRICCIONES

246 C 13

MATRICES DENTRO DEL P.L (SIMPLEX).

247 C 15

SIN MOVER LA ROTULA VARIABLE

248 C

249 C

250 C PARADAS DEL PROGRAMA.

251 C \*\*\*\*\*

252 C

253 C STOP

Descripcion

254 C

255 C 0000

Problema terminado

256 C 1111

Dimensiones no admisibles

257 C 2222

Problema no conocido

258 C 3330

No existe pivote &lt;math&gt;\leq 0&lt;/math&gt; en S.V.A

259 C 3331

Pivote=0 en el S.V.A

260 C 3332

No existe solucion en el Simplex

261 C 4440

Error IN/IO en la rutina DISCO

262 C 4441

Error IN/IO en la rutina LEEDI

263 C 4442

Error IN/IO en la rutina LEIND

264 C 5550

Matriz Singular ( INVER )

265 C 5551

Matriz Singular ( SLNPD )

266 C

267 C LLAMADA A LA RUTINA DE TIEMPO DE C.P.U

268 C

269 C CALL SECON(TCO)

270 C

271 C ASIGNACION DE UNIDADES LOGICAS

272 C

```

273 CALL RMPAR(IP)
274 IN=IP(1)
275 IO=IP(2)
276 IC=IP(3)
277 ID=IP(4)
278 C
279 C PUESTA A CERO DE VARIABLES
280 C
281 NFI=0
282 NITER=0
283 SUM1=0.
284 C
285 C
286 C LECTURA DE DATOS DE CONSTANTES
287 C
288 READ(IN,50) NOMB
289 READ(IN,*) NCL,N1,N2,N3,N4
290 NN1=N1
291 NN2=N2
292 NN3=N3
293 NN4=N4
294 IF(NCL.LT.20) GOTO 1
295 READ(IN,*) NGD,NGF,CADI
296 1 READ(IN,*)NB,ND,NS,NLI,NAI,NBCU,E,SIGMA,NECNP
297 WRITE(IO,20)NOMB,NB,ND,NS,NLI,NAI,NBCU,E
298 IF(NCL.LE.13.OR.NCL.GE.20) GO TO 123
299 WRITE(IO,124) SIGMA
300 123 IF(NCL.EQ.12.OR.NCL.EQ.16) WRITE(IO,1037) NECNP
301 IF(CADI.EQ.0) CADI=1
302 IF(NCL.GT.19) GOTO 2
303 L=NCL-9
304 GOTO(3,4,6,7,5,8,9),L
305 3 WRITE(IO,30)
306 GOTO 40
307 4 WRITE(IO,31)
308 GOTO 40
309 5 WRITE(IO,32)
310 GOTO 40
311 6 WRITE(IO,33)
312 GOTO 40
313 7 WRITE(IO,34)
314 GOTO 40
315 8 WRITE(IO,35)
316 GOTO 40
317 9 WRITE(IO,36)
318 GO TO 40
319 2 IF(NGD.EQ.0) NGD=1
320 L=NCL-19
321 GOTO(11,12,13,14),L
322 11 WRITE(IO,21)
323 GOTO 41
324 12 WRITE(IO,22)
325 GO TO 41
326 13 WRITE(IO,26)
327 GO TO 41

```

```

328 14  WRITE(IO,27)
329 41  WRITE(IO,24) SIGMA,CADI
330     WRITE(IO,25) NGD
331 40  NGL=2*NB
332     NGL1=3*ND
333     NH=2*NB
334     WRITE(IO,1020)
335     KK=1
336     IF(NCL,NE.15,AND,NCL,NE.14,AND,NCL,NE.16,AND,NCL,NE.23) GO TO 1023
337     KK=4
338     IF(N1,EQ.1) WRITE(IO,1024)
339     IF(N1,EQ.1) GO TO 1045
340     WRITE(IO,1025)
341 1045 IF(N2,EQ.1) WRITE(IO,1026)
342     IF(N2,EQ.1) GO TO 1046
343     WRITE(IO,1027)
344 1046 IF(N3,EQ.1) WRITE(IO,1028)
345     IF(N3,EQ.1) GO TO 1023
346     WRITE(IO,1029)
347 1023 IF(N4,EQ.1,OR,NCL,EQ.14,OR,NCL,EQ.16)WRITE(IO,1031)
348     IF(N4,EQ.1,OR,NCL,EQ.14,OR,NCL,EQ.16) KK=5
349     WRITE(IO,1036) KK
350 C
351 C  LLAMADA A LA RUTINA DE LECTURA DE DATOS
352 C
353     NH=2*NB
354     NND=3*ND
355     NBB=4*NB
356     CALL DATOS(IT,X,Y,NI,NJ,RLON,ALFA,LBE,Z,ND,NB,NGL,NLI,NR1,NAI,
357 @     NR,RMP,RINER,DCL,E,NCL,NGB,NGD,NTP,COL,COI,COM,NH,
358 @     AREA,RMP1)
359     TOL=1./10.**IC
360     TOLS=1./10.**ID
361     NGL=2*ND-NR1+NAI
362     NMI=3*ND-NB-NR
363 C
364 C  ANALISIS DE LAS DIMENSIONES MAXIMAS
365 C
366     IF(NCL,GE.20) GO TO 51
367     IF(NCL,NE.13) GO TO 52
368     IF(2*(NH-NMI)+1,GT.47,OR,2*NH+2*(NH-NMI)+1,GT.147) STOP 1111
369     GO TO 60
370 52  IF(NH,GT.50) STOP 1111
371     GO TO 60
372 51  IF(NCL,EQ.20,OR,NCL,EQ.22) GO TO 53
373     IF(NCL,EQ.21) GO TO 54
374     IF(NCL,EQ.23) GO TO 55
375     STOP 2222
376 53  IF(NMI+NH+1,GT.62,OR,2*NH+NGD,GT.101) STOP 1111
377     GO TO 60
378 54  IF(2*NRM+2*NGD,GT.90,OR,2*(NRM+NGD)+2,GT.92) STOP 1111
379     GO TO 60
380 55  IF(NH+NMI+NGD+1,GT.62,OR,2*NH+2*NGD,GT.101) STOP 1111
381 60  LOC=NB+NAI+NBCU/2
382     DO 511 I=1,NB

```

```

383      RLOF(I)=0.0
384 511  CONTINUE
385 C
386 C      LLAMADA AL SEGMENTO CORRESPONDIENTE
387 C
388 510  CALL EXEC(8,5HLECAR)
389 C
390 C      FORMATOS DE ESCRITURA/LECTURA
391 C
392 22   FORMAT(2/,10X,"TIPO DE PROBLEMA : DISE#0 A MINIMO PESO RESTRINGIDO
393 @")
394 50   FORMAT(30A2)
395 21   FORMAT(2/,10X,"TIPO DE PROBLEMA : DISE#0 A MINIMO PESO")
396 26   FORMAT(2/,10X,"TIPO DE PROBLEMA : DISE#0 CON CARGAS VARIABLES")
397 27   FORMAT(2/,10X,"TIPO DE PROBLEMA : DISE#0 EN TEORIA DE 2 ORDEN")
398 25   FORMAT(10X,"GRUPOS DISTINTO DE BARRAS :",3X,I5)
399 24   FORMAT(10X,"TENSION DE FLUENCIA          :",2X,E14.6," T/CM^2"/,
400 @    10X,"CARGA DE DISE#0          :",2X,E14.6)
401 34   FORMAT(2/,10X,"TIPO DE PROBLEMA : ANALISIS DE SHAKE-DOWN")
402 30   FORMAT(2/,10X,"TIPO DE PROBLEMA : ANALISIS DIRECTO 1 ORDEN")
403 35   FORMAT(2/,10X,"TIPO DE PROBLEMA : ANALISIS DIRECTO 2 ORDEN")
404 31   FORMAT(2/,10X,"TIPO DE PROBLEMA : ANALISIS PASO-PASO 1 ORDEN")
405 32   FORMAT(2/,10X,"TIPO DE PROBLEMA : ANALISIS PASO-PASO 2 ORDEN")
406 33   FORMAT(2/,10X,"TIPO DE PROBLEMA : ANALISIS PASO-PASO 1 ORDEN CARGA
407 @ N.P")
408 36   FORMAT(2/,10X,"TIPO DE PROBLEMA : ANALISIS PASO-PASO 2 ORDEN CARGA
409 @ N.P")
410 20   FORMAT(1H1,10X,30A2/10X,60("*")4/10X,"CONSTANTES DE LA ESTRUCTURA
411 @"/10X,27("-"),3/10X,"NUMERO DE BARRAS",I19/10X,"NUMERO DE NUDOS",
412 @I20/10X,"NUMERO DE SOPORTES",I17/10X,"NUMERO DE LIBERTADES",
413 @I15/10X,"NUMERO DE APOYOS INCLINADOS",I8,/10X,"NUMERO DE BARRAS C
414 @ON CARG. UNIF.",I3//,10X,"MODULO DE ELASTICIDAD",8X,E14.6)
415 124  FORMAT(10X,"TENSION DE PLASTIFICACION",4X,E14.6)
416 1020 FORMAT(/10X,"HIPOTESIS DE CALCULO          :"/)
417 1024 FORMAT(10X,"1.- EL EFECTO SECUNDARIO DEL MOMENTO FLECTOR ES IGNORA
418 @DO")
419 1025 FORMAT(10X,"1.- EL EFECTO SECUNDARIO DEL MOMENTO FLECTOR ES CONSID
420 @ERADO")
421 1026 FORMAT(10X,"2.- EL ESFUERZO AXIL EN VIGAS ES DESPRECIADO")
422 1027 FORMAT(10X,"2.- EL ESFUERZO AXIL EN VIGAS ES CONSIDERADO")
423 1028 FORMAT(10X,"3.- EL EFECTO DEL AXIL SOBRE EL MOMENTO PLASTICO ES IG
424 @NORADO")
425 1029 FORMAT(10X,"3.- EL EFECTO DEL AXIL SOBRE EL MOMENTO PLASTICO ES CO
426 @NSIDERADO")
427 1031 FORMAT(10X,"4.- LOS COEF. DE ESTABILIDAD ESTAN LINEALIZADOS")
428 1036 FORMAT(10X,I1,".- NO SE CONSIDERA DEFORMACION DEBIDA AL AXIL NI AL
429 @ CORTANTE")
430 1037 FORMAT(10X,"NUMERO DE ETAPAS DE CARGA",5X,I5)
431      END

```

\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: 1716 COMMON: (NONE)



```

432 SUBROUTINE DATOS(IT,X,Y,NI,NJ,RLON,ALFA,LBE,Z,ND,NB,NGL,NLI,NR1,
433 @ NAI,NR,RMP,RINER,DCL,E,NCL,NGB,NGD,NTP,COL,COI,
434 @ COM,NH,AREA,RMP1)
435 C
436 C SUBROUTINA PARA LECTURA DE DATOS DE ENTRADA
437 C
438 DIMENSION IT(ND),X(ND),Y(ND),NI(NB),NJ(NB),ALFA(NB),LBE(NLI),
439 @ NGB(NB),NTP(NB),AREA(NB),RLON(NB),Z(ND),RMP(NB),
440 @ DCL(NLI),RINER(NB),D(4),IGB(5),RMP1(NB)
441 COMMON /VAR/ IN,IO
442 C
443 C LECTURA DE DATOS DE NUDOS
444 C
445 NR=0
446 NR1=0
447 DO 1 I=1,ND
448 READ(IN,*) KL,IT(I),X(I),Y(I),Z(I)
449 Z(I)=Z(I)*3.141592/180.
450 J=IT(I)+1
451 GOTO(1,233,233,234,235,233,1,236,236),J
452 233 NR=NR+1
453 NR1=NR1+1
454 GO TO 1
455 234 NR=NR+2
456 NR1=NR1+2
457 GO TO 1
458 235 NR=NR+3
459 NR1=NR1+2
460 GO TO 1
461 236 NR=NR+2
462 NR1=NR1+1
463 1 CONTINUE
464 C
465 C LECTURA DE INCIDENCIA DE BARRAS Y CARACTER.
466 C CALCULO DE SU LONGITUD Y ANGULO
467 C
468 C SI EL PROBLEMA ES DISE#0 LEER LOS DATOS REFERIDOS A ESTE
469 C
470 COL=1.E36
471 IF(NCL.LT.20)GO TO 402
472 IF(NGD.EQ.1) GO TO 403
473 DO 404 I=1,NB
474 READ(IN,*) K,NI(I),NJ(I),NGB(I),NTP(I)
475 RLON(I)=SQRT((X(NJ(I))-X(NI(I)))**2+(Y(NJ(I))-Y(NI(I)))**2)
476 IF(COL.GT.RLON(I))COL=RLON(I)
477 ALFA(I)=ATAN2(Y(NJ(I))-Y(NI(I)),X(NJ(I))-X(NI(I)))
478 404 CONTINUE
479 GO TO 420
480 403 DO 406 I=1,NB
481 READ(IN,*) K,NI(I),NJ(I),NTP(I)
482 RLON(I)=SQRT((X(NJ(I))-X(NI(I)))**2+(Y(NJ(I))-Y(NI(I)))**2)
483 IF(COL.GT.RLON(I))COL=RLON(I)
484 ALFA(I)=ATAN2(Y(NJ(I))-Y(NI(I)),X(NJ(I))-X(NI(I)))
485 406 NGB(I)=1
486 GO TO 420

```

```
487 402 RIMA=0.0
488 COL=1.E36
489 COI=1.E36
490 COM=1.E36
491 DO 2 I=1,NB
492 READ(IN,*) KL,NI(I),NJ(I),RMP(I),RINER(I),AREA(I)
493 IF(AREA(I).EQ.0) AREA(I)=1.
494 RLON(I)=SQRT((X(NJ(I))-X(NI(I)))**2+(Y(NJ(I))-Y(NI(I)))**2)
495 ALFA(I)=ATAN2(Y(NJ(I))-Y(NI(I)),X(NJ(I))-X(NI(I)))
496 C
497 C SI MP=0 HACE MP=1 E IGUAL CON INERCIA
498 C
499 IF(RMP(I).EQ.0.0) RMP(I)=1.
500 IF(RINER(I).EQ.0.) RINER(I)=1
501 C
502 C DEFINICION DE LAS COTAS
503 C
504 IF(RIMA.LT.RINER(I))RIMA=RINER(I)
505 IF(COL.GT.RLON(I))COL=RLON(I)
506 IF(COI.GT.RINER(I))COI=RINER(I)
507 IF(COM.GT.RMP(I))COM=RMP(I)
508 2 CONTINUE
509 C
510 C LECTURA DE LIBERTADES DE BARRAS
511 C
512 420 IF(NLI.EQ.0) GO TO 801
513 DO 4 I=1,NLI
514 4 LBE(NLI)=0
515 DO 3 I=1,NLI
516 READ(IN,*) NF, LB, LN, DCL(NF)
517 LBE(I)=LB*100+LN
518 3 CONTINUE
519 C
520 C ESCRITURA DATOS DE ENTRADA
521 C
522 801 WRITE(IO,21)
523 DO 41 I=1,ND
524 41 WRITE(IO,22) I,IT(I),X(I),Y(I),Z(I)
525 WRITE(IO,23)
526 DO 24 I=1,NB
527 ANG=ALFA(I)*180./3.141592
528 24 WRITE(IO,25) I,NI(I),NJ(I),RLON(I),ANG
529 IF(NCL.LT.20)GO TO 410
530 WRITE(IO,415)
531 DO 416 I=1,NB
532 IF(NCL.EQ.21) RMP1(I)=2000.0
533 LG=NTP(I)+1
534 GO TO(500,501,502,503),LG
535 500 IGB(1)=2HA
536 IGB(2)=2HDE
537 IGB(3)=2HCI
538 IGB(4)=2HDI
539 IGB(5)=2HR
540 GO TO 504
541 501 IGB(1)=2H
```

```

542      IGB(2)=2HIP
543      IGB(3)=2HN
544      IGB(4)=2H
545      IGB(5)=2H
546      GO TO 504
547 502  IGB(1)=2H
548      IGB(2)=2HIP
549      IGB(3)=2HE
550      IGB(4)=2H
551      IGB(5)=2H
552      GO TO 504
553 503  IGB(1)=2H
554      IGB(2)=2HHE
555      IGB(3)=2HB
556      IGB(4)=2H
557      IGB(5)=2H
558 504  WRITE(IO,418) I, NGB(I), IGB
559 416  CONTINUE
560      GO TO 417
561 410  IF(NCL.NE.15.AND.NCL.NE.14.AND.NCL.NE.16) GO TO 664
562      WRITE(IO,667)
563      DO 668 I=1,NB
564      WRITE(IO,669) I,RMP(I),RINER(I),AREA(I)
565 668  CONTINUE
566      GO TO 417
567 664  WRITE(IO,367)
568      DO 369 I=1,NB
569 369  WRITE(IO,368) I,RMP(I),RINER(I)
570 417  IF(NLI.EQ.0) RETURN
571      WRITE(IO,26)
572      DO 27 II=1,NLI
573      J=LBE(II)/100
574      I=LBE(II)-J*100
575      WRITE(IO,28) II,J,I,DCL(II)
576 27  CONTINUE
577 C
578 C      FORMATOS DE ESCRITURA
579 C
580      21 FORMAT(2/,10X"DATOS DE NUDOS"/10X,14("-")3/10X," NUDO TIPO CO
581      @ORD. X",3X,"COORD. Y",3X,"ANG. APOYO INC." )
582      22 FORMAT(10X,I5,I7,3F11.3)
583 23  FORMAT(2/10X,"TABLA DE CONECTIVIDAD Y PROP. GEOMETRICAS DE BARRAS
584      @"/10X,51("-")3/10X," BARRA NUDO1 NUDO2 LONGITUD ANGULO
585      @")
586 367  FORMAT(2/10X,"PROPIEDADES MECANICAS DE LAS BARRAS"/10X,35("-")3/,
587      @10X," BARRA MOMENTO PLASTICO MOMENTO DE INERCIA")
588 667  FORMAT(2/10X,"PROPIEDADES MECANICAS DE LAS BARRAS"/10X,35("-")3/,
589      @10X," BARRA M. PLASTICO M. DE INERCIA AREA")
590 25  FORMAT(10X,I5,I8,I9,3X,2F10.4)
591 669  FORMAT(10X,I5,5X,F10.4,4X,F10.4,6X,F10.4)
592 415  FORMAT(2/10X,"TABLA CON LAS DIFERENTES CLASES DE BARRAS"/10X,
593      @41("-")3/,10X," BARRA GRUPO TIPO PERFIL")
594 418  FORMAT(10X,I5,3X,I6,7X,5A2)
595 368  FORMAT(10X,I5,7X,F10.4,11X,F10.4)
596 26  FORMAT(2/,10X,"LIBERTADES EN BARRAS"/,10X,20("-")3/)

```

597 28 FORMAT(10X,"LIBERTAD",I3,2X,"BARRA",I4," NUDO",I4," CARGA",F10.3)  
598 END

FTN4X COMPILER: HP92834 REV.2130 (810716)

\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: 1733 COMMON: (NONE)

```

599 PROGRAM LECAR(5)
600 C
601 C SEGMENTO PARA LECTURA Y ESCRITURA DE DATOS DE CARGA DE LA
602 C ESTRUCTURA
603 C
604 COMMON /VAR/ IN, IO, NOMB(30), ND, NB, INDT, NMI, NLI, NGL, NAI, NBCU, NMH,
605 @ NFI, NGD, NGF, NCL, LHF, LHV, NITER, NND, NH, NBB, NTCPV, NCFP,
606 @ NCPV, NMIN, NGH, NMCO, NMHCO, NECO, NTRA, LV, CADI, SUMF, E,
607 @ SUM, TOL, TOLS, SUM1, PESO, SIGMA, COP, COL, COI, COM, TCO
608 COMMON /CON/ IT(25), RMP(25), RIMON(50), RMOME(50), NI(25), NJ(25),
609 @ ALFA(25), RLON(25), RINER(25), RMP1(25), AXIL(25), XA(25),
610 @ YA(25), ZA(25), DCL(20), R(75), LBE(20), T(50), LICOL(50),
611 @ NCOL(50), NFILA(50), ICOL(50), DCBU(100), T1(50), Q(50),
612 @ NGB(25), NTP(25), LTP(25), AREA(25), TT(50), CFI(50),
613 @ CMH(50), CMV(50), FIFI(25), X(25), Y(25), Z(25),
614 @ RLCPV(10,2), TETA(100,10), RMAT(75,10)
615 COMMON /PAS/ N1, N2, N3, N4, COLAA, COLAF, RMINC(50), RFIC(50), MC(50),
616 @ RLOF(25), NECNF, NENP, COTA(10)
617 C
618 C LLAMADA A LA RUTINA DE LECTURA DE DATOS DE CARGA CORRESPONDIENTE
619 C
620 C DIMENSIONES CARACTERISTICAS DE LAS MATRICES
621 C
622 L1=75
623 L2=100
624 L3=10
625 IF(NCL.EQ.13.OR.NCL.EQ.22)GO TO 1
626 C
627 C LLAMADA A LA RUTINA DE LECTURA DE DATOS DE CARGAS PROPORCIONALES
628 C
629 CALL DACAF(RLON,ALFA,R,ND,NB,TOL,DCBU,NBCU,IO,COP, —
630 @ CMV,NH,NI,NJ,NCL,NECNF,COTA,RMAT,TETA,NBB,NND,L1,L2,L3)
631 GO TO 2
632 C
633 C LLAMADA A LA RUTINA DE LECTURA DE CARGAS VARIABLES O NO PROPORCIONAL
634 C
635 1 CALL DACAV(RMAT,ND,NB,TOL,COP,NND,L3,L1,RLCPV,NTCPV,NCFP,NCPV)
636 2 WRITE(IO,3) TOL,TOLS
637 C
638 C ESCALADO DEL PROBLEMA
639 C
640 IF(NCL.EQ.13.OR.NCL.EQ.22) NECNF=NTCPV+1
641 DO 4 I=1,NB
642 RLON(I)=RLON(I)/COL
643 RLOF(I)=RLOF(I)/COL
644 IF(NCL.EQ.21) GO TO 4
645 IF(NCL.NE.22) GO TO 20
646 RINER(I)=1.0
647 COI=1.0
648 COM=1.0
649 20 RINER(I)=RINER(I)/COI
650 RMP(I)=RMP(I)/COM
651 RMP1(I)=RMP(I)
652 4 CONTINUE
653 IF(NCL.NE.12.AND.NCL.NE.13.AND.NCL.NE.16.AND.NCL.NE.22) GO TO 5

```

```
654 DO 6 I=1,NND
655 II=I/3
656 RH=I/3.
657 HR=1.
658 IF (ABS (II-RH).EQ.0) HR=COL
659 DO 6 J=1,NECNP
660 RMAT (I,J)=RMAT (I,J)/(COP*HR)
661 6 CONTINUE
662 IF (NCL.NE.13.AND.NCL.NE.22) GO TO 7
663 DO 8 I=1,NTCPV
664 DO 8 J=1,2
665 RLCPV (I,J)=RLCPV (I,J)*COL*COP/COM
666 8 CONTINUE
667 GO TO 9
668 7 DO 10 I=1,NECNP
669 COTA (I)=COTA (I)*COP*COL/COM
670 10 CONTINUE
671 9 CALL LEEDI (RMAT,L1,L3,NND,NECNP,63,2)
672 IF (NCL.EQ.13.OR.NCL.EQ.22) GO TO 11
673 IF (NBCU.EQ.0) GO TO 12
674 DO 13 I=1,NECNP
675 DO 13 J=1,NB
676 TETA ((J-1)*4+1,I)=TETA ((J-1)*4+1,I)/COL
677 TETA ((J-1)*4+2,I)=TETA ((J-1)*4+2,I)/COL
678 TETA ((J-1)*4+3,I)=TETA ((J-1)*4+3,I)*COL/COP
679 TETA ((J-1)*4+4,I)=TETA ((J-1)*4+4,I)*COL/COP
680 13 CONTINUE
681 CALL LEEDI (TETA,L2,L3,NBB,NECNP,58,2)
682 GO TO 12
683 5 DO 14 I=1,NND
684 II=I/3
685 RH=I/3.
686 HR=1.
687 IF (ABS (II-RH).EQ.0) HR=COL
688 R (I)=R (I)/(COP*HR)
689 RMAT (I,1)=R (I)
690 14 CONTINUE
691 CALL LEEDI (RMAT,L1,L3,NND,1,63,2)
692 11 IF (NBCU.EQ.0) GO TO 12
693 DO 15 I=1,NB
694 DCBU ((I-1)*4+1)=DCBU ((I-1)*4+1)/COL
695 DCBU ((I-1)*4+2)=DCBU ((I-1)*4+2)/COL
696 DCBU ((I-1)*4+3)=DCBU ((I-1)*4+3)*COL/COP
697 DCBU ((I-1)*4+4)=DCBU ((I-1)*4+4)*COL/COP
698 15 CONTINUE
699 C
700 C PUESTA A CERO DE LAS MATRICES
701 C
702 12 CALL ZERO (RMAT,L1,L3,NND,L3)
703 CALL ZERO (TETA,L2,L3,NBB,L3)
704 IF (NCL.NE.21) GO TO 21
705 DO 22 I=1,NB
706 CFI (I)=1.0
707 22 CONTINUE
708 CALL ELECO (RMP,RMP1,RINER,AREA,NB,SIGMA,NTP,COM,COI,CFI)
```

```
709      CALL ELETE(NTP,CFI,CMH,NB,COM,RMP1,SIGMA,COI)
710      DO 23 I=1,2*ND
711      CMV(I)=CMV(I)*(E*COI)/(COM*COL*COL)
712 23    CONTINUE
713 C
714 C      LLAMADA LA SEGMENTO CORRESPONDIENTE
715 C
716 21    CALL EXEC(8,5HFORMA)
717 C
718 C      FORMATOS DE ESCRITURA
719 C
720 3      FORMAT(4/,10X,"TOLERANCIA DE OPERACION.....",E12.4/
721      @          10X,"TOLERANCIA DE COMPARACION.....",E12.4)
722      WRITE(10,55)
723 55    FORMAT(1H1)
724      END
```

FTN4X COMPILER: HP92834 REV.2130 (810716)

\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: 914 COMMON: (NONE)

```
725 SUBROUTINE DACAF(RLON,ALFA,R,ND,NB,TOL,DCBU,NBCU, ID,COP,  
726 @ CMV,NH,NI,NJ,NCL,NECNF,COTA,RMAT,TETA,NBB,NND,N1,  
727 @ N2,N3)  
728 C  
729 C SUBROUTINA PARA LECTURA DE DATOS DE CARGA  
730 C  
731 DIMENSION ALFA(NB),CMV(ND),R(NND),D(4),DCBU(NBB),NI(NB),NJ(NB),  
732 @ COTA(NECNF),RMAT(N1,N3),TETA(N2,N3),RLON(NB)  
733 COMMON /VAR/ IN,IO  
734 C  
735 C LECTURA DE DATOS DE CARGA  
736 C  
737 COP=0.0  
738 IF(NCL.NE.12.AND.NCL.NE.16) GO TO 1  
739 WRITE(IO,2)  
740 3 NET=NET+1  
741 D(1)=0.0  
742 D(2)=0.0  
743 D(3)=0.0  
744 D(4)=0.0  
745 IF(NET.GT.NECNF) RETURN  
746 READ(IN,*) NETA,COTA(NET)  
747 WRITE(IO,4) NET,COTA(NET)  
748 1 DO 5 I=1,NND  
749 5 R(I)=0.0  
750 READ(IN,*) NCA,D(1),D(2),D(3)  
751 IF(NCA.EQ.0) GO TO 6  
752 WRITE(IO,7)  
753 WRITE(IO,8)  
754 GO TO 9  
755 10 DO 11 I=1,3  
756 11 D(I)=0.  
757 READ(IN,*) NCA,D(1),D(2),D(3)  
758 9 IF(COP.LT.ABS(D(1)))COP=ABS(D(1))  
759 IF(COP.LT.ABS(D(2)))COP=ABS(D(2))  
760 IF(COP.LT.ABS(D(3)))COP=ABS(D(3))  
761 IF(NCA.EQ.0) GO TO 6  
762 WRITE(IO,12) NCA,D(1),D(2),D(3)  
763 DO 13 J=1,3  
764 R((NCA-1)*3+J)=D(J)  
765 13 CONTINUE  
766 GO TO 10  
767 6 IF(NBCU.EQ.0) GO TO 14  
768 C  
769 C LECTURA DE DATOS DE CARGAS UNIFORMES  
770 C  
771 WRITE(IO,15)  
772 DO 16 I=1,NBB  
773 16 DCBU(I)=0.0  
774 DO 17 I=1,NBCU  
775 DO 18 IL=1,4  
776 18 D(IL)=0.0  
777 READ(IN,*)NBC,LTIP,D1,D2  
778 WRITE(IO,19)NBC,LTIP,D1,D2  
779 IF(COP.LT.ABS(D2))COP=ABS(D2)
```



```

780 C
781 C   ALMACENAMIENTO DE LOS DATOS DE CARGA EN EL CONJUNTO
782 C   POR LLAMADA A LA RUTINA DACU
783 C
784     RLO=RLON(NBC)
785     ALF=ALFA(NBC)
786     CALL DACU(NBC,LTIP,D1,D2,RLO,ALF,D)
787     DO 20 J=1,4
788     DCBU((NBC-1)*4+J)=D(J)
789 20  CONTINUE
790 17  CONTINUE
791 14  IF(NCL.NE.12.AND.NCL.NE.16) GO TO 21
792     DO 22 I=1,NND
793     RMAT(I,NET)=R(I)
794     R(I)=0.0
795 22  CONTINUE
796     IF(NBCU.EQ.0) GO TO 23
797     DO 24 I=1,NBE
798     TETA(I,NET)=DCBU(I)
799 24  CONTINUE
800 23  GO TO 3
801 21  IF(NCL.NE.21) RETURN
802     WRITE(IO,31)
803 33  READ(IN,*) I,CH,CV
804     IF(I.EQ.0) RETURN
805     CMV(2*I-1)=CH
806     CMV(2*I)=CV
807     WRITE(IO,34) I,CH,CV
808     GO TO 33
809 C
810 C   FORMATOS DE ESCRITURA
811 C
812 2   FORMAT(3/,10X,"DATOS DE CARGAS NO PROPORCIONAL"/10X,31("*"))
813 4   FORMAT(3/10X,"ETAPA DE CARGA :",I2,5X,"COTA DE LA CARGA :",E14.6)
814 7   FORMAT(2/10X,"DATOS DE CARGAS EN NUDOS"/,10X,24("-")3/)
815 8   FORMAT(10X,"NUDO",2X,"FUERZA X",3X,"FUERZA Y",3X,"MOME Z")
816 12  FORMAT(10X,I3,3(F10.3))
817 15  FORMAT(2/,10X,"CARGAS EN BARRAS"/10X,16("-")3/,10X," BARRA
818 @TIPO          DAT1          DAT2")
819 19  FORMAT(13X,I3,8X,I3,6X,F10.4,5X,F10.4)
820 31  FORMAT(2/,10X,"LIMITACIONES EN LOS MOVIMIENTOS"/10X,31("-"),3/,
821 @ 10X,"NUDO          COTA MOV. X          COTA MOV. Y")
822 34  FORMAT(10X,I3,3X,E14.6,4X,E14.6)
823     END

```

FTN4X COMPILER: HP92834 REV.2130 (810716)

\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: 950 COMMON: (NONE)

```
824     SUBROUTINE DACAV(RMAT,ND,NB,TOL,COP,NND,N3,N1,RLCPV,NTCPV,NCFF,  
825     @          NCPV)  
826 C  
827 C     SUBROUTINA PARA LA LECTURA DE CARGAS FIJAS Y VARIABLES  
828 C  
829     DIMENSION D(4),RMAT(N1,N3),RL(6),RLCPV(N3,2)  
830     COMMON /VAR/ IN,IO  
831 C  
832 C     LECTURA DE DATOS GENERALES  
833 C  
834     READ(IN,*)NCFF,NCPV  
835     WRITE(IO,1) NCFF,NCPV  
836     COP=0.0  
837     IF(NCFF.EQ.0) GO TO 100  
838 C  
839 C     LECTURA Y ALMACENAMIENTO DE LA CARGAS PUNTUALES FIJAS  
840 C  
841     WRITE(IO,2)  
842     WRITE(IO,3)  
843     DO 10 JJ=1,NCFF  
844     DO 4 I=1,3  
845 4     D(I)=0  
846     READ(IN,*) NCA,D(1),D(2),D(3)  
847     IF(COP.LT.ABS(D(1)))COP=ABS(D(1))  
848     IF(COP.LT.ABS(D(2)))COP=ABS(D(2))  
849     IF(COP.LT.ABS(D(3)))COP=ABS(D(3))  
850     WRITE(IO,5) NCA,D(1),D(2),D(3)  
851     DO 6 J=1,3  
852     RMAT((NCA-1)*3+J,1)=D(J)  
853 6     CONTINUE  
854 10     CONTINUE  
855 C  
856 C     LECTURA Y ALMACENAMIENTO DE LAS CARGAS PUNTUALES VARIABLES  
857 C  
858 100     IF(NCPV.EQ.0) GO TO 102  
859     WRITE(IO,33)  
860     DO 34 I=1,6  
861 34     RL(I)=0  
862     DO 35 I=1,4  
863 35     D(I)=0.0  
864     K=1  
865     DO 16 I=1,NCPV  
866     READ(IN,*)NCA,D(1),D(2),D(3),(RL(JK),JK=1,6)  
867     IF(COP.LT.ABS(D(1))) COP=ABS(D(1))  
868     IF(COP.LT.ABS(D(2))) COP=ABS(D(2))  
869     IF(COP.LT.ABS(D(3))) COP=ABS(D(3))  
870     WRITE(IO,17) NCA,D(1),RL(1),D(2),RL(3),D(3),RL(5),RL(2),RL(4),  
871     @          RL(6)  
872     IF(D(1).EQ.0) GO TO 18  
873     K=K+1  
874     RMAT((NCA-1)*3+1,K)=D(1)  
875     RLCPV(K-1,1)=RL(1)  
876     RLCPV(K-1,2)=RL(2)  
877 18     IF(D(2).EQ.0) GO TO 19  
878     K=K+1
```

```
879      RMAT((NCA-1)*3+2,K)=D(2)
880      RLCPV(K-1,1)=RL(3)
881      RLCPV(K-1,2)=RL(4)
882 19    IF(D(3).EQ.0) GO TO 16
883      K=K+1
884      RMAT((NCA-1)*3+3,K)=D(3)
885      RLCPV(K-1,1)=RL(5)
886      RLCPV(K-1,2)=RL(6)
887 16    CONTINUE
888      NTCPV=K-1
889 102    RETURN
890 C
891 C      FORMATOS DE ESCRITURA
892 C
893 1      FORMAT(2/,10X,"DATOS GENERALES DE CARGA :"/10X,24("-")3/10X,
894      @"NUMERO DE NUDOS CON CARGAS PUNTUALES FIJAS      :",I4/10X,
895      @"NUMERO DE NUDOS CON CARGAS PUNTUALES VARIABLES :",I4)
896 2      FORMAT(2/,10X,"CARGAS FIJAS EN NUDOS"/10X,21("-")3/)
897 3      FORMAT(10X,"NUDO FUERZA X      FUERZA Y      MOME Z")
898 5      FORMAT(10X,I3,3(F10.3))
899 33     FORMAT(2/,10X,"DATOS DE CARGAS VARIABLES EN NUDOS"/10X,35("-")3/,
900      @10X,"NUDO FUERZA X      LIMITES FUERZA Y      LIMITES MOME Z
901      @LIMITES")
902 17     FORMAT(9X,I4,1X,F10.3,1X,F10.3,F10.3,2X,F10.3,F9.3,2X,
903      @      F10.3,/,25X,F10.3,12X,F10.3,11X,F10.3)
904      END
```

FTN4X COMPILER: HP92834 REV.2130 (810716)

\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: 806 COMMON: (NONE)

```
905      SUBROUTINE DACU(NBC,LTIP,D1,D2,RLO,ALF,D)
906 C
907 C      SUBROUTINA PARA LA DETERMINACION DEL VECTOR DE CARGAS UNIFORMES
908 C
909      DIMENSION D(4)
910      GO TO (10,11,12,13),LTIP
911 10     D(1)=D1/ABS(COS(ALF))
912      D(2)=RLO
913      D(3)=D2*ABS(COS(ALF))
914      GO TO 14
915 11     D(1)=D1/ABS(SIN(ALF))
916      D(2)=RLO
917      D(4)=D2*ABS(SIN(ALF))
918      GO TO 14
919 12     D(1)=D1
920      D(2)=RLO
921      D(3)=D2*COS(ALF)
922      D(4)=-D2*SIN(ALF)
923      GO TO 14
924 13     D(1)=D1
925      D(2)=RLO
926      D(3)=D2*COS(ALF)
927      D(4)=D2*SIN(ALF)
928 14     RETURN
929      END
```

FTN4X COMPILER: HP92834 REV.2130 (810716)

\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: 167 COMMON: (NONE)

```

930      PROGRAM FORMA(5)
931 C
932 C      SEGMENTO PARA LA FORMACION DE LA MATRIZ RESTRICTIVA Y CALCULO
933 C      DE SOLUCIONES INDEPENDIENTES
934 C
935      DIMENSION A1(75),L1(75),L2(75),L3(75)
936      COMMON /VAR/ IN, IO, NOMB(30), ND, NB, INDT, NMI, NLI, NGL, NAI, NBCU, NMH,
937 @          NFI, NGD, NGF, NCL, LHF, LHV, NITER, NND, NH, NBB, NTCFV, NCFP,
938 @          NCPV, NMIN, NGH, NMCO, NMHCO, NECC, NTRA, LV, CADI, SUMF, E,
939 @          SUM, TOL, TOLS, SUM1, PESQ, SIGMA, COP, COL, COI, COM, TCO
940      COMMON /CON/ IT(25), RMP(25), RIMON(50), RMOME(50), NI(25), NJ(25),
941 @          ALFA(25), RLON(25), RINER(25), RMP1(25), AXIL(25), XA(25),
942 @          YA(25), ZA(25), DCL(20), R(75), LBE(20), T(50), LICOL(50),
943 @          NCQL(50), NFILA(50), ICOL(50), DCBU(100), T1(50), Q(50),
944 @          NGB(25), NTP(25), LTP(25), AREA(25), TT(50), CFI(50),
945 @          CMH(50), CMV(50), FIFI(25), X(25), Y(25), Z(25),
946 @          RLCPV(10,2), TETA(50,50), RMAT(40,75)
947 C
948 C      DIMENSIONES CARACTERISTICAS
949 C
950      LOC=NB+NAI+NBCU/2
951      LB1=40
952      N2=75
953      N3=50
954 C
955 C      LLAMADA A LA RUTINA DE FORMACION Y RESOLUCION
956 C
957      CALL FORM(IT, NI, NJ, ALFA, Z, RMAT, TETA, NB, ND, NGL, NMI, NAI, LOC, INDT,
958 @          TOL, L1, L2, L3, A1, NFILA, NBCU, RLON, NCL, LB1, N2, N3)
959      NMI=NMI+NLI
960 C
961 C      PUESTA A CERO
962 C
963      CALL ZERO(RMAT, LB1, N2, LB1, N2)
964 C
965 C      LLAMADA AL SEGMENTO CORRESPONDIENTE
966 C
967      CALL EXEC(8,5HMECAN)
968      END

```

FTN4X COMPILER: HF92834 REV.2130 (810716)

\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: 445 COMMON: (NONE)

```
969 SUBROUTINE FORM(IT,NI,NJ,ALFA,Z,RMAT,TETA,NB,ND,NGL,NMI,NAI,LOC,
970 @ INDT,TOL,L1,L2,L3,A1,NFILA,NBCU,RLON,NCL,LB1,N2,N3)
971 C
972 C SUBROUTINA PARA LA FORMACION DE LA MATRIZ RESTRICTIVA
973 C
974 DIMENSION IT(ND),NI(NB),NJ(NB),L3(N2),L1(N2),L2(N2),A1(N2),
975 @ ALFA(NB),Z(ND),RLON(NB),RMAT(LB1,N2),TETA(N3,N3)
976 COMMON /VAR/ IN,IO,NOMB(30)
977 DO 50 I=1,NB
978 II=NI(I)
979 JJ=NJ(I)
980 C=COS(ALFA(I))
981 S=SIN(ALFA(I))
982 C=TOLE(C,TOL)
983 S=TOLE(S,TOL)
984 CALL POS(IT,ND,II,M,N)
985 IF(M.EQ.0) GO TO 101
986 RMAT(I,M)=C
987 101 IF(N.EQ.0) GO TO 102
988 RMAT(I,N)=S
989 102 CALL POS(IT,ND,JJ,M,N)
990 IF(M.EQ.0) GO TO 103
991 RMAT(I,M)=-C
992 103 IF(N.EQ.0) GO TO 50
993 RMAT(I,N)=-S
994 50 CONTINUE
995 IF(NAI.EQ.0) GO TO 51
996 LA=1
997 DO 52 J=1,ND
998 IF(Z(J).EQ.0) GO TO 52
999 CALL POS(IT,ND,J,M,N)
1000 RMAT(NB+LA,M)=-SIN(Z(J))
1001 RMAT(NB+LA,N)=COS(Z(J))
1002 LA=LA+1
1003 52 CONTINUE
1004 C
1005 C INTRODUCCION DE LAS CONDICIONES DE BARRA UNICA
1006 C
1007 51 IF(NBCU.EQ.0) GO TO 53
1008 NCOM=NB+NAI
1009 DO 54 J=1,ND
1010 IF(IT(J).NE.6) GO TO 54
1011 NCOM=NCOM+1
1012 DO 55 K=1,NB
1013 IF(J.EQ.NJ(K))GO TO 56
1014 GO TO 55
1015 56 NB1=K
1016 NB2=K+1
1017 C=COS(ALFA(NB1))
1018 S=SIN(ALFA(NB1))
1019 RL1=RLON(NB1)
1020 RL2=RLON(NB2)
1021 II=NI(NB1)
1022 JJ=NI(NB2)
1023 IJ=NJ(NB1)
```

```

1024      CALL POS(IT,ND,II,M,N)
1025      IF(M.EQ.0) GO TO 61
1026      RMAT(NCOM,M)=S/RL1
1027      RMAT(NCOM,M)=TOLE(RMAT(NCOM,M),TOL)
1028 61    IF(N.EQ.0) GO TO 62
1029      RMAT(NCOM,N)=-C/RL1
1030      RMAT(NCOM,N)=TOLE(RMAT(NCOM,N),TOL)
1031 62    CALL POS(IT,ND,II,M,N)
1032      IF(M.EQ.0) GO TO 63
1033      RMAT(NCOM,M)=S/RL2
1034      RMAT(NCOM,M)=TOLE(RMAT(NCOM,M),TOL)
1035 63    IF(N.EQ.0) GO TO 64
1036      RMAT(NCOM,N)=-C/RL2
1037      RMAT(NCOM,N)=TOLE(RMAT(NCOM,N),TOL)
1038 64    CALL POS(IT,ND,IJ,M,N)
1039      IF(M.EQ.0) GO TO 65
1040      RMAT(NCOM,M)=-S*(RL1+RL2)/(RL1*RL2)
1041      RMAT(NCOM,N)=TOLE(RMAT(NCOM,N),TOL)
1042 65    IF(N.EQ.0) GO TO 55
1043      RMAT(NCOM,N)=C*(RL1+RL2)/(RL1*RL2)
1044      RMAT(NCOM,N)=TOLE(RMAT(NCOM,N),TOL)
1045      GO TO 54
1046 55    CONTINUE
1047 54    CONTINUE
1048 C
1049 C      LLAMADA A LA RUTINA DE CALCULO DE SOLUCIONES INDEPENDIENTES
1050 C
1051 53    NO=0
1052      IF(NCL.NE.15.AND.NCL.NE.14.AND.NCL.NE.16.AND.NCL.NE.23) GO TO 58
1053 C
1054 C      EN CASO DE ANALISIS DE 2 ORDEN ORLAR LA MATRIZ RESTRICTIVA
1055 C
1056      DO 1 I=1,NB
1057      RMAT(I,I+NGL)= 1
1058 1     CONTINUE
1059      NGL=NGL+NB
1060      NO=NB
1061 58    CALL RESOL(RMAT,TETA,L1,L2,L3,A1,INDT,TOL,NGL,LOC,KNT,NO,LB1,N2,
1062 @      N3)
1063      NGL=NGL-NO
1064      INDT=INDT-NO
1065      NMI=NMI+KNT
1066      RETURN
1067      END

```

FTN4X COMPILER: HP92834 REV.2130 (810716)

