

2  
201

# UNIVERSIDAD NACIONAL AUTONOMA DE MEXICO

ESCUELA NACIONAL DE ESTUDIOS PROFESIONALES  
"A R A G O N"



SISTEMA DE GRAFICACION BASADO EN  
EL ENLACE DE DBASE III PLUS Y  
LOTUS 1-2-3

T E S I S  
QUE PARA OBTENER EL TITULO DE  
INGENIERO EN COMPUTACION  
P R E S E N T A :  
IMELDA DE LA LUZ FLORES DIAZ

TESIS CON  
FALLA DE ORIGEN



## **UNAM – Dirección General de Bibliotecas Tesis Digitales Restricciones de uso**

### **DERECHOS RESERVADOS © PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL**

Todo el material contenido en esta tesis está protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

# INDICE

# Pag.

## INTRODUCCION

1

## CAPITULO I.- BASES TEORICAS

### 1.1 ANTECEDENTES PARA LA CREACION DEL SISTEMA

1.1.1 El Sistema Integral de Información 3

1.1.2 El Comité Técnico de Información 6

1.1.3 La Dirección de Sistemas de Información 6

1.1.4 El Módulo de Control de Formatos 7

### 1.2 HARDWARE QUE SE REQUIERE 11

### 1.3 TIPOS DE BASES DE DATOS 11

1.3.1 Relacional 11

1.3.2 Jerárquica 12

1.3.3 De Red 12

### 1.4 DBASE III PLUS 12

### 1.5 QUATTRO 21

## CAPITULO II.- METODOLOGIA

### II.1 ANTECEDENTES 53

### II.2 FASES EN EL DESARROLLO DE LOS PROGRAMAS 54

II.2.1 Algoritmo 54

II.2.2 Diagramas 54

II.2.3 Codificación 54

II.2.4 Prueba 55

II.2.5 Mantenimiento 55

### II.3 CALIDAD DE LOS PROGRAMAS 55

II.3.1 Que el programa funcione 56

II.3.2 Que el programa esté libre de errores 56

II.3.3 Que el programa esté bien documentado 56

### II.4 PROGRAMACION MODULAR O DISEÑO MODULAR DESCENDENTE CO TOP-DOWN DE SISTEMAS 56

### II.5 PROGRAMACION ESTRUCTURADA 57

II.5.1 Métodos de programación estructurada 58

## CAPITULO III.- ARCHIVOS Y DIAGRAMAS

### III.1 OBJETIVOS Y ALCANCES DEL SISTEMA 60

### III.2. ARCHIVOS 61

III.2.1 Archivos contenidos en el sistema 61

	# Pág.
111.2.2 Instalación del sistema	
111.2.3 Sisgraf.bat	64
111.2.4 Sistgraf.prg	65
111.2.5 Prin1.prg	66
111.2.6 Prin2.prg	67
111.2.7 Prin3.prg	70
111.2.8 Tecla.prg	72
111.2.9 Sub1.prg	74
111.2.10Macro.1kq	75
	80
<b>111.3. DIAGRAMAS</b>	<b>81</b>
111.3.1 Diagrama de la organización del disco que contiene el Sistema	82
111.3.2 Diagrama de la organización de los archivos dentro del disco duro	84
111.3.3 Diagrama general del Sistema	86
111.3.4 Instala.bas	88
111.3.5 Diagrama general del sisgraf.bat	91
111.3.6 Sisgraf.bat	93
111.3.7 Sistgraf.prg	95
111.3.8 Prin1.prg	97
111.3.9 Prin2.prg	100
111.3.10Prin3.prg	103
111.3.11Tecla.prg	105
111.3.12Sub1.prg	107
111.3.13Macro.1kq	119
 <b>CAPITULO IV. LISTADOS Y PANTALLAS</b>	
<b>IV.1 LISTADOS</b>	
IV.1.1 Instala.bas y sisgraf.bat	122
IV.1.2 Sistgraf.prg	123
IV.1.3 Caenti.dbf	126
IV.1.4 Caform.dbf	129
IV.1.5 Casec.dbf	131
IV.1.6 Casubs.dbf	133
IV.1.7 Contol1.dbf	135
IV.1.8 Variable.dbf	137
IV.1.9 Ejemplo de Tempo.txt	146
IV.1.10Ejemplo de Tempo2.txt	148
IV.1.11Ejemplo de Tempo.cat	150
IV.1.12Prin1.prg	152
IV.1.13Prin2.prg	154
IV.1.14Prin3.prg	156
IV.1.15Tecla.prg	158
IV.1.16Sub1.prg	160
IV.1.17Macro.1kq	162
	169
<b>IV.2 PANTALLAS</b>	
IV.2.1 Seriación de pantallas	171
IV.2.2 Ejemplos de los tipos de gráficas	172
	183

	# Pág.
<u>CONCLUSION</u>	187
<u>BIBLIOGRAFIA</u>	188
<u>ANEXO 1</u>	190

## INTRODUCCION

En la Secretaría de la Contraloría General de la Federación (SECOGEF) es necesario llevar un estricto control de los ingresos y egresos del gasto público que realiza cada una de las entidades paraestatales existentes, y para tal efecto se vale de un conjunto de formatos que nos ayuda a recopilar la información que se requiere, dichos formatos están englobados en apartados, y a su vez estos apartados se encuentran contenidos según su índole en los siguientes documentos:

- Carpeta de Entidades Paraestatales.
- Cuaderno de Información del Gobierno Federal.
- Manual de Información de Intermediarios Financieros.

La SECOGEF cuenta con la Dirección de Sistemas de Información, la cual es la encargada de crear dentro de su centro de cómputo los programas necesarios para que se capture y procese la información que mandan las entidades por medio de los formatos.

Por todo lo antes expuesto se creó un sistema computarizado que como ya se dijo se denominó "Módulo de Control de Formatos del Sistema Integral de Información", el cual muy a menudo tan solo es llamado "Sistema Integral de Información" por la gente que se encuentra familiarizada con él. Dicho sistema fue realizado en Dbase III Plus y Clipper, permitiendo éste el registro de la información por medio de los formatos además de ayudar a que se perciba y registre la puntualidad y el cumplimiento que es necesario analizar y cuyos resultados sirven de base para que la o las persona(s) indicadas lleven a cabo la toma de decisiones. Pero dicho análisis se dificulta debido a que la forma en que la información se recibe de la computadora no es la idónea (se reciben listados de cifras y más cifras); es por ello que me nació la idea de que si esta información estuviera en forma de gráficas se obtendrían mejores y más rápidos resultados, ya que así se podría visualizar más fácilmente que entidades, sectores o subsectores están cumpliendo, quienes no y en que porcentaje lo han hecho; ahorrando de esta manera una considerable cantidad de tiempo (horas-hombre) y esfuerzo por parte del personal destinado a su análisis, verificación e interpretación.

Tras algunas pesquisas en cuanto a cuál sería el lenguaje o paquete más apropiado para el desarrollo del Sistema deseado obtuve las siguientes ventajas propias del Dbase III Plus y que lo hacían el más apropiado para los fines que se perseguían:

- Es rápido.
- Requiere poca memoria.
- No tiene restricciones en cuanto al tipo de máquina.
- Requiere al DOS como sistema operativo.
- Su portabilidad es grande.
- Esta diseñado para equipo pequeño.
- No es caro en cuanto a costo de adquisición y mantenimiento.
- Es muy popular y por lo tanto mucha gente lo sabe manejar.
- Es de fácil manejo.

por todo esto llegué a la conclusión de que dicho sistema de graficación debería estar diseñado tentativamente en Dbase III Plus para la recopilación y selección de la información, así como la especificación de archivos; y en Lotus 123 para la graficación propiamente dicha. Pero comenzando a bosquejar lo que sería la tesis me encontré con que resultaba ser bastante complicado lograr

que se pudieran imprimir las gráficas mediante el Lotus, ya que para lograrlo, dicho paquete se auxilia del Pgraph y no resulta sencillo hacer que automáticamente se imprima la gráfica (sin que el usuario tenga que realizar alguna elección), ya que en este no existen las macros; por todo esto me di a la tarea de buscar otro paquete que resultara ser compatible con Dbase, encontrando finalmente el Quattro, el cual como más adelante se apreciará es ideal para todos los fines que se perseguían.

Finalmente lo que me ayudó a acabarme de decidir por estos dos paquetes fue lo siguiente:

- 1.- La gente que ahí labora conoce y maneja de manera considerablemente buena los 2 paquetes, de tal forma que la comprensión y el mantenimiento del Sistema que se crearía no sería problema para ellos una vez que se hubiera implantado en la SECOGEF.
- 2.- Los archivos en los cuales se estaría basando para la creación del sistema (los que contienen la información y los datos) son creados con Dbase III Plus (ya sean bases de datos o programas en sí).
- 3.- La programación en Dbase es muy accesible, al igual que las macros y la graficación en Quattro.
- 4.- Puede existir enlace entre ambos paquetes.

## CAPITULO I BASES TEORICAS

*Sistema es un conjunto de elementos interrelacionados entre sí.*

### I.1 ANTECEDENTES PARA LA CREACION DEL SISTEMA

#### I.1.1 EL SISTEMA INTEGRAL DE INFORMACION (SII).

##### QUE ES EL SII

*Es un conjunto de información ordenada que reporta los resultados de las actividades que el sector público realiza, siendo este un sistema de Planeación Democrática del Desarrollo Nacional. Habiendo sido presentado éste a fines de 1985 en el seno de la Comisión Intersecretorial del Gasto Financiero.*

##### CUAL ES EL OBJETIVO DEL SII

*Es imprimir solidez, dinamismo, permanencia y equidad al crecimiento de la economía basándose en la adecuada formulación, instrumentación, control y evaluación de cada uno de los programas y del esquema de planeación en su conjunto, lo cual se apoya en que la información sea oportuna y confiable, además de estar definida sobre bases conceptuales adecuadas, de tal forma que se apoye la planeación y la toma de decisiones.*

*Estando todo esto coordinado por el Comité Técnico de Información, del cual se hablará más adelante.*

##### CUAL ES EL CONTENIDO DEL SII

*Es un conjunto de formatos que nos ayudan a recopilar la información que se requiere, dichos formatos están englobados en apartados, y a su vez estos apartados se encuentran contenidos según su índole en los siguientes documentos:*

- Carpeta de Entidades Paraestatales.
- Cuaderno de Información del Gobierno Federal.
- Manual de Información de Intermediarios Financieros.

##### CUAL ES EL OBJETIVO Y EL CONTENIDO DE LA CARPETA DE INFORMACION DE ENTIDADES PARAESTATALES.

*El objetivo de esta carpeta es captar la información respecto a:*

- 1) Situación financiera (comprendiendo los formatos del E-1 al E-18).
- 2) Ocupación (contenida en los formatos del E-19 al E-21).



- 3) Producción y precios (con formatos del E-22 al E-27).
- 4) Deuda pública (abarcando los formatos del E-28 al E-31).
- 5) Presupuesto por programas (comprendiendo los formatos del E-32 al E-34).
- 6) Inversión física (con formatos del E-35 al E-44).
- 7) Apreciaciones y expectativas (contenido en los formatos del E-45 al E-47).
- 8) Registro Único de transferencias (abarcando los formatos del E-48 al E-53).

Todo esto basado como se puede ver en 53 formatos (del E-1 al E-53) que se entregan con periodicidades diferentes (mensual, anual y trimestral); dichos formatos están contenidos en los 8 apartados que se mencionaron.

De dicha carpeta cada entidad llena alrededor de 28 formatos, según le corresponda.

CUAL ES EL OBJETIVO Y EL CONTENIDO DEL CUADERNO DE INFORMACIÓN DEL GOBIERNO FEDERAL

Su objetivo es obtener información sobre los siguientes apartados, basándose en 25 formatos (del D-1 al D-23):

- 1) Presupuesto ejercido (abarcando los formatos del D-1 al D-7.4).
- 2) Ocupación (contenida en los formatos del D-8 al D-10).
- 3) Presupuesto y programas (con formatos que van del D-11 al D-12).
- 4) Inversiones físicas (comprendiendo los formatos del D-13 al D-17).
- 5) Recaudación de ingresos (abarcando los formatos del D-18.1 al D-18.3).
- 6) Informes del gobierno (con formatos que van desde el D-19 al D-23).
- 7) Información de adquisiciones y obras públicas (contenidas en los formatos del D-24 al D-25).

Este cuaderno deberá ser llenado por las Secretarías de Estado.

- iii) Un ejemplo de estos formatos se encuentra en el anexo de la presente Tesis; ahora bien, en caso de que sea necesario consultar los demás favor de remitirse a cualquiera de los documentos -Carpeta de Entidades parasetales, Cuaderno de Información del Gobierno Federal o Manual de Información de Intermediarios Financieros-, los cuales se encuentran disponibles en la Dirección de Sistemas de Información perteneciente a la Secretaría de la Contraloría General de la Federación ubicada en Insurgentes Sur 1785 Col. Cuadalupe Inn.

## CUAL ES EL OBJETIVO Y EL CONTENIDO DEL MANUAL DE INFORMACION DE INTERMEDIARIOS FINANCIEROS

Este manual al que también se le conoce como de Intermediación Financiera, es el único instrumento de captación de información de los bancos, fondos, fideicomisos y otras entidades

coordinadas por la Secretaría de Hacienda y Crédito Público (SHCP) por parte de la Secretaría de Programación y Presupuesto (SPP), la SECOGEF, el Banco de México (BM) y la Secretarías Coordinadoras de Sector; todo esto con el propósito de evitar dispersiones, duplicaciones o peticiones adicionales no justificadas.

Este se dividió en 8 apartados que cubren un total de 45 formatos, con periodicidades diferentes: mensual, trimestral o anual. He aquí los apartados:

- 1) Información programática (contenida en el primer formato).
- 2) Situación financiera (con formatos del 2 al 13).
- 3) Crédito (abarcando los formatos del 14 al 16).
- 4) Empleo y remuneraciones (comprendiendo del 17 al 21 formato).
- 5) Inversiones (con formatos que van del 22 al 29).
- 6) Informe del gobierno (contenido en los formatos del 30 al 34).
- 7) Transferencias (abarcando del formato 35 al 39).
- 8) Deuda (comprendiendo del 40 al 45 formato).

## COMO SE INSTITUCIONALIZO EL SII

Las labores se han concentrado en la expedición de un acuerdo presidencial que hace obligatorio de manera permanente el SII, así como la creación de apoyos para que la información fluya de las entidades a las globalizadoras (SPP, SHCP y SECOGEF) y al Banco de México a través de diskettes de microcomputadora.

## QUIEN COORDINA EL SII Y CUALES FUERON LAS RAZONES PARA CREAR ESTE ORGANISMO.

Se tiene que el Comité Técnico de Información es el encargado de coordinar el SII, habiéndose formado éste para resolver la siguiente problemática: La existencia de múltiples sistemas ocasionó que las solicitudes de información de las Secretarías coordinadoras de sector y sobre todo las de las propias entidades paraestatales fueran repetidas, aunque en formatos distintos, realizándose así esfuerzos duplicatorios innecesarios tanto en la captación de la información como en su procesamiento. La diversidad de los sistemas también complicó la generación oportuna, suficiente y veraz de la información por parte de las entidades paraestatales y propició que éstas destinaran más recursos materiales y humanos significativos para satisfacerlos.

### I.1.2.- EL CONITE TECNICO DE INFORMACION (CTI)

#### QUEL ES EL OBJETIVO DEL CTI

Es coordinar el SII de los Ingresos y Gastos Públicos para hacer compatibles los requerimientos de información que deben rendir las dependencias e intermediarios financieros, así como las entidades paraestatales conforme a su legislación específica y vigilar la operación del Sistema mediante el seguimiento del mismo, y constituir un foro intersecretarial permanente, para el intercambio de opiniones en materia técnica y logística de la información.

#### QUIENES INTEGRAN EL CTI:

El CTI está integrado por representantes de la SPP, la SHCP, la SECOGEF y el Banco de México. Así mismo, se prevé la participación de las Secretarías coordinadoras de sector.

#### COMO ESTA ORGANIZADO EL CTI

Existen tres grupos de trabajo: uno para tratar los aspectos relacionados con la información correspondiente al Gobierno Federal, otro para analizar los aspectos relativos a Organizaciones y Empresas, y un tercero para discutir las cuestiones relacionadas con la información sobre Intermediación Financiera. Así, en el comité se cubre la información relativa a todas las dependencias y entidades del sector público.

#### CUAL FUE LA APORTACION DEL CTI AL SII

Como resultado de diversas sesiones de los 3 grupos de trabajo del CTI se obtuvieron los siguientes documentos que conforman lo que es actualmente el SII:

- Carpeta de Información de Entidades Paraestatales.
- Cuaderno de Información del Gobierno Federal.
- Manual de Información de Intermediarios Financieros.

A la par es elaborado un instructivo que sirva como guía para el llenado de los 3 documentos arriba mencionados, conteniendo también una relación del universo de dependencias que llenaría cada uno de los formatos.

### I.1.3.- LA DIRECCION DE SISTEMAS DE INFORMACION

#### PAPEL QUE DESEMPEÑA LA DIRECCION DE SISTEMAS DE INFORMACION DENTRO DEL SII.

Esta dirección es la encargada de crear dentro de su centro

de cómputo los programas necesarios para que se capture y procese la información que mandan las entidades por medio de los formatos, viéndose esto reflejado con la creación del Módulo de Control de Formatos del SII.

#### QUE ACTIVIDADES SE REALIZAN EN DICHA OPERACION

- 1) Los formatos recibidos se registran en el SII.
- 2) Se emiten reportes comparativos y de validación, así como también oficios de aviso de recepción, a las áreas usuarias de la información contenida en los formatos.
- 3) Los reportes comparativos y la validación se envían a los usuarios tanto internos como externos para su evaluación.
- 4) Se envían oficios de aviso de recepción de la información a la Subsecretaría "B" de la SECOGEF, especialmente a las direcciones de Auditorías a Obras Públicas de Normatividad e Investigación, General de Auditorías Externas y General de Operación Regional.
- 5) Se envían junto con los oficios de aviso, los formatos que contienen la información a las direcciones de Área "A" y "B" para su análisis y resguardo.

#### 1.1.4. - MÓDULO DE CONTROL DE FORMATOS DEL SII

##### CUAL ES EL OBJETIVO GENERAL DEL MÓDULO DE CONTROL DE FORMATOS DEL SII.

Es contar con un sistema o módulo computalizado para controlar y evaluar el flujo de formatos inherentes al SII, permitiendo la emisión de reportes que cuantifiquen y comparen la información solicitada con la recibida por las 3 dependencias de apoyo global o el Banco de México a nivel individual o global.

##### CUALES SON LOS OBJETIVOS BASICOS DEL MÓDULO DE CONTROL DE FORMATOS DEL SII

- 1.- Facilitar la remisión de la información de las "plantillas únicas" de cada dependencia a las áreas usuarias a través de la generación automática de oficios de envío.
- 2.- Proporcionar información por entidad, sector o subsector del grado de cumplimiento en la entrega de los formatos a las dependencias de apoyo global y al Banco de México, permitiendo incluso una evaluación que abarque a todas estas instancias.
- 3.- Ofrecer elementos para juzgar la calidad de la información remitida a partir de la agrupación de observaciones sobre la misma y el número de veces que se sustituyan los formatos.

##### COMO ES LA ESTRUCTURA DEL MÓDULO

Antes que nada es conveniente saber que este sistema o módulo está escrito completamente en el lenguaje del manejador de base de

datos "Dase III Plus" y Clipper.

Ahora bien, la estructura del Sistema de Registro y Control de Formatos comprende un menú principal, el cual sirve como pivote para que a partir de este, sea posible acceder cualquiera de los submódulos disponibles, que son los siguientes:

- a) Registros de formatos.
- b) Universo de formatos por dependencia/entidad.
- c) Universo de formatos por responsable.
- d) Comentarios a las dependencias.
- e) Catálogo de formatos.
- f) Catálogo de entidades.
- g) Catálogo de sectores.
- h) Catálogo de subsectores.
- i) Catálogo de responsables.
- j) Catálogo de tablas.
- k) Creación de índices.
- l) Respaldo de información.
- m) Recuperación de información.

QUE FUNCION SE LLEVA A CABO EN  
EL SUBMÓDULO DE REGISTRO DE FORMATOS

Este submódulo presenta los procedimientos para realizar y verificar el correcto registro de los formatos que remiten a las dependencias de apoyo global y al Banco de México, las dependencias, entidades paraestatales e intermediarios financieros que integran el Sistema Integral de Información del Sector Público. Así como los formatos de las finanzas locales.

QUE HACE EL SUBMÓDULO UNIVERSO  
DE FORMATOS POR DEPENDENCIA/ENTIDAD

Debido a la diversidad de formatos que puede manejar cada dependencia/entidad, relacionados a las características inherentes a su funcionamiento, se hace necesario manejar un universo de formatos específicos que considere los requerimientos de información de las globalizadoras y del Banco de México, a nivel dependencia/entidad.

Este submódulo permite llevar a cabo el control y la verificación de la información que deberán enviar las dependencias y/o entidades a las globalizadoras y al Banco de México.

QUE SE SABE DEL SUBMÓDULO  
UNIVERSO DE FORMATOS POR RESPONSABLE

Este submódulo permite mantener el control y la verificación del conjunto de formatos enviados por cada dependencia/entidad a las dependencias de apoyo global y al Banco de México, con el responsable final de dicha información, el cual recibirá los formatos en función del periodo de tiempo previamente establecido en los instrumentos correspondientes al Gobierno Federal, a la Intermediación Financiera y a las de Empresas Paraestatales.

Este submódulo, es la base para que la ventanilla receptora de los formatos los remita a las áreas usuarias de la propia dependencia o haga de su conocimiento la información que se ha

recibido a partir de los oficios de aviso o remisión.

Es necesario aclarar que solo deberá capturarse información para los responsables que tengan "1" en el campo de los indicadores del submódulo del Catálogo de Entidades.

QUE SE PUEDE LEER EN CUANTO AL  
SUBMÓDULO COMENTARIOS A LA DEPENDENCIA/ENTIDAD

Este submódulo permite mantener un seguimiento de los comentarios que realizan las dependencias de apoyo global o el Banco de México a las dependencias, entidades paraestatales o intermediarios financieros de acuerdo a la entrega de la información solicitada, tanto en cantidad, calidad y oportunidad, a la vez, se especifica, si los informantes están respetando las periodicidades establecidas para cada uno de los formatos y la estructura de los mismos.

QUE HAY CON RESPECTO AL  
SUBMÓDULO CATALOGO DE EMPRESAS

Este permite tener el control de todos y cada uno de los formatos que deberán de remitir las dependencias, empresas paraestatales o intermediarios financieros a las dependencias de apoyo global y al Banco de México, así como la clave con que son identificados.

En este submódulo se tiene la opción de consultar o modificar lo que en él se encuentra contenido.

QUE SE PUEDE LEER EN CUANTO  
AL SUBMÓDULO CATALOGO DE ENTIDADES

Permite mantener el control de las claves y de las entidades correspondientes, no importando en el área en la cual operan (sector o subsector).

QUE CONTIENE EL  
SUBMÓDULO CATALOGO DE SECTORES

En este submódulo se encuentran contenidas todos los sectores existentes, así como sus respectivas claves, para que en caso necesario se pueda hacer uso de esta información sin problema alguno; permitiendo también dicho submódulo tanto su consulta como su actualización.

QUE ESTA CONFORMADO EL  
SUBMÓDULO CATALOGO DE SUBSECTORES

En el presente submódulo se encuentran contenidos tanto los nombres como las claves de cada uno de los subsectores, para que cuando sea necesario se tenga acceso de manera rápida y eficaz a dicha información, ya sea para su consulta o su modificación.

QUE PERMITE HACER EL  
SUBMÓDULO CATALOGO DE RESPONSABLES

Este submódulo permite llevar el manejo y control del catálogo de responsables de una manera adecuada y eficaz.

El catálogo permite conocer a quién se le notificará por medio de un oficio de la llegada de los formatos enviados por las dependencias/entidades a las "ventanillas únicas" de las dependencias de apoyo global y al Banco de México.

Así mismo, el catálogo permite clasificar a los responsables por áreas de especialización.

QUE CONTIENE EL  
SUBMÓDULO CATALOGO DE TABLAS

Este submódulo permite llevar el control y la actualización de las tablas de descripciones que utiliza el sistema de control y registro de formatos.

Este catálogo está compuesto por 11 tablas, que son las siguientes:

TABLA	TIPO DE TABLA
01	TIPO DE ENTIDAD
02	SITUACION OPERATIVA
03	SECTOR
04	OBSERVACIONES
05	CLASIFICACION PRESUPUESTAL
06	PERIODICIDAD
07	PERIODOS
08	FUENTE DE INFORMACION
09	AREA ELABORADORA
10	TIPO DE CUADERNO
11	TEMAS

Las tablas nos permiten conocer la descripción de las claves que se utilizan en todo el sistema.

QUE SE REALIZA EN EL  
SUBMÓDULO CREACION DE INDICES

Se realiza un movimiento técnico que consiste en la creación de índices en los archivos de las bases de datos.

Esta creación de índices permite el acceso directo a los registros que se deseen obtener de las bases de datos.

Se recomienda a los usuarios generar esta opción inmediatamente después de bajar información a la unidad de disco duro.

QUE PODEMOS DECIR DEL  
SUBMÓDULO RESPALDO DE INFORMACION

Este submódulo permite tener una copia de todo el sistema de registro y control de formatos en discos flexibles (diskettes).

La ventaja de contar con este submódulo es que previene la

perdida de información debido a causas fuera del control del usuario, como por ejemplo daños en la unidad de disco duro que ameriten reformatar este último y con ello ocasionar la pérdida de la información.

Es recomendable generar este procedimiento sistemáticamente, de acuerdo con la frecuencia de uso del sistema.

## 1.2 HARDWARE QUE SE REQUIERE

Dadas las facilidades que presentan actualmente las computadoras personales, se optó por desarrollar el "Sistema de Graficación Basado en el Enlace del Dbase III Plus y el Lotus 123" en máquinas de este tipo, por razones de economía y portabilidad, aunado al hecho de que el Sistema de Registro y Control de Formatos del SII está desarrollado en este mismo tipo de máquina.

Los requerimientos mínimos de equipo de cómputo necesarios para la implementación del presente Sistema son:

- Computadora personal XT o AT compatible con IBM.
- Sistema operativo D.O.S. (Disk Operating System) versión 3.1 o más reciente.
- 512 KB de memoria RAM (Random Access Memory).
- Disco duro con 3 MB libres en el momento de utilizar ambos sistemas (el de Registro y Control de Formatos del SII, y el de Graficación Basado en el Enlace del Dbase III Plus y el Lotus 123).
- Impresora con capacidad de impresión de 132 columnas y 215 columnas con letra compactada.

## 1.3 TIPOS DE BASES DE DATOS

Se llama base de datos a una colección de datos interrelacionados y almacenados en conjunto sin redundancias perjudiciales o innecesarias; su finalidad es la de servir a una aplicación o más, de la mejor manera posible; los datos se almacenan de modo que resulten independientes de los programas que los usan; se emplean métodos bien determinados para incluir datos nuevos y para modificar o extraer los datos almacenados.

### 1.3.1 RELACIONAL

En este tipo de base de datos se manejan datos de más de un archivo a la vez y estos archivos se encuentran en tablas, donde cada tabla se asemeja a un archivo secuencial. Cada renglón corresponderá a un registro del archivo y cada columna a un campo. Aquí se emplea el concepto de dominio, que es un conjunto de valores de una columna específica. Así pues las asociaciones entre los renglones se representan únicamente por valores de datos en columnas sacadas de un dominio en común. Se llama relacional porque existe una relación entre tablas por medio de algún campo llave para establecer esas relaciones.



### I.3.2 JERARQUICA

Aquí las unidades de información se estructuran en varios niveles que se parecen gráficamente a un árbol invertido, con la raíz en la parte superior y las ramas hacia abajo. En esta estructura hay una relación superior-subordinado. Debajo del componente de información que constituye la raíz única hay elementos subordinados o nodos, cada uno de los cuales, a su vez, "posee" uno o más elementos adicionales (o ninguno). Cada elemento o rama de esta estructura debajo de la raíz tiene un solo "dueño". Las ramas de una estructura de árbol no están conectadas entre sí.

### I.3.3 DE RED

Este tipo de base de datos es muy similar a la jerárquica, con la única diferencia de que aquí sí se permite conectar los nodos en forma multidireccional. Así, cada nodo puede tener varios "dueños" y puede, a su vez, poseer cualquier número de unidades de información diferentes. Debido a que todas estas relaciones se dan por medio de apuntes por lo mismo aquí se ocupa mucho espacio en memoria. Ahora bien, los programas para el manejo de los datos permiten extraer la información requerida de una estructura como ésta a partir de cualquier registro del archivo.

## I.4 DBASE III PLUS

Es un sistema manejador de base de datos en forma racional.

### CARACTERÍSTICAS

- Facilidad de manejo para uso interactivo o desarrollo de programación.
- Comandos con lenguaje fácil de manejar, y ayuda con pantallas en línea.
- 128 campos o 4000 bytes por registro y más de un millón de registros-archivos.
- 10 archivos de datos en uso al mismo tiempo.
- Ordena archivos por varios campos y busca registros de forma rápida.
- Corre en cualquier PC y compatibles.

### CONVENCIONES

- Nombres de archivos hasta de 8 caracteres de longitud, comenzando con una letra, pudiendo contener letras, números y guión de subrayado exclusivamente.
- Nombres de campos y variables, hasta 10 caracteres de longitud (con las mismas reglas anteriores)
- [ ] indican partes opcionales de una orden o comando.
- < > indican partes que deben ser proporcionadas por el usuario a una orden o comando, para su ejecución.

TIPOS DE DATOS O VARIABLES

- C. - Carácter alfanumérico. - Cualquier letra, número o signo de hasta 254 posiciones de longitud.  
N. - Numérico. - Incluye números, signo y punto decimal.  
D. - Fecha. - en forma MM/DD/AA.  
L. - Lógico. - Puede ser .T., .F., .F. o .f.  
M. - Memo. - Para textos de gran longitud.

OPERADORES MATEMÁTICOS

- |   |           |        |                 |    |                 |
|---|-----------|--------|-----------------|----|-----------------|
| + | =Adición  | -      | =Resta          | *  | =Multiplicación |
| / | =División | ** ó ^ | =Exponenciación | () | =Agrupación     |

Los operadores matemáticos por lo general entregan resultados numéricos.

OPERADORES ALFANUMÉRICOS

- + =Operador de concatenación. - Es usado para unir dos o más variables alfanuméricas en una sola variable alfanumérica.
- =Operador de concatenación. - Es igual que el anterior, pero los espacios en blanco que aparecen a la derecha de cada variable son colocados al final de la variable resultante.

