

**FACULTAD DE INGENIERIA U.N.A.M.
DIVISION DE EDUCACION CONTINUA**

CENTRO DE INFORMACION Y DOCUMENTACION

"ING. BRUNO MASCANZONI"

EL CENTRO DE INFORMACION Y DOCUMENTACION "ING. BRUNO MASCANZONI" TIENE POR OBJETIVO SATISFACER LAS NECESIDADES DE ACTUALIZACION AL PROPORCIONAR LA ADECUADA INFORMACION QUE PERMITA A LOS PROFESIONALES INGENIEROS PROFESORES Y ALUMNOS, ESTAR AL TANTO DEL ESTADO ACTUAL DEL CONOCIMIENTO SOBRE TEMAS ESPECIFICOS ENFATIZANDO LAS INVESTIGACIONES DE VANGUARDIA DE LOS CAMPOS DE LA INGENIERIA TANTO NACIONALES COMO EXTRANJERAS.

POR LO QUE SE PONE A DISPOSICION DE LOS ASISTENTES DE LOS CURSOS DE LA D.E.C.F.I.; ASI COMO AL PUBLICO EN GENERAL.

EN DICHO CENTRO USTED TENDRA LOS SIGUIENTES SERVICIOS:

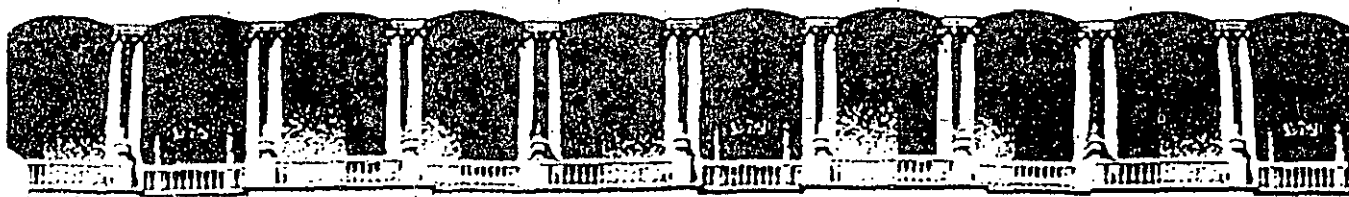
- * PRESTAMO INTERNO
- * PRESTAMO EXTERNO
- * PRESTAMO INTERBIBLIOTECARIO
- * SERVICIO DE FOTOCOPIADO
- * CONSULTA TELEFONICA
- * CONSULTA A LOS BANCOS DE DATOS: LIBRUNAM EN CD-ROM Y EN LINEA

LOS MATERIALES A SU DISPOSICION SON:

- * LIBROS
- * TESIS DE POSGRADO
- * NOTICIAS TECNICAS
- * PUBLICACIONES PERIODICAS
- * PUBLICACIONES DE LA ACADEMIA MEXICANA DE INGENIERIA
- * NOTAS DE LOS CURSOS QUE SE HAN IMPARTIDO DE 1971 A LA FECHA

EN LAS AREAS DE INGENIERIA INDUSTRIAL, CIVIL, ELECTRONICA, CIENCIAS DE LA TIERRA, MECANICA Y ELECTRICA Y COMPUTACION.

EL C.I.D. SE ENCUENTRA UBICADO EN EL MEZZANINE DEL PALACIO DE MINERIA LADO ORIENTE. EN HORARIO DE SERVICIO DE 10:00 A 19:30 HORAS DE LUNES A VIERNES.



**FACULTAD DE INGENIERIA U.N.A.M.
DIVISION DE EDUCACION CONTINUA**

A LOS ASISTENTES A LOS CURSOS DE LA DIVISION DE EDUCACION CONTINUA

Las autoridades de la Facultad de Ingeniería, por conducto del Jefe de la División de Educación Continua, otorgan una constancia de asistencia a quienes cumplan con los requisitos establecidos para cada curso.

El control de asistencia se llevará a cabo a través de la persona que le entregó las notas. Las inasistencias serán computadas por las autoridades de la División, con el fin de entregarle constancia solamente a los alumnos que tengan un mínimo del 80% de asistencias.

Pedimos a los asistentes recoger su constancia el día de la clausura. Estas se retendrán por el período de un año, pasado este tiempo la DECFI no se hará responsable de este documento.

e recomienda a los asistentes participar activamente con sus ideas y experiencias, pues los cursos que ofrece la División están planeados para que los profesores - expongan una tesis, pero sobre todo, para que coordinen las opiniones de todos los interesados, constituyendo verdaderos seminarios.

Es muy importante que todos los asistentes llenen y entreguen su hoja de inscripción al inicio del curso, información que servirá para integrar un directorio de asistentes, que se entregará oportunamente.

Con el objeto de mejorar los servicios que la División de Educación Continua ofrece, al final del curso deberán entregar la evaluación a través de un cuestionario diseñado para emitir juicios anónimos.

Se recomienda llenar dicha evaluación conforme los profesores impartan sus clases, a efecto de no llenar en la última sesión las evaluaciones y con esto sean más fehacientes sus apreciaciones.

¡ G R A C I A S !

UNO DE LOS PROYECTOS QUE ACTUALMENTE ESTA LLEVANDO A CABO LA DECFI, ES LA ORGANIZACIÓN DE CURSOS DE ACTUALIZACIÓN EN TEMAS DE INGENIERÍA, DENTRO DE LOS CUALES SE INCLUYEN - PROGRAMAS DE COMPUTADORA RELACIONADOS CON EL TEMA DEL CURSO, LOS CUALES SE DISTRIBUIRÁN EN SUS VERSIONES FUENTE.

CON EL OBJETO DE CONOCER LOS TEMAS DE MAYOR INTERÉS PARA ESTE TIPO DE CURSOS, ASÍ COMO PARA DEFINIR LOS REQUISITOS TÉCNICOS QUE DEBEN REUNIR LOS PROGRAMAS A DISTRIBUIR, MUCHO AGRADECEREMOS A USTED SE SIRVA LLENAR EL SIGUIENTE CUESTIONARIO, EL CUAL SERÁ DE UNA GRAN AYUDA PARA LA DECFI.

1.- CALIFIQUE CON ESCALA DE CERO A DIEZ LOS SIGUIENTES CURSOS UTILIZANDO LAS LÍNEAS EN BLANCO PARA AQUELLOS QUE USTED PROPONGA (0=NO INTERESA, 10=INTERESA MUCHO)

ANÁLISIS ESTRUCTURAL ()	ESTADÍSTICA ()	CONTROL DE PERSONAL ()
CONTROL DE OBRAS ()	DISEÑO MECÁNICO ()	ALMACENES ()
RUTA CRÍTICA ()	PROGRAMACIÓN ESTRUC. ()	INV. DE OPERACIONES ()
PROGRAMACIÓN LINEAL ()	ESTRUCTURA DE DATOS ()	CONTROL DE CALIDAD ()
MATEMÁTICAS ()	CONTABILIDAD ()	ADMON. PROGRAMACIÓN DE LA PRODUCCIÓN ()
_____ ()	_____ ()	_____ ()
_____ ()	_____ ()	_____ ()
_____ ()	_____ ()	_____ ()

DEBIDO A QUE LA PRINCIPAL CARACTERÍSTICA DE LOS CURSOS SERÍA LA DE DISTRIBUIR PROGRAMAS DE COMPUTADORA QUE PUEDAN SER USADAS POR LOS ASISTENTES EN SUS DIFERENTES EMPRESAS CON EL MENOR ESFUERZO DE ADAPTACIÓN.

2.- ¿PARA QUE TIPO DE COMPUTADORA DESEARÍA QUE SE ESCRIBIERAN LOS PROGRAMAS?

