

EVALUACION DEL PERSONAL DOCENTE

CURSO: INT. AL USO DE LAS COMPUTADORAS PERSONALES

FECHA: 6-17 FEBRERO

CONFERENCISTA	DOMINIO DEL TEMA	USO DE AYUDAS AUDIOVISUALES	COMUNICACION CON EL ASISTENTE	PUNTUALIDAD
ACT. COINTO BARRERA LIBRADO				

EVALUACION DE LA ENSEÑANZA

ORGANIZACION Y DESARROLLO DEL CURSO	
GRADO DE PROFUNDIDAD LOGRADO EN EL CURSO	
ACTUALIZACION DEL CURSO	
APLICACION PRACTICA DEL CURSO	

EVALUACION DEL CURSO

CONCEPTO	CALIF.
CUMPLIMIENTO DE LOS OBJETIVOS DEL CURSO	
CONTINUIDAD EN LOS TEMAS	
CALIDAD DEL MATERIAL DIDACTICO UTILIZADO	

ESCALA DE EVALUACION: 1 A 10

1.- ¿LE AGRADO SU ESTANCIA EN LA DIVISION DE EDUCACION CONTINUA?

SI	NO
----	----

SI INDICA QUE "NO" DIGA PORQUE.

2.- MEDIO A TRAVES DEL CUAL SE ENTERO DEL CURSO:

PERIODICO EXCELSIOR		FOLLETO ANUAL		GACETA UNAM		OTRO MEDIO	
PERIODICO EL UNIVERSAL		FOLLETO DEL CURSO		REVISTAS TECNICAS			

3.- ¿QUE CAMBIOS SUGERIRIA AL CURSO PARA MEJORARLO?

4.- ¿RECOMENDARIA EL CURSO A OTRA(S) PERSONA(S)?

SI		NO	
----	--	----	--

5.- ¿QUE CURSOS LE SERVIRIA QUE PROGRAMARA LA DIVISION DE EDUCACION CONTINUA.?

6.- OTRAS SUGERENCIAS:

7.- ¿EN QUE HORARIO LE SERIA CONVENIENTE SE IMPARTIERAN LOS CURSOS DE LA DIVISION DE EDUCACION CONTINUA?
MARQUE EL HORARIO DE SU AGRADO

LUNES A VIERNES DE 16 A 20 HORAS	MARTES Y JUEVES DE 17 A 21 HS SABADO DE 10 A 14 HS.	OTRO
LUNES, MIERCOLES Y VIERNES DE 17 A 21 HORAS	VIERNES DE 17 A 21 HS. SABADOS DE 10 A 14 HS	



**FACULTAD DE INGENIERIA, UNAM.
DIVISION DE EDUCACION CONTINUA
INTRODUCCION A LAS COMPUTADORAS PERSONALES**

HOJA ELECTRONICA

FEBRERO 1995

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

LOTUS inicia el software que integra aplicaciones específicas, tales como la Hoja Electrónica, Procesamiento de Textos, Base de Datos y Graficación.

Permite realizar un gran número de tareas y facilita el intercambio de información entre una y otra aplicación.

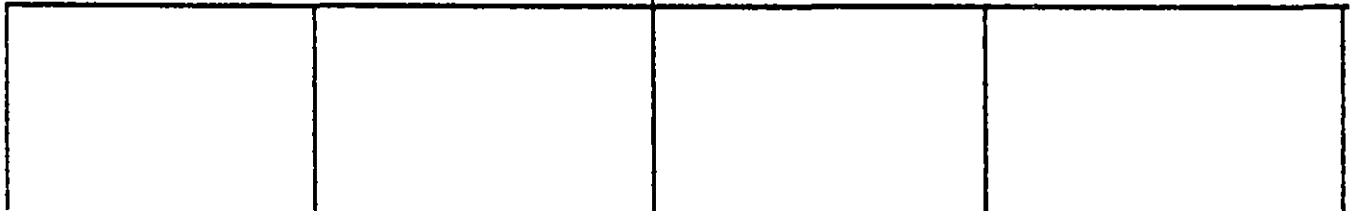
LOTUS se puede definir como:

- * Una potente Hoja Electrónica
- * Con recursos muy prácticos de graficación
- * Con posibilidades de procesar textos
- * Con capacidad de efectuar búsquedas y ordenamientos en bancos de información
- * Con recursos suficientes para graficar e imprimir reportes
- * Con posibilidades prácticas para automatizar procedimientos y programas de aplicación.

COMPONENTES DE LOTUS

El software LOTUS esta compuesto de 5 paquetes:

LOTUS



1-2-3 PRINTGRAPH TRANSLATE INSTALL VIEW

1-2-3

Hoja Electrónica-Gráfica-Base de Datos-
Procesador de Palabras: es el paquete de
cálculo propiamente.

PrintGraph

Impresión de Gráficas. Es el paquete que
imprimirá las gráficas generadas en 1-2-3.
También se pueden definir parámetros y
características para las gráficas, así como
para interface, impresora o graficador del
sistema de cómputo.

Translate

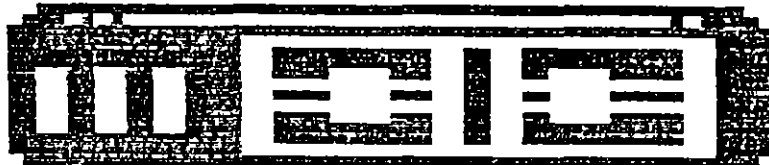
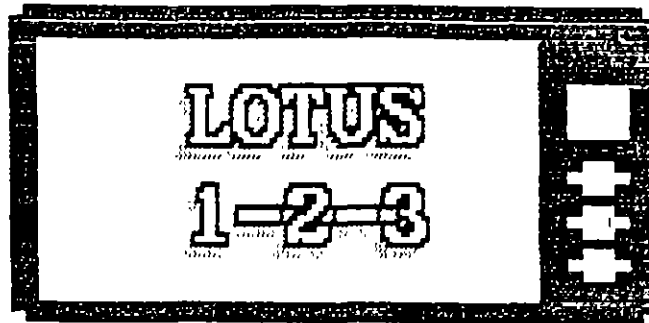
Transferencia de Archivos. Permite
transferir archivos desde y hacia LOTUS
(los software considerados son: VisiCalc,
Data Interchange Format Files, dBASE II,
dBASE III, etc).

Install

Instalación del LOTUS en el Sistema de
Cómputo.
Paquetes que considera el tipo de
computadora y video.

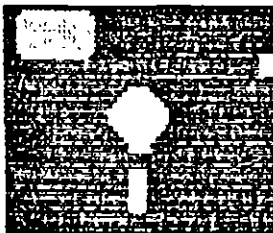
View

Descripción de LOTUS. Este paquete
introduce al usuario a las Hojas
electrónicas y muestra algunas aplicaciones
del paquete 1-2-3.



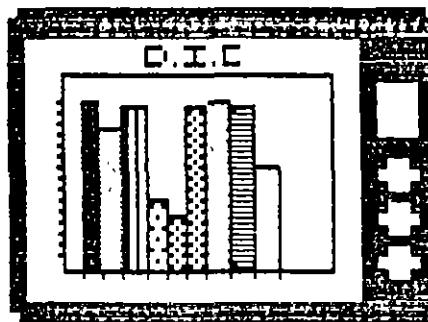
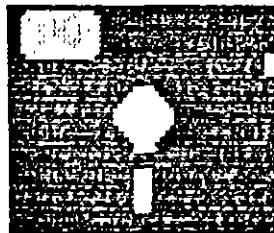
HOJA

**ELECTRONICA
DE DATOS**

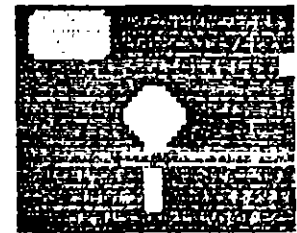


	A	B	C	D
1	25.00			
2	55.00			
3	11.00			
4	42.00			
5	31.00			

GRAFICADOR



**MANEJO DE
INFORMACION POR
MEDIO DE B.D.**



RFC	N.EMP.	SCO.
CEFR	1001	CUE3
EURB	1103	CUE2
MADE	1316	CUE5
GAGS	1574	CUE6

DESCRIPCION DE LOTUS 1-2-3

LOTUS 1-2-3 es un paquete que integra tres herramientas computacionales:

- HOJA ELECTRONICA DE CALCULO, donde podemos almacenar información en columnas y renglones, relacionar la información mediante fórmulas que son calculadas en forma automática. Cuenta con funciones matemáticas, financieras, lógicas, estadísticas y de calendario. Además 1-2-3 tiene capacidad de formato para controlar la apariencia de la hoja y proporciona control de la impresora para obtener los reportes deseados.

- GRAFICAS, donde podemos representar los resultados obtenidos de la hoja de cálculo o los datos proporcionados a ésta. Las gráficas que pueden ser elaboradas con Lotus 1-2-3 son fáciles de generar utilizando tan sólo unas cuantas teclas.

- MANEJO DE INFORMACION POR MEDIO DE BASE DE DATOS, LOTUS 1-2-3 puede organizar y manipular la información contenida en la hoja de cálculo. Organizándola en orden alfabético, ascendente o descendente; localizar o extraer información utilizando criterios de selección.

LOTUS 1-2-3 puede ser utilizado en cualquier Área donde se requiera tener la información ordenada en columnas y renglones, entre otras:

- Planeación
- Análisis Financiero
- Contabilidad
- Proyecciones y Pronósticos
- Control de Inventario

Se obtienen las siguientes ventajas a través de su uso:

- Ahorro de tiempo al efectuar los cálculos.
- La probabilidad de error es disminuida ya que los cálculos no son realizados manualmente.
- Se pueden probar diferentes alternativas con un mínimo de tiempo ya que la hoja es recalculada instantáneamente.
- La organización de la información es más eficiente.

En general LOTUS 1-2-3 es una herramienta poderosa para la solución de problemas y toma de decisiones. Sus características son:

- Fácil de aprender y usar.
- Hoja de cálculo de 8192 renglones por 256 columnas.
- Ayuda en línea para acceder rápidamente referencias.
- Capacidad de graficar y manejar información.
- Ventanas y títulos para consultar más de una sección de la hoja de cálculo al mismo tiempo.
- Cuenta con menús de comandos.
- Ofrece control de periféricos.
- Cuenta con funciones matemáticas, financieras y de fechas.
- Puede manejar Macros ("programas").
- Rápido en la ejecución de cálculos.

Limitaciones:

- Cuando la hoja de cálculo es muy grande se requiere de memoria adicional y se recomienda disco rígido, ya que las hojas demasiado grandes no pueden ser almacenadas completas en disco en el formato de un solo lado. En este último caso se pueden almacenar en partes.

- El tiempo de recálculo con hojas demasiado grandes puede ser muy largo, por lo que se recomienda utilizar el modo de recálculo manual, así sólo se recalcula la hoja hasta que se hayan introducido los datos.

HOJA DE CALCULO

PANEL DE CONTROL

CONTENIDO DE LA CELDA

INDICADOR DE MODO

DIRECCION DE LA CELDA

ANCHO DE LA COLUMNA

FORMATO

D:B

(F2)

[W 16]

+\$C\$5:12

MENU

FIXED SCIENTIFIC CURRENCY GENERAL +/- PERCENT DATE TEXT WIDEN RESI
FIXED NUMBER OF DECIMAL PLACES (X.XX)

	A	B	C	D	E
1					
2					
3					
4					
5					
6					
7					
8					
9					
10					
11					
12					
13					
14					
15					
16					
17					
18					
19					
20					

INDICADOR DE FECHA Y TIEMPO

05-SEP-88 08:27 AM

STATUS

SCROLL

CELDA

LA HOJA DE TRABAJO (WORKSHEET)

Objetivo: Conocer los componentes de la Hoja de Trabajo.

Una hoja de trabajo (WORKSHEET), es semejante a una hoja tabular de contabilidad compuesta de 256 columnas y 8192 renglones, formando celdas en sus intersecciones. Cada celda tiene una dirección compuesta de las coordenadas (su columna y su renglón). Las columnas de la hoja son designadas por letras (A,B,C,.....IV) y los renglones por números (1,2,3,4,.....8192).

Note que la celda D8 es resaltada en la hoja de trabajo, la cual es móvil y es llamada Celda Cursor.

Las tres líneas que se encuentran por arriba de la orilla de la Hoja de Trabajo son llamadas Panel de Control. El Panel de Control es utilizado para mostrar la posición de la celda cursor, los menús de comandos, contenido de la celda, teclas que se han oprimido para alimentar o editar la información en la celda; también muestra el contenido de la celda y el formato con que se muestra la información.

El Indicador de Modo se encuentra en el extremo superior derecho de la pantalla. El indicador de modo es muy útil para conocer de que forma LOTUS 1-2-3 ESTA INTERPRETANDO NUESTRAS INSTRUCCIONES.

En el extremo inferior izquierdo aparece el Indicador de Fecha y Tiempo, que nos indica la fecha (DD-MM-AA) y la hora (HH:MM AM ó PM); en el extremo inferior derecho aparecen los Indicadores de Status (estado) del programa.

CONTENIDO DE LAS CELDAS

Las celdas almacenan información que puede ser de tres tipos:

- **ETIQUETAS.**- Una etiqueta puede ser una serie hasta de 240 caracteres, y se tienen 3 justificaciones para mostrar las etiquetas.

- **NUMEROS.**- LOTUS 1-2-3 puede almacenar números tan grandes como 1.0E+308 y tan pequeños como 1.0E-308.

- **FORMULAS.**- Una fórmula es una instrucción para que LOTUS 1-2-3 calcule un número. Las fórmulas se utilizan para definir relaciones entre las celdas de la hoja de cálculo, es decir que unas celdas van a depender de otras. Siempre que el valor de una celda cambia, 1-2-3 recalcula automáticamente los valores de todas las fórmulas en la Hoja de Cálculo, a menos que se indique recálculo manual.

INDICADORES DE STATUS

SIGNIFICADO

CALC

LA HOJA DE TRABAJO NECESITA RECALCULAR LAS FORMULAS CONTENIDAS EN ELLA, PRESIONE LA TECLA DE RECALCULACION (F9).

CAPS

SE ENCUENTRA ENCENDIDA LA TECLA DE MAYUSCULAS (CapsLock).

CIRC

SE ENCIENDE CUANDO LA HOJA DE TRABAJO CONTIENE UNA FORMULA QUE HACE REFERENCIA A LA MISMA CELDA QUE LA CONTIENE (ESTO SOLO OCURRE CUANDO LA RECALCULACION ES MANUAL).

END

INDICA QUE SE HA HECHO PAUSA DURANTE LA EJECUCION DE UNA MACROINSTRUCCION.

END

INDICA QUE SE PRESIONO LA TECLA (END).

NUM

INDICA QUE LA TECLA (NumLock) ESTA ENCENDIDA.

OVR

INDICA QUE SE PRESIONO LA TECLA (INS) Y EN VEZ DE INSERTAR EN LA EDICION DE UNA CELDA REPLAZA LOS CARACTERES.

SCROLL

INDICA QUE ESTA ENCENDIDA LA TECLA (Scroll Lock).

STEP

INDICA QUE SE HA INVOCADO LA EJECUCION PASO A PASO DE UNA MACROINSTRUCCION.

SST

INDICA QUE UNA MACROINSTRUCCION SE ESTA EJECUTANDO EN MODO PASO A PASO.

INDICADORES

DE MODO

SIGNIFICADO

EDIT

INDICA QUE EL CONTENIDO DE UNA CELDA ESTA EDITANDOSE.

ERROR

MANIFIESTA QUE EXISTE UN ERROR EN LA CELDA ACTUAL; PRESIONE ESCAPE O RETURN PARA QUITARLO.

FILES

UN MENU DE ARCHIVOS HA SIDO DESPLEGADO.

FIND

INDICA QUE UNA BUSQUEDA SE ESTA EFECTUANDO (DEL MODULO DE DATOS).

FORM

SE ESTA EDITANDO UNA LINEA DE FORMATO DURANTE UNA OPCION /DATA PHASE.

HELP

INDICA QUE SE ESTA EN NIVEL DE AYUDA.

LABEL

EN ESPERA DE CARACTERES.

MENU

INDICA QUE SE ESTA EN EL MENU DE COMANDOS DE LOTUS, PARA EFECTUAR UNA ACTIVIDAD U OPERACION.

NAMES

UN MENU DE NOMBRES DE RANGOS O DE GRAFICAS SE HA DESPLEGADO.

POINT

PARA SEÑALAR UNA CELDA O RANGO.

READY

EN ESPERA DE CUALQUIER TIPO DE INFORMACION EN LAS CELDAS.

STAT

INDICA QUE SE HA DESPLEGADO EL GENERAL DE LA HOJA DE TRABAJO.

VALUE

EN ESPERA DE UN VALOR NUMERICO O UNA FORMULA.

WAIT

INDICA AL OPERADOR DE 1-2-3 QUE SE ESTA EFECTUANDO UN PROCESO Y QUE ES NECESARIO ESPERAR.

MOVIMIENTO DE LA CELDA CURSOR

Objetivo: Utilizar las teclas para el movimiento de la celda cursor.

La Hoja de Trabajo de LOTUS 1-2-3 es una colección de celdas, organizadas en columnas y renglones. Cada Celda puede almacenar un sólo elemento de información, sea fórmula, número o etiqueta. La pantalla de la computadora es una ventana dentro de la Hoja de Trabajo. En un momento dado, sólo puede usted ver parte de la hoja, más no toda la hoja. Para observar diferentes partes de la hoja debemos mover la ventana. Las siguientes teclas nos permiten posicionar la Celda Cursor donde se desee.

ENTRADA DE DATOS

Objetivo: Alimentar Números, Etiquetas y Fórmulas a las celdas através del teclado.

Existen tres formas de alimentar los datos a las celdas:

- 1.- Tecleándolos.
- 2.- Copiándolos o moviéndolos desde otra(s) celda(s) en la misma hoja de trabajo.
- 3.- Trayéndolos desde el disco, de hojas de trabajo creadas anteriormente.

En esta sección únicamente estudiaremos como penetrar datos por medio del teclado. Para registrar una entrada en una determinada celda lo que se debe de hacer es posicionar la celda cursor en la celda deseada alimentar a la celda, observe que cada carácter que usted teclea va apareciendo en la segunda línea del Panel de Control. Cuando usted acabe de teclear la entrada, entonces oprima <ENTER> ó alguna de las teclas de movimiento de la celda cursor para terminar la entrada. Al terminar la entrada la segunda línea del Panel de Control desaparece. Si existía algo en la celda, entonces dicho contenido será substituido por lo que acaba de teclear.

Cada entrada a la celda puede ser hasta de 240 caracteres, muchos más de los que 1-2-3 puede mostrar en el Panel de Control. Si se teclean más de 80 caracteres que puede mostrar, entonces la segunda línea mostrada comienza a desplazarse hacia la derecha hasta que usted oprima <ENTER>.

Cada vez que se termina una entrada a una celda tecleando <ENTER> ó algunas de las teclas de movimiento de cursor ocurre lo siguiente:

- 1.- LOTUS 1-2-3 verifica si hay algún error en la entrada, en caso de que ocurra, entonces LOTUS 1-2-3 emite un sonido y aparece el mensaje de ERROR en el Indicador de Modo.
- 2.- Si se tiene recálculo automático, 1-2-3 recalcula las fórmulas en la Hoja de Trabajo.
- 3.- La pantalla es actualizada y LOTUS 1-2-3 regresa a Modo READY esperando instrucciones.

TIPOS DE ENTRADA

Se pueden crear tres tipos de entradas: números, fórmulas y etiquetas. Se debe indicar a 1-2-3 como interpretar la entrada, esto se logra por medio del primer carácter que se teclea.

EL MENU DE COMANDOS

Objetivo: Conocer el funcionamiento y estructura del Menú de comandos de LOTUS 1-2-3.

Durante cada sesión de 1-2-3 usted podrá utilizar los comandos que ejecutan una determinada función, por ejemplo: podrá copiar, mover y borrar datos de la Hoja de Trabajo; transferir datos entre la Hoja de Trabajo y el almacenamiento en disco; imprimir reportes; dibujar gráficas; o procesar bases de datos.

El primer paso es el invocar el Menú de Comandos que se efectúa en modo READY oprimiendo la diagonal /. Oprimiendo </> aparece el Menú de Comandos en el Panel de Control. La primera línea de Panel de Control aún muestra la localidad en la que se encuentra el apuntador de celdas y su contenido. La segunda línea muestra el menú se encuentra resaltado en video inverso por el apuntador de comandos, el cual puede ser movido hacia otras opciones del menú por medio de las teclas de movimiento del apuntador.

En la tercera línea del Panel de Control aparece una breve descripción del comando que se encuentra resaltado por la celda cursor. Dicha descripción puede consistir de alguna de las siguientes opciones: una lista de opciones (submenú) ó una frase acerca de las acciones que el comando efectúa.

La estructura del menú de comandos es de árbol, donde podemos ir seleccionando el comando ó subcomandos que van constituyendo las ramas del mismo.

-- Existen dos formas de seleccionar una opción del menú:

- Posicionando la celda cursor de menús en el comando deseado y luego oprimiendo <ENTER>.
- Tecleando la primera letra del comando, por ejemplo si se desea el comando COPY se oprime la tecla <C>.

PARA CORREGIR LOS ERRORES.

Si usted cometió algún error en la selección de algún comando, siempre se puede regresar al paso anterior, oprimiendo la tecla <ESC>, regresa a la etapa anterior. Si usted se encuentra en algún submenú, <ESC> regresa al menú previo y si usted oprime <ESC> en el menú principal regresará a modo READY.

INDICANDO RANGOS EN LOTUS 1-2-3

Objetivo Particular: Manejar rangos en LOTUS 1-2-3.

Un rango de celdas es un rectángulo de celdas. Este rectángulo puede ser un renglón o más celdas, una columna de celdas adyacentes, ó un bloque de dos ó más renglones y columnas. Una sola celda también es considerada como un rango que consiste de un renglón y una columna. Los rangos son utilizados tanto en comandos como en funciones de LOTUS 1-2-3.

Existen tres formas de indicar rangos:

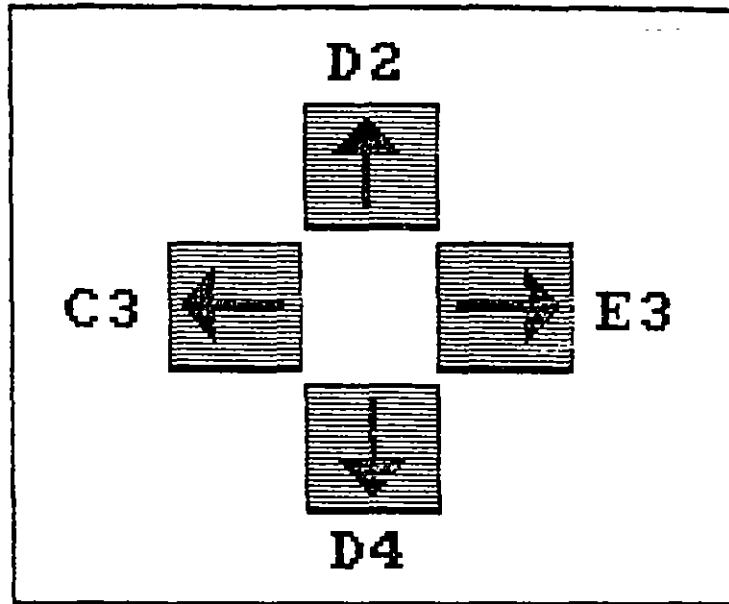
- 1.- Tecleando las direcciones de celdas que se encuentran en esquinas contrarias en el rango deseado.
- 2.- Expandiendo la celda cursor de tal forma que resalte el rango completo (modo POINT).
- 3.- Asignando al rango un nombre específico.

TECLEANDO LAS DIRECCIONES DE LAS CELDAS.

Usted puede especificar un rango tecleando las direcciones de cualquier par de celdas diagonalmente opuestas en el rango. Se debe de utilizar por lo menos un punto (.) para separar dos direcciones. LOTUS 1-2-3 siempre edita las direcciones separadas con dos puntos, los espacios están prohibidos en la especificación de rangos. Por ejemplo:

A5..F150
G11..A20

MOVIMIENTO DE LA CELDA CURSOR



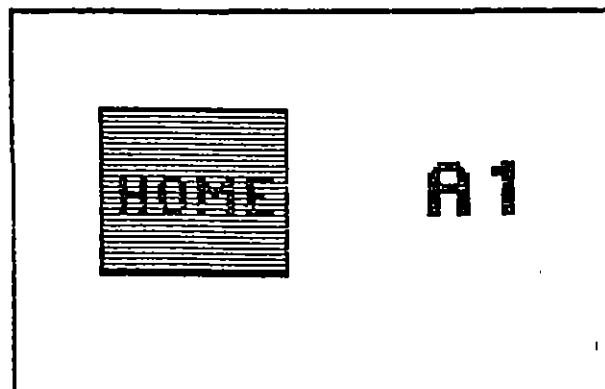
C3:

READY

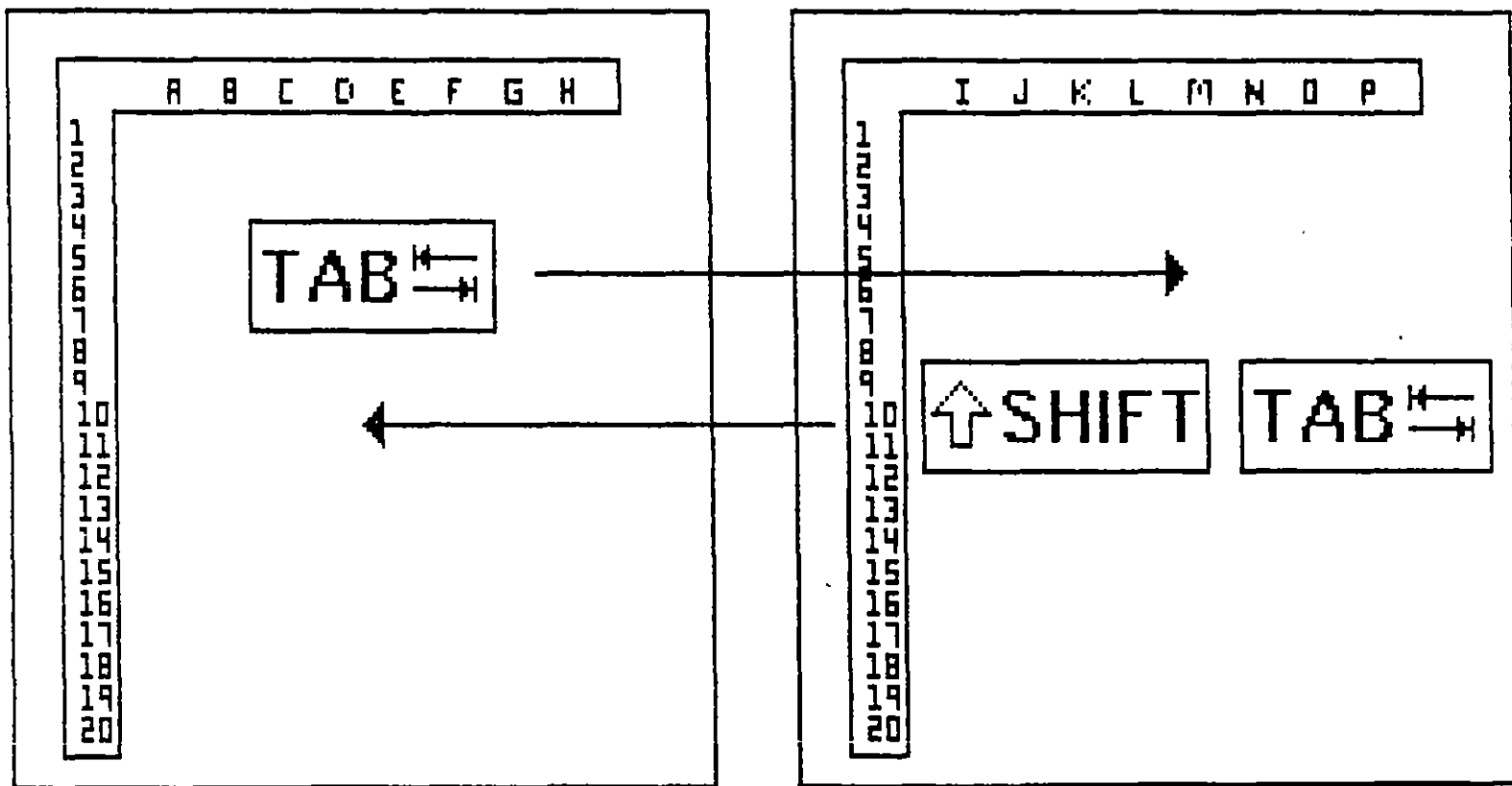
	A	B	C	D	E	F
1						
2						
3						
4						
5						
6						
7						
8						
9						
10						
11						
12						
13						
14						
15						
16						
17						
18						
19						
20						

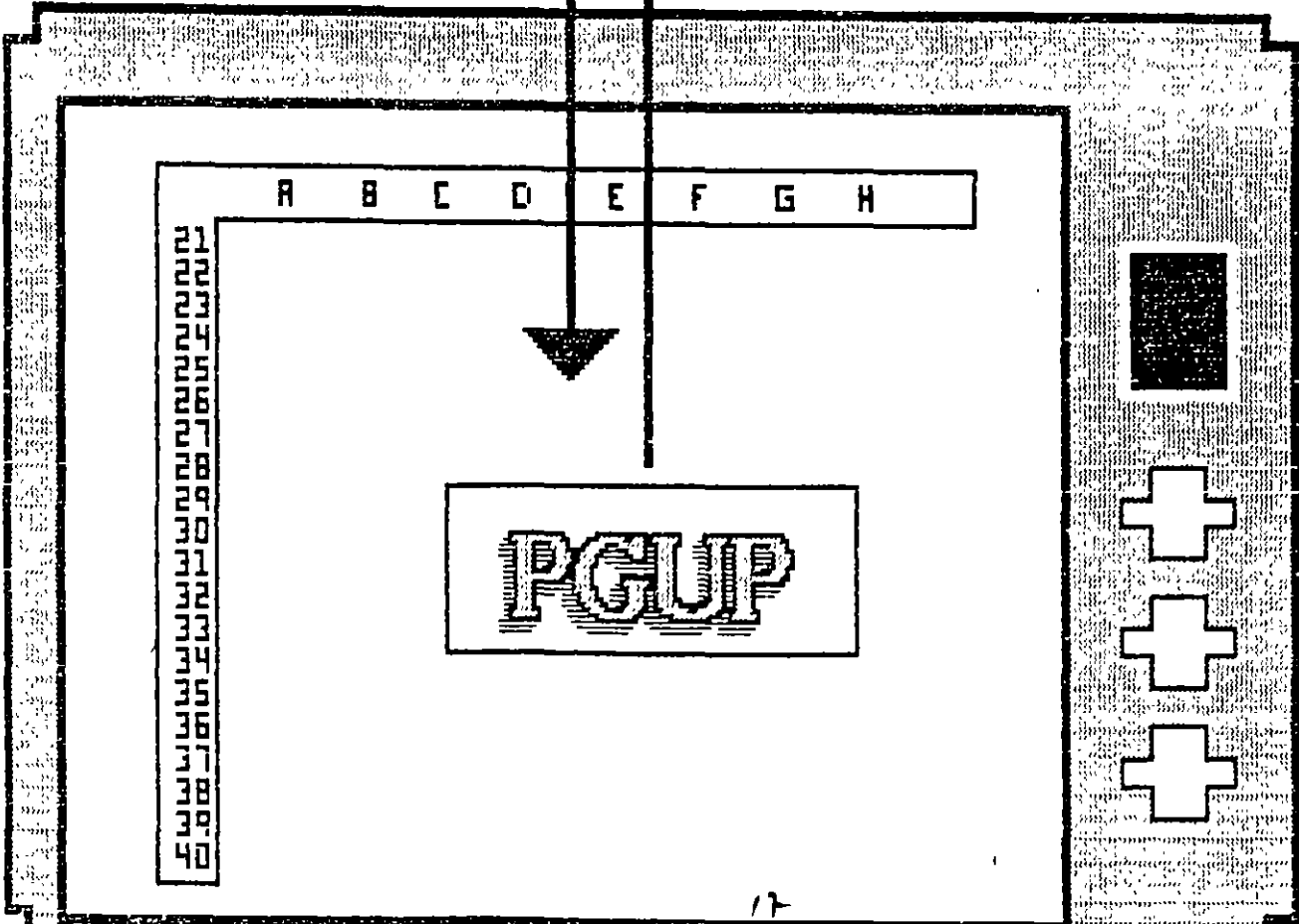
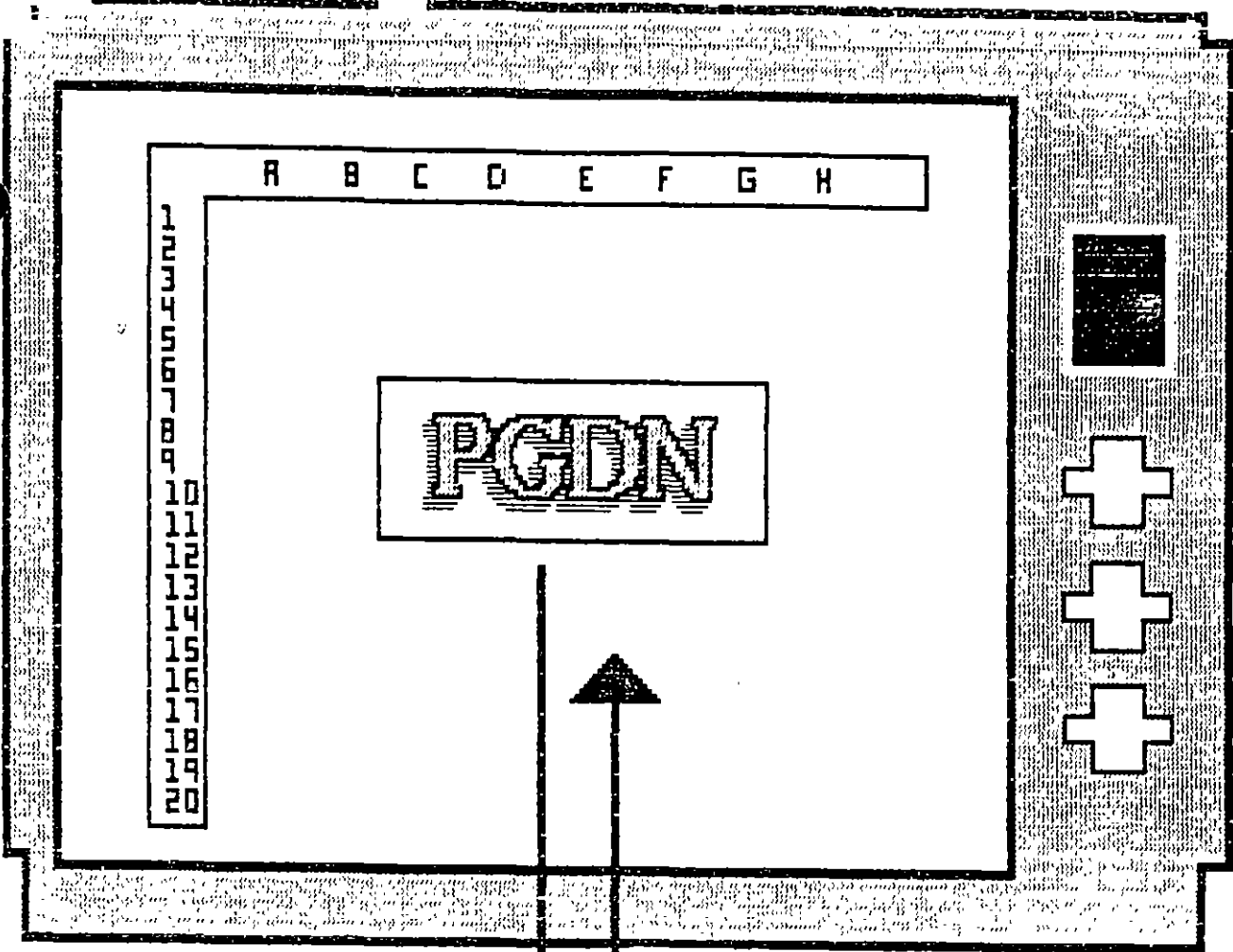
	A	B	C	D	E	F	G
1							
2							
3							
4							
5							
6							
7							
8							
9							
10							
11							
12							
13							
14							
15							
16							
17							
18							
19							
20							

MOVIMIENTO DE LA CELDA CURSOR







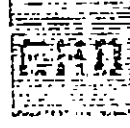

MOVIMIENTO DE LA CELDA CURSOR





	A	B	C	D	E	F	G
1		ENERO	FEBRERO	MARZO	ABRIL	MAYO	JUNIO
2	1980	\$1,600,500	\$8,455,222	\$1,996,543	\$3,434,324	\$6,231,111	\$9,689,234
3	1981	\$9,689,234	\$8,333,229	\$4,664,962	\$4,444,341	\$3,654,431	\$1,826,456
4	1982	\$3,145,645	\$5,456,879	\$7,987,678	\$7,755,354	\$8,445,678	\$3,986,446
5	1983	\$2,111,567	\$6,826,229	\$5,667,894	\$2,111,567	\$1,600,500	\$8,876,550
6	1984	\$5,435,623	\$1,826,456	\$2,322,232	\$5,435,623	\$9,689,234	\$6,466,543
7	1985	\$1,415,111	\$3,986,446	\$9,424,248	\$1,415,111	\$3,145,645	\$3,336,221
8	1986	\$3,434,324	\$8,876,550	\$6,231,111	\$3,434,324	\$7,777,999	\$1,996,543
9	1987	\$4,444,341	\$6,466,543	\$3,654,431	\$3,333,464	\$8,530,989	\$4,664,962
10	1988	\$7,755,354	\$3,336,221	\$8,445,678	\$9,678,123	\$6,199,677	\$7,987,678
11							
12							
13							
14							
15							
16							
17							
18							
19							
20							

EQUIPAMIENTO DE LA CELDA CURSOR

		A10
		G10
		G1

18

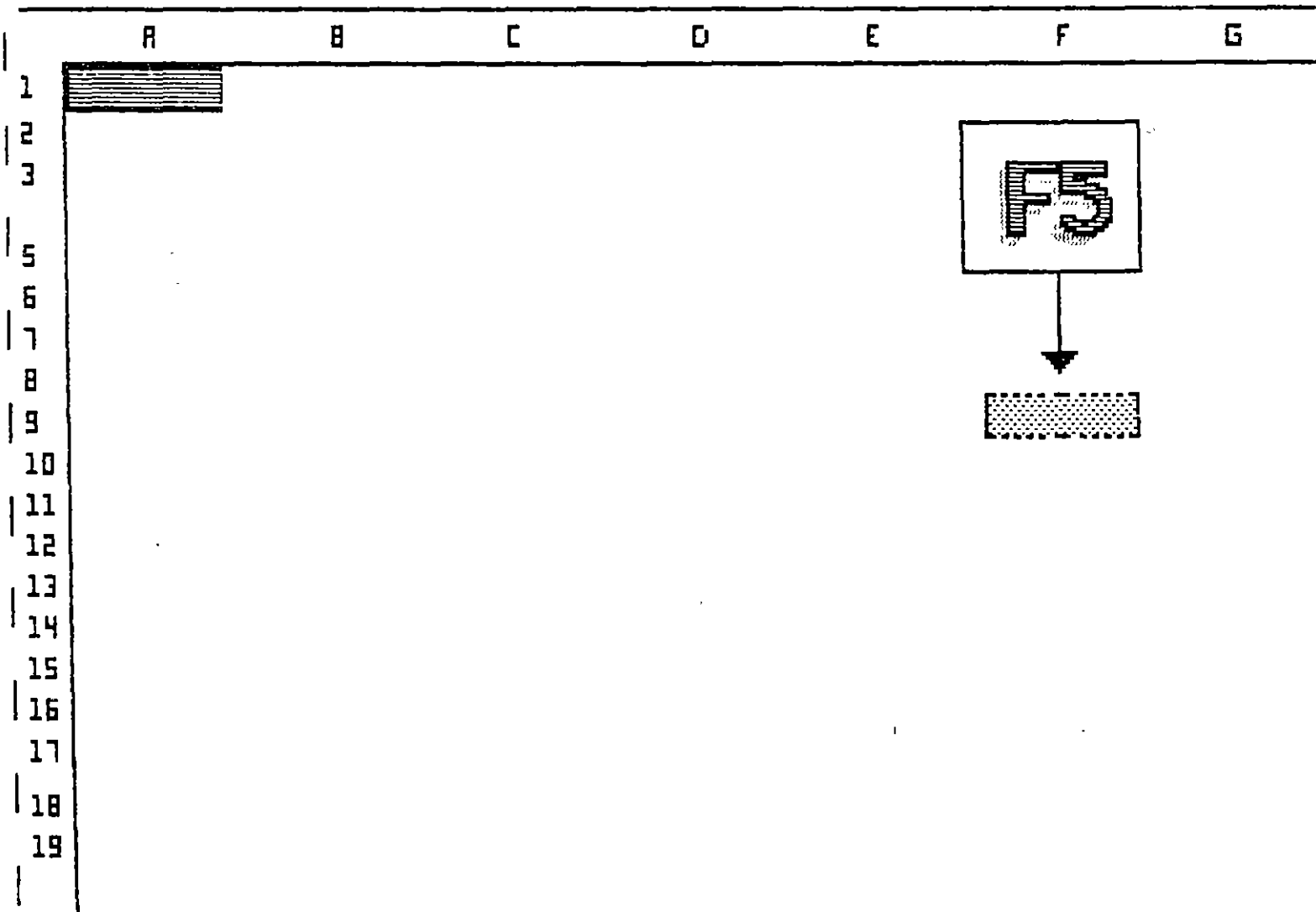
MOVIMIENTO DE LA CELDA

CURSOR CON LA TECLA





F5 (GOTO)

MENSAJE CUANDO PRESIONAMOS F5
A1: ENTER ADDRESS TO GO TO: F9 ←

READY



REGLAS PARA ENTRADA DE ETIQUETAS

- SE ASUME QUE SE ESTA ESCRIBIENDO UNA ETIQUETA SI EL PRIMER CARACTER QUE SE INTRODUCE EN LA CELDA NO ES NINGUNO DE LOS CARACTERES RESERVADOS PARA NUMEROS O FORMULAS.
- EXISTEN PREFIJOS DE ETIQUETAS QUE LE INDICAN A 1-2-3 COMO MOSTRAR LA ETIQUETA ALINEADA DENTRO DE LA CELDA.
 - SI COMIENZA CON  SE JUSTIFICA A LA IZQUIERDA.
 - SI COMIENZA CON  SE JUSTIFICA A DERECHA.
 - SI COMIENZA CON  LA ETIQUETA ES CENTRADA.
 - SI COMIENZA CON  EL TEXTO SE REPITE HASTA LLENAR LA CELDA.
- EL INDICADOR DE MODO CAMBIARA DE READY A LABEL.