OPERADORES DE RELACION

- |    |                            |        |                 |
|----|----------------------------|--------|-----------------|
| <  | .-Menor que                | >      | .-Mayor que     |
| =  | .-Igual a                  | <> ó # | .-Diferente de  |
| <= | .-Menor o igual            | >=     | .-Mayor o igual |
| I  | .-Comparación alfanumérica |        |                 |

Los operadores de relación dan resultados lógicos, y deben ser utilizados como variables del mismo tipo.

OPERADORES LÓGICOS

- .AND. -AND lógico (&).
- .OR. -OR lógico (O).
- .NOT. -NOT lógico (- ó NO).

Los operadores lógicos obtienen resultados lógicos basados en la comparación de 2 expresiones lógicas.

SINTAXIS GENERAL DE COMANDOS

- VERBO <rango> <expresiones> <condición>                donde:
- VERBO        .- Comando de Dbase III Plus.
  - rango        .- ALL, NEXT, x, REST, RECORD n.
  - expresión   .- exp<C>, exp<N>, exp<L>, exp<D>, ó variables, campos, etc., y la combinación de ellos.

condición .- FOR/WHILE

Ejemplo:

<u>DOSPLAY</u>	<u>NEXT 15</u>	<u>Nom.Dir</u>	<u>FOR Nom='Juan'</u>
<u>VERBO</u>	<u>rango</u>	<u>lista de</u> <u>expresiones</u>	<u>Condición</u>

#### EXPRESIONES EN DBASE III PLUS

Una expresión puede estar formada con cualquiera o la combinación de: campos, variables de memoria, constantes y operadores. Ejemplo:

$$\text{Precio} = (\text{costo} * \text{cantidad}) * 1.06$$

<u>campo</u>	<u>variable</u> <u>de</u> <u>memoria</u>	<u>constante</u>
-----		
expresión		

#### REGLAS DE OPERACION EN EL DBASE

1) Funciona en base a comandos, que consisten en sencillos verbos que identifican de alguna forma la orden que debe ejecutar el paquete, por lo que los comandos deben comenzar con alguno de estos verbos.

2) Además del verbo cada comando puede tener frases opcionales que califican o condicionan la ejecución del mismo, sin exceder un total de 254 caracteres, y cumpliendo con la sintaxis o estructura permitida por el comando.

3) Los comandos pueden ser abreviados utilizando los primeros 4 caracteres del verbo que lo compone, y acepta "n" blancos entre sí.

4) Existen también las funciones, las cuales al igual que las variables y expresiones, deberán acompañar a un comando.

#### GLOSARIO DE LAS ORDENES MAS IMPORTANTES

Cada nombre de orden va seguido por la sintaxis de ésta, así como una descripción de lo que hace. Para algunas órdenes se proporcionan ejemplos de aplicación.

Se comenzará por establecer los siguientes convenios:

1) Todas las órdenes están escritas en mayúscula, aunque de hecho se pueden introducir en mayúsculas o minúsculas.

2) Todos los parámetros de las órdenes están escritos con minúsculas.

3) Cualquier parte de la orden o parámetro que vaya encerrado entre los corchetes izquierdo y derecho - [ y ] - es opcional.

4) Cuando en una orden 2 operaciones van separadas por una barra, como ON/OFF, se debe especificar una opción, pero no ambas.

5) Los puntos suspensivos (...) siguiendo a una orden o parámetro significan que éste se puede repetir "indefinidamente", es decir hasta que se acabe con la memoria de la computadora.

6) El parámetro alcance, que es siempre opcional, puede tener 3 significados diferentes dependiendo de la orden: ALL.- para todos los registros; NEXT n.- para los siguientes n registros a partir de la posición actual del puntero de registros; y RECORD.- para un solo registro a partir de la posición actual del puntero de registros.

Orden: @

Sintaxis: @ <línea, columna> [SAY expresión] [(GET variable) [PICTURE expresión] [RANGE exp, exp]] [CLEAR] [DOUBLE]

La orden @ sitúa al cursor en un lugar determinado de la pantalla, que es definido por línea y columna. Esta orden se puede usar con una o más de las siguientes opciones:

- SAY.- visualiza la expresión que está a continuación de ella escrita.

- GET.- permite una edición a toda pantalla de la variable.

- PICTURE.- permite el uso de pantallas, que especifiquen la forma en que los datos serán visualizados o aceptados en respuesta a la GET.

- RANGE.- se usa con la opción GET para especificar el rango de entradas aceptables.

Ejemplo:

Para colocar el mensaje <<Introduzca el nombre del accionista>> en la posición de la pantalla 12,2 y para permitir una edición a toda pantalla del valor contenido en la variable NAC, introduzca:

```
@12,2 SAY "Introduzca el nombre del accionista" GET NAC
```

Cuando se usan las opciones TO y DUBLE, la orden @ dibuja líneas y rectángulos simples o dobles (o una combinación) en la pantalla.

El primer valor representa la coordenada superior izquierda en la pantalla, y el segundo valor la coordenada inferior derecha. Si ambas coordenadas tienen el mismo valor vertical y horizontal, lo que se dibuja es una línea; de otro modo, se dibuja un cuadrado rectangular. Se puede usar la opción CLEAR para borrar una línea o un rectángulo previamente dibujados.

**Ejemplo:**

Para dibujar una línea simple desde la fila 3, columna 5 a la fila 3 columna 50, introduzca lo siguiente:

```
@3.5 TO 3.50
```

Para dibujar un cuadrado a doble línea con vértice superior izquierdo en la fila 4, y el vértice inferior derecho en la fila 18, columna 70, introduzca:

```
@4.1 TO 18.70 DOUBLE
```

Para borrar el recuadro dibujado en el ejemplo anterior, introduzca:

```
@4.1 CLEAR TO 18.70
```

Orden: APPEND FROM

Sintaxis: APPEND FROM nomfich [FOR/WHILE condicion][SDF/DELIMITED]

APPEND FROM copia registros del nomfich y los añade a la base de datos activa. La opción FOR/WHILE especifica una condición que se debe alcanzar para que se copie cualquier registro. Si el nomfich conteniendo los datos a ser copiados no es una base de datos de DASE III PLUS, se debe usar la opción SDF o la opción DELIMITED.

Orden: CLEAR

Sintaxis: CLEAR

La orden CLEAR borra la pantalla y devuelve el cursor a la posición 0,0 (la esquina superior izquierda). Esta orden se puede usar también como una opción de la orden @ para borrar la pantalla por debajo y a la derecha de la posición especificada por dicha orden.

**Ejemplo:**

Para borrar la pantalla anterior, introduzca:

```
CLEAR
```

Para borrar la pantalla por debajo y a la derecha del cursor en 12,20, introduzca:

```
@12.20 CLEAR
```

Orden: CLOSE

Sintaxis: CLOSE tipo-archivo

Esta orden cierra todos los tipos de archivo contenidos en tipo-archivo. Tipo-archivo puede ser uno de los 5 siguientes: ALTERNATE, DATABASES, FORMATS, INDEX, o PROCEDURE.

Orden: CONTINUE  
 Sintaxis: CONTINUE

Esta orden continua una búsqueda iniciada por LOCATE. Después de que LOCATE encuentra el registro que cumple el criterio especificado en la orden se pueden encontrar más registros con el mismo criterio, introduciendo CONTINUE. Utilizar CONTINUE le exige de volver a introducir la orden LOCATE. (ver LOCATE).

Orden: COPY  
 Sintaxis: COPY TO nom-fich [alcance][FIELDS lista-campos]  
 [FOR/WHILE condicion][SDF/DELIMITED][WITH delimitador]]

Esta orden copia una parte o toda la base de datos activa en nom-fich. Si alcance no está presente se asume ALL (todos). La opción FIELDS se usa para especificar los campos que han de ser copiados. La opción FOR/WHILE copia solo aquellos registros en los que se cumpla la condición. Especificando SDF se copiará el archivo en forma de Datos del Sistema; especificando DELIMITED, se copiará el archivo en forma Delimitado.

Ejemplo:

Para copiar los campos APELLIDO, RENTAMES y EXTRAS de la base de datos actual ABC1 en MUNDIAL, introduzca:

COPY TO MUNDIAL FIELDS APELLIDO, RENTAMES, EXTRAS

Orden: CREATE  
 Sintaxis: CREATE nom-fich

La orden CREATE crea un archivo de base de datos y define su estructura. Si CREATE va seguido por un nombre de archivo, se creará una base de datos con ese nombre. La extensión del nombre del archivo .DBF se añade automáticamente al nombre del archivo, a no ser que se especifique otra cosa.

Orden: DIR  
 Sintaxis: DIR [unidad][nomfich]

Esta orden visualiza el directorio de todos los archivos de base de datos o archivos de un tipo específico si se indica una extensión de archivo. Unidad es el indicador de la unidad (A:, o B:, o C: para usuarios de disco duro), y nomfich es el nombre de un archivo con o sin extensión. Las características comodín, el asterisco y la interrogación, se pueden usar para reemplazar en parte o totalmente a nomfich. En el caso de archivos de base de datos, DIR produce también la visualización del número de registros, la fecha de la última actualización y el tamaño del archivo (en bytes).

Ejemplo:

Para visualizar todos los archivos índice de la unidad por defecto actual, introduzca:

DIR \*.NDX

Orden: DO

Sintaxis: DO nomfich (WHILE lista-parámetros)

Esta orden arranca la ejecución de un archivo de órdenes del DBASE III PLUS. La extensión del nombre del archivo .PRG se asume a no ser que se especifique otra cosa. Si se especifica la opción WITH seguida por una lista de marámetros, lista-parámetros, se transfieren dichos parámetros al archivo de órdenes.

Orden: DO CASE

Sintaxis: DO CASE

```

CASE condición
  órdenes...
[CASE condición]
  órdenes...
[OTHERWISE]
  órdenes
ENDCASE
  
```

La orden DO CASE selecciona una opción de entre varias opciones. Las condiciones que siguen a las sentencias CASE son evaluadas hasta que se encuentra una que es cierta. Cuando una condición es cierta, se ejecutan las órdenes entre la sentencia CASE y otro CASE, u OTHERWISE y ENDCASE. Si no se encuentra ninguna condición cierta en las sentencias CASE, las órdenes que sigan a la sentencia opcional OTHERWISE serán ejecutadas. Si no se usa la sentencia OTHERWISE y no se encuentra ninguna condición cierta, el DBASE III PLUS procede con la orden siguiente a la sentencia ENDCASE.

Ejemplo:

En las siguientes órdenes DO CASE, el dBASE escoge entre 3 posibles alternativas: (1) ejecutar un archivo de órdenes llamado MENU, (2) aumentar la base de datos, o (3) salir de dBASE III PLUS.

```

DO CASE
  CASE SELEC=1
    DO MENU
  CASE SELEC=2
    APPEND
  CASE SELEC=3
    QUIT
ENDCASE
  
```

Orden: DO WHILE

SINTAXIS: DO WHILE condición  
órdenes  
ENDDO

La orden DO WHILE ejecuta repetidamente las órdenes entre el DO WHILE y el ENDDO mientras que la condición sea cierta. Cuando el dBASE encuentra un DO WHILE, evalúa la condición que le sigue: si la condición es falsa, procede con la orden que siga al ENDDO, pero si la condición es cierta, ejecuta las órdenes que siguen al DO WHILE hasta que encuentra la orden ENDDO. Cuando encuentra la orden ENDDO se vuelve a ejecutar la condición que sigue al DO

WHILE. Si todavía es cierta, se vuelven a ejecutar las órdenes contenidas entre el DO WHILE y el ENDDO. Si la condición es falsa, el dBASE III PLUS procede con la orden siguiente al ENDDO.

Ejemplo:

Para imprimir los campos APELLIDO, CIUDAD, ESTADO, RENTAMES, FECHAEXP de cada registro hasta el final del archivo (realmente el final de la base de datos), introduzca:

```
DO WHILE .NOT. EOF()
# APELLIDO, CIUDAD, ESTADO, RENTAMES, FECHAEXP
SKIP
ENDDO
```

Orden: EXIT  
Sintaxis: EXIT

Esta orden provoca la salida de un bucle DO WHILE y procede con la primera orden que siga al ENDDO.

Ejemplo:

La siguiente parte del archivo de órdenes usa EXIT para salir del bucle DO WHILE si se introduce un número de pieza 9999.

```
# "Introduzca el número de pieza al inventario."
# "Introduzca 9999 para salir"
DO WHILE .T.
INPUT TO NUMPIEZA
IF NUMPIEZA=9999
EXIT
ENDIF
ENDDO
```

Orden: IF  
Sintaxis: IF condición  
órdenes  
ELSE  
órdenes  
ENDIF

IF es una orden de toma de decisiones que ejecutará órdenes cuando se cumplan ciertas condiciones. Si la condición de la sentencia IF es cierta, se ejecutarán las órdenes entre el IF y el ENDIF. Si la condición es falsa y existe un ELSE, las órdenes que se ejecutarán serán las que estén entre el ELSE y el ENDIF. Por otro lado, si la condición es falsa y no hay un ELSE, se saltará a la sentencia ENDIF sin ejecutar ninguna orden.

Orden: GO o GOTO  
Sintaxis: GO o GOTO BOTTOM/TOP/expresión

La orden GO o GOTO posiciona al puntero al puntero de registros en un registro. GO TOP moverá el puntero al principio de la base de datos, mientras que GO BOTTOM lo moverá al final de la base de datos.



Orden: LIST

Sintaxis: LIST [OFF][alcance][lista-campos][FOR/WHILE condición]  
[TO PRINT]

Esta orden proporciona una lista del contenido en la base de datos. La opción alcance se usa para cuantificar los registros a ser listados. Si el alcance está ausente, se asume ALL (todos). La opción FOR/WHILE especifica una condición que se debe alcanzar para que un registro se liste. La opción OFF evitará que se listen los números de los registros. Si se usa la opción TO PRINT, el listado se imprimirá en la impresora.

Orden: MODIFY COMMAND

Sintaxis: MODIFY COMMAND nomfich

Esta orden pone en marcha el procesador de textos del dBASE III PLUS, que se podrá usar para editar archivos de órdenes o archivos texto ASCII. Al nomfich se le dará la extensión .PRG.

Orden: PROCEDURE

Sintaxis: PROCEDURE

Esta orden identifica el comienzo de cada procedimiento de un archivo de procedimientos.

Aunque el uso de procedimientos de una línea es ineficiente (los procedimientos deben tener cuando menos 3 líneas), el siguiente ejemplo muestra un procedimiento sencillo:  
Ejemplo:

```
PROCEDURE ERRORI
@2,10 SAY "Eso no es una fecha válida. Pruebe nuevamente"
RETURN
```

Orden: READ

Sintaxis: READ [SAVE]

Esta orden permite la entrada de datos a toda pantalla desde una orden @ con la opción GET. Normalmente, una orden READ borra todos los GETS cuando se termina la introducción o la edición de los datos. La opción SAVE sirve para que no se borren los GET al terminar la introducción o la edición de los datos.

Orden: RETURN

Sintaxis: RETURN [TO MASTER]

La orden RETURN finaliza la ejecución de un archivo o procedimiento de órdenes. Si el archivo de órdenes fue llamado por otro archivo de órdenes, el control del programa es devuelto al otro archivo de órdenes. Si el archivo de órdenes no fue llamado por otro archivo de órdenes, el control vuelve al punto de petición. Si se usa la opción TO MASTER, el control vuelve al archivo de órdenes de más alto nivel.

Orden: RUN  
 Sintaxis: RUN nomfich

Esta orden ejecuta un programa que no es del dBASE III desde el entorno de dBASE III. El programa debe tener la extensión .COM o .EXE. Cuando el programa cumple su ejecución, el control será devuelto al dBASE. Se pueden ejecutar también órdenes de DOS con RUN, suponiendo que hay suficiente espacio en memoria.

Orden: SAVE  
 Sintaxis: SAVE TO nomfich [ALL LIKE/EXCEP comodín]

Esta orden copia las variables de memoria en un archivo de disco. Los comodines que son asterisco o signo de interrogación se usan con las opciones LINK y EXCEPT. Los asteriscos se pueden usar para representar uno o más caracteres, y la interrogación para representar un carácter.

Ejemplo:

Para salvar todas las variables de memoria existentes que tengan 6 letras y terminen con las letras IMP, en un archivo en disco llamo FIGURAS, introduzca:

SAVE TO FIGURAS ALL LIKE ###IMP

Orden: SET  
 Sintaxis: SET

Hace que aparezca el menú SET. El menú SET se puede usar para seleccionar la mayoría de los parámetros.

Orden: SET FORMAT  
 Sintaxis: SET FORMAT TO nomfich

Esta orden permite seleccionar un nomfich para el formato de salida por pantalla. Si nomfich tiene la extensión .FRM no es necesario indicar dicha extensión.

Orden: USE  
 Sintaxis: USE [nomfich][INDEX lista-fich][ALIAS alias]

Esta orden abre un archivo de base de datos y sus archivos índice asociados en un área de trabajo.

#### I.5 QUATRO

El presente paquete es una hoja de cálculo electrónica: la

- (2) Hoja electrónica.- Paquete que permite a los usuarios crear, manipular y analizar rápidamente datos que se organizan en columnas y renglones.

cual ofrece 254 columnas por 2,048 renglones o más de 500,000 celdas.

Una característica de este programa además de las ventajas que por sí solo posee, es la capacidad de poder programar funciones, denominando a esto MACROS de teclado, es decir, cualquier operación que se realice de manera manual se podrá ejecutar de manera automática.

#### ACCESO AL SISTEMA CUATRO

Para tener acceso al sistema, lo primero que hay que hacer una vez que la computadora tenga cargado el Sistema Operativo y el mismo Quattro es teclear la letra Q ó q, quedando:

C>Q ó C>q

Al teclear esto entramos a la Hoja Electrónica.

Una hoja electrónica es como ya se dijo una colección de CELDAS organizadas en renglones y columnas. Cada celda puede almacenar una pieza sencilla o compleja de información (Número, fórmula o etiqueta). La pantalla de la computadora es una ventana de la Hoja Electrónica. Es posible ver cualquier parte de la Hoja Electrónica con tan solo desplazarnos en ella, pero no es posible ver toda la Hoja Electrónica a un mismo tiempo.

Los 2 primeros renglones que se encuentran en la pantalla los denominaremos PANEL DE CONTROL, este se encuentra dividido en 3:

-El primer renglón indica en que celda se encuentra el CURSOR DE CELDAS, además del contenido y el formato de esa celda. Por ejemplo puede decir:

M10: (C2) U 3450

La celda en la que se encuentra el cursor es M10, se encuentra en formato CURRENCY a 2 decimales, se encuentra desprotegida (UNPROTECTED) y el contenido es el número 3450.

-El segundo renglón es de EDICION, todo lo que escribimos siempre pasará por este renglón y posteriormente pasará al primer renglón de contenido; en este renglón podemos realizar las modificaciones a la celda. Si deseamos modificar la información que se encuentra en el primer renglón lo debemos bajar al segundo presionando la tecla F2 (EDIT).

-El segundo renglón también nos muestra en su segunda mitad derecha una breve explicación de los comandos.

Una barra es colocada en video inverso dentro de la Hoja Electrónica (visualizándose en pantalla), la cual la denominaremos CURSOR DE CELDAS y cuya función es señalarnos la celda en la cual estamos posicionados en ese momento. El CURSOR DE CELDAS es el foco de atención de la Hoja Electrónica, ya que conociendo cual es su funcionamiento podemos, por ejemplo, mover el cursor a una celda en particular o teclear una entrada ó un comando que afecte el contenido de la celda.

Lo anterior podría ser tan solo una aplicación, ya que existen comandos que procesan al mismo tiempo un grupo de celdas, esto es llamado RANGO (RANGE'). En algunas situaciones se puede indicar el rango expandiendo el CURSOR a muchas celdas a la vez. Para lograr esto tenemos que utilizar las flechas que se

encuentran en la parte derecha del teclado de la computadora.

#### MOVIENDOSE A UNA LOCALIDAD ESPECIFICA

Solo con prestar la tecla <F5> que es la de <GoTo> y después prestando las coordenadas de la celda a la que nos queremos dirigir, automáticamente se posiciona el cursor en esa localidad, no importando que tan lejos se encuentre.

#### INGRESANDO DATOS A LAS CELDAS

Cuando se comienza una sesión de Quattro se ve una Hoja Electrónica vacía. Una de las principales tareas es llenar las celdas con números, fórmulas o etiquetas para alguna aplicación específica.

Existen 3 formas básicas para dar la entrada a una celda:

1.- Teclando un número, una fórmula o una etiqueta directamente del teclado.

2.- Con las opciones /Copy o /Move. Lo que hacen es copiar o mover entradas que ya se encontraban en otras celdas de la Hoja Electrónica.

3.- Bajar datos de un archivo de un disco, ya sea creado con Quattro o con otro programa.

#### INGRESANDO DATOS

Para dar entrada a una celda particular solo debemos mover el cursor a la celda que se desea, dar la entrada numérica o alfanumérica y al terminar presionar la tecla RETURN. Todo carácter que se teclee como entrada aparecerá en la segunda línea del panel de control, arriba de la hoja electrónica.

#### TIPOS DE ENTRADAS

Existen 3 tipos de entradas que se pueden crear: Números, fórmulas o etiquetas.

#### INGRESO DE NUMEROS Y FORMULAS

Si se comienza la entrada a una celda con cualquiera de los siguientes caracteres, la Hoja Electrónica "pensará" que se desea ingresar un número o una fórmula:

0 1 2 3 4 5 6 7 8 9 + - . ( @ # \$

Es sencillo visualizar porque la Hoja Electrónica asume que se desea ingresar una entrada numérica o una fórmula si se inicia con cualquiera de estos pocos caracteres, ya que estos son las construcciones básicas de los bloques de números y expresiones

aritméticas. Los caracteres @ y # son usados para crear fórmulas más sofisticadas que envuelven funciones y condiciones lógicas. El carácter % es usado para propósitos similares.

La Hoja Electrónica anuncia su decisión después de que se teclée el primer carácter cambiando el indicador READY en la esquina superior derecha por el de VALUE.

#### REGLAS PARA LA ENTRADA DE NUMEROS

Existen algunas reglas que se deben de seguir para la entrada de números. Si una entrada no sigue las siguientes reglas, la Hoja Electrónica no la captará cuando se prestone RETURN sonando un BEEP y colocando el cursor en el lugar del problema y cambiando de VALUE a modo de edición EDIT en la esquina superior derecha.

1.- Un número puede empezar con un dígito (0....,9), un signo de más o de menos, un punto o un signo de i.

2.- Un número puede terminar con el carácter % para indicar que es un por ciento. Este carácter tiene el mismo efecto que dividir entre 100 (divide el número precedente entre 100)

3.- Un número puede tener solo un punto decimal.

4.- No se pueden utilizar ni comas ni espacios cuando se introduzca un número. Sin embargo la Hoja Electrónica puede mostrar los números en la pantalla con comas y signos de pesos con comandos específicos de formateo.

5.- Se puede determinar un número con potencias de 10, llamado de notación científica.

#### ENTRADA DE ETIQUETAS

Si el primer carácter que se teclée en una celda no es ninguno de los listados anteriormente para entradas numéricas o fórmulas, significará que se está ingresando una etiqueta. Esto lo anunciará la Hoja Electrónica cambiando el indicador de READY por el de LABEL.

#### ESPECIFICANDO EL RANGO DE LAS FORMULAS

Las Hojas Electrónicas pueden calcular con rangos los valores tan bien como con valores individuales. Los rangos de valores no pueden ser procesados con la operaciones aritméticas ordinarias.

Por ejemplo, no se pueden sumar o restar 2 rangos como 2 valores independientes, esto es, no se pueden procesar rangos con funciones especiales, llamadas funciones.

Un rango es un rectángulo de celdas contiguas, este rectángulo puede ser de un renglón o dos o más celdas adyacentes, una columna de celdas adyacentes o un bloque que contenga 2 o más renglones o columnas. Una celda sencilla también podemos decir que es un rango.

Quattro identifica un rango por sus primeras y últimas coordenadas, existen varias formas de nombrar un rango, una de

ellas es ingresar 2 esquinas del rango y separarlo por un punto (aunque la computadora al momento de procesar coloque dos puntos seguidos), por ejemplo, un rango puede ser:

A20..H70

este rango contiene todos los renglones del 20 al 70 y todas las columnas de la A a la H.

Podemos decir que para casi cualquier comando y cualquier función del Quattro es necesario utilizar RANGOS, de hecho existe un comando denominado /Range.

### CORRIGIENDO ERRORES

Cuando el indicador se encuentra en VALUE o en LABEL, podemos cancelar alguna entrada y regresar a la modalidad de READY, solo con presionar la tecla (ESC). Si queremos hacer alguna cancelación, debemos realizarla antes de presionar la tecla RETURN o antes de hacer algún movimiento del cursor de celdas.

En caso de que el programa se encuentre en la modalidad POINT será necesario presionar la tecla (ESC) 2 veces para salir a la modalidad READY, si deseamos salir de manera más rápida de alguna modalidad solo con presionar las teclas de (ctrl) y luego sin soltar esta presionar la tecla de (Break), así se regresará de manera inmediata a la modalidad READY.

### REEMPLAZO

La forma más sencilla de cambiar una entrada sería DESPUES de haber sido ingresada, colocar el cursor de celdas en aquella que queremos modificar y después ingresar la entrada nueva en esa celda. El contenido original de la celda se pierde, por lo que existe la necesidad de tener la seguridad de que el cambio a realizar no vaya a contener información importante.

### BORRADO

Para borrar el contenido de una o más celdas se utiliza el comando /RANGE ERASE. Este comando no realiza la misma operación que la de reemplazar con espacios, ya que los espacios para la computadora son caracteres. Una celda borrada es aquella que no contiene caracteres.

La forma más rudimentaria para blanquear una celda es con la tecla (Back-Space), presionando esta tecla hasta que ya no exista ningún carácter en el panel de control y después dando entrada, con esta secuencia de caracteres nulos, se blanqueará esta celda.

### Jerarquía de las Operaciones en las Formulas

Los caracteres que indican operaciones aritméticas, los llamaremos OPERANDOS, y son los que a continuación aparecen:

+	SUMA
-	RESTA

\*  
MULTIPLICACION  
DIVISION

En las fórmulas, las operaciones son realizadas de izquierda a derecha. Existen 2 formas para indicar que las operaciones se realicen en otro orden. Una manera es explícita, es decir utilizando paréntesis en la sección de la fórmula que se quiera dar prioridad.

Es posible colocar paréntesis dentro de paréntesis, esto es llamado PARENTESIS ANIDADOS.

La otra forma de alterar el orden de las operaciones es utilizando la JERARQUIA DE OPERACION, esto es utilizando las ideas del álgebra.

Las operaciones siguientes y su significado son escritos de mayor a menor jerarquía, para que se conozca de esta forma la jerarquía que cada una posee, no olvidando que: 1) Los operadores con mayor jerarquía son realizados primero, a no ser que se alteren por medio de paréntesis; y 2) Operadores de jerarquía igual serán realizados de izquierda a derecha.

^	EXPONENCIACION	7
-	NEGATIVO	6
*	MULTIPLICACION	5
/	DIVISION	5
+	SUMA	4
-	RESTA	4
=	IGUAL	3
<	MENOR QUE	3
<=	MENOR O IGUAL QUE	3
>	MAYOR QUE	3
>=	MAYOR O IGUAL QUE	3
<>	DIFERENTE	3
#NOT#	NEGACION LOGICA	2
#AND#	'Y' LOGICO	1
#OR#	'O' LOGICO	1

MODULO DE EDICION 'EDIT'

Algunas ocasiones es necesario realizar algún otro cambio que no sea el de blanquear toda la secuencia de caracteres de una celda.

Para este propósito, la Hoja Electrónica tiene una modalidad de Edición (EDIT MODE). Se puede invocar este modo cuando se quiere mover el cursor de celdas a cualquier celda en la que se requiera un cambio.

Para entrar al modo de EDIT se necesita presionar la tecla funcional (F2), que es la tecla de EDIT. El indicador en la parte superior derecha de la pantalla cambiará y colocará la palabra EDIT. También la Hoja Electrónica pasa automáticamente a la modalidad de edición EDIT cuando se ha ingresado alguna fórmula o número que no pueda entender (no se usó el prefijo correcto, se olvidó cerrar un paréntesis, etc.).

En el modo de edición (EDIT), la Hoja Electrónica coloca una copia del contenido de la celda, ya sea número, etiqueta o fórmula en la segunda línea del panel de control, colocando el cursor al final de la entrada, en este punto se pueden borrar o insertar caracteres.

## LA TECLA FUNCIONAL (F9) (CALC)

En modalidad READY, presionando la tecla (F9) causa que la Hoja Electrónica realice nuevamente una pasada calculando todas las fórmulas dentro de la hoja. Pero en modalidad EDIT esta tecla convierte la fórmula en el número correspondiente de la celda en que nos encontremos posicionados (este proceso no es reversible).

## SALIDA DEL MODO DE EDICION

Si se invocó el modo EDIT desde el modo VALUE o LABEL es posible regresar a dicho modo presionando la tecla (F2) nuevamente.

## FUNCIONES

Se puede pensar que las funciones son como fórmulas previamente construidas. Como las fórmulas el resultado de estas funciones pueden ser valores numéricos o lógicos. Este valor depende usualmente de argumentos, números, celdas o referencias de rango.

Para realizar una función se tecldea primero el nombre de la misma seguido de sus argumentos. El nombre la función siempre principia con el signo @; el resto del nombre puede ser tecldeado con mayúsculas o con minúsculas o con cualquier combinación. Los argumentos que siguen a una función deben de estar entre paréntesis y separados por comas o indicando un rango dependiendo de la función.

Ejemplo:

@nombrefunción(arg1, arg2, ..., argN)

@nombrefunción(rango)

## FUNCIONES MATEMATICAS

@ABS(x)	Valor absoluto
@ACOS(x)	Arco coseno
@ASIN(x)	Arco seno
@ATAN(x)	Arco tangente
@COS(x)	Coseno
@SIN(x)	Seno
@TAN(x)	Tangente
@EXP(x)	Exponencial
@INT(x)	Parte entera
@MOD(x)	Residuo
@RAND(x)	Número random
@SQRT(x)	Raíz cuadrada
@PI	Valor de la constante PI
@ROUND	Redondeo a un número de n lugares decimales



### FUNCIONES LOGICAS

@FALSE	El valor 0 (falso)
@TRUE	El valor 1 (verdadero)
@ISNAC(x)	El valor 1 (verdadero) si la expresión x tiene un valor NA
@ISERR(x)	El valor 1 (verdadero) si la expresión en x tiene el valor ERR
@IF(cond, x, y)	El valor x si la condición es verdadera, y el valor y si la condición es falsa

Nota.- Las funciones @FALSE y @TRUE no tienen argumento.