PRIMERA OPCIÓN MARCA _____	MODELO _____	LENGUAJE _____
SEGUNDA OPCIÓN MARCA _____	MODELO _____	LENGUAJE _____
TERCERA OPCIÓN MARCA _____	MODELO _____	LENGUAJE _____

SI USTED CONOCE ALGUNAS OTRAS PERSONAS INTERESADAS EN ESTE TIPO DE CURSOS, MUCHO LE AGRADECEREMOS HACERLE LLEGAR UNA COPIA DE ESTA HOJA Y ENVIARLA POSTERIORMENTE A:

DIVISIÓN DE EDUCACIÓN CONTINUA

PALACIO DE MINERÍA

CALLE DE TACUBA No. 5

DELEGACIÓN CUAUHTEMOC

06000 MÉXICO, D.F.

10.- DIRECCION DE OFICINA:

39	CALLE, NUMERO EXTERIOR E INTERIOR	72	A 3	M 7
			80	80
8	COLONIA	37		
38	DELEGACION O CIUDAD	57	ESTADO	
	CODIGO POSTAL 			
	60	64		

11.- ASOCIACIONES A LAS QUE PERTENECE :

PRINCIPAL :

65	66

OTRAS :

67	68

69	70

71	72

73	74

A 4	M 8
80	80

FECHA DE ELABORACION

_____ A _____ DE _____ DE 19 _____

FIRMA

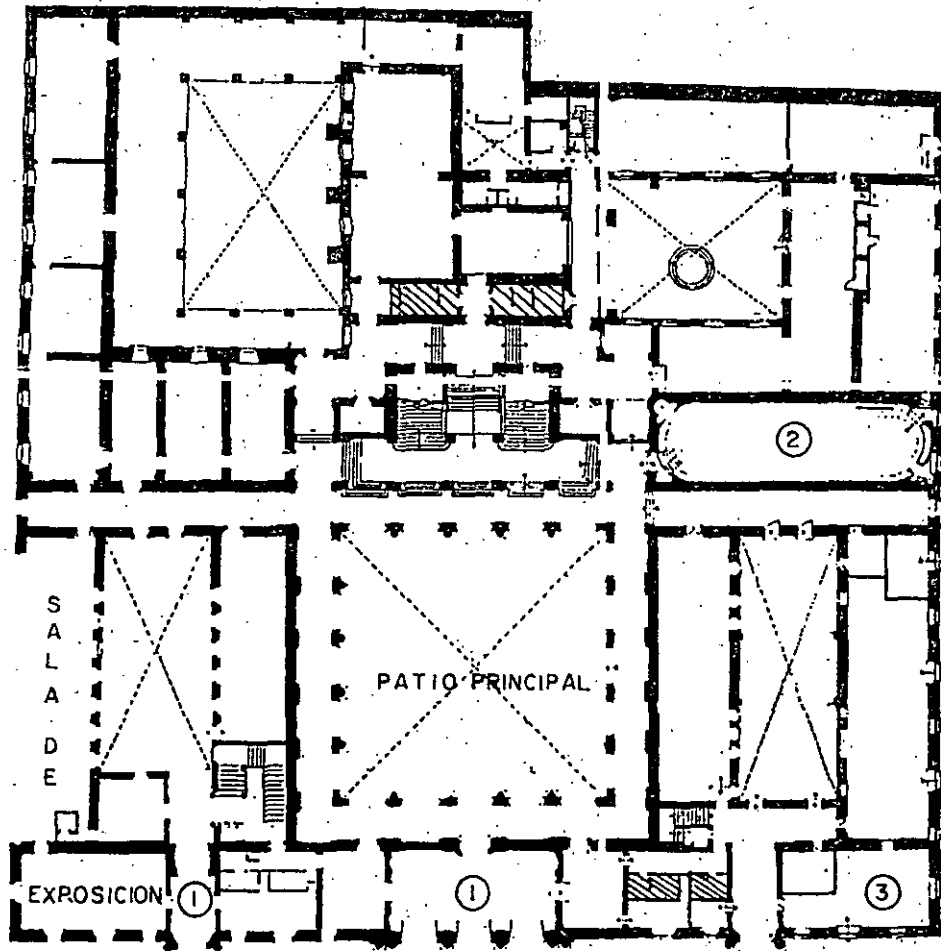
PARA USO EXCLUSIVO DE LA DIVISION DE EDUCACION CONTINUA

CODIFICO:

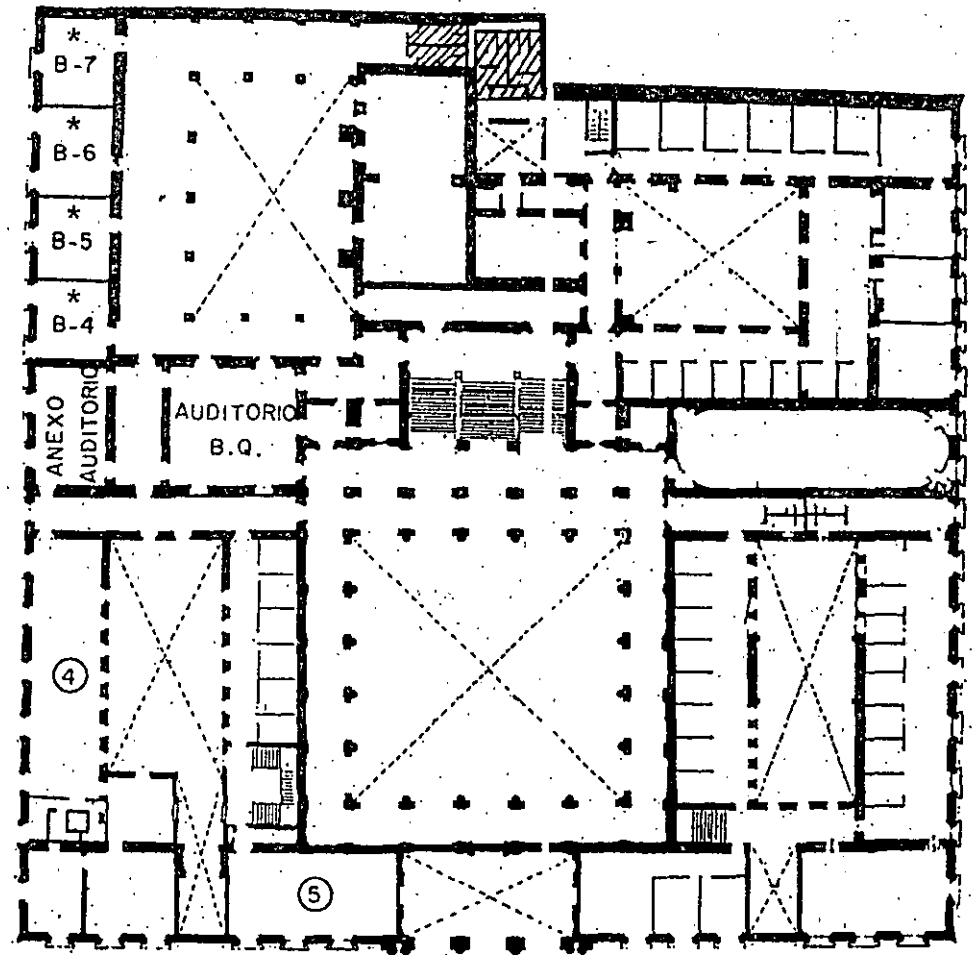
REVISO:

OBSERVACIONES:

