

PONTIFICIA UNIVERSIDAD CATOLICA DEL ECUADOR

PROGRAMAS ACADEMICOS AMBATO I N F O R M A T I C A

CALCULO Y DISEÑO DE ESTRUCTURAS APORTICADAS DE HORMIGON ARMADO

PROMOCION 1992 - 1993

ROMMEL RAFAEL ALEMAN TINTIN

MARCO POLO SILVA SEGOVIA

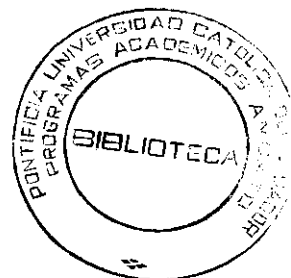
DIRECTOR DE TESIS

ING. EDISON ALVAREZ

ASESOR DE TESIS

ING. PATRICIO BELTRAN

AMBATO - ECUADOR



PONTIFICIA UNIVERSIDAD
CATOLICA DEL ECUADOR

PROGRAMAS ACADÉMICOS AMBATO

CÁLCULO Y DISEÑO DE
ESTRUCTURAS APORTICADAS
DE HORMIGÓN ARMADO

TOMO II

ROMMEL R. ALEMAN TINTIN
MARCO P. SILVA SEGOVIA

CAPITULO VI

6. MANUAL DEL USUARIO

INDICE

INTRODUCCION	1
CARACTERISTICAS	1
REQUISITOS DEL SISTEMA	2
GENERALIDADES	2
RESPALDOS DEL SISTEMA	4
INSTALACION DEL PAQUETE	4
EJECUTANDO EL PAQUETE	9
CREANDO UN PROYECTO	11
OBTENIENDO REPORTES	18
CONFIGURANDO SUS SISTEMA	19
DANDOLE MANTENIMIENTO A SU SISTEMA	22
TERMINANDO LOS TRABAJOS CON CIVIL	28
APPENDICE A	29

MANUAL DEL USUARIO

INTRODUCCION

Uno de los objetivos de importancia en el campo de la ingeniería constituye el diseño del hormigón armado como parte elemental de una estructura que ofrezca seguridad, economía y eficiencia, razón por la cual día a día se desarrollan nuevos métodos y técnicas para diseño, contando entre las principales el método tradicional de coeficientes de trabajo o método elástico y el método de última resistencia o de rotura.

Si bien es cierto que los métodos antes mencionados han sido tratados ampliamente por diversos autores, dicho tratamiento ha sido individualizado por cada uno sin realizar un estudio comparativo de los dos métodos, con el objeto de obtener mayor provecho de las ventajas que tiene cada uno de ellos.

Los estudiantes de esta obra, podrán aprovechar de los pasos metodológicos y técnicos que hemos empleado en el estudio, de tal manera que aparte de simplificar los trabajos les permita tomar nuevas iniciativas que procuren un mayor perfeccionamiento dentro de las técnicas que recomendamos.

Este trabajo ha sido basado en los conocimientos y experiencias del ingeniero civil en estructuras Marco Silva Acosta, quien ha desarrollado una alternativa al método de última resistencia, para calcular y diseñar estructuralmente el hormigón armado, que consiste en optimizar la participación del acero como refuerzo en el hormigón, para la construcción de viviendas con estructuras aporticadas.

CARACTERISTICAS.

El sistema a partir de datos por defecto, proporcionados por los mismos datos adicionales ingresados por el usuario, realiza y proporciona automáticamente:

- Los cálculos de prediseño,
- El mosaico de cargas,
- Cargas sobre las vigas,
- Momento flexionante máximo,
- Peralte de la viga,
- Cargas sobre las columnas,
- Cálculos de diseño,
- Planilla de secciones,
- Planilla de rigideces,
- Resumen de rigideces,
- Planilla de momentos flexionantes,
- Resumen de momentos flexionantes,
- Matriz de rigideces,
- Matriz inversa de rigideces,
- Giros de nudos finales,
- Momentos finales de nudo,
- Momento de borde, momento máximo, V. total, flecha,

- Armadura de vigas,
- Armaduras de columnas,
- Armaduras para losas,
- Armadura para plintos,
- Sección de estribos y
- Geometría.

Adicionalmente el sistema puede mantener un número ilimitado de proyectos.

Todos los procesos y parámetros que son manejados por estos, están debidamente controlados, proporcionando mensajes de advertencia, error o inconsistencia que se pueda producir al momento de ser ingresados.

REQUISITOS DEL SISTEMA.

Para el correcto desenvolvimiento del sistema la información a este deberá ser previamente analizada y revisada por un profesional de la rama, es decir, un Ingeniero Civil.

En cuanto se refiere a los requerimientos de hardware, se deberá tener la siguiente configuración mínima de equipo:

- Un computador AT 80286 o superior,
- Memoria interna (RAM) de 1 Mbyte o superior,
- Un disco duro de 50 Mbyte o superior,
- Una o dos unidades de alta densidad para respaldos,
- Una impresora de 80 o 132 columnas.

GENERALIDADES.

durante la operación del sistema los usuarios se toparán con características comunes a todos los módulos:

1. Todos los mensajes producidos por el sistema, serán mostrados en una ventana ubicada en la parte superior derecha del monitor.
2. La tecla ESC sirve para retroceder un paso en cualquiera de los módulos.
3. La tecla F1 proporciona el acceso a la ayuda en línea, que se incluye en el sistema, pudiendo ser usada en cualquier momento que esta sea requerida.
4. La tecla F2 proporciona un listado de los proyectos existentes para ser seleccionados, esta función es proporcionada siempre y cuando se encuentre en uso la opción de "Datos" ya sea en el modulo de prediseño o de diseño.
5. La tecla F3 activa una calculadora, la cual solo podrá ser usada cuando nos encontremos en una operación de edición de datos.
6. La tecla F4 proporciona un calendario/agenda permanente, que al

igual que la calculadora solo podrá ser usada cuando el sistema se encuentra en una operación de edición de datos.

7. La tecla F5 proporciona una ventana con los códigos ASCII.
8. La tecla F6 proporciona una ventana con los caracteres especiales.
9. El sistema muestra una línea de estado en la parte inferior de la pantalla con la siguiente información: la hora actual, el nombre del proyecto, el puerto de impresora y el directorio por defecto.
10. El sistema es interactivo y validará la información ingresada, si los datos no están dentro del rango, se proporcionará un mensaje con el rango permitido.
11. En los módulos en que se emiten reportes, verifique que su impresora esté encendida y lista.
12. Si el sistema encontrara un error al momento de la ejecución, este mostrará una ventana con la siguiente información.

 Civil 02 Número de Error : Mensaje del Error : Error de la línea de código : Error en la línea número : Programa que produjo el error : Reintentar Cancelar
--

Además proporciona dos opciones:

Reintentar y Cancelar, la primera forza al sistema a reintentar la operación y la otra a cancelar todos los procesos.

13. Todas las ventanas usadas por el sistema están identificadas por claves como:

Civil nn

Donde nn es un número (1..10) que identifica a la ventana para su configuración.

14. El sistema de menús utilizado en el paquete, proporciona un control total sobre todas las opciones de los submenús y entre estos también, su uso se basa en la utilización de las flechas de control del cursor o en la letra de la opción que se encuentra en color diferente.
15. Al producirse un apagón intempestivo del equipo ya sea intencional o producido por un corte de energía, el sistema puede perder la información básica de los archivos de índice, en

cuyo caso, se deberá llevar a cabo una operación de reconstrucción de archivos que se detalla más adelante.

RESPALDOS DEL SISTEMA.

En todo sistema se tiene seguridades especiales para proteger la información contenida en los archivos que se encuentran en los discos, sin embargo siempre existen situaciones eventuales como por ejemplo fallos en la energía eléctrica, fallos de Hardware, etc. que producen daños irreversibles en los archivos de datos. Para evitar este tipo de accidentes es necesario respaldar la información durante períodos de tiempo que varían dependiendo de las necesidades, para este efecto el sistema proporciona un módulo de mantenimiento que facilitará estas operaciones.

INSTALACION DEL PAQUETE.

Para poder instalar el paquete CIVIL deberá usted tener una copia autorizada de este, caso contrario la instalación no se hará en forma correcta.

Los pasos que se dan a continuación deberán ser seguidos con precaución:

1. Verifique que el número de BUFFERS y FILES tengan los siguientes valores:

FILES = 70

BUFFERS = 30,8 (dependiendo de la capacidad de disco)

2. Tener previamente instalado el FOXPRO/LN 2.0
3. Verificar que en el "PATH" se incluya el subdirectorío del FOXPRO2.

SET PATH = C:\FOXPRO2;%PATH%

4. Introduzca el diskette que contiene el sistema civil en uno de los drives A: o B:, dependiendo de en que presentación usted contiene al paquete.
5. Para iniciar la instalación digite la siguiente secuencia de comandos:

Sintaxis:

INSTALL [d:][vía]

Ejemplo:

INSTALL C:\CIVIL <ENTER>

Inmediatamente aparecerá en pantalla un menú que contiene 3 opciones:

Cálculo y diseño de Estructuras Aporticadas
Romel Alemán - Marco Silva - Edwin Chauca

Seleccione la opción a ejecutar:

[V]. Verificar Existencia de FOX
[I]. Instalar Producto (Civil+ 1.00)
[T]. Terminar

Pulse una letra para seleccionar programa ->

De este menú seleccione la opción I que corresponde a la instalación del paquete. Inmediatamente se mostrará en pantalla un mensaje que dice:

Instalando Producto Civil+ 1.00

Presione <ENTER> para continuar.

1 file(s) copied

1 file(s) copied

Pak 1.6 Copyright 1988 - 89 No Gate Consulting.

.
.
.

6. El sistema luego de ser instalado, arrancará automáticamente presentando la siguiente pantalla:



Es importante leer los mensajes mostrados en las pantallas, y si es el caso ejecutar la operación que se indica.

presione una tecla o click del mouse ... Para continuar.

- De inmediato se mostrara el mensaje:

Actualizando parámetros de funcionamiento...

7. Inmediatamente después de presionar <ENTER> aparecerá en pantalla el menú principal.

8. Seleccione de este menú la opción "Sistema", utilizando para el efecto las flechas de control de cursor izquierda a derecha.
9. Seleccione de este submenú la opción "Monitor", presionando la letra M o seleccionando la opción con las flechas de control de cursor abajo o arriba y presionando <ENTER>.
10. Inmediatamente aparecerá la ventana etiquetada como "Civil 02", en la cual se encuentra la siguiente información, que podrá ser editada:

Civil 02																															
*** PARAMETROS DE MONITOR ***																															
TIPO : Mono	<table border="1" style="width: 100%; border-collapse: collapse;"> <tbody> <tr><td>[01]</td><td>11</td><td>W+/N,B+/N,N/W,N+/W,N/W,N/W,W/N,N+/N,N/W,W/N,-</td></tr> <tr><td>[02]</td><td>1</td><td>W+/N,B+/N,N/W,N+/W,N/W,N/W,W/N,N+/N,N/W,W/N,-</td></tr> <tr><td>[03]</td><td>1</td><td>W+/N,B+/N,N/W,N+/W,N/W,N/W,W/N,N+/N,N/W,W/N,-</td></tr> <tr><td>[04]</td><td>1</td><td>W+/N,B+/N,N/W,N+/W,N/W,N/W,W/N,N+/N,N/W,W/N,-</td></tr> <tr><td>[05]</td><td>1</td><td>W+/N,B+/N,N/W,N+/W,N/W,N/W,W/N,N+/N,N/W,W/N,-</td></tr> <tr><td>[06]</td><td>1</td><td>W+/N,B+/N,N/W,N+/W,N/W,N/W,W/N,N+/N,N/W,W/N,-</td></tr> <tr><td>[07]</td><td>10</td><td>W+/N,B+/N,N/W,N+/W,N/W,N/W,W/N,N+/N,N/W,W/N,-</td></tr> <tr><td>[08]</td><td>2</td><td>W+/N,B+/N,N/W,N+/W,N/W,N/W,W/N,N+/N,N/W,W/N,-</td></tr> <tr><td>[09]</td><td>10</td><td>W+/N,B+/N,N/W,N+/W,N/W,N/W,W/N,N+/N,N/W,W/N,-</td></tr> <tr><td>[10]</td><td>10</td><td>W+/N,B+/N,N/W,N+/W,N/W,N/W,W/N,N+/N,N/W,W/N,-</td></tr> </tbody> </table>	[01]	11	W+/N,B+/N,N/W,N+/W,N/W,N/W,W/N,N+/N,N/W,W/N,-	[02]	1	W+/N,B+/N,N/W,N+/W,N/W,N/W,W/N,N+/N,N/W,W/N,-	[03]	1	W+/N,B+/N,N/W,N+/W,N/W,N/W,W/N,N+/N,N/W,W/N,-	[04]	1	W+/N,B+/N,N/W,N+/W,N/W,N/W,W/N,N+/N,N/W,W/N,-	[05]	1	W+/N,B+/N,N/W,N+/W,N/W,N/W,W/N,N+/N,N/W,W/N,-	[06]	1	W+/N,B+/N,N/W,N+/W,N/W,N/W,W/N,N+/N,N/W,W/N,-	[07]	10	W+/N,B+/N,N/W,N+/W,N/W,N/W,W/N,N+/N,N/W,W/N,-	[08]	2	W+/N,B+/N,N/W,N+/W,N/W,N/W,W/N,N+/N,N/W,W/N,-	[09]	10	W+/N,B+/N,N/W,N+/W,N/W,N/W,W/N,N+/N,N/W,W/N,-	[10]	10	W+/N,B+/N,N/W,N+/W,N/W,N/W,W/N,N+/N,N/W,W/N,-
[01]	11	W+/N,B+/N,N/W,N+/W,N/W,N/W,W/N,N+/N,N/W,W/N,-																													
[02]	1	W+/N,B+/N,N/W,N+/W,N/W,N/W,W/N,N+/N,N/W,W/N,-																													
[03]	1	W+/N,B+/N,N/W,N+/W,N/W,N/W,W/N,N+/N,N/W,W/N,-																													
[04]	1	W+/N,B+/N,N/W,N+/W,N/W,N/W,W/N,N+/N,N/W,W/N,-																													
[05]	1	W+/N,B+/N,N/W,N+/W,N/W,N/W,W/N,N+/N,N/W,W/N,-																													
[06]	1	W+/N,B+/N,N/W,N+/W,N/W,N/W,W/N,N+/N,N/W,W/N,-																													
[07]	10	W+/N,B+/N,N/W,N+/W,N/W,N/W,W/N,N+/N,N/W,W/N,-																													
[08]	2	W+/N,B+/N,N/W,N+/W,N/W,N/W,W/N,N+/N,N/W,W/N,-																													
[09]	10	W+/N,B+/N,N/W,N+/W,N/W,N/W,W/N,N+/N,N/W,W/N,-																													
[10]	10	W+/N,B+/N,N/W,N+/W,N/W,N/W,W/N,N+/N,N/W,W/N,-																													

11. Finalmente seleccione la opción "Terminar" y presione <ENTER>. Al salir del sistema aparecerá una pantalla que contiene la siguiente información:

Civil 07	
CERRANDO LAS BASES DE DATOS E INDICES...	

Civil 07	
DESACTIVANDO VENTANAS, ENCERANDO VARIABLES DE MEMORIA...	

Revise el archivo CONFIG.SYS

FILES=70

BUFFERS=30,8

Revise el archivo AUTOEXEC.BAT

PATH C:\DOS:C:\FOXPRO2

Para ejecutar el paquete digite la siguiente lista de Comandos:

CD\CIVIL <ENTER>

CIVIL <ENTER>

Presione <ENTER> para continuar.

EJECUTANDO EL PAQUETE.

Para la ejecución de CIVIL , solo es necesario digitar la siguiente lista de comandos:

1.

CD\CIVIL <ENTER>

CIVIL <ENTER>

2.

Inmediatamente se mostrará un mensaje en la parte superior izquierda de la pantalla que dice:

Espere un momento...

3. A continuación se mostrará la pantalla de presentación inicial, luego de lo cual el sistema pasará a la pantalla de carga de parámetros donde se muestra un mensaje que dice:

presione una tecla o click del mouse ... Para continuar.

- De inmediato se mostrara el mensaje:

Actualizando parámetros de funcionamiento...

CREANDO UN PROYECTO.

1. Del menú principal seleccione el submenú Prediseño.

Prediseño

Datos
Cálculos

2. Seleccione la opción "Datos" usando las flechas de control del cursor y presionando <ENTER>, a continuación se mostrara una ventana etiquetada como "Civil 09". En la parte inferior de esta ventana se muestra un menú de barra con opciones, del cual debe seleccionar la opción "añadir", de inmediato el cursor se ubicara en el campo:

Civil 09

Proyecto NUMERO :

Nombre del Cliente :

Domicilio de la obra :

Tipo de obra :

Ing. Calculista :

Fecha (mm-dd-aa) :

** Datos para Prediseño **

f'c del hormigón:

fy del acero:

No. vanos X:

No. vanos Y:

Espesor loseta:

Peso bloques:

Peso paredes:

Espesor acabados:

Ancho viga:

Ancho columna:

No. de pisos:

Altura de losa:

Carga muerta:

Carga viva:

Carga sísmica:

Carga Ultima:

% de acero mínimo:

Cte.Esfuerzo Unit.:

Prox. Anter. Inicio Fin Buscar Editar añadir borrar Salir

En la parte inferior de la ventana podemos observar un menú, el cual tiene las siguientes opciones:

Prox (Para adelantar un registro).
 Ante (Para retroceder un registro).
 Inicio (Ir al inicio del archivo).
 Fin (Ir al Fin del archivo).
 Buscar (Busca un registro digitando su número, o presionando F2 y seleccionando de la ventana de consulta el registro deseado).
 Editar (Editar o modificar el registro presente).
 añadir (Añade una registro al archivo).
 borrar (Eliminar un registro del archivo).
 Salir (Salir al menú principal).

En el cual aparece un número para el proyecto, el cual es dado automáticamente por el sistema, este número lo podemos cambiar introduciendo otro valor y presionando <ENTER>.

El sistema verifica que el número del proyecto no haya sido dado anteriormente a otro proyecto. A continuación se mostrara el resto de campos.

3. Al llegar al ultimo campo editable aparecerá una tabla con descripciones de tipos de construcción, esta tabla selecciona la carga viva con la cual se calculara la carga muerta, la carga sísmica, la carga última y otro cálculos, la selección se la hace utilizando las flechas de control de cursor y presionando <ENTER>. Si la opción de la tabla necesita alguna aclaración, se mostrará una ventana con una nota de aclaración.

Almacenes
Armerías
Areas de Reuniones ->
Auditorios y galerías
Auditorios y galerías
Cornisas, marquesinas y balcón
Facilidades de salida públicas
Garajes
Garajes
Hospitales
Bibliotecas
.
.
.
Salas de descanso ->

|| Civil 07 ||

Las ocupaciones residenciales incluyen residencias privadas, apartamentos y cuartos de huéspedes de hoteles.
--

4. El valor de la tabla será traspasado al campo de carga viva, este y los siguientes campos se mostrarán en edición, al llegar al campo del "% de acero mínimo", se hará el calculo del valor del esfuerzo unitario, luego de lo cual se mostrara en la parte lateral derecha de la pantalla, una ventana etiquetada como (Civil 00), En donde se mostrara los vanos en el sentido X, en el sentido Y y las alturas entre pisos para ser editados.

|| Civil 0

X01 =
X02 =
X03 =
.
.
.
X20 =

Ingreso de los vanos en el eje X

|| Civil 0

Y01 =
Y02 =
Y03 =
.
.
.
Y20 =

Ingreso de los vanos en el eje Y

|| Civil 0

P06 =
P05 =
P04 =
.
.
.
P01 =

Ingreso de las alturas entre pisos

5. Finalmente si algún dato fue mal ingresado, podemos usar la opción "Editar", para reingresar estos.
6. Cuando se a terminado la edición de los datos, para terminar seleccionamos la opción "Salir", con lo que regresaremos al menú principal.
7. De este menú seleccionamos la opción "Cálculos", utilizando las flechas de control de cursor o la letra "C". En la pantalla se mostrara una ventana etiquetada como (Civil 05), en la cual se muestra la información de identificación del proyecto y la opción de estar seguro de querer continuar. La contestación puede ser ingresada como S, N o seleccionada con la barra espaciadora.

|| Civil 05 ||

CALCULOS PARA PREDISEÑO

Proyecto NUMERO :
 Nombre del Cliente :
 Domicilio de la obra :
 Tipo de obra :
 Ing. Calculista :
 Fecha (mm-dd-aa) :

Esta seguro de querer continuar [S/N] :

Luego de ser contestada, si esta es positiva se mostrara una nueva ventana de mensajes etiquetada como (Civil 06), en la que se mostrara las fases del cálculo y las tablas que se van generando.

|| Civil 06 ||

GENERANDO MATRIZ DE MOSAICO DE CARGAS...

8. Después de que el proceso de Cálculo ha sido completado, podemos obtener los reportes de prediseño, para lo cual seleccionaremos el submenú de "Reportes".

Reportes

A. Datos Generales	(Pre-diseño)
B. Mosaico de cargas	(Pre-diseño)
C. Carga sobre las vigas	(Pre-diseño)
D. Momento Flexionante máximo	(Pre-diseño)
E. Peralte de la viga	(Pre-diseño)
F. Carga sobre las columnas	(Pre-diseño)
.	
.	
.	
W. Geometría	(Diseño)

9. De este menú seleccionamos cualquiera de los reportes de prediseño como:

Datos Generales
 Mosaico de cargas
 Carga sobre las vigas
 Momento Flexionante máximo
 Peralte de la viga
 Carga sobre las columnas

10. Para utilizar cada uno de los reportes simplemente seleccionamos uno de ellos, utilizando las flechas de control de cursor arriba o abajo y presionando <ENTER> o utilizando la letra de color diferente que se encuentra a la izquierda de la opción.

Civil 05	
DATOS DEL PROYECTO	
Proyecto NUMERO :	
Nombre del Cliente :	
Domicilio de la obra :	
Tipo de obra :	
Ing. Calculista :	
Fecha (mm-dd-aa) :	
Reporte por Monitor o Impresora :	

Al seleccionar una de las opciones, aparecerá en pantalla una ventana etiquetada como (Civil 05), en la cual se mostrara el nombre del reporte y los datos generales del proyecto. Se muestra también la siguiente pregunta:

Reporte por Monitor o Impresora : Monitor

La respuesta puede ser introducida o seleccionada, la primera ingresando la letra inicial del dispositivo y la segunda seleccionando con la barra espaciadora.

11. Luego de realizados los cálculos del prediseño, podemos pasar al modulo de diseño, de donde seleccionaremos la opción "Datos", de inmediato se mostrara en pantalla la ventana etiquetada como (Civil 04), la cual muestra los datos y cálculos generados por el prediseño. En la parte inferior de esta ventana se muestra un menú del cual seleccionaremos la opción "Editar" para ingresar los valores de:

Diseño

Datos
Cálculos

Civil 04	
Proyecto No.:	
Nombre del Cliente:	
Cte.hormigón :	Peso sobre columnas:
Cte.flexión viga :	Ancho del Plinto:
Cte.flexión columna :	Esfuer.Admi.Suelo :
Cte.corte y torsión :	% Acero mínimo:
Altura de Losa:	Espacio de Estribos:
Peralte de la viga:	Anál.Pórtico EJE:
Momento Flexionante:	
Peralte de la Columna:	
Prox Ante Inicio Fin Buscar Editar añadir borrar Salir	

En la parte inferior de la ventana podemos observar un menú, el cual tiene las siguientes opciones:

Prox (Para adelantar un registro).
 Ante (Para retroceder un registro).
 Inicio (Ir al inicio del archivo).
 Fin (Ir al Fin del archivo).
 Buscar (Busca un registro digitando su número, o presionando F2 y seleccionando de la ventana de consulta el registro deseado).
 Editar (Editar o modificar el registro presente).
 añadir (Añade una registro al archivo).
 borrar (Eliminar un registro del archivo).
 Salir (Salir al menú principal).

Estas peticiones pueden ser ingresadas o editadas luego de haber realizado los cálculos de prediseño.

12. Luego de haber ingresado o editado todas las peticiones de valores de la opción "Datos", del submenú de Diseño, podemos pasar a realizar los cálculos de diseño, seleccionando de este menú la opción "Cálculos", luego de lo cual se mostrara en pantalla una ventana etiquetada como (Civil 05), que muestra los datos generales del proyecto.

Civil 05	
CALCULOS DE DISEÑO	
Proyecto NUMERO :	
Nombre del Cliente :	
Domicilio de la obra :	
Tipo de obra :	
Ing. Calculista :	
Fecha (mm-dd-aa) :	
Esta seguro de querer continuar [S/N] :	

Verificaremos la pregunta que se encuentra en esta ventana, luego de ser contestada, si esta es positiva se mostrará una nueva ventana de mensajes etiquetada como (Civil 06), en la que se mostrará las fases del cálculo y las tablas que se van generando.

|| Civil 06 ||

GENERANDO PLANILLA DE RIGIDECEC...

13. Finalmente y luego de terminados los cálculos de Diseño, podremos emitir los reportes de este modulo, Utilizando el submenú de reportes y seleccionando cualquiera de los reportes de diseño. Para seleccionar estos deberemos seguir el mismo procedimiento descrito en el prediseño.

Reportes

G. Datos de Diseño	(Diseño)
H. Planilla de secciones	(Diseño)
I. Planilla de Rigideces	(Diseño)
J. Resumen de Rigideces	(Diseño)
K. Planilla MF	(Diseño)
L. Resumen MF	(Diseño)
M. Matriz de Rigidez	(Diseño)
N. Matriz Inversa de Rigidez	(Diseño)
O. Giros de Nudo finales	(Diseño)
P. Momentos Finales de Nudo	(Diseño)
Q. Mborde, Mmax, V Total, Flecha	(Diseño)
R. Armadura de Vigas	(Diseño)
S. Armadura de Columnas	(Diseño)
T. Armadura para Losas	(Diseño)
U. Armadura para Plintos	(Diseño)
V. Sección de Estribos	(Diseño)
W. Geometría	(Diseño)

14. De este menú seleccionamos cualquiera de los reportes de diseño como:

Datos de Diseño	(Diseño)
Planilla de secciones	(Diseño)
Planilla de Rigideces	(Diseño)
Resumen de Rigideces	(Diseño)
Planilla MF	(Diseño)
Resumen MF	(Diseño)
Matriz de Rigidez	(Diseño)
Matriz Inversa de Rigidez	(Diseño)
Giros de Nudo finales	(Diseño)
Momentos Finales de Nudo	(Diseño)
Mborde, Mmax, V Total, Flecha	(Diseño)
Armadura de Vigas	(Diseño)
Armadura de Columnas	(Diseño)
Armadura para Losas	(Diseño)

Armadura para Plintos	(Diseño)
Sección de Estribos	(Diseño)
Geometría	(Diseño)

15. Para utilizar cualquiera de los proyectos ya existentes, deberemos ubicar el proyecto, utilizando para el efecto una de las opciones de Datos del Prediseño o del Diseño y luego recalcular estos en el prediseño y en el diseño respectivamente.

OBTENIENDO REPORTES.

- Para obtener un reporte, seleccionamos del menú principal la opción "Reportes", que activa el correspondiente submenú, en el cual se encuentran las siguientes opciones:

Reportes

A. Datos Generales	(Pre-diseño)
B. Mosaico de cargas	(Pre-diseño)
C. Carga sobre las vigas	(Pre-diseño)
D. Momento Flexionante máximo	(Pre-diseño)
E. Peralte de la viga	(Pre-diseño)
F. Carga sobre las columnas	(Pre-diseño)
G. Datos de Diseño	(Diseño)
H. Planilla de secciones	(Diseño)
I. Planilla de Rigideces	(Diseño)
J. Resumen de Rigideces	(Diseño)
K. Planilla MF	(Diseño)
L. Resumen MF	(Diseño)
M. Matriz de Rigidez	(Diseño)
N. Matriz Inversa de Rigidez	(Diseño)
O. Giros de Nudo finales	(Diseño)
P. Momentos Finales de Nudo	(Diseño)
Q. Mborde, Mmax, V Total, Flecha	(Diseño)
R. Armadura de Vigas	(Diseño)
S. Armadura de Columnas	(Diseño)
T. Armadura para Losas	(Diseño)
U. Armadura para Plintos	(Diseño)
V. Sección de Estribos	(Diseño)
W. Geometría	(Diseño)

Para seleccionar una de estas opciones podemos hacerlo de dos maneras: la primera utilizando las flechas de control de cursor, para ubicar la opción y luego presionar <ENTER>, la segunda es utilizando la letra que se encuentra en color diferente al lado izquierdo de la opción.

Luego de seleccionada la opción se mostrará en pantalla la ventana etiquetada como (Civil 05), en la cual se puede leer el nombre del reporte y los datos generales del proyecto a más de una confirmación que dice:

Reporte por Monitor o Impresora :

Que sirve para seleccionar la salida, pudiendo ser esta por monitor o por impresora, para seleccionar esto, utilizamos la barra espaciadora o la letra inicial de la palabra Monitor o Impresora.

No olvide que para retroceder un paso, simplemente deberá presionar la tecla ESC.

CONFIGURANDO SU SISTEMA.

Al instalar su sistema ya usamos una de las opciones de este menú (Monitor), que sirve para seleccionar el tipo de monitor y las combinaciones de colores de todas las ventanas usadas por el sistema.

A hora veremos como configurar sus sistema para que el Hardware y el Software trabajen en forma óptima.

Sistema

Datos por defecto Monitor Impresora Usuarios

1. Datos por defecto.- Al seleccionar esta opción se visualizará en pantalla la ventana etiquetada como (Civil 09), la cual contiene la información o datos por defecto que son utilizados por los módulos de Prediseño y diseño, los cuales serán mostrados cada vez que se cree un nuevo proyecto.

Civil 09

** DATOS POR DEFECTO PARA EL PREDISEÑO **

Resistencia del hormigón : 210 Kg/cm²
 Resistencia del acero : 4200 Kg/cm²
 Espesor de loseta : 0.0500 m
 Peso de bloques : 0.0072 Tn/m²
 Peso de paredes : 0.1100 Tn/m²
 Espesor de acabados : 0.0200 m
 % de acero mínimo : 0.0160
 Constante de esfuerzo unitario : 16.73

** DATOS POR DEFECTO PARA EL DISEÑO **

Coeficiente de hormigón : 0.85
 Coeficiente de flexión de viga : 0.90
 Coeficiente de flexión de columna : 0.70
 Coeficiente de corte y torsión : 0.80

2. Monitor.- Al seleccionar esta opción se visualizará la ventana etiquetada como (Civil 02), la misma que contiene información sobre los parámetros de Monitor y colores de las ventanas definidas por el sistema. Las peticiones son las siguientes:

Civil 02	
*** PARAMETROS DE MONITOR ***	
TIPO : Mono	
<pre> [01] 11 W+/N,B+/N,N/W,N+/W,N/W,N/W,W/N,N+/N,N/W,W/N,- [02] 1 W+/N,B+/N,N/W,N+/W,N/W,N/W,W/N,N+/N,N/W,W/N,- [03] 1 W+/N,B+/N,N/W,N+/W,N/W,N/W,W/N,N+/N,N/W,W/N,- [04] 1 W+/N,B+/N,N/W,N+/W,N/W,N/W,W/N,N+/N,N/W,W/N,- [05] 1 W+/N,B+/N,N/W,N+/W,N/W,N/W,W/N,N+/N,N/W,W/N,- [06] 1 W+/N,B+/N,N/W,N+/W,N/W,N/W,W/N,N+/N,N/W,W/N,- [07] 10 W+/N,B+/N,N/W,N+/W,N/W,N/W,W/N,N+/N,N/W,W/N,- [08] 2 W+/N,B+/N,N/W,N+/W,N/W,N/W,W/N,N+/N,N/W,W/N,- [09] 10 W+/N,B+/N,N/W,N+/W,N/W,N/W,W/N,N+/N,N/W,W/N,- [10] 10 W+/N,B+/N,N/W,N+/W,N/W,N/W,W/N,N+/N,N/W,W/N,- </pre>	

3. Impresora.- Al seleccionar esta opción se visualizará la ventana etiquetada como (Civil 02), la misma que contiene información sobre los parámetros del puerto de la impresora y los tamaños de letra para los títulos y resultados. Las peticiones son las siguientes:

Civil 02	
** PARAMETROS DE IMPRESORA **	
PUERTO : Lpt1	*CADENA DE CARACTERES PARA TITULOS Y RESU*
TITULOS	CHR(27)+CHR(14) CHR(20)
RESULTADOS	CHR(27)+CHR(27)+CHR(27)+CHR(15) CHR(18)+CHR(27)+CHR(80)

4. Usuarios.- Al seleccionar esta opción se mostrará en pantalla la ventana etiquetada como (Civil 04), en la cual se muestra la siguiente información:

Civil 04	
Clave No.:	1
Nombre del Usuario:	MARCO POLO SILVA SEGOVIA
Clave del Usuario:	MARCO
Prioridad de acceso:	99
Prox Ante Inicio Fin Buscar Editar añadir borRar Salir	

La clave puede ser una combinación de 5 caracteres cualquiera.

La prioridad de acceso puede ser un número entre 1 a 99, mientras mas grande es el valor, mayor será la prioridad de acceso. Este parámetro sólo será verificado en el submenú de "Sistema".

En la parte inferior de la ventana podemos observar un menú, el cual tiene las siguientes opciones:

Prox	(Para adelantar un clave usuario).
Ante	(Para retroceder una clave de usuario).
Inicio	(Ir al inicio del archivo de claves).
Fin	(Ir al Fin del archivo de claves).
Buscar	(Busca una clave digitando su número, o presionando F2 y seleccionando de la ventana de consulta el usuario deseado).
Editar	(Editar o modificar el registro presente).
añadir	(Añade una clave de usuario al archivo).
borRar	(Eliminar un registro de clave de usuario).
Salir	(Salir al menú principal).

Quando en este modulo no se a ingresado ninguna clave de usuario el sistema queda abierto a cualquier usuario, lo que no pasa si se a incluido las respectivas claves.

Si el sistema tiene activo su módulo de seguridad, se podrá verificar que al momento de ingresar a este, se activara una ventana en la cual se pide ingresar su clave, sin la cual usted no podrá acceder al sistema.

Civil+ 1.00

Multi-User FOXPRO/LN 2.00
Serial Number XPDQWEASTY4
M.P.R.S.S.
Su clave :

ACCESO :CONCEDIDO ...

MARCO POLO SILVA SEGOVIA

BIENVENIDO

La clave de acceso podrá ser ingresada hasta por tres ocasiones, luego de lo cual el sistema abortará su funcionamiento.

DANDOLE MANTENIMIENTO A SUS SISTEMA.

Este submenú del sistema provee al usuario un conjunto de herramientas, que le permitirán ver los recursos del sistema, ver el directorio de archivos, copiar archivos, respaldar información con Backup, restaurar la información respaldada con Backup y reconstruir los archivos de índice perdidos o dañados.

Utilitarios

Información Sistema
Ver Directorio
Copiar Archivo
Respaldar
Restaurar
Reconstruir

1. Información Sistema.- Al seleccionar esta opción, se visualizará en pantalla la ventana etiquetada como (Civil 10), la misma que nos mostrará la siguiente información:

Civil 10	
INFORMACION DE CONFIGURACION	
FILES :	70
BUFFERS :	30
MONITOR :	VGA/Mono
PANTALLA :	25 BY 80
FOXPRO ver. :	FoxPro/LAN 2.00
DIRECTORIO :	\CIVIL\
MEM. BASE FOX :	52000
W/EMS MEMORIA :	184192
DISCO Kbytes :	28,033,024

Si el sistema al ser ejecutado no encuentra el archivo CONFIG.SYS mostrará un mensaje pidiendo se ingrese la letra del drive en el cual se encuentre este archivo.

2. Ver Directorio.- Esta opción sirve para visualizar el directorio de archivos del sistema.

Sintaxis:

[d:][via][nombre[.ext]]

[nombre[.ext]] nombre del archivo o conjunto de archivos (*; ?).

Civil 07	
CAMINO:	c:\civil*.* <ENTER>

Luego de presionado <Enter> aparecerá la ventana etiquetada como (Civil 10), en la cual se podrá visualizar el listado de los nombres de archivo que se encuentren en el disco.

Civil 10

MAT19.DBF	MENU003.FXP	MATRIZ.MEM	MONITOR.DBF
MRAC.MEM	MRAV.MEM	MRK.MEM	NP1.MEM
NVX1.MEM	NVX4.MEM	NVY1.MEM	PRECOD.IDX
PREPROY.IDX	PRIORI.MEM	PROCOLOR.APP	PROGS.BAT
TABCOD.IDX	MAT03.DBF	TABLA01.FPT	TABLA02.DBF
U.MEM	MAT04.DBF	USUCOD.IDX	USUIDE.IDX
VANX.MEM	VANY.MEM	Y01.IDX	Y02.IDX
Y03.IDX	Y04.IDX	Y05.IDX	Y06.IDX

3. Copiar Archivo.- Esta opción provee al usuario la facilidad de realizar copias de archivos individuales, o grupos usando los comodines de sistema operativo (*, ?).

Sintaxis:

[d:][vía][nombre[.ext]] [d:][vía][nombre[.ext]]

Al ingresar a la opción se mostrara en pantalla la ventana etiquetada como (Civil 10) la cual muestra el directorio del drive y directorio por defecto.

Civil 10

MAT19.DBF	MENU003.FXP	MATRIZ.MEM	MONITOR.DBF
MRAC.MEM	MRAV.MEM	MRK.MEM	NP1.MEM
NVX1.MEM	NVX4.MEM	NVY1.MEM	PRECOD.IDX
PREPROY.IDX	PRIORI.MEM	PROCOLOR.APP	PROGS.BAT
TABCOD.IDX	MAT03.DBF	TABLA01.FPT	TABLA02.DBF
U.MEM	MAT04.DBF	USUCOD.IDX	USUIDE.IDX
VANX.MEM	VANY.MEM	Y01.IDX	Y02.IDX
Y03.IDX	Y04.IDX	Y05.IDX	Y06.IDX

|| Civil 07 ||

COPY:>C:\CIVIL*.DEF A: <ENTER>

4. Respalidar.- Esta opción es un macro de sistema operativo que utiliza el comando BACKUP de este. Su uso y sintáxis se describe a continuación:

Sintáxis:

```
d:[vía][fichero[.ext]]d:[/S][/M][/D:mm/dd/aa]
[/Thh:mm:ss][/F][/L[:[d:][vía]fichero[.ext]]]
```

Debiéndose especificar:

d:[vía] para la unidad y la vía del fichero o ficheros de los que quiere realizar copia de seguridad.
[fichero[.ext]] nombre del fichero origen a ser copiado

- d: para indicar la unidad de destino
- /S si realiza la copia de seguridad de ficheros de subdirectorios, además del directorio actual
- /M si lo que desea es realizar la copia de los archivos modificados desde el ultimo BACKUP
- /A para añadir los archivos que deben copiarse a los que tengamos en el disco de seguridad.
- /D para realizar copia sólo de los archivos modificados a partir de la fecha indicada.
- /T para realizar copia de los archivos modificados a partir de la hora indicada.
- /F para que realice formateo del disco de copia, si no esta formateado, además de hacer la copia.
- /L para crear un archivo cronológico que contenga:
- fecha y hora de la copia de seguridad
 - número de disquette
 - vía de acceso y nombre de todos los archivos copiados.

Al ingresar a la opción se mostrará en pantalla la ventana etiquetada como (Civil 10) la cual muestra el directorio del drive y directorio por defecto.

Civil 10

MAT19.DBF	MENU003.FXP	MATRIZ.MEM	MONITOR.DBF
MRAC.MEM	MRAV.MEM	MRK.MEM	NP1.MEM
NVX1.MEM	NVX4.MEM	NVY1.MEM	PRECOD.IDX
PREPROY.IDX	PRIORI.MEM	PROCOLOR.APP	PROGS.BAT
TABCOD.IDX	MAT03.DBF	TABLA01.FPT	TABLA02.DBF
U.MEM	MAT04.DBF	USUCOD.IDX	USUIDE.IDX
VANX.MEM	VANY.MEM	Y01.IDX	Y02.IDX
Y03.IDX	Y04.IDX	Y05.IDX	Y06.IDX

Civil 07

BACKUP:>

5. Restaurar.- Esta opción es un macro de sistema operativo que utiliza el comando RESTORE de este. Su uso y sintáxis se describe a continuación:

Sintáxis:

d: [d:][vía][fichero[.ext]][/S][/P] [/B:dd-mm-aa][/A:mm-dd-aa][/M][/N][/L:hora][/E:hora]

Siendo:

/S	Todos los archivos de subdirectorios y de los directorios.
/P	Muestra los ficheros de sólo lectura antes de almacenarlos.
/B:fecha	Para restaurar los ficheros modificados en o antes de la fecha especificada.
/A:fecha	Para restaurar los ficheros modificados en o después de la fecha especificada.
/M	Para restaurar los ficheros modificados o suprimidos desde la copia de seguridad.
/N	Para restaurar los archivos que no existen en

la unidad de destino.
 /L:hora Para restaurar los ficheros modificados en o
 antes de la hora especificada.

Al ingresar a la opción se mostrará en pantalla la ventana etiquetada como (Civil 10) la cual muestra el directorio del drive y directorio por defecto.

Civil 10			
MAT19.DBF	MENU003.FXP	MATRIZ.MEM	MONITOR.DBF
MRAC.MEM	MRAV.MEM	MRK.MEM	NP1.MEM
NVX1.MEM	NVX4.MEM	NVY1.MEM	PRECOD.IDX
PREPROY.IDX	PRIORI.MEM	PROCOLOR.APP	PROGS.BAT
TABCOD.IDX	MAT03.DBF	TABLA01.FPT	TABLA02.DBF
U.MEM	MAT04.DBF	USUCOD.IDX	USUIDE.IDX
VANX.MEM	VANY.MEM	Y01.IDX	Y02.IDX
Y03.IDX	Y04.IDX	Y05.IDX	Y06.IDX

Civil 07
RESTORE:>

6. Reconstruir.- La función de reconstrucción de archivos de índice con la cual se a equipado al sistema, es de gran ayuda al momento de los cortes de energía, durante los cuales se producen daños en estos.

Para ejecutar esta opción bastará con seleccionarla del menú y automáticamente empezará a realizar el trabajo.

Civil 10

2 records copied
2 records indexed
2 records indexed

TABLA DE CRAGA VIVA

25 records copied
25 records indexed

USUARIOS GENERALES

1 records copied

TERMINANDO LOS TRABAJOS CON CIVIL.

La operación de terminar los trabajos, si bien es cierto no ostenta un paso de gran importancia, es también un paso importante en el mantenimiento del sistema, ya que de su correcto uso dependerá la integridad de los datos que el sistema contenga.

La forma correcta de terminar los trabajos es utilizando la opción "Terminar" del menú principal, la cual al ser activada mostrará una ventana de mensajes etiquetada como (Civil 07), en la cual irán apareciendo mensajes como:

|| Civil 07 ||

CERRANDO LAS BASES DE DATOS E INDICES...

|| Civil 07 ||

DESACTIVANDO VENTANAS, ENCERANDO VARIABLES DE MEMORIA...

ETC.

APENDICE A

CORRIDA DE PRUEBA

**** DATOS DEL PROYECTO ****

Proyecto NUMERO : 1
Nombre del Cliente : Arq. Carlos Naranjo
Domicilio de la obra : Latacunga
Tipo de obra : Residencia
Ing. Calculista : Ing. Marco Silva Acosta
Fecha (mm-dd-aa) : 04-20-93

** DATOS PARA PREDISEÑO **

f'c del hormigón: 210 Kg/cm²
fy del acero: 4200 Kg/cm²
Nº vanos X: 3
Nº vanos Y: 2
Espesor loseta: 0.0500 m
Peso bloques: 0.0072 Tn/m²
Peso paredes: 0.1100 Tn/m²
Espesor acabado: 0.0200 m
Ancho viga: 20 cm
Ancho columna: 30 cm
Nº de pisos: 6
Altura de losa: 20 cm
Carga muerta: 496 Kg/m²
Carga viva: 200 Kg/m²
Carga sísmica: 124 Kg/m²
Carga Ultima: 1245 Kg/m²
% de acero minimo: 0.0400
Cte.Esfuerzo Unit.: 16.73

MOSAICO DE CARGAS

PROYECTO DE CONSTRUCCION: Arq. Carlos Naranjo

DIRECCION: Latacunga

CALCULO: Ing. Marco Silva Acosta

FECHA: 04-20-93

CARGAS SOBRE LAS VIGAS SENTIDO X-X (Kg/m)

	A	B	C	D
01	2153.27	2189.12	2153.27	
	2153.27	2189.12	2153.27	
02	1882.97	1896.35	1882.97	
	1882.97	1896.35	1882.97	

CARGAS SOBRE LAS VIGAS SENTIDO Y-Y (Kg/m)

	A	B	B	C	C	D
01	2075.00	2075.00	2075.00	2075.00	2075.00	2075.00
02	1494.00	1494.00	1494.00	1494.00	1494.00	1494.00

CARGA SOBRE LAS VIGAS

PROYECTO DE CONSTRUCCION: Arq. Carlos Naranjo
DIRECCION: Latacunga
CALCULO: Ing. Marco Silva Acosta
FECHA: 04-20-93

CARGA SOBRE LAS VIGAS SENTIDO X-X (Kg/m)

	A	B	C	D
01	2153.27	2189.12	2153.27	
02	4036.24	4085.48	4036.24	
03	1882.97	1896.35	1882.97	

CARGA SOBRE LAS VIGAS SENTIDO Y-Y (Kg/m)

	A	B
01	2075.00	4150.00
02	1494.00	2988.00

MOMENTO FLEXIONANTE MAXIMO

PROYECTO DE CONSTRUCCION: Arq. Carlos Naranjo

DIRECCION: Latacunga

CALCULO: Ing. Marco Silva Acosta

FECHA: 04-20-93

MOMENTO FLEXIONANTE MAXIMO SENTIDO X-X (Kg.m)

	A	B	C	D
01	4852.04	5124.38	4852.04	
02	9095.00	9563.43	9095.00	
03	4242.95	4439.04	4242.95	

MOMENTO FLEXIONANTE MAXIMO SENTIDO Y-Y (Kg.m)

	A	B
01	4322.91	8645.83
02	1613.52	3227.04

PERALTE DE LA VIGA

PROYECTO DE CONSTRUCCION: Arq. Carlos Naranjo

DIRECCION: Latacunga

CALCULO: Ing. Marco Silva Acosta

FECHA: 04-20-93

PERALTE DE LA VIGA SENTIDO X-X (cm)

	A	B	C	D
01	38.08	39.13	38.08	
02	52.13	53.46	52.13	
03	35.60	36.42	35.60	

PERALTE DE LA VIGA SENTIDO Y-Y (cm)

	A	B
01	35.94	50.83
02	21.95	31.05

CARGA SOBRE LAS COLUMNAS

PROYECTO DE CONSTRUCCION: Arq. Carlos Naranjo
DIRECCION: Latacunga
CALCULO: Ing. Marco Silva Acosta
FECHA: 04-20-93

CARGA SOBRE LAS COLUMNAS (Kg)

	A	B	C	D
01	48555.00	98043.75	98043.75	48555.00
02	83514.60	168635.25	168635.25	83514.60
03	34959.60	70591.50	70591.50	34959.60

PERALTE DE COLUMNAS (Kg)

	A	B	C	D
01	13.94	28.16	28.16	13.94
02	23.99	48.44	48.44	23.99
03	10.04	20.28	20.28	10.04

*** DATOS DEL PROYECTO ***

Proyecto NUMERO : 1
 Nombre del Cliente : Arq. Carlos Naranjo
 Domicilio de la obra : Latacunga
 Tipo de obra : Residencia
 Ing. Calculista : Ing. Marco Silva Acosta
 Fecha (mm-dd-aa) : 04-20-93

** DATOS PARA DISEÑO **

Cte.hormigón β :0.85
 Cte.flexión viga ϕ :0.90
 Cte.flexión columna ϕ :0.70
 Cte.corte y torsión τ :0.80
 Altura de Losa: 20 cm
 Peralte de la viga: 53 cm
 Momento Flexionante: 9563.433 Kg-m
 Peralte de la Columna: 48 cm
 Peso sobre columna:168635.250 Kg
 Ancho del Plinto : 2.00 m
 Esfuer.Admi.Suelo δ : 20 Tn/m²
 :
 :
 :
 Análisis Portico "EJE" : 2

PLANILLA DE SECCIONES

PROYECTO DE CONSTRUCCION: Arq. Carlos Naranjo
 DIRECCION: Latacunga
 CALCULO: Ing. Marco Silva Acosta
 FECHA: 04-20-93

ANALISIS PORTICO (X-X) 2 (cm²)

Nivel	A	B	C	D
15.14	20/53	20/53	20/53	
	30/48	30/48	30/48	30/48
12.68	20/53	20/53	20/53	
	30/48	30/48	30/48	30/48
10.22	20/53	20/53	20/53	
	30/48	30/48	30/48	30/48
7.76	20/53	20/53	20/53	
	30/48	30/48	30/48	30/48
5.30	20/53	20/53	20/53	
	30/48	30/48	30/48	30/48
2.84	20/53	20/53	20/53	
	30/48	30/48	30/48	30/48

PLANILLA DE RIGIDECES

PROYECTO DE CONSTRUCCION: Arq. Carlos Naranjo
 DIRECCION: Latacunga
 CALCULO: Ing. Marco Silva Acosta
 FECHA: 04-20-93

ANALISIS PORTICO (X-X) 2 (dm4/m)

Nivel	A	B	C	D
15.14		4.771	4.681	4.771
		2.385	2.340	2.385
	11.239	11.239	11.239	11.239
	5.619	5.619	5.619	5.619
12.68		4.771	4.681	4.771
		2.385	2.340	2.385
	11.239	11.239	11.239	11.239
	5.619	5.619	5.619	5.619
10.22		4.771	4.681	4.771
		2.385	2.340	2.385
	11.239	11.239	11.239	11.239
	5.619	5.619	5.619	5.619
7.76		4.771	4.681	4.771
		2.385	2.340	2.385
	11.239	11.239	11.239	11.239
	5.619	5.619	5.619	5.619
5.30		4.771	4.681	4.771
		2.385	2.340	2.385
	11.239	11.239	11.239	11.239
	5.619	5.619	5.619	5.619
2.84		4.771	4.681	4.771
		2.385	2.340	2.385
	9.735	9.735	9.735	9.735
	4.867	4.867	4.867	4.867

RESUMEN DE RIGIDECES

PROYECTO DE CONSTRUCCION: Arq. Carlos Naranjo
DIRECCION: Latacunga
CALCULO: Ing. Marco Silva Acosta
FECHA: 04-20-93

ANALISIS PORTICO (X-X) 2 (dm⁴/m)