### FUNCIONES ESPECIALES

@ERR	El valor ERR (error)
@CHOOSE(x, v0, v1, ..., vn)	Selecciona el valor del argumento
@HLOOKUP(x, range, offset)	Búsqueda horizontal en la tabla
@VLOOKUP(x, range, offset)	Búsqueda vertical en la tabla

### FUNCIONES ESTADISTICAS

@COUNT(lista)	Cuenta el número de conceptos en una lista
@SUM(lista)	Suma los valores en una lista
@AVG(lista)	Calcula el promedio de los valores en una lista
@MIN(lista)	Extrae el valor mínimo de una lista
@MAX(lista)	Extrae el valor máximo de una lista
@STD(lista)	Calcula la desviación estándar de un conjunto de números
@VAR(lista)	Calcula la varianza de un rango de números

### FUNCIONES ESTADISTICAS DE UNA BASE DE DATOS

Estas funciones operan con los valores de campos particulares de una base de datos donde los registros fueron encontrados en base a un criterio determinado.

@DCOUNT(contrada, offset, criterio).- cuenta el número de elementos de los conceptos de una lista.

DSUM(contrada, offset, criterio).- Suma los valores de una lista de base de datos.

@DAVG(contrada, offset, criterio).- Calcula el promedio de los valores de una lista de base de datos.

@DMIN(contrada, offset, criterio).- Extrae el valor mínimo de una lista.

@DHAX(entrada, offset, criterio).- Extrae el valor máximo de una lista.

@DSTD(entrada, offset, criterio).- Calcula desviación estándar de todos los números de una lista.

@DYAR(entrada, offset, criterio).- Calcula la varianza de todos los elementos de una lista.

Donde:

entrada.- Es el rango donde se encuentra localizada la base de datos, incluyendo los nombres de campo.

offset.- Es la columna en la cual deseamos realizar la operación de la función, esta columna se cuenta a partir de la primera columna de la base de datos y es la columna 0 (cero).

criterio.- Es lo que deseamos que realice la operación, por ejemplo, aquellas personas que se encuentran en el PISO 4.

#### ENTRADA AL MENU DE COMANDOS

Ahora bien, si tecleamos el caracter / llegamos al menú principal de Quattro que aparece a continuación:

WORKSHEET	RANGE	COPY	MOVE	FILE	PRINT
DATA	SISTEM	ADD-IN/MACRO	INSTALL	QUIT	

#### MENÚ DE ACCESO AL SISTEMA

Existen 2 formas de hacer una elección a un menú:

1.- Utilizar las teclas de movimiento del cursor para escoger la opción deseada, cuando la opción tome la manera inversa en cuanto a color que aparece en el monitor, podemos aceptar la opción presionando la tecla RETURN.

2.- Como segunda forma de elegir una opción del menú es únicamente presionando la primera tecla de la opción a escoger.

ORDEN Y ORDENAMIENTO DEL MENÚ Y SUBMENUS  
DENTRO DEL CUATRERO

En las páginas subsiguientes aparecen los diferentes submenús que conforman cada opción del menú, así como sus respectivas ramificaciones:

```

-----
:Worksheet:
:   :Global      :Format        :Fixed         :
:   :           :              :Scientific   :
:   :           :              :Currency     :
:   :           :              :.(coma)      :
:   :           :              :General      :
:   :           :              :+/-         :
:   :           :              :Porcent      :
:   :           :              :Date (1/2/3) :
:   :           :              :Text         :
:   :           :              :Hidden       :
:   :           :              :Reset        :
:   :           :-----
:   :           :Label-Prefix :Left/Right/Center :
:   :           :-----
:   :           :Column-Width :
:   :           :-----
:   :           :Recalculation: Natural       :
:   :           :              :Columnwise   :
:   :           :              :Rowwise      :
:   :           :              :Automatic    :
:   :           :              :Manual       :
:   :           :              :Iteration    :
:   :           :-----
:   :           :Protection   :Enable/Disable  :
:   :           :-----
:   :           :Default      :Printer         :Interface      :
:   :           :              :                :Auto-LF       :
:   :           :              :                :Left/Right    :
:   :           :              :                :Top/Bottom    :
:   :           :              :                :Page-Length   :
:   :           :              :                :Wait          :
:   :           :              :                :Setup         :
:   :           :              :                :Name          :
:   :           :              :                :Quit          :
:   :           :              :-----
:   :           :              :Directory      :None/A..G     :
:   :           :              :-----
:   :           :              :Status         :
:   :           :              :Update        :
:   :           :              :Others        :
:   :           :              :Resource      :
:   :           :              :Quit          :
:   :           :-----
:   :           :Zero         :
:   :           :-----

```

```
: :-----: :  
: :Insert :Row/Column :  
: :-----: :  
: :Delete :Row/Column :  
: :-----: :  
: :Column-Width: Set/Reset/Hide/Display :  
: :-----: :  
: :Titles :Horiz/Vert/Both/Clear :  
: :-----: :  
: :Window :Horiz/Vert/Clear :  
: : :Sync/Unsync :  
: :-----: :  
: :Erase (Yes/No) :  
: :Status :  
: :Page :  
: :-----: :
```

## COMANDOS DEL WORKSHEET

```

-----
: Range
:
: Format
: Fixed
: Scientific
: Currency
: , (coma)
: General
: -/+
: Percent
: Date (1/2/3)
: Text
: Reset
: Hidden
:
: -----
: Label-Prefix : Left/Right/Center
:
: Erase
:
: Name
: Create/Delete
:
: Labels
: Left/Right
: Up/Down
:
: Table
:
: Reset
:
: -----
: Justify
:
: Protect/Unprotect
:
: Input
:
: Transpose
:
: Value
:
: Search/Re- : Range
: place : Search String
: : Replace String
: : Go
: : Reset
: -----
: Copy
: -----
: Move
: -----

```

COMANDOS RANGE, COPY y MOVE

```

-----
:File      :Retrive
:          :
:          :Save (Cancel/Replace)
:          :
:          :Combine      :Copy (Entire File/Named Range)
:          :              :Edd/Subtract (Entire File/Named Range)
:          :
:          :Xtract      :Formulas/Values
:          :
:          :Erase       :Workshrc
:          :              :Print
:          :              :Graph
:          :              :Other
:          :
:          :List        :Workshrc
:          :              :Print
:          :              :Graph
:          :              :Other
:          :
:          :Import      :Text/Numbers
:          :
:          :Directory   :
:          :
:          :/sqz+      :Remove Blank
:          :              :Storage of Values
:          :              :Verston
-----
:Print     :Printer/File:Range
:          :
:          :           :Line
:          :           :Page
:          :
:          :           :Options : Margins(Left/Right/Top/Bottom
:          :           :           :Footer
:          :           :           :Header
:          :           :           :Setup
:          :           :           :Page-Length
:          :           :           :Quit
:          :
:          :           :Borders  :Columns/Row
:          :
:          :           :Other    :As-Displayed
:          :           :           :Cell-Formulas
:          :           :           :Unformatted
:          :           :           :Formatted
:          :
:          :           :Clear   :Range
:          :           :           :All
:          :           :           :Borders
:          :           :           :Format
:          :
:          :           :Align
:          :           :           :Go
:          :           :           :Quit
-----

```

```

-----
: Data      : Fill
:           : Table (1/2)
:           : Distribution
:           :
:           : Sort      : Data-Range
:           :           : Primary-Key (Ascending/Descending)
:           :           : Secondary-Key (Ascending/Descending)
:           :           : Reset
:           :           : Go
:           :           : S
:           :           : 4
:           :           : 5
:           :           : Quit
:           :
:           : Query     : Input
:           :           : Criterion
:           :           : Output
:           :           : Find
:           :           : Extract
:           :           : Unique
:           :           : Delete
:           :           : Reset
:           :           : Logical-Criteria
:           :           : Assgn-Names
:           :           : Quit
:           :
:           : Matrix    : Invert/Multiplay
:           :
:           : Regression : x-Range
:           :           : y-Range
:           :           : Output Range
:           :           : Intercep
:           :           : Compute
:           :           : Intercept
:           :           : Zero
:           :           : Reset
:           :           : Go
:           :           : Quit
:           :
:           : Parse     : Input- Column
:           :           : Output Range
:           :           : Format-Line
:           :           : Create
:           :           : Edit
:           :           : Reset
:           :           : Go
:           :           : Quit
-----

```

COMANDOS DATA

```

-----
: Graph :
: :Type : Stacked-Bar :
: : : Bar :
: : : Line :
: : : Pie :
: : : XY :
: : : 3-Dimensional :
: : : Area :
: : : Rotated Bar :
: : : Markers :
: : : Combined Lines and Markers :
: :-----:
: :Reset : Graph :
: : : X, A, ..., F :
: : : Quit :
: :-----:
: :Options : Color/B/N :
: : : Legend :
: : : Data-Labels :
: : : Quit :
: :-----:
: : :Format : Graph :
: : : : Quit :
: :-----:
: : : : A, ..., F : Lines :
: : : : : Symbols :
: : : : : Both :
: : : : : Neither :
: : : : : Vertical Bars :
: :-----:
: : :Titles : First/Second :
: : : : X & Y axes :
: : : : Main Font :
: : : : Character Size :
: :-----:
: : :Orient : Horiz/Vert :
: : : : Both :
: : : : Clear :
: :-----:
: : :Scale : X Y-axes : Automatic :
: : : : : Manual :
: : : : : Lower/Upper :
: : : : : Format :
: : : : : Indicator :
: : : : : Quit :
: :-----:
: : :Patterns: A, ..., F :
: : : : Markers: Update :
: : : : : View :
: : : : : Quit :
: :-----:
: : : :Ship Factor : Release IA notes :
: :-----:
: :Name : Use :
: : : Create :

```



```

:           :Delete
:           :Reset
:-----:
: X
: A.....F
: View
: Save
: Quit
:-----:
: Graph Custom: A.....F :Range
:           :Legend
:           :Interior Labels
:-----:
:           :Placement of:Center
: Labels    :Left
:           :Above
:           :Right
:           :Below
:           :None (to reset)
:-----:
:           :Color
:-----:
:           :Override :Line
:           :Type     :Markers
:           :         :Combine Line-Mark.
:           :         :Bar
:           :         :None (for labels)
:-----:
:           :Markers  :A-Filled Square
:           :         :B-Plus
:           :         :C-8-point Star
:           :         :D-Empty Square
:           :         :E-X
:           :         :F-F
:           :         :G-Filled Triangle
:           :         :H-Hourglass
:           :         :I-6-point Star
:           :         :J-Square with x ins
:           :         :K-Shadowed Cross
:           :         :L-Vertical Line
:           :         :M-Horizontal Line
:-----:
:           :Fill Pattern: A-Empty
:           :         :B-Filled
:           :         :C- ----
:           :         :D-Lt ///
:           :         :E-Hvy //
:           :         :F-Lt \\\
:           :         :G-Hvy \
:           :         :H-+++++++
:           :         :I-Crosshatch
:           :         :J-Hatch
:           :         :K-Light Dots
:           :         :L-Heavy Dots
:-----:
:           :Reset
:           :View
:           :Quit

```







```

:-----:
:Install :Hardware :Color :
: : :IBM Monochrome :
: : :Screen Snows :
: : :Use Special Drive :
: : :Drive Name :
: : :Graph Scree :
: : :Resolution :
: : :Aspect Ratio :
: : :Quit :
:-----:
: :Colors :Menu :Frame :
: : : :Banner :
: : : :Text :
: : : :Key Letter :
: : : :Highlight :
: : : :Settings :
: : : :Explanation :
: : : :Drop Shadw :
: : : :Quit :
:-----:
: : :Spreads-: Borders :
: : :heet : Cells :
: : : :Titles :
: : : :Descriptor Line :
: : : :Highlight :
: : : :Status Line :
: : : :Input Line :
: : : :Inprotected :
: : : :Labels :
: : : :Error Massages :
: : : :Quit :
:-----:
: : :Condi- : On/Off :
: : :tional : Err :
: : : :Smallest Normal Value :
: : : :Greatest Normal Value :
: : : :Below Normal Color :
: : : :Normal Cell Color :
: : : :Above Normal Color :
: : : :Quit :
:-----:
: : :Palettes: Color :
: : : :Monochrome :
: : : :Black & White 80 :
: : : :Quit :
:-----:
: : :Help :Banner :
: : : :Text :
: : : :Keyword :
: : : :Highlight :
: : : :Quit :
:-----:
: :Startup :Autoload File :
: : :Startup Macro :
: : :Extension :
: : :Beep :
:-----:

```



BAJANDO ENTRADAS DE ARCHIVOS DE DISCO

Se pueden cargar datos de un disco a la Hoja Electrónica como entradas numéricas y/o entradas de etiquetas. El uso más común es volver a cargar una Hoja Electrónica ya creada previamente y almacenada en disco.

El procedimiento para guardar una Hoja Electrónica que se encuentra en la memoria de la computadora a disco es el siguiente:

- 1) Entramos a la modalidad READY
- 2) Prestionar /File
- 3) Dentro del comando File pedir la opción de Save
- 4) Prestionar la tecla RETURN

Después de haber prestionado RETURN la computadora buscará en el disco los nombres de los archivos que correspondan a la Hoja Electrónica que se encuentren almacenados y nos los mostrará como también simultáneamente nos mostrará el mensaje:

ENTER FILE NAME:

5) Ingresar el nombre con el que deseamos que se llame nuestra Hoja Electrónica en el disco, después de haberlo ingresado la computadora procederá a checar si no es alguno de los que se tenían en disco, en caso de que ya haya sido dado el nombre que le dimos a la Hoja Electrónica nos mostrará las siguientes opciones con la indicación de:

CANCEL REFLACE

Indicando la opción de CANCEL que cancelará el guardar esa hoja a disco, es decir, CANCELAR la opción. Siendo la opción de REFLACE la operación de almacenar en disco el contenido que en ese momento posee la Hoja Electrónica que tenemos en la memoria principal de la computadora.

En caso de que se haya escrito otro nombre que no se encontraba en el disco, la computadora sin ninguna opción más guardará el contenido en la Hoja Electrónica de la memoria principal al disco.

Para llamar a una Hoja Electrónica de un disco a la memoria principal el procedimiento es similar:

- 1) Encontrarse en modalidad READY
- 2) Prestionar /File
- 3) Dentro del submenú del comando File seleccionar la opción:

Retrieve

- 4) Prestionar la tecla RETURN

Después de haber prestionado RETURN la computadora buscará en el disco los nombres de los archivos que correspondan a las Hojas Electrónicas que se encuentran almacenadas, los mostrará y

simultáneamente nos mostrará el mensaje:

ENTER FILE NAME:

5) En este momento podemos escoger algún nombre que nos proponga la computadora (moviéndonos con el cursor y presionando RETURN) o bien escribir el nombre del archivo que se quiera bajar a memoria principal.

NOTA: Cada vez que se baje algún archivo de disco a memoria, el contenido de la memoria de la computadora primero se borrará y luego cargará el archivo que tiene la Hoja Electrónica que deseamos bajar.

### GRAFICACION

Los comandos de graficación del Quattro son herramientas muy poderosas y flexibles para crear representaciones visuales de datos.

El comando /Graph tiene características que ayudan a dibujar gráficas fácilmente, dibujos que comunican los datos numéricos de una manera sencilla, las gráficas pueden ayudar a realizar relaciones y sacar conclusiones de un conjunto de datos que por sí solos es difícil que nos diga algo.

### COMANDOS DE GRAFICACION

GRAPH. - Realiza la gráfica y la impresión de los valores contenidos en la Hoja Electrónica.

TYPE. - Aquí se especifica el tipo de gráfica que se va a utilizar.

LINE. - Gráfica de líneas.

BAR. - Gráfica de barras.

PIE. - Gráfica de pastel.

STACKED-BAR. - Barras encimadas.

XY. - Gráfica por puntos.

3 DIMENSIONAL. - Gráfica en tercera dimensión.

AREA. - Gráfica el área de la gráfica como si estuviera llena.

ROTATED BAR. - Nos entrega rotada la gráfica.

MARKERS. - La gráfica aparece dentro de un recuadro.

X. - Define el conjunto de valores o etiquetas usadas para indicar ejes en x (parte de abajo de la gráfica).

A,B,C,D,E,F. - Define rango de números a dibujar en la gráfica. Es posible graficar hasta 6 rangos de números (A,B,C,D,E,F) en una misma gráfica. Quattro utiliza esta escala de rangos y la etiqueta de y-axis ejes en y para definir la línea vertical de la parte izquierda de la gráfica.

RESET. - Elimina TODAS las indicaciones dadas para



realizar una gráfica, o cualquiera de los 7 rangos X y A..F. También se borran estas indicaciones cuando se elige QUIT.

VIEW.- Le da instrucciones a Quattro para que dibuje la gráfica, basado en las últimas indicaciones dadas, dependen del monitor usado para tener acceso a esta opción.

MONITOR MONOCROMATICO.- No se puede ver la gráfica en la pantalla.

MONITOR GRAFICO (O MONITOR MONOCROMATICO CON TARJETA DE GRAFICAS).- Quattro limpia la pantalla y muestra la gráfica. Presionando cualquier tecla se limpia la gráfica y se regresa a ver la Hoja Electrónica.

SAVE.- Almacena una gráfica como archivo de disco. Las gráficas que han sido salvadas pueden ser mandadas a imprimir por medio del comando PRINT que aparece dentro del mismo GRAPH, con tan solo darle las especificaciones que requiere y elegir GO.

OPTIONS.- Opciones especiales de gráficas

LEGEND.- Etiqueta de cualquier rango de datos. La leyenda aparece abajo de la gráfica.

FORMAT - Se coloca formato para una gráfica o para uno de los rangos de A-F.

LINES.- La gráfica o alguno de los rangos puede verse únicamente las líneas.

SYMBOLS.- La gráfica o alguno de los rangos puede verse únicamente los símbolos dependiendo del rango que se haya utilizado.

BOTH.- La gráfica mostrará ambos, líneas y símbolos en la gráfica.

NEITHER - La computadora no mostrará ningún rango en la gráfica.

TITLES.- Coloca los títulos en la gráfica.

FIRST.- Coloca el primer título en la primera línea de la gráfica.

SECOND.- Coloca el segundo título en la segunda línea de la gráfica.

X-AXIS.- Da el título al eje x.

Y-AXIS.- Da el título al eje y.

GRID.- Muestra una red sobre la gráfica, pueden

ser líneas horizontales, líneas verticales, ambas líneas o pueden eliminarse con la opción CLEAR.

SCALE. - Formatea escala de los ejes. El Quattro automáticamente calcula la escala, a no ser que se le especifique que la escala se dará de manera MANUAL. Si se utiliza la opción de escala MANUAL es necesario especificar los límites superiores de la gráfica. No es posible dar la escala a una gráfica de PIE.

X-AXIS. - Pide la escala de los ejes X en una gráfica.

Y-AXIS. - Pide la escala de los ejes Y en una gráfica.

AUTOMATIC. - La computadora realiza el cálculo de los ejes automáticamente.

MANUAL. - Se ingresa la escala de manera manual.

QUIT. - Salir del menú.

COLOR. - La gráfica se desplegará en color en monitor de color.

B&N. - La gráfica se desplegará en blanco y negro en cualquier monitor.

DATA LABELS. - Etiquetar puntos de la gráfica en las Hojas Electrónicas.

QUIT. - Salir del menú de OPCIONES.

NAME. - Asigna nombre a definiciones gráficas. Guardando la Hoja Electrónica después de haberse nombrado.

USE. - Indica que se usará el mismo nombre de la gráfica.

CREATE. - Permite ingresar la opción de un nuevo nombre.

DELETE. - Borra un nombre de una gráfica.

RESET. - Borra todos los nombres de las gráficas.

#### COMO DEFINIR UNA GRAFICA

Para definir una gráfica se utiliza el comando /Graph, el cual especifica aspectos y da información esencial para la

creación de dicha gráfica como son:

- 1.- Que tipo de gráfica se desea dibujar: Línea, Barra, Barra en forma de pila, Pastel (Pie), XY, etc.
- 2.- El rango de las celdas en los cuales están los datos a dibujar. Hay que notar que la definición de una gráfica envuelve la localidad, no los valores contenidos en estas celdas.
- 3.- Etiquetas gráficas. El Quattro incluye la facilidad de adicionar etiquetas, títulos y leyendas en las gráficas.

Todas las demás especificaciones que nos proporciona el menú de Graph son opcionales.

En cualquier punto de la definición de la gráfica es posible verla con tan solo seleccionar /Graph View. Si el Quattro tiene suficiente información, dibujará la gráfica.

#### GRAFICANDO

El camino más rápido para dibujar una gráfica es el siguiente:

- 1.- Seleccionar el comando /Graph; y a continuación aparecerá el siguiente submenú de gráficas:

```
TYPE X A B C D E F RESET VIEW SAVE OPTIONS NAME
GRAPH-CUSTOMIZE PRINT QUIT
```

- 2.- Seleccione TYPE. Aquí se podrá elegir entre los diferentes tipos de gráficas que maneja Quattro.
- 3.- Después de la elección de tipo de gráfica se escoge según se vaya necesitando de entre los rangos (X, A, B, C, D, E o F) para dar los valores de los datos.
- 4.- Una vez hechas estas especificaciones vuelve a aparecer el menú principal de gráficas, en donde podemos seleccionar la opción VIEW para poder ver la gráfica en pantalla.
- 5.- En caso de que se desee guardar o salvar nuestra gráfica se hace uso del comando Save, pudiendo dar el nombre que queramos a nuestra gráfica.

#### MÚLTIPLES CONJUNTOS DE DATOS

La representación de muchos conjuntos de datos en la misma gráfica es muy sencilla. El Quattro puede mostrar hasta 6 rangos a la vez -en adición al rango A-. De los valores de datos se pueden especificar los rangos B, C, D, E Y F.  
Ejemplo:

```
El rango B va de B5.E5
El rango C va de B6.E6
El rango D va de B7.E7
Etc.
```

Ahora solo con presionar View veremos una gráfica con todos los valores de los rangos. Este tipo de graficación es muy útil por ejemplo si se tuviera para un mismo mes 6 diferentes tipos de cosechas, y se quisiera visualizar todo el año y todas las cosechas a la vez, ya que en este caso aparecerían las 6 gráficas correspondientes a las diferentes cosechas agrupadas y conformando su respectivo mes).

#### MACROS DEL TECLADO

El Quattro tiene la facilidad de permitir almacenar secuencias de teclas para uso futuro durante la misma sesión del Quattro o para uso futuro. Las teclas pueden llamar uno o más comandos de formateo, pedir entradas, generar gráficas, etc.

Una secuencia de teclazos es comúnmente llamada MACROS DE TECLADO, de manera abreviada MACRO.

Podemos llamar a cada macro con una de las letras del abecedario: A, B, C, ..., Z.

#### UN EJEMPLO SENCILLO

Supongamos que una Hoja Electrónica del Quattro provee los datos de muchos tipos de reportes. Para cada reporte, queremos ajustar el ancho de las columnas A y B, en orden para producir el mejor efecto visual.

Antes de mandar a imprimir el reporte se deben de ingresar los comandos apropiados:

Anotaremos las teclas que vamos presionando conforme se realiza de manera manual la ejecución de nuestras órdenes:

TECLAS	RESULTADO
F5 A1 RETURN	Mover el apuntador a A1
^WCS25 RETURN	Use el comando ^Worksheet Column-Width Set para colocar el ancho de la columna A a 25 caracteres
--->	Mover el cursor a la celda de la derecha, a la celda B1
^WCS12~	Usar el comando ^Worksheet Column-Width Set de la columna B a 12 caracteres

#### CREACIÓN DE UNA MACRO

-Podemos ingresar la macro siguiente, en la celda X1 y grabar todos estos teclazos como una entrada de etiqueta:

'(GOTO)A1~/WCS25~(RIGHT)/WCS12~

-Indicamos el nombre de la macro con una letra presionando la secuencia de teclas:

<Range Name Create  
Enter Name: C RETURN  
Enter Range: X1 RETURN

Esto coloca a la secuencia de teclas que comienzan en la celda X1 como macro C.

-Utilización de la macro: Solo es necesario presionar la tecla <Alt> y sin soltar esta tecla presionar la tecla C, el Quattro instantáneamente efectúa la secuencia de teclas almacenadas en la celda X1.

#### CREANDO Y NOMBRANDO UNA MACRO

Como sugerencia para teclear una macro podemos mencionar como pasos a realizar los siguientes:

a) Teclear entradas y/o comandos, notando exactamente que son las mismas teclas que presionaría si estuviera de manera manual, no olvidándose ningún RETURN representado por el caracter TILDE ~.

b) Ingresar las etiquetas de la macro, esto es precederlas de LABEL-PREFIX '.

Algunas secuencias de las teclas llaman a comandos dentro de la Hoja Electrónica o indican que se va a ingresar números o fórmulas, es necesario definir esta secuencia con un prefijo indicando que es una etiqueta, es decir principiar con los caracteres ' ^ ', en las siguiente figura se muestran 3 etiquetas que pueden actuar como secuencias de caracteres:

(1) +(up)+(up)(up)+(up)(up)(up)~

(2) /WCS10~(RIGTH)/WCS10~(RIGTH)/WCS10~(LEFT)(LEFT)

(3) ASOSORAMIENTO Y SOLUCIONES DE INFORMÁTICA~

La macro (1) ingresa una fórmula que suma las 3 celdas arriba de la celda en la que se encontraba el cursor. Para ingresar esta macro como etiqueta, se debe de comenzar con un apóstrofo (') u otro prefijo de etiqueta.

Se debe de teclear:

'+(up)+(up)(up)+(up)(up)(up)~

Aparecerá en la celda:

+(up)+(up)(up)+(up)(up)(up)~

La regla general es:

Lo que se vea en las celdas de la Hoja Electrónica será lo que se realice cuando se ejecute esta MACRO.

La Macro (2) ejecuta varias veces el comando /Worksheet Column-Width Set varias veces. Esta macro hace 3 columnas consecutivas a 10 caracteres de ancho y regresa el apuntador de celdas a su celda original, para ingresarse se debe de iniciar con apóstrofe u otro indicador de etiquetas " o '.

La Macro (3) Únicamente ingresa una etiqueta a la celda en la que se encuentra el cursor.

Los ejemplos que a continuación aparecen ilustran como indicar teclas especiales en una macro:

-Las teclas con múltiples caracteres son indicadas encerrándolas en llaves, por ejemplo: (HOME), (ESC), etc.

-El movimiento del cursor es de manera similar, por ejemplo: (LEFT), (RIGHT), (UP), (DOWN), ETC (Las letras mayúsculas o minúsculas equivalen a lo mismo).

-La tecla RETURN dentro de una macro se representa con el caracter TILDE ~.

#### LISTA COMPLETA DE INDICADORES DE TECLAS ESPECIALES USADOS EN MACROS

~	Tecla <Return>
(UP)	Mover el cursor hacia arriba
(DOWN)	Mover el cursor hacia abajo
(RIGHT)	Mover el cursor hacia la derecha
(LEFT)	Mover el cursor hacia la izquierda
(HOME)	Tecla de HOME
(END)	Tecla de END
(PGDN)	Tecla de PgDn
(ESC)	Tecla de ESC
(DEL)	Tecla de DEL
(EDIT)	<F2>
(NAME)	<F3>
(ABS)	<F4>
(GOTO)	<F5>
(WINDOW)	<F6>
(QUERY)	<F7>
(TABLE)	<F8>
(CALC)	<F9>
(GRAPH)	<F10>
(#)	Pausa para entrada manual hasta que se ingrese la tecla <Return>

-Asignar un nombre de rango especial a la macro en el cual indique la celda donde comienza la secuencia de teclas.

-Este procedimiento es usado por el comando /Range Name, asignando el nombre de la macro precedido de un backslash (\) y después la letra donde comenzó la secuencia de teclas. Por ejemplo:

\A \B \C ... \Z

## MACROS DE AUTO-EJECUCION

Una característica especial es que podemos asignar una macro adicional con el comando

`/Range Name Create`

llamada

`0`

la macro `0` (cero). Esta macro especial indica que una macro es auto-ejecutable.

Si una Hoja Electrónica creada incluye una macro llamada `0`, el Quattro automáticamente comenzará a ejecutar esa macro cuando se cargue una Hoja Electrónica con un

`/File Retrieve`

El Quattro incorpora una característica adicional que abre el rango de posibilidades. En lugar de presionar varias veces teclas que ya se almacenaron previamente, Quattro puede realizar decisiones y seguir ejecutando teclas de lo que se hará posteriormente. Estas decisiones pueden ser basadas en valores de la Hoja Electrónica o en acciones tomadas durante la ejecución de una macro.

Por ejemplo se puede tener una macro que:

-Se dirija a diferentes localidades de la hoja electrónica, es decir un (GOTO) de un lenguaje de alto nivel.

Imprimir una lista de correos utilizando la información de una base de datos, deteniéndose automáticamente cuando se imprima el último registro.

Imprimir un menú que se defina en el panel de control, permitiendo al usuario seleccionar alguna opción de éste. La macro toma entonces acción basándose en la elección del usuario.

## LOS COMANDOS /X

Existe un conjunto de comandos extra dentro del Quattro. los comandos `/X`, estos comandos son invisibles, es decir, nuestro paquete no los reconocerá si se teclan de forma inmediata. Sólo son utilizables estos comandos dentro de una macro.

Esta facilidad permite la capacidad de programación dentro de la Hoja Electrónica.

Los comandos `/X` a diferencia de otros comandos del Quattro en su camino fundamental afectan las entradas de la Hoja Electrónica.

## LISTA DE COMANDOS /X

CONDICION	FUNCION
/Xlcondición~...	IF-THEN
/XGlocalidad~	GOTO
/XClocalidad~	Llamado a subrutina
/XR	Regreso de subrutina
/XQ	Salir de la ejecución de una macro

## COMANDOS DE ENTRADA DEL TECLADO

/XMlocalidad~	Proceso para definir un menú
/Xlmensaje~localidad~	Mostrar un mensaje en el panel de control, aceptar una entrada alfanumérica (/XL) o una entrada numérica (/XN) del teclado y coloca esa entrada en la celda marcada
/XNmensaje~localidad~	

## LOS COMANDOS /X DE PROGRAMACION

Estos comandos son una herramienta básica en un lenguaje de programación, pues son las transferencias de control incondicional 'IF', condicional 'GOTO', llamados a subrutinas 'CALL', regresos de subrutinas 'RETURN', etc.

Para variables en las celdas de la Hoja Electrónica, se sugiere dar a los rangos nombres que tengan algún significado, como CONTADOR, NUMERO, COLUMNA, ETC.

Ahora bien, al tener:

```
/Xlcondición~...
```

si la condición es verdadera (TRUE), es decir el valor de la condición no sea cero, se continúan leyendo los teclazos en la misma celda. Si la condición es falsa (FALSE), cero, se continúan leyendo los teclazos de las celdas de abajo. Es necesario que después de haber colocado la condición en este comando se coloque el caracter tilde.

Ejemplo:

```
/Xlcontador<=10~..
```

Cuando se tiene:

```
/XGlocalidad~
```

Lo que se hace es mandar la ejecución de la macro a la localidad especificada. Se puede especificar esta localidad como una dirección de celda, un rango, o un nombre de rango. Aquí no que hace Cuatro es que abandona la celda donde se encuentra esta instrucción y continúa leyendo la macro en la celda especificada (o en el caso de un rango empieza en el rango



señalado en la esquina superior izquierda del mismo).