PALACIO DE MINERIA



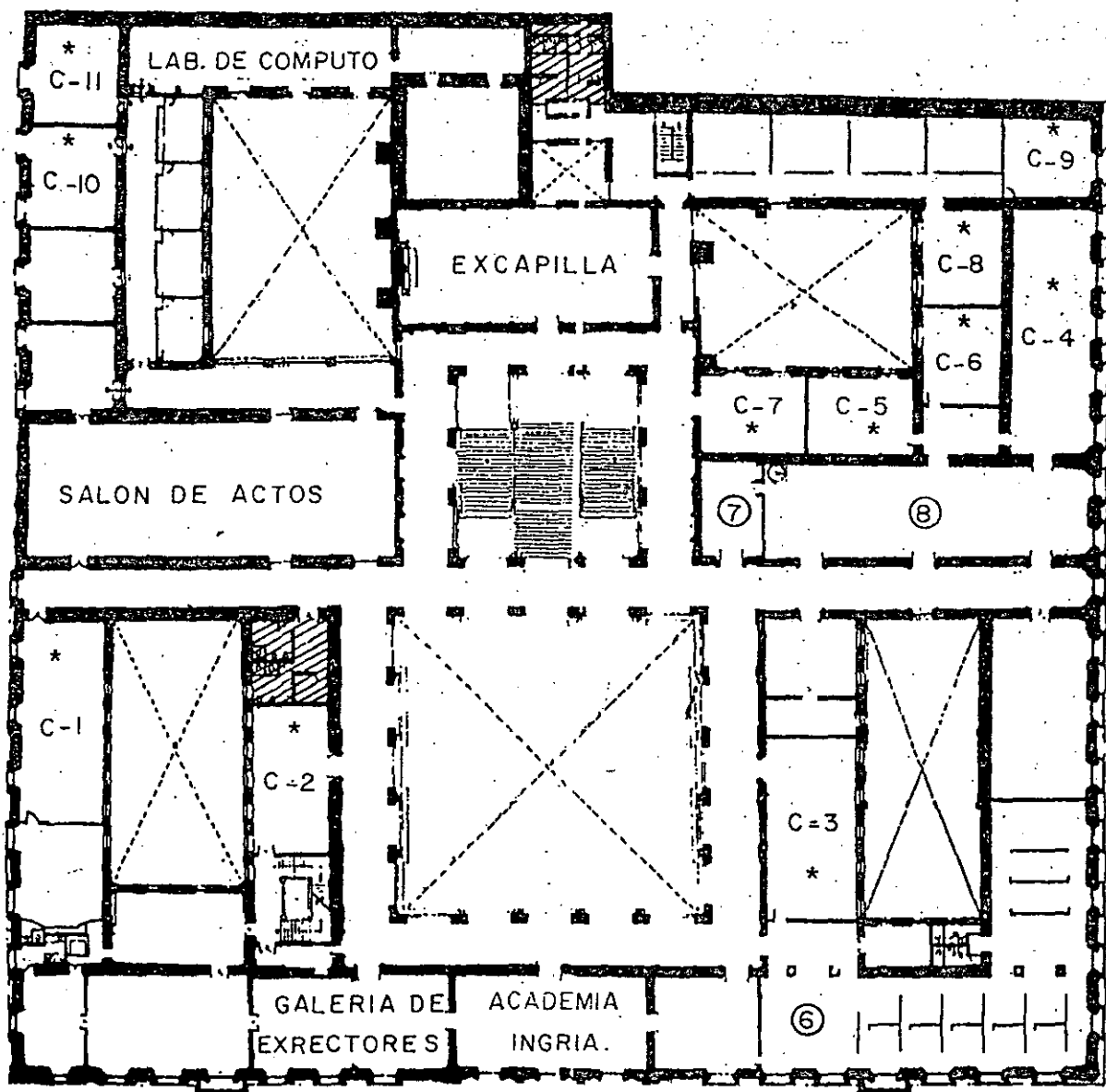
PLANTA BAJA



MEZZANINNE



DIVISION DE EDUCACION CONTINUA
FACULTAD DE INGENIERIA U.N.A.M.
CURSOS ABIERTOS



GUIA DE LOCALIZACION

- 1 - ACCESO
- 2 - BIBLIOTECA HISTORICA
- 3 - LIBRERIA U N A M
- 4 - CENTRO DE INFORMACION Y DOCU-
MENTACION "ING. BRUNO
MASCANZONI"
- 5 - PROGRAMA DE APOYO A LA
TITULACION
- * AULAS
- 6 - OFICINAS GENERALES
- 7 - ENTREGA DE MATERIAL Y CONTROL
DE ASISTENCIA.
- 8 - SALA DE DESCANSO
- ▨ SANITARIOS

1er. PISO



**FACULTAD DE INGENIERIA U.N.A.M.
DIVISION DE EDUCACION CONTINUA**

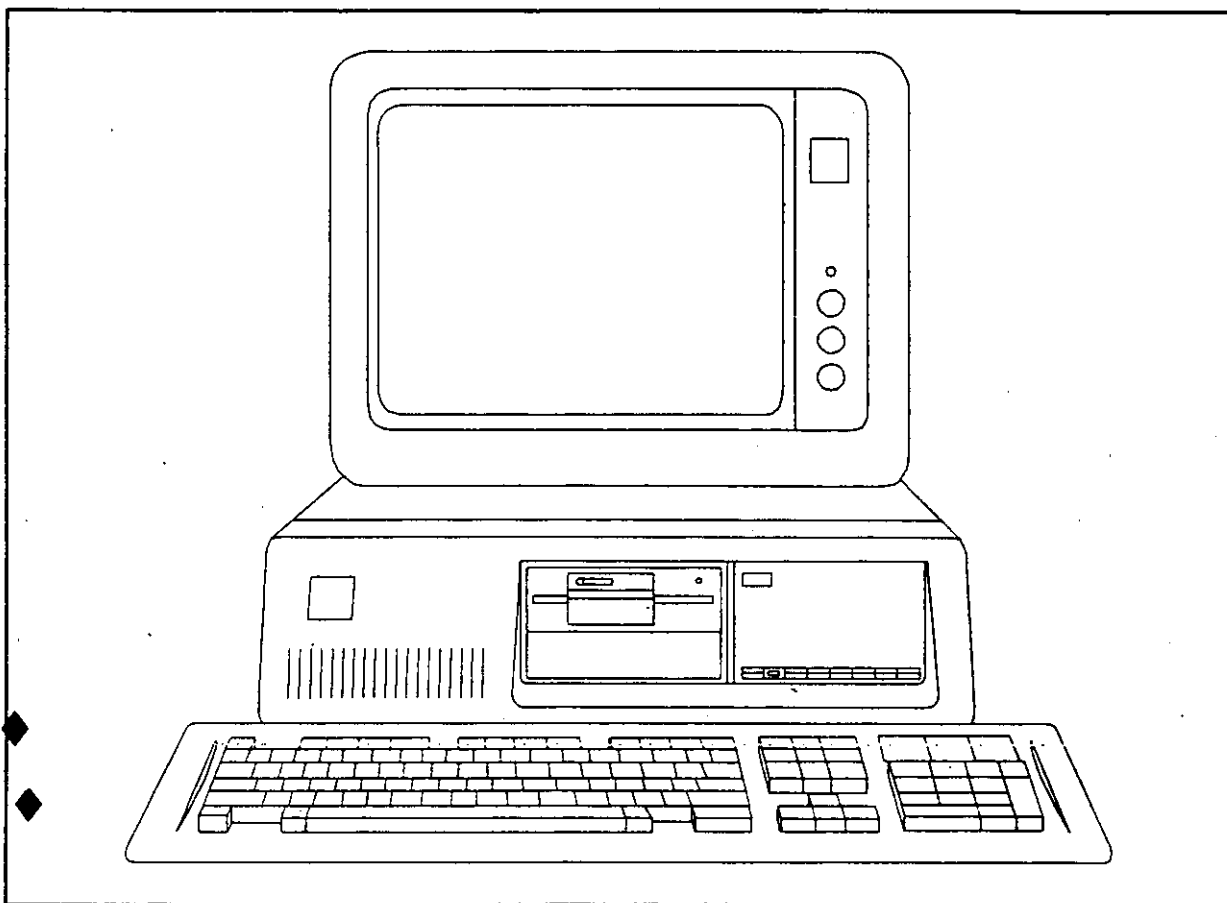
INTRODUCCION A LAS
COMPUTADORAS PERSONALES

MATERIAL DE APOYO
DIDACTICO

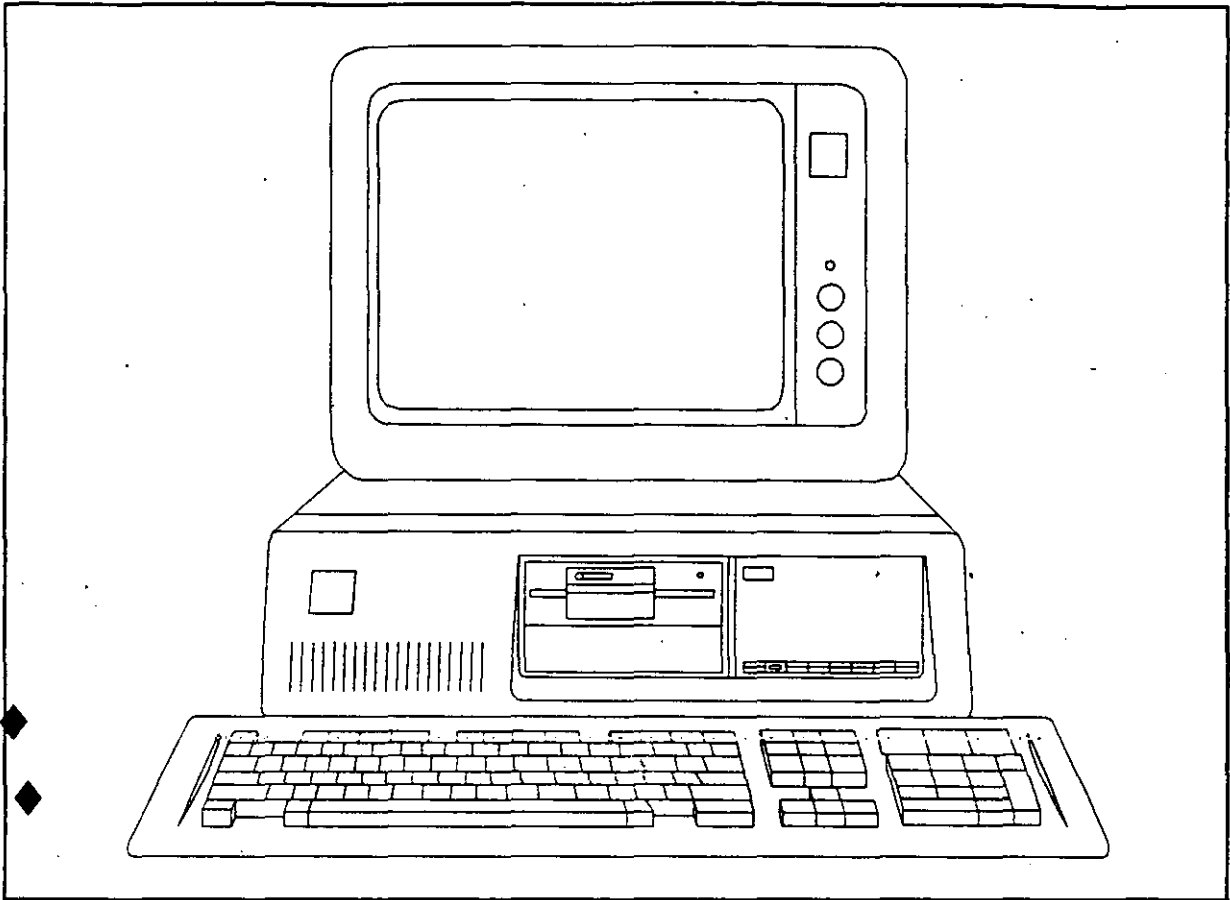
SEPTIEMBRE-OCTUBRE, 92.