Nivel	A	B	C	D
15.14	16.010	20.692	20.692	16.010
12.68	27.249	31.931	31.931	27.249
10.22	27.249	31.931	31.931	27.249
7.76	27.249	31.931	31.931	27.249
5.30	27.249	31.931	31.931	27.249
2.84	25.745	30.427	30.427	25.745

PLANILLA MOMENTO EMPOTRAMIENTO PE

PROYECTO DE CONSTRUCCION: Arq. Carlos Naranjo
 DIRECCION: Latacunga
 CALCULO: Ing. Marco Silva Acosta
 FECHA: 04-20-93

ANALISIS PORTICO (X-X) 2 (Kg.m)

Nivel	A	B	B	C	C	D
15.14	9095.00	-9095.00	9563.43	-9563.43	9095.00	-9095.00
12.68	9095.00	-9095.00	9563.43	-9563.43	9095.00	-9095.00
10.22	9095.00	-9095.00	9563.43	-9563.43	9095.00	-9095.00
7.76	9095.00	-9095.00	9563.43	-9563.43	9095.00	-9095.00
5.30	9095.00	-9095.00	9563.43	-9563.43	9095.00	-9095.00
2.84	9095.00	-9095.00	9563.43	-9563.43	9095.00	-9095.00

RESUMEN DE MOMENTOS DE EMPOTRAMI

PROYECTO DE CONSTRUCCION: Arq. Carlos Naranjo
DIRECCION: Latacunga
CALCULO: Ing. Marco Silva Acosta
FECHA: 04-20-93

ANALISIS PORTICO (X-X) 2 (Kg.m)

Nivel	A	B	C	D
15.14	9095.00	468.43	-468.43	-9095.00
12.68	9095.00	468.43	-468.43	-9095.00
10.22	9095.00	468.43	-468.43	-9095.00
7.76	9095.00	468.43	-468.43	-9095.00
5.30	9095.00	468.43	-468.43	-9095.00
2.84	9095.00	468.43	-468.43	-9095.00

Time	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
1	25.74	2.38	0.00	0.00	5.61	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
2	2.38	30.42	2.34	0.00	0.00	5.61	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
3	0.00	2.34	30.42	2.38	0.00	0.00	5.61	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
4	0.00	0.00	2.38	25.74	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
5	4.86	0.00	0.00	0.00	27.24	2.38	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
6	4.86	0.00	0.00	0.00	27.24	2.38	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
7	0.00	0.00	4.86	0.00	2.38	31.93	2.34	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
8	0.00	0.00	0.00	0.00	0.00	2.34	31.93	2.38	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
9	0.00	0.00	0.00	0.00	0.00	0.00	2.38	27.24	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
10	0.00	0.00	0.00	0.00	5.61	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
11	0.00	0.00	0.00	0.00	0.00	5.61	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
12	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
13	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
14	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
15	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
16	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
17	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
18	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
19	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
20	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
21	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
22	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
23	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
24	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00

[illegible]

GIROS DE NUDOS FINALES

PROYECTO DE CONSTRUCCION: Arq. Carlos Naranjo
DIRECCION: Latacunga
CALCULO: Ing. Marco Silva Acosta
FECHA: 04-20-93

ANALISIS PORTICO (X-X) 2

	01	02	03	04
el	-312.514	9.932	-9.932	312.514
	-220.385	-0.370	0.370	220.385
	-237.125	2.223	-2.223	237.125
	-249.180	5.981	-5.981	249.180
	-175.582	-11.286	11.286	175.582
	-513.082	44.634	-44.634	513.082

MOMENTOS FINALES DE NUDO

PROYECTO DE CONSTRUCCION: Arq. Carlos Naranjo
 DIRECCION: Latacunga
 CALCULO: Ing. Marco Silva Acosta
 FECHA: 04-20-93

ANALISIS PORTICO (X-X) 2

*** V I G A S *** (Kg.m)

Nivel	A	B	B	C	C	D
15.14	6753.07	-10106.22	9667.92	-9667.92	10106.22	-6753.07
12.68	8230.19	-9567.79	9537.01	-9537.01	9567.79	-8230.19
10.22	7920.18	-9661.00	9577.43	-9577.43	9661.00	-7920.18
7.76	7968.74	-9650.17	9568.63	-9568.63	9650.17	-7968.74
5.30	8042.44	-9622.60	9562.56	-9562.56	9622.60	-8042.44
2.84	7627.38	-9793.26	9586.68	-9586.68	9793.26	-7627.38

*** C O L U M N A S *** (Kg.m)

Nivel	A1	B2	C3	D4
15.14	-6753.29	438.21	-438.21	6753.29
	-4856.88	124.00	-124.00	4856.88
12.68	-3373.75	-93.23	93.23	3373.75
	-3787.30	3.79	-3.79	3787.30
10.22	-4133.17	79.71	-79.71	4133.17
	-4065.43	58.59	-58.59	4065.43
7.76	-3903.61	22.90	-22.90	3903.61
	-3809.55	8.33	-8.33	3809.55
5.30	-4233.23	51.65	-51.65	4233.23
	-4750.90	109.54	-109.54	4750.90
2.84	-3042.32	96.68	-96.68	3042.32
	-1521.31	48.34	-48.34	1521.31

Mborde, Mmax, V Total, Flecha

PROYECTO DE CONSTRUCCION: Arq. Carlos Naranjo
 DIRECCION: Latacunga
 CALCULO: Ing. Marco Silva Acosta
 FECHA: 04-20-93

ANALISIS PORTICO (X-X) 2

(q)	4036.24	4085.48	4036.24
(M)	6753.07 -10106.22	9667.92 -9667.92	10106.22 -6753.07
(Vo)	10494.23 -10494.23	10826.52 -10826.52	10494.23 -10494.23
(Vh)	-644.83 644.83	0.00 0.00	644.83 -644.83
(Vt)	9849.39 11139.07	10826.52 10826.52	11139.07 9849.39
(Mb)	-365.58 -17486.76	-2542.63 -16793.20	-2725.68 -13140.56
(X)	2.44	2.65	2.76
(Mmax)	5264.36	4677.22	5264.36

(q)	4036.24	4085.48	4036.24
(M)	8230.19 -9567.79	9537.01 -9537.01	9567.79 -8230.19
(Vo)	10494.23 -10494.23	10826.52 -10826.52	10494.23 -10494.23
(Vh)	-257.23 257.23	0.00 0.00	257.23 -257.23
(Vt)	10237.00 10751.46	10826.52 10826.52	10751.46 10237.00
(Mb)	-1544.24 -16649.88	-2411.72 -16662.29	-2485.71 -14916.14
(X)	2.53	2.65	2.66
(Mmax)	4751.70	4808.13	4751.70

(q)	4036.24	4085.48	4036.24
(M)	7920.18 -9661.00	9577.43 -9577.43	9661.00 -7920.18
(Vo)	10494.23 -10494.23	10826.52 -10826.52	10494.23 -10494.23
(Vh)	-334.77 334.77	0.00 0.00	334.77 -334.77
(Vt)	10159.46 10829.00	10826.52 10826.52	10829.00 10159.46
(Mb)	-1293.94 -16802.79	-2452.14 -16702.71	-2519.21 -14546.42
(X)	2.51	2.65	2.68
(Mmax)	4865.79	4767.71	4865.79

(q)	4036.24	4085.48	4036.24
(M)	7968.74 -9650.17	9568.63 -9568.63	9650.17 -7968.74
(Vo)	10494.23 -10494.23	10826.52 -10826.52	10494.23 -10494.23
(Vh)	-323.35 323.35	0.00 0.00	323.35 -323.35
(Vt)	10170.88 10817.58	10826.52 10826.52	10817.58 10170.88
(Mb)	-1333.71 -16783.17	-2443.35 -16693.92	-2517.17 -14603.77
(X)	2.52	2.65	2.68
(Mmax)	4845.99	4776.51	4845.99

(q)	4036.24	4085.48	4036.24
(M)	8042.44 -9622.60	9562.56 -9562.56	9622.60 -8042.44
(Vo)	10494.23 -10494.23	10826.52 -10826.52	10494.23 -10494.23
(Vh)	-303.87 303.87	0.00 0.00	303.87 -303.87
(Vt)	10190.35 10798.11	10826.52 10826.52	10798.11 10190.35
(Mb)	-1392.41 -16740.60	-2437.28 -16687.85	-2504.60 -14692.47

(q)	4036.24		4085.48		4036.24	
(M)	7627.38	-9793.26	9586.68	-9586.68	9793.26	-7627.38
(Vo)	10494.23	-10494.23	10826.52	-10826.52	10494.23	-10494.23
(Vh)	-416.51	416.51	0.00	0.00	416.51	-416.51
(Vt)	10077.71	10910.74	10826.52	10826.52	10910.74	10077.71
(Eb)	-1064.08	-16997.99	-2461.39	-16711.96	-2588.53	-14190.68
(X)	2.49		2.65		2.70	
(Hmax)	4953.67		4758.46		4953.67	

ARMADURA DE VIGAS

PROYECTO DE CONSTRUCCION: Arq. Carlos Naranjo
DIRECCION: Latacunga
CALCULO: Ing. Marco Silva Acosta
FECHA: 04-20-93

ANALISIS PORTICO (X-X) 2 (cm²)

As1	-0.131	1.883	-6.254	-0.909	1.883	-6.006	-0.975	1.883	-4.700
As2	-0.552	1.699	-5.955	-0.863	1.699	-5.959	-0.889	1.699	-5.335
As3	-0.463	1.740	-6.009	-0.877	1.740	-5.974	-0.901	1.740	-5.202
As4	-0.477	1.733	-6.002	-0.874	1.733	-5.970	-0.900	1.733	-5.223
As5	-0.496	1.724	-5.987	-0.872	1.724	-5.968	-0.896	1.724	-5.255
As6	-0.381	1.772	-6.079	-0.880	1.772	-5.977	-0.926	1.772	-5.075

MATE

PROYECTO D

Nivel	01	020	021	022	023	024
1	0.04 10	0.00	0.00	0.00	0.00	0.00
2	-0.00 10	0.00	0.00	0.00	0.00	0.00
3	0.00 10	0.00	0.00	0.00	0.00	0.00
4	0.00 10	0.00	0.00	0.00	0.00	0.00
5	-0.00 10	0.00	0.00	0.00	0.00	0.00
6	0.00 10	0.00	0.00	0.00	0.00	0.00
7	0.00 10	0.00	0.00	0.00	0.00	0.00
8	0.00 10	0.00	0.00	0.00	0.00	0.00
9	0.00 10	0.00	-0.00	0.00	0.00	0.00
10	0.00 10	0.00	0.00	0.00	0.00	0.00
11	0.00 10	0.00	0.00	0.00	0.00	0.00
12	0.00 10	0.00	0.00	0.00	0.00	-0.00
13	0.00 10	0.00	0.00	-0.00	0.00	0.00
14	0.00 10	0.00	-0.00	0.00	-0.00	0.00
15	0.00 10	0.00	0.00	-0.00	0.00	-0.00
16	0.00 10	-0.00	0.00	0.00	-0.00	0.00
17	0.00 10	0.00	-0.01	0.00	0.00	0.00
18	0.00 10	0.00	0.00	-0.01	0.00	0.00
19	0.00 13	-0.00	0.00	0.00	-0.01	0.00
20	0.00 10	0.04	0.00	0.00	0.00	-0.01
21	0.00 10	0.00	0.06	-0.00	0.00	0.00
22	0.00 10	0.00	-0.00	0.05	-0.00	0.00
23	0.00 11	0.00	0.00	-0.00	0.05	-0.00
24	0.00 10	-0.01	0.00	0.00	-0.00	0.06

ANALISIS PORTICO (X-X) 2 (cm²)

As6	-39.009	-16.736	-16.736	-39.009
As5	-41.043	-26.398	-26.398	-41.043
As4	-46.525	-36.023	-36.023	-46.525
As3	-50.738	-45.454	-45.454	-50.738
As2	-57.195	-54.983	-54.983	-57.195
As1	-54.266	-63.760	-63.760	-54.266

** ACERO MINIMO ASUMIDO **

As6	23.040	23.040	23.040	23.040
As5	23.040	23.040	23.040	23.040
As4	23.040	23.040	23.040	23.040
As3	23.040	23.040	23.040	23.040
As2	23.040	23.040	23.040	23.040
As1	23.040	23.040	23.040	23.040

ARMADURA DE LOSAS

PROYECTO DE CONSTRUCCION: Arq. Carlos Naranjo
DIRECCION: Latacunga
CALCULO: Ing. Marco Silva Acosta
FECHA: 04-20-93

ANALISIS PORTICO (X-X) (cm²)

AsX	5.758	5.981	5.758
-----	-------	-------	-------

ANALISIS PORTICO (Y-Y) (cm²)

AsY	2.759	5.323
-----	-------	-------

ARMADURA DE PLINTOS

PROYECTO DE CONSTRUCCION: Arq. Carlos Naranjo
DIRECCION: Latacunga
CALCULO: Ing. Marco Silva Acosta
FECHA: 04-20-93

ANALISIS PORTICO (X-X) 2

Ac (m ²)	4.175	8.431	8.431	4.175
Bp (m)	2.043	2.903	2.903	2.043
Ap (m)	2.043	2.903	2.903	2.043
Dp (m)	0.500	0.881	0.881	0.519
AspX (cm ²)	34.077	85.340	85.312	35.370
AspY (cm ²)	34.077	85.340	85.312	35.370

ARMADURA PARA CORTE (ESTRIBOS)

PROYECTO DE CONSTRUCCION: Arq. Carlos Naranjo
DIRECCION: Latacunga
CALCULO: Ing. Marco Silva Acosta
FECHA: 04-20-93

ESTRIBOS PARA VIGAS (cm²)

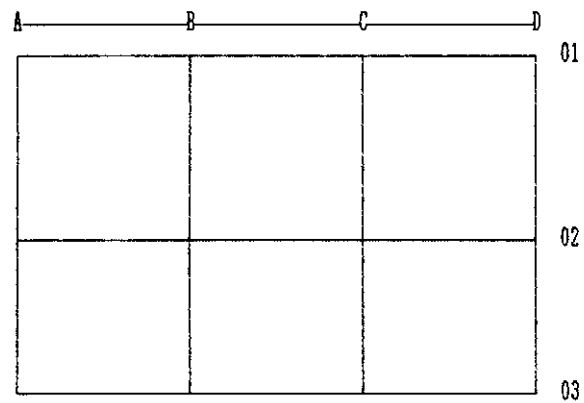
EJE X-X	EJE Y-Y
0.166	0.441

ESTRIBOS PARA COLUMNAS (cm²)

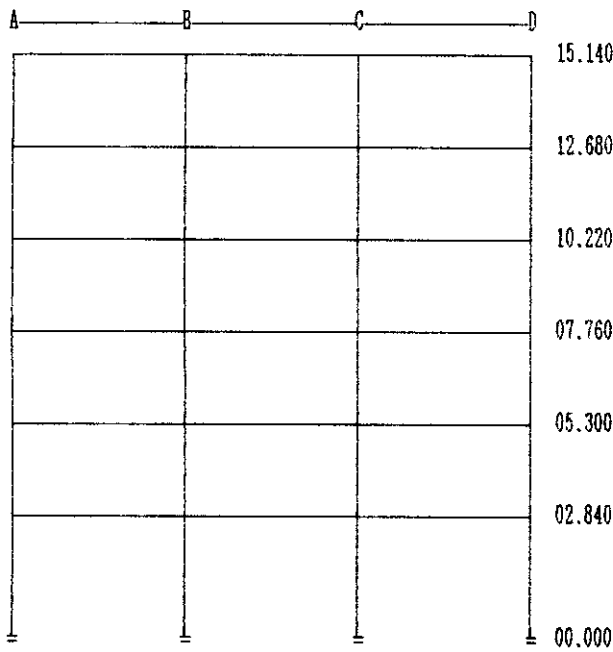
EJE X-X	EJE Y-Y
0.250	0.400

DIRECCION: Natacunga
CALCULO: Ing. Marco Silva Acosta
FECHA: 04-20-93

PLANTA TIPO



PORTICO TIPO



EJE SELECCIONADO --> 2

CAPITULO VII

7. DOCUMENTACION DEL CODIGO FUENTE.

INDICE

CIVIL.PRG	CARATULA INICIAL	1
DEFARCH.PRG	DEFINICION DE ARCHIVOS	2
DEFOBJE.PRG	DEFINICION DE VENTANAS GLOBALES Y MENUS	3
DISE001.PRG	EDICION DE DATOS PARA EL PREDISEÑO	7
DISE002.PRG	CALCULOS Y PROCESOS DE PREDISEÑO	14
MAX00.PRG	PANTALLA BASE Y LOGOTIPO DEL SISTEMA	33
MAX01.PRG	VALIDACION Y VERIFICACION CLAVES USUARIO	35
MENU001.PRG	MENU DE OPCIONES PARA PREDISEÑO	37
MENU002.PRG	MENU DE OPCIONES PARA DISEÑO	38
MENU003.PRG	MENU DE OPCIONES DE REPORTES	39
MENU004.PRG	MENU DE OPCIONES DE PARAMETROS DEL SISTEMA	41
MENU005.PRG	MENU DE OPCIONES UTILIDADES DEL SISTEMA	42
MENU006.PRG	OPCION DE TERMINAR TRABAJOS	43
PRED001.PRG	ENTRADA Y ACTUALIZACION DE DATOS DE DISEÑO	44
PRED002.PRG	CALCULOS DE DISEÑO	54
REPO001.PRG	REPORTE DATOS DEL PROYECTO PREDISEÑO	64
REPO002.PRG	REPORTE MOSAICO DE CARGAS	67
REPO003.PRG	REPORTE CARGA SOBRE LAS VIGAS	71
REPO004.PRG	REPORTE MOMENTO FLEXIONANTE MAXIMO	74
REPO005.PRG	REPORTE PERALTE DE LA VIGA	77
REPO006.PRG	REPORTE CARGA SOBRE LAS COLUMNAS	80
REPO007.PRG	REPORTE DATOS DEL PROYECTO DISEÑO	84
REPO008.PRG	REPORTE PLANILLA DE SECCIONES	87
REPO009.PRG	REPORTE PLANILLA DE RIGIDECESES	90
REPO010.PRG	REPORTE RESUMEN DE RIGIDECESES	93
REPO011.PRG	REPORTE PLANILLA DE MOMENTOS DE EMP.PERF.	97
REPO012.PRG	REPORTE RESUMEN DE MOMENTOS DE EMP.PERF.	100
REPO013.PRG	REPORTE MATRIZ DE RIDEZ	103
REPO014.PRG	REPORTE MATRIZ DE RIGIDEZ INVERSA	106
REPO015.PRG	REPORTE GIROS DE NUDOS FINALES	109
REPO016.PRG	REPORTE MOMENTOS FINALES DE NUDO	112
REPO017.PRG	REPORTE MBORDE,MMAX,V.TOTAL,FLECHA	115
REPO018.PRG	REPORTE ARMADURA DE VIGAS	119
REPO019.PRG	REPORTE ARMADURA DE LUMNAS	121
REPO020.PRG	REPORTE ARMADURA DE LOSAS	124
REPO021.PRG	REPORTE ARMADURA DE PLINTOS	127
REPO022.PRG	REPORTE ARMADURA PARA CORTE DE ESTRIBOS	131
REPO023.PRG	REPORTE GEOMETRIA DE LA ESTRUCTURA	133
RUTINAS.PRG	SUBROUTINAS DE ERRORES Y VENTANA DE SELEC	138
SETENV.PRG	CONFIGURACION DEL AMBIENTE	140
SIST001.PRG	DATOS POR DEFECTO DEL SISTEMA	142
SIST002.PRG	CONFIGURACION DE MONITOR	144
SIST003.PRG	CONFIGURACION DE IMPRESORA	146
SIST004.PRG	ACTUALIZACION DE CLAVES DE USUARIO	148
SYSTEM.PRG	ACTIVACION DEL MENU PRINCIPAL	155
TERM001.PRG	RUTINA DE TERMINACION DE TRABAJOS	156
UTIL001.PRG	INFORMACION DE AMBIENTE DEL SISTEMA	157
UTIL002.PRG	MACRO DEL COMANDO DIR	159
UTIL003.PRG	MACRO DEL COMANDO COPY	160
UTIL004.PRG	MACRO DEL COMANDO BACKUP	161
UTIL005.PRG	MACRO DEL COMANDO RESTORE	162
UTIL006.PRG	RECONSTRUCCION DE ARCHIVOS DE INDICE	164

```

1  *:*****
2  *:
3  *: Procedure file: C:\CIVIL\CIVIL.PRG
4  *:
5  *:      System: Anal.y Diseño de Estructuras Aporticadas
6  *:      Author: Rommel R. Alemán T. - Marco P. Silva S.
7  *:      Copyright (c) 1993, Pontificia Universidad Catolica del E.
8  *:      Last modified: 05/28/93      11:39
9  *:
10 *:      Procs & Fncts: ESCRIBIR
11 *:
12 *:      Calls: SETENV.PRG
13 *:              : ESCRIBIR      (procedure in CIVIL.PRG)
14 *:              : MAX00.PRG
15 *:
16 *:      Documented 05/28/93 at 12:12      FoxDoc version 2.10
17 *:*****
18      DO setenv
19      CLEAR
20      DEFINE WINDOW fondo FROM 00,00 TO 23,79 NONE SHADOW COLOR SCHEME 7
21      DEFINE WINDOW univer FROM 01,02 TO 05,76 SHADOW COLOR SCHEME 1
22      DEFINE WINDOW marca FROM 07,05 TO 16,74 SHADOW COLOR SCHEME 5
23      DEFINE WINDOW nombre FROM 18,11 TO 22,67 SHADOW COLOR SCHEME 1
24      ACTIVATE WINDOW fondo
25      ACTIVATE WINDOW univer
26      DO escribir WITH ' Pontificia Universidad Católica del Ecuador Programas Academicos AMBATO',
00
27      DO escribir WITH '      CALCULO Y DISEÑO DE ESTRUCTURAS APORTICADAS DE HORMIGON ARMADO      ',.01
28      DO escribir WITH '                        ** TRABAJO PRACTICO DE GRADO **                        ',.02
29      ACTIVATE WINDOW marca
30      DO escribir WITH '      '
31      DO escribir WITH '      '
32      DO escribir WITH '      '
33      DO escribir WITH '      '
34      DO escribir WITH '      '
35      DO escribir WITH '      '
36      DO escribir WITH '      '
37      DO escribir WITH '      '
38      ACTIVATE WINDOW nombre
39      DO escribir WITH '      Autores: Romel Alemán Tintín',.00
40      DO escribir WITH '      Edwin Chauca Navas',.01
41      DO escribir WITH '      Marco Silva Segovia',.02
42      STORE 0 TO A
43      FOR i=1 TO 500
44      [ STORE A+1 TO A
45      ]ENDFOR
46      DEACTIVATE WINDOW fondo
47      FOR i=1 TO 600
48      [ STORE A+1 TO A
49      ]ENDFOR
50      DEACTIVATE WINDOW univer
51      FOR i=1 TO 700
52      [ STORE A+1 TO A
53      ]ENDFOR
54      DEACTIVATE WINDOW marca
55      FOR i=1 TO 800

```

```

56   STORE A+1 TO A
57   ]ENDFOR
58   DEACTIVATE WINDOW nombre
59   DO max00
60   RETURN
68   *!*****
69   *!
70   *!      Procedure: ESCRIBIR
71   *!
72   *!      Called by: CIVIL.PRG
73   *!
74   *!*****
68   PROCEDURE escribir
69   PARAMETER texto, linea
70   FOR i=1 TO LEN(texto)
71   [ @linea,i-1 SAY SUBSTR(texto,i,1)
72   [   FOR j=1 TO 50
73   [     ]ENDFOR
74   [ ]ENDFOR
75   RETURN
77   *: EOF: CIVIL.ACT

```

```

1  *:*****
2  *:
3  *:      Program: C:\CIVIL\DEFARCH.PRG
4  *:
5  *:      System: Anal.y Diseño de Estructuras Aporticadas
6  *:      Author: Rommel R. Alemán T. - Marco P. Silva S.
7  *:      Copyright (c) 1993, Pontificia Universidad Catolica del E.
8  *:      Last modified: 04/20/93      16:30
9  *:
10 *:      Called by: MAX00.PRG
11 *:      : UTIL006.PRG
12 *:
13 *:      Uses: MAESTRO.DBF      Alias: PREDISE
14 *:      : TABLA01.DBF      Alias: CARVIVA
15 *:      : TABLA02.DBF      Alias: TABLA
16 *:      : USUARIOS.DBF      Alias: USUARIOS
17 *:      : MAT01.DBF      Alias: MAT01
18 *:      : MAT02.DBF      Alias: MAT02
19 *:      : MAT03.DBF      Alias: MAT03
20 *:      : MAT04.DBF      Alias: MAT04
21 *:      : MAT05.DBF      Alias: MAT05
22 *:      : MAT06.DBF      Alias: MAT06
23 *:      : MAT07.DBF      Alias: MAT07
24 *:      : MAT08.DBF      Alias: MAT08
25 *:      : MAT09.DBF      Alias: MAT09
26 *:      : MAT10.DBF      Alias: MAT10
27 *:      : MAT11.DBF      Alias: MAT11
28 *:      : MAT12.DBF      Alias: MAT12
29 *:      : MAT13.DBF      Alias: MAT13
30 *:      : MAT14.DBF      Alias: MAT14
31 *:      : MAT15.DBF      Alias: MAT15
32 *:      : MAT16.DBF      Alias: MAT16
33 *:      : MAT17.DBF      Alias: MAT17
34 *:      : MAT18.DBF      Alias: MAT18
35 *:      : MAT19.DBF      Alias: MAT19
36 *:
37 *:      Indexes: PRECOD.IDX
38 *:      : PREPROY.IDX
39 *:      : TABCOD.IDX
40 *:      : USUCOD.IDX
41 *:      : USUIDE.IDX
42 *:      : Y01.IDX
43 *:      : Y02.IDX
44 *:      : Y03.IDX
45 *:      : Y04.IDX
46 *:      : Y05.IDX
47 *:      : Y06.IDX
48 *:      : Y07.IDX
49 *:      : Y08.IDX
50 *:      : Y09.IDX
51 *:      : Y10.IDX
52 *:      : Y11.IDX
53 *:      : Y12.IDX
54 *:      : Y13.IDX
55 *:      : Y14.IDX
56 *:      : Y15.IDX

```

```

57 *:           : Y16.IDX
58 *:           : Y17.IDX
59 *:           : Y18.IDX
60 *:           : Y19.IDX
61 *:
62 *:   Other Files: HELP
63 *:
64 *:   Documented 05/28/93 at 12:12           FoxDoc version 2.10
65 *:*****
66   PARAMETER DIREC
67   CLOSE DATA
68   USE maestro IN 01 INDEX precod, preproy ALIAS predise
69   USE tabla01 IN 02 INDEX tabcod ALIAS carviva
70   USE tabla02 IN 03 ALIAS tabla
71   USE usuarios IN 04 INDEX usucod, usuide ALIAS usuarios
72   USE mat01 IN 05 INDEX y01 ALIAS mat01
73   USE mat02 IN 06 INDEX y02 ALIAS mat02
74   USE mat03 IN 07 INDEX y03 ALIAS mat03
75   USE mat04 IN 08 INDEX y04 ALIAS mat04
76   USE mat05 IN 09 INDEX y05 ALIAS mat05
77   USE mat06 IN 10 INDEX y06 ALIAS mat06
78   USE mat07 IN 11 INDEX y07 ALIAS mat07
79   USE mat08 IN 12 INDEX y08 ALIAS mat08
80   USE mat09 IN 13 INDEX y09 ALIAS mat09
81   USE mat10 IN 14 INDEX y10 ALIAS mat10
82   USE mat11 IN 15 INDEX y11 ALIAS mat11
83   USE mat12 IN 16 INDEX y12 ALIAS mat12
84   USE mat13 IN 17 INDEX y13 ALIAS mat13
85   USE mat14 IN 18 INDEX y14 ALIAS mat14
86   USE mat15 IN 19 INDEX y15 ALIAS mat15
87   USE mat16 IN 20 INDEX y16 ALIAS mat16
88   USE mat17 IN 21 INDEX y17 ALIAS mat17
89   USE mat18 IN 22 INDEX y18 ALIAS mat18
90   USE mat19 IN 23 INDEX y19 ALIAS mat19
91   SET HELP TO HELP
92   ON KEY LABEL f1 HELP
93   RETURN
95   *: EOF: DEFARCH.ACT

```

```

1  *:*****
2  *:
3  *:      Program: C:\CIVIL\DEFOBJE.PRG
4  *:
5  *:      System: Anal.y Diseño de Estructuras Aporticadas
6  *:      Author: Rommel R. Alemán T. - Marco P. Silva S.
7  *:      Copyright (c) 1993, Pontificia Universidad Catolica del E.
8  *:      Last modified: 05/09/93      17:35
9  *:
10 *:      Called by: MAX00.PRG
11 *:
12 *:      Calls: MENU001.PRG
13 *:              : MENU002.PRG
14 *:              : MENU003.PRG
15 *:              : MENU004.PRG
16 *:              : MENU005.PRG
17 *:              : MENU006.PRG
18 *:
19 *:      Memory Files: COLORES.MEM
20 *:
21 *:      Documented 05/28/93 at 12:12      FoxDoc version 2.10
22 *:*****
23  RESTORE FROM colores
24  DEFINE WINDOW tbla FROM 08,50 TO 16,75 PANEL SHADOW;
25  TITLE ' Civil 01 ' COLOR SCHEME mcr1
26  DEFINE WINDOW sistema FROM 03,04 TO 21,75 DOUBLE SHADOW;
27  TITLE ' Civil 02 ' COLOR SCHEME mcr2
28  DEFINE WINDOW vano FROM 02,65 TO 23,79 DOUBLE SHADOW;
29  TITLE ' Civil 03 ' COLOR SCHEME mcr3
30  DEFINE WINDOW disenio FROM 06,02 TO 20,77 DOUBLE SHADOW;
31  TITLE ' Civil 04 ' COLOR SCHEME mcr4
32  DEFINE WINDOW reportes FROM 06,06 TO 20,73 DOUBLE SHADOW;
33  TITLE ' Civil 05 ' COLOR SCHEME mcr5
34  DEFINE WINDOW impresor FROM 19,00 TO 23,79 DOUBLE SHADOW;
35  TITLE ' Civil 06 ' COLOR SCHEME mcr6
36  DEFINE WINDOW mensajes FROM 08,02 TO 12,76 DOUBLE SHADOW;
37  TITLE ' Civil 07 ' COLOR SCHEME mcr7
38  DEFINE WINDOW memos FROM 08,03 TO 16,76 DOUBLE SHADOW;
39  TITLE ' Civil 08 ' COLOR SCHEME mcr8
40  DEFINE WINDOW trabajo FROM 01,00 TO 23,79 PANEL SHADOW;
41  TITLE ' Civil 09 ' COLOR SCHEME mcr9
42  DEFINE WINDOW tabla FROM 03,02 TO 21,77 PANEL SHADOW;
43  TITLE ' Civil 10 ' CLOSE FLOAT GROW ZOOM COLOR SCHEME mcr10
44  DEFINE MENU menuprin COLOR SCHEME 2
45  DEFINE PAD predisenio OF menuprin PROMPT ' \<Prediseño ' AT 00,01
46  DEFINE PAD disenio OF menuprin PROMPT ' \<Diseño ' AT 00,15
47  DEFINE PAD reportes OF menuprin PROMPT ' \<Reportes ' AT 00,26
48  DEFINE PAD sistema OF menuprin PROMPT ' \<Sistema ' AT 00,39
49  DEFINE PAD util OF menuprin PROMPT ' \<Utilitarios ' AT 00,51
50  DEFINE PAD termina OF menuprin PROMPT ' \<Termina ' AT 00,68
51  ON PAD predisenio OF menuprin ACTIVATE POPUP menupred
52  ON PAD disenio OF menuprin ACTIVATE POPUP menudise
53  ON PAD reportes OF menuprin ACTIVATE POPUP menurepo
54  ON PAD sistema OF menuprin ACTIVATE POPUP menusist
55  ON PAD util OF menuprin ACTIVATE POPUP menuutil
56  ON PAD termina OF menuprin ACTIVATE POPUP menuterm

```

```

57  DEFINE POPUP      menupred FROM 01,01 SHADOW COLOR SCHEME 2
58  DEFINE BAR 1 OF  menupred PROMPT '\<Datos'
59  DEFINE BAR 2 OF  menupred PROMPT '\<Cálculos'
60  ON SELECTION POPUP menupred DO menu001
61  DEFINE POPUP      menudise FROM 01,15 SHADOW COLOR SCHEME 2
62  DEFINE BAR 1 OF  menudise PROMPT '\<Datos'
63  DEFINE BAR 2 OF  menudise PROMPT '\<Cálculos'
64  ON SELECTION POPUP menudise DO menu002
65  DEFINE POPUP      menurepo FROM 01,26 SHADOW COLOR SCHEME 2
66  DEFINE BAR 1 OF  menurepo PROMPT '\<A. Datos Generales      (Pre-diseño)'
67  DEFINE BAR 2 OF  menurepo PROMPT '\<B. Mosaico de cargas    (Pre-diseño)'
68  DEFINE BAR 3 OF  menurepo PROMPT '\<C. Carga sobre las vigas (Pre-diseño)'
69  DEFINE BAR 4 OF  menurepo PROMPT '\<D. Momento Flexionante Maximo (Pre-diseño)'
70  DEFINE BAR 5 OF  menurepo PROMPT '\<E. Peralte de la viga    (Pre-diseño)'
71  DEFINE BAR 6 OF  menurepo PROMPT '\<F. Carga sobre las columnas (Pre-diseño)'
72  DEFINE BAR 7 OF  menurepo PROMPT '\<G. Datos de Diseño      (Diseño)'
73  DEFINE BAR 8 OF  menurepo PROMPT '\<H. Planilla de Secciones (Diseño)'
74  DEFINE BAR 9 OF  menurepo PROMPT '\<I. Planilla de Rigideces (Diseño)'
75  DEFINE BAR 10 OF menurepo PROMPT '\<J. Resumen de Rigideces (Diseño)'
76  DEFINE BAR 11 OF menurepo PROMPT '\<K. Planilla MF          (Diseño)'
77  DEFINE BAR 12 OF menurepo PROMPT '\<L. Resumen MF          (Diseño)'
78  DEFINE BAR 13 OF menurepo PROMPT '\<M. Matriz de Rigides    (Diseño)'
79  DEFINE BAR 14 OF menurepo PROMPT '\<N. Matriz Inversa de Rigides (Diseño)'
80  DEFINE BAR 15 OF menurepo PROMPT '\<O. Giros de Nudo Finales (Diseño)'
81  DEFINE BAR 16 OF menurepo PROMPT '\<M. Momentos Finales de Nudo (Diseño)'
82  DEFINE BAR 17 OF menurepo PROMPT '\<Q. Mborde, Mmax, V Total, Flecha (Diseño)'
83  DEFINE BAR 18 OF menurepo PROMPT '\<R. Armadura de Vigas      (Diseño)'
84  DEFINE BAR 19 OF menurepo PROMPT '\<S. Armadura de Columnas   (Diseño)'
85  DEFINE BAR 20 OF menurepo PROMPT '\<T. Armadura para Losas    (Diseño)'
86  DEFINE BAR 21 OF menurepo PROMPT '\<U. Armadura para Plintos  (Diseño)'
87  DEFINE BAR 22 OF menurepo PROMPT '\<V. Sección de Estribos   (Diseño)'
88  DEFINE BAR 23 OF menurepo PROMPT '\<W. Geometría             (Diseño)'
89  ON SELECTION POPUP menurepo DO menu003
90  DEFINE POPUP      menusist FROM 01,39 SHADOW COLOR SCHEME 2
91  DEFINE BAR 1 OF  menusist PROMPT '\<Datos por defecto'
92  DEFINE BAR 2 OF  menusist PROMPT '\<Monitor'
93  DEFINE BAR 3 OF  menusist PROMPT '\<Impresora'
94  DEFINE BAR 4 OF  menusist PROMPT '\<Usuarios'
95  ON SELECTION POPUP menusist DO menu004
96  DEFINE POPUP      menuutil FROM 01,51 SHADOW COLOR SCHEME 2
97  DEFINE BAR 1 OF  menuutil PROMPT '\<Información Sistema'
98  DEFINE BAR 2 OF  menuutil PROMPT '\<Ver Directorio'
99  DEFINE BAR 3 OF  menuutil PROMPT '\<Copiar Archivo'
100 DEFINE BAR 4 OF  menuutil PROMPT '\<Respaldar'
101 DEFINE BAR 5 OF  menuutil PROMPT '\<Resta\urar'
102 DEFINE BAR 6 OF  menuutil PROMPT '\<Reconstruir'
103 ON SELECTION POPUP menuutil DO menu005
104 DEFINE POPUP      menuterm FROM 01,68 SHADOW COLOR SCHEME 2
105 DEFINE BAR 1 OF  menuterm PROMPT '\<Salir'
106 ON SELECTION POPUP menuterm DO menu006
108 *: EOF: DEFOBJE.ACT

```

```

1  *:*****
2  *:
3  *: Procedure file: C:\CIVIL\DISK001.PRG
4  *:
5  *:      System: Anal.y Diseño de Estructuras Aportricadas
6  *:      Author: Rommel R. Alemán T. - Marco P. Silva S.
7  *:      Copyright (c) 1993, Pontificia Universidad Catolica del E.
8  *:      Last modified: 05/16/93      16:16
9  *:
10 *:      Set by: MENU002.PRG
11 *:
12 *:      Calls: SETUP_REP      (procedure in PRED001.PRG)
13 *:              : DESPLEGAR      (procedure in PRED001.PRG)
14 *:              : MOSTRAR      (procedure in PRED001.PRG)
15 *:
16 *:      Documented 05/28/93 at 12:12      FoxDoc version 2.10
17 *:*****
18  RELEASE ALL
19  STORE '' TO mproyecto
20  SET FUNCTION f2 TO 'B:'
21  STORE 0 TO maux
22  DO setup_rep
23  SELECT predise
24  SET ORDER TO 1
25  IF EOF()
26  GO TOP
27  ENDIF
28  STORE codigo TO mcodigo
29  STORE DATE() TO mfecha
30  ACTIVATE WINDOW disenio
31  DO desplegar
32  DO mostrar
33  STORE .T. TO reporte
34  DO WHILE reporte
35  ACTIVATE MENU salesrep PAD search
36  ENDDO
37  DEACTIVATE WINDOW disenio
38  RELEASE MENU salesrep
39  CLOSE PROCEDURE
40  @24,23 SAY LEFT(mproyecto,25) COLOR SCHEME 5
41  RETURN
56 *!*****
57 *!
58 *!      Procedure: INCREMENTAR
59 *!
60 *!      Called by: SETUP_REP      (procedure in PRED001.PRG)
61 *!
62 *!      Calls: DESPLEGAR      (procedure in PRED001.PRG)
63 *!              : MOSTRAR      (procedure in PRED001.PRG)
64 *!              : STANDBY      (procedure in PRED001.PRG)
65 *!              : EDITAR      (procedure in PRED001.PRG)
66 *!
67 *!      Memory Files: DEFECTO.MEM
68 *!
69 *!*****
56  PROCEDURE incrementar

```

```

57     WAIT 'No puede incrementar proyectos de diseño ...' WINDOW
58     RETURN
72  *!*****
73  *!
74  *!     Procedure: BORRAR
75  *!
76  *!     Called by: SETUP_REP           (procedure in PRED001.PRG)
77  *!
78  *!     Calls: MOSTRAR                (procedure in PRED001.PRG)
79  *!
80  *!     Uses: &NOM1.DBF
81  *!           : &NOM2.DBF
82  *!           : &NOM3.DBF
83  *!
84  *!*****
72  PROCEDURE borrar
73  WAIT 'No puede eliminar proyectos de diseño ...' WINDOW
74  RETURN
85  *!*****
86  *!
87  *!     Procedure: DESPLEGAR
88  *!
89  *!     Called by: PRED001.PRG
90  *!                 : DISE001.PRG
91  *!                 : SIST004.PRG
92  *!                 : INCREMETAR      (procedure in PRED001.PRG)
93  *!
94  *!*****
85  PROCEDURE desplegar
86  CLEAR
87  @01,01 SAY '      Proyecto No.: '
88  @02,01 SAY '      Nombre del Cliente: '
89  @03,01 SAY '      Cte.hormigón β: '
90  @04,01 SAY '      Cte.flexión viga φ: '
91  @05,01 SAY '      Cte.flexión columna φ: '
92  @06,01 SAY '      Cte.corte y torsión τ: '
93  @07,01 SAY '      Altura de Losa: '
94  @08,01 SAY '      Peralte de la viga: '
95  @09,01 SAY '      Momento Flexionante: ' COLOR SCHEME 8
96  @10,01 SAY '      Peralte de la Columna: '
97  @03,35 SAY '      Peso sobre columna: ' COLOR SCHEME 8
98  @04,35 SAY '      Ancho del Plinto : '
99  @05,35 SAY '      Esfuer.Admi.Suelo δ: '
100 @06,35 SAY '      % Acero minimo : '
101 @07,35 SAY '      Espacio de Estribos: '
102 @08,35 SAY '      Anál. Portico EJE: '
103 RETURN
116 *!*****
117 *!
118 *!     Procedure: EDITAR
119 *!
120 *!     Called by: SETUP_REP           (procedure in PRED001.PRG)
121 *!                 : INCREMETAR      (procedure in PRED001.PRG)
122 *!
123 *!     Calls: MOSTRAR                (procedure in PRED001.PRG)
124 *!           : ESCOJER              (procedure in RUTINAS.PRG)

```

```

125 *!           : DATOS           (procedure in PRED001.PRG)
126 *!
127 *!*****
116   PROCEDURE editar
117   IF RECCOUNT()#0
118       @01,24 GET codigo PICTURE '####'
119       CLEAR GETS
120       @02,24 SAY proyecto
121       @03,24 GET beta PICTURE '#.##'          RANGE 0.01, 9.99
122       @04,24 GET fi1 PICTURE '#.##'          RANGE 0.01, 9.99
123       @05,24 GET fi2 PICTURE '#.##'          RANGE 0.01, 9.99
124       @06,24 GET fi3 PICTURE '#.##'          RANGE 0.01, 9.99
125       @07,24 GET h1 PICTURE '### cm'          RANGE 1,999
126       @08,24 GET D PICTURE '#### cm'          RANGE 1,9999
127       @09,24 SAY m1 PICTURE '#####.### Kg-m'
128       @10,24 GET H PICTURE '#### cm'          RANGE 1,9999
129       @03,56 GET pc PICTURE '#####.### Kg' RANGE 000000.001, 999999.999
130       @04,56 GET bp PICTURE '###.## m'          RANGE 000.01, 999.99
131       @05,56 GET ss PICTURE '#### Tn/m²'          RANGE 1,9999
132       @06,56 GET ph1 PICTURE '##.####'          RANGE 0.0100,0.0800
133       @07,56 GET ee PICTURE '## cm'          RANGE 10,25
134       STORE VAL(xx) TO mxx
135       STORE nvy TO mnvy
136       @08,56 GET mxx PICTURE '99' RANGE 1,mnvy+1
137       READ
138       IF csp <=0
139           REPLA csp WITH 1
140       ENDIF
141       IF ee<=0
142           REPLA ee WITH 10
143       ENDIF
144       REPLA xx WITH STR(mxx,2)
145       DO mostrar
146   ENDIF
147   RETURN
157 *!*****
158 *!
159 *!   Procedure: FINAL
160 *!
161 *!   Called by: SETUP_REP          (procedure in PRED001.PRG)
162 *!
163 *!   Calls: MOSTRAR          (procedure in PRED001.PRG)
164 *!
165 *!*****
157   PROCEDURE final
158   GO BOTTOM
159   DO mostrar
160   RETURN
171 *!*****
172 *!
173 *!   Procedure: PROXIMO
174 *!
175 *!   Called by: SETUP_REP          (procedure in PRED001.PRG)
176 *!
177 *!   Calls: STANDBY          (procedure in PRED001.PRG)
178 *!           : MOSTRAR          (procedure in PRED001.PRG)

```

```

179 *!
180 *!*****
171     PROCEDURE proximo
172     IF RECCOUNT()#0
173     SKIP
174     IF EOF()
175     GO BOTTOM
176     DO standby WITH 'El registro presente es el ultimo...'
177     ELSE
178     DO mostrar
179     ENDIF
180     ENDIF
181     RETURN
192 *!*****
193 *!
194 *!     Procedure: ANTERIOR
195 *!
196 *!     Called by: SETUP_REP           (procedure in PRED001.PRG)
197 *!
198 *!     Calls: STANDBY                 (procedure in PRED001.PRG)
199 *!           : MOSTRAR               (procedure in PRED001.PRG)
200 *!
201 *!*****
192     PROCEDURE anterior
193     IF RECCOUNT()#0
194     SKIP -1
195     IF BOF()
196     GO TOP
197     DO standby WITH 'El registro presente es el primero...'
198     ELSE
199     DO mostrar
200     ENDIF
201     ENDIF
202     RETURN
210 *!*****
211 *!
212 *!     Procedure: SALIR
213 *!
214 *!     Called by: SETUP_REP           (procedure in PRED001.PRG)
215 *!
216 *!*****
210     PROCEDURE salir
211     STORE proyecto TO mproyecto
212     STORE .F. TO reporte
213     DEACTIVATE MENU
214     RETURN
234 *!*****
235 *!
236 *!     Procedure: SETUP_REP
237 *!
238 *!     Called by: PRED001.PRG
239 *!           : DISE001.PRG
240 *!           : SIST004.PRG
241 *!
242 *!     Calls: PROXIMO                 (procedure in PRED001.PRG)
243 *!           : ANTERIOR              (procedure in PRED001.PRG)

```

```

244 *!          : INICIO          (procedure in PRED001.PRG)
245 *!          : FINAL           (procedure in PRED001.PRG)
246 *!          : BUSCAR          (procedure in PRED001.PRG)
247 *!          : EDITAR          (procedure in PRED001.PRG)
248 *!          : INCREMETAR      (procedure in PRED001.PRG)
249 *!          : BORRAR          (procedure in PRED001.PRG)
250 *!          : SALIR           (procedure in PRED001.PRG)
251 *!
252 *!*****
234   PROCEDURE setup_rep
235   DEFINE MENU salesrep COLOR SCHEME 3
236   DEFINE PAD NEXT   OF salesrep PROMPT '\<Prox' AT 12,01
237   DEFINE PAD PREV   OF salesrep PROMPT '\<Ante' AT 12,08
238   DEFINE PAD first  OF salesrep PROMPT '\<Inicio' AT 12,15
239   DEFINE PAD LAST   OF salesrep PROMPT '\<Fin' AT 12,24
240   DEFINE PAD search OF salesrep PROMPT '\<Buscar' AT 12,30
241   DEFINE PAD EDIT   OF salesrep PROMPT '\<Editar' AT 12,39
242   DEFINE PAD APPEND OF salesrep PROMPT 'aña\<Dir' AT 12,48
243   DEFINE PAD DELETE OF salesrep PROMPT 'bor\<Rar' AT 12,57
244   DEFINE PAD QUIT   OF salesrep PROMPT '\<Salir' AT 12,66
245   ON SELECTION PAD NEXT   OF salesrep DO proximo
246   ON SELECTION PAD PREV   OF salesrep DO anterior
247   ON SELECTION PAD first  OF salesrep DO inicio
248   ON SELECTION PAD LAST   OF salesrep DO final
249   ON SELECTION PAD search OF salesrep DO buscar
250   ON SELECTION PAD EDIT   OF salesrep DO editar
251   ON SELECTION PAD APPEND OF salesrep DO incremetar
252   ON SELECTION PAD DELETE OF salesrep DO borrar
253   ON SELECTION PAD QUIT   OF salesrep DO salir
254   RETURN
272 *!*****
273 *!
274 *!   Procedure: MOSTRAR
275 *!
276 *!   Called by: PRED001.PRG
277 *!           : DISE001.PRG
278 *!           : SIST004.PRG
279 *!           : INCREMETAR      (procedure in PRED001.PRG)
280 *!           : BORRAR          (procedure in PRED001.PRG)
281 *!           : EDITAR          (procedure in PRED001.PRG)
282 *!           : FINAL           (procedure in PRED001.PRG)
283 *!           : PROXIMO         (procedure in PRED001.PRG)
284 *!           : ANTERIOR        (procedure in PRED001.PRG)
285 *!           : BUSCAR          (procedure in PRED001.PRG)
286 *!           : INICIO          (procedure in PRED001.PRG)
287 *!
288 *!*****
272   PROCEDURE mostrar
273   @01,24 SAY codigo PICTURE '####'
274   @02,24 SAY proyecto
275   @03,24 SAY beta   PICTURE '#.##'
276   @04,24 SAY fi1    PICTURE '#.##'
277   @05,24 SAY fi2    PICTURE '#.##'
278   @06,24 SAY fi3    PICTURE '#.##'
279   @07,24 SAY h1     PICTURE '### cm'
280   @08,24 SAY D      PICTURE '#### cm'

```

```

281 @09,24 SAY m1 PICTURE '#####.### Kg-m'
282 @10,24 SAY H PICTURE '#### cm'
283 @03,56 SAY pc PICTURE '#####.### Kg'
284 @04,56 SAY bp PICTURE '###.## m'
285 @05,56 SAY ss PICTURE '#### Tn/m²'
286 @06,56 SAY ph1 PICTURE '##.####'
287 @07,56 SAY ee PICTURE '## cm'
288 @08,56 SAY xx PICTURE '99'
289 RETURN
301 *!*****
302 *!
303 *! Procedure: BUSCAR
304 *!
305 *! Called by: SETUP_REP (procedure in PRED001.PRG)
306 *!
307 *! Calls: ESCOJE (procedure in RUTINAS.PRG)
308 *! : MOSTRAR (procedure in PRED001.PRG)
309 *! : STANDBY (procedure in PRED001.PRG)
310 *!
311 *!*****
312 PROCEDURE buscar
313 last_rec=RECNO()
314 STORE 0 TO mcodigo
315 @01,24 GET mcodigo PICTURE '#####'
316 READ
317 IF mcodigo=0
318 DEFINE POPUP localiza FROM 00,45 TO 20,77 PROMPT FIELD proyecto SHADOW COLOR SCHEME 7
319 ON SELECTION POPUP localiza DO escoje WITH PROMPT(), BAR()
320 ACTIVATE POPUP localiza
321 STORE codigo TO mcodigo
322 ENDIF
323 IF mcodigo=0
324 DO mostrar
325 RETURN
326 ENDIF
327 SEEK mcodigo
328 IF .NOT. FOUND()
329 DO standby WITH 'El registro no ha sido encontrado en el archivo'
330 GO last_rec
331 ENDIF
332 DO mostrar
333 RETURN
334 *!*****
335 *!
336 *! Procedure: INICIO
337 *!
338 *! Called by: SETUP_REP (procedure in PRED001.PRG)
339 *!
340 *! Calls: MOSTRAR (procedure in PRED001.PRG)
341 *!
342 *!*****
343 PROCEDURE inicio
344 GO TOP
345 DO mostrar
346 RETURN
347 *!*****