\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: 764 COMMON: (NONE)

```
1068 SUBROUTINE RESOL(RMAT,TETA,L1,L2,L3,A1,INDT,TOL,N,M,KNT,NO,LB1,N2,
1069 @ N3)
1070 C
1071 C SUBROUTINA PARA EL CALCULO DE SOLUCIONES INDEPENDIENTES
1072 C
1073 DIMENSION RMAT(LB1,N2),TETA(N3,N3),L1(N2),L2(N2),L3(N2),A1(N2)
1074 COMMON /VAR/ IN,IO,NOMB(30)
1075 DO 1 I=1,M
1076 C
1077 C BUSQUEDA DEL PIVOTE
1078 C
1079 MAYOR=1
1080 AMAX=RMAT(I,1)
1081 DO 2 J=1,N-ND
1082 IF(ABS(RMAT(I,J)).LT.ABS(AMAX))GO TO 2
1083 MAYOR=J
1084 AMAX=RMAT(I,J)
1085 2 CONTINUE
1086 IF(ABS(AMAX).LT.TOL) GO TO 1
1087 DO 45 J=1,N
1088 RMAT(I,J)=RMAT(I,J)/AMAX
1089 RMAT(I,J)=TOLE(RMAT(I,J),TOL)
1090 45 CONTINUE
1091 L2(I)=MAYOR
1092 C
1093 C LLAMADA A LA RUTINA DE TRIANGULACION
1094 C
1095 CALL TRIAN(RMAT,I,MAYOR,TOL,M,N,LB1,N2)
1096 1 CONTINUE
1097 C
1098 C ORDENACION INTERNA Y GENERACION DE CONJUNTOS DE ORDENACION
1099 C
1100 INDP=N-M
1101 KNT=0
1102 DO 3 I=1,M
1103 IF(L2(I).EQ.0) KNT=KNT+1
1104 3 CONTINUE
1105 INDT=INDP+KNT
1106 DO 4 I=1,N
1107 4 L1(I)=I
1108 DO 5 J=1,M
1109 DO 5 K=1,N
1110 IF(L2(J).EQ.L1(K)) L1(K)=0
1111 5 CONTINUE
1112 K=1
1113 DO 6 J=1,N
1114 IF(L1(J).EQ.0) GO TO 6
1115 L3(K)=J
1116 K=K+1
1117 6 CONTINUE
1118 KH=N
1119 DO 8 I=1,INDT
1120 DO 9 J=1,N
1121 9 A1(J)=0.0
1122 DO 10 J=1,M
```



```
1123 10  A1(L2(J))=-RMAT(J,L3(I))
1124     A1(L3(I))=1
1125     DO 11 J=1,N-NO
1126     TETA(J,KH)=A1(J)
1127 11   CONTINUE
1128     KH=KH-1
1129 8    CONTINUE
1130     LKJ=N-NO
1131     RETURN
1132     END
```

FTN4X COMPILER: HP92834 REV.2130 (810716)

\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: 418 COMMON: (NONE)

```
1133      SUBROUTINE POS(IT,ND,II,M,N)
1134 C
1135 C      RUTINA PARA REORDENAMIENTO DE LA MATRIZ RESTRICTIVA
1136 C
1137      DIMENSION IT(ND)
1138      N=0
1139      M=0
1140      IF(II.EQ.1) GO TO 1
1141 C
1142 C      EN CASO DE APOYOS INTRASLACIONALES NO MONTAR
1143 C
1144      IF(IT(II).EQ.4.OR.IT(II).EQ.3) RETURN
1145      DO 10 I=1,II-1
1146      IL=IT(I)+1
1147      GO TO(30,20,20,10,10,30,30,20,20),IL
1148 30      M=M+2
1149      GO TO 10
1150 20      M=M+1
1151 10      CONTINUE
1152 1      IL=IT(II)+1
1153      GO TO (40,50,60,100,100,40,40,50,60),IL
1154 40      N=M+2
1155      M=M+1
1156 100     RETURN
1157 50      N=M+1
1158      M=0
1159      GO TO 100
1160 60      M=M+1
1161      N=0
1162      GO TO 100
1163      END
```

FTN4X COMPILER: HP92834 REV.2130 (810716)

\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: 118 COMMON: (NONE)

```
1164 PROGRAM MECAN(5)
1165 C
1166 C SEGMENTO PARA EL CALCULO DE MECANISMOS INDEPENDIENTES
1167 C
1168 COMMON /VAR/ IN, IO, NOMB(30), ND, NB, INDT, NMI, NLI, NGL, NAI, NBCU, NMH,
1169 @ NFI, NGD, NGP, NCL, LHF, LHV, NITER, NND, NH, NBB, NTCPV, NCFP,
1170 @ NCPV, NMIN, NGH, NMCO, NMHCO, NECO, NTRA, LV, CADI, SUMF, E,
1171 @ SUM, TOL, TOLS, SUM1, PESO, SIGMA, COP, COL, COI, COM, TCO
1172 COMMON /CON/ IT(25), RMP(25), RIMON(50), RMOIME(50), NI(25), NJ(25),
1173 @ ALFA(25), RLON(25), RINER(25), RMP1(25), AXIL(25), XA(25),
1174 @ YA(25), ZA(25), DCL(20), R(75), LBE(20), T(50), LICOL(50),
1175 @ NCOL(50), NFILA(50), ICOL(50), DCBU(100), T1(50), Q(50),
1176 @ NGB(25), NTP(25), LTP(25), AREA(25), TT(50), CFI(50),
1177 @ CMH(50), CMV(50), FIFI(25), X(25), Y(25), Z(25),
1178 @ RLCPV(10,2), TETA(50,50), RMAT(50,50)
1179 LOC=NB+NAI+NBCU/2
1180 LB1=50
1181 C
1182 C LLAMADA A LA RUTINA DE CALCULO DE MECANISMOS INDEPENDIENTES
1183 C
1184 CALL MECA(IT, NI, NJ, RLON, ALFA, LBE, Z, TETA, NB, ND, NGL, NMI, NLI,
1185 @ NH, NAI, LOC, INDT, RMAT, LB1, TOL, NCL)
1186 C
1187 C LLAMADA AL SEGMENTO CORRESPONDIENTE
1188 C
1189 CALL EXEC(8, 5HTECAN)
1190 END
```

FTN4X COMPILER: HP92834 REV.2130 (810716)

\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: 47 COMMON: (NONE)

```

1191 SUBROUTINE MECA(IT,NI,NJ,RLON,ALFA,LBE,Z,TETA,NB,ND,NGL,NMI,
1192 @ NLI,NH,NAI,LOC,INDT,RMAT,LB1,TOL,NCL)
1193 C
1194 C SUBROUTINA PARA EL CALCULO DE LOS MECANISMOS INDEPENDIENTES
1195 C
1196 DIMENSION IT(ND),NI(NB),NJ(NB),ALFA(NB),LBE(20),Z(ND),
1197 @ TETA(LB1,LB1),RMAT(LB1,LB1),RLON(NB)
1198 COMMON /VAR/ IN,IO,NOMB(30)
1199 IF(INDT.NE.0) GO TO 222
1200 IF(NCL.NE.15.AND.NCL.NE.14.AND.NCL.NE.16.AND.NCL.NE.23) GO TO 2
1201 222 IF(NCL.NE.15.AND.NCL.NE.14.AND.NCL.NE.16.AND.NCL.NE.23) GO TO 100
1202 INDT=INDT+NB
1203 NGL=NGL+NB
1204 100 KH=NGL
1205 DO 1 I=1,INDT
1206 DO 101 J=1,ND
1207 U=0
1208 V=0
1209 CALL POS (IT,ND,J,M,N)
1210 IF(M.NE.0) U=TETA(M,KH)
1211 IF(N.NE.0) V=TETA(N,KH)
1212 LLJ=2*J-1
1213 RMAT(I,LLJ)=U
1214 RMAT(I,LLJ+1)=V
1215 101 CONTINUE
1216 KH=KH-1
1217 1 CONTINUE
1218 L1=2*ND
1219 NO=0
1220 IF(NCL.EQ.15.OR.NCL.EQ.14.OR.NCL.EQ.16.OR.NCL.EQ.23)NO=NB
1221 NUJ=NGL-NO
1222 CALL LEEDI(TETA,LB1,LB1,NUJ,NGL,51,2)
1223 IF(INDT-NO.EQ.0) GO TO 223
1224 DO 20 I=1,INDT-NO
1225 DO 20 J=1,L1
1226 TETA(INDT-NO-I+1,J)=RMAT(I,J)
1227 20 CONTINUE
1228 223 IF(NCL.NE.15.AND.NCL.NE.14.AND.NCL.NE.16.AND.NCL.NE.23) GO TO 121
1229 II=0
1230 DO 21 I=INDT-NB+1,INDT
1231 II=II+1
1232 DO 21 J=1,L1
1233 TETA(INDT-II+1,J)=RMAT(I,J)
1234 21 CONTINUE
1235 121 DO 22 I=1,INDT
1236 DO 22 J=1,L1
1237 RMAT(I,J)=TETA(I,J)
1238 22 CONTINUE
1239 CALL LEEDI(TETA,LB1,LB1,NUJ,NGL,51,1)
1240 CALL LEEDI(RMAT,LB1,LB1,INDT,L1,51,2)
1241 C
1242 C PUESTA A CERO DE LAS MATRICES
1243 C
1244 2 DO 3 I=1,LB1
1245 DO 3 J=1,LB1

```

```
1246      RMAT(I,J)=TETA(I,J)
1247      TETA(I,J)=0.
1248 3      CONTINUE
1249 C
1250 C      SI SE TRATA DE ANALISIS DE 2 ORDEN FORMAR LA MATRIZ D
1251 C
1252      IF(NCL.NE.15.AND.NCL.NE.14.AND.NCL.NE.16.AND.NCL.NE.23) GO TO 10
1253      CALL METRA(TETA,RMAT,ALFA,NI,NJ,IT,RLON,LB1,NB,ND,NB,INDT,NGL,TOL)
1254      CALL LEEDI(TETA,LB1,LB1,NB,NH,52,2)
1255      CALL ZERO(TETA,LB1,LB1,NB,NH)
1256      INDT=INDT-NB
1257 C
1258 C      GENERAR LOS MECANISMOS DE NUDO
1259 C
1260 10     CALL MECNU(TETA,NI,NJ,IT,LBE,ND,NB,NLI,INDT,NMI,LB1,LB1,MNM)
1261 C
1262 C      GENERAR ECUACIONES ASOCIADAS A TRASLACIONALES
1263 C
1264      IF(INDT.EQ.0) GO TO 4
1265      CALL METRA(TETA,RMAT,ALFA,NI,NJ,IT,RLON,LB1,NB,ND,INDT,INDT,NGL,
1266 @      TOL)
1267 C
1268 C      GRABAR EN FICHERO
1269 C
1270 4      CALL LEEDI(TETA,LB1,LB1,NMI,NH,55,2)
1271      IF(NCL.NE.15.AND.NCL.NE.14.AND.NCL.NE.16.AND.NCL.NE.23) RETURN
1272 C
1273 C      EN CASO DE ANALISIS DE 2 ORDEN CALCULAR Y GUARDAR V(N*DELTA)
1274 C
1275      CALL ZERO(RMAT,LB1,LB1,NMI,NB)
1276      DO 8 I=1,INDT
1277      LO=0
1278      DO 8 J=1,NH,2
1279      LO=LO+1
1280      RMAT(I,LO)=TETA(I,J)
1281 8      CONTINUE
1282      CALL LEEDI(RMAT,LB1,LB1,NMI,NB,54,2)
1283      CALL LEEDI(TETA,LB1,LB1,NB,NH,52,1)
1284      NV=2*N
1285      CALL ZERO(RMAT,LB1,LB1,NV,NV)
1286      DO 12 I=1,NB
1287      LO=0
1288      DO 12 J=1,NH,2
1289      LO=LO+1
1290      RMAT(I,LO)=TETA(I,J)
1291 12     CONTINUE
1292      CALL LEEDI(RMAT,LB1,LB1,NB,NB,53,2)
1293      CALL LEEDI(TETA,LB1,LB1,NMI,NH,55,1)
1294      RETURN
1295      END
```

\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: 791 COMMON: (NONE)

```

1296 SUBROUTINE MECNU(TETA,NI,NJ,IT,LBE,ND,NB,NLI,INDT,NMI,LB1,LB2,NMN)
1297 C
1298 C GENERACION DE LOS MECANISMOS DE NUDO
1299 C
1300 DIMENSION TETA(LB1,LB2),NI(NB),NJ(NB),IT(ND),LBE(20)
1301 LL=1
1302 DO 202 J=1,ND
1303 IF(IT(J).EQ.4.OR.IT(J).EQ.7.OR.IT(J).EQ.8) GO TO 202
1304 DO 228 K=1,NB
1305 LCX=0
1306 IF(NLI.EQ.0) GO TO 610
1307 DO 609 IL=1,NLI
1308 LB=LBE(IL)/100
1309 IF(LB.NE.K) GO TO 609
1310 LN=LBE(IL)-LB*100
1311 IF(LN.NE.J) GO TO 609
1312 LCX=1
1313 GO TO 613
1314 609 CONTINUE
1315 613 IF(LCX.EQ.0) GO TO 610
1316 IF(LN.EQ.NI(K)) GO TO 205
1317 610 IF(NI(K).NE.J) GO TO 205
1318 TETA(INDT+LL,2*(K-1)+1)=1
1319 GO TO 228
1320 205 IF(NLI.EQ.0) GO TO 612
1321 IF(LCX.EQ.0) GO TO 612
1322 IF(LN.EQ.NJ(K)) GO TO 228
1323 612 IF(NJ(K).NE.J) GO TO 228
1324 TETA(INDT+LL,2*(K-1)+2)= 1
1325 228 CONTINUE
1326 LL=LL+1
1327 202 CONTINUE
1328 C
1329 C GENERAR ECUACIONES ASOCIADAS A LIBERTADES
1330 C
1331 NMN=LL-1
1332 IF(NLI.EQ.0) GO TO 601
1333 NCOM=NMN+INDT+1
1334 KL=0
1335 DO 602 I=NCOM,NCOM+NLI-1
1336 KL=KL+1
1337 LB=LBE(KL)/100
1338 LNN=LBE(KL)-LB*100
1339 IF(LNN.EQ.NJ(LB)) GO TO 603
1340 TETA(I,2*LB-1)= 1
1341 GO TO 602
1342 603 TETA(I,2*LB)= 1
1343 602 CONTINUE
1344 601 RETURN
1345 END

```

\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: 306 COMMON: (NONE)



```
1346 SUBROUTINE METRA(TETA, RMAT, ALFA, NI, NJ, IT, RLON, LB1, NB, ND, NN, MM, NGL,
1347 @ TOL)
1348 C
1349 C RUTINA PARA LA DETERMINACION DE LAS ECUACIONES ASOCIADAS A
1350 C TRASLACIONALES
1351 C
1352 DIMENSION TETA(LB1, LB1), RMAT(LB1, LB1), ALFA(NB), NI(NB), NJ(NB),
1353 @ RLON(NB), IT(ND)
1354 DO 1 I=1, NN
1355 KH=NGL-MM+I
1356 C
1357 C CHEQUEO DE BARRAS
1358 C
1359 DO 1 K=1, NB
1360 U=0
1361 V=0
1362 U1=0
1363 V1=0
1364 U2=0
1365 V2=0
1366 II=NI(K)
1367 JJ=NJ(K)
1368 CALL POS(IT, ND, II, M, N)
1369 CALL POS(IT, ND, JJ, M1, N1)
1370 IF(M1.NE.0) U2=RMAT(M1, KH)
1371 IF(M.NE.0) U1=RMAT(M, KH)
1372 U=U1-U2
1373 IF(N.NE.0) V1=RMAT(N, KH)
1374 IF(N1.NE.0) V2=RMAT(N1, KH)
1375 V=V1-V2
1376 TETA(I, 2*K-1)=-{(U*SIN(ALFA(K))-V*COS(ALFA(K)))/RLON(K)}
1377 TETA(I, 2*K-1)=TOLE(TETA(I, 2*K-1), TOL)
1378 TETA(I, 2*K)=TETA(I, 2*K-1)
1379 1 CONTINUE
1380 RETURN
1381 END
```

FTN4X COMPILER: HP92834 REV.2130 (810716)

\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: 333 COMMON: (NONE)

```

1382 PROGRAM TECAN(5)
1383 C
1384 C SEGMENTO PARA EL CALCULO DEL VECTOR TRABAJO EN CADA MECANISMO
1385 C INDEPENDIENTE.
1386 C
1387 COMMON /VAR/ IN, IO, NOMB(30), ND, NB, INDT, NMI, NLI, NGL, NAI, NBCU, NMH,
1388 @ NFI, NGD, NGP, NCL, LHF, LHV, NITER, NND, NH, NBB, NTCPV, NCFP,
1389 @ NCPV, NMIN, NGH, NMCO, NMHCO, NECO, NTRA, LV, CADI, SUMF, E,
1390 @ SUM, TOL, TOLS, SUM1, PESO, SIGMA, COP, COL, COI, COM, TCO
1391 COMMON /CON/ IT(25), RMP(25), RIMON(50), RMOME(50), NI(25), NJ(25),
1392 @ ALFA(25), RLON(25), RINER(25), RMP1(25), AXIL(25), XA(25),
1393 @ YA(25), ZA(25), DCL(20), R(75), LBE(20), T(50), LICOL(50),
1394 @ NCOL(50), NFILA(50), ICOL(50), DCBU(100), T1(50), Q(50),
1395 @ NGB(25), NTP(25), LTP(25), AREA(25), T2(50), CFI(50),
1396 @ CMH(50), CMV(50), FIFI(25), X(25), Y(25), Z(25),
1397 @ RLCPV(10,2), TETA(50,50), RMAT(50,50), TCNP(50,10),
1398 @ TFI(25,10)
1399 COMMON /PAS/ N1, N2, N3, N4, COLAA, COLAF, RMINC(50), RFIC(50), MC(50),
1400 @ RLOF(25), NECNP, NENP, COTA(10), ROV(25,6)
1401 COMMON /PAS2/ NX(4), RCA(75)
1402 LB1=50
1403 IF(NCL.EQ.12.OR.NCL.EQ.16) NENP=1
1404 IN=100
1405 C
1406 C LLAMADA A LA RUTINA QUE GENERA EL VECTOR O LA MATRIZ DE TRABAJO
1407 C
1408 IF(NCL.NE.12.AND.NCL.NE.13.AND.NCL.NE.16.AND.NCL.NE.22) GO TO 130
1409 GO TO 131
1410 130 DO 132 I=1,NND
1411 RCA(I)=R(I)
1412 132 CONTINUE
1413 GO TO 10
1414 131 DO 1 I=1,NECNP
1415 DO 2 II=1,NND
1416 R(II)=0.0
1417 2 CONTINUE
1418 IF(NCL.EQ.13.OR.NCL.EQ.22) GO TO 3
1419 DO 4 II=1,NBB
1420 DCBU(II)=0.0
1421 4 CONTINUE
1422 3 DO 5 II=1,NMI
1423 T(II)=0.0
1424 T1(II)=0.0
1425 T2(II)=0.0
1426 5 CONTINUE
1427 CALL LEIND(R,NND,I,NECNP,NND,63,1)
1428 IF(NBCU.EQ.0.OR.NCL.EQ.13.OR.NCL.EQ.22) GO TO 10
1429 CALL LEIND(DCBU,NBB,I,NECNP,NBB,58,1)
1430 10 CALL FORCA (T,T1,RMAT,ALFA,RLON,DCBU,LBE,R,NI,NJ,DCL,IT,NMI,ND,NB,
1431 @ NBCU,INDT,LB1,NLI,NGL,NCL,FIFI,NND,NBB)
1432 IF(NCL.NE.12.AND.NCL.NE.13.AND.NCL.NE.16.AND.NCL.NE.22) GO TO 6
1433 DO 7 II=1,NMI
1434 TCNP(II,I)=T1(II)
1435 7 CONTINUE
1436 DO 101 II=1,NB

```

```
1437      TFI(II,I)=FIFI(II)
1438      FIFI(II)=0.0
1439 101    CONTINUE
1440 1      CONTINUE
1441      GO TO 8
1442 6      DO 9 I=1,NMI
1443      T2(I)=T1(I)
1444 9      CONTINUE
1445      IF(NCL.EQ.15.OR.NCL.EQ.14.OR.NCL.EQ.16.OR.NCL.EQ.23)
1446 @CALL LEEDI(TETA,LB1,LB1,NMI,NH,55,1)
1447 C
1448 C      LLAMADA AL SEGMENTO CORRESPONDIENTE SEGUN TIPO DE PROBLEMA
1449 C
1450 8      IF(NBCU.EQ.0) GO TO 15
1451      IF(NCL.NE.16) GO TO 107
1452      DO 106 I=1,NB
1453      FIFI(I)=TFI(I,1)
1454 106    CONTINUE
1455 107    CALL EXEC(8,5HITERA)
1456 15     NZ=NCL-9
1457      GO TO (13,12,12,12,12,13,12,100,100,100,16,14,12,16),NZ
1458 16     DO 456 I=1,NB
1459      T(I)=1.0
1460 456   CONTINUE
1461      CALL EXEC(8,5HDISEN)
1462 14     CALL LEEDI(TETA,LB1,LB1,NMI,NH,55,1)
1463      DO 26 I=1,NMI
1464      TETA(I,NH+1)=T(I)
1465 26    CONTINUE
1466      CALL LEEDI(TETA,LB1,LB1,NMI,NH+1,55,2)
1467      CALL EXEC(8,5HCOMUN)
1468 13     CALL EXEC(8,5HCOLAP)
1469 12     IF(NCL.NE.16) GO TO 102
1470      DO 103 I=1,NB
1471      FIFI(I)=TFI(I,1)
1472 103   CONTINUE
1473 102   CALL EXEC(8,5HPASO1)
1474 100   STOP 2222
1475      END
```

FTN4X COMPILER: HP92834 REV.2130 (810716)

\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: 541 COMMON: (NONE)

```

1476      SUBROUTINE FORCA (T,T1,RMAT,ALFA,RLON,DCBU,LBE,R,NI,NJ,DCL,IT,NMI,
1477      @          ND,NB,NBCU,INDT,LB1,NLI,NGL,NCL,FIFI,NND,NBB)
1478 C
1479 C      ESTA SUBRUTINA FORMA EL VECTOR DE TRABAJO
1480 C
1481      DIMENSION T(NMI),T1(NMI),RMAT(LB1,LB1),ALFA(NB),DCBU(NBB),IT(ND),
1482      @          DCL(NLI),LBE(NLI),NI(NB),NJ(NB),R(NND),RLON(NB),
1483      @          FIFI(NB)
1484 C
1485 C      CALCULO TRABAJO FUERZAS EXTERNAS DEBIDA A CARGAS EN NUDOS
1486 C
1487      DO 1 I=1,NMI
1488      T(I)=0.0
1489 1      CONTINUE
1490      LL=1
1491      DO 2 J=1,ND
1492      T(INDT+LL)=R(3*J)
1493      IF(IT(J).EQ.4) GO TO 2
1494      LL=LL+1
1495 2      CONTINUE
1496 C
1497 C      TRABAJO EN LAS LIBERTADES
1498 C
1499      NMN=LL-1
1500      IF(NLI.EQ.0) GO TO 3
1501      NCOM=NMN+INDT+1
1502      KL=0
1503      DO 4 I=NCOM,NCOM+NLI-1
1504      KL=KL+1
1505      T(I)=DCL(KL)
1506 4      CONTINUE
1507 C
1508 C      EN CASO DE ANALISIS DE 2 ORDEN DETERMINAR TRABAJO PARA EL CALCULO
1509 C      DE AXILES
1510 C
1511 3      L1=2*ND
1512      INDO=INDT
1513      IF(NCL.EQ.15.OR.NCL.EQ.14.OR.NCL.EQ.16.OR.NCL.EQ.23) INDO=INDT+NB
1514      CALL LEEDI(RMAT,LB1,LB1,INDO,L1,51,1)
1515      IF(NCL.NE.15.AND.NCL.NE.14.AND.NCL.NE.16.AND.NCL.NE.23) GO TO 10
1516      NN=INDT+1
1517      MM=INDT+NB
1518      CALL TRABA(RMAT,IT,FIFI,R,NJ,NI,RLON,ALFA,DCBU,LB1,NB,ND,NBCU,NN,
1519      @          MM,NND,NBB,NMI)
1520 C
1521 C      TRABAJO EN LOS MECANISMO TRASLACIONALES
1522 C
1523 10      IF(INDT.EQ.0) GO TO 11
1524      NN=1
1525      MM=INDT
1526      CALL TRABA(RMAT,IT,T,R,NJ,NI,RLON,ALFA,DCBU,LB1,NB,ND,NBCU,NN,MM,
1527      @          NND,NBB,NMI)
1528 11      DO 12 I=1,NMI
1529      T1(I)=T(I)
1530 12      CONTINUE

```

1531 RETURN  
1532 END

FTN4X COMPILER: HP92834 REV.2130 (810716)

\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: 370 COMMON: (NONE)

```

1533 SUBROUTINE TRABA(RMAT, IT, T, R, NJ, NI, RLO, ALFA, DCBU, LB1, NB, ND, NBCU,
1534 @ NN, MM, NND, NBB, NMI)
1535 C
1536 C RUTINA PARA EL CALCULO DE LOS TERMINOS DE TRABAJO
1537 C
1538 DIMENSION RMAT(LB1, LB1), ALFA(NB), DCBU(NBB), IT(ND), NI(NB), NJ(NB),
1539 @ R(NND), T(NB), RLO(NB)
1540 C
1541 C TRABAJO EN LOS MECANISMO TRASLACIONALES
1542 C
1543 DO 1 I=NN, MM
1544 DO 2 K=1, 3*ND, 3
1545 KK=(K-1)/3+1
1546 T(I-NN+1)=T(I-NN+1)+R(K)*RMAT(I, 2*KK-1)+R(K+1)*RMAT(I, 2*KK)
1547 2 CONTINUE
1548 C
1549 C CALCULO DEL TRABAJO DEBIDO A CARGAS EN BARRAS
1550 C
1551 IF(NBCU.EQ.0) GO TO 1
1552 DO 3 K=1, NB
1553 NBV=0
1554 IF(IT(NJ(K)).NE.6) GO TO 3
1555 NBV=K
1556 I1=NI(NBV)
1557 I2=NJ(NBV)
1558 AL=ALFA(NBV)
1559 BETA=(RMAT(I, 2*I2)*COS(AL)-RMAT(I, 2*I2-1)*SIN(AL)-RMAT(I, 2*I1)*
1560 @ COS(AL)+RMAT(I, 2*I1-1)*SIN(AL))/RLO(NBV)
1561 A=DCBU(4*NBV-2)-DCBU(4*NBV-3)
1562 B=(DCBU(4*NBV-2))**2-(DCBU(4*NBV-3))**2
1563 T(I-NN+1)=T(I-NN+1)+(A*(DCBU(4*NBV)*RMAT(I, 2*I1-1)+DCBU(4*NBV-1)*
1564 @ RMAT(I, 2*I1))+0.5*BETA*B*(DCBU(4*NBV-1)*COS(AL)-
1565 @ DCBU(4*NBV)*SIN(AL))
1566 3 CONTINUE
1567 1 CONTINUE
1568 RETURN
1569 END

```

FTN4X COMPILER: HP92834 REV.2130 (810716)

\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: 474 COMMON: (NONE)

```

1570      PROGRAM ITERA(5)
1571 C
1572 C      SEGMENTO PARA LA FORMACION DE ECUACIONES ASOCIADAS A BARRAS
1573 C      CARGADAS UNIFORMENTE
1574 C
1575      COMMON /VAR/ IN, IO, NOMB(30), ND, NB, INDT, NMI, NLI, NGL, NAI, NBCU, NMH,
1576 @          NFI, NGD, NGP, NCL, LHF, LHV, NITER, NND, NH, NBB, NTCPV, NCFP,
1577 @          NCPV, NMIN, NGH, NMCO, NMHCO, NECO, NTRA, LV, CADI, SUMF, E,
1578 @          SUM, TOL, TOLS, SUM1, PESO, SIGMA, COP, COL, COI, COM, TCO
1579      COMMON /CON/ IT(25), RMP(25), RIMON(50), RMOME(50), NI(25), NJ(25),
1580 @          ALFA(25), RLON(25), RINER(25), RMP1(25), AXIL(25), XA(25),
1581 @          YA(25), ZA(25), DCL(20), R(75), LBE(20), T(50), LICOL(50),
1582 @          NCOL(50), NFILA(50), ICOL(50), DCBU(100), T1(50), Q(50),
1583 @          NGB(25), NTP(25), LTP(25), AREA(25), T2(50), CFI(50),
1584 @          CMH(50), CMV(50), FIFI(25), X(25), Y(25), Z(25),
1585 @          RLOPV(10,2), TETA(50,50), RMAT(50,50), TCNP(50,10),
1586 @          TFI(25,10)
1587      COMMON /PAS/ N1, N2, N3, N4, COLAA, FINAL, COR(50), RFIC(50), MC(50),
1588 @          RLOF(25), NECNP, NENP, COTA(10), ROV(25,6)
1589      COMMON /PAS2/ NX(4), RCA(75)
1590 C
1591 C      PUESTA A CERO DE CONJUNTOS
1592 C
1593      LB1=50
1594      DO 1 I=1, LB1
1595          ICOL(I)=0
1596          NCOL(I)=0
1597          NFILA(I)=0
1598          LICOL(I)=0
1599          RMOME(I)=0
1600          T(I)=0
1601          RIMON(I)=0
1602 1      CONTINUE
1603          CALL ZERO(RMAT, LB1, LB1, NH, NH)
1604          CALL ZERO(TETA, LB1, LB1, NH, NH)
1605          DO 21 I=1, NND
1606              R(I)=RCA(I)
1607 21      CONTINUE
1608 C
1609 C      LLAMADA A LA RUTINA DE FORMACION ECUACIONES ASOCIADAS A BARRAS
1610 C
1611          NITER=NITER+1
1612          CALL ITER (TETA, DCBU, NBCU, NMI, NB, ND, NI, NJ, IT, NH, RLON, NITER, LB1,
1613 @          ALFA, RMAT, NCL, NBB, NECNP, FIFI, TFI, INDT, NENP, R, NND)
1614 C
1615 C      LLAMADA A LA RUTINA DE FORMACION DEL VECTOR DE TRABAJO EN BARRAS
1616 C      CARGADAS UNIFORMEMENTE
1617 C
1618          IF(NCL.NE.12.AND.NCL.NE.16) GO TO 2
1619          DO 3 I=1, NECNP
1620              DO 4 II=1, NBB
1621                  DCBU(II)=0.0
1622 4          CONTINUE
1623          DO 5 II=1, NMI
1624              T1(II)=0.0

```

```

1625      T(II)=0.0
1626 5     CONTINUE
1627      CALL LEIND(DCBU,NBB,I,NECNP,NBB,58,1)
1628      DO 6 II=1,NMI
1629      T1(II)=TCNP(II,I)
1630 6     CONTINUE
1631 2     CALL FORCU(T,T1,DCBU,NBCU,NMI,NB,ND,NI,NJ,IT,NH,RLON,NITER,LB1,
1632 @      ALFA,COR,NBB,RMAT,T2)
1633      IF(NCL.NE.12.AND.NCL.NE.16) GO TO 7
1634      DO 8 II=1,NMI
1635      TCNP(II,I)=T1(II)
1636 8     CONTINUE
1637 3     CONTINUE
1638 7     IF(NCL.NE.13.AND.NCL.NE.22) GO TO 15
1639      DO 16 II=1,NMI
1640      TCNP(II,1)=T1(II)
1641 16    CONTINUE
1642 15    IF(NCPV.NE.0) GO TO 9
1643      DO 10 I=1,NMI
1644      IF(NCL.EQ.12.OR.NCL.EQ.16) T1(I)=TCNP(I,1)
1645 10    CONTINUE
1646 C     .
1647 C     LLAMADA AL SEGMENTO CORRESPONDIENTE
1648 C     .
1649 9     NV=NCL
1650      IF(NCL.EQ.10.OR.NCL.EQ.15.OR.(NCL.EQ.23.AND.IN.NE.100)) GO TO 13
1651      IF(NV.NE.11.AND.NV.NE.12.AND.NV.NE.13.AND.NV.NE.14.AND.NV.NE.16)
1652 @GO TO 11
1653      CALL EXEC(8,5HPAS01)
1654 11    IF(FINAL.NE.2.AND.FINAL.NE.1) GO TO 35
1655      IF(NCL.EQ.10.OR.NCL.EQ.15.OR.NCL.EQ.23) GO TO 13
1656 35    IF(NCL.NE.20.AND.NCL.NE.23) STOP 2222
1657      DO 456 I=1,NB
1658      T(I)=1.0
1659 456   CONTINUE
1660      IN=1
1661      CALL EXEC(8,5HDISEN)
1662 13    IN=1
1663      CALL EXEC(8,5HCOLAP)
1664      END

```

FTN4X COMPILER: HP92834 REV.2130 (810716)

\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: 515 COMMON: (NONE)



```

1665      SUBROUTINE ITER(TETA,DCBU,NBCU,NMI,NB,ND,NI,NJ,IT,NH,RLON,NITER,
1666 @          LB1,ALFA,RMAT,NCL,NBB,NECNP,FIFI,TFI,INDT,NENP,R,
1667 @          NND)
1668 C
1669 C      CORRECCION DE LA MATRIZ DE EQUILIBRIO
1670 C
1671      DIMENSION TETA(LB1,LB1),IT(ND),DCBU(NBB),RLON(NB),ALFA(NB),NI(NB),
1672 @          NJ(NB),RMAT(LB1,LB1),FIFI(NB),TFI(NB,NECNP),R(NND)
1673      COMMON /VAR/ IN,IO
1674 C
1675 C      OPERACIONES CON FICHEROS
1676 C
1677      CALL LEEDI(TETA,LB1,LB1,NMI,NH,55,1)
1678      IF(NITER.EQ.1) GO TO 2
1679      DO 10 I=1,ND
1680      IF(IT(I).NE.6) GO TO 10
1681      DO 11 J=1,NB
1682      IF(NJ(J).NE.I) GO TO 11
1683      RLON(J)=DCBU(4*J-2)
1684      RLON(J+1)=DCBU(4*J+2)
1685      GO TO 10
1686 11      CONTINUE
1687 10      CONTINUE
1688 2      K=NMI-NBCU/2
1689      J=1
1690 66      IF(IT(NJ(J)).NE.6) GO TO 6
1691      K=K+1
1692      SUL=RLON(J)+RLON(J+1)
1693      A=DCBU((J-1)*4+2)/(SUL-DCBU((J-1)*4+2))
1694      TETA(K,2*J-1)=-1
1695      TETA(K,2*J)=-1
1696      TETA(K,2*J+1)=A
1697      TETA(K,2*J+2)=A
1698      J=J+1
1699 6      J=J+1
1700      IF(J.GT.NB) GO TO 100
1701      GO TO 66
1702 100      CALL LEEDI(TETA,LB1,LB1,NMI,NH,55,2)
1703      IF(NCL.NE.15.AND.NCL.NE.14.AND.NCL.NE.16.AND.NCL.NE.23) RETURN
1704      LL1=2*ND
1705      INDO=INDT+NB
1706      CALL LEEDI(RMAT,LB1,LB1,INDO,LL1,51,1)
1707      IF(NCL.NE.16) GO TO 50
1708      DO 51 I=1,NECNP
1709      DO 52 J=1,NBB
1710      DCBU(J)=0.0
1711 52      CONTINUE
1712      CALL LEIND(R,NND,I,NECNP,NND,63,1)
1713      CALL LEIND(DCBU,NBB,I,NECNP,NBB,58,1)
1714 50      NN=INDT+1
1715      MM=INDT+NB
1716      DO 58 IK=1,NB
1717      FIFI(IK)=0.0
1718 58      CONTINUE
1719 C

```

```
1720 C      MODIFICACION DE LOS DESPLAZAMIENTOS VIRTUALES EN NUDOS VARIABLES
1721 C
1722      DO 70 IL=MM,NN,-1
1723      J=1
1724 76      I1=NJ(J)
1725      IF(IT(I1).NE.6) GO TO 75
1726      I2=NI(J)
1727      I3=NI(J+1)
1728      UI=RMAT(IL,2*I2-1)
1729      UJ=RMAT(IL,2*I3-1)
1730      VI=RMAT(IL,2*I2)
1731      VJ=RMAT(IL,2*I3)
1732      VZ=RMAT(IL,2*I1)
1733      UZ=RMAT(IL,2*I1-1)
1734      SUL=RLON(J)+RLON(J+1)
1735      AL=ALFA(J)
1736      DETA=((VJ-VI)*COS(AL)-(UJ-UI)*SIN(AL))/SUL
1737      ALA=(UZ-UI)*COS(AL)+(VZ-VI)*SIN(AL)
1738      VZ1=VI+DETA*RLON(J)*COS(AL)+ALA*SIN(AL)
1739      UZ1=UI-DETA*RLON(J)*SIN(AL)+ALA*COS(AL)
1740      RMAT(IL,2*I1-1)=UZ1
1741      RMAT(IL,2*I1)=VZ1
1742      J=J+1
1743 75      J=J+1
1744      IF(J.GT.NB) GO TO 70
1745      GO TO 76
1746 70      CONTINUE
1747      CALL TRABA(RMAT,IT,FIFI,R,NJ,NI,RLON,ALFA,DCBU,LB1,NB,ND,NBCU,
1748 @          NN,MM,NND,NBB,NMI)
1749      IF(NCL.NE.16) GO TO 110
1750      DO 53 II=1,NB
1751      TFI(II,I)=FIFI(II)
1752      FIFI(II)=0.0
1753 53      CONTINUE
1754 51      CONTINUE
1755      DO 54 I=1,NB
1756      FIFI(I)=TFI(I,NENP)
1757 54      CONTINUE
1758      CALL LEIND(DCBU,NBB,NENP,NECNP,NBB,58,1)
1759 110     CALL LEEDI(TETA,LB1,LB1,NMI,NH,55,1)
1760      RETURN
1761      END
```

FTN4X COMPILER: HP92834 REV.2130 (810716)

\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: 878 COMMON: (NONE)

```

1762      SUBROUTINE FORCU(T,T1,DCBU,NBCU,NMI,NB,ND,NI,NJ,IT,NH,RLON,NITER,
1763      @          LB1,ALFA,COR,NBB,RMAT,T2)
1764 C
1765 C      CORRECCION DEL VECTOR DE TRABAJO
1766 C
1767      DIMENSION IT(ND),T(NMI),T1(NMI),DCBU(NBB),RLON(NB),NI(NB),NJ(NB),
1768      @          ALFA(NB),COR(NB),RMAT(LB1,LB1),T2(NMI)
1769      IF(NITER.EQ.1) GO TO 2
1770      DO 10 I=1,ND
1771      IF(IT(I).NE.6) GO TO 10
1772      DO 11 J=1,NB
1773      IF(NJ(J).NE.I) GO TO 11
1774      RLON(J)=DCBU(4*J-2)
1775      RLON(J+1)=DCBU(4*J+2)
1776      GO TO 10
1777 11      CONTINUE
1778 10      CONTINUE
1779 2      K=NMI-NBCU/2
1780      J=1
1781 66      IF(IT(NJ(J)).NE.6) GO TO 6
1782      K=K+1
1783      T1(K)=0
1784      SUL=RLON(J)+RLON(J+1)
1785      ZL=DCBU(4*J-3)
1786      BETA=RLON(J)**2-ZL**2
1787      BETA1=SUL-RLON(J)
1788      T1(K)=0.5*(BETA+RLON(J)*(BETA1**2-DCBU(4*J+1)**2)/BETA1)*
1789      @      (-DCBU(4*J)*SIN(ALFA(J))+DCBU(4*J-1)*COS(ALFA(J)))
1790      T2(K)=T1(K)
1791      J=J+1
1792 6      J=J+1
1793      IF(J.GT.NB) GO TO 100
1794      GO TO 66
1795 100     DO 456 I=1,NMI
1796 456     T(I)=T1(I)
1797      RETURN
1798      END

```

FTN4X COMPILER: HP92834 REV.2130 (810716)

\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: 393 COMMON: (NONE)

```

1799      PROGRAM COLAP(5)
1800 C
1801 C      SEGMENTO PARA EL CALCULO DE LA CARGA DE COLAPSO.
1802 C
1803      COMMON /VAR/ IN, IO, NOMB(30), ND, NB, INDT, NMI, NLI, NGL, NAI, NBCU, NMH,
1804 @          NFI, NGD, NGP, NCL, LHF, LHV, NITER, NND, NH, NBB, NTCPV, NCFP,
1805 @          NCPV, NMIN, NGH, NMCO, NMHCO, NECO, NTRA, LV, CADI, SUMF, E,
1806 @          SUM, TOL, TOLS, SUM1, PESO, SIGMA, COP, COL, COI, COM, TCO
1807      COMMON /CON/ IT(25), RMP(25), RIMON(50), RMOME(50), NI(25), NJ(25),
1808 @          ALFA(25), RLON(25), RINER(25), RMP1(25), AXIL(25), XA(25),
1809 @          YA(25), ZA(25), DCL(20), R(75), LBE(20), T(50), LICOL(50),
1810 @          NCOL(50), NFILA(50), ICOL(50), DCBU(100), T1(50), Q(50),
1811 @          NGB(25), NTP(25), LTP(25), AREA(25), T2(50), CFI(50),
1812 @          CMH(50), CMV(50), FIFI(25), X(25), Y(25), Z(25),
1813 @          RLCPV(10, 2), TETA(50, 50), RMAT(50, 50)
1814      COMMON /PAS/ NMC, NC, NETAPA, COLAA, COLAP, RMINC(50), RFIC(50), MC(50),
1815 @          RLOF(25), NECNF, NENP, COTA(10)
1816 C
1817 C      PUESTA A CERO
1818 C
1819      IF(NCL.NE.15.AND.NCL.NE.23) GO TO 15
1820      DO 10 I=1, NH
1821      NFILA(I)=0
1822      LICOL(I)=0
1823      RMOME(I)=0.0
1824      RIMON(I)=0.0
1825      NCOL(I)=0
1826      ICOL(I)=0
1827 10  CONTINUE
1828 15  LB1=50
1829      NMIN=NMI
1830      IF(ISSW(0).GE.0) GO TO 41
1831 C
1832 C      IMPRESION DE LA MATRIZ DE ENTRADA (OPCIONAL)
1833 C
1834      CALL ESCRM(TETA, T, LB1, LB1, NMI, NMI, NH, 0)
1835 C
1836 C      LLAMADA A LA RUTINA DE CALCULO DE CARGA DE COLAPSO
1837 C
1838 41  CALL LOPI(TETA, T, RIMON, RMP, NMI, NH, SUM, NB, IO, NH, NCOL, LB1, TOL,
1839 @          TOLS, NFILA, RMOME, RFIC, NMIN, NGH, NMCO, NMHCO, NECO, NTRA, LV)
1840      IF(ISSW(1).GE.0) GO TO 42
1841 C
1842 C      IMPRESION DE LA MATRIZ DE SALIDA (OPCIONAL)
1843 C
1844      CALL ESCRM(TETA, T, LB1, LB1, NMI, NMI, NH, 0)
1845      DO 11 I=1, NH
1846      IF(NFILA(I).EQ.0) GO TO 11
1847      WRITE(IO, 12) I
1848 11  CONTINUE
1849 C
1850 C      DETECCION DEL FINAL
1851 C
1852 42  IF(NBCU.EQ.0) GO TO 1
1853      IF(SUM1.LT.TOL) SUM1=1

```

```
1854      PO=ABS(SUM-SUM1)/SUM1
1855      IF(PO.GT.0.001) GO TO 2
1856      NFI=1
1857 2     SUM1=SUM
1858 C
1859 C     LLAMADA AL SEGMENTO CORRESPONDIENTE
1860 C
1861 1     CALL EXEC(8,5HINFOR)
1862 C
1863 C     FORMATOS DE ESCRITURA
1864 C
1865 12    FORMAT(5X,"LA FILA ",I3," DE LA MATRIZ ES DE COLAPSO")
1866      END
```

FTN4X COMPILER: HP92834 REV.2130 (810716)

\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: 227 COMMON: (NONE)

```

1867 SUBROUTINE LOPI(A,B,RIMON,RMP,N,M,ROLA,NB,IO,NH,MAYOR,LB1,
1868 @ TOL,TOLS,NFILA,RMOME,RMP1,NMIN,NGH,NMCO,NMHCO,
1869 @ NECO,NTRA,LV)
1870 C
1871 C SUBROUTINA PARA EL CALCULO DE LA CARGA DE COLAPSO
1872 C
1873 DIMENSION A(LB1,LB1),B(N),RIMON(NH),RMP(NB),MAYOR(N),RMP1(NH),
1874 @ B1(50,1),NFILA(NH),RMOME(NH)
1875 DO 1 I=1,NB
1876 RMP1(2*I-1)=RMP(I)
1877 RMP1(2*I)=RMP(I)
1878 1 CONTINUE
1879 DO 5 I=1,N
1880 B1(I,1)=B(I)
1881 5 CONTINUE
1882 CALL PRE(A,B1,MAYOR,LB1,N,M,N,TOL,IO)
1883 CALL COLAI(A,B1,MAYOR,RIMON,ROLA,IO,NH,TOL,RMP1,RMP1,N,LB1,M,N,
1884 @ TOLS,NFILA,RMOME,NB,NMIN,NGH,NMCO,NMHCO,NECO,NTRA,LV)
1885 DO 8 KL=1,N
1886 B(KL)=B1(KL,1)
1887 8 CONTINUE
1888 RETURN
1889 END

```

FTN4X COMPILER: HP92834 REV.2130 (810716)

\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: 308 COMMON: (NONE)

```

1890 SUBROUTINE PRE(A,B1,MAYOR,LB1,N,M,NHC,TOL,IO)
1891 C
1892 C PREPARACION DE LA MATRIZ DEL S.V.A.
1893 C
1894 DIMENSION A(LB1,LB1),B1(NHC,1),MAYOR(NHC)
1895 DO 1 I=1,N
1896 MAYOR(I)=1
1897 PIVOT=A(I,1)
1898 DO 2 J=2,M
1899 IF(ABS(PIVOT).GT.ABS(A(I,J))) GO TO 2
1900 MAYOR(I)=J
1901 PIVOT=A(I,J)
1902 2 CONTINUE
1903 IF(ABS(PIVOT).LT.TOL) GO TO 1000
1904 DO 3 J=1,M
1905 A(I,J)=A(I,J)/PIVOT
1906 A(I,J)=TOLE(A(I,J),TOL)
1907 3 CONTINUE
1908 B1(I,1)=B1(I,1)/PIVOT
1909 B1(I,1)=TOLE(B1(I,1),TOL)
1910 C
1911 C LLAMADA A LA RUTINA DE PIVOTAMIENTO
1912 C
1913 CALL PIV(A,I,MAYOR(I),B1,N,M,LB1,NHC,TOL,LB1,1,1)
1914 DO 15 II=1,N
1915 B1(II,1)=TOLE(B1(II,1),TOL)
1916 DO 15 JJ=1,M
1917 A(II,JJ)=TOLE(A(II,JJ),TOL)
1918 15 CONTINUE
1919 1 CONTINUE
1920 RETURN
1921 1000 STOP 3330
1922 END