I N D I C E

INTRODUCCIÓN A LAS COMPUTADORAS PERSONALES	000001
INTRODUCCIÓN	000003
INTRODUCCIÓN A LAS COMPUTADORAS PERSONALES	000004
INTRODUCCIÓN AL MANEJO DE INFORMACIÓN	000018
QUÉ SON Y POR QUE UTILIZAR LAS COMPUTADORAS PERSONALES	000027
INTRODUCCIÓN A MS-DOS	000044
INGENIERÍA DE SOFTWARE	000121
EJEMPLOS Y PRÁCTICAS	000185
LA COMPUTACIÓN EN MÉXICO	000205
GLOSARIO	000251
BIBLIOGRAFIA	000265



**INTRODUCCION A LAS
COMPUTADORAS PERSONALES**



INTRODUCCION

INTRODUCCION

LAS NOTAS DE ESTE CURSO SON RESULTADO DE LA COMPILACION Y EXPERIENCIA DEL COORDINADOR Y LOS INSTRUCTORES EN APROXIMADAMENTE 20 OCASIONES QUE SE HA IMPARTIDO EN LA DIVISION DE EDUCACION CONTINUA DE LA FACULTAD DE INGENIERIA.

DE NINGUNA MANERA SE PRETENDE CUBRIR LAS EXPECTATIVAS DE UN LIBRO DE TEXTO NI MUCHO MENOS; COMO PODRA COMPROBARSE SE CUENTA CON INFORMACION DE TRES TIPOS.

- APUNTES Y NOTAS PREPARADAS DIRECTAMENTE POR LOS PROFESORES.
- UNA SELECCION DE ARTICULOS Y COPIAS DE LIBROS Y REVISTAS QUE COMPLEMENTAN O CUBREN PUNTOS ESPECIFICOS DE LA TEORIA DEL CURSO.
- PRACTICAS Y SESIONES GUIADAS CON EJEMPLOS QUE INTEGRAN LA PARTE MEDULAR DE LAS EXPOSICIONES EN LABORATORIO; EN ESTA PARTE VALE ACLARAR QUE LA SELECCION DE EJEMPLOS VARIA Y SE ADECUA DEPENDIENDO DEL PERFIL DE LOS PARTICIPANTES EN CADA CASO, EJEMPLOS INGENIERILES, ADMINISTRATIVOS, ETC.

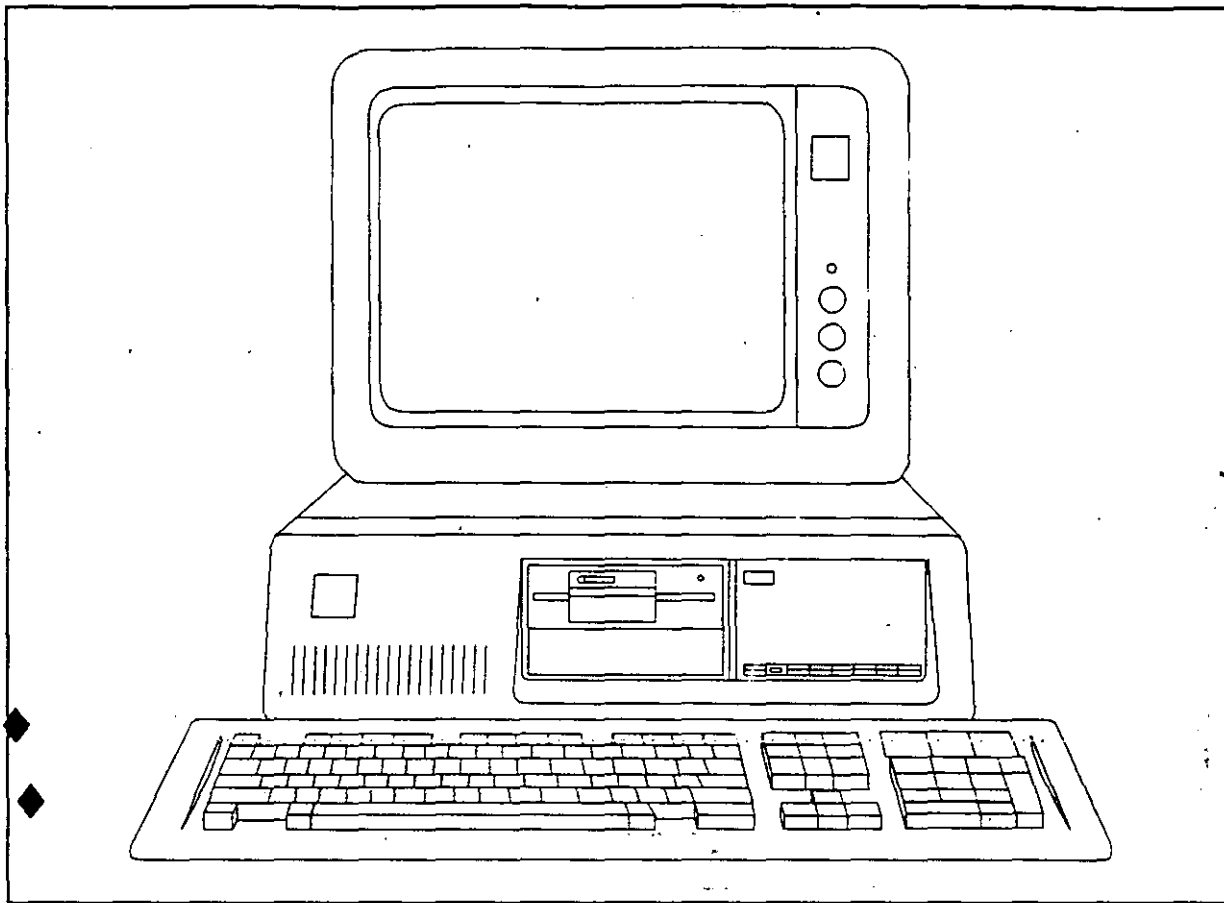
ES NECESARIO MENCIONAR QUE LA TOTALIDAD DE LOS PUNTOS PLANTEADOS EN EL TEMARIO TEORICO SE CUBREN EN LAS NOTAS; ASI MISMO, EXISTE INFORMACION ADICIONAL COMPLEMENTARIA, YA QUE EL CURSO SE CONSIDERA UN CURSO INFORMATIVO Y NO FORMATIVO.

SE INTEGRA UNA GUIA DE ACCESO A LAS NOTAS CON LAS ACLARACIONES RESPECTIVAS Y LA UBICACION DENTRO DE SU CARPETA.

CUALQUIER OBSERVACION O SUGERENCIA PARA MEJORAR LAS PRESENTES NOTAS SERA BIEN RECIBIDA POR CUALQUIERA DE LOS INSTRUCTORES O POR EL COORDINADOR DEL CURSO.

ATENTAMENTE.

ING. HECTOR ARRONA URREA.



**QUE SON Y POR QUE UTILIZAR
COMPUTADORAS PERSONALES**

INTRODUCCION A LAS COMPUTADORAS PERSONALES

¿QUE SON Y PORQUE UTILIZAR COMPUTADORAS PERSONALES?

¿NECESIDAD Y/O AVENTURA?

En la actualidad, nadie tiene que decirnos que el mundo de la computación es diferente, esto lo entendemos pero no lo asimilamos, nos intimidan los términos no familiares, los lenguajes extraños, los tecnicismos y en ocasiones hasta el idioma inglés.

A pesar de que cada vez oímos con mayor frecuencia sobre lo que hacen o dicen que hacen las computadoras, es posible que nos encontremos como millones de personas, sin una visión clara de lo que son, lo que hacen, para qué sirven y cómo podemos utilizarlas.

Si este es tu primer contacto formal con el mundo de las computadoras, de algo debes estar seguro, será una experiencia apasionante aunque en ocasiones sea frustrante, muchas de estas frustraciones proveendrán de las limitaciones humanas, como por ejemplo, el tiempo necesario para observar las cantidades impresionantes de información y conocimientos.

Las personas que ya tienen tiempo practicando o utilizando las computadoras, tienden a olvidar lo que les llevó introducirse y acumular la gran cantidad de términos y conocimiento; se nos olvida que en alguna ocasión nosotros también empezamos, lo que origina en parte la imagen de superhombres que en ocasiones nos formamos los "computólogos" ó informáticos.