```

```

347 *!
348 *!      Procedure: STANDBY
349 *!
350 *!      Called by: INCREMETAR      (procedure in PRED001.PRG)
351 *!                  : PROXIMO      (procedure in PRED001.PRG)
352 *!                  : ANTERIOR      (procedure in PRED001.PRG)
353 *!                  : BUSCAR        (procedure in PRED001.PRG)
354 *!
355 *!*****
356      PROCEDURE standby
357      PARAMETER mensaje
358      ACTIVATE WINDOW mensajes
359      STORE INT((59 - LEN(mensaje))/2) TO ln
360      @01,ln SAY mensaje
361      READ
362      DEACTIVATE WINDOW mensajes
363      RETURN
364      *: EOF: DISE001.ACT

```

```

1  *:*****
2  *:
3  *:      Program: C:\CIVIL\DISK002.PRG
4  *:
5  *:      System: Anal.y Diseño de Estructuras Aporticadas
6  *:      Author: Rommel R. Alemán T. - Marco P. Silva S.
7  *:      Copyright (c) 1993, Pontificia Universidad Catolica del E.
8  *:      Last modified: 05/26/93      16:10
9  *:
10 *:      Called by: MENU002.PRG
11 *:
12 *:      Memory Files: &NOM01
13 *:                  : &NOM02
14 *:                  : &NOM03
15 *:                  : MRAV.MEM
16 *:                  : MRAC.MEM
17 *:                  : MRE.MEM
18 *:                  : U.MEM
19 *:                  : MATRIZ.MEM
20 *:                  : GIRO.MEM
21 *:
22 *:      Documented 05/28/93 at 12:13      FoxDoc version 2.10
23 *:*****
24      RELEASE ALL
25      IF hl<=0 .OR. D<=0 .OR. ml<=0 .OR. H<=0 .OR. pc<=0 .OR. bp<=0 .OR. ss<=0 .OR. ee<=0
26      WAIT 'ERROR: Datos incompatibles o no existen...' WINDOW
27      RETURN
28      ENDIF
29      ACTIVATE WINDOW reportes
30      CLEAR
31      SELECT predise
32      SET ORDER TO 1
33      @00,13 TO 02,55
34      STORE 'CALCULOS DE DISEÑO' TO titulo
35      STORE ROUND((40 - LEN(titulo)) / 2,0) + 14 TO mlon
36      @01,mlon SAY titulo
37      @04,01 SAY "      Proyecto NUMERO : "
38      @04,24 SAY codigo PICTURE "####"
39      @05,01 SAY "      Nombre del Cliente : "
40      @05,24 SAY proyecto
41      @06,01 SAY "Domicilio de la obra : "
42      @06,24 SAY direccion
43      @07,01 SAY "      Tipo de obra : "
44      @07,24 SAY obra
45      @08,01 SAY "      Ing. Calculista : "
46      @08,24 SAY calculo
47      @09,01 SAY "      Fecha (mm-dd-aa) : "
48      @09,24 SAY fecha
49      STORE SPACE(1) TO res
50      @11,01 SAY 'Esta seguro de querer continuar [S/N] : ' GET res PICTURE '@m S, N'
51      READ
52      STORE (nvx + 1) * np TO memoria
53      IF memoria>40
54      WAIT '['+STR(memoria,2)+'> 40] Insuficiente memoria ...' WINDOW
55      DEACTIVATE WINDOW reportes
56      RETURN

```

```

57  └─ENDIF
58  ┌─IF LASTKEY()=27 .OR. res='N'
59  │   DEACTIVATE WINDOW reportes
60  └─RETURN
61  └─ENDIF
62  ACTIVATE WINDOW impresor
63  CLEAR
64  @01.01 SAY 'GENERANDO PLANILLA DE RIGIDEZES...'
65  STORE codigo TO numero
66  STORE STR(numero,5) TO mnum
67  STORE 'nvx'+ALLTRIM(mnum) TO nom01
68  STORE 'nvy'+ALLTRIM(mnum) TO nom02
69  STORE 'np'+ALLTRIM(mnum) TO nom03
70  RESTORE FROM &nom01 ADDITIVE
71  RESTORE FROM &nom02 ADDITIVE
72  RESTORE FROM &nom03 ADDITIVE
73  STORK nvx TO mnvx
74  STORK nvy TO mnvy
75  STORK np TO mnp
76  STORK VAL(xx) TO portico
77  DIMENSION x1(mnp,mnvx), x2(mnp,mnvx), x3(mnp,mnvx+1), x4(mnp,mnvx+1)
78  FOR K=1 TO mnp
79  │   STORE 0 TO nivel
80  │   FOR k1=mnp TO K STEP -1
81  │   │   STORE nivel+hp(k1) TO nivel
82  │   └─ENDFOR
83  │   FOR i=1 TO mnvx
84  │   │   STORE 0 TO kv
85  │   │   STORE (bv/10*(D/10)^3)/(12*X(i)) TO kv
86  │   │   STORE kv TO x1(K,i)
87  │   └─ENDFOR
88  │   FOR i=1 TO mnvx
89  │   │   STORE ((bv/10 * (D/10)^3)/(12*X(i)))/2 TO av
90  │   │   STORE av TO x2(K,i)
91  │   └─ENDFOR
92  │   FOR i=1 TO mnvx+1
93  │   │   STORE (bc/10 * (H/10)^3)/(12 * hp(K)) TO kc
94  │   │   STORE kc TO x3(K,i)
95  │   └─ENDFOR
96  │   FOR i=1 TO mnvx+1
97  │   │   STORE ((bc/10 * (H/10)^3)/(12 * hp(K)))/2 TO ac
98  │   │   STORE ac TO x4(K,i)
99  │   └─ENDFOR
100 └─ENDFOR
101 SAVE TO mnav ALL LIKE x2
102 SAVE TO mrac ALL LIKE x4
103 SELECT mat07
104 SET ORDER TO 1
105 DELETE ALL
106 PACK
107 FOR i=1 TO mnp
108 │   APPEND BLANK
109 │   REPLA y WITH i
110 │   STORE 1 TO K
111 │   FOR j=1 TO mnvx
112 │   │   STORE 'X'+ALLTRIM(STR(K,2)) TO macro1

```

```

113 | STORE K + 1 TO K
114 | STORE 'X'+ALLTRIM(STR(K,2)) TO macro2
115 | REPLA &macro1 WITH ROUND(x1(i,j),3)
116 | REPLA &macro2 WITH ROUND(x1(i,j),3)
117 | STORE K + 1 TO K
118 | _ENDFOR
119 | _ENDFOR
120 | SELECT mat08
121 | SET ORDER TO 1
122 | DELETE ALL
123 | PACK
124 | STORE 1 TO K
125 | FOR i=1 TO mnp
126 | | APPEND BLANK
127 | | REPLA y WITH K
128 | | FOR j=1 TO mnvx+1
129 | | | STORE 'X'+ALLTRIM(STR(j,2)) TO MACRO
130 | | | REPLA &macro WITH ROUND(x3(i,j),3)
131 | | _ENDFOR
132 | | IF i#mnp
133 | | | APPEND BLANK
134 | | | STORE K + 1 TO K
135 | | | REPLA y WITH K
136 | | | FOR j=1 TO mnvx+1
137 | | | | STORE 'X'+ALLTRIM(STR(j,2)) TO MACRO
138 | | | | REPLA &macro WITH ROUND(x3(i,j),3)
139 | | | _ENDFOR
140 | | | STORE K + 1 TO K
141 | | _ENDIF
142 | _ENDFOR
143 | SELECT mat09
144 | SET ORDER TO 1
145 | DELETE ALL
146 | PACK
147 | FOR i=1 TO mnp
148 | | APPEND BLANK
149 | | REPLA y WITH i
150 | | STORE 1 TO K
151 | | FOR j=1 TO mnvx
152 | | | STORE 'X'+ALLTRIM(STR(K,2)) TO macro1
153 | | | STORE K + 1 TO K
154 | | | STORE 'X'+ALLTRIM(STR(K,2)) TO macro2
155 | | | REPLA &macro1 WITH ROUND(x2(i,j),3)
156 | | | REPLA &macro2 WITH ROUND(x2(i,j),3)
157 | | | STORE K + 1 TO K
158 | | _ENDFOR
159 | _ENDFOR
160 | SELECT mat10
161 | SET ORDER TO 1
162 | DELETE ALL
163 | PACK
164 | STORE 1 TO K
165 | FOR i=1 TO mnp
166 | | APPEND BLANK
167 | | REPLA y WITH K
168 | | FOR j=1 TO mnvx+1

```

```

169      STORE 'X'+ALLTRIM(STR(j,2)) TO MACRO
170      REPLA &macro WITH ROUND(x4(i,j),3)
171      ENDFOR
172      IF i#mnp
173          APPEND BLANK
174          STORE K + 1 TO K
175          REPLA y WITH i
176          FOR j=1 TO mnvx+1
177              STORE 'X'+ALLTRIM(STR(j,2)) TO MACRO
178              REPLA &macro WITH ROUND(x4(i,j),3)
179          ENDFOR
180          STORE K + 1 TO K
181      ENDIF
182  ENDFOR
183  CLEAR
184  @01.01 SAY 'GENERANDO RESUMEN DE RIGIDESES...'
185  RELEASE ALL
186  SELECT predise
187  SET ORDER TO 1
188  STORE codigo TO numero
189  STORE STR(numero,5) TO mnum
190  STORE 'nvx'+ALLTRIM(mnum) TO nom01
191  STORE 'nvy'+ALLTRIM(mnum) TO nom02
192  STORE 'np'+ALLTRIM(mnum) TO nom03
193  RSTORE FROM &nom01 ADDITIVE
194  RSTORE FROM &nom02 ADDITIVE
195  RSTORE FROM &nom03 ADDITIVE
196  STORE nvx TO mnvx
197  STORE nvy TO mnvy
198  STORE np TO mnp
199  STORE VAL(xx) TO portico
200  DIMENSION x1(mnp,mnvx+1), x2(mnp*2,mnvx+1), x3(mnp,mnvx+1)
201  DIMENSION x11(mnp*2,mnvx*2), x22(mnp*2+1,(mnvx+1)*2)
202  STORE 0 TO x1,x2,x3,x11,x22
203  FOR K=1 TO mnp
204      FOR i=1 TO mnvx
205          STORE 0 TO kv
206          STORE (bv/10*(D/10)^3)/(12*X(i)) TO kv
207          STORE kv TO x1(K,i)
208      ENDFOR
209      FOR i=1 TO mnvx+1
210          STORE (bc/10 * (H/10)^3)/(12 * hp(K)) TO kc
211          STORE kc TO x2(K,i)
212      ENDFOR
213  ENDFOR
214  FOR i=1 TO mnp
215      STORE 1 TO K
216      FOR j=1 TO mnvx
217          STORE x1(i,j) TO x11(i,K)
218          STORE K+1 TO K
219          STORE x1(i,j) TO x11(i,K)
220          STORE K+1 TO K
221      ENDFOR
222  ENDFOR
223  STORE 0 TO x1
224  FOR i=1 TO mnp

```

```

225     STORE 1 TO K
226     FOR j=1 TO mnvx*2
227         IF j=1 .OR. j=mnvx*2
228             STORE x11(i,j) TO x1(i,K)
229             STORE K+1 TO K
230         ELSE
231             IF MOD(j,2) = 0
232                 STORE x11(i,j)+x11(i,j+1) TO x1(i,K)
233                 STORE K+1 TO K
234             ENDIF
235         ENDIF
236     ENDFOR
237 ENDFOR
238     STORE 1 TO K
239     FOR i=1 TO mnp
240         FOR j=1 TO mnvx+1
241             STORE x2(i,j) TO x22(K,j)
242         ENDFOR
243         STORE K+1 TO K
244         FOR j=1 TO mnvx+1
245             STORE x2(i,j) TO x22(K,j)
246         ENDFOR
247         STORE K+1 TO K
248     ENDFOR
249     STORE 0 TO x2
250     STORE 1 TO K
251     FOR i=1 TO mnp*2-1
252         FOR j=1 TO mnvx+1
253             IF i=1
254                 STORE x22(i,j) TO x2(K,j)
255             ELSE
256                 IF MOD(i,2) = 0
257                     STORE x22(i,j)+x22(i+1,j) TO x2(K,j)
258                 ENDIF
259             ENDIF
260         ENDFOR
261         IF i=1
262             STORE K + 1 TO K
263         ELSE
264             IF MOD(i,2)=0
265                 STORE K+1 TO K
266             ENDIF
267         ENDIF
268     ENDFOR
269     FOR i=1 TO mnp
270         STORE 0 TO nivel
271         FOR K=mnp TO i STEP -1
272             STORE nivel+hp(K) TO nivel
273         ENDFOR
274         FOR j=1 TO mnvx+1
275             STORE x1(i,j)+x2(i,j) TO x3(i,j)
276         ENDFOR
277     ENDFOR
278     SAVE TO mrk ALL LIKE x3
279     CLEAR
280     @01.01 SAY 'GENERANDO PLANILLA DE MOMENTOS DE EMPOTRAMIENTO PERFECTO...'

```

```

281  RELEASE ALL
282  SELECT predise
283  SET ORDER TO 1
284  STORE codigo TO numero
285  STORE STR(numero,5) TO mnum
286  STORE 'nvx'+ALLTRIM(mnum) TO nom01
287  STORE 'nvy'+ALLTRIM(mnum) TO nom02
288  STORE 'np'+ALLTRIM(mnum) TO nom03
289  RESTORE FROM &nom01 ADDITIVE
290  RESTORE FROM &nom02 ADDITIVE
291  RESTORE FROM &nom03 ADDITIVE
292  STORE nvx TO mnvx
293  STORE nvy TO mnvy
294  STORE np TO mnp
295  STORE VAL(xx) TO portico
296  DIMENSION x1(mnvy+1,mnvx+1)
297  DIMENSION x11(mnp*2,mnvx*2), x22(mnp*2,(mnvx+1)*2)
298  STORE 0 TO x1,x2,x11,x22
299  SELECT mat03
300  SET ORDER TO 1
301  GO TOP
302  STORE 0 TO i,j
303  FOR i=1 TO RECCOUNT()
304  |   FOR j=1 TO mnvx
305  | |   STORE 'X'+ALLTRIM(STR(j,2)) TO MACRO
306  | |   STORE (&macro*X(j)^2)/12 TO x1(i,j)
307  | |   ENDFOR
308  |   SKIP
309  |   ENDFOR
310  STORE portico TO i
311  STORE 1 TO K
312  FOR j=1 TO mnvx
313  |   STORE x1(i,j) TO x22(i,K)
314  |   STORE K+1 TO K
315  |   STORE -x1(i,j) TO x22(i,K)
316  |   STORE K+1 TO K
317  |   ENDFOR
318  FOR i=1 TO mnp
319  |   FOR j=1 TO mnvx*2
320  | |   STORE x22(portico,j) TO x11(i,j)
321  | |   ENDFOR
322  |   ENDFOR
323  SELECT mat15
324  SET ORDER TO 1
325  ZAP
326  FOR i=1 TO mnp
327  |   APPEND BLANK
328  |   REPLA y WITH i
329  |   STORE 0 TO nivel
330  |   FOR K=mnp TO i STEP -1
331  | |   STORE nivel+hp(K) TO nivel
332  | |   ENDFOR
333  |   FOR j=1 TO mnvx*2
334  | |   STORE 'X'+ALLTRIM(STR(j,2)) TO MACRO
335  | |   REPLA &macro WITH x11(i,j)
336  | |   ENDFOR

```

```

337  ┌-ENDFOR
338      CLEAR
339      @01,01 SAY 'GENERANDO RESUMEN DE MOMENTOS DE EMPOTRAMIENTO...'
340      RELEASE ALL
341      SELECT predise
342      SET ORDER TO 1
343      STORE codigo TO numero
344      STORE STR(numero,5) TO mnum
345      STORE 'nvx'+ALLTRIM(mnum) TO nom01
346      STORE 'nvy'+ALLTRIM(mnum) TO nom02
347      STORE 'np'+ALLTRIM(mnum) TO nom03
348      RESTORE FROM &nom01 ADDITIVE
349      RESTORE FROM &nom02 ADDITIVE
350      RESTORE FROM &nom03 ADDITIVE
351      STORE nvx TO mnvx
352      STORE nvy TO mnvy
353      STORE np TO mnp
354      STORE VAL(xx) TO portico
355      DIMENSION x1(mnvy+1,mnvx*2)
356      DIMENSION x11(mnp*2,mnvx*2), x22(mnp*2,(mnvx+1)*2)
357      STORE 0 TO x1,x11,x22
358      SELECT mat03
359      SET ORDER TO 1
360      GO TOP
361      STORE 0 TO i,j
362      ┌-FOR i=1 TO RECCOUNT()
363          ┌-FOR j=1 TO mnvx
364              STORE 'X'+ALLTRIM(STR(j,2)) TO MACRO
365              STORE (&macro*X(j)^2)/12 TO x1(i,j)
366          ┌-ENDFOR
367          SKIP
368      ┌-ENDFOR
369      STORE portico TO i
370      STORE 1 TO K
371      ┌-FOR j=1 TO mnvx
372          STORE x1(i,j) TO x22(i,K)
373          STORE K+1 TO K
374          STORE -x1(i,j) TO x22(i,K)
375          STORE K+1 TO K
376      ┌-ENDFOR
377      ┌-FOR i=1 TO mnp
378          ┌-FOR j=1 TO mnvx*2
379              STORE x22(portico,j) TO x11(i,j)
380          ┌-ENDFOR
381      ┌-ENDFOR
382      RELEASE x1
383      DIMENSION x1(mnp,mnvx*2)
384      STORE 0 TO x1
385      ┌-FOR i=1 TO mnp
386          STORE 1 TO K
387          ┌-FOR j=1 TO mnvx*2
388              ┌-IF j=1 .OR. j=mnvx*2
389                  STORE x11(i,j) TO x1(i,K)
390                  STORE K+1 TO K
391              ┌-ELSE
392                  ┌-IF MOD(j,2) = 0

```

```

393         STORE x11(i,j)+x11(i,j+1) TO x1(i,K)
394         STORE K+1 TO K
395     _ENDIF
396 _ENDIF
397 _ENDFOR
398 _ENDFOR
399     DIMENSION cl(mnp*(mnvx+1))
400     STORE 1 TO K
401     FOR i=mnp TO 1 STEP -1
402     _FOR j=1 TO mnvx+1
403         STORE ((-1) * x1(i,j)) TO cl(K)
404         STORE K + 1 TO K
405     _ENDFOR
406 _ENDFOR
407     SAVE TO u ALL LIKE cl
408     CLEAR
409     @01.01 SAY 'GENERANDO MATRIZ DE RIGIDES...'
410     RELEASE ALL
411     SELECT predise
412     SET ORDER TO 1
413     STORE codigo TO numero
414     STORE STR(numero,5) TO mnum
415     STORE 'nvx'+ALLTRIM(mnum) TO nom01
416     STORE 'nvy'+ALLTRIM(mnum) TO nom02
417     STORE 'np'+ALLTRIM(mnum) TO nom03
418     RESTORE FROM &nom01 ADDITIVE
419     RESTORE FROM &nom02 ADDITIVE
420     RESTORE FROM &nom03 ADDITIVE
421     STORE nvx TO mnvx
422     STORE nvy TO mnvy
423     STORE np TO mnp
424     STORE VAL(xx) TO portico
425     CLEAR
426     @01.01 SAY 'GENERANDO MATRIZ DE RIGIDEZ ...'
427     STORE 0 TO nx
428     STORE (mnvx+1)*mnp TO nx
429     DIMENSION mr(nx,nx)
430     STORE 0 TO mr
431     RESTORE FROM mrk ADDITIVE
432     STORE 1 TO K
433     FOR i=mnp TO 1 STEP -1
434     _FOR j=1 TO mnvx+1
435         STORE x3(i,j) TO mr(K,K)
436         STORE K + 1 TO K
437     _ENDFOR
438 _ENDFOR
439     RELEASE x3
440     RESTORE FROM mrav ADDITIVE
441     STORE 1 TO K, 1
442     FOR i=mnp TO 1 STEP -1
443     _FOR j=1 TO mnvx
444         STORE x2(i,j) TO mr(K,K+1)
445         STORE x2(i,j) TO mr(K+1,K)
446         STORE K + 1 TO K
447     _ENDFOR
448     STORE K+1 TO K

```

```

449  └─ENDFOR
450  RELEASE x2
451  RESTORE FROM mac ADDITIVE
452  STORE mnx+2 TO K
453  STORE 1 TO 1
454  STORE 1 TO bandera
455  └─FOR i=mp TO 1 STEP -1
456  └─FOR j=1 TO mnx+1
457  STORE x4(i,j) TO mr(K,1)
458  └─IF K>nx
459  STORE 0 TO bandera
460  └─V=EXIT
461  └─ENDIF
462  STORE K + 1 TO K
463  STORE 1 + 1 TO 1
464  └─ENDFOR
465  └─IF bandera=0
466  └─V=EXIT
467  └─ENDIF
468  └─ENDFOR
469  STORE 1 TO K
470  STORE mnx+2 TO 1
471  STORE 1 TO bandera
472  └─FOR i=1 TO mnp
473  └─FOR j=1 TO mnx+1
474  STORE x4(i,j) TO mr(K,1)
475  └─IF l>=nx
476  STORE 0 TO bandera
477  └─V=EXIT
478  └─ENDIF
479  STORE K + 1 TO K
480  STORE 1 + 1 TO 1
481  └─ENDFOR
482  └─IF bandera=0
483  └─V=EXIT
484  └─ENDIF
485  └─ENDFOR
486  RELEASE x4
487  SELECT mat05
488  SET ORDER TO 1
489  DELETE ALL
490  PACK
491  STORE '' TO sumac
492  └─FOR i=1 TO nx
493  APPEND BLANK
494  REPLA y WITH i
495  └─FOR j=1 TO nx
496  STORE 'X'+ALLTRIM(STR(j,2)) TO mac
497  REPLA mac WITH mr(i,j)
498  └─ENDFOR
499  └─ENDFOR
500  DIMENSION X(nx,nx), y(nx,nx)
501  STORE 0 TO X,y
502  GO TOP
503  └─FOR i=1 TO nx
504  └─FOR j=1 TO nx

```

```

505      STORE 'X'+ALTRIM(STR(j,2)) TO MACRO
506      STORE &macro TO X(j,i), y(j,i)
507      └─ENDFOR
508      SKIP
509      └─ENDFOR
510      RESTORE FROM u ADDITIVE
511      DIMENSION g(n)
512      STORE 0 TO g(n)
513      CLEAR
514      @01.01 SAY 'OBTENIENDO MATRIZ DE RIGIDEZ INVERSA ...'
515      FOR j=1 TO nx
516      └─FOR i=1 TO nx
517      └─└─STORE 0 TO y(j,i)
518      └─└─ENDFOR
519      └─STORE 1 TO y(j,j)
520      └─ENDFOR
521      FOR j=1 TO nx
522      └─FOR i=j TO nx
523      └─└─IF X(i,j)≠0
524      └─└─└─EXIT
525      └─└─ENDIF
526      └─ENDFOR
527      FOR k=1 TO nx
528      └─STORE X(j,k) TO s
529      └─STORE X(i,k) TO X(j,k)
530      └─STORE s TO X(i,k)
531      └─STORE y(j,k) TO s
532      └─STORE y(i,k) TO y(j,k)
533      └─STORE s TO y(i,k)
534      └─ENDFOR
535      STORE 1/X(j,i) TO t
536      FOR k=1 TO nx
537      └─STORE t * X(j,k) TO X(j,k)
538      └─STORE t * y(j,k) TO y(j,k)
539      └─ENDFOR
540      FOR i=1 TO nx
541      └─IF i=j
542      └─└─LOOP
543      └─ENDIF
544      └─STORE (-1) * X(i,j) TO t
545      └─FOR k=1 TO nx
546      └─└─STORE X(i,k) + t * X(j,k) TO X(i,k)
547      └─└─STORE y(i,k) + t * y(j,k) TO y(i,k)
548      └─└─ENDFOR
549      └─ENDFOR
550      └─ENDFOR
551      SELECT mat06
552      SET ORDER TO 1
553      ZAP
554      FOR i=1 TO nx
555      └─APPEND BLANK
556      └─REPLA y WITH i
557      └─FOR j=1 TO nx
558      └─└─STORE 'X'+ALTRIM(STR(j,2)) TO MACRO
559      └─└─REPLA &macro WITH ROUND(y(i,j),3)
560      └─└─ENDFOR

```

```

561  ENDFOR
562  GO TOP
563  CLEAR
564  @01.01 SAY 'OBTENIENDO LOS GIROS DE NUDO ...'
565  FOR i=1 TO nx
566      FOR j=1 TO nx
567          STORE girog(i) + (y(i,j) * c1(j)) TO girog(i)
568      ENDFOR
569      STORE ROUND(girog(i),3) TO girog(i)
570  ENDFOR
571  SAVE TO matriz ALL LIKE nx
572  SAVE TO giro ALL LIKE girog
573  RELEASE X,Y
574  SELECT predise
575  SET ORDER TO 1
576  STORE codigo TO numero
577  STORE STR(numero,5) TO anua
578  STORE 'nvx' + ALLTRIM(anua) TO nom01
579  STORE 'nvy' + ALLTRIM(anua) TO nom02
580  STORE 'np' + ALLTRIM(anua) TO nom03
581  RESTORE FROM anoma1 ADDITIVE
582  RESTORE FROM anoma2 ADDITIVE
583  RESTORE FROM anoma3 ADDITIVE
584  RESTORE FROM matriz ADDITIVE
585  STORE nvx TO nvx
586  STORE nvy TO nvy
587  STORE np TO np
588  CLEAR
589  @01.01 SAY 'GENERANDO GIROS DE NUDOS FINALES...'
590  RELEASE ALL
591  SELECT predise
592  SET ORDER TO 1
593  STORE codigo TO numero
594  STORE STR(numero,5) TO anua
595  STORE 'nvx' + ALLTRIM(anua) TO nom01
596  STORE 'nvy' + ALLTRIM(anua) TO nom02
597  STORE 'np' + ALLTRIM(anua) TO nom03
598  RESTORE FROM anoma1 ADDITIVE
599  RESTORE FROM anoma2 ADDITIVE
600  RESTORE FROM anoma3 ADDITIVE
601  STORE nvx TO nvx
602  STORE nvy TO nvy
603  STORE np TO np
604  STORE VAL(x) TO potico
605  RESTORE FROM giro ADDITIVE
606  DIMENSION x1(np,nvx+1)
607  STORE 0 TO x1
608  STORE 1 TO K
609  FOR i=1 TO np
610      FOR j=1 TO nvx+1
611          STORE girog(K) TO x1(i,j)
612      STORE K + 1 TO K
613  ENDFOR
614  ENDFOR
615  SELECT mat11
616  SET ORDER TO 1

```

```

617 DELETE ALL
618 PACK
619 STORE 1 TO 1
620 FOR i=mnv TO 1 STEP-1
621     APPEND BLANK
622     REPLA y WITH 1
623     STORE 1 TO K
624     STORE 'X'+ALLTRIM(STR(K,2)) TO macro1
625     REPLA &macro1 WITH ROUND(x1(i,K),3)
626     STORE K+1 TO K
627     FOR j=2 TO mnvx
628         STORE 'X'+ALLTRIM(STR(K,2)) TO macro1
629         STORE K + 1 TO K
630         STORE 'X'+ALLTRIM(STR(K,2)) TO macro2
631         REPLA &macro1 WITH ROUND(x1(i,j),3)
632         REPLA &macro2 WITH ROUND(x1(i,j),3)
633         STORE K + 1 TO K
634     ENDFOR
635     STORE 'X'+ALLTRIM(STR(K,2)) TO macro1
636     REPLA &macro1 WITH ROUND(x1(i,mnvx+1),3)
637     STORE i+1 TO i
638 ENDFOR
639 SELECT mat12
640 SET ORDER TO 1
641 DELETE ALL
642 PACK
643 SELECT mat11
644 SET ORDER TO 1
645 GO TOP
646 FOR i=1 TO mnv
647     SELECT mat12
648     SET ORDER TO 1
649     APPEND BLANK
650     REPLA y WITH i
651     FOR j=1 TO mnvx*2 STEP +2
652         SELECT mat11
653         SET ORDER TO 1
654         STORE 'X'+ALLTRIM(STR(j,2)) TO macro1
655         STORE 'X'+ALLTRIM(STR(j+1,2)) TO macro2
656         STORE &macro1 TO aux1
657         STORE &macro2 TO aux2
658         SELECT mat12
659         SET ORDER TO 1
660         REPLA &macro1 WITH aux2
661         REPLA &macro2 WITH aux1
662     ENDFOR
663     SELECT mat11
664     SET ORDER TO 1
665     SKIP
666 ENDFOR
667 SELECT mat13
668 SET ORDER TO 1
669 ZAP
670 STORE mnv*2-1 TO 1
671 STORE 0 TO bandera
672 FOR i=1 TO mnv

```

```

673   FOR K=1 TO 2
674   |   IF i=mnv .AND. K=2
675   |   |   STORE 1 TO bandera
676   |   |   v-----EXIT
677   |   |   ELSE
678   |   |   |   APPEND BLANK
679   |   |   |   REPLA y WITH 1
680   |   |   |   FOR j=1 TO mnvx+1
681   |   |   |   |   STORE 'X'+ALLTRIM(STR(j,2)) TO MACRO
682   |   |   |   |   REPLA &macro WITH x1(i,j)
683   |   |   |   ENDFOR
684   |   |   |   STORE 1 - 1 TO 1
685   |   |   ENDIF
686   |   ENDFOR
687   |   IF bandera=1
688   |   |   v-----EXIT
689   |   |   ENDIF
690   |   ENDFOR
691   SELECT mat13
692   SET ORDER TO 1
693   GO TOP
694   SELECT mat14
695   SET ORDER TO 1
696   DELETE ALL
697   PACK
698   FOR i=1 TO mnv*2-1
699   |   APPEND BLANK
700   |   REPLA y WITH i
701   |   FOR j=1 TO mnvx+1
702   |   |   SELECT mat13
703   |   |   SET ORDER TO 1
704   |   |   STORE 'X'+ALLTRIM(STR(j,2)) TO MACRO
705   |   |   STORE &macro TO aux
706   |   |   SELECT mat14
707   |   |   SET ORDER TO 1
708   |   |   REPLA &macro WITH aux
709   |   ENDFOR
710   |   SELECT mat13
711   |   SET ORDER TO 1
712   |   SKIP
713   |   SELECT mat14
714   |   SET ORDER TO 1
715   |   ENDFOR
716   SELECT mat14
717   SET ORDER TO
718   GO TOP
719   FOR i=1 TO mnv*2-2 STEP +2
720   |   GOTO i
721   |   STORE y TO aux
722   |   SKIP
723   |   REPLA y WITH aux
724   |   GOTO i
725   |   REPLA y WITH aux+1
726   |   ENDFOR
727   SET ORDER TO 1
728   CLEAR

```

```

729 @01.01 SAY 'GENERANDO MOMENTOS FINALES DE NUDO...'
730 RELEASE ALL
731 SELECT predise
732 SET ORDER TO 1
733 STORE codigo TO numero
734 STORE STR(numero,5) TO mnum
735 STORE 'nvx'+ALLTRIM(mnum) TO nom01
736 STORE 'nvy'+ALLTRIM(mnum) TO nom02
737 STORE 'np'+ALLTRIM(mnum) TO nom03
738 RESTORE FROM &nom01 ADDITIVE
739 RESTORE FROM &nom02 ADDITIVE
740 RESTORE FROM &nom03 ADDITIVE
741 RESTORE FROM matriz ADDITIVE
742 RESTORE FROM giro ADDITIVE
743 STORE nvx TO mnvx
744 STORE nvy TO mnvy
745 STORE np TO mnp
746 STORE proyecto TO mproyecto
747 STORE direccion TO mdireccion
748 STORE calculo TO mcalculo
749 STORE fecha TO mfecha
750 DIMENSION matril(mnp,mnvx*2), matri2(mnp*2,mnvx+1)
751 STORE 0 TO matril, matri2
752 SELECT mat07
753 SET ORDER TO 1
754 GO TOP
755 SELECT mat09
756 SET ORDER TO 1
757 GO TOP
758 SELECT mat11
759 SET ORDER TO 1
760 GO TOP
761 SELECT mat12
762 SET ORDER TO 1
763 GO TOP
764 SELECT mat15
765 SET ORDER TO 1
766 GO TOP
767 SELECT mat16
768 SET ORDER TO 1
769 ZAP
770 FOR i=1 TO mnp
771     APPEND BLANK
772     REPLA y WITH i
773     FOR j=1 TO mnvx*2
774         STORE 'X'+ALLTRIM(STR(j,2)) TO MACRO
775         STORE 0 TO mkv,mav,mtkv,mtav,mnf,mfv
776         SELECT mat07
777         SET ORDER TO 1
778         STORE &macro TO mkv
779         SELECT mat09
780         SET ORDER TO 1
781         STORE &macro TO mav
782         SELECT mat11
783         SET ORDER TO 1
784         STORE &macro TO mtkv

```

```

785      SELECT mat12
786      SET ORDER TO 1
787      STORE &macro TO mtav
788      SELECT mat15
789      SET ORDER TO 1
790      STORE &macro TO mmf
791      STORE mmf + (mkv * mtkv) + (mtav * mav) TO mfv
792      SELECT mat16
793      SET ORDER TO 1
794      STORE ROUND(mfv,3) TO matril(i,j)
795      REPLA &macro WITH matril(i,j)
796  _ENDFOR
797      SELECT mat07
798      SET ORDER TO 1
799      SKIP
800      SELECT mat09
801      SET ORDER TO 1
802      SKIP
803      SELECT mat11
804      SET ORDER TO 1
805      SKIP
806      SELECT mat12
807      SET ORDER TO 1
808      SKIP
809      SELECT mat15
810      SET ORDER TO 1
811      SKIP
812      SELECT mat16
813      SET ORDER TO 1
814  _ENDFOR
815      SELECT mat08
816      SET ORDER TO 1
817      GO TOP
818      SELECT mat10
819      SET ORDER TO 1
820      GO TOP
821      SELECT mat13
822      SET ORDER TO 1
823      GO TOP
824      SELECT mat14
825      SET ORDER TO 1
826      GO TOP
827      SELECT mat17
828      SET ORDER TO 1
829      ZAP
830      STORE 0 TO matril2
831  _FOR i=1 TO mnp*2-1
832      APPEND BLANK
833      REPLA y WITH i
834      _FOR j=1 TO mnvx+1
835          STORE 'X'+ALLTRIM(STR(j,2)) TO MACRO
836          STORE 0 TO mkc,mac,mtkc,mtac,mfc
837          SELECT mat08
838          SET ORDER TO 1
839          STORE &macro TO mkc
840          SELECT mat10

```

```

841      SET ORDER TO 1
842      STORE &macro TO mac
843      SELECT mat13
844      SET ORDER TO 1
845      STORE &macro TO mtkc
846      SELECT mat14
847      SET ORDER TO 1
848      STORE &macro TO mtac
849      SELECT mat17
850      SET ORDER TO 1
851      IF i = mnp*2-1
852          STORE (mkc * mtkc) TO mfc
853          STORE ROUND(mfc,3) TO matri2(i,j)
854          REPLA &macro WITH matri2(i,j)
855      ELSE
856          STORE (mkc * mtkc) + (mtac * mac) TO mfc
857          STORE ROUND(mfc,3) TO matri2(i,j)
858          REPLA &macro WITH matri2(i,j)
859      ENDIF
860  ENDFOR
861      SELECT mat08
862      SET ORDER TO 1
863      SKIP
864      SELECT mat10
865      SET ORDER TO 1
866      SKIP
867      SELECT mat13
868      SET ORDER TO 1
869      SKIP
870      SELECT mat14
871      SET ORDER TO 1
872      SKIP
873      SELECT mat17
874      SET ORDER TO 1
875  ENDFOR
876      SELECT mat08
877      SET ORDER TO 1
878      GO BOTTOM
879      SELECT mat10
880      SET ORDER TO 1
881      GO BOTTOM
882      SELECT mat13
883      SET ORDER TO 1
884      GO BOTTOM
885      SELECT mat14
886      SET ORDER TO 1
887      GO BOTTOM
888      SELECT mat17
889      SET ORDER TO 1
890      APPEND BLANK
891      REPLA y WITH i
892      FOR j=1 TO mnvx+1
893          STORE 'X'+ALLTRIM(STR(j,2)) TO MACRO
894          STORE 0 TO mac,mtac,mfc
895          SELECT mat10
896          SET ORDER TO 1

```

```

897     STORE &macro TO mac
898     SELECT mat14
899     SET ORDER TO 1
900     STORE &macro TO mtac
901     SELECT mat17
902     SET ORDER TO 1
903     STORE (mtac * mac) TO mfc
904     STORE ROUND(mfc,3) TO matri2(i,j)
905     REPLA &macro WITH matri2(i,j)
906   _ENDFOR
907   CLAR
908   @01.01 SAY 'GENERANDO Mborde, Mmax, V Total, Flecha...'
909   RELEASE ALL
910   SELECT predise
911   SET ORDER TO 1
912   STORE codigo TO numero
913   STORE STR(numero,5) TO mnum
914   STORE 'nvx'+ALLTRIM(mnum) TO nom01
915   STORE 'nvy'+ALLTRIM(mnum) TO nom02
916   STORE 'np'+ALLTRIM(mnum) TO nom03
917   RESTORE FROM &nom01 ADDITIVE
918   RESTORE FROM &nom02 ADDITIVE
919   RESTORE FROM &nom03 ADDITIVE
920   RESTORE FROM matriz ADDITIVE
921   STORE nvx      TO mnvx
922   STORE nvy      TO mnvy
923   STORE np       TO mnp
924   STORE fc       TO mfc
925   STORE fy       TO mfy
926   STORE fil      TO mfil
927   STORE 14/mfy   TO mp
928   STORE D/100    TO mdv
929   STORE H/200    TO mdc
930   STORE mdc+mdv  TO md
931   STORE proyecto TO mproyecto
932   STORE direccion TO mdireccion
933   STORE calculo  TO mcalculo
934   STORE fecha    TO mfecha
935   STORE xx       TO mxx
936   STORE VAL(mxx) TO eje
937   DIMENSION q(mnvx), m(mnp,mnvx*2), AS(mnp,mnvx*3)
938   DIMENSION vo(mnvx*2), vh(mnvx*2), vt(mnvx*2)
939   DIMENSION mb(mnvx*2), mx(mnvx), mmax(mnvx)
940   STORE 0 TO q,m,AS,vo,vh,vt,mb,mx,mmax
941   SELECT mat03
942   SET ORDER TO 1
943   SEEK eje
944   FOR i=1 TO mnvx
945     STORE 'X'+ALLTRIM(STR(i,2)) TO MACRO
946     STORE &macro TO q(i)
947   _ENDFOR
948   SELECT mat16
949   SET ORDER TO 1
950   GO TOP
951   FOR i=1 TO mnp
952     FOR j=1 TO mnvx*2

```

```

993      STORE 'X'+ALLTRIM(STR(j,2)) TO MACRO
994      STORE &macro TO m(i,j)
995      ENDFOR
996      SKIP
997      ENDFOR
998      STORE m(i)*m(fy*md*(1 - 0.59 * mp * (mfy / mfc)) TO das
999      FOR i=1 TO mp
1000      STORE 1 TO K
1001      FOR j=1 TO mnvx
1002      STORE ROUND((q(j)*K(j))/2.3) TO voi, vo(K)
1003      STORE K+1 TO K
1004      STORE ROUND(voi * (-1),3) TO vod, vo(K)
1005      STORE K+1 TO K
1006      ENDFOR
1007      STORE 1 TO K
1008      FOR j=1 TO mnvx
1009      STORE ROUND((m(i,K)+m(i,K+1))/K(j),3) TO vhi, vh(K)
1010      STORE K+1 TO K
1011      STORE ROUND(vhi * (-1),3) TO vhd, vh(K)
1012      STORE K+1 TO K
1013      ENDFOR
1014      STORE 1 TO K
1015      FOR j=1 TO mnvx
1016      STORE ROUND((vo(K)+vh(K),3) TO vti, vt(K)
1017      STORE ROUND((vo(K)-vh(K),3) TO vtd, vt(K+1)
1018      STORE K+2 TO K
1019      ENDFOR
1020      STORE 1 TO K
1021      FOR j=1 TO mnvx
1022      STORE ROUND( (vt(K) * md - (q(j) * md^2 / 2) - m(i,K)), 3) TO mbi, mb(K)
1023      STORE K+1 TO K
1024      STORE ROUND( (-vt(K) * md + (q(j) * md^2 / 2) + m(i,K)), 3) TO mbd, mb(K)
1025      STORE K+1 TO K
1026      ENDFOR
1027      STORE 1 TO K
1028      FOR j=1 TO mnvx
1029      STORE ROUND( vt(K) / q(j), 3) TO mx, mx(j)
1030      STORE K+2 TO K
1031      ENDFOR
1032      STORE 1 TO K
1033      FOR j=1 TO pos
1034      STORE ROUND( vt(K)^2 / (2*q(j)) - m(i,K), 3) TO mmax, mmax(j)
1035      STORE K+2 TO K
1036      ENDFOR
1037      STORE 1 TO K, 1
1038      FOR j=1 TO mnvx
1039      STORE ROUND( mb(K)/das, 3) TO asi, AS(i,1)
1040      STORE 1+1 TO 1
1041      STORE ROUND( mmax(j)/das, 3) TO ASC, AS(i,1)
1042      STORE 1+1 TO 1
1043      STORE ROUND( mb(K+1)/das, 3) TO asd, AS(i,1)
1044      STORE K+2 TO K
1045      STORE 1+1 TO 1
1046      ENDFOR
1047      ENDFOR
1048      ENDFOR

```

```

1009     SELECT mat18
1010     SET ORDER TO 1
1011     ZAP
1012     FOR i=1 TO mnp
1013         APPEND BLANK
1014         REPLA y WITH i
1015         STORE 1 TO K
1016         FOR j=1 TO mnvx*3
1017             STORE 'X'+ALLTRIM(STR(j,2)) TO MACRO
1018             REPLA &macro WITH AS(i,K)
1019             STORE K+1 TO K
1020         ENDFOR
1021     ENDFOR
1022     DEACTIVATE WINDOW impresor
1023     DEACTIVATE WINDOW reportes
1024     RETURN
1026     *: EOF: DISK002.ACT

```

TEXT

```

57 STORE SYS(5)+SYS(2003)+SPACE(15) TO mdirec
58 STORE 'NINGUNO'+SPACE(18) TO mnomproyec
59 STORE SYS(6) TO mpuerto
60 @24,12 SAY 'Proyecto :'+mnomproyec+'Puerto :'+mpuerto+''+LEFT(mdirec,17) COLOR SCHEME 5
61 @07,01 FILL TO 19,36 COLOR SCHEME 5 && w+/B
62 @07,37 FILL TO 19,78 COLOR SCHEME 5 && w+/B
63 ACTIVATE WINDOW demo
64 @00,00 SAY ' MaxComp+ 1.00 ' COLOR SCHEME 7 && br+/B
65 @01,00 SAY ' INGENIERIA CIVIL ' COLOR SCHEME 7
66 @02,00 SAY ' Aleman-Chauca-Silva ' COLOR SCHEME 7
67 FOR MCOL = 03 TO 52
68 MOVE WINDOW demo TO 08.MCOL
69 STORE SECONDS() TO stime
70 DO WHILE SECONDS() <= stime + .015
71 ENDDO
72 ENDFOR
73 ACTIVATE SCREEN BOTTOM
74 @07,01 FILL TO 19,36 COLOR SCHEME 5 && n/B
75 WAIT "Presione una tecla o click del mouse... para continuar!" WINDOW
76 RELEASE WINDOW demo
77 @08,52 TO 12,75 DOUBLE COLOR SCHEME 7 && w+/B
78 @09,53 SAY ' MaxComp+ 1.00 ' COLOR SCHEME 7 && br+/B
79 @10,53 SAY ' INGENIERIA CIVIL ' COLOR SCHEME 7 && br+/B
80 @11,53 SAY ' Aleman-Chauca-Silva ' COLOR SCHEME 7
81 ACTIVATE WINDOW temp
82 CLEAR
83 @01,01 SAY 'Actualizando parametros de funcionamiento...'
84 DO defobje
85 DO defarch
86 SELECT predise
87 SET ORDER TO 1
88 GO TOP
89 STORE proyecto TO mproyecto
90 SET PROCEDURE TO rutinas
91 ON ERROR DO stop WITH ERROR(), MESSAGE(), MESSAGE(1), PROGRAM(), LINENO()
92 DEACTIVATE WINDOW temp
93 @24,23 SAY LEFT(mproyecto,25) COLOR SCHEME 5
94 DO SYSTEM
95 RETURN
96 *: EOF: MAX00.ACT

```

```

1  *:*****
2  *:
3  *:      Program: C:\CIVIL\MAX01.PRG
4  *:
5  *:      System: Anal.y Diseño de Estructuras Aporticadas
6  *:      Author: Rommel R. Alemán T. - Marco P. Silva S.
7  *:      Copyright (c) 1993, Pontificia Universidad Catolica del E.
8  *:      Last modified: 05/16/93      17:11
9  *:
10 *:      Called by: SYSTEM.PRG
11 *:
12 *:      Calls: MAX00.PRG
13 *:
14 *:      Memory Files: PRIORI.MEM
15 *:
16 *:      Documented 05/28/93 at 12:12      FoxDoc version 2.10
17 *:*****
18 SET INTENSITY OFF
19 SELECT usuarios
20 SET ORDER TO 2
21 IF RECCOUNT() <= 0
22     STORE 99 TO mprioridad
23     SAVE TO priori ALL LIKE mprioridad
24     SET INTENSITY ON
25 RETURN
26 ENDIF
27 DEFINE WINDOW max01 FROM 08,21 TO 16,59 DOUBLE SHADOW COLOR SCHEME 07
28 DEFINE WINDOW mensaje FROM 03,35 TO 07,79 DOUBLE SHADOW COLOR SCHEME 10
29 ACTIVATE WINDOW max01
30 @00,06 SAY "      Newmax+ 1.02      "
31 @02,06 SAY "Multi-User FoxPRO + 1.00"
32 @03,06 SAY "Serial Number XPDQWEASTY"
33 @04,06 SAY "I. N. G. E. C. O. M. P. "
34 @05,06 SAY "Su clave : "
35 ro=1
36 DO WHILE ro <= 3
37     mclave=SPACE(5)
38     STORE 0 TO xx
39     @05,17 GET mclave COLOR SCHEME 2
40     READ
41     STORE UPPER(mclave) TO mclave
42     IF mclave=SPACE(5)
43         WAIT 'CLAVE: no valida (presione ESPACE)' WINDOW
44     LOOP
45 ENDIF
46 STORE TRIM(mclave) TO zz
47 STORE LEN(zz) TO nu
48 IF nu=0
49     STORE ro + 1 TO ro
50 LOOP
51 ENDIF
52 SEEK mclave
53 IF EOP()
54     WAIT 'ACCESO :NEGADO...' WINDOW
55     STORE ro + 1 TO ro
56 ELSE

```

```

57      CLEAR
58      STORE ALLTRIM(profesor) TO prof
59      STORE LEN(prof) TO lprof
60      STORE INT((38-lprof)/2) TO lprof
61      @02,lprof SAY prof
62      @04,14 SAY 'BIENVENIDO'
63      WAIT 'ACCESO :CONCEDIDO ...' WINDOW
64      SET INTENSITY ON
65      DEACTIVATE WINDOW max01
66      STORE prioridad TO mprioridad
67      SAVE TO priori ALL LIKE mprioridad
68 <----- RETURN
69      └─ENDIF
70      └─ENDDO
71      DEACTIVATE WINDOW max01
72      DO max00
73      RETURN
74
76      *: KOF: MAX01.ACT

```

```

1  *:*****
2  *:
3  *:      Program: C:\CIVIL\MENU001.PRG
4  *:
5  *:      System: Anal.y Diseño de Estructuras Aporticadas
6  *:      Author: Rommel R. Alemán T. - Marco P. Silva S.
7  *:      Copyright (c) 1993, Pontificia Universidad Catolica del E.
8  *:      Last modified: 05/28/93      12:08
9  *:
10 *:      Called by: DEFOBJK.PRG
11 *:
12 *:      Calls: STOP      (procedure in Rutinas.PRG)
13 *:      : PRED001.PRG
14 *:      : PRED002.PRG
15 *:
16 *:      Documented 05/28/93 at 12:12      FoxDoc version 2.10
17 *:*****
18      STORE BAR() TO opcion1
19      HIDE POPUP menupred
20      SET PROCEDURE TO rutinas
21      ON ERROR DO stop WITH ERROR(), MESSAGE(), MESSAGE(1), PROGRAM(),LINENO()
22      DO CASE
23      =CASE opcion1 = 1
24          DO pred001
25      =CASE opcion1 = 2
26          DO pred002
27      =ENDCASE
28      SHOW POPUP menupred
29      HIDE WINDOW ALL
30      RETURN
32      *: EOF: MENU001.ACT

```

```

1  *:*****
2  *:
3  *:      Program: C:\CIVIL\MENU002.PRG
4  *:
5  *:      System: Anal.y Diseño de Estructuras Aporticadas
6  *:      Author: Rommel R. Alemán T. - Marco P. Silva S.
7  *:      Copyright (c) 1993, Pontificia Universidad Catolica del E.
8  *:      Last modified: 05/16/93      16:05
9  *:
10 *:      Called by: DEFOBJE.PRG
11 *:
12 *:      Calls: STOP              (procedure in RUTINAS.PRG)
13 *:              : DISE001.PRG
14 *:              : DISE002.PRG
15 *:
16 *:      Documented 05/28/93 at 12:12              FoxDoc version 2.10
17 *:*****
18     STORE BAR() TO opcion1
19     HIDE POPUP menudise
20     SET PROCEDURE TO rutinas
21     ON ERROR DO stop WITH ERROR(), MESSAGE(), MESSAGE(1), PROGRAM(),LINEO()
22     DO CASE
23     CASE opcion1 = 1
24         DO dise001
25     CASE opcion1 = 2
26         DO dise002
27     ENDCASE
28     SHOW POPUP menudise
29     HIDE WINDOW ALL
30     RETURN
32     *: EOF: MENU002.ACT