PERFILLO DE ETIQUETAS (2)

R1: "SUELDO ←

READY

	A	B	C	D	E	F	G
1	SUELDO						
2							
3							
4							
5							
6							
7							
8							
9							
10							
12							
13							
14							
15							
16							
17							
18							
19							
20							

PREPILIO DE ETIQUETAS (1)

RI: 'NOMBRE'←

READY

	A	B	C	D	E	F	G
1	NOMBRE---						
2							
3							
4							
5							
6							
7							
8							
9							
10							
11							
12							
13							
14							
15							
16							
17							
18							
19							
20							

REGLAS PARA LA ENTRADA DE NUMEROS Y FORMULAS

- SE ASUME QUE SE ESTA PENETRANDO UN NUMERO O FORMULA SI SE COMIENZA CON ALGUNO DE LOS SIGUIENTES CARACTERES:

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

- SE OBSERVARA QUE EL INDICADOR DE MODO CAMBIARA DE MODO READY A VALUE .
- LAS REGLAS PARA INTRODUCIR NUMEROS SON:
 - 1.- UN NUMERO PUEDE COMENZAR CON UN DIGITO (0-9), +, -, \$.
 - 2.- UN NUMERO QUE TERMINA CON % LO DIVIDE ENTRE 100.
 - 3.- UN NUMERO DEBE TENER SOLO UN PUNTO DECIMAL.
 - 4.- NO SE DEBE USAR COMAS O ESPACIOS EN UN NUMERO.
 - 5.- SE PUEDEN INTRODUCIR NUMEROS EN NOTACION CIENTIFICA.

PERFILLO DE ENQUETAS (A)

91: CLAVE 4-1

READY

R B C D E F G

1 -- CLAVE --

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

PERFILLO DE ETIQUETAS (/)

READY

R1: /RFE4

	R	B	C	D	E	F	G
1	R	F	C	R	F	C	R
2							
3							
4							
5							
6							
7							
8							
9							
10							
11							
12							
13							
14							
15							
16							
17							
18							
19							
20							

RANGOS

	A	B	C	D	E	F	G	
1								
2								
3								
4								
5								
6								
7								
8								
9								
10								
11								
12								
13								
14								
15								
16								
17								
18								
19	ESTOS SON RANGOS				ESTOS NO SON RANGOS			
20								

INTRODUCIENDO FORMULAS

Objetivo Particular: Introducir fórmulas en celdas, fórmulas que contengan referencias absolutas y relativas a otras celdas.

Una fórmula es una instrucción para que 1-2-3 calcule un número. Es posible introducir una fórmula en cualquier celda de la hoja electrónica de trabajo. Por definición, una fórmula puede contener números y operadores. Los números pueden ser celdas simples o rangos de celdas que contienen a su vez números o fórmulas. Los operadores pueden ser lógicos o matemáticos.

OPERADORES MATEMATICOS

Los operadores matemáticos relacionan valores que se encuentran en celdas y son los siguientes:

<u>OPERADOR</u>	<u>OPERACION</u>	<u>PRECEDENCIA</u>
^	Exponenciación	7
-	Negativo	6
+	Positivo	6
*	Multiplicación	5
/	División	5
+	Adición	4
-	Substracción	4

Las operaciones se efectúan siempre siguiendo el orden de precedencia: las operaciones con operadores de precedencia superior son efectuadas primero. Las operaciones con orden de precedencia igual son ejecutadas de izquierda a derecha.

El orden en que se ejecutan las instrucciones puede ser modificado utilizando los paréntesis efectuándose primero aquellas operaciones que se encuentran dentro de los paréntesis. Dentro de los paréntesis se sigue el mismo orden de precedencia para las operaciones.

Se pueden manejar números negativos precediendo a los números con el signo (-).

OPERADORES LOGICOS.

En este caso se tienen expresiones que tienen que ser evaluadas como verdaderas (TRUE) o como falsas (FALSE). Dichas expresiones son llamadas fórmulas ó expresiones lógicas. Los operadores que combinan los números son llamados operadores lógicos que son los siguientes:

<u>OPERADOR</u>	<u>SIGNIFICADO</u>	<u>PRECEDENCIA</u>
=	Igual	3
<	Menor que	3
>	Mayor que	3
<=	Menor ó igual que	3
>=	Mayor ó igual que	3
<>	Diferente que	3
#NOT#	Negación	2
#AND#	"Y" lógico	1
#OR#	"O" lógico	1

FUNCIONES (@).

En LOTUS 1-2-3 es posible especificar un número en una fórmula directamente, simplemente tecleando los dígitos ó se puede especificar indirectamente utilizando una dirección de alguna celda. Existe aún otra forma de especificar un número en una fórmula que esta utilizando una función @. En el sentido estricto una función @ es una fórmula que va a ser colocada dentro de otra fórmula.

Todas las funciones comienzan con el carácter @ seguida por el nombre de la función y a continuación tenemos los números que son los argumentos de la función entre paréntesis y los argumentos individuales separados por comas. Algunas funciones permiten que se especifiquen rangos de celdas como argumento, en este caso LOTUS 1-2-3 usa todos los números almacenados en el rango para calcular el valor de la función. Por ejemplo:

@SUM(A1..A15)

@AVG(A1,A2,A3,A4,A5,A20..A40)

WORKSHEET

WORKSHEET RANGE COPY MOVE FILE PRINT GRAPH DATA SYSTEM QUIT

GLOBAL INSERT DELETE COLUMN ERASE TITLES WINDOW STATUS PAGE-BREAK

COLUMN ROW

HORIZONTAL VERTICAL SYNC UNSYNC CLEAR

COLUMN ROW

BOTH HORIZONTAL VERTICAL CLEAR

SET-WIDTH RESET-WIDTH HIDE DISPLAY

FORMAT LABEL-PREFIX COLUMN-WIDTH RECALCULATION PROTECTION DEFAULT ZERO

LEFT RIGHT CENTER

ENABLE DISABLE

NATURAL COLUMNWISE ROWWISE AUTOMATIC MANUAL ITERATION

FIXED SCIENTIFIC CURRENCY , GENERAL +/- PERCENT DATE TEXT HIDDEN

TIME

PRINTER DIRECTORY STATUS UPDATE OTHER QUI

INTERNATIONAL HELP CLOCK

PUNCTUATION CURRENCY DATE TIME QUI

INTERFACE AUTO-LF LEFT RIGHT TOP BOTTOM PAGE-LENGTH WAIT SETUP NAME QUI

RANGE

WORKSHEET RANGE COPY MOVE FILE PRINT GRAPH DATA SYSTEM QUIT

FORMAT LABEL ERASE NAME JUSTIFY PROTECT UNPROTECT INPUT VALUE TRANSPOSE

LEFT RIGHT CENTER

CREATE DELETE LABELS RESET TABLE

FIXED SCIENTIFIC CURRENCY , GENERAL +/- PERCENT DATE TEXT HIDDEN RESET

COPY

Y

MOVE

WORKSHEET RANGE COPY MOVE FILE PRINT GRAPH DATA SYSTEM QUIT

ENTER RANGE TO COPY FROM:

ENTER TO COPY TO:

ENTER RANGE TO MOVE FROM:

ENTER TO MOVE TO:

.11:
WORKSHEET RANGE COPY MOVE FILE PRINT GRAPH DATA SYSTEM QUIT
COPY A CELL OR RANGE OF CELLS

	A	B	C	D	E	F	G
1	100	600					
2	200	700					
3	300	800					
4	400	900					
5	500	1000					
6							
7							
8							
9							
10							
11					100	600	
12					200	700	
13					300	800	
14					400	900	
15					500	1000	
16							
17							
18							
19							
20							

RANGO FUENTE

DESTINO

COPY

/COPY
ENTER RANGE TO COPY FROM: A1..B5 ←
ENTER RANGE TO COPY TO: E11 ←

MOVE

A*

WORKSHEET RANGE COPY **MOVE** FILE PRINT GRAPH DATA QUIT

MENU

MOVE A CELL OR RANGE OF CELLS

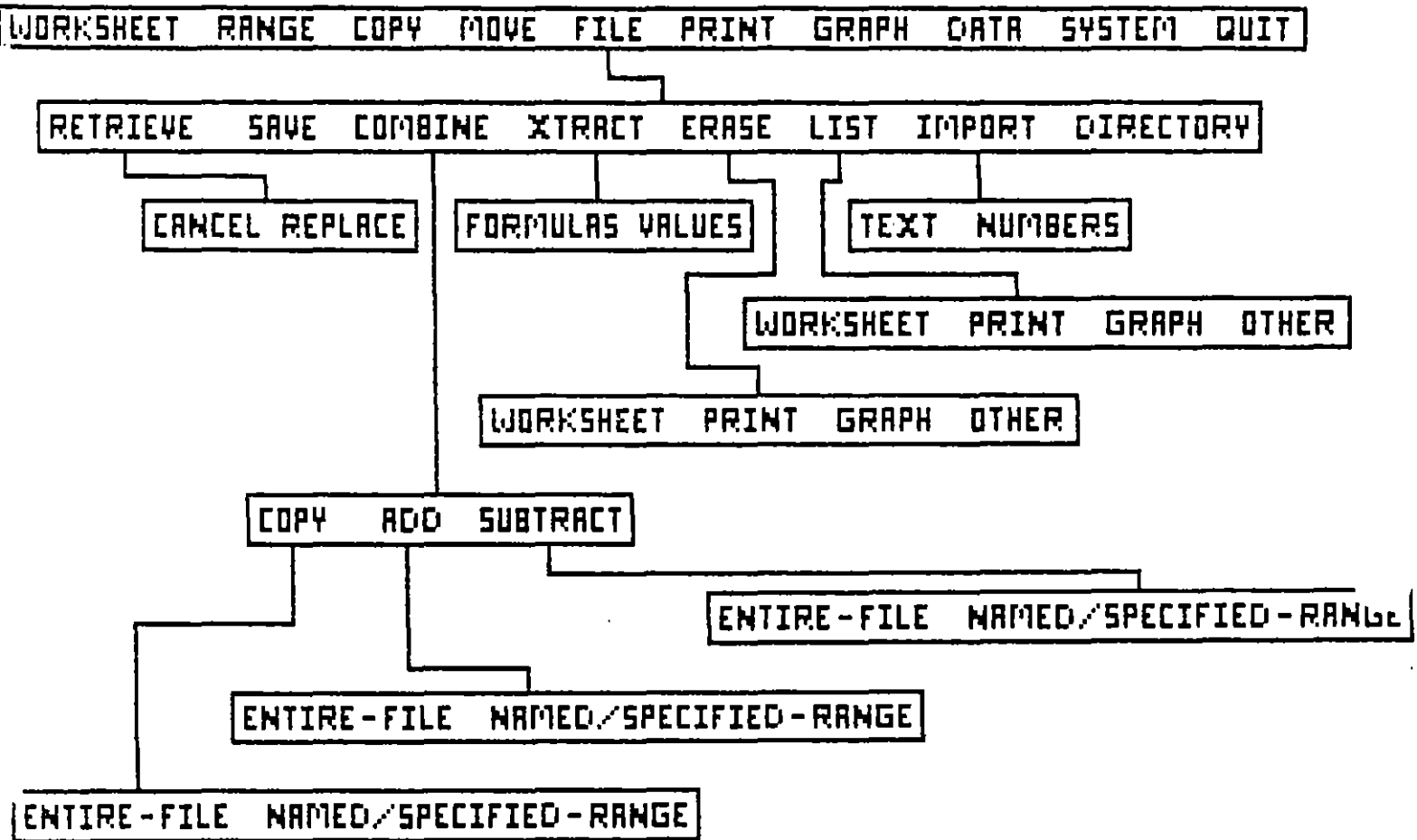
	A	B	C	D	E	F	G
1	100	600					
2	200	700					
3	300	100	600				
4	400	200	700				
5	500	300	100	600			
6		400	200	700			
7		500	300	100	600		
8			400	200	700		
9			500	300	100	600	
10				400	200	700	
11				500	300	800	
12					400	980	
1					500	1000	
14							
15							
16							
17							
18							
19							
20							

/MOVE

ENTER RANGE TO MOVE FROM: A1..B3 ↵

ENTER RANGE TO MOVE TO: E3 ↵

FILE



PRINT

WORKSHEET RANGE COPY MOVE FILE PRINT GRAPH DATA SYSTEM QUIT

PRINTER FILE

RANGE LINE PAGE OPTIONS CLEAR ALIGN GO QUIT

ALL RANGE BORDERS FORMAT

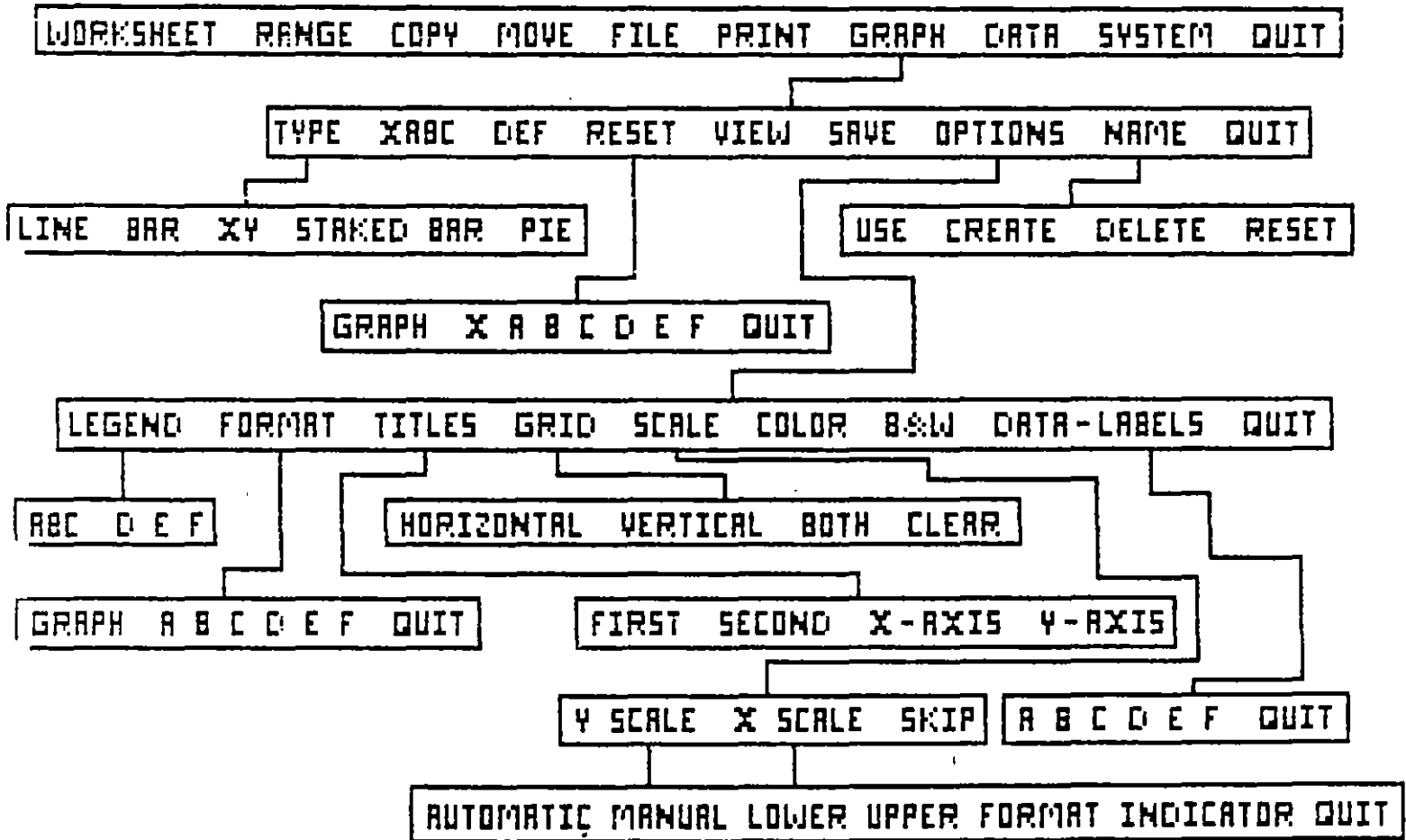
HEADER FOOTER MARGINS BORDERS SETUP PG-LENGTH OTHER QUIT

LEFT RIGHT TOP BOTTOM

COLUMNS ROWS

AS-DISPLAYED CELL-FORMULAS FORMATTED UNFORMATTED

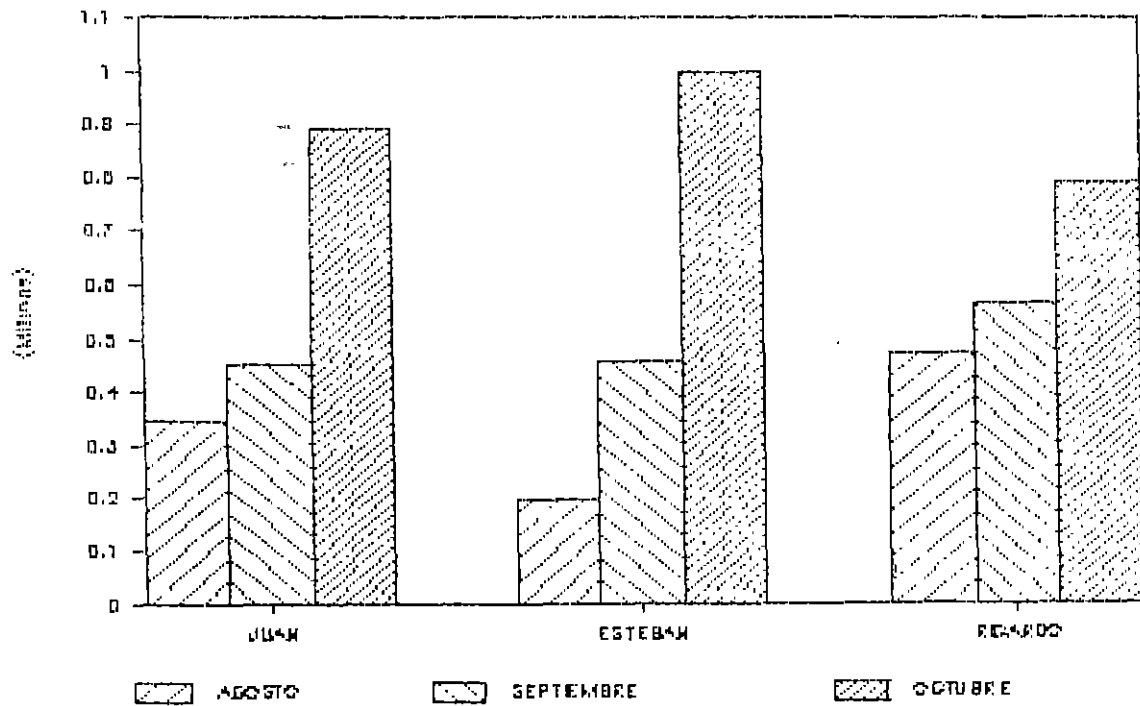
GRAPH



GRAFICA TIPO

BARRAS

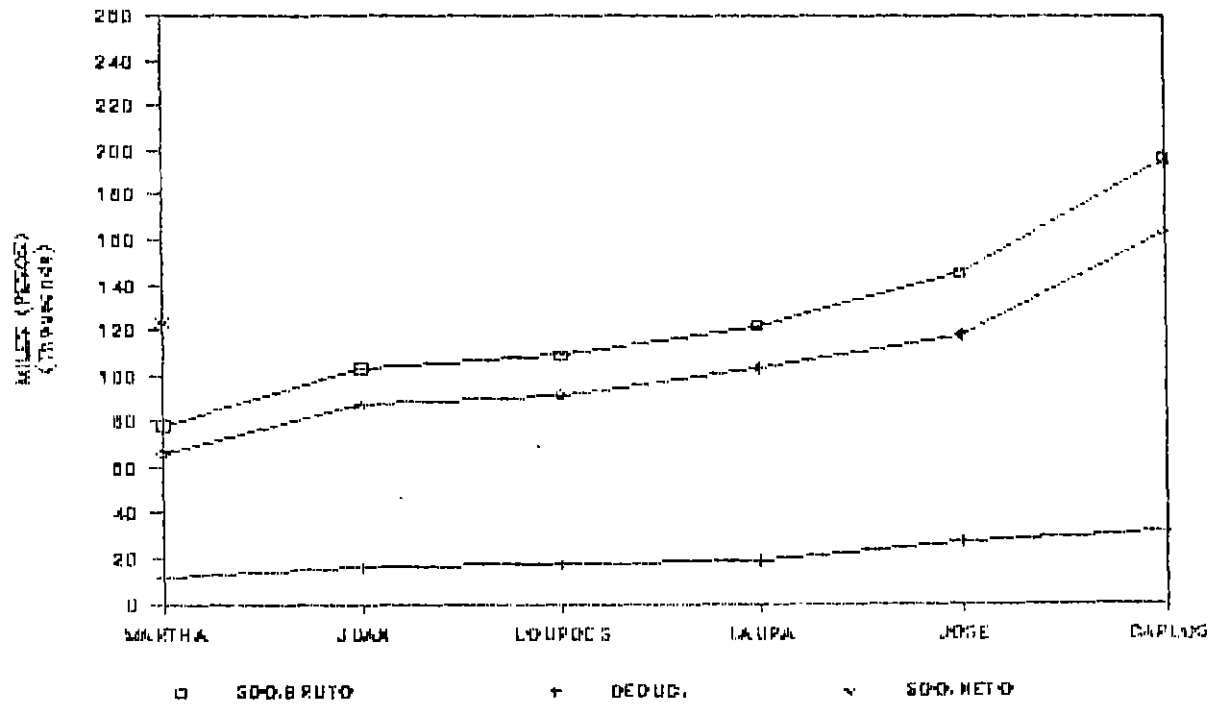
ANALISIS DE VENTAS



GRAFICA TIPO

LINEA

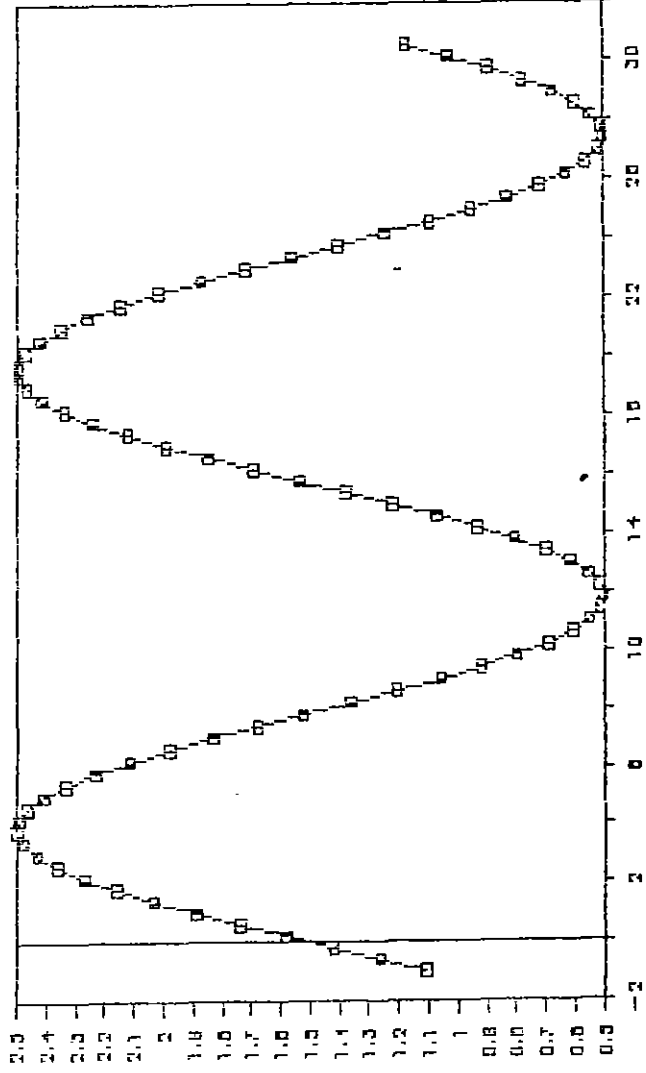
PLANTILLA DE PERSONAL



GRAFICA

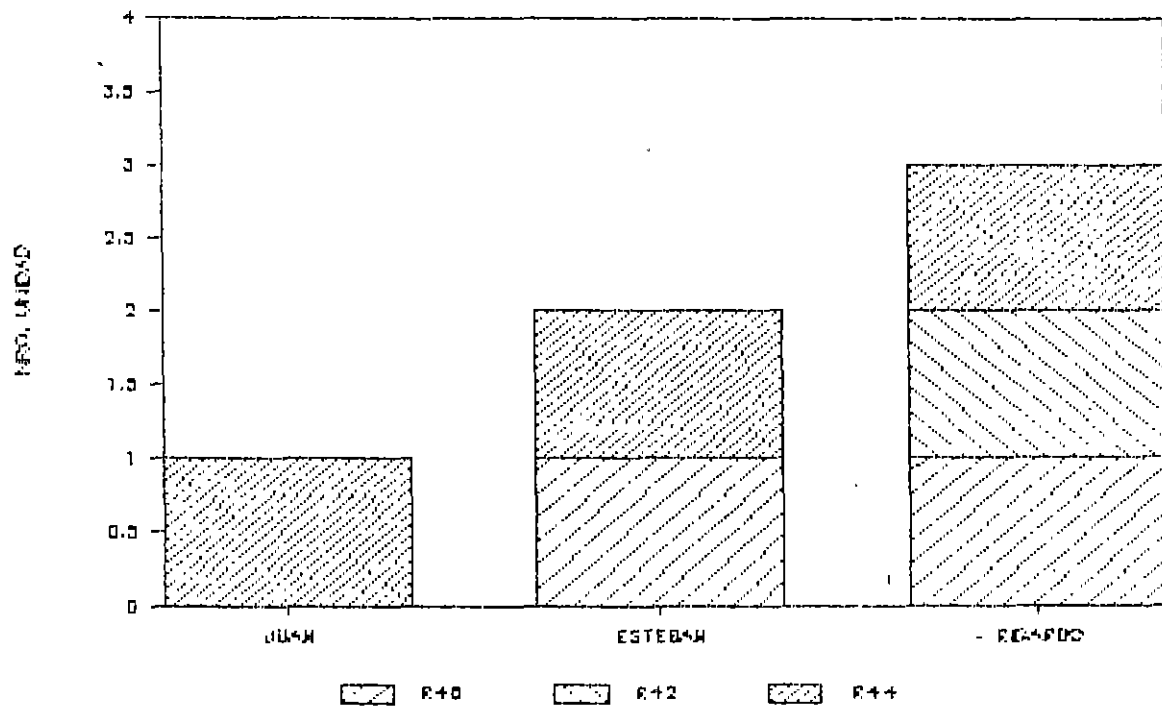
TIPO

XY



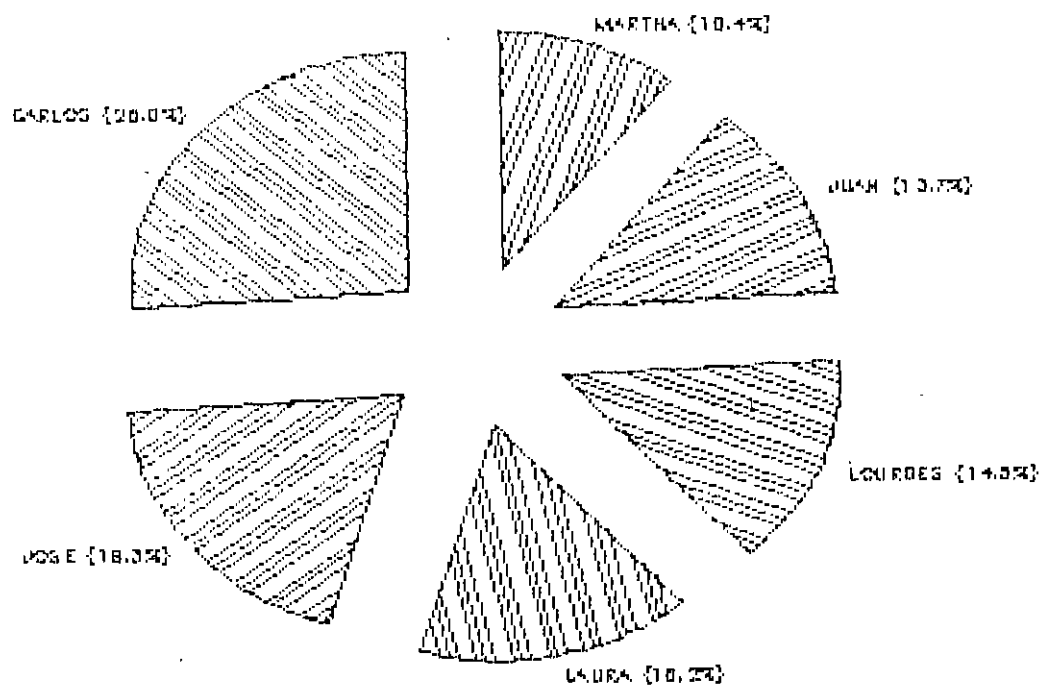
GRAFICA TIPO STACKED-BAR

ANALISIS DE PRODUCTIVIDAD



GRAFICA TIPO

PIE



MENU DE PRINT GRAPH

IMAGE - SELECT SETTINGS GO ALIGN PAGE EXIT

IMAGE HARDWARE ACTION SAVE RESET QUIT

SIZE FONT RANGE - COLORS QUIT

PAUSE EJECT QUIT

1 2 X A B C D E F

FULL HALF MANUAL QUIT

TOP LEFT WIDTH HEIGHT ROTATION QUIT

GRAPHS - DIRECTORY FONTS - DIRECTORY INTERFACE PRINTER SIZE - PAPER QUIT

1 2 3 4 5 6 7 8

LENGTH WIDTH QUIT

1 2 3 4 5 6 7 8 9

DATA

WORKSHEET RANGE COPY MOVE FILE PRINT GRAPH DATA SYSTEM QUIT

FILL TABLE SORT QUERY DISTRIBUTION MATRIX REGRESSION PARSA

1 2 RESET

INVERT MULTIPLY

DATA-RANGE- PRIMARY-KEY SECONDARY-KEY RESET GO QUIT

INPUT CRITERION OUTPUT FIND EXTRACT UNIQUE DELETE RESETE QUIT

CANCEL DELETE

X-RANGE Y-RANGE OUTPUT-RANGE INTERCEPT RESET GO QUIT

COMPUTE ZERO

NOMBRE DEL CAMPO

CAMPO DE ENTRADA

	A	B	C	D	E
1			PLANTILLA		
2			DE		
3			PERSONAL		
4	NOMBRE DEL EMPLEADO	PUESTO	AREA	PERCEPCIONES	DEDUCCIONES
5	HUERTA ESPEJO LIZBETH	25	APOYO TEC.	\$1,250,000.00	\$135,000.00
6	GARCIA GAMA SANDRA	30	R.SISTEMAS	\$1,500,000.00	\$200,000.00
7	GUZMAN LARA JOSE	17	TEC. PLAN.	\$1,000,000.00	\$100,000.00
8	JIMENEZ ROCHA MAURICIO	19	PUBLICIDAD	\$1,300,000.00	\$125,000.00
9	MARTINEZ ORTIZ EDUARDO	20	D.SISTEMAS	\$1,150,000.00	\$105,000.00
10	CANO BALTAZAR GABRIELA	11	PROGRAMAC.	\$1,000,000.00	\$105,000.00
11	MORENO GUERRERO ANA	12	CAPTURA	\$1,000,000.00	\$115,000.00
12	ALFARO FRUTOS EDUARDO	18	ORGANIZAC.	\$1,150,000.00	\$125,000.00
13					
14					
15			REGISTRO		
16					
17					
18					
19					
20					

BASE DE DATOS



FACULTAD DE INGENIERIA, UNAM.
DIVISION DE EDUCACION CONTINUA

INTRODUCCION A LAS COMPUTADORAS PERSONALES

CARACTERES ASCII

FEBRERO 1995

TABLAS DE CARACTERES ASCII

Valor decimal	Valor hexadecimal	Carácter de control	Carácter
0	00	NUL	Nulo
1	01	SOH	☺
2	02	STX	☹
3	03	ETX	♥
4	04	EOT	♦
5	05	ENQ	♣
6	06	ACK	♠
7	07	BEL	Zumbido, Pitido, "bip"
8	08	BS	☐
9	09	HT	Tabulación
10	0A	LF	Avance de línea
11	0B	VT	Cursor a posición inicial
12	0C	FF	Avance de página
13	0D	CR	Retorno de carro, introducir
14	0E	SO	🎵
15	0F	SI	⚙
16	10	DLE	▶
17	11	DC1	◀
18	12	DC2	↕

Valor decimal	Valor hexadecimal	Carácter de control	Carácter
19	13	DC3	!!
20	14	DC4	¶
21	15	NAK	§
22	16	SYN	≡
23	17	ETB	⏏
24	18	CAN	↑
25	19	EM	↓
26	1A	SUB	→
27	1B	ESC	←
28	1C	FS	Cursor a derecha
29	1D	GS	Cursor a izquierda
30	1E	RS	Cursor arriba
31	1F	US	Cursor abajo
32	20		Espacio
33	21		!
34	22		..
35	23		#
36	24		\$
37	25		%
38	26		&
39	27		.
40	28		(
41	29)
42	2A		•
43	2B		+
44	2C		.
45	2D		-
46	2E		.
47	2F		/
48	30		0
49	31		1
50	32		2
51	33		3
52	34		4
53	35		5
54	36		6
55	37		7
56	38		8
57	39		9
58	3A		:
59	3B		;
60	3C		<
61	3D		=
62	3E		>
63	3F		?
64	40		ⓐ
65	41		A
66	42		B
67	43		C
68	44		D
69	45		E
70	46		F