Es conveniente situarnos en el mundo actual, la computación es una necesidad del mundo contemporáneo y si está a nuestra mano, porque no utilizarla. Cabe aclarar un principio básico, "La computadora es un medio, no un fin" es una herramienta que debemos de utilizar, dicha utilización será en algunos casos más a detalle y en otros sólo contemplaremos productos finales de alguna aplicación; en pocas palabras, es necesario hacer a un lado el halo de misterio que rodea a la computación, hay que adentrarnos pausada y ordenadamente en una aventura que pocas veces es reversible; en alguna ocasión en la introducción de un libro se leía, ADVERTENCIA: "El ambiente que genera el mundo de las computadoras tiende a ser extremadamente adictivo; miles de personas se introducen cada día, no hay cura conocida para la dependencia de las computadoras que a menudo resulta de dicha adicción, tenga precaución.

Porque no hacer de esta adicción producto de una necesidad, una aventura pero una aventura provechosa.

Por último, el concepto de necesidad se genera a partir de los adelantos tecnológicos en la época que vivimos; la computación no es del mañana es del hoy, la dificultad es solo inicial y en la mayoría de los casos producto del desconocimiento.

Piensen en un campo en el que todavía no se aplique la computación, si se tardan menos de un minuto en encontrar alguno es posible que :

- 1) Ya tenga nociones de computación.
- 2) Esa necesidad se puede canalizar, ya encontramos una nueva aplicación.
- 3) El concepto actual no sea el adecuado.

Esperamos que al finalizar este curso "INTRODUCCION A LAS COMPUTADORAS PERSONALES" podamos todos contar con los elementos básicos para decidir como concretizar esta necesidad de la manera más productiva.

La mayor parte de los usuarios finales del mundo actual de las computadora no sabían, por ejemplo, más de electrónica de lo que ustedes saben ahora, incluso esto se limita al manejo de un aparato de T.V. o de un equipo de alta fidelidad, no se necesita ningún antecedente especial, sino solo una gran curiosidad y el suficiente empeño para dominar lo necesario del lenguaje computacional y así comprender qué es y para qué podemos utilizar una computadora adicional; a esto es necesario manejar algún vocabulario del idioma inglés y adquirir cierto nivel (dependiendo éste de lo que se pretenda lograr) en la llamada cultura de la computación.

INTRODUCCION A LA COMPUTACION

Cada vez es más frecuente el encontrarse, en nuestra sociedad, con más términos del llamado idioma computacional, conversaciones donde palabras como calculadora, computadora, microcomputadora, computadora personal, byte, bit, microprocesador se emplean frecuentemente. Es una realidad que la mayoría de los términos empleados tienen su origen en el inglés, o en nuestra propia lengua, ya sea uniendo o abreviando palabras.

Por ejemplo, una pequeña computadora, independientemente que cueste alrededor de 3,000 dólares, es diminuta en comparación con los sistemas grandes por eso les llamamos microcomputadoras, una de éstas, pequeña que es fácilmente transportable, que se usa en las empresas y por particulares también se le llama COMPUTADORA PERSONAL (últimamente se usa un nuevo tipo "portable" para la presentación en forma de maletín), pero tiene algo en común con los sistemas mayores, un dispositivo electrónico llamado MICROPROCESADOR, esto es un micro-procesador, un pequeño procesador de información que solo es un circuito integrado o IC que se asemeja a un miriápodo y que también se llaman CHIP el cual consta de una combinación de elementos de circuitos que están integrados (unidos o combinados) en un paquete de miles de transistores con una función en conjunto.

Actualmente las computadoras personales utilizan uno o más microprocesadores, lo mismo que las computadoras de tamaño mediano a las que les llaman MINICOMPUTADORAS, las que se utilizan en empresas y entidades de tamaño medio, supuestamente estos equipos inician donde las microcomputadoras terminan, pero esta división está cada vez menos clara. Existe una categoría más de computadoras que agrupa a las de tipo grande utilizadas en las grandes oficinas almacenando una gran cantidad de información, a éstas que se les denomina MACROCOMPUTADORA ó MAINFRAME.

Como se puede observar, las computadoras se clasifican de acuerdo con tres prefijos bastante comunes, - MICRO, MINI, MACRO-. Generalmente micro significa pequeño, (técnicamente una millonésima de) y mini, para la mayoría de la gente sería una miniatura; macro sería grande, así de fácil es en la mayoría de los casos el vocabulario de la computación.

Al clasificar una computadora de acuerdo con su tamaño, nos dice algo del trabajo que puede resolver, cuanto podrán hacer en un minuto, en un segundo, en un microsegundo (millonésima de segundo) o en un nanosegundo (billonésima de segundo), así se mide el tiempo en la computadora, generalmente las macrocomputadoras son mucho más veloces en ejecución (process time) y de mayores capacidades de almacenamiento de información (memory), esto se debe a que las grandes tienen un hardware y un software más sofisticado) entiéndase más caro y complicado). Ambos términos pueden definirse literalmente.

El hardware es el material rígido, sólido, es la parte física, lo que se puede tocar; las cajas de metal o plástico que contienen a los diferentes dispositivos electromecánicos y electrónicos. El software es la parte lógica: se encuentra en cintas o discos o incluso en un listado, el software es el "ente" que le dice al hardware qué hacer y cómo hacerlo.

Independientemente del tamaño, cualquier computadora, desde la micro a la macro, necesita forzosamente del software (los programas o sistemas) ya que sin su presencia solo tendríamos "fierros" sin organización ni control. Este punto (el software es el común denominador de las computadoras de todos tamaños y formas. Los programas o sistemas dan "vida" a las computadoras y las capacitan para hacer su trabajo, solo el que le diga, quien la programe (¡ nada más !). Al igual que las computadoras el software se clasifica en tres categorías.

El primer grupo de software comprende a los llamados programas de aplicación o de usuario, también se llaman paquetes de usuario final, y son los que mayormente se utilizan, un programa de usuario es un conjunto de instrucciones escritas en un lenguaje que puede entender la computadora y que le dice como hacer una tarea específica, cómo calcular el valor de nuestras incógnitas en una ecuación o actualizar un directorio, una simulación, etc. Estos paquetes solo los ejecutamos (o corremos) nos piden datos de entrada y nos proporcionan datos de salida, están en "paquete" y realmente no nos importa como están constituidos por dentro, siempre y cuando hagan lo que dicen que hacen, este grupo es el que mayor se difunde y anuncia, para su uso no se necesitan casi ningún conocimiento específico pero si una buena noción y la lectura de una manual de usuarios.

otra categoría sería la de software para Desarrollo, la que algunos dividen en Desarrollo específico y Desarrollo General, veamos:

Desarrollo General, este software consta de lenguajes de alto nivel muy parecidos al inglés y a veces al español que permiten escribir programas (en BASIC, FORTRAN, PASCAL, etc.) muchos paquetes están escritos en estos lenguajes, este grupo implica conocer a niveles más profundos el mundo de la computación, por ejemplo Técnicas para Desarrollo de Sistemas, y programas, cursos completos de lenguajes como los ya mencionados y sus aplicaciones (ahora si es importante que tienen dentro los paquetes) este grupo requiere mayor conocimiento pero no tan especializado como el tercer grupo o sea el software de Desarrollo específico como sería diseñar a los programas que permiten que la computadora ejecute programas, programas que permiten funcionar a los paquetes o lenguajes. El Sistema Operativo (O.S.) es un programa que opera, administra y controla cada dispositivo de hardware y ejecuta cada instrucción de los programas de usuario los guía a través de cada parte del hardware a la que necesita llegar, le otorga el tiempo suficiente. Por ejemplo las siglas CP/M se refieren a un sistema operativo (Programa de Control para Microprocesadores); este es uno de los más populares ó DOS (Sistema Operativo para Disco) que es el programa que nos permitirá dialogar con nuestra computadora personal.

Hardware (parte física), Software (parte lógica), hay una tercera palabra de este tipo Firmware que se refiere al software que ya viene programado o incluido en cierto tipo de hardware, es decir es una parte física pero programada. Los programas de tipo firmware dan instrucciones de operación a varias partes de una computadora, trabajan en coordinación con el Sistema Operativo a veces en el caso de las microcomputadoras el mismo O.S. viene ya cargado en hardware, es decir es firmware, puesto que un programa es un conjunto de instrucciones, cada una de estas debe de tener un sentido o un significado para la computadora; de otro modo, poco se podría realizar. La computadora no comprende si no están estas instrucciones en un lenguaje que comprenda, la palabra lenguaje tiene un significado habitual, es un medio de comunicación entre la gente (el usuario) y las computadoras, COBOL, ENSAMBLADOR, PASCAL por mencionar algunos; cada uno de ellos, es un conjunto de órdenes que tienen sentido para las personas. Cada orden es traducida automáticamente (compilada) en grupos de caracteres y símbolos que si tienen sentido para la computadora. Los símbolos y caracteres generados cuando escribimos, por ejemplo en BASIC, le dicen a la computadora que haga algo muy específico (no ambiguo) tal como PRINT (imprimir) o RUN (ejecutar el programa).

Hasta el momento sabemos que todas las partes de una computadora están "unidas" por el software y a veces también por el firmware, pero ¿cuáles son las partes de una computadora?