```

```

1  *:*****
2  *:
3  *:      Program: C:\CIVIL\MENU003.PRG
4  *:
5  *:      System: Anal.y Diseño de Estructuras Aporticadas
6  *:      Author: Rommel R. Alemán T. - Marco P. Silva S.
7  *:      Copyright (c) 1993, Pontificia Universidad Catolica del E.
8  *:      Last modified: 05/16/93      16:05
9  *:
10 *:      Called by: DEFOBJE.PRG
11 *:
12 *:      Calls: STOP                      (procedure in RUTINAS.PRG)
13 *:              : REPO001.PRG
14 *:              : REPO002.PRG
15 *:              : REPO003.PRG
16 *:              : REPO004.PRG
17 *:              : REPO005.PRG
18 *:              : REPO006.PRG
19 *:              : REPO007.PRG
20 *:              : REPO008.PRG
21 *:              : REPO009.PRG
22 *:              : REPO010.PRG
23 *:              : REPO011.PRG
24 *:              : REPO012.PRG
25 *:              : REPO013.PRG
26 *:              : REPO014.PRG
27 *:              : REPO015.PRG
28 *:              : REPO016.PRG
29 *:              : REPO017.PRG
30 *:              : REPO018.PRG
31 *:              : REPO019.PRG
32 *:              : REPO020.PRG
33 *:              : REPO021.PRG
34 *:              : REPO022.PRG
35 *:              : REPO023.PRG
36 *:
37 *:      Documented 05/28/93 at 12:12          FoxDoc version 2.10
38 *:*****
39  STORE BAR() TO opcion1
40  HIDE POPUP menurepo
41  SET PROCEDURE TO rutinas
42  ON ERROR DO stop WITH ERROR(), MESSAGE(), MESSAGE(1), PROGRAM(),LINENO()
43  =DO CASE
44  =CASE opcion1 = 1
45  DO repo001
46  =CASE opcion1 = 2
47  DO repo002
48  =CASE opcion1 = 3
49  DO repo003
50  =CASE opcion1 = 4
51  DO repo004
52  =CASE opcion1 = 5
53  DO repo005
54  =CASE opcion1 = 6
55  DO repo006
56  =CASE opcion1 = 7

```

```

57      DO repo007
58  ==CASE opcion1 = 8
59      DO repo008
60  ==CASE opcion1 = 9
61      DO repo009
62  ==CASE opcion1 = 10
63      DO repo010
64  ==CASE opcion1 = 11
65      DO repo011
66  ==CASE opcion1 = 12
67      DO repo012
68  ==CASE opcion1 = 13
69      DO repo013
70  ==CASE opcion1 = 14
71      DO repo014
72  ==CASE opcion1 = 15
73      DO repo015
74  ==CASE opcion1 = 16
75      DO repo016
76  ==CASE opcion1 = 17
77      DO repo017
78  ==CASE opcion1 = 18
79      DO repo018
80  ==CASE opcion1 = 19
81      DO repo019
82  ==CASE opcion1 = 20
83      DO repo020
84  ==CASE opcion1 = 21
85      DO repo021
86  ==CASE opcion1 = 22
87      DO repo022
88  ==CASE opcion1 = 23
89      DO repo023
90  ==ENDCASE
91      SHOW POPUP menurepo
92      HIDE WINDOW ALL
93      RETURN
95      *: KOF: MENU003.ACT

```

```

1  *:*****
2  *:
3  *:      Program: C:\CIVIL\MENU004.PRG
4  *:
5  *:      System: Anal.y Diseño de Estructuras Aporticadas
6  *:      Author: Rommel R. Alemán T. - Marco P. Silva S.
7  *:      Copyright (c) 1993, Pontificia Universidad Catolica del E.
8  *:      Last modified: 05/16/93      16:05
9  *:
10 *:      Called by: DEFOBJE.PRG
11 *:
12 *:      Calls: STOP      (procedure in RUTINAS.PRG)
13 *:      : SIST001.PRG
14 *:      : SIST002.PRG
15 *:      : SIST003.PRG
16 *:      : SIST004.PRG
17 *:
18 *:      Memory Files: PRIORI.MEM
19 *:
20 *:      Documented 05/28/93 at 12:12      FoxDoc version 2.10
21 *:*****
22  STORE BAR() TO opcion1
23  HIDE POPUP menusist
24  RESTORE FROM priori ADDITIVE
25  SET PROCEDURE TO rutinas
26  ON ERROR DO stop WITH ERROR(), MESSAGE(), MESSAGE(1), PROGRAM(),LINENO()
27  DO CASE
28  CASE opcion1 = 1
29      IF mprioridad>=40
30          DO sist001
31      ELSE
32          WAIT 'USUARIO NO AUTORIZADO...' WINDOW
33      ENDIF
34  CASE opcion1 = 2
35      IF mprioridad>=60
36          DO sist002
37      ELSE
38          WAIT 'USUARIO NO AUTORIZADO...' WINDOW
39      ENDIF
40  CASE opcion1 = 3
41      IF mprioridad>=80
42          DO sist003
43      ELSE
44          WAIT 'USUARIO NO AUTORIZADO...' WINDOW
45      ENDIF
46  CASE opcion1 = 4
47      IF mprioridad>=99
48          DO sist004
49      ELSE
50          WAIT 'USUARIO NO AUTORIZADO...' WINDOW
51      ENDIF
52  ENDCASE
53  SHOW POPUP menusist
54  HIDE WINDOW ALL
55  RETURN
56  *: EOF: MENU004.ACT

```

```

1  *:*****
2  *:
3  *:      Program: C:\CIVIL\MENU005.PRG
4  *:
5  *:      System: Anal.y Diseño de Estructuras Aporticadas
6  *:      Author: Rommel R. Alemán T. - Marco P. Silva S.
7  *:      Copyright (c) 1993, Pontificia Universidad Catolica del E.
8  *:      Last modified: 05/16/93      16:05
9  *:
10 *:      Called by: DEFOBJE.PRG
11 *:
12 *:      Calls: STOP              (procedure in RUTINAS.PRG)
13 *:              : UTIL001.PRG
14 *:              : UTIL002.PRG
15 *:              : UTIL003.PRG
16 *:              : UTIL004.PRG
17 *:              : UTIL005.PRG
18 *:              : UTIL006.PRG
19 *:
20 *:      Documented 05/28/93 at 12:12      FoxDoc version 2.10
21 *:*****
22     STORE BAR() TO opcion1
23     HIDE POPUP menuutil
24     SET PROCEDURE TO rutinas
25     ON ERROR DO stop WITH ERROR(), MESSAGE(), MESSAGE(1), PROGRAM(),LINENO()
26     DO CASE
27     ==CASE opcion1 = 1
28         DO util001
29     ==CASE opcion1 = 2
30         DO util002
31     ==CASE opcion1 = 3
32         DO util003
33     ==CASE opcion1 = 4
34         DO util004
35     ==CASE opcion1 = 5
36         DO util005
37     ==CASE opcion1 = 6
38         DO util006
39     ==ENDCASE
40     SHOW POPUP menuutil
41     HIDE WINDOW ALL
42     RETURN
44     *: EOF: MENU005.ACT

```

```

1  *:*****
2  *:
3  *:      Program: C:\CIVIL\MENU006.PRG
4  *:
5  *:      System: Anal.y Diseño de Estructuras Aporticadas
6  *:      Author: Rommel R. Alemán T. - Marco P. Silva S.
7  *:      Copyright (c) 1993, Pontificia Universidad Catolica del E.
8  *:      Last modified: 05/16/93      16:06
9  *:
10 *:      Called by: DEFOBJE.PRG
11 *:
12 *:      Calls: STOP      (procedure in RUTINAS.PRG)
13 *:      : TERM001.PRG
14 *:
15 *:      Documented 05/28/93 at 12:12      FoxDoc version 2.10
16 *:*****
17     STORE BAR() TO opcion1
18     HIDE POPUP menuterm
19     SET PROCEDURE TO rutinas
20     ON ERROR DO stop WITH ERROR(), MESSAGE(), MESSAGE(1), PROGRAM(),LINE0()
21     DO CASE
22     CASE opcion1 = 1
23         DO term001
24     ENDCASE
25     SHOW POPUP menuterm
26     HIDE WINDOW ALL
27     RETURN
28     *: EOF: MENU006.ACT

```

```

1  *:*****
2  *:
3  *: Procedure file: C:\CIVIL\PRED001.PRG
4  *:
5  *:      System: Anal.y Diseño de Estructuras Aporticadas
6  *:      Author: Rommel R. Alemán T. - Marco P. Silva S.
7  *:      Copyright (c) 1993, Pontificia Universidad Catolica del E.
8  *:      Last modified: 05/16/93      16:12
9  *:
10 *: Procs & Fncts: SETUP_REP
11 *:                : DESPLEGAR
12 *:                : MOSTRAR
13 *:                : INCREMENTAR
14 *:                : BORRAR
15 *:                : EDITAR
16 *:                : FINAL
17 *:                : PROXIMO
18 *:                : ANTERIOR
19 *:                : SALIR
20 *:                : BUSCAR
21 *:                : INICIO
22 *:                : STANDBY
23 *:                : DATOS
24 *:
25 *:      Set by: MENU001.PRG
26 *:
27 *:      Calls: SETUP_REP      (procedure in PRED001.PRG)
28 *:                : DESPLEGAR      (procedure in PRED001.PRG)
29 *:                : MOSTRAR      (procedure in PRED001.PRG)
30 *:
31 *:      Documented 05/28/93 at 12:12      FoxDoc version 2.10
32 *:*****
33   SET FUNCTION f2 TO "B;"
34   STORE "" TO mproyecto
35   STORE 0 TO maux
36   DO setup_rep
37   SELECT predise
38   SET ORDER TO 1
39   IF EOF()
40   GO TOP
41   ENDIF
42   STORE codigo TO mcodigo
43   STORE DATE() TO mfecha
44   ACTIVATE WINDOW trabajo
45   DO desplegar
46   DO mostrar
47   STORE .T. TO reporte
48   DO WHILE reporte
49   ACTIVATE MENU salesrep PAD search
50   ENDDO
51   DEACTIVATE WINDOW trabajo
52   RELEASE MENU salesrep
53   CLOSE PROCEDURE
54   @24,23 SAY LEFT(mproyecto,25) COLOR SCHEME 5
55   RETURN
70  *!*****

```

```

71 *!
72 *!      Procedure: INCREMETAR
73 *!
74 *!      Called by: SETUP_REP      (procedure in PRED001.PRG)
75 *!
76 *!      Calls: DESPLEGAR      (procedure in PRED001.PRG)
77 *!           : MOSTRAR      (procedure in PRED001.PRG)
78 *!           : STANDBY      (procedure in PRED001.PRG)
79 *!           : EDITAR      (procedure in PRED001.PRG)
80 *!
81 *!      Memory Files: DEFECTO.MEM
82 *!
83 *!*****
70      PROCEDURE incremetar
71      RESTORE FROM defecto ADDITIVE
72      GO BOTTOM
73      STORE codigo TO mcodigo
74      DO desplegar
75      STORE mcodigo+1 TO mcodigo
76      @01,24 GET mcodigo PICTURE '*****' RANGE 1,99999
77      READ
78      IF LASTKEY()=27
79      <=====RETURN
80      <-----ENDIF
81      SEEK mcodigo
82      IF FOUND()
83      DO mostrar
84      DO standby WITH "No puedo añadir el registro: el registro ya existe"
85      <=====RETURN
86      <-----ELSE
87      APPEND BLANK
88      REPLACE NEXT 1 codigo WITH mcodigo
89      REPLACE NEXT 1 fc      WITH mfc
90      REPLACE NEXT 1 fy      WITH mfy
91      REPLACE NEXT 1 el      WITH mel
92      REPLACE NEXT 1 pb      WITH mpb
93      REPLACE NEXT 1 pp      WITH mpp
94      REPLACE NEXT 1 ea      WITH mea
95      REPLACE NEXT 1 ph      WITH mph
96      REPLACE NEXT 1 ph1     WITH mph
97      REPLACE NEXT 1 R       WITH mr
98      REPLACE NEXT 1 beta    WITH mbeta
99      REPLACE NEXT 1 fi1     WITH mfi1
100     REPLACE NEXT 1 fi2     WITH mfi2
101     REPLACE NEXT 1 fi3     WITH mfi3
102     <-----ENDIF
103     DO editar
104     RETURN
118 *!*****
119 *!
120 *!      Procedure: BORRAR
121 *!
122 *!      Called by: SETUP_REP      (procedure in PRED001.PRG)
123 *!
124 *!      Calls: MOSTRAR      (procedure in PRED001.PRG)
125 *!

```

```

126 *!           Uses: &NOM1.DBF
127 *!           : &NOM2.DBF
128 *!           : &NOM3.DBF
129 *!
130 *!*****
118     PROCEDURE borrar
119     STORE SPACE(1) TO do_it
120     ACTIVATE WINDOW mensajes
121     @01,02 SAY "Esta usted seguro de querer borrar el registro [S/N] : " GET do_it PICT "@M N,S"
122     READ
123     IF do_it # "S"
124         DEACTIVATE WINDOW mensajes
125     <===== RETURN
126     <----- ENDIF
127     IF RECCOUNT()#0
128         CLEAR
129         @01,02 SAY "ELIMINANDO REGISTRO..."
130         STORE codigo TO mnum
131         DELETE NEXT 1
132         IF .NOT. EOF()
133             SKIP
134         ELSE
135             GO BOTTOM
136         ENDIF
137         STORE ALLTRIM(STR(mnum)) TO maux
138         STORE "NVX"+maux+".MEM" TO nom1
139         STORE "NVY"+maux+".MEM" TO nom2
140         STORE "NP"+maux+".MEM" TO nom3
141         DELETE FILE &nom1
142         DELETE FILE &nom2
143         DELETE FILE &nom3
144     ENDIF
145     CLEAR
146     DEACTIVATE WINDOW mensajes
147     DO mostrar
148     RETURN
149 *!*****
150 *!
151 *!           Procedure: DESPLEGAR
152 *!
153 *!           Called by: PRED001.PRG
154 *!                   : DISE001.PRG
155 *!                   : SIST004.PRG
156 *!                   : INCREMENTAR (procedure in PRED001.PRG)
157 *!
158 *!*****
159     PROCEDURE desplegar
160     CLEAR
161     @01,01 SAY "    Proyecto NUMERO : "
162     @03,01 SAY "    Nombre del Cliente : "
163     @04,01 SAY "    Domicilio de la obra : "
164     @05,01 SAY "    Tipo de obra : "
165     @06,01 SAY "    Ing. Calculista : "
166     @07,01 SAY "    Fecha (mm-dd-aa) : "
167     @09,00 SAY SPACE(26)+"** Datos para Prediseño **"+SPACE(26) COLOR SCHEME 7
168     @11,01 SAY "    f'c del hormigón:"

```

```

169 @12,01 SAY "      fy del acero:"
170 @13,01 SAY "      NO vanos X:"
171 @14,01 SAY "      NO vanos Y:"
172 @15,01 SAY "      Espesor loseta:"
173 @16,01 SAY "      Peso bloques:"
174 @17,01 SAY "      Peso paredes:"
175 @18,01 SAY "      Espesor acabado:"
176 @19,01 SAY "      Ancho viga:"
177 @11,38 SAY "      Ancho columna:"
178 @12,38 SAY "      NO de pisos:"
179 @13,38 SAY "      Altura de losa:"
180 @14,38 SAY "      Carga muerta:"
181 @15,38 SAY "      Carga viva:"
182 @16,38 SAY "      Carga sismica:"
183 @17,38 SAY "      Carga Ultima:"
184 @18,38 SAY "      % de acero minimo:"
185 @19,38 SAY "      Cte.Esfuerzo Unit.:"
186 RETURN
199 *!*****
200 *!
201 *!      Procedure: EDITAR
202 *!
203 *!      Called by: SETUP_REP      (procedure in PRED001.PRG)
204 *!                  : INCREMENTAR      (procedure in PRED001.PRG)
205 *!
206 *!      Calls: MOSTRAR      (procedure in PRED001.PRG)
207 *!                  : ESCOJER      (procedure in RUTINAS.PRG)
208 *!                  : DATOS      (procedure in PRED001.PRG)
209 *!
210 *!*****
199 PROCEDURE editar
200 IF RECCOUNT()#0
201     STORE nvx TO onvx
202     STORE nvy TO onvy
203     STORE np TO onp
204     @01,24 GET codigo PICTURE "####"
205     CLEAR GETS
206     @03,24 GET proyecto
207     @04,24 GET direccion
208     @05,24 GET obra
209     @06,24 GET calculo
210     @07,24 GET fecha
211     @11,24 GET fc PICTURE "### Kg/cm²" RANGE 1,999
212     @12,24 GET fy PICTURE "#### Kg/cm²" RANGE 1,9999
213     @13,24 GET nvx PICTURE "###" RANGE 1,20
214     @14,24 GET nvy PICTURE "###" RANGE 1,20
215     @15,24 GET el PICTURE "### m" RANGE 0.0001,9.9999
216     @16,24 GET pb PICTURE "### Tn/m²" RANGE 0.0001,9.9999
217     @17,24 GET pp PICTURE "### Tn/m²" RANGE 0.0001,9.9999
218     @18,24 GET ea PICTURE "### m" RANGE 0.0001,9.9999
219     @19,24 GET bv PICTURE "### cm" RANGE 1,999
220     @11,61 GET bc PICTURE "### cm" RANGE 1,999
221     @12,61 GET np PICTURE "###" RANGE 1,10
222     @13,61 GET hl PICTURE "### cm"
223     @14,61 GET cm PICTURE "#### Kg/m²"
224 READ

```

```

225 | IF LASTKEY()=27
226 |     DO mostrar
227 | <-----RETURN
228 | ENDIF
229 |     SELECT carviva
230 |     SET ORDER TO 1
231 |     DEFINE POPUP cargas FROM 00,45 TO 20,77 PROMPT FIELD categoria SHADOW COLOR SCHEME 7
232 |     ON SELECTION POPUP cargas DO escojer WITH PROMPT(), BAR()
233 |     ACTIVATE POPUP cargas
234 |     ACTIVATE WINDOW mensajes
235 |     STORE carga TO maux
236 |     CLEAR
237 |     @00,00
238 |     ? notas
239 |     WAIT 'Presione ENTER para continuar...' WINDOW
240 |     DEACTIVATE WINDOW mensajes
241 |     SELECT predise
242 |     SET ORDER TO 1
243 |     REPLA cv WITH maux
244 |     @15,61 GET cv PICTURE "#### Kg/m²" RANGE 1,9999
245 |     @16,61 GET cs PICTURE "### Kg/m²"
246 |     @17,61 GET cu PICTURE "#### Kg/m²"
247 |     @18,61 GET ph PICTURE "#.####" RANGE 0.0100,0.0800
248 |     READ
249 |     DO CASE
250 |     =CASE ph>=0.0100 .AND. ph<=0.0129
251 |         REPLA R WITH 14.34
252 |     =CASE ph>=0.0130 .AND. ph<=0.0150
253 |         REPLA R WITH 15.94
254 |     =CASE ph>=0.0151 .AND. ph<=0.0800
255 |         REPLA R WITH 16.73
256 |     ENDCASE
257 |     @19,61 SAY R PICTURE "##.##"
258 |     READ
259 |     STORE nvx TO mnvx
260 |     STORE nvy TO mnvy
261 |     STORE np TO mnp
262 |     STORE codigo TO mnumero
263 |     DO datos WITH mnvx, mnvy, mnp, mnumero
264 |     DO mostrar
265 | ENDIF
266 | RETURN
276 | *****
277 | *!
278 | *! Procedure: FINAL
279 | *!
280 | *! Called by: SETUP_REP (procedure in PRED001.PRG)
281 | *!
282 | *! Calls: MOSTRAR (procedure in PRED001.PRG)
283 | *!
284 | *****
276 | PROCEDURE final
277 | GO BOTTOM
278 | DO mostrar
279 | RETURN
290 | *****

```

```

291 *!
292 *!      Procedure: PROXIMO
293 *!
294 *!      Called by: SETUP_REP      (procedure in PRED001.PRG)
295 *!
296 *!      Calls: STANDBY      (procedure in PRED001.PRG)
297 *!           : MOSTRAR      (procedure in PRED001.PRG)
298 *!
299 *!*****
300 PROCEDURE proximo
301 IF RECCOUNT()#0
302     SKIP
303     IF BOF()
304         GO BOTTOM
305         DO standby WITH "El registro presente es el ultimo..."
306     ELSE
307         DO mostrar
308     ENDIF
309 ENDIF
310 RETURN
311 *!*****
312 *!
313 *!      Procedure: ANTERIOR
314 *!
315 *!      Called by: SETUP_REP      (procedure in PRED001.PRG)
316 *!
317 *!      Calls: STANDBY      (procedure in PRED001.PRG)
318 *!           : MOSTRAR      (procedure in PRED001.PRG)
319 *!
320 *!*****
321 PROCEDURE anterior
322 IF RECCOUNT()#0
323     SKIP -1
324     IF BOF()
325         GO TOP
326         DO standby WITH "El registro presente es el primero..."
327     ELSE
328         DO mostrar
329     ENDIF
330 ENDIF
331 RETURN
332 *!*****
333 *!
334 *!      Procedure: SALIR
335 *!
336 *!      Called by: SETUP_REP      (procedure in PRED001.PRG)
337 *!
338 *!*****
339 PROCEDURE salir
340 STORE proyecto TO mproyecto
341 STORE .F. TO reporte
342 DEACTIVATE MENU
343 RETURN
344 *!*****
345 *!
346 *!      Procedure: SETUP_REP

```

```

356 *!
357 *!      Called by: PRED001.PRG
358 *!           : DISK001.PRG
359 *!           : SIST004.PRG
360 *!
361 *!      Calls: PROXIMO      (procedure in PRED001.PRG)
362 *!           : ANTERIOR    (procedure in PRED001.PRG)
363 *!           : INICIO      (procedure in PRED001.PRG)
364 *!           : FINAL       (procedure in PRED001.PRG)
365 *!           : BUSCAR      (procedure in PRED001.PRG)
366 *!           : EDITAR      (procedure in PRED001.PRG)
367 *!           : INCREMENTAR (procedure in PRED001.PRG)
368 *!           : BORRAR      (procedure in PRED001.PRG)
369 *!           : SALIR       (procedure in PRED001.PRG)
370 *!
371 *!*****
353 PROCEDURE setup_rep
354 DEFINE MENU salesrep COLOR SCHEME 3
355 DEFINE PAD proximo OF salesrep PROMPT '\<Prox' AT 21,03
356 DEFINE PAD anterior OF salesrep PROMPT '\<Ante' AT 21,10
357 DEFINE PAD inicio OF salesrep PROMPT '\<Inicio' AT 21,17
358 DEFINE PAD fin OF salesrep PROMPT '\<Fin' AT 21,26
359 DEFINE PAD buscar OF salesrep PROMPT '\<Buscar' AT 21,32
360 DEFINE PAD editar OF salesrep PROMPT '\<Editar' AT 21,41
361 DEFINE PAD anadir OF salesrep PROMPT 'aña\<Dir' AT 21,50
362 DEFINE PAD borrar OF salesrep PROMPT 'bor\<Rar' AT 21,59
363 DEFINE PAD salir OF salesrep PROMPT '\<Salir' AT 21,68
364 ON SELECTION PAD proximo OF salesrep DO proximo
365 ON SELECTION PAD anterior OF salesrep DO anterior
366 ON SELECTION PAD inicio OF salesrep DO inicio
367 ON SELECTION PAD fin OF salesrep DO final
368 ON SELECTION PAD buscar OF salesrep DO buscar
369 ON SELECTION PAD editar OF salesrep DO editar
370 ON SELECTION PAD anadir OF salesrep DO incremetar
371 ON SELECTION PAD borrar OF salesrep DO borrar
372 ON SELECTION PAD salir OF salesrep DO salir
373 RETURN
391 *!*****
392 *!
393 *!      Procedure: MOSTRAR
394 *!
395 *!      Called by: PRED001.PRG
396 *!           : DISK001.PRG
397 *!           : SIST004.PRG
398 *!           : INCREMENTAR (procedure in PRED001.PRG)
399 *!           : BORRAR      (procedure in PRED001.PRG)
400 *!           : EDITAR      (procedure in PRED001.PRG)
401 *!           : FINAL       (procedure in PRED001.PRG)
402 *!           : PROXIMO      (procedure in PRED001.PRG)
403 *!           : ANTERIOR    (procedure in PRED001.PRG)
404 *!           : BUSCAR      (procedure in PRED001.PRG)
405 *!           : INICIO      (procedure in PRED001.PRG)
406 *!
407 *!*****
391 PROCEDURE mostrar
392 @01.24 SAY codigo PICTURE "*****"

```

```

393 @03,24 SAY proyecto
394 @04,24 SAY direccion
395 @05,24 SAY obra
396 @06,24 SAY calculo
397 @07,24 SAY fecha
398 @11,24 SAY fc PICTURE "### Kg/cm²"
399 @12,24 SAY fy PICTURE "#### Kg/cm²"
400 @13,24 SAY nvx PICTURE "##"
401 @14,24 SAY nvy PICTURE "##"
402 @15,24 SAY el PICTURE "$.#### m"
403 @16,24 SAY pb PICTURE "$.#### Tn/m²"
404 @17,24 SAY pp PICTURE "$.#### Tn/m²"
405 @18,24 SAY ea PICTURE "$.#### m"
406 @19,24 SAY bv PICTURE "### cm"
407 @11,61 SAY bc PICTURE "### cm"
408 @12,61 SAY np PICTURE "##"
409 @13,61 SAY hl PICTURE "## cm"
410 @14,61 SAY cm PICTURE "#### Kg/m²"
411 @15,61 SAY cv PICTURE "#### Kg/m²"
412 @16,61 SAY cs PICTURE "### Kg/m²"
413 @17,61 SAY cu PICTURE "#### Kg/m²"
414 @18,61 SAY ph PICTURE "$.####"
415 @19,61 SAY R PICTURE "##.##"
416 RETURN
428 *!*****
429 *!
430 *! Procedure: BUSCAR
431 *!
432 *! Called by: SETUP_REP (procedure in PRED001.PRG)
433 *!
434 *! Calls: ESCOJE (procedure in RUTINAS.PRG)
435 *! : MOSTRAR (procedure in PRKD001.PRG)
436 *! : STANDBY (procedure in PRED001.PRG)
437 *!
438 *!*****
428 PROCEDURE buscar
429 last_rec=RECNO()
430 STORE 0 TO mcodigo
431 @01,24 GET mcodigo PICTURE '####'
432 READ
433 IF mcodigo=0
434 DEFINE POPUP localiza FROM 00,45 TO 20,77 PROMPT FIELD proyecto SHADOW COLOR SCHEME 7
435 ON SELECTION POPUP localiza DO escoje WITH PROMPT(), BAR()
436 ACTIVATE POPUP localiza
437 STORE codigo TO mcodigo
438 ENDIF
439 IF mcodigo=0
440 DO mostrar
441 <====RETURN
442 ENDIF
443 SEEK mcodigo
444 IF .NOT. FOUND()
445 DO standby WITH "El registro no ha sido encontrado en el archivo"
446 GO last_rec
447 ENDIF
448 DO mostrar

```

```

449     RETURN
459  *!*****
460  *!
461  *!     Procedure: INICIO
462  *!
463  *!     Called by: SETUP_REP           (procedure in PRED001.PRG)
464  *!
465  *!     Calls: MOSTRAR                 (procedure in PRED001.PRG)
466  *!
467  *!*****
459     PROCEDURE inicio
460     GO TOP
461     DO mostrar
462     RETURN
473  *!*****
474  *!
475  *!     Procedure: STANDBY
476  *!
477  *!     Called by: INCREMENTAR         (procedure in PRED001.PRG)
478  *!                  : PROXIMO         (procedure in PRED001.PRG)
479  *!                  : ANTERIOR        (procedure in PRED001.PRG)
480  *!                  : BUSCAR          (procedure in PRED001.PRG)
481  *!
482  *!*****
473     PROCEDURE standby
474     PARAMETER mensaje
475     ACTIVATE WINDOW mensajes
476     STORE INT((59 - LEN(mensaje))/2) TO ln
477     @01,ln SAY mensaje
478     READ
479     DEACTIVATE WINDOW mensajes
480     RETURN
492  *!*****
493  *!
494  *!     Procedure: DATOS
495  *!
496  *!     Called by: EDITAR               (procedure in PRED001.PRG)
497  *!
498  *!     Memory Files: &NOM01
499  *!                  : &NOM02
500  *!                  : &NOM03
501  *!
502  *!*****
492     PROCEDURE datos
493     PARAMETER mvx, mvy, mp, numero
494     IF mvx=0 .OR. mvy=0 .OR. mp=0
495     |   WAIT 'El No. de vanos en X o Y es igual a 0; el No. de pisos es igual a 0' WINDOW
496     <====RETURN
497     |_ENDIF
498     STORE STR(numero,5) TO mnum
499     STORE 'nvx'+ALLTRIM(mnum) TO nom01
500     STORE 'nvy'+ALLTRIM(mnum) TO nom02
501     STORE 'np'+ALLTRIM(mnum) TO nom03
502     DIMENSION X(mvx+1)
503     STORE 0 TO X
504     ACTIVATE WINDOW vano

```

```

505     FOR i=1 TO mvx
506         @i-1.01 SAY 'X'
507         @i-1.02 SAY i PICTURE '## =' FUNCTION '1'
508         @i-1.07 GET X(i) PICTURE '#.##' RANGE 0,9.99
509     ENDFOR
510     READ
511     DEACTIVATE WINDOW vano
512     SAVE TO &nom01 ALL LIKE X
513     DIMENSION y(mvy+1)
514     STORE 0 TO y
515     ACTIVATE WINDOW vano
516     FOR i=1 TO mvy
517         @i-1.01 SAY 'Y'
518         @i-1.02 SAY i PICTURE '## =' FUNCTION '1'
519         @i-1.07 GET y(i) PICTURE '#.##' RANGE 0,9.99
520     ENDFOR
521     READ
522     DEACTIVATE WINDOW vano
523     SAVE TO &nom02 ALL LIKE y
524     DIMENSION hp(mp+1)
525     STORE 0 TO hp
526     ACTIVATE WINDOW vano
527     STORE mp TO j
528     FOR i=1 TO mp
529         @i-1.01 SAY 'P'
530         @i-1.02 SAY j PICTURE '## =' FUNCTION '1'
531         @i-1.07 GET hp(i) PICTURE '#.##' RANGE 0,9.99
532         STORE j-1 TO j
533     ENDFOR
534     READ
535     SAVE TO &nom03 ALL LIKE hp
536     DEACTIVATE WINDOW vano
537     RETURN
539     *: EOF: PREDO01.ACT

```

```

1  *:*****
2  *:
3  *:      Program: C:\CIVIL\PREDO02.PRG
4  *:
5  *:      System: Anal.y Diseño de Estructuras Aporticadas
6  *:      Author: Rommel R. Alemán T. - Marco P. Silva S.
7  *:      Copyright (c) 1993, Pontificia Universidad Catolica del E.
8  *:      Last modified: 04/28/93      18:11
9  *:
10 *:      Called by: MENU001.PRG
11 *:
12 *:      Memory Files: &NOM01
13 *:                  : &NOM02
14 *:                  : &NOM03
15 *:
16 *:      Documented 05/28/93 at 12:12      FoxDoc version 2.10
17 *:*****
18  RELEASE ALL
19  SELECT predise
20  SET ORDER TO 1
21  IF nvx<=0 .OR. nvx<=0 .OR. bv<=0 .OR. bc<=0 .OR. np<=0
22  WAIT 'ERROR: Datos incompatibles o no existen...' WINDOW
23  RETURN
24  ENDIF
25  ACTIVATE WINDOW reportes
26  CLEAR
27  SELECT predise
28  SET ORDER TO 1
29  @00,13 TO 02,55
30  STORE 'CALCULOS PARA PREDISEÑO' TO titulo
31  STORE ROUND((40 - LEN(titulo)) / 2,0) + 14 TO mlon
32  @01,mlon SAY titulo
33  @04,01 SAY "      Proyecto NUMERO :"
34  @04,24 SAY codigo PICTURE "####"
35  @05,01 SAY "      Nombre del Cliente :"
36  @05,24 SAY proyecto
37  @06,01 SAY "Domicilio de la obra :"
38  @06,24 SAY direccion
39  @07,01 SAY "      Tipo de obra :"
40  @07,24 SAY obra
41  @08,01 SAY "      Ing. Calculista :"
42  @08,24 SAY calculo
43  @09,01 SAY "      Fecha (mm-dd-aa) :"
44  @09,24 SAY fecha
45  STORE SPACE(1) TO res
46  @11,01 SAY 'Esta seguro de querer continuar [S/N] :' GET res PICTURE '@m S, N'
47  READ
48  IF LASTKEY()=27 .OR. res='N'
49  DEACTIVATE WINDOW reportes
50  RETURN
51  ENDIF
52  ACTIVATE WINDOW impresor
53  CLEAR
54  @01,01 SAY 'PROCESANDO...'
55  STORE codigo TO numero
56  STORE nvx TO mnvx

```

```

57     STORE nvy TO mnvy
58     STORE np TO mnp
59     STORE STR(numero,5) TO mnum
60     STORE 'nvx'+ALLTRIM(mnum) TO nom01
61     STORE 'nvy'+ALLTRIM(mnum) TO nom02
62     STORE 'np'+ALLTRIM(mnum) TO nom03
63     RESTORE FROM &nom01 ADDITIVE
64     RESTORE FROM &nom02 ADDITIVE
65     RESTORE FROM &nom03 ADDITIVE
66     STORE codigo TO numero
67     STORE X(1) TO lm
68     FOR i=1 TO nvx-1
69         IF lm <= X(i+1)
70             STORE X(i+1) TO lm
71         ENDIF
72     ENDFOR
73     STORE lm TO vxm
74     STORE y(1) TO lm
75     FOR i=1 TO nvy-1
76         IF lm <= y(i+1)
77             STORE y(i+1) TO lm
78         ENDIF
79     ENDFOR
80     STORE lm TO vym
81     IF vxm > vym
82         STORE vxm TO lm
83     ELSE
84         STORE vym TO lm
85     ENDIF
86     STORE 0 TO maux
87     STORE ROUND(lm*(800+0.0712*fy)*100/36000,0) TO maux
88     DO CASE
89     CASE maux<=15
90         STORE 15 TO maux
91     CASE maux>15 .AND. maux<=20
92         STORE 20 TO maux
93     CASE maux>20 .AND. maux<=25
94         STORE 25 TO maux
95     OTHERWISE
96         WAIT 'Revise Datos del PROYECTO ...' WINDOW
97     ENDCASE
98     CLEAR
99     @01,01 SAY 'OBTENIENDO VALOR DE ALTURA DE LOSA...'
100    REPLA hl WITH maux
101    STORE 0 TO maux
102    STORE (el*2.4)+((hl-el)/100*0.1*3.6*2.4)+(8*pb)+(pp*1)+(ea*1.8) TO maux
103    REPLA cm WITH ROUND(maux*1000,0)
104    STORE 0 TO maux
105    STORE 1.4 * cm + 1.7 * cv + 1.7 * 0.25 * cm TO maux
106    CLEAR
107    @01,01 SAY 'OBTENIENDO VALOR DE CARGA ULTIMA...'
108    REPLA cu WITH ROUND(maux,0)
109    STORE 0 TO maux
110    STORE 0.25*cm TO maux
111    CLEAR
112    @01,01 SAY 'OBTENIENDO VALOR DE CARGA SISMICA...'

```

```

113  REPLA cs WITH ROUND(maux,0)
114  STORE cu TO mcu
115  STORE beta TO mbeta
116  STORE fc TO mfc
117  STORE ph TO mph
118  STORE fy TO mfy
119  STORE bc TO mbc
120  STORE R TO mr
121  STORE bv TO mbv
122  RELEASE maux
123  DIMENSION mat1(mnvy,mnvx), mat2(mnvy,mnvx), aux(mnvx,mnvy)
124  STORE 0 TO mat1,mat2
125  CLEAR
126  @01,01 SAY 'GENERANDO MATRIZ DE MOSAICO DE CARGAS...'
127  STORE 1 TO l
128  FOR i=mnvy TO 1 STEP -1
129      FOR j=1 TO mnvx
130          STORE 0 TO q, mm
131          IF X(j) > y(i)
132              STORE y(i)/X(j) TO mm
133              q=cu*y(i)/3 * (3-mm^2)/2
134              STORE q TO mat1(i,j)
135          ELSE
136              q=cu*X(j)/3
137              STORE q TO mat1(i,j)
138          ENDIF
139      ENDFOR
140      STORE l+1 TO l
141  ENDFOR
142  STORE 1 TO l
143  FOR i=1 TO mnvx
144      STORE 1 TO K
145      FOR j=mnvy TO 1 STEP -1
146          STORE 0 TO q, mm
147          IF y(j) > X(i)
148              STORE X(i)/y(j) TO mm
149              q=cu*X(i)/3 * (3-mm^2)/2
150              STORE q TO aux(i,j)
151          ELSE
152              q=cu*y(j)/3
153              STORE q TO aux(i,j)
154          ENDIF
155      ENDFOR
156      STORE l+1 TO l
157  ENDFOR
158  FOR i=1 TO mnvx
159      STORE 1 TO K
160      FOR j=mnvy TO 1 STEP -1
161          STORE aux(i,j) TO mat2(K,i)
162          STORE K+1 TO K
163      ENDFOR
164  ENDFOR
165  SELECT mat01
166  SET ORDER TO 1
167  ZAP
168  STORE 1 TO l

```

```

169   FOR i=mnvy TO 1 STEP -1
170     FOR K=1 TO 2
171       APPEND BLANK
172       REPLA y WITH 1
173       FOR j=1 TO mnvx
174         STORE 'X'+ALLTRIM(STR(j,2)) TO MACRO
175         REPLA &macro WITH mat1(i,j)
176       ENDFOR
177       STORE 1+1 TO 1
178     ENDFOR
179   ENDFOR
180   GO TOP
181   SELECT mat02
182   SET ORDER TO 1
183   ZAP
184   FOR i=1 TO mnvy
185     APPEND BLANK
186     REPLA y WITH i
187     STORE 1 TO K
188     FOR j=1 TO mnvx
189       STORE 'X'+ALLTRIM(STR(K,2)) TO macro1
190       STORE K+1 TO K
191       STORE 'X'+ALLTRIM(STR(K,2)) TO macro2
192       REPLA &macro1 WITH mat2(i,j)
193       REPLA &macro2 WITH mat2(i,j)
194       STORE K+1 TO K
195     ENDFOR
196   ENDFOR
197   GO TOP
198   CLEAR
199   @01,01 SAY 'GENERANDO MATRIZ DE CARGAS SOBRE LAS VIGAS...'
200   RELEASE mat1,mat2,aux
201   SELECT mat03
202   SET ORDER TO 1
203   ZAP
204   APPEND BLANK
205   REPLA y WITH 1
206   FOR j=1 TO mnvx
207     STORE 'X'+ALLTRIM(STR(j,2)) TO MACRO
208     SELECT mat01
209     SET ORDER TO 1
210     STORE &macro TO maux
211     SELECT mat03
212     SET ORDER TO 1
213     REPLA &macro WITH maux
214   ENDFOR
215   SELECT mat01
216   SET ORDER TO 1
217   SKIP
218   FOR i=2 TO mnvy
219     SELECT mat03
220     SET ORDER TO 1
221     APPEND BLANK
222     REPLA y WITH i
223     FOR j=1 TO mnvx
224       STORE 'X'+ALLTRIM(STR(j,2)) TO MACRO

```

```

225      SELECT mat01
226      SET ORDER TO 1
227      STORE &macro TO maux
228      SELECT mat03
229      SET ORDER TO 1
230      REPLA &macro WITH maux
231  —ENDFOR
232      SELECT mat01
233      SET ORDER TO 1
234      SKIP
235      FOR j=1 TO mnvx
236          STORE 'X'+ALLTRIM(STR(j,2)) TO MACRO
237          SELECT mat01
238          SET ORDER TO 1
239          STORE &macro TO maux
240          SELECT mat03
241          SET ORDER TO 1
242          REPLA &macro WITH &macro+maux
243  —ENDFOR
244      SELECT mat01
245      SET ORDER TO 1
246      SKIP
247      SELECT mat03
248      SET ORDER TO 1
249  —ENDFOR
250      SELECT mat03
251      SET ORDER TO 1
252      APPEND BLANK
253      REPLA y WITH mnvy+1
254      FOR j=1 TO mnvx
255          STORE 'X'+ALLTRIM(STR(j,2)) TO MACRO
256          SELECT mat01
257          SET ORDER TO 1
258          STORE &macro TO maux
259          SELECT mat03
260          SET ORDER TO 1
261          REPLA &macro WITH maux
262  —ENDFOR
263      GO TOP
264      SELECT mat04
265      SET ORDER TO 1
266      ZAP
267      FOR i=1 TO mnvy
268          APPEND BLANK
269          REPLA y WITH i
270          STORE 1 TO j
271          STORE 'X'+ALLTRIM(STR(j,2)) TO MACRO
272          SELECT mat02
273          SET ORDER TO 1
274          STORE &macro TO maux
275          SELECT mat04
276          SET ORDER TO 1
277          REPLA &macro WITH maux
278          STORE 2 TO K
279      —FOR j=2 TO mnvx
280          STORE 'X'+ALLTRIM(STR(K,2)) TO macrol

```

```

281      STORE K+1 TO K
282      STORE 'X'+ALLTRIM(STR(K,2)) TO macro2
283      SELECT mat02
284      SET ORDER TO 1
285      STORE &macro1 TOiaux1
286      STORE &macro2 TOiaux2
287      SELECT mat04
288      SET ORDER TO 1
289      STORE 'X'+ALLTRIM(STR(j,2)) TO MACRO
290      REPLA &macro WITHiaux1+iaux2
291      STORE K+1 TO K
292  └─ENDFOR
293      STORE mnvx*2 TO j
294      STORE 'X'+ALLTRIM(STR(j,2)) TO MACRO
295      SELECT mat02
296      SET ORDER TO 1
297      STORE &macro TOiaux
298      SELECT mat04
299      SET ORDER TO 1
300      STORE mnvx+1 TO j
301      STORE 'X'+ALLTRIM(STR(j,2)) TO MACRO
302      REPLA &macro WITHiaux
303      SELECT mat02
304      SET ORDER TO 1
305      SKIP
306      SELECT mat04
307      SET ORDER TO 1
308  └─ENDFOR
309      GO TOP
310      CLEAR
311      @01,01 SAY 'CALCULO DEL MOMENTO FLEXIONANTE MAXIMO...'
312      SELECT mat03
313      SET ORDER TO 1
314      GO TOP
315      DIMENSION aux1(RECCOUNT()*mnvx)
316      STORE 0 TO aux1
317      STORE 1 TO K
318      FOR i=1 TO RECCOUNT()
319  └─FOR j=1 TO mnvx
320      STORE 'X'+ALLTRIM(STR(j,2)) TO MACRO
321      STORE (&macro*X(j)^2)/12 TO aux1(K)
322      STORE K+1 TO K
323  └─ENDFOR
324      SKIP
325  └─ENDFOR
326      SELECT mat04
327      SET ORDER TO 1
328      GO TOP
329      DIMENSION aux2(RECCOUNT()*mnvy)
330      STORE 0 TO aux2
331      STORE 1 TO l
332      STORE mnvy TO K
333      FOR i=1 TO RECCOUNT()
334  └─FOR j=1 TO mnvy
335      STORE 'X'+ALLTRIM(STR(j,2)) TO MACRO
336      STORE (&macro*y(K)^2)/12 TO aux2(l)

```

```

337     STORE l+1 TO 1
338     ENDFOR
339     STORE K-1 TO K
340     SKIP
341     ENDFOR
342     STORE aux1(1) TO lm
343     FOR i=1 TO (RECCOUNT()*mnvx)-1
344         IF lm <= aux1(i+1)
345             STORE aux1(i+1) TO lm
346         ENDIF
347     ENDFOR
348     STORE lm TO may1
349     STORE aux2(1) TO lm
350     FOR i=1 TO (RECCOUNT()*mnvy)-1
351         IF lm <= aux2(i+1)
352             STORE aux2(i+1) TO lm
353         ENDIF
354     ENDFOR
355     STORE lm TO may2
356     IF may1 > may2
357         STORE may1 TO lm
358     ELSE
359         STORE may2 TO lm
360     ENDIF
361     SELECT predise
362     SET ORDER TO 1
363     REPLA m1 WITH lm
364     CLEAR
365     @01,01 SAY 'CALCULO DEL PERALTE DE LA VIGA...'
366     RELEASE aux1, aux2
367     SELECT mat03
368     SET ORDER TO 1
369     GO TOP
370     DIMENSION aux1(RECCOUNT()*mnvx)
371     STORE 0 TO aux1
372     STORE 1 TO K
373     FOR i=1 TO RECCOUNT()
374         FOR j=1 TO mnvx
375             STORE 'X'+ALLTRIM(STR(j,2)) TO MACRO
376             STORE (&macro*X(j)^2)/12 TO m1
377             STORE m1 * 100 TO m2
378             STORE (m2/(mr*mbv))^(1/2) TO D
379             STORE D TO aux1(K)
380             STORE 1+K TO K
381         ENDFOR
382     SKIP
383     ENDFOR
384     SELECT mat04
385     SET ORDER TO 1
386     GO TOP
387     DIMENSION aux2(RECCOUNT()*mnvy)
388     STORE 0 TO aux2
389     STORE 1 TO 1
390     STORE mnvy TO K
391     FOR i=1 TO RECCOUNT()
392         FOR j=1 TO mnvy

```

```

393      STORE 'X'+ALLTRIM(STR(j,2)) TO MACRO
394      STORE (&macro*y(K)^2)/12 TO m1
395      STORE m1 * 100 TO m2
396      STORE (m2/(mr*mbv))^(1/2) TO D
397      STORE D TO aux2(1)
398      STORE l+1 TO l
399  _ENDFOR
400      STORE K-1 TO K
401      SKIP
402  _ENDFOR
403      STORE aux1(1) TO lm
404      FOR i=1 TO (RECCOUNT()*mnvx)-1
405      _IF lm <= aux1(i+1)
406      _STORE aux1(i+1) TO lm
407      _ENDIF
408  _ENDFOR
409      STORE lm TO may1
410      STORE aux2(1) TO lm
411      FOR i=1 TO (RECCOUNT()*mnvy)-1
412      _IF lm <= aux2(i+1)
413      _STORE aux2(i+1) TO lm
414      _ENDIF
415  _ENDFOR
416      STORE lm TO may2
417      _IF may1 > may2
418      _STORE may1 TO lm
419      _ELSE
420      _STORE may2 TO lm
421      _ENDIF
422      SELECT predise
423      SET ORDER TO 1
424      REPLA D WITH lm
425      CLEAR
426      @01,01 SAY ' CALCULO DE LA CARGA SOBRE LAS COLUMNAS...'
427      RELEASE aux1, aux2
428      DIMENSION aux1((mnvy+1)*(mnvx+1)), aux2((mnvy+1)*(mnvx+1)), aux3(mnvy+1,mnvx+1)
429      STORE 0 TO aux1,aux2, aux3
430      STORE 0 TO mpc
431      STORE 1 TO l, o. m
432      FOR j=mnvy TO 0 STEP -1
433      _STORE 1 TO K.N
434      _FOR i=mnvx+1 TO 1 STEP -1
435      _STORE 0 TO mx1,mx2,my1,my2
436      _IF i-1 = 0
437      _STORE 0 TO mx1
438      _ELSE
439      _STORE X(i-1) TO mx1
440      _ENDIF
441      _IF i=mnvx+1
442      _STORE 0 TO mx2
443      _ELSE
444      _STORE X(i) TO mx2
445      _ENDIF
446      _STORE y(j+1) TO my1
447      _IF j=0
448      _STORE 0 TO my2

```

```

449      ELSE
450          STORE y(j) TO my2
451      ENDIF
452      STORE ((mx1+mx2)*(my1+my2)/4)*mcu*mp TO mpc
453      STORE mpc TO aux1(o), aux3(m,N)
454      STORE o+1 TO o
455      STORE K + 1 TO K
456      STORE N + 1 TO N
457  ENDFOR
458      STORE l + 1 TO l
459      STORE m + 1 TO m
460  ENDFOR
461      SELECT mat19
462      SET ORDER TO 1
463      ZAP
464      FOR i=1 TO mnvy+1
465          APPEND BLANK
466          REPLA y WITH i
467          STORE 1 TO K
468          FOR j=mnvx+1 TO 1 STEP -1
469              STORE 'X'+ALLTRIM(STR(K,2)) TO MACRO
470              REPLA &macro WITH aux3(i,j)
471              STORE K + 1 TO K
472          ENDFOR
473      ENDFOR
474      RELEASE aux3
475      STORE aux1(1) TO lm
476      FOR i=1 TO (mnvy*(mnvx+1))-1
477          IF lm <= aux1(i+1)
478              STORE aux1(i+1) TO lm
479          ENDIF
480      ENDFOR
481      SELECT predise
482      SET ORDER TO 1
483      REPLA pc WITH lm
484      CLEAR
485      @01.01 SAY ' CALCULO DEL PERALTE DE COLUMNAS...'
486      STORE 0 TO mpc, mh
487      STORE 1 TO l, o
488      FOR j=mnvy TO 0 STEP -1
489          STORE 1 TO K
490          FOR i=mnvx+1 TO 1 STEP -1
491              STORE 0 TO mx1,mx2,my1,my2
492              IF i-1 = 0
493                  STORE 0 TO mx1
494              ELSE
495                  STORE X(i-1) TO mx1
496              ENDIF
497              IF i=mnvx+1
498                  STORE 0 TO mx2
499              ELSE
500                  STORE X(i) TO mx2
501              ENDIF
502              STORE y(j+1) TO my1
503              IF j=0
504                  STORE 0 TO my2

```

```

505      ELSE
506          STORE y(j) TO my2
507      ENDIF
508          STORE ((mx1+mx2)*(my1+my2)/4)*mcu*anp TO mpc
509          STORE 0 TO mag
510          STORE mpc/(mbeta*(0.25*mfc+mph*0.5*mfy)) TO mag
511          STORE mag/mbc TO mh
512          STORE mh TO aux2(o)
513          STORE o+1 TO o
514          STORE K + 1 TO K
515      ENDFOR
516          STORE l + 1 TO l
517      ENDFOR
518          STORE aux2(1) TO lm
519          FOR i=1 TO (mnvy*(mnvx+1))-1
520              IF lm <= aux2(i+1)
521                  STORE aux2(i+1) TO lm
522              ENDIF
523          ENDFOR
524          SELECT predise
525          SET ORDER TO 1
526          REPLA H WITH lm
527          DEACTIVATE WINDOW mensajes
528          DEACTIVATE WINDOW reportes
529          RETURN
531      *: EOF: PRED002.ACT