**PRECAUCION.** - No hay que confundir esta instrucción de programación de 'goto' con la tecla funcional F5 (GOTO). El comando /XG solamente determina a donde el Quattro tiene que seguir ejecutando la Macro.

Si tenemos

/Xlocalidad y /XR

vemos que es similar a la instrucción /XG (GOTO), pues continúa leyendo la macro en la localidad especificada. Cuando un comando /XR es encontrado, regresa a leer la macro en la instrucción siguiente de donde fue encontrada la la instrucción /Xlocalidad. Las subrutinas pueden ser anidadas.

Cuando se encuentra con la instrucción

/XQ

da por terminada la ejecución de la macro.

#### GRABACION AUTOMATICA DE UNA MACRO

El Quattro posee una ventaja muy grande, y esta es la grabación automática de una macro, la cual se realiza oprimiendo al mismo tiempo las teclas <FB> y <ALT> (modo RECORD), en cuyo caso se nos solicita el nombre con que deseamos reconocerla. Una vez teclado dicho nombre (con un RETURN al final) se comienzan a grabar todos los pasos que vamos realizando en la selección de las diferentes opciones de los menús para realizar determinada acción; pudiendo de esta manera tener grabada una macro que nos lleve por este camino sin necesidad de tener que hacerlo manualmente. Para dar por terminada nuestra macro lo que se debe de hacer es oprimir nuevamente las teclas <ALT> y <FB> al mismo tiempo.

## CAPITULO II

### METODOLOGIA

Para la elaboración de la presente tesis me he basado en dos métodos o estrategias: La Programación Estructurada y El Diseño Modular (descendente o top-down) de Sistemas; es por ello que a continuación aparece una breve descripción de lo que se necesita conocer para entender dichas metodologías, además de mencionar algunas otras existentes.

#### II.1 ANTECEDENTES

En sus inicios, la programación de los sistemas de cómputo estuvo determinada por el trabajo individual de los programadores; es decir no había un grupo que se coordinara en la tarea del desarrollo de los sistemas, ni una forma sistemática de estudiar la complejidad de los mismos.

Como consecuencia de ello, el usuario, al hacer uso del sistema tenían una serie de dificultades que, en gran medida, se podían atribuir a que la programación del sistema no cumplía con los requerimientos.

Por otra parte, cuando un programador codificaba individualmente sin atenderse a un método específico, ocasionaba que otros programadores tuvieran dificultad para implementar cambios sustanciales al sistema, tales como reprogramar módulos incorrectos, integrar otras funciones al sistema y cualquier otra especificación para que este fuera actualizado.

Esto, naturalmente incrementaba los costos de mantenimiento, además de que un sistema de estas características ocasionaba grandes retrasos. Asimismo, los programadores no aplicaban una metodología en el desarrollo de la programación de los sistemas que permitiera obtener un producto que cumpliera con ciertas normas de calidad. Esto se debía a que el programador aplicaba la mayor parte de su esfuerzo a la programación y prueba del sistema y desatendía por completo las fases de análisis y diseño, además de que olvidaba que un programa también es un documento para comunicar la solución de un problema.

Resultaba así, que mientras el equipo de cómputo progresaba grandemente y su costo disminuía, la eficiencia de la programación se incrementaba con la complejidad del sistema y, como consecuencia, sus costos se incrementaban. Esta situación llegó a agravarse tanto que dió origen a lo que se llamó crisis del software o de la programación; pero esto motivó la búsqueda de una solución a los problemas, con el objeto de:

- Abatir costos de mantenimiento.
- Obtener alta confiabilidad.
- Entregar a tiempo.
- Lograr una administración efectiva.
- Otros.

Con este fin nació la programación estructurada o sistemática, así como el diseño modular (descendente o top-down) de

sistemas.

## II.2 FASES EN EL DESARROLLO DE PROGRAMAS

Todo programa, como conducto final de un proceso de manufacturas, emerge como consecuencia de una serie de pasos o fases que deben seguirse rigurosamente. Se identifican las cuatro fases siguientes:

### II.2.1 ALGORITMO

En esta fase se lleva a cabo el análisis del problema y el desarrollo de la solución. Estas dos actividades forman una estrecha simbiosis. Durante esta fase se hace un análisis cuidadoso del problema propuesto con el objeto de lograr una representación mental (abstracción) de los elementos esenciales del mismo. Una vez logrado este primer nivel de abstracción se puede continuar subdividiendo estos elementos, siempre y cuando sea posible, en partes más simples de la misma forma hasta llegar a un nivel de gran detalle.

Es evidente que se hace necesario contar con una notación conveniente para ir concretando la solución, ya que mentalmente no se podría retener la gran cantidad de detalles que se originan conforme se refina la solución. A todo este proceso se le conoce como refinamiento a pasos y es central a la programación estructurada y al diseño modular (descendente) de sistemas. Esto no es más que un proceso de ensayo y error. Obsérvese que el problema comienza con un enunciado que dice lo "que" se intenta resolver y se termina diciendo "como" se intenta resolverlo.

### II.2.2 DIAGRAMAS

Existen varios instrumentos de análisis de programación que ayudan a desarrollar soluciones a los problemas. Uno de éstos es el diagrama de bloques de programas, el cual es un diagrama general que representa los pasos que se van a realizar dentro de la máquina a fin de producir las salidas requeridas. Otro de éstos instrumentos es el diagrama de flujo de un programa, el cual proviene del diagrama de bloque de un sistema. En muchos casos, el diagrama de un programa principal (o macro) identifica los principales módulos que es necesario realizar y entonces se utilizan diagramas micro detallados para mostrar los pasos del procesamiento en los módulos especificados. Por lo tanto, los programadores pueden preparar un solo diagrama de programa macro (siendo clasificados así en algunos casos los diagramas de bloques por ciertos autores), y varios diagramas micro durante el análisis de la programación.

### II.2.3 CODIFICACION

De la fase anterior surge un programa abstracto, el cual, en esta tercera fase, se traduce o se implementa en un lenguaje de

programación determinado. El proceso de codificar un programa en un lenguaje de programación es totalmente directo.

#### II.2.4 PRUEBA

Después de la implementación de la solución presentada, es necesario que ésta sea probada, para así asegurar que resolverá correctamente el problema original.

La depuración del programa y la eliminación de errores en la codificación o en la carga de la computadora (vía terminal o cualquier otra), se realiza mediante la "compilación" del programa.

Para la prueba del programa se deberán diseñar conjuntos de datos, en los que se puedan contemplar todos los casos o variaciones posibles en la presentación del problema, principalmente en casos extremos.

#### II.2.5 MANTENIMIENTO

Durante la fase de mantenimiento del programa se realizan las modificaciones que son necesarias cuando cambian algunas características del problema que no implican un cambio en el diseño original del programa. La "vida útil" es el tiempo durante el cual el programa funciona sin que sea modificado sustancialmente; es decir, el tiempo en que es útil para la solución del problema original.

La fase de mantenimiento se mantendrá presente durante toda la vida útil del programa, y es en la que se determina cuándo el programa ha dejado de ser útil.

### II.3 CALIDAD DE LOS PROGRAMAS

La forma tradicional de diseño condujo a programas difíciles para darles mantenimiento (escasa calidad), debido, por una parte a su casi nula estructuración y escasa claridad, y por otra a artificios "inteligentes" de programadores "abusados", artificios que por lo general sólo servían para realizar el ego del programador y no contribuían en nada a hacer del programa un vehículo para comunicar ideas.

Debido a esta situación se hizo patente la necesidad de realizar el diseño de programas de una manera sistemática, logrando con ello programas de mayor calidad. Lograr programas de mayor calidad mediante un diseño sistemático es el objetivo fundamental de la programación estructurada y del diseño modular (descendente) de sistemas.

No hay dos personas que estén de acuerdo sobre lo que es la calidad de programas; sin embargo, algunos criterios que nos dan una idea de lo que se entiende por calidad de programas son los siguientes:

### II.3.1 QUE EL PROGRAMA FUNCIONE

Esto significa que el programa debe cumplir con los requerimientos especificados. Es importante que las especificaciones sean revisadas continuamente a través de todo el proceso de diseño e implementación del programa. Un programa que no hace o hace parcialmente lo que se supone debe hacer, refleja que el diseñador tiene una escasa comprensión de las especificaciones requeridas del programa.

### II.3.2 QUE EL PROGRAMA ESTE LIBRE DE ERRORES

Generalmente los errores de un programa son debidos a que el programador es poco cuidadoso; tiene un entendimiento poco claro de las especificaciones o de los rasgos del lenguaje en que codifica o falla al anticipar una situación particular.

### II.3.3 QUE EL PROGRAMA ESTE BIEN DOCUMENTADO

Tradicionalmente, la eficiencia de un programa se medía en términos de la cantidad de memoria que ocupaba y de la velocidad de ejecución del mismo. Esto se ajustaba cuando las máquinas eran lentas y de poca capacidad de memoria y por consiguiente el programador dedicaba tiempo en reducir el requerimiento que demandaba su programa de estos recursos.

En la actualidad, estos criterios de eficiencia ya no son válidos, excepto en casos muy especiales. Debido a que los costos del hardware han descendido y los costos humanos se han incrementado. De modo que si un programador intenta hacer eficiente un programa bajo el criterio de espacio/velocidad a costa de producir un programa difícil de darle mantenimiento es claro que habrá logrado un producto de baja calidad.

## II.4 PROGRAMACION MODULAR O DISEÑO MODULAR DESCENDENTE (O TOP-DOWN) DE SISTEMAS

El concepto básico de la programación modular es muy simple, consiste en dividir un programa en módulos.

En realidad es un método de diseño que tiende a dividir el problema de forma lógica, en partes permanentemente diferenciadas que pueden ser utilizadas, programadas y puestas a punto independientemente.

La división de un problema en módulos o programas independientes exige otro módulo que controle y relacione a todos los demás, es el denominado módulo base o principal del problema.

Realmente la programación modular es un intento para diseñar programas, de forma tal que cualquier función lógica pueda ser intercambiada sin afectar a otras partes del programa.

En esta estrategia el programador puede escribir primero un programa de control principal que sirva para definir los segmentos principales, o módulos, que se requieren para resolver un problema. El programa de control principal especifica el orden en que se va a procesar cada uno de los módulos subordinados del

programa. La etapa de análisis de programación continúa hasta haber reducido todos los módulos hasta el punto en que el programador esté seguro de tener un módulo de solución que resolverá la tarea. Cuando se emplea esta técnica de diseño modular (descendente) de programas, una instrucción del programa de control principal transfiere el control a una rutina subordinada (subrutina) o módulo del programa. Una vez terminada la operación específica de procesamiento que realiza el módulo, es posible que otra instrucción de ramificación transfiera el control del programa a otro módulo o lo devuelva al programa de control principal. Así pues, los módulos o subrutinas son en realidad programas dentro de un programa. Lo normal es que un módulo tenga un solo punto de entrada y un solo punto de salida.

Algunas de las ventajas de utilizar esta técnica de construcción son:

1.- Es posible dividir programas complejos en elementos más sencillos y manejables.

2.- Varios programas pueden codificar varios módulos al mismo tiempo.

3.- Se puede crear una biblioteca de módulos y utilizar los módulos en otros programas según se requiera.

4.- Probablemente sea más fácil determinar la localización de errores de programa en un módulo específico, lo que simplifica la implementación y el mantenimiento.

## II.5 PROGRAMACION ESTRUCTURADA

Los primitivos propósitos de la programación estructurada dirigen sus esfuerzos a buscar métodos de minimizar la probabilidad de error en el proceso de programación. El factor humano es una de las fuentes más importantes en la incurrancia de errores. Uno de los objetivos de la programación estructurada es la minimización del error humano.

Aunque el profesor Edgar W. Dijkstra de la Universidad de Eindhoven se le considera el padre de la programación estructurada, muchos otros investigadores han trabajado en su desarrollo y de ellos podemos citar a Hoare, Wirth, Knuth, Dahl, Bohm, Jacopini, Warnier, etc.

Es difícil dar una definición de programación estructurada por no existir generalmente una definición que sea aceptada a todos los niveles. Podríamos por ello enunciar la programación estructurada como una técnica de construcción de programas que utilizan al máximo los recursos del lenguaje, limita el conjunto de estructuras aplicables a leer y presenta una serie de reglas que coordinan adecuadamente el desarrollo de las diferentes fases de la programación.

En la programación estructurada se describe el uso disciplinado de las tres estructuras lógicas estándar (estructura de secuencia simple, estructura de selección o *if-then* y estructura de ciclo o *do-while*). Aunque es posible combinar o "anidar" los patrones lógicos en la práctica, poseyendo cada patrón un solo punto de entrada y un solo punto de salida. También pueden leer todos los patrones de arriba hacia abajo. Esta sencillez inherente conduce a un programa con una lógica más comprensible.

A continuación aparecen algunas de las ventajas que nos proporciona el uso de técnicas de programación estructurada:

- 1.- Se pueden utilizar técnicas comprobadas para atacar un problema a fin de que los programadores puedan evitar que los programas sean innecesariamente complejos y confusos.
- 2.- Los programas se pueden descomponer en módulos y asignarse a varias personas para acelerar el proceso de la programación.
- 3.- Ya que presentan sólo una parte de un programa que puede ser extenso y complejo, es posible que estos módulos sean más fáciles de comprender, probar y depurar.
- 4.- La probabilidad de que los errores sobrevivan al proceso de desarrollo es menor y es probable que se acelere la implementación del programa y que las labores posteriores de mantenimiento sean más fáciles y económicas.

## II.5.1 METODOS DE PROGRAMACION ESTRUCTURADA

Existen diferentes métodos de programación estructurada. Entre los más conocidos se encuentran: Jackson, Bertini y Warnier.

### METODO JACKSON

Esta metodología creada por el inglés Michael Jackson se basa en que la estructura de un programa está en función de la estructura de los datos que manipula. Jackson emplea métodos según su orden jerárquico dentro de los diferentes niveles donde se encuentra. Cada módulo es un dato o conjunto de datos.

Las estructuras básicas de este método son las siguientes:

- Secuencial. - Un número determinado de módulos se ejecutan una sola vez en el orden jerárquico preestablecido.
- Repetitiva. - Un módulo se ejecuta desde cero hasta  $n$  veces.
- Alternativa. - Se selecciona para la ejecución un módulo entre varios posibles.

Con estas estructuras básicas se puede obtener cualquier otra que intervenga en el diseño del programa.

El uso del método de Jackson supone la lectura arriba-abajo y de izquierda a derecha.

Los pasos a seguir para resolver un determinado problema siguiendo esta metodología son:

- 1.- Establecer o definir los datos de entrada y salida (resultados).
- 2.- Creación de la estructura del programa a partir de las diferentes estructuras de los datos.
- 3.- Utilizar los recursos que posee el método para conseguir los resultados.
- 4.- Escribir el pseudocódigo y codificar.

METODO BERTINI

La metodología Bertini consiste en la descomposición de un problema en niveles, teniendo cada uno de ellos un inicio, un conjunto de procesos y un fin.

Esta metodología representa la estructura de los programas y no las operaciones del tratamiento. Aquí también se manejan las tres estructuras que en el caso anterior

Según Bertini las instrucciones se ejecutan de derecha a izquierda pero el programador puede leerlo al revés si le resulta más cómodo.

METODO WARNIER

Se basa en una metodología matemática que establece un único lenguaje de comunicación entre usuarios, analistas y programadores, lo que permite la comprensión de forma sencilla por cualquier programador. La representación de cualquier proceso se puede hacer mediante llaves.

El método se basa en la descomposición por niveles del problema. En cada nivel se detallan los tratamientos que permiten la solución del problema planteado.

Las estructuras utilizadas son conceptualmente idénticas a las de Jackson si bien su presentación varía.

Los pasos a seguir en esta metodología son:

- 1.- Estudio de los datos de salida.
- 2.- Estudio de los datos de entrada teniendo en cuenta la organización de los datos en el fichero lógico de salida y analizando las posibles fases de tratamiento.
- 3.- Cuadro sinóptico de descomposición formado a partir de la estructura de los datos de entrada, salida y del tratamiento.
- 4.- Diagrama de flujo y lista de instrucciones. El diagrama se obtiene de forma automática del cuadro sinóptico y su uso puede no ser imprescindible. La lista de instrucciones -su orden de ejecución- produce en casi su totalidad la escritura del programa con independencia del lenguaje.



## CAPITULO III ARCHIVOS Y DIAGRAMAS

### III.1 OBJETIVOS Y ALCANCES DEL SISTEMA

#### III.1.1 OBJETIVOS DEL SISTEMA

1) Obtener de un archivo bastante grande otro, en el cual sólo se encuentre contenida la información que se requiere (cantidad de formatos recibidos y solicitudes para cada entidad, clasificados éstos por su clave respectiva), para poder llevar a cabo la gráfica basada en las elecciones hechas por el usuario (un formato en particular de todas las entidades, un formato en particular de todos los sectores, un formato en particular de todos los subsectores, una entidad en particular con todos los formatos, un sector en particular con todos los formatos o un subsector en particular con todos los formatos).

2) En base al archivo creado se obtiene la gráfica en la que se ve reflejado el porciento de cumplimiento de cada entidad, sector o subsector (lo cual se obtiene en base a la cantidad de formatos que se han recibido respecto a los que les fueron solicitados), teniendo la capacidad de ser creada dicha gráfica de 3 diferentes tipos (líneas, barras o pastel), así como 2 alternativas de salida (por pantalla o por impresora); todo lo cual será a gusto del usuario.

#### III.1.2 ALCANCES DEL SISTEMA

- 1) No pretende capturar información.
- 2) La información que requiere la obtiene de otro sistema independiente completamente del que nos ocupa.
- 3) Selecciona del archivo creado por el sistema del cual se toma la información, la que requiere solamente creando un nuevo archivo resultado del "filtrado" del primero.
- 4) En base a los datos contenidos en el archivo producto del filtrado y a las elecciones hechas por el usuario se obtiene la gráfica.
- 5) Da a elegir entre 3 tipos diferentes de gráficas -líneas, barras y pastel- y 2 salidas -impresora y pantalla- para poder visualizar de la manera más adecuada el porciento de cumplimiento de las entidades, sectores o subsectores.

## III.2 ARCHIVOS

### III.2.1 ARCHIVOS CONTENIDOS EN EL SISTEMA

#### INSTA.BAS

Este es un programa elaborado en turbo basic, el cual tiene como finalidad instalar el Sistema en el disco duro; creando a su vez el "SISGRAF.BAT" en el directorio raíz.

#### INSTA.EXE

Este es el programa ejecutable del "INSTA.BAS".

#### SISGRAF.BAT

Es un archivo batch (por lote), el cual es creado en el directorio raíz por el "INSTA.EXE" cuando se instala el sistema en el disco duro.

#### SISGRAF.PRG

Este programa es el encargado de que se imprima en pantalla de forma adecuada el menú de opciones de nuestro Sistema, así como de que en cada submenú se elija una sola alternativa, guardándose posteriormente dicha información las variables: menú, opción, gráfica y salida.

Tomando como base el submenú en el que se está desplazando el usuario, este programa envueta a que se continúe el flujo del programa según corresponda a "PRIN1.PRG", a "PRIN2.PRG" o a "PRIN3.PRG".

#### SISTGRAF

Este es el programa ejecutable del "SISTGRAF.PRG" y para poder obtenerlo como tal, lo que se hizo fue compilar con CLIPPER (CLIPPER.EXE) el programa fuente (sistgraf), pudiendonos percatar de que en este punto también se toman todos los demás programas que son requeridos por "SISGRAF.PRG" para trabajar adecuadamente; obteniendo así el programa objeto, el cual a su vez fue ligado por medio del LINK (LINK.EXE del Sistema Operativo) a la librería DBU.LIB de Clipper; obteniendo de esta manera finalmente nuestro programa ejecutable "SISTGRAF.EXE".

(3) Los archivos con extensión bat en su nombre se denominan archivos por lote (batch) y no contienen un programa como los archivos .com o .exe, sino que son listas de comandos de DOS que se deben ejecutar en secuencia. Mientras que los archivos .com y .exe están en formato binario (y parecen conjuntos incomprensibles de dígitos), los archivos por lote se pueden listar en la pantalla o en la impresora y tienen un aspecto de archivos ordinarios de texto.

COENTL.DBF

En esta base de datos está contenido el catálogo de entidades.

CAFORM.DBF

Esta es una base de datos en la cual se encuentra almacenado el catálogo de formatos, así como sus respectivas claves.

CABSEC.DBF

En esta base de datos está contenido el catálogo de sectores y sus respectivas claves.

CASUBS.DBF

Esta es una base de datos en la cual se encuentran almacenados tanto el catálogo de subsectores como las claves que le corresponden a cada uno.

CONTRACLI.DBF

En esta base de datos tenemos toda la información recabada por el Módulo de Control de Formatos del Sistema Integral de Información con respecto a las claves del sector, subsector, entidad y formato, así como el número de formatos recibidos y solicitados.

VARIABLE.DBF

En este archivo se guarda la estructura con la cual se van a almacenar los siguientes datos: subtítulo de la gráfica, nombre del eje x, tipo de gráfica y la salida por la que elegimos se graficará.

Este a su vez se copia en el archivo "TEMPOR.TXT", el cual contendrá tanto la estructura como la información, habiendo tomado esta primera del archivo "VARIABLE.DBF".

PRINL.PRG

Permite el desplazamiento del usuario a lo largo del primer submenú del programa que se visualiza en pantalla, enviando posteriormente el flujo del programa al "SUB1.PRG".

PRIND.PRG

Permite que por medio de las flechas el usuario se desplace a todo lo largo del segundo submenú que aparece en pantalla, encausando finalmente el flujo del programa al "SUB1.PRG".

Por medio de este programa el usuario tiene la capacidad de desplazarse en el tercer menú que aparece en pantalla, para que finalmente mande el flujo del programa al "SUBI.PRG".

#### TECLA:ESC

Aquí se verifica que lo teclado por el usuario sea correcto, esto es, que corresponda el código a los caracteres de flechas, return o esc; encausando por medio de estos dos últimos el flujo del programa hacia donde corresponda.

#### SUBI.PRG

En este programa se lleva a cabo el filtrado de la información que se va a graficar, es decir, nosotros tenemos la base de datos "CONTROL1.DBF", la cual contiene toda la información del Sistema Integral de Información y en "SUBI.PRG" en base a todas las elecciones hechas por el usuario (al visualizar las diferentes pantallas que este programa va creando), se toma y ordena (indexa) la información requerida, grabándola a su vez en los archivos temporales ("TEMPO.TXT", "TEMPO2.TXT" y "TEMPO.CAT").

#### MACRO:G1

Como su nombre lo dice es una macro, la cual primeramente fue creada en Lotus 1-2-3 y posteriormente trasladada y completada en Quattro para hacer posible la graficación en papel; dicha macro contiene las órdenes necesarias para que se grafique la información que hemos obtenido de la selección del usuario, la cual se encuentra contenida en "TEMPO2.TXT" (valores de las variables: subtítulo, título del eje y, tipo de gráfica y salida) "TEMPO.CAT" (valores del eje x) y "TEMPO.TXT" (archivo que se deriva del "TEMPO.DBF" y cuyo contenido al igual que el de "TEMPO.TXT" es la selección de los datos a graficar, sólo que en el .DBF está contenida dicha información en forma de base de datos, en tanto que en .TXT se tiene en forma de texto, esto debido a que de esta manera se nos facilita la graficación en Quattro).

- (4) Los archivos indexados tienen la particularidad de que sus registros están organizados empleando como base un índice, lo cual agiliza el acceso a dichos registros, ya que para realizar una localización emplea éste índice en vez de comenzar la búsqueda desde el principio del archivo.

### 111.2.2 INSTALACION DEL SISTEMA

A continuación se da una breve descripción de la forma en que se instala y se trabaja el Sistema en cuanto a la ubicación de los archivos dentro de los directorios, visualizando así de una manera más palpable como es que se desarrolla esto.

Estando en el directorio principal del disco flexible que contiene el Sistema corremos el programa "INSTALA.EXE", el cual nos pedirá que le especifiquemos la unidad y el nombre del directorio donde se instalará el Sistema, así como la unidad y el nombre del directorio donde se encuentra la base de datos, la cual se presupone está ubicada dentro del mismo directorio de Dbase; y finalmente nos pide le digamos la unidad y el nombre del directorio donde se encuentra Quattro. Si todos los datos proporcionados son correctos el programa instalará automáticamente el Sistema.

El procedimiento que se lleva a cabo en cuanto a la manipulación de los archivos se describe brevemente a continuación:

Es generado en la unidad y en el directorio especificados para el la ubicación del Sistema, un archivo por lotes llamado "SISGRAF.BAT"; el cual tendrá el control absoluto de la manipulación de los archivos y de los cambios de directorio.

A continuación el archivo "INSTALA.BAS" translada del subdirectorio `lib` (que se encuentra en el disco del Sistema) los archivos "SISGRAF.EXE" y "VARIABLE.DBF" a la unidad y directorio especificados para la base de datos. También es creado en el mismo directorio un archivo llamado "QUATIRO.DIR", el cual es utilizado por la macro cuando requiere cambiar su directorio de trabajo de Dase a Quattro, lugar donde se localiza la el paquete Quattro y la macro. Es en "QUATIRO.DIR" donde se encuentran grabados los datos necesarios para efectuar dicho cambio como son la unidad y directorio donde se encuentra ubicado el directorio Quattro. Así mismo translada del directorio de `lib` (del disco del Sistema) el archivo "MACRO.DAQ" al directorio de "QUATIRO" en disco duro, creando en este mismo directorio el archivo "DBASE.DIR", que también es un archivo de datos que utiliza la macro para cambiar su directorio de trabajo de Quattro a Dbase, lugar donde se localizan todos los archivos temporales, los cuales contienen la información que es requerido por la macro para graficar ya sea por pantalla o por impresora. Es en "DBASE.DIR" donde se encuentran grabados los datos necesarios para efectuar dicho cambio como son la unidad y directorio donde se encuentra ubicado el directorio destino que en este caso es en el que se encuentre la base de datos.

Al finalizar, nuestro programa "INSTALA" nos posiciona en el directorio en donde fue instalado inicialmente "SISGRAF" y dentro de éste para que de esa manera podamos llevar a cabo una nueva elección de graficación.

### III.2.3 SISGRAF.BAT

En el momento que entramos a "SISGRAF" éste se encarga de cambiarnos al directorio donde se encuentran las bases de datos CIBICED y pide el programa "SISGRAF"; con el cual, al correrlo se obtienen todas las características de la gráfica, las cuales se graban al igual que la información seleccionada en los archivos temporales "TEMPO.TXT", "TEMPO2.TXT" y "TEMPO.CAT".

Si se detecta la existencia del primero se efectúa la rutina de Quattro, de lo contrario se regresa al directorio donde se le instaló el Sistema, dándose por terminada la sesión.

Al trabajar con la rutina de Quattro, esta cambia el flujo del programa al directorio de Quattro, a la vez que renombra el archivo llamado "MACRO.WKD" por "QUATTRO.WKD"; dicho archivo es una hoja de cálculo, la cual se carga automáticamente al correr Quattro, que es el paso siguiente.

Ahora bien, dentro de dicha hoja de cálculo existe una macro cero que a su vez, como es sabido corre automáticamente, la cual cambia el directorio de trabajo del Quattro al directorio donde se encuentra la base de datos, leyendo de la hoja de cálculo los archivos temporales antes mencionados y generando la gráfica en base a ellos, ya sea por pantalla o por impresora.

Al salir de Quattro la hoja de cálculo regresa a su nombre original y se realiza el cambio a la unidad y al directorio donde tenemos las bases de datos, borrando ahí todos los archivos temporales que existan y dirigiendo el flujo del programa de manera que se entre nuevamente al menú de "SISGRAF", para que así estemos capacitados para solicitar un nuevo proceso.

## III.2.4 SISTGRAF.PRG

A continuación aparece una descripción de como funciona el presente programa, tomando como guía el listado que aparece en la correspondiente parte de esta Tesis.

Lo primero que se hace es: Se limpia la pantalla y se anula la posibilidad que tiene la computadora de mandar mensajes por ella. Se deshabilita la tecla esc que existe en el teclado. Se limpian y cierran todos los archivos y se ordena que la barra de estatus no se encienda. Se manda que no se desplieguen los letreros que nos piden confirmación alguna y a su vez se confirma la sobrescritura de archivos. Se hace publica la variable "tecla" y se les asignan valores a las variables "opcion1", "opcion2", "opcion3", "opcion4", "opcion5", "opcion6", "opcion01", "opcion02", "opcion03", "opcion04", "opcion05", "opcion06", "grafica1", "grafica2", "grafica3", "salida1" y "salida2". Se le asignan 79 espacios en blanco a la variable "línea", esto para que cuando se necesite borrar una línea completa lo único que se haga sea invocarla. Se inicializan con uno las variables "menu", "opcion", "grafica" y "salida" y se indica que los colores a ocupar por la pantalla serán los asignados desde un principio.

Después de todo esto se dan las especificaciones necesarias para la creación del menú en pantalla conteniendo: en la parte superior los tres submenús; en la parte media la opción, el tipo de gráfica y la salida, las cuales se van modificando según sea nuestra posición o nuestra elección en cada submenú; y en la parte inferior las indicaciones para el desplazamiento y elección que se desea hacer; así como los recuadros dobles y sencillos que se especifican, aunado esto a los colores de pantalla que fueron seleccionados.

En el momento de terminar con todo lo antes mencionado se entra en un "do while" y se pregunta si el valor de la variable "menu" es 1, 2 ó 3, mandando a ejecutar el programa "PRIN1", "PRIN2" ó "PRIN3" respectivamente según sea el caso.

Finalmente si se tiene que "tecla" vale 5 (enter) se hace un salto fuera del "do while"; en caso contrario se continúa en éste indefinidamente.

(5) Para la creación de los recuadros que aparecen en la parte inferior de la pantalla del menú fue necesario valerse de la repetición de algunos caracteres ASCII.

### III.2.5 PRINI.PRG

A continuación se da una explicación del funcionamiento del programa "PRINI.PRG" siguiendo la secuencia del listado que aparece en la sección correspondiente de la presente Tesis.

El presente programa se inicia con la escritura de la palabra opciones, así como un recuadro doble que la enmarque, todo esto en pantalla; acción seguida se ejecuta el programa "TECLA" y se define el color con el que vamos a trabajar en este momento, que será letra negra (o sin color) sobre fondo blanco.

En caso de que se tenga un valor de tecla igual a 1 ó 2 (esto significa que el usuario oprimió ya sea la flecha hacia abajo o ya sea la flecha hacia arriba, respectivamente) se realizará lo siguiente:

-Se regresará a la pantalla el color que tenía cuando iniciamos(6).