Casualmente en cualquier computadora hay también tres sistemas básicos de hardware.

El primer grupo de hardware se refiere al mundo exterior, al usuario, es decir a las unidades de entrada que permiten comunicarse con la computadora, una entrada (input) es un hardware que nos permite ingresar información en una computadora, el segundo grupo podría ser entonces un dispositivo (hardware) que permite a la computadora comunicarse con el usuario, un dispositivo de salida (output).

El tercer grupo incluye el "cerebro" de la computadora: la unidad central de procesamiento o CPU que para el caso de las computadoras personales ya mencionábamos se llama microprocesador, el cual realiza cálculos, compara, interpreta, procesa información, los CPU'S no piensan a menos que hayan sido programados para ello. Incluso entonces el "pensamiento" de la CPU está rígidamente estructurado o programado para hacer cosas muy específicas. Por lo tanto, no es posible que un CPU se vuelva o tome decisiones contra su dueño o cualquier otro (generalmente es más probable lo contrario) ya que un CPU es frío e independiente, ignora todo excepto algunas palabras previamente programadas, esto podría provocar un ataque del usuario al equipo, pero nunca viceversa.

En este tercer grupo, los CPU necesitan algo para recordar que hacer, una memoria que les permita actuar. En lenguaje de computadoras, la memoria significa lo mismo que el lenguaje normal, es la capacidad de llamada repetitiva o de recordar, funciona como la memoria humana, incluyendo el problema de tener que encontrar la información que sabemos que está almacenada en ella, esto se llama acceder o acceso a memoria.

La memoria de la computadora aparece en diferentes formas mecánicas y electrónicas, pero podríamos pensar en dos categorías: Temporal y Permanente, el factor que separa a las dos es la electricidad, cuando se desconecta la corriente de una memoria temporal, todo lo almacenado se borra, dura sólo mientras que la corriente está conectada. La memoria permanente, no depende de la presencia de corriente para almacenar información.

Además de esta clasificación, la memoria también se diferencia por el método de acceso o de recuperar (leer) información de ella. cuando obtenemos información, estamos leyendo de ella, cuando introducimos información estamos escribiendo en ella; así una memoria de la cual solo podemos leer es una memoria solo - lectura o ROM (Read Only Memory).

Algunas ROM'S vienen de fábrica con un programa almacenado; otras se pueden programar por el usuario y otras pueden ser borradas y reprogramadas en un equipo especial. las ROM'S que usted puede programar una vez o ROM'S programables se llamas PROM'S y las ROM'S que pueden borrarse electrónicamente y reprogramarse son las EPROM'S, esto es la memoria permanente (a no ser que se borre a propósito). Otro tipo de memoria como si fuera permanente, sería la cinta magnética; una vez que la información está en la cinta o en un disco, permanece en ella a no ser que la cinta o disco se destruyan o borren. este tipo de memoria (cinta o disco) puede ser leída o reproducida y se llama memoria masiva (mass - storage) porque almacena grandes masas de información.

Cuando se trata de recuperar información, es más probable que busquemos ciertos trozos, pequeñas partes de información dispersa aquí y allí, el tiempo ahora es mucho más importante.

Mientras buscamos información específica en la memoria masiva, la computadora está holgazaneando tranquilamente sin hacer nada. Si llevara un minuto o dos encontrar cierta información, alguna persona impaciente lo notaría. Pero un minuto es una eternidad en el tiempo de la computadora, entonces el "tiempo de acceso" es muy importante.

Generalmente las cintas magnéticas tienen un tiempo de acceso grande, ya que son lentas, son mucho más lentas que los discos ya que en una cinta acceder al azar información, significa recorrer desde el principio hasta donde esté lo que buscamos , a esto se llama "memoria secuencial".

La memoria masiva más popular es el disco flexible, floppy disk, diskette, es más fácil y rápido acceder información en un disco que en una cinta, es posible ir directo a cualquier lugar del disco en una fracción de segundos, a este tipo de memoria se le llama memoria de acceso Random aleatorio o al azar, por que el acceso es casi instantáneo independientemente de la posición en el disco a la que se pretenda llegar, RANDOM ACCESS MEMORY O RAM simplemente.

La ROM es permanente y solo podemos leer, en la RAM podemos leer y escribir al azar, algunas Ram son temporales pero su acceso es más rápido y son más baratas que las ROM.

Las computadoras vienen con algún ROM dentro, donde almacenan el firmware y con RAM (medidas en K's: 8K, 16K, 48K, 64K, etc.) y es fácilmente expandible.

Kilobyte (kilo - significa 1,000). En lenguaje de computadora son 1,024 bytes y no 1,000 pero entonces que es un byte?

El software toma los signos y caracteres de nuestro lenguaje y los convierte en una forma que pueda ser tratada por la computadora. Realmente la computadora solo comprende el equivalente eléctrico de si o no, conectado o desconectado, con corriente o uno y sin corriente o un cero. A estos 0's o 1's se les llama bit (binary digit) dígito binario; un bit es el trozo más pequeño de información que puede manejar una computadora. Cuando tecleamos algo en una computadora, generamos electrónicamente una serie de bits (unos y ceros) y se envían al CPU.

Lo que tecleamos realmente son bits que la computadora puede interpretar según una tabla o código específico.

Por lo tanto, un byte es un grupo de bits, ocho específicamente, dos bytes serán 16 bits, 128 bytes son 1,024 bits y 1,024 bytes un kilobyte.

Nótese que todos los números se elevan a una potencia de 2, es decir, 2 multiplicado por sí mismo, entonces 1,024 bytes es igual a 2 multiplicado por sí mismo diez veces, de esto la palabra kilobyte.

Como se puede observar aprender el lenguaje de las computadoras es como aprender francés o español si ya se sabe latín. Si conocemos lo básico de nuestra lengua, solo es cuestión de familiarizarnos con nuevos usos de palabras ya conocidas.

Tendencias actuales, computadoras "amigables" y Computadoras Personales.

A menudo tendemos a calificar las cosas no familiares como no amistosas, por ello es bastante normal que al principio consideremos a las computadoras como algo extraño y poco amigable.

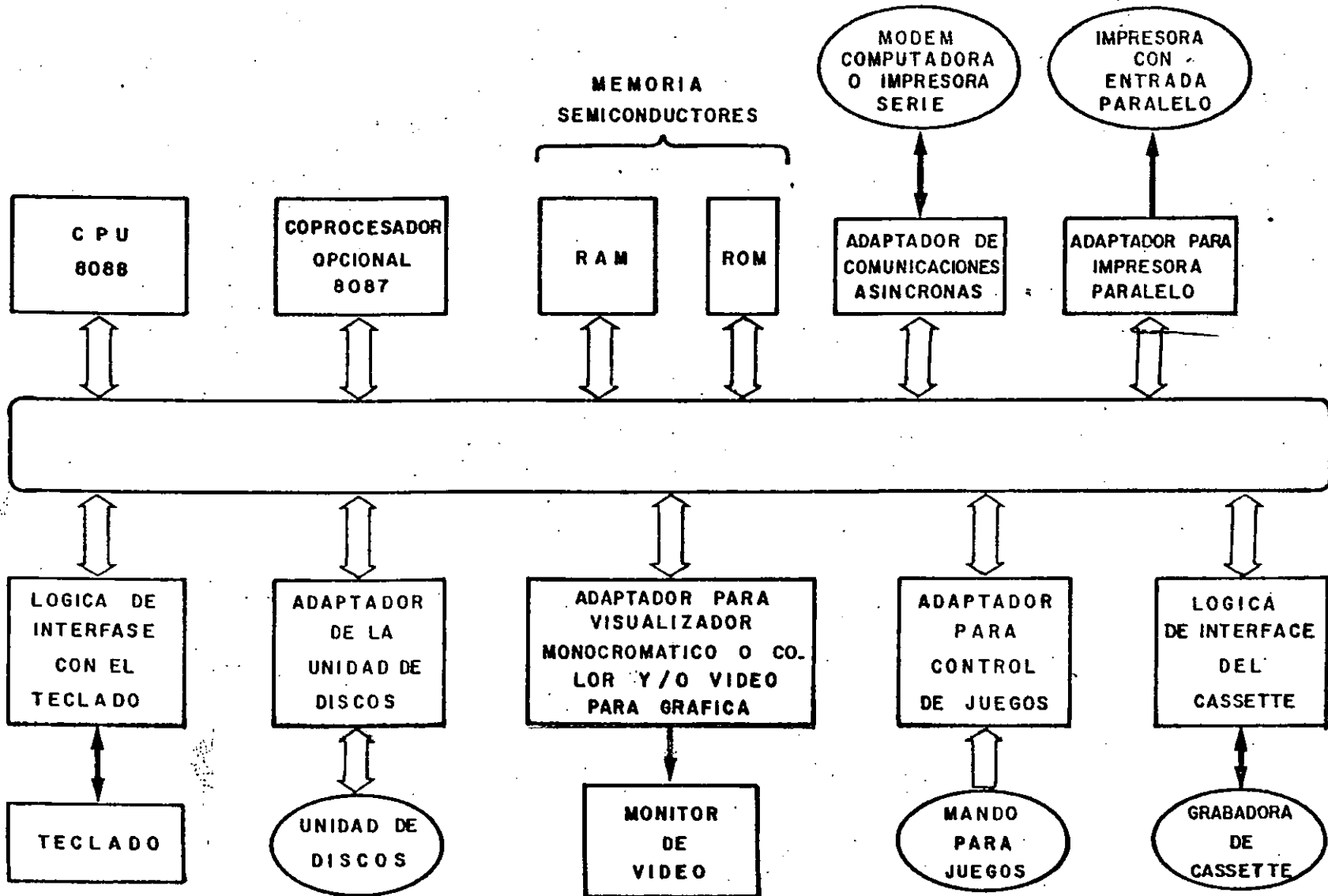
Teniendo en cuenta lo anterior, vale la pena aclarar que en los principios de la computación la preocupación primordial fue desarrollar el hardware necesario, conforme se satisfizo este punto el software empezó a tener un desarrollo importante, ya tenían una computadora, ya podían platicar con ella pero en lenguaje de ella, "lenguaje de máquina" o lenguaje de unos y ceros, el software permitió primero "ensamblar" instrucciones para que la computadora entendiera, después "interpretar" instrucciones tipo mnemonicos, ADD para sumar y su equivalente podría ser 11001111 etc.