```

FTN4X COMPILER: HP92834 REV.2130 (810716)

\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: 280 COMMON: (NONE)

```

1923      SUBROUTINE COLAI(A,B1,MAYOR,RIMON,ROLA,IG,NH,TOL,RMP1,RMP2,NHC,
1924      @          LB1,M,N,TOLS,NFILA,RMOME,NB,NMIN,NGH,NMCO,NMHCO,
1925      @          NECO,NTRA,LV)
1926 C
1927 C      ALGORITMO DEL S.V.A.
1928 C
1929      DIMENSION A(LB1,LB1),B1(NHC,1),MAYOR(NHC),RIMON(NH),RMP1(NH),
1930      @          ZCON(50),NFILA(NH),RG(50),RMOME(NH),RMP2(NH)
1931      ROLA=0.
1932      LCA=0
1933      DO 5 J=1,M
1934      RIMON(J)=0.0
1935 5      CONTINUE
1936 500    DCOL=1.E36
1937      LFI=0
1938      DO 6 K=1,N
1939      RG(K)=0.0
1940      L=MAYOR(K)
1941      NN=L/2
1942      R=L/2.
1943      IF(ABS(NN-R).GT.0.00001) NN=NN+1
1944      IF(ABS(B1(K,1)).LT.TOL) GO TO 16
1945      IF(B1(K,1).GT.0) GO TO 7
1946      DCOL1=(-RMP1(L)-RIMON(L))/B1(K,1)
1947      RG(K)=DCOL1
1948      DCOL1=TOLE(DCOL1,TOL)
1949      IF(DCOL1.LT.0) DCOL1=0
1950      GO TO 8
1951 7      DCOL1=(RMP2(L)-RIMON(L))/B1(K,1)
1952      DCOL1=TOLE(DCOL1,TOL)
1953      RG(K)=DCOL1
1954      IF(DCOL1.LT.0) DCOL1=0.0
1955 8      IF(DCOL1.GT.DCOL) GO TO 6
1956      DCOL=DCOL1
1957      LFI=K
1958      GO TO 6
1959 16    RG(K)=1.E36
1960 6      CONTINUE
1961      ROLA=ROLA+DCOL
1962      DO 9 J=1,N
1963      L=MAYOR(J)
1964      RIMON(L)=RIMON(L)+B1(J,1)*DCOL
1965      RIMON(L)=TOLE(RIMON(L),TOL)
1966 9      CONTINUE
1967 50    DO 10 J=1,M
1968      ZCON(J)=0.0
1969 10    CONTINUE
1970      DO 20 I=1,M
1971      IF(ABS(RIMON(I)).LE.TOLS) GO TO 21
1972      IF(ABS(RIMON(I)-RMP2(I)).LE.TOLS) GO TO 22
1973      IF(ABS(RIMON(I)+RMP1(I)).LE.TOLS) GO TO 23
1974      GO TO 354
1975 21    ZCON(I)=ABS(A(LFI,I))
1976      GO TO 354
1977 22    ZCON(I)=-A(LFI,I)*ABS(B1(LFI,1))/B1(LFI,1)

```



```
1978      GO TO 354
1979 23    ZCON(I)=A(LFI,I)*ABS(B1(LFI,1))/B1(LFI,1)
1980 354    ZCON(I)=TOLE(ZCON(I),TOL)
1981 20     CONTINUE
1982      LCO=0
1983      MA=1
1984      ZMA=ZCON(1)
1985      DO 30 J=1,M
1986      IF(ZCON(J).LE.TOL) GO TO 30
1987      IF(ZCON(J).LT.ZMA) GO TO 30
1988      LCO=1
1989      MA=J
1990      ZMA=ZCON(J)
1991 30     CONTINUE
1992      IF(LCA.EQ.1) GO TO 54
1993      IF(LCO.EQ.0) GO TO 2000
1994      PIVOT=A(LFI,MA)
1995      DO 31 J=1,M
1996      A(LFI,J)=A(LFI,J)/PIVOT
1997      A(LFI,J)=TOLE(A(LFI,J),TOL)
1998 31     CONTINUE
1999      B1(LFI,1)=B1(LFI,1)/PIVOT
2000      B1(LFI,1)=TOLE(B1(LFI,1),TOL)
2001      MAYOR(LFI)=MA
2002 C
2003 C      LLAMADA A LA RUTINA DE PIVOTAMIENTO
2004 C
2005      KJM=KJM+1
2006      CALL PIV(A,LFI,MA,B1,N,M,LB1,NHC,TOL,LB1,1,1)
2007      DO 801 II=1,N
2008      B1(II,1)=TOLE(B1(II,1),TOL)
2009      DO 801 JJ=1,M
2010      A(II,JJ)=TOLE(A(II,JJ),TOL)
2011 801     CONTINUE
2012      GO TO 500
2013 2000    NECO=0
2014      DO 653 I=1,N
2015      TA=DCOL*TOLS
2016      IF(DCOL.GT.1.) TA=TOLS/DCOL
2017      IF(ISSW(8).LT.0) WRITE(IO,8543) I,DCOL,RG(I),TOLS
2018      RG(I)=TOLE(RG(I),TOL)
2019      IF(ABS(DCOL-ABS(RG(I))).LE.TA) NFILA(I)=1
2020      RG(I)=0.0
2021 653     CONTINUE
2022      LCA=1
2023      IK=1
2024 C
2025 C      OBTENCION DE LAS SECCIONES QUE INTERVIENEN EN EL MECANISMO
2026 C      DE COLAPSO
2027 C
2028 51     IF(NFILA(IK).EQ.0) GO TO 52
2029      LFI=IK
2030      GO TO 50
2031 54     IF(LCO.EQ.1) NFILA(IK)=0
2032 52     IK=IK+1
```

```
2033      IF(IK.GT.N) GO TO 53
2034      GO TO 51
2035 53    DO 58 I=1,M
2036      RG(I)=0.0
2037 58    CONTINUE
2038      DO 55 I=1,N
2039      IF(NFILA(I).NE.1) GO TO 55
2040      NECO=NECO+1
2041      DO 56 J=1,M
2042      IF(ABS(A(I,J)).GT.TOL) RG(J)=1
2043 56    CONTINUE
2044 55    CONTINUE
2045      NN=0
2046      DO 57 I=1,M
2047      RMOE(I)=RIMON(I)
2048      IF(RG(I).EQ.1) GOTO 57
2049      RIMON(I)=1.E36
2050      NN=NN+1
2051 57    CONTINUE
2052      NMCO=M-NN
2053      RETURN
2054 1000  STOP 3331
2055 C
2056 C      FORMATOS DE ESCRITURA
2057 C
2058 8543  FORMAT(10X,"Fila,Incr. de carga,Incr. de fila,Tol",I5,3E14.5)
2059      END
```

FTN4X COMPILER: HP92834 REV.2130 (810716)

\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: 1272 COMMON: (NONE)

```

2060      PROGRAM INFOR(5)
2061 C
2062 C      SEGMENTO PARA DETERMINAR LOS VIRTUALES A PARTIR DE LA MATRIZ
2063 C      FINAL DEL S.V.A Y FORMACION DE LA MATRIZ DEL SISTEMA
2064 C
2065      DIMENSION LCO(50)
2066      COMMON /VAR/ IN, IO, NOMB(30), ND, NB, INDT, NMI, NLI, NVIR, NAI, NBCU, NMH,
2067 @          NFI, NGD, NGP, NCL, LHF, LHV, NITER, NND, NH, NBB, NTCPV, NCFP,
2068 @          NCPV, NMIN, NGH, NMCO, NMHCO, NECO, NTRA, LV, CADI, SUMF, E,
2069 @          SUM, TOL, TOLS, SUM1, PESO, SIGMA, COP, COL, COI, COM, TCO
2070      COMMON /CON/ IT(25), RMP(25), RIMON(50), RMOME(50), NI(25), NJ(25),
2071 @          ALFA(25), RLON(25), RINER(25), RMP1(25), AXIL(25), XA(25),
2072 @          YA(25), ZA(25), DCL(20), R(75), LBE(20), T(50), LICOL(50),
2073 @          NCOL(50), NFILA(50), ICOL(50), DCBU(100), T1(50), Q(50),
2074 @          NGB(25), NTP(25), LTP(25), AREA(25), T2(50), CFI(50),
2075 @          CMH(50), CMV(50), FIFI(25), X(25), Y(25), Z(25),
2076 @          RLCPV(10,2), TETA(50,50), RMAT(50,50)
2077      COMMON /PAS/ N1, N2, N3, N4, COLAA, COLAP, RMINC(50), RFIC(50), MC(50),
2078 @          RLOF(25), NECNP, NENP, COTA(10)
2079      LB1=50
2080 C
2081 C      LLAMADA A LA RUTINA DE GENERACION DE CONJUNTOS PARA EL CALCULO
2082 C      DE LOS VIRTUALES.
2083 C
2084 C      PONER A CERO LA MATRIZ
2085 C
2086      DO 1 I=1, NH
2087      LICOL(I)=0
2088      Q(I)=0
2089 1      CONTINUE
2090      CALL ZERO(RMAT, LB1, LB1, NH, NH)
2091      CALL VIRTU(NMI, NH, NCOL, ICOL, NVIR, LB1, RMAT, TETA, Q, TOL, SUM, RMOME,
2092 @          RIMON, T, NMH, LICOL)
2093      MCONO=0
2094      DO 10 I=1, NH
2095      DO 11 J=1, NVIR
2096      IF(ICOL(J), EQ, 0) GO TO 11
2097      IF(ABS(RMAT(J, I)), GT, TOL) GO TO 10
2098 11      CONTINUE
2099      MCONO=MCONO+1
2100      LCO(I)=1
2101 10      CONTINUE
2102      IF(ISSW(8), GE, 0) GO TO 12
2103      CALL ESCRM(RMAT, T, LB1, LB1, NMI, NVIR, NH, 2)
2104      WRITE(IO, 745)
2105      DO 101 I=1, NH
2106      WRITE(IO, 102) I, Q(I), ICOL(I), NCOL(I)
2107 101      CONTINUE
2108      WRITE(IO, 13) MCONO, NH-MCONO
2109      IF(MCONO, EQ, 0) GO TO 12
2110      DO 15 I=1, NH
2111      IF(LCO(I), EQ, 0) GO TO 15
2112      WRITE(IO, 16) I
2113 15      CONTINUE
2114 12      NGH=NVIR

```

```

2115      DO 2 I=1,NH
2116      DO 3 J=1,NMI
2117      IF(NCOL(J).NE.1) GO TO 3
2118      GO TO 2
2119 3     CONTINUE
2120      IF(RIMON(I).EQ.1.E36) NN=NN+1
2121 2     CONTINUE
2122      NMHCO=NVIR-NN
2123 C
2124 C     LLAMADA A LA RUTINA DE FORMACION DEL VECTOR DE CARGAS CONDENSADO
2125 C
2126      CALL FOMTI(RMAT,RIMON,RMOME,R,RLON,RINER,RMP,ICOL,DCBU,IT,ND,NB,
2127 @          LB1,NVIR,NMH,TOLS,1.,NH,Q,ALFA,SUM,NJ,NITER,NI,NBCU,
2128 @          TOLS,AXIL,E,COL,COI,COM,N4,NBB,NND)
2129 C
2130 C     LLAMADA A LA RUTINA DE FORMACION DE LA MATRIZ DEL SISTEMA
2131 C
2132      CALL FOMHI(TETA,RMAT,RIMON,RMOME,RLON,RINER,RMP,ICOL,ND,NB,LB1,
2133 @          NVIR,NMH,TOLS,1.,NH,NJ,NI,TOLS,NCOL,LV,NTRA,AXIL,E,
2134 @          COL,COI,COM,N4)
2135 C
2136 C     LLAMADA AL SEGMENTO CORRESPONDIENTE
2137 C
2138      CALL EXEC(8,5HROTAC)
2139 C
2140 C     FORMATOS DE ESCRITURA
2141 C
2142 765    FORMAT(10X,"LA MATRIZ ANTERIOR ES LA MATRIZ Bh"//)
2143 102    FORMAT(5X,"I,B,Mh,M",I5,E14.5,2I5)
2144 13     FORMAT(10X,"N. DE MOMENTOS CONOCIDOS.....",I5/
2145 @       10X,"N. DE MOMENTOS DESCONOCIDOS..",I5)
2146 16     FORMAT(5X,"LA SECCION ",I2," ES DE MOMENTO CONOCIDA")
2147      END

```

FTN4X COMPILER: HP92834 REV.2130 (810716)

\*\* NO WARNINGS \*\*    NO ERRORS \*\*    PROGRAM:    496    COMMON: (NONE)

```

2148 SUBROUTINE VIRTU(NMI,NH,NCOL,IC,NVIR,LB1,RMAT,TETA,Q,TOLS,SUM,
2149 @ RMOME,RIMON,T,NMH,LICOL)
2150 C
2151 C GENERACION AUTOMATICA DE CODIGOS PARA EL ORDENAMIENTO
2152 C DE LOS VIRTUALES.
2153 C
2154 DIMENSION IC(NH),NCOL(NH),RIMON(NH),RMOME(NH),
2155 @ TETA(LB1,LB1),RMAT(LB1,LB1),T(NMI),Q(NH),LICOL(NH)
2156 CALL FORVI(TETA,RMAT,NCOL,IC,LB1,LB1,LB1,LB1,NH,NH,NMI,NH,NVIR)
2157 DO 1 I=1,NH
2158 LICOL(I)=IC(I)
2159 1 CONTINUE
2160 DO 13 I=1,NH
2161 DO 16 K=1,NMI
2162 IF(NCOL(K).NE.I) GO TO 16
2163 Q(I)=T(K)*SUM
2164 GO TO 13
2165 16 CONTINUE
2166 13 CONTINUE
2167 NMH=0
2168 DO 18 I=1,NVIR
2169 DO 18 J=1,NH
2170 IF(RIMON(J).NE.1.E36) GO TO 19
2171 IF(IC(I).EQ.J) NMH=NMH+1
2172 GO TO 18
2173 19 IF(IC(I).NE.J)GO TO 18
2174 DO 20 KK=1,NH
2175 Q(KK)=Q(KK)+RIMON(IC(I))*RMAT(I,KK)
2176 20 CONTINUE
2177 IC(I)=0
2178 18 CONTINUE
2179 NMHI=NVIR-NMH
2180 DO 21 I=1,NVIR
2181 IF(IC(I).NE.0) RETURN
2182 21 CONTINUE
2183 RETURN
2184 END

```

FTN4X COMPILER: HP92834 REV.2130 (810716)

\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: 279 COMMON: (NONE)

```
2185     SUBROUTINE VIRT( NMI, NH, NCOL, IC, NVIR, LB1, RMAT, TETA, TOL, NDC,  
2186     @           W, L1, L2, L3, IT, NJ, L6, L7 )  
2187 C  
2188 C     SUBROUTINA PARA LA GENERACION DE LA MATRIZ DE VIRTUALES  
2189 C     A PARTIR DE LAS ECUACIONES DE EQUILIBRIO SIN TRIANGULAR  
2190 C  
2191     DIMENSION TETA(LB1, LB1), RMAT(LB1, LB1), NCOL(NH), IC(NH), W(L1, L2),  
2192     @           IT(L6), NJ(L7)  
2193     DO 1 I=1, NMI  
2194     CALL PRTRI( I, TETA, W, NCOL, LB1, LB1, L1, L2, NH, NH, L3, TOL, MAYOR,  
2195     @           IT, NJ, L6, L7 )  
2196     IF( NDC.EQ.0 ) GO TO 123  
2197     CALL PIV( TETA, I, MAYOR, W, NMI, NH, LB1, L1, TOL, LB2, L2, L3 )  
2198     GO TO 1  
2199 123     CALL TRIAN( TETA, I, MAYOR, TOL, NMI, NH, LB1, LB1 )  
2200 1     CONTINUE  
2201     CALL FORVI( TETA, RMAT, NCOL, IC, LB1, LB1, LB1, LB1, NH, NH, NMI, NH, NVIR )  
2202     RETURN  
2203     END
```

FTN4X COMPILER: HP92834 REV.2130 (810716)

\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: 165 COMMON: (NONE)

```

2204 SUBROUTINE PRTRI (I,TETA,B,NC,L1,L2,L3,L4,L5,N1,N2,TOL,MAYOR,IT,
2205 @ NJ,L6,L7)
2206 C
2207 C ELECCION DEL PIVOTE PARA QUE LOS VIRTUALES NO DEFENDAN DE LOS NUDDOS
2208 C MOVILES
2209 C
2210 DIMENSION TETA(L1,L2),B(L3,L4),NC(L5),IT(L6),NJ(L7)
2211 MAYOR=1
2212 AMAX=TETA(I,1)
2213 L=0
2214 DO 2 J=1,N1
2215 L=J-L
2216 R=J/2.
2217 M=J/2
2218 RR=R-M
2219 MM=NJ(L)
2220 IF (ABS(RR).GT.TOL) GO TO 6
2221 IF (IT(MM).EQ.6.AND.ABS(TETA(I,J)).GT.TOL) GOTO 5
2222 6 IF (ABS(TETA(I,J)).LT.ABS(AMAX)) GO TO 2
2223 5 MAYOR=J
2224 AMAX=TETA(I,J)
2225 IF (ABS(RR).GT.TOL) GO TO 2
2226 IF (IT(MM).EQ.6.AND.ABS(TETA(I,J)).GT.TOL) GOTO 7
2227 2 CONTINUE
2228 7 IF (ABS(AMAX).LT.TOL) GO TO 10
2229 DO 3 J=1,N1
2230 TETA(I,J)=TETA(I,J)/AMAX
2231 TETA(I,J)=TOLE(TETA(I,J),TOL)
2232 3 CONTINUE
2233 DO 4 J=1,N2
2234 B(I,J)=B(I,J)/AMAX
2235 B(I,J)=TOLE(B(I,J),TOL)
2236 4 CONTINUE
2237 NC(I)=MAYOR
2238 RETURN
2239 10 NC(I)=0
2240 RETURN
2241 END

```

FTN4X COMPILER: HP92834 REV.2130 (810716)

.\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: 343 COMMON: (NONE)

```
2242 SUBROUTINE FORVI(TETA,RMAT,NC,IC,L1,L2,L3,L4,L5,L6,N1,N2,NV)
2243 C
2244 C FORMACION DE LOS VIRTUALES
2245 C
2246 DIMENSION TETA(L1,L2),RMAT(L3,L4),NC(L5),IC(L6),MM(50)
2247 K=0
2248 DO 10 I=1,N1
2249 IF(NC(I).EQ.0) GO TO 10
2250 K=K+1
2251 10 CONTINUE
2252 NV=N2-K
2253 DO 20 K=1,N2
2254 MM(K)=K
2255 20 CONTINUE
2256 DO 30 K=1,N2
2257 DO 30 LH=1,N1
2258 IF(NC(LH).EQ.MM(K)) MM(K)=0
2259 30 CONTINUE
2260 K=1
2261 DO 40 J=1,N2
2262 IF(MM(J).EQ.0) GO TO 40
2263 IC(K)=J
2264 K=K+1
2265 40 CONTINUE
2266 DO 12 I=1,NV
2267 DO 11 J=1,N1
2268 RMAT(I,NC(J))=-TETA(J,IC(I))
2269 11 CONTINUE
2270 RMAT(I,IC(I))=1
2271 12 CONTINUE
2272 RETURN
2273 END
```

FTN4X COMPILER: HP92834 REV.2130 (810716)

\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: 258 COMMON: (NONE)



```

2274 SUBROUTINE FOMHI(TETA,RMAT,RIMON,RMOME,RLON,RINER,RMP,ICOL,ND,NB,
2275 @ LB1,NVIR,NMH,TOLS,E,NH,NJ,NI,TOLA,NFA,LL,NTRA,
2276 @ AXIL,EE,COL,COI,COM,N4)
2277 C
2278 C FORMACION DE LA MATRIZ BH^ * A * BH
2279 C
2280 DIMENSION TETA(LB1, LB1), RMAT(LB1, LB1), RIMON(NH), RMOME(NH), NJ(NB),
2281 @ RLON(NB), RINER(NB), RMP(NB), ICOL(NH), NI(NB), NFA(NH),
2282 @ AXIL(NB)
2283 DO 1 I=1, NH
2284 NFA(I)=0
2285 1 CONTINUE
2286 CALL ZERO(TETA, LB1, LB1, NH, NH)
2287 DO 2 I=1, NVIR
2288 RR=0
2289 K=0
2290 DO 3 II=1, NVIR
2291 IF(ICOL(II).EQ.0) GO TO 3
2292 K=K+1
2293 RR=0
2294 DO 4 J=1, NB
2295 P=RLON(J)/(6.*RINER(J)*E)
2296 CALL FLEX(AXIL(J), RLON(J), RINER(J), C1, C2, EE, N4, COM, COL, COI)
2297 RMOME(2*J-1)=P*(C1*RMAT(II, 2*J-1)-C2*RMAT(II, 2*J))
2298 RMOME(2*J)=P*(C1*RMAT(II, 2*J)-C2*RMAT(II, 2*J-1))
2299 4 CONTINUE
2300 DO 5 KK=1, NH
2301 RR=RR+RMAT(I, KK)*RMOME(KK)
2302 5 CONTINUE
2303 L=(ICOL(II)+1)/2
2304 TETA(I, K)=RR
2305 3 CONTINUE
2306 LL=NMH
2307 L=0
2308 DO 6 K=1, NH
2309 L=K-L
2310 COMPAR=ABS(ABS(RIMON(K))-RMP(L))
2311 IF((RIMON(K).GE.1.E30).OR.(COMPAR.GT.TOLA)) GO TO 6
2312 LL=LL+1
2313 NFA(LL-NMH)=K
2314 NSG=1
2315 IF(RIMON(K).LT.0) NSG=-1
2316 TETA(I, LL)=RMAT(I, K)*NSG
2317 IF(I.NE.NVIR) GO TO 6
2318 TETA(NVIR+1, LL)=RIMON(K)*NSG
2319 6 CONTINUE
2320 2 CONTINUE
2321 NTRA=LL-NMH
2322 RETURN
2323 END

```

\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: 514 COMMON: (NONE)

```

2324 SUBROUTINE FOMTI(RMAT,RIMON,RMOME,R,RLON,RINER,RMP,ICOL,DCBU,
2325 @ IT,ND,NB,LB1,NVIR,NMH,TOLS,E,NH,Q,ALFA,SUM,NJ,
2326 @ NITER,NI,NBCU,TOLA,AXIL,EE,COL,COI,COM,N4,NBB,
2327 @ NND)
2328 C
2329 C FORMACION DEL VECTOR BH^ * A *BP
2330 C
2331 DIMENSION RMAT(LB1, LB1), RIMON(NH), RMOME(NH), R(NND), AXIL(NB),
2332 @ RLON(NB), RINER(NB), RMP(NB), ICOL(NH), DCBU(NBB), ALFA(NB),
2333 @ Q(NH), Q1(54), NJ(NB), NI(NB), IT(ND)
2334 DO 1 I=1,NH
2335 RMOME(I)=0
2336 R(I)=0.
2337 1 CONTINUE
2338 C
2339 C FORMACION DEL SISTEMA DE COMPATIBILIDAD
2340 C
2341 DO 2 I=1,NVIR
2342 RR=0
2343 DO 3 J=1,NB
2344 F=RLON(J)/(6.*RINER(J)*E)
2345 CALL FLEX(AXIL(J),RLON(J),RINER(J),C1,C2,EE,N4,COM,COL,COI)
2346 Q1(2*J-1)=C1*F*Q(2*J-1)-P*C2*Q(2*J)
2347 Q1(2*J)=-C2*P*Q(2*J-1)+C1*F*Q(2*J)
2348 3 CONTINUE
2349 DO 4 J=1,NH
2350 R(I)=R(I)+RMAT(I,J)*Q1(J)
2351 4 CONTINUE
2352 K=0
2353 DO 5 II=1,NVIR
2354 IF(ICOL(II).EQ.0) GO TO 5
2355 K=K+1
2356 RR=0
2357 DO 6 J=1,NB
2358 F=RLON(J)/(6.*RINER(J)*E)
2359 CALL FLEX(AXIL(J),RLON(J),RINER(J),C1,C2,EE,N4,COM,COL,COI)
2360 RMOME(2*J-1)=F*(C1*RMAT(II,2*J-1)-C2*RMAT(II,2*J))
2361 RMOME(2*J)=F*(C1*RMAT(II,2*J)-C2*RMAT(II,2*J-1))
2362 6 CONTINUE
2363 DO 7 KK=1,NH
2364 RR=RR+RMAT(I, KK)*RMOME(KK)
2365 7 CONTINUE
2366 L=(ICOL(II)+1)/2
2367 R(I)=R(I)-RR*RMP(L)*10.
2368 5 CONTINUE
2369 IF(NBCU.EQ.0) GO TO 20
2370 C
2371 C SI HAY CARGAS UNIFORMES LE SUMAMOS EL APORTE DE ESTA
2372 C
2373 RS1=0.
2374 DO 8 IM=1,ND
2375 IF(IT(IM).NE.6) GO TO 8
2376 DO 9 J=1,NB
2377 IF(IM.NE.NJ(J)) GO TO 9
2378 DO 10 JK=J,J+1

```

```
2379      RL=RLON(JK)
2380      QW=(-DCBU(4*JK)*SIN(ALFA(JK))+DCBU(4*JK-1)*COS(ALFA(JK)))*SUM
2381      C=RL-DCBU(4*JK-3)
2382      TE1=2*C*RL**2-C**3
2383      TE1=TE1*QW*C/(24*RL*RINER(J)*E)
2384      TE2=-4.*C*RL**2-C**3+4*RL*C**2
2385      TE2=TE2*QW*C/(24.*RL*RINER(J)*E)
2386      IF(ABS(AXIL(JK)).LT.TOLA) GO TO 11
2387      YH=COM*COL/(EE*COI)
2388      TE1=TE1-QW*AXIL(JK)*YH*RLON(JK)**5/(240.*(E*RINER(JK))**2)
2389      TE2=TE2+QW*AXIL(JK)*YH*RLON(JK)**5/(240.*(E*RINER(JK))**2)
2390 11     RS1=RS1+TE1*RMAT(I,2*JK-1)+TE2*RMAT(I,2*JK)
2391 10     CONTINUE
2392      GO TO 8
2393 9      CONTINUE
2394 8      CONTINUE
2395 20     R(I)=R(I)+RS1
2396      R(I)=-R(I)
2397 2      CONTINUE
2398      RETURN
2399      END
```

FTN4X COMPILER: HP92834 REV.2130 (610716)

\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: 1142 COMMON: (NONE)

```
2400      SUBROUTINE FLEX(AX,RL,RI,C1,C2,E,N4,COM,COL,COI)
2401 C
2402 C      FORMACION DE LA SUBMATRIZ DE FLEXIBILIDAD
2403 C
2404      IF(N4.EQ.1) GO TO 10
2405      U=RL*COL*SQRT(ABS(AX*COM)/(E*RI*COI*COL))*0.5
2406      IF(ABS(AX).LE.0.000001) GO TO 1
2407      IF(AX.LT.0) GO TO 2
2408      TH=(EXP(2*U)-EXP(-2*U))/(EXP(2*U)+EXP(-2*U))
2409      SH=(EXP(2*U)-EXP(-2*U))*0.5
2410      SG=-1
2411      GO TO 4
2412 2      TH=SIN(2*U)/COS(2*U)
2413      SH=SIN(2*U)
2414      SG=1
2415 4      C1=SG*3.*(TH-2*U)/(TH*2*U**2)
2416      C2=SG*3.*(2*U-SH)/(2*SH*U**2)
2417      RETURN
2418 1      C1=2.
2419      C2=1.
2420      RETURN
2421 10     C1=2.
2422      C2=1.
2423      C=((AX*COM*(RL*COL)**2)/(E*RI*COI*COL))*6.
2424      C1=C1-C/45.
2425      C2=C2-C*7./360.
2426      RETURN
2427      END
```

FTN4X COMPILER: HP92834 REV.2130 (810716)

\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: 356 COMMON: (NONE)

```

2428 PROGRAM ROTAC(5)
2429 C
2430 C SEGMENTO PARA LA FORMACION DEL SISTEMA Y RESOLUCION.
2431 C DETERMINA LA SOLUCION DE MOMENTOS Y LAS ROTACIONES PLASTICAS
2432 C
2433 DIMENSION AA(50)
2434 COMMON /VAR/ IN, IO, NOMB(30), ND, NB, INDT, NMI, NLI, NVIR, NAI, NBCU, NMH,
2435 @ NFI, NGD, NGP, NCL, LHF, LHV, NITER, NND, NH, NBB, NTCPV, NCFP,
2436 @ NCPV, NMIN, NGH, NMCO, NMHCO, NECO, NTRA, LV, CADI, SUMF, E,
2437 @ SUM, TOL, TOLS, SUM1, PESO, SIGMA, COP, COL, COI, COM, TCO
2438 COMMON /CON/ IT(25), RMP(25), RIMON(50), RMOME(50), NI(25), NJ(25),
2439 @ ALFA(25), RLON(25), RINER(25), RMP1(25), AXIL(25), XA(25),
2440 @ YA(25), ZA(25), DCL(20), R(75), LBE(20), T(50), LICOL(50),
2441 @ NCOL(50), NFILA(50), ICOL(50), DCBU(100), T1(50), Q(50),
2442 @ NGB(25), NTP(25), LTP(25), AREA(25), T2(50), CFI(50),
2443 @ CMH(50), CMV(50), FIFI(25), X(25), Y(25), Z(25),
2444 @ RLCPV(10, 2), TETA(50, 50), RMAT(50, 50)
2445 COMMON /PAS/ N1, N2, N3, N4, COLAA, COLAP, RMIND(50), RFIC(50), MC(50),
2446 @ RLOF(25), NECNP, NENP, COTA(10)
2447 LB1=50
2448 CALL ECUA(TETA, RMAT, RIMON, RMOME, R, RLON, RINER, RMP, ICOL, DCBU, IT,
2449 @ ND, NB, LB1, NVIR, NMH, TOL, 1., NH, Q, ALFA, SUM, NJ, NITER,
2450 @ NCOL, NI, NBCU, TOLS, NTRA, LV, NND, NBB, LVUE, COM, E, COI, COL)
2451 C
2452 C SI SE DESEA SE COMPRUEBA EQUILIBRIO, COMPATIBILIDAD, SEGURIDAD
2453 C Y PARIDAD.
2454 C
2455 IF(ISSW(12).GE.0) GO TO 25
2456 C
2457 C COMPROBACION DE EQUILIBRIO
2458 C
2459 CALL LEEDI(TETA, LB1, LB1, NMI, NH, 55, 1)
2460 DO 60 I=1, NMI
2461 D=0
2462 DO 61 J=1, NH
2463 D=D+TETA(I, J)*RMOME(J)
2464 61 CONTINUE
2465 WRITE(IO, 62) I, D, T1(I)*SUM
2466 60 CONTINUE
2467 C
2468 C COMPROBACION DE LA COMPATIBILIDAD
2469 C
2470 DO 65 J=1, NB
2471 P=RLON(J)/(6.*RINER(J))
2472 CALL FLEX(AXIL(J), RLON(J), RINER(J), C1, C2, E, N4, COM, COL, COI)
2473 AA(2*J-1)=P*(C1*RMOME(2*J-1)-C2*RMOME(2*J))+R(2*J-1)
2474 AA(2*J)=P*(C1*RMOME(2*J)-C2*RMOME(2*J-1))+R(2*J)
2475 65 CONTINUE
2476 DO 66 I=1, NVIR
2477 D=0
2478 DO 67 J=1, NH
2479 D=D+RMAT(I, J)*AA(J)
2480 67 CONTINUE
2481 WRITE(IO, 68) I, D
2482 66 CONTINUE

```

```

2483 C
2484 C   COMPROBACION DE LA SEGURIDAD
2485 C
2486   LLKK=0
2487   DO 69 I=1,NH
2488   LLKK=I-LLKK
2489   WRITE(IO,70) I,ABS(RMOME(I))-RMP(LLKK)
2490 69   CONTINUE
2491 C
2492 C   COMPROBACION DE LA PARIDAD
2493 C
2494   L=0
2495   DO 71 I=1,NH
2496   L=I-L
2497   WRITE(IO,72) I,RMOME(I)*R(I),ABS(ABS(RMOME(I))-RMP(L))*R(I)
2498 71   CONTINUE
2499 25   IF(ISSW(8).GE.0) GO TO 21
2500   LVUE=LVUE-1
2501   WRITE(IO,22) LVUE
2502 21   IF(NBCU.EQ.0) GO TO 20
2503   IF(ISSW(9).GE.0) GO TO 20
2504   SD=SUM*COM/(COP*CDL)
2505   WRITE(IO,18) SD
2506   WRITE(IO,11)
2507   DO 12 I=1,NB
2508   RR=RMOME(2*I-1)*COM
2509   RR1=RMOME(2*I)*COM
2510   DIS=DCBU(4*I-2)*COL
2511   WRITE(IO,13) I,2*I-1,RR,DIS,2*I,RR1
2512 12   CONTINUE
2513   WRITE(IO,16)
2514 20   IF(NFI.EQ.1.OR.NBCU.EQ.0) GO TO 1
2515   IF(NCL.EQ.15.OR.NCL.EQ.23) GO TO 1
2516 C
2517 C   EN CASO DE CARGAS UNIFORMES Y SEGUN EL TIPO DE PROBLEMA SE
2518 C   ANALIZA EL MOMENTO MAXIMO Y SE CAMBIA DE POSICION SI PROCEDE
2519 C   LA ROTULA VARIABLE
2520 C
2521   IF(ISSW(15).LT.0) GO TO 1
2522   CALL CAMBI(RMOME,NB,NI,NJ,IT,ND,NH,NBCU,DCBU,RLON,ALFA,TOL,RMP,LF,
2523 @     SUM,TOLS,NBB,COR,NCL)
2524   IF(LF.EQ.0) GO TO 1
2525 C
2526 C   LLAMADA AL SEGMENTO CORRESPONDIENTE
2527 C
2528   CALL EXEC(8,5HITERA)
2529 1   CALL EXEC(8,5HMOVIM)
2530 C
2531 C   FORMATOS DE ESCRITURA
2532 C
2533 62   FORMAT(10X,"Ecuacion de Equilibrio ",I3,2X,E14.5," = ",E14.5)
2534 68   FORMAT(10X,"Ecuacion de Compatib. ",I3,2X,E14.5," = 0.0")
2535 70   FORMAT(10X,"Seguridad de la seccion",I3,2X,E14.5," <= 0.0")
2536 72   FORMAT(10X,"Paridad de la seccion ",I3,2X,E14.5," >= 0.0",
2537 @     /,38X,E14.5," = 0.0")

```

```
2538 22  FORMAT(10X,"NUMERO DE AJUSTES DE PASIVAS....",I3)
2539 18  FORMAT(2/,10X,"CARGA DE COLAPSO.....",E14.4/)
2540 11  FORMAT(10X," BARRA SECCION      MOMENTO      DISTANCIA")
2541 13  FORMAT(10X,I5,2X,I4,3X,E12.4,2X,E12.4/17X,I4,3X,E12.4)
2542 16  FORMAT(2,5X,50("="))
2543      END
```

FTN4X COMPILER: HP92834 REV.2130 (810716)

\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: 1027 COMMON: (NONE)



```

2544 SUBROUTINE ECUA(TETA,RMAT,RIMON,RMOME,R,RLON,RINER,RMP,ICOL,DCBU,
2545 @ IT,ND,NB,LB1,NVIR,NMH,TOLS,E,NH,Q,ALFA,SUM,NJ,
2546 @ NITER,NFA,NI,NBCU,TOLA,NTRB,LL,NND,NBB,LVUE,COM,
2547 @ EE,COI,COL)
2548 C
2549 C RUTINA PARA LA FORMACION DE LA MATRIZ DE ENTRADA AL SIMPLEX.
2550 C DETERMINACION DE LA DISTRIBUCION DE MOMENTOS Y ROTACIONES.
2551 C
2552 DIMENSION TETA(LB1,LB1),RMAT(LB1,LB1),RIMON(NH),RMOME(NH),R(NND),
2553 @ RLON(NB),RINER(NB),RMP(NB),ICOL(NH),DCBU(NBB),ALFA(NB),
2554 @ Q(NH),NJ(NB),NI(NB),NFA(LB1),IT(ND),NA1(3),NBAS(100),
2555 @ KG(50),RFG(50)
2556 COMMON /VAR/ IN,IO,NOMB(30)
2557 NVU=NVIR
2558 LVUE=0
2559 NTRA=NTRB
2560 NTRC=NTRA
2561 LLNN=LL
2562 NTRB=0
2563 C
2564 C OPERACIONES CON FICHEROS Y LLAMADA A LA RUTINA DE PROGRAMACION
2565 C
2566 C
2567 C OPERACIONES DE GRABACION EN DISCO
2568 C
2569 CALL DISCO(4,59,IER)
2570 IF(IER.LT.0) GO TO 1113
2571 NPAL=(NVIR+1)*2
2572 DO 1 J=1,LL
2573 CALL DISCO(2,59,IER,TETA(1,J),NPAL)
2574 IF(IER.LT.0) GO TO 1113
2575 1 CONTINUE
2576 CALL DISCO(2,59,IER,R,NPAL)
2577 IF(IER.LT.0) GO TO 1113
2578 C
2579 C LLAMADA A LA RUTINA DE RESOLUCION
2580 C
2581 200 DO 2 I=1,NVIR
2582 R(I)=TOLE(R(I),TOLS)
2583 DO 2 J=1,LL
2584 TETA(I,J)=TOLE(TETA(I,J),TOLS)
2585 2 CONTINUE
2586 LVUE=LVUE+1
2587 TOL1=TOLS
2588 ZAT=0.0
2589 DO 810 II=1,NVIR
2590 IF(ABS(R(II)).GT.ZAT) ZAT=ABS(R(II))
2591 810 CONTINUE
2592 TOLS=ZAT*TOLS
2593 IF(ISSW(4).GE.0) GO TO 122
2594 CALL ESCRM(TETA,R,LB1,LB1,NND,NVIR,LL,1)
2595 122 CALL SIMP(TETA,R,NVIR,LL,LL,LB1,NND,TOLS,LB1,NBAS,LL,NVIR)
2596 TOLS=TOL1
2597 C
2598 C ANALISIS DE LA DISTRIBUCION DE MOMENTOS PARA LA DETECCION DE

```

```
2599 C ROTULAS PASIVAS
2600 C
2601 NVIT=NVIR
2602 NVIR=NVUJ
2603 DO 3 I=1,NH
2604 RMOME(I)=RIMON(I)
2605 3 CONTINUE
2606 L=0
2607 DO 4 I=1,NVUJ
2608 IF(ICOL(I).EQ.0) GO TO 4
2609 L=L+1
2610 NB1=(ICOL(I)+1)/2
2611 R(L)=R(L)-RMP(NB1)*10.
2612 4 CONTINUE
2613 DO 5 I=1,NH
2614 S=0
2615 LD=0
2616 DO 6 J=1,NVUJ
2617 IF(ICOL(J).EQ.0) GO TO 6
2618 LD=LD+1
2619 S=S+RMAT(J,I)*R(LD)
2620 6 CONTINUE
2621 RMOME(I)=S+Q(I)
2622 5 CONTINUE
2623 DO 81 I=1,NH
2624 RFG(I)=0
2625 81 CONTINUE
2626 DO 82 I=1,NTRC
2627 NSG=1
2628 IF(RMOME(NFA(I)).LT.0) NSG=-1
2629 RFG(NFA(I))=R(I+NMH)*NSG
2630 82 CONTINUE
2631 C
2632 C ESCRITURA SI PROCEDE DE LA DISTRIBUCION DE MOMENTOS Y
2633 C ROTACIONES PLASTICAS
2634 C
2635 IF(ISSW(8).GE.0) GO TO 150
2636 DO 152 I=1,NH
2637 WRITE(IO,151) I,RMOME(I)*COM,RFG(I)*COM*COL/(EE*COI)
2638 152 CONTINUE
2639 150 NL=0
2640 C
2641 C SE ANALIZA SI SE HA PRODUCIDO ALGUNA VIOLACION EN SEGURIDAD O
2642 C EN PARIDAD
2643 C
2644 NL=0
2645 DO 848 II=1,NH
2646 NL=II-NL
2647 GF=RMOME(II)*RMP(NL)/ABS(RMOME(II))
2648 PQ=ABS(RMP(NL)-ABS(RMOME(II)))/RMP(NL)
2649 IF(PQ.LE.0.0025) RMOME(II)=GF
2650 IF(ABS(ABS(RMOME(II))-RMP(NL)).LE.TOLA) RMOME(II)=GF
2651 848 CONTINUE
2652 NL=0
2653 DO 7 I=1,NH
```

```

2654      NL=I-NL
2655      IF(ABS(RMOME(I))-TOLA.LT.RMP(NL)) GO TO 77
2656      PQ=ABS(RMP(NL)-ABS(RMOME(I)))/RMP(NL)
2657      IF(PQ.LE.0.0025) GO TO 77
2658      IF(ABS(RMOME(I)).LT.RMP(NL))GO TO 77
2659      DO 8 IL=1,NTRA
2660      IF(I.EQ.NFA(IL)) GO TO 77
2661 8      CONTINUE
2662      GO TO 9
2663 77     IF(ISSW(8).GE.0) GO TO 79
2664      WRITE(IO,7632) RMOME(I)*RFG(I),ABS((ABS(RMOME(I))-RMP(NL))*RFG(I))
2665 79     IF(RMOME(I)*RFG(I).LT.0)GO TO 9
2666      IF(ABS((ABS(RMOME(I))-RMP(NL))*RFG(I)).GT.TOLS*COM) GO TO 9
2667 7      CONTINUE
2668      GO TO 100
2669 C
2670 C      SE HA DETECTADO UNA ROTULA PASIVA O UNA INCOMPATIBILIDAD
2671 C
2672 C
2673 C      SE RECUPERA LA MATRIZ INICIAL
2674 C
2675 9      CALL DISCO(4,59,IER)
2676      IF(IER.LT.0) GO TO 1113
2677      CALL ZERO(TETA,LB1,LB1,NH,NH)
2678      DO 11 J=1,LLNN
2679      CALL DISCO(1,59,IER,TETA(1,J),NPAL)
2680      IF(IER.LT.0) GO TO 1113
2681 11     CONTINUE
2682      CALL DISCO(1,59,IER,R,NPAL)
2683      IF(IER.LT.0) GO TO 1113
2684 C
2685 C      SE MODIFICA EN FUNCION DE LOS DATOS ANTES OBTENIDOS
2686 C
2687      DO 754 II=1,NVIT
2688      NBAS(II)=100
2689      KG(II)=0
2690 754    CONTINUE
2691      L=0
2692      NVOR=NVIR
2693      LL=LLNN
2694      NVIR=NVU
2695      DO 801 I=NTRA+1,NTRB
2696      NFA(I)=0
2697 801    CONTINUE
2698      NCDW=0
2699 C
2700 C      PRIMERO SE INTRODUCEN LOS MOMENTOS QUE HAN SUPERADO MP
2701 C
2702 343    NL=0
2703      DO 12 I=1,NH
2704      NL=I-NL
2705      IF(ISSW(8).LT.0)WRITE(IO,761) I,NCDW,RMOME(I),RMP(NL),TOLA
2706      IF(NCDW.EQ.1) GO TO 344
2707      IF(ABS(ABS(RMOME(I))-TOLA).LT.RMP(NL)) GO TO 12
2708      GO TO 346