-Se ejecuta la siguiente fórmula, auxiliándonos de la tabla que aparece posteriormente, la cual nos ayudará a comprender de una mejor manera dicha fórmula:

#### 5. + OPCION : 2

Esta aparece en el listado en varias ocasiones y como podremos apreciar tiene la finalidad de posicionarnos en un renglón determinado de nuestro submenú de opciones que se visualiza en la pantalla en el menú del Sistema, habiendo obtenido con anterioridad el valor de la variable "opcion" (dependiendo de en cuál de las seis opciones existentes estamos parados en este momento):

OPCION	FORMULA	REGLON
1	6 + 1 + 1 + 1 + 1 = 10	8
2	6 + 1 + 1 + 1 + 1 = 10	10
3	6 + 1 + 1 + 1 + 1 = 10	12
4	6 + 1 + 1 + 1 + 1 = 10	14
5	6 + 1 + 1 + 1 + 1 = 10	16
6	6 + 1 + 1 + 1 + 1 = 10	18

Es pertinente aclarar que el seis que aparece como constante en la fórmula se debe a que existen en la parte superior de la pantalla seis renglones ocupados con el título, el recuadro y los renglones en blanco que se necesitan.

- (d) Para hacer más entendible el concepto del cambio del color en la pantalla se enuncia lo siguiente: cuando aparece un símbolo \* significa que se indicó parpadear, si aparece el signo ^ significa que el color será intenso, las letras nos dan los colores (v.blanco, azul, gr.verde, nrisn color o negro, etc.), si se tiene un \* significa letra blanca sobre fondo negro y si se tiene solamente la ^ indicará que tiene el mismo fondo de antes.



Ahora bien, se puede observar que se hace referencia a "OP2", y, para poder entender su funcionamiento aparece la siguiente explicación: al hacer referencia a "OP2" de lo que se está hablando es de lo que está dentro del contenido de esta variable, es decir, OP2 puede contener cualquiera de las siguientes 6 variables: opcion1, opcion2, opcion3, opcion4, opcion5 u opcion6; y a su vez cada una de ellas contendrá respectivamente una de las siguientes: Formato/ENTIDADES, Formato/SECTORES, Formato/SUBSECTORES, Entidad/FORMATOS, Sector/FORMATOS y Subsector/FORMATOS; y como podemos ver lo que en realidad se necesita es escribir cualquiera de las últimas 6 especificaciones, pero para poder conseguirlo sin tener que alargar mucho el presente programa fue necesario trabajarlas como ya se mencionó.

Si se encuentra que la variable "tecla" es igual a uno incrementamos en una unidad la variable "opcion" para que de esa manera el cursor avance una línea hacia abajo en nuestro menú; pero si se encuentra con que "opcion" vale siete, entonces le asigna el valor de uno para que nuestro cursor aparezca en la parte superior de nuestro submenú.

Si tenemos el caso de que la variable "tecla" es igual a dos, decrementamos en una unidad el valor de la variable "opcion" para el cursor que se encuentra posicionado en un renglón del primer submenú de nuestro programa ascienda un renglón; en caso de que la variable "opcion" tuviera el valor de cero se le asignará el valor de seis, para que de esa manera de la impresión de que se le dió la vuelta al submenú al hacer nuestra aparición en el último renglón.

Se manda que el color de la pantalla sea letras negras sobre fondo blanco.

Las líneas que dicen:

```
OPCION=SUBSECTOR+ENTIDAD+OPCION1 TO OP
```

```
OPCION=OPCION+SUBSECTOR+OPCION2 TO OP2
```

```
STORE "OPCION" OF TO OP3
```

se interpretan de la siguiente manera: En cuanto a la primera diremos que almacena del valor de opcion más cien (esto para evitar que se pierdan algunos registros, ya que al tomar el valor numérico de su clave y al sumarlo al de otro campo pueden dar el mismo resultado las sumas, por ejemplo 1+4 y 2+3, en cuyo caso el programa elegirá la primera y descartará la otra, dando esto como resultado una elección incorrecta pues son campos diferentes y por lo tanto se deben tomar ambos; se convierten en alfanuméricos, los caracteres del noveno en adelante y se asigna la variable "op" (por ejemplo puede quedar op="02"). En la segunda línea lo que se hace es almacenar del valor de "op" convertido en alfanumérico los caracteres del segundo en adelante a la variable "op2", anteponiéndole la palabra opcion (continuando con la secuencia del ejemplo de la primera línea tendremos el siguiente resultado op2="opcion"+"2"=opcion2). En la tercera línea lo que sucede es que a la variable "op3" se le asigna el valor de "op" antecedido por la palabra opcion (y siguiendo con nuestro ejemplo tenemos que op3="opcion"+"02"=opcion02).

Todo esto se hizo para poder obtener de la opcion1 a la opcion4 y de la opcion5 a la opcion6 complementando así

variables subindexadas), ya que se necesitan ambas dependiendo del mensaje de que se trate (el del submenú o el del primer recuadro de los que aparecen en forma horizontal), esto debido a que ambos tienen diferente longitud.

Posteriormente se imprime en el lugar que corresponda los contenidos existentes dentro de las variables contenidas en op3 y op2.

En caso de que el valor de "tecla" sea 3 ó 4 se realizará lo siguiente:

La pantalla empezará trabajando con los colores que de antemano tiene asignados y escribirá la palabra opciones, así como un recuadro doble que enmarque dicha palabra, todo esto en el lugar que se le ha especificado.

Se cambia la especificación de los colores a letras negras sobre fondo blanco.

Si se tiene que el valor de "tecla" es igual a 3 (flecha hacia la derecha) se da el valor de 2 a la variable "menu" (para que se realice un cambio al segundo submenú). Pero si en vez de ser 3 es 4 (flecha hacia la izquierda) el valor de "tecla", entonces la variable menu valdrá 3 (para que se realice un cambio al tercer submenú).

Y finalmente en el caso de que la variable "tecla" valga 5 (enter) se ejecuta el programa "SUB1".

### III.2.6 PRIN2.PRG

Este programa es muy parecido al programa "PRIN1.PRG", de hecho tiene la misma mecánica, lo único que cambia es que aquí el submenú sobre el que estamos trabajando es el segundo (selección de tipo de gráfica) en vez del primero (selección de tipo de opción).

A continuación aparece una breve explicación del funcionamiento del programa "PRIN2.PRG" siguiendo la secuencia del listado que aparece en la sección correspondiente de la presente Tesis.

Dicho programa se inicia con la escritura las palabras "tipo de gráfica", así como un recuadro doble que las enmarque, todo esto en pantalla; acción seguida se ejecuta el programa "TECLA" y se define el color con el que vamos a trabajar en este momento, que será letra negra (o sin color) sobre fondo blanco.

En caso de que se tenga un valor de tecla igual a 1 ó 2 (esto significa que el usuario oprimió ya sea la flecha hacia abajo o ya sea la flecha hacia arriba, respectivamente) se realizará lo siguiente:

Se regresará a la pantalla el color que tenía cuando iniciamos.

La tabla que a continuación aparece nos ayudará a comprender de una mejor manera la siguiente fórmula:

$$6 + \text{GRAFICA} * 2$$

La cual aparece en el listado en varias ocasiones y como podremos apreciar tiene la finalidad de posicionarnos en un renglón determinado de nuestro submenú de tipo de gráfica que se visualiza en la pantalla en el menú del Sistema, habiendo obtenido con anterioridad el valor de la variable "grafica" (dependiendo de en cuál de las tres opciones existentes estamos posicionados en este momento):

GRAFICA	FORMULA	RENGLON
1	= 6 + ( 1 * 2 ) =	8
2	=> 6 + ( 2 * 2 ) =	10
3	=> 6 + ( 3 * 2 ) =	12

Es pertinente aclarar que el seis que aparece como constante en la fórmula se debe a que existen en la parte superior de la pantalla seis renglones ocupados con el título, el recuadro y los renglones en blanco que se necesitan.

Ahora bien, se puede observar que se hace referencia a "&GR2", y que al hacer dicha referencia de lo que se está hablando es de lo que contiene lo que esta dentro de esta variable, es decir, &GR2 puede poseer cualquiera de las siguientes 3 variables: grafica1, grafica2 o grafica3; y a su vez cada una de ellas contendrá respectivamente una de las siguientes LINEAS, BARRAS y PASTEL ; y como se puede ver lo que en realidad se necesita es escribir cualquiera de las últimas 3 especificaciones, pero para poder conseguirlo sin tener que alargar demasiado el presente programa fue necesario trabajarlas como ya se mencionó.

Si se encuentra que la variable "tecla" es igual a uno incrementamos en una unidad la variable "grafica" para que de esa manera el cursor avance una línea hacia abajo en nuestro submenú; pero si se encuentra con que "grafica" vale cuatro, entonces le asigna el valor de uno para que nuestro cursor aparezca en la parte superior de nuestro submenú.

Si tenemos el caso de que la variable "grafica" es igual a dos, decrementamos en una unidad el valor de la variable "grafica" para que el cursor que se encuentra posicionado en un renglón del segundo submenú de nuestro programa ascienda un renglón; en caso de que la variable "grafica" tuviera el valor de cero se le asignará el valor de tres, para que de esa manera de la impresión de que se le dió la vuelta al submenú al hacer nuestra aparición en el último renglón.

Se manda que el color de la pantalla sea letras negras sobre fondo blanco.

Las líneas que dicen:

```
STORE SUBSTR(STR(GRAFICA),10) TO GR1
```

```
STORE "GRAFICA"+GR TO GR2
```

se interpretan de la siguiente manera: En cuanto a la primera línea lo que se hace es almacenar del valor de "grafica" convertido en alfanumérico los caracteres del décimo en adelante a la variable "gr". (por ejemplo gr=1) En la segunda línea lo que sucede es que a la variable "gr2" se le asigna el valor de "gr" antecedida por la palabra grafica (y continuando con nuestro ejemplo tenemos que gr2="grafica"+ "1"="grafica1").

Posteriormente se imprime en el lugar que corresponda el contenido existente dentro de la variable contenida en gr2.

En caso de que el valor de "tecla" sea 3 ó 4 se realizará lo siguiente:

La pantalla empezará trabajando con los colores que de antemano tiene asignados y escribirá las palabras tipo de gráfica, así como un recuadro doble que las enmarque, todo esto en el lugar que se le ha especificado.

Se cambia la especificación de los colores a letras negras sobre fondo blanco.

Si se tiene que el valor de "tecla" es igual a 3 (flecha hacia la derecha) se da el valor de 3 a la variable "menu" (para que se realice un cambio al tercer submenú). Pero si en vez de ser 3 es 4 (flecha hacia la izquierda) el valor de "tecla", entonces la variable menú valdrá 1 (para que se realice un cambio al primer submenú).

Y finalmente en el caso de que la variable "tecla" valga 5 (enter) se ejecuta el programa "SUB1".

## III.2.7 PRIN3.PRG

Este programa es muy parecido a los programas "PRIN1.PRG" y "PRIN2.PRG", de hecho tiene la misma mecánica, lo único que cambia es que aquí el submenú sobre el que estamos trabajando es el tercero (elección de tipo de salida) en vez del primero (elección de tipo de opción) o del segundo (elección de tipo de gráfica), respectivamente.

A continuación aparece una breve explicación del funcionamiento del programa "PRIN3.PRG" siguiendo la secuencia del listado que aparece en la sección correspondiente de la presente Tesis.

Dicho programa se inicia con la escritura la palabra "salida", así como un recuadro doble que las enmarque, todo esto en pantalla; acción seguida se ejecuta el programa "TECLA" y se define el color con el que vamos a trabajar en este momento, que será letra negra (o sin color) sobre fondo blanco.

En caso de que se tenga un valor de tecla igual a 1 ó 2 (esto significa que el usuario oprimió la flecha hacia abajo o la flecha hacia arriba, respectivamente) se realizará lo siguiente:

Se regresará a la pantalla el color que tenía cuando iniciamos.

La tabla que a continuación aparece nos ayudará a comprender de una mejor manera la siguiente fórmula:

$$7 + \text{SALIDA} * 2$$

La cual aparece en el listado en varias ocasiones y como podremos apreciar tiene la finalidad de posicionarnos en un renglón determinado de nuestro submenú de tipo de salida que se visualiza en la pantalla en el submenú del Sistema, habiendo obtenido con anterioridad el valor de la variable "salida" (dependiendo de en cuál de las dos opciones existentes estamos posicionados en este momento):

SALIDA	FORMULA	REGLON
1	=: 7 + ( 1 * 2 ) =	9
2	=: 7 + ( 2 * 2 ) =	11

Es pertinente aclarar que el siete que aparece como constante en la fórmula se debe a que existen en la parte superior de la pantalla siete renglones ocupados con el título, el recuadro y los renglones en blanco que se necesitan.

Ahora bien, se puede observar que se hace referencia a "SA2", y que al hacer dicha referencia de lo que se está hablando es de lo que contiene lo que está dentro de esta variable, es decir, SA2 puede poseer cualquiera de las siguientes 2 variables: salida1 o salida2; y a su vez cada una de ellas contendrá respectivamente una de las siguientes PANTALLA e IMPRESORA; y como se puede ver lo que en realidad se necesita es escribir cualquiera de las últimas 2 especificaciones, pero para poder conseguirlo sin tener que alargar demasiado el presente programa fue necesario

trabajarlas como ya se mencionó.

Se le asigna a la variable "salida" un valor igual a 3 menos el valor que contiene actualmente dicha variable, para que de esta manera siempre nos estemos posicionando en el único otro renglón que existe en nuestro submenú, no importando si lo que se tecleó fue la flecha hacia abajo o hacia arriba.

Se manda que el color de la pantalla sea letras negras sobre fondo blanco.

Las líneas que dicen:

```
STORE SUBSTR(STR(SALIDA),10) TO SA
```

```
STORE "SALIDA"+SA TO SA2
```

se interpretan de la siguiente manera: En cuanto a la primera línea lo que se hace es almacenar del valor de "salida" convertido en alfanumérico los caracteres del décimo en adelante a la variable "sa". (por ejemplo sa=1). En la segunda línea lo que sucede es que a la variable "sa2" se le asigna el valor de "sa" antecedido por la palabra salida (continuando con nuestro ejemplo tenemos que sa2="salida"+ "1"=salida!).

Posteriormente se imprime en el lugar que corresponda el contenido existente dentro de la variable contenida en sa2.

En caso de que el valor de "tecla" sea 3 ó 4 se realizará lo siguiente:

La pantalla empezará trabajando con los colores que de antemano tiene asignados y escribirá la palabra salida, así como un recuadro doble que la enmarque, todo esto en el lugar que se le ha especificado.

Se cambia la especificación de los colores a letras negras sobre fondo blanco.

Si se tiene que el valor de "tecla" es igual a 3 (flecha hacia la derecha) se da el valor de 1 a la variable "menu" (para que se realice un cambio al primer submenú). Pero si en vez de ser 3 es 4 (flecha hacia la izquierda) el valor de "tecla", entonces la variable menú valdrá 2 (para que se realice un cambio al segundo submenú).

Y finalmente en el caso de que la variable "tecla" valga 5 (enter) se ejecuta el programa "SUB1".

### III.2.8 TECLA.PRG

A continuación aparece una breve explicación del funcionamiento de este programa siguiendo la secuencia del listado que aparece en la sección correspondiente de la presente Tesis.

Se le da el valor de cero a la variable "tecla" y se dice que mientras persista dicho valor tome como suyo cualquiera que se le de por teclado; después de lo cual se empieza a analizar al valor que tiene almacenado "tecla" en este momento (esto después de que le ha teclado algún valor el usuario).

En el caso de que "tecla" valga 24 (código de control de la tecla del cursor que indica flecha hacia abajo) o 50 (código ascii del teclado numérico de la tecla con el número dos), se le cambia dicho valor por 1.

En el caso de que "tecla" valga 5 (código de control de la tecla del cursor que indica flecha hacia arriba) o 56 (código ascii del teclado numérico de la tecla con el número ocho), se le cambia dicho valor por 2.

En el caso de que "tecla" valga 4 (código de control de la tecla del cursor que indica flecha hacia la derecha) o 54 (código ascii del teclado numérico de la tecla con el número seis), se le cambia dicho valor por 3.

En el caso de que "tecla" valga 19 (código de control de la tecla del cursor que indica flecha hacia la izquierda) o 52 (código ascii del teclado numérico de la tecla con el número cuatro), se le cambia dicho valor por 4.

En el caso de que "tecla" valga 13 (código de control de la tecla enter) se le cambia dicho valor por 5.

En el caso de que "tecla" valga 27 (código de control de la tecla esc) se regresa al color de default de pantalla que tiene Dbase y se limpia ésta, dejando habilitada nuevamente la tecla esc del teclado.

En caso de que el usuario del Sistema pulse cualquier otra tecla se escuchará un bip, habilitado éste por el renglón que dice: ? CHR(7).

### III.2.0 SUB1.PRG

A continuación aparece una descripción de como funciona el presente programa, tomando como guía el listado que aparece en la correspondiente parte de esta Tesis.

Se declaran como públicas las variables "z" y "clave", ya que estos serán ocupados a lo largo del Sistema en diferentes sitios.

En caso de que la variable "opción" sea igual a uno, dos o tres se usará el catálogo de formatos (caform), indicando esto que se va a elegir un formato y que en base a él vamos a filtrar la información, pues de él es de quien se necesita toda la información. Posteriormente se les asigna un contenido a determinadas variables de la siguiente manera: a "titulo" se le da "selección de formato", a "mensaje" "< > y < > para escoger formato", a "campo1" "forclav", a "campo2" "nombre" y a "selección" "formato":.

En caso de que la variable "opción" sea igual a cuatro se usará el catálogo de entidades (caent), indicando esto que se va a elegir una entidad y que en base a ella vamos a filtrar la información, pues es de ella de quien se necesita toda la información. Posteriormente se les asigna un contenido a las siguientes variables: a "titulo" se le da "selección de entidad", a "mensaje" "< > y < > para escoger entidad", a "campo1" "claent", a "campo2" "nom\_ent" y a "selección" "entidad":.

En caso de que la variable "opción" sea igual a cinco se usará el catálogo de sectores (casec), indicando esto que se va a elegir un sector y que en base a él vamos a filtrar la información, pues es de él de quien se necesita toda la información. Posteriormente se les asigna un contenido a las siguientes variables: a "titulo" se le da "selección de sector", a "mensaje" "< > y < > para escoger sector", a "campo1" "clasec", a "campo2" "nom\_sector" y a "selección" "sector":.

En caso de que la variable "opción" sea igual a seis se usará el catálogo de subsectores (casubs), indicando esto que se va a elegir un subsector y que en base a él vamos a filtrar la información, pues es de él de quien se necesita toda la información. Posteriormente se les asigna un contenido a las siguientes variables: a "titulo" se le da "selección de subsector", a "mensaje" "< > y < > para escoger subsector", a "campo1" "clasubs", a "campo2" "nom\_subs" y a "selección" "subsector":.

Se le da la orden a la pantalla para que maneje los colores que de antemano tenía asignados.

Ahora se procede a la creación de la segunda pantalla, para lo cual nos vamos de las variables "línea", "titulo" y "mensaje", añadidas a la creación de recuadros y letreros, acompañados por los cambios de color en la pantalla cuando se requiere.

Con la orden "go top" el puntero de la pantalla se va hasta donde se inicia esta (inicio de archivo).

Se inicializa la variable "x" con un valor igual a 3, esto para que ahí sea donde se inicialice la cuenta de posición en pantalla, ya que los primeros dos renglones están ocupados (el primero con el valor de la variable "titulo" y el segundo con una línea).

Mediante el ciclo "do while" se imprime en la segunda pantalla perteneciente al Sistema todo el catálogo que corresponde según sea el caso, al contenido final existente



dentro de %campo1 y %campo2.

Se cierra el ciclo "do while" entrando posteriormente a otro, el cual hace lo siguiente:

Se va al principio del archivo y se reconoce el número de renglón de dicho registro (RECNO()) incrementándole en 2 unidades el número de renglón para posicionarnos ahí con un valor de 4 en cuanto a el número de la columna, para poder imprimir en este sitio el valor contenido en la variable que está dentro de %CAMPO1. Posteriormente se hace lo mismo pero ahora ubicándose en la columna 14 e imprimiendo el valor contenido en la variable que está dentro de %CAMPO2.

Dentro de otro ciclo "do while" anidado se inicializa "z" con cero recibiendo en dicha variable lo que se manda desde el teclado. Se ordena que el color de pantalla sea el mismo que se tenía asignado desde el principio, y se entra en un nuevo ciclo "do while", el cual sirve para validar o no lo teclado; si se tecléo cualquiera de las flechas o la tecla enter se sale fuera del este último ciclo, si se oprimió la tecla esc (z=27) lo que se realiza en realidad es un return para volver al menú anterior y en caso de que se oprima cualquier otra tecla se dejará escuchar un bip (^CHR(7)). Aquí termina este "do while" anidado.

Se va al principio del archivo y se reconoce el número de renglón de dicho registro (RECNO()) incrementándole en 2 unidades el número de renglón para posicionarnos ahí con un valor de 4 en cuanto a el número de la columna, para poder imprimir en este sitio el valor contenido en la variable que está dentro de %CAMPO1. Posteriormente se hace lo mismo pero ahora ubicándose en la columna 14 e imprimiendo el valor contenido en la variable que se encuentra dentro de %CAMPO2.

En caso de que "z" sea igual a 24 o a 50 (flecha hacia abajo) se realiza un brinco hacia el renglón de abajo en el catálogo que se tiene en pantalla, y en el caso que sea fin de archivo se posiciona en el renglón superior de dicho catálogo.

En caso de que "z" sea igual a 5 o a 56 (flecha hacia arriba) se realiza un brinco hacia el renglón de arriba en el catálogo que se tiene en pantalla, y en el caso que sea principio de archivo se posiciona en el último registro (renglón) de dicho catálogo. Y en cualquier otro caso se almacena en las variables "clave" y "nombre" los contenidos de las variables que están dentro de %CAMPO1 y %CAMPO2, respectivamente; cerrando todas las bases de datos para terminar así con el ciclo "do while".

Se crea la tercera pantalla en la cual aparecen tres recuadros: en el primero está el nombre del Sistema, en el segundo dos letreros (los cuales dicen "espere un momento" y "procesando información") y el tercero en el cual se pueden visualizar todas las elecciones que se han hecho y por tanto la información que se está procesando.

Tomando en cuenta la salida que se eligió se decide si cambiar a la cuarta pantalla o no, ya que en ésta, la única variante con respecto a la tercera es que pide se prepare la impresora. Direccionalándose la salida a pantalla (set device to screen) o a impresora (set device to printer).

Abriéndose la base de datos "CONTROL1" se pregunta por la variable "opcion".

Si ésta es igual a uno (un formato en específico contra todas las entidades) se indexa la base de datos en base a "cliente" y se graba en "TEMPO", reescribiendo en este archivo solamente los datos que resulten pertenecer a la clave del formato seleccionado

Cforclave=clave, siendo "forclave" el nombre del campo dentro de "CONTROLI" que contiene las claves de los formatos, y "clave" es el nombre de la variable que contiene la clave del formato seleccionado en nuestra segunda pantalla. Usando la base de datos "CAENTI" (catálogo de entidades), se procede a copiar en el archivo "TEMPO.CAT" todos los campos que existen en "claenti" (claves de las entidades) en forma delimitada (deli) o entrecomillada para poder leerlo de una forma más sencilla desde Quattro; es decir, se tiene "TEMPO.DBF" y de él se copia el contenido de "solicitado" y "recibido" al archivo "TEMPO.TXT".

Si "opcion" vale dos (un formato en específico contra todos los sectores) solamente se manejará un registro para cada llave ("clasec" clase de sector) del indexado y se grabará en la base de datos "TEMPO", la cual contendrá únicamente 7 registros, los cuales pertenecen a cada uno de los 7 sectores que existen. Después de lo cual, se abre nuevamente "TEMPO" y se procede a limpiar los campos "solicitado" y "recibido", ya que pueden contener valores iniciales diferentes de cero, cosa que no sería muy conveniente, pues en el momento de llevar a cabo la suma que más tarde se efectúa esta no resultaría correcta. Posteriormente se le asigna el valor de "clave" a "forclave", esto para garantizar que se tiene la clave del formato correcta. Se abre "CONTROLI" y se manda el puntero al primer registro de esta base de datos. Se hace una reasignación de variables para poder realizar la suma y entrando a "TEMPO" se sacan los totales de "recibido" y "solicitado" cuando estos tienen el mismo sector ("clasec"="clase"). Finalmente usando la base de datos "CASEC" (catálogo de sectores), se procede a copiar en el archivo "TEMPO.CAT" todos los campos que existen en "clasec" (claves de los sectores) en forma delimitada (deli) o entrecomillada para poder leerlo de una forma más sencilla desde Quattro; es decir, se tiene "TEMPO.DBF" y de él se copia el contenido de "solicitado" y "recibido" al archivo "TEMPO.TXT".

Si "opcion" vale tres (un formato en específico contra todos los subsectores) solamente se manejará un registro para cada llave del indexado que en este caso será la suma de "clasec" por 10 y de "clasubs", ya que ahora debemos tener en cuenta ambas por la forma en que están organizadas las bases de datos y siendo necesario el multiplicar por 10 la primera pues de no hacerlo así el "unique" omite ciertos registros cuya suma daba como resultado la misma cantidad ( $9+2=10+1$ ); grabándose todo esto en la base de datos "TEMPO", la cual contendrá únicamente 13 registros, los cuales pertenecen a cada uno de los 13 subsectores que existen. Después de lo cual, se abre nuevamente "TEMPO" y se procede a limpiar los campos "solicitado" y "recibido", ya que pueden contener valores iniciales diferentes de cero, cosa que no sería muy conveniente, pues en el momento de llevar a cabo la suma que más tarde se efectúa esta no resultaría correcta. Posteriormente se le asigna el valor de "clave" a "forclave", esto para garantizar que se tiene la clave del formato correcta. Se abre "CONTROLI" y se manda el puntero al primer registro de esta base de datos. Se hace una reasignación de variables para poder realizar la suma y entrando a "TEMPO" se sacan los totales de "recibido" y "solicitado" cuando estos tienen el mismo subsector ("clasec"="clase" y "clasubs"="ssector"). Finalmente usando la base de datos "CASUBS" (catálogo de subsectores), se procede a copiar en el archivo "TEMPO.CAT" todos los campos que existen en "clasubs" (claves de los subsectores) en forma delimitada (deli)

o entrecomillada para poder leerlo de una forma más sencilla desde Quattro; es decir, se tiene "TEMPO.DBF" y de él se copia el contenido de "solicitado" y "recibido" al archivo "TEMPO.TXT".

Si "opcion" vale cuatro (una entidad en específico contra todos los formatos) se copian en "TEMPO" los registros cuya "clasi" (clave de entidad, cuyo valor está dado en forma numérica) sea igual al valor numérico de "clave" (que inicialmente es alfanumérico). Se abre "caform" (catálogo de formatos) y se copia en "TEMPO.CAT" todos los campos que existen en "forclav" (claves de los formatos) en forma delimitada (deli) o entrecomillada para poder leerlo de una forma más sencilla desde Quattro; es decir, se tiene "TEMPO.DBF" y de él se copia el contenido de "solicitado" y "recibido" al archivo "TEMPO.TXT".

Si "opcion" vale cinco (un sector en específico contra todos los formatos) solamente se manejará un registro para cada clave ("forclav" clave de formato) del indexado y se grabará en la base de datos "TEMPO", cerrando posteriormente el "index" y anulando el "unique". Después de lo cual, se abre nuevamente "TEMPO" y se procede a limpiar los campos "solicitado" y "recibido", ya que pueden contener valores iniciales diferentes de cero, cosa que no sería muy conveniente, pues en el momento de llevar a cabo la suma que más tarde se efectúa, ésta no resultaría correcta. Posteriormente se manda el puntero al primer registro de la base de datos "CONTROL1" y se hace una reasignación de variables si se cumple que "clasec" (clave de sector cuyo contenido es numérico) y el valor numérico de "clave" (cuyo contenido inicialmente es alfanumérico) son iguales, dicha reasignación se lleva a cabo para poder realizar la suma de los registros "solicitado" y "recibido". Se entra a "TEMPO" y se sacan dichos totales siempre y cuando cuando los registros cumplan con la siguiente condición: que su valor de "forclav" (clave de formato) y de "clave" sean iguales. Finalmente usando la base de datos "CAFORM" (catálogo de formatos), se procede a copiar en el archivo "TEMPO.CAT" todos los campos que existen en "forclav" (claves de los formatos) en forma delimitada (deli) o entrecomillada para poder leerlo de una forma más sencilla desde Quattro; es decir, se tiene "TEMPO.DBF" y de él se copia el contenido de "solicitado" y "recibido" al archivo "TEMPO.TXT".

Si "opcion" vale seis (un subsector en específico contra todos los formatos) solamente se manejará un registro para cada clave ("forclav" clave de formato) del indexado y se grabará en la base de datos "TEMPO", cerrando posteriormente el "index" y anulando el "unique". Después de lo cual, se abre nuevamente "TEMPO" y se procede a limpiar los campos "solicitado" y "recibido". Posteriormente se manda el puntero al primer registro de la base de datos "CONTROL1" y se hace una reasignación de variables si se cumple que "clasec" (clave de sector cuyo contenido es numérico) y el valor numérico de "clave" (cuyo contenido inicialmente es alfanumérico) son iguales, y además de que "clasubs" (cuyo valor es numérico) sea igual al resultado que se entrega si se almacena del valor "clave" convertido a alfanumérico los caracteres del cuarto en adelante; dicha reasignación se lleva a cabo para poder efectuar la suma de los registros "solicitado" y "recibido". Se entra a "TEMPO" y se sacan dichos totales siempre y cuando cuando los registros cumplan con la siguiente condición: que el valor de "forclav" (clave de formato) y de "clave" sean iguales. Finalmente usando la base de datos "CAFORM" (catálogo de formatos), se procede a copiar en el

archivo "TEMPO.CAT" todos los campos que existen en "forclav" (claves de los formatos) en forma delimitada (deli) o entrecomillada para poder leerlo de una forma más sencilla desde Quattro; es decir, se tiene "TEMPO.DBF" y de él se copia el contenido de "solicitado" y "recibido" al archivo "TEMPO.TXT".

Al terminar el "do case" se cierran todos los archivos y las bases de datos, para que en la próxima línea se abra "TEMPO" nuevamente.

Si se tiene que "opcion" es mayor de tres (esto significa que se van a graficar todos los formatos contra una entidad, sector o subsector en específico) se manda indexar "TEMPO" basándose en el valor numérico que entrega "forclav" a partir del tercer carácter; todo esto para un indexado correcto pues si tenemos que "forclav"=E-1 por ejemplo, nuestro archivo se indexará tomando en cuenta el "1".

Para sacar el porcentaje de cumplimiento de las entidades se necesita hacer la siguiente operación "recibido/solicitado x 100" y como es de esperarse habrá ocasiones en las que no se haya solicitado formato alguno, es por ello que en este caso se tendrá que solicitado es igual a cero, pero como no se permiten realizar divisiones entre cero fue necesario hacer lo siguiente: primero, cuando se encuentre que solicitado=0 se reemplaza recibido=1 (ya que por lógica si solicitado vale cero recibido también) y segundo, cuando se encuentra que solicitado=0 se reemplaza con solicitado=1; de esta forma la operación si se puede realizar y no se altera en lo más mínimo el resultado.