Se crean lenguajes, lenguajes de alto nivel o lenguajes amigables pero con un concepto contrario, las tendencias actuales nos dicen que los "friendly language" se acercan lo más posible al usuario y se alejan del lenguaje de máquina ya que existen procedimientos (intérpretes y compiladores) para traducir ese lenguaje amigable, guiado, dirigido, sin complicaciones, de autoinstrucción a aquel que la computadora puede interpretar.

Por otro lado, la tecnología evoluciona de los bulbos a los transistores a los chips a las tecnologías LSI (large scale integrated) y VLST (vey large scale integrated) en aspectos de hardware, lo que ocasiona que en un IC se concentren miles o millones de dispositivos, eso origina la disminución de tamaños y la evolución desde las macrocomputadoras, minicomputadora, microcomputadora y finalmente computadoras personales, de escritorio, portátiles, etc.

La conclusión, la herramienta con mayor facilidad para el usuario, tanto en hardware (tamaño personal) como en software (amigable y totalmente accesible al usuario final) como en COSTO y difusión, esto es la orientación actual.

ORGANIZACION DE COMPUTADORAS

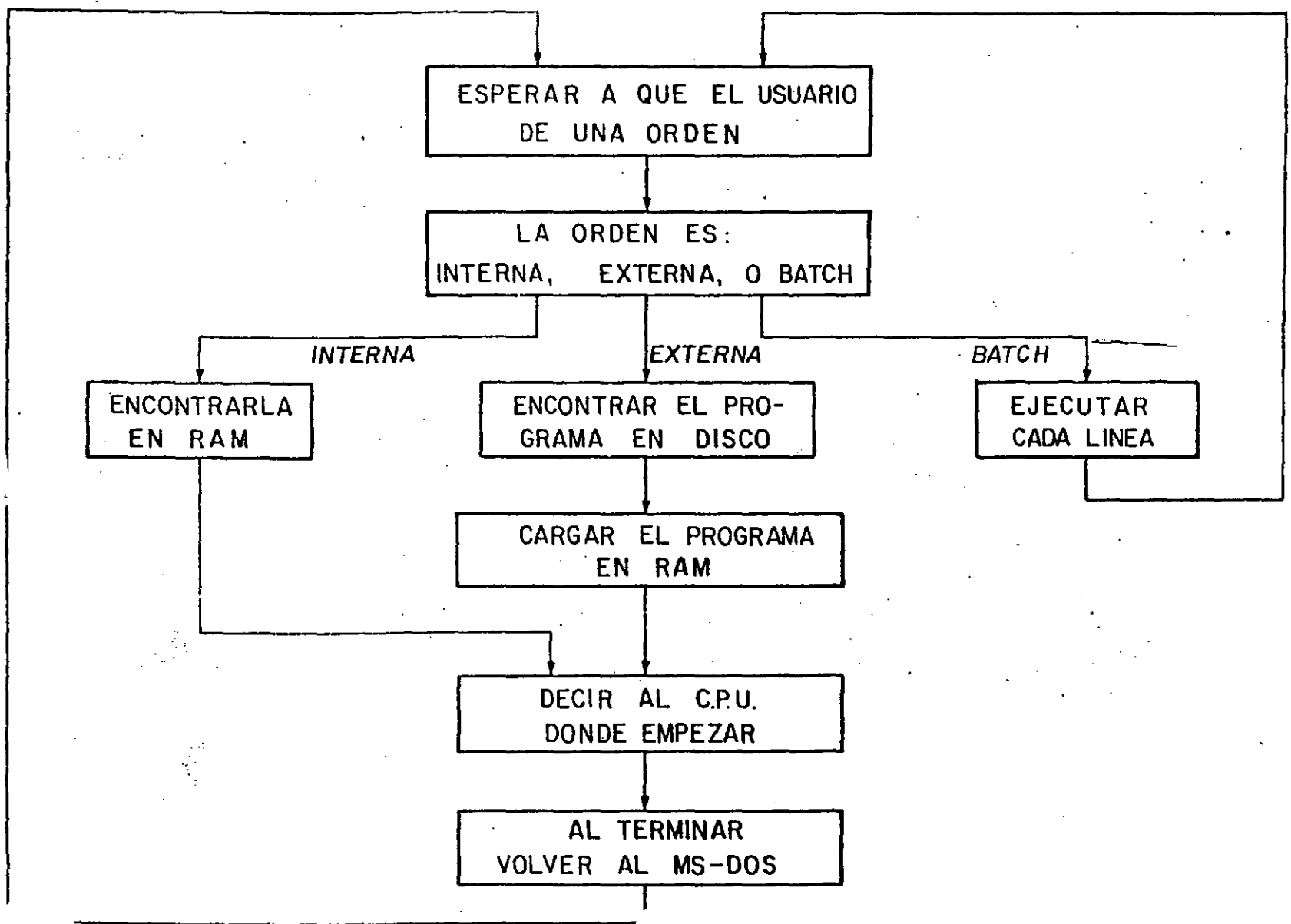


000013

MS-DOS Y COMO EJECUTA PROGRAMAS

0000014

10000000



LA COMPUTADORA Y SU MEDIO AMBIENTE

U
S
U
A
R
I
O

ENTRADAS
TECLADOS, ETC.