```

```
2709 344 IF(ABS(ABS(RMOME(I))-RMP(NL))/RMP(NL).GT.0.0025) GO TO 12
2710 C
2711 C AUNQUE Mi=MP SI ES FIJO => Bhi=0 NO SE INTRODUCER
2712 C
2713 346 DO 58 II=1,NVU
2714 IF(ICOL(II).EQ.0) GO TO 58
2715 IF(ABS(RMAT(II,I)).LE.TOL) GO TO 58
2716 GO TO 59
2717 58 CONTINUE
2718 GO TO 12
2719 59 DO 39 ILL=1,NTRA
2720 IF(NFA(ILL).EQ.I) GO TO 12
2721 39 CONTINUE
2722 L=L+1
2723 IF(NVOR.EQ.NVU) GO TO 27
2724 IF(NBAS(L+NVU).EQ.0) GO TO 12
2725 27 NTRB=NTRA+L
2726 NTRC=NTRB
2727 NFA(NTRB)=I
2728 RIMON(I)=1.E30
2729 LL=LL+1
2730 NVIR=NVIR+1
2731 NSG=1
2732 IF(RMOME(I).LT.0) NSG=-1
2733 KG(NVIR)=NSG
2734 DO 13 J=1,NVU
2735 TETA(J,LL)=RMAT(J,I)*NSG
2736 13 CONTINUE
2737 DO 14 K=1,LL
2738 TETA(NVIR+1,K)=TETA(NVIR,K)
2739 TETA(NVIR,K)=0
2740 14 CONTINUE
2741 TETA(NVIR+1,LL)=RMP(NL)
2742 LLL=0
2743 R(NVIR)=0
2744 DO 15 K=1,NVU
2745 IF(ICOL(K).EQ.0) GO TO 15
2746 LLL=LLL+1
2747 NB1=(ICOL(K)+1)/2
2748 TETA(NVIR,LLL)=RMAT(K,I)
2749 R(NVIR)=R(NVIR)+RMP(NB1)*RMAT(K,I)*10.
2750 15 CONTINUE
2751 R(NVIR)=R(NVIR)+RMP(NL)*(ABS(RMOME(I))/RMOME(I))-Q(I)
2752 12 CONTINUE
2753 C
2754 C VUELTA A RESOLVER
2755 C
2756 IF(NCOW.EQ.1) GO TO 345
2757 NCOW=1
2758 GO TO 343
2759 345 DO 456 II=NVU+1,NVIR
2760 LL=LL+1
2761 TETA(II,LL)=KG(II)
2762 456 CONTINUE
2763 GO TO 200
```

```
2764 C
2765 C SE HA OBTENIDO UNA SOLUCION COMPATIBLE EN EQUILIBRIO, SEGURA Y
2766 C PARITARIA
2767 C
2768 100 DO 16 I=1, NH
2769 RIMON(I)=0
2770 16 CONTINUE
2771 DO 17 I=1, NTRC
2772 NSG=1
2773 IF(RMOME(NFA(I)).LT.0) NSG=-1
2774 RIMON(NFA(I))=R(I+NMH)*NSG
2775 17 CONTINUE
2776 DO 18 I=1, NH
2777 R(I)=RIMON(I)
2778 RIMON(I)=0.0
2779 18 CONTINUE
2780 NVIR=NVLI
2781 NTRB=NTRC-NTRA
2782 RETURN
2783 C
2784 C PARADAS POR ERRORES EN EL PROCESO
2785 C
2786 1113 STOP 4440
2787 C
2788 C FORMATOS DE ESCRITURA
2789 C
2790 151 FORMAT(5X, "SECCION", 15, " MOMENTO ROT. PLAST", E14.4, E14.4)
2791 7632 FORMAT(10X, "M*FI, (M-MP)*FI", 2E14.5)
2792 761 FORMAT(10X, "I, NCDW, RMOME, RMP, TOL", 2I5, 3E14.6)
2793 END
```

FTN4X COMPILER: HP92834 REV.2130 (810716)

\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: 1943 COMMON: (NONE)

```

2794 SUBROUTINE CAMBI(RMOME,NB,NI,NJ,IT,ND,NH,NBCU,DCBU,RLON,ALFA,TOL,
2795 @ RMP,LFIN,SUM,TOLS,NBB,COR,NCL)
2796 C
2797 C CALCULO DE LOS MOMENTOS MAXIMOS EN BARRAS CARGADAS UNIFORMEMENTE
2798 C
2799 DIMENSION RMOME(NH),NI(NB),NJ(NB),IT(ND),DCBU(NBB),RLON(NB),
2800 @ ALFA(NB),RMP(NB),COR(NB)
2801 COMMON /VAR/ IN,IO
2802 LFIN=0
2803 DO 1 I=1,ND
2804 IF(IT(I).NE.6) GO TO 1
2805 DO 2 J=1,NB
2806 IF(I.NE.NJ(J)) GO TO 2
2807 SUL=RLON(J)+RLON(J+1)
2808 C=SUL-DCBU(4*J-3)-DCBU(4*(J+1)-3)
2809 A=C*(0.5*C+DCBU(4*(J+1)-3))/SUL
2810 B=DCBU(4*K-3)+A
2811 Q=(-DCBU(4*J)*SIN(ALFA(J))+DCBU(4*J-1)*COS(ALFA(J)))*SUM
2812 COK=RMOME(2*J-1)+RMOME(2*J+1)
2813 COK1=RMOME(2*J-1)
2814 IF(NCL.NE.15) GO TO 10
2815 C COK=COK-(COR(J)+COR(J+1))*SUM
2816 C COK1=COK1-COR(J)*SUM
2817 10 Z=B-COK/(Q*SUL)
2818 C RMAX=-RMOME(2*K-1)+Q*Z*(B-A-Z)+Q*0.5*(Z-DCBU(4*K-3))**2
2819 RMAX=-COK1+Z*COK/SUL-0.5*Q*((Z**2-DCBU(4*J-3)**2)*(SUL-Z)+
2820 @ Z*((SUL-Z)**2-DCBU(4*(J+1)-3)**2))/SUL
2821 IF(ABS(Z).LE.SUL*TOL) GO TO 1
2822 IF(ABS(Z-DCBU(4*J-2)).LE.TOLS) GO TO 1
2823 IF(ABS(SUL-ABS(Z)).LE.SUL*TOL) GO TO 1
2824 IF(Z.LT.DCBU(4*J-3)) GO TO 1
2825 IF(Z.GT.SUL) GO TO 1
2826 IF(ABS(RMAX)+TOLS.LT.RMP(J)) GO TO 1
2827 DCBU(4*(J+1)-2)=SUL-Z
2828 DCBU(4*J-2)=Z
2829 LFIN=1
2830 GO TO 1
2831 2 CONTINUE
2832 1 CONTINUE
2833 RETURN
2834 END

```

FTN4X COMPILER: HP92834 REV.2130 (810716)

\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: 509 COMMON: (NONE)

```

2835      PROGRAM MOVIM(5)
2836 C
2837 C      SEGMENTO PARA CALCULAR LOS MOVIMIENTOS
2838 C
2839      COMMON /VAR/ IN, IO, NOMB(30), ND, NB, INDT, NMI, NLI, NVIR, NAI, NBCU, NMH,
2840 @          NFI, NGD, NGP, NCL, LHF, LHV, NITER, NND, NH, NBB, NTCPV, NCFP,
2841 @          NCPV, NMIN, NGH, NMCO, NMHCO, NECO, NTRA, LV, CADI, SUMF, E,
2842 @          SUM, TOL, TOLS, SUM1, PESO, SIGMA, COP, COL, COI, COM, TCO
2843      COMMON /CON/ IT(25), RMP(25), RIMON(50), RMOME(50), NI(25), NJ(25),
2844 @          ALFA(25), RLON(25), RINER(25), RMP1(25), AXIL(25), X1(25),
2845 @          Y1(25), Z1(25), DCL(20), R(75), LBE(20), T(50), LICOL(50),
2846 @          NCOL(50), NFILA(50), ICOL(50), DCBU(100), T1(50), Q(50),
2847 @          NGB(25), NTP(25), LTP(25), AREA(25), T2(50), CFI(50),
2848 @          CMH(50), CMV(50), FIFI(25), X(25), Y(25), Z(25),
2849 @          RLCPV(10, 2), TETA(50, 50), RMAT(50, 50)
2850      COMMON /PAS/ N1, N2, N3, N4, COLAA, FINAL, COR(50), RFIC(50), MC(50),
2851 @          RLOF(25), NECNP, NENP, COTA(10), ROV(25, 6)
2852 C
2853 C      SIGNIFICADO DE LAS VARIABLES
2854 C      *****
2855 C
2856 C      X.....MOVIMIENTO HORIZONTAL
2857 C      Y.....MOVIMIENTO VERTICAL
2858 C      Z.....GIRO DEL NUDO
2859 C      DCL.....GIRO EN LIBERTADES
2860 C      DCBU.....DATOS DE CARGAS EN BARRAS
2861 C      R.....ROTACION PLASTICA
2862 C      RMOME.....DISTRIBUCION DE MOMENTOS
2863 C      SUM.....CARGA DE COLAPSO
2864 C      RIMON.....PARTE PLASTICA +ELASTICA+CARGA UNIFORME(AM+FI+DR)
2865 C      T.....MOVIMIENTOS ASOCIADOS AL VECTOR DE TRABAJO
2866 C
2867      LB1=50
2868      FINAL=0.0
2869      NFI=0
2870 C
2871 C      PUESTA A CERO
2872 C
2873      DO 1 I=1, NH
2874          T(I)=0.0
2875          RIMON(I)=0.0
2876          IF(NCPV.NE.0.OR.NCL.NE.15.OR.NCL.NE.23) GO TO 5
2877          CFI(I)=0.
2878 5      CALL ZERO(RMAT, LB1, LB1, NH, NH)
2879          CALL ZERO(TETA, LB1, LB1, NH, NH)
2880 1      CONTINUE
2881      DO 2 I=1, ND
2882          X(I)=0.0
2883          Y(I)=0.0
2884          Z(I)=0.0
2885          IF(NCPV.NE.0.OR.NCL.NE.15.OR.NCL.NE.23) GO TO 2
2886          X1(I)=0.0
2887          Y1(I)=0.0
2888          Z1(I)=0.0
2889 2      CONTINUE

```

```

2890      DO 3 I=1,NLI
2891 3      DCL(I)=0.0
2892 C
2893 C      LLAMADA A LA RUTINA DE CALCULO DE MOVIMIENTO
2894 C
2895      CALL MOVI(T,RIMON,RMOME,RLON,RINER,RMAT,TETA,R,LICOL,X,Y,Z,IT,
2896 @          DCL,NB,ND,NH,1.,NMI,LB1,NLI,TOL,NVIR,INDT,SUM,DCBU,
2897 @          NBCU,ALFA,NJ,NI,AXIL,N4,COM,COL,E,COI,NBB,NND)
2898 C
2899 C      EN CASO DE ANALIS DE 2 ORDEN DETERMINAR AXILES Y MODIFICAR
2900 C
2901      IF(NCL,NE.15.AND.NCL,NE.23) GO TO 44
2902      DO 88 I=1,ND
2903      Z1(I)=Z(I)
2904 88      CONTINUE
2905 C
2906 C      DETERMINACION DE LOS AXILES Y DELTAS DE CADA BARRA
2907 C
2908      CALL DELTA(Y,X,ALFA,Z,NI,NJ,NB,ND,NB,IT,RLON)
2909      CALL CAXIL(TETA,RMOME,Z,FIFI,RLON,T,AXIL,NB,NH,N1,SUM,COM,COL,
2910 @          E,COI,LB1,TOL)
2911      NCTR=0
2912 C
2913 C      SE ANALIZA LA CONVERGENCIA DE LA SOLUCION OBTENIDA
2914 C
2915      DO 60 I=1,NH
2916      PO=ABS(RMOME(I))-ABS(CFI(I))
2917      CORL=ABS(RMOME(I))
2918      IF(CORL,LT,TOLS)CORL=COM
2919      PO=PO/CORL
2920      IF(ABS(PO).GT.0.005) GO TO 6
2921 60      CONTINUE
2922      DO 61 I=1,NB
2923      PO=ABS(AXIL(I))-ABS(RLOF(I))
2924      IF(N2.EQ.1.AND.AREA(I).GE.100000.) PO=0
2925      CORL=ABS(AXIL(I))
2926      IF(CORL,LT,TOLS) CORL=1
2927      PO=PO/CORL
2928      IF(ABS(PO).GT.0.005) GO TO 6
2929 61      CONTINUE
2930      IF(ABS(SUM-SUMF)/ABS(SUM).GE.0.005) GO TO 6
2931      NCDTR=1
2932      GO TO 4422
2933 C
2934 C      LA SOLUCION OBTENIDA NO ES CORRECTA Y SE MODIFICAN LAS ECUACIONES
2935 C
2936 6      SUMF=SUM
2937      CALL MODIF(TETA,T,Z,FIFI,AXIL,AREA,LB1,SIGMA,NB,ND,SUMF,INDT,
2938 @          TOL,TOLS,NH,RMP,RMP1,T2,T1,NMI,N1,N2,N3,COM,COL,COI,
2939 @          E,COR)
2940      NCPV=NCPV+1
2941      DO 8 I=1,ND
2942      X1(I)=X(I)
2943      Y1(I)=Y(I)
2944 8      CONTINUE

```



```
2945      LL=0
2946      DO 9 I=1,NH
2947      LL=I-LL
2948      RLOF(LL)=AXIL(LL)
2949      CFI(I)=RMOME(I)
2950 9      CONTINUE
2951 C
2952 C      ESCRITURA DE LA ITERACION SI PROCEDE
2953 C
2954 4422  IF(ISSW(2).GE.0) GO TO 16
2955      CALL SECON(TT8)
2956      IF(NCPV.EQ.1) WRITE(IO,10) NOMB
2957      SS=SUM*COM/(COP*COL)
2958      WRITE(IO,11) NCPV,SS
2959      WRITE(IO,12) (AXIL(I)*COM/COL,I=1,NB)
2960      WRITE(IO,13)
2961      WRITE(IO,14) (Z(I)*COM*COL*COL/(E*COI),I=1,NB)
2962      WRITE(IO,15)
2963      CALL SECON(TT9)
2964      TCO=TCO-TT9+TT8
2965 16     IF(NCOTR.EQ.1) GO TO 4444
2966      IF(NCL.EQ.15.OR.NCL.EQ.23) CALL EXEC(8,5HCOLAP)
2967      LF=0
2968 4444  IF(NBCU.NE.0) CALL CAMBI(RMOME,NB,NI,NJ,IT,ND,NH,NBCU,DCBU,RLON,
2969      @      ALFA,TOL,RMP,LF,SUM,TOLS,NBB,COR,NCL)
2970      IF(NCL.EQ.15.AND.LF.EQ.0) GO TO 44
2971 C
2972 C      MODIFICAR LOS DATOS QUE SE VAN A SUPONER EN SU CASO
2973 C
2974      IF(LF.EQ.0) GO TO 4
2975      CALL EXEC(8,5HITERA)
2976 4      CALL DELTA(Y1,X1,ALFA,Z,NI,NJ,NB,ND,NB,IT,RLON)
2977      CCP=SUMF*COM/COL
2978      CALL TRAB2(TETA,T,COR,T1,T2,AXIL,Z,COM,COM,COI,CCP,E,LB1,NMI,NB,
2979      @      ND,N2,AREA)
2980      DO 20 I=1,NB
2981      T(I)=1.0
2982 C
2983 C      MODIFICACION DEL MOMENTO PLASTICO SI PROCEDE
2984 C
2985      IF(N3.EQ.1) GO TO 20
2986      IF(-AXIL(I)*COM/(COL*SIGMA*AREA(I)).LE.0.1525424) GO TO 20
2987      T(I)=1.18*(1+AXIL(I)*COM/(COL*SIGMA*AREA(I)))
2988 20     CONTINUE
2989      CALL EXEC(8,5HDISEN)
2990 C
2991 C      OBTENCION DEL TIEMPO TOTAL DE RESOLUCION
2992 C
2993 44     IF(NCL.NE.15.AND.NCL.NE.23) GO TO 45
2994      DO 46 I=1,ND
2995      Z(I)=Z1(I)
2996 46     CONTINUE
2997 45     CALL SECON(RT)
2998      TCO=RT-TCO
2999 C
```

```
3000 C      LLAMADA AL SEGMENTO DE ESCRITURA DE RESULTADOS
3001 C
3002      CALL EXEC(8,5HRESUL)
3003 C
3004 C      FORMATOS DE ESCRITURA
3005 C
3006 10     FORMAT(1H1,10X,30A2/10X,60("*")////////)
3007 11     FORMAT(10X,"ETAPA.....",I5,/10X,
3008 @      "CARGA DE COLAPSO.....",F12.7,/10X,"ESF. AXIL :")
3009 12     FORMAT(10X,6E10.4)
3010 13     FORMAT(2/,10X,"DESPLAZAMIENTOS RELATIVOS DE LAS BARRAS :")
3011 14     FORMAT(10X,6E10.4)
3012 15     FORMAT(/,10X,60("-"),/)
3013 835    FORMAT(10X,"Secc. Mfn. Mfa. Porc.",I5,3E14.5)
3014 836    FORMAT(10X,"Barra Axn. Axa. Porc.",I5,3E14.5)
3015 838    FORMAT(10X,"PORCENTAJE DE DIFERANCIA EN CARGA",E14.5)
3016      END
```

FTN4X COMPILER: HP92834 REV.2130 (810716)

\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: 1195 COMMON: (NONE)

```

3017 SUBROUTINE MOVI(T,RIMON,RMOME,RLON,RINER,RMAT,TETA,R,ICOL,X,Y,Z,
3018 @ IT,DCL,NB,ND,NH,E,NMI,LB1,NLI,TOL,NEIN,INDT,SUM,
3019 @ DCBU,NBCU,ALFA,NJ,NI,AXIL,N4,COM,COL,EE,COI,NBB,
3020 @ NND)
3021 C
3022 C RUTINA DE CALCULO DE MOVIMIENTOS GENERALIZADOS
3023 C
3024 DIMENSION T(NMI),RIMON(NH),RMOME(NH),RLON(NB),RINER(NB),R(NND),
3025 @ TETA(LB1,LB1),ICOL(NH),DCL(NLI),X(ND),Y(ND),Z(ND),
3026 @ IT(ND),DCBU(NBB),ALFA(NB),NJ(NB),NI(NB),AXIL(NB),
3027 @ RMAT(LB1,LB1)
3028 COMMON /VAR/ IN,IO
3029 C
3030 C LLAMADA A LA RUTINA DE CALCULO DE DEFORMACIONES
3031 C
3032 CALL DEFO(T,RIMON,RMOME,RLON,RINER,R,IT,NB,ND,NH,E,TOL,N4,
3033 @ DCBU,NBCU,ALFA,NJ,AXIL,COM,COL,EE,COI,NBB,NND,RIMON,
3034 @ NCL,RLON,RIMON,SUM)
3035 C
3036 C LECTURA DEL FICHERO
3037 C
3038 CALL LEEDI(RMAT,LB1,LB1,NMI,NH,55,1)
3039 C
3040 C PREPARACION DEL SISTEMA
3041 C
3042 K=0
3043 DO 1 I=1,NH
3044 DO 2 J=1,NEIN
3045 IF(I.EQ.ICOL(J)) GO TO 1
3046 2 CONTINUE
3047 K=K+1
3048 DO 3 KK=1,NMI
3049 TETA(K,KK)=RMAT(KK,I)
3050 3 CONTINUE
3051 T(K)=RIMON(I)
3052 1 CONTINUE
3053 C
3054 C LLAMADA A LA RUTINA DE RESOLUCION
3055 C
3056 CALL SLNPD(TETA,T,NMI,T,NMI,LB1,TOL,NMI)
3057 CALL MOVIT(TETA,IT,NI,NJ,RLON,ALFA,DCBU,T,DCL,X,Y,Z,NB,ND,
3058 @ NH,LB1,INDT,NBCU,NMI,NLI,NBB)
3059 RETURN
3060 END

```

FTN4X COMPILER: HP92834 REV.2130 (810716)

\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: 334 COMMON: (NONE)

```

3061 SUBROUTINE MOVIT(TETA,IT,NI,NJ,RLON,ALFA,DCBU,T,DCL,X,Y,Z,NB,ND,
3062 @ NH,LB1,INDT,NBCU,NMI,NLI,NBB)
3063 C
3064 C DETERMINACION DE LOS MOVIMIENTOS EN COORDENADAS GLOBALES
3065 C
3066 DIMENSION TETA(LB1,LB1),RLON(NB),ALFA(NB),DCL(NLI),X(ND),Y(ND),
3067 @ Z(ND),IT(ND),DCBU(NBB),NI(NB),NJ(NB),T(NMI)
3068 CALL ZERO(TETA,LB1,LB1,LB1,LB1)
3069 C
3070 C LECTURA DEL FICHERO
3071 C
3072 IF(INDT.EQ.0) GO TO 459
3073 L1=2*ND
3074 CALL LEEDI(TETA,LB1,LB1,INDT,L1,51,1)
3075 IF(NBCU.EQ.0) GO TO 459
3076 I=1
3077 93 IF(IT(NJ(I)).NE.6) GO TO 89
3078 RLI=RLON(I)+RLON(I+1)
3079 II=NI(I)
3080 JJ=NI(I+1)
3081 IZ=NJ(I)
3082 AL=ALFA(I)
3083 DO 92 J=1,INDT
3084 UI=TETA(J,2*II-1)
3085 UJ=TETA(J,2*JJ-1)
3086 VJ=TETA(J,2*JJ)
3087 VI=TETA(J,2*II)
3088 TE=(VJ*COS(AL)-UJ*SIN(AL)-VI*COS(AL)+UI*SIN(AL))/RLI
3089 TE=TOLE(TE,TOL)
3090 TETA(J,2*IZ-1)=UI-TE*RLON(I)*SIN(AL)
3091 TETA(J,2*IZ)=VI+TE*RLON(I)*COS(AL)
3092 92 CONTINUE
3093 I=I+1
3094 89 I=I+1
3095 91 IF(I.LE.NB) GO TO 93
3096 459 IF(NBCU.EQ.0) GO TO 578
3097 K=0
3098 IJ=1
3099 572 IL=NJ(IJ)
3100 IF(IT(IL).NE.6) GO TO 571
3101 K=K+1
3102 TETA(INDT+K,2*IL-1)=-SIN(ALFA(IJ))*DCBU(4*IJ-2)
3103 TETA(INDT+K,2*IL)=COS(ALFA(IJ))*DCBU(4*IJ-2)
3104 IJ=IJ+1
3105 571 IJ=IJ+1
3106 IF(IJ.GT.NB) GO TO 573
3107 GO TO 572
3108 573 NCOM=NMI-K+1
3109 C
3110 C CALCULO DE MOVIMIENTOS
3111 C
3112 578 L=0
3113 DO 8 J=1,2*ND,2
3114 L=L+1
3115 SUMH=0

```

```
3116 .      SUMV=0
3117      IF(INDT.EQ.0) GO TO 123
3118      DO 9 I=1,INDT
3119      SUMH=SUMH+T(I)*TETA(I,J)
3120      SUMV=SUMV+T(I)*TETA(I,J+1)
3121 9      CONTINUE
3122 123     LK=INDT
3123      DO 16 I=NCOM,NMI
3124      LK=LK+1
3125      SUMH=SUMH+T(I)*TETA(LK,J)
3126      SUMV=SUMV+T(I)*TETA(LK,J+1)
3127 16     CONTINUE
3128      X(L)=SUMH
3129      Y(L)=SUMV
3130 8      CONTINUE
3131 C
3132 C      CALCULO DE GIROS DE NUDOS
3133 C
3134      K=INDT+1
3135      DO 12 I=1,ND
3136      IF(IT(I).EQ.4.OR.IT(I).EQ.7.OR.IT(I).EQ.8) GO TO 12
3137      Z(I)=T(K)
3138      K=K+1
3139 12     CONTINUE
3140 C
3141 C      SI HAY LIBERTADES CALCULAR GIROS EN LIBERTADES
3142 C
3143      IF(NLI.EQ.0) RETURN
3144      KK=K
3145      DO 13 I=1,NLI
3146      DCL(I)=T(KK)
3147      KK=KK+1
3148 13     CONTINUE
3149      RETURN
3150      END
```

FTN4X COMPILER: HP92834 REV.2130 (810716)

\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: 737 COMMON: (NONE)

```

3151 SUBROUTINE MODIF(TETA,T,Z,FIFI,AXIL,AREA,LB1,SIGMA,NB,ND,SUMF,
3152 @ INDT,TOL,TOLS,NH,RMP,RMP1,T2,T1,NMI,N1,N2,N3,COM,
3153 @ COL,COI,E,COR)
3154 C
3155 C RUTINA PARA ALTERAR EL TRABAJO DE LAS CARGAS EXTERNAS.
3156 C
3157 DIMENSION TETA(LB1,LB1),T(NMI),Z(NB),FIFI(NB),AXIL(NB),AREA(NB),
3158 @ RMP(NB),RMP1(NB),T2(NMI),T1(NMI),COR(NB)
3159 CALL TRAB2(TETA,T,COR,T1,T2,AXIL,Z,COM,COL,COI,SUMF,E,LB1,NMI,
3160 @ NB,ND,N2,AREA)
3161 CALL LEEDI(TETA,LB1,LB1,NMI,NH,55,1)
3162 C
3163 C REDUCCION DEL MOMENTO PLASTICO
3164 C
3165 IF(N3.EQ.1) RETURN
3166 DO 10 I=1,NB
3167 RMP(I)=RMP1(I)
3168 IF(-AXIL(I)*COM/(COL*SIGMA*AREA(I)).LE.0.1525424) GO TO 10
3169 RMP(I)=1.18*RMP1(I)*(1+AXIL(I)*COM/(COL*SIGMA*AREA(I)))
3170 10 CONTINUE
3171 RETURN
3172 END

```

FTN4X COMPILER: HP92834 REV.2130 (810716)

\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: 270 COMMON: (NONE)

```
3173 SUBROUTINE APCUN(IT,NJ,RLON,ALFA,DCBU,RINER,T,NH,ND,NB,NBB,SUM,E,  
3174 @ COM,COL,EE,COI,AXIL)  
3175 C  
3176 C RUTINA PARA DETERMINAR LA DEFORMACION DEBIDA A CARGAS UNIFORMES  
3177 C  
3178 DIMENSION IT(ND),NJ(NB),RLON(NB),ALFA(NB),RINER(NB),DCBU(NBB),  
3179 @ T(NH),AXIL(NB)  
3180 DO 1 IM=1,ND  
3181 IF(IT(IM).NE.6) GO TO 1  
3182 DO 2 J=1,NB  
3183 IF(IM.NE.NJ(J)) GO TO 2  
3184 Q=(-DCBU(4*J)*SIN(ALFA(J))+DCBU(4*J-1)*COS(ALFA(J)))*SUM  
3185 C=RLON(J)-DCBU(4*J-3)  
3186 TE=Q*C/(24.*E*RLON(J)*RINER(J))  
3187 T(2*J-1)=TE*(2*C*RLON(J)**2-C**3)  
3188 T(2*J)=TE*(-4*C*RLON(J)**2-(C-4*RLON(J))*C**2)  
3189 C  
3190 C CORREGIR EN CASO DE ANALISIS DE 2 ORDEN  
3191 C  
3192 IF(ABS(AXIL(J)).LT.0.0001) GO TO 2  
3193 HY=COM*COL/(EE*COI)  
3194 T(2*J-1)=T(2*J-1)-Q*HY*AXIL(J)*RLON(J)**5/(240.*(E*RINER(J))**2)  
3195 T(2*J)=T(2*J)+Q*HY*AXIL(J)*RLON(J)**5/(240.*(E*RINER(J))**2)  
3196 2 CONTINUE  
3197 1 CONTINUE  
3198 RETURN  
3199 END
```

FTN4X COMPILER: HP92834 REV.2130 (810716)

\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: 457 COMMON: (NONE)

```

3200 SUBROUTINE DEFO(T,RIMON,RMOME,RLON,RINER,R,IT,NB,ND,NH,E,TOL,N4,
3201 @ DCBU,NBCU,ALFA,NJ,AXIL,COM,COL,EE,COI,NBB,NND,C5,
3202 @ NCL,CC1,C3,SUM)
3203 C
3204 C RUTINA DE CALCULO DE LAS DEFORMACIONES
3205 C
3206 DIMENSION T(NH),RIMON(NH),RMOME(NH),RLON(NB),RINER(NB),R(NND),
3207 @ IT(ND),DCBU(NBB),ALFA(NB),NJ(NB),AXIL(NB),C5(NH),C3(NH),
3208 @ CC1(NB)
3209 COMMON /VAR/ IN,IO
3210 C
3211 C CALCULO DE LA PARTE ELASTICA
3212 C
3213 DO 1 J=1,NB
3214 K=2*J-1
3215 P=RLON(J)/(RINER(J)*E*6.)
3216 CALL FLEX(AXIL(J),RLON(J),RINER(J),C1,C2,EE,N4,COM,COL,COI)
3217 IF(NCL.NE.14.AND.NCL.NE.16) GO TO 6
3218 RIMON(K+1)=(C5(K+1)*C1-C2*C5(K))*P
3219 RIMON(K)=(C5(K)*C1-C2*C5(K+1))*P
3220 CALL FLEX(CO1(J),RLON(J),RINER(J),C1,C2,EE,N4,COM,COL,COI)
3221 RIMON(K+1)=RIMON(K+1)-(C3(K+1)*C1-C2*C3(K))*P
3222 RIMON(K)=RIMON(K)-(C3(K)*C1-C2*C3(K+1))*P
3223 RIMON(K)=TOLE(RIMON(K),TOL)
3224 RIMON(K+1)=TOLE(RIMON(K+1),TOL)
3225 GO TO 1
3226 6 RIMON(K+1)=(RMOME(K+1)*C1-C2*RMOME(K))*P
3227 RIMON(K)=(RMOME(K)*C1-C2*RMOME(K+1))*P
3228 RIMON(K+1)=TOLE(RIMON(K+1),TOL)
3229 RIMON(K)=TOLE(RIMON(K),TOL)
3230 1 CONTINUE
3231 C
3232 C SUMA DE LA PARTE PLASTICA
3233 C
3234 DO 2 J=1,NH
3235 RIMON(J)=RIMON(J)+R(J)
3236 2 CONTINUE
3237 C
3238 C EN CASO DE CARGA UNIFORME CALCULO Y SUMA DE LA PARTE CORRESPONDIENTE
3239 C
3240 IF(ISSW(8).LT.0) WRITE(IO,10)
3241 IF(ISSW(8).GE.0.OR.NBCU.NE.0) GO TO 4
3242 DO 5 I=1,NH
3243 WRITE(IO,9) I,RIMON(I)-R(I),R(I),RIMON(I)
3244 5 CONTINUE
3245 4 IF(NBCU.EQ.0) RETURN
3246 CALL APCUN(IT,NJ,RLON,ALFA,DCBU,RINER,T,NH,ND,NB,NBB,SUM,E,COM,
3247 @ COL,EE,COI,AXIL)
3248 IF(ISSW(8).GE.0) GO TO 12
3249 DO 7 I=1,NH
3250 WRITE(IO,8) I,RIMON(I)-R(I),R(I),T(I),RIMON(I)+T(I)
3251 7 CONTINUE
3252 12 DO 3 I=1,NH
3253 RIMON(I)=RIMON(I)+T(I)
3254 T(I)=0.0

```



```
3255 S CONTINUE
3256 C
3257 C FORMATS DE ESCRITURA
3258 C
3259 10 FORMAT(3/,10X,"DEFORMACIONES PARCIALES Y TOTALES * MP*L (E*I) ="
3260 @/10X,46("-")3/10X,"SECCION DEF. ELAST. DEF. PLAST. DEF.
3261 @ UNIF. DEF. TOTAL "/)
3262 9 FORMAT(10X,I5,6X,E10.4,6X,E10.4,21X,E10.4)
3263 8 FORMAT(10X,I5,6X,E10.4,6X,E10.4,5X,E10.4,8X,E10.4)
3264 RETURN
3265 END
```

FTN4X COMPILER: HP92834 REV.2130 (810716)

\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: 836 COMMON: (NONE)

```

3266 SUBROUTINE CAXIL(RMAT,RMOME,W2,FIFI,RLOF,RZ,AXIL,NB,NH,NN1,
3267 @ COLAA,COM,COL,E,COI,LB1,TOL)
3268 C
3269 C ESTA RUTINA CALCULO LOS AXILES
3270 C
3271 DIMENSION RMAT(LB1,LB1),RMOME(NH),W2(NB),FIFI(NB),RLOF(NB),
3272 @ RZ(NH),AXIL(NB)
3273 COMMON /VAR/ IN,IO
3274 CALL LEEDI(RMAT,LB1,LB1,NB,NH,52,1)
3275 DO 1 I=1,NB
3276 D=0
3277 DO 2 J=1,NH
3278 D=D+RMAT(I,J)*RMOME(J)
3279 2 CONTINUE
3280 L=NB-I+1
3281 AF=W2(L)
3282 IF(NN1.EQ.1) AF=0.0
3283 RZ(I)=FIFI(I)*COLAA-D-((RMOME(2*L)+RMOME(2*L-1))*AF/(RLOF(L)**2))*
3284 @ COM*COL/(E*COI)
3285 1 CONTINUE
3286 CALL LEEDI(RMAT,LB1,LB1,NB,NB,53,1)
3287 DO 3 I=1,NB
3288 DO 4 J=1,NB
3289 RMAT(I,J)=-RMAT(I,J)*W2(J)*COM*COL/(E*COI)
3290 4 CONTINUE
3291 RMAT(I,NB+1-I)=RMAT(I,NB+1-I)+1
3292 3 CONTINUE
3293 CALL SLNPD(RMAT,RZ,NB,AXIL,NB,LB1,TOL,NB)
3294 RETURN
3295 END

```

FTN4X COMPILER: HP92834 REV.2130 (810716)

\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: 344 COMMON: (NONE)

```
3296      SUBROUTINE DELTA(Y1,X1,ALFA,Z,NI,NJ,NB,ND,NN,IT,RLON)
3297 C
3298 C      RUTINA PARA DETERMINAR EL DELTA DE TODAS LA BARRAS
3299 C
3300      DIMENSION Y1(ND),X1(ND),Z(NN),ALFA(NB),NI(NB),NJ(NB),IT(ND),
3301 @      RLON(NB)
3302      I=1
3303 1      I1=NI(I)
3304      I2=NJ(I)
3305      IF(IT(NJ(I)).EQ.6) GO TO 3
3306 C
3307 C      SI LA BARRA TIENE UN NUDO VARIABLE SE AJUSTA DE FORMA QUE
3308 C      PERMANEZCA ALINEADO EL NUDO VARIABLE CON LOS NUDOS EXTREMOS
3309 C
3310      Z(I)=(Y1(I2)-Y1(I1))*COS(ALFA(I))-(X1(I2)-X1(I1))*SIN(ALFA(I))
3311      GO TO 4
3312 3      I2=NI(I+1)
3313      DEL=(Y1(I2)-Y1(I1))*COS(ALFA(I))-(X1(I2)-X1(I1))*SIN(ALFA(I))
3314      Z(I)=RLON(I)*DEL/(RLON(I)+RLON(I+1))
3315      Z(I+1)=RLON(I+1)*DEL/(RLON(I)+RLON(I+1))
3316      I=I+1
3317 4      I=I+1
3318      IF(I.LE.NB) GO TO 1
3319      RETURN
3320      END
```

FTN4X COMPILER: HP92834 REV.2130 (810716)

\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: 296 COMMON: (NONE)

```

3321 SUBROUTINE TRAB2(TETA,T,COR,T1,T2,AXIL,Z,COM,COL,COI,SUMF,E,LB1,
3322 @ NMI,NB,ND,N2,AREA)
3323 C
3324 C RUTINA PARA LA OBTENCION DEL TRABAJO EN CADA MECANISMO
3325 C INDEPENDIENTE CON EFECTOS DE 2 ORDEN
3326 C
3327 DIMENSION TETA(LB1,LB1),T(NMI),T1(NMI),T2(NMI),COR(NB),Z(NB),
3328 @ AXIL(NB),AREA(NB)
3329 DO 1 I=1,NB
3330 IF(N2.EQ.1.AND.AREA(I).GT.10000.0) AXIL(I)=0.0
3331 T(I)=AXIL(I)*Z(I)*COM*COL/(E*COI*SUMF)
3332 COR(I)=T(I)
3333 1 CONTINUE
3334 CALL LEEDI(TETA,LB1,LB1,NMI,NB,54,1)
3335 DO 7 I=1,NMI
3336 D=0
3337 DO 8 J=1,NB
3338 D=D+TETA(I,J)*T(J)
3339 8 CONTINUE
3340 T1(I)=T2(I)+D
3341 7 CONTINUE
3342 DO 9 I=1,NMI
3343 T(I)=T1(I)
3344 9 CONTINUE
3345 RETURN
3346 END

```

FTN4X COMPILER: HP92834 REV.2130 (810716)

\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: 263 COMMON: (NONE)

```

3347 PROGRAM PAS01(5)
3348 C
3349 C SEGMENTO PARA LA PREPARACION Y OBTENCION DE MATRICES EN EL
3350 C ANALISIS ELASTOPLASTICO DE 1 ORDEN
3351 C
3352 DIMENSION T2(50,1),GGG(50,35)
3353 COMMON /VAR/ IN,IG,NOMB(30),ND,NB,INDT,NMI,NLI,NVIR,NAI,NBCU,NMH,
3354 @ NFI,NGD,NGP,NCL,LHF,LHV,NITER,NND,NH,NBB,NTCPV,NCFP,
3355 @ NCPV,NMIN,NGH,NMCO,NMHCO,NECO,NTRA,LV,CADI,SUMF,E,
3356 @ SUM,TOL,TOLS,SUM1,PESO,SIGMA,COP,COL,COI,COM,TCO
3357 COMMON /CON/ IT(25),RMP(25),RIMON(50),RMOME(50),NI(25),NJ(25),
3358 @ ALFA(25),RLON(25),RINER(25),RMP1(25),AXIL(25),XA(25),
3359 @ YA(25),ZA(25),DCL(20),R(75),LBE(20),T(50),LICOL(50),
3360 @ NCOL(50),NFILA(50),ICOL(50),DCBU(100),T1(50),Q(50),
3361 @ NSD(50),LTP(25),AREA(25),TT(50),RZ(50),
3362 @ CMH(50),CMV(50),FIFI(25),X(25),Y(25),Z(25),
3363 @ RLCPV(10,2),TETA(50,50),RMAT(50,50),TCNP(50,10)
3364 COMMON /PAS/ NMC,NC,NETAPA,IND,COLAA,COLAP,RMINC(50),RFIC(50),
3365 @ MC(50),RLOF(25),NECNP,NENP,COTA(10)
3366 LB1=50
3367 M2=10
3368 M3=M2+25
3369 C
3370 C PUESTA A CERO DE VARIABLES
3371 C
3372 NMC=0
3373 NC=0
3374 COLAA=0.0
3375 NETAPA=0
3376 COLAP=0
3377 SUMF=0
3378 DO 1 I=1,NH
3379 NFILA(I)=0
3380 LICOL(I)=0
3381 1 CONTINUE
3382 CALL ZERO(RMAT,LB1,LB1,NMI,NH)
3383 DO 2 I=1,ND
3384 X(I)=0.0
3385 Y(I)=0.0
3386 Z(I)=0.0
3387 ZA(I)=0.0
3388 YA(I)=0.0
3389 ZA(I)=0.0
3390 2 CONTINUE
3391 IF(NCL.EQ.14.OR.NCL.EQ.16) CALL LEEDI(GGG,LB1,M3,NMI,NB,54,1)
3392 DO 3 I=1,NMI
3393 T2(I,1)=T1(I)
3394 IF(NCL.NE.14) GO TO 7
3395 GGG(I,NB+1)=T1(I)
3396 GO TO 3
3397 7 IF(NCL.NE.16) GO TO 3
3398 DO 8 IJ=1,NECNP
3399 GGG(I,NB+IJ)=TCNP(I,IJ)
3400 8 CONTINUE
3401 3 CONTINUE

```

```

3402 C
3403 C      OBTENCION DE BH Y BP
3404 C
3405      IF(NCL.NE.12.AND.NCL.NE.13.AND.NCL.NE.22) GO TO 4
3406      CALL CAGE(NMI,NH,NCOL,ICOL,NVIR,LB1,RMAT,TETA,TOL,1.,TCNP,M2,
3407 @          NECNP,IT,NJ,ND,NB,Q,1)
3408      GO TO 5
3409 4      IF(NCL.NE.14.AND.NCL.NE.16) GO TO 6
3410      NVV=NB+1
3411      IF(NCL.EQ.16) NVV=NB+NECNP
3412      NV1=NB+1
3413      CALL CAGE(NMI,NH,NCOL,ICOL,NVIR,LB1,RMAT,TETA,TOL,1.,GGG,M3,
3414 @          NVV,IT,NJ,ND,NB,Q,NV1)
3415      CALL LEEDI(GGG,LB1,M3,NH,NB,54,2)
3416      CALL LEEDI(GGG,LB1,M3,NH,NB,62,2)
3417      IF(NCL.NE.16) GO TO 5
3418      DO 9 IJ=1,NH
3419      DO 9 IK=1,NECNP
3420      TCNP(IJ,IK)=GGG(IJ,IK+NB)
3421 9      CONTINUE
3422      GO TO 5
3423 6      CALL CAGE(NMI,NH,NCOL,ICOL,NVIR,LB1,RMAT,TETA,TOL,1.,T2,1,1,
3424 @          IT,NJ,ND,NB,Q,1)
3425 5      CALL LEEDI(RMAT,LB1,LB1,NVIR,NH,50,2)
3426      IF(NCL.NE.22.AND.NCL.NE.13) GO TO 10
3427      CALL LEEDI(TCNP,LB1,M2,NH,NECNP,53,2)
3428      GO TO 11
3429 C
3430 C      FORMACION DE LA MATRIZ INVERSA Y=-(BH^ * A * BH)**(-1)
3431 C
3432 10     CALL FORY(TETA,RMAT,RMOME,RLON,RINER,ICOL,NB,LB1,NVIR,1.,NH,
3433 @          TOL,57)
3434 C
3435 C      FORMACION DE LA MATRIZ Z Y Z*A+I
3436 C
3437      CALL FORZ(RMAT,TETA,RLON,RINER,1.,LB1,NH,NVIR,57,NB)
3438 C
3439 C      LLAMADA AL SEGMENTO CORRESPONDIENTE
3440 C
3441      CALL EXEC(8,5HPAS1B)
3442 11     CALL EXEC(8,5HSHD01)
3443      END

```

FTN4X COMPILER: HP92834 REV.2130 (810716)

\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: 4072 COMMON: (NONE)

```

3444 SUBROUTINE CAGE(NMI,NH,NCOL,ICOL,NVIR,LB1,RMAT,TETA,TOL,LK,TCNP,
3445 @ M2,MM,IT,NJ,ND,NB,Q,NN)
3446 C
3447 C ESTA RUTINA CALCULA EL VECTOR DE CARGAS GENERALIZADO BF
3448 C
3449 DIMENSION RMAT(LB1,LB1),TETA(LB1,LB1),TCNP(LB1,M2),IT(ND),NJ(NB),
3450 @ Q(NH),ICOL(NH),NCOL(NH)
3451 CALL VIRTI(NMI,NH,NCOL,ICOL,NVIR,LB1,RMAT,TETA,TOL,LK,TCNP,LB1,M2,
3452 @ MM,IT,NJ,ND,NB)
3453 DO 1 I=1,MM
3454 DO 2 J=1,NH
3455 DO 2 K=1,NMI
3456 IF(NCOL(K).NE.J) GO TO 2
3457 Q(J)=TCNP(K,I)
3458 2 CONTINUE
3459 DO 4 K=1,NH
3460 TCNP(K,I)=Q(K)
3461 Q(K)=0.0
3462 4 CONTINUE
3463 1 CONTINUE
3464 DO 5 K=1,NH
3465 Q(K)=TCNP(K,NN)
3466 5 CONTINUE
3467 RETURN
3468 END

```

FTN4X COMPILER: HP92834 REV.2130 (810716)

\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: 225 COMMON: (NONE)

```

3469 SUBROUTINE FORY(TETA,RMAT,RMOME,RLON,RINER,ICOL,NB,LB1,NVIR,E,
3470 @ NH,TOL,LR)
3471 C
3472 C FORMACION DE LA MATRIZ INVERSA Y=-(BH^ * A * BH)**(-1)
3473 C
3474 DIMENSION TETA(LB1,LB1),RMAT(LB1,LB1),RMOME(NH),RINER(NH),
3475 @ ICOL(NH),RLON(NH)
3476 DO 2 I=1,NVIR
3477 RR=0
3478 K=0
3479 DO 3 II=1,NVIR
3480 IF(ICOL(II).EQ.0) GO TO 3
3481 K=K+1
3482 RR=0
3483 DO 4 J=1,NB
3484 P=RLON(J)/(6.*RINER(J)*E)
3485 RMOME(2*J-1)=P*(2.*RMAT(II,2*J-1)-RMAT(II,2*J))
3486 RMOME(2*J)=P*(2.*RMAT(II,2*J)-RMAT(II,2*J-1))
3487 4 CONTINUE
3488 DO 5 KK=1,NH
3489 RR=RR+RMAT(I,KK)*RMOME(KK)
3490 5 CONTINUE
3491 TETA(I,K)=RR
3492 3 CONTINUE
3493 2 CONTINUE
3494 CALL INVER(TETA,NVIR,LB1,IER)
3495 IF(IER.NE.0) GOTO 10
3496 DO 6 I=1,NVIR
3497 DO 6 J=1,NVIR
3498 TETA(I,J)=-TETA(I,J)
3499 6 CONTINUE
3500 CALL LEEDI(TETA,LB1,LB1,NVIR,NVIR,LR,2)
3501 RETURN
3502 10 STOP 5550
3503 END

```

FTN4X COMPILER: HP92834 REV.2130 (810716)

\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: 327 COMMON: (NONE)



```

3504 SUBROUTINE FORZ(RMAT,TETA,RLON,RINER,E,LB1,NH,NVIR,LR,NB)
3505 C
3506 C ESTA RUTINA FORMA LA MATRIZ ( Z ) Y LA ( Z * A ) + ( I )
3507 C
3508 DIMENSION RMAT(LB1,LB1),TETA(LB1,LB1),RLON(NB),RINER(NB)
3509 DO 1 I=1,NVIR
3510 DO 1 J=1,NH
3511 TETA(J,I)=RMAT(I,J)
3512 1 CONTINUE
3513 CALL LEEDI(RMAT,LB1,LB1,NVIR,NVIR,LR,1)
3514 CALL PROMA(TETA,RMAT,LB1,LB1,LB1,NH,NVIR,NVIR)
3515 C
3516 C GRABA LA MATRIZ BH*Y
3517 C
3518 CALL LEEDI(TETA,LB1,LB1,NH,NVIR,56,2)
3519 CALL LEEDI(RMAT,LB1,LB1,NVIR,NH,50,1)
3520 CALL PROMA(TETA,RMAT,LB1,LB1,LB1,NH,NVIR,NH)
3521 C
3522 C GRABA LA MATRIZ BH*Y*BH^
3523 C
3524 CALL LEEDI(TETA,LB1,LB1,NH,NH,59,2)
3525 DO 2 J=1,NH
3526 DO 2 K=1,NB
3527 P=RLON(K)/(6.*RINER(K)*E)
3528 RMAT(J,2*K-1)=(TETA(J,2*K-1)*2.-TETA(J,2*K))*P
3529 RMAT(J,2*K)=(TETA(J,2*K)*2.-TETA(J,2*K-1))*P
3530 2 CONTINUE
3531 C
3532 C FORMACION DE LA MATRIZ BH*Y*BH^*A+I
3533 C
3534 DO 3 J=1,NH
3535 RMAT(J,J)=RMAT(J,J)+1.
3536 3 CONTINUE
3537 RETURN
3538 END