Valor decimal	Valor hexadecimal	Carácter
71	47	G
72	48	H
73	49	I
74	4A	J
75	4B	K
76	4C	L
77	4D	M
78	4E	N
79	4F	O
80	50	P
81	51	Q
82	52	R
83	53	S
84	54	T
85	55	U
86	56	V
87	57	W
88	58	X
89	59	Y
90	5A	Z
91	5B	[
92	5C	\
93	5D]
94	5E	^
95	5F	.
96	60	:
97	61	a
98	62	b
99	63	c
100	64	d
101	65	e
102	66	f
103	67	g
104	68	h
105	69	i
106	6A	j
107	6B	k
108	6C	l
109	6D	m
110	6E	n
111	6F	o
112	70	p
113	71	q
114	72	r
115	73	s
116	74	t
117	75	u
118	76	v
119	77	w
120	78	x
121	79	y
122	7A	z

Valor decimal	Valor hexadecimal	Carácter
123	7B	
124	7C	
125	7D	
126	7E	
127	7F	
128	80	␣
129	81	Ç
130	82	ü
131	83	é
132	84	à
133	85	â
134	86	ä
135	87	ç
136	88	è
137	89	é
138	8A	ê
139	8B	ë
140	8C	ì
141	8D	í
142	8E	î
143	8F	Ï
144	90	Ä
145	91	É
146	92	æ
147	93	AE
148	94	o
149	95	õ
150	96	ö
151	97	u
152	98	ù
153	99	ÿ
154	9A	Û
155	9B	Ü
156	9C	é
157	9D	£
158	9E	¥
159	9F	Pl
160	A0	ƒ
161	A1	á
162	A2	í
163	A3	ó
164	A4	ù
165	A5	ñ
166	A6	Ñ
167	A7	o
168	A8	o
169	A9	o
170	AA	l
171	AB	l
172	AC	1/2
173	AD	1/4

Valor decimal	Valor hexadecimal	Carácter
174	AE	
175	AF	“
176	B0	”
177	B1	⋄
178	B2	⋄
179	B3	—
180	B4	⋄
181	B5	⋄
182	B6	⋄
183	B7	⋄
184	B8	⋄
185	B9	⋄
186	BA	⋄
187	BB	=
188	BC	⋄
189	BD	⋄
190	BE	⋄
191	BF	⋄
192	C0	⋄
193	C1	⋄
194	C2	⋄
195	C3	⋄
196	C4	⋄
197	C5	⋄
198	C6	+
199	C7	⋄
200	C8	⋄
201	C9	⋄
202	CA	⋄
203	CB	⋄
204	CC	⋄
205	CD	⋄
206	CE	⋄
207	CF	⋄
208	D0	⋄
209	D1	⋄
210	D2	⋄
211	D3	⋄
212	D4	⋄
213	D5	⋄
214	D6	⋄
215	D7	⋄
216	D8	⋄
217	D9	⋄
218	DA	⋄
219	DB	⋄
220	DC	⋄
221	DD	⋄
222	DE	⋄
223	DF	⋄
224	E0	⋄
225	E1	⋄

Valor decimal	Valor hexadecimal	Carácter
226	E2	┌
227	E3	┐
228	E4	└
229	E5	┘
230	E6	◊
231	E7	◊
232	E8	◊
233	E9	◊
234	EA	◊
235	EB	◊
236	EC	◊
237	ED	◊
238	EE	◊
239	EF	◊
240	F0	◊
241	F1	◊
242	F2	◊
243	F3	◊
244	F4	◊
245	F5	◊
246	F6	◊
247	F7	◊
248	F8	◊
249	F9	◊
250	FA	◊
251	FB	◊
252	FC	◊
253	FD	◊
254	FE	◊
255	FF	◊

(espacio en blanco 'FF')



FACULTAD DE INGENIERIA, UNAM.
DIVISION DE EDUCACION CONTINUA

INTRODUCCION A LAS COMPUTADORAS PERSONALES

PROCESADOR DE TEXTO

FEBRERO 1995

INTRODUCCION

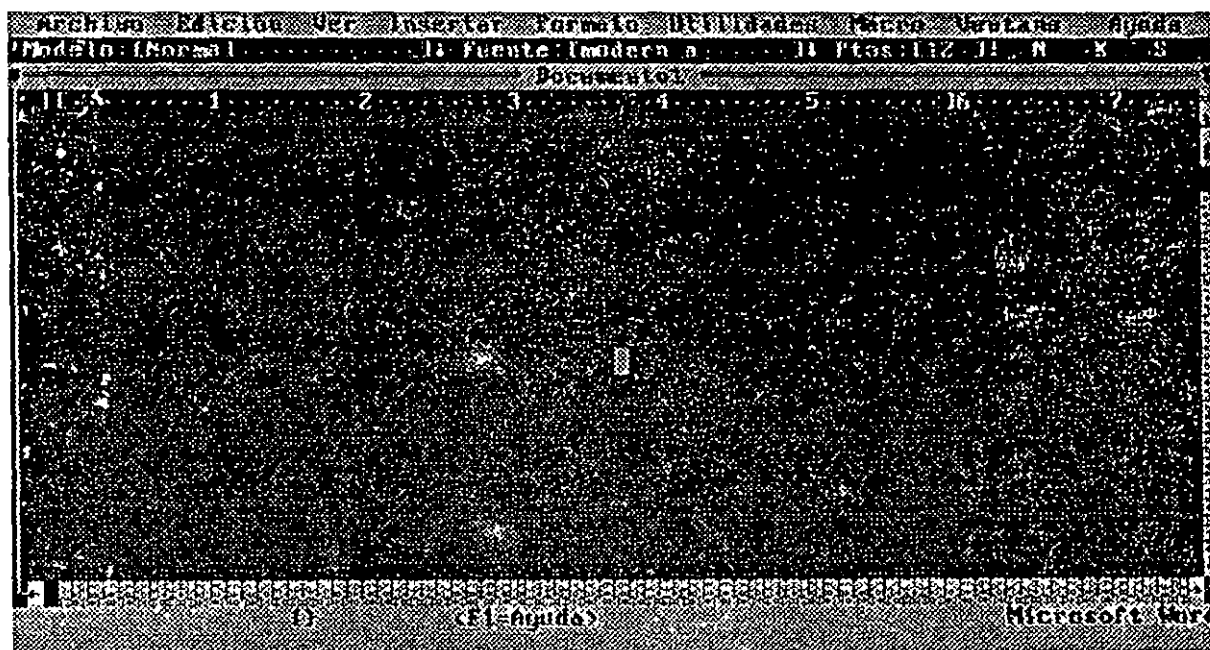
Word es un procesador de palabras que lo podríamos clasificar entre los más completos y populares del mundo. En el, podemos combinar nuestros documentos con Hojas de cálculo, gráficos, imágenes y texto de cualquier otros procesadores.

Para iniciar Word desde su directorio de trabajo procedemos de la siguiente manera:

- I.- Estando en en indicador de sistema operativo, tecleamos la palabra WORD y presionamos ENTER.

```
C:\WORD>WORD
```

Y Word nos muestra su pantalla de trabajo...



- A.- Area de Trabajo en donde escribirá su documento en la ventana.
- B.- Barra de Menús que contiene los comandos para trabajar con Word.
- C.- Barra de Estado que presenta información acerca del documento activo.
- D.- Barra de Mensajes en la cual podrá ver los mensajes de Word.
- E.- Con F1 o haciendo clic con el mouse consigue ayuda.
- F.- Barra de Título.
- G.- Barras de Desplazamiento
- H.- Bordos de la Ventana.
- I.- Marca de Fin.

DESCRIPCION DE LA PANTALLA DE WORD



AREA DE TRABAJO:

Es la hoja en blanco donde aparece el cursor junto con la marca de fin de documento. En esta área podrá escribir su documento y conformando a su gusto los datos que vaya insertando.

BARRA DE MENUS:

En Word se revisa y se da formato a documentos, seleccionando comandos de la barra de menús. Los comandos están agrupados por relaciones de función, en un mismo menú. Por ejemplo, los comandos utilizados para Cinta, Regla, Barra de Herramientas se encuentran en el Menú Ver.

BARRA DE MENSAJES:

Presenta información acerca de su trabajo con Word, indicándole, por ejemplo, si el programa está realizando la tarea de imprimir, repaginar o guardar un documento, etc.,. También le indica lo que pasa cuando el cursor se encuentra posicionado en un comando específico.

BARRA DE ESTADO:

Le proporciona información básica acerca de el estado de trabajo de Word y también acerca del documento en el cual está trabajando. Por ejemplo:

- Número de página
- Número de sección
- Portapapeles
- Estado del Teclado
- Número de Línea
- Número de Columna
- Ayuda de Word

Números de Página, Sección, Columna y Línea.

Estos indicadores señalan la posición actual del cursor en el documento. El número de página indica en que página se encuentra trabajando. Si el documento tiene varias secciones, el número de sección le indica en que sección específica del documento se encuentra el cursor. El número de columna le indica la posición horizontal del cursor en la pantalla, a partir del margen izquierdo.

Portapapeles.

El portapapeles muestra el texto copiado o borrado por última vez. Los puntos suspensivos (...) indican que el texto es mayor que el espacio disponible en esta parte de la barra de estado.

Ayuda de Word.



F1=AYUDA Al presionar F1 o hacer clic con el mouse, Word presenta información sobre el comando con el que se encuentre trabajando.

Estado del Teclado.

Este indicador le recuerda que tiene activada una o más de las funciones siguientes, que están representadas en la pantalla con una abreviatura de dos letras.

INDICADORES:

Código	Tecla/Comando/Opción	Significado
BD	BLOQ DESPL	Teclas de dirección para desplazar.
BM	BLOQ MAYUS	Todas las letras en mayúsculas.
BN	BLOQ NUM	Escribir números con el teclado numérico.
DL	Utilidades	Dibujar líneas con el cursor.
EX	F8	Extender la selección.
GM	CTRL + F3	Grabar un macro.
ME	"Paso a Paso" en modo escalonado.	Ejecución de macros la caja de diálogo de Macro Ejecutar.
MR	Utilidades Marcas de revisión.	Marcas de revisión activadas.
MX	CTRL + F10	Maximizar la ventana.
PP	Ver Página	Modo de presentación de página.
SC	CTRL + MAYUS + F8	Seleccionar columnas.
SO	INS	Modo de sobrescritura.

BARRA DE TITULO:

Presenta el nombre del documento activo. Antes de guardar por primera vez un documento tendrá un nombre provisional que Word pone automáticamente (Documento x), dónde x es un número que comienza en uno y aumenta de uno en uno cada vez que crea archivos nuevos, o bien que se abran.

BARRAS DE DESPLAZAMIENTO:

Nos permiten desplazarnos a diferentes partes del documento para ver partes que no se veían en la pantalla. Estas son una referencia posicional con relación a cualquier parte de un documento. Existen dos tipos de barras de desplazamiento: Vertical y Horizontal.

BORDES DE LA VENTANA:

Separan el documento del resto de la pantalla.

MARCAS DE FIN:

La marca de fin (" _ ") indica el final del documento. No es posible borrar o modificar esta marca.

CREACION DE UN DOCUMENTO CON WORD.



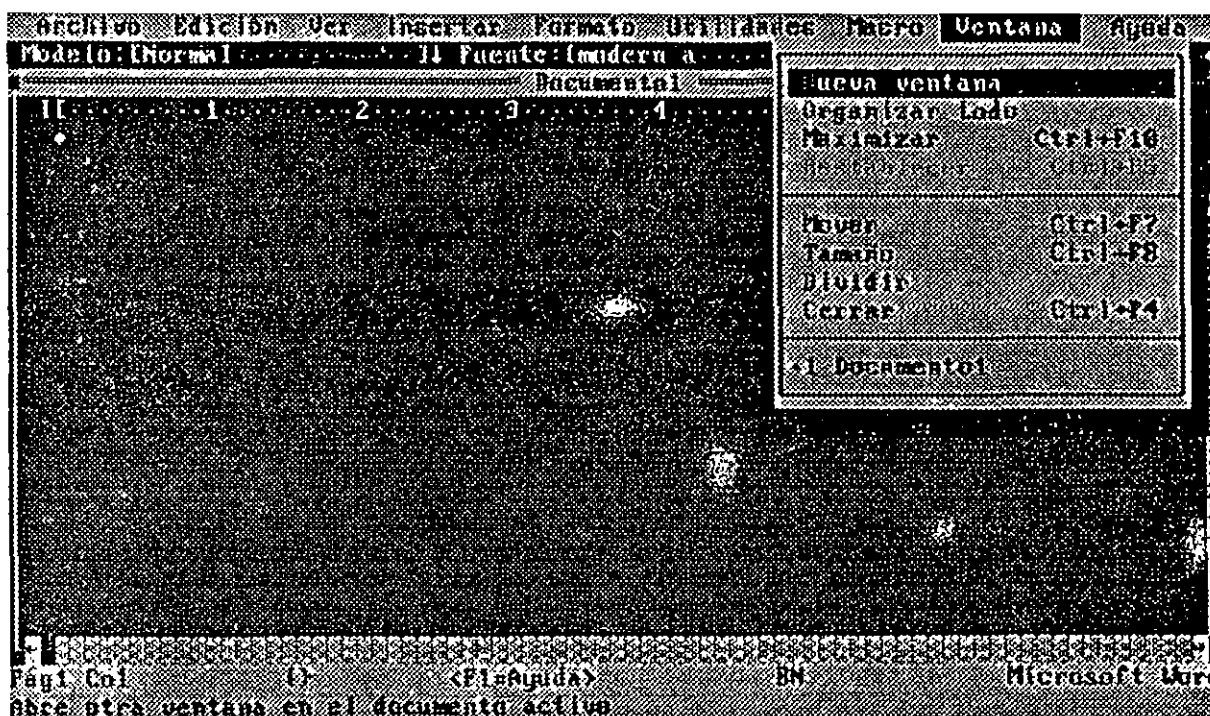
La creación de un documento con WORD consta generalmente de los siguientes pasos:

- Crear o abrir un documento.
- Escribir y dar formato al texto.
- Revisar el texto y el formato.
- Guardar el documento.
- Imprimir el documento.
- Cerrar el documento y/o salir de word.

Selección de comandos.

Un comando es una instrucción que se le da a WORD para que realice una acción determinada. Para escoger un comando, tendrá que seleccionar uno de los menús de la Barra de Menús y escoger el comando deseado dentro del menú.

La ventana de word en pantalla se verá de la siguiente forma:



- A = Barra de Menús: presenta los nombres de Menús.
- B = Usted escogerá comandos del Menú.
- C = La Barra de Información le ofrece información sobre los comandos.
- D = Los comandos ejecutan acciones.

Selección de Comandos en la Barra de Menús:



- 1.- Presionar la tecla ALT para activar la Barra de Menús. Una letra en cada nombre de menú aparecerá resaltada. Esta letra resaltada también se llama Tecla de Acceso Rápido.
- 2.- Presionar la letra resaltada del menú que desee utilizar. Aparecerán los comandos de ese menú.
- 3.- Presionar la letra resaltada del comando que desee utilizar.
Si el nombre del comando aparece seguido de puntos suspensivos (...) significará que al escoger el comando aparecerá una caja de diálogo. Se utiliza una caja de diálogo para suministrar información que WORD necesita para ejecutar el comando. Si el nombre del comando no aparece seguido de puntos suspensivos, no aparecerá la caja de diálogo, sino que Word ejecutará el comando inmediatamente.

Cancelar la Barra de Menú.

Para cancelar la Barra de Menús Presione la Tecla ESC.

Escribir Texto.

En word el texto puede contener letras, números, espacios y símbolos especiales. Escribir texto es muy parecido a utilizar una máquina de escribir, no obstante existen algunas diferencias:

-Un subrayado intermitente, denominado cursor, indica su posición actual dentro del documento. Mientras escribe, el cursor se mueve hacia la derecha.

-Si escribe texto nuevo en un lugar en el que ya existe texto, el texto nuevo se inserta en la posición del cursor y el texto existente se desplaza hacia la derecha.

-No presione la tecla ENTER al final de las líneas. El texto que escriba pasará automáticamente a la línea siguiente si no cabe dentro de los márgenes de la página. Esto se llama Ajuste automático de líneas. Si agrega o borra texto, el ajuste automático de líneas cambiará las líneas de texto para acomodar el texto nuevo o borrado.

Insertar Texto.

- 1.- Utilice las teclas de dirección para situar el cursor en el lugar en el que desee insertar el texto o señale con el mouse en dicho lugar y haga clic con el botón izquierdo.
- 2.- Comience a escribir. Word insertará el texto. El cursor y cualquier texto existente se moverán hacia la derecha.

Escribir encima de un texto existente.



Escribir texto nuevo encima de un texto existente (sobrescribir) es otra de las opciones de Word.

- 1.- Presione la tecla INS para activar la función de sobrescritura. Las letras “SO” aparecerán en la barra de estado al pie de la pantalla indicando el modo de sobrescritura.
- 2.- Comience a escribir. Word escribirá el texto nuevo encima del texto ya existente.
- 3.- Presione INS nuevamente para desactivar el modo de sobrescritura.

Corregir un error.

Presione la tecla Retroceso o Backspace. Esta tecla borra los caracteres al retroceder el cursor sobre ellos, menos cuando esté activada la función de sobre escritura.

Borrar texto permanente.

- 1.- Seleccione el texto que desea borrar.
- 2.- Presione la tecla SUPR o DEL para eliminar el texto del documento y dejar intacto el portapapeles.

También podemos usar la tecla de Retroceso para borrar texto permanente.

Insertar un espacio de no separación.

A veces deseará evitar que se produzca un salto de línea entre dos palabras específicas. Por ejemplo, el nombre y el apellido de un cliente o una fórmula matemática para conservarla en la misma línea; quizá desea que Word evite espacios entre caracteres en una instrucción. En tales casos podrá usar el espacio de no separación.

- 1.- Presione CTRL + MAYUSCULAS + BARRA ESPACIADORA, para insertar un espacio de no separación entre palabras que desee conservar siempre en la misma línea.

Unir y Dividir Párrafos.

Borrando la marca de fin de párrafo que se encuentra al final de un párrafo, éste se une con el párrafo siguiente. El primer párrafo tomará los formatos del segundo.



Unir dos Párrafos.

- 1.- Lleve el cursor sobre el primer carácter del segundo párrafo.
- 2.- Presione la tecla de Retroceso.
- 3.- Presione la barra espaciadora para insertar un espacio entre las frases en el lugar en el que se unieron los dos párrafos.

Dividir un Párrafo en dos.

- 1.- Lleve el cursor sobre el carácter que desee tener como primer carácter de párrafo.
- 2.- Presione ENTER.

Repetir lo que se acaba de escribir.

Esta función es de gran utilidad cuando desea insertar la misma frase en varios lugares del documento.

- 1.- Escriba el texto.
- 2.- Mueva el cursor al lugar donde desee insertar la misma frase o texto.
- 3.- Presione la tecla F4.
- 4.- Repita los pasos 2 y 3 en todos los lugares donde desee insertar el texto.

Iniciar un nuevo párrafo.

Un párrafo es cualquier texto y formato que preceda a un carácter especial llamado marca de párrafo. Esta marca contiene las instrucciones de formato, como márgenes y tabulaciones, correspondientes a ese párrafo.

- 1.- Presione ENTER.

Word insertará una marca de párrafo (esta puede ser visible o invisible).

Iniciar una nueva línea sin crear un nuevo párrafo.

Cuando desee que un texto se mantenga en un solo párrafo, por ejemplo en una tabla, una dirección o un poema, podrá hacerlo sin crear nuevos párrafos, manteniendo así el mismo formato de párrafo.

- 1.- Presione Mayúsculas+ENTER.

Word iniciará una línea nueva sin crear nuevo párrafo, insertando el símbolo de nueva línea.



Iniciar una página manualmente.

Un salto de página indica a la impresora a dónde debe terminar una página y iniciar otra. Word inicia páginas nuevas automáticamente cuando usted escribe un documento.

- 1.- Lleve el cursor al lugar dónde desee insertar un salto de página.
- 2.- Presione CTRL+ENTER.

Word presentará una línea punteada en la pantalla para indicar el salto de página. La línea de puntos que representa un salto de página automático es menos densa que la que representa el salto de página manual.

Seleccionar texto con el teclado.

Para seleccionar texto utilizando el teclado primero deberá pasar al texto utilizando cualquiera de las teclas mencionadas de desplazamiento.

Podrá utilizar la tecla de extensión F8 para seleccionar una unidad de texto (palabra, frase, párrafo o documento entero) y para extender la selección.

Seleccionar texto con el Mouse.

Para seleccionar texto utilizando el Mouse, primero coloque el cursor al inicio del texto que desea seleccionar y mantenga presionado el botón izquierdo del Mouse, arrastre el puntero hasta el otro extremo del texto y suelte el botón.

Seleccionar una columna.

- 1.- Mueva el cursor sobre el carácter situado en la esquina superior izquierda de la columna que desee seleccionar.
- 2.- Presione CTRL+MAYUS+F8 para activar la función de seleccionar columnas.
- 3.- Presione las teclas de dirección o sitúe el puntero del mouse y haga clic para extender la selección.
- 4.- Presione CTRL+MAYUS+F8 para desactivar la función de selección de

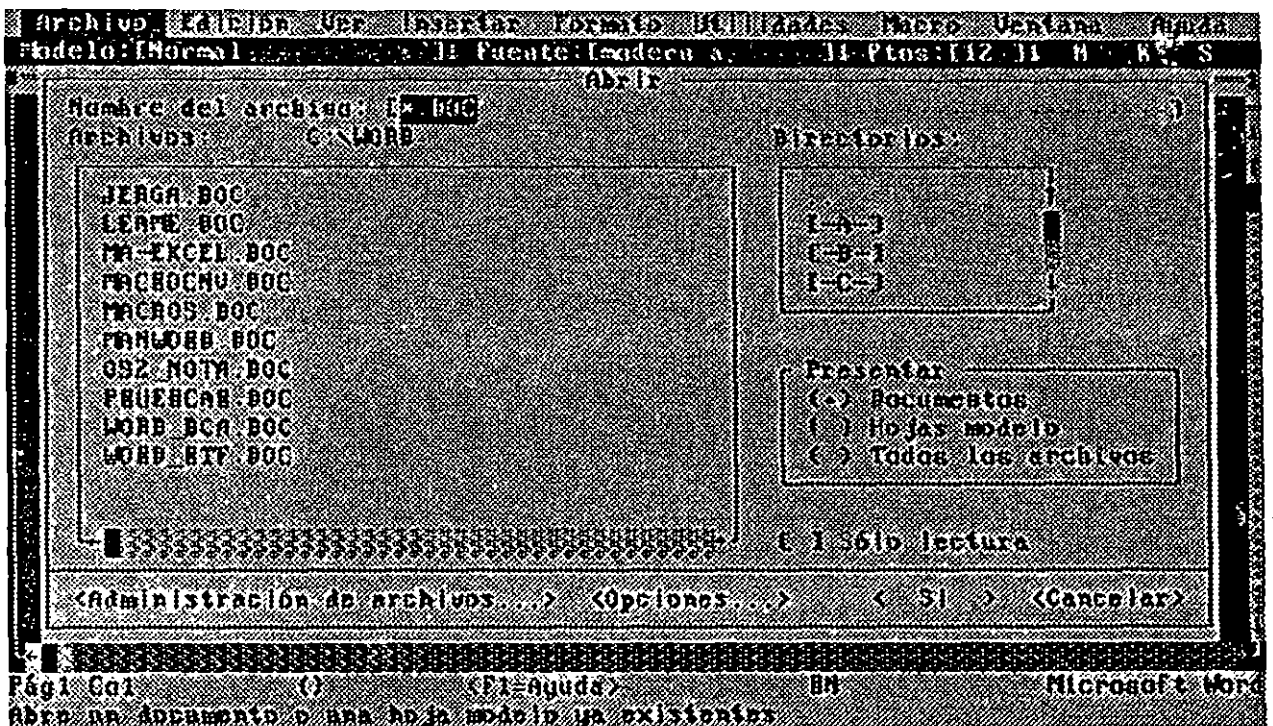


El Menú Archivo consta de los siguientes comandos:

COMANDO NUEVO:

Crea un documento nuevo.

Al iniciar WORD se creará un documento automáticamente. Podrá crear un documento en cualquier momento mientras trabaje con WORD. Al crear un documento, word le asigna un nombre temporal y lo presenta como título de ventana. Puede empezar a escribir inmediatamente.



- A.- La marca de fin indica el final del documento.
- B.- El nombre del documento aparece en la barra de título.
- C.- El cursor indica el lugar en el que se encuentra dentro del documento.

El texto que escriba se almacenará temporalmente en la memoria de la computadora. Si desea conservar el texto para utilizarlo posteriormente, deberá guardar el documento en un disco. Al guardar el documento deberá asignarle un nombre único.



- 1.- Escoja el comando Abrir del Menú Archivo.
- 2.- Haga uno de estos pasos:

-En el cuadro "nombre de Archivo", escriba el nombre del documento a recuperar.

-Sí el documento no se encuentra en la unidad de disco o directorio en uso, incluya la ruta de acceso con el nombre de archivo (por ejemplo:

a:\VENTAS\COTIZA).

-Presionar la tecla de Retroceso si necesita corregir algún error.

-Presione ALT + A para pasar a la lista de Archivos, presione la flecha hacia abajo para seleccionar el nombre del archivo que desea recuperar.

-Si el archivo no se encuentra en la unidad o directorio en uso, presione ALT + D para pasar a la lista de "Directorios". Presione la flecha hacia abajo hasta seleccionar la unidad o el directorio apropiado, y presione ENTER.

-Seleccione los dos puntos (..) para pasar a un nivel anterior en la escritura de directorios. Después, seleccione el nombre del documento en la lista de "Archivos".

- 3.- Seleccione la opción "SI" o presione ENTER.

COMANDO CERRAR:

Cierra un documento de Word en cualquier momento.

- 1.- Escoja el comando Archivo Cerrar.

Si ha hecho cambios en el documento desde la última vez que lo guardó, Word presentará un mensaje preguntándole si desea guardar los cambios.

-Elija "SI" si desea guardar el documento e incorporar los cambios que haya hecho.

-Elija "NO" si desea descartar todos los cambios que no haya guardado anteriormente.

-Elija "Cancelar" (ESC) para cancelar el comando Archivo Cerrar.



CERRAR TODO:

Cierra todos los documentos, glosarios y hojas modelo.

Con este comando puede cerrar todos los documentos simultáneamente.

- 1.- Escoja el comando Archivo Cerrar Todo.
- 2.- Aparecerá una caja de diálogo, dónde preguntará si desea guardar los documentos abiertos. Haga uno de los siguientes pasos:
 - Elija "SI" si desea guardar los cambios realizados.
 - Elija "NO" si no desea guardar los cambios.
 - Elija "Cancelar" para cancelar el comando Cerrar Todo.

COMANDO GUARDAR:

Guarda el documento u hoja modelo en uso.

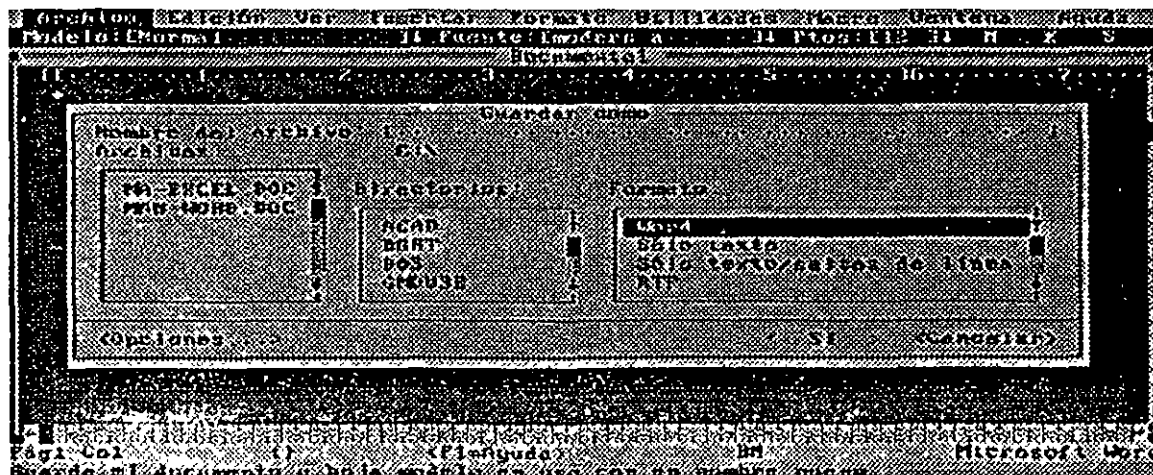
Este comando guarda el documento de forma completa y actualiza las versiones en el disco cada vez que usted lo escoge.

- 1.- Escoja el comando Archivo Guardar.
- 2.- Aparecerá en la barra de estado un mensaje indicando que está guardando el archivo con su nombre y con cuantos caracteres cuenta.

COMANDO GUARDAR COMO:

Guarda el documento u hoja modelo en uso con un nombre nuevo.

Para guardar con otro nombre a un archivo se utiliza el comando Archivo Guardar Como.





Nombre del Archivo: El nombre que desea asignar al archivo.

Archivos: Una lista de documentos existentes en la unidad de disco o directorio en uso.

Directorios: Una lista de las unidades de disco y los subdirectorios disponibles.

Formato: El formato con el que desee guardar el archivo. Por Default es Word (*.DOC).

Opciones: Abre una caja de diálogo en el que podrá especificar el directorio desde el que se van a abrir los documentos o en el que se guardarán.

Cuando guarde por primera vez un documento Word le asignará la extensión DOC.

- 1.- Escoja el comando Archivo Guardar como.
- 2.- En el cuadro "Nombre de Archivo", escriba un nombre de hasta 8 caracteres.
- 3.- Elija "SI" o presione ENTER.

COMANDO GUARDAR TODO:

Guarda todos los documentos, hojas modelo y glosarios abiertos.

Este comando se utiliza para guardar de un solo paso los documentos abiertos, incluyendo cualquier hoja modelo que esté activa y el archivo del glosario en uso.

- 1.- Escoja el comando Archivo Guardar Todo.

Mientras que Word esté guardando los documentos, aparecerá el mensaje "Guardando" en la barra de estado, seguido de la ruta de acceso y del nombre de cada documento que se vaya guardando. Cuando desaparezca el mensaje todos los archivos habrán sido guardados.



COMANDOS DEL MENU EDICION

COMANDO DESHACER:

Invierte la última acción.

Este comando invierte el efecto, es decir, anula la última acción de revisión o formato. También podrá deshacer las acciones de escribir, ordenar y separar palabras en sílabas. Es necesario usar este comando inmediatamente después de realizar la acción que desee anular.

Procedimiento.

- 1.- Escoja el comando Edición Deshacer para restablecer la forma que tenía el documento antes de realizar el último cambio.

COMANDO REPETIR:

Repite la última acción.

Con este comando se puede acelerar el proceso de modificación.

- 1.- Escoja el comando Edición Repetir.

COMANDO CORTAR:

Borra la selección y la coloca en el portapapeles.

El portapapeles es un lugar de almacenamiento temporal para una porción de texto tomada de un documento. Usted borra (corta) o copia texto colocándolo en el portapapeles, pudiendo luego insertar (pegar) el texto almacenado en el documento activo o en cualquier otro.

Procedimiento.

- 1.- Seleccione el texto que desee borrar (cortar).
- 2.- Escoja el comando Edición Cortar.

Al teclear este comando, el texto seleccionado desaparecerá temporalmente de la pantalla, hasta que se mande a otro lugar del documento.

También puede presionar las teclas MAYUSCULAS+SUPR para borrar texto y transfe-

... rirlo al portapapeles.



COMANDO COPIAR:

Copia la selección y la coloca en el portapapeles.

Cuando copia un texto, word lo deja en su lugar original dentro del documento e inserta una copia de éste en otro lugar.

Procedimiento.

- 1.- Seleccione el texto que desee copiar.
- 2.- Escoja el comando Edición Copiar.

Word pondrá una copia del texto en el portapapeles.

- 3.- Mueva el cursor al lugar donde desee insertar la copia.
- 4.- Escoja el comando Edición Pegar.

COMANDO PEGAR:

Inserta el contenido del portapapeles en la posición del cursor.

Procedimiento.

- 1.- Coloque el cursor en el lugar que desee colocar el texto.
- 2.- Escoja el comando Edición Pegar.

También puede presionar la combinación de teclas MAYUSCULAS+INS para insertar el texto desde el portapapeles.

COMANDO HALLAR:

Busca texto o Formatos específicos.

El texto de búsqueda puede contener hasta 256 caracteres, incluyendo los espacios. Cuanto más largo sea el texto de búsqueda, más tiempo utilizará Word para buscarlo, especialmente si se trata de un documento largo.

Procedimiento.



- 1.- Lleve el cursor al lugar en el que desee iniciar la búsqueda o seleccione el texto.
- 2.- Escoja el comando Edición Hallar.
- 3.- En el cuadro "Hallar el Texto", escriba el texto que desee buscar.
- 4.- Realice uno de los siguientes pasos:
 - Seleccione la opción "Arriba" para buscar hacia el principio del documento.
 - Seleccione la opción "Abajo" para buscar hacia abajo. (Opción predeterminada).
- 5.- Elija "SI".

COMANDO REEMPLAZAR:

Busca textos o formato y los reemplaza.

Procedimiento.

- 1.- Coloque el cursor al principio de la porción del documento en la que desee reemplazar el texto o seleccione dicha porción.
- 2.- Escoja el comando Edición Reemplazar.
- 3.- En el cuadro "Hallar el texto" escriba el texto que desee reemplazar.
- 4.- En el cuadro "Reemplazar con" escriba el texto de reemplazo.
- 5.- Deje sin marcar la casilla "confirmar cambios" si desea que Word reemplace el texto automáticamente cada vez que lo encuentre, sin pedirle confirmación.
- 6.- Elija "SI".
- 7.- Si la opción "Confirmar cambios" está activada, Word se detendrá cada vez que encuentre el texto. Entonces deberá seguir uno de estos pasos:
 - Elija "SI" para reemplazar texto.
 - Elija "NO" para dejar el texto intacto y proceder a la siguiente aparición del mismo.
 - Elija "Cancelar" para suspender la acción de reemplazo.

COMANDO IR A:

Pasa a un lugar específico en el documento activo.

Si esta revisando un documento ya impreso o repaginado, podrá moverse rápidamente al

primer carácter de una página específica utilizando el comando Edición Ir a. Este comando es muy útil, especialmente si se trabaja con un documento muy grande.



Procedimiento.

- 1.- Escoja el comando Edición Ir a.
- 2.- En el cuadro "Ir a" escriba el número de la página a dónde desee ir.
Word indica el número de la página presente en la pantalla en la barra de estado.
- 3.- Elija "SI" o presione ENTER.

COMANDO GLOSARIO:

Inserta o define elementos de glosario.

Cada porción de texto guardado es llamado elemento de glosario. Para crear un elemento glosario, escriba el texto en el documento, seleccione el texto y guárdelo con un nombre único.

Crear un Elemento de Glosario.

Un elemento de Glosario puede abarcar cualquier cantidad de texto. El nombre para el elemento puede contener un máximo de 31 caracteres (letras o números).

Procedimiento.

- 1.- Escriba el texto que desee guardar como elemento de glosario.
- 2.- Seleccione el texto.
- 3.- Escoja el comando Edición Glosario.
- 4.- En el cuadro "Nombre del Elemento de glosario" dé nombre en elemento de glosario.
- 5.- Elija la opción "Definir".

Insertar el texto de un elemento de glosario.

Al insertar un elemento de glosario, Word copiará el texto del elemento de glosario en la posición en la que se encuentre el cursor en el documento.

- 1.- Coloque el cursor al lugar donde desee insertar el elemento de glosario.
- 2.- Escoja el comando Edición Glosario.
- 3.- En el cuadro "Nombres" seleccione el nombre del elemento de glosario.
- 4.- Elija la opción "Insertar".

Eliminar un elemento de glosario.

- 1.- Escoja el comando Edición Glosario.
- 2.- En el cuadro "Nombres", seleccione el nombre del elemento de glosario que desee borrar.
- 3.- Elija la opción "Eliminar".

4.- Elija "Cerrar" para cerrar la caja de diálogo.



COMANDOS DEL MENU VER

COMANDO ESQUEMA:

Presenta el esquema de un documento (Activa/desactiva).

Un esquema en Word, no es un documento separado, sino que es otra presentación del mismo documento que aparece en la pantalla con sólo presionar unas teclas.

Procedimiento.

1.-Escoja el comando Ver Esquema.

Aparecerá una marca junto a Esquema en el menú para indicarle que está activado el modo de presentación de esquema.

Para volver al modo de presentación normal escoja nuevamente el comando Ver Esquema.

COMANDO PAGINA:

Presenta en la pantalla columnas y objetos con posición fija (Activa/Desactiva).

Word tiene un modo de pantalla que le permitirá ver la distribución en la página de los párrafos lado a lado al escribirlos y darles formato.

Procedimiento.

1.- Escoja el comando Ver Página.

Si la ventana está vacía, Word insertará una marca de párrafo al activar el modo de presentación de página. (Nota.- No es posible borrar esta marca de párrafo).

COMANDO CINTA:

Activa o desactiva la cinta.

La cinta presenta el nombre de la fuente, el tamaño de la fuente y el formato (negrita, cursiva, subrayado, etc.) de los caracteres del texto seleccionado. Si utiliza una hoja modelo, la cinta indicará el modelo que se haya aplicado al texto seleccionado.

La cinta indica los formatos del texto seleccionado, a menos que el texto contenga varios formatos diferentes. Y si esto sucede (formatos diferentes) aparecerán dos signos de interrogación en la abreviatura de la regla correspondiente a dicho formato.



VER Y OCULTAR CINTA:

Procedimiento.