```

```

1  *:*****
2  *:
3  *:      Program: C:\CIVIL\REP0001.PRG
4  *:
5  *:      System: Anal.y Diseño de Estructuras Aporticadas
6  *:      Author: Rommel R. Alemán T. - Marco P. Silva S.
7  *:      Copyright (c) 1993, Pontificia Universidad Catolica del E.
8  *:      Last modified: 05/26/93      16:19
9  *:
10 *:      Called by: MENU003.PRG
11 *:
12 *:      Memory Files: IMPRESORA.MEM
13 *:
14 *:      Documented 05/28/93 at 12:13      FoxDoc version 2.10
15 *:*****
16  RELEASE ALL
17  RESTORE FROM impresora ADDITIVE
18  ACTIVATE WINDOW reportes
19  CLEAR
20  SELECT predise
21  SET ORDER TO 1
22  @00,13 TO 02,55
23  STORE 'DATOS DEL PROYECTO'      TO titulo
24  STORE ROUND((40 - LEN(titulo)) / 2,0) + 14 TO mlon
25  @01,mlon SAY titulo
26  @04,01 SAY "      Proyecto NUMERO : "
27  @04,24 SAY codigo PICTURE "*****"
28  @05,01 SAY "      Nombre del Cliente : "
29  @05,24 SAY proyecto
30  @06,01 SAY "Domicilio de la obra : "
31  @06,24 SAY direccion
32  @07,01 SAY "      Tipo de obra : "
33  @07,24 SAY obra
34  @08,01 SAY "      Ing. Calculista : "
35  @08,24 SAY calculo
36  @09,01 SAY "      Fecha (mm-dd-aa) : "
37  @09,24 SAY fecha
38  STORE SPACE(9) TO reporte
39  @11,01 SAY 'Reporte por Monitor o Impresora : ' GET reporte PICTURE '@m Monitor, Impresora'
40  READ
41  IF LASTKEY()=27
42  |   DEACTIVATE WINDOW reportes
43  <====RETURN
44  |_ENDIF
45  STORE UPPER(reportes) TO reporte
46  ACTIVATE WINDOW impresor
47  CLEAR
48  IF reporte='MONITOR'
49  |   @01,01 SAY 'Generando Reporte ...'
50  |   SET DEVICE TO FILE reporte.txt
51  |   @00,26 SAY "*** DATOS DEL PROYECTO ***"
52  |   @02,05 SAY "      Proyecto NUMERO : "
53  |   @02,28 SAY codigo PICTURE "*****"
54  |   @03,05 SAY "      Nombre del Cliente : "
55  |   @03,28 SAY proyecto
56  |   @04,05 SAY "Domicilio de la obra : "

```

```

57 @04,28 SAY direccion
58 @05,05 SAY " Tipo de obra : "
59 @05,28 SAY obra
60 @06,05 SAY " Ing. Calculista : "
61 @06,28 SAY calculo
62 @07,05 SAY " Fecha (mm-dd-aa) : "
63 @07,28 SAY fecha
64 @09,00 SAY SPACE(26)+"** DATOS PARA PREDISEÑO **"+SPACE(26) COLOR SCHEME 7
65 @11,05 SAY " f'c del hormigón: "
66 @11,28 SAY fc PICTURE "### Kg/cm²"
67 @12,05 SAY " fy del acero: "
68 @12,28 SAY fy PICTURE "### Kg/cm²"
69 @13,05 SAY " Nº vanos X: "
70 @13,28 SAY nvx PICTURE "##"
71 @14,05 SAY " Nº vanos Y: "
72 @14,28 SAY nvy PICTURE "##"
73 @15,05 SAY " Espesor loseta: "
74 @15,28 SAY el PICTURE "#.### m"
75 @16,05 SAY " Peso bloques: "
76 @16,28 SAY pb PICTURE "#.### Tn/m²"
77 @17,05 SAY " Peso paredes: "
78 @17,28 SAY pp PICTURE "#.### Tn/m²"
79 @18,05 SAY " Espesor acabado: "
80 @18,28 SAY ea PICTURE "#.### m"
81 @19,05 SAY " Ancho viga: "
82 @19,28 SAY bv PICTURE "### cm"
83 @20,05 SAY " Ancho columna: "
84 @20,28 SAY bc PICTURE "### cm"
85 @21,05 SAY " Nº de pisos: "
86 @21,28 SAY np PICTURE "##"
87 @22,05 SAY " Altura de losa: "
88 @22,28 SAY hl PICTURE "## cm"
89 @23,05 SAY " Carga muerta: "
90 @23,28 SAY cm PICTURE "### Kg/m²"
91 @24,05 SAY " Carga viva: "
92 @24,28 SAY cv PICTURE "### Kg/m²"
93 @25,05 SAY " Carga sísmica: "
94 @25,28 SAY cs PICTURE "### Kg/m²"
95 @26,05 SAY " Carga Última: "
96 @26,28 SAY cu PICTURE "### Kg/m²"
97 @27,05 SAY " % de acero mínimo: "
98 @27,28 SAY ph PICTURE "#.###"
99 @28,05 SAY " Cte.Esfuerzo Unit.: "
100 @28,28 SAY R PICTURE "###"
101 SET DEVICE TO SCREEN
102 MODI COMM reporte.txt NOEDIT WINDOW trabajo
103 ELSE
104 CLEAR
105 @01,01 SAY 'Imprimiendo ...'
106 SET DEVICE TO PRINT no EJECT
107 @00,13 SAY &nexp+"** DATOS DEL PROYECTO **"+&nex
108 @02,05 SAY " Proyecto NUMERO : "
109 @02,28 SAY codigo PICTURE "###"
110 @03,05 SAY " Nombre del Cliente : "
111 @03,28 SAY proyecto
112 @04,05 SAY "Domicilio de la obra : "

```

```

113 @04,28 SAY direccion
114 @05,05 SAY " Tipo de obra : "
115 @05,28 SAY obra
116 @06,05 SAY " Ing. Calculista : "
117 @06,28 SAY calculo
118 @07,05 SAY " Fecha (mm-dd-aa) : "
119 @07,28 SAY fecha
120 @09,00 SAY SPACE(26)+"** DATOS PARA PREDISEÑO **"+SPACE(26) COLOR SCHEME 7
121 @11,05 SAY " f'c del hormigón: "
122 @11,28 SAY fc PICTURE "### Kg/cm²"
123 @12,05 SAY " fy del acero: "
124 @12,28 SAY fy PICTURE "#### Kg/cm²"
125 @13,05 SAY " N° vanos X: "
126 @13,28 SAY nvx PICTURE "##"
127 @14,05 SAY " N° vanos Y: "
128 @14,28 SAY nvy PICTURE "##"
129 @15,05 SAY " Espesor loseta: "
130 @15,28 SAY el PICTURE "#.### m"
131 @16,05 SAY " Peso bloques: "
132 @16,28 SAY pb PICTURE "#.### Tn/m²"
133 @17,05 SAY " Peso paredes: "
134 @17,28 SAY pp PICTURE "#.### Tn/m²"
135 @18,05 SAY " Espesor acabado: "
136 @18,28 SAY ea PICTURE "#.### m"
137 @19,05 SAY " Ancho viga: "
138 @19,28 SAY bv PICTURE "### cm"
139 @20,05 SAY " Ancho columna: "
140 @20,28 SAY bc PICTURE "### cm"
141 @21,05 SAY " N° de pisos: "
142 @21,28 SAY np PICTURE "##"
143 @22,05 SAY " Altura de losa: "
144 @22,28 SAY hl PICTURE "## cm"
145 @23,05 SAY " Carga muerta: "
146 @23,28 SAY cm PICTURE "#### Kg/m²"
147 @24,05 SAY " Carga viva: "
148 @24,28 SAY cv PICTURE "#### Kg/m²"
149 @25,05 SAY " Carga sísmica: "
150 @25,28 SAY cs PICTURE "### Kg/m²"
151 @26,05 SAY " Carga Última: "
152 @26,28 SAY cu PICTURE "#### Kg/m²"
153 @27,05 SAY " % de acero mínimo: "
154 @27,28 SAY ph PICTURE "#.###"
155 @28,05 SAY " Cte.Esfuerzo Unit.: "
156 @28,28 SAY R PICTURE "##.##"
157 EJECT
158 SET DEVICE TO SCREEN
159 _ENDIF
160 DEACTIVATE WINDOW impresor
161 DEACTIVATE WINDOW reportes
162 RETURN
163 *: EOF: REPO001.ACT

```

```

1  *:*****
2  *:
3  *:      Program: C:\CIVIL\REP0002.PRG
4  *:
5  *:      System: Anal.y Diseño de Estructuras Aporticadas
6  *:      Author: Rommel R. Alemán T. - Marco P. Silva S.
7  *:      Copyright (c) 1993, Pontificia Universidad Catolica del E.
8  *:      Last modified: 05/26/93      16:19
9  *:
10 *:      Called by: MENU003.PRG
11 *:
12 *:      Memory Files: IMPRESORA.MEM
13 *:
14 *:      Documented 05/28/93 at 12:13      FoxDoc version 2.10
15 *:*****
16  RELEASE ALL
17  RESTORE FROM impresora ADDITIVE
18  ACTIVATE WINDOW reportes
19  CLEAR
20  SELECT predise
21  SET ORDER TO 1
22  IF nvx=0 .OR. nvy=0 .OR. fy=0 .OR. el=0 .OR. pb=0 .OR. pp=0 .OR. ea=0
23  |   WAIT 'ERROR: Datos incompatibles o no existen...' WINDOW
24  |   DEACTIVATE WINDOW reportes
25  <=====RETURN
26  └─ENDIF
27  @00,13 TO 02,55
28  STORE 'MOSAICO DE CARGAS'      TO titulo
29  STORE ROUND((40 - LEN(titulo)) / 2,0) + 14 TO mlon
30  @01,mlon SAY titulo
31  @04,01 SAY "      Proyecto NUMERO :"
32  @04,24 SAY codigo PICTURE "*****"
33  @05,01 SAY "      Nombre del Cliente :"
34  @05,24 SAY proyecto
35  @06,01 SAY "Domicilio de la obra :"
36  @06,24 SAY direccion
37  @07,01 SAY "      Tipo de obra :"
38  @07,24 SAY obra
39  @08,01 SAY "      Ing. Calculista :"
40  @08,24 SAY calculo
41  @09,01 SAY "      Fecha (mm-dd-aa) :"
42  @09,24 SAY fecha
43  STORE SPACE(9) TO reporte
44  @11,01 SAY 'Reporte por Monitor o Impresora :` GET reporte PICTURE `@m Monitor, Impresora`
45  READ
46  |   IF LASTKEY()=27
47  |   |   DEACTIVATE WINDOW reportes
48  |   <=====RETURN
49  └─ENDIF
50  STORE UPPER(reportes) TO reporte
51  ACTIVATE WINDOW impresor
52  CLEAR
53  STORE nvx TO mnvx
54  STORE nvy TO mnvy
55  STORE proyecto TO mproyecto
56  STORE direccion TO mdireccion

```

```

57  STORE calculo    TO mcalculo
58  STORE fecha     TO mfecha
59  IF reporte='MONITOR'
60      @01,01 SAY 'Generando Reporte ...'
61      SET DEVICE TO FILE reporte.txt
62      SELECT mat01
63      SET ORDER TO 1
64      GO TOP
65      STORE 0 TO i,j
66      @00,01 SAY 'MOSAICO DE CARGAS'
67      @02,01 SAY 'PROYECTO DE CONSTRUCCION: '+mproyecto
68      @03,01 SAY '                               DIRECCION: '+mdireccion
69      @04,01 SAY '                               CALCULO: '+mcalculo
70      @05,01 SAY '                               FECHA: '+DTC(mfecha)
71      @07,01 SAY 'CARGAS SOBRE LAS VIGAS SENTIDO X-X'
72      @08,01 SAY '-----'
73      FOR i=1 TO mnvx+1
74          @10,(i)*10+5 SAY CHR(64+i)
75      ENDFOR
76      STORE 1 TO K
77      FOR i=1 TO RECCOUNT()
78          IF MOD(i,2)#0
79              @10+i,05 SAY K PICTURE '##' FUNCTION '1'
80              STORE K + 1 TO K
81          ENDIF
82          FOR j=1 TO mnvx
83              STORE 'X' TO mac1
84              IF j<10
85                  STORE STR(j,1) TO mac2
86              ELSE
87                  STORE STR(j,2) TO mac2
88              ENDIF
89              STORE ALLTRIM(mac1)+ALLTRIM(mac2) TO MACRO
90              @10+i,(j)*10 SAY &macro PICTURE '#####.##'
91          ENDFOR
92          SKIP
93      ENDFOR
94      SELECT mat02
95      SET ORDER TO 1
96      GO TOP
97      STORE 0 TO i,j
98      @01,01 SAY 'CARGAS SOBRE LAS VIGAS SENTIDO Y-Y'
99      @02,01 SAY '-----'
100     STORE 1 TO K
101     FOR i=1 TO mnvx*2
102         IF i<=1
103             @04,(i)*10+5 SAY CHR(64+i)
104         ELSE
105             IF MOD(i,2)=0
106                 STORE K+1 TO K
107             ENDIF
108             @04,(i)*10+5 SAY CHR(64+K)
109         ENDIF
110     ENDFOR
111     FOR i=1 TO RECCOUNT()
112         @05+i,05 SAY i PICTURE '##' FUNCTION 'L'

```

```

113   FOR j=1 TO mnvx*2
114       STORE 'X' TO mac1
115       IF j<10
116           STORE STR(j,1) TO mac2
117       ELSE
118           STORE STR(j,2) TO mac2
119       ENDIF
120       STORE ALLTRIM(mac1)+ALLTRIM(mac2) TO MACRO
121       @05+i,(j)*10 SAY &macro PICTURE '#####.##'
122   ENDFOR
123   SKIP
124 ENDFOR
125 SET DEVICE TO SCREEN
126 MODI COMM reporte.txt NOEDIT WINDOW trabajo
127 ELSE
128     CLEAR
129     @01,01 SAY 'Imprimiendo ...'
130     SET DEVICE TO PRINT no EJECT
131     SET PRINT ON
132     SELECT mat01
133     SET ORDER TO 1
134     GO TOP
135     STORE 0 TO i,j
136     ? &mexp+'MOSAICO DE CARGAS'+&mnex AT 22
137     ?
138     ? 'PROYECTO DE CONSTRUCCION: '+mproyecto
139     ? '          DIRECCION: '+mdireccion
140     ? '          CALCULO: '+mcalculo
141     ? '          FECHA: '+DTC(mfecha)
142     ?
143     ?
144     ? 'CARGAS SOBRE LAS VIGAS SENTIDO X-X (Kg/m)' AT 22
145     ? '-----' AT 22
146     ? &mcom
147     FOR i=1 TO mnvx+1
148         ?? CHR(64+i) AT (i)*10+5
149     ENDFOR
150     STORE 1 TO K
151     FOR i=1 TO RECCOUNT()
152         ?
153         IF MOD(i,2)#0
154             ?? K PICTURE '##' FUNCTION '1' AT 05
155             STORE K + 1 TO K
156         ENDIF
157         FOR j=1 TO mnvx
158             STORE 'X' TO mac1
159             IF j<10
160                 STORE STR(j,1) TO mac2
161             ELSE
162                 STORE STR(j,2) TO mac2
163             ENDIF
164             STORE ALLTRIM(mac1)+ALLTRIM(mac2) TO MACRO
165             ?? &macro PICTURE '#####.##' AT (j)*10
166         ENDFOR
167         IF i>50
168             EJECT

```

```

169      └─ENDIF
170      SKIP
171      └─ENDFOR
172      SELECT mat02
173      SET ORDER TO 1
174      GO TOP
175      STORE 0 TO i,j
176      ? &mnco
177      ?
178      ? 'CARGAS SOBRE LAS VIGAS SENTIDO Y-Y (Kg/m)' AT 22
179      ? '-----' AT 22
180      ? &mcom
181      STORE 1 TO K
182      FOR i=1 TO mnvx*2
183      └─IF i<=1
184      └─?? CHR(64+i) AT (i)*10+5
185      └─ELSE
186      └─IF MOD(i,2)=0
187      └─STORE K+1 TO K
188      └─ENDIF
189      └─?? CHR(64+K) AT (i)*10+5
190      └─ENDIF
191      └─ENDFOR
192      FOR i=1 TO RECCOUNT()
193      ?
194      ?? i PICTURE '**' FUNCTION 'L' AT 05
195      └─FOR j=1 TO mnvx*2
196      └─STORE 'X' TO mac1
197      └─IF j<10
198      └─STORE STR(j,1) TO mac2
199      └─ELSE
200      └─STORE STR(j,2) TO mac2
201      └─ENDIF
202      └─STORE ALLTRIM(mac1)+ALLTRIM(mac2) TO MACRO
203      └─?? &macro PICTURE '#####.##' AT (j)*10
204      └─ENDFOR
205      └─IF i>50
206      └─EJECT
207      └─ENDIF
208      SKIP
209      └─ENDFOR
210      ? &mnco
211      EJECT
212      SET PRINT OFF
213      SET DEVICE TO SCREEN
214      └─ENDIF
215      DEACTIVATE WINDOW impresor
216      DEACTIVATE WINDOW reportes
217      RETURN
218      *: EOF: REPO002.ACT

```

```

1  *:*****
2  *:
3      Program: C:\CIVIL\BEPO003.PRG
4  *:
5      System: Anal.y Diseño de Estructuras Aporticadas
6  *:      Author: Rommel R. Alemán T. - Marco P. Silva S.
7  *:      Copyright (c) 1993, Pontificia Universidad Catolica del E.
8  *:      Last modified: 05/26/93      16:41
9  *:
10 *:      Called by: MENU003.PRG
11 *:
12 *:      Memory Files: IMPRESORA.MEM
13 *:
14 *:      Documented 05/28/93 at 12:13      FoxDoc version 2.10
15 *:*****
16     RELEASE ALL
17     RESTORE FROM impresora ADDITIVE
18     ACTIVATE WINDOW reportes
19     CLEAR
20     SELECT predise
21     SET ORDER TO 1
22     IF nvx=0 .OR. nvy=0 .OR. fy=0 .OR. el=0 .OR. pb=0 .OR. pp=0 .OR. ea=0
23     |     WAIT "ERROR: Datos incompatibles o no existen..." WINDOW
24     |     DEACTIVATE WINDOW reportes
25     <=====RETURN
26     _ENDIF
27     @00,13 TO 02,55
28     STORE "CARGA SOBRE LAS VIGAS"      TO titulo
29     STORE ROUND((40 - LEN(titulo)) / 2,0) + 14 TO mlon
30     @01,mlon SAY titulo
31     @04,01 SAY "      Proyecto NUMERO :"
32     @04,24 SAY codigo PICTURE "####"
33     @05,01 SAY "      Nombre del Cliente :"
34     @05,24 SAY proyecto
35     @06,01 SAY "Domicilio de la obra :"
36     @06,24 SAY direccion
37     @07,01 SAY "      Tipo de obra :"
38     @07,24 SAY obra
39     @08,01 SAY "      Ing. Calculista :"
40     @08,24 SAY calculo
41     @09,01 SAY "      Fecha (mm-dd-aa) :"
42     @09,24 SAY fecha
43     STORE SPACE(9) TO reporte
44     @11,01 SAY "Reporte por Monitor o Impresora : " GET reporte PICTURE "@m Monitor, Impresora"
45     READ
46     IF LASTKEY()=27
47     |     DEACTIVATE WINDOW reportes
48     <=====RETURN
49     _ENDIF
50     STORE UPPER(reportes) TO reporte
51     ACTIVATE WINDOW impresor
52     CLEAR
53     STORE nvx TO mnvx
54     STORE nvy TO mnvy
55     STORE proyecto TO mproyecto
56     STORE direccion TO mdireccion

```

```

57     STORE calculo    TO mcalculo
58     STORE fecha      TO mfecha
59     IF reporte='MONITOR'
60         @01.01 SAY 'Generando Reporte ...'
61         SET DEVICE TO FILE reporte.txt
62         SET DEVICE TO SCREEN
63         MODI COMM reporte.txt NORDIT WINDOW trabajo
64     ELSE
65         CLEAR
66         @01.01 SAY 'Imprimiendo ...'
67         SET DEVICE TO PRINT no EJECT
68         SET PRINT ON
69         SELECT mat03
70         SET ORDER TO 1
71         GO TOP
72         STORE 0 TO i,j
73         ? &mexp+'CARGA SOBRE LAS VIGAS'+&mnex AT 18
74         ?
75         ? 'PROYECTO DE CONSTRUCCION: '+mproyecto
76         ? '                               DIRECCION: '+mdireccion
77         ? '                               CALCULO: '+mcalculo
78         ? '                               FECHA: '+DTC(mfecha)
79         ?
80         ?
81         ? 'CARGA SOBRE LAS VIGAS SENTIDO X-X (Kg/m)' AT 22
82         ? '-----' AT 22
83         ? &mcom
84         FOR i=1 TO mnvx+1
85             ?? CHR(64+i) AT (i)*10+5
86         ENDFOR
87         FOR i=1 TO RECCOUNT()
88             ?
89             ?? i PICTURE '##' FUNCTION '1' AT 05
90             FOR j=1 TO mnvx
91                 STORE 'X' TO mac1
92                 IF j<10
93                     STORE STR(j,1) TO mac2
94                 ELSE
95                     STORE STR(j,2) TO mac2
96                 ENDIF
97                 STORE ALLTRIM(mac1)+ALLTRIM(mac2) TO MACRO
98                 ?? &macro PICTURE '#####.##' AT (j)*10
99             ENDFOR
100            IF i>50
101                EJECT
102            ENDIF
103            SKIP
104        ENDFOR
105        SELECT mat04
106        SET ORDER TO 1
107        GO TOP
108        STORE 0 TO i,j
109        ? &mnco
110        ?
111        ? 'CARGA SOBRE LAS VIGAS SENTIDO Y-Y (Kg/m)' AT 22
112        ? '-----' AT 22

```

```

113      ? &macro
114      FOR i=1 TO mny
115          ?? CHR(64+i) AT (i)*10+5
116      ENDFOR
117      FOR i=1 TO RECCOUNT()
118          ?
119          ?? i PICTURE '##' FUNCTION 'L' AT 05
120          FOR j=1 TO mny
121              STORE 'X' TO mac1
122              IF j<10
123                  STORE STR(j,1) TO mac2
124              ELSE
125                  STORE STR(j,2) TO mac2
126              ENDIF
127              STORE ALTRIM(mac1)+ALTRIM(mac2) TO MACRO
128              ?? &macro PICTURE '#####' AT (j)*10
129          ENDFOR
130          IF i>50
131              EJECT
132          ENDIF
133          SKIP
134      ENDFOR
135      ? &macro
136      EJECT
137      SET PRINT OFF
138      SET DEVICE TO SCREEN
139  ENDIF
140  DEACTIVATE WINDOW impresor
141  DEACTIVATE WINDOW reportes
142  RETURN
143  *: EOF: REP0003.ACT
144

```

```

1  *:*****
2  *:
3  *:      Program: C:\CIVIL\REPO004.PRG
4  *:
5  *:      System: Anal.y Diseño de Estructuras Aporticadas
6  *:      Author: Rommel R. Alemán T. - Marco P. Silva S.
7  *:      Copyright (c) 1993, Pontificia Universidad Catolica del E.
8  *:      Last modified: 05/09/93      16:33
9  *:
10 *:      Called by: MENU003.PRG
11 *:
12 *:      Memory Files: IMPRESORA.MEM
13 *:                  : &NOM01
14 *:                  : &NOM02
15 *:                  : &NOM03
16 *:
17 *:      Documented 05/28/93 at 12:13      FoxDoc version 2.10
18 *:*****
19  RELEASE ALL
20  RESTORE FROM impresora ADDITIVE
21  ACTIVATE WINDOW reportes
22  CLEAR
23  SELECT predise
24  SET ORDER TO 1
25  IF nvx=0 .OR. nvy=0 .OR. fy=0 .OR. el=0 .OR. pb=0 .OR. pp=0 .OR. ea=0
26  WAIT "ERROR: Datos incompatibles o no existen..." WINDOW
27  DEACTIVATE WINDOW reportes
28  RETURN
29  ENDIF
30  @00.13 TO 02.55
31  STORE "MOMENTO FLEXIONANTE MAXIMO" TO titulo
32  STORE ROUND((40 - LEN(titulo)) / 2.0) + 14 TO mlon
33  @01,mlon SAY titulo
34  @04.01 SAY "      Proyecto NUMERO :"
35  @04.24 SAY codigo PICTURE "*****"
36  @05.01 SAY "      Nombre del Cliente :"
37  @05.24 SAY proyecto
38  @06.01 SAY "Domicilio de la obra :"
39  @06.24 SAY direccion
40  @07.01 SAY "      Tipo de obra :"
41  @07.24 SAY obra
42  @08.01 SAY "      Ing. Calculista :"
43  @08.24 SAY calculo
44  @09.01 SAY "      Fecha (mm-dd-aa) :"
45  @09.24 SAY fecha
46  STORE SPACE(9) TO reporte
47  @11.01 SAY "Reporte por Monitor o Impresora : " GET reporte PICTURE "@m Monitor, Impresora"
48  READ
49  DEACTIVATE WINDOW reportes
50  IF LASTKEY()=27
51  RETURN
52  ENDIF
53  STORE UPPER(reportes) TO reporte
54  ACTIVATE WINDOW impresor
55  CLEAR
56  STORE codigo TO numero

```

```

57 STORE STR(numero,5) TO mnum
58 STORE 'nvx'+ALLTRIM(mnum) TO nom01
59 STORE 'nvx'+ALLTRIM(mnum) TO nom02
60 STORE 'np'+ALLTRIM(mnum) TO nom03
61 RESTORE FROM &nom01 ADDITIVE
62 RESTORE FROM &nom02 ADDITIVE
63 RESTORE FROM &nom03 ADDITIVE
64 STORE nvx TO mnvx
65 STORE nvx TO mnvy
66 STORE proyecto TO mproyecto
67 STORE direccion TO mdireccion
68 STORE calculo TO mcalculo
69 STORE fecha TO mfecha
70 IF reporte='MONITOR'
71 SET DEVICE TO FILE reporte.txt
72 SET DEVICE TO SCREEN
73 MODI COMM reporte.txt NORDIT WINDOW trabajo
74 ELSE
75 SET DEVICE TO PRINT no EJECT
76 SET PRINT ON
77 ? &mexp+'MOMENTO FLEXIONANTE MAXIMO'+&mnex AT 13
78 ?
79 ? 'PROYECTO DE CONSTRUCCION: '+mproyecto
80 ? ' DIRECCION: '+mdireccion
81 ? ' CALCULO: '+mcalculo
82 ? ' FECHA: '+DTC(mfecha)
83 ?
84 ?
85 ? 'MOMENTO FLEXIONANTE MAXIMO SENTIDO X-X (Kg.m)' AT 20
86 ? '-----' AT 20
87 ? &mcom
88 SELECT mat03
89 SET ORDER TO 1
90 GO TOP
91 FOR i=1 TO mnvx+1
92 ?? CHR(64+i) AT (i)*10+5
93 ENDFOR
94 FOR i=1 TO RECCOUNT()
95 ?
96 ?? i PICTURE '**' FUNCTION '1' AT 05
97 FOR j=1 TO mnvx
98 STORE 'X'+ALLTRIM(STR(j,2)) TO MACRO
99 ?? (&macro*X(j)^2)/12 PICTURE '*****.**' AT (j)*10
100 ENDFOR
101 IF i>50
102 EJECT
103 ENDIF
104 SKIP
105 ENDFOR
106 ? &mco
107 ?
108 ? 'MOMENTO FLEXIONANTE MAXIMO SENTIDO Y-Y (Kg.m)' AT 20
109 ? '-----' AT 20
110 ? &mcom
111 SELECT mat04
112 SET ORDER TO 1

```

```

113      GO TOP
114      FOR i=1 TO mnvy
115          ?? CHR(64+i) AT (i)*10+5
116      ENDFOR
117      STORE mnvy TO K
118      FOR i=1 TO RECCOUNT()
119          ?
120          ?? i PICTURE '##' FUNCTION 'L' AT 05
121          FOR j=1 TO mnvy
122              STORE 'X'+ALLTRIM(STR(j,2)) TO MACRO
123              ?? (&macro*y(K)^2)/12 PICTURE '#####.##' AT (j)*10
124          ENDFOR
125          IF i>50
126              EJECT
127          ENDIF
128          STORE K-1 TO K
129          SKIP
130      ENDFOR
131      ? &mnco
132      EJECT
133      SET PRINT OFF
134      SET DEVICE TO SCREEN
135  ENDFIF
136      DEACTIVATE WINDOW impresor
137      RETURN
139      *: EOF: REPO004.ACT

```

```

1  *:*****
2  *:
3  *:      Program: C:\CIVIL\REPO005.PRG
4  *:
5  *:      System: Anal.y Diseño de Estructuras Aporticadas
6  *:      Author: Rommel R. Alemán T. - Marco P. Silva S.
7  *:      Copyright (c) 1993, Pontificia Universidad Catolica del E.
8  *:      Last modified: 05/09/93      16:33
9  *:
10 *:      Called by: MENU003.PRG
11 *:
12 *:      Memory Files: IMPRESORA.MEM
13 *:                  : &NOM01
14 *:                  : &NOM02
15 *:                  : &NOM03
16 *:
17 *:      Documented 05/28/93 at 12:13                      FoxDoc version 2.10
18 *:*****
19      RELEASE ALL
20      RESTORE FROM impresora ADDITIVE
21      ACTIVATE WINDOW reportes
22      CLEAR
23      SELECT predise
24      SET ORDER TO 1
25      IF nvx=0 .OR. nvy=0 .OR. R=0 .OR. bv=0
26      |      WAIT "ERROR: Datos incompatibles o no existen..." WINDOW
27      |      DEACTIVATE WINDOW reportes
28  <=====RETURN
29  └─ENDIF
30      @00,13 TO 02,55
31      STORE "PERALTE DE LA VIGA" TO titulo
32      STORE ROUND((40 - LEN(titulo)) / 2,0) + 14 TO mlon
33      @01,mlon SAY titulo
34      @04,01 SAY "      Proyecto NUMERO :"
35      @04,24 SAY codigo PICTURE "####"
36      @05,01 SAY "      Nombre del Cliente :"
37      @05,24 SAY proyecto
38      @06,01 SAY "Domicilio de la obra :"
39      @06,24 SAY direccion
40      @07,01 SAY "      Tipo de obra :"
41      @07,24 SAY obra
42      @08,01 SAY "      Ing. Calculista :"
43      @08,24 SAY calculo
44      @09,01 SAY "      Fecha (mm-dd-aa) :"
45      @09,24 SAY fecha
46      STORE SPACE(9) TO reporte
47      @11,01 SAY "Reporte por Monitor o Impresora :." GET reporte PICTURE "@m Monitor, Impresora"
48      READ
49      DEACTIVATE WINDOW reportes
50      IF LASTKEY()=27
51  <=====RETURN
52  └─ENDIF
53      STORE UPPER(reportes) TO reporte
54      ACTIVATE WINDOW impresor
55      CLEAR
56      STORE codigo TO numero

```

```

57 STORE STR(numero,5) TO mnum
58 STORE 'nvx'+ALLTRIM(mnum) TO nom01
59 STORE 'nvy'+ALLTRIM(mnum) TO nom02
60 STORE 'np'+ALLTRIM(mnum) TO nom03
61 RESTORE FROM &nom01 ADDITIVE
62 RESTORE FROM &nom02 ADDITIVE
63 RESTORE FROM &nom03 ADDITIVE
64 STORE nvx TO mnvx
65 STORE nvy TO mnvy
66 STORE proyecto TO mproyecto
67 STORE direccion TO mdireccion
68 STORE calculo TO mcalculo
69 STORE fecha TO mfecha
70 STORE R TO mr
71 STORE bv TO mbv
72 IF reporte='MONITOR'
73 SET DEVICE TO FILE reporte.txt
74 SET DEVICE TO SCREEN
75 MODI COMM reporte.txt NORDIT WINDOW trabajo
76 ELSE
77 SET DEVICE TO PRINT no EJECT
78 SET PRINT ON
79 ? &mexp+'PERALTE DE LA VIGA'+&mnex AT 21
80 ?
81 ? 'PROYECTO DE CONSTRUCCION: '+mproyecto
82 ? ' DIRECCION: '+mdireccion
83 ? ' CALCULO: '+mcalculo
84 ? ' FECHA: '+DTC(mfecha)
85 ?
86 ?
87 ? 'PERALTE DE LA VIGA SENTIDO X-X (cm)' AT 24
88 ? '-----' AT 24
89 ? &mcom
90 SELECT mat03
91 SET ORDER TO 1
92 GO TOP
93 FOR i=1 TO mnvx+1
94 ?? CHR(64+i) AT (i)*10+5
95 ENDFOR
96 FOR i=1 TO RECCOUNT()
97 ?
98 ?? i PICTURE '##' FUNCTION '1' AT 05
99 FOR j=1 TO mnvx
100 STORE 'X'+ALLTRIM(STR(j,2)) TO MACRO
101 STORE (&macro*X(j)^2)/12 TO m1
102 STORE m1 * 100 TO m2
103 STORE (m2/(mr*mbv))^(1/2) TO D
104 ?? D PICTURE '#####.##' AT (j)*10
105 ENDFOR
106 IF i>50
107 EJECT
108 ENDIF
109 SKIP
110 ENDFOR
111 ? &mnco
112 ?

```

```

113 ? 'PERALTE DE LA VIGA SENTIDO Y-Y (cm)' AT 24
114 ? '-----' AT 24
115 ? &mcom
116 SELECT mat04
117 SET ORDER TO 1
118 GO TOP
119 FOR i=1 TO mnvy
120     ?? CHR(64+i) AT (i)*10+5
121 ENDFOR
122 STORE mnvy TO K
123 FOR i=1 TO RECCOUNT()
124     ?
125     ?? i PICTURE '##' FUNCTION 'L' AT 05
126     FOR j=1 TO mnvy
127         STORE 'X'+ALLTRIM(STR(j,2)) TO MACRO
128         STORE (&macro*y(K)^2)/12 TO m1
129         STORE m1 * 100 TO m2
130         STORE (m2/(m1*mbv))^(1/2) TO D
131         ?? D PICTURE '#####.##' AT (j)*10
132     ENDFOR
133     IF i>50
134         EJECT
135     ENDIF
136     STORE K-1 TO K
137     SKIP
138 ENDFOR
139 ? &mcco
140 EJECT
141 SET PRINT OFF
142 SET DEVICE TO SCREEN
143 ENDIF
144 DEACTIVATE WINDOW impresor
145 RETURN
147 *: KOF: REPO005.ACT

```

```

1  *:*****
2  *:
3  *:      Program: C:\CIVIL\REP0006.PRG
4  *:
5  *:      System: Anal.y Diseño de Estructuras Aporticadas
6  *:      Author: Rommel R. Alemán T. - Marco P. Silva S.
7  *:      Copyright (c) 1993, Pontificia Universidad Catolica del E.
8  *:      Last modified: 05/09/93      16:34
9  *:
10 *:      Called by: MENU003.PRG
11 *:
12 *:      Memory Files: IMPRESORA.MEM
13 *:                  : &NOM01
14 *:                  : &NOM02
15 *:                  : &NOM03
16 *:
17 *:      Documented 05/28/93 at 12:13      FoxDoc version 2.10
18 *:*****
19  RELEASE ALL
20  RESTORE FROM impresora ADDITIVE
21  ACTIVATE WINDOW reportes
22  CLEAR
23  SELECT predise
24  SET ORDER TO 1
25  IF nvx=0 .OR. nvy=0 .OR. np=0 .OR. cu=0 .OR. beta=0 .OR. fc=0 .OR. ph=0 .OR. fy=0 .OR. bc=0
26  WAIT "ERROR: Datos incompatibles o no existen..." WINDOW
27  DEACTIVATE WINDOW reportes
28  ←RETURN
29  └─ENDIF
30  @00,13 TO 02.55
31  STORE "CARGA SOBRE LAS COLUMNAS" TO titulo
32  STORE ROUND((40 - LEN(titulo)) / 2,0) + 14 TO mlon
33  @01,mlon SAY titulo
34  @04,01 SAY "      Proyecto NUMERO :"
35  @04,24 SAY codigo PICTURE "*****"
36  @05,01 SAY "      Nombre del Cliente :"
37  @05,24 SAY proyecto
38  @06,01 SAY "Domicilio de la obra :"
39  @06,24 SAY direccion
40  @07,01 SAY "      Tipo de obra :"
41  @07,24 SAY obra
42  @08,01 SAY "      Ing. Calculista :"
43  @08,24 SAY calculo
44  @09,01 SAY "      Fecha (mm-dd-aa) :"
45  @09,24 SAY fecha
46  STORE SPACE(9) TO reporte
47  @11,01 SAY "Reporte por Monitor o Impresora :" GET reporte PICTURE "@m Monitor, Impresora"
48  READ
49  DEACTIVATE WINDOW reportes
50  └─IF LASTKEY()=27
51  ←RETURN
52  └─ENDIF
53  STORE UPPER(reporte) TO reporte
54  ACTIVATE WINDOW impresor
55  CLEAR
56  STORE codigo TO numero

```

```

57     STORE STR(numero,5) TO mnum
58     STORE 'nvx'+ALLTRIM(mnum) TO nom01
59     STORE 'nvy'+ALLTRIM(mnum) TO nom02
60     STORE 'np'+ALLTRIM(mnum) TO nom03
61     RESTORE FROM &nom01 ADDITIVE
62     RESTORE FROM &nom02 ADDITIVE
63     RESTORE FROM &nom03 ADDITIVE
64     STORE nvx TO mnvx
65     STORE nvy TO mnvy
66     STORE np TO mnp
67     STORE cu TO mcu
68     STORE beta TO mbeta
69     STORE fc TO mfc
70     STORE ph TO mph
71     STORE fy TO mfy
72     STORE bc TO mbc
73     STORE proyecto TO mproyecto
74     STORE direccion TO mdireccion
75     STORE calculo TO mcalculo
76     STORE fecha TO mfecha
77     IF reporte='MONITOR'
78         SET DEVICE TO FILE reporte.txt
79         SET DEVICE TO SCREEN
80         MODI COMM reporte.txt NOEDIT WINDOW trabajo
81     ELSE
82         SET DEVICE TO PRINT no EJECT
83         SET PRINT ON
84         STORE 0 TO i,j
85         ? &mexp+'CARGA SOBRE LAS COLUMNAS'+&mnex AT 16
86         ?
87         ? 'PROYECTO DE CONSTRUCCION: '+mproyecto
88         ? '                               DIRECCION: '+mdireccion
89         ? '                               CALCULO: '+mcalculo
90         ? '                               FECHA: '+DTC(mfecha)
91         ?
92         ?
93         ? 'CARGA SOBRE LAS COLUMNAS (Kg)' AT 27
94         ? '-----' AT 27
95         ? &mcom
96         FOR i=1 TO mnvx+1
97             ?? CHR(64+i) AT (i)*10+5
98         ENDFOR
99         STORE 0 TO mpc
100        STORE 1 TO l
101        FOR j=mnvy TO 0 STEP -1
102            STORE 1 TO K
103            ?
104            ?? 1 PICTURE '##' FUNCTION '1' AT 05
105            FOR i=mnvx+1 TO 1 STEP -1
106                STORE 0 TO mx1,mx2,my1,my2
107                IF i-1 = 0
108                    STORE 0 TO mx1
109                ELSE
110                    STORE X(i-1) TO mx1
111                ENDIF
112            IF i=mnvx+1

```

```

113      STORE 0    TO mx2
114      ELSE
115      STORE X(i) TO mx2
116      ENDIF
117      STORE y(j+1) TO my1
118      IF j=0
119      STORE 0 TO my2
120      ELSE
121      STORE y(j)  TO my2
122      ENDIF
123      STORE ((mx1+mx2)*(my1+my2)/4)*mcu*mp TO mpc
124      ?? mpc PICTURE '#####.##' AT i*10
125      STORE K + 1 TO K
126      ENDFOR
127      STORE l + 1 TO l
128      IF i>50
129      EJECT
130      ENDIF
131      ENDFOR
132      ? &mnco
133      ?
134      ?
135      ? 'PERALTE DE COLUMNAS (Kg)' AT 29
136      ? '-----' AT 29
137      ? &mcom
138      FOR i=1 TO mnvx+1
139      ?? CHR(64+i) AT (i)*10+5
140      ENDFOR
141      STORE 0 TO mpc, mh
142      STORE 1 TO l
143      FOR j=mnvy TO 0 STEP -1
144      STORE 1 TO K
145      ?
146      ?? l PICTURE '##' FUNCTION 'l' AT 05
147      FOR i=mnvx+1 TO 1 STEP -1
148      STORE 0 TO mx1,mx2,my1,my2
149      IF i-1 = 0
150      STORE 0    TO mx1
151      ELSE
152      STORE X(i-1) TO mx1
153      ENDIF
154      IF i=mnvx+1
155      STORE 0    TO mx2
156      ELSE
157      STORE X(i) TO mx2
158      ENDIF
159      STORE y(j+1) TO my1
160      IF j=0
161      STORE 0 TO my2
162      ELSE
163      STORE y(j)  TO my2
164      ENDIF
165      STORE ((mx1+mx2)*(my1+my2)/4)*mcu*mp TO mpc
166      STORE 0 TO mag
167      STORE mpc/(mbeta*(0.25*mf+mph*0.5*mfy)) TO mag
168      STORE mag/mbc TO mh

```

```

169      ?? mh PICTURE '#####.##' AT i*10
170      STORE K + 1 TO K
171      ENDFOR
172      STORE l + 1 TO l
173      IF i>50
174      EJECT
175      ENDIF
176      ENDFOR
177      ? &mnco
178      EJECT
179      SET PRINT OFF
180      SET DEVICE TO SCREEN
181      ENDIF
182      DEACTIVATE WINDOW impresor
183      RETURN
185      *: KOF: BEP0006.ACT

```

```

1  *:*****
2  *:
3  *:      Program: C:\CIVIL\REPO007.PRG
4  *:
5  *:      System: Anal.y Diseño de Estructuras Aporticadas
6  *:      Author: Rommel R. Alemán T. - Marco P. Silva S.
7  *:      Copyright (c) 1993, Pontificia Universidad Catolica del E.
8  *:      Last modified: 05/26/93      16:11
9  *:
10 *:      Called by: MENU003.PRG
11 *:
12 *:      Memory Files: IMPRESORA.MEM
13 *:
14 *:      Documented 05/28/93 at 12:13      FoxDoc version 2.10
15 *:*****
16      RELEASE ALL
17      IF hl<=0 .OR. D<=0 .OR. ml<=0 .OR. H<=0 .OR. pc<=0 .OR. bp<=0 .OR. ss<=0 .OR. ee<=0
18      WAIT "ERROR: Datos incompatibles o no existen..." WINDOW
19      RETURN
20  ENDIF
21      RESTORE FROM impresora ADDITIVE
22      ACTIVATE WINDOW reportes
23      CLEAR
24      SELECT predise
25      SET ORDER TO 1
26      @00,13 TO 02,55
27      STORE "DATOS DEL PROYECTO" TO titulo
28      STORE ROUND((40 - LEN(titulo)) / 2,0) + 14 TO mlon
29      @01,mlon SAY titulo
30      @04,01 SAY "      Proyecto NUMERO :"
31      @04,24 SAY codigo PICTURE "####"
32      @05,01 SAY "      Nombre del Cliente :"
33      @05,24 SAY proyecto
34      @06,01 SAY "Domicilio de la obra :"
35      @06,24 SAY direccion
36      @07,01 SAY "      Tipo de obra :"
37      @07,24 SAY obra
38      @08,01 SAY "      Ing. Calculista :"
39      @08,24 SAY calculo
40      @09,01 SAY "      Fecha (mm-dd-aa) :"
41      @09,24 SAY fecha
42      STORE SPACE(9) TO reporte
43      @11,01 SAY "Reporte por Monitor o Impresora :" GET reporte PICTURE "@m Monitor, Impresora"
44      READ
45      DEACTIVATE WINDOW reportes
46      IF LASTKEY()=27
47      RETURN
48  ENDIF
49      STORE UPPER(reportes) TO reporte
50      ACTIVATE WINDOW impresor
51      CLEAR
52      IF reportes="MONITOR"
53      @01,01 SAY "Generando Reporte ..."
54      SET DEVICE TO FILE reporte.txt
55      @00,26 SAY "*** DATOS DEL PROYECTO ***"
56      @02,05 SAY "      Proyecto NUMERO :"

```

```

57 @02.28 SAY codigo PICTURE "#####"
58 @03.05 SAY " Nombre del Cliente : "
59 @03.28 SAY proyecto
60 @04.05 SAY "Domicilio de la obra : "
61 @04.28 SAY direccion
62 @05.05 SAY " Tipo de obra : "
63 @05.28 SAY obra
64 @06.05 SAY " Ing. Calculista : "
65 @06.28 SAY calculo
66 @07.05 SAY " Fecha (mm-dd-aa) : "
67 @07.28 SAY fecha
68 @09.00 SAY SPACE(26)+"** DATOS PARA DISEÑO **"
69 @11.05 SAY " Cte.hormigón β: "
70 @11.27 SAY beta PICTURE "###"
71 @12.05 SAY " Cte.flexión viga φ: "
72 @12.27 SAY fi1 PICTURE "###"
73 @13.05 SAY "Cte.flexión columna φ: "
74 @13.27 SAY fi2 PICTURE "###"
75 @14.05 SAY "Cte.corte y torsión τ: "
76 @14.27 SAY fi3 PICTURE "###"
77 @15.05 SAY " Altura de Losa: "
78 @15.27 SAY hl PICTURE "### cm"
79 @16.05 SAY " Peralte de la viga: "
80 @16.27 SAY D PICTURE "### cm"
81 @17.05 SAY " Momento Flexionante: "
82 @17.27 SAY ml PICTURE "#####.### Kg-m"
83 @18.05 SAY "Peralte de la Columna: "
84 @18.27 SAY H PICTURE "### cm"
85 @19.05 SAY " Peso sobre columna: "
86 @19.27 SAY pc PICTURE "#####.### Kg"
87 @20.05 SAY " Ancho del Plinto : "
88 @20.27 SAY bp PICTURE "###.## m"
89 @21.05 SAY " Esfuer.Admi.Suelo δ: "
90 @21.27 SAY ss PICTURE "### Tn/m²"
91 @22.05 SAY " : "
92 @22.27 SAY " : "
93 @23.05 SAY " : "
94 @23.27 SAY " : "
95 @24.05 SAY "Análisis Portico "EJK" : "
96 @24.33 SAY xx PICTURE "99"
97 SET DEVICE TO SCREEN
98 MODI COMM reporte.txt NOEDIT WINDOW trabajo
99 ELSE
100 @01.05 SAY "Imprimiendo ..."
101 SET DEVICE TO PRINT no EJECT
102 @00.13 SAY &mexp+"*** DATOS DEL PROYECTO ***"+&mhex
103 @02.05 SAY " Proyecto NUMERO : "
104 @02.28 SAY codigo PICTURE "#####"
105 @03.05 SAY " Nombre del Cliente : "
106 @03.28 SAY proyecto
107 @04.05 SAY "Domicilio de la obra : "
108 @04.28 SAY direccion
109 @05.05 SAY " Tipo de obra : "
110 @05.28 SAY obra
111 @06.05 SAY " Ing. Calculista : "
112 @06.28 SAY calculo

```

```

113 @07.05 SAY " Fecha (mm-dd-aa) : "
114 @07.28 SAY fecha
115 @09.00 SAY SPACE(26)+ "*** DATOS PARA DISEÑO ***"
116 @11.05 SAY ' Cte.hormigón β: '
117 @11.27 SAY beta PICTURE '#.##'
118 @12.05 SAY ' Cte.flexión viga φ: '
119 @12.27 SAY fi1 PICTURE '#.##'
120 @13.05 SAY 'Cte.flexión columna φ: '
121 @13.27 SAY fi2 PICTURE '#.##'
122 @14.05 SAY 'Cte.corte y torsión τ: '
123 @14.27 SAY fi3 PICTURE '#.##'
124 @15.05 SAY ' Altura de Losa: '
125 @15.27 SAY hl PICTURE '### cm'
126 @16.05 SAY ' Peralte de la viga: '
127 @16.27 SAY D PICTURE '#### cm'
128 @17.05 SAY ' Momento Flexionante: '
129 @17.27 SAY m1 PICTURE '#####.### Kg-m'
130 @18.05 SAY 'Peralte de la Columna: '
131 @18.27 SAY H PICTURE '#### cm'
132 @19.05 SAY ' Peso sobre columna: '
133 @19.27 SAY pc PICTURE '#####.### Kg'
134 @20.05 SAY ' Ancho del Plinto : '
135 @20.27 SAY bp PICTURE '###.## m'
136 @21.05 SAY ' Esfuer.Admi.Suelo δ: '
137 @21.27 SAY ss PICTURE '#### Tn/m²'
138 @22.05 SAY ' : '
139 @22.27 SAY ' : '
140 @23.05 SAY ' : '
141 @23.27 SAY ' : '
142 @24.05 SAY 'Análisis Portico "EJE" : '
143 @24.33 SAY xx PICTURE '99'
144 EJECT
145 SET DEVICE TO SCREEN
146 ENDIF
147 DEACTIVATE WINDOW impresor
148 RETURN
150 *: EOF: REPO007.ACT

```

```

1  *:*****
2  *:
3  *:      Program: C:\CIVIL\REPO008.PRG
4  *:
5  *:      System: Anal.y Diseño de Estructuras Aporticadas
6  *:      Author: Rommel R. Alemán T. - Marco P. Silva S.
7  *:      Copyright (c) 1993, Pontificia Universidad Catolica del E.
8  *:      Last modified: 05/26/93      16:11
9  *:
10 *:      Called by: MENU003.PRG
11 *:
12 *:      Memory Files: IMPRESORA.MEM
13 *:      : &NOM03
14 *:
15 *:      Documented 05/28/93 at 12:13      FoxDoc version 2.10
16 *:*****
17  RELEASE ALL
18  IF nvx<=0 .OR. nvy<=0 .OR. bv<=0 .OR. bc<=0 .OR. np<=0
19  WAIT 'ERROR: Datos incompatibles o no existen...' WINDOW
20  <=====RETURN
21  └─ENDIF
22  RESTORE FROM impresora ADDITIVE
23  ACTIVATE WINDOW reportes
24  CLEAR
25  SELECT predise
26  SET ORDER TO 1
27  @00.13 TO 02,55
28  STORE 'PLANILLA DE SECCIONES' TO titulo
29  STORE ROUND((40 - LEN(titulo)) / 2,0) + 14 TO mlon
30  @01.mlon SAY titulo
31  @04.01 SAY "      Proyecto NUMERO :"
32  @04.24 SAY codigo PICTURE "*****"
33  @05.01 SAY "      Nombre del Cliente :"
34  @05.24 SAY proyecto
35  @06.01 SAY "Domicilio de la obra :"
36  @06.24 SAY direccion
37  @07.01 SAY "      Tipo de obra :"
38  @07.24 SAY obra
39  @08.01 SAY "      Ing. Calculista :"
40  @08.24 SAY calculo
41  @09.01 SAY "      Fecha (mm-dd-aa) :"
42  @09.24 SAY fecha
43  STORE SPACE(9) TO reporte
44  @11.01 SAY 'Reporte por Monitor o Impresora : ' GET reporte PICTURE '@m Monitor, Impresora'
45  READ
46  IF LASTKEY()=27
47  <=====RETURN
48  └─ENDIF
49  STORE UPPER(reporte) TO reporte
50  ACTIVATE WINDOW impresor
51  CLEAR
52  STORE codigo TO numero
53  WAIT 'PROYECTO No.'+STR(numero,7) WINDOW
54  STORE STR(numero,5) TO mnum
55  STORE 'np'+ALLTRIM(mnum) TO nom03
56  RESTORE FROM &nom03 ADDITIVE