```

FTN4X COMPILER: HP92834 REV.2130 (810716)

\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: 311 COMMON: (NONE)

```

3539 PROGRAM PAS1B(5)
3540 C
3541 C SEGMENTO PARA LA PREPARACION Y OBTENCION DE LOS VECTORES DE
3542 C DE CARGA GENERALIZADO Y DE LA MATRIZ PARA EL CALCULO DE LOS
3543 C MOVIMIENTOS EN UN ANALISIS ELASTOPLASTICO
3544 C
3545 DIMENSION T2(50,1),GGG(50,35)
3546 COMMON /VAR/ IN,IO,NOMB(30),ND,NB,INDT,NMI,NLI,NVIR,NAI,NBCU,NMH,
3547 @ NFI,NGD,NGP,NCL,LHF,LHV,NITER,NND,NH,NBB,NTCPV,NCFP,
3548 @ NCPV,NMIN,NGH,NMCO,NMHCO,NECO,NTRA,LV,CADI,SUMF,E,
3549 @ SUM,TOL,TOLS,SUM1,PESO,SIGMA,COP,COL,COI,COM,TCO
3550 COMMON /CON/ IT(25),RMP(25),RIMON(50),RMOE(50),NI(25),NJ(25),
3551 @ ALFA(25),RLON(25),RINER(25),RMP1(25),AXIL(25),XA(25),
3552 @ YA(25),ZA(25),DCL(20),R(75),LBE(20),T(50),LICOL(50),
3553 @ NCOL(50),NFILA(50),ICOL(50),ICBU(100),T1(50),Q(50),
3554 @ NSD(50),LTP(25),AREA(25),TT(50),RZ(50),
3555 @ CMH(50),CMV(50),FIFI(25),X(25),Y(25),Z(25),
3556 @ RLCPV(10,2),TETA(50,50),RMAT(50,50),TCNP(50,10)
3557 COMMON /PAS/ NMC,NC,NETAPA,IND,COLAA,COLAP,RMINC(50),RFIC(50),
3558 @ MC(50),RLOF(25),NECNP,NENP,COTA(10)
3559 LB1=50
3560 M2=10
3561 M3=M2+25
3562 C
3563 C FORMACION DEL VECTOR DE CARGAS DE CALCULO
3564 C
3565 IF(NCL.EQ.12.OR.NCL.EQ.16) GO TO 1
3566 C
3567 C CASO DE CARGA PROPORCIONAL
3568 C
3569 CALL FORBL(TETA,RMAT,ICBU,IT,NJ,RLON,ALFA,RINER,Q,NH,NBCU,NBB,
3570 @ ND,NB,LB1,COM,COL,E,COI,AXIL)
3571 IF(NCL.EQ.14)CALL LEEDI(GGG,LB1,M3,NH,NB,54,1)
3572 GO TO 2
3573 C
3574 C CASO DE CARGA NO PROPORCIONAL
3575 C
3576 IF(NCL.EQ.16)CALL LEEDI(GGG,LB1,M3,NH,NB,54,1)
3577 CALL LEEDI(RMAT,LB1,LB1,NH,NH,54,2)
3578 CALL FORBF(TETA,RMAT,ICBU,IT,NJ,RLON,ALFA,RINER,Q,TCNP,NECNP,NCL,
3579 @ NH,NBCU,NBB,ND,NB,LB1,M2,NENP,COM,COL,E,COI,AXIL)
3580 CALL LEEDI(RMAT,LB1,LB1,NH,NH,54,1)
3581 C
3582 C FORMACION DE LA MATRIZ PARA EL CALCULO DE LOS DESPLAZAMIENTOS
3583 C
3584 IF(NCL.NE.14.AND.NCL.NE.16) GO TO 11
3585 CALL PROMA(RMAT,GGG,LB1,LB1,M3,NH,NH,NB)
3586 CALL LEEDI(RMAT,LB1,LB1,NH,NB,54,2)
3587 11 CALL LEEDI(TETA,LB1,LB1,NMI,NH,55,1)
3588 CALL QBTTER(TETA,RMAT,ICOL,NVIR,NH,NMI,LB1)
3589 C
3590 C LLAMADA A LA RUTINA DE INVERSION
3591 C
3592 CALL INVER(RMAT,NMI,LB1,IER)
3593 IF(IER.NE.0) GO TO 200

```

```
3594 CALL LEEDI(RMAT, LB1, LB1, NMI, NMI, 60, 2)
3595 C
3596 C PUESTA A CERO
3597 C
3598 DO 10 I=1, NH
3599 T(I)=Q(I)
3600 RMOME(I)=0.0
3601 RIMON(I)=0.0
3602 LICOL(I)=0
3603 10 CONTINUE
3604 IND=0
3605 CALL EXEC(8, 5HPASQ2)
3606 200 STOP 5550
3607 END
```

FTN4X COMPILER: HP92834 REV.2130 (810716)

\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: 3855 COMMON: (NONE)

```

3608 SUBROUTINE FORBF(TETA,RMAT,DCBU,IT,NJ,RLON,ALFA,RINER,Q,TCNP,
3609 @ NN,NCL,NH,NBCU,NBB,ND,NB,LB1,M2,NENP,COM,COL,EE,
3610 @ COI,AXIL)
3611 C
3612 C ESTA RUTINA DETERMINA EL VECTOR BF EN CASO DE CARGA NO PROPORCIONAL
3613 C O CARGA VARIABLE
3614 C
3615 DIMENSION TETA(LB1,LB1),RMAT(LB1,LB1),DCBU(NBB),IT(ND),NJ(NB),
3616 @ RLON(NB),ALFA(NB),RINER(NB),Q(NH),TCNP(LB1,M2),TT(50),
3617 @ AXIL(NB)
3618 DO 1 I=1,NH
3619 DO 2 J=1,NN
3620 D=0
3621 DO 3 K=1,NH
3622 D=D+RMAT(I,K)*TCNP(K,J)
3623 3 CONTINUE
3624 TT(J)=D
3625 2 CONTINUE
3626 DO 4 II=1,NN
3627 RMAT(I,II)=TT(II)
3628 4 CONTINUE
3629 1 CONTINUE
3630 DO 5 I=1,NH
3631 DO 5 J=1,NN
3632 TCNP(I,J)=RMAT(I,J)
3633 5 CONTINUE
3634 C
3635 C SUMA APORTE CARGA UNIFORME
3636 C
3637 DO 20 II=1,NB
3638 AXIL(II)=0.0
3639 20 CONTINUE
3640 IF(NBCU.EQ.0) GO TO 15
3641 DO 6 I=1,NN
3642 DO 16 II=1,NH
3643 Q(II)=0.0
3644 16 CONTINUE
3645 IF(NCL.EQ.13.OR.NCL.EQ.22) GO TO 11
3646 DO 8 J=1,NBB
3647 DCBU(J)=0.0
3648 8 CONTINUE
3649 CALL LEIND(DCBU,NBB,I,NN,NBB,58,1)
3650 11 CALL APCUN(IT,NJ,RLON,ALFA,DCBU,RINER,Q,NH,ND,NB,NBB,1.,1.,COM,
3651 @ COL,EE,COI,AXIL)
3652 DO 10 J=1,NH
3653 D=0
3654 DO 9 K=1,NH
3655 D=D+TETA(J,K)*Q(K)
3656 9 CONTINUE
3657 TCNP(J,I)=TCNP(J,I)+D
3658 10 CONTINUE
3659 IF(NCL.EQ.13.OR.NCL.EQ.22) RETURN
3660 6 CONTINUE
3661 CALL LEIND(DCBU,NBB,1,NN,NBB,58,1)
3662 15 IF(NCL.EQ.13.OR.NCL.EQ.22) RETURN

```

```
3663      DO 12 I=1,NH
3664      Q(I)=TCNF(I,1)
3665 12    CONTINUE
3666      NENF=1
3667      RETURN
3668      END
```

FTN4X COMPILER: HP92834 REV.2130 (810716)

\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: 579 COMMON: (NONE)

```

3669 SUBROUTINE FORBL(TETA,RMAT,DCBU,IT,NJ,RLON,ALFA,RINER,Q,NH,
3670 @ NBCU,NBB,ND,NB,LB1,COM,COL,EE,COI,AXIL)
3671 C
3672 C ESTA RUTINA DETERMINA EL VECTOR BF EN CASO DE CARGA PROPORCIONAL
3673 C
3674 DIMENSION TETA(LB1,LB1),RMAT(LB1,LB1),DCBU(NBB),IT(ND),NJ(NB),
3675 @ RLON(NB),ALFA(NB),RINER(NB),Q(NH),TT(50),AXIL(NB)
3676 COMMON /VAR/IN,IO
3677 DO 1 I=1,NH
3678 D=0
3679 DO 2 J=1,NH
3680 D=D+RMAT(I,J)*Q(J)
3681 2 CONTINUE
3682 TT(I)=D
3683 1 CONTINUE
3684 C
3685 C SUMA APORTE CARGA UNIFORME
3686 C
3687 DO 20 II=1,NB
3688 AXIL(II)=0.0
3689 20 CONTINUE
3690 IF(NBCU.EQ.0) GO TO 5
3691 DO 6 I=1,NH
3692 Q(I)=0.0
3693 6 CONTINUE
3694 CALL APCUN(IT,NJ,RLON,ALFA,DCBU,RINER,Q,NH,ND,NB,NBB,1.,1.,COM,
3695 @ COL,EE,COI,AXIL)
3696 DO 10 I=1,NH
3697 D=0
3698 DO 9 J=1,NH
3699 D=D+TETA(I,J)*Q(J)
3700 9 CONTINUE
3701 TT(I)=TT(I)+D
3702 10 CONTINUE
3703 5 DO 12 I=1,NH
3704 Q(I)=TT(I)
3705 12 CONTINUE
3706 RETURN
3707 END

```

FTN4X COMPILER: HP92834 REV.2130 (810716)

\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: 396 COMMON: (NONE)

```
3708      SUBROUTINE ORTER(TETA,RMAT,ICOL,NVIR,NH,NMI,LB1)
3709 C
3710 C      RUTINA PARA LA OBTENCION DE LA MATRIZ DE EQUILIBRIO REDUCIDA
3711 C
3712      DIMENSION TETA(LB1,LB1),RMAT(LB1,LB1),ICOL(NH)
3713      KK=0
3714      DO 1 I=1,NH
3715      IF(NVIR.EQ.0) GO TO 5
3716      DO 2 J=1,NVIR
3717      IF(I.EQ.ICOL(J)) GO TO 1
3718 2     CONTINUE
3719 5     KK=KK+1
3720      DO 3 K1=1,NMI
3721      RMAT(KK,K1)=TETA(K1,I)
3722 3     CONTINUE
3723 1     CONTINUE
3724      RETURN
3725      END
```

F/N4X COMPILER: HP92834 REV.2130 (810716)

\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: 107 COMMON: (NONE)

```

3726 PROGRAM PASO2(5)
3727 C
3728 C SEGMENTO PARA EL CALCULO DE LOS INCREMENTOS DE ROTACION PLASTICA
3729 C DE LA ETAPA, COMPROBACION DE LA DESCARGA Y CHEQUEO FINAL.
3730 C
3731 DIMENSION MSD(50),RL1(50),RMD(50)
3732 COMMON /VAR/ IN, IO, NOMB(30), ND, NB, INDT, NMI, NLI, NVIR, NAI, NBCU, NMH,
3733 @ NFI, NGD, NGP, NCL, LHF, LHV, NITER, NND, NH, NBB, NTCPV, NCFP,
3734 @ NCFV, NMIN, NGH, NMCO, NMHCO, NECO, NTRA, LV, CADI, SUMF, E,
3735 @ SUM, TOL, TOLS, SUM1, PESQ, SIGMA, COP, COL, COI, COM, TCO
3736 COMMON /CON/ IT(25),RMP(25),RIMON(50),RMOME(50),NI(25),NJ(25),
3737 @ ALFA(25),RLON(25),RINER(25),RMP1(25),AXIL(25),XA(25),
3738 @ YA(25),ZA(25),DCL(20),R(75),LBE(20),T(50),LICOL(50),
3739 @ NCOL(50),NFILA(50),ICOL(50),DCBU(100),RMAX(50),Q(50),
3740 @ NSD(50),LTP(25),AREA(25),T2(50),RZ(50),
3741 @ CMH(50),CMV(50),FIFI(25),X(25),Y(25),Z(25),
3742 @ RLCPV(10,2),TETA(50,50),RMAT(50,50),TCNP(50,10)
3743 COMMON /PAS/ NMC,NC,NETAPA,IND,COLAA,COLAF,RMINC(50),RFIC(50),
3744 @ MC(50),RLDF(25),NECNP,NENP,COTA(10),ROV(25,6)
3745 COMMON /PAS2/ NN1,NN2,NN3,NN4,NVV(300),RF1(50),RM1(50),QL1(50),
3746 @ RMPI(50)
3747 LB1=50
3748 DO 15 I=1,NH
3749 T2(I)=0.0
3750 RMAX(I)=0.0
3751 RZ(I)=0.0
3752 15 CONTINUE
3753 C
3754 C CALCULO DE LA ETAPA 0
3755 C
3756 IF(NCL.EQ.16.AND.NGP.EQ.1.AND.NGD.EQ.0) GO TO 141
3757 IF(NCL.NE.12.OR.IND.NE.1)GO TO 99
3758 141 IF(NMC.EQ.0) GO TO 1
3759 CALL LEEDI(RMAT,LB1,LB1,NMC,NMC,61,1)
3760 GO TO 6
3761 99 IF(NETAPA.EQ.1) GOTO 7
3762 IF(NMC.EQ.0.AND.NETAPA.EQ.0) GO TO 1
3763 7 IF(NMC.EQ.0) GO TO 189
3764 C
3765 C COMIENZA LA ETAPA GENERICA I
3766 C
3767 CALL ZERO(RMAT,LB1,LB1,NMC,NMC)
3768 189 CALL PASO1(RMAT,TETA,T2,NFILA,R,NMC,NH,NVIR,NC,TOL,LB1,LFIN)
3769 NGD=0
3770 IF(LFIN.EQ.1) GOTO 2
3771 6 CALL OBTFI(RMAT,R,Q,NFILA,RFIC,NMC,NH,LB1,RF1,QL1,NCL,RMPI,COLAF,
3772 @ TOLS)
3773 IF(NCL.NE.14.AND.NCL.NE.16) GO TO 4
3774 CALL CALIN(RMD,TETA,RFIC,Q,NH,NMC,0,LB1,MC,TOL,TOLS,RMOME,RMP,
3775 @ NB,NCL,RF1,QL1,RL1,NFILA)
3776 CALL DECET(RMOME,RMD,MC,RMP,COTA,NC,RINP,NCL,NENP,NH,NB,NECNP,
3777 @ RL1,AXIL,TOLS)
3778 IF(NCL.NE.16) GO TO 167
3779 IF(ABS(COLAA+RINP).GT.ABS(COTA(NENP)))RINP=COTA(NENP)-COLAA
3780 167 DO 20 I=1,NH

```



```
3781      RMD(I)=RFIC(I)*RINP+RF1(I)
3782      RL1(I)=RFIC(I)
3783      RFIC(I)=RMD(I)
3784 20    CONTINUE
3785 4     CALL DESCA(RMAT,TETA,NFILA,R,RFIC,RMOME,NMC,NH,LDE,LB1,MC,NFF,
3786 @      COTA,NECNP,NENP,NCL,MSD,INDA,NSD,DCBU,ROV,NBB,NBCU,
3787 @      NJ,IT,ND,NB,RIMON)
3788      IF(NCL.NE.14.AND.NCL.NE.16) GO TO 19
3789      DO 27 IJ=1,NH
3790      RFIC(IJ)=RL1(IJ)
3791 27    CONTINUE
3792 19    IF(INDA.EQ.0) GO TO 11
3793      DO 14 II=1,INDA
3794      RFIC(MSD(II))=0.0
3795      RF1(MSD(II))=0.0
3796      RL1(MSD(II))=0.0
3797      IF(NCL.NE.11.AND.NCL.NE.12) GO TO 14
3798      WRITE(IO,13) MSD(II)
3799 14    CONTINUE
3800 11    KK=0
3801      IF(NMC.EQ.0.AND.NFF.NE.1) GO TO 122
3802      GO TO 111
3803 122   CALL ZERO(RMAT,LB1,LB1,NH,NH)
3804      GO TO 1
3805 111   IF(NFF.EQ.1.AND.LDE.EQ.1) KK=1
3806      IF(KK.EQ.0) GOTO 5
3807      NFF=0
3808      GOTO 7
3809 5     IF(LDE.EQ.1) GOTO 6
3810      IF(NFF.EQ.1) GOTO 3
3811 1     CALL EXEC(8,5HPASO3)
3812 2     CALL DEPSE(R,RMOME,NFILA,RFIC,NMC,NH,NFF,COTA,NECNP,NENP,NCL,RMAT,
3813 @     LB1,TOL)
3814      IF(NFF.EQ.2) GOTO 3
3815      GOTO 4
3816 3     CALL SECON(RV)
3817      TEM=RV-TCO-PESO
3818      WRITE(IO,10) TEM
3819      STOP
3820 C
3821 C     FORMATOS DE ESCRITURA
3822 C
3823 13    FORMAT(10X,"LA SECCION ",I2," SE HA DESCARGADO")
3824 10    FORMAT(2/10X,"*** TIEMPO DE C.P.U. EMPLEADO.....",F10.3,"SG.")
3825      END
```

FTN4X COMPILER: HP92834 REV.2130 (810716)

\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: 837 COMMON: (NONE)

```
3826 SUBROUTINE PASOI(RMAT,TETA,T2,NFILA,R,NMC,NH,NVIR,NC,TOL,LB1,LFIN)
3827 C
3828 C TRATAMIENTO DE UNA ETAPA GENERICA. FORMACION DE MATRICES.
3829 C
3830 DIMENSION RMAT(LB1,LB1),TETA(LB1,LB1),T2(NH),NFILA(NH),R(NH)
3831 COMMON /VAR/ IN,IO
3832 NMC=NMC+1
3833 LFIN=0
3834 CALL LEIND(T2,NH,NC,NH,NVIR,50,1)
3835 CALL LEEDI(RMAT,LB1,LB1,NH,NVIR,56,1)
3836 DO 3 I=1,NH
3837 D=0
3838 DO 4 J=1,NVIR
3839 D=D+RMAT(I,J)*T2(J)
3840 4 CONTINUE
3841 TETA(I,NMC)=D
3842 3 CONTINUE
3843 NFILA(NMC)=NC
3844 C
3845 C FORMACION DE LA INVERSA LA MATRIZ GENERICA Z
3846 C
3847 IF(NMC.EQ.1) GOTO 12
3848 LL=NMC-1
3849 CALL LEEDI(RMAT,LB1,LB1,LL,LL,61,1)
3850 RAYOR=0.0
3851 DO 6 I=1,NMC-1
3852 D=0
3853 IF(RAYOR.LT.ABS(TETA(NC,I)))RAYOR=ABS(TETA(NC,I))
3854 DO 7 J=1,NMC-1
3855 D=D+RMAT(I,J)*TETA(NC,J)
3856 7 CONTINUE
3857 R(I)=D
3858 R(I)=TOLE(R(I),TOL)
3859 6 CONTINUE
3860 I=0
3861 DO 8 I=1,NMC-1
3862 D=D+TETA(NC,I)*R(I)
3863 8 CONTINUE
3864 C
3865 C OBTENCION DE H Y CHEQUEO DE POSIBLES COLAPSO
3866 C
3867 H=TETA(NC,NMC)-D
3868 IF(ABS(H).GT.RAYOR/1.E3) GOTO 301
3869 LFIN=1
3870 RETURN
3871 301 DKK=1/H
3872 DO 9 I=1,NMC-1
3873 R(I)=-R(I)*DKK
3874 9 CONTINUE
3875 DO 10 I=1,NMC-1
3876 RMAT(I,NMC)=R(I)
3877 RMAT(NMC,I)=R(I)
3878 10 CONTINUE
3879 RMAT(NMC,NMC)=DKK
3880 DO 85 I=1,NMC-1
```

```
3881      DO 85 J=1,NMC-1
3882      RMAT(I,J)=RMAT(I,J)+RMAT(NMC,J)*RMAT(I,NMC)/DKK
3883 85    CONTINUE
3884      GOTO 11
3885 12    H=TETA(NC,1)
3886      IF(ABS(H).GT.0.000001) GOTO 302
3887      LFIN=1
3888      RETURN
3889 302   DKK=1/H
3890      RMAT(1,1)=DKK
3891 11    RETURN
3892      END
```

FTN4X COMPILER: HP92834 REV.2130 (810716)

\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: 531 COMMON: (NONE)

```
3893 SUBROUTINE DBTFI(RMAT,R,Q,NFILA,RFIC,NMC,NH,LB1,RF1,QL1,NCL,RMPI,
3894 @ COLAP,TOLS)
3895 C
3896 C OBTENCION DEL INCREMENTO DE ROTACION PLASTICA
3897 C
3898 DIMENSION RMAT(LB1,LB1),R(NH),Q(NH),NFILA(NH),RFIC(NH),RF1(NH),
3899 @ QL1(NH),RMPI(NH)
3900 COMMON /VAR/ IN,IO
3901 CALL LEEDI(RMAT,LB1,LB1,NMC,NMC,61,2)
3902 DO 7 I=1,NH
3903 RFIC(I)=0.0
3904 RF1(I)=0.0
3905 7 CONTINUE
3906 IF(NMC.EQ.0) RETURN
3907 DO 1 I=1,NMC
3908 R(I)=0
3909 DO 2 J=1,NMC
3910 R(I)=R(I)+RMAT(I,J)*Q(NFILA(J))
3911 2 CONTINUE
3912 R(I)=-R(I)
3913 1 CONTINUE
3914 DO 3 I=1,NMC
3915 RFIC(NFILA(I))=R(I)
3916 3 CONTINUE
3917 IF(NCL.NE.14.AND.NCL.NE.16) RETURN
3918 DO 4 I=1,NMC
3919 R(I)=0
3920 DO 5 J=1,NMC
3921 R(I)=R(I)+RMAT(I,J)*(QL1(NFILA(J))-RMPI(NFILA(J)))
3922 5 CONTINUE
3923 R(I)=-R(I)
3924 4 CONTINUE
3925 DO 6 I=1,NMC
3926 RF1(NFILA(I))=R(I)
3927 6 CONTINUE
3928 RETURN
3929 END
```

FTN4X COMPILER: HP92834 REV.2130 (810716)

\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: 326 COMMON: (NONE)

```

3930 SUBROUTINE DESCA (RMAT,TETA,NFILA,R,RFIC,RMOME,NMC,NH,LDE,LB1,MC,
3931 @ NFF,COTA,NECNF,NENP,NCL,MSD,INDA,NSD,DCBU,ROV,
3932 @ NBB,NJ,IT,ND,NB,RIMON)
3933 C
3934 C TRATAMIENTO Y CHEQUEO DE POSIBLES DESCARGAS
3935 C
3936 DIMENSION RMAT(LB1,LB1),TETA(LB1,LB1),NFILA(NH),RMOME(NH),
3937 @ RFIC(NH),R(NH),MC(NH),COTA(NECNF),NSD(NH),MSD(NH),
3938 @ DCBU(NBB),ROV(NB,6),NJ(NB),IT(ND),RIMON(NH)
3939 COMMON /VAR/ IN,IO
3940 LDE=0
3941 NSG=1
3942 IF(NCL.NE.12.AND.NCL.NE.16) GO TO 1
3943 IF(COTA(NENP).LT.0) NSG=-1
3944 1 INDA=0
3945 DO 15 I=1,NH
3946 NSD(I)=0
3947 MSD(I)=0
3948 IF(NSG*RFIC(I)*RMOME(I).GE.0.0) GOTO 15
3949 LDE=1
3950 RFIC(I)=0
3951 MC(I)=0
3952 INDA=INDA+1
3953 MSD(INDA)=I
3954 IF(NBCU.EQ.0) GO TO 100
3955 RI=I/2.
3956 JI=I/2
3957 IF(RI-JI.NE.0) GO TO 100
3958 IF(IT(NJ(JI)).NE.6) GO TO 100
3959 NSD(I)=NSD(I)+1
3960 ROV(JI,2*NSD(I)-1)=DCBU(4*J-3)
3961 ROV(JI,2*NSD(I))=RIMON(I)
3962 100 IF(NMC.EQ.1) GOTO 92
3963 DO 16 J=1,NMC
3964 IF(NFILA(J).EQ.I) GOTO 17
3965 16 CONTINUE
3966 GOTO 15
3967 17 IF(J.GT.NMC-1) GO TO 81
3968 DO 18 II=1,NMC
3969 DO 18 K=J,NMC-1
3970 D=RMAT(II,K)
3971 RMAT(II,K)=RMAT(II,K+1)
3972 RMAT(II,K+1)=D
3973 18 CONTINUE
3974 DO 19 K=J,NMC-1
3975 NFILA(K)=NFILA(K+1)
3976 19 CONTINUE
3977 DO 20 II=1,NH
3978 DO 20 K=J,NMC-1
3979 TETA(II,K)=TETA(II,K+1)
3980 20 CONTINUE
3981 DO 21 II=1,NMC
3982 DO 21 K=J,NMC-1
3983 D=RMAT(K,II)
3984 RMAT(K,II)=RMAT(K+1,II)

```

```
3985      RMAT(K+1,II)=D
3986 21    CONTINUE
3987 81    NMC=NMC-1
3988      NFILA(NMC+1)=0
3989      DO 50 K =1,NMC
3990      DO 50 J=K,NMC
3991      RMAT(K,J)=RMAT(K,J)-RMAT(NMC+1,K)*RMAT(J,NMC+1)/RMAT(NMC+1,NMC+1)
3992 50    CONTINUE
3993      DO 51 II=1,NMC
3994      DO 51 J=1,NMC
3995      RMAT(J,II)=RMAT(II,J)
3996 51    CONTINUE
3997      CALL LEEDI(RMAT,LB1,LB1,NMC,NMC,61,2)
3998      GOTO 15
3999 92    MC(NFILA(1))=0
4000      NMC=0
4001 15    CONTINUE
4002      RETURN
4003      END
```

FTN4X COMPILER: HP92834 REV.2130 (810716)

\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: 686 COMMON: (NONE)

```

4004 SUBROUTINE DEPSE(R,RMOME,NFILA,RFIC,NMC,NH,NFF,COTA,N1,N2,N3,RMAT,
4005 @ LB1,TOL)
4006 C
4007 C QUEQUEO DE POSIBLES PSEUDOS.
4008 C
4009 DIMENSION R(NH),RMOME(NH),NFILA(NH),RFIC(NH),RMAT(LB1,LB1),
4010 @ COTA(N1)
4011 COMMON /VAR/ IN,IO
4012 N=0
4013 R(NMC)=-1
4014 DO 1 I=1,NMC
4015 IF(R(I)*RMOME(NFILA(I)).LE.TOL)N=N+1
4016 1 CONTINUE
4017 IF(N.EQ.0.OR.N.EQ.NMC) GOTO 2
4018 D=0
4019 DO 3 I=1,NMC
4020 D=D+RMOME(NFILA(I))*R(I)
4021 3 CONTINUE
4022 NS=1
4023 IF(D.LT.0) NS=-1
4024 DO 4 I=1,NMC
4025 RFIC(NFILA(I))=R(I)*NS
4026 4 CONTINUE
4027 N=0
4028 DO 5 I=1,NMC
4029 IF(RFIC(NFILA(I))*RMOME(NFILA(I)).LE.TOL) N=N+1
4030 5 CONTINUE
4031 IF(N.EQ.0.OR.N.EQ.NMC) GO TO 2
4032 WRITE(IO,100)
4033 NFF=1
4034 NMC=NMC-1
4035 CALL LEEDI(RMAT,LB1,LB1,NMC,NMC,61,1)
4036 RETURN
4037 2 NFF=2
4038 RETURN
4039 C
4040 C FORMATOS DE ESCRITURA
4041 C
4042 100 FORMAT(2/,10X,"EL MECANISMO OBTENIDO ES UN PSEUDO *****")
4043 END

```

FTN4X COMPILER: HP92834 REV.2130 (810716)

\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: 292 COMMON: (NONE)

```

4044      PROGRAM PAS03(5)
4045 C
4046 C      SEGMENTO PARA EL CALCULO DE LOS INCRMENTOS DE MOMENTOS DE UNA
4047 C      ETAPA
4048 C
4049      COMMON /VAR/ IN, IO, NOMB(30), ND, NB, INDT, NMI, NLI, NVIR, NAI, NBCU, NMH,
4050 @          NFI, NGD, NGP, NCL, LHF, LHV, NITER, NND, NH, NBB, NTCPV, NCFP,
4051 @          NCPV, NMIN, NGH, NMCO, NMHCO, NECO, NTRA, LV, CADI, SUMF, E,
4052 @          SUM, TOL, TOLS, SUM1, PESO, SIGMA, COP, COL, COI, COM, TCO
4053      COMMON /CON/ IT(25), RMP(25), RIMON(50), RMOME(50), NI(25), NJ(25),
4054 @          ALFA(25), RLON(25), RINER(25), RMP1(25), AXIL(25), XA(25),
4055 @          YA(25), ZA(25), DCL(20), R(75), LBE(20), T(50), LICOL(50),
4056 @          NCOL(50), NFILA(50), ICOL(50), DCBU(100), T4(50), Q(50),
4057 @          NSD(50), LTP(25), AREA(25), T2(50), RZ(50),
4058 @          C4(50), C3(50), FIFI(25), X(25), Y(25), Z(25),
4059 @          RLCPV(10, 2), TETA(50, 50), RMAT(50, 50), TCNP(50, 10),
4060 @          TFI(25, 10)
4061      COMMON /PAS/ NMC, NC, NETAPA, IND, COLAA, COLAP, RMINC(50), RFIC(50),
4062 @          MC(50), RLOF(25), NECNP, NENP, COTA(10), ROV(25, 6)
4063      COMMON /PAS2/ NN1, NN2, NN3, NN4, C1(25), C2(25), W1(25), W2(25), W3(50),
4064 @          RF1(50), RM1(50), QL1(50), RMPI(50)
4065      LB1=50
4066      IND=0
4067      CALL CALIN(RMINC, TETA, RFIC, Q, NH, NMC, NBCU, LB1, MC, TOL, TOLS,
4068 @          RMOME, RMP, NB, NCL, RF1, QL1, RM1, NFILA)
4069      IF(NBCU, EQ, 0) GOTO 8
4070 C
4071 C      LLAMADA AL SEGMENTO CORRESPONDIENTE
4072 C
4073      CALL EXEC(8, 5HPAS3B)
4074 B      CALL EXEC(8, 5HPAS3C)
4075      END

```

FTN4X COMPILER: HP92834 REV.2130 (810716)

\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: 48 COMMON: (NONE)



```

4076 SUBROUTINE CALIN(RMINC,TETA,RFIC,Q,NH,NMC,NBCU,LB1,MC,TOL,
4077 @ TOLS,RMOME,RMP,NB,NCL,RF1,QL1,RM1,NFILA)
4078 C
4079 C OBTENCION DE LOS INCREMENTOS DE MOMENTOS
4080 C
4081 DIMENSION RMINC(NH),TETA(LB1,LB1),RFIC(NH),Q(NH),MC(NH),
4082 @ RMOME(NB),RMP(NB),RF1(NH),RM1(NH),QL1(NH),NFILA(NH)
4083 COMMON /VAR/ IN,IO
4084 JJ=0
4085 DO 1 I=1,NH
4086 JJ=I-JJ
4087 RMINC(I)=0
4088 RM1(I)=0.0
4089 IF(NMC.EQ.0) GOTO 2
4090 DO 3 J=1,NMC
4091 RMINC(I)=RMINC(I)+TETA(I,J)*RFIC(NFILA(J))
4092 IF(NCL.NE.14.AND.NCL.NE.16) GO TO 3
4093 RM1(I)=RM1(I)+TETA(I,J)*RF1(NFILA(J))
4094 3 CONTINUE
4095 2 RMINC(I)=RMINC(I)+Q(I)
4096 IF(NCL.EQ.14.OR.NCL.EQ.16) RM1(I)=RM1(I)+QL1(I)
4097 RMINC(I)=TOLE(RMINC(I),TOL)
4098 RM1(I)=TOLE(RM1(I),TOL)
4099 IF(MC(I).NE.1) GO TO 5
4100 RMINC(I)=0.0
4101 RM1(I)=0.0
4102 5 IF(ABS(RMINC(I)).GT.RMP(JJ)/1000.) GO TO 1
4103 RMINC(I)=0.0
4104 RM1(I)=0.0
4105 1 CONTINUE
4106 RETURN
4107 END

```

FTN4X COMPILER: HP92834 REV.2130 (810716)

\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: 361 COMMON: (NONE)

```

4108 PROGRAM PAS3B(5)
4109 C
4110 C SEGMENTO PARA EL TRATAMIENTO DE LAS CARGAS UNIFORMES. OBTENCION
4111 C DEL INCRMENTO DE CARGA SEGURO DE LA ETAPA, Y SI PROCEDE ACTUALI-
4112 C ZACION DE LAS ECUACIONES Y VARIABLES.
4113 C
4114 DIMENSION RM1X(50),RMAX(50)
4115 COMMON /VAR/ IN,IG,NOMB(30),ND,NB,INDT,NMI,NLI,NVIR,NAI,NBCU,NMH,
4116 @ NFI,NGD,NGP,NCL,LHF,LHV,NITER,NND,NH,NBB,NTCPV,NCFP,
4117 @ NCPV,NMIN,NGH,NMCO,NMHCO,NECO,NTRA,LV,CADI,SUMF,E,
4118 @ SUM,TOL,TOLS,SUM1,PESO,SIGMA,COP,COL,COI,COM,TCO
4119 COMMON /CON/ IT(25),RMP(25),RIMON(50),RMOE(50),NI(25),NJ(25),
4120 @ ALFA(25),RLON(25),RINER(25),RMP1(25),AXIL(25),XA(25),
4121 @ YA(25),ZA(25),DCL(20),R(75),LBE(20),T(50),LICOL(50),
4122 @ NCOL(50),NFILA(50),ICOL(50),DCBU(100),T4(50),Q(50),
4123 @ NSD(50),LTP(25),AREA(25),T2(50),RZ(50),
4124 @ C6(50),C3(50),FIFI(25),X(25),Y(25),Z(25),
4125 @ RLOPV(10,2),TETA(50,50),RMAT(50,50),TCNP(50,10),
4126 @ TFI(25,10)
4127 COMMON /PAS/ NMC,NC,NETAPA,IND,COLAA,COLAP,RMINC(50),RFIC(50),
4128 @ MC(50),RLOF(25),NECNF,NENP,COTA(10),ROV(25,6)
4129 COMMON /PAS2/ NN1,NN2,NN3,NN4,C1(25),C2(25),W1(25),W2(25),W3(50),
4130 @ RF1(50),RM1(50),QL1(50),RMPI(50)
4131 LB1=50
4132 IF(ISSW(15).LT.0) GOTO 10
4133 DO 1 I=1,NH
4134 RMAX(I)=RMINC(I)
4135 RM1X(I)=RM1(I)
4136 1 CONTINUE
4137 CALL TRCUN(IT,RZ,NJ,RMAX,RLON,DCBU,ALFA,RMINC,MC,RMP,RMOE,RMAT,
4138 @ TETA,T2,Q,NI,ICOL,ND,NH,NB,TOLS,LB1,NVIR,NMC,NMI,NFILA,
4139 @ COLAA,NBCU,TOL,NBB,NCL,NENP,NECNF,TCNP,LB1,10,COTA,RM1X,
4140 @ RM1,AXIL,T,RINER,1.,C1,C2,W1,W2,SUMF,E,COM,COL,COI,X,Y,
4141 @ Z,RIMON,LBE,NSD,ROV,C6,FIFI,TFI,W3,NN1,T4,XA,YA,ZA,COP,
4142 @ AREA,NN2,INDT,NND)
4143 C
4144 C LLAMADA AL SEGMENTO CORRESPONDIENTE
4145 C
4146 10 CALL EXEC(8,5HPAS3C)
4147 END

```

FTN4X COMPILER: HP92834 REV.2130 (810716)

\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: 334 COMMON: (NONE)

```

4148 SUBROUTINE TRCUN(IT,RZ,NJ,RMAX,RLON,DCBU,ALFA,RMINC,MC,RMP,RMOME,
4149 @ RMAT,TETA,T2,Q,NI,ICOL,ND,NH,NB,TOLS,LB1,NVIR,
4150 @ NMC,NMI,NFILA,COLAA,NBCU,TOL,NBB,NCL,NENP,NECNP,
4151 @ TCNP,NN1,NN2,COTA,RM1X,RM1,AXIL,T,RINER,EY,C1,C2,
4152 @ W1,W2,SUMF,EG,COM,COL,COI,X,Y,Z,RIMON,LBE,NSD,
4153 @ ROV,C6,FIFI,TFI,W3,NN0,T4,XA,YA,ZA,COP,AREA,NN8,
4154 @ INDT,NND)
4155 C
4156 C TRATAMIENTO DE LAS CARGAS UNIFORMES
4157 C
4158 DIMENSION IT(ND),RZ(NH),RMAX(NH),RLON(NB),ALFA(NB),DCBU(NBB),
4159 @ RMINC(NH),RMP(NB),RMOME(NH),MC(NH),TETA(LB1,LB1),Q(NH),
4160 @ RMAT(LB1,LB1),T2(NH),NI(NB),ICOL(NH),NFILA(NH),NJ(NB),
4161 @ ZO(4),RO(4),TCNP(NN1,NN2),COTA(NECNP),RM1X(NH),RM1(NH),
4162 @ AXIL(NB),T(NH),RINER(NB),C1(NB),C2(NB),W1(NB),W2(NB),
4163 @ NA1(3),X(ND),Y(ND),Z(ND),RIMON(NH),LBE(20),ROV(NB,6),
4164 @ C6(NH),FIFI(NB),TFI(NB,NECNP),XB(25),YB(25),ZB(25),
4165 @ RES(25),W3(NH),RZZ(25),ATB(25),T4(NH),XA(ND),YA(ND),
4166 @ ZA(ND),AREA(NB),B(100)
4167 COMMON /VAR/ IN,IO
4168 LKJ=0
4169 DO 110 I=1,NH
4170 LKJ=I-LKJ
4171 RZ(I)=RMOME(I)
4172 IF(NCL.EQ.14.OR.NCL.EQ.16) GO TO 110
4173 C1(LKJ)=0.0
4174 C2(LKJ)=0.0
4175 W1(LKJ)=0.0
4176 W2(LKJ)=0.0
4177 RM1X(I)=0.0
4178 RM1(I)=0.0
4179 110 CONTINUE
4180 C
4181 C SE RECORRE CADA BARRA CON CARGA UNIFORME DETERMINANDOSE EL INCREMENTO
4182 C DE CARGA QUE PRODUCE UN MOMENTO IGUAL A +A- EL PLASTICO.PARA ELLO
4183 C SE RESUELVE F(2 GRADO)=0 , SE DETERMINA LA Z MAXIMA Y SI ESTA QUEDA
4184 C DENTRO DE LA BARRA ,Y SI EL INCREMENTO DE F HALLADO ES EL MENOR QUE
4185 C PROVOCA LA APARICION DE UNA ROTULA SE REALIZAN LAS SUSTITUCIONES
4186 C PERTINENTES
4187 C
4188 DO 1 I=1,ND
4189 LAYOR=0
4190 DO 543 II=1,4
4191 RO(II)=0.0
4192 ZO(II)=0.0
4193 543 CONTINUE
4194 IF(IT(I).NE.6) GO TO 1
4195 DO 2 J=1,NB
4196 IF(I.NE.NJ(J)) GO TO 2
4197 IF(ABS(RMP(J))-ABS(RZ(2*J))).LT.TOLS) GOTO 2
4198 SUL=RLON(J)+RLON(J+1)
4199 FG=RZ(2*J-1)+RM1X(2*J-1)
4200 A=RZ(2*J-1)+RZ(2*J+1)+RM1X(2*J-1)+RM1X(2*J+1)
4201 CC=SUL-DCBU(4*J-3)-DCBU(4*J+1)
4202 D=CC*(0.5*CC+DCBU(4*J+1))/SUL

```

```
4203      BB=(DCBU(4*J-3)+D)/SUL
4204      Q1=0
4205      IF(NCL.NE.12.AND.NCL.NE.16) GO TO 350
4206      IF(NENF.EQ.1) GO TO 350
4207      DO 300 IW=1,NENP-1
4208      CALL LEIND(DCBU,NBB,IW,NECNF,NBB,58,1)
4209      Q1=Q1+(-DCBU(4*J)*SIN(ALFA(J))+DCBU(4*J-1)*COS(ALFA(J)))*COTA(IW)
4210 300    CONTINUE
4211      CALL LEIND(DCBU,NBB,NENP,NECNF,NBB,58,1)
4212 350    QQ=(-DCBU(4*J)*SIN(ALFA(J))+DCBU(4*J-1)*COS(ALFA(J)))
4213      WW=QQ*COLAA+Q1
4214      C=D*(DCBU(4*J-3)+D*0.5)
4215      E=2*(SUL**2)
4216      G1=WW*(RMP(J)+FG-BB*A+WW*C)+A*A/E
4217      G2=WW*(-RMP(J)+FG-BB*A+WW*C)+A*A/E
4218      HH=RMAX(2*J-1)+RMAX(2*J+1)
4219      H=-FG*QQ-RMAX(2*J-1)*WW+BB*A*QQ+HH*(BB*WW-2*A/E)-2*C*QQ*WW
4220      H1=H-RMP(J)*QQ
4221      H2=H+RMP(J)*QQ
4222      RJ=-RMAX(2*J-1)*QQ+HH*(BB*QQ-HH/E)-C*QQ**2
4223      IF(ABS(RJ).GT.TOL) GOTO 107
4224      RO(1)=G1/H1
4225      RO(2)=G2/H2
4226      RO(3)=RO(1)
4227      RO(4)=RO(2)
4228      GOTO 108
4229 107    F1=H1**2+4*G1*RJ
4230      F2=H2**2+4*G2*RJ
4231      IF(F1.LT.0) GOTO 109
4232      RO(1)=(-H1+SQRT(F1))/(2*RJ)
4233      RO(3)=(-H1-SQRT(F1))/(2*RJ)
4234      GOTO 113
4235 109    RO(1)=-1.E30
4236      RO(3)=-1.E30
4237 113    IF(F2.LT.0) GOTO 111
4238      RO(2)=(-H2+SQRT(F2))/(2*RJ)
4239      RO(4)=(-H2-SQRT(F2))/(2*RJ)
4240      GOTO 108
4241 111    RO(2)=-1.E30
4242      RO(4)=-1.E30
4243 108    DO 3 II=1,4
4244      IF(ABS(A+HH*RO(II)).LT.TOL) GO TO 129
4245      ZO(II)=BB*SUL-((A+HH*RO(II))*SUL*2.)/(E*(WW+QQ*RO(II)))
4246      GO TO 3
4247 129    ZO(II)=BB*SUL
4248 3      CONTINUE
4249      RAYOR=1.E36
4250      DO 4 II=1,4
4251      NSIG=1
4252      IF(NCL.NE.16.AND.NCL.NE.12) GO TO 8003
4253      NSIG=COTA(NENF)/ABS(COTA(NENF))
4254 8003    IF(NCL.EQ.14.OR.NCL.EQ.16) GO TO 8001
4255      IF(RO(II).LT.-NSIG*TOLS) GOTO 4
4256 8001    IF(ABS(ZO(II)).LE.SUL/200.) GO TO 4
4257      IF(ABS(SUL-ABS(ZO(II))).LE.SUL/200.) GO TO 4
```

```

4258 IF(ZO(II).LT.DCBU(4*J-3).OR.ZO(II).GT.(SUL-DCBU(4*J+1))) GO TO 4
4259 IF(ZO(II).EQ.BB*SUL) GO TO 4
4260 IF(ABS(ZO(II)-RLON(J)).LT.0.001*RLON(J)) GO TO 4
4261 IF(RAYOR.LT.RD(II)) GO TO 4
4262 IF(RD(II).EQ.-1.E30) GO TO 4
4263 LAYOR=J
4264 MAYOR=II
4265 RAYOR=RD(II)
4266 ZAYOR=ZO(II)
4267 QAYOR=QQ
4268 4 CONTINUE
4269 IF(LAYOR.EQ.0) GO TO 1
4270 C
4271 C LA Z OBTENIDA ES EN PRINCIPIO VALIDA
4272 C
4273 SUL=RLON(LAYOR)+RLON(LAYOR+1)
4274 SU1=SUL-ZAYOR
4275 CALL CAMZ(SU1,ZAYOR,RZ(2*LAYOR-1),RZ(2*LAYOR+1),1.,WW,
4276 @ DCBU(4*LAYOR-3),DCBU(4*LAYOR+1),RZ(2*LAYOR))
4277 WW1=Q1+QQ*SUMF
4278 CALL CAMZ(SU1,ZAYOR,W3(2*LAYOR-1),W3(2*LAYOR+1),1.,WW1,
4279 @ DCBU(4*LAYOR-3),DCBU(4*LAYOR+1),WK)
4280 CALL CAMZ(SU1,ZAYOR,RMAX(2*LAYOR-1),RMAX(2*LAYOR+1),QAYOR,1.,
4281 @ DCBU(4*LAYOR-3),DCBU(4*LAYOR+1),RMAX(2*LAYOR))
4282 IF(NCL.NE.14.AND.NCL.NE.16) GO TO 5
4283 C
4284 C ACTUALIZACION DE LOS DELTAS DE LA ETAPA ANTERIOR
4285 C
4286 CALL DELMO(Y(NI(LAYOR+1)),Y(NI(LAYOR)),X(NI(LAYOR+1)),
4287 @ X(NI(LAYOR)),ALFA(LAYOR),DEL11,DEL12,ZAYOR,SUL)
4288 C
4289 C ACTUALIZACION DE LOS DELTAS Y AXILES SUPUESTOS
4290 C
4291 V2=Y(NI(LAYOR+1))+YA(NI(LAYOR+1))
4292 U2=X(NI(LAYOR+1))+XA(NI(LAYOR+1))
4293 V1=Y(NI(LAYOR))+YA(NI(LAYOR))
4294 U1=X(NI(LAYOR))+XA(NI(LAYOR))
4295 CALL DELMO(V2,V1,U2,U1,ALFA(LAYOR),DEL21,DEL22,ZAYOR,LAYOR)
4296 C
4297 C MODIFICACION DE LOS TERMINOS LIBRES
4298 C
4299 CALL CAMZ(SU1,ZAYOR,RM1X(2*LAYOR-1),RM1X(2*LAYOR+1),QAYOR,0.,
4300 @ DCBU(4*LAYOR-3),DCBU(4*LAYOR+1),RM1X(2*LAYOR))
4301 RM1X(2*LAYOR+2)=-RM1X(2*LAYOR)
4302 5 RZ(2*LAYOR+2)=-RZ(2*LAYOR)
4303 RMAX(2*LAYOR+2)=-RMAX(2*LAYOR)
4304 C
4305 C OBTENCION DEL INCREMENTO DE P
4306 C
4307 CALL DECET(RZ,RMAX,MC,RMP,COTA,NC,RINP,NCL,NENP,NH,NB,NECNP,RM1X,
4308 @ AXIL,TOLS,TOL)
4309 IF(NCL.NE.12.AND.NCL.NE.16) GO TO 8002
4310 IF(ABS(COLAA+RINF).GT.ABS(COTA(NENP))) GO TO 1
4311 8002 NC1=2*LAYOR
4312 NC2=2*LAYOR+2