1.- Escoja el comando Ver Cinta.

Si utiliza el mouse, señale el icono de la regla en la esquina superior derecha de la ventana y haga clic con el botón derecho del mouse.

COMANDO REGLA:

Activa o desactiva la regla.

La regla presenta las sangrías y tabulaciones del párrafo que contiene el cursor.

Si mueve el cursor a un párrafo que tenga un formato distinto, la regla cambiará para indicar los formatos del párrafo seleccionado. Si selecciona dos o más párrafos con formatos distintos, la regla indicará los formatos de las tabulaciones comunes a los párrafos.

Cada documento puede contener una regla.

Procedimiento.

Ver y ocultar la regla.

1.- Escoja el comando Ver Regla.

Si trabaja con mouse, señale el icono de la regla en la esquina superior derecha de la ventana y haga clic con el botón izquierdo del mouse.

COMANDO BARRA DE ESTADO:

Activa o desactiva la barra de estado.

La barra de estado le proporciona información acerca de su sesión de trabajo con Word y acerca del documento en el que esté trabajando.

Procedimiento.

Ver y Ocultar la barra de estado.



- 1.- Escoja el comando Ver Barra de Estado.

COMANDO NOTAS AL PIE/ANOTACIONES:

Abre la ventana de Notas al Pie/Anotaciones para modificarlas (Activa/Desactiva).

Word tiene un panel especial en el que se pueden ver y revisar las notas al pie y anotaciones sin tener que desplazarse hasta el final del documento.

Procedimiento.

Abrir o Cerrar el panel de notas al pie/anotaciones.

- 1.- Para abrir el panel de notas al pie, escoja el comando Ver Notas al Pie/Anotaciones.
- 2.- Para cerrar el panel de notas al pie, escoja el comando Ver Notas al Pie/Anotaciones.

COMANDO PREFERENCIAS:

Establece opciones de pantalla para el documento.

Activa y Desactiva:

- Las barras de desplazamiento
- Los saltos de línea
- Los números de línea

Cambia:

- La velocidad el cursor
- Los colores de la ventana.

Además coincidir el número de líneas de la pantalla y colores para los formatos de carácter.

Procedimiento.

- 1.- Escoja el comando **Ver-Preferencias**.
- 2.- Seleccione la opción que desee.



COMANDOS DEL MENU FORMATO

COMANDO CARACTER:

Cambia el aspecto de los caracteres seleccionados.

Usos típicos de los formatos de carácter que podrá aplicar con Word.

Formato	Uso típico
Negrita, cursiva, ejemplo, subrayado doble, mayúsculas, versales	Realzar o dar énfasis a letras, palabras, frases o párrafos; por para títulos, advertencias o palabras clave.
Tachado	Indicar el texto que se debe eliminar de un documento o cláusulas de un contrato que han sido anuladas.
Subíndices, registrada y Superíndice	Indicar las referencias de notas al pie, símbolos de marca de derechos de autor; en fórmulas científicas.
Texto Oculto	Ocultar texto con el objeto de agregar comentarios, incluir comandos para elementos de índices o tablas de contenido, o agregar marcadores especiales que faciliten la introducción de datos de un formulario

Procedimiento.

Poner Formatos de Carácter.

- 1.- Seleccione el texto al cual desea poner formato de carácter.
- 2.- Escoja el comando Formato Carácter.
- 3.- Seleccione las opciones deseadas.

Quitar Formatos de Carácter.

- 1.- Seleccione los caracteres cuyo formato desee quitar.
- 2.- Realice uno de los siguientes pasos:

-Para quitar formatos uno por uno, presione la combinación de teclas de formato rápido o quite la marca de la casilla apropiada en la caja de diálogo.

-Para quitar todos los formatos con excepción de la fuente y el tamaño, presione CTRL+R.

-Para volver al formato carácter normal (quitar todos los formatos), presione CTRL+Barra espaciadora.



COMANDO PARRAFO:

Cambia el aspecto, alineación e interlineado de los párrafos seleccionados.

Dar formato a los párrafos con las teclas de formato rápido.

Para Aplicar este formato	Mantenga presionada la tecla CTRL y presione:
Centrado	C
Alineación a la Izquierda	I
Alineación a la Derecha	D
Justificado	J
Aumentar la sangría izquierda	U
Reducir la sangría izquierda	Y
Sangría Francesa	G
Doble Espacio	2
Espacio Sencillo	1
Espacio de una línea antes del párrafo	A
Párrafo Normal	X
Ambas Sangrías (izq. y der)	L

Procedimiento.

Dar formato con el comando Formato Párrafo

- 1.- Seleccione el texto, frase o párrafo al que desee dar formato.
- 2.- Escoja el comando Formato Párrafo.
- 3.- Seleccione las opciones que desee.
- 4.- Escoja "SI" o presione ENTER.

COMANDO SECCION:

Cambia el formato de la página o secciones seleccionadas.

Procedimiento.

- 1.- Coloque el cursor dentro de la sección cuyo formato desee cambiar.
- 2.- Escoja el comando Formato Sección.
- 3.- En los cuadros "Columnas", "Notas al Pie" y "Números de Líneas", seleccione las opciones deseadas para esta sección.

- 4.- En el cuadro "Empezar", seleccione una opción para indicar a Word el lugar el que desee tener el salto de página para la nueva sección.



COMANDO MARGENES:

Cambia los márgenes de las páginas o secciones seleccionadas.

El tamaño de página determinado de Word es de 21 por 29.7 cm. Cada documento nuevo tendrá este tamaño de página y un margen superior e inferior de 2.5 cm. y un margen izquierdo y derecho de 3 cm. Sin embargo podrá cambiar el tamaño de la página y los márgenes de cualquier documento para acomodar estas medidas al tipo de papel que utilice normalmente.

Procedimiento.

Cambiar las medidas para el ancho y el largo de la página.

- 1.- Escoja el comando Formato Márgenes.
- 2.- En los cuadros "Ancho" y "Largo", escriba las medidas del papel en el que vaya a imprimir su documento.
- 3.- Elija "SI" o presione ENTER.

Cambiar Márgenes.

El margen es el espacio en blanco que se encuentra entre el área en la que se imprime el texto y el borde del papel.

Procedimiento.

- 1.- Escoja el comando Formato Márgenes.
- 2.- En los cuadros "Superior", "Inferior", "Izquierdo" y/o "Derecho", especifique las medidas que desee.

Word medirá los márgenes a partir del borde correspondiente del papel.

- 3.- Elija "SI" o presione ENTER.

COMANDO TABULACIONES:

Establece o elimina tabulaciones para los párrafos seleccionados.

Trabajar con tabulaciones en Word es como hacerlo en una máquina de escribir. Al presionar la tecla TABULACION, el cursor salta a la tabulación siguiente y el espacio se llena con un carácter de tabulación.



Todos los párrafos de un documento nuevo tendrán tabulaciones predeterminadas a cada 1.25 cm. Puede usar estas tabulaciones o:

- Cambiar el intervalo de las tabulaciones predeterminadas.
- Fijar tabulaciones propias en cualquier posición. Puede especificar la alineación del texto con relación a las tabulaciones que fije e indicar el carácter de relleno.

Cambiar las tabulaciones predeterminadas.

- 1.- Escoja el comando Utilidades Personalizar.
- 2.- En el cuadro "Tabs. predeterminadas", escriba una medida para la distancia que desee entre las tabulaciones.
- 3.- Elija "SI" o presione ENTER.

Fijar Tabulaciones con medidas precisas.

- 1.- Coloque el cursor dentro del párrafo en el que desee fijar las tabulaciones o seleccione un grupo de párrafos.
- 2.- Escoja el comando Formato Tabulaciones.
- 3.- En el cuadro "Posición" escriba una medida. Indique también la abreviatura para la unidad de medida si no desea utilizar centímetros.
- 4.- En el cuadro "Alineación", seleccione una opción.
- 5.- Elija el botón "Fijar".

El valor que escriba aparecerá en el cuadro de la lista de la opción "Posición".

- 6.- Repita los pasos 3 al 5 para cada tabulación que desee fijar.

Las tabulaciones no aparecerán en la regla hasta que se elija "SI" y cierre la caja de diálogo.

- 7.- Elija "SI" o presione ENTER.

COMANDO BORDES:

Dibuja líneas o cuadros alrededor de los párrafos seleccionados.

Es posible colocar línea y cuadros alrededor de los párrafos, llamados bordes de párrafos, utilizando el comando **Formato Bordes**. Los bordes de párrafos se ajustan automáticamente si se cambia el tamaño del párrafo insertando o eliminando texto, o si se cambia el tamaño de la fuente.



Procedimiento.

- 1.- Coloque el cursor dentro del párrafo al que desee agregar el borde.
- 2.- Escoja el comando Formato Bordes.
- 3.- Seleccione la opción "Encuadrar cada párrafo" o la opción "Líneas".
- 4.- Si seleccionó la opción "Líneas", marque las casillas "Izquierda", "Derecha", "Superior" o "Inferior" para especificar el lugar en el que desee agregar los bordes de líneas.
- 5.- En el cuadro "Tipo de Línea", deje la opción propuesta, "Normal" o seleccione "Negrita", "Doble" o "Gruesa".
- 6.- Si utiliza una impresora con capacidad para imprimir en color, seleccione un color en la opción "Color" del cuadro "Tipo de línea".
- 7.- Elija "SI" o presione ENTER.

COMANDO ENCABEZADO/PIE DE PAGINA:

Convierte los párrafos seleccionados en encabezados o pies de página.

Para crear los encabezados o pies de página y para determinar su alineación, se utiliza el comando Formato Encabezado/Pie de Página. Antes de trabajar con encabezados y pies de página, asegúrese de que el modo de presentación de página esté desactivado.

Procedimiento.

- 1.- Seleccione el párrafo o los párrafos que desee usar como encabezado o pie de página. Asegúrese de que el párrafo o párrafos figuren como el primer texto de la primer página en la que desee imprimirlos, generalmente el primer párrafo de una sección.
- 2.- Escoja el comando Formato Encabezado/Pie de Página. Este comando no estará disponible (aparecerá inactivo en el menú) si está activado el modo de presentación de página.
- 3.- Para asignar el formato de encabezado al párrafo, seleccione la opción "Encabezado"; para asignarle el formato de pie de página, seleccione la opción "Pie de Página".
- 4.- Para alinear el encabezado o pie de página con el margen izquierdo, seleccione la opción



FACULTAD DE INGENIERIA, UNAM.
DIVISION DE EDUCACION CONTINUA

INTRODUCCION A LAS COMPUTADORAS PERSONALES

BASE DE DATOS

FEBRERO 1995

CURSO:

dBASE III PLUS

DIC, S.A. DE C.V. CURSO : dBASE III PLUS
CAPITULO : I
INTRODUCCION

1. Que es dBASE III PLUS.

dBASE III PLUS es un Sistema Manejador de Bases de (DBMS) Que se puede definir como el conjunto de programas que permiten fácilmente la administración de grandes volúmenes de información.

El tipo de operaciones que permite al usuario es el de crear actualizar, ordenar, clasificar, seleccionar, emitir listados y reportes, elaboración de etiquetas para listas de correos, y un sin número de operaciones con la información que contenga el sistema que se desee controlar por medio de la computadora. Por ejemplo algunas de las aplicaciones más frecuentes en las que dBASE III PLUS es utilizado son:

- Control de Inventarios
- Listas de Precios
- Directorios telefónicos
- Nóminas
- Contabilidad
- Control de Ordenes de Trabajo
- Control de Chequera
- Consiliación Bancaria

Y en general todos aquellos sistemas que manejen volúmenes de información considerable y que se desee llevar un control más fácil de esa información. Antes de continuar con el estudio de este paquete, es necesario definir algunos conceptos básicos que serán de gran utilidad posteriormente.

DATOS.- Son la materia prima que al ser procesada dan como resultado la información, por ejemplo:

Son datos las horas trabajadas y el salario por hora, pero el importe a pagar por las horas trabajadas a la semana es la información. Los datos son elementos que dan informes sobre el nombre, tipo e intervalo de valores. Ejemplo de esto es; el nombre, dirección, ciudad, estado, clave postal y el número del cliente.

CAMPO.- Un campo define la localización física de almacenamiento de una unidad de datos o información.

Un campo puede tener uno o más *BYTES* de longitud, siendo este la unidad más pequeña de datos o información que se puede definir y manejar, es el común denominador entre el *USUARIO* y la *COMPUTADORA*.

Un ejemplo de campos en una base de datos es:

Campos	Datos (capturados por el usuario)
NOMBRE:	DISTRIBUIDORA INTEGRAL DE COMPUTACION, SA. DE C.V.
DIRECCION:	MEDELLIN 184 DESPACHO 105
COLONIA:	ROMA
CIUDAD:	MEXICO
ESTADO:	DISTRITO FEDERAL
C.P.:	06700
TELEFONO:	584-88-22

CARACTERISTICAS PRINCIPALES DE UNA BASE DE DATOS

-INDEPENDENCIA DE LOS DATOS.- No es necesario que el usuario conozca en forma detallada la localización física de los datos que utiliza, este tipo de problemas los resolverá el Sistema por sí solo, valiéndose de un diccionario de datos.

-SEGURIDAD.- Existen sistemas en los que se maneja el acceso a una base de datos mediante *claves de acceso*, lo cual sugiere que no todos los usuarios tendrán acceso a determinados niveles de información, para no afectar la base de datos.

-CONSULTAR EN LINEA.- Permite el acceso a datos en forma interactiva, lo que significa que el usuario consultará información en menor tiempo, evitando reportes que requieren de grandes inversiones de tiempo.

-LENGUAJE FORMAL DE ALTO NIVEL.- Permite a los usuarios y programadores, desarrollar programas específicos de aplicación con más rapidéz que los lenguajes de programación convencionales.

CURSO : DBASE III PLUS
DIC, S.A

Base de datos relacional.

dBase III PLUS cumple con el modelo para el manejo de bases de datos tipo relacional. Una base de datos relacional se define como una o más tablas rectangulares de renglones y columnas. Cada tabla recibe el nombre de relación. Los renglones son llamados registros y las columnas se denominan campos. A cada registro se asigna un número "el número de registro", y cada campo a su vez debe tener un nombre que lo defina el nombre del campo y lo identifique de entre los demás.

	SALON	GRADO	NOMBRE DEL ALUMNO
001	11	2	Alvarez López Joaquín
002	15	4	Amezcuca Cárdenas Juan José
003	19	5	Amigo Chávez Luis Alberto
.	.	.	
.	.	.	
425	9	2	Zurita Perea Carlos Gerardo

La administración de una base de datos relacional provee la capacidad de referencias cruzadas entre relaciones (archivos tipo base de datos en Dbase III). La figura que se muestra a continuación nos ejemplifica el concepto de relación. Los estudiantes pueden ser relacionados con sus profesores, basandonos en el contenido de las columnas (campos). De esta manera Joaquín Alvarez López y su maestra Olivia Cortés pueden ser localizados utilizando, la referencia cruzada del número de salón y grado de las dos tablas.

	SALON	GRADO	NOMBRE DEL ALUMNO
001	11	2	Alvarez López Joaquín
002	15	4	Amezcuca Cardenas Juan José
003	19	5	Amigo Chavez Luis Alberto
004			

	SALON	GRADO	NOMBRE DEL PROFESOR
005	11		
006	14		
007	18		
001	14	6	Rafael Pérez Cabrera
002	11	2	Olivia Cortez López
003	7	5	Alejandro Belmar Mendez
004	15	1	Norma A. Jiménez Hawley

CAPITULO II

DISEÑO Y CREACION DE UNA BASE DE DATOS

CICLOS DE DESARROLLO DE UN SISTEMA

ANALISIS

El objetivo de esta etapa es definir ¿QUE? se va a hacer, aunque resulte lógico no siempre lo es, ya que muchas veces se inicia con algo y se termina con otro producto diferente precisamente por no saber hacia donde dirigimos nuestro sistema. Esta etapa la podemos dividir en las siguientes actividades:

a) **ANALISIS DE REQUERIMIENTOS.**- En esta etapa se realiza un levantamiento de documentos fuente, así como las entrevistas a los usuarios que los solicitaron como los que lo operan en forma mecánica o manual. Otro aspecto tocado aquí también son los resultados que el usuario pretende obtener del sistema.

b) **ANALISIS DE DATOS.**- Una vez hecho lo anterior es necesario identificar cuales son los datos que se requieren para poder llegar a los resultados que el usuario quiere. Se deben agrupar de tal forma que exista una relacion entre ellos y no se dupliquen.

DISEÑO

En esta etapa se define el ¿como? se va a realizar. A su vez esta se divide en los siguientes pasos:

a) **DISEÑO CONCEPTUAL.**- Aquí se define la estructura del SISTEMA así como las entidades que se van a definir en la BASE DE DATOS.

b) **DISEÑO LOGICO.**- Esta etapa consiste en definir las relaciones que tendrán las entidades y cual será la forma de acceso, también se definen los archivos de índices que se utilizarán.

c) **DISEÑO FISICO.**- Esta etapa consiste en establecer los mecanismos de operación, así como los recursos de cómputo que se van a requerir para poder procesar toda la información del Sistema.

PROGRAMACION

Esta etapa es la más laboriosa, ya que se tienen que definir todos los programas y la interrelación que existe entre ellos.

IMPLEMENTACION

Esta etapa consiste en implantar el sistema y probarlo con datos reales, quizás no existan pero estos deben simularse de tal forma que pueda trabajarse el sistema a plena carga para ver su comportamiento y eficiencia.

AYUDA DE dBASE III PLUS HELP

En el caso de que necesite ayuda, el dBASE III PLUS le puede ofrecer una breve información sobre cuestiones que van desde el punto indicativo hasta la creación de una base de datos, información que esté contenida en un archivo **HELP** del disco del sistema y que es accesible mientras se permanece en el dBASE III PLUS. Una serie de menús le ayudará a encontrar la información que necesite.

El dBASE III PLUS le ofrece ayuda aún sin que usted la solicite. Si el dBASE III PLUS no entiende como intenta usted usar una orden le pregunta si necesita ayuda de los archivos **HELP**.

Otra forma de obtener ayuda es por medio de la tecla **F1** o tecleando la palabra **HELP**, usted puede pedir ayuda específica de un comando tecleando la palabra **HELP** y nombre del **<COMANDO>**.

ASSIST (ASISTENTE)

Todas las ayudas del asistente que se refieren al cambio de archivos están contenidas en el menú **UPDATE**. Para modificar la estructura de la base de datos, use la opción **DATA BASE FILE** del menú **MODIFY**. Desde el punto indicador introduzca **ASSIST** o la tecla **F2**; el menú **SET UP** aparece resaltado. Como lo que queremos es ir a **UPDATE**, pulse **U** para abrir el menú **UPDATE**. Observe que el menú **UPDATE** proporciona acceso a las mismas ordenes **BROWSE**, **EDIT**, **DELETE**, **RECALL**, **PACK** y **REPLACE** que se han usado a lo largo de este pequeño capítulo.

Al solicitar ayuda del **ASSIST**, dBASE III PLUS despliega el menú **SET UP** en pantalla y usted podrá ver que el asistente o ayudante está dividido en las siguientes áreas:

- 1.- **MENU BAR (MENU HORIZONTAL)**: La barra del menú consiste en una serie de catálogos detallados en la primera línea de la pantalla.
- 2.- **CLOCK DISPLAY (RELOJ)**: El sistema muestra la fecha actual en el extremo superior derecho de la pantalla.
- 3.- **PULL DOWN MENU (MENU VERTICAL)**: El menú vertical se presenta en un cuadro, este contiene una lista de opciones.
- 4.- **COMMAND LINE (LINEA DE COMANDOS)**: Esta línea le muestra los comandos de dBASE III PLUS.
- 5.- **STATUS LINE (LINEA DE ESTADO)**: Esta sección despliega información actual del programa. Así mismo la línea de estado posee 6 secciones:

- a) **MODE (Modo)**.- Este muestra el modo actual del **ASSIST**.
- b) **DRIVE**.- Aquí se muestra la unidad de disco activa.

- c) **FILE.**- Archivo con el cual se está trabajando(current file).
- d) **LOCATION.**- Esta área muestra la localización del cursor en su menu vertical.
- e) **INSERT/OVERTYPE.**- Esta área indica el tipo de estado, cualquiera de los dos, **INSERT** u **OVERTYPE**.
- f) **LOCKS.**- Esta indica si están activadas o no las teclas **CAPS** o **NUM LOCK**. Es muy útil, debido a que muchos teclados no tienen indicador en el cual se muestre que las teclas de estado estén activas.
- g) **MESSAGE AREA.**- El área de mensaje es usada para desplegar información de ayuda cuando se esté operando un programa.

SET STATUS

Este comando de DBase III Plus tiene el siguiente formato:

SET STATUS ON/OFF

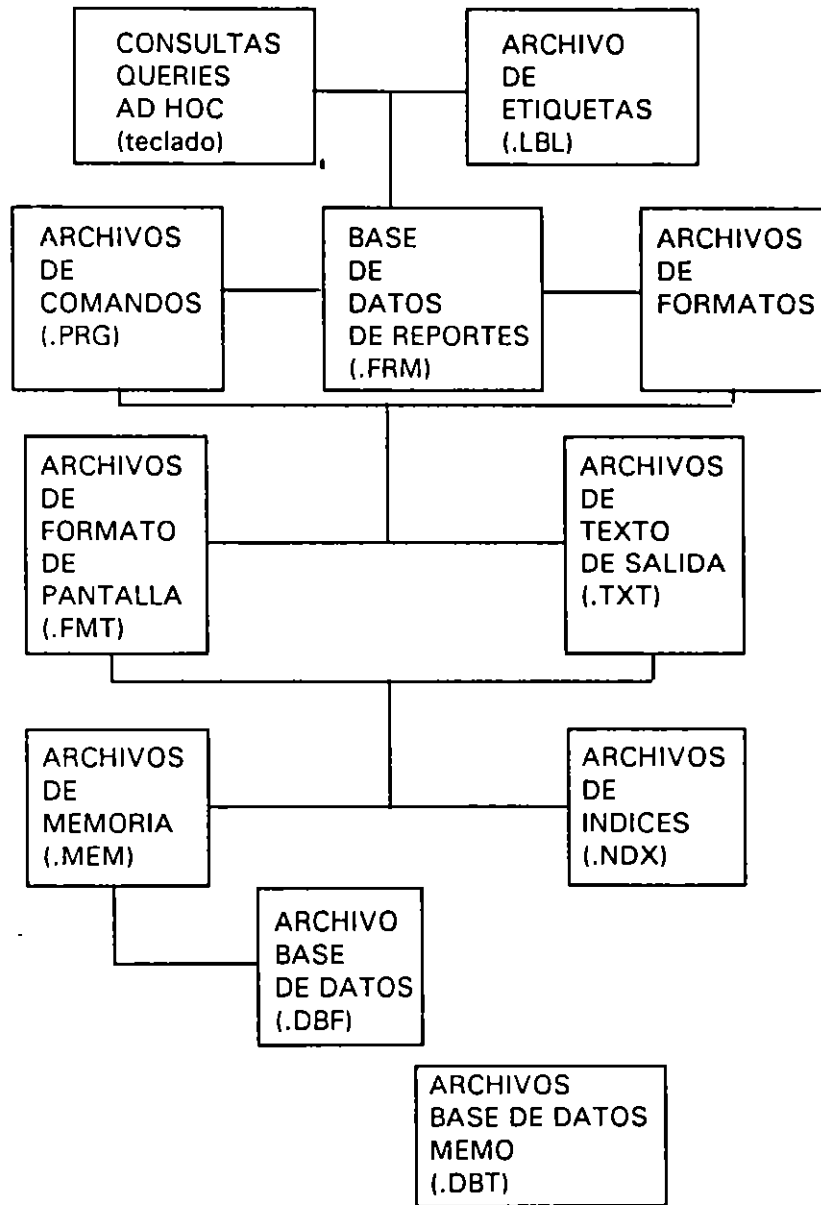
ESPECIFICACION DE UNIDAD DE TRABAJO

SET DEFAULT TO

Este comando permite que nosotros podamos cambiar la unidad de trabajo, y su formato es el siguiente:

SET DAFUALT TO <Drive>
Ejm: **SET DAFUALT TO B:**

ESQUEMA DE ADMINISTRACION DE UNA BASE DE DATOS.



ARCHIVOS EN DBASE III PLUS

DBASE III PLUS guarda la información en disco en forma de archivos en trece formatos especializados. Cada uno sirve para una necesidad específica de procesamiento para Dbase III PLUS.

Cada nombre de archivo en disco se forma de la siguiente manera:

-El nombre de archivo propiamente dicho de hasta 8 caracteres. El primer caracter debe ser letra, y es asignado por el usuario.

-Una extensión denominada el "identificador" de archivos que consiste de un punto y tres letras. Normalmente es asignado por Dbase III PLUS al crear el archivo.

El usuario puede establecer un identificador de archivo en particular si al nombrar inicialmente el archivo, incluye la extensión que el desea que este nuevo archivo tenga. En lo sucesivo cuando se desee trabajar con este archivo se deberá siempre invocar con el nombre completo incluyendo la extensión particular con que fué creado.

CLASE DE ARCHIVO	IDENTIFICADORES DE ARCHIVO
1 - Archivos Base de Datos	.DBF
2.- Archivos Base de Datos Memo	.DBT
3.- Archivos de Indices (INDEX)	.NDX
4.- Archivos de Memoria	.MEM
5.- Archivos de Comandos	.PRG
6.- Archivo de Formato de Pantalla	.FMT
7.- Archivos de Etiquetas	.LBL
8.- Archivos de Formato de Reportes	.FRM
9.- Archivos de Texto de Salida	.TXT
10.- Archivos de pantallas	.SCR
11.- Archivos de catalaogos	.CAT
12.- Archivos de vistas	.VUE
13.- Archivos de query	.QRY

A continuación se describe cada uno de ellos:

1.- **Archivos Base de Datos (.DBF).**- Estos archivos almacenan datos en forma de registros ; campos (renglones y columnas). Cada registro contiene un conjunto unico de información. Dbase III puede controlar hasta un billon de registros. Cada registro puede contener hasta 4000 Bytes, los cuales se deben contener en no más de 128 campos de datos.

2.- **Archivos Base de Datos Memo (.DBT).**- Son archivos auxiliares a los archivos base de datos (.DBF). Utilizados para contener los campos memo. Todos los campos memo de un archivo .DBT son almacenados en un archivo .DB del mismo nombre.

Cada registro base de datos puede contener hasta 128 campos memo. Cada campo memo puede contener el mismo tipo de información que los campos tipo carácter, la diferencia es que a un campo tipo carácter le caben hasta 256 caracteres como máximo, y un campo tipo memo puede contener hasta 4000 caracteres. Cada campo memo requiere un mínimo de 512 Bytes por registro si cualquier dato es introducido. Cada campo memo ocupa solo 16 espacios en el archivo tipo base de datos (.DBF), ya que el texto es almacenado en un archivo auxiliar (.DBT) separado dentro del disco.

3.- **Archivos de Índices (INDEX .NDX).**- Proveen la facilidad de utilizar la base de datos en un orden lógico. El orden físico es el orden en el cual los registros fueron introducidos, y el orden lógico, es un orden basado en un criterio alfabético numérico considerando el contenido de uno o más campos de la base de datos.

Los archivos index ejecutan la traducción entre una llave (campo o expresión de campos que resulta un elemento de interés, tal como un nombre) y su correspondiente número de registro en la base de datos. Cuando utilizamos una base de datos con un archivo de índices nos da la apariencia de estar ordenada con respecto a una llave. Esto se denomina orden lógico. La llave (la cual se puede componer de uno o varios campos), también puede ser utilizada para proveer acceso directo (random) a un registro en particular.

4.- **Archivos de Memoria (.MEM).**- Los archivos de memoria pueden contener hasta 256 variables de memoria. Se utilizan para almacenar en disco (sobre un archivo), el contenido de las variables de memoria y poder más tarde utilizar las variables con este valor almacenado que resulta de nuestro interés. Estos archivos son creados y almacenados en disco a través del comando **SAVE**, y devueltos a la memoria a través del comando **RESTORE**.

5.- **Archivos de Comandos (.PRG).**- Contienen conjuntos de instrucciones de Dbase III, que se han almacenado como programas. Son archivos ASCII y pueden ser creados a través del comando **MODIFY COMMAND**, o en el modo "no documento" de la mayoría de los programas procesadores de texto.

6.- **Archivos de Formatos de Pantalla (.FMT).**- Son archivos creados como pantallas de captura de usuarios y se utilizan en la entrada de datos e impresión de salida.

7.- **Archivos de Etiquetas (.LBL).**- Los archivos de etiquetas contienen la información necesaria para los comandos **LABEL** de impresión de etiquetas.

8.- **Archivos de Formato de Reportes (.FRM).**- Estos archivos, contienen información que necesita el comando **REPORT**, para la elaboración de reportes. Estos archivos pueden ser creados y alterados a través del comando **MODIFY REPORT**.

9.- **Archivos de Texto de Salida (.TXT).**- Son principalmente utilizados, para crear la interface entre Dbase III y otros programas. Son archivos ASCII y contienen solo caracteres ASCII posibles de imprimir. Los archivos de texto de salida serán leídos a Dbase III mediante una forma especial del commando **APPEND FROM**. Son creados a través de un formato especial del **COPY**. también pueden ser utilizados para las actividades de procesamiento de registros de Dbase III mediante el comando **SET ALTERNATE**.

ESTRUCTURA DE LA BASE DE DATOS

La estructura de un archivo tipo Base de Datos se establece al definir, cada uno de los campos de la base. Esto se efectua utilizando el comando **CREATE**, y el comando **MODIFY STRUCTURE**. La definicion de los campos consiste de:

1.- **Nombre del Campo.**- Puede tener hasta 16 caracteres de longitud. Debe comenzar con una letra, y no puede contener espacios en blanco. Letras números y caracteres underscore (6) son permitidos.

2.- **Tipo de Campo.**- Los tipos de campo en Dbase III son:

IDENTIFICADOR	TIPO
C	Caracter
D	Fecha
L	Lógico
N	Numérico
M	Memo

3.- **Ancho del Campo.**- Es el número máximo de caracteres o dígitos que pueden ser contenidos en el campo. En el caso de campos numéricos, debemos tomar la consideración de que el punto decimal (en caso de que se utilicen números reales), ocupa un espacio, como cualquier otro dígito.

CADA UNO DE LOS TIPOS DE CAMPO, SE DESCRIBE A CONTINUACION:

DEFINICION DE LOS TIPOS DE CAMPOS

Existen cinco tipos de campos utilizados en Dbase III y son:

1.- **Campos Tipo Caracter (C).**- SE utilizan para almacenar cualquier caracter posible de impresión. Se pueden capturar desde el teclado. Todos estos caracteres se encuentran dentro del código ASCII, y son posibles de obtener en una impresora, incluye caracteres tales como, letras, números, algunos símbolos especiales y espacios en blanco. El tamaño máximo de este campo es 254 caracteres.

2.- **Campos Tipo Fecha (D).**- Almacenan fechas, los datos tipo fecha se pueden introducir y desplegar en una variedad de formas utilizando la función **PICTURE**. El formato de default de captura y despliegue es MM/DD/AA. El ancho del campo es siempre de 8 caracteres. Una fecha se puede sumar a un número; una fecha puede ser restada de otra fecha; y un número puede ser restado a una fecha.

3.- **Campos Tipo Numéricos (N).**- Estos campos son de dos tipos enteros y reales (incluyen decimales). Un número entero es un número que no contiene decimales (por ejemplo, el número de estudiantes en un salón). La precisión de un campo numérico es 15.9 dígitos (el punto decimal no cuenta). Esto significa que los 15 dígitos más significativos de un campo de cualquier número son siempre confiables.

4.- **Campos Tipo Lógicos (L).**- Estos campos aceptan un solo caracter que representa dos posibles valores verdadero (**TRUE**) y falso (**FALSE**). El valor de verdad lógico es posible digitarlo como T, t, o Y, y, y el valor lógico de falso como F, f, N, o n.

5.- **Campos Tipo Memo (M).**- Son diseñados para contener grandes bloques de información textual. Son almacenados en un archivo auxiliar a un archivo tipo base de datos. Están indicados en un archivo tipo Base de Datos (.DBF) por la palabra "memo". El tamaño del campo memo es variable. Si no se introducen datos el ancho del campo es cero, conforme se introducen datos, se asigna el espacio en bloques de 512 Bytes. El tamaño máximo de este tipo de campos es de 4096 Bytes. Cada campo memo utiliza 10 Bytes en cada registro del archivo tipo base de datos (.DBF).

MODIFICACION DE UNA BASE DE DATOS

MODIFY STRUCTURE

Este comando se utiliza para cambiar la estructura de una base de datos existente.

Sintáxis:

MODIFY STRUCTURE <nombre del archivo>

Cuando las modificaciones de la estructura son completados, los datos son automáticamente adicionados desde un archivo de respaldo.

APERTURA Y CIERRE DE UNA BASE DE DATOS

USE

Abre un archivo tipo base de datos, y hasta 7 archivos tipo indice (index) en el área de trabajo. Si el archivo abierto contiene campos **MEMO**, el archivo **.DBT** asociado se abre automáticamente.

Sintáxis:

USE [<nombre del archivo>] [**INDEX**<lista de archivos indice>]
[ALIAS <nombre alias del archivo>]

Si utilizamos **USE** sin parámetros, entonces se cierra la base de datos que se está utilizando y los archivos indice de ésta área.

CLOSE

Sintáxis:

CLOSE [**ALTERNATE/DATABASES/FORMAT/INDEX/PROCEDURE**]

Cierra los tipos de archivos especificados.

SELECT

Sintáxis:

SELECT <área de trabajo/alias>

Este comando se utiliza para poder cambiar de área de trabajo de entre 10 de que se disponga en DBase. Estas 10 áreas permiten el uso simultáneo de hasta 10 archivos tipo base de datos. Todas las áreas de trabajo son seleccionables, mediante el uso de un apuntador.

CAPITULO III ACTUALIZACION DE LA BASE DE DATOS

ALIMENTACION DE DATOS

APPEND

Una vez definido el archivo de base de datos, es importante saber como almacenar los datos correspondientes, para posteriormente poder procesar y así obtener la información requerida.

APPEND.- Este comando permite adicionar un nuevo registro al final del archivo de la base de datos activo.

Sintáxis:

APPEND [BLANK]

La forma más simple es dando **APPEND** solamente. La opción **BLANK** permite adicionar un registro en blanco al final del archivo, pero sin mandar la pantalla de captura. Esta opción también se usa con **REPLACE**.

MODIFICACION DE DATOS

Para poder realizar esta función, DBase cuenta con una variedad de instrucciones para realizarla.

EDIT

La instrucción **EDIT** es la que nos permite modificar todos o algunos de los campos de un registro determinado.

Sintáxis:

**EDIT [<scope>] [FIELDS <lista>] [WHILE <condicion>]
[FOR <condicion>]**

El formato más simple de **EDIT**, es el cual muestra en la pantalla todos los campos del registro en el cual se encuentra activo en el archivo de base de datos, al terminar de editarlo, pasa automáticamente al siguiente hasta que llega al fin del archivo.

Cuando el número de campos no se puede desplegar en pantalla al mismo tiempo, DBase presenta cuantas pantallas sean necesarias para poder editar cada registro.

EDIT tiene distintos modos de uso (<scope>) o ámbito y en tre ellos encontramos los siguientes:

ALL NEXT <n>
 RECORD <n> REST

ALL le indica a DBase, que el comando actuará sobre todos los registros del archivo, comenzando siempre con el primero. **RECORD <n>** le indica que sólo afectará al registro especificado n y cuya posición es **A**, dejando posicionado el apuntador en este registro. **NEXT <n>** le indica que el comando se efectuará sobre los siguientes n registros, incluyendo el registro donde está posicionado el cursor y, por último, **REST** le indica que afectará a todos los registros a partir de donde se encuentre el cursor.

Ejemplos:

.EDIT ALL	idéntico a EDIT , ya que esta es por omisión.
.EDIT RECORD 5	en este caso sólo se editará el registro 5.
.EDIT NEXT 3	editaré el registro donde está posicionado el cursor y los siguiente dos registros.
.EDIT REST	editaré a partir de donde está posicionado el apuntador hasta el último registro de la base de datos, dejando el apuntador en la etiqueta de fin de archivo (EOF).

CHANGE

Se utiliza para proporcionar ayuda con facilidades en pantalla en la edición de campos y registros previamente especificados en la base de datos activa.

Sintáxis:

CHANGE [<comodín>] [**FIELDS** <lista de campos>]
 [**FOR/WHILE** <comodín>]

Todos los registros son secuencialmente especificados, a menos que se especifique otra cosa mediante comodín o **FOR** o **WHILE**. Todos los campos se presentan para ser editados a menos que se especifique otra cosa mediante fields. Para editar campos memo se indica con ^Pg Dn y se termina con ^Pg Up, o se anula con ESC.

REPLACE

Se utiliza para cambiar al contenido de los campos especificados en la base de datos activa.

Sintáxis:

REPLACE [<comodín>] <campo> **WITH** <exp>
 [, <campo> **WITH** <exp>...] [**FOR/WHILE** <condición>]

A menos que se especifique otra cosa mediante **FOR** o **WHILE**, sólo el registro vigente será afectado. Si sufrió algún cambio un campo indexado, se deberá actualizar el archivo index.

BORRADO DE DATOS

DELETE

Marca los registros de la base de datos para su posterior eliminación mediante los comandos **PACK** o **ZAP**.

Sintáxis:

DELETE [<comodín> [**FIELDS**<lista de campos>]
[**FOR/WHILE** <comodín>]

Si no existe ninguna especificación a través de <comodín> o mediante **FOR** o **WHILE**, sólo se elimina el registro donde se encuentra el cursor. Los registros marcados para su eliminación en la base de datos son marcados por dBase con un asterisco (*). La combinación ^U actúa como switch que marca/desmarca los registros para su eliminación.

PACK

Elimina en forma permanente todos los registros previamente marcados con un asterisco (*) para ser eliminados.

SET DELETED

Oculta (**ON**) o descubre (**OFF**) el proceso de los registros marcados para ser eliminados.

Sintáxis:

SET DELETED ON/OFF

ZAP

Elimina todos los registros de la base de datos activa o en uso.