En la siguiente línea lo que se hace es se tienen "TEMPO.DBF" y de él se copia el contenido de "solicitado" y "recibido" al archivo "TEMPO.TXT", para que de esa manera se puedan importar más fácilmente dichos datos con el Quattro.

Se manda a imprimir la quinta pantalla, la cual tiene únicamente en el centro con letras negras y fondo blanco brillante el mensaje que dice 'espere un momento', la cual estará presente durante todo el tiempo restante hasta que entremos a Quattro.

Ahora se comienza a trabajar con el archivo "VARIABLE" (no debe perderse de vista que dicho archivo contiene la estructura que se necesita para la buena integración de los datos que a continuación se manejan), en el cual lo primero que se hace es reemplazar el contenido de "subst" con el contenido de "nombre"; y dependiendo del valor de "opcion", de "grafica" y de "salida" se almacenan los valores del eje x y del subtítulo, así como el camino para graficar en sus tres variantes y elegir la salida, estos dos últimos necesarios para Quattro.

Y para terminar el programa "SUBI.PRG" lo que se hace es guardar los valores que se obtuvieron (subtítulo, eje x, gráfica y salida) en el archivo temporal "TEMPO2", que será con el que se trabaje desde Quattro cuando se requieran dichos datos.

### III.2.10 MACRO. \$KQ

Esta es la macro de Quattro que se encuentra dentro del directorio de QUATTRO y, en la cual al ver su listado se puede uno percatar de que lo que a continuación se narra es la descripción de lo que ella realiza siguiendo la secuencia que aparece en dicho listado, pudiendo ser localizada en la sección correspondiente de la presente Tesis, que como es sabido contiene todas las órdenes que en otro caso se necesitarían dar manualmente para que se pudiera obtener la gráfica ya sea en pantalla o en papel.

- Cambia el directorio de trabajo actual del de QUATTRO al de DBASE, valiéndose para ello del "path" o camino contenido en el archivo "DBASE.DIR". Todo esto para poder obtener los datos que necesita la macro, y que se encuentran contenidos en los archivos temporales localizados en este último directorio.
- Define las coordenadas del punto1.
- Va a la celda AE1 e importa en este sitio lo contenido en el archivo "TEMPO.TXT", que vienen a ser los datos de la cantidad de formatos recibidos y solicitados.
- Va a la celda AD1 y ahí importa el contenido del archivo "TEMPO.CAT", el cual posee la información correspondiente a las claves.
- Va a la celda BA1 e importa en este sitio el archivo "TEMPO2.TXT", el cual contiene los parámetros referentes a el título de la gráfica, el nombre del eje x y las especificaciones en cuanto a el tipo de gráfica.
- Calcula la relación del primer dato, es decir realiza la división de la cantidad de formatos recibidos entre los solicitados.
- Copia la misma relación a las celdas siguientes hasta terminar con toda la columna.
- Define el formato de porcentaje para el rango de datos especificado.
- Se ubica en la primera celda del rango.
- Inicializa gráfica y se le asigna al rango A como valores los porcentajes.
- Se ubica en la primera celda y se le asigna al rango X como valores el de las claves.
- Define el formato en porcentaje del eje Y.
- Define los títulos generales.
- Translada el contenido de las celdas y define los títulos variables.
- Translada y opera la celda con la definición de tipo de gráfica.
- Cambia el directorio de trabajo del de DBASE al de QUATTRO, para poder llevar a cabo la graficación, todo esto valiéndose del "path" o camino que está contenido en el archivo "QUATTRO.DIR", que como ya dijimos a su vez se encuentra dentro del directorio de DBASE.
- Translada y opera la celda con la definición de salida.
- Si así se pidió se visualiza en pantalla la gráfica.
- Se llevan a cabo las órdenes necesarias para poder imprimir la gráfica en caso de que así se hubiera solicitado.
- Sale de Quattro.

## III.3 DIAGRAMAS

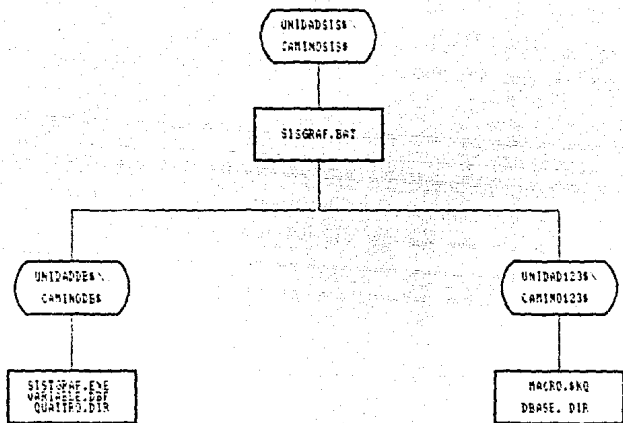
### III.3.1 DIAGRAMA DE LA ORGANIZACION DEL DISCO QUE CONTIENE EL SISTEMA.

NOTA: Los archivos que aparecen debajo de las líneas punteadas no se necesitan incluir en el disco para que el sistema funcione, ya que al tener los dos archivos ejecutables, éstos los incluyen (el instala.exe al instala.bas y el sistgraf.exe a los .prg).

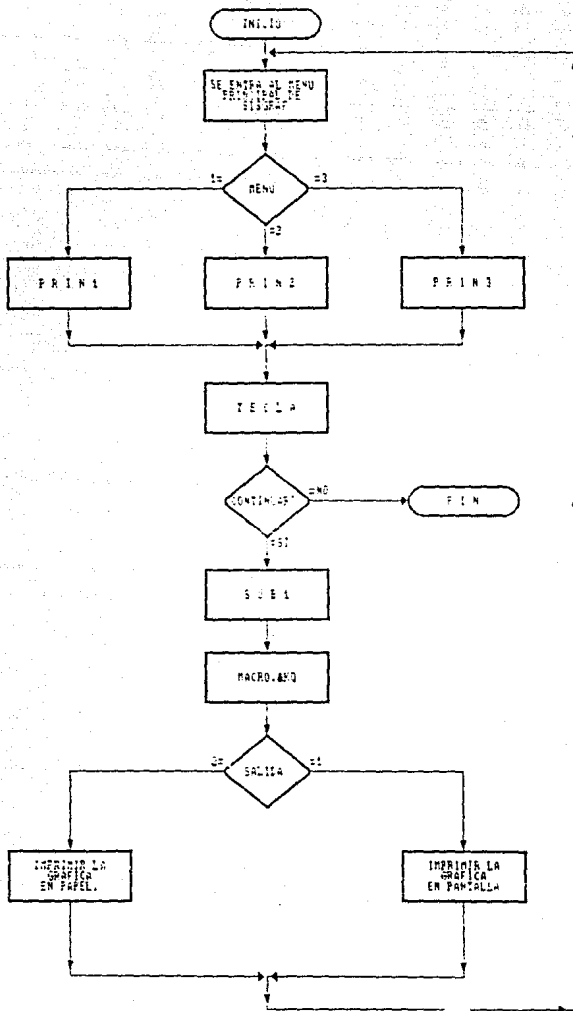




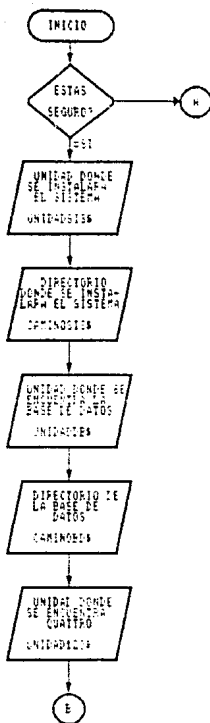
**III.3.2 DIAGRAMA DE LA ORGANIZACION  
DE LOS ARCHIVOS DENTRO DEL DISCO DURO.**

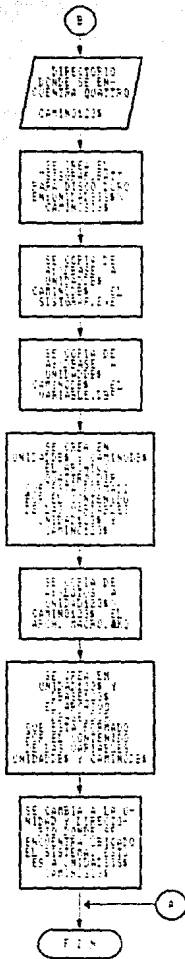


### III.3.3 DIAGRAMA GENERAL DEL SISTEMA



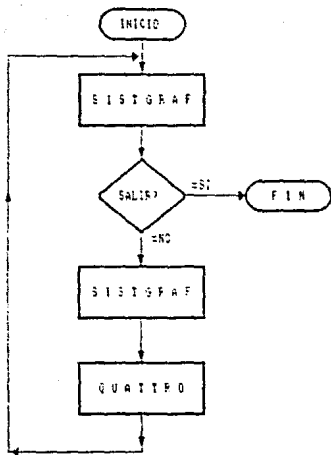
**III.34 INSTALASI**



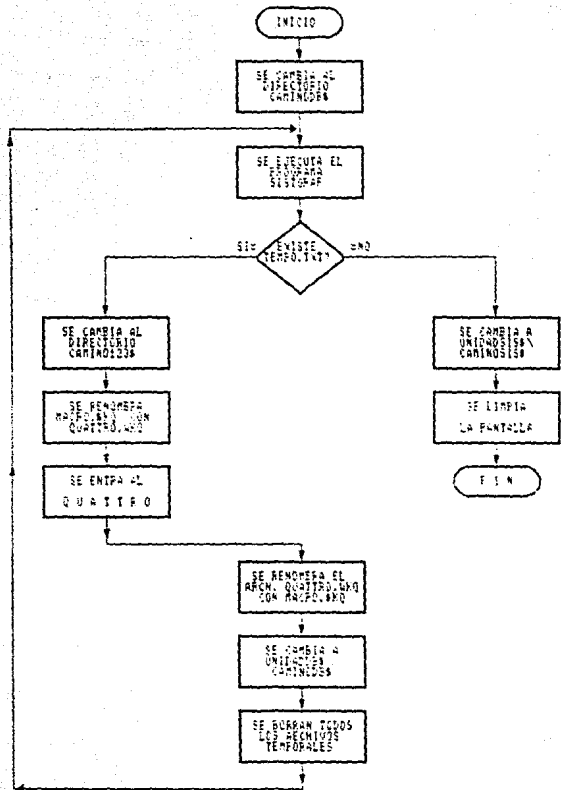


### III.3.5 DIAGRAMA GENERAL DEL SISGRAF.BAT

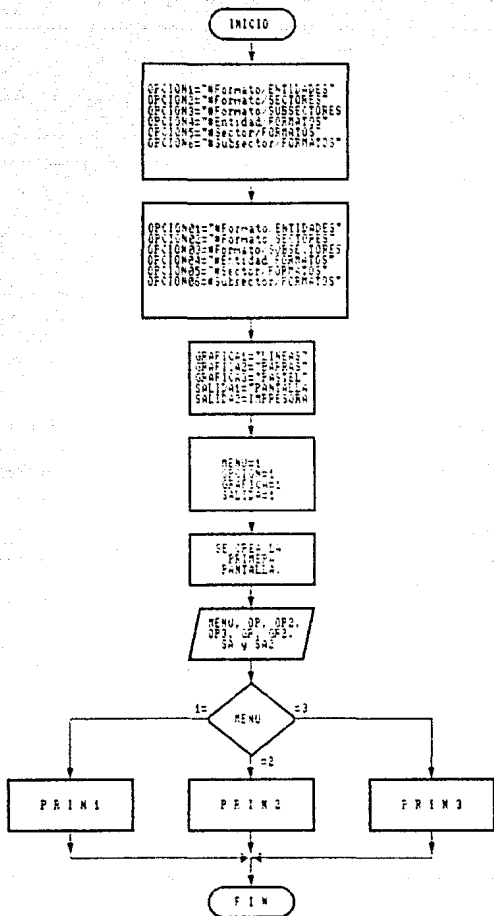




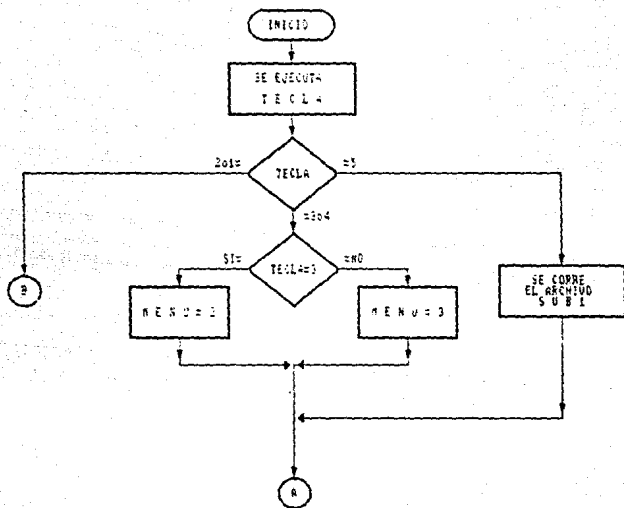
**III.36 SIGRAF.BAT**

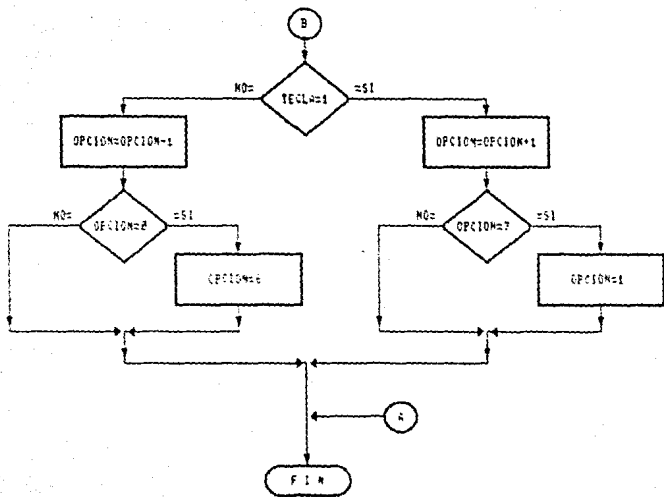


III.37 SISTGRAF.PRG



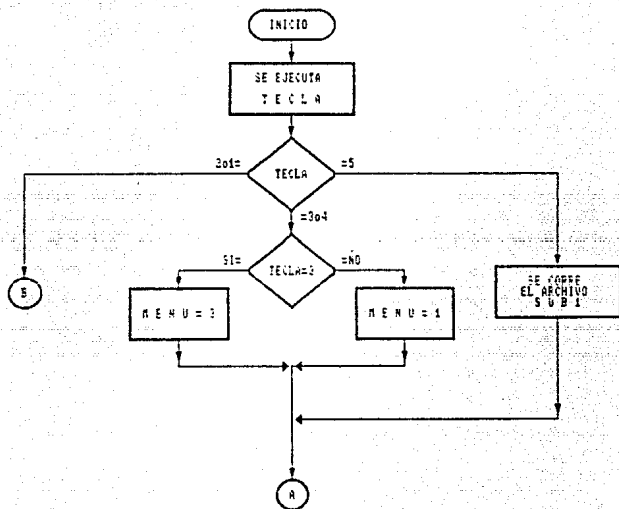
III38 PRIN1.PRG

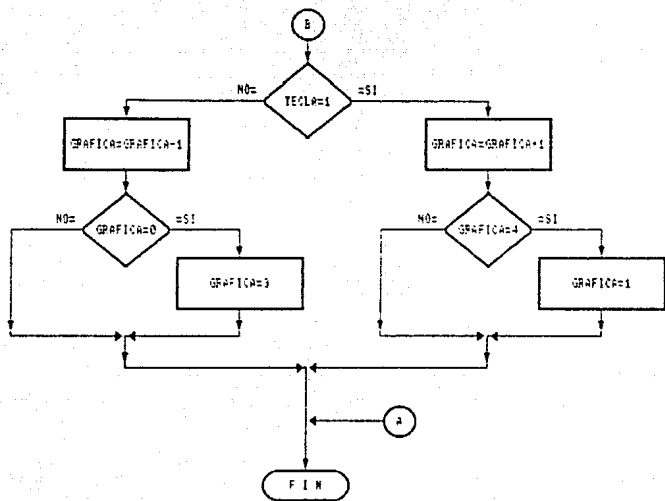




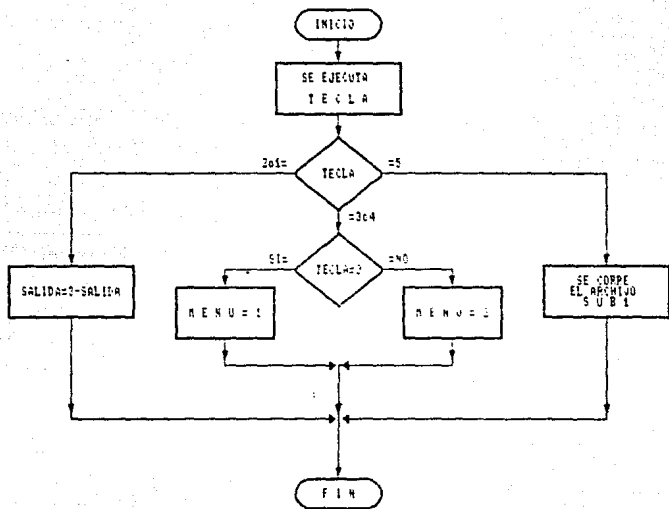


III.39 PRIN2.PRG

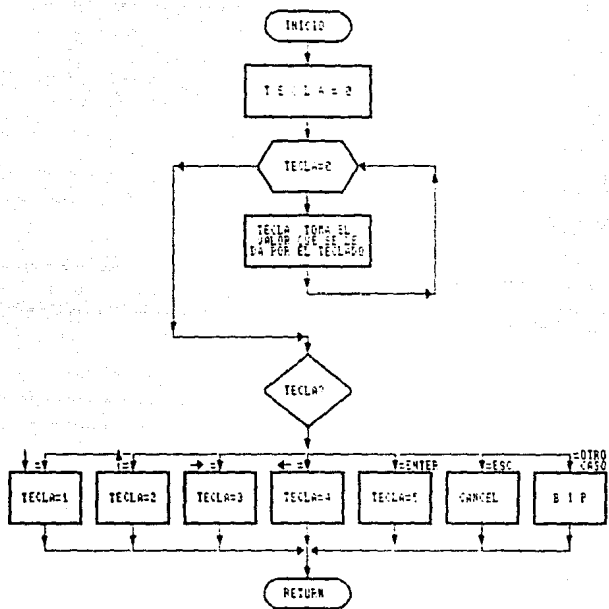




III.3.10 PRIN3.PRG

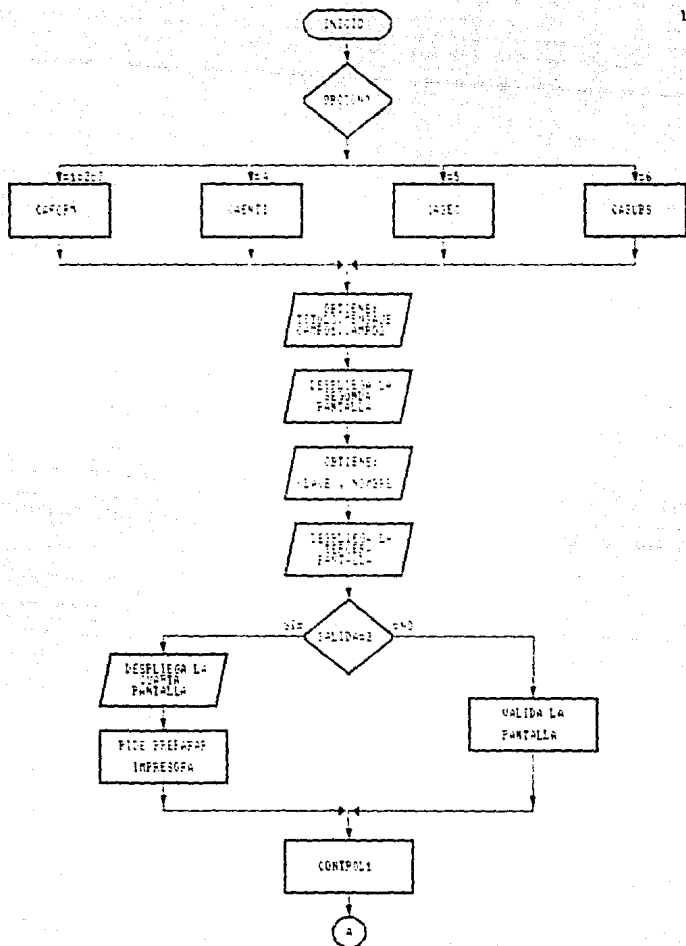


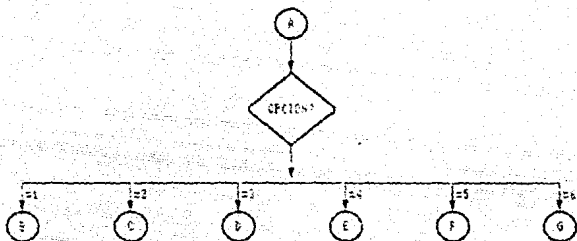
**III.3.11 T E C L A . P R G**

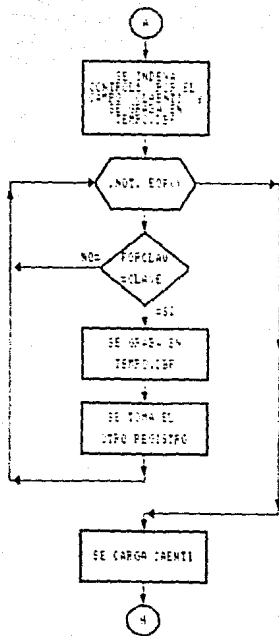


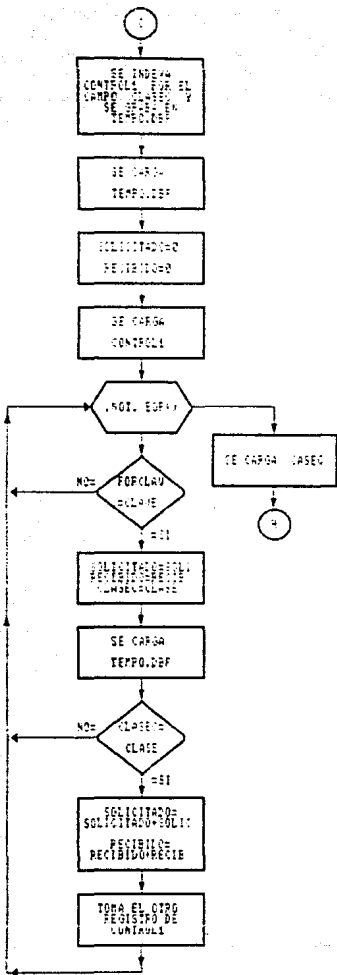
III.3.12 SUB 1. P R G

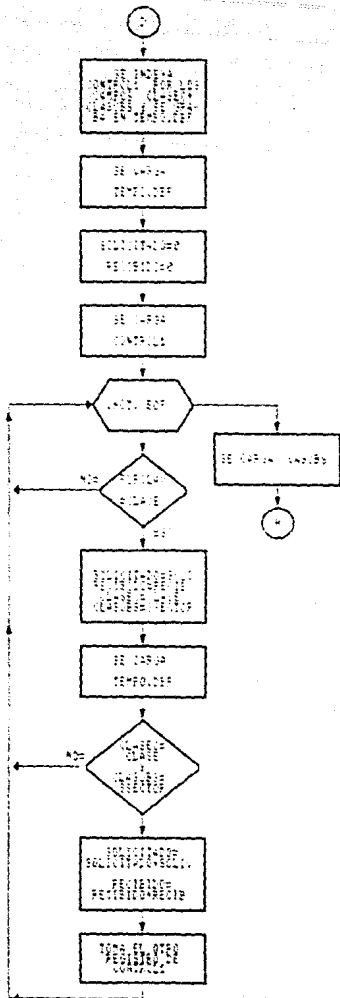


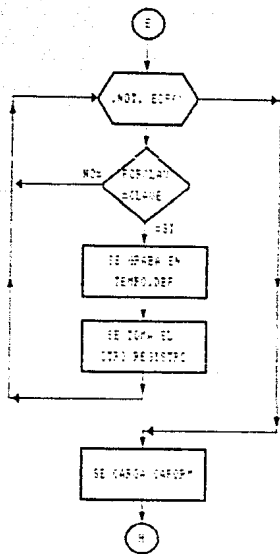


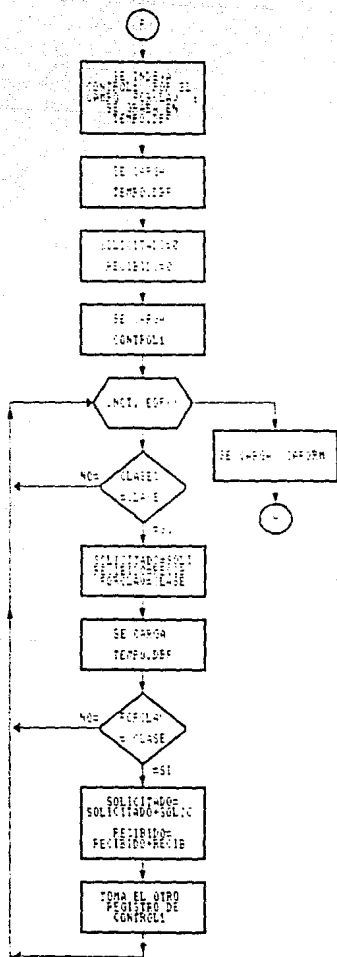


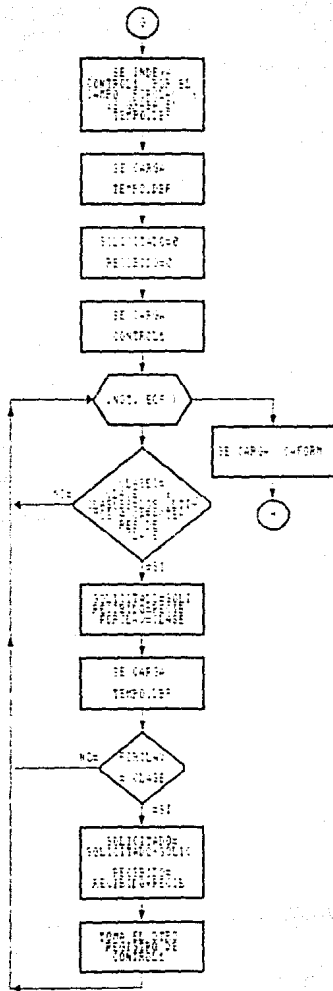




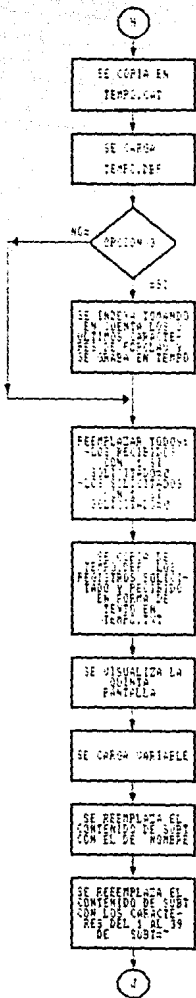


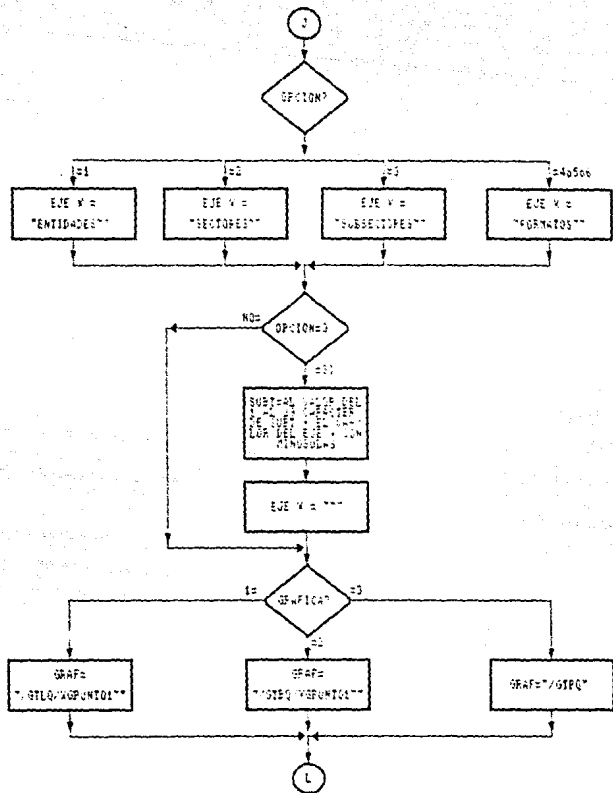


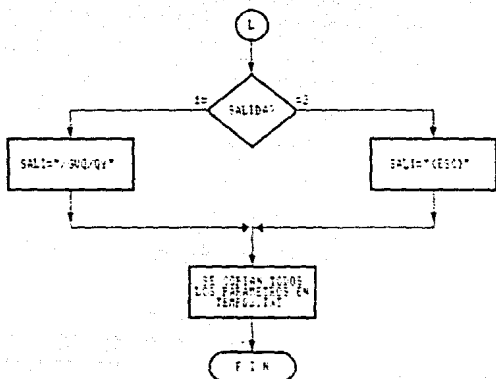




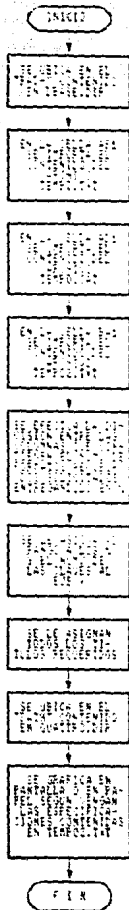








III.3.13 MACRO.SKQ



**CAPITULO IV**  
**LISTADOS Y PANTALLAS**

#### IV.1 LISTADOS

**IV.1.1 INSTALAS  
Y SIGRAF.BAT**



```

CLS
PRINT "INSTALACION DEL SISTEMA 'GISGRAF' EN DISCO DURO"
PRINT ""
PRINT "ESTAS SEGURO QUE DESEAS INSTALAR EL SISTEMA (S/N) "
REPITE:
Z$=""
WHILE Z$=""
  Z$=IN$EY$
WEND
IF Z$<"S" AND Z$<"s" AND Z$<"N" AND Z$<"n" THEN PRINT CHR$(7) : GOTO REPITE
CLS
IF Z$="N" OR Z$="n" THEN END
SIS:
INPUT "UNIDAD DONDE SE INSTALARA EL SISTEMA (C:) > ",UNIDAD$1$
IF UNIDAD$1$="" THEN
  UNIDAD$1$="C:"
ELSE
  IF UNIDAD$1$<"C:" AND UNIDAD$1$<"c:" AND UNIDAD$1$<"D" AND UNIDAD$1$<"d"
  THEN
    PRINT CHR$(7)
    GOTO SIS
  END IF
END IF
PRINT ""
INPUT "DIRECTORIO DONDE SE INSTALARA EL SISTEMA (:) > ",CAMINO$1$
IF CAMINO$1$="" THEN CAMINO$1$ = "\
IF LEFT$(CAMINO$1$,1) <> "\" THEN CAMINO$1$="\\"+CAMINO$1$
PRINT ""
PRINT ""
DB:
INPUT "UNIDAD DONDE SE ENCUENTRA LA BASE DE DATOS (C:) > ",UNIDADDB$
IF UNIDADDB$="" THEN
  UNIDADDB$="C:"
ELSE
  IF UNIDADDB$<"C:" AND UNIDADDB$<"c:" AND UNIDADDB$<"D" AND UNIDADDB$<"d"
  THEN
    PRINT CHR$(7)
    GOTO DB
  END IF
END IF
PRINT ""
INPUT "DIRECTORIO DONDE SE ENCUENTRA LA BASE DE DATOS (DBASE) > ",CAMINODB$
IF CAMINODB$="" THEN CAMINODB$ = "\DBASE"
IF LEFT$(CAMINODB$,1) <> "\" THEN CAMINODB$="\\"+CAMINODB$
PRINT ""
PRINT ""
QUATTRO:
INPUT "UNIDAD DONDE SE ENCUENTRA QUATTRO (C:) > ",UNIDAD123$
IF UNIDAD123$="" THEN
  UNIDAD123$="C:"
ELSE
  IF UNIDAD123$<"C:" AND UNIDAD123$<"c:" AND UNIDAD123$<"D" AND UNIDAD123$<"d"
  THEN
    PRINT CHR$(7)
    GOTO QUATTRO
  END IF
END IF
PRINT ""
INPUT "DIRECTORIO DONDE SE ENCUENTRA QUATTRO (QUATTRO) > ",CAMINO123$
IF CAMINO123$="" THEN CAMINO123$ = "\QUATTRO"
IF LEFT$(CAMINO123$,1) <> "\" THEN CAMINO123$="\\"+CAMINO123$