```

```
4313 IF(NC.EQ.NC1) GO TO 401
4314 IF(NC.EQ.NC2) GO TO 401
4315 GO TO 1
4316 C
4317 C SI EL INCREMENTO DE F CORRESPONDE A LA FORMACION DE UNA ROTULA
4318 C EN EL NUDO VARIABLE QUE ESTAMOS ANALIZANDO SE REALIZAN LAS MODI-
4319 C FICACIONES. EN CASO CONTRARIO SE IGNORA EL CALCULO REALIZADO Y
4320 C SE PASA A ESTUDIAR OTRA BARRA CARGADA
4321 C
4322 401 NC=2*LAYOR
4323 DCBU(4*LAYOR-2)=ZAYOR
4324 DCBU(4*LAYOR+2)=SUL-ZAYOR
4325 RLON(LAYOR)=ZAYOR
4326 RLON(LAYOR+1)=SUL-ZAYOR
4327 RMINC(NC)=RMAX(NC)
4328 RMINC(NC+2)=-RMAX(NC)
4329 W3(NC)=WK
4330 W3(NC+2)=-WK
4331 RMOME(NC)=RZ(NC)
4332 RMOME(NC+2)=RZ(NC+2)
4333 RM1(NC)=RM1X(NC)
4334 RM1(NC+2)=RM1X(NC+2)
4335 IF(NCL.NE.14.AND.NCL.NE.16) GO TO 405
4336 C2(LAYOR)=DEL11
4337 C2(LAYOR+1)=DEL12
4338 W2(LAYOR)=DEL21
4339 W2(LAYOR+1)=DEL22
4340 C
4341 C MODIFICACION DE LOS DATOS DE CARGAS UNIFORMES
4342 C
4343 405 IF(NCL.NE.12.AND.NCL.NE.16) GO TO 215
4344 CALL DISCO(4,56,IER)
4345 IF(IER.LT.0) STOP 4440
4346 DO 216 IJ=1,NECNP
4347 CALL LEIND(DCBU,NBB,IJ,NECNP,NBB,58,1)
4348 DCBU(4*LAYOR-2)=ZAYOR
4349 DCBU(4*LAYOR+2)=SUL-ZAYOR
4350 CALL LEIND(DCBU,NBB,IJ,NECNP,NBB,56,2)
4351 216 CONTINUE
4352 CALL DISCO(4,58,IER)
4353 IF(IER.LT.0) STOP 4441
4354 DO 219 IJ=1,NECNP
4355 CALL LEIND(DCBU,NBB,IJ,NECNP,NBB,56,1)
4356 CALL LEIND(DCBU,NBB,IJ,NECNP,NBB,58,2)
4357 219 CONTINUE
4358 CALL LEIND(DCBU,NBB,NENF,NECNP,NBB,58,1)
4359 C
4360 C MODIFICACION DE LOS VIRTUALES Y DE BH^ * Y Y DE BF
4361 C
4362 215 CALL LEEDI(RMAT,LB1,LB1,NVIR,NH,50,1)
4363 DO 7 IJ=1,NVIR
4364 RMAT(IJ,NC)=(ZAYOR*RMAT(IJ,NC+1)-(SUL-ZAYOR)*RMAT(IJ,NC-1))/SUL
4365 RMAT(IJ,NC+2)=-RMAT(IJ,NC)
4366 7 CONTINUE
4367 CALL LEEDI(TETA,LB1,LB1,NVIR,NVIR,57,1)
```

```

4368 CALL PROMA(TETA,RMAT,LB1,LB1,LB1,NVIR,NVIR,NH)
4369 CALL LEEDI(RMAT,LB1,LB1,NVIR,NH,50,2)
4370 DO 8 IJ=1,NVIR
4371 DO 8 JJ=1,NH
4372 RMAT(JJ,IJ)=TETA(IJ,JJ)
4373 8 CONTINUE
4374 CALL LEEDI(RMAT,LB1,LB1,NH,NVIR,56,2)
4375 C
4376 C OBTENCION DE LA Z
4377 C
4378 DO 9 IJ=1,NMC
4379 CALL LEIND(T2,NH,NFILA(IJ),NH,NVIR,50,1)
4380 DO 11 II=1,NH
4381 D=0
4382 DO 12 JJ=1,NVIR
4383 D=D+RMAT(II,JJ)*T2(JJ)
4384 12 CONTINUE
4385 TETA(II,IJ)=D
4386 11 CONTINUE
4387 9 CONTINUE
4388 CALL LEEDI(TETA,LB1,LB1,NH,NMC,60,2)
4389 CALL LEEDI(TETA,LB1,LB1,NH,NMC,49,2)
4390 C
4391 C MODIFICACION DE LA Z TOTAL
4392 C
4393 IF(NCL.NE.14.AND.NCL.NE.16) GO TO 251
4394 CALL LEEDI(TETA,LB1,LB1,NVIR,NH,50,1)
4395 CALL PROMA(RMAT,TETA,LB1,LB1,LB1,NH,NVIR,NH)
4396 CALL LEEDI(RMAT,LB1,LB1,NH,NH,59,2)
4397 C
4398 C MODIFICACION DE B*h
4399 C
4400 CALL LEEDI(RMAT,LB1,LB1,NH,NB,54,1)
4401 DO 351 IK=1,NB
4402 RMAT(NC,IK)=(ZAYOR*RMAT(NC+1,IK)-(SUL-ZAYOR)*RMAT(NC-1,IK))/SUL
4403 RMAT(NC+2,IK)=-RMAT(NC,IK)
4404 351 CONTINUE
4405 CALL LEEDI(RMAT,LB1,LB1,NH,NB,54,2)
4406 C
4407 C MODIFICACION DE BF
4408 C
4409 251 CALL CAMZ(SU1,ZAYOR,Q(NC-1),Q(NC+1),QAYOR,1.,DCBU(4*LAYOR-3),
4410 @ DCBU(4*LAYOR+1),Q(NC))
4411 Q(NC+2)=-Q(NC)
4412 IF(NCL.NE.14.AND.NCL.NE.16) GO TO 252
4413 CALL CAMZ(SU1,ZAYOR,T(NC-1),T(NC+1),QAYOR,1.,DCBU(4*LAYOR-3),
4414 @ DCBU(4*LAYOR+1),T(NC))
4415 T(NC+2)=-T(NC)
4416 252 IF(NCL.NE.12.AND.NCL.NE.16) GO TO 217
4417 DO 206 IJ=1,NECNF
4418 CALL LEIND(DCBU,NBB,IJ,NECNF,NBB,58,1)
4419 W=-DCBU(4*LAYOR)*SIN(ALFA(LAYOR))+DCBU(4*LAYOR-1)*COS(ALFA(LAYOR))
4420 CALL CAMZ(SU1,ZAYOR,TCNP(NC-1,IJ),TCNP(NC+1,IJ),W,1.,
4421 @ DCBU(4*LAYOR-3),DCBU(4*LAYOR+1),TCNP(NC,IJ))
4422 TCNP(NC+2,IJ)=-TCNP(NC,IJ)

```

```

4423 206 CONTINUE
4424 CALL LEIND(DCBU,NBB,NENP,NECNP,NBB,58,1)
4425 GO TO 227
4426 C
4427 C MODIFICACION DE LAS ECUACIONES DE EQUILIBRIO
4428 C
4429 217 CALL LEIND(B,NND,1,1,NND,63,1)
4430 227 CALL ITER(RMAT,DCBU,NBCU,NMI,NB,ND,NI,NJ,IT,NH,RLON,2,LB1,ALFA,
4431 @ TETA,NCL,NBB,NECNP,FIFI,TFI,INDT,NENP,B,NND)
4432 IF(NCL,NE,14,AND,NCL,NE,16) GO TO 1234
4433 C
4434 C OBTENCION DE LOS NUEVOS AXILES
4435 C
4436 DO 726 IKM=1,NB
4437 B(IKM)=0.0
4438 726 CONTINUE
4439 IF(NCL,NE,16) GO TO 770
4440 IF(NENP,EQ,1) GO TO 770
4441 DO 671 IK=1,NB
4442 DO 673 IH=1,NENP-1
4443 B(IK)=B(IK)+TFI(IK,IH)*COTA(IH)
4444 673 CONTINUE
4445 671 CONTINUE
4446 C
4447 C OBTENCION DE LOS AXILES DE LA ETAPA ANTERIOR
4448 C
4449 770 NGT=0
4450 685 DO 675 IK=1,NB
4451 COLK=COLAA
4452 IF(ABS(COLK).LT.TOL) GO TO 676
4453 ATB(IK)=FIFI(IK)+B(IK)/COLK
4454 GO TO 675
4455 676 COLK=1
4456 ATB(IK)=B(IK)
4457 675 CONTINUE
4458 IF(NGT,EQ,1) GO TO 881
4459 CALL CAXIL(RMAT,RMOME,C2,ATB,RLON,RES,C1,NB,NH,NNO,COLK,COM,COL,
4460 @ EG,COI,LB1,TOL)
4461 C
4462 C MODIFICACION DE C6=A(*)M DE LA ETAPA ANTERIOR
4463 C
4464 DO 408 IK=1,NB
4465 CC=C1(IK)*RLON(IK)**3/(RINER(IK)**2)
4466 D1=CC/45.
4467 D2=CC*7./360.
4468 C6(2*IK-1)=-RZ(2*IK-1)*D1+RZ(2*IK)*D2
4469 C6(2*IK)=RZ(2*IK-1)*D2-RZ(2*IK)*D1
4470 408 CONTINUE
4471 C
4472 C OBTENCION DE LOS AXILES SUPUESTOS
4473 C
4474 NGT=1
4475 BG=COLAA
4476 COLAA=SUMF
4477 GO TO 685

```



```
4478 881 CALL CAXIL(RMAT,W3,W2,ATB,RLON,RES,W1,NB,NH,NNO,COLK,COM,COL,EG,
4479 @ CDI,LB1,TOL)
4480 COLAA=BG
4481 DO 723 IKM=1,NB
4482 AXIL(IKM)=W1(IKM)
4483 723 CONTINUE
4484 1234 KK=0
4485 CALL LEEDI(RMAT,LB1,LB1,NMI,NH,55,1)
4486 DO 13 IJ=1,NH
4487 DO 14 JJ=1,NVIR
4488 IF(IJ.EQ.ICOL(JJ)) GOTO 13
4489 14 CONTINUE
4490 KK=KK+1
4491 DO 15 K1=1,NMI
4492 TETA(KK,K1)=RMAT(K1,IJ)
4493 15 CONTINUE
4494 13 CONTINUE
4495 CALL INVER(TETA,NMI,LB1,IER)
4496 IF(IER.NE.0) GOTO 200
4497 C
4498 C PONER EN MEMORIA LA MATRIZ Z
4499 C
4500 DO 16 IJ=1,NMI
4501 DO 16 JJ=1,NMI
4502 RMAT(IJ,JJ)=TETA(IJ,JJ)
4503 16 CONTINUE
4504 CALL LEEDI(TETA,LB1,LB1,NH,NMC,60,1)
4505 CALL LEEDI(RMAT,LB1,LB1,NMI,NMI,60,2)
4506 GO TO 1
4507 2 CONTINUE
4508 1 CONTINUE
4509 RETURN
4510 200 STOP 5550
4511 END
```

FTN4X COMPILER: HP92834 REV.2130 (810716)

\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: 4109 COMMON: (NONE)

```
4512 SUBROUTINE CAMZ (RLZ,Z,R1,R2,Q,CAR,A,D,X)
4513 C
4514 C   ESTA RUTINA DETERMINA EL VALOR ACTUAL DE R1 Y R2 EN FUNCION DE Z
4515 C
4516 RL=RLZ+Z
4517 X=(-RLZ*R1+Z*R2-Q*CAR*0.5*(RLZ*(Z**2-A**2)+Z*(RLZ**2-D**2)))/RL
4518 RETURN
4519 END
```

FTN4X COMPILER: HP92834 REV.2130 (810716)

\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: 102 COMMON: (NONE)

```
4520      SUBROUTINE DELMO(A,B,C,D,E,D1,D2,SUL,ZAYOR)
4521 C
4522 C      ESTA RUTINA DETERMINA EL DELTA EN BARRAS CARGADAS UNIFORMENTE
4523 C      ESTE DELTA NO ES REAL SINO ESTA REFERIDO A LA LINEA QUE UNE LOS
4524 C      PUNTOS EXTREMOS DE LA BARRA
4525 C
4526      DELOR=(A-B)*COS(E)-(C-D)*SIN(E)
4527      D1=DELOR*ZAYOR/SUL
4528      D2=DELOR*(SUL-ZAYOR)/SUL
4529      RETURN
4530      END
```

FTN4X COMPILER: HP92834 REV.2130 (810716)

\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: 71 COMMON: (NONE)

```

4531      PROGRAM PAS3C(5)
4532 C
4533 C      SEGMENTO PARA EL CALCULO DE LA CARGA DE LA ETAPA
4534 C
4535      COMMON /VAR/ IN, IO, NOMB(30), ND, NB, INDT, NMI, NLI, NVIR, NAI, NBCU, NMH,
4536 @          NFI, NGD, NGP, NCL, LHF, LHV, NITER, NND, NH, NBB, NTCPV, NCFP,
4537 @          NCPV, NMIN, NGH, NMCO, NMHCO, NECC, NTRA, LV, CADI, SUMF, E,
4538 @          SUM, TOL, TOLS, SUM1, PESO, SIGMA, COP, COL, COI, COM, TCO
4539      COMMON /CON/ IT(25), RMP(25), RIMON(50), RMOME(50), NI(25), NJ(25),
4540 @          ALFA(25), RLON(25), RINER(25), RMP1(25), AXIL(25), XA(25),
4541 @          YA(25), ZA(25), DCL(20), R(75), LBE(20), T(50), LICOL(50),
4542 @          NCOL(50), NFILA(50), ICOL(50), DCBU(100), T4(50), Q(50),
4543 @          NSD(50), LTP(25), AREA(25), T2(50), RZ(50),
4544 @          C6(50), C3(50), FIFI(25), X(25), Y(25), Z(25),
4545 @          RLCPV(10, 2), TETA(50, 50), RMAT(50, 50), TCNP(50, 10),
4546 @          TFI(25, 10)
4547      COMMON /PAS/ NMC, NC, NETAPA, IND, COLAA, COLAP, RMINC(50), RFIC(50),
4548 @          MC(50), RLOF(25), NECNP, NENP, COTA(10), ROV(25, 6)
4549      COMMON /PAS2/ NN1, NN2, NN3, NN4, C1(25), C2(25), W1(25), W2(25), W3(50),
4550 @          RF1(50), RM1(50), QL1(50), RMPI(50)
4551      LB1=50
4552      CALL OBTCO(RIMON, RFIC, RMOME, RMP, MC, COLAP, COLAA, NH, NB, NMC, NC, TOL,
4553 @          TOLS, LB1, RMINC, COTA, NENP, NCL, IND, NECNP, RM1, RF1, AXIL)
4554 C
4555 C      LLAMADA AL SEMENTO CORRESPONDIENTE
4556 C
4557      CALL EXEC(8, 5HPAS04)
4558      END

```

FTN4X COMPILER: HP92834 REV.2130 (810716)

\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: 40 COMMON: (NONE)

```

4559 SUBROUTINE OBTCO(RIMON,RFIC,RMOME,RMP,MC,COLAP,COLAA,NH,NB,NMC,
4560 @ NC,TOL,TOLS,LB1,RMINC,COTA,NENP,NCL,INDI,NEC,RM1,
4561 @ RF1,AXIL)
4562 C
4563 C OBTENCION DE LA CARGA DE LA ETAPA
4564 C
4565 DIMENSION RIMON(NH),RFIC(NH),RMOME(NH),RMINC(NH),MC(NH),RMP(NB),
4566 @ COTA(NEC),RM1(NH),RF1(NH),AXIL(NB)
4567 COMMON /VAR/ IN,IO
4568 INDI=0
4569 COL1=COLAA
4570 CALL DECET(RMOME,RMINC,MC,RMP,COTA,NC,RINF,NCL,NENP,NH,NB,NEC,RM1,
4571 @ AXIL,TOLS,TOL)
4572 COLAP=RINF
4573 IF(NCL.NE.12.AND.NCL.NE.16) GO TO 3
4574 IF(ABS(COLAA+COLAP).GT.ABS(COTA(NENP))) GO TO 5
4575 GO TO 3
4576 5 COLAP=COTA(NENP)-COLAA
4577 INDI=1
4578 COLAA=COTA(NENP)
4579 GO TO 3
4580 3 COLAA=COLAP+COLAA
4581 MC(NC)=1
4582 8 LKB=0
4583 C
4584 C OBTENCION DEL MOMENTO Y DE LA ROTACION PLASTICA
4585 C
4586 DO 2 I=1,NH
4587 RMINC(I)=RMINC(I)*COLAP+RM1(I)
4588 RMOME(I)=RMOME(I)+RMINC(I)
4589 RFIC(I)=RFIC(I)*COLAP+RF1(I)
4590 RIMON(I)=RIMON(I)+RFIC(I)
4591 2 CONTINUE
4592 RETURN
4593 END

```

FTN4X COMPILER: HP92834 REV.2130 (810716)

\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: 286 COMMON: (NONE)

```
4594 SUBROUTINE DECET(RMOME, RMINC, MC, RMP, COTA, NC, RINP, NCL, NENP, NH, NB,  
4595 @ NECNP, RM1, AXIL, TOLS, TOL)  
4596 C  
4597 C ESTA SUBROUTINA DETERMINA EL INCREMENTO DE CARGA SEGURO DE LA ETAPA  
4598 C  
4599 DIMENSION RMINC(NH), MC(NH), RMP(NB), COTA(NECNP), RMOME(NH), RM1(NH),  
4600 @ AXIL(NB)  
4601 COMMON /VAR/ IN, IO  
4602 RINP=1.E36  
4603 J=0  
4604 DO 1 I=1, NH  
4605 J=I-J  
4606 IF(MC(I).EQ.1) RMINC(I)=0  
4607 IF(ABS(RMINC(I)).LE.TOL) GOTO 1  
4608 NSG=1  
4609 IF(NCL.EQ.12.OR.NCL.EQ.16) NSG=COTA(NENP)/ABS(COTA(NENP))  
4610 U=NSG*RMP(J)*ABS(RMINC(I))/RMINC(I)  
4611 IF(NCL.EQ.14.OR.NCL.EQ.16) U=U-RM1(I)  
4612 RIN=(U-RMOME(I))/RMINC(I)  
4613 IF(NSG.EQ.-1.AND.NCL.EQ.12) GO TO 2  
4614 IF(RIN.GT.RINP.OR.ABS(RINP).LE.TOL) GOTO 1  
4615 4 IF(ABS(ABS(RIN)-RINP).LT.TOL.AND.ABS(AXIL(NBB)).GE.ABS(AXIL(J)))  
4616 @GO TO 1  
4617 GO TO 3  
4618 2 IF(ABS(RIN).GT.ABS(RINP)) GO TO 1  
4619 GO TO 4  
4620 3 RINP=RIN  
4621 NC=I  
4622 NBB=J  
4623 1 CONTINUE  
4624 IF(ABS(RINP).LE.TOL) RINP=0.0  
4625 RETURN  
4626 END
```

FTN4X COMPILER: HP92834 REV.2130 (810716)

\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: 338 COMMON: (NONE)

```

4627 PROGRAM PAS04(5)
4628 C
4629 C SEGMENTO PARA LA OBTENCION DE LOS MOVIMIENTOS Y ESCRITURA DE
4630 C LA ETAPA.
4631 C
4632 DIMENSION RRR(100),RR1(75),X1(25),Y1(25),Z1(25),DC(100)
4633 COMMON /VAR/ IN,IO,NOMB(30),ND,NB,INDT,NMI,NLI,NVIR,NAI,NBCU,NMH,
4634 @ NFI,NGD,NGP,NCL,LHF,LHV,NITER,NND,NH,NBB,NTCPV,NCFP,
4635 @ NCPV,NMIN,NGH,NMCO,NMHCO,NECO,NTRA,LV,CADI,SUMF,E,
4636 @ SUM,TOL,TOLS,SUM1,PESO,SIGMA,COP,COL,COI,COM,TCO
4637 COMMON /CON/ IT(25),RMP(25),RIMON(50),RMOME(50),NI(25),NJ(25),
4638 @ ALFA(25),RLON(25),RINER(25),RMP1(25),AXIL(25),XA(25),
4639 @ YA(25),ZA(25),DCL(20),R(75),LBE(20),T(50),LICOL(50),
4640 @ NCOL(50),NFILA(50),ICOL(50),DCBU(100),RMAX(50),Q(50),
4641 @ NSD(50),LTP(25),AREA(25),T2(50),RZ(50),
4642 @ C6(50),C3(50),FIFI(25),X(25),Y(25),Z(25),
4643 @ RLCPV(10,2),TETA(50,50),RMAT(50,50),TCNP(50,10)
4644 COMMON /PAS/ NMC,NC,NETAPA,IND,COLAA,COLAP,RMINC(50),RFIC(50),
4645 @ MC(50),RLOF(25),NECNP,NENP,COTA(10),ROV(25,6)
4646 COMMON /PAS2/ NN1,NN2,NN3,NN4,C1(25),NVV(250),RF1(50),QL1(50)
4647 LB1=50
4648 NCCLL=NCL
4649 C
4650 C SI HAY BARRAS CON CARGA UNIFORME SE DETERMINA EL DESPLAZAMIENTO
4651 C TOTAL
4652 C
4653 DO 33 I=1,ND
4654 X1(I)=X(I)
4655 Y1(I)=Y(I)
4656 Z1(I)=Z(I)
4657 33 CONTINUE
4658 IF(NBCU.EQ.0) GO TO 30
4659 IF(NCL.EQ.14.OR.NCL.EQ.16) NCCLL=15
4660 CCAA=COLAA
4661 DO 36 I=1,NBB
4662 DC(I)=DCBU(I)
4663 36 CONTINUE
4664 IF(NCL.NE.16.AND.NCL.NE.12) GO TO 1700
4665 IF(NENP.EQ.1) GO TO 1700
4666 IF(ABS(CCAA).LT.TOL) CCAA=1
4667 DO 1701 I=1,NENP-1
4668 CALL LEIND(DCBU,NBB,I,NECNP,NBB,58,1)
4669 DO 1703 J=1,NB
4670 DC(4*J)=DC(4*J)+DCBU(4*J)*COTA(I)/CCAA
4671 DC(4*J-1)=DC(4*J-1)+DCBU(4*J-1)*COTA(I)/CCAA
4672 1703 CONTINUE
4673 1701 CONTINUE
4674 CALL LEIND(DCBU,NBB,NENP,NECNP,NBB,58,1)
4675 1700 CALL OBTMO(RMAT,NI,NJ,RLON,ALFA,DC,XA,YA,ZA,RMOME,RIMON,ICOL,
4676 @ 1.,NH,NVIR,NB,ND,INDT,LB1,NBCU,NMI,NLI,RRR,RR1,RINER,
4677 @ CCAA,IT,DCL,NBB,E,COM,COL,COI,AXIL,RMOME,NCCLL,C1,C3)
4678 DO 31 I=1,ND
4679 X(I)=XA(I)
4680 Y(I)=YA(I)
4681 Z(I)=ZA(I)

```

```

4682 31 CONTINUE
4683 GO TO 32
4684 30 CALL OSTMO(RMAT, NI, NJ, RLON, ALFA, DCBU, XA, YA, ZA, RMINC, RFIC, ICOL,
4685 @ 1., NH, NVIR, NB, ND, INDT, LB1, NBCU, NMI, NLI, RRR, RR1, RINER,
4686 @ COLAP, IT, DCL, NBB, E, COM, COL, COI, AXIL, RMOME, NCL, C1, C3)
4687 DO 21 I=1, ND
4688 X(I)=XA(I)+X1(I)
4689 Y(I)=YA(I)+Y1(I)
4690 Z(I)=ZA(I)+Z1(I)
4691 21 CONTINUE
4692 32 IF(NBCU.EQ.0) GO TO 16
4693 IF(NCL.EQ.14.OR.NCL.EQ.16) GO TO 35
4694 CALL VARIA(RMOME, RIMON, X, Y, Z, DCBU, IT, NJ, COTA, ALFA, RLON, RINER, LBE,
4695 @ NSD, ROV, NB, NH, ND, NBB, NLI, NECNP, NENP, NI, NCL, DCL, COLAA)
4696 35 DO 37 I=1, ND
4697 XA(I)=X(I)-X1(I)
4698 YA(I)=Y(I)-Y1(I)
4699 ZA(I)=Z(I)-Z1(I)
4700 37 CONTINUE
4701 16 IF(NCL.EQ.14.OR.NCL.EQ.16) GO TO 13
4702 IF(NCL.EQ.12.AND.ABS(COLAA-ABS(COTA(NENP))), LT.TOLS) GO TO 13
4703 IF(ABS(COLAP*COM/(COP*COL)).LE.TOLS) GO TO 10
4704 13 NETAPA=NETAPA+1
4705 IF(NCL.EQ.14.AND.ISSW(3).GE.0) GO TO 1
4706 IF(NCL.EQ.16.AND.ISSW(3).GE.0) GO TO 1
4707 CALL SECON(BVC)
4708 CALL IMPRE(RMINC, RFIC, RMOME, RIMON, NETAPA, COLAP, COLAA, X, Y, Z, ND, NH,
4709 @ NB, RMF, DCL, NBCU, DCBU, NLI, IT, NJ, NI, LBE, COP, COL, COI, COM, E,
4710 @ NBB, NCL, IND, NENP, AXIL)
4711 CALL SECON(BVC1)
4712 PESO=PESO+BVC1-BVC
4713 1 IF(NCL.NE.12) GO TO 10
4714 IF(IND.NE.1) GO TO 10
4715 NENP=NENP+1
4716 IF(NCL.NE.16) COLAA=0.0
4717 IF(NENP.GT.NECNP) GO TO 11
4718 DO 3 I=1, NH
4719 Q(I)=TCNP(I, NENP)
4720 T(I)=TCNP(I, NENP)
4721 3 CONTINUE
4722 IF(NBCU.EQ.0) GO TO 10
4723 CALL LEIND(DCBU, NBB, NENP, NECNP, NBB, 58, 1)
4724 10 IF(NCL.EQ.14.OR.NCL.EQ.16) GO TO 101
4725 IF(ABS(COLAP*COM/(COP*COL)).LE.TOLS) GO TO 20
4726 C
4727 C LLAMADA AL SEGMENTO CORRESPONDIENTE
4728 C
4729 CALL EXEC(8, 5HPAS02)
4730 101 CALL EXEC(8, 5HPAS05)
4731 20 NGD=1
4732 22 CALL EXEC(8, 5HPAS02)
4733 11 CALL SECON(RV)
4734 TEMP=RV-TCO-PESO
4735 WRITE(10, 15) TEMP
4736 STOP

```



4737 C

4738 C     FORMATOS DE ESCRITURA

4739 C

4740 15     FORMAT(2/,10X,"\*\*\* TIEMPO DE C.P.U. EMPLEADO.....",F10.3,"SG.")

4741     END

FTN4X COMPILER: HP92834 REV.2130 (810716)

\*\* NO WARNINGS \*\* NO ERRORS \*\*     PROGRAM: 1560     COMMON: (NONE)

```

4742      SUBROUTINE OBTMO(RMAT,NI,NJ,RLON,ALFA,DCBU,X,Y,Z,RMOME,RIMON,ICOL,
4743 @      E,NH,NVIR,NB,ND,INDT,LB1,NBCU,NMI,NLI,RRR,RR1,RINER,
4744 @      COLAA,IT,DCL,NBB,EE,COM,COL,COI,AXIL,RMO,NCL,C1,C3)
4745 C
4746 C      OBTENCION DE LOS MOVIMIENTOS DE LA ETAPA
4747 C
4748      DIMENSION RMAT(LB1,LB1),NI(NB),NJ(NB),RLON(NB),RINER(NB),ALFA(NB),
4749 @      DCBU(NBB),DCL(NLI),X(ND),Y(ND),Z(ND),RMOME(NH),ICOL(NH),
4750 @      RIMON(NH),RRR(NH),RR1(NH),IT(ND),AXIL(NB),RMO(NH),
4751 @      C1(NB),C3(NH)
4752      COMMON /VAR/ IN,IO
4753 C
4754 C      LLAMADA A LA RUTINA DE CALCULO DE DEFORMACIONES
4755 C
4756      DO 1 I=1,NH
4757      RRR(I)=0.0
4758      RR1(I)=0.0
4759 1      CONTINUE
4760      CALL DEFO(RR1,RRR,RMOME,RLON,RINER,RIMON,IT,NB,ND,NH,E,TOL,1,
4761 @      DCBU,NBCU,ALFA,NJ,AXIL,COM,COL,EE,COI,NBB,NH,RMO,NCL,
4762 @      C1,C3,COLAA)
4763 C
4764 C      PREPARACION DEL VECTOR DE DEFORMACIONES
4765 C
4766      KK1=0
4767      DO 802 I=1,NH
4768      DO 803 J=1,NVIR
4769      IF(I.EQ.ICOL(J)) GO TO 802
4770 803      CONTINUE
4771      KK1=KK1+1
4772      RR1(KK1)=RRR(I)
4773 802      CONTINUE
4774      CALL LEEDI(RMAT,LB1,LB1,NMI,NMI,60,1)
4775      DO 804 I=1,NMI
4776      D=0
4777      DO 805 J=1,NMI
4778      D=D+RMAT(I,J)*RR1(J)
4779 805      CONTINUE
4780      RRR(I)=D
4781 804      CONTINUE
4782 C
4783 C      LLAMADA A LA RUTINA DE CALCULO DE MOVIMIENTOS
4784 C
4785      CALL MOVIT(RMAT,IT,NI,NJ,RLON,ALFA,DCBU,RRR,DCL,X,Y,Z,NB,ND,
4786 @      NH,LB1,INDT,NBCU,NMI,NLI,NBB)
4787      RETURN
4788      END

```

FTN4X COMPILER: HP92834 REV.2130 (810716)

\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: 383 COMMON: (NONE)

```

4789 SUBROUTINE IMPRE(RMINC,RFIC,RMOME,RIMON,NETAPA,COLAP,COLAA,X,Y,
4790 @ Z,ND,NH,NB,RMP,DCL,NBCU,DCBU,NLI,IT,NJ,NI,LBE,
4791 @ COP,COL,COI,COM,E,NBB,NCL,INDO,NENP,AXIL)
4792 C
4793 C ESCRITURA DE LA ETAPA
4794 C
4795 DIMENSION RMINC(NH),RFIC(NH),RMOME(NH),RIMON(NH),X(ND),Y(ND),
4796 @ RMP(NB),DCBU(NBB),LBE(NLI),NI(NB),NJ(NB),DCL(NLI),
4797 @ Z(ND),IT(ND),AXIL(NB)
4798 COMMON /VAR/ IN,IO,NOMB(30)
4799 WRITE(IO,15) NOMB
4800 CO=COLAP*COM/(COP*COL)
4801 CA=COLAA*COM/(COP*COL)
4802 IF(NCL.NE.12.AND.NCL.NE.16) GO TO 16
4803 IF(INDO.NE.1) GO TO 17
4804 WRITE(IO,18) NENP
4805 GO TO 16
4806 17 WRITE(IO,19) NENP
4807 16 WRITE(IO,1) NETAPA,CO,CA
4808 IF(NCL.EQ.14.OR.NCL.EQ.16) GO TO 561
4809 WRITE(IO,2)
4810 GO TO 562
4811 561 WRITE(IO,578)
4812 562 DO 3 I=1,NB
4813 IL=NI(I)
4814 JL=NJ(I)
4815 K1=2*I-1
4816 K2=2*I
4817 G1=RMINC(K1)*COM
4818 H1=RMOME(K1)*COM
4819 H2=RMOME(K2)*COM
4820 G2=RMINC(K2)*COM
4821 F1=RFIC(K1)*COM*COL/(E*COI)
4822 E1=RIMON(K1)*COM*COL/(E*COI)
4823 E2=RIMON(K2)*COM*COL/(E*COI)
4824 F2=RFIC(K2)*COM*COL/(E*COI)
4825 HA=RMP(I)*COM
4826 HG=AXIL(I)*COM/COL
4827 IF(NCL.EQ.14.OR.NCL.EQ.16) GO TO 560
4828 WRITE(IO,4) I,IL,HA,H1,G1,E1,F1,JL,H2,G2,E2,F2
4829 GO TO 3
4830 560 WRITE(IO,579) I,IL,HA,HG,H1,G1,E1,F1,JL,H2,G2,E2,F2
4831 3 CONTINUE
4832 WRITE(IO,5)
4833 DO 6 I=1,ND
4834 XX=X(I)*COM*COL*COL/(E*COI)
4835 YY=Y(I)*COM*COL*COL/(E*COI)
4836 ZZ=Z(I)*COM*COL/(E*COI)
4837 WRITE(IO,7) I,XX,YY,ZZ
4838 6 CONTINUE
4839 IF(NLI.EQ.0) GO TO 20
4840 WRITE(IO,21)
4841 DO 22 I=1,NLI
4842 LB=LBE(I)/100
4843 LN=LBE(I)-LB*100

```

```

4844      DC=DCL(I)*COM*COL/(E*COI)
4845      WRITE(IO,23) I, LB, LN, DC
4846 22    CONTINUE
4847 20    IF(NBCU.EQ.0) RETURN
4848      WRITE(IO,8)
4849      DO 9 I=1,ND
4850      IF(IT(I).NE.6) GO TO 9
4851      DO 10 J=1,NB
4852      IF(NJ(J).NE.1) GO TO 10
4853      DC=DCBU(4*J-2)*COL
4854      WRITE(IO,11) I, NI(J), DC
4855      GO TO 9
4856 10    CONTINUE
4857 9      CONTINUE
4858      RETURN
4859 C
4860 C      FORMATOS DE ESCRITURA
4861 C
4862 15    FORMAT(1H1,10X,30A2/10X,60("*")3/)
4863 18    FORMAT(10X,"FINAL DE LA ETAPA DE CARGA **** ",I2//)
4864 19    FORMAT(10X,"ETAPA DE CARGA.....",I8/)
4865 1      FORMAT(10X,"ETAPA.....",I8/10X,
4866 @      "INCREMENTO DE CARGA.....",E13.6/10X,
4867 @      "CARGA TOTAL.....",E13.6//)
4868 2      FORMAT(10X,"DISTRIBUCION DE MOMENTOS Y ROTACIONES PLASTICAS EN LA
4869 @ETAPA :"/10X,59("-")//10X,"BARRA NUDO M. PLASTICO MOMENTO I
4870 @NCR. MOM. ROTACION INCR. ROT."/)
4871 578    FORMAT(10X,"DISTRIBUCION DE ESFUERZOS Y ROTACIONES PLASTICAS EN LA
4872 @ ETAPA :"/10X,60("-")//10X,"BARRA NUDO M.PLAST. AXIL MOMEN
4873 @TO INCR. MOM. ROTACION INCR. ROT."/)
4874 579    FORMAT(10X,I4,I5,2X,E9.4,1X,E9.3,E10.4,1X,E10.4,1X,E10.4,1X,
4875 @      E10.4,/,14X,I5,21X,E10.4,1X,E10.4,1X,E10.4,1X,E10.4)
4876 4      FORMAT(10X,I4,2X,I3,4X,E10.4,2X,E10.4,2X,E10.4,2X,E10.4,2X,E10.4,
4877 @/16X,I3,16X,E10.4,2X,E10.4,2X,E10.4,2X,E10.4)
4878 5      FORMAT(///,10X,"MOVIMIENTOS DE LOS NUDOS :"/10X,24("-")//10X,
4879 @ "NUDO MOV. X MOV. Y GIRO"/)
4880 7      FORMAT(10X,I3,1X,F12.5,1X,F12.5,1X,F12.5)
4881 21     FORMAT(2/,10X,"GIRO EN LIBERTADES :"/10X,19("-"),2/,10X,
4882 @ "LIBERTAD BARRA NUDO GIRO"/)
4883 23     FORMAT(12X,I2,9X,I2,7X,I2,4X,E12.4)
4884 8      FORMAT(2/,10X,"POSICION DE LOS NUDOS MOVILES :"/10X,29("-")2/,
4885 @10X,"NUDO NUDO ORIG. DISTANCIA"/)
4886 11     FORMAT(10X,I3,7X,I3,9X,F12.4)
4887      END

```

FTN4X COMPILER: HP92834 REV.2130 (810716)

\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: 1447 COMMON: (NONE)

```

4888      SUBROUTINE VARIA(RMOME,RIMON,X,Y,Z,DCBU,IT,NJ,COTA,ALFA,RLON,
4889 @          RINER,LBE,NSD,ROV,NB,NH,ND,NBB,NLI,NECNP,NENP,
4890 @          NI,NCL,DCL,COLAA)
4891 C
4892 C      RUTINA PARA LA DETERMINACION DE MOVIMIENTOS SEGUN LA ECUACION
4893 C      GENERAL DE LA ELASTICA
4894 C
4895      DIMENSION RMOME(NH),RIMON(NH),X(ND),Y(ND),Z(ND),DCBU(NBB),
4896 @          IT(ND),NJ(NB),NI(NB),COTA(NECNP),ALFA(NB),RINER(NB),
4897 @          RLON(NB),LBE(20),NSD(NH),ROV(NB,6),DCL(20)
4898      COMMON /VAR/ IN,IO
4899      DO 1 I=1,ND
4900      IF(IT(I).NE.6) GO TO 1
4901      DO 2 J=1,NB
4902      IF(I.NE.NJ(J)) GO TO 2
4903      IF(NSD(2*J).EQ.0.AND.NSD(2*J+2).EQ.0) GO TO 2
4904 C
4905 C      EL MOVIMIENTO DEL NUDO VARIABLE SE DETERMINA POR LA ECUACION
4906 C      UNIVERSAL DE LA ELASTICA EN PRESENCIA DE DISCONTINUIDADES DE
4907 C      GIRO
4908 C
4909      SUL=RLON(J)+RLON(J+1)
4910      ZZ=DCBU(4*J-2)
4911      QQ=0.0
4912 C
4913 C      EN CASO DE CARGA NO PROPORCIONAL SE DETERMINA LA CARGA EN BARRA
4914 C
4915      IF(NCL.NE.12.AND.NCL.NE.16) GO TO 3
4916      IF(NENP.EQ.1) GO TO 3
4917      DO 4 II=1,NENP-1
4918      CALL LEIND(DCBU,NBB,II,NECNP,NBB,58,1)
4919      QQ=QQ+(-DCBU(4*J)*SIN(ALFA(J))+DCBU(4*J-1)*COS(ALFA(J)))*COTA(II)
4920 4      CONTINUE
4921      CALL LEIND(DCBU,NBB,NENP,NECNP,NBB,58,1)
4922 3      QQ=QQ+(-DCBU(4*J)*SIN(ALFA(J))+DCBU(4*J-1)*COS(ALFA(J)))*COLAA
4923      C=SUL-DCBU(4*J-3)-DCBU(4*J+1)
4924 C
4925 C      DETERMINACION DEL CORTANTE EN EL ORIGEN DE LA BARRA
4926 C
4927      V=(RMOME(2*J-1)+RMOME(2*J+1)-QQ*C*(0.5*C+DCBU(4*J+1)))/SUL
4928      VA=(-X(NI(J))*SIN(ALFA(J))+Y(NI(J))*COS(ALFA(J)))*RINER(J)
4929      VH=(X(NI(J))*COS(ALFA(J))+Y(NI(J))*SIN(ALFA(J)))*RINER(J)
4930 C
4931 C      DETERMINACION DEL ANGULO GIRADO POR LA BARRA EN LA SECCION INICIAL
4932 C
4933      IF(NLI.EQ.0) GO TO 10
4934      DO 11 II=1,NLI
4935      LB=LBE(II)/100
4936      LN=LBE(II)-LB*100
4937      IF(J.NE.LB) GO TO 11
4938      IF(LN.NE.NI(LB)) GO TO 11
4939      ZA=DCL(II)*RINER(J)
4940      GO TO 12
4941 11      CONTINUE
4942 10      ZA=(Z(NI(J))-RIMON(2*J-1))*RINER(J)

```

```
4943 12 IF(NSD(2*J).EQ.0) GO TO 13
4944 NN=NSD(2*J)
4945 16 DO 15 II=1,NN
4946 IF(ROV(J,2*II-1).GE.ZZ) GO TO 15
4947 ZA=ZA+ROV(J,2*II)
4948 VA=VA+ROV(J,2*II)*(ZZ-ROV(J,2*II-1))
4949 15 CONTINUE
4950 GO TO 17
4951 13 IF(NSD(2*J+2).EQ.0) GO TO 17
4952 NN=NSD(2*J+2)
4953 GO TO 16
4954 17 Z(I)=ZA-RMOME(2*J-1)*ZZ+0.5*V*ZZ**2+(QQ*(ZZ-DCBU(4*J-3))**3)/6.
4955 Z(I)=Z(I)/RINER(J)
4956 VV=VA+ZA*ZZ-0.5*RMOME(2*J-1)*ZZ**2+(V*ZZ**3)/6.
4957 VV=(VV+(QQ*(ZZ-DCBU(4*J-3))**4)/24.)/RINER(J)
4958 VVH=VH/RINER(J)
4959 X(I)=-VV*SIN(ALFA(J))+VVH*COS(ALFA(J))
4960 Y(I)=VV*COS(ALFA(J))+VVH*SIN(ALFA(J))
4961 Z(I)=Z(I)+RIMON(2*J)
4962 GO TO 1
4963 2 CONTINUE
4964 1 CONTINUE
4965 RETURN
4966 END
```

FTN4X COMPILER: HP92834 REV.2130 (810716)