```

```

57 STORE nvx TO mnvx
58 STORE nvy TO mnvy
59 STORE np TO mnp
60 STORE proyecto TO mproyecto
61 STORE direccion TO mdireccion
62 STORE calculo TO mcalculo
63 STORE fecha TO mfecha
64 STORE xx TO mxx
65 IF reporte='MONITOR'
66 SET DEVICE TO FILE reporte.txt
67 SET DEVICE TO SCREEN
68 MODI COMM reporte.txt NOEDIT WINDOW trabajo
69 ELSE
70 SET DEVICE TO PRINT no EJECT
71 SET PRINT ON
72 STORE 0 TO i,j
73 ? &mexp+'PLANILLA DE SECCIONES'+&mnnx AT 17
74 ?
75 ? 'PROYECTO DE CONSTRUCCION: '+mproyecto
76 ? ' DIRECCION: '+mdireccion
77 ? ' CALCULO: '+mcalculo
78 ? ' FECHA: '+DTOC(mfecha)
79 ?
80 ? 'ANALISIS PORTICO (X-X) '+mxx+' (cm²)' AT 27
81 ?
82 ? &mcom
83 ? 'Nivel' AT 00
84 FOR i=1 TO mnvx+1
85 ?? CHR(64+i) AT (i)*10+2
86 ENDFOR
87 FOR K=1 TO mnp
88 STORE 0 TO nivel
89 FOR k1=mnp TO K STEP -1
90 STORE nivel+hp(k1) TO nivel
91 ENDFOR
92 ? nivel PICTURE '##.##'
93 FOR i=1 TO mnvx
94 STORE '' TO viga
95 STORE STR(bv,2) TO viga
96 STORE viga+'/' TO viga
97 STORE viga+STR(D,2) TO viga
98 ?? viga AT (i+1)*10-5
99 ENDFOR
100 ?
101 FOR i=1 TO mnvx+1
102 STORE '' TO viga
103 STORE STR(bc,2) TO viga
104 STORE viga+'/' TO viga
105 STORE viga+STR(H,2) TO viga
106 ?? viga AT i*10
107 ENDFOR
108 ENDFOR
109 ? &mcco
110 EJECT
111 SET PRINT OFF
112 SET DEVICE TO SCREEN

```

```
113  L-ENDIF
114      DEACTIVATE WINDOW impresor
115      DEACTIVATE WINDOW reportes
116      RETURN
118      *: EOF: REP0008.ACT
```

```

1  *:*****
2  *:
3  *:      Program: C:\CIVIL\REP0009.PRG
4  *:
5  *:      System: Anal.y Diseño de Estructuras Aporticadas
6  *:      Author: Rommel R. Alemán T. - Marco P. Silva S.
7  *:      Copyright (c) 1993, Pontificia Universidad Catolica del E.
8  *:      Last modified: 05/26/93      16:11
9  *:
10 *:      Called by: MENU003.PRG
11 *:
12 *:      Memory Files: IMPRESORA.MEM
13 *:              : &NOM01
14 *:              : &NOM02
15 *:              : &NOM03
16 *:
17 *:      Documented 05/28/93 at 12:14      FoxDoc version 2.10
18 *:*****
19     RELEASE ALL
20     IF nvx<=0 .OR. nvy<=0 .OR. bv<=0 .OR. bc<=0 .OR. np<=0
21     WAIT "ERROR: Datos incompatibles o no existen..." WINDOW
22     RETURN
23     ENDIF
24     RESTORE FROM impresora ADDITIVE
25     ACTIVATE WINDOW reportes
26     CLEAR
27     SELECT predise
28     SET ORDER TO 1
29     @00,13 TO 02,55
30     STORE 'PLANILLA DE RIGIDECES' TO titulo
31     STORE ROUND((40 - LEN(titulo)) / 2,0) + 14 TO mlon
32     @01,mlon SAY titulo
33     @04,01 SAY "      Proyecto NUMERO : "
34     @04,24 SAY codigo PICTURE "####"
35     @05,01 SAY "      Nombre del Cliente : "
36     @05,24 SAY proyecto
37     @06,01 SAY "Domicilio de la obra : "
38     @06,24 SAY direccion
39     @07,01 SAY "      Tipo de obra : "
40     @07,24 SAY obra
41     @08,01 SAY "      Ing. Calculista : "
42     @08,24 SAY calculo
43     @09,01 SAY "      Fecha (mm-dd-aa) : "
44     @09,24 SAY fecha
45     STORE SPACE(9) TO reporte
46     @11,01 SAY 'Reporte por Monitor o Impresora : ' GET reporte PICTURE '@a Monitor, Impresora'
47     READ
48     IF LASTKEY()=27
49     RETURN
50     ENDIF
51     STORE UPPER(reportes) TO reporte
52     ACTIVATE WINDOW impresor
53     CLEAR
54     STORE codigo TO numero
55     STORE STR(numero,5) TO mnum
56     STORE 'nvx'+ALLTRIM(mnum) TO nom01

```

```

57  STORE 'nvx'+ALLTRIM(mnum) TO nom02
58  STORE 'np'+ALLTRIM(mnum) TO nom03
59  RESTORE FROM &nom01 ADDITIVE
60  RESTORE FROM &nom02 ADDITIVE
61  RESTORE FROM &nom03 ADDITIVE
62  STORE nvx TO mnvx
63  STORE nvy TO mnvy
64  STORE np TO mnp
65  STORE proyecto TO mproyecto
66  STORE direccion TO mdireccion
67  STORE calculo TO mcalculo
68  STORE fecha TO mfecha
69  STORE xx TO mxx
70  IF reporte='MONITOR'
71      SET DEVICE TO FILE reporte.txt
72      SET DEVICE TO SCREEN
73      MODI COMM reporte.txt NORDIT WINDOW trabajo
74  ELSE
75      SET DEVICE TO PRINT no EJECT
76      SET PRINT ON
77      STORE 0 TO i,j
78      ? &mexp+'PLANILLA DE RIGIDEZES'+&mnex AT 17
79      ?
80      ? 'PROYECTO DE CONSTRUCCION: '+mproyecto
81      ? '          DIRECCION: '+mdireccion
82      ? '          CALCULO: '+mcalculo
83      ? '          FECHA: '+DTC(mfecha)
84      ?
85      ? 'ANALISIS PORTICO (X-X) '+mxx+' (dm4/m)' AT 27
86      ?
87      ? &mcom
88      ? 'Nivel' AT 00
89      FOR i=1 TO mnvx+1
90          ?? CHR(64+i) AT (i)*10+2
91      ENDFOR
92      FOR K=1 TO mnp
93          STORE 0 TO nivel
94          FOR k1=1 TO K STEP -1
95              STORE nivel+hp(k1) TO nivel
96          ENDFOR
97          ? nivel PICTURE '##.##'
98          FOR i=1 TO mnvx
99              STORE 0 TO kv
100             STORE (bv/10*(D/10)^3)/(12*X(i)) TO kv
101             ?? kv PICTURE '###.###' AT (i+1)*10-4
102         ENDFOR
103         ?
104         FOR i=1 TO mnvx
105             STORE ((bv/10 * (D/10)^3)/(12*X(i)))/2 TO av
106             ?? av PICTURE '###.###' AT (i+1)*10-4
107         ENDFOR
108         ?
109         FOR i=1 TO mnvx+1
110             STORE (bc/10 * (H/10)^3)/(12 * hp(K)) TO kc
111             ?? kc PICTURE '###.###' AT i*10
112         ENDFOR

```

```

113      ?
114      FOR i=1 TO mnvx+1
115          STORE ((bc/10 * (H/10)^3)/(12 * hp(K)))/2 TO ac
116          ?? ac PICTURE '###.###' AT i*10
117      ENDFOR
118  ENDFOR
119      ? &mnco
120      EJECT
121      SET PRINT OFF
122      SET DEVICE TO SCREEN
123  ENDIF
124      DEACTIVATE WINDOW impresor
125      DEACTIVATE WINDOW reportes
126      RETURN
128      *: EOF: REPO009.ACT

```

```

1  *:*****
2  *:
3  *:      Program: C:\CIVIL\REPO010.PRG
4  *:
5  *:      System: Anal.y Diseño de Estructuras Aporticadas
6  *:      Author: Rommel R. Alemán T. - Marco P. Silva S.
7  *:      Copyright (c) 1993, Pontificia Universidad Catolica del E.
8  *:      Last modified: 05/26/93      16:11
9  *:
10 *:      Called by: MENU003.PRG
11 *:
12 *:      Memory Files: IMPRESORA.MEM
13 *:                  : &NOM01
14 *:                  : &NOM02
15 *:                  : &NOM03
16 *:
17 *:      Documented 05/28/93 at 12:14      FoxDoc version 2.10
18 *:*****
19     RELEASE ALL
20     IF nvx<=0 .OR. nvx<=0 .OR. bv<=0 .OR. bc<=0 .OR. np<=0
21     WAIT 'ERROR: Datos incompatibles o no existen...' WINDOW
22     RETURN
23     ENDIF
24     RESTORE FROM impresora ADDITIVE
25     ACTIVATE WINDOW reportes
26     CLEAR
27     SELECT predise
28     SET ORDER TO 1
29     @00,13 TO 02,55
30     STORE 'RESUMEN DE RIGIDECES' TO titulo
31     STORE ROUND((40 - LEN(titulo)) / 2,0) + 14 TO mlon
32     @01,mlon SAY titulo
33     @04,01 SAY "      Proyecto NUMERO : "
34     @04,24 SAY codigo PICTURE "*****"
35     @05,01 SAY "      Nombre del Cliente : "
36     @05,24 SAY proyecto
37     @06,01 SAY "Domicilio de la obra : "
38     @06,24 SAY direccion
39     @07,01 SAY "      Tipo de obra : "
40     @07,24 SAY obra
41     @08,01 SAY "      Ing. Calculista : "
42     @08,24 SAY calculo
43     @09,01 SAY "      Fecha (mm-dd-aa) : "
44     @09,24 SAY fecha
45     STORE SPACE(9) TO reporte
46     @11,01 SAY 'Reporte por Monitor o Impresora : ' GET reporte PICTURE '@m Monitor, Impresora'
47     READ
48     IF LASTKEY()=27
49     RETURN
50     ENDIF
51     STORE UPPER(reportes) TO reporte
52     ACTIVATE WINDOW impresor
53     CLEAR
54     STORE codigo TO numero
55     STORE STR(numero,5) TO mnum
56     STORE 'nvx'+ALLTRIM(mnum) TO nom01

```

```

57 STORE 'nvy'+ALLTRIM(mnum) TO nom02
58 STORE 'np'+ALLTRIM(mnum) TO nom03
59 RESTORE FROM &nom01 ADDITIVE
60 RESTORE FROM &nom02 ADDITIVE
61 RESTORE FROM &nom03 ADDITIVE
62 STORE nvx TO mnvx
63 STORE nvy TO mnvy
64 STORE np TO mnp
65 STORE proyecto TO mproyecto
66 STORE direccion TO mdireccion
67 STORE calculo TO mcalculo
68 STORE fecha TO mfecha
69 STORE xx TO mxx
70 DIMENSION x1(mnp,mnvx+1), x2(mnp,mnvx+1), x3(mnp,mnvx+1)
71 DIMENSION x11(mnp*2,mnvx*2), x22(mnp*2,(mnvx+1)*2)
72 STORE 0 TO x1,x2,x3,x11,x22
73 IF reporte='MONITOR'
74 SET DEVICE TO FILE reporte.txt
75 SET DEVICE TO SCREEN
76 MODI COMM reporte.txt NOEDIT WINDOW trabajo
77 ELSE
78 FOR K=1 TO mnp
79 FOR i=1 TO mnvx
80 STORE 0 TO kv
81 STORE (bv/10*(D/10)^3)/(12*X(i)) TO kv
82 STORE kv TO x1(K,i)
83 ENDFOR
84 FOR i=1 TO mnvx+1
85 STORE (bc/10 * (H/10)^3)/(12 * hp(K)) TO kc
86 STORE kc TO x2(K,i)
87 ENDFOR
88 ENDFOR
89 FOR i=1 TO mnp
90 STORE 1 TO K
91 FOR j=1 TO mnvx
92 STORE x1(i,j) TO x11(i,K)
93 STORE K+1 TO K
94 STORE x1(i,j) TO x11(i,K)
95 STORE K+1 TO K
96 ENDFOR
97 ENDFOR
98 CLEAR
99 STORE 0 TO x1
100 FOR i=1 TO mnp
101 STORE 1 TO K
102 FOR j=1 TO mnvx*2
103 IF j=1 .OR. j=mnvx*2
104 STORE x11(i,j) TO x1(i,K)
105 STORE K+1 TO K
106 ELSE
107 IF MOD(j,2) = 0
108 STORE x11(i,j)+x11(i,j+1) TO x1(i,K)
109 STORE K+1 TO K
110 ENDFIF
111 ENDFIF
112 ENDFOR

```

```

113      ENDFOR
114      STORE 1 TO K
115      FOR i=1 TO mnp
116          FOR j=1 TO mnvx+1
117              STORE x2(i,j) TO x22(K,j)
118          ENDFOR
119          STORE K+1 TO K
120          FOR j=1 TO mnvx+1
121              STORE x2(i,j) TO x22(K,j)
122          ENDFOR
123          STORE K+1 TO K
124      ENDFOR
125      CLEAR
126      STORE 0 TO x2
127      STORE 1 TO K
128      FOR i=1 TO mnp*2-1
129          FOR j=1 TO mnvx+1
130              IF i=1
131                  STORE x22(i,j) TO x2(K,j)
132              ELSE
133                  IF MOD(i,2) = 0
134                      STORE x22(i,j)+x22(i+1,j) TO x2(K,j)
135                  ENDIF
136              ENDIF
137          ENDFOR
138          IF i=1
139              STORE K+1 TO K
140          ELSE
141              IF MOD(i,2) = 0
142                  STORE K+1 TO K
143              ENDIF
144          ENDIF
145      ENDFOR
146      SET DEVICE TO PRINT no KJECT
147      SET PRINT ON
148      STORE 0 TO i,j
149      ? &mexp+ 'RESUMEN DE RIGIDECES'+&mnex AT 17
150      ?
151      ? 'PROYECTO DE CONSTRUCCION: '+mproyecto
152      ? '          DIRECCION: '+mdireccion
153      ? '          CALCULO: '+mcalculo
154      ? '          FECHA: '+DTOC(mfecha)
155      ?
156      ? 'ANALISIS PORTICO (X-X) '+mxx+ ' (dm4/m)' AT 27
157      ?
158      ? &mcom
159      ? 'Nivel' AT 00
160      FOR i=1 TO mnvx+1
161          ?? CHR(64+i) AT (i)*10+2
162      ENDFOR
163      FOR i=1 TO mnp
164          STORE 0 TO nivel
165          FOR K=mnp TO i STEP -1
166              STORE nivel+hp(K) TO nivel
167          ENDFOR
168          ? nivel PICTURE '##.##'

```

```

169      FOR j=1 TO mnvx+1
170          ?? x1(i,j)+x2(i,j) PICTURE '####.###' AT (j+1)*10-11
171          STORE x1(i,j)+x2(i,j) TO x3(i,j)
172      ENDFOR
173  ENDFOR
174      ? &nnco
175      EJECT
176      SET PRINT OFF
177      SET DEVICE TO SCREEN
178  ENDIF
179      DEACTIVATE WINDOW impresor
180      DEACTIVATE WINDOW reportes
181      RETURN
183      *: EOF: REPO010.ACT

```

```

1  *:*****
2  *:
3  *:      Program: C:\CIVIL\REPO011.PRG
4  *:
5  *:      System: Anal.y Diseño de Estructuras Aporticadas
6  *:      Author: Rommel R. Alemán T. - Marco P. Silva S.
7  *:      Copyright (c) 1993, Pontificia Universidad Catolica del E.
8  *:      Last modified: 05/26/93      16:12
9  *:
10 *:      Called by: MENU003.PRG
11 *:
12 *:      Memory Files: IMPRESORA.MEM
13 *:                  : &NOM01
14 *:                  : &NOM02
15 *:                  : &NOM03
16 *:
17 *:      Documented 05/28/93 at 12:14      FoxDoc version 2.10
18 *:*****
19      RELEASE ALL
20      IF nvx<=0 .OR. nvx<=0 .OR. hv<=0 .OR. bc<=0 .OR. np<=0
21      WAIT 'ERROR: Datos incompatibles o no existen...' WINDOW
22      RETURN
23  ENDIF
24      RESTORE FROM impresora ADDITIVE
25      ACTIVATE WINDOW reportes
26      CLEAR
27      SELECT predise
28      SET ORDER TO 1
29      @00,13 TO 02,55
30      STORE 'PLANILLA MOMENTO EMPOTRAMIENTO PERF.' TO titulo
31      STORE ROUND((40 - LEN(titulo)) / 2,0) + 14 TO mlon
32      @01,mlon SAY titulo
33      @04,01 SAY "      Proyecto NUMERO :"
34      @04,24 SAY codigo PICTURE "####"
35      @05,01 SAY "      Nombre del Cliente :"
36      @05,24 SAY proyecto
37      @06,01 SAY "Domicilio de la obra :"
38      @06,24 SAY direccion
39      @07,01 SAY "      Tipo de obra :"
40      @07,24 SAY obra
41      @08,01 SAY "      Ing. Calculista :"
42      @08,24 SAY calculo
43      @09,01 SAY "      Fecha (mm-dd-aa) :"
44      @09,24 SAY fecha
45      STORE SPACE(9) TO reporte
46      @11,01 SAY 'Reporte por Monitor o Impresora :' GET reporte PICTURE '@m Monitor, Impresora'
47      READ
48      IF LASTKEY()=27
49      RETURN
50  ENDIF
51      STORE UPPER(reportes) TO reporte
52      ACTIVATE WINDOW impresor
53      CLEAR
54      STORE codigo TO numero
55      STORE STR(numero,5) TO mnum
56      STORE 'nvx'+ALLTRIM(mnum) TO nom01

```

```

57 STORE 'nvy'+ALLTRIM(mnum) TO nom02
58 STORE 'np'+ALLTRIM(mnum) TO nom03
59 RESTORE FROM &nom01 ADDITIVE
60 RESTORE FROM &nom02 ADDITIVE
61 RESTORE FROM &nom03 ADDITIVE
62 STORE nvx TO mnvx
63 STORE nvy TO mnvy
64 STORE np TO mnp
65 STORE VAL(xx) TO portico
66 STORE xx TO mxx
67 STORE proyecto TO mproyecto
68 STORE direccion TO mdireccion
69 STORE calculo TO mcalculo
70 STORE fecha TO mfecha
71 DIMENSION x1(mnp,mnvx+1), x2(mnp,mnvx+1)
72 DIMENSION x11(mnp*2,mnvx*2), x22(mnp*2,(mnvx+1)*2)
73 STORE 0 TO x1,x2,x11,x22
74 IF reporte='MONITOR'
75     SET DEVICE TO FILE reporte.txt
76     SET DEVICE TO SCREEN
77     MODI COMM reporte.txt NOEDIT WINDOW trabajo
78 ELSE
79     SELECT mat03
80     SET ORDER TO 1
81     GO TOP
82     STORE 0 TO i,j
83     FOR i=1 TO RECCOUNT()
84         FOR j=1 TO mnvx
85             STORE 'X'+ALLTRIM(STR(j,2)) TO MACRO
86             STORE (&macro*X(j)^2)/12 TO x1(i,j)
87         ENDFOR
88         SKIP
89     ENDFOR
90     STORE portico TO i
91     STORE 1 TO K
92     FOR j=1 TO mnvx
93         STORE x1(i,j) TO x22(i,K)
94         STORE K+1 TO K
95         STORE -x1(i,j) TO x22(i,K)
96         STORE K+1 TO K
97     ENDFOR
98     FOR i=1 TO mnp
99         FOR j=1 TO mnvx*2
100             STORE x22(portico,j) TO x11(i,j)
101         ENDFOR
102     ENDFOR
103     SET DEVICE TO PRINT no EJECT
104     SET PRINT ON
105     STORE 0 TO i,j
106     ? &nexp+'PLANILLA MOMENTO EMPOTRAMIENTO PERF.'+'&mnex AT 3
107     ?
108     ? 'PROYECTO DE CONSTRUCCION: '+mproyecto
109     ? '          DIRECCION: '+mdireccion
110     ? '          CALCULO: '+mcalculo
111     ? '          FECHA: '+DTOC(mfecha)
112     ?

```

```

113      ? 'ANALISIS PORTICO (X-X) '+mx+ ' (Kg.m)' AT 27
114      ?
115      ? &mcom
116      ? 'Nivel' AT 00
117      STORE 1 TO K
118      FOR i=1 TO mnvx*2
119          IF i<=1
120              ?? CHR(64+i) AT (i)*10+2
121          ELSE
122              IF MOD(i,2)=0
123                  STORE K+1 TO K
124              ENDIF
125              ?? CHR(64+K) AT (i)*10+2
126          ENDIF
127      ENDFOR
128      FOR i=1 TO mnp
129          STORE 0 TO nivel
130          FOR K=mnp TO i STEP -1
131              STORE nivel+hp(K) TO nivel
132          ENDFOR
133          ? nivel PICTURE '##.##'
134          FOR j=1 TO mnvx*2
135              ?? x11(i,j) PICTURE '#####.##' AT (j+1)*10-11
136          ENDFOR
137      ENDFOR
138      ? &mcco
139      EJECT
140      SET PRINT OFF
141      SET DEVICE TO SCREEN
142  ENDIF
143      DEACTIVATE WINDOW impresor
144      DEACTIVATE WINDOW reportes
145      RETURN
146      *: EOF: RKPO011.ACT

```

```

1  *:*****
2  *:
3  *:      Program: C:\CIVIL\RKPO012.PRg
4  *:
5  *:      System: Anal.y Diseño de Estructuras Aporticadas
6  *:      Author: Rommel R. Alemán T. - Marco P. Silva S.
7  *:      Copyright (c) 1993, Pontificia Universidad Catolica del E.
8  *:      Last modified: 05/26/93      16:12
9  *:
10 *:      Called by: MENU003.PRg
11 *:
12 *:      Memory Files: IMPRESORA.MEM
13 *:              : &NOM01
14 *:              : &NOM02
15 *:              : &NOM03
16 *:
17 *:      Documented 05/28/93 at 12:14      FoxDoc version 2.10
18 *:*****
19  RELEASE ALL
20  IF nvx<=0 .OR. nvy<=0 .OR. bv<=0 .OR. bc<=0 .OR. np<=0
21  WAIT 'ERROR: Datos incompatibles o no existen...' WINDOW
22  RETURN
23  ENDIF
24  RESTORE FROM impresora ADDITIVE
25  ACTIVATE WINDOW reportes
26  CLEAR
27  SELECT predise
28  SET ORDER TO 1
29  @00,13 TO 02,55
30  STORE 'RESUMEN DE MOMENTOS DE EMPOTRAMIENTO' TO titulo
31  STORE ROUND((40 - LEN(titulo)) / 2,0) + 14 TO mlon
32  @01,mlon SAY titulo
33  @04,01 SAY "      Proyecto NUMERO : "
34  @04,24 SAY codigo PICTURE "*****"
35  @05,01 SAY "      Nombre del Cliente : "
36  @05,24 SAY proyecto
37  @06,01 SAY "Domicilio de la obra : "
38  @06,24 SAY direccion
39  @07,01 SAY "      Tipo de obra : "
40  @07,24 SAY obra
41  @08,01 SAY "      Ing. Calculista : "
42  @08,24 SAY calculo
43  @09,01 SAY "      Fecha (mm-dd-aa) : "
44  @09,24 SAY fecha
45  STORE SPACE(9) TO reporte
46  @11,01 SAY 'Reporte por Monitor o Impresora : ' GET reporte PICTURE '@m Monitor, Impresora'
47  READ
48  DEACTIVATE WINDOW reportes
49  IF LASTKEY()=27
50  RETURN
51  ENDIF
52  STORE UPPER(reportes) TO reporte
53  ACTIVATE WINDOW impresor
54  CLEAR
55  STORE codigo TO numero
56  STORE STR(numero,5) TO mnum

```

```

57 STORE `nvx'+ALLTRIM(mnum) TO nom01
58 STORE `nvx'+ALLTRIM(mnum) TO nom02
59 STORE `np'+ALLTRIM(mnum) TO nom03
60 RESTORE FROM &nom01 ADDITIVE
61 RESTORE FROM &nom02 ADDITIVE
62 RESTORE FROM &nom03 ADDITIVE
63 STORE nvx TO mnx
64 STORE nvx TO mny
65 STORE np TO mnp
66 STORE VAL(xx) TO portico
67 STORE xx TO mxx
68 STORE proyecto TO mproyecto
69 STORE direccion TO mdireccion
70 STORE calculo TO mcalculo
71 STORE fecha TO mfecha
72 DIMENSION x1(mny+1,mnx+1)
73 DIMENSION x11(mnp*2,mnx*2), x22(mnp*2,(mnx+1)*2)
74 STORE 0 TO x1,x11,x22
75
76 IF reporte="MONITOR"
77 SET DEVICE TO FILE reporte.txt
78 SET DEVICE TO SCREEN
79 MODI COMM reporte.txt NOEDIT WINDOW trabajo
80 ELSE
81 SELECT mat03
82 SET ORDER TO 1
83 GO TOP
84 STORE 0 TO i,j
85 FOR i=1 TO RECCOUNT()
86 FOR j=1 TO mnx
87 STORE 'X'+ALLTRIM(STR(j,2)) TO MACRO
88 STORE (&macro*(j)^2)/12 TO x1(i,j)
89 ENDFOR
90 SKIP
91 STORE portico TO i
92 STORE 1 TO k
93 FOR j=1 TO mny
94 STORE x1(i,j) TO x22(i,k)
95 STORE k+1 TO k
96 STORE -x1(i,j) TO x22(i,k)
97 STORE k+1 TO k
98 ENDFOR
99 FOR i=1 TO mnp
100 FOR j=1 TO mnx*2
101 STORE x22(portico,j) TO x11(i,j)
102 ENDFOR
103 ENDFOR
104 RELEASE x1
105 DIMENSION x1(mnp,mnx*2)
106 STORE 0 TO x1
107 FOR i=1 TO mnp
108 STORE 1 TO k
109 FOR j=1 TO mnx*2
110 IF j=1 .OR. j=mnx*2
111 STORE x11(i,j) TO x1(i,k)
112 STORE k+1 TO k

```

```

113      ELSE
114      IF MOD(j,2) = 0
115      STORE x11(i,j)+x11(i,j+1) TO x1(i,K)
116      STORE K+1 TO K
117      ENDIF
118      ENDIF
119      ENDFOR
120      ENDFOR
121      SET DEVICE TO PRINT no EJECT
122      SET PRINT ON
123      STORE 0 TO i,j
124      ? &mexp+'RESUMEN DE MOMENTOS DE EMPOTRAMIENTO'+&mhex AT 3
125      ?
126      ? 'PROYECTO DE CONSTRUCCION: '+mproyecto
127      ? '          DIRECCION: '+mdireccion
128      ? '          CALCULO: '+mcalculo
129      ? '          FECHA: '+DTC(mfecha)
130      ?
131      ? 'ANALISIS PORTICO (X-X) '+mxx+' (Kg.m)' AT 27
132      ?
133      ? &mcom
134      ? 'Nivel' AT 00
135      FOR i=1 TO mnvx+1
136      ?? CHR(64+i) AT (i)*10+2
137      ENDFOR
138      FOR i=1 TO mnp
139      STORE 0 TO nivel
140      FOR K=mnp TO i STEP -1
141      STORE nivel+hp(K) TO nivel
142      ENDFOR
143      ? nivel PICTURE '###.##'
144      FOR j=1 TO mnvx+1
145      ?? x1(i,j) PICTURE '#####.##' AT (j+1)*10-11
146      ENDFOR
147      ENDFOR
148      ? &mcco
149      EJECT
150      SET PRINT OFF
151      SET DEVICE TO SCREEN
152      ENDIF
153      DEACTIVATE WINDOW impresor
154      RETURN
155      *: EOF: REPO012.ACT

```

```

1  *:*****
2  *:
3  *:      Program: C:\CIVIL\REPO013.PRG
4  *:
5  *:      System: Anal.y Diseño de Estructuras Aporticadas
6  *:      Author: Rommel R. Alemán T. - Marco P. Silva S.
7  *:      Copyright (c) 1993, Pontificia Universidad Catolica del E.
8  *:      Last modified: 05/28/93      16:12
9  *:
10 *:      Called by: MENU003.PRG
11 *:
12 *:      Memory Files: IMPRESORA.MEM
13 *:              : &NOM01
14 *:              : &NOM02
15 *:              : &NOM03
16 *:              : MATRIZ.MEM
17 *:
18 *:      Documented 05/28/93 at 12:14      FoxDoc version 2.10
19 *:*****
20      RELEASE ALL
21      IF nvx<=0 .OR. nvy<=0 .OR. bv<=0 .OR. bc<=0 .OR. np<=0
22      WAIT 'ERROR: Datos incompatibles o no existen...' WINDOW
23      <=====RETURN
24      <-----ENDIF
25      RESTORE FROM impresora ADDITIVE
26      ACTIVATE WINDOW reportes
27      CLEAR
28      SELECT predise
29      SET ORDER TO 1
30      @00,13 TO 02,55
31      STORE 'MATRIZ DE RIGIDEZ'      TO titulo
32      STORE ROUND((40 - LEN(titulo)) / 2,0) + 14 TO mlon
33      @01,mlon SAY titulo
34      @04,01 SAY "      Proyecto NUMERO : "
35      @04,24 SAY codigo PICTURE "####"
36      @05,01 SAY "      Nombre del Cliente : "
37      @05,24 SAY proyecto
38      @06,01 SAY "Domicilio de la obra : "
39      @06,24 SAY direccion
40      @07,01 SAY "      Tipo de obra : "
41      @07,24 SAY obra
42      @08,01 SAY "      Ing. Calculista : "
43      @08,24 SAY calculo
44      @09,01 SAY "      Fecha (mm-dd-aa) : "
45      @09,24 SAY fecha
46      STORE SPACE(9) TO reporte
47      @11,01 SAY 'Reporte por Monitor o Impresora : ' GET reporte PICTURE '@m Monitor, Impresora'
48      READ
49      IF LASTKEY()=27
50      <=====RETURN
51      <-----ENDIF
52      STORE UPPER(reportes) TO reporte
53      ACTIVATE WINDOW impresor
54      @01,01 SAY 'GENERANDO REPORTE ...'
55      CLEAR
56      STORE codigo TO numero

```

```

57     STORE STR(numero,5) TO mnum
58     STORE 'nvx' + ALLTRIM(mnum) TO nom01
59     STORE 'nvy' + ALLTRIM(mnum) TO nom02
60     STORE 'np' + ALLTRIM(mnum) TO nom03
61     RESTORE FROM &nom01 ADDITIVE
62     RESTORE FROM &nom02 ADDITIVE
63     RESTORE FROM &nom03 ADDITIVE
64     RESTORE FROM matriz ADDITIVE
65     STORE nvx TO mnvx
66     STORE nvy TO mnvy
67     STORE np TO mnp
68     STORE proyecto TO mproyecto
69     STORE direccion TO mdireccion
70     STORE calculo TO mcalculo
71     STORE fecha TO mfecha
72     STORE xx TO mxx
73     DIMENSION x1(nx,nx)
74     STORE 0 TO x1
75     IF reporte='MONITOR'
76         SET DEVICE TO FILE reporte.txt
77         SET DEVICE TO SCREEN
78         MODI COMM reporte.txt NOEDIT WINDOW trabajo
79     ELSE
80         SELECT mat05
81         SET ORDER TO 1
82         GO TOP
83         STORE 0 TO i,j
84         FOR i=1 TO nx
85             FOR j=1 TO nx
86                 STORE 'X'+ALLTRIM(STR(j,2)) TO MACRO
87                 STORE &macro TO x1(i,j)
88             ENDFOR
89             SKIP
90         ENDFOR
91         SET DEVICE TO PRINT no EJECT
92         SET PRINT ON
93         STORE 0 TO i,j
94         ? &mexp+'MATRIZ DE RIGIDEZ'+&anex AT 11
95         ?
96         ? 'PROYECTO DE CONSTRUCCION: '+mproyecto
97         ? '                               DIRECCION: '+mdireccion
98         ? '                               CALCULO: '+mcalculo
99         ? '                               FECHA: '+DTC(mfecha)
100        ?
101        ? 'ANALISIS PORTICO (X-X) '+mxx AT 27
102        ?
103        ? &mcom
104        ? 'Nivel' AT 00
105        FOR i=1 TO nx
106            ?? CHR(233)+ALLTRIM(STR(i,2)) AT (i)*10+2
107        ENDFOR
108        FOR i=1 TO nx
109            ? i PICTURE '##'
110            FOR j=1 TO nx
111                ?? x1(i,j) PICTURE '#####.##' AT (j+1)*10-11
112            ENDFOR

```

```
113      L-ENDFOR
114      ? &nnco
115      EJECT
116      SET PRINT OFF
117      SET DEVICE TO SCREEN
118  L-ENDIF
119  DEACTIVATE WINDOW impresor
120  DEACTIVATE WINDOW reportes
121  RETURN
123  *: EOF: REP0013.ACT
```

```

1  *:*****
2  *:
3  *:      Program: C:\CIVIL\REP0014.PRG
4  *:
5  *:      System: Anal.y Diseño de Estructuras Aperticadas
6  *:      Author: Rommel R. Alemán T. - Marco P. Silva S.
7  *:      Copyright (c) 1993, Pontificia Universidad Catolica del E.
8  *:      Last modified: 05/26/93      16:12
9  *:
10 *:      Called by: MENU003.PRG
11 *:
12 *:      Memory Files: IMPRESORA.MEM
13 *:                  : &NOM01
14 *:                  : &NOM02
15 *:                  : &NOM03
16 *:                  : MATRIZ.MEM
17 *:
18 *:      Documented 05/28/93 at 12:14      FoxDoc version 2.10
19 *:*****
20  RELEASE ALL
21  IF nvx<=0 .OR. nvy<=0 .OR. bv<=0 .OR. bc<=0 .OR. np<=0
22  WAIT 'ERROR: Datos incompatibles o no existen...' WINDOW
23  <====RETURN
24  LENDIF
25  RESTORE FROM impresora ADDITIVE
26  ACTIVATE WINDOW reportes
27  CLEAR
28  SELECT predise
29  SET ORDER TO 1
30  @00,13 TO 02,55
31  STORE 'MATRIZ DE RIGIDEZ INVERSA' TO titulo
32  STORE ROUND((40 - LEN(titulo)) / 2,0) + 14 TO mlon
33  @01,mlon SAY titulo
34  @04,01 SAY "      Proyecto NUMERO : "
35  @04,24 SAY codigo PICTURE "####"
36  @05,01 SAY "      Nombre del Cliente : "
37  @05,24 SAY proyecto
38  @06,01 SAY "Domicilio de la obra : "
39  @06,24 SAY direccion
40  @07,01 SAY "      Tipo de obra : "
41  @07,24 SAY obra
42  @08,01 SAY "      Ing. Calculista : "
43  @08,24 SAY calculo
44  @09,01 SAY "      Fecha (mm-dd-aa) : "
45  @09,24 SAY fecha
46  STORE SPACE(9) TO reporte
47  @11,01 SAY 'Reporte por Monitor o Impresora : ' GET reporte PICTURE '@m Monitor, Impresora'
48  READ
49  IF LASTKEY()=27
50  <====RETURN
51  LENDIF
52  STORE UPPER(reportes) TO reporte
53  ACTIVATE WINDOW impresor
54  CLEAR
55  STORE codigo TO numero
56  STORE STR(numero,5) TO mnum

```

```

57 STORE 'nvx'+ALLTRIM(mnum) TO nom01
58 STORE 'nvy'+ALLTRIM(mnum) TO nom02
59 STORE 'np'+ALLTRIM(mnum) TO nom03
60 RESTORE FROM &nom01 ADDITIVE
61 RESTORE FROM &nom02 ADDITIVE
62 RESTORE FROM &nom03 ADDITIVE
63 RESTORE FROM matriz ADDITIVE
64 STORE nvx TO mnvx
65 STORE nvy TO mnvy
66 STORE np TO mnp
67 STORE proyecto TO mproyecto
68 STORE direccion TO mdireccion
69 STORE calculo TO mcalculo
70 STORE fecha TO mfecha
71 STORE xx TO mxx
72 DIMENSION x1(nx,nx)
73 STORE 0 TO x1
74 IF reporte='MONITOR'
75     SET DEVICE TO FILE reporte.txt
76     SET DEVICE TO SCREEN
77     MODI COMM reporte.txt NOEDIT WINDOW trabajo
78 ELSE
79     SELECT mat06
80     SET ORDER TO 1
81     GO TOP
82     STORE 0 TO i,j
83     FOR i=1 TO nx
84         FOR j=1 TO nx
85             STORE 'X'+ALLTRIM(STR(j,2)) TO MACRO
86             STORE &macro TO x1(i,j)
87         ENDFOR
88         SKIP
89     ENDFOR
90     SET DEVICE TO PRINT no EJECT
91     SET PRINT ON
92     STORE 0 TO i,j
93     ? &mexp+'MATRIZ DE RIGIDEZ INVERSA'+&mnex AT 3
94     ?
95     ? 'PROYECTO DE CONSTRUCCION: '+mproyecto
96     ? 'DIRECCION: '+mdireccion
97     ? 'CALCULO: '+mcalculo
98     ? 'FECHA: '+DTOC(mfecha)
99     ?
100    ? 'ANALISIS PORTICO (X-X) '+mxx AT 27
101    ?
102    ? &mcom
103    ? 'Nivel' AT 00
104    FOR i=1 TO nx
105        ?? CHR(233)+ALLTRIM(STR(i,2)) AT (i)*10+2
106    ENDFOR
107    FOR i=1 TO nx
108        ? i PICTURE '##'
109        FOR j=1 TO nx
110            ?? x1(i,j) PICTURE '#####.##' AT (j+1)*10-11
111        ENDFOR
112    ENDFOR

```

```
113      ? &nnco
114      EJECT
115      SET PRINT OFF
116      SET DEVICE TO SCREEN
117  └─ENDIF
118      DEACTIVATE WINDOW impresor
119      DEACTIVATE WINDOW reportes
120      RETURN
122      *: EOF: REPO014.ACT
```

```

1  *:*****
2  *:
3  *:      Program: C:\CIVIL\RRP0015.PRG
4  *:
5  *:      System: Anal.y Diseño de Estructuras Aporticadas
6  *:      Author: Rommel R. Alemán T. - Marco P. Silva S.
7  *:      Copyright (c) 1993, Pontificia Universidad Catolica del E.
8  *:      Last modified: 05/26/93      16:12
9  *:
10 *:      Called by: MENU003.PRG
11 *:
12 *:      Memory Files: IMPRESORA.MEM
13 *:                  : &NOM01
14 *:                  : &NOM02
15 *:                  : &NOM03
16 *:                  : MATRIZ.MEM
17 *:                  : GIRO.MEM
18 *:
19 *:      Documented 05/28/93 at 12:14      FoxDoc version 2.10
20 *:*****
21      RELEASE ALL
22      IF nvx<=0 .OR. nvy<=0 .OR. bv<=0 .OR. bc<=0 .OR. np<=0
23      WAIT "ERROR: Datos incompatibles o no existen..." WINDOW
24      <=====RETURN
25      <ENDIF
26      RESTORE FROM impresora ADDITIVE
27      ACTIVATE WINDOW reportes
28      CLEAR
29      SELECT predise
30      SET ORDER TO 1
31      @00,13 TO 02,55
32      STORE "GIROS DE NUDOS FINALES" TO titulo
33      STORE ROUND((40 - LEN(titulo)) / 2,0) + 14 TO mlon
34      @01,mlon SAY titulo
35      @04,01 SAY "      Proyecto NUMERO : "
36      @04,24 SAY codigo PICTURE "####"
37      @05,01 SAY "      Nombre del Cliente : "
38      @05,24 SAY proyecto
39      @06,01 SAY "Domicilio de la obra : "
40      @06,24 SAY direccion
41      @07,01 SAY "      Tipo de obra : "
42      @07,24 SAY obra
43      @08,01 SAY "      Ing. Calculista : "
44      @08,24 SAY calculo
45      @09,01 SAY "      Fecha (mm-dd-aa) : "
46      @09,24 SAY fecha
47      STORE SPACE(9) TO reporte
48      @11,01 SAY "Reporte por Monitor o Impresora : " GET reporte PICTURE "@m Monitor, Impresora"
49      READ
50      IF LASTKEY()=27
51      <=====RETURN
52      <ENDIF
53      STORE UPPER(reportes) TO reporte
54      ACTIVATE WINDOW impresor
55      CLEAR
56      STORE codigo TO numero

```

```

57     STORE STR(numero,5) TO mnum
58     STORE 'nvx'+ALLTRIM(mnum) TO nom01
59     STORE 'nvy'+ALLTRIM(mnum) TO nom02
60     STORE 'np'+ALLTRIM(mnum) TO nom03
61     RESTORE FROM &nom01 ADDITIVE
62     RESTORE FROM &nom02 ADDITIVE
63     RESTORE FROM &nom03 ADDITIVE
64     RESTORE FROM matriz ADDITIVE
65     RESTORE FROM giro ADDITIVE
66     STORE nvx TO mnvx
67     STORE nvy TO mnvy
68     STORE np TO mnp
69     STORE proyecto TO mproyecto
70     STORE direccion TO mdireccion
71     STORE calculo TO mcalculo
72     STORE fecha TO mfecha
73     STORE xx TO mxx
74     DIMENSION x1(mnp,mnvx+1)
75     STORE 0 TO x1
76     IF reporte='MONITOR'
77         SET DEVICE TO FILE reporte.txt
78         SET DEVICE TO SCREEN
79         MODI COMM reporte.txt NOEDIT WINDOW trabajo
80     ELSE
81         STORE 1 TO K
82         FOR i=1 TO mnp
83             FOR j=1 TO mnvx+1
84                 STORE giros(K) TO x1(i,j)
85                 STORE K + 1 TO K
86             ENDFOR
87         ENDFOR
88         SET DEVICE TO PRINT no EJECT
89         SET PRINT ON
90         STORE 0 TO i,j
91         ? &mexp+'GIROS DE NUDOS FINALES'+&mnex AT 3
92         ?
93         ? 'PROYECTO DE CONSTRUCCION: '+mproyecto
94         ? ' DIRECCION: '+mdireccion
95         ? ' CALCULO: '+mcalculo
96         ? ' FECHA: '+DTC(mfecha)
97         ?
98         ? 'ANALISIS PORTICO (X-X) '+mxx AT 27
99         ?
100        ? &mcom
101        ? 'Nivel' AT 00
102        FOR i=1 TO mnvx+1
103            ?? CHR(233)+ALLTRIM(STR(i,2)) AT (i)*10+2
104        ENDFOR
105        FOR i=1 TO mnp
106            ? i PICTURE '##'
107            FOR j=1 TO mnvx+1
108                ?? x1(i,j) PICTURE '#####' AT (j+1)*10-11
109            ENDFOR
110        ENDFOR
111        ? &mco
112        EJECT

```

```
113      SET PRINT OFF
114      SET DEVICE TO SCREEN
115  ┌─ENDIF
116      DEACTIVATE WINDOW impresor
117      DEACTIVATE WINDOW reportes
118      RETURN
120      *: EOF: REPO015.ACT
```

```

1  *:*****
2  *:
3  *:      Program: C:\CIVIL\REPO016.PRG
4  *:
5  *:      System: Anal.y Diseño de Estructuras Aporticadas
6  *:      Author: Rommel R. Alemán T. - Marco P. Silva S.
7  *:      Copyright (c) 1993, Pontificia Universidad Catolica del E.
8  *:      Last modified: 05/26/93      16:12
9  *:
10 *:      Called by: MENU003.PRG
11 *:
12 *:      Memory Files: IMPRESORA.MEM
13 *:                  : &NOM01
14 *:                  : &NOM02
15 *:                  : &NOM03
16 *:                  : MATRIZ.MEM
17 *:                  : GIRO.MEM
18 *:
19 *:      Documented 05/28/93 at 12:14      FoxDoc version 2.10
20 *:*****
21      RELEASE ALL
22      IF nvx<=0 .OR. nvy<=0 .OR. bv<=0 .OR. bc<=0 .OR. np<=0
23      WAIT 'ERROR: Datos incompatibles o no existen...' WINDOW
24      RETURN
25      ENDIF
26      RESTORE FROM impresora ADDITIVE
27      ACTIVATE WINDOW reportes
28      CLEAR
29      SELECT predise
30      SET ORDER TO 1
31      @00,13 TO 02,55
32      STORE 'MOMENTOS FINALES DE NUDO'      TO titulo
33      STORE ROUND((40 - LEN(titulo)) / 2,0) + 14 TO mlon
34      @01,mlon SAY titulo
35      @04,01 SAY "      Proyecto NUMERO : "
36      @04,24 SAY codigo PICTURE "*****"
37      @05,01 SAY "      Nombre del Cliente : "
38      @05,24 SAY proyecto
39      @06,01 SAY "Domicilio de la obra : "
40      @06,24 SAY direccion
41      @07,01 SAY "      Tipo de obra : "
42      @07,24 SAY obra
43      @08,01 SAY "      Ing. Calculista : "
44      @08,24 SAY calculo
45      @09,01 SAY "      Fecha (mm-dd-aa) : "
46      @09,24 SAY fecha
47      STORE SPACE(9) TO reporte
48      @11,01 SAY 'Reporte por Monitor o Impresora : ' GET reporte PICTURE '@m Monitor, Impresora'
49      READ
50      IF LASTKEY()=27
51      RETURN
52      ENDIF
53      STORE UPPER(reportes) TO reporte
54      ACTIVATE WINDOW impresor
55      CLEAR
56      STORE codigo TO numero

```

```

57  STORE STR(numero,5) TO mnum
58  STORE 'nvx'+ALLTRIM(mnum) TO nom01
59  STORE 'nvy'+ALLTRIM(mnum) TO nom02
60  STORE 'np'+ALLTRIM(mnum) TO nom03
61  RESTORE FROM &nom01 ADDITIVE
62  RESTORE FROM &nom02 ADDITIVE
63  RESTORE FROM &nom03 ADDITIVE
64  RESTORE FROM matriz ADDITIVE
65  RESTORE FROM giro ADDITIVE
66  STORE nvx TO mnvx
67  STORE nvy TO mnvy
68  STORE np TO mnp
69  STORE proyecto TO mproyecto
70  STORE direccion TO mdireccion
71  STORE calculo TO mcalculo
72  STORE fecha TO mfecha
73  STORE xx TO mxx
74  IF reporte='MONITOR'
75      SET DEVICE TO FILE reporte.txt
76      SET DEVICE TO SCREEN
77      MODI COMM reporte.txt NOEDIT WINDOW trabajo
78  ELSE
79      SET DEVICE TO PRINT no EJECT
80      SET PRINT ON
81      STORE 0 TO i,j
82      ? &mexp+'MOMENTOS FINALES DE NUDO'+&mnex AT 10
83      ?
84      ? 'PROYECTO DE CONSTRUCCION: '+mproyecto
85      ? '          DIRECCION: '+mdireccion
86      ? '          CALCULO: '+mcalculo
87      ? '          FECHA: '+DTC(mfecha)
88      ?
89      ? 'ANALISIS PORTICO (X-X) '+mxx AT 27
90      ?
91      ? &mcon
92      ? '*** V I G A S *** (Kg.m)'
93      ?
94      ?
95      ? 'Nivel' AT 00
96      STORE 1 TO K
97      FOR i=1 TO mnvx*2
98          IF i<=1
99              ?? CHR(64+i) AT (i)*12+5
100         ELSE
101             IF MOD(i,2)=0
102                 STORE K+1 TO K
103             ENDIF
104             ?? CHR(64+K) AT (i)*12+5
105         ENDIF
106     ENDFOR
107     SELECT mat16
108     SET ORDER TO 1
109     GO TOP
110     FOR i=1 TO mnp
111         STORE 0 TO nivel
112         FOR K=mnp TO i STEP -1

```

```

113      STORE nivel+hp(K) TO nivel
114      ENDFOR
115      ? nivel PICTURE '##.##'
116      FOR j=1 TO mnvx*2
117          STORE 'X'+ALLTRIM(STR(j,2)) TO MACRO
118          ?? &macro PICTURE '#####.##' AT (j+1)*12-13
119      ENDFOR
120      SKIP
121      ENDFOR
122      ?
123      ?
124      ?
125      ? '*** C O L U M N A S *** (Kg.m)'
126      ?
127      ?
128      ? 'Nivel' AT 00
129      FOR i=1 TO mnvx+1
130          ?? CHR(64+i)+ALLTRIM(STR(i,2)) AT (i)*12+2
131      ENDFOR
132      STORE 0 TO 1
133      SELECT mat17
134      SET ORDER TO 1
135      GO TOP
136      FOR i=1 TO mnp*2
137          STORE 0 TO nivel
138          IF MOD(i,2) # 0
139              STORE 1 + 1 TO 1
140              FOR K=mnp TO 1 STEP -1
141                  STORE nivel+hp(K) TO nivel
142              ENDFOR
143              ? nivel PICTURE '##.##'
144          ENDIF
145          FOR j=1 TO mnvx+1
146              STORE 'X'+ALLTRIM(STR(j,2)) TO MACRO
147              ?? &macro PICTURE '#####.##' AT (j+1)*12-13
148          ENDFOR
149          SKIP
150          ?
151      ENDFOR
152      ? &mnco
153      EJECT
154      SET PRINT OFF
155      SET DEVICE TO SCREEN
156      ENDFIF
157      DEACTIVATE WINDOW impresor
158      DEACTIVATE WINDOW reportes
159      RETURN
161      *: EOF: REPO016.ACT