VIAS ALTERNAS DE ENTRADAS DE DATOS

APPEND FROM

Adiciona registros de otros archivos a los archivos tipo base de datos.

Sintáxis:

APPEND FROM <archivo> [**FOR** <exp>] [**SDF**] [**DELIMITED**]

BROWSE

Este comando contiene un menú de asistencia, el cual permite la edición y la adición de registros a través de la utilización de ventanas en archivos de bases de datos. Despliega hasta 17 registros por pantalla con el número de campos posibles a visualizar. Sintaxis:

BROWSE [FIELDS <lista de campos>]

La opción **FIELDS** especifica los campos a desplegar y su orden de despliegue. La utilización de la combinación de teclas ^HOME despliega el menú de opciones que contiene lo siguiente:

- BOTTOM** Se posiciona en el último registro del archivo
- TOP** Se posiciona en el primer registro del archivo
- LOCK** Define el número de campos que permanecerán fijos en la parte izquierda de la pantalla.
- MENU** Habilita o inhabilita las teclas de control del menú en el campo **BROWSE**.
- RECORD #** Se posiciona en el número de registro especificado.
- FREEZE** Edita el contenido de un solo campo.

Las opciones utilizables del menú se muestran en texto inverso, así como los campos del registro sobre el que se está trabajando. Explicaciones adicionales aparecen en la parte superior e inferior de la pantalla. Las teclas de control del menú que se pueden utilizar en **BROWSE** son las siguientes:

Las teclas de dirección se desplazan a través de todas las pantallas.

Las combinaciones [Ctrl] [←] y [Ctrl] [→] desplazan los campos sobre las pantallas.

Las teclas [Pg Dn] y [Pg Up] se desplazan a través de todos los registros del archivo.

La combinación ^Y elimina aquello que esté donde se encuentre el cursor hacia el final del campo.

^U Marca y desmarca el registro.

SET CARRY

Habilita (ON) o inhabilita (OFF) la posibilidad de escribir el contenido del último registro adicionado mediante **APPEND**.

**CAPITULO IV
RECUPERACION DE DATOS
LISTA DE DATOS**

LIST

Se utiliza para revisar el contenido de un archivo tipo base de datos.

Sintaxis:

**LIST [OFF] <comodín> [<lista de expresiones>]
[FOR/WHILE <condición>] [TO PRINT]**

La opción **OFF** elimina el despliegue de los números de registros, <comodín> puede asumir los valores **ALL**, **NEXT n**, o **RECORD n**; <lista de expresiones> se utiliza para limitar la aparición de determinados campos en el despliegue de información. Las opciones **FOR** y **WHILE** se utilizan para desplegar solo los registros que cumplan la <condición>, según sea el caso. La opción **TO PRINT** se emplea para que el despliegue de información se obtenga en impresora.

DISPLAY

Este comando sirve para desplegar el contenido de un archivo tipo base de datos.

Sintaxis:

**DISPLAY [OFF] [<comodín>] [FOR/WHILE <condición>]
[TO PRINT]**

Si no se especifica nada por el <comodín> o por las opciones **FOR** y **WHILE**, solamente se despliega la información del registro en el cual se encuentra el apuntador. La opción **OFF** evita que el número de registro sea desplegado. Si se van a desplegar más de 20 registros, dBASE III PLUS hace pausas cada 20 registros desplegados seguidos de un mensaje "Press any key to continue" (Presione una tecla para continuar), esto permite revisar la información desplegada con más calma. Los títulos de las columnas son nombres de los campos o de las variables o las expresiones que se desean desplegar. El contenido de campos memo no será desplegado a menos que sea especificada su aparición en la lista de expresiones, en lugar del nombre del campo aparecerá **MEMO**; si se van a desplegar campos memo su contenido se despliega en 50 columnas.

CONSULTA DE DATOS

BROWSE

Es un comando que cuenta con un menú de asistencia, el cual permite la edición y la adición de registros a través de la utilización de ventanas en archivos de bases de datos. Despliega hasta 17 registros por pantalla con el número de campos posibles a visualizar.

Sintáxis:

BROWSE [FIELDS <lista de campos>]

LOCALIZACION DE DATOS

LOCATE

Efectúa una búsqueda secuencial sobre archivos tipo base de datos activos, hasta encontrar el final del archivo, o el primer registro que cumpla con la condición que se especifique.

Sintáxis:

LOCATE [<comodín>] FOR <condición>

CURSO: dBASE III PLUS
CAPITULO: V
CLASIFICACION DE DATOS

ORDENAMIENTO SORT

Una de las funciones más importantes de las computadoras es, precisamente el de ordenar y/o clasificar la información, de ahí que en algunos países, las computadoras sean llamadas "ordenadores".

En **dBASE III PLUS**, existen 6 métodos diferentes de ordenar la información. Un método usado es el comando **SORT** que tiene la siguiente sintáxis:

```
SORT TO <archivo nuevo> ON <campo1> [/A] [/C] [/D] [, <campo2> [/A]
[/C] [/D]....] [<comodin> | [WHILE <condición>] [FOR <condición>]
```

Cuando el comando **SORT** es ejecutado, **dBASE III PLUS** crea una segunda base de datos en el disco, usando la base de datos que se encuentre activa.

Por ejemplo:

- . USE bdirectorio
- . SORT TO bordenada ON baprpap

Una vez que se han dado los comandos anteriores, se tiene una copia de la base de datos del directorio de alumnos, con igual número de registros e igual estructura pero ya están ordenados alfabéticamente por apellido paterno, pero si en este momento se quiere un listado, la información se presentará en forma desordenada por que no se ha activado la base ordenada, por ejemplo:

- . USE bordenada
- . DISPLAY ALL

De esta forma se obtiene la información ordenada y además se renumeró la base de datos con el nuevo orden, de tal forma que coincide el número de registro con el orden de la información.

```
DIR
Database Files  # Records  Last Update  Size
BGRUPOS.DBF      0      01/24/89     322
BDIRECTO.DBF    184     01/31/89   44544
BDELEG.DBF      18     06/04/87    1024
BMONEDA.DBF     38     06/01/87    2560
BCURSOS.DBF     18     01/26/89    2560
BORDEN.DBF     184     01/31/89   44154
```

95164 bytes in 6 files.
229376 bytes remaining on drive.

```
.USE BDIRECTO
.SORT TO borden ON baprpat
borden.dbf already exists, overwrite it? (Y/N) Yes
    00% Sorted
    100% Sorted      184 Records sorted
```

6

```
. USE BORDER!
. DISPLAY ALL CAPrPAT,BNOMBRE,BEMPRESA
```

Record#	BAPrPAT	BNOMBRE	BEMPRESA
1	AGUILAR	MARIA ONELIA	SERVICETEC, S.A. DE C.V.
2	AGUILAR	RENE	ARRENDADORA AROS, S.A. DE C.V.
3	AGUILAR	MANUEL	
4	AGUIRRE	RENE	PADILLA ESPEJEL Y ASOCIADOS,S.A. DE C.V.
5	AGUIRRE	RAMON	DEPTO. DEL DISTRITO FEDERAL.
6	ALARDIN	MARCIAL	ELECTRONICA ZONDA, S.A. DE C.V.
7	ANSELMO	FERNANDO	PROD. SAN CRISTOBAL, S.A. DE C.V.
8	ARCE	MANUEL FRANCISCO	

Press any key to continue...

Record#	BAPrPAT	BNOMBRE	BEMPRESA
9	ARISTA	ISABEL	BOBINADORES UNIDOS, S.A. DE C.V.
10	ARREDONDO	ARTURO	VISTEAR, S.A. DE C.V.
11	ARTEAGA	YOLANDA	BOBINADORES UNIDOS, S.A. DE C.V.
12	BARRERA	COINTO	DISTRIBUIDORA INTEGRAL DE COMPUTACION
13	BAUTISTA	JORGE	ACABADOS Y ESTAMPADOS PLUS ULTRA, S.A.
14	BRIONES	IGNACIO	JHI, S.A. DE C.V.
15	BUSTANI	JOSE	
16	CABRERA	ROBERTO	DESP. CABRERA Y ASOC., S.A. DE C.V.
17	CADENA	ROBERTO	JHI, S.A. DE C.V.

Press any key to continue...

Record#	BAPrPAT	BNOMBRE	BEMPRESA
.	.	.	.
.	.	.	.
.	.	.	.
.	.	.	.
.	.	.	.

Etc..

COMANDO INDEX

El otro método para ordenar registros es el comando **INDEX**.

INDEX ON <expresión> TO <archivo de índices>

El comando **INDEX**, no crea otra base de datos, ni ordena físicamente a los registros de la base de datos en uso, lo que hace es generar una tabla de índices, en la que continen la llave (es un campo o expresión válida de la base de datos en uso), por la que fué indexada y la dirección del registro en la base de datos. Está tabla es guardada en el disco con la extensión **.NTX**.

El archivo de index, reside en la memoria RAM, al momento de que se activa la base de datos con la cual esta relacionada, por la llave del index.

Como se genera un Index:

- 1) Activar la base de datos .**USE** bdirector
- 2) Crear el Index con la llave .**INDEX OR** bnumralumn **INTO** inum

De esta manera se crea el INDEX, quedando en la memoria el archivo de Index, y la información ya se encuentra ordenada.

. USE BDIRECTO

. display structure

Structure for database: A:BDIRECTO.dbf

Number of data records: 184

.Date of last update : 01/31/89

Field	Field Name	Type	Width	Dec
1	BNUMrALUMN	Numeric	6	
2	BAPrPAT	Character	15	
3	BAPrMAT	Character	15	
4	BNOMBRE	Character	20	
5	BPROFESION	Character	5	
6	BEMPRESA	Character	40	
7	BPUESTO	Character	20	
8	BDiReMP	Character	40	
9	BCOLreMP	Character	25	
10	BDELreMP	Numeric	2	
11	BCIUDreMP	Character	15	
12	BEDOreMP	Character	10	
13	BCPreMP	Character	5	
14	BTELreMP	Character	7	
15	BTEL2reMP	Character	7	
16	BEXTreMP	Character	4	
** Total **			237	

. INDEX ON BNUMrALUMN TO INUM

INUM.ndx already exists, overwrite it? (Y/N) Yes

00% indexed

100% indexed 184 Records indexed

. DISPLAY ALL BNUMrALUMN,BAPrPAT,BAPrMAT,BNOMBRE

Record#	BNUMrALUMN	BAPrPAT	BAPrMAT	BNOMBRE
1	1	BARRERA	LIBRADO	COINTO
2	2	MARTINEZ	SALDAÑA	SERGIO
4	2	AGUIRRE	VELAZQUEZ	RAMON
3	3	DOMINGUEZ	PASTRANA	LEONARDO
5	3	COQUET	RAMOS	JUAN BENITO
6	4	CALVO	MARROQUIN	MARIO
7	5	SERRANO	MIGALLON	FERNANDO
8	6	RETA	MARTINEZ	CARLOS
9	7	GIL	ELIZONDO	JUAN
10	9	ARCE	RINCON	MANUEL FRANCISCO
11	10	COSIO	VIDAURR	GUILLERMO
12	11	FUENTES	BOVE	MANUEL
13	12	VELEZ	BAÑUELOS	RODOLFO
14	13	VELDEZ	DELGADILLO	FRANCISCO
15	14	BUSTANI	HID	JOSE
16	15	HERNANDEZ	VALENZUELA	FERNANDO

Press any key to continue...

*** INTERRUPTED ***

. display status

Currently Selected Database:

Select area: 1, Database in Use: A:BDIRECTO.dbf Alias: BDIRECTO

Master index file: A:INUM.ndx Key: BNUMrALUMN

Alternate file: A:EJINDEX.txt

File search path:

Default disk drive: A:

Print destination: PRN:

Margin = 0

Current work area = 1

Press any key to continue...

*** INTERRUPTED ***

Una vez que se ha creado el archivo de indices, es necesario que al activar la base de datos se también activado el archivo de indices. Para activar el index se puede hacer de dos maneras, una activando el index al mismo tiempo que la base de datos.

.USE bdirector INDEX inum

O bien se puede traer la base de datos sola y despues el archivo o archivos de indices necesarios.

.USE bdirector

.SET INDEX TO inum, idire

23

REGLAS DE OPERACION DE LOS COMANDOS.

Cuando utilizamos el lenguaje de comandos de DBASE III existen una serie de reglas que se recomienda observar para estar seguros de que los comandos se escriben apropiadamente:

- 1.- Cada línea de comando, deberá comenzar con un verbo tomado del grupo de los comandos válidos.
- 2.- En la utilización de comandos, debemos seguir la sintáxis previamente descrita para cada comando.
- 3.- Un comando consiste de un verbo y opcionalmente una o más cláusulas que califican este verbo. Las cláusulas opcionales (específicamente las que comienzan con **FOR**, **WHILE** y **NEXT**), pueden ocurrir en cualquier orden. Por ejemplo, las siguientes expresiones son equivalentes:

DISPLAY NEXT 25 FOR NOMBRE = 'PEREZ'

DISPLAY FOR NOMBRE = 'PEREZ' NEXT 25

- 4.- La máxima longitud de un comando es **254** caracteres.
- 5.- Las palabras en una línea de comandos pueden estar separadas por cualquier número de espacios en blanco, sin embargo, estos espacios también se contabilizan en el límite de **254** caracteres.
- 6.- Los comandos y algunas palabras clave pueden ser abreviados, a los primeros 4 caracteres (por ejemplo, **DISPLAY MEMORY** puede ser abreviado como **DISP MEMO**).
- 7.- Los verbos de los comandos, las palabras clave (**STATUS**, **MEMORY**), los nombres de los campos, los nombres de las variables de memoria, y los nombres de los archivos, pueden ser referenciados escribiendolos en mayúsculas, minúsculas, o, en cualquier combinación de estos dos tipos de letra.
- 8.- Aunque DBASE III no prohíbe la utilización de cualquier palabra, es preferible evitar la utilización de las palabras clave como nombres de archivo, nombres de campo, o variables de memoria, ya que esto puede causar dificultades y, confuciones. Por ejemplo hagamos la suposición de que un campo de un registro cualquiera, se llama **STATUS**, no podría ser desplegado por el comando **DISPLAY STATUS** por que este comando despliega el estado de procesamiento actual, en lugar del contenido de este campo llamado **STATUS**.

EXPRESIONES

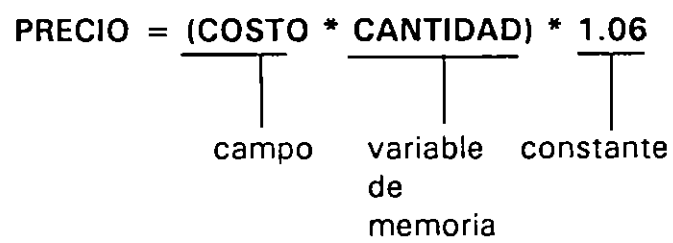
El termino expresión se utiliza ya que se refiere a un concepto más amplio que el de un campo, ya que DBASE III puede operar sobre expresiones, las cuales son mas complejas que un campo, por si mismo. Una expresión puede estar formada por cualquier combinación de:

- campos
- variables de memoria
- constantes
- funciones
- operadores

Ejemplo:

Una base de datos tiene un campo numérico llamado COSTO Una variable de memoria denominada CANTIDAD, contiene la cantidad vendida. El porcentaje de impuesto correspondiente es 6%. Esto significa que el precio total de, la cantidad a pagar se encierra en la siguiente expresión:

EXPRESION



Este es un ejemplo de utilización de una variable numérica, en la sección de referencia a comandos puede ser, descrita como; expresión, exp, o Nexp, dependiendo del comando o función y del espacio que se tenga disponible.

TIPOS DE OPERADORES.

Dbase III cuenta con cuatro tipos de operadores y son.- matemáticos, relacionales, lógicos, y de cadena.

1.- *Operadores Matemáticos:* generan resultados numéricos.

+	Adición
-	Substracción
*	Multiplicación
/	División
** ó ^	Exponenciación
()	Para agrupar

2.- *Operadores Relacionales:* El resultado que se obtiene es verdadero o falso. Estos operadores de comparación pueden ser utilizados en expresiones numéricas, de carácter o de fecha, pero al utilizarlos, ambas expresiones deben ser del mismo tipo.

<	Menor Que
>	Mayor Que
=	Igual
< > ó #	Diferente
< =	Menor o Igual
> =	Mayor o Igual

3.- *Operadores Lógicos:* Obtiene un resultado lógico de comparar dos expresiones lógicas.

.AND.	AND lógico
.OR.	OR lógico
.NOT.	NOT lógico
()	Paréntesis para Agrupar

4.- *Operadores de Cadena:* Estos operadores trabajan exclusivamente con cadenas de caracteres para lograr, la unión y poder determinar a partir de una comparación si dos cadenas de caracteres son iguales.

,	(coma) Operador de concatenación: Se utiliza para la unión de dos o más cadenas de caracteres, dando como resultado una cadena más larga que contiene a la cadenas unidas.
\$	Comparación de subcadena. (por ejemplo, si A, y B son cadenas de caracteres, A\$B me devuelve un valor lógico de verdadero si A, es idéntica o esta contenida en B.

PRECEDENCIA DE EJECUCION DE OPERADORES.

Cada tipo de operador, tiene un conjunto de reglas que gobiernan el orden en el cual serán ejecutadas las operaciones. Esto se refiere al concepto denominado niveles de precedencia de operadores.

Los operadores de comparación de cadenas, tienen un solo nivel de precedencia. Los operadores en estas categorías, se ejecutan de izquierda a derecha.

Los niveles de precedencia para operadores matemáticos son:

- 1.- +/- Unario
- 2.- Exponenciación
- 3.- Multiplicación/División
- 4.- Adición/Substracción

Los niveles de precedencia para los operadores lógicos son:

- 1.- .NOT.
- 2.- .AND.
- 3.- .OR.

Cuando algunos de estos.- tipos se utilizan en una misma expresión, los niveles de precedencia para estos tipos son:

- 1.- Matemáticos y Cadena
- 2.- Comparación
- 3.- Lógicos

Todas las operaciones del mismo nivel de precedencia son ejecutadas en orden de izquierda a derecha. Paréntesis son utilizados para alterar el orden en el cual se efectúan las operaciones.



FACULTAD DE INGENIERIA, UNAM.
DIVISION DE EDUCACION CONTINUA

INTRODUCCION A LAS COMPUTADORAS PERSONALES

MATERIAL DE APOYO DIDACTICO

FEBRERO 1995

COMPONENTES DE UN SISTEMA COMPUTACIONAL

Un sistema computacional lo podemos dividir en dos partes básicas: **SOFTWARE Y HARDWARE**.

SOFTWARE

Consiste en todas las secuencias de código y programas que dan instrucciones al hardware para desarrollar varias actividades para el usuario.

El software a su vez, se puede dividir en tres principales:

**SISTEMAS OPERATIVOS.
LENGUAJES DE PROGRAMACION.
SOFTWARE DE APLICACION.**

SISTEMAS OPERATIVOS

Es usualmente dado por el fabricante del sistema. Es el código y programas necesitados para hacer que el hardware desarrolle sus funciones generales.

El sistema operativo es una colección de programas que actúa como interfase entre el hardware y el usuario. En general el Sistema Operativo tiene dos funciones:

a) Provee a los usuarios de una variedad de facilidades para simplificar su uso, diseño y mantenimiento de programas de aplicación.

b) Controla el uso del hardware y los recursos del sistema para asegurar una operación eficiente.

LENGUAJES DE PROGRAMACION

Nos permite desarrollar nuestra propia aplicación, desarrollando las tareas que en específico le hemos indicado. Estas tareas las desarrollará de acuerdo a las entradas de datos que se le este dando y obtendremos la salida de acuerdo a las necesidades del usuario del sistema. Estas aplicaciones se desarrollarán de acuerdo a la sintaxis del lenguaje que se ha escogido para la aplicación. Por esto existen diferentes tipos de lenguajes, algunos con fines científicos, administrativos, etc. Entre los diferentes lenguajes podemos encontrar Pascal, Quick Basic, Fortran, Cobol, C, etc.

SOFTWARE DE APLICACION

Se conoce también como paquetería. La cual es una aplicación en una forma general, es decir, la puede utilizar cualquier persona y adptarla a sus necesidades, ya que la aplicación da muchas herramientas en un solo paquete. Entre algunas aplicaciones podemos encontrar hojas electrónicas, procesadores de palabras, bases de datos, graficadores etc.

HARDWARE

Consiste en todas las partes físicas del sistema de cómputo. Estas partes las podemos dividir en cuatro grupos principales: Dispositivos de entrada, dispositivos de salida, dispositivos de entrada\salida y dispositivos de almacenamiento.

DISPOSITIVOS DE ENTRADA

Los dispositivos de entrada los podemos definir como aquellos que nos permiten entrar en contacto el usuario con la computadora. Entre éstos podemos encontrar: el teclado.

DISPOSITIVOS DE SALIDA

Son los dispositivos que dan salida a datos que el usuario requiere. Entre los dispositivos de salida tenemos: monitor, impresora, etc.

DISPOSITIVOS DE ENTRADA\SALIDA

Estos dispositivos permiten tanto la entrada de datos del usuario, como la salida de datos que el usuario requiere, permite cambio en los datos y reemplazos, además de anexar más datos y permitir revisión de los mismos. Entre estos tenemos; drives, que permiten la lectura y escritura de los discos flexible, y cualquier dispositivo que permita la lectura y escritura de datos en dispositivos de almacenamiento.

DISPOSITIVOS DE ALMACENAMIENTO

Como su nombre lo indica son utilizados para el almacenamiento de información, la cual puede ser leída las veces que sean necesarias. Entre los dispositivos de almacenamiento tenemos: discos duros y flexibles, cintas, tambores, etc.

ESPECIFICACIONES DEL HARDWARE

TECLADO

El teclado actúa como un puente entre el usuario y la máquina, ya que realiza la función de comunicarlo con la computadora. A través del teclado el usuario puede darle instrucciones u órdenes (command, en inglés), o también darles datos para que se procese.

El teclado se divide en tres partes principales:

1) 10 teclas de función, de F1 a F10. Estas teclas varían su función de acuerdo al software con el que se esté trabajando, ya sea sistema operativo, lenguaje o paquetes.

2) Teclas alfanuméricas, es decir, teclas que incluyen 26 letras del alfabeto (A-Z), números (0-9), signos de puntuación (., etc.), ortográficos (' "~ , etc.), aritméticos (+, - , etc.).

3) Teclas numéricas y de movimiento de cursor. Estas teclas se encuentran distribuidas de forma similar a una calculadora.

Existen además teclas de control, las cuales se encuentran distribuidas en todo el tablero, como: insert (ins), delete (del), Escape (esc), Tabulador (tab), Return (ret), etc. Donde cada una tiene una función específica de acuerdo al software que se esté usando.

A continuación se explicará la función de algunas teclas importantes para el Sistema Operativo D.O.S., propósito de éste manual.

F1

Copia y visualiza un caracter de la línea de comandos previamente establecida.

F2

Copia todos los caracteres hasta uno señalado de la línea previa de comandos.

F3

Copia todos los caracteres restantes de la línea en curso.

F4

Elimina todos los caracteres hasta caracter especificado de la línea en curso.

ESC

Es una abreviación de ESCAPE. Su función es ignorar la orden o dato que acabas de escribir. En algunas ocasiones ésta línea es borrada de la pantalla y en otras el caracter "\" aparece al final de la línea y el cursor se coloca en la línea inmediata siguiente.

TAB

Actúa como tabulador, similar a una máquina de escribir, ya que mueve el cursor al siguiente tabulador marcado.

SHIFT

Existen dos teclas shift, que cumplen con la misma tarea. Su función es escribir ya sea en mayúscula, obtener los símbolos que se encuentran marcadas en la parte superior de las teclas numéricas y de puntuación.

BACK SPACE

Borra el caracter en el que se posiciona el cursor. Y todos los caracteres que se encuentran a la izquierda de donde está el cursor, se recorren a la izquierda una posición por caracter borrado.

ALT

Abreviación de ALTERNATE. Su función es similar al de las teclas de SHIFT Y CTRL, es decir, cambia el significado de otra tecla cuando se presionan simultáneamente.

RETURN

Su propósito es terminar la línea que acabas de escribir y avanza a la siguiente línea. Es decir, mueve el cursor del último caracter de la línea al primer caracter de la línea siguiente.

CAPS LOCK

Se encuentran en la parte derecha del teclado. Y permite que escribas, una vez que se ha presionado, todo el tiempo en letras mayúsculas, sin necesidad de presionar SHIFT. Sólo afecta a los signos de puntuación. Cuando CAPS LOCK ha sido presionado o no, es decir, si esta activo o no.

SCROLL LOCK

No realiza ninguna función si se utiliza sola. En la mayoría de los casos se usa con CTRL, lo que hace que la computadora pare todo lo que esté haciendo y en espera de alguna orden. Si éste ha sido presionado se indicará con un indicador luminoso. Esta tecla también se encuentra en la parte derecha del teclado.

NUM LOCK

Si éste es presionado se encenderá el indicador luminoso, que se encuentra en la parte superior derecha del tablero. Su función es permitirnos usar los números que se encuentran a la derecha del teclado, si éste es nuevamente presionado el indicador luminoso se apagará y no permitirá el uso de los movimientos de cursor, sino que se usará como teclas numéricas. El movimiento de cursor en cada una de la teclas es indicado por las flechas marcadas.

CTRL

Esta tecla por si sola no tiene ninguna función. Para darle significado tiene que ser presionada con otra tecla simultáneamente, con lo que llevará a cabo alguna función, que dependerá del sistema en que se encuentre trabajando.

La función de la siguientes teclas en el Sistema Operativo es nulo.

HOME

Su función es mover el cursor a la primera línea en la primera posición de la pantalla.

END

Mueve el cursor al final de la línea en la que nos encontramos trabajando.

INSERT

Es un switch que indica a la terminal si se encuentra en modo de inserción o no. Si nos encontramos en modo de inserción se pueden insertar caracteres en cualquier parte de la línea, sólo se tiene que colocar el cursor en el lugar deseado.

PG UP

Su función es movernos entre diferentes páginas de un texto. En éste caso será hacia arriba, es decir en páginas anteriores de donde nos encontramos situados.

PG DN

Abreviación de PAGE DOWN. Su función es la misma que la tecla anterior, sólo que aquí el movimiento es hacia abajo, es decir las páginas posteriores del texto.

DEL

Tiene la misma función que BACK SPACE, sólo que en el sistema operativo ésta tecla no tiene ninguna función.

MONITOR

Dispositivo de salida que permite al usuario revisar la entrada de sus datos y también la salida de datos.

TIPOS

Existen dos tipos de monitores:

Monocromático (un solo color) y policromático (diferentes colores).

Los monitores monocromáticos son de color verde o ámbar.

MODOS DE OPERACION

Existen dos modos de operación, el modo gráfico y de texto. En el modo gráfico existen tres tipos de resoluciones:

Baja resolución	320 x 200 puntos
Media resolución	640 x 200 puntos
Alta resolución	640 x 400 puntos

En el modo de texto existen dos modos de visualización:

25 líneas x 40 columnas

25 líneas x 80 columnas

IMPRESORA

2 Dispositivos de salida que presenta en forma impresa la información que el usuario requiere, lo que le permite a éste consultarla el número de veces que sea necesaria.

Existen diferentes tipos de impresoras:

DE MARGARITA

Esta impresora usa un mecanismo de rueda de margarita para generar caracteres sólidos de alta calidad. Su mecanismo es similar a una máquina de escribir. Su funcionamiento es mucho más lento que las impresoras de matriz y son también de costo más elevado, además de que tiene la desventaja que no permiten el modo de graficación.

DE MATRIZ

En este tipo de impresora los caracteres son compuestos de puntos creados al accionar un grupo de agujas, las que, en diferentes combinaciones, forman los diferentes caracteres. La mayoría son bidireccionales, es decir, escriben de izquierda a derecha y de derecha a izquierda, además de que son graficadoras. Este tipo de impresoras son las más usadas por su precio que es más accesible.

DE CHORRO DE TINTA

Como su nombre lo indica, disparan pequeñas cantidades de tinta que van dirigidas de acuerdo al carácter que tengan que imprimir, y están dirigidas por medios magnéticos.

TERMICAS

La cabeza de la impresora transmite calor a un papel sensible, de modo que cambia el color en las áreas afectadas. Son silenciosas y de costo razonable.

DE RAYO LASER

Es una variante de la anterior, su funcionamiento es a través de un haz de luz sobre la superficie del papel.

Para poder comenzar a hablar de los dispositivos de almacenamiento, tenemos que comenzar por hablar de la unidades en que se manejan, es decir en la unidades en las que se comercializan, que son: bits y bytes.

BIT Y BYTE

Es la mínima unidad de información, ya que 8 bits forman un byte, el cual a su vez forma un caracter. Estos bits se manejan en forma binaria, es decir sólo podrán tener dos valores posibles: 0 ó 1, y es así como se guarda la información.

Un KiloByte se encuentra formado por 1024 bytes, que es una de las unidades en que se comercializan los diskettes o discos flexibles. Ya que estos tienen una capacidad de 360 Kbytes. Es decir $360 \text{ Kbytes} \times 1024 \text{ bytes}$ es 368,640 bytes de capacidad.

Y un Megabyte son 1024 Kbytes. Los discos duros más comunes que se comercializan tienen una capacidad de 20 Megabytes, lo que será $20 \text{ Mbytes} \times 1024 \text{ Kbytes} \times 1024 \text{ bytes}$ que es igual a 20,971,440 bytes de capacidad.

DISCOS

Existen dos tipos de discos: Flexibles y Duros. Los discos flexibles son usados cuando la información no es demasiada, o se esté necesitando la información en diferentes lugares, lo cual haría más fácil su transportación.

Los discos duros son usados por empresas con volúmenes grandes de información, que no necesitan estar moviendo su información.

Sin importar que tipo de disco es su división para almacenar la información es la siguiente:

PISTA

Se divide al disco, que es de forma circular en Pistas concéntricas, similares a un L.P.

SECTOR

Se divide al disco en partes semejantes a una rebanada de pastel, partiendo desde el centro.

TRACK

Es la intersección entre una pista y un sector. Por sus capacidades los discos flexibles los dividimos en una densidad y doble densidad.

Una densidad
Doble densidad

Por sus dimensiones, los discos flexibles se dividen en discos de 5 1/4 pulg. y de 3 1/2" pulg., su uso dependerá del tipo de disco que utiliza la máquina.

MEMORIA ROM (Read Only Memory)

Memoria de sólo lectura. Tiene programas almacenados que son hechos por el fabricante de la máquina, su función es darle arranque a la computadora, Estos programas son fijos y el usuario no puede entrar a éstos, por lo tanto no podrá modificarlos.

El objetivo de la memoria ROM es darle arranque al sistema, revisando el funcionamiento del hardware, alistando al sistema para que el usuario pueda empezar a utilizarlo.

MEMORIA RAM (Random Acces Memory)

Memoria de acceso aleatorio. Es una memoria interna en que las instrucciones de programas o datos que se estén almacenando son guardados temporalmente antes de guardarlos en dispositivos de almacenamiento, es decir, discos o cintas. Esto le da más rapidez, ya que mientras se está realizando cualquier función no se tiene que ir a dispositivos a tomar los datos que se necesitan, ya que los tenemos cargados en memoria RAM. Entre más memoria RAM se tenga más capacidad de almacenamiento temporal se tiene.

Dibujo de Unidad principal con dos drives uno de lado derecho izquierdo y otro con drives superior e inferior, indicando el prompt A o B y otra unidad.

NOTA:

En caso de que su computadora no se encuentre en ninguno de los dos casos anteriores, su computadora seguramente se encuentra conectada a una red, con lo cual el sistema operativo ya esté cargado, y el prompt cambie de acuerdo a la distribución de la red.

ORDENES EXTERNAS E INTERNAS

Cuando el Sistema Operativo se carga en la computadora las órdenes más usadas quedan almacenadas en memoria RAM, lo que hace que cuando se requiera alguna orden se lea de esta memoria y sea más rápida su ejecución. estas órdenes se clasifican como órdenes internas. Todas las **órdenes internas** se encuentran en el archivo Command.com.

Las **Ordenes Externas**, no se consideran muy usadas por lo que no se cargan en memoria RAM, sino que se dejan almacenadas en el disco y cuando se utilicen, éstas serán leídas del disco y luego ejecutadas, lo que lo hará más lento. Por ésta razón son externos. Además de que en disco se encuentra un archivo con el nombre de ésta orden.

NOTA:

Para cada una de las órdenes se indicará si son externas o internas. Nomenclatura

[]

Indica que la entrada que se encuentra entre ellos es opcional.

< >

Si lo que está entre ellos está en

Minúsculas: son datos necesarios que no se pueden omitir.

Por ejemplo:

<nombre archivo> es el nombre del archivo que no se puede omitir.

Mayúsculas: es la tecla que se deberá de oprimir.

Por ejemplo:

<RET> No se debe de oprimir la tecla RETURN.

/

La barra diagonal [slash], se emplea para indicar que lo que sigue es un modificador de la orden.

SINTAXIS

La sintaxis nos indicará como se debe de dar la orden de ejecutar. Es el cómo debemos de escribir la oración. La sintaxis de una orden la podemos dividir para éste primer bloque en 3 partes importantes:

[Unidad Fuente] ORDEN [Unidad Destino] [/Modificadores]

UNIDAD FUENTE:

Es la unidad disco en la que nos encontramos trabajando, o donde se encuentra el archivo se es externo. Esto se puede omitir, y si se omite tomará como unidad fuente la unidad en la que nos encontramos trabajando.

ORDEN:

Es la instrucción a ejecutar.

UNIDAD DESTINO:

Indica en que unidad se va a dejar los resultados de la acción realizada, se puede omitir y se toma por omisión la unidad en la que nos encontramos trabajando.

MODIFICADORES:

Son los modificadores de la acción, se pueden omitir también, de acuerdo a como queremos que se realice la acción.

ACCION

RESPUESTAS/EXPLICACION

Encendido de la computadoras.

1. Encender regulador.

2. Colocar el disco del
Sistema Operativo en
la Unidad A

3. Encender la unidad
principal y el monitor.

Si se cuenta con él.

Lo primero que se realiza es chequeo que realiza la computadora de memoria RAM (SELF TESTING). y posteriormente la computadora busca el archivo COMMAND.COM para alistar el sistema.

Una vez cargado el sistema, lo primero que pide es la fecha. Que es el primer comando de éste bloque.

Date

Se visualizará en la pantalla lo siguiente:

Current Date is Tue 1-01-1980
Enter new date:

Esta es la fecha que se tiene que cambiar. Para dar la fecha actual, el formato es el siguiente:

mm-dd-aa o mm/dd/aa

ACCION

RESPUESTAS/EXPLICACION

Teclear:
mm/dd/aa <RET>

mm mes actual
dd día actual
aa año actual,
pueden ser los dos
últimos dígitos o los
cuatro.

Si se comete error de
teclear <RET>, puede
regresarse a corregir
con backspace y rees-
cribir la fecha.

Si se comete error, es
decir una fecha que no
existe o algún caracter
no válido dentro de la
fecha, y se tecllea
RETURN, el sistema
volverá a pedir la fecha
para que se de el dato
correcto.

Si en lugar de dar la
fecha se da un RETURN,
el sistema tomará por
fecha la fecha que
indicó como actual y no
la modificará.

En éste caso será
Tue 1-01-1980.

Colocar la fecha
correcta es importante,
ya que ésta fecha se le
dará a los archivos que
se realicen o se modi-
fiquen esto nos permite
llevar control de cuando
se hicieron los archivos
o la fecha de la última
modificación del archi-
vo.

ACCION

RESPUESTAS/EXPLICACION

Time

Current Time is 0:00:14.03
Enter new time:

Teclear la hora actual:
hh:mm:ss:cc <RET>

Si la fecha fue correctamente tecleada o si de dió RETURN, el sistema preguntará por la hora correcta.

Se desplegará el siguiente mensaje:

El formato para escribir la hora correcta es:

hh hora
mm minutos
ss segundos
cc centésimas de seg.

De la misma manera que con la fecha, si hay error antes de dar <RET>, se puede regresar a corregir con BACKSPACE. Si no se corrige y se da RET, el sistema volverá a pedir la hora correctamente.

Si no se da la hora, y se teclea RET, el sistema tomará nuevamente la hora que ya tenía y no la modifica.

Su Importancia es la misma que la de la fecha, ya que la hora quedará grabada en cada uno de los archivos que se modifiquen o se realicen. Y sirve para llevar el control.

ACCION

RESPUESTAS/EXPLICACION

Prompt

Estas dos órdenes se pueden dar en cualquier momento, no necesariamente al principio. También sirven para pedir fecha u hora sin necesidad de modificarlas.

A>

Una vez aceptada la fecha y la hora, el sistema desplegará lo siguiente.

Es el prompt, que indica la unidad en la que nos encontramos trabajando. Además de que nos indica que se encuentra esperando una orden.

Para probar nuevamente las órdenes de fecha y hora.

Teclear:
A>DATE <RET>

Nota:
Las órdenes pueden escribirse en mayúsculas o en minúsculas.

A>TIME <RET>

Como lo único que se deseaba es ver la fecha y hora, no es necesario modificarlas, sólo darle RETURN.

ACCION

RESPUESTAS/EXPLICACION

CLS (Clear Screen)

Orden de limpiar pantalla. Su única función es limpiar la pantalla de lo que se tenía escrito, dejando el cursor en la parte superior izquierda de la pantalla.

Teclear:
A>CLS<RET>

Comando TREE.

Despliega en pantalla la estructura de los directorios grabados en disco.

GUIA RAPIDA DE COMANDOS DEL MS-DOS

Comandos internos.

Son aquellos comandos que están contenidos dentro de los componentes de una computadora (procesador central). Dichos comandos ya están grabados como información permanente en dichos componentes.

Comandos externos.

Son aquellos comandos que no están contenidos dentro de los componentes internos de una computadora, es decir, que son información que se tiene en dispositivos de almacenamiento externo (diskettes). Un ejemplo de dichos comandos, son todos los programas contenidos en los diskettes del sistema operativo (MS-DOS disk).