\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: 971 COMMON: (NONE)

```

4967      PROGRAM PAS05(5)
4968 C
4969 C      SEGMENTO PARA LA OBTENCION DE LOS AXILES Y MODIFICACION DEL
4970 C      VECTOR DE CARGAS EN UN ANALISIS ELASTOPLASTICO DE 2 ORDEN
4971 C
4972      DIMENSION RES(50),CU1(50),CU2(50),CU3(50),CU4(50)
4973      COMMON /VAR/ IN, IO, NOMB(30), ND, NB, INDT, NMI, NLI, NVIR, NAI, NBCU, NMH,
4974 @          NFI, NGD, NGP, NCL, LHF, LHV, NITER, NND, NH, NBB, NTCPV, NCFP,
4975 @          NCFV, NMIN, NGH, NMCO, NMHCO, NECO, NTRA, LV, CADI, SUMF, E,
4976 @          SUM, TOL, TOLS, SUM1, PESO, SIGMA, COP, COL, CDI, COM, TCO
4977      COMMON /CON/ IT(25),RMP(25),RIMON(50),RMOME(50),NI(25),NJ(25),
4978 @          ALFA(25),RLON(25),RINER(25),RMP1(25),AXIL(25),XA(25),
4979 @          YA(25),ZA(25),DCL(20),R(75),LBE(20),T(50),LICOL(50),
4980 @          NCOL(50),NFILA(50),ICOL(50),DCBU(100),T4(50),Q(50),
4981 @          NSI(50),LTP(25),AREA(25),T2(50),RZ(50),
4982 @          C6(50),C3(50),FIFI(25),X(25),Y(25),Z(25),
4983 @          RLCFV(10,2),TETA(50,50),RMAT(50,50),TCNF(50,10),
4984 @          TFI(25,10)
4985      COMMON /PAS/ NMC, NC, NETAPA, IND, COLAA, COLAP, RMINC(50), RFIC(50),
4986 @          MC(50), RLOF(25), NECNF, NENF, COTA(10)
4987      COMMON /PAS2/ NN1, NN2, NN3, NN4, C1(25), C2(25), W1(25), W2(25), W3(50),
4988 @          RF1(50), RM1(50), QL1(50), RMP1(50), NBVC, NMA(50),
4989 @          NMF(50)
4990      LB1=50
4991 C
4992 C      PUESTA A CERO DE CONJUNTOS
4993 C
4994      CALL LEEDI(RMAT, LB1, LB1, NB, NH, 52, 1)
4995      DO 1 I=1, NB
4996      T4(2*I-1)=0.0
4997      T4(2*I)=0.0
4998      RZ(I)=0.0
4999      T2(I)=0.0
5000 1      CONTINUE
5001 C
5002 C      EN CASO DE CARGA NO PROPORCIONAL FORMAR EL VECTOR DE CARGAS
5003 C      EXTERNAS RESULTANTE
5004 C
5005      IF(NENF.EQ.1) GO TO 677
5006      DO 671 IK=1, NB
5007      DO 673 IH=1, NENF-1
5008      RES(IK)=RES(IK)+TFI(IK, IH)*COTA(IH)
5009 673      CONTINUE
5010 671      CONTINUE
5011 677      DO 675 IK=1, NB
5012      COLK=COLAA
5013      IF(ABS(COLK).LT.TOL) GO TO 676
5014      RES(IK)=FIFI(IK)+RES(IK)/COLK
5015      GO TO 675
5016 676      COLK=1
5017 675      CONTINUE
5018 C
5019 C      DETERMINACION DEL DELTA ACTUAL DE LA BARRA Y DE LOS AXILES
5020 C
5021      CALL DELTA(Y, X, ALFA, T2, NI, NJ, NB, ND, NB, IT, RLON)

```

```

5022      LL=0
5023      CALL CAXIL(RMAT,RMOME,T2,RES,RLON,RZ,AXIL,NB,NH,NN1,
5024 @      COLK,COM,COL,E,COI,LB1)
5025 C
5026 C      SI NN2=1 DESPRECIAR AXIL EN BARRAS
5027 C
5028      DO 2 I=1,NB
5029      RES(I)=0.0
5030      IF(NN2.EQ.1.AND.AREA(I).GT.10000.) AXIL(I)=0.0
5031 2      CONTINUE
5032      RCOF=COM/(COP*COL)
5033      IF(RCOF.LT.1.) RCOF=1.
5034      RCOL=ABS(TOLS*SUMF*RCOF)
5035 C
5036 C      DETERMINACION DE A(*) * M
5037 C
5038      DO 3 I=1,NB
5039      C=AXIL(I)*RLON(I)**3/(RINER(I)**2)
5040      D1=C/45.
5041      D2=7.*C/360.
5042      T4(2*I-1)=-RMOME(2*I-1)*D1+RMOME(2*I)*D2
5043      T4(2*I)=RMOME(2*I-1)*D2-RMOME(2*I)*D1
5044 3      CONTINUE
5045 C
5046 C      MODIFICACION DE LOS VECTORES
5047 C
5048 C
5049 C      OBTENCION DE Z*INCR(A(N)*M)
5050 C
5051      CALL LEEDI(RMAT,LB1,LB1,NH,NH,59,1)
5052      IF(NBCU.EQ.0) GO TO 151
5053      DO 152 I=1,NB
5054      QQ=-DCBU(4*I)*SIN(ALFA(I))+DCBU(4*I-1)*SIN(ALFA(I))
5055      CC=QQ*RLON(I)**5/(240.*RINER(I)**2)
5056      CU1(2*I-1)=-CC*AXIL(I)
5057      CU1(2*I)=CC*AXIL(I)
5058      CU2(2*I-1)=-CC*C1(I)
5059      CU2(2*I)=CC*C1(I)
5060 152      CONTINUE
5061 151      DO 4 I=1,NH
5062      D=0
5063      DD=0
5064      DDD=0
5065      DDDD=0
5066      DO 5 J=1,NH
5067      D=D+RMAT(I,J)*T4(J)
5068      DD=DD+RMAT(I,J)*C6(J)
5069      DDD=DDD+RMAT(I,J)*CU1(J)
5070      DDDD=DDDD+RMAT(I,J)*(CU1(J)-CU2(J))
5071 5      CONTINUE
5072      QL1(I)=D
5073      QL1(I)=TOLE(QL1(I),TOL)
5074      RES(I)=-DD
5075      RES(I)=TOLE(RES(I),TOL)
5076      CU3(I)=DDD

```



```
5077      CU4(I)=DDDD
5078 4      CONTINUE
5079 C
5080 C      OBTENCION DE B*h*INCR(N*DELTA)
5081 C
5082      CALL LEEDI(RMAT, LB1, LB1, NH, NB, 54, 1)
5083      DO 6 I=1, NH
5084          D=0
5085          DD=0.0
5086          DO 7 J=1, NB
5087              D=D+RMAT(I, J)*AXIL(J)*T2(J)
5088              DD=DD+RMAT(I, J)*C1(J)*C2(J)
5089 7      CONTINUE
5090          DD=TOLE(DD, TOL)
5091          D=TOLE(D, TOL)
5092          RES(I)=RES(I)-DD
5093          QL1(I)=QL1(I)+D
5094 6      CONTINUE
5095          CCAA=COLAA
5096          IF(NCL.NE.16) GO TO 65
5097          IF(NENP.EQ.1) GO TO 65
5098          P=0
5099          DO 62 I=1, NENP-1
5100              P=P+COTA(I)
5101 62     CONTINUE
5102          CCAA=CCAA+P
5103 65     DO 8 I=1, NH
5104             RES(I)=(RES(I)+QL1(I))*((CCAA-COLAP)/CCAA)
5105             RES(I)=RES(I)+CU4(I)*(CCAA-COLAP)
5106             RES(I)=TOLE(RES(I), TOL)
5107             RES(I)=RES(I)*COM*COL/(E*COI)
5108             QL1(I)=QL1(I)/CCAA+CU3(I)
5109             QL1(I)=QL1(I)*COM*COL/(E*COI)
5110             QL1(I)=TOLE(QL1(I), TOL)
5111 8      CONTINUE
5112          DO 9 I=1, NH
5113              Q(I)=T(I)+QL1(I)
5114              QL1(I)=RES(I)
5115              QL1(I)=TOLE(QL1(I), TOL)
5116              Q(I)=TOLE(Q(I), TOL)
5117 9      CONTINUE
5118 C
5119 C      COMPROBACION DE LA SOLUCION OBTENIDA
5120 C
5121          DO 60 I=1, NH
5122              PO=(ABS(RMOME(I))-ABS(W3(I)))
5123              CORL=RMOME(I)
5124              IF(ABS(CORL).LT.TOLS) CORL=COM
5125              PO=PO/CORL
5126              IF(ABS(PO).GT.0.005) GO TO 61
5127 60     CONTINUE
5128          DO 765 I=1, NB
5129              PO=ABS(AXIL(I))-ABS(W1(I))
5130              CORL=ABS(AXIL(I))
5131              IF(CORL.LT.TOLS) CORL=1.0
```

```
5132      PO=ABS(PO)/CORL
5133      IF(PO.GT.0.005) GO TO 61
5134 765   CONTINUE
5135      IF(ABS(SUMF-COLAA)/ABS(COLAA).LT.0.005) GO TO 10
5136 C
5137 C      SI LA TOLERANCIA ES LA DESEADA PASAR A UNA NUEVA ETAPA. SI NO ITERAF
5138 C
5139 61     DO 11 I=1,ND
5140      X(I)=X(I)-XA(I)
5141      Y(I)=Y(I)-YA(I)
5142      Z(I)=Z(I)-ZA(I)
5143 11     CONTINUE
5144      DO 12 I=1,NH
5145      W3(I)=RMOME(I)
5146      T4(I)=RIMON(I)
5147      RMOME(I)=RMOME(I)-RMINC(I)
5148      RIMON(I)=RIMON(I)-RFIC(I)
5149      MC(I)=NMA(I)
5150      NFILA(I)=NMF(I)
5151 12     CONTINUE
5152      SUMF=COLAA
5153      COLAA=COLAA-COLAP
5154      DO 13 I=1,NB
5155      W1(I)=AXIL(I)
5156      W2(I)=T2(I)
5157 13     CONTINUE
5158      NETAPA=NETAPA-1
5159      NC=NCFP
5160      NMC=NBVC
5161      INPA=1
5162      IF(NBVC.EQ.0) GO TO 14
5163 C
5164 C      MODIFICACION DE Z INV
5165 C
5166      CALL LEEDI(RMAT,LB1,LB1,NMC,NMC,48,1)
5167      CALL LEEDI(TETA,LB1,LB1,NH,NMC,49,1)
5168      CALL LEEDI(RMAT,LB1,LB1,NMC,NMC,61,2)
5169      GO TO 14
5170 C
5171 C      ESCRITURA DE LA ETAPA
5172 C
5173 10     CALL SECON (BVC)
5174      INPA=0
5175      CALL IMPRE(RMINC,RFIC,RMOME,RIMON,NETAPA,COLAP,COLAA,X,Y,Z,ND,NH,
5176 @          NB,RMP,DCL,NBCU,DCBU,NLI,IT,NJ,NI,LBE,COP,COL,COI,COM,E,
5177 @          NBB,NCL,IND,NENP,AXIL)
5178      NGP=0
5179      IF(IND.EQ.1) NGP=1
5180      CALL SECON(BVC1)
5181      PESO=PESO+BVC1-BVC
5182 C
5183 C      SI PROCEDE ACTUALIZAR MOMENTOS PLASTICOS E INCREMENTOS DE MOMENTOS
5184 C
5185 14     IF(NN3.EQ.1) GO TO 16
5186      DO 17 I=1,NH
```

```
5187      RMPI(I)=0.0
5188 17    CONTINUE
5189      DO 18 I=1,NB
5190      P=RMP1(I)
5191      IF(-AXIL(I)*COM/(COL*SIGMA*AREA(I)).LE.0.1525424) GO TO 19
5192      P=1.18*RMP1(I)*(1+AXIL(I)*COM/(COL*SIGMA*AREA(I)))
5193      IF(RIMON(2*I-1).EQ.0.0) GO TO 20
5194      RMPI(2*I-1)=P*RMOME(2*I-1)/(ABS(RMOME(2*I-1)))-C3(2*I-1)
5195 20     IF(RIMON(2*I).EQ.0.0) GO TO 21
5196      RMPI(2*I)=P*RMOME(2*I)/(ABS(RMOME(2*I)))-C3(2*I)
5197 21     IF(ABS(ABS(RMOME(2*I-1))-RMP(I)).GT.RMP(I)/1000.) GO TO 22
5198      RMOME(2*I-1)=P*RMOME(2*I-1)/(ABS(RMOME(2*I-1)))
5199 22     IF(ABS(ABS(RMOME(2*I))-RMP(I)).GT.RMP(I)/1000.) GO TO 19
5200      RMOME(2*I)=P*RMOME(2*I)/(ABS(RMOME(2*I)))
5201 19     RMP(I)=P
5202 18     CONTINUE
5203 16     IF(INFA.EQ.1) GO TO 30
5204      DO 33 I=1,NH
5205      C4(I)=T4(I)
5206      C3(I)=RMOME(I)
5207      Q(I)=T(I)
5208      QL1(I)=0.0
5209      NMA(I)=MC(I)
5210      NMF(I)=NFILA(I)
5211      W3(I)=RMOME(I)
5212 33     CONTINUE
5213      SUMF=COLAA
5214      IF(NMC.EQ.0) GO TO 35
5215      CALL LEEDI(RMAT,LB1,LB1,NMC,NMC,61,1)
5216      CALL LEEDI(RMAT,LB1,LB1,NMC,NMC,48,2)
5217      CALL LEEDI(TETA,LB1,LB1,NH,NMC,49,2)
5218 35     NBVC=NMC
5219      DO 36 I=1,NB
5220      C1(I)=AXIL(I)
5221      W1(I)=AXIL(I)
5222      C2(I)=T2(I)
5223      W2(I)=T2(I)
5224 36     CONTINUE
5225      DO 75 I=1,ND
5226      XA(I)=0.0
5227      YA(I)=0.0
5228      ZA(I)=0.0
5229 75     CONTINUE
5230      NCFP=NC
5231      IF(NCL.NE.16.OR.IND.NE.1) GO TO 30
5232      NENP=NENP+1
5233      COLAA=0.0
5234      IF(NENP.GT.NECNP) GO TO 55
5235      IND=0
5236      DO 56 I=1,NH
5237      Q(I)=TCNP(I,NENP)
5238      T(I)=TCNP(I,NENP)
5239      QL1(I)=0.0
5240 56     CONTINUE
5241      SUMF=0.0
```

```
5242      IF(NBCU.EQ.0) GO TO 30
5243      CALL LEIND(DCBU,NBB,NENF,NECNF,NBB,58,1)
5244 30    CALL EXEC(8,5HPAS02)
5245 55    CALL SECON (RV)
5246      TEMP=RV-TCO-PESO
5247      WRITE(IO,57) TEMP
5248 57    FORMAT(2/,10X,"*** TIEMPO DE C.P.U. EMPLEADO.....",F10.3,"SG.")
5249      STOP
5250      END
```

FTN4X COMPILER: HP92834 REV.2130 (810716)

\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: 2854 COMMON: (NONE)

```

5251      PROGRAM SHD01(5)
5252 C
5253 C      SEGMENTO PARA LA PREPARACION Y OBTENCION LA ENVOLVENTE MAXIMA
5254 C      Y MINIMA DE MOMENTOS ELASTICOS.
5255 C
5256      COMMON /VAR/ IN, IO, NOMB(30), ND, NB, INDT, NMI, NLI, NVIR, NAI, NBCU, NMH,
5257 @          NFI, NGD, NGP, NCL, LHF, LHV, NITER, NND, NH, NBB, NTCPV, NCFP,
5258 @          NCPV, NMIN, NGH, NMCO, NMHCO, NECO, NTRA, LV, CADI, SUMF, E,
5259 @          SUM, TOL, TOLS, SUM1, PESO, SIGMA, COP, COL, COI, COM, TCO
5260      COMMON /CON/ IT(25), RMP(25), RIMON(50), RMOME(50), NI(25), NJ(25),
5261 @          ALFA(25), RLON(25), RINER(25), RMP1(25), AXIL(25), XA(25),
5262 @          YA(25), ZA(25), DCL(20), R(75), LBE(20), T(50), LICOL(50),
5263 @          NCOL(50), NFILA(50), ICOL(50), DCBU(100), T1(50), Q(50),
5264 @          NGB(25), NTP(25), LTP(25), AREA(25), TT(50), CFI(50),
5265 @          CMH(50), CMV(50), FIFI(25), X(25), Y(25), Z(25),
5266 @          RLCPV(10,2), TETA(50,50), RMAT(50,50), TCNP(50,10)
5267      COMMON /PAS/ NMC, NC, NETAPA, IND, COLAA, COLAP, RPOS(50), RMEN(50),
5268 @          MC(50), RITE(25), NECNP, NENP, COTA(10)
5269      LB1=50
5270      M2=10
5271 C
5272 C      LECTURA DE FICHERO
5273 C
5274      CALL LEEDI(RMAT, LB1, LB1, NVIR, NH, 50, 1)
5275      IF(NCL.EQ.22) CALL LEEDI(TCNP, LB1, M2, NH, NECNP, 53, 1)
5276      IF(NITER.NE.2) GO TO 5
5277      NCL=13
5278 C
5279 C      PUESTA A CERO
5280 C
5281 C      CALL ZERO(TETA, LB1, LB1, LB1, LB1)
5282 C
5283 C      FORMACION DE LA MATRIZ  $Y = -(BH^A * A * BH)^{**}(-1)$ 
5284 C
5285      CALL FORY(TETA, RMAT, RMOME, RLON, RINER, ICOL, NB, LB1, NVIR, 1., NH,
5286 @          TOL, 57)
5287 C
5288 C      FORMACION DE LA MATRIZ  $Z = A + I$ 
5289 C
5290      CALL FORZ(RMAT, TETA, RLON, RINER, 1., LB1, NH, NVIR, 57, NB)
5291 C
5292 C      FORMACION DEL VECTOR DE CARGAS DE CALCULO
5293 C
5294      CALL FORBF(TETA, RMAT, DCBU, IT, NJ, RLON, ALFA, RINER, Q, TCNP, NECNP, NCL,
5295 @          NH, NBCU, NBB, ND, NB, LB1, M2, NENP, COM, COL, E, COI, AXIL)
5296      CALL DEMAX(RPOS, RMEN, TCNP, RLCPV, NTCPV, NH, LB1, M2)
5297      DO 6 I=1, NH
5298      RPOS(I)=RPOS(I)*COM
5299      RMEN(I)=RMEN(I)*COM
5300 C      CONTINUE
5301      IF(NCL.EQ.22) GO TO 2
5302      DO 1 I=1, NH
5303      RIMON(I)=RPOS(I)/COM
5304      RMOME(I)=RMEN(I)/COM
5305 C      CONTINUE

```

```
5306      GO TO 15
5307 2    DO 3 I=1,NB
5308      RITE(I)=RINER(I)*COI
5309 3    CONTINUE
5310 C
5311 C    LLAMADA AL SEGMENTO CORRESPONDIENTE
5312 C
5313      CALL EXEC(8,5HDISEN)
5314 15   CALL EXEC(8,5HSHD02)
5315      END
```

FTN4X COMPILER: HF92834 REV.2130 (810716)

\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: 235 COMMON: (NONE)

```
5316      SUBROUTINE DEMAX(RPOS, RMEN, TCNP, RLCPV, NTCPV, NH, N1, N2)
5317 C
5318 C      ESTA SUBROUTINA CALCULA LA ENVOLVENTE DE LOS MAXIMOS
5319 C
5320      DIMENSION RPOS(NH), RMEN(NH), TCNP(N1, N2), RLCPV(N2, 2)
5321      COMMON /VAR/ IN, IO
5322      DO 1 JJ=1, NH
5323      RPOS(JJ)=0.0
5324      RMEN(JJ)=0.0
5325 1      CONTINUE
5326      NTC=NTCPV+1
5327      DO 2 I=1, NH
5328      IF (NTCPV.EQ.0) GO TO 3
5329      DO 4 J=2, NTCPV+1
5330      K=1
5331      IF (TCNP(I, J).GE.0) K=2
5332      RMEN(I)=RMEN(I)+TCNP(I, J)*RLCPV(J-1, K)
5333      K=2
5334      IF (TCNP(I, J).GE.0) K=1
5335      RPOS(I)=RPOS(I)+TCNP(I, J)*RLCPV(J-1, K)
5336 4      CONTINUE
5337 3      RMEN(I)=RMEN(I)+TCNP(I, 1)
5338      RPOS(I)=RPOS(I)+TCNP(I, 1)
5339 2      CONTINUE
5340      RETURN
5341      END
```

FTN4X COMPILER: HP92834 REV.2130 (810716)

\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: 239 COMMON: (NONE)

```

5342 PROGRAM SHD02(5)
5343 C
5344 C SEGMENTO PARA LA DETERMINACION DE LA CARGA DE SHAKE-DOW
5345 C
5346 COMMON /VAR/ IN, IO, NOMB(30), ND, NB, INDT, NMI, NLI, NVIR, NAI, NBCU, NMH,
5347 @ NFI, NGD, NGP, NCL, LHF, LHV, NITER, NND, NH, NBB, NTCPV, NCFP,
5348 @ NCPV, NMIN, NGH, NMCO, NMHCO, NECD, NTRA, LV, CADI, SUMF, E,
5349 @ SUM, TOL, TOLS, SUM1, PESO, SIGMA, COP, COL, COI, COM, TCO
5350 COMMON /CON/ IT(25), RMP(25), RPOS(50), RMEN(50), NI(25), NJ(25),
5351 @ TETA(47, 147)
5352 COMMON /PAS/ R(81), NBAS(100)
5353 C
5354 C PUESTA A CERO
5355 C
5356 LB1=47
5357 LB2=147
5358 DO 1 I=1, LB1
5359 R(I)=0.0
5360 NBAS(I)=0
5361 1 CONTINUE
5362 CALL ZERO(TETA, LB1, LB2, LB1, LB2)
5363 SUM=0.0
5364 CALL LEEDI(TETA, LB1, LB2, NVIR, NH, 50, 1)
5365 CALL CSHDO(TETA, RPOS, RMEN, R, RMP, LB1, NB, NVIR, NH, TOL, SUM, LB2, NBAS)
5366 C
5367 C OBTENCION DE LA SOLUCION DEL PROBLEMA PRIMAL
5368 C
5369 DO 2 I=1+2*NH, 2*NH+NVIR
5370 RMP(I-2*NH)=TETA(2*NVIR+2, I)-TETA(2*NVIR+2, I+NVIR)
5371 2 CONTINUE
5372 SUM=TETA(2*NVIR+2, 2*NH+2*NVIR+1)
5373 CALL ZERO(TETA, LB1, LB2, LB1, LB2)
5374 CALL LEEDI(TETA, LB1, LB2, NVIR, NH, 50, 1)
5375 DO 3 I=1, NH
5376 D=0
5377 DO 4 J=1, NVIR
5378 D=D+TETA(J, I)*RMP(J)
5379 4 CONTINUE
5380 R(I)=D
5381 3 CONTINUE
5382 CALL SECON(T)
5383 C
5384 C ESCRITURA DE RESULTADOS
5385 C
5386 TCO=T-TCO
5387 CALL ESSHD(SUM, RPOS, RMEN, NH, NB, NI, NJ, TCO, COP, COL, COM, RMP, R, LB1,
5388 @ NITER)
5389 STOP
5390 END

```



\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: 246 COMMON: (NONE)

```
5391 SUBROUTINE CSHDO(TETA,RPOS,RMEN,R,RMP,LB1,NB,NVIR,NH,TOL,OPTM,  
5392 @ LB2,NBAS)  
5393 C  
5394 C ESTA RUTINA DETERMINA LA CARGA DE SHAKE-DOW  
5395 C  
5396 C PREPARACION DE LAS ECUACIONES  
5397 C  
5398 C DIMENSION TETA(LB1,LB2),RPOS(NH),RMEN(NH),RMP(NB),R(LB1),NBAS(LB1)  
5399 C  
5400 C FORMACION DE LAS RESTRICCIONES DEL PROBLEMA DUAL  
5401 C  
5402 C DO 1 I=1,NVIR  
5403 C DO 1 J=1,NH  
5404 C TETA(I,J+NH)=TETA(I,J)  
5405 C TETA(I+NVIR,J)=TETA(I,J)  
5406 C TETA(I+NVIR,J+NH)=-TETA(I,J)  
5407 C TETA(I,J)=-TETA(I,J)  
5408 1 CONTINUE  
5409 C DO 2 I=1,NH  
5410 C TETA(2*NVIR+1,I)=-RPOS(I)  
5411 C TETA(2*NVIR+1,I+NH)=RMEN(I)  
5412 2 CONTINUE  
5413 C  
5414 C FORMACION DE LA PARTE CORRESPONDIENTE A VARIABLES DE HOLGURAS  
5415 C  
5416 C DO 3 I=1,2*NVIR+1  
5417 C TETA(I,2*NH+I)=1  
5418 C R(I)=0.0  
5419 3 CONTINUE  
5420 C  
5421 C FORMACION DEL VECTOR DE TERMINOS INDEPENDIENTES  
5422 C  
5423 C R(2*NVIR+1)=-1  
5424 C  
5425 C FORMACION DE LA FUNCION OBJETIVO  
5426 C  
5427 C DO 4 I=1,NB  
5428 C TETA(2*NVIR+2,2*I-1)=RMP(I)  
5429 C TETA(2*NVIR+2,2*I)=RMP(I)  
5430 C TETA(2*NVIR+2,NH+2*I-1)=RMP(I)  
5431 C TETA(2*NVIR+2,NH+2*I)=RMP(I)  
5432 4 CONTINUE  
5433 C M=2*NVIR+1  
5434 C N=2*NH+2*NVIR+1  
5435 C AC=TOL  
5436 C  
5437 C LLAMADA AL ALGORITMO DE RESOLUCION  
5438 C  
5439 C IF(ISSW(5).GE.0) GO TO 122  
5440 C CALL ESCRM(TETA,R,LB1,LB2,LB1,M,N,1)  
5441 122 CALL SIMP(TETA,R,M,N,N,LB1,LB2,TOL,LB1,NBAS,N,M)  
5442 C OPTM=-TOL  
5443 C TOL=AC  
5444 C RETURN  
5445 C END
```

FTN4X COMPILER: HP92834 REV.2130 (810716)

\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: 418 COMMON: (NONE)

```

5446      SUBROUTINE ESSHD(SUM,RPOS,RMEN,NH,NB,NI,NJ,TCO,COP,COL,COM,RMP,R,
5447 @          LB1,NITER)
5448 C
5449 C      RUTINA PARA LA ESCRITURA DE RESULTADOS DEL SHAKE-DOW
5450 C
5451      DIMENSION RPOS(NH),RMEN(NH),NI(NB),NJ(NB),RMP(NB),R(LB1)
5452      COMMON /VAR/ IN,IO,NOMB(30)
5453      DO 1 I=1,NH
5454      RPOS(I)=RPOS(I)*COM
5455      RMEN(I)=RMEN(I)*COM
5456      R(I)=R(I)*COM
5457 1      CONTINUE
5458      DO 2 I=1,NB
5459      RMP(I)=RMP(I)*COM
5460 2      CONTINUE
5461      IF(NITER.EQ.2) GO TO 21
5462      WRITE(IO,10) NOMB
5463 21     WRITE(IO,11) SUM
5464      WRITE(IO,12)
5465      DO 15 I=1,NB
5466      WRITE(IO,16) I,NI(I),RPOS(2*I-1),RMEN(2*I-1),R(2*I-1),NJ(I),
5467 @          RPOS(2*I),RMEN(2*I),R(2*I)
5468 15     CONTINUE
5469      WRITE(IO,17) TCO
5470      RETURN
5471 C
5472 C      FORMATOS DE ESCRITURA
5473 C
5474 10     FORMAT(1H1,10X,30A2/10X,60("*")3/)
5475 11     FORMAT(10X,"CARGA DE SHAKE-DOWN.....",E14.5,3/)
5476 12     FORMAT(10X,"ENVOLVENTE DE MOMENTOS ELASTICOS MAX Y MIN. Y MOM. RES
5477 @IDUALES"/10X,61("-")3/,10X,"BARRA      NUDO      MOM. MAX +      MOM
5478 @. MIN -      MOM. RESD."2/)
5479 16     FORMAT(10X,I4,4X,I3,5X,E12.4,4X,E12.4,5X,E10.4/18X,I3,5X,E12.4,4X,
5480 @          E12.4,5X,E10.4)
5481 17     FORMAT(6/,10X,"*** TIEMPO DE C.P.U EMPLEADO.....",F10.3,"SG.")
5482      END

```

FTN4X COMPILER: HP92834 REV.2130 (810716)

\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: 450 COMMON: (NONE)

```

5483      PROGRAM DISEN(5)
5484 C
5485 C      SEGMENTO PARA LA FORMACION DE LA MATRIZ FINAL DEL PROBLEMA DE
5486 C      DISEÑO Y RESOLUCION,
5487 C
5488      DIMENSION NAMI(3),TERI(65),CD(25),NBAS(100)
5489      COMMON /VAR/ IN,IO,NOMB(30),ND,NB,INDT,NMI,NLI,NGL,NAI,NBCU,NMH,
5490 @          NFI,NGD,NGF,NCL,LHF,LHV,NITER,NND,NH,NBB,NTCPV,NCFP,
5491 @          NCPV,NMIN,NGH,NMCO,NMHCO,NECO,NTRA,LV,CADI,SUMF,E,
5492 @          SUM,TOL,TOLS,SUM1,FESO,SIGMA,COP,COL,COI,COM,TCO
5493      COMMON /CON/ IT(25),RMP(25),RIMON(50),RMOME(50),NI(25),NJ(25),
5494 @          ALFA(25),RLON(25),RINER(25),RMP1(25),AXIL(25),XA(25),
5495 @          YA(25),ZA(25),DCL(20),R(75),LBE(20),T(50),LICOL(50),
5496 @          NCOL(50),NFILA(50),ICOL(50),DCBU(100),T1(50),Q(50),
5497 @          NGB(25),NTP(25),LTP(25),AREA(25),TT(50),CFI(50),
5498 @          CMH(50),CMV(50),FIFI(25),X(25),Y(25),Z(25),
5499 @          RLCPV(10,2),TETA(62,101)
5500      COMMON /PAS/ NMC,NC,NETAPA,IND,COLAA,FINAL,RPOS(50),RMEN(50),
5501 @          MC(50),RITE(25),NECNF,NENF,COTA(10)
5502      LB1=62
5503      LB2=101
5504 C
5505 C      PUESTA A CERO
5506 C
5507      FINAL=0
5508      CALL ZERO(TETA,LB1,LB2,LB1,LB2)
5509      DO 510 I=1,NND
5510      R(I)=0.0
5511 510      CONTINUE
5512      IF(NCL.EQ.23) GO TO 10
5513      DO 11 I=1,NB
5514      T(I)=1.0
5515 11      CONTINUE
5516 C
5517 C      LLAMADA A LA RUTINA DE FORMACION DE LA MATRIZ Y RESOLUCION
5518 C
5519 10      OPTM=0.0
5520 C
5521 C      OBTENCION DE LA MATRIZ TETA
5522 C
5523      CALL LEEDI(TETA,LB1,LB2,NMI,NH,55,1)
5524      IF(NCL.EQ.20.OR.NCL.EQ.23) GO TO 1
5525      DO 2 I=1,NMI
5526      D=0
5527      DO 3 J=1,NH
5528      D=D+RMEN(J)*TETA(I,J)
5529 3      CONTINUE
5530      TERI(I)=D/COM
5531 2      CONTINUE
5532      DO 4 I=1,NH
5533      TERI(I+NMI)=- (RPOS(I)-RMEN(I))/COM
5534 4      CONTINUE
5535      GO TO 5
5536 1      DO 6 I=1,NMI
5537      TERI(I)=T1(I)*COP*COL

```

```

5538 6      CONTINUE
5539      COJ=COM
5540      COM=1.
5541      IF(NCL.NE.23) GO TO 5
5542      DO 40 I=1,NB
5543      RMP(I)=RMP1(I)*COJ
5544 40     CONTINUE
5545 5      CALL FORDI(TETA,TERI,RLON,RMOME,RMP1,NB,NMI,NGB,NGD,TOL,OPTM,
5546 @          LB1,LB2,R,NND,NH,COP,COL,RMEN,NCL,COM,NITER,T,SIGMA,
5547 @          NBAS)
5548 C
5549 C      SI HAY CARGAS UNIFORMES CALCULAR LOS MOMENTOS E ITERAR SI PROCEDE
5550 C
5551      IF(NBCU.EQ.0.OR.NCL.EQ.23) GO TO 7
5552      IF(ABS(OPTM-PESO).LT.TOLS) GO TO 7
5553      PESO=OPTM
5554      SUM=1.0
5555      CALL CAMBI(RMOME,NB,NI,NJ,IT,ND,NH,NBCU,DCBU,RLON,ALFA,TOL,RMP1,
5556 @          LF,SUM,TOLS,NBB)
5557      IF(LF.EQ.0) GO TO 7
5558 C
5559 C      LLAMADA AL SEGMENTO CORRESPONDIENTE
5560 C
5561      CALL EXEC(8,5HITERA)
5562 7      NAMI(1)=2HDI
5563      NAMI(2)=2HSE
5564      NAMI(3)=2HP
5565      CALL ZERO(TETA,LB1,LB2,LB1,LB2)
5566      SUM=1.0
5567      IF(NCL.NE.22) GO TO 25
5568      DO 45 I=1,NB
5569      CO(I)=1
5570      X(I)=RINER(I)*COI
5571 45     CONTINUE
5572      COT=COM
5573      CALL ELECO(RMP,RMP1,RINER,AREA,NB,SIGMA,NTP,COM,COI,CO)
5574      IF(ISSW(10).GE.0) GO TO 26
5575      WRITE(IO,451)
5576      DO 29 I=1,NB
5577      AA=RMP(I)*COM
5578      BB=RINER(I)*COI
5579      WRITE(IO,27) I,AA,BB
5580 29     CONTINUE
5581      WRITE(IO,46)
5582 26     DO 30 I=1,NB
5583      SS=RINER(I)*COI
5584      IF(SS.LE.1) SS=1
5585      ERROR=100*ABS((RINER(I)*COI-X(I)))/SS
5586      IF(ERROR.LE.1.0) GO TO 30
5587      GO TO 31
5588 30     CONTINUE
5589      DO 33 I=1,NB
5590      RMP(I)=RMP(I)*COM
5591      RMP1(I)=RMP1(I)*COM
5592 33     CONTINUE

```

```
5593      CALL EXEC(8,5HDISEP)
5594 31    DO 32 I=1,NTCPV
5595      DO 32 J=1,2
5596      RLCPV(I,J)=RLCPV(I,J)*COT/COM
5597 32    CONTINUE
5598      CALL EXEC(8,5HSHD01)
5599 25    IF(NCL.NE.23) CALL EXEC(8,NAMI)
5600 C
5601 C      PUESTA A CERO
5602 C
5603 8      CALL ZERO(TETA,LB1,LB2,LB1,LB2)
5604      NCOP=0
5605      TOPES=0
5606 C
5607 C      SI EL ERROR ES MENOR QUE 1.0% TERMINA LA ETAPA DE DISE#0
5608 C
5609      IF(ISSW(2).LT.0) WRITE(IO,8811)
5610      DO 41 I=1,NB
5611      SS=RMP(I)
5612      IF(SS.LE.1) SS=1
5613      ERROR=100*ABS((RMP(I)-RMP1(I)))/SS
5614      IF(ISSW(2).LT.0) WRITE(IO,876) I,ERROR
5615      IF(ERROR.LE.1.0) GO TO 41
5616      NCOP=1
5617      TOPES=1
5618 41    CONTINUE
5619      IF(TOPES.EQ.0) NCOP=0
5620      FINAL=30.
5621      IF(NCOP.EQ.1) FINAL=33
5622      CALL EXEC(8,NAMI)
5623 C
5624 C      FORMATOS DE ESCRITURA
5625 C
5626 451    FORMAT(/10X,50("=")/10X,"BARRA      M. PLASTICO      INERCIA")
5627 27    FORMAT(10X,I3,3X,E12.4,3X,E12.4)
5628 46    FORMAT(10X,33("-"))
5629 8811  FORMAT(10X,72("="))
5630 876   FORMAT(10X,"SECCION :",I5,"  ERROR :",E14.4," %")
5631      END
```

FTN4X COMPILER: HP92834 REV.2130 (810716)

\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: 1133 COMMON: (NONE)

```
5632 SUBROUTINE FORDI(TETA,TERI,RLON,RMOME,RMP1,NB,NMI,NGB,NGD,TOL,
5633 @ OPTM,LB1,LB2,R,NND,NH,COP,COL,RMEN,NCL,COM,NITER,
5634 @ T,SIGMA,NBAS)
5635 C
5636 C RUTINA PARA FORMAR EL SISTEMA Y RESOLUCION MEDIANTE P.L
5637 C
5638 DIMENSION TETA(LB1,LB2),TERI(LB1),NGB(NB),RLON(NB),R(NND),
5639 @ RMP1(NB),RMOME(NH),RMEN(NH),T(NB),NBAS(LB1)
5640 NFIL=NMI+NH
5641 IF(NCL.EQ.23) NFIL=NFIL+NGD
5642 NF1=NFIL+1
5643 NCOL=2*NH+NGD
5644 IF(NCL.EQ.23) NCOL=NCOL+NGD
5645 NN=NCOL-NH
5646 IF(NCL.NE.22.OR.NITER.EQ.0) GO TO 20
5647 CALL LEEDI(TETA,LB1,LB2,NF1,NCOL,61,1)
5648 GO TO 21
5649 20 DO 10 I=1,NMI
5650 DO 11 J=1,NB
5651 R(NGB(J))=R(NGB(J))-(TETA(I,2*J-1)+TETA(I,2*J))*T(J)
5652 11 CONTINUE
5653 DO 12 J=1,NGD
5654 TETA(I,NH+J)=R(J)
5655 R(J)=0
5656 12 CONTINUE
5657 10 CONTINUE
5658 DO 13 I=NMI+1,NMI+NH
5659 TETA(I,I-NMI)=1
5660 13 CONTINUE
5661 L=0
5662 DO 14 J=1,NH
5663 L=J-L
5664 TETA(NMI+J,NH+NGB(L))=-2*T(L)
5665 14 CONTINUE
5666 DO 15 I=NMI+1,NMI+NH
5667 TETA(I,NH+NGD+I-NMI)=1
5668 15 CONTINUE
5669 DO 16 I=1,NB
5670 R(NGB(I))=R(NGB(I))+RLON(I)
5671 16 CONTINUE
5672 DO 17 J=NH+1,NH+NGD
5673 TETA(NFIL+1,J)=R(J-NH)
5674 17 CONTINUE
5675 IF(NCL.NE.23) GO TO 30
5676 J=0
5677 DO 31 I=NMI+NH+1,NFIL
5678 J=J+1
5679 TETA(I,NH+J)=1
5680 TETA(I,2*NH+NGD+J)=-1
5681 TERI(I)=11.4*SIGMA*2.
5682 31 CONTINUE
5683 30 IF(NCL.EQ.22) CALL LEEDI(TETA,LB1,LB2,NF1,NCOL,61,2)
5684 21 TOL1=TOL
5685 IF(ISSW(6).GE.0) GO TO 122
5686 CALL ESCRM(TETA,TERI,LB1,LB2,LB1,NFIL,NCOL,1)
```



```
5687 C
5688 C      LLAMADA A LA RUTINA DE RESOLUCION
5689 C
5690 122    CALL SIMP(TETA,TERI,NFIL,NCOL,NN,LB1,LB1,TOL,LB2,NBAS,NCOL,NFIL)
5691      OPTM=TOL
5692      TOL=TOL1
5693      L=0
5694      DO 18 I=1,NH
5695      L=I-L
5696      RMP1(L)=TERI(NGB(L)+NH)*COM
5697 18     CONTINUE
5698      L=0
5699      DO 19 I=1,NH
5700      L=I-L
5701      RMOME(I)=TERI(I)*COM-RMP1(L)*T(L)
5702      IF(NCL.EQ.22) RMOME(I)=RMOME(I)-RMEN(I)
5703 19     CONTINUE
5704      RETURN
5705      END
```

FTN4X COMPILER: HP92834 REV.2130 (810716)

\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: 646 COMMON: (NONE)

```

5706      PROGRAM DISEP(5)
5707 C
5708 C      SEGMENTO PARA LA ELECCION DEL PERFIL MAS ADECUADO
5709 C
5710      DIMENSION LL(2),RA(50),CO(25)
5711      COMMON /VAR/ IN,IO,NOMB(30),ND,NB,INDT,NMI,NLI,NGL,NAI,NBCU,NMH,
5712 @          NFI,NGD,NGP,NCL,LHF,LHV,NITER,NND,NH,NBB,NTCPV,NCFF,
5713 @          NCPV,NMIN,NGH,NMCO,NMHCO,NECO,NTRA,LV,CADI,SUMF,E,
5714 @          SUM,TOL,TOLS,SUM1,PESO,SIGMA,COP,COL,COI,COM,TCO
5715      COMMON /CON/ IT(25),RMP(25),RIMON(50),RMOE(50),NI(25),NJ(25),
5716 @          ALFA(25),RLON(25),RINER(25),RMP1(25),AXIL(25),XA(25),
5717 @          YA(25),ZA(25),DCL(20),R(75),LBE(20),T(50),LICOL(50),
5718 @          NCOL(50),NFILA(50),ICOL(50),DCBU(100),T1(50),Q(50),
5719 @          NGB(25),NTP(25),LTP(25),AREA(25),T2(50),CFI(50),
5720 @          CMH(50),CMV(50),FIFI(25),X(25),Y(25),Z(25),
5721 @          RLCPV(10,2),TETA(50,50)
5722      COMMON /PAS/ NMC,NC,NETAPA,IND,COLAA,FINAL,RPOS(50),RMEN(50),
5723 @          MC(50),RITE(25),NECNP,NENP,COTA(10)
5724      LB1=50
5725      DO 50 I=1,NB
5726      CO(I)=T(I)
5727 50    CONTINUE
5728      DO 67 I=1,NMI
5729      T(I)=T1(I)
5730 67    CONTINUE
5731 C
5732 C      LLAMADA A LA RUTINA DE ELECCION
5733 C
5734      IF(NCL.NE.23) GO TO 45
5735      IF(FINAL.NE.33) GO TO 45
5736      CALL ELECO(RMP,RMP1,RINER,AREA,NB,SIGMA,NTP,COM,COI,CO)
5737      IF(ISSW(2).GE.0) GO TO 756
5738      DO 757 II=1,NB
5739      WRITE(IO,1287) II,RMP(II)*COM,RINER(II)*COI,AREA(II)
5740 757    CONTINUE
5741      WRITE(IO,881)
5742 756    GO TO 35
5743 45      PES=PESO
5744      PESO=0.0
5745      CALL ELEC(RMP,RMP1,RINER,PESO,LTP,NB,SIGMA,NTP,RLON,AREA)
5746      NITER=1
5747      PESO=PESO*COL
5748      COT=COM
5749      COM=1.E36
5750      COI=1.E36
5751      NCO=0
5752      DO 1 I=1,NB
5753      IF(COM.GT.RMP(I))COM=RMP(I)
5754      IF(COI.GT.RINER(I))COI=RINER(I)
5755 1      CONTINUE
5756      DO 2 I=1,NB
5757      RMP(I)=RMP(I)/COM
5758      IF(NCL.NE.22) GO TO 32
5759      IF(ABS(RITE(I)-RINER(I)).GT.TOLS) NCO=1
5760 32      RINER(I)=RINER(I)/COI

```

```
5761 2   CONTINUE
5762     IF(NCL.NE.22) GO TO 31
5763     DO 10 I=1,NTCPV
5764     DO 10 J=1,2
5765     RLCPV(I,J)=RLCPV(I,J)*COT/COM
5766 10   CONTINUE
5767 C
5768 C   SACO LA MATRIZ DE EQUILIBRIO DEL DISCO
5769 C
5770 C*****
5771 31   IF(NCL.EQ.23) GO TO 3
5772 C*****
5773     IF(NCL.EQ.20) GO TO 35
5774     IF(NCO.EQ.0) GO TO 3
5775     IF(NCL.EQ.22) GO TO 3
5776 35   CALL LEEDI(TETA,LB1,LB1,NMI,NH,55,1)
5777 C
5778 C   LLAMADA AL SEGMENTO CORRESPONDIENTE
5779 C
5780     CALL EXEC(8,5HCOLAP)
5781 C
5782 C   ESCRITURA DEL DISEÑO
5783 C
5784 3     NITER=2
5785 30   WRITE(IO,12) NOMB
5786     WRITE(IO,13) PESO
5787     WRITE(IO,14)
5788     DO 15 I=1,NB
5789     N=LTP(I)/10
5790     L=N*10
5791     J=LTP(I)-L
5792     GO TO (16,17,18),J
5793 16   LL(1)=2HIP
5794     LL(2)=2HN-
5795     GO TO 19
5796 17   LL(1)=2HIP
5797     LL(2)=2HE-
5798     GO TO 19
5799 18   LL(1)=2HHE
5800     LL(2)=2HB-
5801 19   AA=RMP(I)*COM
5802     WRITE(IO,20) I,RMP1(I),AA,LL,L
5803     RMP1(I)=RMP(I)
5804 15   CONTINUE
5805     WRITE(IO,21)
5806     IF(NCL.EQ.22) GO TO 11
5807     NCL=15
5808     SUMF=0.0
5809     DO 88 I=1,NMI
5810     T1(I)=T2(I)
5811     T(I)=T2(I)
5812 88   CONTINUE
5813     DO 89 I=1,NB
5814     AXIL(I)=0.0
5815     Z(I)=0.0
```

```
5816 89    CONTINUE
5817        GO TO 35
5818 11    CALL EXEC(8,5HSHD01)
5819 C    
5820 C     FORMATOS DE ESCRITURA
5821 C    
5822 1287    FORMAT(10X,"BARRA,Mp,Inercia,Area",I5,3E14.5)
5823 881     FORMAT(10X,72("="))
5824 12     FORMAT(1H1,10X,30A2/10X,60("*")3/)
5825 13     FORMAT(10X,"PESO DE LA ESTRUCTURA :",5X,E14.5," KG.")
5826 14     FORMAT(3/,10X,"SOLUCION TEORICA Y PRACTICA DE DISEÑO "/10X,
5827        @37("-")/10X,"BARRA    M.PLAST E        M.PLAST P        TIPO-PERFIL"/)
5828 20     FORMAT(10X,I4,3X,E10.4,3X,E10.4,5X,2A2,I3)
5829 21     FORMAT(2/)
5830        END
```

FTN4X COMPILER: HP92834 REV.2130 (810716)