```

```

CLS
PRINT "  INSTALACION DE SISTEMA"
PRINT ""
PRINT "GENERANDO ARCHIVO "+UNIDAD$IS$+CAMINOSIS$+"SISGRAF.BAT"
OPEN UNIDAD$IS$+CAMINOSIS$+"SISGRAF.BAT" FOR OUTPUT AS #1
PRINT #1, "ECHO OFF"
PRINT #1, "CLS"
PRINT #1, UNIDADDB$
PRINT #1, "CD "+CAMINODB$
PRINT #1, "CICLO"
PRINT #1, "SISGRAF"
PRINT #1, "IF EXIST TEMP0.TXT GOTO QUATTRO"
PRINT #1, "GOTO FIN"
PRINT #1, "QUATTRO"
PRINT #1, UNIDAD123$
PRINT #1, "CD "+CAMINO123$
PRINT #1, "REN MACRO.%0 QUATTRO.WK0"
PRINT #1, "0"
PRINT #1, "REN QUATTRO.WK0 MACRO.%10"
PRINT #1, UNIDADDB$
PRINT #1, "CD "+CAMINODB$
PRINT #1, "DEL TEMP0.*"
PRINT #1, "GOTO CICLO"
PRINT #1, "FIN"
PRINT #1, UNIDAD$IS$
PRINT #1, "CD "+CAMINOSIS$
PRINT #1, "CLS"
CLOSE
PRINT
PRINT "TRASLADANDO ARCHIVOS DE TRABAJO AL DIRECTORIO ";UNIDADDB$;CAMINODB$
PRINT SFC(0);"SISGRAF.EXE"
SHELL "COPY %1\BASE\SISGRAF.EXE "+UNIDADDB$+CAMINODB$
PRINT SFC(0);"VARIABLE.DBF"
SHELL "COPY %1\BASE\VARIABLE.DBF "+UNIDADDB$+CAMINODB$
PRINT "CREA ARCHIVO DE TRABAJO EN EL DIRECTORIO ";UNIDADDB$;CAMINODB$
PRINT SFC(0);"QUATTRO.DIR"
OPEN UNIDADDB$+CAMINODB$+"QUATTRO.DIR" FOR OUTPUT AS #1
WRITE #1, UNIDAD123$+CAMINO123$+"*"
CLOSE
PRINT
PRINT "TRASLADANDO ARCHIVO DE TRABAJO AL DIRECTORIO ";UNIDAD123$;CAMINO123$
PRINT SFC(0);"MACRO.%0"
SHELL "COPY %1\QUATTRO\MACRO.%0 "+UNIDAD123$+CAMINO123$
PRINT "CREA ARCHIVO DE TRABAJO EN EL DIRECTORIO ";UNIDAD123$;CAMINO123$
PRINT SFC(0);"DBASE.DIR"
OPEN UNIDAD123$+CAMINO123$+"DBASE.DIR" FOR OUTPUT AS #1
WRITE #1, UNIDADDB$+CAMINODB$+"*"
CLOSE
SHELL UNIDAD$IS$
SHELL "CD "+CAMINOSIS$
CLS

```

## IV.12 SISTGRAF.PRG

```

SET ECHO OFF
CLEAR
SET TALK OFF
SET ESCAPE OFF
CLEAR ALL
CLOSE ALL
SET STATUS OFF
SET CONFIRM OFF
SET SAFETY OFF
PUBLIC TCCLA
STORE "Formato / ENTIDADES" TO OPCION1
STORE "Formato / SECTORES" TO OPCION2
STORE "Formato / SUBSECTORES" TO OPCION3
STORE "Entidad / FORMATOS" TO OPCION4
STORE "Sector / FORMATOS" TO OPCION5
STORE "Subsector / FORMATOS" TO OPCION6
STORE "Formato / ENTIDADES" TO OPCION01
STORE "Formato / SECTORES" TO OPCION02
STORE "Formato / SUBSECTORES" TO OPCION03
STORE "Entidad / FORMATOS" TO OPCION04
STORE "Sector / FORMATOS" TO OPCION05
STORE "Subsector / FORMATOS" TO OPCION06
STORE "LINEAS" TO GRAFICA1
STORE "BARRAS" TO GRAFICA2
STORE "PASIEL" TO GRAFICA3
STORE "PANTALLA" TO SALIDA1
STORE "IMPRESORA" TO SALIDA2
STORE 1 TO MENU.OPCION,GRAFICA,SALIDA
DO WHILE .T.
SET COLOR TO W/W,GR*/B,W
CLEAR
CLOSE ALL
@ 1, 19 TO 4, 51 DOUBLE
@ 6, 3 TO 19, 31
@ 5, 9 TO 7, 25 DOUBLE
@ 6, 33 TO 13, 53
@ 5, 34 TO 7, 52 DOUBLE
@ 6, 55 TO 13, 75
@ 5, 58 TO 7, 72 DOUBLE
@ 2, 21 SAY "PROGRAMA DE GRAFICACION DE FORMATOS DEL"
@ 3, 25 SAY "SISTEMA INTEGRAL DE INFORMACION"
@ 6, 9 SAY "OPCIONES"
@ 6, 35 SAY "TIPO DE GRAFICA"
@ 6, 59 SAY "SALIDA"
@ 8, 5 SAY OPCION1
@ 10, 5 SAY OPCION2
@ 12, 5 SAY OPCION3
@ 14, 5 SAY OPCION4
@ 16, 5 SAY OPCION5
@ 18, 5 SAY OPCION6
@ 8, 37 SAY GRAFICA1
@ 10, 37 SAY GRAFICA2
@ 12, 37 SAY GRAFICA3
@ 9, 57 SAY SALIDA1
@ 11, 57 SAY SALIDA2
@ 16, 34 SAY "OPCION 1:"
@ 17, 34 SAY "TIPO DE GRAFICA 1:"
@ 18, 34 SAY "SALIDA 1:"
SET COLOR TO W/W
@ 15, 33 TO 19, 75 DOUBLE
SET COLOR TO W/N
@ 20, 0 TO 24, 75 DOUBLE

```

```

@ 10, 07 SAY CHR(200)
@ 21, 1 SAY " < > Y < > PARA ESCOGER OPCION " < > X < > Y < > PARA CHR
DIAR DE MENU "
EL PROGRAMA " PRESIONA <RETURN> PARA CONTINUAR " < > X < > Y < > PARA SALIR D
@ 22, 0 SAY CHR(204)+REPLICATE("M",38)+CHR(206)+REPLICATE("M",39)+CHR(165)
@ 24, 39 SAY CHR(202)
SET COLOR TO W/N
@ 21, 6 SAY CHR(24)
@ 21, 12 SAY CHR(25)
@ 21, 48 SAY CHR(27)
@ 21, 51 SAY CHR(26)
@ 23, 14 SAY "RETURN"
@ 23, 48 SAY "ESC"
SET COLOR TO N/W
@ 0PL SUBSTR(STR(OPCION+100),9) TO OP1
STORE "OPCION"+SUBSTR(OP,2) TO OP2
STORE "OPCION"+OP TO OP3
@ 8*OPCION+2,5 SAY %OP2
@ 16, 50 SAY %OP3
STORE SUBSTR(STR(GRAFICA),10) TO GR
STORE "GRAFICA"+GR TO GR2
@ 8*GRAFICA+2, 37 SAY %GR2
@ 17, 53 SAY %GR2
STORE SUBSTR(STR(SALIDA),10) TO SA
STORE "SALIDA"+SA TO SA2
@ 7*SALIDA+2, 57 SAY %SA2
@ 18, 53 SAY %SA2
DO WHILE .T.
SET COLOR TO N/W
DO CASE
CASE MENU=1
DO PRINT
CASE MENU=2
DO PRINT
CASE MENU=3
DO PRINT
ENDCASE
IF TECLA=5
EXIT
ELSE
LOOP
ENDIF
ENDDO
ENDDO

```

IV.13 CAENTI.DBF

## CLIENTE NOM\_ENT

462 PRODUCTORA E IMPORTADORA DE PAPEL, S.A. DE C.V.  
 750 LOTERIA NACIONAL PARA LA ASISTENCIA PUBLICA  
 400 INSTITUTO MEXICANO DEL CAFE  
 375 FORESTAL VICENTE GUERRERO  
 490 PRODUCTOS FORESTALES MEXICANOS  
 490 TELEFONOS DE MEXICO, S.A. DE C.V.  
 255 AEROPUERTOS Y SERVICIOS AUXILIARES  
 110 CAMINOS Y PUENTES FEDS. DE INGRES. Y SERVS. CONEXOS  
 195 FERROCARRILES NACIONALES DE MEXICO  
 500 DIST. E IMPULSORA COMERCIAL CONASUPO, S.A. DE C.V.  
 637 INST. NAL. DE REG. Y SERVS. SOC. DE LOS TRAB. DEL EDO.  
 641 INSTITUTO MEXICANO DEL SEGURO SOCIAL  
 164 COMISION FEDERAL DE ELECTRICIDAD (C. F. E.)  
 168 COMPANIA DE LUZ Y FUERZA DEL CENTRO, S. A.  
 572 PETROLEOS MEXICANOS  
 802 ALTOS HORNOS DE MEXICO (AHMIA)  
 432 SIDERURGICA LAZARO CARRERAS LAS TRUCHAS, S. A.  
 234 CONSTRUCT. NACIONAL DE CARRERAS DE FERROCARRIL, S.A.  
 692 AZUCAR, S. A. DE C. V.  
 296 FERTILIZANTES MEXICANOS, S. A.

## IV.14 C A F O R M . D B F



FORCLAV	NOMBRE
E-1	COMPLIMIENTO FINANCIERO
E-4	CUMPLIMIENTO DISPOSICIONES AUSTERIDAD
E-5	INGRESO EFECTIVO
E-6	EGRESO EFECTIVO
E-7	DISPONIBILIDADES FINANCIERAS
E-8	DEPOSITOS EN EL SISTEMA BANCARIO (MONEDA NACIONAL)
E-9	DETALLE DE DEPOSITOS MONEDA NACIONAL
E-10	DETALLE DE DEPOSITOS (MONEDA EXTRANJERA)
E-11	DEPOSITOS DE OTRAS INVERSIONES (MONEDA NACIONAL)
E-12	BALANCE DE DIVISAS
E-13	BALANCE DE DIVISAS (PROGRAMA-SEGUIMIENTO)
E-14	BALANCE GENERAL
E-15	ESTADO DE RESULTADOS
E-16	ORIGEN Y APLICACION DE RECURSOS
E-17	ESTADO DE VARIANTES EN EL CAPITAL CONTABLE
E-18	ESTADO DE COSTOS DE PRODUCCION Y VENTAS
E-19	EMPLEO Y REMUNERACIONES
E-20	PROGRAMA NACIONAL DE CAPACITACION Y PRODUCTIVIDAD
E-22	METAS ANUALES DE PRODUCCION: PRESP. Y MODIF.

## IV.15 CASE C.DBF

CLASEC	NOM_SECTOR
04	GOBERNACION
06	HACIENDA Y CREDITO PUBLICO
08	AGRICULTURA Y RECURSOS HIDRAULICOS
09	COMUNICACIONES Y TRANSPORTES
10	COMERCIO Y FOMENTO INDUSTRIAL
12	SALUD
13	ENERGIA, MINAS E INDUSTRIA PARAESTATAL

**IV.16 CASUBS.DBF**

CLASUBS	NON_SUBS
04-00	INDUSTRIA DEL PAPEL
06-00	ASISTENCIA PUBLICA
08-01	AGROPECUARIO
08-02	FORESTAL
09-00	COMUNICACIONES
09-01	AEREO - CARRETERO
10-01	COMERCIO - CONASUFC
12-01	SEGURIDAD SOCIAL
13-01	ENERGETICO
13-02	SIDERURGICO
13-03	METALMECANICO Y BIENES DE CAPITAL
13-05	AZUCARERO
13-07	FERTILIZANTES

**IV.17 CONTROL1.DBF**

LAUREA	NUMERO	CLASSE	ESCLUSO	SOLICITATO	RECIBIDO
4	460	E-1			4
4	460	E-4			4
4	460	E-5			4
4	460	E-6			4
4	460	E-7			4
4	460	E-8			4
4	460	E-9			4
4	460	E-10			4
4	460	E-11			4
4	460	E-12			4
4	460	E-13			4
4	460	E-14			4
4	460	E-15			4
4	460	E-16			4
4	460	E-17			4
4	460	E-18			4
4	460	E-19			4
4	460	E-20			4
6	750	E-1			6
6	750	E-4			6
6	750	E-5			6
6	750	E-6			6
6	750	E-7			6
6	750	E-8			6
6	750	E-9			6
6	750	E-10			6
6	750	E-11			6
6	750	E-12			6
6	750	E-13			6
6	750	E-14			6
6	750	E-15			6
6	750	E-16			6
6	750	E-17			6
6	750	E-18			6
6	750	E-19			6
6	750	E-20			6
8	400	E-1			8
8	400	E-4			8
8	400	E-5			8
8	400	E-6			8
8	400	E-7			8
8	400	E-8			8
8	400	E-9			8
8	400	E-10			8
8	400	E-11			8
8	400	E-12			8
8	400	E-13			8
8	400	E-14			8
8	400	E-15			8
8	400	E-16			8
8	400	E-17			8

CLASEC	LASUBS	CLAENTI	FORCLAV	SOLICITADO	RECIBIDO
00	1	400	E-18	1	1
00	1	400	E-19	0	0
00	1	400	E-20	0	0
00	1	400	E-21	0	0
00	355	E-1	0	0	0
00	355	E-4	0	0	0
00	355	E-5	0	0	0
00	355	E-6	0	0	0
00	355	E-7	0	0	0
00	355	E-8	0	0	0
00	355	E-9	0	0	0
00	355	E-9	0	0	0
00	355	E-10	0	0	0
00	355	E-11	0	0	0
00	355	E-12	0	0	0
00	355	E-13	0	0	0
00	355	E-14	0	0	0
00	355	E-15	0	0	0
00	355	E-16	0	0	0
00	355	E-17	0	0	0
00	355	E-18	0	0	0
00	355	E-19	0	0	0
00	355	E-20	0	0	0
00	355	E-21	0	0	0
00	490	E-1	0	0	0
00	490	E-4	0	0	0
00	490	E-5	0	0	0
00	490	E-6	0	0	0
00	490	E-7	0	0	0
00	490	E-8	0	0	0
00	490	E-9	0	0	0
00	490	E-10	0	0	0
00	490	E-11	0	0	0
00	490	E-12	0	0	0
00	490	E-13	0	0	0
00	490	E-14	0	0	0
00	490	E-15	0	0	0
00	490	E-16	0	0	0
00	490	E-17	0	0	0
00	490	E-18	0	0	0
00	490	E-19	0	0	0
00	490	E-20	0	0	0
00	490	E-21	0	0	0
00	430	E-1	0	0	0
00	430	E-4	0	0	0
00	430	E-5	0	0	0
00	430	E-6	0	0	0
00	430	E-7	0	0	0
00	430	E-8	0	0	0
00	430	E-9	0	0	0
00	430	E-10	0	0	0
00	430	E-11	0	0	0





CLASS	CLASOPS	CLASNT	FORCLAV	SOLICITAC	RECIBIO
9	1	195	E-7	0	0
9	1	195	E-8	0	0
9	1	195	E-9	0	0
9	1	195	E-10	0	0
9	1	195	E-11	0	0
9	1	195	E-12	0	0
9	1	195	E-13	0	0
9	1	195	E-14	0	0
9	1	195	E-15	0	0
9	1	195	E-16	0	0
9	1	195	E-17	0	0
9	1	195	E-18	0	0
9	1	195	E-19	0	0
9	1	195	E-20	0	0
10	1	500	E-1	0	0
10	1	500	E-4	0	0
10	1	500	E-5	0	0
10	1	500	E-6	0	0
10	1	500	E-7	0	0
10	1	500	E-8	0	0
10	1	500	E-9	0	0
10	1	500	E-10	0	0
10	1	500	E-11	0	0
10	1	500	E-12	0	0
10	1	500	E-13	0	0
10	1	500	E-14	0	0
10	1	500	E-15	0	0
10	1	500	E-16	0	0
10	1	500	E-17	0	0
10	1	500	E-18	0	0
10	1	500	E-19	0	0
10	1	500	E-20	0	0
11	1	637	E-1	0	0
11	1	637	E-4	0	0
11	1	637	E-5	0	0
11	1	637	E-7	0	0
11	1	637	E-8	0	0
11	1	637	E-9	0	0
11	1	637	E-10	0	0
11	1	637	E-11	0	0
11	1	637	E-12	0	0
11	1	637	E-13	0	0
11	1	637	E-14	0	0
11	1	637	E-15	0	0
11	1	637	E-16	0	0
11	1	637	E-17	0	0
11	1	637	E-18	0	0
11	1	637	E-19	0	0
11	1	637	E-20	0	0

CLASIFICACION	CLASIFICACION	CLIENTE	FOROCLAV	SOLICITADO	RECIBIDO
10	1	641	E-1	5	1
10	1	641	E-2	0	0
10	1	641	E-3	0	0
10	1	641	E-4	0	0
10	1	641	E-5	2	2
10	1	641	E-6	4	1
10	1	641	E-7	1	1
10	1	641	E-8	3	1
10	1	641	E-9	5	0
10	1	641	E-10	0	0
10	1	641	E-11	0	0
10	1	641	E-12	0	0
10	1	641	E-13	0	0
10	1	641	E-14	0	0
10	1	641	E-15	0	0
10	1	641	E-16	0	0
10	1	641	E-17	0	0
10	1	641	E-18	0	0
10	1	641	E-19	0	0
10	1	641	E-20	4	4
10	1	641	E-21	3	3
16	1	164	E-1	0	0
16	1	164	E-4	0	0
16	1	164	E-5	0	1
16	1	164	E-6	0	0
16	1	164	E-7	0	0
16	1	164	E-8	0	0
16	1	164	E-9	0	0
16	1	164	E-10	4	4
16	1	164	E-11	0	0
16	1	164	E-12	0	0
16	1	164	E-13	0	0
16	1	164	E-14	0	2
16	1	164	E-15	0	0
16	1	164	E-16	0	0
16	1	164	E-17	0	0
16	1	164	E-18	0	2
16	1	164	E-19	0	0
16	1	164	E-20	5	4
16	1	164	E-21	4	1
16	1	190	E-1	0	0
16	1	190	E-4	0	0
16	1	190	E-5	2	0
16	1	190	E-6	4	4
16	1	190	E-7	0	2
16	1	190	E-8	0	1
16	1	190	E-9	6	6
16	1	190	E-10	6	6
16	1	190	E-11	5	3
16	1	190	E-12	0	0
16	1	190	E-13	0	0
16	1	190	E-14	0	0
16	1	190	E-15	0	0

CLASE	CLASUBS	CLAENTI	FORCLAV	SOLICITADO	RECIBIDO
18	1	190	E-16	2	2
18	1	190	E-17	3	3
18	1	190	E-18	4	4
18	1	190	E-19	5	5
18	1	190	E-20	4	4
18	1	190	E-21	1	1
18	1	572	E-1	0	0
18	1	572	E-4	0	0
18	1	572	E-5	0	0
18	1	572	E-6	3	3
18	1	572	E-7	0	0
18	1	572	E-8	0	0
18	1	572	E-9	0	0
18	1	572	E-10	0	0
18	1	572	E-11	0	0
18	1	572	E-12	2	1
18	1	572	E-13	4	4
18	1	572	E-14	4	4
18	1	572	E-15	1	1
18	1	572	E-16	1	1
18	1	572	E-17	3	3
18	1	572	E-18	4	4
18	1	572	E-19	3	3
18	1	572	E-20	5	5
18	1	572	E-21	5	5
18	1	572	E-22	3	3
18	1	572	E-23	3	3
18	1	572	E-24	3	3
18	1	572	E-25	3	3
18	1	572	E-26	3	3
18	1	572	E-27	3	3
18	1	572	E-28	3	3
18	1	572	E-29	3	3
18	1	572	E-30	3	3
18	1	572	E-31	3	3
18	1	572	E-32	3	3
18	1	572	E-33	3	3
18	1	572	E-34	3	3
18	1	572	E-35	3	3
18	1	572	E-36	3	3
18	1	572	E-37	3	3
18	1	572	E-38	3	3
18	1	572	E-39	3	3
18	1	572	E-40	3	3
18	1	572	E-41	3	3
18	1	572	E-42	3	3
18	1	572	E-43	3	3
18	1	572	E-44	3	3
18	1	572	E-45	3	3
18	1	572	E-46	3	3
18	1	572	E-47	3	3
18	1	572	E-48	3	3
18	1	572	E-49	3	3
18	1	572	E-50	3	3
18	1	572	E-51	3	3
18	1	572	E-52	3	3
18	1	572	E-53	3	3
18	1	572	E-54	3	3
18	1	572	E-55	3	3
18	1	572	E-56	3	3
18	1	572	E-57	3	3
18	1	572	E-58	3	3
18	1	572	E-59	3	3
18	1	572	E-60	3	3
18	1	572	E-61	3	3
18	1	572	E-62	3	3
18	1	572	E-63	3	3
18	1	572	E-64	3	3
18	1	572	E-65	3	3
18	1	572	E-66	3	3
18	1	572	E-67	3	3
18	1	572	E-68	3	3
18	1	572	E-69	3	3
18	1	572	E-70	3	3
18	1	572	E-71	3	3
18	1	572	E-72	3	3
18	1	572	E-73	3	3
18	1	572	E-74	3	3
18	1	572	E-75	3	3
18	1	572	E-76	3	3
18	1	572	E-77	3	3
18	1	572	E-78	3	3
18	1	572	E-79	3	3
18	1	572	E-80	3	3
18	1	572	E-81	3	3
18	1	572	E-82	3	3
18	1	572	E-83	3	3
18	1	572	E-84	3	3
18	1	572	E-85	3	3
18	1	572	E-86	3	3
18	1	572	E-87	3	3
18	1	572	E-88	3	3
18	1	572	E-89	3	3
18	1	572	E-90	3	3
18	1	572	E-91	3	3
18	1	572	E-92	3	3
18	1	572	E-93	3	3
18	1	572	E-94	3	3
18	1	572	E-95	3	3
18	1	572	E-96	3	3
18	1	572	E-97	3	3
18	1	572	E-98	3	3
18	1	572	E-99	3	3
18	1	572	E-100	3	3

CLASSE	CLASSE	AURENTI	FORLAV	FOLICITAZ	RE.ELLE
18	000	E-11	0	0	0
18	000	E-12	0	0	0
18	000	E-13	4	4	4
18	000	E-14	0	0	0
18	000	E-15	0	0	0
18	000	E-16	0	0	0
18	000	E-17	0	0	0
18	000	E-18	0	0	0
18	000	E-19	0	0	0
18	000	E-20	4	4	4
18	000	E-21	0	0	0
18	000	E-1	6	6	6
18	000	E-2	0	0	0
18	000	E-3	1	1	1
18	000	E-4	0	0	0
18	000	E-5	0	0	0
18	000	E-6	4	4	4
18	000	E-7	0	0	0
18	000	E-8	4	4	4
18	000	E-9	4	4	4
18	000	E-10	0	0	0
18	000	E-11	0	0	0
18	000	E-12	0	0	0
18	000	E-13	0	0	0
18	000	E-14	0	0	0
18	000	E-15	0	0	0
18	000	E-16	0	0	0
18	000	E-17	0	0	0
18	000	E-18	0	0	0
18	000	E-19	0	0	0
18	000	E-20	0	0	0
18	000	E-21	1	1	1
18	690	E-1	2	2	2
18	690	E-4	0	0	0
18	690	E-5	0	0	0
18	690	E-6	0	0	0
18	690	E-7	0	0	0
18	690	E-8	0	0	0
18	690	E-9	0	0	0
18	690	E-10	0	0	0
18	690	E-11	6	6	6
18	690	E-12	0	0	0
18	690	E-13	0	0	0
18	690	E-14	0	0	0
18	690	E-15	0	0	0
18	690	E-16	0	0	0
18	690	E-17	0	0	0
18	690	E-18	0	0	0
18	690	E-19	0	0	0
18	690	E-20	0	0	0
18	690	E-21	0	0	0
18	296	E-1	0	0	0
18	296	E-4	0	0	0
18	296	E-5	0	0	0

CLASS	CLASS	CLASS	CLASS	CLASS	CLASS
11	11	11	11	11	11
12	12	12	12	12	12
13	13	13	13	13	13
14	14	14	14	14	14
15	15	15	15	15	15
16	16	16	16	16	16
17	17	17	17	17	17
18	18	18	18	18	18
19	19	19	19	19	19
20	20	20	20	20	20
21	21	21	21	21	21
22	22	22	22	22	22
23	23	23	23	23	23
24	24	24	24	24	24
25	25	25	25	25	25
26	26	26	26	26	26
27	27	27	27	27	27
28	28	28	28	28	28
29	29	29	29	29	29
30	30	30	30	30	30
31	31	31	31	31	31
32	32	32	32	32	32
33	33	33	33	33	33
34	34	34	34	34	34
35	35	35	35	35	35
36	36	36	36	36	36
37	37	37	37	37	37
38	38	38	38	38	38
39	39	39	39	39	39
40	40	40	40	40	40
41	41	41	41	41	41
42	42	42	42	42	42
43	43	43	43	43	43
44	44	44	44	44	44
45	45	45	45	45	45
46	46	46	46	46	46
47	47	47	47	47	47
48	48	48	48	48	48
49	49	49	49	49	49
50	50	50	50	50	50
51	51	51	51	51	51
52	52	52	52	52	52
53	53	53	53	53	53
54	54	54	54	54	54
55	55	55	55	55	55
56	56	56	56	56	56
57	57	57	57	57	57
58	58	58	58	58	58
59	59	59	59	59	59
60	60	60	60	60	60
61	61	61	61	61	61
62	62	62	62	62	62
63	63	63	63	63	63
64	64	64	64	64	64
65	65	65	65	65	65
66	66	66	66	66	66
67	67	67	67	67	67
68	68	68	68	68	68
69	69	69	69	69	69
70	70	70	70	70	70
71	71	71	71	71	71
72	72	72	72	72	72
73	73	73	73	73	73
74	74	74	74	74	74
75	75	75	75	75	75
76	76	76	76	76	76
77	77	77	77	77	77
78	78	78	78	78	78
79	79	79	79	79	79
80	80	80	80	80	80
81	81	81	81	81	81
82	82	82	82	82	82
83	83	83	83	83	83
84	84	84	84	84	84
85	85	85	85	85	85
86	86	86	86	86	86
87	87	87	87	87	87
88	88	88	88	88	88
89	89	89	89	89	89
90	90	90	90	90	90
91	91	91	91	91	91
92	92	92	92	92	92
93	93	93	93	93	93
94	94	94	94	94	94
95	95	95	95	95	95
96	96	96	96	96	96
97	97	97	97	97	97
98	98	98	98	98	98
99	99	99	99	99	99
100	100	100	100	100	100

#### IV.18 VARIABLE.DBF





#### IV.19 EJEMPLO DE TEXPO.TXT

1.1  
1.1  
1.1  
1.1  
1.1  
6.2  
1.1  
1.1  
1.1  
1.1  
6.8  
1.1  
1.1  
1.1  
1.1  
2.3  
1.1  
2.1  
4.0

#### IV.110 EJEMPLO DE TEXP02.TXT

"REQUERIMIENTO FINANCIERO", "ENTIDADES", "GTLO. XPUNTO", "AÑO/OY"

## IV.111 EJEMPLO DE TEXPO.CAT

"460"  
"750"  
"400"  
"355"  
"490"  
"430"  
"085"  
"110"  
"195"  
"500"  
"657"  
"641"  
"164"  
"180"  
"572"  
"383"  
"632"  
"234"  
"622"  
"298"

IV.12 PRIN1.PRG

```

* 5. 9 SAY " O P C I O N E S "
* 5. 0 TO 7. 26 DOUBLE
DO WHILE .T.
  DO TECLA
  SET COLOR TO N/W
  DO CASE
    CASE TECLA=1 OR. TECLA=2
      SET COLOR TO W+W,GR+V,W
      @ 5+OPCION*2.5 SAY "OP2"
      IF TECLA=1
        OPCION=OPCION+1
        IF OPCION=7
          OPCION=1
        ENDIF
      ELSE
        OPCION=OPCION-1
        IF OPCION=0
          OPCION=6
        ENDIF
      ENDIF
      SET COLOR TO N/W
      STORE SUBSTR(STR(OPCION+100),9) TO OP
      STORE "OPCION"+SUBSTR(OP,2) TO OP2
      STORE "OPCION"+OP TO OP3
      @ 5+OPCION*2.5 SAY "OP2"
      @ 16. 55 SAY "OP3"
      LOOP
    CASE TECLA=3 OR. TECLA=4
      SET COLOR TO W+W,GR+V,W
      @ 6. 7 SAY " O P C I O N E S "
      @ 5. 0 TO 7. 26 DOUBLE
      SET COLOR TO N/W
      IF TECLA=3
        MENU=2
      ELSE
        MENU=3
      ENDIF
      EXIT
    CASE TECLA=5
      DO SUB1
      EXIT
  ENDCASE