```

```

2 *:
3 *:      Program: C:\CIVIL\REPO017.PRG
4 *:
5 *:      System: Anal.y Diseño de Estructuras Aporticadas
6 *:      Author: Rommel R. Alemán T. - Marco P. Silva S.
7 *:      Copyright (c) 1993, Pontificia Universidad Catolica del E.
8 *:      Last modified: 05/26/93      16:12
9 *:
10 *:      Called by: MENU003.PRG
11 *:
12 *:      Memory Files: IMPRESORA.MEM
13 *:                  : &NOM01
14 *:                  : &NOM02
15 *:                  : &NOM03
16 *:                  : MATRIZ.MEM
17 *:
18 *:      Documented 05/28/93 at 12:14      FoxDoc version 2.10
19 *:*****
20      RELEASE ALL
21      IF nvx<=0 .OR. nvx<=0 .OR. bv<=0 .OR. bc<=0 .OR. np<=0
22      WAIT "ERROR: Datos incompatibles o no existen..." WINDOW
23      RETURN
24      ENDIF
25      RESTORE FROM impresora ADDITIVE
26      ACTIVATE WINDOW reportes
27      CLEAR
28      SELECT predise
29      SET ORDER TO 1
30      @00,13 TO 02,55
31      STORE "Mborde, Mmax, V Total, Flecha" TO titulo
32      STORE ROUND((40 - LEN(titulo)) / 2,0) + 14 TO mlon
33      @01,mlon SAY titulo
34      @04,01 SAY "      Proyecto NUMERO :"
35      @04,24 SAY codigo PICTURE "####"
36      @05,01 SAY "      Nombre del Cliente :"
37      @05,24 SAY proyecto
38      @06,01 SAY "Domicilio de la obra :"
39      @06,24 SAY direccion
40      @07,01 SAY "      Tipo de obra :"
41      @07,24 SAY obra
42      @08,01 SAY "      Ing. Calculista :"
43      @08,24 SAY calculo
44      @09,01 SAY "      Fecha (mm-dd-aa) :"
45      @09,24 SAY fecha
46      STORE SPACE(9) TO reporte
47      @11,01 SAY "Reporte por Monitor o Impresora :" GET reporte PICTURE "@m Monitor, Impresora"
48      READ
49      IF LASTKEY()=27
50      RETURN
51      ENDIF
52      STORE UPPER(reporte) TO reporte
53      ACTIVATE WINDOW impresor
54      CLEAR
55      STORE codigo TO numero
56      STORE STR(numero,5) TO mnum
57      STORE 'nvx'+ALLTRIM(mnum) TO nom01

```

```

58 STORE 'nvy'+ALLTRIM(mnum) TO nom02
59 STORE 'np'+ALLTRIM(mnum) TO nom03
60 RESTORE FROM &nom01 ADDITIVE
61 RESTORE FROM &nom02 ADDITIVE
62 RESTORE FROM &nom03 ADDITIVE
63 RESTORE FROM matriz ADDITIVE
64 STORE nvx TO mnvx
65 STORE nvy TO mnvy
66 STORE np TO mnp
67 STORE fc TO mfc
68 STORE fy TO mfy
69 STORE fil TO mfil
70 STORE l4/mfy TO mp
71 STORE D/100 TO mdv
72 STORE H/200 TO mdc
73 STORE mdc+mdv TO md
74 STORE proyecto TO mproyecto
75 STORE direccion TO mdireccion
76 STORE calculo TO mcalculo
77 STORE fecha TO mfecha
78 STORE xx TO mxx
79 STORE VAL(mxx) TO eje
80 IF reporte='MONITOR'
81 SET DEVICE TO FILE reporte.txt
82 SET DEVICE TO SCREEN
83 MODI COMM reporte.txt NORDIT WINDOW trabajo
84 ELSE
85 DIMENSION q(mnvx), m(mnp,mnvx*2), AS(mnp,mnvx*3)
86 DIMENSION vo(mnvx*2), vh(mnvx*2), vt(mnvx*2)
87 DIMENSION mb(mnvx*2), mx(mnvx), mmax(mnvx)
88 STORE 0 TO q,m,AS,vo,vh,vt,mb,mx,mmax
89 SELECT mat03
90 SET ORDER TO 1
91 SEEK eje
92 FOR i=1 TO mnvx
93 STORE 'X'+ALLTRIM(STR(i,2)) TO MACRO
94 STORE &macro TO q(i)
95 ENDFOR
96 SELECT mat16
97 SET ORDER TO 1
98 GO TOP
99 FOR i=1 TO mnp
100 FOR j=1 TO mnvx*2
101 STORE 'X'+ALLTRIM(STR(j,2)) TO MACRO
102 STORE &macro TO m(i,j)
103 ENDFOR
104 SKIP
105 ENDFOR
106 STORE mfil*mfy*md*(1 - 0.59 * mp * (mfy / mfc)) TO das
107 SET DEVICE TO PRINT no EJECT
108 SET PRINT ON
109 STORE 0 TO i,j
110 ? &mexp+'Mborde, Mmax, V Total, Flecha'+&mnex AT 8
111 ?
112 ? 'PROYECTO DE CONSTRUCCION: '+mproyecto
113 ? ' DIRECCION: '+mdireccion

```

```

114 ? '          CALCULO: '+mcalculo
115 ? '          FECHA: '+DTOC(mfecha)
116 ?
117 ? 'ANALISIS PORTICO (X-X) '+mxx AT 27
118 ?
119 ? &mcom
120 ?
121 FOR i=1 TO mnp
122   ? '(q)'
123   STORE 1 TO pos
124   FOR j=1 TO mnvx
125     ?? q(j) PICTURE '#####.##' AT pos*10
126     STORE pos+2 TO pos
127   ENDFOR
128   ? '(M)'
129   STORE 1 TO pos
130   FOR j=1 TO mnvx*2
131     ?? m(i,j) PICTURE '#####.##' AT pos*10
132     STORE pos+1 TO pos
133   ENDFOR
134   ? '(Vo)'
135   STORE 1 TO K
136   STORE 1 TO pos
137   FOR j=1 TO mnvx
138     STORE ROUND((q(j)*X(j))/2,3) TO voi, vo(K)
139     STORE K+1 TO K
140     STORE ROUND(voi * (-1),3) TO vod, vo(K)
141     STORE K+1 TO K
142     ?? voi PICTURE '#####.##' AT pos*10
143     STORE pos+1 TO pos
144     ?? vod PICTURE '#####.##' AT pos*10
145     STORE pos+1 TO pos
146   ENDFOR
147   ? '(Vh)'
148   STORE 1 TO K
149   STORE 1 TO pos
150   FOR j=1 TO mnvx
151     STORE ROUND((m(i,K)+m(i,K+1))/X(j),3) TO vhi, vh(K)
152     STORE K+1 TO K
153     STORE ROUND(vhi * (-1),3) TO vhd, vh(K)
154     STORE K+1 TO K
155     ?? vhi PICTURE '#####.##' AT pos*10
156     STORE pos+1 TO pos
157     ?? vhd PICTURE '#####.##' AT pos*10
158     STORE pos+1 TO pos
159   ENDFOR
160   ? '(Vt)'
161   STORE 1 TO K
162   STORE 1 TO pos
163   FOR j=1 TO mnvx
164     STORE ROUND((vo(K)+vh(K)),3) TO vti, vt(K)
165     STORE ROUND((vo(K)-vh(K)),3) TO vtd, vt(K+1)
166     STORE K+2 TO K
167     ?? vti PICTURE '#####.##' AT pos*10
168     STORE pos+1 TO pos
169     ?? vtd PICTURE '#####.##' AT pos*10

```

```

170      STORE pos+1 TO pos
171    ┌ ENDFOR
172      ? '(Mb)'
173      STORE 1 TO K
174      STORE 1 TO pos
175      FOR j=1 TO mnvx
176        STORE ROUND( (vt(K) * md - (q(j) * md^2 / 2) - m(i,K)), 3) TO mbi, mb(K)
177        STORE K+1 TO K
178        STORE ROUND( (-vt(K) * md + (q(j) * md^2 / 2) + m(i,K)), 3) TO mbd, mb(K)
179        STORE K+1 TO K
180        ?? mbi PICTURE '#####.##' AT pos*10
181        STORE pos+1 TO pos
182        ?? mbd PICTURE '#####.##' AT pos*10
183        STORE pos+1 TO pos
184      └ ENDFOR
185      ? '(K)'
186      STORE 1 TO K
187      STORE 1 TO pos
188      FOR j=1 TO mnvx
189        STORE ROUND( vt(K) / q(j), 3) TO mx, mx(j)
190        STORE K+2 TO K
191        ?? mx PICTURE '#####.##' AT pos*10
192        STORE pos+2 TO pos
193      └ ENDFOR
194      ? '(Mmax)'
195      STORE 1 TO K
196      STORE 1 TO pos
197      FOR j=1 TO mnvx
198        STORE ROUND( vt(K)^2 / (2*q(j)) - m(i,K), 3) TO mmax, mmax(j)
199        STORE K+2 TO K
200        ?? mmax PICTURE '#####.##' AT pos*10
201        STORE pos+2 TO pos
202      └ ENDFOR
203      ? REPLICATE('-', (mnvx*2+1)*11)
204      ?
205      STORE 1 TO K, 1
206      STORE 1 TO pos
207      FOR j=1 TO mnvx
208        STORE ROUND( mb(K)/das, 3) TO asi, AS(i,1)
209        STORE l+1 TO l
210        STORE ROUND( mmax(j)/das, 3) TO ASC, AS(i,1)
211        STORE l+1 TO l
212        STORE ROUND( mb(K+1)/das, 3) TO asd, AS(i,1)
213        STORE K+2 TO K
214        STORE l+1 TO l
215      └ ENDFOR
216    └ ENDFOR
217    ? &nnc0
218    EJECT
219    SET PRINT OFF
220    SET DEVICE TO SCREEN
221  └ ENDF
222    DEACTIVATE WINDOW impresor
223    DEACTIVATE WINDOW reportes
224    RETURN
226    *: EOF: REPO017.ACT

```

```

1  *:*****
2  *:
3  *:      Program: C:\CIVIL\REP0018.PRG
4  *:
5  *:      System: Anal y Diseño de Estructuras Aporticadas
6  *:      Author: Rommel R. Alemán T. - Marco P. Silva S.
7  *:      Copyright (c) 1993, Pontificia Universidad Catolica del E.
8  *:      Last modified: 05/26/93      16:12
9  *:
10 *:      Called by: MENU003.PRG
11 *:
12 *:      Memory Files: IMPRESORA.MEM
13 *:
14 *:      Documented 05/28/93 at 12:14                      FoxDoc version 2.10
15 *:*****
16      RELEASE ALL
17      IF nvx<=0 .OR. nvy<=0 .OR. bv<=0 .OR. bc<=0 .OR. np<=0
18      WAIT 'ERROR: Datos incompatibles o no existen...' WINDOW
19      <====RETURN
20      ENDIF
21      RESTORE FROM impresora ADDITIVE
22      ACTIVATE WINDOW reportes
23      CLEAR
24      SELECT predise
25      SET ORDER TO 1
26      @00,13 TO 02,55
27      STORE 'ARMADURA DE VIGAS' TO titulo
28      STORE ROUND((40 - LEN(titulo)) / 2,0) + 14 TO mlon
29      @01,mlon SAY titulo
30      @04,01 SAY "      Proyecto NUMERO : "
31      @04,24 SAY codigo PICTURE "#####"
32      @05,01 SAY "      Nombre del Cliente : "
33      @05,24 SAY proyecto
34      @06,01 SAY "Domicilio de la obra : "
35      @06,24 SAY direccion
36      @07,01 SAY "      Tipo de obra : "
37      @07,24 SAY obra
38      @08,01 SAY "      Ing. Calculista : "
39      @08,24 SAY calculo
40      @09,01 SAY "      Fecha (mm-dd-aa) : "
41      @09,24 SAY fecha
42      STORE SPACE(9) TO reporte
43      @11,01 SAY 'Reporte por Monitor o Impresora : ' GET reporte PICTURE '@m Monitor, Impresora'
44      READ
45      IF LASTKEY()=27
46      <====RETURN
47      ENDIF
48      STORE UPPER(reportes) TO reporte
49      ACTIVATE WINDOW impresor
50      CLEAR
51      STORE nvx TO mnvx
52      STORE nvy TO mnvy
53      STORE np TO mnp
54      STORE proyecto TO mproyecto
55      STORE direccion TO mdireccion
56      STORE calculo TO mcalculo

```

```

57     STORE fecha      TO mfecha
58     STORE xx         TO mxx
59     IF reporte='MONITOR'
60         SET DEVICE TO FILE reporte.txt
61         SET DEVICE TO SCREEN
62         MODI COMM reporte.txt NOEDIT WINDOW trabajo
63     ELSE
64         SET DEVICE TO PRINT no EJECT
65         SET PRINT ON
66         STORE 0 TO i,j
67         ? &mexp+'ARMADURA DE VIGAS'+&mhex AT 14
68         ?
69         ? 'PROYECTO DE CONSTRUCCION: '+mproyecto
70         ? '                               DIRECCION: '+mdireccion
71         ? '                               CALCULO: '+mcalculo
72         ? '                               FECHA: '+DTC(mfecha)
73         ?
74         ? 'ANALISIS PORTICO (X-X) '+mxx+' (cm²)' AT 27
75         ?
76         ? &mcom
77         ?
78         ?
79         SELECT mat18
80         SET ORDER TO 1
81         GO TOP
82         FOR i=1 TO mnp
83             ? 'As'+ALLTRIM(STR(i,2))
84             STORE 1 TO pos
85             FOR j=1 TO mnvx*3
86                 STORE 'X'+ALLTRIM(STR(j,2)) TO MACRO
87                 ?? &macro PICTURE '#####.###' AT pos*10
88                 STORE pos+1 TO pos
89             ENDFOR
90             SKIP
91         ENDFOR
92         ? &mcco
93         EJECT
94         SET PRINT OFF
95         SET DEVICE TO SCREEN
96     ENDIF
97     DEACTIVATE WINDOW impresor
98     DEACTIVATE WINDOW reportes
99     RETURN
101    *: EOF: REPO018.ACT

```

```

1  *:*****
2  *:
3  *:      Program: C:\CIVIL\REPO019.PRG
4  *:
5  *:      System: Anal.y Diseño de Estructuras Aporticadas
6  *:      Author: Rommel R. Alemán T. - Marco P. Silva S.
7  *:      Copyright (c) 1993, Pontificia Universidad Catolica del E.
8  *:      Last modified: 05/26/93      16:12
9  *:
10 *:      Called by: MENU003.PRG
11 *:
12 *:      Memory Files: IMPRESORA.MEM
13 *:
14 *:      Documented 05/28/93 at 12:14      FoxDoc version 2.10
15 *:*****
16  RELEASE ALL
17  IF nvx<=0 .OR. nvy<=0 .OR. bv<=0 .OR. bc<=0 .OR. np<=0
18  WAIT 'ERROR: Datos incompatibles o no existen...' WINDOW
19  <====RETURN
20  └─ENDIF
21  RESTORE FROM impresora ADDITIVE
22  ACTIVATE WINDOW reportes
23  CLEAR
24  SELECT predise
25  SET ORDER TO 1
26  @00,13 TO 02,55
27  STORE 'ARMADURA DE COLUMNAS'      TO titulo
28  STORE ROUND((40 - LEN(titulo)) / 2,0) + 14 TO mlon
29  @01,mlon SAY titulo
30  @04,01 SAY "      Proyecto NUMERO : "
31  @04,24 SAY codigo PICTURE "####"
32  @05,01 SAY "      Nombre del Cliente : "
33  @05,24 SAY proyecto
34  @06,01 SAY "Domicilio de la obra : "
35  @06,24 SAY direccion
36  @07,01 SAY "      Tipo de obra : "
37  @07,24 SAY obra
38  @08,01 SAY "      Ing. Calculista : "
39  @08,24 SAY calculo
40  @09,01 SAY "      Fecha (mm-dd-aa) : "
41  @09,24 SAY fecha
42  STORE SPACE(9) TO reporte
43  @11,01 SAY 'Reporte por Monitor o Impresora : ' GET reporte PICTURE '@m Monitor, Impresora'
44  READ
45  IF LASTKEY()=27
46  <====RETURN
47  └─ENDIF
48  STORE UPPER(reportes) TO reportes
49  ACTIVATE WINDOW impresor
50  CLEAR
51  STORE nvx      TO mnvx
52  STORE nvy      TO mnvy
53  STORE np       TO mnp
54  STORE bc       TO mbc
55  STORE beta     TO mbeta
56  STORE fi2      TO mfi2

```

```

57     STORE fc      TO mfc
58     STORE H       TO mh
59     STORE fy      TO mfy
60     STORE phl     TO mph
61     STORE proyecto TO mproyecto
62     STORE direccion TO mdireccion
63     STORE calculo  TO mcalculo
64     STORE fecha    TO mfecha
65     STORE VAL(xx)  TO mxx
66     DIMENSION pc(mnvx+1)
67     STORE 0 TO pc
68     IF reporte='MONITOR'
69         SET DEVICE TO FILE reporte.txt
70         SET DEVICE TO SCREEN
71         MODI COMM reporte.txt NOEDIT WINDOW trabajo
72     ELSE
73         SET DEVICE TO PRINT no EJECT
74         SET PRINT ON
75         STORE 0 TO i,j
76         ? &mexp+'ARMADURA DE COLUMNAS'+&mnex AT 14
77         ?
78         ? 'PROYECTO DE CONSTRUCCION: '+mproyecto
79         ? '          DIRECCION: '+mdireccion
80         ? '          CALCULO: '+mcalculo
81         ? '          FECHA: '+DTC(mfecha)
82         ?
83         ? 'ANALISIS PORTICO (X-X) '+ALLTRIM(STR(mxx))+ ' (cm²)' AT 27
84         ?
85         ? &mcom
86         ?
87         ?
88         SELECT mat19
89         SET ORDER TO 1
90         GO TOP
91         SEEK mxx
92         FOR i=1 TO mnvx+1
93             STORE 'X'+ALLTRIM(STR(i,2)) TO MACRO
94             STORE &macro TO pc(i)
95         ENDFOR
96         SELECT mat17
97         SET ORDER TO 1
98         GO BOTTOM
99         FOR i=mnv TO 1 STEP -1
100            SKIP -1
101            ? 'As'+ALLTRIM(STR(i,2))
102            STORE 1 TO pos
103            FOR j=1 TO mnvx+1
104                STORE 'X'+ALLTRIM(STR(j,2)) TO MACRO
105                STORE &macro TO mu
106                STORE ((pc(j)/mnv*i + 200* ABS(mu)/mbc) - (mbeta*0.85*mfc*mbc*mh)) / (mfi2*mfy) TO
AS
107                ?? AS PICTURE '#####.###' AT pos*10
108                STORE pos+1 TO pos
109            ENDFOR
110            SKIP -2
111        ENDFOR

```

```

112      ?
113      ?
114      ? '** ACKERO MINIMO ASUMIDO **'
115      ?
116      GO BOTTOM
117      FOR i=mnv TO 1 STEP -1
118      |   SKIP -1
119      |   ? 'As'+ALLTRIM(STR(i,2))
120      |   STORE 1 TO pos
121      |   FOR j=1 TO mnvx+1
122      |   |   STORE 'X'+ALLTRIM(STR(j,2)) TO MACRO
123      |   |   STORE &macro TO mu
124      |   |   STORE ((pc(j)/mnp*i + 200* ABS(mu)/mbc) - (mbeta*0.85*mfc*mbc*mh)) / (mfi2*mfy) TO
AS
125      |   |   IF AS<=0
126      |   |   |   STORE mph*mbc*mh TO AS
127      |   |   |   ENDIF
128      |   |   ?? AS PICTURE '#####.###' AT pos*10
129      |   |   STORE pos+1 TO pos
130      |   |   ENDFOR
131      |   |   SKIP -2
132      |   |   ENDFOR
133      |   ? &mnco
134      |   EJECT
135      |   SET PRINT OFF
136      |   SET DEVICE TO SCREEN
137      |   ENDIF
138      DEACTIVATE WINDOW impresor
139      DEACTIVATE WINDOW reportes
140      RETURN
142      *: EOF: REPO019.ACT

```

```

1  *:*****
2  *:
3  *:      Program: C:\CIVIL\REP0020.PRG
4  *:
5  *:      System: Anal.y Diseño de Estructuras Aporticadas
6  *:      Author: Rommel R. Alemán T. - Marco P. Silva S.
7  *:      Copyright (c) 1993, Pontificia Universidad Catolica del E.
8  *:      Last modified: 05/26/93      16:12
9  *:
10 *:      Called by: MENU003.PRG
11 *:
12 *:      Memory Files: IMPRESORA.MEM
13 *:                  : &NOM01
14 *:                  : &NOM02
15 *:                  : &NOM03
16 *:
17 *:      Documented 05/28/93 at 12:15      FoxDoc version 2.10
18 *:*****
19  RELEASE ALL
20  IF nvx<=0 .OR. nvy<=0 .OR. bv<=0 .OR. bc<=0 .OR. np<=0
21  WAIT 'ERROR: Datos incompatibles o no existen...' WINDOW
22  <====RETURN
23  └─ENDIF
24  RESTORE FROM impresora ADDITIVE
25  ACTIVATE WINDOW reportes
26  CLEAR
27  SELECT predise
28  SET ORDER TO 1
29  @00,13 TO 02,55
30  STORE 'ARMADURA DE LOSAS' TO titulo
31  STORE ROUND((40 - LEN(titulo)) / 2,0) + 14 TO mlon
32  @01,mlon SAY titulo
33  @04,01 SAY " Proyecto NUMERO :"
34  @04,24 SAY codigo PICTURE "#####"
35  @05,01 SAY " Nombre del Cliente :"
36  @05,24 SAY proyecto
37  @06,01 SAY "Domicilio de la obra :"
38  @06,24 SAY direccion
39  @07,01 SAY " Tipo de obra :"
40  @07,24 SAY obra
41  @08,01 SAY " Ing. Calculista :"
42  @08,24 SAY calculo
43  @09,01 SAY " Fecha (mm-dd-aa) :"
44  @09,24 SAY fecha
45  STORE SPACE(9) TO reporte
46  @11,01 SAY 'Reporte por Monitor o Impresora : ' GET reporte PICTURE '@m Monitor, Impresora'
47  READ
48  IF LASTKEY()=27
49  <====RETURN
50  └─ENDIF
51  STORE UPPER(reportes) TO reporte
52  ACTIVATE WINDOW impresor
53  CLEAR
54  STORE codigo TO numero
55  STORE STR(numero,5) TO mnum
56  STORE 'nvx'+ALLTRIM(mnum) TO nom01

```

```

57 STORE 'nvy'+ALLTRIM(mnum) TO nom02
58 STORE 'np'+ALLTRIM(mnum) TO nom03
59 RESTORE FROM &nom01 ADDITIVE
60 RESTORE FROM &nom02 ADDITIVE
61 RESTORE FROM &nom03 ADDITIVE
62 STORE nvx TO mnvx
63 STORE nvy TO mnvy
64 STORE np TO mnp
65 STORE beta TO mbeta
66 STORE fil TO mfil
67 STORE fc TO mfc
68 STORE el TO mel
69 STORE hl TO mhl
70 STORE cu TO mcu
71 STORE fy TO mfy
72 STORE proyecto TO mproyecto
73 STORE direccion TO mdireccion
74 STORE calculo TO mcalculo
75 STORE fecha TO mfecha
76 STORE VAL(xx) TO mxx
77 IF reporte='MONITOR'
78 SET DEVICE TO FILE reporte.txt
79 SET DEVICE TO SCREEN
80 MODI COMM reporte.txt NOEDIT WINDOW trabajo
81 ELSE
82 SET DEVICE TO PRINT no EJECT
83 SET PRINT ON
84 STORE 0 TO i,j
85 ? &mexp+'ARMADURA DE LOSAS'+&mnex AT 14
86 ?
87 ? 'PROYECTO DE CONSTRUCCION: '+mproyecto
88 ? ' DIRECCION: '+mdireccion
89 ? ' CALCULO: '+mcalculo
90 ? ' FECHA: '+DTC(mfecha)
91 ?
92 ? 'ANALISIS PORTICO (X-X) (cm²)' AT 27
93 ? &mcom
94 ?
95 STORE mfil*mbeta*mfc*100*mel*100*((mhl-3)-mel*100/2) TO muf
96 STORE (14*(hl-3))/(mbeta*mfc) TO A
97 ? 'AsX'
98 STORE 1 TO pos
99 FOR i=1 TO mnvx
100 STORE mcu*X(i)^2/8 TO mu
101 IF mu<muf .AND. A<mel
102 STORE mu*100 / (mfil*mfy*mhl*(1-0.59*14/mfc)) TO AS
103 ELSE
104 STORE mu*100 / (mfil*mfy*(mhl-A/2)) TO AS
105 ENDIF
106 ?? AS PICTURE '####.###' AT pos*10
107 STORE pos + 1 TO pos
108 ENDFOR
109 ? &mco
110 ? 'ANALISIS PORTICO (Y-Y) (cm²)' AT 27
111 ? &mcom
112 ?

```

```

113      ? 'AsY'
114      STORE 1 TO pos
115      FOR i=1 TO mny
116          STORE mu*y(i) 2/8 TO nu
117          IF nu<mu1.AND. A<ae1
118              STORE nu*100 / (m11*mfy*ml*(1-0.59*14/m1c)) TO AS
119          ELSE
120              STORE nu*100 / (m11*mfy*(ml-A/2)) TO AS
121          ENDIF
122      ?? AS PICTURE '####.###' AT pos*10
123      STORE pos + 1 TO pos
124      ENDFOR
125      ? &muco
126      EJECT
127      SKT PRINT OFF
128      SKT DEVICE TO SCREEN
129      ENDIF
130      DRACTIVATE WINDOW impresor
131      DRACTIVATE WINDOW reportes
132      RETURN
133      *: KOF: REP0020.ACT
134

```

```

1  *:*****
2  *:
3  *:      Program: C:\CIVIL\REPO021.PRG
4  *:
5  *:      System: Anal.y Diseño de Estructuras Aporticadas
6  *:      Author: Rommel R. Alemán T. - Marco P. Silva S.
7  *:      Copyright (c) 1993, Pontificia Universidad Catolica del E.
8  *:      Last modified: 05/26/93      16:12
9  *:
10 *:      Called by: MENU003.PRG
11 *:
12 *:      Memory Files: IMPRESORA.MEM
13 *:
14 *:      Documented 05/28/93 at 12:15                      FoxDoc version 2.10
15 *:*****
16      RELEASE ALL
17      IF nvx<=0 .OR. nvx<=0 .OR. bv<=0 .OR. bc<=0 .OR. np<=0
18      WAIT 'ERROR: Datos incompatibles o no existen...' WINDOW
19      <=====RETURN
20      └─ENDIF
21      RESTORE FROM impresora ADDITIVE
22      ACTIVATE WINDOW reportes
23      CLEAR
24      SELECT predise
25      SET ORDER TO 1
26      @00,13 TO 02,55
27      STORE 'ARMADURA DE PLINTOS' TO titulo
28      STORE ROUND((40 - LEN(titulo)) / 2,0) + 14 TO mlon
29      @01,mlon SAY titulo
30      @04,01 SAY "      Proyecto NUMERO : "
31      @04,24 SAY codigo PICTURE "#####"
32      @05,01 SAY "      Nombre del Cliente : "
33      @05,24 SAY proyecto
34      @06,01 SAY "Domicilio de la obra : "
35      @06,24 SAY direccion
36      @07,01 SAY "      Tipo de obra : "
37      @07,24 SAY obra
38      @08,01 SAY "      Ing. Calculista : "
39      @08,24 SAY calculo
40      @09,01 SAY "      Fecha (mm-dd-aa) : "
41      @09,24 SAY fecha
42      STORE SPACE(9) TO reporte
43      @11,01 SAY 'Reporte por Monitor o Impresora : ' GET reporte PICTURE '@m Monitor, Impresora'
44      READ
45      IF LASTKEY()=27
46      <=====RETURN
47      └─ENDIF
48      STORE UPPER(reportes) TO reporte
49      ACTIVATE WINDOW impresor
50      CLEAR
51      STORE nvx TO mnvx
52      STORE nvx TO mnvy
53      STORE np TO mnp
54      STORE bc TO mbc
55      STORE fi2 TO mfi2
56      STORE fc TO mfc

```

```

57     STORE H      TO mh
58     STORE fy     TO mfy
59     STORE csp/100 TO mcsp
60     STORE bp     TO mbp, mbpl
61     STORE ss     TO mss
62     STORE beta   TO mbeta
63     STORE proyecto TO mproyecto
64     STORE direccion TO mdireccion
65     STORE calculo TO mcalculo
66     STORE fecha  TO mfecha
67     STORE VAL(xx) TO mxx
68     DIMENSION pc(mnvx+1)
69     STORE 0 TO pc
70     IF reporte='MONITOR'
71         SET DEVICE TO FILE reporte.txt
72         SET DEVICE TO SCREEN
73         MODI COMM reporte.txt NOEDIT WINDOW trabajo
74     ELSE
75         SET DEVICE TO PRINT no EJECT
76         SET PRINT ON
77         STORE 0 TO i,j
78         ? &mexp+'ARMADURA DE PLINTOS'+&mnex AT 14
79         ?
80         ? 'PROYECTO DE CONSTRUCCION: '+mproyecto
81         ? '          DIRECCION: '+mdireccion
82         ? '          CALCULO: '+mcalculo
83         ? '          FECHA: '+DTOC(mfecha)
84         ?
85         ? 'ANALISIS PORTICO (X-X) '+ALLTRIM(STR(mxx)) AT 27
86         ?
87         ? &mcom
88         ?
89         ?
90         SELECT mat19
91         SET ORDER TO 1
92         GO TOP
93         SEEK mxx
94         FOR i=1 TO mnvx+1
95             STORE 'X'+ALLTRIM(STR(i,2)) TO MACRO
96             STORE &macro TO pc(i)
97         ENDFOR
98         SELECT mat17
99         SET ORDER TO 1
100        GO BOTTOM
101        ? 'Ac (m²)'
102        STORE 1 TO pos
103        FOR j=1 TO mnvx+1
104            STORE 'X'+ALLTRIM(STR(j,2)) TO MACRO
105            STORE &macro TO mu
106            STORE (pc(j)) / (mss*1000) TO ac
107            ?? ac PICTURE '#####.###' AT pos*10
108            STORE pos+1 TO pos
109        ENDFOR
110        ? 'Bp (m)'
111        STORE 1 TO pos
112        FOR j=1 TO mnvx+1

```

```

113     STORE 'X'+ALLTRIM(STR(j,2)) TO MACRO
114     STORE &macro TO mu
115     STORE (pc(j)) / (mss*1000) TO ac
116     STORE ac^(1/2) TO mbp
117     ?? mbp PICTURE '#####.###' AT pos*10
118     STORE pos+1 TO pos
119   _ENDFOR
120   ? 'Ap (m)'
121   STORE 1 TO pos
122   _FOR j=1 TO mnvx+1
123     STORE 'X'+ALLTRIM(STR(j,2)) TO MACRO
124     STORE &macro TO mu
125     STORE (pc(j)) / (mss*1000) TO ac
126     STORE ac^(1/2) TO ap
127     ?? ap PICTURE '#####.###' AT pos*10
128     STORE pos+1 TO pos
129   _ENDFOR
130   ? 'Dp (m)'
131   STORE 1 TO pos
132   _FOR j=1 TO mnvx+1
133     STORE 'X'+ALLTRIM(STR(j,2)) TO MACRO
134     STORE &macro TO mu
135     STORE (pc(j)) / (mss*1000) TO ac
136     STORE ac^(1/2) TO mbp
137     STORE ac^(1/2) TO ap
138     STORE (pc(j) / ac) + (mu*6/(mbp*ap^2)) TO ss2
139     STORE ss2*ap*1.5 TO qu
140     STORE 0.53*mbeta*mc^(1/2) TO vc
141     STORE (((ap - mh/100)/2)*qu)/(qu+(vc*10000/2)) TO dp
142     *   IF DP1<30
143     *     STORE 30 TO DP
144     *   ELSE
145     *     STORE ROUND((DP1/10)+0.5,0) * 10 TO DP
146     *   ENDIF
147     ?? dp PICTURE '#####.###' AT pos*10
148     STORE pos+1 TO pos
149   _ENDFOR
150   ? 'AspX (cm²)'
151   STORE 1 TO pos
152   _FOR j=1 TO mnvx+1
153     STORE 'X'+ALLTRIM(STR(j,2)) TO MACRO
154     STORE &macro TO mu
155     STORE (pc(j)) / (mss*1000) TO ac
156     STORE ac^(1/2) TO ap
157     STORE ac^(1/2) TO mbp
158     STORE (pc(j) / ac) + (mu*6/(mbp*ap^2)) TO ss2
159     STORE ss2*ap*1.5 TO qu
160     STORE 0.53*mbeta*mc^(1/2) TO vc
161     STORE (((ap - mh/100)/2)*qu)/(qu+(vc*10000/2)) TO dp
162     *   IF DP1<30
163     *     STORE 30 TO DP
164     *   ELSE
165     *     STORE ROUND((DP1/10)+0.5,0) * 10 TO DP
166     *   ENDIF
167     STORE (14/mfy)*mbp*dp*10000 TO aspx
168     ?? aspx PICTURE '#####.###' AT pos*10

```

```

169      STORE pos+1 TO pos
170  ┌   ENDFOR
171      ? 'AspY (cm²)'
172      STORE 1 TO pos
173  ┌   FOR j=1 TO mnvx+1
174      │   STORE 'X'+ALLTRIM(STR(j,2)) TO MACRO
175      │   STORE &macro TO mu
176      │   STORE (pc(j)) / (mss*1000) TO ac
177      │   STORE ac^(1/2) TO ap
178      │   STORE ac^(1/2) TO mbp
179      │   STORE (pc(j) / ac) + (mu*6/(mbp*ap^2)) TO ss2
180      │   STORE ss2*ap*1.5 TO qu
181      │   STORE 0.53*mbeta*mc^(1/2) TO vc
182      │   STORE (((ap - mh/100)/2)*qu)/(qu+(vc*10000/2)) TO dp
183      │   *   IF DP1<30
184      │   *       STORE 30 TO DP
185      │   *   ELSE
186      │   *       STORE ROUND((DP1/10)+0.5,0) * 10 TO DP
187      │   *   ENDIF
188      │   STORE (14/mfy)*ap*dp*10000 TO aspy
189      │   ?? aspy PICTURE '####.###' AT pos*10
190      │   STORE pos+1 TO pos
191  └   ENDFOR
192      ? &mnco
193      EJECT
194      SET PRINT OFF
195      SET DEVICE TO SCREEN
196  └   ENDFIF
197      DEACTIVATE WINDOW impresor
198      DEACTIVATE WINDOW reportes
199      RETURN
201      *: EOF: REPO021.ACT

```

```

1  *:*****
2  *:
3  *:      Program: C:\CIVIL\REPO022.PRG
4  *:
5  *:      System: Anal.y Diseño de Estructuras Aperticadas
6  *:      Author: Rommel R. Alemán T. - Marco P. Silva S.
7  *:      Copyright (c) 1993, Pontificia Universidad Catolica del E.
8  *:      Last modified: 05/26/93      16:12
9  *:
10 *:      Called by: MENU003.PRG
11 *:
12 *:      Memory Files: IMPRESORA.MEM
13 *:
14 *:      Documented 05/28/93 at 12:15      FoxDoc version 2.10
15 *:*****
16      RELEASE ALL
17      IF nvx<=0 .OR. nvy<=0 .OR. bv<=0 .OR. bc<=0 .OR. np<=0
18      WAIT 'ERROR: Datos incompatibles o no existen...' WINDOW
19      <=====RETURN
20      ENDIF
21      RESTORE FROM impresora ADDITIVE
22      ACTIVATE WINDOW reportes
23      CLEAR
24      SELECT predise
25      SET ORDER TO 1
26      @00,13 TO 02.55
27      STORE 'ARMADURA PARA CORTE (ESTRIBOS)' TO titulo
28      STORE ROUND((40 - LEN(titulo)) / 2,0) + 14 TO mlon
29      @01,mlon SAY titulo
30      @04,01 SAY "      Proyecto NUMERO : "
31      @04,24 SAY codigo PICTURE "####"
32      @05,01 SAY "      Nombre del Cliente : "
33      @05,24 SAY proyecto
34      @06,01 SAY "Domicilio de la obra : "
35      @06,24 SAY direccion
36      @07,01 SAY "      Tipo de obra : "
37      @07,24 SAY obra
38      @08,01 SAY "      Ing. Calculista : "
39      @08,24 SAY calculo
40      @09,01 SAY "      Fecha (mm-dd-aa) : "
41      @09,24 SAY fecha
42      STORE SPACE(9) TO reporte
43      @11,01 SAY 'Reporte por Monitor o Impresora : ' GET reporte PICTURE '@m Monitor, Impresora'
44      READ
45      DEACTIVATE WINDOW reportes
46      IF LASTKEY()=27
47      <=====RETURN
48      ENDIF
49      STORE UPPER(reportes) TO reporte
50      ACTIVATE WINDOW impresor
51      CLEAR
52      STORE bc TO mbc
53      STORE bv TO mbv
54      STORE H TO mh
55      STORE D TO md
56      STORE fy TO mfy

```

```

57     STORE ee      TO mee
58     STORE proyecto TO mproyecto
59     STORE direccion TO mdireccion
60     STORE calculo  TO mcalculo
61     STORE fecha    TO mfecha
62     STORE VAL(xx)  TO mxx
63     DIMENSION pc(mnvx+1)
64     STORE 0 TO pc
65     IF reporte='MONITOR'
66         SET DEVICE TO FILE reporte.txt
67         SET DEVICE TO SCREEN
68         MODI COMM reporte.txt NOEDIT WINDOW trabajo
69     ELSE
70         SET DEVICE TO PRINT no EJECT
71         SET PRINT ON
72         STORE 0 TO i,j
73         ? &mexp+'ARMADURA PARA CORTE (ESTRIBOS)'+&mnnx AT 14
74         ?
75         ? 'PROYECTO DE CONSTRUCCION: '+mproyecto
76         ? '          DIRECCION: '+mdireccion
77         ? '          CALCULO: '+mcalculo
78         ? '          FECHA: '+DTC(mfecha)
79         ?
80         ?
81         ? 'ESTRIBOS PARA VIGAS (cm²)' AT 27
82         ? '-----' AT 27
83         ? &ncom
84         ?
85         STORE (3.5*mbv*mee)/mfy TO asvx
86         STORE (3.5*md*mee)/mfy TO asvy
87         ? 'EJE X-X' AT 30
88         ?? 'EJE Y-Y' AT 60
89         ?
90         ? asvx PICTURE '###.###' AT 30
91         ?? asvy PICTURE '###.###' AT 60
92         ? &ncco
93         ?
94         ? 'ESTRIBOS PARA COLUMNAS (cm²)' AT 26
95         ? '-----' AT 26
96         ? &ncom
97         ?
98         STORE (3.5*mbc*mee)/mfy TO ascx
99         STORE (3.5*mh*mee)/mfy TO ascy
100        ? 'EJE X-X' AT 30
101        ?? 'EJE Y-Y' AT 60
102        ?
103        ? ascx PICTURE '###.###' AT 30
104        ?? ascy PICTURE '###.###' AT 60
105        ? &ncco
106        EJECT
107        SET PRINT OFF
108        SET DEVICE TO SCREEN
109    ENDIF
110    RETURN
112    *: KOF: REP0022.ACT

```

```

1  *:*****
2  *:
3  *:      Program: C:\CIVIL\REP0023.PRG
4  *:
5  *:      System: Anal.y Diseño de Estructuras Aporticadas
6  *:      Author: Rommel R. Alemán T. - Marco P. Silva S.
7  *:      Copyright (c) 1993, Pontificia Universidad Catolica del E.
8  *:      Last modified: 05/26/93      16:13
9  *:
10 *:      Called by: MENU003.PRG
11 *:
12 *:      Memory Files: IMPRESORA.MEM
13 *:              : &NOM01
14 *:              : &NOM02
15 *:              : &NOM03
16 *:
17 *:      Documented 05/28/93 at 12:15      FoxDoc version 2.10
18 *:*****
19      RELEASE ALL
20      IF nvx<=0 .OR. nvy<=0 .OR. bv<=0 .OR. bc<=0 .OR. np<=0
21      WAIT 'ERROR: Datos incompatibles o no existen...' WINDOW
22      <=====RETURN
23      └─ENDIF
24      RESTORE FROM impresora ADDITIVE
25      ACTIVATE WINDOW reportes
26      CLEAR
27      SELECT predise
28      SET ORDER TO 1
29      @00,13 TO 02,55
30      STORE 'GEOMETRIA' TO titulo
31      STORE ROUND((40 - LEN(titulo)) / 2,0) + 14 TO mlon
32      @01,mlon SAY titulo
33      @04,01 SAY "      Proyecto NUMERO : "
34      @04,24 SAY codigo PICTURE "####"
35      @05,01 SAY "      Nombre del Cliente : "
36      @05,24 SAY proyecto
37      @06,01 SAY "Domicilio de la obra : "
38      @06,24 SAY direccion
39      @07,01 SAY "      Tipo de obra : "
40      @07,24 SAY obra
41      @08,01 SAY "      Ing. Calculista : "
42      @08,24 SAY calculo
43      @09,01 SAY "      Fecha (mm-dd-aa) : "
44      @09,24 SAY fecha
45      STORE SPACE(9) TO reporte
46      @11,01 SAY 'Reporte por Monitor o Impresora : ' GET reporte PICTURE '@m Monitor, Impresora'
47      READ
48      IF LASTKEY()=27
49      DEACTIVATE WINDOW reportes
50      <=====RETURN
51      └─ENDIF
52      STORE UPPER(reportes) TO reporte
53      ACTIVATE WINDOW impresor
54      CLEAR
55      STORE codigo TO numero
56      STORE STR(numero,5) TO mnum

```

```

57 STORE 'nvx'+ALLTRIM(mnum) TO nom01
58 STORE 'nv'+ALLTRIM(mnum) TO nom02
59 STORE 'np'+ALLTRIM(mnum) TO nom03
60 RESTORE FROM &nom01 ADDITIVE
61 RESTORE FROM &nom02 ADDITIVE
62 RESTORE FROM &nom03 ADDITIVE
63 STORE nvx TO mnvx
64 STORE nvy TO mnvy
65 STORE np TO mnp
66 STORE bc TO mbc
67 STORE bv TO mbv
68 STORE H TO mh
69 STORE D TO md
70 STORE fy TO mfy
71 STORE ee TO mee
72 STORE proyecto TO mproyecto
73 STORE direccion TO mdireccion
74 STORE calculo TO mcalculo
75 STORE fecha TO mfecha
76 STORE VAL(xx) TO mxx
77 DIMENSION pc(mnvx+1)
78 STORE 0 TO pc
79 IF reporte='MONITOR'
80 SET DEVICE TO FILE reporte.txt
81 SET DEVICE TO SCREEN
82 MODI COMM reporte.txt NOEDIT WINDOW trabajo
83 ELSE
84 SET DEVICE TO PRINT no EJECT
85 SET PRINT ON
86 STORE 0 TO i,j
87 ? &mexp+'GEOMETRIA'+&mnex AT 14
88 ?
89 ? 'PROYECTO DE CONSTRUCCION: '+mproyecto
90 ? ' DIRECCION: '+mdireccion
91 ? ' CALCULO: '+mcalculo
92 ? ' FECHA: '+DTC(mfecha)
93 ?
94 ? 'PLANTA TIPO'
95 ? '-----'
96 ? &mcom
97 ? CHR(65) AT 1
98 STORE 2 TO pos
99 FOR i=1 TO mnvx
100 ?? REPLICATE(CHR(196),ROUND(X(i)*3.29,0)) AT pos
101 STORE pos + ROUND(X(i)*3.29,0) TO pos
102 ?? CHR(65+i) AT pos
103 STORE pos + 1 TO pos
104 ENDFOR
105 ? CHR(218) AT 1
106 STORE 2 TO pos
107 STORE 1 TO l2
108 FOR i=1 TO mnvx
109 ?? REPLICATE(CHR(196),ROUND(X(i)*3.29,0)) AT pos
110 STORE pos + ROUND(X(i)*3.29,0) TO pos
111 IF i#mnvx
112 ?? CHR(194) AT pos

```

```

113      ELSE
114          ?? CHR(191) AT pos
115          ?? 12 PICTURE '##' FUNCTION 'L' AT pos+3
116      ENDIF
117      STORE pos + 1 TO pos
118  ENDFOR
119      STORE 12+1 TO 12
120      STORE mnvy TO 11
121      FOR l=1 TO mnvy
122          FOR j=1 TO ROUND(y(11),0)
123              ?
124              STORE 1 TO pos
125              FOR K=1 TO mnvx+1
126                  ?? CHR(179) AT pos
127                  STORE pos + ROUND(X(K)*3.29,0) + 1 TO pos
128              ENDFOR
129          ENDFOR
130          IF l#mnvy
131              ? CHR(195) AT 1
132              STORE 2 TO pos
133              FOR i=1 TO mnvx
134                  ?? REPLICATE(CHR(196),ROUND(X(i)*3.29,0)) AT pos
135                  STORE pos + ROUND(X(i)*3.29,0) TO pos
136                  IF i#mnvx
137                      ?? CHR(197) AT pos
138                  ELSE
139                      ?? CHR(180) AT pos
140                      ?? 12 PICTURE '##' FUNCTION 'L' AT pos+3
141                  ENDIF
142                  STORE pos + 1 TO pos
143              ENDFOR
144          ENDIF
145          STORE 11-1 TO 11
146      ENDFOR
147      ? CHR(192) AT 1
148      STORE 12+1 TO 12
149      STORE 2 TO pos
150      FOR i=1 TO mnvx
151          ?? REPLICATE(CHR(196),ROUND(X(i)*3.29,0)) AT pos
152          STORE pos + ROUND(X(i)*3.29,0) TO pos
153          IF i#mnvx
154              ?? CHR(193) AT pos
155          ELSE
156              ?? CHR(217) AT pos
157              ?? 12 PICTURE '##' FUNCTION 'L' AT pos+3
158          ENDIF
159          STORE pos + 1 TO pos
160      ENDFOR
161      ? &nco
162      EJECT
163      ?
164      ? 'PORTICO TIPO'
165      ? '-----'
166      ? &mcom
167      STORE 0 TO 12
168      FOR i=1 TO mnp

```

```

169      STORE 12+hp(i) TO 12
170  ┌   └─ENDFOR
171      ?
172      ?
173      ?
174      ? CHR(65) AT 1
175      STORE 2 TO pos
176      ┌   └─FOR i=1 TO mnvx
177          ?? REPLICATE(CHR(196),ROUND(X(i)*3.29,0)) AT pos
178          STORE pos + ROUND(X(i)*3.29,0) TO pos
179          ?? CHR(65+i) AT pos
180          STORE pos + 1 TO pos
181  ┌   └─ENDFOR
182      ? CHR(218) AT 1
183      STORE 2 TO pos
184      ┌   └─FOR i=1 TO mnvx
185          ?? REPLICATE(CHR(196),ROUND(X(i)*3.29,0)) AT pos
186          STORE pos + ROUND(X(i)*3.29,0) TO pos
187          ┌   └─IF i#mnvx
188              ?? CHR(194) AT pos
189          └─ELSE
190              ?? CHR(191) AT pos
191              ?? 12 PICTURE '##.###' FUNCTION 'L' AT pos+3
192          └─ENDIF
193          STORE pos + 1 TO pos
194  └─ENDFOR
195      ┌   └─FOR l=1 TO mnp
196          ┌   └─FOR j=1 TO ROUND(hp(l),0)
197              ?
198              STORE 1 TO pos
199              ┌   └─FOR K=1 TO mnvx+1
200                  ?? CHR(179) AT pos
201                  STORE pos + ROUND(X(K)*3.29,0) + 1 TO pos
202              └─ENDFOR
203          └─ENDFOR
204          ┌   └─IF l#mnp
205              ? CHR(195) AT 1
206              STORE 2 TO pos
207              ┌   └─FOR i=1 TO mnvx
208                  ?? REPLICATE(CHR(196),ROUND(X(i)*3.29,0)) AT pos
209                  STORE pos + ROUND(X(i)*3.29,0) TO pos
210                  ┌   └─IF i#mnvx
211                      ?? CHR(197) AT pos
212                  └─ELSE
213                      ?? CHR(180) AT pos
214                      STORE 12-hp(l) TO 12
215                      ?? 12 PICTURE '##.###' FUNCTION 'L' AT pos+3
216                  └─ENDIF
217                  STORE pos + 1 TO pos
218              └─ENDFOR
219          └─ENDIF
220      └─ENDFOR
221      ? CHR(207) AT 1
222      STORE 2 TO pos
223      ┌   └─FOR i=1 TO mnvx
224          STORE pos + ROUND(X(i)*3.29,0) TO pos

```

```

225      IF i#mnvx
226          ?? CHR(207) AT pos
227      ELSE
228          ?? CHR(207) AT pos
229          STORE 12-hp(mnp) TO 12
230          ?? 12 PICTURE '##.###' FUNCTION 'L' AT pos+3
231      ENDIF
232      STORE pos + 1 TO pos
233  ENDFOR
234  ?
235  ? ' EJE SELECCIONADO --> '+ALLTRIM(STR(mxx))
236  ? &mnco
237  EJECT
238  SET PRINT OFF
239  SET DEVICE TO SCREEN
240  ENDIF
241  DEACTIVATE WINDOW reportes
242  DEACTIVATE WINDOW impresor
243  RETURN
244  *: EOF: REP0023.ACT

```

```

1  *:*****
2  *:
3  *: Procedure file: C:\CIVIL\ROUTINAS.PRG
4  *:
5  *:      System: Anal.y Diseño de Estructuras Aporticadas
6  *:      Author: Rommel R. Alemán T. - Marco P. Silva S.
7  *:      Copyright (c) 1993, Pontificia Universidad Catolica del E.
8  *:      Last modified: 05/22/93      11:59
9  *:
10 *:  Procs & Fncts: STOP
11 *:                      : ESCOJE
12 *:                      : ESCOJER
13 *:
14 *:      Documented 05/28/93 at 12:12      FoxDoc version 2.10
15 *:*****
29 *!*****
30 *!
31 *!      Procedure: STOP
32 *!
33 *!      Called by: MAX00.PRG
34 *!                      : MENU001.PRG
35 *!                      : MENU002.PRG
36 *!                      : MENU003.PRG
37 *!                      : MENU004.PRG
38 *!                      : MENU005.PRG
39 *!                      : MENU006.PRG
40 *!
41 *!*****
29  PROCEDURE stop
30  PARAMETER merror, mensa1, mensa2, mprog, mlinnum
31  SET PRINT OFF
32  SET DEVICE TO SCREEN
33  ACTIVATE WINDOW memos
34  CLEAR
35  @00.02 SAY '      Número de Error :'+ALLTRIM(STR(merror))
36  @01.02 SAY '      Mensaje del Error :'+ALLTRIM(mensa1)
37  @02.02 SAY ' Error de la línea de codigo :'+ALLTRIM(mensa2)
38  @04.02 SAY ' Error en la línea número :'+ALLTRIM(STR(mlinnum))
39  @05.02 SAY 'Programa que produjo el error :'+ALLTRIM(mprog)
40  STORE 0 TO op
41  @06.25 PROMPT '\<Reintentar'
42  @06.38 PROMPT '\<Cancelar'
43  MENU TO op
44  DO CASE
45  CASE op=1
46  IF ERROR()=125
47  SET DEVICE TO PRINT
48  SET PRINT ON
49  ENDIF
50  RETRY
51  CASE op=2
52  <====CANCEL
53  ENDCASE
54  DEACTIVATE WINDOW memos
55  RETURN
63 *!*****