Ejemplo de comandos internos:

DIR (Directory)	Desplegar directorio.
DEL (Delete)	Borrar archivos.
COPY	Copiar archivos.
ERASE	Misma operación.
TYPE	Describe archivo.
MD (Make Directory)	Crear directorio.
RD (Remove Directory)	Remover directorio.
CD (Change Directorio)	Cambiar de directorio.
REN (Rename)	Renombrar archivo.

Ejemplo de comandos externos:

FORMAT.COM	Formateo de discos o diskettes.
DISKCOPY.COM	Copiado de diskettes.
ATTRIB.EXE	Modificar atributos de archivo
EDLIN.COM	Editor de líneas.
RECOVER.COM	Recupera archivos.
PRINT.COM	Comando de impresión.
TREE.COM	Despliega arbol de directorios.
MODE.COM	Modos de impresión.
BACKUP.COM	Crea un respaldo.
RESTORE.COM	Reescribe un respaldo
XCOPY.EXE	Copiado de archivos.
AUTOEXEC.BAT	Alta de dispositivos y comandos.

Uso de comodines.

Por comodín entendemos que es el caracter que nos ayuda a generalizar, abreviar y especificar nombres de archivos que se tengan que copiar, borrar, respaldar, etc.. El comodín se representa por el caracter (*) que conocemos con el nombre de asterisco. Ejemplos:

- 1.- *.* Indica todos los archivos con cualquier extensión.
- 2.- *.BAK Indica todos los archivos con extensión BAK.
- 3.- CAT0.* Indica los archivos CAT0 con cualquier extensión.

USO DE COMANDOS BASICOS

(INTERNOS)

Comando DIR.

El uso de este comando es muy sencillo contando además con algunas opciones.

- 1.- DIR (desplegar directorio en forma simple).
- 2.- DIR/P (desplegar directorio en forma pausada).
- 3.- DIR/W (desplegar directorio en forma horizontal).
- 3.- DIR PRN (listar directorio en impresora).

Comando DEL.

Este comando se utiliza para borrar archivos, ya sean específicos, único o todos los contenidos en un directorio.

- 1.- DEL ARCHIVO.EXT (borra archivo único).
- 2.- DEL ARCHIVO.* (borra archivos específicos).
- 3.- DEL *.* (borra todo archivo en directorio).

SINTAXIS PARA MANEJO DE COMANDOS Y ARCHIVOS.

- Por nombre de archivo entendemos que es la clave de identificación de un archivo.
- Por extensión de un archivo entendemos que es parte de la identificación de un archivo, pero que puede indicarnos su origen y contenido, es decir, que nos sirve para darnos cuenta si es un escrito, una tabla, un programa o un gráfico o dibujo.
- El DOS reconoce para el nombre de un archivo hasta un máximo de 8 caracteres.
- El dos reconoce para la extensión de un archivo hasta un máximo de 3 caracteres.

Ejemplos:

Nombre	Extensión	No.
ARCHIVO	.EXT	7 caracteres en nombre y 3 en extensión.
SAE915-B	.001	8 caracteres en nombre y 3 en extensión.
ARCO	.TXT	4 caracteres en nombre y 3 en extensión.
INVESTIG	.LA	8 caracteres en nombre y 2 en extensión.
CAT-91	.A	6 caracteres en nombre y 1 en extensión.
CARSA		5 caracteres en nombre y 0 en extensión.

Notas: Observe que no se puede omitir el nombre del archivo pero en el caso de la extensión si se puede omitir. Además, el punto después de el nombre del archivo funciona como un separador de la extensión. Al hacer uso de comandos para manejar archivos, el copiar o borrar archivos requiere del uso del punto separador en caso de existir extensión, pero en caso de no contar con una extensión, el punto separador se omite.

Comando MD.

Sirve para crear directorios, su uso es muy simple, aunque sólo se tiene que tener cuidado en no repetir un nombre que ya exista en disco, de lo contrario en pantalla aparecerá un mensaje que indique que no se puede crear con el mismo nombre.

1.- MD CATALOGO (crea un directorio con nombre CATALOGO).

Comando RD.

Cuando un directorio ya no se use o tenga que ser borrado por otra razón, este comando nos ayuda a efectuar dicha operación, bastará que el directorio se encuentre vacío y desde el directorio inmediatamente anterior se indique el borrado.

1.- RD CATALOGO (remueve el directorio CATALOGO).

Comando CD.

Este comando es muy útil en lo referente a la "navegación" dentro del árbol de directorios.

1.- CD PROD90 (entra al directorio PROD90).

2.- CD PROD90\MAT (entra al subdirectorio MAT, dentro del directorio PROD90).

3.- CD\DIREC (entra a un directorio desde cualquier otro donde uno se encuentre).

Comando REN.

Este comando es muy útil en el manejo del archivo, algunas veces es necesario renombrar un archivo con fines de organización, o bien para evitar la duplicidad de archivo, es una herramienta indispensable para la organización del archivo.

1.- REN ARCH1.DOC ARCH-1.DOC (cambia nombre de un archivo).

Comando COPY.

Este comando se utiliza para copiar archivos, ya sean únicos, específicos o todos los contenidos en un directorio.

- 1.- COPY *.* A: (copia todo archivo a la unidad **A**).
- 2.- COPY CAT3.DOC A: (copia un archivo a la unidad **A**).
- 3.- COPY *.CAT A: (copia cualquier archivo con extensión CAT a la unidad **A**).

Nota: Se toma como convención el hecho de usar solamente la unidad **a**:

-El uso de mayúsculas es otra convención en los comandos y la indicación de archivos.

Comando ERASE.

Este comando tiene la misma función que DEL.

Comando TYPE.

Este comando sirve para desplegar información de archivos en pantalla sin tener que entrar a un programa, puede servir para identificar que tipo de archivo es, aunque cabe destacar que mucha de la información que se le solicite a la computadora la describa será ilegible para nosotros, puesto que puede estar gravada en código máquina. La información que sí será reconocida por lo general está con formato ASCII (tal como es).

- 1.- TYPE ARCHIVO.EXT (describe un archivo específico).
- 2.- TYPE CAT28 (describe un archivo sin extensión).

Comando ATTRIB.

Este comando cambia los atributos de lectura de archivos, es de gran ayuda para proteger archivos que puedan ser borrados.

- 1.- ATTRIB +R ARCH.EXT (protege archivo específico).
- 2.- ATTRIB +R *.* (protege todos los archivos).
- 3.- ATTRIB +R *.DOC (protege archivos con extensión DOC).
- 4.- ATTRIB +R ARCH.* (protege archivos con el mismo nombre y diferente extensión).

Nota: Para quitar el atributo se procede de la misma forma pero cambiando el símbolo (+) por el de (-).

Comando EDLIN.

Este comando sirve para crear, por ejemplo archivos con extensión BAT (lista de ejecución de comandos) que bien podemos llamar *subcomando*. Ejemplo:

- 1.- EDLIN LISTA.BAT **aparecerá el mensaje:**

New file

*

- 2.- Posteriormente se procede a "cargar" las instrucciones que la computadora seguirá cada que se le indique este subcomando, al inicio de cada archivo se indica el número correspondiente de instrucción. Ejemplo:

```
1i
1 DATE
2 TIME
3 CLS
4 PATH C:\DOS;C:\MOUSE;C:\WINDOWS
5 CD\MOUSE
6 MOUSE
```

- 3.- Una vez escritas las instrucciones deseadas se oprimen simultáneamente las teclas Ctrl y C.

- 4.- Por último, para salvar el archivo escriba una <E> seguida de un <Enter>.

EXTERNOS)

Comando FORMAT.

Para que un diskette pueda ser usado para gravar información en él, primero debe ser inicializado con FORMAT, aunque debemos recordar que existen distintos tipos de diskettes, en este caso veremos los más comunes.

Tipos de Diskettes:

- **TPI** Traks Per Inches (traks por pulgada).
- **DS** Double Sided (doble lado).
- **DD** Double Density (doble densidad).
- **HD** High Density (alta densidad).
- a) - Discos de 5 1/4" de 360 KB - 48 TPI (DS/DD)
- b) - Discos de 5 1/4" de 1.2 MB - 96 TPI (DS/HD)
- c) - Discos de 3 1/2" de 720 KB - 135 TPI (DS/DD)
- d) - Discos de 3 1/2" de 1.44 MB - 270 TPI (DS/HD)

1.- **FORMAT A:/4** (da formato en diskette a 360 de capacidad).

2.- **FORMAT A:** (da formato en diskette a 1.2 MB de cap.).

Nota: Sólo computadoras de tipo **AT** con drives **HD** pueden dar formatos de alta densidad.

Comando DISKCOPY.

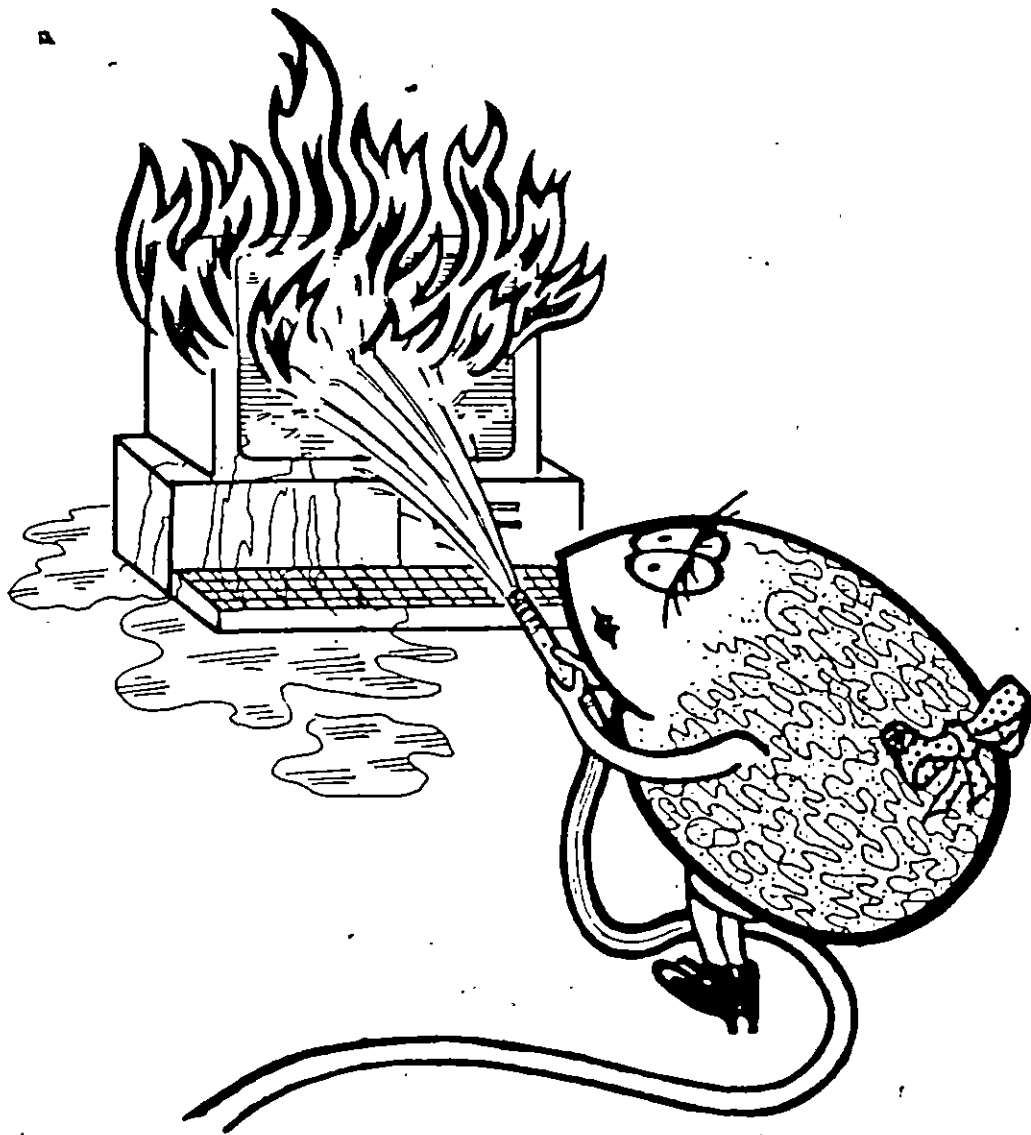
Este comando reproduce diskettes, es decir, hace copias de diskettes.

1.- **DISKCOPY A: B:** (copia disco de la unidad **A:** a otro disco en la unidad **B:**).

2.- **DISKCOPY B: A:** (copia disco de la unidad **B:** a otro disco en la unidad **A:**).

Nota: La mayoría de las computadoras que cuentan con disco duro y un solo drive reconocen su único drive como A: o B:.

3. Introducción al sistema operativo MS-DOS



3.1 Generalidades

El sistema operativo MS-DOS mantiene muchas características de su antecesor, el CP/M, del que se puede considerar que hereda sus pretensiones de favorecer la transportabilidad.

Externamente mantiene un gran parecido en cuanto a:

— el uso de las convenciones en el nombre de los archivos

— la utilización de los comodines
— el vocabulario de las órdenes.

Está diseñado para funcionar con un microprocesador de 16 bits, y una de sus versiones, el IBM-DOS, es la que se adoptó para los diferentes modelos de ordenadores personales IBM, lo que ha contribuido a su gran popularidad. (Hoy día muchos ordenadores personales lo han adoptado.) El IBM-DOS y el MS-DOS son prácticamente iguales desde el punto de vista del usuario y en adelante,

para simplificar, nos referiremos a ellos con el nombre de *MS-DOS*.

3.2 La estructura en tres niveles

El conjunto de programas que constituyen el sistema operativo *MS-DOS* está agrupado en tres niveles:

1. El *COMMAND*
2. El *D.O.S.*
3. El *B.I.O.S.*

En el nivel más externo, o próximo al usuario, se halla el *COMMAND* o interpretador de órdenes cuya función principal es editar o procesar las órdenes; contiene, además, las llamadas órdenes residentes y los procedimientos generales del sistema operativo (carga y ejecución de programas). Este programa, el *COMMAND.COM*, utiliza rutinas del nivel inferior, el *D.O.S.*, descrito a continuación.

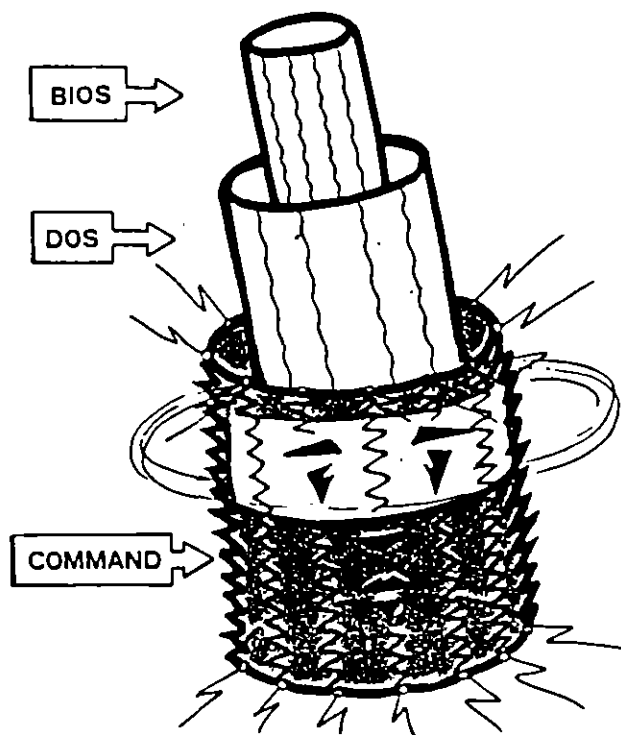


Fig. 3.1 Los programas del S.O. forman un conjunto muy unido.

En el nivel inmediatamente inferior está el *D.O.S.*, propiamente dicho, cuyo núcleo fundamental se encarga de controlar la(s) unidad(es) de disco(s), interviene en la creación y manipulación de los archivos y también incluye una serie de rutinas de «servicio» que se utilizan de vez en cuando desde los programas de aplicación.

En el estrato inferior, más cercano al equipo físico, está el *B.I.O.S.*, cuya función es la de relacionarse directamente con el ordenador y los periféricos, aunque para estas operaciones utiliza las rutinas especializadas de la *ROM-BIOS*. Es el intermediario entre el *D.O.S.* y los periféricos y hace de centinela detectando cuándo se producen situaciones de error para avisar al *D.O.S.* a fin de que se tomen las medidas necesarias para que no se produzcan daños irreparables al sistema.

3.3 La configuración del sistema

En los ordenadores compatibles *PC*, que son los principales usuarios del sistema operativo *DOS*, el *S.O.* suele estar guardado en un disquete o en el disco duro. Esto supone que al poner marcha el ordenador hay que cargar o guardar en la memoria interna los archivos de que consta. Esta operación puede efectuarse de este modo:

1. Apagar el ordenador.
2. Colocar el disquete del *MS-DOS* en la unidad A.
3. Encender el ordenador.

Inmediatamente después del encendido, el ordenador ejecuta las verificaciones internas que correspondan y pasa a leer el *S.O.* del disco de la unidad A. En el caso de que estuviera vacía o el disco no contuviera el sistema operativo producirá un mensaje de error. Después de leer el *S.O.*, este mismo hace que se ejecute el programa *AUTOEXEC.BAT*, si existe, y seguidamente muestra el cursor del *MS-DOS* indicando que espera un orden del usuario:

A>

En argot informático, el término *arranque* se utiliza para indicar la carga del sistema operativo desde el principio, sin haber desc

tado el ordenador. Es algo similar a lo que ocurre cuando se enchufa el ordenador, pero evitando las verificaciones internas de los circuitos; el segundo método es el que efectúa normalmente cada vez que se pone en marcha el ordenador y se le llama comúnmente *arranque en frío* —antes de cargar el S.O. comprueba el estado del sistema informático.

Según sea el sistema operativo, el ordenador mostrará al usuario una apariencia externa distinta en algunos de sus aspectos. Los dos más evidentes son:

- el significado de las teclas y
- el cursor o la unidad de discos activa.

3.3.1 El teclado

Es el periférico de entrada más utilizado y el que se asume siempre por defecto. Actualmente es el medio más corriente para establecer comunicación entre el usuario y el ordenador.

El sistema operativo puede modificar el significado de las teclas. Es corriente asignar a algunas teclas de función, o a grupos de otras teclas usadas conjuntamente, la capacidad de facilitar la edición de comandos.

Teclas	Significado
F1	copia un carácter de la última línea que se escribió
F3	copia toda la línea
CTRL-C	cancela la ejecución de una orden
CTRL-NUMLOCK	detiene momentáneamente la ejecución del programa, que continúa cuando se pulsa cualquier tecla
SHIFT-PrtSc	imprime sobre papel el contenido de la pantalla del monitor
CTRL-P	direcciona la salida de información hacia la impresora
CTRL-ALT-DEL	arranque del sistema "en caliente", carga el sistema operativo

Fig. 3.2 Significado de algunas teclas especiales en el sistema operativo MS-DOS.

Como todos los teclados de ordenador se fabrican adaptados a las peculiaridades de la lengua inglesa, cuando se pretende escribir en otros idiomas surgen inconvenientes por la falta de algunos caracteres (Ñ, Ç, acentos...). Para superar ese pro-

blema el S.O. puede encargarse de ejecutar un programa específico para remodelar el teclado adaptándolo a las nuevas necesidades. Hay versiones de estos programas para varias lenguas (el que configura el teclado en castellano suele llamarse KEYBSP.COM).

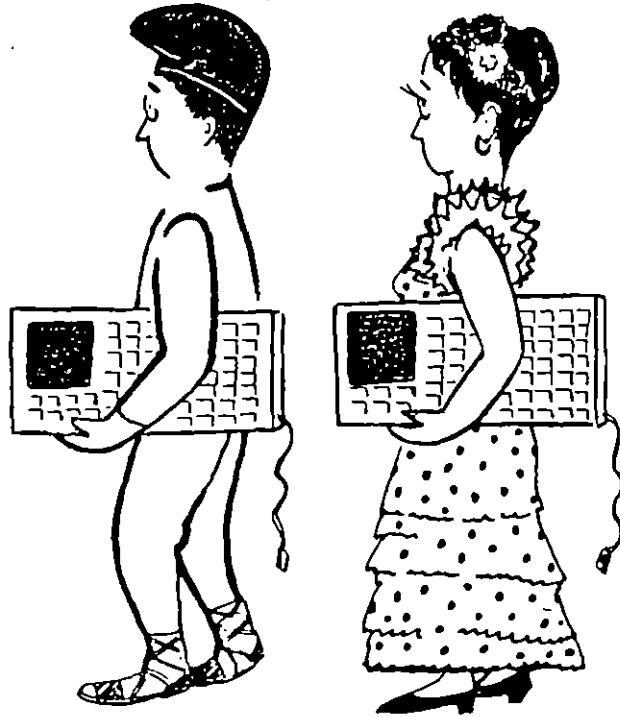


Fig. 3.3 Para escribir correctamente en un idioma hay que disponer de los caracteres necesarios. Existen programas que cambian el significado de las teclas (reconfiguran el teclado).

3.3.2 Las unidades de disco

Son periféricos de entrada/salida secundarios; es una alternativa al teclado, pero de utilidad distinta. Se recurre a la unidad de discos para almacenar, de forma permanente y masiva, programas y datos o recuperarlos para su uso.

El ordenador puede relacionarse, sucesivamente, con una o varias unidades que se identifican por medio de una letra, normalmente de la A a la D. Hay dos tipos de unidades de disco, de disco flexible y de disco fijo o «duro». Estos últimos son discos magnéticos para almacenar información con una capacidad de 30 a 60 veces superior a la de un disco flexible, o disquete, y una velocidad de acceso mayor; se les suele asignar la letra C y D.

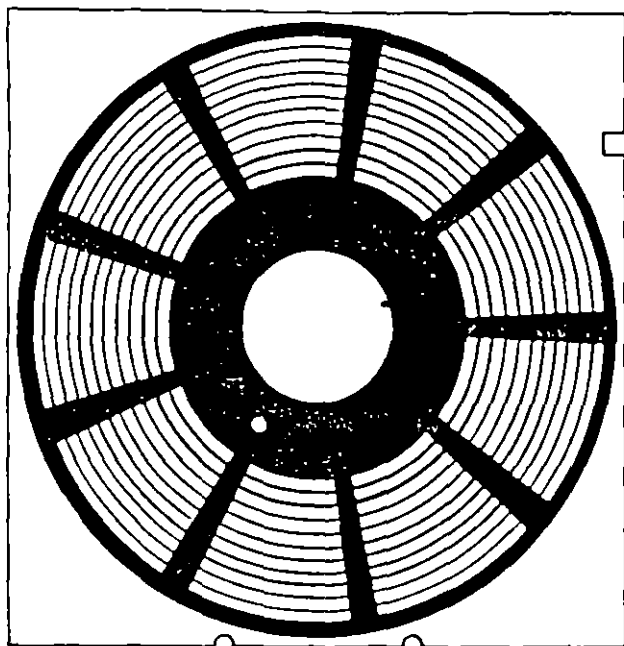


Fig. 3.4 La zona de grabación de un disco magnético debe estar repartida en pistas, distribuidas concéntricamente, y sectores, de disposición radial.

Cuando se necesita leer o escribir en otra unidad basta con teclear el nombre de la nueva seguido de dos puntos (:), por ejemplo A: o B: y el S.O. hace las gestiones oportunas para que, a partir de este momento, todas las órdenes de disco se dirijan a la nueva unidad.

```

┌
└ A>B:<RETURN>
  B>

```

Fig. 3.5

Al disco que se utiliza en un momento determinado se le llama *disco activo*. Su letra distintiva (A, B, C, ...) aparece siempre a la izquierda del cursor del MS-DOS (>).

Vocabulario:

MS-DOS
 Interpretador de comandos
 D.O.S.
 B.I.O.S.
 Teclado
 Unidad de disco
 Arranque «en frío»
 Arranque «en caliente»
 Disco activo.

Ejercicios:

- 3.1. Explicar con palabras propias el significado de cada uno de los términos que aparecen en el vocabulario.
- 3.2. Poner en marcha el ordenador y observar las acciones que van ejecutándose automáticamente.
- 3.3. Exponer tres razones que justifiquen la necesidad del sistema operativo.
- 3.4. Hacer una lista de las ventajas que supone que el S.O. reconfigure el teclado.
- 3.5. Ir a una casa de venta de productos informáticos pedir información sobre dos o tres «marcas» de discos duros; comparar sus cualidades con las de un disco flexible corriente.
- 3.6. Buscar información sobre otros sistemas operativos: nombre, microprocesador sobre el que funciona, para qué tipo de aplicaciones es más idóneo... y hacer una tabla comparativa con el MS-DOS.
- 3.7. Un programa escrito por un programador puede ser muy grande, pero nunca podrá ocupar toda la memoria del ordenador. Explicar por qué razón.

4. Los archivos MS-DOS



4.1 Concepto de archivo

El ordenador siempre guarda la información que contiene en forma de archivos. Conceptualmente una *archivo* es un bloque de datos contiguos —una lista de bits con un indicador de principio, el contenido y un indicador de final— que está localizado en alguna parte de la arquitectura del ordenador (memoria interna o externa) donde pueden almacenarse los datos. Por lo tanto, son archivos tanto los lenguajes de programación como los programas del sistema operativo o los documentos creados con un editor de textos.

La arquitectura de un ordenador es el conjunto de elementos conectados con él.

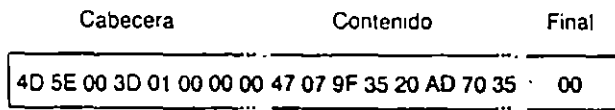


Fig. 4.1 La mayoría de los archivos tienen su contenido precedido de un conjunto de octetos que sirven de cabecera y terminan con alguna marca de final.

Los archivos pueden ser de dos tipos fundamentales:

- a) *ejecutables*, o programas propiamente dichos; lista de instrucciones para el procesador.
- b) *no ejecutables*, o datos; información que puede utilizar un programa, pero nunca puede ejecutarse.

Pueden contener informaciones de todo tipo, textos, números, gráficos, programas, por ejemplo:

— un fichero de facturas	No ejecutable
— un programa editor de textos	Ejecutable
— una lista de órdenes del MS-DOS	Ejecutable
— una lista de resultados de los exámenes	No ejecutable
— un documento escrito con un editor	No ejecutable
— el sistema operativo	Ejecutable
— un dibujo	No ejecutable
— ...	

35

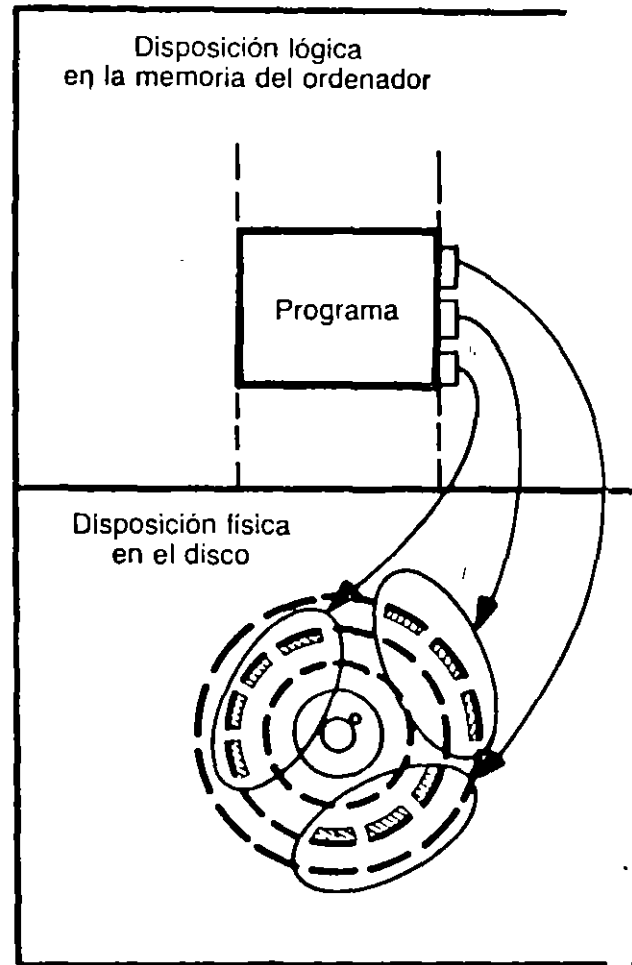


Fig. 4.2 En la memoria interna los archivos tienen siempre una disposición compacta (disposición lógica), pero cuando se almacenan en un dispositivo externo pueden fragmentarse en varios trozos dispersos por el disco (disposición física). Guardar un archivo en la memoria central no es muy diferente de hacerlo en el disco, aunque lo segundo no es tan automático como lo primero porque se utilizan convenciones diferentes.

Un archivo debe tener siempre un formato (orden interno) conocido para que cuando se le accede a través del sistema operativo, puedan recuperarse esos datos sin que pierdan su sentido. En el caso de un archivo con un programa, los datos tienen un agrupamiento lógico, no conocido previamente pero que el ordenador discrimina en el momento que lo va leyendo para ejecutarlo.

Para el ordenador el dispositivo de almacenamiento de información primario es la memoria central. Pero esta memoria es temporal por lo que...

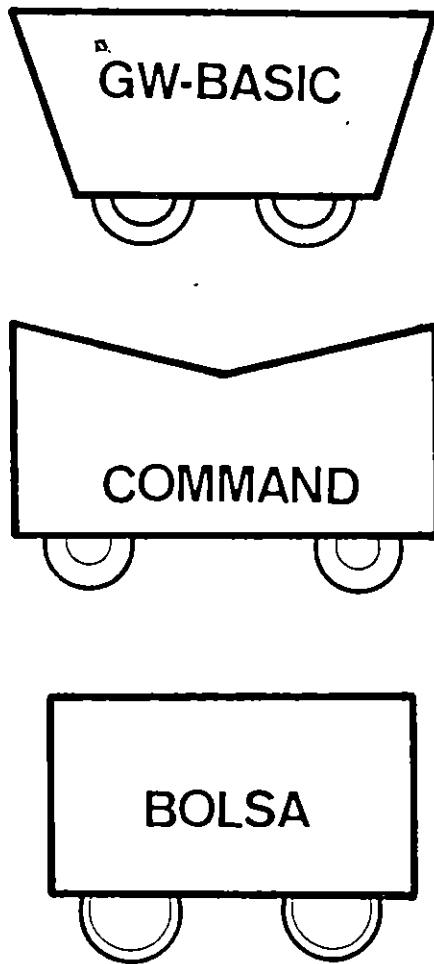


Fig. 4.3 El contenido de los archivos puede ser muy diferente, aunque cuando se guardan en la memoria externa se almacenan de la misma manera.

se requiere un sistema de almacenamiento permanente habrá que recurrir al uso de la memoria externa, o secundaria, generalmente sobre soporte magnético: discos o cintas.

4.2 El nombre de los archivos

Los archivos se pueden manipular como bloques de información y desplazar de un lugar a otro; por ello necesitan un nombre que los identifique y diferencie de los demás. El *nombre* es una palabra que individualiza a un conjunto de datos más o menos amplio de forma inequívoca para futuras consultas u operaciones con él, que generalmente son

las de guardar o recuperar todo o parte de él en la memoria externa.

4.2.1 Las convenciones del nombre

En el sistema operativo MS-DOS, los nombres de los archivos pueden tener tres partes; de ellas la única imprescindible es el nombre propiamente dicho.

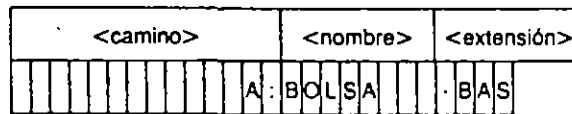


Fig. 4.4 El nombre y la extensión son de longitud fija (máximo de 8 y 3 caracteres respectivamente), pero el «camino» puede ser muy largo.

Aunque el nombre se pueda utilizar solo, generalmente se combina con una extensión de la que se separa por medio de un punto:

A: BOLSA.BAS	extensión .BAS
BOLSA.BAS	extensión .BAS
BOLSA	no tiene extensión

El nombre propiamente dicho puede constar de hasta ocho caracteres, que pueden ser letras, dígitos o algunos otros:

\$, &, #, %, @, —

aunque no es aconsejable utilizar caracteres extraños.

Ejemplos:

PEPITO
BASIC1
FORMA\$15
P
ABCDEFGH
CARTA-25

La *extensión* de los nombres de los archivos puede constar de hasta tres caracteres y sirve como complemento o clave para clasificar rápidamente los programas de un disco. Suele utilizarse para indicar el tipo de archivo o el tipo de datos que contiene. Algunas extensiones tienen también una significación concreta para el sistema operativo:

El sistema operativo MS-DOS no distingue entre mayúsculas y minúsculas.

EXTENSIONES SIGNIFICATIVAS EN EL MS-DOS	
Extensión	Significado
.....COM	programa ejecutable en código máquina
.....BAT	programa ejecutable en forma de lista de órdenes del Sistema Operativo
.....EXE	programa ejecutable en código máquina
.....SYS	archivo del Sistema Operativo

Fig. 4.5 Extensiones más significativas de los nombres de los archivos.

Los archivos que tienen la extensión .COM o .EXE son programas que pueden ejecutarse directamente, escribiendo simplemente su nombre sin necesidad de escribir también la extensión. Están escritos en el lenguaje de la máquina.

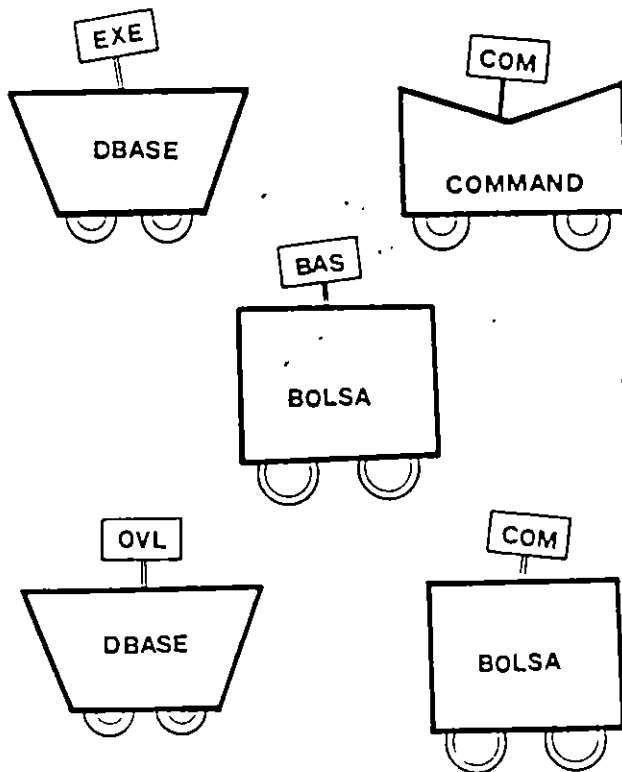


Fig. 4.6 Las extensiones complementan el nombre de los archivos. Algunas se utilizan para identificar los directamente utilizables, .COM y .EXE, otras indican el lenguaje en el que está escrito el programa, .BAS o .PAS, otras que se trata de un fichero de texto, .TXT, etc. 37

Los archivos que llevan la extensión .BAT, también son ejecutables, pero su contenido es una lista de programas. AUTOEXEC.BAT, por ejemplo es el archivo que contiene las órdenes que deben ejecutarse al inicializar el sistema. Escribir AUTOEXEC equivale a escribir una por una las órdenes (KEYBSP, TIME, DATE, etc...) que se hayan prescrito en el archivo AUTOEXEC.BAT.

He aquí un ejemplo de AUTOEXEC.BAT sencillo para cargar el intérprete BASIC:

```

PEM Sistema operativo MS-DOS 3.1
KEYBSP
DATE
GW BASIC

```

Fig. 4.7 Este pequeño «programita» constituido por órdenes del S.O. y un programa en código máquina (GW BASIC) se det guardar en un archivo con la extensión .BAT.

Extensión	Significado
MANUAL.TXT	archivo de texto
BOLSA.BAS	archivo de un programa fuente BASIC
BOLSA.PAS	archivo de un programa fuente Pascal
BOLSA.BAK	copia de un arch. del mismo nombre con otra extensión por si se pierde el primero
WORDP.MSG	archivo con mensajes del programa WORDP
ALA1987.\$\$\$	archivo donde un programa ha guardado provisionalmente datos
FICHEROS.OVL	archivo con parte de un programa ejecutable que se utiliza ocasionalmente

Fig. 4.8 Convenciones que se suelen utilizar en las extensiones; aunque están muy difundidas, en modo alguno son obligatorias.

4.2.2 Los comodines

Para buscar con mayor facilidad un archivo hay dos caracteres que sirven de comodín. En la especificación (nombre y extensión) de un archivo pueden incluirse uno o varios caracteres-comodín para que sustituyan a un carácter (?) o varios (*):

- ? sustituye a cualquier letra en esa posición
- * sustituye a cualquier número de caracteres en esa posición.

Por ejemplo:

CA?A	se refiere tanto a CARA como a CAJA, CALA, CAPA, CAÑA, etc.
C??A	puede referirse a CARA, COLA, CITA, CREA, CIMA, etc.
CASA.???	se refiere a todos los archivos CASA sea cual fuere su extensión.
CASA.*	se refiere a todos los archivos que se llamen CASA con cualquier tipo de extensión, o con ninguna. Para el S.O. esta especificación es idéntica a la anterior (CASA.???).
C*	se refiere a todos los archivos que empiecen por C y que no tengan extensión.
J*.BAS	se refiere a todos los archivos cuyo nombre empiece por Jota y lleven la extensión .BAS.
.	se refiere a todos los archivos del disco activo. Sustituye a cualquier nombre y cualquier extensión.

3.3 Cómo ejecutar un programa

Para ejecutar un programa basta con escribir su nombre a nivel de sistema operativo, pero debe estar escrito en el lenguaje de la máquina, lo que el S.O. reconoce porque tendrá la extensión .COM o .EXE (no hace falta escribirlos).


 A>GWASIC<RETURN>

Fig. 4.9 Es la orden necesaria para entrar en GWBASIC; si no produce el efecto deseado es porque el programa intérprete del GWBASIC no está en el disco que haya en la lectora A. Cuando se escribe una orden, el procedimiento que sigue el ordenador es el siguiente:

Comprueba si existe ese archivo en el disco activo

entonces Lo carga en la memoria interna
Empieza a ejecutarlo
si no Enseñaría un mensaje de error (...).