```



## IV.13 PRIN2.PRG

```

ENDDO      @ 8. 05 SAY " TIPO DE GRAFICA "
@ 5. 04 TO 7. 02 DOUBLE
DO WHILE .T.
  DO TECLA
  SET COLOR TO N/W
  DO CASE
    CASE TECLA=1 .OR. TECLA=2
      SET COLOR TO W+W,GR+B,W
      @ 8+GRAFICA*2, 07 SAY &GR
      IF TECLA=1
        GRAFICA=GRAFICA+1
        IF GRAFICA=4
          GRAFICA=1
        ENDIF
      ELSE
        GRAFICA=GRAFICA-1
        IF GRAFICA=0
          GRAFICA=3
        ENDIF
      ENDIF
      SET COLOR TO N/W
      STORE SUBSTR(STR(GRAFICA),10) TO GR
      STORE "GRAFICA"+GR TO GR2
      @ 8+GRAFICA*2, 07 SAY &GR2
      @ 17, 53 SAY &GR2
      LOOP
    CASE TECLA=3 .OR. TECLA=4
      SET COLOR TO W+W,LP+B,W
      @ 6. 05 SAY " TIPO DE GRAFICA "
      @ 5. 04 TO 7. 57 DOUBLE
      SET COLOR TO N/W
      IF TECLA=3
        MENU=3
      ELSE
        MENU=1
      ENDIF
      EXIT
    CASE TECLA=5
      DO SUB1
      EXIT
  ENDCASE
ENDDO

```

IV.14 PRIN3.PRG

```

@ 5, 59 SAY " S A L I D A "
@ 5, 58 TO 7, 72 DOUBLE
DO WHILE .I.
  DO TECLA
  SET COLOR TO N/W
  DO CASE
    CASE TECLA=1 .OR. TECLA=2
      SET COLOR TO W+W,GR+7R,W
      @ 7+SALIDA+2, 57 SAY %SA2
      SALIDA=3-SALIDA
      SET COLOR TO N/W
      STORE SUBSTF(STF(SALIDA),10) TO SA
      STORE "SALIDA"+SA TO SA2
      @ 7+SALIDA+2, 57 SAY %SA2
      @ 18, 53 SAY %SA2
      LOOP
    CASE TECLA=3 .OR. TECLA=4
      SET COLOR TO W+W,GR+7R,W
      @ 6, 59 SAY " S A L I D A "
      @ 5, 58 TO 7, 72 DOUBLE
      SET COLOR TO N/W
      IF TECLA=3
        MENU=1
      ELSE
        MENU=2
      ENDIF
      EXIT
    CASE TECLA=5
      DO SUB1
      EXIT
  ENDCASE
ENDDO

```

**IV.15 TECLA.PRG**

```

8.1.1: END
DO WHILE .T.
  TECLA=0
  DO WHILE (TECLA=0)
    TECLA=INKEY()
  ENDDO
  DO CASE
    CASE TECLA=24 .OR. TECLA=50
      TECLA=1
      EXIT
    CASE TECLA=5 .OR. TECLA=56
      TECLA=2
      EXIT
    CASE TECLA=4 .OR. TECLA=54
      TECLA=3
      EXIT
    CASE TECLA=19 .OR. TECLA=52
      TECLA=4
      EXIT
    CASE TECLA=13
      TECLA=5
      EXIT
    CASE TECLA=27
      SET COLOR TO
      CLEAR
      SET ESCAPE ON
      CANCEL
    OTHERWISE
      ? CHR(7)
      LOOP
  ENDCASE
ENDDO

```

IV.116 SUB 1. P R G

```

CANCEL PUELO 2
PUBLIC CLAVE
DD CASE
  CASE OPCION=1 .OR. OPCION=2 .OR. OPCION=3
    USE CAFORM
    STORE "SELECCION DE FORMATO" TO TITULO
    STORE "CLAVE" PARA ESCOGER FORMATO " TO MENSAJE
    STORE "FORMCLAV" TO CAMPO1
    STORE "NOMBRE" TO CAMPO2
    STORE "Formato:" TO SELECCION
  CASE OPCION=4
    USE CAENTI
    STORE "SELECCION DE ENTIDAD" TO TITULO
    STORE "CLAVE" PARA ESCOGER ENTIDAD " TO MENSAJE
    STORE "CLMENT" TO CAMPO1
    STORE "NOM_ENT" TO CAMPO2
    STORE "Entidad:" TO SELECCION
  CASE OPCION=5
    USE CASEC
    STORE "SELECCION DE SECTOR" TO TITULO
    STORE "CLAVE" PARA ESCOGER SECTOR " TO MENSAJE
    STORE "CLASEC" TO CAMPO1
    STORE "NOM_SECTOR" TO CAMPO2
    STORE "Sector:" TO SELECCION
  CASE OPCION=6
    USE CASUB
    STORE "SELECCION DE SUBSECTOR" TO TITULO
    STORE "CLAVE" PARA ESCOGER SUBSECTOR " TO MENSAJE
    STORE "CLASUB" TO CAMPO1
    STORE "NOM_SUSS" TO CAMPO2
    STORE "Subsector:" TO SELECCION
  ENDCASE
  SET COLOR TO W/W,GR+/R,W
  CLEAR
  @ 2, 1 TO 25, 79 DOUBLE
  SET COLOR TO
  @ 1, 1 SAY LINEA
  @ 24, 1 SAY LINEA
  @ 1, 18 SAY TITULO
  @ 24, 1 SAY "PRESIONA < > PARA CONTINUAR ; < > PARA REGRESAR"
  AL MENU
  SET COLOR TO W+
  @ 24, 18 SAY "RETURN"
  @ 24, 48 SAY "ESC"
  SET COLOR TO W/W
  @ 25, 21 SAY MENSAJE
  SET COLOR TO W+/W,GR+/R,W
  @ 25, 20 SAY CHR(165)
  @ 25, 56 SAY CHR(204)
  @ 25, 25 SAY CHR(24)
  @ 25, 29 SAY CHR(25)
  GO TOP
  *
  DO WHILE (.NOT. EOF)
    @ 1, 4 SAY %CAMPO1
    @ 7, 14 SAY %CAMPO2
    x=y+1
  SKIP
  ENDDO

```



```

GO TOP
DO WHILE .T.
  SET COLOR TO W/N
  * RECORD()=1. 4 SAY SCAMP01
  * RECORD()+2.14 SAY SCAMP02
DO WHILE .T.
  @4
  DO WHILE .T.
    Z=IN(1)
  ENDDO
  SET COLOR TO W*W,OR+*B,W
DO CASE
  CASE Z=24 .OR. Z=50 .OR. Z=51 .OR. Z=52 .OR. Z=53
    EXIT
  CASE Z=17
    RETURN
  OTHERWISE
    @CHR(1)
  LOOP
ENDCASE
ENDDO
* RECORD()+1. 4 SAY SCAMP01
* RECORD()+2.14 SAY SCAMP02
DO CASE
  CASE Z=24 .OR. Z=50
    SKIP
    IF EOF()
      GO TOP
    ENDIF
    LOOP
  CASE Z=51 .OR. Z=52
    SKIP -1
    IF BOF()
      GO EDITOR
    ENDIF
    LOOP
  OTHERWISE
    STORE SCAMP01 TO CLAVE
    STORE SCAMP02 TO NOMBRE
    CLOSE DATABASES
    EXIT
ENDCASE
ENDDO
IF Z=17
  RETURN
ENDIF
SET COLOR TO
CLEAR
* 1. 19 TO 4. 51 DOUBLE
* 2. 21 SAY "PROGRAMA DE GRAFICACION DE FORMATOS DEL"
* 3. 25 SAY "SISTEMA INTEGRAL DE INFORMACION"
SET COLOR TO W*W,OR+*B,W
* 4. 19 CLEAR TO 15. 31
* 5. 19 TO 15 , 51 DOUBLE
SET COLOR TO W/N
* 20. 0 TO 24. 77 DOUBLE
* 20. 39 SAY CHR(200)
* 21. 1 SAY "O P C I O N : "
* 22. 1 SAY SELECCION
* 21. 37 SAY "?: TIPO DE GRAFICA :
* 23. 39 SAY "?: B A = 1 D A :
* 22. 0 SAY CHR(204)+REPLICATE(CHR(205),38)+CHR(206)+REPLICATE(CHR(205),39)+CHR(
+100)
* 24. 39 SAY CHR(200)
SET COLOR TO W*W
* 21. 17 SAY SCOP

```

```

@ 23, 17 SAY CLAVE
@ 21, 59 SAY AGR2
@ 23, 59 SAY A5A2
IF SALIDA = 0
  SET COLOR TO W+W
  @ 11, 21 SAY "P R E S P A R E   L A   I N F O R M A C I O N"
  SET COLOR TO N+W
  @ 13, 21 SAY "presione cualquier tecla para continuar"
  SET COLOR TO
  @ 19, 1 SAY " " = REGRESAR
  SET COLOR TO W+W
  @ 19, 2 SAY "ESC"
  SET COLOR TO
  PUBLIC MAL
  STORE 0 TO MAL
  DO WHILE MAL = 0
    ? CHR(7)
    STORE 0 TO I
    DO WHILE I.T.
      ? INPR(I)
      IF I=0
        ? " "
        EXIT
      ENDIF
    ENDDO
    IF I=27
      RETURN
    ENDIF
    @ 0,0 SAY " "
    SET DEVICE TO PRINTER
    MAL = 1
  ENDDO
  SET DEVICE TO SCREEN
  @ 19, 1 SAY " "
ENDIF
SET COLOR TO W+W,OR+W,B,W
@ 10, 20 CLEAR TO 14, 50
SET COLOR TO W+W
@ 11, 24 SAY "E S F I Z A R E   U N   M O M E N T O"
SET COLOR TO N+W
@ 13, 27 SAY "Procesando informacion"
SET COLOR TO
USE CONTROL1
DO CASE
  CASE OFCION=1
    INDEX ON CLAEINTI TO TEMPO
    COPY TO TEMPO FOR FORCLAV=CLAVE
    USE CAENTI
    COPY TO TEMPO,CAT FIELDS CLAEINTI BELI
  CASE OFCION=2
    SET UNIQUE ON
    INDEX ON CLASEC TO TEMPO
    COPY TO TEMPO
    CLOSE INDEX
    SET UNIQUE OFF
    SELECT 2
    USE TEMPO
    REPLACE ALL SOLICITADO WITH 0
    REPLACE ALL RECIBIDO WITH 0
    SELECT 1
    GO TOP
    DO WHILE .NOT. EOF()
      IF FORCLAV=CLAVE
        STORE SOLICITADO TO SOLIC
        STORE RECIBIDO TO RECIB
        STORE CLASEC TO CLASE
        SELECT 1

```

```

LOCATE FOR CLASEC=CLASE
REPLACE SOLICITADO WITH SOLICITADO+SOLIC
REPLACE RECIBIDO WITH RECIBIDO +RECIB
SELECT 1
ENDIF
SKIP
ENDDD
USE CASEC
COPY TO TEMPO.CAT FIELDS CLASEC DELI
CASE OPCION=2
SET UNIQUE ON
INDEX ON CLASEC+10 + CLASUBS TO TEMPO
COPY TO TEMPO
CLOSE INDEX
SET UNIQUE OFF
SELECT 1
USE TEMPO
REPLACE ALL SOLICITADO WITH 0
REPLACE ALL RECIBIDO WITH 0
SELECT 1
GO TOP
DO WHILE (.NOT. EOF())
  IF FORCLAV=CLAVE
    STORE SOLICITADO TO SOLIC
    STORE RECIBIDO TO RECIB
    STORE CLASEC TO CLASE
    STORE CLASUBS TO SSECTOR
    SELECT 1
    LOCATE FOR CLASEC=CLASE .AND. CLASUBS=SSECTOR
    REPLACE SOLICITADO WITH SOLICITADO+SOLIC
    REPLACE RECIBIDO WITH RECIBIDO +RECIB
    SELECT 1
  ENDIF
  SKIP
ENDDD
USE CASUBS
COPY TO TEMPO.CAT FIELDS CLASUBS DELI
CASE OPCION=4
COPY TO TEMPO (FOR CURRENT=VAL(CLAVE))
USE CAFORM
COPY TO TEMPO.CAT FIELDS FORCLAV DELI
CASE OPCION=5
SET UNIQUE ON
INDEX ON FORCLAV TO TEMPO
COPY TO TEMPO
CLOSE INDEX
SET UNIQUE OFF
SELECT 2
USE TEMPO
REPLACE ALL SOLICITADO WITH 0
REPLACE ALL RECIBIDO WITH 0
SELECT 1
GO TOP
DO WHILE (.NOT. EOF())
  IF CLASEC=VAL(CLAVE)
    STORE SOLICITADO TO SOLIC
    STORE RECIBIDO TO RECIB
    STORE FORCLAV TO CLASE
    SELECT 1
    LOCATE FOR FORCLAV=CLASE
    REPLACE SOLICITADO WITH SOLICITADO+SOLIC
    REPLACE RECIBIDO WITH RECIBIDO +RECIB
    SELECT 1
  ENDIF
  SKIP
ENDDD

```

```

USE CAPFORM
COPY TO TEMPORAT FIELDS FORCLAY DELI
CASE OPCION=1
SET UNIQUE ON
INDEX ON FORCLAY TO TEMP0
COPY TO TEMP0
CLOSE INDEX
SET UNIQUE OFF
SELECT 2
USE TEMP0
REPLACE ALL SOLICITADO WITH 0
REPLACE ALL RECIBIDO WITH 0
SELECT 1
GO TOP
DO WHILE (.NOT. EOF)
  IF CLASE=VAL(CLAVE) .AND. CLASURG=VAL(SUBSTR(CLAVE,4))
    STORE SOLICITADO TO SOLIC
    STORE RECIBIDO TO RECIB
    STORE FORCLAY TO CLASE
    SELECT 2
    LOCATE FOR FORCLAY=CLASE
    REPLACE SOLICITADO WITH SOLICITADO+SOLIC
    REPLACE RECIBIDO WITH RECIBIDO +RECIB
    SELECT 1
  ENDIF
SKIP
ENDDO
USE CAPFORM
COPY TO TEMPORAT FIELDS FORCLAY DELI
ENDCASE
CLOSE ALL
USE TEMP0
IF OPCION = 2
  INDEX ON VAL(SUBSTR(FORCLAY,1)) TO TEMP0
ENDIF
REPLACE ALL RECIBIDO WITH 1 FOR SOLICITADO=0
REPLACE ALL SOLICITADO WITH 1 FOR SOLICITADO=0
COPY TO TEMP0 FIELDS SOLICITADO , RECIBIDO DELI
SET COLOR TO N W
CLEAR
SET COLOR TO N/W
@ 12,31 SAY "ESPERE UN MOMENTO"
SET COLOR TO
USE VARIABLE
GO )
REPLACE SUBT WITH NOMBRE
DO CASE
  CASE OPCION = 1
    REPLACE EJEY WITH "ENTIDADES"
  CASE OPCION = 2
    REPLACE EJEY WITH "SECTORES"
  CASE OPCION = 3
    REPLACE EJEY WITH "SUBSECTORES"
  OTHERWISE
    REPLACE EJEY WITH "FORMATOS"
ENDCASE
IF GRAFICA = 3
  REPLACE SUBT WITH SUBSTR(SUBT,1,27)+"/"+LOWER(EJEY)
  REPLACE EJEY WITH ""
ENDIF
STORE TRIM(SUBT) TO SUBTITULO
IF LEN(SUBTITULO) = 40
  SUBTITULO = SUBSTR(SUBTITULO,1,39)+""
ELSE
  SUBTITULO = SUBTITULO + ""
ENDIF

```

```
REPLACE SPAN WITH "GUNT01"
ENDIF
DO CASE
CASE GRAFICA = 1
  REPLACE GRAF WITH "GUNT01"
CASE GRAFICA = 2
  REPLACE SPAN WITH "GUNT01"
CASE GRAFICA = 3
  REPLACE GRAF WITH "GUNT01"
ENDCASE
IF SALIDA = 1
  REPLACE SALI WITH "GUNT01"
ELSE
  REPLACE SALI WITH "GUNT01"
ENDIF
COPY TO TEMP02.DBF
CLOSE ALL
```

IV.17 MACRO.SKQ

```

(GOTO)A27"/FINDBASE.DIR"
/FO
<
<
<
(MENCUNTO)A25"
(GOTO)A61"
(FINTERFO)A7"
(GOTO)A61"
(FINTEMPO)CAT"
(GOTO)A61"
(FINTEMPOD)TNT"
(GOTO)A61"/A61/A61"
(EN)DOWN)LEFT"
(EN)DOWN"
(RIGHT)
(GOTO)A61"
(REF)EN)DOWN"
(GOTO)A61"
(ORGA)EN)DOWN)TO"
(GOTO)A61"
(EN)DOWN)R"
(GOTO)A61"
(DEF)PER)ESC)G"
(DEF)Y)DE)COMPL)M)ENT)O"
(DEF)S)IST)E)M)A)DE)G)R)A)F)I)C)A)C)I)O)N)D)E)L)S)I)
(MER)A)A)27"
(A)O)T)A
GO
(MER)A)A)31"
/ROT"
<
<
<
GO
(MER)A)A)34"
(GOTO)A27"/FINQUATTRO.DIR"
/FO
<
<
(MER)A)A)37"(GOTO)A61"
(MER)A)A)38)DOWN)ESC)ESC)D)ESC)G"
/JOY

```

\* CAMBIA EL DIRECTORIO DE TRABAJO

\* DEFINE LAS COORDENADAS DEL PUNTO:

\* VA A LA CELDA A61

\* LEE EL ARCHIVO DE DATOS

\* VA A LA CELDA A61

\* LEE EL ARCHIVO CON EL RANGO Y

\* VA A LA CELDA A61

\* LEE EL ARCHIVO DE PARAMETROS

\* CALCULA LA RELACION DEL PRIMER DAT

\* COPIA LA MISMA RELACION

\* A LAS CELDAS SIGUIENTES

\* DEFINE EL FORMATO DE PORCENTAJE

\* PARA EL RANGO DE DATOS ESPECIFICAD

\* SE UBICA EN LA PRIMER CELDA DEL RA

\* INICIALIZA GRAFICA Y DEFINE EL RAN

\* SE UBICA EN LA PRIMER CELDA

\* Y DEFINE EL RANGO

\* DEFINE EL FORMATO EN PORCENTAJE DE

\* DEFINE LOS TITULOS

\* GENERALES

\* TRASLADA EL CONTENIDO DE LAS

\* CELDAS, Y DEFINE LOS TITULOS

\* VARIABLES.

\* TRASLADA Y OPERA LA CELDA CON

\* LA DEFINICION DE TIPO DE GRAFICA

\* CAMBIA EL DIRECTORIO DE TRABAJO

\* TRASLADA Y OPERA LA CELDA CON

\* LA DEFINICION DE SALIDA

\* IMPRIME LA GRAFICA

\* SALE DE LOTUS

IV2 PANTALLAS



#### IV.2.1 SERIACION DE PANTALLAS

## PRIMERA PANTALLA

PROGRAMA DE GRAFICACION DE FORMATOS DEL  
SISTEMA INTEGRAL DE INFORMACION

O P C I O N E S

# Formato / ENTIDADES  
# Formato / SECTORES  
# Formato / SUBSECTORES  
# Entidad / FORMATOS  
# Sector / FORMATOS  
# Subsector / FORMATOS

TIPO DE GRAFICA

L I N E A S  
B A R R A S  
P A S T E L

S A L I D A

P A N T A L L A  
I M P R E S O R A

O P C I O N : # formato / ENTIDADES  
T I P O D E G R A F I C A : L I N E A S  
S A L I D A : P A N T A L L A

PARA ESCOGER OPCION

PARA CAMBIAR DE MENU

PRESSIONA RETURN PARA CONTINUAR

TECLA PARA SALIR DEL PROGRAMA

## SEGUNDA PANTALLA

## SELECCION DE FORMATO

E-1	SEGUIMIENTO FINANCIERO
E-4	CUMPLIMIENTO DISPOSICIONES ANTERIORMENTE
E-5	INGRESO EFECTIVO
E-6	EGRESO EFECTIVO
E-7	DISPONIBILIDADES FINANCIERAS
E-8	DEPOSITOS EN EL SISTEMA BANCARIO (MONEDA NACIONAL)
E-9	DETALLE DE DEPOSITOS MONEDA NACIONAL
E-10	DETALLE DE DEPOSITOS (MONEDA EXTRANJERA)
E-11	DEPOSITOS DE OTRAS INVERSIONES (MONEDA NACIONAL)
E-12	BALANCE DE DIVISAS
E-13	BALANCE DE DIVISAS (PROGRAMA-SEGUIMIENTO)
E-14	BALANCE GENERAL
E-15	ESTADO DE RESULTADOS
E-16	ORIGEN Y APLICACION DE RECURSOS
E-17	ESTADO DE VARIANTES EN EL CAPITAL CONTABLE
E-18	ESTADO DE CUENTAS DE TRANSACCIONES Y CUENTAS
E-19	EMPLEO Y REMUNERACIONES
E-20	PROGRAMA NACIONAL DE CAPACITACION Y PRODUCTIVIDAD
E-22	METAS ANUALES DE PRODUCCION, PRESUP. Y MODIF.

FRESINA (RETURN) PARA CONTINUAR      ESCN PARA REGRESAR AL MENU

## TERCERA PANTALLA

PROGRAMA DE GRAFICACION DE FORMATOS DEL  
SISTEMA INTEGRAL DE INFORMACION

PREPARE LA IMPRESORA  
Presione cualquier tecla para continuar

ES: 5 DEGRABAR

SELECCION : F formato / ENTIDADES	TIPO DE GRAFICA : LINEAS
Formato : E-1	SOLICITA : IMPRESORA

## CUARTA PANTALLA



PROGRAMA DE GRAFICACION DE FORMATOS DEL  
SISTEMA INTEGRAL DE INFORMACION

ESPERE UN MOMENTO

Procesando informacion

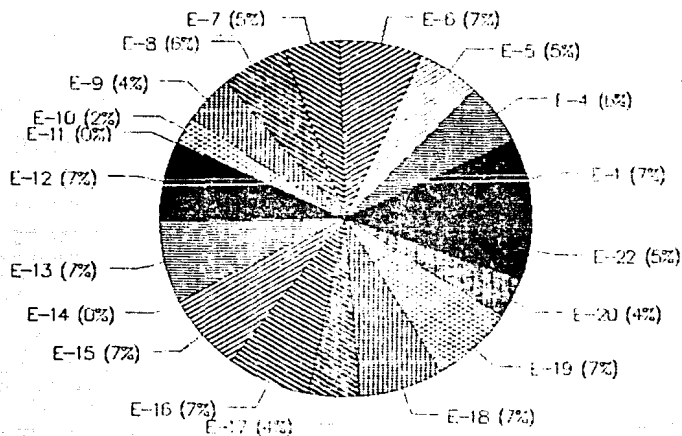
OPCION : # formato / ENTIDADES	TIPO DE GRAFICA : LINEAS
Formato : E-1	SALIDA : IMPRESORA

## QUINTA PANTALLA

ESPERE UM MOMENTO

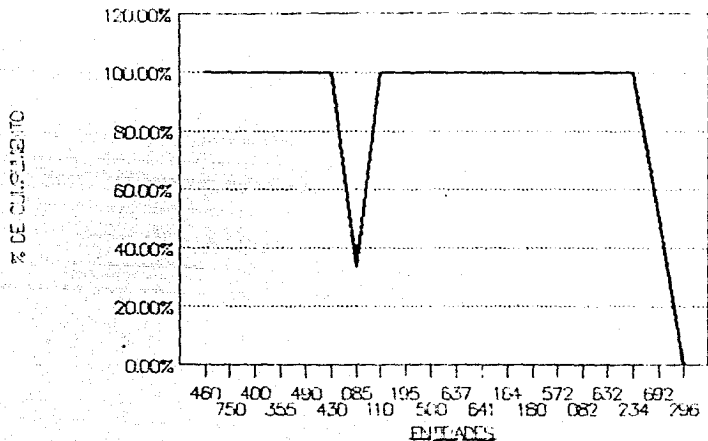
## IV.2.2 EJEMPLOS DE TIPOS DE GRAFICAS

# SISTEMA DE GRAFICACION DEL S.I.I. INDUSTRIA DEL PAPEL /formatos



# SISTEMA DE GRAFICACION DEL S.I.I.

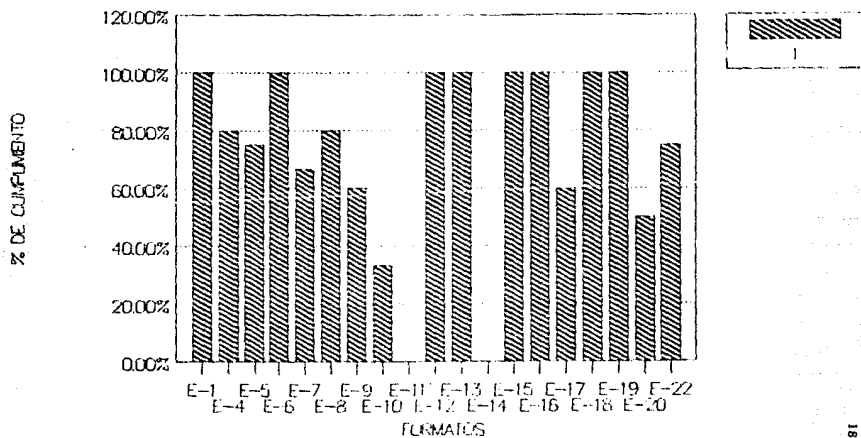
## SEGUIMIENTO FINANCIERO



1

# SISTEMA DE GRAFICACION DEL S.I.I.

## PRODUCTORA E IMPORTADORA DE PAPEL, S.A.



## CONCLUSION

A lo largo de la presente tesis se ha podido constatar:

1) Al implantar el Sistema de Graficación en la Secretaría de la Contraloría General de la Federación se ha cumplido la finalidad por la que fue creada: ahorrar tiempo (horas-hombre) en el análisis y selección de la información, repercutiendo por lo tanto en el costo que esto implica, ya que los resultados que se visualizan en una gráfica antes se obtenían en alrededor de un día de trabajo, lo cual traducido a remuneración del empleado sería aproximadamente de \$100,000.00 (esto por gráfica). Aunque no debemos perder de vista el hecho de que en muchas ocasiones es requerido el análisis de lo equivalente a 2 o más gráficas.

2) Resulta de gran utilidad en diversas ocasiones el realizar el enlace de 2 paquetes que son compatibles, como en este caso el dBase III Plus y el Quattro, ya que ambos se han complementado, pudiéndose llegar de esa manera a crear un Sistema que sin alguno de ellos no sería capaz de funcionar.



## BIBLIOGRAFIA

- BARRAS, EDWARD M.**  
Lotus 1-2-3 Guía para Usuarios  
 Mc Graw-Hill  
 México, 1986.
- FAST, ROBERT**  
Programas Prácticos del Lotus 1-2-3  
 Mc Graw-Hill  
 México, 1985.
- GANE, CHRIS y SARSON, TRISH**  
Introducción a sistemas de análisis. 1984  
 Prentice Hall  
 Englewood Cliffs, 1979.
- JAMES, MARTIN**  
Organización de las bases de datos  
 Prentice Hall  
 México, 1985.
- LISKIN, MIRIAM**  
dBase III Plus  
 Mc Graw-Hill  
 México, 1988.
- HANUAL DE QUATTRO**  
Borland International  
 U. S. A. . 1987.
- MICROSOFT MS-DOS**  
Guía del Usuario  
 México, 1987.
- PANIAGUA ZWANZIGER, ABELARDA**  
dBase III Plus  
 Mc Graw-Hill  
 México, 1988.
- SANDERS, DONALD H.**  
Informática: Presente y Futuro  
 Mc Graw-Hill  
 México, 1990.
- VAN GIGCH, JOHN P.**  
Teoría General de Sistemas  
Aplicada  
 Trillas  
 México, 1981.
- WALSH E., HYLES**  
Entendiendo a las Computadoras  
 Continental  
 México, 1984.

YOURDON, EDWARD y CONSTANTINE, LARRY L.  
Structured Design  
Prentice Hall  
Englewood Cliffs, 1981.

ZINKY, MARGARET y WERNER, JUSTIN  
Programming in C++  
Scott, Foresman and Company  
Londres, 1988.

## A N E X O 1

SISTEMA INTEGRAL DE INFORMACION  
 COMPROMISO DE (DEFICIT) O SUPERAVIT FINANCIERO 1

191

SECTOR: \_\_\_\_\_

ENTIDAD: \_\_\_\_\_

( EN MILLONES DE PESOS )

CUADRO C-2

C O N C E P T O	ENERO	FEBRERO	MARZO	ENERO-MARZO
<b>A. INGRESOS</b>				
COMUNAL POR VENTA DE BILM.V. Y SERV.				
Cuentas del INPS e ISSSTE 2 /				
OTROS INGRESOS				
OPERACIONES AJENAS NETAS				
TECEROS				
NECUPERABLES				
<b>B. GASTO PROGRAMABLE</b>				
GASTO DE OPERACION				
SERVICIOS PERSONALES				
MATERIALES Y SUMINISTROS				
SERVICIOS GUBERNALES				
IMPUESTOS				
OTROS				
OPERACIONES EXTRAORDINARIAS				
INVERSION FISICA				
INVERSION FINANCIERA				
OPERACIONES AJENAS NETAS				
TECEROS				
NECUPERABLES				
<b>C. (DEF.) O SUP. ANTES DE INT.</b>				
<b>D. INTERES Y GASTOS DE LA DEUDA</b>				
<b>E. (DEF.) O SUP. ANTES DE TRANSF.</b>				
<b>F. TRANSFERENCIAS DEL GOBIERNO FEDERAL</b>				
PARA GASTO CORRIENTE				
PARA INVERSION FISICA				
PARA INVERSION FINANCIERA				
PARA AMORTIZACION DE LA DEUDA				
PARA PROY. DE INTERESES				
<b>G. (DEF.) O SUP. TOTAL</b>				
<b>H. EMPLEAMIENTO NETO</b>				
INTEROS				
DISPOSICIONES				
AMORTIZACIONES				
ESTEROS				
DISPOSICIONES				
AMORTIZACIONES				
<b>I. VARIACION DE DISPONIBILIDADES</b>				
INTEROS				
INICIAL				
FINAL				
EXTEROS				
INICIAL				
FINAL				

1/ LOS IMPUESTOS DE NOTAR EN EL REVERSO DE ESTE CUADRO  
 2/ SOLO PARA SER LLENADO POR EL INPS E ISSSTE  
 3/ Nota

PERIODO DE INFORMACION  
 AREA RECEPTORA

11