\*\* NO WARNINGS \*\*    NO ERRORS \*\*    PROGRAM:    904    COMMON: (NONE)

```

5831 SUBROUTINE ELEC(RMP,RMP1,RINER,PESO,LTIP,NB,SIGMA,NTP,RLON,AREA)
5832 C
5833 C SUBROUTINA PARA LA ELECCION DEL PERFIL
5834 C
5835 DIMENSION TIP1(21,5),TIP2(18,5),TIP3(19,5),RMP(NB),RMP1(NB),
5836 @ RINER(NB),LTIP(NB),RLON(NB),NTP(NB),AREA(NB)
5837 C
5838 C ALMACENAMIENTO DEL CONJUNTO DE PERFILES
5839 C
5840 DATA TIP2/8.,10.,12.,14.,16.,18.,20.,22.,24.,27.,30.,33.,36.,40.,
5841 @45.,50.,55.,60.,11.6,19.7,30.4,44.2,61.9,83.2,110.,143.,183.,242.,
5842 @314.,402.,510.,654.,851.,1100.,1390.,1760.,80.1,171.,318.,541.,
5843 @869.,1320.,1940.,2770.,3890.,5790.,8360.,11770.,16270.,23130.,
5844 @33740.,48200.,67120.,92080.,6.,8.1,10.4,12.9,
5845 @15.8,18.8,22.4,26.2,30.7,36.1,42.2,49.1,57.1,66.3,77.6,90.7,106.,
5846 @122.,7.64,10.3,13.2,16.4,20.1,23.9,28.5,33.4,39.1,45.9,53.8,62.6,
5847 @72.7,84.5,98.8,116.,134.,156./
5848 DATA TIP1/8.,10.,12.,14.,16.,18.,20.,22.,24.,26.,28.,30.,
5849 @32.,34.,36.,38.,40.,45.,
5850 @50.,55.,60.,11.4,19.9,31.8,47.7,68.0,93.4,125.,162.,206.,257.,
5851 @316.,381.,457.,540.,638.,741.,857.,1200.,1620.,2120.,2730.,77.8,
5852 @171.,328.,573.,935.,1450.,2140.,3060.,4250.,5740.,7590.,9800.,
5853 @12510.,15700.,19610.,24010.,29210.,45850.,68740.,99180.,139000.,
5854 @5.95,8.32,11.1,14.4,17.9,21.9,26.3,31.1,36.2,41.9,48.,54.2,61.1,
5855 @68.1,76.2,84.,92.6,115.,141.,167.,199.0,7.57,10.6,14.2,18.2,22.8,
5856 @27.9,33.4,39.5,46.1,53.3,61.,69.,77.7,86.7,97.,107.,118.,147.,
5857 @179.,212.,254./
5858 DATA TIP3/10.,12.,14.,16.,18.,20.,22.,24.,26.,28.,30.,32.,34.,36.,
5859 @40.,45.,50.,55.,60.,52.1,82.6,123.,177.,241.,321.,414.,527.0,
5860 @641.,797.,934.,1070.,1200.,1340.,1620.,1990.,2410.,2800.,3210.,
5861 @450.,864.,1510.,2490.,3830.,5700.,8090.,11260.,14920.,19270.,
5862 @25170.,30820.,36660.,43190.,57680.,79890.,107200.,136700.,171000.,
5863 @20.4,26.7,33.7,42.6,51.2,61.3,71.5,83.2,93.0,103.,117.,127.,
5864 @134.,142.,155.,171.,187.,199.,212.,26.,34.,43.,54.3,65.3,78.1,
5865 @91.,106.,118.4,131.4,149.1,161.3,170.9,180.6,197.8,218.,238.6,
5866 @254.1,270./
5867 C
5868 C ELECCION DEL PERFIL. SI EL USUARIO HA DEJADO LIBRE EL TIPO DE PERFIL
5869 C EL PROGRAMA ELIGIRA EL DE MINIMO PESO.
5870 C
5871 DO 1 I=1,18
5872 TIP1(I,2)=TIP1(I,2)*SIGMA*2.
5873 TIP2(I,2)=TIP2(I,2)*SIGMA*2.
5874 TIP3(I,2)=TIP3(I,2)*SIGMA*2.
5875 1 CONTINUE
5876 TIP1(19,2)=TIP1(19,2)*SIGMA*2.
5877 TIP1(20,2)=TIP1(20,2)*SIGMA*2.
5878 TIP1(21,2)=TIP1(21,2)*SIGMA*2.
5879 TIP3(19,2)=TIP3(19,2)*SIGMA*2.
5880 PESO=0
5881 DO 2 I=1,NB
5882 LK=NTP(I)+1
5883 GOTO(4,5,6,7),LK
5884 4 RAYO1=1.E36
5885 RAYO2=1.E36

```

```
5886      RAY03=1.E36
5887      DO 11 J=1,21
5888      RAY1=TIP1(J,2)-RMP1(I)
5889      IF(J.GT.18) GO TO 55
5890      RAY2=TIP2(J,2)-RMP1(I)
5891 55     IF(J.GT.19) GO TO 56
5892      RAY3=TIP3(J,2)-RMP1(I)
5893 56     IF(RAY1.GT.RAY01.OR.RAY1.LT.0) GO TO 12
5894      RAY01=RAY1
5895      MAY01=J
5896 12     IF(J.GT.18) GO TO 13
5897      IF(RAY2.GT.RAY02.OR.RAY2.LT.0) GO TO 13
5898      RAY02=RAY2
5899      MAY02=J
5900 13     IF(J.GT.19) GO TO 11
5901      IF(RAY3.GT.RAY03.OR.RAY3.LT.0) GO TO 11
5902      RAY03=RAY3
5903      MAY03=J
5904 11     CONTINUE
5905      PES=TIP1(MAY01,4)
5906      IF(PES.GT.TIP2(MAY02,4)) PES=TIP2(MAY02,4)
5907      IF(PES.GT.TIP3(MAY03,4)) PES=TIP3(MAY02,4)
5908      PES0=PES0+PES*RLON(I)/100.
5909      IF(PES.EQ.TIP1(MAY01,4)) GO TO 14
5910      IF(PES.EQ.TIP2(MAY02,4)) GO TO 15
5911      RINER(I)=TIP3(MAY03,3)
5912      RMP(I)=TIP3(MAY03,2)
5913      AREA(I)=TIP3(MAY03,5)
5914      LTIP(I)=TIP3(MAY03,1)*10+3
5915      GO TO 2
5916 14     RINER(I)=TIP1(MAY01,3)
5917      RMP(I)=TIP1(MAY01,2)
5918      AREA(I)=TIP1(MAY01,5)
5919      LTIP(I)=TIP1(MAY01,1)*10+1
5920      GO TO 2
5921 15     RINER(I)=TIP2(MAY02,3)
5922      RMP(I)=TIP2(MAY02,2)
5923      AREA(I)=TIP2(MAY02,5)
5924      LTIP(I)=TIP2(MAY02,1)*10+2
5925      GO TO 2
5926 5      RAYOR=1.E36
5927      DO 20 J=1,21
5928      RAY=TIP1(J,2)-RMP1(I)
5929      IF(RAY.GT.RAYOR.OR.RAY.LT.0) GO TO 20
5930      RAYOR=RAY
5931      MAYOR=J
5932 20     CONTINUE
5933      RINER(I)=TIP1(MAYOR,3)
5934      RMP(I)=TIP1(MAYOR,2)
5935      AREA(I)=TIP1(MAYOR,5)
5936      LTIP(I)=TIP1(MAYOR,1)*10+1
5937      PES0=PES0+TIP1(MAYOR,4)*RLON(I)/100
5938      GO TO 2
5939 6      RAYOR=1.E36
5940      DO 21 J=1,18
```

```
5941      RAY=TIP2(J,2)-RMP1(I)
5942      IF(RAY.GT.RAYOR.OR.RAY.LT.0) GO TO 21
5943      RAYOR=RAY
5944      MAYOR=J
5945 21    CONTINUE
5946      RINER(I)=TIP2(MAYOR,3)
5947      AREA(I)=TIP2(MAYOR,5)
5948      RMP(I)=TIP2(MAYOR,2)
5949      LTIP(I)=TIP2(MAYOR,1)*10+2
5950      PESO=PESO+TIP2(MAYOR,4)*RLON(I)/100
5951      GO TO 2
5952 7     RAYOR=1.E36
5953      DO 22 J=1,19
5954      RAY=TIP3(J,2)-RMP1(I)
5955      IF(RAY.GT.RAYOR.OR.RAY.LT.0) GO TO 22
5956      RAYOR=RAY
5957      MAYOR=J
5958 22    CONTINUE
5959      RINER(I)=TIP3(MAYOR,3)
5960      AREA(I)=TIP3(MAYOR,5)
5961      RMP(I)=TIP3(MAYOR,2)
5962      LTIP(I)=TIP3(MAYOR,1)*10+3
5963      PESO=PESO+TIP3(MAYOR,4)*RLON(I)/100
5964 2     CONTINUE
5965      RETURN
5966      END
```

FTN4X COMPILER: HP92834 REV.2130 (810716)

\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: 1466

COMMON: (NONE)

```

5967      SUBROUTINE ELECO(RMP,RMP1,RINER,AREA,NB,SIGMA,NTP,COM,COI,CO)
5968 C
5969 C      RUTINA PARA LA OBTENCION DE LAS PROPIEDADES EN FUNCION DE MP
5970 C
5971      DIMENSION RMP(NB),RMP1(NB),RINER(NB),AREA(NB),NTP(NB),CO(NB)
5972      COMMON /VAR/ IN,IO
5973      DO 1 I=1,NB
5974      S=RMP1(I)/(2*SIGMA)
5975      LK=NTP(I)
5976      IF(LK.EQ.0) LK=1
5977      GO TO (2,3,4),LK
5978 2      RINER(I)=2.911824*(S**1.3622)
5979      AREA(I)=1.537023*(S**0.6435)
5980      IF(RINER(I).LT.77.8) RINER(I)=77.8
5981      IF(AREA(I).LT.7.57) AREA(I)=7.57
5982      GO TO 1
5983 3      RINER(I)=2.649372*(S**1.4)
5984      AREA(I)=1.72521*(S**0.601)
5985      IF(RINER(I).LT.80.1)RINER(I)=80.1
5986      IF(AREA(I).LT.7.64)AREA(I)=7.64
5987      GO TO 1
5988 4      IF(S.LT.640) GO TO 5
5989      RINER(I)=0.7729575*(S**1.5222)
5990      AREA(I)=4.3286*(S**0.515)
5991      GO TO 1
5992 5      RINER(I)=1.751099*(S**1.4)
5993      AREA(I)=2.376434*(S**0.605)
5994      IF(RINER(I).LT.450) RINER(I)=450
5995      IF(AREA(I).LT.26) AREA(I)=26.
5996 1      CONTINUE
5997 10     DO 11 I=1,NB
5998      RMP(I)=RMP1(I)
5999 11     CONTINUE
6000      COM=1.E36
6001      COI=1.E36
6002      DO 12 I=1,NB
6003      IF(COM.GT.RMP(I).AND,RMP(I).GT.1.) COM=RMP(I)
6004      IF(COI.GT.RINER(I)) COI=RINER(I)
6005 12     CONTINUE
6006      DO 13 I=1,NB
6007      RMP(I)=RMP(I)/COM
6008      RMP1(I)=RMP(I)
6009      RINER(I)=RINER(I)/COI
6010      RMP(I)=RMP1(I)*CO(I)
6011 13     CONTINUE
6012      RETURN
6013      END

```

FTN4X COMPILER: HP92834 REV.2130 (810716)

\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: 503 COMMON: (NONE)



```
6014     SUBROUTINE ELETE(NTP,CFI,CMH,NB,COM,RMP1,SIGMA,COI)
6015 C
6016 C     RUTINA PARA LA OBTENCION DE ALFA Y BETA
6017 C
6018     DIMENSION NTP(NB),CFI(NB),CMH(NB),RMP1(NB)
6019     DO 1 I=1,NB
6020     SS=COM/(2.*SIGMA)
6021     S=RMP1(I)*SS
6022     LK=NTP(I)
6023     IF(LK.EQ.0) LK=1
6024     GO TO (2,3,4),LK
6025 2     CFI(I)=-1.36222
6026     CMH(I)=COI/(2.911824*(SS**1.36222))
6027     GO TO 1
6028 3     CFI(I)=-1.4
6029     CMH(I)=COI/(2.649372*(SS**1.4))
6030     GO TO 1
6031 4     CFI(I)=-1.4
6032     CMH(I)=COI/(1.751099*(SS**1.4))
6033     IF(S.LT.640) GO TO 1
6034     CFI(I)=-1.5222
6035     CMH(I)=COI/(0.7729575*(SS**1.5222))
6036 1     CONTINUE
6037     RETURN
6038     END
```

FTN4X COMPILER: HP92834 REV.2130 (810716)

\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: 229 COMMON: (NONE)

```

6039 PROGRAM RESUL(5)
6040 C
6041 C SEGMENTO PARA LA ESCRITURA DE RESULTADOS
6042 C
6043 COMMON /VAR/ IN, IO, NOMB(30), ND, NB, INDT, NMI, NLI, NVIR, NAI, NBCU, NMH,
6044 @ NFI, NGD, NGP, NCL, LHF, LHV, NITER, NND, NH, NBB, NTCPV, NCFP,
6045 @ NCPV, NMIN, NGH, NMCO, NMHCO, NECO, NTRA, LV, CADI, SUMF, E,
6046 @ SUM, TOL, TOLS, SUM1, PESO, SIGMA, COP, COL, COI, COM, TCO
6047 COMMON /CON/ IT(25), RMP(25), RIMON(50), RMOME(50), NI(25), NJ(25),
6048 @ ALFA(25), RLON(25), RINER(25), RMP1(25), AXIL(25), XA(25),
6049 @ YA(25), ZA(25), DCL(20), R(75), LBE(20), T(50), LICOL(50),
6050 @ NCOL(50), NFILA(50), ICOL(50), DCBU(100), T1(50), Q(50),
6051 @ NGB(25), NTP(25), LTP(25), AREA(25), T2(50), CFI(50),
6052 @ CMH(50), CMV(50), FIFI(25), X(25), Y(25), Z(25),
6053 @ RLCPV(10, 2), TETA(50, 50), RMAT(50, 50), TCNP(50, 10)
6054 COMMON /PAS/ NMC, NC, NETAPA, COLAA, COLAP, RMINC(50), RFIC(50), MC(50),
6055 @ RLOF(25), NECNP, NENP, COTA(10)
6056 C
6057 C LLAMADA A LA RUTINA DE ESCRITURA
6058 C
6059 C
6060 C PASO A VALORES REALES
6061 C
6062 LB1=50
6063 DO 1 I=1, NB
6064 AXIL(I)=AXIL(I)*COM/COL
6065 RMP(I)=RMP(I)*COM
6066 1 CONTINUE
6067 DO 2 I=1, 2*NB
6068 RMOME(I)=RMOME(I)*COM
6069 R(I)=R(I)*COM*COL/(E*COI)
6070 2 CONTINUE
6071 DO 3 I=1, ND
6072 X(I)=X(I)*COM*COL*COL/(E*COI)
6073 Y(I)=Y(I)*COM*COL*COL/(E*COI)
6074 Z(I)=Z(I)*COM*COL/(E*COI)
6075 3 CONTINUE
6076 SUM=SUM*COM/(COP*COL)
6077 IF(NBCU.EQ.0) GO TO 5
6078 DO 6 I=1, NB
6079 DCBU(4*I-2)=DCBU(4*I-2)*COL
6080 6 CONTINUE
6081 5 IF(NLI.EQ.0) GO TO 7
6082 DO 8 I=1, NLI
6083 DCL(I)=DCL(I)*COM*COL/(E*COI)
6084 8 CONTINUE
6085 7 CALL RESUT(X, Y, Z, DCL, RMOME, R, NI, NJ, NB, ND, NLI, SUM, LB1, NH, LBE, NBCU,
6086 @ DCBU, NITER, IT, LTP, NCL, RMP1, RMP, NTM, PESO, LHF, LHV, CADI, AXIL,
6087 @ NBB, NND)
6088 C
6089 C TERMINACION
6090 C
6091 IF(NCL.EQ.10.OR.NCL.EQ.15) GO TO 100
6092 GO TO 9
6093 100 NH=2*NB

```

```
6094      NGHH=NGH+NTRA
6095      NMC=NMC+1
6096 9     WRITE(IO,13) TCO
6097      IF(ISSW(8).GE.0) STOP
6098      WRITE(IO,10)
6099      IF(NCL.EQ.15) WRITE(IO,11)NCPV
6100      WRITE(IO,12)NITER,NMIN,NGH,NMCO,NMHCO,NECO,NTRA,NMIN,NH,NGHH,LV
6101      STOP
6102 C
6103 C      FORMATOS DE ESCRITURA
6104 C
6105 10     FORMAT(1H1,2/,10X,"INFORMACION ADICIONAL      :"/)
6106 11     FORMAT(10X,"NUMERO DE ITERACIONES TOTALES PARA 2 ORDEN.....",I3/)
6107 12     FORMAT(
6108 @10X,"NUMERO DE ITERACIONES EN CASO DE C.U.....",I3/
6109 @10X,"NUMERO DE MECANISMOS INDEPENDIENTES.....",I3/
6110 @10X,"GRADO HIPERESTATICO.....",I3/
6111 @10X,"NUMERO DE MOMENTOS EN LA COTA EN ECUACIONES DE COLAPSO..",I3/
6112 @10X,"NUMERO DE MOMENTOS HIPERSTATICOS EN LA COTA.....",I3/
6113 @10X,"NUMERO DE ECUACIONES DE COLAPSO.....",I3/
6114 @10X,"NUMERO DE ROTULAS PASIVAS.....",I3/
6115 @10X,"DIMENSION DE LA MATRIZ DEL S.V.A.....",I3,
6116 @" X",I3/
6117 @10X,"DIMENSION DE LA MATRIZ PARA EL CALCULO DE DEFORMACIONES.",I3,
6118 @" X",I3)
6119 13     FORMAT(2/10X,"*** TIEMPO C.P.U EMPLEADO.....",F10.3," SG.")
6120      END
```

FTN4X COMPILER: HP92834 REV.2130 (810716)

\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: 817 COMMON: (NONE)

```

6121      SUBROUTINE RESUT(X,Y,Z,DCL,RMOME,R,NI,NJ,NB,ND,NLI,SUM,LB1,NH,LBE,
6122 @          NBCU,DCBU,NITER,IT,LTP,NCL,RMP1,RMP,NTM,PESO,
6123 @          LHF,LHV,CADI,AXIL,NBB,NND)
6124 C
6125 C      RUTINA PARA LA ESCRITURA DE LOS RESULTADOS
6126 C
6127      DIMENSION X(ND),Y(ND),Z(ND),DCL(NLI),RMOME(NH),R(NND),NI(ND),
6128 @          NJ(ND),DCBU(NBB),IT(ND),RMP1(NB),RMP(NB),LTP(NB),
6129 @          LBE(NLI),LL(2),AXIL(NB)
6130      COMMON /VAR/ IN,IO,NOMB(30)
6131      WRITE(IO,50) NOMB
6132      IF(LHF.NE.0.OR.LHV.NE.0.OR.CADI.NE.1) GO TO 301
6133      WRITE(IO,30) SUM
6134      GO TO 163
6135 301  WRITE(IO,302) CADI
6136 163  IF(NCL.EQ.10) GO TO 165
6137      IF(NCL.EQ.15) GO TO 180
6138      WRITE(IO,176) PESO
6139      WRITE(IO,166)
6140      DO 167 I=1,NB
6141          IL=NI(I)
6142          JL=NJ(I)
6143          N=LTP(I)/10
6144          L=N*10
6145          J=LTP(I)-L
6146          GO TO(168,169,170),J
6147 168  LL(1)=2HIP
6148          LL(2)=2HN-
6149          GO TO 171
6150 169  LL(1)=2HIP
6151          LL(2)=2HE-
6152          GO TO 171
6153 170  LL(1)=2HHE
6154          LL(2)=2HB-
6155 171  WRITE(IO,172) I,IL,RMOME(2*I-1),RMP1(I),RMP(I),LL,L,JL,RMOME(2*I)
6156 167  CONTINUE
6157      GO TO 175
6158 180  WRITE(IO,181)
6159      GO TO 183
6160 165  WRITE(IO,1)
6161 183  DO 10 I=1,NB
6162          IL=NI(I)
6163          JL=NJ(I)
6164          K1=2*I-1
6165          K2=2*I
6166          IF(NCL.EQ.15) GO TO 184
6167          WRITE(IO,2) I,IL,RMP(I),RMOME(K1),R(K1),JL,RMOME(K2),R(K2)
6168          GO TO 10
6169 184  WRITE(IO,185) I,IL,RMP(I),RMOME(K1),R(K1),AXIL(I),JL,RMOME(K2),
6170 @          R(K2)
6171 10   CONTINUE
6172 175  IF(LHF.NE.0.OR.LHV.NE.0.OR.CADI.NE.1) GO TO 305
6173      WRITE(IO,3)
6174      GO TO 306
6175 305  WRITE(IO,307)

```

```

6176 306 DO 13 I=1,ND
6177 WRITE(IO,4) I,X(I),Y(I),Z(I)
6178 13 CONTINUE
6179 IF(NBCU.EQ.0) GO TO 126
6180 WRITE(IO,131)
6181 DO 127 I=1,ND
6182 IF(IT(I).NE.6) GO TO 127
6183 DO 128 J=1,NB
6184 IF(NJ(J).NE.1) GO TO 128
6185 WRITE(IO,130) I,NI(J),DCBU(4*J-2)
6186 GO TO 127
6187 128 CONTINUE
6188 127 CONTINUE
6189 126 IF(NCL.EQ.10.OR.NCL.EQ.15) GO TO 310
6190 WRITE(IO,14)
6191 DO 15 I=1,NB
6192 IL=NI(I)
6193 JL=NJ(I)
6194 K1=2*I-1
6195 K2=2*I
6196 WRITE(IO,333)I,IL,R(K1),JL,R(K2)
6197 15 CONTINUE
6198 310 IF(NLI.EQ.0) RETURN
6199 WRITE(IO,16)
6200 DO 20 I=1,NLI
6201 LB=LBE(I)/100
6202 LN=LBE(I)-LB*100
6203 WRITE(IO,21) I,LB,LN,DCL(I)
6204 20 CONTINUE
6205 RETURN
6206 C
6207 C FORMATOS DE ESCRITURA
6208 C
6209 50 FORMAT(1H1,10X,30A2/10X,60(' '*1)3/)
6210 302 FORMAT(10X,"CARGA DE DISEÑO :",5X,E14.5/10X,15("-")/)
6211 30 FORMAT(10X,"CARGA DE COLAPSO :",5X,E14.5,/10X,16("-")/)
6212 176 FORMAT(10X,"PESO DE LA ESTRUCTURA :",5X,E14.5," KG.")
6213 166 FORMAT(3/,10X,"DISTRIBUCION DE MOMENTOS Y SOLUCION DE DISEÑO :"/,
6214 @10X,45("-"),3/,10X,"BARRA NUDO MOMENTO M.PLAST E M.PLA
6215 @ST P TIPO-PERFIL"/)
6216 172 FORMAT(10X,I4,3X,I3,1X,E12.5,2X,E12.5,1X,E12.5,5X,2A2,I3/17X,
6217 @I3,1X,E12.5)
6218 181 FORMAT(3/,10X,"DISTRIBUCION DE ESFUERZOS Y ROTACIONES PLASTICAS EN
6219 @ EL COLAPSO:"/10X,62("-")3/10X,"BARRA NUDO MOM. PLAST. MO
6220 @MENTO ROTACION AXIL "/)
6221 1 FORMAT(3/,10X,"DISTRIBUCION DE MOMENTOS Y ROTACIONES PLASTICAS EN
6222 @EL COLAPSO:"/10X,61("-"),3/,10X,"BARRA NUDO MOM. PLAST.
6223 @MOMENTO ROTACION"/)
6224 185 FORMAT(10X,I4,4X,I3,3X,E12.5,2X,E12.5,1X,E11.5,3X,E10.4/18X,I3,
6225 @17X,E12.5,1X,E11.5/)
6226 2 FORMAT(10X,I4,4X,I3,2X,E12.5,4X,E12.5,5X,E12.5/18X,I3,18X,E12.5,
6227 @ 5X,E12.5/)
6228 3 FORMAT(3/,10X,"MOVIMIENTOS DE LOS NUDOS EN EL INSTANTE DEL COLAPSO
6229 @ :"/10X,51("'-")3/10X,"NUDO MOV. X MOV. Y GIRO",/)
6230 307 FORMAT(3/10X,"MOVIMIENTOS DE LOS NUDOS EN EL DISEÑO ELASTOPLASTICO

```

```
6231 @ : "/10X,51("-")3/,10X,"NUDO      MOV. X      MOV. Y      GIRO"/)
6232 131  FORMAT(3/,10X,"POSICION DE LOS NUDOS MOVILES : "/10X,29("-")3/
6233 @10X,"NUDO      NUDO ORIG.      DISTANCIA"/)
6234 130  FORMAT(10X,I3,7X,I3,9X,F12.4)
6235 4    FORMAT(10X,I3,1X,F12.5,1X,F12.5,1X,F12.5/)
6236 21   FORMAT(12X,I2,9X,I2,7X,I2,4X,F12.5)
6237 14   FORMAT(3/,10X,"ROTACIONES PLASTICAS : "/10X,20("-"),3/,10X,
6238 @'BARRA      NUDO      ROTACION'/)
6239 16   FORMAT(3/,10X,"GIROS EN LIBERTADES : "/10X,19("-"),3/,10X,
6240 @'LIBERTAD   BARRA      NUDO      GIRO',/)
6241 333  FORMAT(10X,I4,4X,I3,6X,E12.5/18X,I3,6X,E12.5)
6242      END
```

FTN4X COMPILER: HP92834 REV.2130 (810716)

\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: 1547 COMMON: (NONE)

```

6243      PROGRAM COMUN(5)
6244 C
6245 C      SEGMENTO PARA EL CALCULO DE MATRICES FIJAS EN EL PROBLEMA DE
6246 C      DISEÑO CON RESTRICCIONES.
6247 C
6248      DIMENSION TR2(50,1),NC(50,5)
6249      COMMON /VAR/ IN,IO,NCMB(30),ND,NB,INDT,NMI,NLI,NGL,NAI,NBCU,NMH,
6250 @          NFI,NGD,NGF,NCL,LHF,LHV,NITER,NND,NH,NBB,NTCPV,NCFP,
6251 @          NCFV,NMIN,NGH,NMCO,NMHCO,NECO,NTRA,LV,CADI,SUMF,E,
6252 @          SUM,TOL,TOLS,SUM1,PESO,CN(6)
6253      COMMON /CON/ IT(25),RMP(25),RIMON(50),RMOE(50),NI(25),NJ(25),
6254 @          ALFA(25),RLON(25),RINER(25),RMP1(25),AXIL(25),XA(25),
6255 @          YA(25),ZA(25),DCL(20),R(75),LBE(20),T(50),LICOL(50),
6256 @          NCOL(50),NFILA(50),ICOL(50),DCBU(100),T1(50),Q(50),
6257 @          NGB(25),NTP(25),LTP(25),AREA(25),T2(50),CFI(50),
6258 @          CMH(50),CMV(50),FIFI(25),X(25),Y(25),Z(25),
6259 @          RLCPV(10,2),TETA(50,50),RMAT(50,50),TCNP(50,10)
6260      LB1=50
6261      CALL ZERO(RMAT,LB1,LB1,NMI,NH)
6262      DO 1 I=1,NMI
6263      TR2(I,1)=T1(I)
6264 1      CONTINUE
6265      CALL LEEDI(TETA,LB1,LB1,INDT,2*ND,51,1)
6266      DO 2 I=1,INDT
6267      DO 2 J=1,2*ND
6268      RMAT(J,I)=TETA(I,J)
6269 2      CONTINUE
6270      DO 3 I=1,2*ND
6271      DO 3 J=INDT+1,NMI
6272      RMAT(I,J)=0.0
6273 3      CONTINUE
6274      CALL LEEDI(RMAT,LB1,LB1,2*ND,NMI,61,2)
6275 C
6276 C      OBTENCION DE LOS VIRTUALES Bh
6277 C
6278      CALL ZERO(RMAT,LB1,LB1,2*ND,NMI)
6279      CALL LEEDI(TETA,LB1,LB1,NMI,NH,55,1)
6280      CALL CAGE(NMI,NH,NCOL,ICOL,NVIR,LB1,RMAT,TETA,TOL,1,TR2,1,1,IT,
6281 @          NJ,ND,NB,Q,1)
6282      CALL LEEDI(RMAT,LB1,LB1,NVIR,NH,50,2)
6283      CALL ZERO(RMAT,LB1,LB1,NMI,NH)
6284 C
6285 C      OBTENCION DE TETA REDUCIDA INVERSA
6286 C
6287      CALL LEEDI(TETA,LB1,LB1,NMI,NH,55,1)
6288      CALL OBTER(TETA,RMAT,ICOL,NVIR,NH,NMI,LB1)
6289      CALL INVER(RMAT,NMI,LB1,IER)
6290      IF(IER.NE.0) STOP 5550
6291 C
6292 C      OBTENCION DE U*TETA(-1)r
6293 C
6294      CALL LEEDI(TETA,LB1,LB1,2*ND,NMI,61,1)
6295      CALL PROMA(TETA,RMAT,LB1,LB1,LB1,2*ND,NMI,NMI)
6296      CALL LEEDI(TETA,LB1,LB1,2*ND,NMI,61,2)
6297 C

```

```
6298 C GRABAR LOS DATOS NECESARIOS EN UN FICHERO
6299 C
6300 CALL ZERO(TETA, LB1, LB1, LB1, LB1)
6301 DO 4 I=1, NH
6302 TETA(I, 4)=Q(I)
6303 NC(I, 1)=ICOL(I)
6304 IF(I.GT.NB) GO TO 4
6305 TETA(I, 5)=RMP(I)
6306 TETA(I, 6)=RLON(I)
6307 TETA(I, 1)=CFI(I)
6308 TETA(I, 2)=CMH(I)
6309 NC(I, 3)=NGB(I)
6310 NC(I, 4)=NTP(I)
6311 4 CONTINUE
6312 DO 6 I=1, 6
6313 TETA(I, 7)=CN(I)
6314 6 CONTINUE
6315 DO 7 I=1, 2*ND
6316 TETA(I, 3)=CMV(I)
6317 7 CONTINUE
6318 TETA(7, 7)=TOL
6319 TETA(8, 7)=TOLS
6320 TETA(10, 7)=CADI
6321 TETA(9, 7)=E
6322 NC(1, 5)=IO
6323 NC(2, 5)=NVIR
6324 NC(3, 5)=NB
6325 NC(4, 5)=ND
6326 NC(5, 5)=NGD
6327 NC(6, 5)=NCL
6328 NC(7, 5)=NMI
6329 DO 5 I=1, 30
6330 NC(I, 2)=NOMB(I)
6331 5 CONTINUE
6332 CALL LEEDI(TETA, LB1, LB1, LB1, 7, 59, 2)
6333 CALL LEEDI(NC, LB1, 5, LB1, 5, 62, 2)
6334 C
6335 C LLAMADA A UN NUEVO PROGRAMA PRINCIPAL
6336 C
6337 CALL EXEC(24, 5HPLADI)
6338 STOP
6339 END
```

FTN4X COMPILER: HP92834 REV.2130 (810716)

\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: 858 COMMON: (NONE)



```
2 BLOCK DATA DOW
3 COMMON /VAR/ NVAR(90)
4 COMMON /CON/ NCON(19000)
5 END
```

FTN4X COMPILER: HP92834 REV.2130 (810716)

\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: (NONE) COMMON: (NONE)

BLOCK COMMON CON SIZE: 19000

BLOCK COMMON VAR SIZE: 90

```
6      PROGRAM PLADI
7 C
8 C
9 C      *****
10 C     *
11 C     *
12 C     *   PROGRAMA PARA EL DISEÑO CON RESTRICCIONES EN SERVI-
13 C     *   CIO Y EN ESTADO ULTIMO DE ESTRUCTURAS METALICAS
14 C     *   PLANAS FORMADAS POR BARRAS PRISMATICAS.
15 C     *
16 C     *****
17 C
18 C
19 C     CATEDRA DE ELASTICIDAD Y RESISTENCIA DE MATERIALES.
20 C
21 C     ESCUELA TECNICA SUPERIOR DE INGENIEROS INDUSTRIALES. SEVILLA
22 C
23 C
24 C
25 C     SEVILLA, NOVIEMBRE 1.985
26 C
27 C
28 C
29 C
30 C     PROGRAMA PRINCIPAL RESIDENTE EN MEMORIA
31 C
32 C
33 C
34 C
35 C     LLAMADA AL SEGMENTO CORRESPONDIENTE
36 C
37 C     CALL EXEC(8,5HRSU1)
38 C     END
```

FTN4X COMPILER: HP92834 REV.2130 (810716)

\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: 19 COMMON: (NONE)

```

39 PROGRAM DRSU1(5)
40 C
41 C ESTE SEGMENTO ESTRAE DEL FICHERO LOS DATOS NECESARIOS DEL PROGRAMA
42 C PLASI
43 C
44 COMMON /VAR/ IK, IO, NOMB(30), NH, NB, ND, NGD, NCL, NVIR, TOL, TOLS, NMI,
45 @ CN(6), SECO, E, PORCE, CONTA
46 COMMON /CON/ COE(25), COT(25), COU(50), Q(50), RMP(25), RLON(25),
47 @ ICOL(50), NGB(25), NTP(25), RMCME(50), TETA(50,50),
48 @ NC(50,5)
49 LB1=50
50 CONTA=0
51 CALL LEEDI(TETA, LB1, LB1, LB1, 7, 59, 1)
52 CALL LEEDI(NC, LB1, 5, LB1, 5, 62, 1)
53 NB=NC(3,5)
54 NH=2*NB
55 ND=NC(4,5)
56 NGD=NC(5,5)
57 NCL=NC(6,5)
58 NMI=NC(7,5)
59 NVIR=NC(2,5)
60 IO=NC(1,5)
61 E=TETA(9,7)
62 TOL=TETA(7,7)
63 TOLS=TETA(8,7)
64 SECO=TETA(10,7)
65 DO 1 I=1,30
66 NOMB(I)=NC(I,2)
67 1 CONTINUE
68 DO 2 I=1, NH
69 Q(I)=TETA(I,4)
70 ICOL(I)=NC(I,1)
71 IF(I.GT.NB) GO TO 2
72 RMP(I)=TETA(I,5)
73 RLON(I)=TETA(I,6)
74 NGB(I)=NC(I,3)
75 NTP(I)=NC(I,4)
76 COE(I)=TETA(I,1)
77 COT(I)=TETA(I,2)
78 2 CONTINUE
79 DO 14 I=1,6
80 CN(I)=TETA(I,7)
81 14 CONTINUE
82 DO 15 I=1,2*ND
83 COU(I)=TETA(I,3)
84 15 CONTINUE
85 C
86 C LLAMADA AL SEGMENTO CORRESPONDIENTE
87 C
88 CALL EXEC(8,5HRSU2)
89 END

```

\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: 249 COMMON: (NONE)

```

90      PROGRAM DRSU2(5)
91 C
92 C      ESTE SEGMENTO FORMA LA PARTE CORRESPONDIENTE A CARGA ULTIMA EN
93 C      LA MATRIZ DE DISEÑO. Y OBTIENE LA SOLUCION INICIAL
94 C
95      DIMENSION RMP1(25),R(100),RINER(25),AREA(25),NBAS(100)
96      COMMON /VAR/ IK,IO,NOMB(30),NH,NB,ND,NGD,NCL,NVIR,TOL,TOLS,NMI,
97 @      CN(6),SECO,E,PORCE,CONTA,PESO,INDI
98      COMMON /CON/ COE(25),COT(25),COU(50),Q(50),RMP(25),RLON(25),
99 @      ICOL(50),NGB(25),NTP(25),SS(50),G1(50),G2(50),
100 @      B10(50),B(100),A(90,92)
101      LB1= 90
102      LB2=92
103 C
104 C      PUESTA A CERO
105 C
106      DO 1 I=1,LB1
107      B(I)=0.0
108      DO 1 J=1,LB2
109      A(I,J)=0.0
110 1      CONTINUE
111      IF(NMI+NH+NGD+1.GT.90.OR.2*(NGD+NH)+2.GT.92) STOP 1111
112 C
113 C      FORMACION DE LAS RESTRICCIONES DE EQUILIBRIO EN ESTADO ULTIMO
114 C
115      LL=0
116      DO 7 I=1,NH
117      LL=I-LL
118      L=NGB(LL)
119      CALL LEIND(R,NMI,I,NH,NMI,55,1)
120      DO 8 J=1,NMI
121      A(J,L)=A(J,L)-R(J)
122      A(J,I+NGD)=R(J)
123 8      CONTINUE
124 7      CONTINUE
125      CALL LEIND(R,NMI,NH+1,NH+1,NMI,55,1)
126      DO 9 I=1,NMI
127      A(I,NH+NGD+1)=-R(I)
128 9      CONTINUE
129 C
130 C      FORMACION DE LAS RESTRICCIONES DE SEGURIDAD EN EL ESTADO ULTIMO
131 C
132      N1=NMI
133      L=0
134      DO 11 I=1,NH
135      L=I-L
136      LL=NGB(L)
137      A(N1+I,LL)=A(N1+I,LL)-2.0
138      A(N1+I,I+NGD)=1.0
139      A(N1+I,NH+NGD+1+I)=1.0
140 11     CONTINUE
141 C
142 C      FORMACION DE LAS RESTRICCIONES DE TAMAÑO MINIMO
143 C
144      DO 12 I=1,NGD

```

```

145      R(I)=0.0
146      A(N1+NH+I,I)=1.0
147      A(N1+NH+I,2*NH+NGD+1+I)=-1.0
148 12    CONTINUE
149      DO 18 I=1,NB
150      L=NTP(I)
151      C=11.4*CN(1)*2./CN(5)
152      IF(L.EQ.2) C=C*11.6/11.4
153      IF(L.EQ.3) C=C*52.1/11.4
154      R(NGB(I))=C
155 18    CONTINUE
156      DO 19 I=1,NGD
157      B(N1+NH+I)=R(I)
158 19    CONTINUE
159 C
160 C      FORMACION DE LA RESTRICCION DE SEGURIDAD FRENTE AL COLAPSO
161 C
162      NFIL=NMI+NH+NGD+1
163      NCOL=NGD+2*NH+NGD+2
164      A(NFIL,NGD+NH+1)=1
165      A(NFIL,NCOL)=-1.
166      B(NFIL)=SECO*CN(2)*CN(3)/CN(5)
167 C
168 C      FOMACION DE LA FUNCION OBJETIVO
169 C
170      L=0
171      DO 13 I=1,NB
172      L=NGB(I)
173      A(NFIL+1,L)=A(NFIL+1,L)+RLON(I)
174 13    CONTINUE
175      IF(ISSW(6).GE.0) GO TO 20
176      CALL ESCRM(A,B,LB1,LB2,LB1,NFIL,NCOL,1)
177 C
178 C      RESOLUCION
179 C
180 20    OL=TOL
181      CALL SIMP(A,B,NFIL,NCOL,NCOL,LB1,LB1,TOL,LB2,NBAS,NCOL,NFIL)
182      TOL=OL
183      DO 14 I=1,NB
184      RMP(I)=B(NGB(I))*CN(5)
185      R(I)=1.0
186 14    CONTINUE
187 C
188 C      ESCRITURA DE LA SOLUCION DE MINIMO PESO
189 C
190      WRITE(IO,15)
191      DO 16 I=1,NB
192      WRITE(IO,17) I,RMP(I)
193      RMP1(I)=RMP(I)
194 16    CONTINUE
195      INDI=1
196      DO 23 I=1,2*ND
197      DD=COU(I)
198      COU(I)=COU(I)*(CN(3)*CN(3)*CN(5))/(E*CN(4))
199 23    CONTINUE

```

```
200 C
201 C   OBTENCION DE LAS PROPIEDADES MECANICAS
202 C
203   CALL ELECO(RMP,RMP1,RINER,AREA,NB,CN(1),NTP,CN(5),CN(4),R)
204   CALL ELETE(NTP,COE,COT,NB,CN(5),RMP1,CN(1),CN(4))
205   DO 24 I=1,2*ND
206   COU(I)=COU(I)*(E*CN(4))/(CN(3)*CN(3)*CN(5))
207 24   CONTINUE
208 C
209 C   LLAMADA A OTRO SEGMENTO
210 C
211   CALL EXEC(8,5HDRSU6)
212 C
213 C   FORMATOS DE ESCRITURA
214 C
215 15   FORMAT(1H1,///10X,"SOLUCION DE MINIMO PESO ( Solucion Inicial )"//
216   @10X,"BARRA      MOMENTO PLAST.  ")
217 17   FORMAT(10X,I3,6X,E12.4)
218   END
```

FTN4X COMPILER: HP92834 REV.2130 (810716)

\*\* NO WARNINGS \*\* NO ERRORS \*\* PROGRAM: .1286 COMMON: (NONE)

```

219 PROGRAM DRSU3(5)
220 C
221 C ESTE SEGMENTO OBTIENE Mo y Uo COMPROBANDO QUE LAS RESTRICCIONES
222 C
223 DIMENSION R(50),R1(50)
224 COMMON /VAR/ IK,IO,NOMB(30),NH,NB,ND,NGD,NCL,NVIR,TOL,TOLS,NMI,
225 @ CN(6),SECO,E,PORCE,CONTA,PESO,INDI,NV(2),NEQ,NCOT,POR
226 COMMON /CON/ COE(25),COT(25),COU(50),Q(50),RMP(25),RLON(25),
227 @ ICOL(50),NGB(25),NTP(25),RMOME(50),G1(50),G2(50),
228 @ G10(50),RRR(50),B(100),RMAT(50,50),TETA(50,50)
229 LB1=50
230 NV(1)=0
231 NV(2)=0
232 NND=2*ND
233 100 EE=1.0
234 RC=0.0
235 CONF=CN(5)*CN(3)*CN(3)/(CN(4)*E)
236 CALL SOELA(TETA,RMAT,RMP,Q,RMOME,R,COE,COT,RLON,ICOL,LB1,NH,
237 @ NB,EE,NVIR,NND,NMI,B,CN(5),CN(3),CN(4),CN(2))
238 CALL GEREH(TETA,RMAT,RMP,R,Q,COE,COT,RLON,ICOL,LB1,NH,NB,NVIR,NND,
239 @ G1,G2,G10)
240 DO 3 I=1,2*ND
241 BV=B(I)*CONF
242 BC=COU(I)*CONF
243 IF(ISSW(8),LT.0)WRITE(IO,4) I,BV,BC
244 3 CONTINUE
245 L=0
246 DO 5 I=1,NH
247 L=I-L
248 IF(ISSW(8),LT.0) WRITE(IO,6) I,RMOME(I),RMP(L)*CN(5)*0.9
249 5 CONTINUE
250 C
251 C COMPROBACION DE RESTRICCIONES EN DESPLAZAMIENTOS
252 C
253 II=0
254 DO 10 I=1,2*ND
255 IF(ABS(COU(I)),LE.TOL) GO TO 10
256 RC1=ABS(B(I))-ABS(COU(I))
257 PD=RC1/ABS(B(I))
258 RC1=ABS(B(I)/COU(I))
259 IF(PD.GT.0.001) GO TO 109
260 GO TO 10
261 109 IF(RC1.LE.RC) GO TO 10
262 RC=RC1
263 II=I
264 10 CONTINUE
265 IF(II.EQ.0.AND.ISSW(11),LT.0) WRITE(IO,80)
266 C
267 C COMPROBACION RESTRICCIONES DE MOMENTOS
268 C
269 IG=0
270 IL=0
271 GG=0.0
272 L=0
273 DO 14 I=1,NH

```



```

274 L=I-L
275 PO=(ABS(RMOME(I))-RMP(L)*CN(5)*0.9)/(RMP(L)*CN(5)*0.9)
276 IF(PO.LE.0.001) GO TO 14
277 GG1=ABS(RMOME(I))/(RMP(L)*CN(5)*0.9)
278 IF(GG1.LE.GG) GO TO 14
279 IG=I
280 IL=L
281 GG=GG1
282 14 CONTINUE
283 IF(IG.EQ.0) GO TO 15
284 IF(ISSW(11).LT.0)WRITE(IO,83) IG
285 GK=ABS(RMOME(IG))/(RMP(IL)*CN(5)*0.9)
286 CG=-COE(IL)
287 GGK=GK**CG
288 GO TO 16
289 15 IF(ISSW(11).LT.0)WRITE(IO,84)
290 IF(II.EQ.0.AND.INDI.EQ.1) GO TO 1000
291 IF(II.EQ.0)GO TO 1200
292 GO TO 17
293 16 IF(IG.EQ.0) GO TO 17
294 IF(II.NE.0) GO TO 18
295 21 IF(ISSW(11).LT.0)WRITE(IO,85)
296 RC=GGK
297 NV(1)=1
298 NV(2)=IG
299 GO TO 20
300 18 GGKK=ABS(B(II))/ABS(COU(II))
301 IF(GGKK.LT.GGK) GO TO 21
302 RC=GGKK
303 NV(1)=2
304 NV(2)=II
305 IF(ISSW(11).LT.0)WRITE(IO,86)
306 GO TO 20
307 17 NV(1)=2
308 NV(2)=II
309 RC=ABS(B(II))/ABS(COU(II))
310 C
311 C ESCALADO DEL PROBLEMA
312 C
313 20 DO 11 I=1,NB
314 CC=-COE(I)
315 RMP(I)=RMP(I)*RC**(1,0/CC)
316 11 CONTINUE
317 INDI=0
318 GO TO 100
319 C
320 C DETERMINACION DEL ANCHO DEL POLIEDRO
321 C
322 1200 IF(NV(1).EQ.0.AND.INDI.EQ.1) GO TO 1000
323 IF(NV(1).EQ.0) GO TO 64
324 LL=NV(2)
325 LK1=56
326 LK2=NND
327 PORCE=20.0
328 IF(NV(1).EQ.2) GO TO 101

```