La orden de la figura anterior hará que se cargue y ejecute el intérprete del GWBASIC.

Si las extensiones no fueran las correctas se produce un mensaje de error.

Si el archivo buscado no se encuentra en la lectora de discos activa será necesario afrontar el mensaje de error y, después de recuperarlo, especificar la lectora donde esté o cambiar el disco.

Cuando termine su ejecución, el S.O. libera la zona de memoria que se hubiera ocupado y enseña otra vez el cursor A> indicando que espera una orden del usuario.

4.4 Ordenes internas y externas

Suele decirse que el MS-DOS tiene órdenes (mandatos) internas y externas. Una *orden interna* es la que forma parte del sistema operativo y puede ejecutarse sin ayuda externa, porque ya está guardada en la memoria del ordenador. Una *orden externa* es la que no está en la memoria central y si se quiere ejecutar hay que leerla primero de un archivo del disco.

En sentido estricto, estas órdenes externas no se diferencian en nada de los demás programas ejecutables, pero como algunas son absolutamente imprescindibles, en cierta manera deben considerarse que pertenecen también al S.O., o por lo menos a los programas del sistema. El caso más

Cuando el MS-DOS enseña el mensaje: A/R/I (abandonar, reintentar o ignorar), la A da por terminada la ejecución de esa orden que produjo el error, la R hace que se vuelva a ejecutar, y la I pide que el MS-DOS ignore el error (y probablemente habrá otros después).

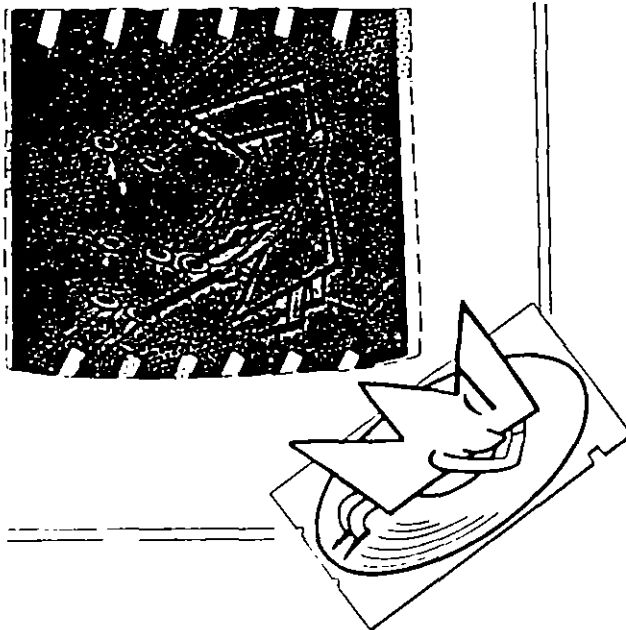


Fig. 4.10 Las órdenes externas permanecen en el disco mientras no se las llame a ejecutar un trabajo. Las internas están en la memoria del ordenador conjuntamente con los programas del S.O.

FORMAT sirve para inicializar un disco nuevo, dibujando —magnéticamente— las pistas y sectores donde guardar los programas. Sin esta operación el S.O. es incapaz de escribir en él.

DISKCOPY hace una copia de todo el contenido de un disco con otro.

representativo es la orden FORMAT, sin la cual no sería posible utilizar los discos. Otras se incluyen por la frecuencia de su uso: DISKCOPY.

Las órdenes externas suelen guardarse en un disco, normalmente el del sistema, para que no ocupen espacio en la memoria del ordenador, ya que no se usan tanto como las órdenes internas o residentes.

4.5 La orden y los parámetros

Desde cierto punto de vista, el conjunto de órdenes del sistema operativo casi constituye un minilenguaje de programación, con variables y órdenes de control de flujo elementales y, como ellos, también requiere que se respeten ciertas normas sintácticas.

Una orden consta de una palabra y un número variable de parámetros, separados de ella por separadores (espacio en blanco, coma), por ejemplo:

A>DIR no tiene parámetros
A>CD SISTEMA tiene un parámetro (SISTEMA)

Se llama *parámetro* al dato que modifica o complementa la ejecución de una orden.

El ordenador supone siempre que está trabajando con un sólo disco, que es aquel cuya letra aparece en el cursor. Siempre que efectue una lectura o una escritura lo hará allí. Para cambiar hay que especificárselo en el nombre del fichero o por medio de un parámetro.

```

A> DIR <RETURN>

Volumen en unidad A etiquetado SISTEMA
Directorio de A:\

COMMAND COM 24492 1-01-80 12:00p
BOLSA BAS 15340 1-05-87 2:30p
      2 Archivos(s) 242723 octetos libres

A> DIR B:<RETURN>

Volumen en unidad A etiquetado EJEMPLO
Directorio de B:\

KEYBSP COM 3187 12-08-87 12:10p
GBASIC EXE 37 1-01-85 12:00p
      2 Archivos(s) 152224 octetos libres

A> B.<RETURN>
B>
    
```

Fig. 4.11 Para cambiar de disco activo, se escribe después del cursor la letra de la unidad seguida de «:» y se pulsa <RETURN>.

Vocabulario:

- Archivo
- Archivo ejecutable
- Archivo no ejecutable
- Nombre de archivo
- Extensión del nombre
- Comodin
- Orden interna
- Orden externa
- Parámetro.

Ejercicios:

- 4.1. Explicar con palabras propias el significado de uno de los términos que aparecen en el vocabulario.
- 4.2. Investigar en el manual del MS-DOS el significado de la orden FORMAT.
- 4.3. Tomar un disco vacío, o de contenido desconocido, y formatearlo.

Sugerencias de ampliación

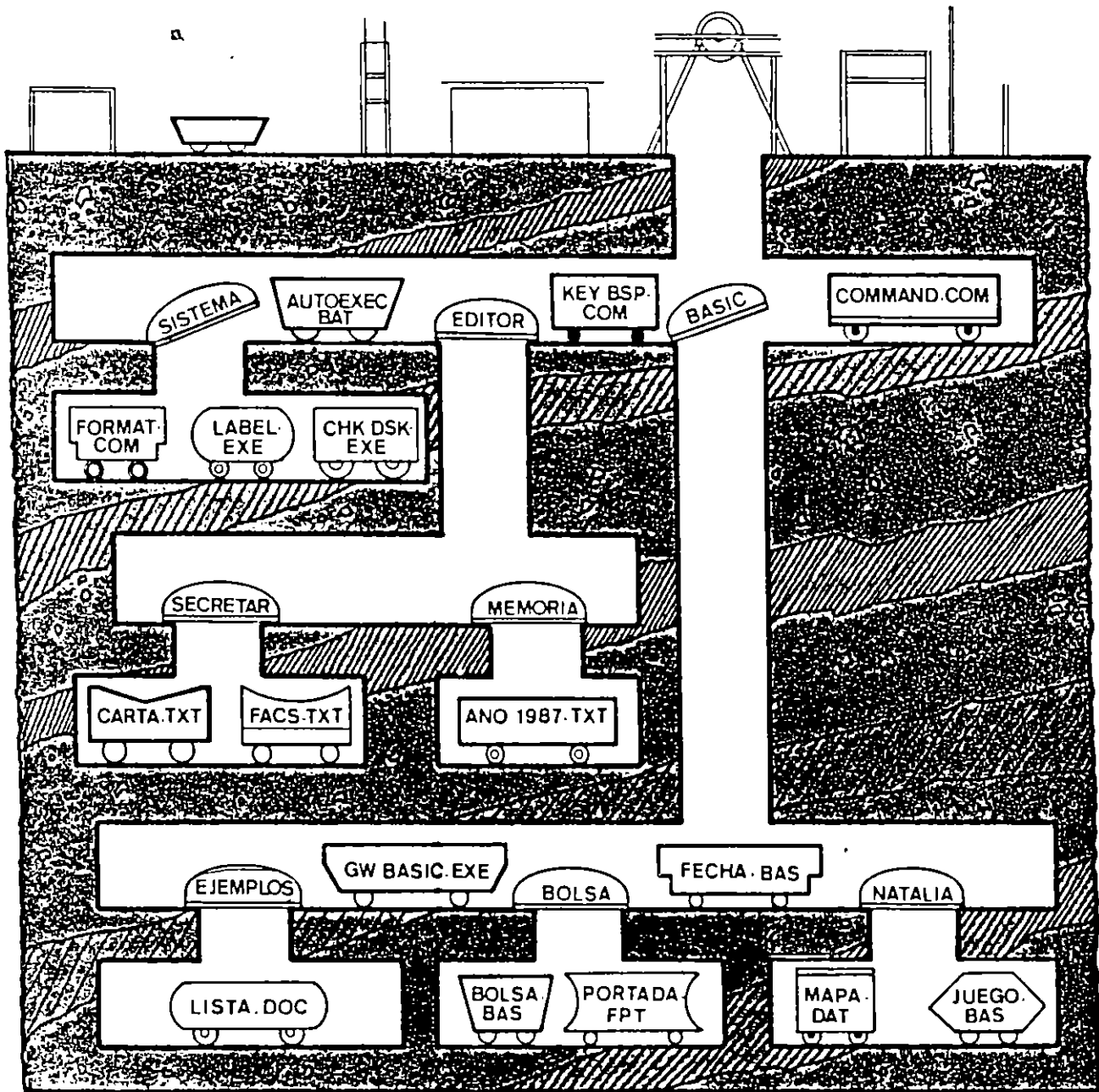
Pistas y sectores

Los discos flexibles son el método de almacenamiento de información más corriente de los ordenadores. Su ventaja principal sobre los discos rígidos es que permiten el acceso aleatorio a la información sin necesidad de procesarla.

Para que el ordenador pueda encontrar la información que busca, tiene que leer el disco con cierto orden. Con este fin, el disco está distribuido en pistas circulares consecutivamente, en sectores separados por espacios (trozos de pista sin información).

En el sistema MS-DOS suelen utilizarse discos flexibles de densidad sencilla y doble cara. MS-DOS los formatea en 40 pistas y le proporciona una cabida aproximada de 1 MB.

5. Las órdenes y los directorios



5.1 Las principales órdenes del MS-DOS

De las 39 órdenes fundamentales del MS-DOS, las más comúnmente utilizadas son: COPY, DIR, ERASE, FORMAT, RENAME, SYS, TYPE. Veamos una descripción de las más importantes.

DIR

Sirve para listar el contenido de un directorio del disco, es decir, los archivos que contiene.

El listado del directorio proporciona también información acerca del tamaño de los archivos y de las fechas en que se crearon o modificaron por última vez.

```

A>DIR<RETURN>

Volumen en unidad A etiquetado PRACTICA
Directorio de A \

COMMAND COM      24492  12-08-87  12.10p
KEYBSP  COM      3187  12-08-87  12.10p
AUTOEXEC BAT      37  12-08-87  12.30p
SISTEMA    <DIR>    12-08-87  16.10p
EDITOR    <DIR>    12-08-87  16.30p
BASIC     <DIR>    12-08-87  16.40p
          3 Archivo(s)      52224 octetos libres

A>

```

Fig. 5.1 Ejemplo del directorio de un disco.

Si sólo se precisa información de uno de los archivos, se escribe DIR y a continuación el nombre del archivo.

```

A>DIR COLOR BAS<RETURN>

Volumen en unidad A etiquetado PRACTICA
Directorio de A \BASIC\EJEMPLOS

COLOR  BAS      6110  12-08-87  12.30p
          1 Archivo(s)      52224 octetos libres

A>

```

Fig. 5.2

TYPE

Muestra el contenido del archivo que se especifique en el periférico de salida activado en ese momento: pantalla o impresora.

Si es un programa en lenguaje máquina su aparición será caótica, por lo que suele utilizarse sólo para archivos que contengan datos o textos. Por ejemplo:

```

A>TYPE AUTOEXEC.BAT<RETURN>
REM Sistema operativo MS-DOS 3.1
KEYBSP
DATE
GW BASIC

A>

```

Fig. 5.3 TYPE visualiza el contenido de un archivo, por pantalla o impresora.

ERASE o DEL

Borra el archivo que se especifique del disco, por lo que hay que usarlo con cuidado. Utiliza como parámetro el nombre del archivo que se quiere borrar.

Como en DIR, permite usar comodines en el nombre del archivo, por ejemplo:

```

A>DEL *.BAS<RETURN>

A>

```

Fig. 5.4 DEL borra uno o varios archivos del disco.

y después ya no tiene remedio.

RENAME o REN

Cambia el nombre del archivo que se especifique por otro que se indique. Necesita dos parámetros: el nombre viejo y el nuevo; también admite los comodines.

```

A>RENAME *.BAS *.GBA<RETURN>

A>

```

Fig. 5.5 El primer nombre es el viejo y el segundo el nuevo.

COPY

Es una orden que puede realizar tareas muy diversas, que, en lo esencial, consisten en leer datos de un lugar y colocarlos en otro.

El uso más corriente es para copiar archivos entre discos. Resulta muy práctico el uso de comodines para copiar varios archivos a la vez.

Sus parámetros fundamentales son:

- lo que se quiere copiar
- dónde colocarlo.

```

A>COPY BOLSA BAS BOLSA.TXT<RETURN>
BOLSA BAS
          1 Archivo(s) copiado(s)

A>

```

Fig. 5.6

En este caso ha copiado el archivo BOLSA.BAS y lo ha dejado en el mismo directorio, pero con un nombre diferente: BOLSA.TXT. En este otro caso hará lo mismo, pero con todos los archivos cuya extensión sea .BAS:

```
A>COPY *.BAS *.TXT<RETURN>
COLOR BAS
PRUEBA BAS
CAMBIO BAS
LISTADOR BAS
      4 Archivo(s) copiado(s)

A>
```

Fig. 5.7

Ahora ha copiado en el disco B: todos los archivos que había en el disco A:

```
A>COPY A:.* B:<RETURN>
COLOR BAS
PRUEBA BAS
CAMBIO BAS
LISTADOR BAS
      4 Archivo(s) copiado(s)

A>
```

Fig. 5.8

Una opción importante es copiar todos los archivos de un disco a otro:

5.2 Concepto de directorio

En principio, un *directorio* es una lista de los archivos que hay en un disco. Pero en el caso de los discos duros esta lista puede ser enorme. Para facilitar las búsquedas entre la masa de archivos existentes y evitar errores se creó el sistema de directorios y subdirectorios, organizándolos en forma de árbol.

Los directorios constituyen unidades separadas y desde uno de ellos los archivos de otros directorios no son accesibles directamente. En un di-

Estructura	Nombre del directorio
Directorio RAIZ	
Directorio SISTEMA	\ SISTEMA
Directorio EDITOR	\ EDITOR
Directorio SECRETAR	\ EDITOR \ SECRETAR
Directorio MEMORIA	\ EDITOR \ MEMORIA
Directorio BASIC	\ BASIC
Directorio EJEMPLOS	\ BASIC \ EJEMPLOS
Directorio BOLSA	\ BASIC \ BOLSA
Directorio NATALIA	\ BASIC \ NATALIA

Fig. 5.9 Ejemplo de la estructura de los directorios de un disco.

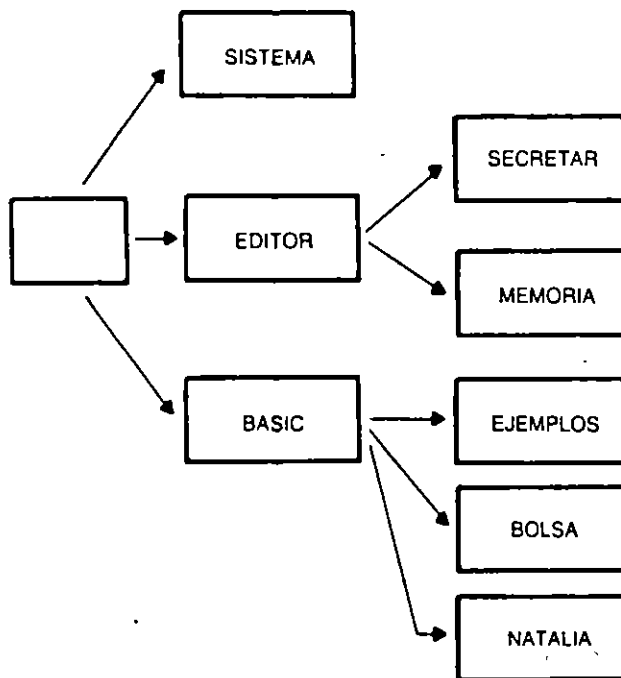


Fig. 5.10 Árbol de subdirectorios.

rectorio sólo se puede leer, ejecutar, modificar, etc. los archivos que están incluidos en él.

Los directorios también pueden contener otras informaciones que no sean nombres de archivos: los nombres de otros directorios. Este hecho permite organizar jerárquicamente los archivos de un disco, con un directorio *raíz* (root), y unas ramas dependientes del mismo, que determinan el *camino* (path) a seguir para encontrar un archivo.


```

A> DIR

Volumen en unidad A etiquetado PRACTICA
Directorio de A:\
Archivos
COMMAND  COM      24492  12-08-87  12:10p
KEYBSP   COM      2187   12-08-87  12:10p
DIRECTORIOS
AUTOEXEC.BAT  37     12-08-87  12:30p
SYSTEMA    <DIR>    12-08-87  16:10p
EDITOR     <DIR>    12-08-87  16:30p
BASIC      <DIR>    12-08-87  16:40p
5 Archivo(s) 52224 octetos libres

```

Fig. 5.11

Sin embargo, en el directorio A:\ no están todos los archivos del disco; por ejemplo, en A:\SISTEMA hay unos cuantos más de utilidad general. A:\ SISTEMA desglosado es:

- A es la unidad de disco donde se encuentra el disco con ese archivo
- : detrás de una letra indica que ésta es la identificación de una unidad de disco
- \ representa al directorio raíz
- SISTEMA es el nombre de un directorio

Pero para acceder a ellos hay que cambiar de directorio. descender un nivel, empleando la orden CD con el argumento del nombre del directorio a donde queremos ir:

```

A>CD SISTEMA <RETURN>

```

Fig. 5.12

Aparentemente no ha cambiado nada, pero ya hemos bajado, como se puede comprobar con la misma CD; si se usa CD sin parámetros el S.O. muestra cuál es el camino activado en ese momento:

```

A> CD <RETURN>
A \SISTEMA

```

Fig. 5.13

y si ahora se pide el listado de archivos se debería ver:

```

A>DIR<RETURN>

Volumen en unidad A etiquetado PRACTICA
Directorio de A \SISTEMA

        <DIR>          1-01-80  12:11a
        <DIR>          1-01-80  12:11a
FORMAT  COM      10123  12-08-87  12:10p
LABEL   EXE      2882   12-08-87  12:10p
CHKDSK  EXE      9840   12-08-87  12:30p
5 Archivo(s) 52224 octetos libres

```

Fig. 5.14

Obsérvese que ese directorio no se parece en nada al que produjo la misma orden DIR cuando estaba en el directorio A:\. Además, si ahora quisiéramos ejecutar algún archivo del directorio A:\, el sistema operativo no lo encontraría, porque no está en el directorio activo, que en este momento es A:\SISTEMA:

```

A>AUTOEXEC<RETURN>
Mandato o nombre de archivo erróneo
A>

```

Fig. 5.15

En este caso no ha encontrado el archivo AUTOEXEC.BAT en el directorio activo, porque e.

en otro. Eso quiere decir que cuando se está situado en un directorio si se guarda un archivo en el disco se quedará en ese directorio y sólo podrán leerse los archivos que están en él.

Para ejecutarlo deberíamos volver al directorio A:\, de esta forma:

```

┌───┐
└───┘ A>CD \ <RETURN>

A>CD
A \

A>.
```

Fig. 5.16

Como se ha visto, a veces no es suficiente con conocer el nombre del archivo y el de su extensión, ya que si el disco está estructurado en directorios es necesario conocer también en cuál está, y qué camino hay que recorrer hasta llegar al directorio deseado.

La forma de especificar el camino de acceso al archivo GWBASIC.EXE, situado en el subdirectorio BASIC, por ejemplo, sería

\BASIC\GWBASIC.EXE

donde el primer signo \ se refiere al directorio raíz, que en este ejemplo es el directorio «padre» de BASIC, es decir, su inmediato predecesor jerárquico. Siguiendo con el mismo ejemplo, si BASIC tuviese a su vez un directorio llamado BOLSA, el acceso al mismo vendría determinado por el camino:

\BASIC\BOLSA

y BASIC sería el padre de BOLSA.

Conociendo el camino completo desde el directorio raíz, podemos colocarnos directamente en el directorio deseado con una sola orden. Por ejemplo:

```

ir a BASIC con      CD\BASIC
ir a BOLSA con     CD\BASIC\BOLSA
ir a MEMORIA con  CD\EDITOR\MEMORIA
ir al raíz con    CD\
```

44

Obsérvese que en los directorios que no son el raíz aparecen los símbolos . y .. al principio. Con el (.) se indica el nombre del directorio activo y con (..) el del directorio «padre». Por ejemplo, en la lista del directorio A:\BASIC aparecen unos cuantos nombres con la indicación <DIR> (EJEMPLOS, BOLSA y NATALIA). En este caso, BASIC es el padre de EJEMPLOS, BOLSA y NATALIA. Y BASIC, a su vez, es hijo del directorio raíz.

La forma de activar, o subir, al directorio «padre» es:

```

┌───┐
└───┘ A>CD .. <RETURN>

A>.
```

Fig. 5.17

Con esta orden (CD ..) se puede ir subiendo niveles por el árbol con facilidad. Para ir todavía más rápido el directorio puede activarse directamente con:

```

┌───┐
└───┘ A>CD \<RETURN>

A>.
```

Fig. 5.18

desde cualquier directorio, por escondido que esté.

Vocabulario:

Directorio
Directorio raíz
Camino

Ejercicios:

- 5 1. Explicar con palabras propias el significado de cada uno de los términos que aparecen en el vocabulario.
- 5 2. Copiar en un disco de pruebas formateado varios archivos de otro disco:

- a) Probando de uno en uno.
- b) A la vez todos los que tengan la misma extensión.
- c) Todos los que empiecen por la misma letra y tengan la misma extensión.

- d) Todos los que empiecen por la misma letra con independencia de la extensión.
- e) Todo el disco a la vez.

Tomar la precaución de borrar todos los archivos cada vez que se haga un apartado. Usar diversos sistemas para eliminar los archivos.

Pedir el directorio después de cada orden de copia y observar el resultado.

5.3. Inspeccionar la apariencia de un archivo de texto (extensión .TXT) y también de uno de datos.

5.4. Siguiendo las instrucciones del manual del MS-DOS hacer pruebas de creación de directorios y subdirectorios. Llenarlos de contenido copiando archivos de otro disco repitiendo las pruebas propuestas en el ejercicio 5.2.

5.5. Tomar un disco nuevo y después de formatearlo crear la estructura de directorios que aparece en el apartado 2 del capítulo.

. . . .

Sugerencias de ampliación

El redireccionamiento de la E/S: la impresora

La forma más fácil de hacer que la salida de datos del ordenador se produzca por la impresora es pulsando CTRL-P.

Desde el momento en que se haga, todo lo que aparezca por la pantalla aparecerá también por la impresora (si está conectada, tiene papel, etc. ¡claro!) hasta que vuelva a pulsarse ese carácter de control.

Lo usual es añadir ese carácter al final de una línea del MS-DOS:

```
A>DIR<CTRL-P><RETURN>
.....
.....
A>
```

Fig. 5.19 Desde ahora lo que aparezca por pantalla también saldrá por impresora.

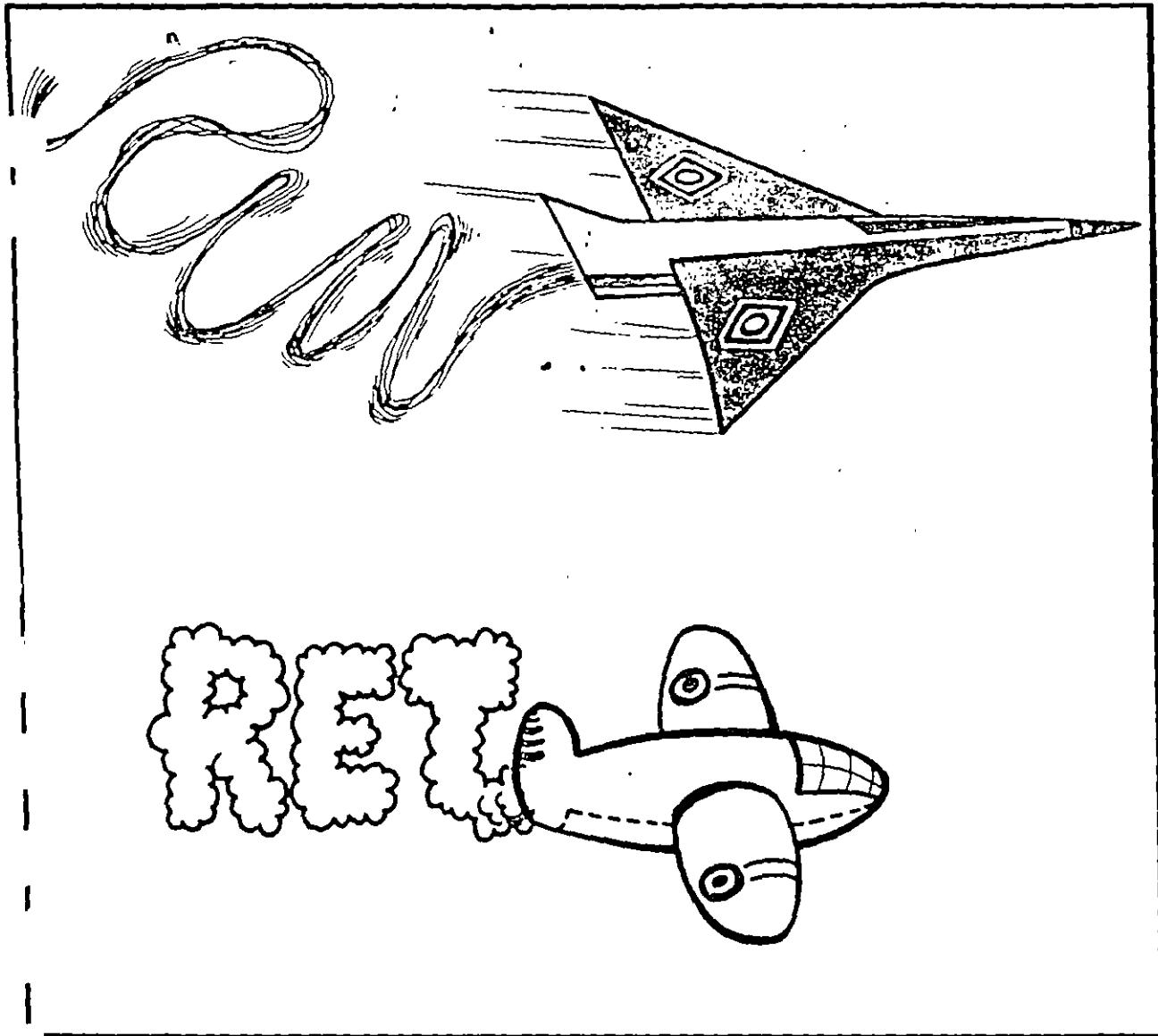
y entonces repetir esa orden para desconectar la impresora:

```
A><CTRL-P><RETURN>
.....
A>
```

Fig. 5.20 Desde ahora la salida se efectuará solamente por impresora.

- BREAK
- CHDIR (CD)
- CTTY
- ECHO
- EXIT
- FOR
- GOTO
- IF
- MKDIR (MD)
- PAUSE
- REN
- RMDIR (RD)
- SET
- SHIFT
- VER
- VERIFY
- VOL

Otras órdenes importantes del MS-DOS.



1 ¿Qué es un lenguaje informático?

La descripción de las tareas que debe efectuar el ordenador sólo se puede realizar en un lenguaje informático de programación.

Un *lenguaje informático*, como cualquier otro lenguaje, consta de un conjunto de reglas (sintaxis o

reglas sintácticas), símbolos y palabras reservadas (vocabulario) agrupadas en forma de instrucciones comprensibles para el ordenador.

Las reglas sintácticas, que son propias de cada lenguaje, establecen las condiciones para escribir las sentencias correctamente. Los errores sintácticos hacen que el programa no sea inteligible para el ordenador.

Lenguaje
reglas + palabras

PALABRAS:
Palabras reservadas
Etiquetas
Operadores
Separadores

EJEMPLO DE REGLA:
Las palabras reservadas
van precedidas y seguidas
de un separador.

```

10 PRINT "HOLA"
20 PRINT NUMERO% * DINERO%
    
```

Fig. 6.1

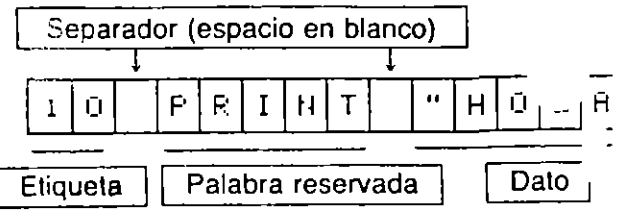


Fig. 6.2

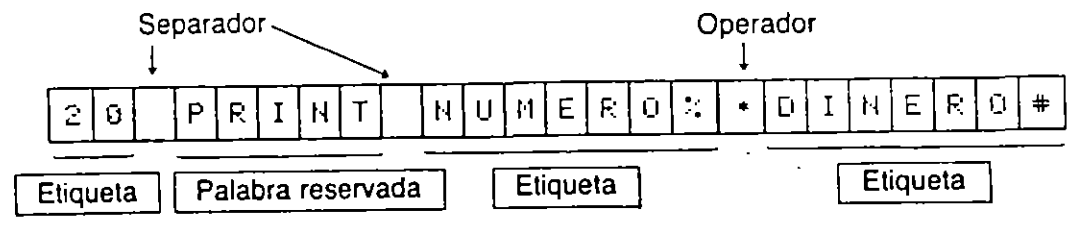


Fig. 6.3

```

PRINT33
SYNTAX ERROR
OK
PRINT 33
33
OK
    
```

Fig. 6.4 Cuando no hay separador puede surgir el error.

En un *lenguaje de bajo nivel* cada instrucción responde a una acción ejecutable por el computador, mientras que en los *lenguajes de alto nivel* una instrucción suele corresponder a varias acciones (incluso centenares!).

Lenguajes de bajo nivel	Lenguajes de alto nivel
Lenguaje máquina ensamblador	BASIC Pascal Logo C Prolog ...

Fig. 6.5

Cada lenguaje informático utiliza un conjunto de símbolos y palabras que constituyen su vocabulario. El de los lenguajes modernos es más limitado que el del BASIC, pero permiten ampliarlo con órdenes nuevas que funcionan como palabras reservadas definidas por el usuario.

Una regla de uso muy extendido entre los lenguajes, y muy importante en GWBASIC, es que las palabras reservadas han de ir separadas de las demás por un separador. En el ejemplo anterior, PRINT33 da un error de sintaxis porque PRINT, palabra reservada, debe ir separada de 33 por un espacio en blanco u otro separador para que el ordenador la pueda distinguir de lo que le sigue: PRINT 33. Esta obligación de separar las palabras por separadores es una *regla sintáctica*.

6.2 Tipos de lenguajes

Atendiendo al número de instrucciones necesarias para realizar una tarea específica podemos clasificar los lenguajes informáticos en dos grandes bloques: lenguajes de bajo nivel y lenguajes de alto nivel.

6.3 Lenguajes de bajo nivel

Los lenguajes informáticos más próximos al hardware reciben el nombre de lenguajes de bajo nivel. Se les ha llamado también de «primer nivel», puesto que fueron los que originalmente se utilizaron para programar los primeros computadores.

Cada microprocesador tiene su propio lenguaje con un conjunto de instrucciones diferente. Cada uno similar, por lo que hay tantos lenguajes de bajo nivel como modelos de microprocesador.

En realidad los ordenadores digitales solamente entienden un único lenguaje, cuyas instrucciones están escritas en código binario, es decir, están formadas por secuencias de ceros y unos. Almacenan toda la información en memoria en forma binaria, de modo que tanto las instrucciones como los datos se hallan codificados.

Los lenguajes de bajo nivel son los que están más cercanos a la máquina (se parecen más al funcionamiento real del ordenador) en el sentido de que cada acción debe especificarse mediante una instrucción.

6.3.1 El lenguaje de la máquina

Un programa escrito en lenguaje máquina, o más propiamente en *código máquina*, es un conjunto de instrucciones codificadas en binario que son capaces de relacionarse directamente con los registros y circuitería del microprocesador del ordenador y que, por lo tanto, resulta directamente ejecutable por éste, sin necesidad de otros programas intermediarios.

Para los programadores es posible escribir programas directamente en lenguaje máquina, pero las instrucciones son difíciles de recordar y los programas resultan largos y laboriosos de escribir y también de corregir y depurar.

Dada la dificultad de programar en código máquina se han diseñado otros lenguajes que, aun-

que no disminuyen el número de instrucciones, resultan más próximos al usuario puesto que las sustituyen por códigos nemotécnicos. Estos son los llamados lenguajes ensambladores.

6.3.2 El lenguaje ensamblador

Es otro lenguaje de programación de bajo nivel, pero simbólico porque las instrucciones se construyen usando códigos de tipo nemotécnico, lo cual facilita la escritura y depuración de los programas, pero no los acorta puesto que para cada acción se necesita una instrucción.

Para ejecutar un programa escrito en lenguaje ensamblador es preciso traducirlo antes a código máquina. De esta tarea se encarga el *programa ensamblador* (assembler) que va traduciendo línea a línea a la vez que comprueba la existencia de errores. Si localiza alguno da un mensaje de error.

Aparece entonces el concepto de *máquina virtual*: si a una máquina inicial la dotamos de un programa complementario, de modo que vista exteriormente se comporte como si admitiera otro lenguaje de programación diferente del lenguaje máquina, estaremos en presencia de una máquina virtual que trabaja en este nuevo lenguaje.

A los ensambladores se les llama de bajo nivel porque, tal como ya se ha descrito, una instrucción en este lenguaje se convierte en una instrucción en lenguaje máquina —relación uno a uno—. En

BASIC	Lenguaje Ensamblador			
100 TOT = 0	NUM	DTA	+0	Variable NUM.
	TOT	DTA	+0	Variable TOT.
105 INPUT NUM	BGN	LOC	+0	Limpiar Acc.
		STO	TOT	Guardar 0 en total
	RPT	IRQ		Leer pregunta
		INP		Leer número
110 IF NUM = 0 THEN 125		STO	NUM	Guardar número
		LOA	NUM	Cargar núm al Acc.
		BZE	OUT	Si 0 saltar a OUT
115 TOT = TOT + NUM		LOA	TOT	Cargar total
		ADD	NUM	Sumar número al total
		STO	TOT	Guardar total actual
120 GOTO 105		BRN	RPT	Saltar a RPT
125 PRINT TOT	OUT	LOA	TOT	Cargar total al Acc.
		OUP		Escribir total
		ORQ		Escribir pregunta
130 END		END		Fin del programa.

Fig. 6.6 El mismo programa escrito en dos lenguajes. BASIC y ensamblador, puestos en correspondencia.

cuanto a potencia no aporta nada respecto al código máquina. Suelen reconocerse como lenguajes pertenecientes a la segunda generación.

Resultan de gran aplicación para el desarrollo de programas del sistema (sistemas operativos, programas de utilidades, rutinas,...).

6.4 Necesidad de los lenguajes de alto nivel

Se comprende que usar el lenguaje ensamblador es más cómodo que utilizar el código máquina, pero en realidad se trata de las mismas instruccio-

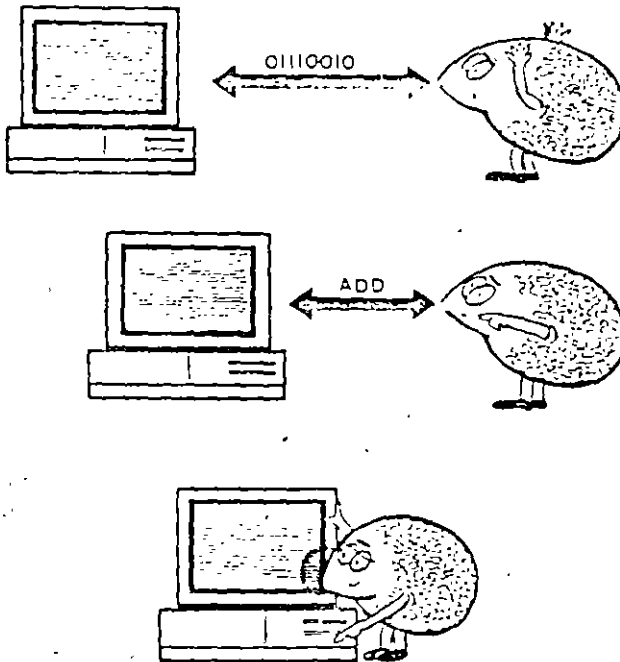


Fig. 6.7 Los lenguajes de alto nivel acercan la máquina al programador.

nes y por lo tanto la escritura de un programa sigue siendo igual de laboriosa.

A lo largo de las tres últimas décadas y a medida que los ordenadores han aumentado su potencia y las aplicaciones su complejidad, han ido apareciendo otros lenguajes de programación más sencillos, flexibles y cómodos de utilizar. A diferencia de los anteriores, estos nuevos lenguajes están más cercanos al usuario en el sentido de que las instrucciones utilizan vocabulario más nemotécnico y cada instrucción se convierte en varias instrucciones en lenguaje máquina, por lo que los programas resultan más fáciles y cortos de redactar y corregir. A este tipo de lenguajes se les llama *lenguajes de alto nivel*.

Al contrario que los de bajo nivel, estos lenguajes no están tan íntimamente relacionados con la máquina por lo que resultan mucho más independientes de ella. Esta independencia de la máquina permite escribir los programas sin necesidad de tener un conocimiento profundo de su funcionamiento físico.

6.5 Lenguajes de alto nivel

Los lenguajes de alto nivel tienen dos características fundamentales que les diferencian de los de bajo nivel:

- Independencia de la máquina.
- Orientados a las aplicaciones.