```

```

64 *!
65 *!      Procedure: ESCOJE
66 *!
67 *!      Called by: BUSCAR          (procedure in PRED001.PRG)
68 *!
69 *!*****
70      PROCEDURE escoje
71      PARAMETER mcategoria,maux
72      DEACTIVATE POPUP localiza
73      RETURN
74 *!*****
75 *!
76 *!      Procedure: ESCOJER
77 *!
78 *!      Called by: EDITAR          (procedure in PRED001.PRG)
79 *!
80 *!*****
81      PROCEDURE escojer
82      PARAMETER mcategoria,maux
83      DEACTIVATE POPUP cargas
84      RETURN
85 *! EOF: RUTINAS.ACT

```

```

1  *:*****
2  *:
3  *:      Program: C:\CIVIL\SETENV.PRG
4  *:
5  *:      System: Anal.y Diseño de Estructuras Aporticadas
6  *:      Author: Rommel R. Alemán T. - Marco P. Silva S.
7  *:      Copyright (c) 1993, Pontificia Universidad Catolica del E.
8  *:      Last modified: 05/23/93      10:49
9  *:
10 *:      Called by: CIVIL.PRG
11 *:              : MAX00.PRG
12 *:
13 *:      Memory Files: IMPRESORA.MEM
14 *:              : COLORES.MEM
15 *:
16 *:      Other Files: &MPTO
17 *:              : HELP
18 *:
19 *:      Documented 05/28/93 at 12:12      FoxDoc version 2.10
20 *:*****
21  IF DISKSPACE(<) < 1000000
22  WAIT 'El espacio minimo en disco es de 1Mbyte ...' WINDOW
23  <====QUIT
24  └─ENDIF
25  IF VAL(SYS(12)) < 64000
26  WAIT 'Insuficiente memoria para ejecutar CIVIL...' WINDOW
27  <====QUIT
28  └─ENDIF
29  RESTORE FROM impresora
30  RESTORE FROM colores ADDITIVE
31  SET ALTERNATE      OFF      && Default
32  SET ALTERNATE TO   && Default
33  SET AUTOSAVE      OFF
34  SET BELL          ON      && Default
35  SET BLOCKSIZE TO  33      && Default
36  SET BLINK         OFF      && Default
37  SET BORDER TO     SINGLE
38  SET CENTURY       OFF
39  SET CLEAR         ON
40  SET CLOCK         ON      && Default
41  SET CLOCK TO      24.01
42  SET COLOR SET TO  &mtpo
43  SET COMPATIBLE    OFF
44  SET CONFIRM       OFF
45  SET CONSOLE       ON
46  SET CURRENCY      LEFT
47  SET DATE          AMERICAN && Default
48  SET DEBUG         OFF
49  SET DEFAULT TO    && LEAVE SET TO CURRENT DEFAULT DRIVE
50  SET DELETE        ON
51  SET DEVICE TO     SCREEN  && Default
52  SET DECIMALS TO   18
53  SET ECHO          OFF      && Default
54  SET ESCAPE        OFF      && Default
55  SET EXACT         ON      && Default
56  SET FULLPATH      ON

```

```

57  SET HELP          ON
58  SET INTENSITY     OFF
59  SET MARK TO       "-"      && THIS IS THE DATE FIELD SEPARATOR
60  SET MEMOWIDTH TO  70
61  SET PATH TO       ""      && ACCEPT DEFAULT PATHS
62  SET POINT TO      "."
63  SET PRINT         OFF
64  SET PRINTER TO    &empto
65  SET SAFETY        OFF
66  SET SCOREBOARD    OFF
67  SET SEPARATOR TO  ","
68  SET SHADOWS       ON
69  SET STATUS        OFF
70  SET SYSMENU       ON
71  SET TALK          OFF
72  SET TOPIC TO
73  SET HELP TO HELP
74  ON KEY LABEL f1 HELP
75  ON KEY LABEL f3 ACTIVATE WINDOW calculator
76  ON KEY LABEL f4 ACTIVATE WINDOW calendar
77  ON KEY LABEL f5 ACTIVATE WINDOW ascii
78  ON KEY LABEL f6 ACTIVATE WINDOW special
79  RETURN
81  *: EOF: SETENV.ACT

```

```

1  *:*****
2  *:
3  *:      Program: C:\CIVIL\SIST001.PRG
4  *:
5  *:      System: Anal.y Diseño de Estructuras Aporticadas
6  *:      Author: Rommel R. Alemán T. - Marco P. Silva S.
7  *:      Copyright (c) 1993, Pontificia Universidad Catolica del E.
8  *:      Last modified: 04/28/93      18:52
9  *:
10 *:      Called by: MENU004.PRG
11 *:
12 *:      Memory Files: DEFECTO.MEM
13 *:
14 *:      Documented 05/28/93 at 12:15      FoxDoc version 2.10
15 *:*****
16  RELEASE ALL
17  ACTIVATE WINDOW trabajo
18  CLEAR
19  RESTORE FROM defecto
20  *store 0 to mfc, mn, mfy, mel, mpb, mpp, mea, mr
21  *store 0 to e1, e2, e3, mph, mbeta, mfil, mfi2, mfi3
22  @02,10 SAY ** DATOS POR DEFECTO PARA EL PREDISEÑO **
23  @04,10 SAY '      Resistencia del hormigón : '
24  @05,10 SAY '      Resistencia del acero : '
25  @06,10 SAY '      Espesor de loseta : '
26  @07,10 SAY '      Peso de bloques : '
27  @08,10 SAY '      Peso de paredes : '
28  @09,10 SAY '      Espesor de acabados : '
29  @10,10 SAY '      % de acero minimo : '
30  @11,10 SAY '      Constante de esfuerzo unitario : '
31  @13,10 SAY ** DATOS POR DEFECTO PARA EL DISEÑO **
32  @15,10 SAY '      Coeficiente de hormigón : '
33  @16,10 SAY '      Coeficiente de flexión de viga : '
34  @17,10 SAY '      Coeficiente de flexión de columna : '
35  @18,10 SAY '      Coeficiente de corte y torsión : '
36  @10,47 SAY mph PICTURE '###.###'
37  @11,47 SAY mr PICTURE '###.##'
38  IF mfc <=0
39  |   SELECT tabla
40  |   BROWSE WINDOW tbla
41  |   STORE fc TO mfc
42  |   STORE N TO mn
43  |_ENDIF
44  @04,47 GET mfc PICTURE '### Kg/cm²'
45  @05,47 GET mfy PICTURE '### Kg/cm²'
46  @06,47 GET mel PICTURE '#.### m'
47  @07,47 GET mpb PICTURE '#.### Tn/m²'
48  @08,47 GET mpp PICTURE '#.### Tn/m²'
49  @09,47 GET mea PICTURE '#.### m'
50  @15,47 GET mbeta PICTURE '#.###'
51  @16,47 GET mfil PICTURE '#.###'
52  @17,47 GET mfi2 PICTURE '#.###'
53  @18,47 GET mfi3 PICTURE '#.###'
54  READ
55  STORE ROUND((mbeta^2*(mfc*6090)/(mfy*(6090+mfy)))*0.75,4) TO mph
56  =DO CASE

```

```

57  CASE mph>=0.0100 .AND. mph<=0.0129
58      STORE 14.34 TO mr
59  CASE mph>=0.0130 .AND. mph<=0.0150
60      STORE 15.94 TO mr
61  CASE mph>=0.0151 .AND. mph<=0.0160
62      STORE 16.73 TO mr
63  OTHERWISE
64      STORE 16.73 TO mr
65  ENDCASE
66      @10,47 SAY mph PICTURE '##.###'
67      @11,47 SAY mr  PICTURE '##.##'
68      IF LASTKEY()=27
69          DEACTIVATE WINDOW trabajo
70  <====RETURN
71  ENDF
72      SAVE TO defecto ALL LIKE m????
73      DEACTIVATE WINDOW trabajo
74      RETURN
76      *: EOF: SIST001.ACT

```

```

1  *;*****
2  *:
3  *:      Program: C:\CIVIL\SIST002.PRG
4  *:
5  *:      System: Anal.y Diseño de Estructuras Aporticadas
6  *:      Author: Rommel R. Alemán T. - Marco P. Silva S.
7  *:      Copyright (c) 1993, Pontificia Universidad Catolica del E.
8  *:      Last modified: 04/28/93      18:57
9  *:
10 *:      Called by: MENU004.PRG
11 *:
12 *:      Memory Files: COLORES.MEM
13 *:
14 *:      Documented 05/28/93 at 12:15      FoxDoc version 2.10
15 *;*****
16  RELEASE ALL
17  ACTIVATE WINDOW sistema
18  CLEAR
19  RESTORE FROM colores ADDITIVE
20  *STORE 0 TO MCR1,MCR2,MCR3,MCR4,MCR5,MCR6,MCR7,MCR8,MCR9,MCR10
21  *STORE SPACE(5) TO MTP0
22  @00,20 TO 02,50 DOUBLE COLOR SCHEME 7
23  @01,21 SAY '*** PARAMETROS DE MONITOR ***' COLOR SCHEME 11
24  @02,01 TO 04,16
25  @03,03 SAY 'TIPO : ' GET mtp0 PICTURE '@M Color, Mono'
26  @05,01 TO 18,68
27
28  @06,02 SAY '[01]'
29  @06,10 SAY SCHEME(mcr1)
30  @06,07 GET mcr1 PICTURE '##' RANGE 1,19
31
32  @07,02 SAY '[02]'
33  @07,10 SAY SCHEME(mcr2)
34  @07,07 GET mcr2 PICTURE '##' RANGE 1,19
35
36  @08,02 SAY '[03]'
37  @08,10 SAY SCHEME(mcr3)
38  @08,07 GET mcr3 PICTURE '##' RANGE 1,19
39
40  @09,02 SAY '[04]'
41  @09,10 SAY SCHEME(mcr4)
42  @09,07 GET mcr4 PICTURE '##' RANGE 1,19
43
44  @10,02 SAY '[05]'
45  @10,10 SAY SCHEME(mcr5)
46  @10,07 GET mcr5 PICTURE '##' RANGE 1,19
47
48  @11,02 SAY '[06]'
49  @11,10 SAY SCHEME(mcr6)
50  @11,07 GET mcr6 PICTURE '##' RANGE 1,19
51
52  @12,02 SAY '[07]'
53  @12,10 SAY SCHEME(mcr7)
54  @12,07 GET mcr7 PICTURE '##' RANGE 1,19
55
56  @13,02 SAY '[08]'

```

```

57 @13,10 SAY SCHEME(mcr8)
58 @13,07 GET mcr8 PICTURE '##' RANGE 1,19
59
60 @14,02 SAY '[09]'
61 @14,10 SAY SCHEME(mcr9)
62 @14,07 GET mcr9 PICTURE '##' RANGE 1,19
63
64 @15,02 SAY '[10]'
65 @15,10 SAY SCHEME(mcr10)
66 @15,07 GET mcr10 PICTURE '##' RANGE 1,19
67 READ
68 IF LASTKEY()=27
69     DEACTIVATE WINDOW sistema
70 <=====RETURN
71 ENDIF
72 SAVE TO colores ALL LIKE m????
73 SET COLOR SET TO &mtpo
74 RELEASE WINDOW ALL
75 DEFINE WINDOW tbla FROM 08,50 TO 16,75 PANEL SHADOW:
76     TITLE 'Civil 01' COLOR SCHEME mcr1
77 DEFINE WINDOW sistema FROM 03,04 TO 21,75 DOUBLE SHADOW:
78     TITLE 'Civil 02' COLOR SCHEME mcr2
79 DEFINE WINDOW vano FROM 02,65 TO 23,79 DOUBLE SHADOW:
80     TITLE 'Civil 03' COLOR SCHEME mcr3
81 DEFINE WINDOW disenio FROM 06,02 TO 20,77 DOUBLE SHADOW:
82     TITLE 'Civil 04' COLOR SCHEME mcr4
83 DEFINE WINDOW reportes FROM 06,06 TO 20,73 DOUBLE SHADOW:
84     TITLE 'Civil 05' COLOR SCHEME mcr5
85 DEFINE WINDOW impresor FROM 19,00 TO 23,79 DOUBLE SHADOW:
86     TITLE 'Civil 06' COLOR SCHEME mcr6
87 DEFINE WINDOW mensajes FROM 08,02 TO 12,76 DOUBLE SHADOW:
88     TITLE 'Civil 07' COLOR SCHEME mcr7
89 DEFINE WINDOW memos FROM 08,03 TO 16,76 DOUBLE SHADOW:
90     TITLE 'Civil 08' COLOR SCHEME mcr8
91 DEFINE WINDOW trabajo FROM 01,00 TO 23,79 PANEL SHADOW:
92     TITLE 'Civil 09' COLOR SCHEME mcr9
93 DEFINE WINDOW tabla FROM 03,02 TO 21,77 PANEL SHADOW:
94     TITLE 'Civil 10' CLOSE FLOAT GROW ZOOM COLOR SCHEME mcr10
95 DEACTIVATE WINDOW sistema
96 *: EOF: SIST002.ACT

```

```

1  *:*****
2  *:
3  *:      Program: C:\CIVIL\SIST003.PRG
4  *:
5  *:      System: Anal.y Diseño de Estructuras Aporticadas
6  *:      Author: Rommel R. Alemán T. - Marco P. Silva S.
7  *:      Copyright (c) 1993, Pontificia Universidad Catolica del E.
8  *:      Last modified: 05/15/93      20:06
9  *:
10 *:      Called by: MENU004.PRG
11 *:
12 *:      Memory Files: IMPRESORA.MEM
13 *:
14 *:      Documented 05/28/93 at 12:15      FoxDoc version 2.10
15 *:*****
16  RELEASE ALL
17  ACTIVATE WINDOW sistema
18  CLEAR
19  RESTORE FROM impresora ADDITIVE
20  *STORE SPACE(4) TO MPTO
21  STORE LEFT(mexp+SPACE(47),47) TO mexp
22  STORE LEFT(mnex+SPACE(47),47) TO mnex
23  STORE LEFT(mcom+SPACE(47),47) TO mcom
24  STORE LEFT(mnco+SPACE(47),47) TO mnco
25  @00,20 TO 02.50 DOUBLE COLOR SCHEME 7
26  @01,21 SAY '** PARAMETROS DE IMPRESORA **' COLOR SCHEME 11
27  @04,01 TO 06,17
28  @05,03 SAY 'PUERTO : ' GET mpto PICTURE '@M Lpt1, Lpt2, Lpt3'
29  @04,19 TO 06,68
30  @05,20 SAY '*CADENA DE CARACTERES PARA TITULOS Y RESULTADOS*'
31  @08,20 TO 09,68
32  @08,21 GET mexp
33  @08,20 SAY CHR(180)
34  @08,19 SAY CHR(218)
35  @08,01 TO 10,17
36  @10,19 SAY CHR(192)
37  @09,19 SAY CHR(180)
38  @09,18 SAY CHR(196)
39  @09,17 SAY CHR(195)
40  @09,05 SAY 'TITULOS'
41  @10,20 TO 11,68
42  @10,21 GET mnex
43  @10,20 SAY CHR(180)
44  @13,20 TO 14,68
45  @13,21 GET mcom
46  @13,20 SAY CHR(180)
47  @13,19 SAY CHR(218)
48  @13,01 TO 15,17
49  @15,19 SAY CHR(192)
50  @14,19 SAY CHR(180)
51  @14,18 SAY CHR(196)
52  @14,17 SAY CHR(195)
53  @14,04 SAY 'RESULTADOS'
54  @15,20 TO 16,68
55  @15,21 GET mnco
56  @15,20 SAY CHR(180)

```

```
57     READ
58     IF LASTKEY()=27
59     DEACTIVATE WINDOW sistema
60 <====RETURN
61     ENDIF
62     SAVE TO impresora ALL LIKE m???
63     DEACTIVATE WINDOW sistema
64     @24,57 SAY mpto COLOR SCHEME 5
65     RETURN
67     *: EOF: SIST003.ACT
```

```

1  *:*****
2  *:
3  *: Procedure file: C:\CIVIL\SIST004.PRG
4  *:
5  *:      System: Anal.y Diseño de Estructuras Aporticadas
6  *:      Author: Rommel R. Alemán T. - Marco P. Silva S.
7  *:      Copyright (c) 1993, Pontificia Universidad Catolica del E.
8  *:      Last modified: 05/15/93      12:08
9  *:
10 *:      Set by: MENU004.PRG
11 *:
12 *:      Calls: SETUP_REP      (procedure in PRED001.PRG)
13 *:              : DESPLEGAR      (procedure in PRED001.PRG)
14 *:              : MOSTRAR      (procedure in PRED001.PRG)
15 *:
16 *:      Documented 05/28/93 at 12:15      FoxDoc version 2.10
17 *:*****
18     RELEASE ALL
19     SET FUNCTION f2 TO 'B;'
20     STORE 0 TO mauX
21     DO setup_rep
22     SELECT usuarios
23     SET ORDER TO 1
24     IF EOF()
25     [   GO TOP
26     ]ENDIF
27     STORE codigo TO mcodigo
28     ACTIVATE WINDOW disenio
29     DO desplegar
30     DO mostrar
31     STORE .T. TO reporte
32     [DO WHILE reporte
33     [   ACTIVATE MENU salesrep PAD search
34     ]ENDDO
35     DEACTIVATE WINDOW disenio
36     RELEASE MENU salesrep
37     CLOSE PROCEDURE
38     RETURN
53  *!*****
54  *!
55  *!      Procedure: INCREMENTAR
56  *!
57  *!      Called by: SETUP_REP      (procedure in PRED001.PRG)
58  *!
59  *!      Calls: DESPLEGAR      (procedure in PRED001.PRG)
60  *!              : MOSTRAR      (procedure in PRED001.PRG)
61  *!              : STANDBY      (procedure in PRED001.PRG)
62  *!              : EDITAR      (procedure in PRED001.PRG)
63  *!
64  *!      Memory Files: DEFECTO.MEM
65  *!
66  *!*****
67  *!      PROCEDURE incrementar
68  *!      GO BOTTOM
69  *!      STORE codigo TO mcodigo
70  *!      DO desplegar

```

```

57     STORE mcodigo+1 TO mcodigo
58     @02.33 GET mcodigo PICTURE '####' RANGE 1.99999
59     READ
60     IF LASTKEY()=27
61     <=====RETURN
62     ENDIF
63     SEEK mcodigo
64     IF FOUND()
65     DO mostrar
66     DO standby WITH "No puedo añadir el registro: el registro ya existe"
67     <=====RETURN
68     ELSE
69     APPEND BLANK
70     REPLACE NEXT 1 codigo WITH mcodigo
71     ENDIF
72     DO editar
73     RETURN
87  *!*****
88  *!
89  *!     Procedure: BORRAR
90  *!
91  *!     Called by: SETUP_REP           (procedure in PRED001.PRG)
92  *!
93  *!     Calls: MOSTRAR                (procedure in PRED001.PRG)
94  *!
95  *!     Uses: &NOM1.DBF
96  *!           : &NOM2.DBF
97  *!           : &NOM3.DBF
98  *!
99  *!*****
87  PROCEDURE borrar
88  STORE SPACE(1) TO do_it
89  ACTIVATE WINDOW mensajes
90  @01.02 SAY "Esta usted seguro de querer borrar el registro [S/N] : " GET do_it PICT "@M N,S"
91  READ
92  DEACTIVATE WINDOW mensajes
93  IF do_it # "S"
94  <=====RETURN
95  ENDIF
96  IF RECCOUNT()#0
97  DELETE NEXT 1
98  IF .NOT. EOF()
99  SKIP
100 ELSE
101 GO BOTTOM
102 ENDIF
103 ENDIF
104 DO mostrar
105 RETURN
116 *!*****
117 *!
118 *!     Procedure: DESPLEGAR
119 *!
120 *!     Called by: PRED001.PRG
121 *!           : DISE001.PRG
122 *!           : SIST004.PRG

```

```

123 *!           : INCREMETAR           (procedure in PRED001.PRG)
124 *!
125 *!*****
116     PROCEDURE desplegar
117     CLEAR
118     @02,10 SAY '           Clave No.: '
119     @05,10 SAY '     Nombre del Usuario: '
120     @07,10 SAY '     Clave del Usuario: '
121     @09,10 SAY '     Prioridad de acceso: '
122     RETURN
135 *!*****
136 *!
137 *!     Procedure: EDITAR
138 *!
139 *!     Called by: SETUP_REP           (procedure in PRED001.PRG)
140 *!           : INCREMETAR           (procedure in PRED001.PRG)
141 *!
142 *!     Calls: MOSTRAR                 (procedure in PRED001.PRG)
143 *!           : ESCOJER               (procedure in RUTINAS.PRG)
144 *!           : DATOS                 (procedure in PRED001.PRG)
145 *!
146 *!*****
135     PROCEDURE editar
136     IF RECCOUNT()#0
137         @02,33 GET codigo PICTURE '####'
138         CLEAR GETS
139         @05,33 GET profesor PICTURE '!!!!!!!!!!!!!!!!!!!!!!'
140         @07,33 GET identida PICTURE '!!!!'
141         @09,33 GET prioridad PICTURE '##' RANGE 1,99
142         READ
143         DO mostrar
144     ENDIF
145     RETURN
155 *!*****
156 *!
157 *!     Procedure: FINAL
158 *!
159 *!     Called by: SETUP_REP           (procedure in PRED001.PRG)
160 *!
161 *!     Calls: MOSTRAR                 (procedure in PRED001.PRG)
162 *!
163 *!*****
155     PROCEDURE final
156     GO BOTTOM
157     DO mostrar
158     RETURN
169 *!*****
170 *!
171 *!     Procedure: PROXIMO
172 *!
173 *!     Called by: SETUP_REP           (procedure in PRED001.PRG)
174 *!
175 *!     Calls: STANDEY                 (procedure in PRED001.PRG)
176 *!           : MOSTRAR               (procedure in PRED001.PRG)
177 *!
178 *!*****

```

```

169     PROCEDURE proximo
170     IF RECCOUNT()#0
171     SKIP
172     IF EOF()
173     GO BOTTOM
174     DO standby WITH 'El registro presente es el ultimo...'
175     ELSE
176     DO mostrar
177     ENDIF
178     ENDIF
179     RETURN
190     *!*****
191     *!
192     *!     Procedure: ANTERIOR
193     *!
194     *!     Called by: SETUP_REP           (procedure in PRED001.PRG)
195     *!
196     *!     Calls: STANDBY                 (procedure in PRED001.PRG)
197     *!           : MOSTRAR                (procedure in PRED001.PRG)
198     *!
199     *!*****
200     PROCEDURE anterior
201     IF RECCOUNT()#0
202     SKIP -1
203     IF BOF()
204     GO TOP
205     DO standby WITH 'El registro presente es el primero...'
206     ELSE
207     DO mostrar
208     ENDIF
209     ENDIF
210     RETURN
211     *!*****
212     *!
213     *!     Procedure: SALIR
214     *!
215     *!     Called by: SETUP_REP           (procedure in PRED001.PRG)
216     *!
217     *!*****
218     PROCEDURE salir
219     STORE .F. TO reporte
220     DEACTIVATE MENU
221     RETURN
222     *!*****
223     *!
224     *!     Procedure: SETUP_REP
225     *!
226     *!     Called by: PRED001.PRG
227     *!           : DISE001.PRG
228     *!           : SIST004.PRG
229     *!
230     *!     Calls: PROXIMO                 (procedure in PRED001.PRG)
231     *!           : ANTERIOR              (procedure in PRED001.PRG)
232     *!           : INICIO                (procedure in PRED001.PRG)
233     *!           : FINAL                 (procedure in PRED001.PRG)
234     *!           : BUSCAR                (procedure in PRED001.PRG)

```

```

244 *!           : EDITAR           (procedure in PRED001.PRG)
245 *!           : INCREMETAR       (procedure in PRED001.PRG)
246 *!           : BORRAR           (procedure in PRED001.PRG)
247 *!           : SALIR            (procedure in PRED001.PRG)
248 *!
249 *!*****
231 PROCEDURE setup_rep
232 DEFINE MENU salesrep COLOR SCHEME 3
233 DEFINE PAD NEXT OF salesrep PROMPT '\<Prox' AT 12,01
234 DEFINE PAD PREV OF salesrep PROMPT '\<Ante' AT 12,08
235 DEFINE PAD first OF salesrep PROMPT '\<Inicio' AT 12,15
236 DEFINE PAD LAST OF salesrep PROMPT '\<Fin' AT 12,24
237 DEFINE PAD search OF salesrep PROMPT '\<Buscar' AT 12,30
238 DEFINE PAD EDIT OF salesrep PROMPT '\<Editar' AT 12,39
239 DEFINE PAD APPEND OF salesrep PROMPT 'aña\<Dir' AT 12,48
240 DEFINE PAD DELETE OF salesrep PROMPT 'bor\<Rar' AT 12,57
241 DEFINE PAD QUIT OF salesrep PROMPT '\<Salir' AT 12,66
242 ON SELECTION PAD NEXT OF salesrep DO proximo
243 ON SELECTION PAD PREV OF salesrep DO anterior
244 ON SELECTION PAD first OF salesrep DO inicio
245 ON SELECTION PAD LAST OF salesrep DO final
246 ON SELECTION PAD search OF salesrep DO buscar
247 ON SELECTION PAD EDIT OF salesrep DO editar
248 ON SELECTION PAD APPEND OF salesrep DO incremetar
249 ON SELECTION PAD DELETE OF salesrep DO borrar
250 ON SELECTION PAD QUIT OF salesrep DO salir
251 RETURN
269 *!*****
270 *!
271 *! Procedure: MOSTRAR
272 *!
273 *! Called by: PRED001.PRG
274 *!           : DISE001.PRG
275 *!           : SIST004.PRG
276 *!           : INCREMETAR       (procedure in PRED001.PRG)
277 *!           : BORRAR           (procedure in PRED001.PRG)
278 *!           : EDITAR           (procedure in PRED001.PRG)
279 *!           : FINAL            (procedure in PRED001.PRG)
280 *!           : PROXIMO          (procedure in PRED001.PRG)
281 *!           : ANTERIOR         (procedure in PRED001.PRG)
282 *!           : BUSCAR           (procedure in PRED001.PRG)
283 *!           : INICIO           (procedure in PRED001.PRG)
284 *!
285 *!*****
269 PROCEDURE mostrar
270 @02,33 SAY codigo PICTURE '####'
271 @05,33 SAY profesor PICTURE '!!!!!!!!!!!!!!!!!!!!!!!!!!!!'
272 @07,33 SAY identida PICTURE '!!!!!!'
273 @09,33 SAY prioridad PICTURE '##'
274 RETURN
286 *!*****
287 *!
288 *! Procedure: BUSCAR
289 *!
290 *! Called by: SETUP_REP         (procedure in PRED001.PRG)
291 *!

```

```

292 *!          Calls: ESCOJE                (procedure in RUTINAS.PRG)
293 *!          : MOSTRAR                    (procedure in PRED001.PRG)
294 *!          : STANDBY                    (procedure in PRED001.PRG)
295 *!
296 *!*****
286     PROCEDURE buscar
287     last_rec=RECNO()
288     STORE 0 TO mcodigo
289     @02,33 GET mcodigo PICTURE '####'
290     READ
291     IF mcodigo=0
292     DEFINE POPUP localiza FROM 00,45 TO 20,77 PROMPT FIELD profesor SHADOW COLOR SCHEME 7
293     ON SELECTION POPUP localiza DO escoje WITH PROMPT(), BAR()
294     ACTIVATE POPUP localiza
295     STORE codigo TO mcodigo
296     ENDIF
297     IF mcodigo=0
298     DO mostrar
299     <===== RETURN
300     ENDIF
301     SEEK mcodigo
302     IF .NOT. FOUND()
303     DO standby WITH 'El registro no ha sido encontrado en el archivo'
304     GO last_rec
305     ENDIF
306     DO mostrar
307     RETURN
317 *!*****
318 *!
319 *!          Procedure: INICIO
320 *!
321 *!          Called by: SETUP_REP            (procedure in PRED001.PRG)
322 *!
323 *!          Calls: MOSTRAR                  (procedure in PRED001.PRG)
324 *!
325 *!*****
317     PROCEDURE inicio
318     GO TOP
319     DO mostrar
320     RETURN
331 *!*****
332 *!
333 *!          Procedure: STANDBY
334 *!
335 *!          Called by: INCREMENTAR          (procedure in PRED001.PRG)
336 *!          : PROXIMO                      (procedure in PRED001.PRG)
337 *!          : ANTERIOR                     (procedure in PRED001.PRG)
338 *!          : BUSCAR                       (procedure in PRED001.PRG)
339 *!
340 *!*****
331     PROCEDURE standby
332     PARAMETER mensaje
333     ACTIVATE WINDOW mensajes
334     STORE INT((59 - LEN(mensaje))/2) TO ln
335     @01,ln SAY mensaje
336     READ

```

337 DEACTIVATE WINDOW mensajes
338 RETURN
340 *: EOF: SIST004.ACT

```
1  *:*****
2  *:
3  *:      Program: C:\CIVIL\SYSTEM.PRG
4  *:
5  *:      System: Anal.y Diseño de Estructuras Aporticadas
6  *:      Author: Rommel R. Alemán T. - Marco P. Silva S.
7  *:      Copyright (c) 1993, Pontificia Universidad Catolica del E.
8  *:      Last modified: 05/23/93      10:52
9  *:
10 *:      Called by: MAX00.PRG
11 *:
12 *:      Calls: MAX01.PRG
13 *:
14 *:      Documented 05/28/93 at 12:12      FoxDoc version 2.10
15 *:*****
16      DO max01
17      ACTIVATE MENU menuprin
18      RELEASE MENU menuprin
19      CLOSE ALL
20      CLEAR ALL
21      QUIT
22      RETURN
24      *: EOF: SYSTEM.ACT
```

```

1  *:*****
2  *:
3  *:      Program: C:\CIVIL\TERM001.PRG
4  *:
5  *:      System: Anal.y Diseño de Estructuras Aporticadas
6  *:      Author: Rommel R. Alemán T. - Marco P. Silva S.
7  *:      Copyright (c) 1993, Pontificia Universidad Catolica del E.
8  *:      Last modified: 05/28/93      11:35
9  *:
10 *:      Called by: MENU006.PRG
11 *:
12 *:      Documented 05/28/93 at 12:15      FoxDoc version 2.10
13 *:*****
14  ACTIVATE WINDOW mensajes
15  CLEAR
16  @01,01 SAY 'REUVICANDO ARCHIVO DE AYUDA EN LINEA...'
17  SET HELP TO
18  FOR i=1 TO 500
19  [   FOR j=1 TO 10
20  [   ENDFOR
21  [   ENDFOR
22  CLEAR
23  @01,01 SAY 'SERRANDO BASES DE DATOS, INDICES, ETC...'
24  [   FOR i=1 TO 500
25  [   FOR j=1 TO 10
26  [   ENDFOR
27  [   ENDFOR
28  CLEAR
29  @01,01 SAY 'DESACTIVANDO VENTANAS, ENSERANDO VARIABLES DE MEMORIA...'
30  [   FOR i=1 TO 500
31  [   FOR j=1 TO 10
32  [   ENDFOR
33  [   ENDFOR
34  CLEAR
35  @01,01 SAY 'TERMINACION NORMAL DEL SISTEMA CIVIL...'
36  [   FOR i=1 TO 500
37  [   FOR j=1 TO 10
38  [   ENDFOR
39  [   ENDFOR
40  *DEACTIVATE MENU
41  *CLOSE ALL
42  *CLEAR ALL
43  QUIT
44  RETURN
46  *: EOF: TERM001.ACT

```

```

1  *:*****
2  *:
3  *:      Program: C:\CIVIL\UTIL001.PRG
4  *:
5  *:      System: Anal.y Diseño de Estructuras Aporticadas
6  *:      Author: Rommel R. Alemán T. - Marco P. Silva S.
7  *:      Copyright (c) 1993, Pontificia Universidad Catolica del E.
8  *:      Last modified: 05/22/93      13:35
9  *:
10 *:      Called by: MENU005.PRG
11 *:
12 *:      Other Files: CONFIG.SYS
13 *:
14 *:      Documented 05/28/93 at 12:15      FoxDoc version 2.10
15 *:*****
16     STORE 'c:' TO mdrive
17     DO WHILE .NOT. FILE(mdrive + '\CONFIG.SYS')
18         STORE ' ' TO mdrive
19         ACTIVATE WINDOW mensajes
20         @00,01 SAY "Su sistema no contiene el archivo CONFIG.SYS"
21         @01,01 SAY "Ingrese el drive o [ESCAPE] para regresar al menu."
22         @02,01 SAY "Localización de CONFIG.SYS es el drive: " GET mdrive PICTURE '! '
23         READ
24         IF mdrive = SPACE(1)
25             DEACTIVATE WINDOW mensajes
26         RETURN
27     ENDIF
28     STORE mdrive + ':' TO mdrive
29     ENDDO
30     config = FOPEN(mdrive + '\CONFIG.SYS')
31     STORE 0 TO fileset, bufferset
32     STORE .T. TO trabajo
33     DO WHILE trabajo .AND. FERROR() = 0
34         inputline = FGETS(config)
35         DO CASE
36             CASE 'FILES' $ UPPER(inputline)
37                 STORE VAL(LTRIM(SUBSTR(inputline,AT('=',inputline)+1))) TO fileset
38             CASE 'BUFFERS' $ UPPER(inputline)
39                 STORE VAL(LTRIM(SUBSTR(inputline,AT('=',inputline)+1))) TO bufferset
40             ENDCASE
41             IF fileset # 0 .AND. bufferset # 0
42                 STORE .F. TO trabajo
43             ENDIF
44         ENDDO
45         = FCLOSE(config)
46         ACTIVATE WINDOW tabla
47         @00,23 SAY 'INFORMACION DE CONFIGURACION'
48         @02,01 SAY '      FILES : ' + LTRIM(STR(fileset,4))
49         @03,01 SAY '      BUFFERS : ' + LTRIM(STR(bufferset,4))
50         @05,01 SAY '      MONITOR : ' + SYS(2006)
51         @06,01 SAY '      PANTALLA : ' + STR(SROWS(),2) + ' BY ' + TRIM(STR(SCOLS(),3))
52         @08,01 SAY '      FOXPRO ver. : ' + VERSION(1)
53         @09,01 SAY '      DIRECTORIO : ' + CURDIR()
54         @11,01 SAY '      MEM. BASE FOX : ' + LTRIM(SYS(12))
55         @12,01 SAY '      W/EMS MEMORIA : ' + LTRIM(SYS(1001))
56         @14,01 SAY '      DISCO Kbytes : ' + LTRIM(TRANSFORM(DISKSPACE(), '999,999,999'))

```

```
57      WAIT 'Presione una tecla para continuar...' WINDOW
58      DEACTIVATE WINDOW tabla
59      RETURN
60
62      *: EOF: UTIL001.ACT
```

```

1  *:*****
2  *:
3  *:      Program: C:\CIVIL\UTIL002.PRG
4  *:
5  *:      System: Anal.y Diseño de Estructuras Aporticadas
6  *:      Author: Rommel R. Alemán T. - Marco P. Silva S.
7  *:      Copyright (c) 1993, Pontificia Universidad Catolica del E.
8  *:      Last modified: 02/28/93      9:12
9  *:
10 *:      Called by: MENU005.PRG
11 *:
12 *:      Documented 05/28/93 at 12:15      FoxDoc version 2.10
13 *:*****
14      ACTIVATE WINDOW mensajes
15      CLEAR
16      STORE SPACE(55) TO camino
17      @01.01 SAY "CAMINO:" GET camino
18      READ
19      IF LASTKEY() = 27
20      |      DEACTIVATE WINDOW tabla
21      |      DEACTIVATE WINDOW mensajes
22  <===== RETURN
23  |_____ ENDIF
24      aux=TRIM(camino)
25      ACTIVATE WINDOW tabla
26      CLEAR
27      DIR &aux
28      WAIT 'Presione ENTER ...' WINDOW
29      DEACTIVATE WINDOW tabla
30      DEACTIVATE WINDOW mensajes
31      RETURN
32
34      *: EOF: UTIL002.ACT

```

```

1  *:*****
2  *:
3  *:      Program: C:\CIVIL\UTIL003.PRG
4  *:
5  *:      System: Anal.y Diseño de Estructuras Aporticadas
6  *:      Author: Rommel R. Alemán T. - Marco P. Silva S.
7  *:      Copyright (c) 1993, Pontificia Universidad Catolica del E.
8  *:      Last modified: 05/23/93      10:50
9  *:
10 *:      Called by: MENU005.PRG
11 *:
12 *:      Documented 05/28/93 at 12:15      FoxDoc version 2.10
13 *:*****
14  ACTIVATE WINDOW tabla
15  CLEAR
16  DIR *.*
17  WAIT 'Presione ENTER ...' WINDOW
18  ACTIVATE WINDOW mensajes
19  STORE SPACE(55) TO comando
20  @01,01 SAY "COPY:>" GET comando
21  READ
22  IF LASTKEY() = 27
23  DEACTIVATE WINDOW mensajes
24  DEACTIVATE WINDOW tabla
25  <=====RETURN
26  ENDIF
27  aux=TRIM(comando)
28  ! /40k COPY &aux
29  WAIT 'Presione ENTER ...' WINDOW
30  DEACTIVATE WINDOW mensajes
31  DEACTIVATE WINDOW tabla
32  RETURN
33
35  *: EOF: UTIL003.ACT

```

```

1  *:*****
2  *:
3  *:      Program: C:\CIVIL\UTIL004.PRG
4  *:
5  *:      System: Anal.y Diseño de Estructuras Aporticadas
6  *:      Author: Rommel R. Alemán T. - Marco P. Silva S.
7  *:      Copyright (c) 1993, Pontificia Universidad Catolica del E.
8  *:      Last modified: 05/23/93      11:09
9  *:
10 *:      Called by: MENU005.PRG
11 *:
12 *:      Documented 05/28/93 at 12:15      FoxDoc version 2.10
13 *:*****
14     ACTIVATE WINDOW tabla
15     CLEAR
16     DIR *.*
17     WAIT 'Presione ENTER ...' WINDOW
18     ACTIVATE WINDOW mensajes
19     STORE SPACE(55) TO comando
20     @01.01 SAY "BACKUP:>" GET comando
21     READ
22     IF LASTKEY() = 27
23     |     DEACTIVATE WINDOW mensajes
24     |     DEACTIVATE WINDOW tabla
25 <=====RETURN
26     |ENDIF
27     aux=TRIM(comando)
28     ! /128k backup &aux
29     WAIT 'Presione ENTER ...' WINDOW
30     DEACTIVATE WINDOW mensajes
31     DEACTIVATE WINDOW tabla
32     RETURN
33
35     *: EOF: UTIL004.ACT

```

```

1  *:*****
2  *:
3  *:      Program: C:\CIVIL\UTIL005.PRG
4  *:
5  *:      System: Anal.y Diseño de Estructuras Aporticadas
6  *:      Author: Rommel R. Alemán T. - Marco P. Silva S.
7  *:      Copyright (c) 1993, Pontificia Universidad Catolica del E.
8  *:      Last modified: 05/23/93      10:50
9  *:
10 *:      Called by: MENU005.PRG
11 *:
12 *:      Documented 05/28/93 at 12:15      FoxDoc version 2.10
13 *:*****
14  ACTIVATE WINDOW tabla
15  CLEAR
16  DIR *.*
17  WAIT 'Presione ENTER ...' WINDOW
18  ACTIVATE WINDOW mensajes
19  STORE SPACE(55) TO comando
20  @01,01 SAY "RESTORE:>" GET comando
21  READ
22  IF LASTKEY() = 27
23  |   DEACTIVATE WINDOW mensajes
24  |   DEACTIVATE WINDOW tabla
25  <====RETURN
26  |_ENDIF
27  aux=TRIM(comando)
28  ! /40k RESTORE &aux
29  WAIT 'Presione ENTER ...' WINDOW
30  DEACTIVATE WINDOW mensajes
31  DEACTIVATE WINDOW tabla
32  RETURN
33
35  *: EOF: UTIL005.ACT

```

```

1  *:*****
2  *:
3  *:      Program: C:\CIVIL\UTIL006.PRG
4  *:
5  *:      System: Anal.y Diseño de Estructuras Aperticadas
6  *:      Author: Rommel R. Alemán T. - Marco P. Silva S.
7  *:      Copyright (c) 1993, Pontificia Universidad Catolica del E.
8  *:      Last modified: 04/20/93      16:38
9  *:
10 *:      Called by: MENU005.PRG
11 *:
12 *:      Calls: DEFARCH.PRG
13 *:
14 *:      Uses: MAESTRO.DBF      Alias: PREDISE
15 *:            : TABLA01.DBF      Alias: CARVIVA
16 *:            : USUARIOS.DBF      Alias: USUARIOS
17 *:            : MAT01.DBF      Alias: MAT01
18 *:            : MAT02.DBF      Alias: MAT02
19 *:            : MAT03.DBF      Alias: MAT03
20 *:            : MAT04.DBF      Alias: MAT04
21 *:            : MAT05.DBF      Alias: MAT05
22 *:            : MAT06.DBF      Alias: MAT06
23 *:            : MAT07.DBF      Alias: MAT07
24 *:            : MAT08.DBF      Alias: MAT08
25 *:            : MAT09.DBF      Alias: MAT09
26 *:            : MAT10.DBF      Alias: MAT10
27 *:            : MAT11.DBF      Alias: MAT11
28 *:            : MAT12.DBF      Alias: MAT12
29 *:            : MAT13.DBF      Alias: MAT13
30 *:            : MAT14.DBF      Alias: MAT14
31 *:            : MAT15.DBF      Alias: MAT15
32 *:            : MAT16.DBF      Alias: MAT16
33 *:            : MAT17.DBF      Alias: MAT17
34 *:            : MAT18.DBF      Alias: MAT18
35 *:            : MAT19.DBF      Alias: MAT19
36 *:
37 *:      Indexes: PRECOD.IDX
38 *:            : PREPROY.IDX
39 *:            : TABCOD.IDX
40 *:            : USUCOD.IDX
41 *:            : USUIDE.IDX
42 *:            : Y01.IDX
43 *:            : Y02.IDX
44 *:            : Y03.IDX
45 *:            : Y04.IDX
46 *:            : Y05.IDX
47 *:            : Y06.IDX
48 *:            : Y07.IDX
49 *:            : Y08.IDX
50 *:            : Y09.IDX
51 *:            : Y10.IDX
52 *:            : Y11.IDX
53 *:            : Y12.IDX
54 *:            : Y13.IDX
55 *:            : Y14.IDX
56 *:            : Y15.IDX

```

```

57 *:          : Y16.IDX
58 *:          : Y17.IDX
59 *:          : Y18.IDX
60 *:          : Y19.IDX
61 *:
62 *: Documented 05/28/93 at 12:15 FoxDoc version 2.10
63 *:*****
64 CLOSE DATA
65 ACTIVATE WINDOW tabla
66 SET TALK ON
67 ?
68 ? -
69 ? - ARCHIVO DE PREDISEÑO Y DISEÑO
70 ? -
71 ?
72 USE maestro
73 PACK
74 INDEX ON codigo TO precod
75 INDEX ON proyecto TO preproy
76 ?
77 ? -
78 ? - TABLA DE CARGA VIVA
79 ? -
80 ?
81 USE tabla01
82 PACK
83 INDEX ON codigo TO tabcod
84 ?
85 ? -
86 ? - USUARIOS GENERALES
87 ? -
88 ?
89 USE usuarios
90 PACK
91 INDEX ON codigo TO usucod
92 INDEX ON identida TO usuide
93 ?
94 ? -
95 ? - MATRIZ DATOS 01
96 ? -
97 ?
98 USE mat01
99 PACK
100 INDEX ON y TO y01
101 ?
102 ? -
103 ? - MATRIZ DATOS 02
104 ? -
105 ?
106 USE mat02
107 PACK
108 INDEX ON y TO y02
109 ?
110 ? -
111 ? - MATRIZ DATOS 03
112 ? -

```

```
113 ?  
114 USE mat03  
115 PACK  
116 INDEX ON y TO y03  
117 ?  
118 ? '  
119 ? ' MATRIZ DATOS 04 '  
120 ? '  
121 ?  
122 USE mat04  
123 PACK  
124 INDEX ON y TO y04  
125 ?  
126 ? '  
127 ? ' MATRIZ DATOS 05 '  
128 ? '  
129 ?  
130 USE mat05  
131 PACK  
132 INDEX ON y TO y05  
133 ?  
134 ? '  
135 ? ' MATRIZ DATOS 06 '  
136 ? '  
137 ?  
138 USE mat06  
139 PACK  
140 INDEX ON y TO y06  
141 ?  
142 ? '  
143 ? ' MATRIZ DATOS 07 '  
144 ? '  
145 ?  
146 USE mat07  
147 PACK  
148 INDEX ON y TO y07  
149 ?  
150 ? '  
151 ? ' MATRIZ DATOS 08 '  
152 ? '  
153 ?  
154 USE mat08  
155 PACK  
156 INDEX ON y TO y08  
157 ?  
158 ? '  
159 ? ' MATRIZ DATOS 09 '  
160 ? '  
161 ?  
162 USE mat09  
163 PACK  
164 INDEX ON y TO y09  
165 ?  
166 ? '  
167 ? ' MATRIZ DATOS 10 '  
168 ? '
```

```
169      ?
170      USE mat10
171      PACK
172      INDEX ON y TO y10
173      ?
174      ? '
175      ? ' MATRIZ DATOS 11 '
176      ? '
177      ?
178      USE mat11
179      PACK
180      INDEX ON y TO y11
181      ?
182      ? '
183      ? ' MATRIZ DATOS 12 '
184      ? '
185      ?
186      USE mat12
187      PACK
188      INDEX ON y TO y12
189      ?
190      ? '
191      ? ' MATRIZ DATOS 13 '
192      ? '
193      ?
194      USE mat13
195      PACK
196      INDEX ON y TO y13
197      ?
198      ? '
199      ? ' MATRIZ DATOS 14 '
200      ? '
201      ?
202      USE mat14
203      PACK
204      INDEX ON y TO y14
205      ?
206      ? '
207      ? ' MATRIZ DATOS 15 '
208      ? '
209      ?
210      USE mat15
211      PACK
212      INDEX ON y TO y15
213      ?
214      ? '
215      ? ' MATRIZ DATOS 16 '
216      ? '
217      ?
218      USE mat16
219      PACK
220      INDEX ON y TO y16
221      ?
222      ? '
223      ? ' MATRIZ DATOS 17 '
224      ? '

```

225 ?
226 USE mat17
227 PACK
228 INDEX ON y TO y17
229 ?
230 ? '
231 ? ' MATRIZ DATOS 18 '
232 ? '
233 ?
234 USE mat18
235 PACK
236 INDEX ON y TO y18
237 ?
238 ? '
239 ? ' MATRIZ DATOS 19 '
240 ? '
241 ?
242 USE mat19
243 PACK
244 INDEX ON y TO y19
245 DO defarch
246 SET TALK OFF
247 DEACTIVATE WINDOW tabla
248 RETURN
249
251 *: KOF: UTIL006.ACT

CAPITULO VIII

8. RECOMENDACIONES Y SUGERENCIAS A LA TERMINACION DEL PROYECTO

INDICE

RECOMENDACIONES Y SUGERENCIAS A LA TERMINACION DEL PROYECTO	1
DESCRIPCION DEL PROYECTO	1
ESPECTATIVAS INICIALES	1
ESTADO ACTUAL DEL PROYECTO	1
AREAS REMANENTES DE INTERES	1
LECCIONES TECNICAS APRENDIDAS	1
LECCIONES ADMINISTRATIVAS APRENDIDAS	2
RECOMENDACIONES PARA FUTUROS PROYECTOS	2

8. RECOMENDACIONES Y SUGERENCIAS A LA TERMINACIÓN DE PROYECTO.

8.1. DESCRIPCIÓN DEL PROYECTO.

El presente trabajo a sido el resultado de la investigación de métodos y tecnologías de la Ingeniería Civil, en la especialización de estructuras, para brindar a los profesionales y estudiantes de esta rama una herramienta para el análisis y diseño de estructuras de hormigón armado utilizando algoritmos matemáticos asistidos por un computador.

Durante el desarrollo del proyecto se ha ido intercambiando conocimientos de lo que es el análisis y desarrollo de sistemas de información con técnicas, métodos y algoritmos matemáticos, entre el usuario final y el equipo de desarrollo de sistemas.

8.2. EXPECTATIVAS INICIALES

Al inicio del proyecto se propuso el desarrollo de un sistema de software para que sirviera como herramienta de aprendizaje de las técnicas de análisis y diseño de estructuras de hormigón armado de tipo residencial.

El sistema es limitado pero no quiere decir esto que no pueda ser ampliado o generalizado a otro tipo de cálculos.

8.3. ESTADO ACTUAL DEL PROYECTO.

El proyecto a su terminación cumplió con las expectativas iniciales planteadas a nosotros como una inquietud de un conjunto de profesionales de la Ingeniería Civil.

8.4. AREAS REMANENTES DE INTERES.

Como áreas de interés, tenemos la mejora de los algoritmos para generalizar el método de cálculo utilizado en el presente trabajo.

8.5. LECCIONES TECNICAS APRENDIDAS.

- Mejora en la administración de lenguajes no convencionales a la aplicación de problemas matemáticos.
- Aprendizaje y perfeccionamiento de técnicas de programación estructurada.
- Optimización del uso de recursos de memoria del equipo.
- Cambio en las técnicas de uso de matrices de memoria por pseudomatrices basadas en archivos de disco.

8.6. LECCIONES ADMINISTRATIVAS APRENDIDAS.

En el presente trabajo, la aplicación de técnicas de administración de proyectos, no fue un punto de interés, ya que todo el proyecto se lo llevó a cabo basándose en la amistad y experiencia de cada uno de los elementos integrantes del equipo.

8.7. RECOMENDACIONES PARA FUTUROS PROYECTOS.

- Mejor planificación de los tiempos de cada una de las etapas del desarrollo de un proyecto informático.
- Para que un proyecto pueda ser cabalmente terminado, es recomendable no forzar los plazos de entrega de estos.
- Mejorar la interrelación de estudiantes y cuerpo directivo de trabajo de Tesis, asignando tiempos de análisis y correcciones de los correspondientes borradores.

F I N