La independencia de la máquina significa que el mismo lenguaje puede funcionar (al menos en teoría) en distintos ordenadores, por lo que tanto el lenguaje como los programas escritos con él serán *transportables* de un ordenador a otro. En la práctica, esta característica resulta limitada por la

Ensamblador	C. máquina	Comentarios
MOV DX,0040	BA 40 00	Poner en el registro DX el número 40 en hexadecimal
MOV DS,DX	8E DA	Pasar el contenido de DX a DS
MOV AL,[0017]	A0 17 00	Colocar en AL el contenido de la dirección de memoria 17 hexadecimal
OR AL,40	OC 40	Aplicar a lo que haya en AL la operación OR con el número 40 hexadecimal
MOV [0017],AL	A2 17 00	Poner lo que haya en AL en la dirección 17
INT 20	CD 20	Terminar el programa

Fig. 6.8 Un programa en código máquina para activar las teclas de mayúsculas fijas en los compatibles PC.

diversidad de versiones y dialectos que se construyen para cada lenguaje.

Que sean *orientados a las aplicaciones* quiere decir que cuando se diseña uno de estos lenguajes se tiene en cuenta que su finalidad es la resolución de problemas, prescindiendo de la arquitectura del ordenador. De esta forma los lenguajes de alto nivel acercan la máquina al programador que, para resolver un problema al estilo informático, se verá más independizado de las características físicas del ordenador y podrá concentrar su atención en el método de resolución del problema y usar unos formalismos más parecidos a los del lenguaje corriente.

6.6 Comparación entre los lenguajes de alto nivel y bajo nivel

Tanto los lenguajes de bajo nivel como de alto nivel son competentes en su campo de aplicabilidad, pero los segundos ofrecen al programador claras ventajas sobre los primeros.

- Mayor especialización de los lenguajes: los lenguajes orientados a los problemas resultan más eficaces para la resolución de aquel tipo de problemas, a costa de una menor eficiencia para otros. Se trata de recurrir al lenguaje más adecuado para resolver un problema de un tipo concreto.
- Mayor transportabilidad: gracias a que los lenguajes de alto nivel son en gran proporción in-

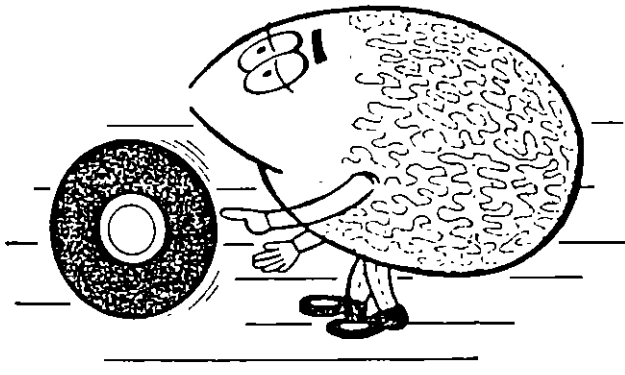


Fig. 6.9 Los lenguajes de alto nivel hacen que los programas sean más transportables entre ordenadores. Un programa escrito en un lenguaje de alto nivel puede funcionar en distintos ordenadores con un número mínimo de modificaciones

dependientes de la máquina, un mismo lenguaje, y los programas escritos con él, podrán funcionar en diversos ordenadores.

- Facilidad de programación: dado que cada sentencia puede sustituir a un gran número de instrucciones es evidente que la escritura de un programa será más fácil y más económica en tiempo de programación.

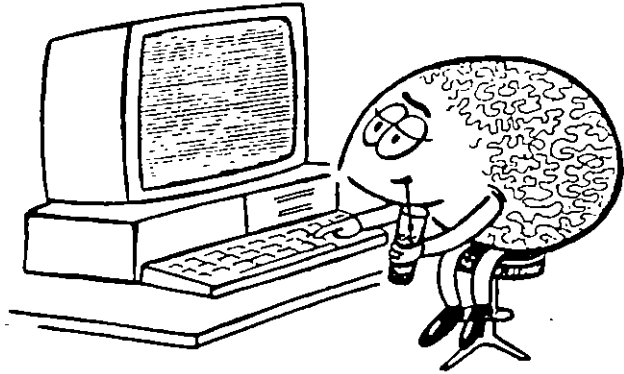


Fig. 6.10 El programador se encuentra más cómodo programando en un lenguaje de alto nivel porque no tiene que ocuparse de todos los detalles.

- Mayor legibilidad de los programas: puesto que el lenguaje está orientado al problema, los programas visualizan la resolución paso a paso a la vez que la localización de errores resulta más evidente.
- Facilidad de modificación: los programas escritos en un lenguaje de alto nivel son más flexi-

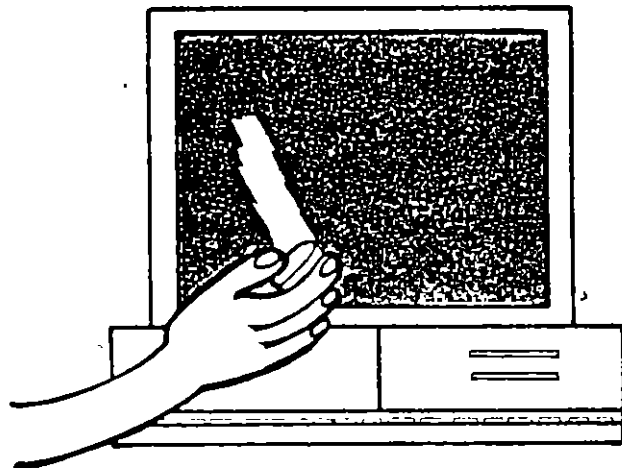


Fig. 6.11 La modificación de un programa escrito en un lenguaje de alto nivel requiere menos cambios en todo el programa

bles y la modificación de alguna de sus instrucciones no requiere la reescritura de todo el programa.

- Facilidad de aprendizaje: debido a que el vocabulario es cercano al del usuario y a que las sentencias guardan un cierto parecido con las frases del idioma escrito.

A pesar de estas ventajas, en algunas ocasiones es conveniente el uso de lenguajes de bajo nivel para programar cierto tipo de aplicaciones, ya que:

- con los lenguajes de bajo nivel se obtienen programas cuyo código objeto —programa ejecutable— es más corto.
- hay mayor interacción con el sistema operativo optimizando su utilidad.

6.7 ¿Por qué hay tantos lenguajes diferentes?

Hay muchos lenguajes de programación distintos, y las razones que puede explicar tal diversidad son:

- la adquisición de mayor potencia —rapidez— en un área de trabajo específica,
- el modo de representación interna de las variables en la memoria del ordenador,
- un intento de adaptación a las peculiaridades del ordenador con el que tienen que funcionar.

Entre los lenguajes de alto nivel hay algunos que podríamos llamar de *propósito específico* puesto que están mejor preparados para adaptarse a una aplicación concreta y específica, aun a costa de su limitación en otras actividades: PROLOG, PEPAMACA,... y otros de propósito general: PL1; BASIC,... cuyo campo de aplicabilidad es amplio.

Sin embargo, no todos los de propósito general se adaptan bien a las necesidades de todos los campos. Algunos se adaptan mejor para aplicaciones de gestión y administración (el COBOL); otros son más adecuados para usarlos en cálculos científicos (el FORTRAN); otros tienen mayor aplicabilidad en el campo de la enseñanza (Pascal y LOGO), etc.

Además, cada lenguaje de alto nivel suele tener sus dialectos, versiones ligeramente modificadas

del mismo, que funcionan en distintos ordenadores, lo que hace que la transportabilidad antes toda sea más teórica que práctica.

Vocabulario:

Lenguaje informático
 Lenguaje de bajo nivel
 Lenguaje de alto nivel
 Código máquina
 Lenguaje máquina
 Lenguaje ensamblador
 Programa ensamblador
 Transportabilidad
 Lenguaje orientado a los problemas
 Lenguaje orientado a la máquina

Ejercicios:

- 6.1. Explicar con palabras propias el significado de uno de los términos que aparecen en el vocabulario.
- 6.2. ¿Cuál es la diferencia fundamental entre los lenguajes de alto nivel y bajo nivel?
- 6.3. ¿Qué aportan a la programación los lenguajes de bajo nivel?
- 6.4. ¿Por qué algunos programas se escriben en lenguajes de bajo nivel?
- 6.5. ¿Por qué cada procesador tiene un código máquina distinto?
- 6.6. Dar un par de razones que justifiquen la variedad de los lenguajes de programación existente.
- 6.7. Consultar en una enciclopedia informática el nombre de lenguajes de:

alto nivel
 bajo nivel
 propósito específico
 propósito general

- 6.8. Buscar analogías y diferencias entre el código máquina y el lenguaje ensamblador.
- 6.9. ¿Por qué es tan difícil programar en código máquina?
- 6.10. Buscar información que permita decidir cuáles serían los lenguajes de programación más apropiados para realizar:

- aplicaciones para gestión de oficinas
- complejos cálculos científicos
- un sistema experto en medicina
- programas gráficos para juegos
- un simulador de vuelo de un F-18
- manipulación de bases de datos
- traducciones entre dos idiomas
- el control de un robot industrial.

Sugerencias de ampliación

Clasificación de los lenguajes de alto nivel

Lenguajes de propósito general:

— PL/1	— Pascal	— Logo
— BASIC	— Módulo II	— Forth
— C	— COBOL	— Ada

Lenguajes para cálculo científico:

— FORTRAN	— APL	— Ada
— Pascal	— ALGOL	

Lenguajes orientados a la gestión:

— COBOL	— RPG
---------	-------

Lenguajes de simulación en general:

— GPSS	— SIMULA	— MIMIC
--------	----------	---------

En el sistema educativo:

— ASSET	— C	— Módulo II
— Logo	— BASIC	— PILOT
— Pascal		

Lenguajes orientados al objeto:

— SMALLTALK

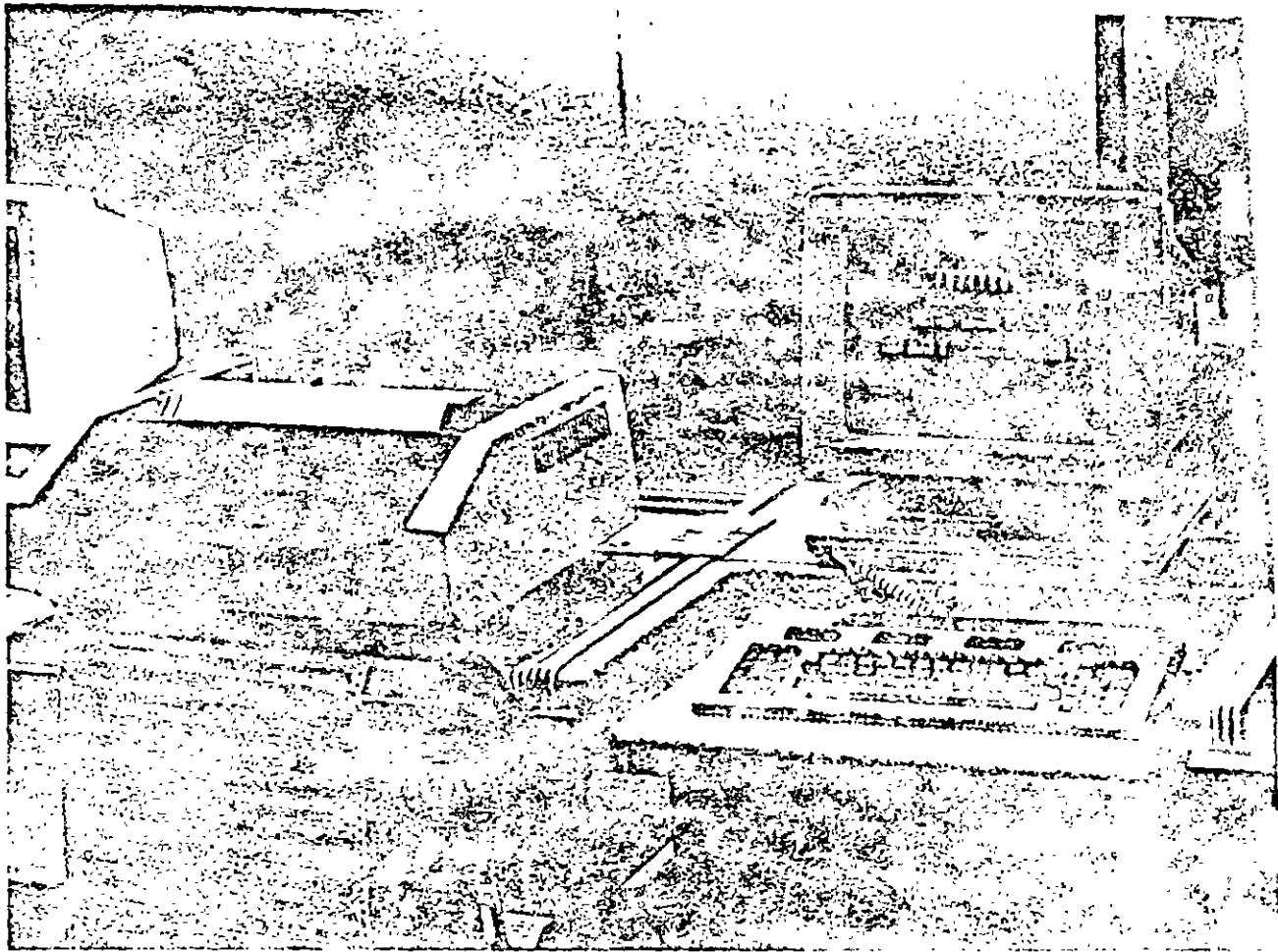
Lenguajes interrogativos:

— PROLOG	— DBASE
----------	---------

Lenguajes para la I.A.

— LISP	— PROLOG
--------	----------

7. Los lenguajes de alto nivel



7.1 ¿Qué es un lenguaje informático de alto nivel?

Hoy en día, la mayoría de los programas de ordenador se escriben utilizando lenguajes de alto nivel, que se han venido desarrollando a lo largo de los treinta últimos años y de los que ya existe una gran variedad. Estos lenguajes permiten la realización de programas más cortos a la vez que más independientes del ordenador (transportables).

Los programas escritos en estos lenguajes son

mucho más fáciles de redactar porque evitan al programador el trabajo de detallar cada una de las operaciones del lenguaje máquina. Pero esta ventaja tiene su contrapartida en las transformaciones a que tiene que someterse el texto del programa así creado hasta que se convierta en un programa ejecutable (en código máquina).

Cuando se trabaja con un lenguaje de alto nivel, el texto del programa, o programa fuente, no es directamente ejecutable por el ordenador; por lo que hay que realizar una transformación o compilación

texto fuente. La creación del texto fuente se realiza por medio de un editor de textos y su resultado generalmente adquiere la forma de un archivo.

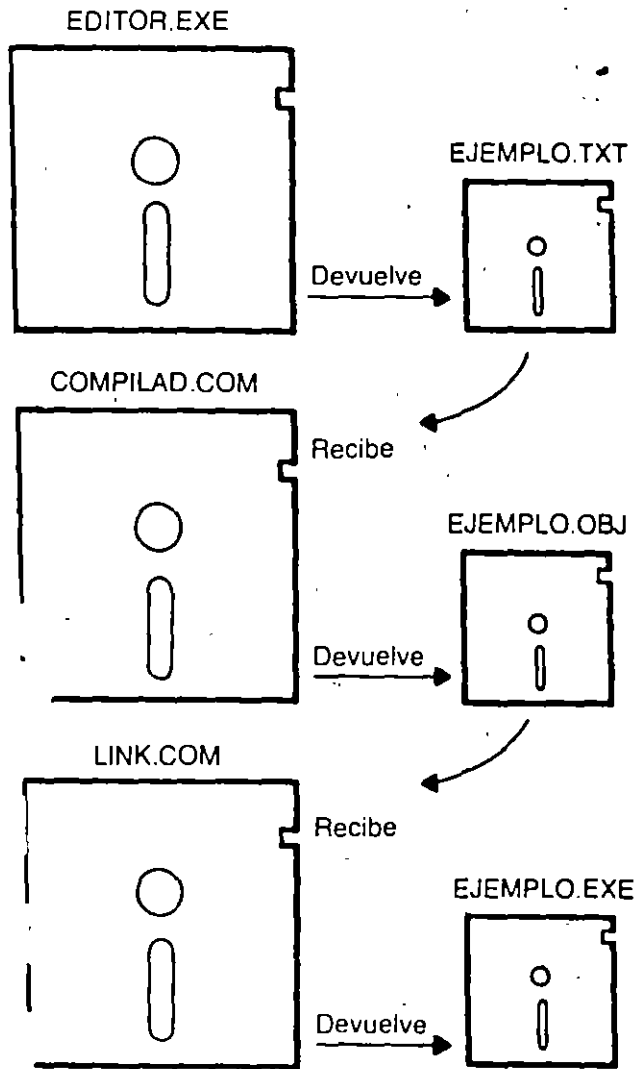


Fig. 7.1

- 1. Uso de un programa editor para escribir el programa se utilizan las reglas del editor; el texto del programa debe respetar la sintaxis del lenguaje resultado: EJEMPLO.TXT
- 2. Uso de un programa compilador generalmente no requiere la intervención del usuario resultado: EJEMPLO.OBJ
- 3. Uso de un programa linkador tampoco requiere generalmente la intervención del usuario; el fichero que crea ya es directamente ejecutable resultado: EJEMPLO.EXE

Los lenguajes de alto nivel, juntamente con los avances tecnológicos que han permitido construir ordenadores cada vez más potentes, de tamaño más reducido y más económicos, han contribuido a la popularización de los ordenadores permitiendo un uso que ya no está restringido a una pequeña élite de «sabios informáticos».

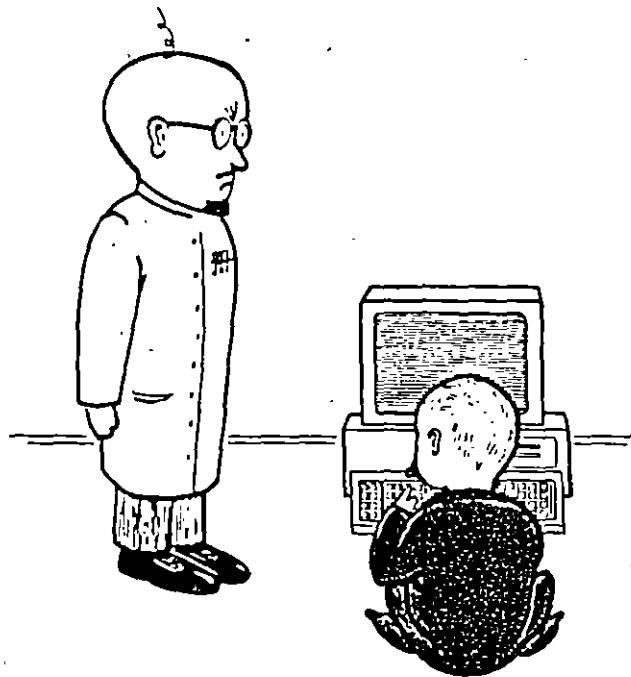


Fig. 7.2 Los lenguajes de alto nivel han ampliado el número de posibles programadores.

7.2 La sintaxis y el vocabulario

Un programa en un lenguaje de alto nivel es un conjunto de *sentencias* cuya construcción estará regida por una serie de *reglas* que permiten combinar los elementos del lenguaje. Estas normas son mucho menos estrictas y del nivel de detalle menos minucioso que en el lenguaje de bajo nivel.

Cada *sentencia* corresponde a varias instrucciones máquina y tiene un formato que combina palabras reservadas —vocabulario— con partes variables —parámetros—. En las *sentencias* pueden combinarse las palabras reservadas con:

- textos
- números
- operadores
- variables.

54

PROGRAMA Lista de sentencias
 SENTENCIA Lista de instrucciones
 INSTRUCCIÓN Palabra y parámetros

Por ejemplo, en BASIC el vocabulario está formado por un conjunto de palabras reservadas —PRINT, INPUT, IF, GOTO, ... y signos especiales— de manera que la construcción de las sentencias se hará por combinación de los elementos de éste mediante reglas del tipo «las palabras reservadas deberán ir precedidas y seguidas de un separador», o «para la reunión de dos o más sentencias en una misma línea de programa será necesario utilizar el separador "dos puntos" (:) entre cada sentencia».

Cada lenguaje de alto nivel tiene su propio vocabulario y conjunto de reglas para construir las sentencias. Los siguientes ejemplos, utilizando diversos lenguajes de alto nivel, indican a un ordenador que lea un número y que, si éste es mayor que cero, escriba que es positivo.

En BASIC :
100 INPUT A :IF A>0 THEN PRINT "Número positivo"
En Logo :
HAZ "NUMERO LEEPALABRA SI :NUMERO > 0 [ESCRIBE (Número positivo)]
En Pascal :
Program Sencillo; var número : real; Begin Readln (Número); If Número > 0 Writeln ('Número positivo'); End.

Fig. 7.3

7.3 Los traductores: compiladores e intérpretes

Los *programas traductores* están a un nivel similar a los programas del sistema. Se encargan de materializar o crear la máquina virtual para que se pueda programar en un lenguaje de alto nivel. Los traductores son programas que transforman un programa escrito en un lenguaje determinado a otro escrito en lenguaje máquina.

Un programa ensamblador es también un tra-

ductor entre el programa escrito en lenguaje ensamblador y el código máquina.

Según el método que se utilice en la traducción los programas traductores serán:

- Compiladores
- Intérpretes.

7.4 Los compiladores

Los *compiladores* traducen en su totalidad el texto del programa original escrito en lenguaje de alto nivel a un programa análogo, pero escrito en código máquina, o a un código intermedio que luego puede ser traducido a código máquina (véase fig. 7.1). Cada fase del proceso se almacena en la memoria del ordenador y puede guardarse en la memoria externa.

El *programa objeto* que se ha generado normalmente es más largo que el *programa fuente* del que proviene puesto que cada instrucción en alto nivel, una vez traducida, puede convertirse en varias instrucciones objeto.

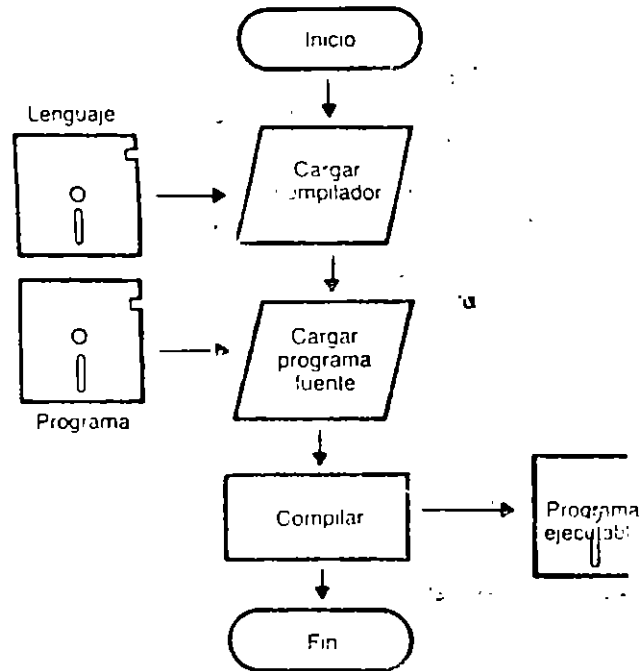


Fig. 7.4 Esquema de la relación programa fuente, compilador, ordenador y programa objeto.

Si cada vez que se ejecuta un programa hace falta compilarlo, el proceso es más largo que si el programa es interpretado, pero una vez que se ha compilado puede guardarse el código objeto y entonces queda traducido para siempre y podrá ejecutarse sin necesidad de ser compilado de nuevo. Sólo en el caso de que deban realizarse modificaciones en el programa, éstas se harán sobre el programa fuente y entonces será necesario compilarlo de nuevo antes de su ejecución.

Los compiladores son propios de cada lenguaje de programación y sólo sirven para trabajar en aquel lenguaje. Para programar en Pascal se necesitará un compilador de Pascal, y si se quiere programar en Lisp deberá usarse un compilador de este lenguaje.

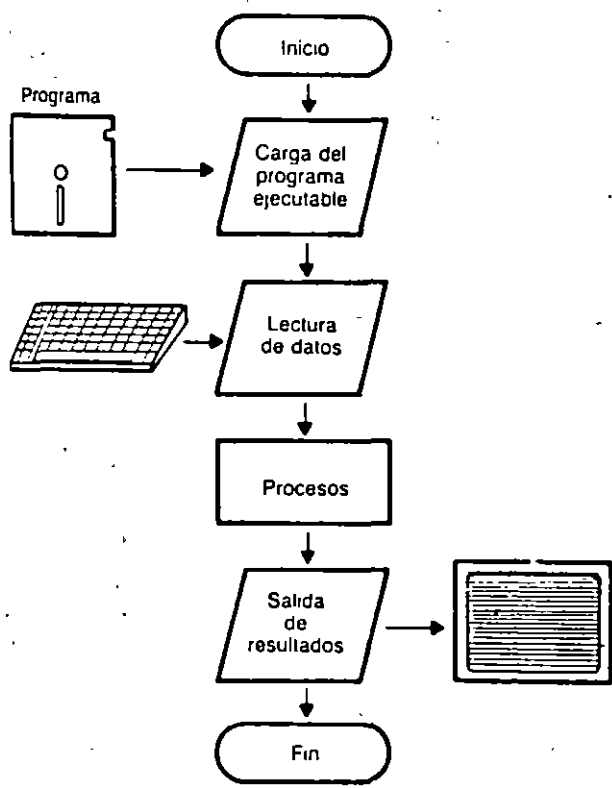


Fig. 7.5 Esquema de la ejecución de un programa compilado.

7.5 Los intérpretes

Otra modalidad de la traducción es lo que se llama interpretación. Un intérprete es un programa

traductor que coexiste en la memoria del ordenador con el programa del usuario y lo va traduciendo a medida que se ejecutan sus sentencias, al igual que lo haría un intérprete de una lengua extranjera en una «traducción simultánea».

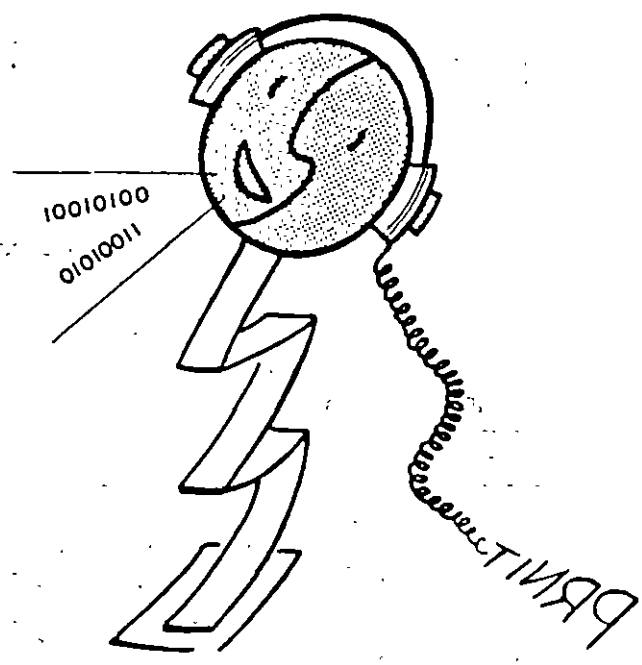


Fig. 7.6 El ordenador necesita un traductor para entender los programas escritos en un lenguaje de alto nivel.

Generalmente, los programas intérpretes constan también de un editor más o menos completo con el que el texto del programa puede crearse y modificarse.

El proceso de análisis y traducción se realiza a la vez que el de ejecución, por lo que, en este caso, no se crea ningún programa objeto capaz de ser guardado en la memoria externa. Por lo tanto, un programa escrito en un lenguaje interpretado deberá traducirse cada vez que se ejecuta, lo que hará que deba invertirse más tiempo en el proceso de ejecución que si el programa estuviera compilado.

Para realizar modificaciones a un programa en un lenguaje interpretado bastará con cambiar la o las líneas que se deseen.

Los intérpretes, al igual que los compiladores, también son propios de cada lenguaje de alto nivel; pueden ser residentes o no residentes. Se dice que un programa es residente si está incorporado a la

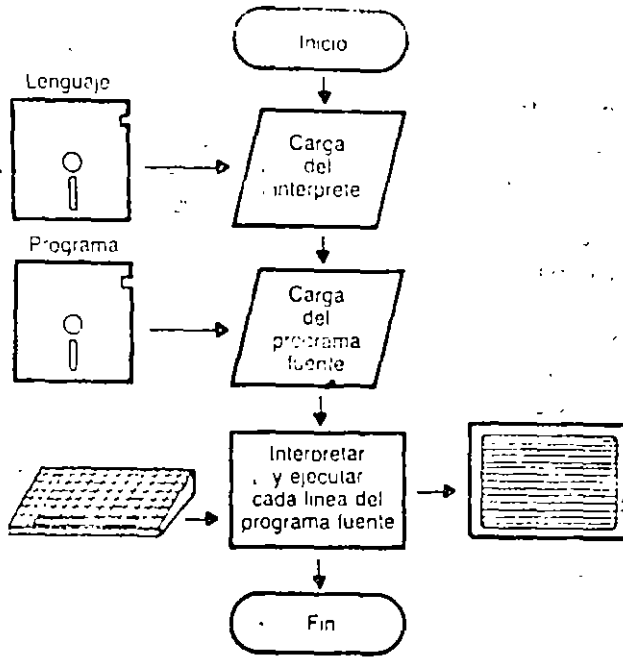


Fig. 7.7 La ejecución de un programa de alto nivel por medio de un interpretador (por ejemplo del GWBASIC).

ROM del ordenador y *no residente* si hay que cargarlo en la máquina procedente de un soporte externo.

Los programas compiladores suelen ser más complejos que los intérpretes, pero el código objeto generado por un compilador es más veloz de ejecución que el programa interpretado.

7.6 ¿Lenguajes interpretados o compilados?

Por lo general los ordenadores pequeños —personales y domésticos— suelen trabajar con lenguajes interpretados puesto que muchos de ellos ya salen de fábrica con un lenguaje incorporado en la ROM y porque un programa fuente en lenguaje de alto nivel ocupa menos espacio y deja más memoria libre para el usuario, lo que es de considerable importancia en los ordenadores de poca capacidad.

Sin embargo, en las dos últimas décadas se han diseñado compiladores de algunos de los lenguajes informáticos más corrientes que están al alcance de los microordenadores y ordenadores domés-

ticos, lo que ha contribuido a la difusión del uso de lenguajes como el Pascal.

Para las fases de desarrollo y prueba de un programa, los lenguajes interpretados ofrecen la ventaja de ser más flexibles por lo que permiten realizar pruebas y modificaciones de manera sencilla y rápida. Los errores de sintaxis se pueden localizar y corregir fácilmente y es posible diseñar conjuntos de pruebas que comprueben el funcionamiento de módulos concretos del programa sin necesidad de ejecutarlo todo.

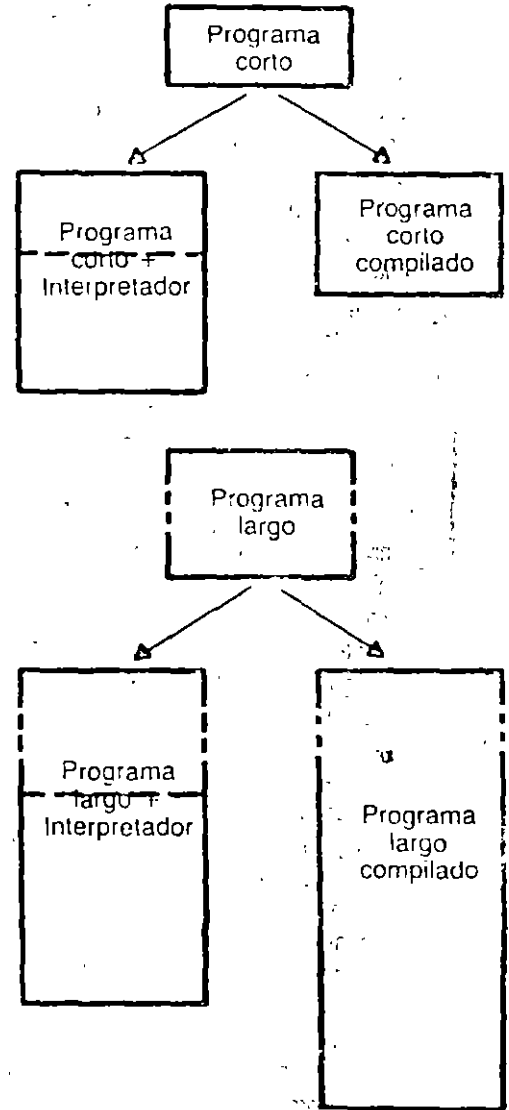


Fig. 7.8 Comparación entre la ocupación de memoria de un programa interpretado y compilado.

Los lenguajes interpretados permiten mayor interacción entre el usuario-programador y la máquina, lo que la convierte en un instrumento más accesible desde el primer momento.

Cuando el programa ya está probado y funciona correctamente la compilación tiene sus ventajas ya que a igual magnitud de programa, si éste está compilado, se ejecuta con mayor rapidez; ya no es necesario que se traduzca sentencia por sentencia cada vez que se ejecuta.

La ocupación de la memoria dependerá, evidentemente, de la longitud del programa, pero si el lenguaje es interpretado el programa deberá compartirla con el interpretador. Por lo tanto, un programa corto interpretado más el interpretador ocupará más memoria que el mismo programa compilado. Si el programa es largo, al compilarlo se alargará mucho más debido al código que se genera y, en este caso, probablemente sobrepasará la cantidad de memoria que se requiere cuando es interpretado.

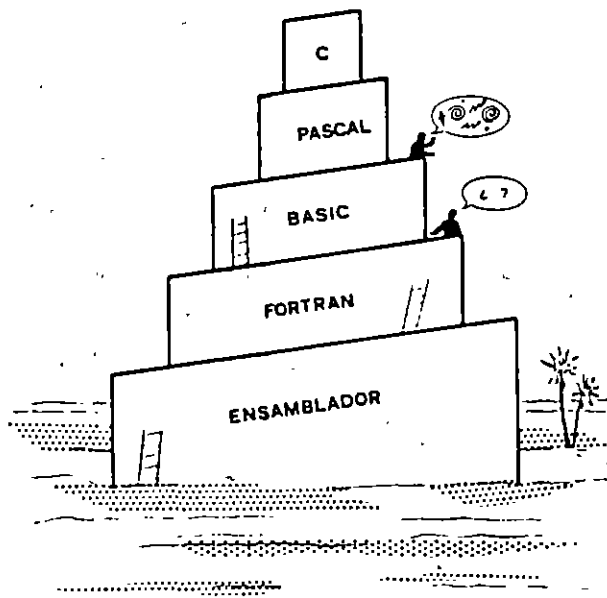


Fig. 7.9 Un programa escrito en Pascal es totalmente incomprensible para un intérprete de BASIC.

7.7 Los lenguajes estructurados

Una de las aportaciones de los lenguajes de alto nivel ha sido la de posibilitar el uso de metodologías de análisis y programación estructurada —el conocimiento de cómo se relacionan entre sí los centenares de sentencias que pueden constituir un programa—. Algunos lenguajes son más apropiados que otros para adaptarse a esta técnica de programación; a aquellos que contienen las estructuras básicas les llamamos *lenguajes estructurados*.

Los lenguajes estructurados permiten escribir los programas como secuencias de pequeños módulos, cada uno de ellos realiza una tarea concreta y la relación entre ellos queda determinada claramente y sin ambigüedades. La redacción de cada módulo estará de acuerdo con alguna de las estructuras admitidas o bien será combinación de varias.

Algunos módulos podrán constituirse como subprogramas o subrutinas, en algunos casos llamadas procedimientos. Los subprogramas y subrutinas se invocarán desde el programa principal cada vez que se requieran.

En particular, los procedimientos tienen su propio nombre que los define y es al que se invocará cada vez que se vaya a utilizar el procedimiento.

Otra particularidad de los procedimientos es que pueden llamarse a sí mismos, característica que se conoce con el nombre de «recursividad» y que dota al lenguaje de una gran potencia.

El primer lenguaje estructurado fue el ALGOL. Hacia la década de los años setenta aparecieron el Lisp, el Logo y el Pascal entre otros, y más recientemente el Prolog y el ADA.

Vocabulario:

- Sentencias
- Reglas
- Compilador
- Intérprete
- Programa fuente
- Programa objeto
- Programa residente
- Subrutina.

Ejercicios:

- 7.1. Explicar con palabras propias el significado de cada uno de los términos que aparecen en el vocabulario.
- 7.2. A partir de los tres ejemplos del apartado 2 deducir algunas de las reglas de cada uno de los tres lenguajes utilizados.

7.3. Explicar la razón por la que es necesario utilizar un programa traductor cuando se escriben programas en un lenguaje de alto nivel.

7.4. ¿Cuál es la función de un programa compilador? ¿Y la de un interpretador?

7.5. Citar analogías y diferencias entre los programas interpretadores y los compiladores.

7.6. ¿Cuándo es más práctico usar un lenguaje interpretado? ¿Y uno compilado?

7.7. La comunicación con el ordenador se puede realizar a distintos niveles: código máquina, lenguaje ensamblador, lenguaje de alto nivel interpretado y lenguaje de alto nivel compilado.

Explicar las diferencias entre las cuatro posibilidades anteriores con relación a la cantidad de memoria necesaria, la facilidad de programación y depuración, la facilidad de uso, el tiempo de programación y el tiempo de ejecución del programa.

7.8. Citar los nombres de algunos lenguajes de los llamados estructurados.

Sugerencias de ampliación

Comparando la compilación con la interpretación:

Facilidad para el desarrollo de programas, localización de errores, realización de pruebas	Interpretación
Facilidad para realizar modificaciones	Interpretación
Posibilidad de convertir una lista en un programa ejecutable (véase el caso de Logo, Lisp...)	Interpretación
Posibilidad de trabajo en modo interactivo	Interpretación
Rapidez en la ejecución de un programa	Compilación
Buena optimización del código objeto	Algunos compiladores
Cantidad de memoria ocupada	Depende de la longitud del programa fuente y la del interpretador
Potencia del lenguaje	Equivalente