"CEREBRO"
C P U
MICROPROCE-
SADOR



SALIDAS
PANTALLAS
IMPRESORAS

U
S
U
A
R
I
O

M E M O R I A

ROM

RAM

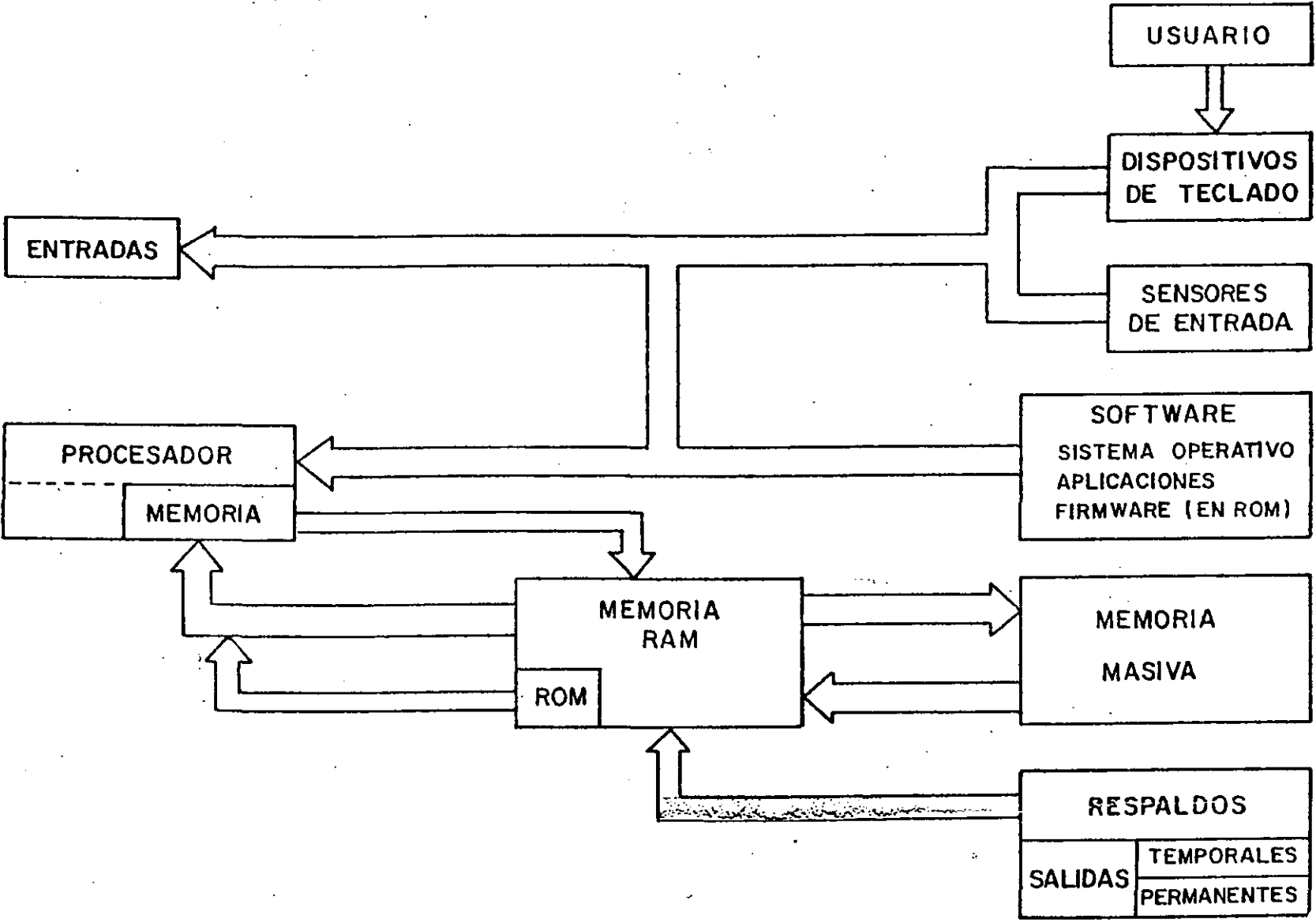
MEMORIA
MASIVA

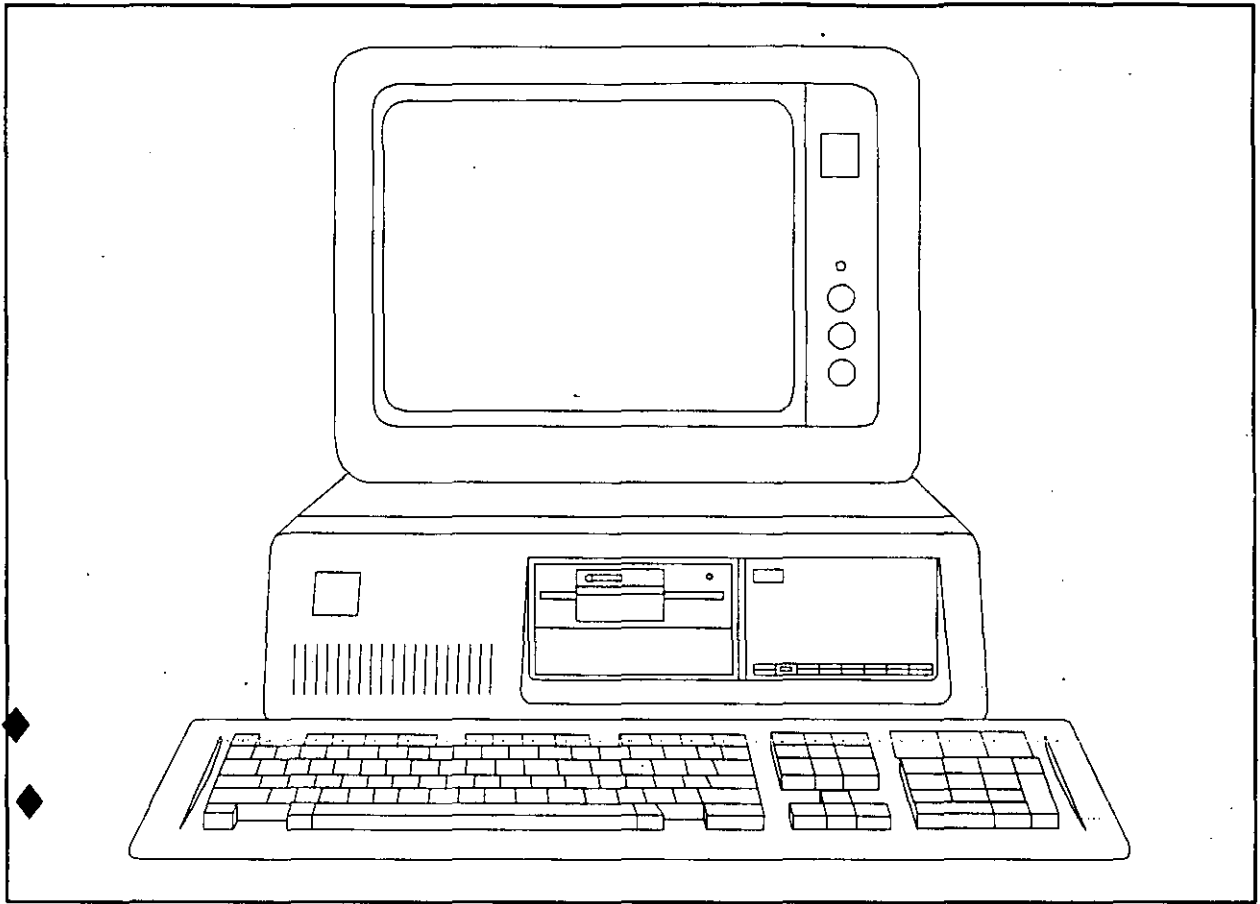
DISCOS, CINTAS
CARTUCHO. ETC.

ADMINISTRADORES DEL
EQUIPO, EQUIPOS, REDES
CENTRO DE COMPUTO

0000015

PARTES BASICAS DE UNA COMPUTADORA PERSONAL





INTRODUCCION AL MANEJO DE INFORMACION

CODIGOS, DATOS Y ARCHIVOS

INTRODUCCION:

AL HABLAR DE COMPUTADORAS PUEDEN VENIR A NUESTRA MENTE MUCHOS CONCEPTOS. EL PUBLICO EN GENERAL QUIZA ASOCIARA A LAS COMPUTADORAS CON LOS TERMINOS AMBIGUOS "CEREBRO ELECTRONICO", QUIZA OTROS MAS INFORMADOS PENSARAN EN MAQUINAS PARA REALIZAR CALCULOS MUY COMPLEJOS Y ALGUNOS MAS LAS IDENTIFICARAN COMO MAQUINAS PARA ALMACENAR Y RECUPERAR INFORMACION.

CUALQUIERA QUE FUERA LA IDEA QUE SE TENGA DE UNA COMPUTADORA, SI ESTA ES CORRECTA TODAS COINCIDIRAN EN ACEPTAR QUE SE LE PUEDE DEFINIR COMO UNA MAQUINA PARA MANEJAR INFORMACION. DICHO SEA DE PASO, EL TERMINO QUE SE UTILIZA EN EL CODIGO ADUANERO PARA DESCRIBIR ESTA MERCANCIA ES PRECISAMENTE: MAQUINAS AUTOMATICAS PARA EL TRATAMIENTO DE INFORMACION

POR OTRO LADO, CUANDO UNA PERSONA INICIA SUS ESTUDIOS SOBRE COMPUTADORAS, INVARIABLEMENTE APARECEN LOS TERMINOS BIT, BYTE, NIBBLE, PALABRA, ETC, Y SE LE HACE HINCAPIE EN QUE DENTRO DE LA COMPUTADORA TODO SE RESUME A DOS ESTADOS, EL SI Y EL NO, O EL PRENDIDO Y EL APAGADO, EL VERDADERO O EL FALSO. EN OTRAS PALABRAS, DENTRO DE UNA COMPUTADORA SOLO HAY BITS.

EN UN PRINCIPIO TODO LO ANTERIOR PARECE TOTALMENTE LOGICO, SIN EMBARGO, AL TRATAR DE ASOCIAR LOS CONCEPTOS DE LOS PRIMEROS PARRAFOS CON LOS DE ESTE ULTIMO SE GENERAN MUCHAS DUDAS. COMO PUEDE SER QUE UNA COMPUTADORA PUEDA MANEJAR CALCULOS COMPLEJOS, LOS NOMBRES DE MILES DE PERSONAS, LOS SALDOS DE OTROS TANTOS CUENTAHABIENTES Y AUN DIRIGIR EL TRAZO DE UN BOLIGRAFO EN UN GRAFICADOR, SOLO UTILIZANDO ELEMENTOS BIESTABLES (BITS)?.

LA RESPUESTA A ESTA INTERROGANTE PUEDE DARSE CUANDO SE COMPRENDEN ALGUNOS CONCEPTOS FUNDAMENTALES QUE SE CUBREN A CONTINUACION.

USO DE CODIGOS PARA CODIFICAR LAS INSTRUCCIONES DEL LENGUAJE DE MAQUINA:

CONCEPTO DE CODIGOS PARA LAS INSTRUCCIONES EN LENGUAJE DE MAQUINA.

OPERACION DE UNA COMPUTADORA: CICLO DE EJECUCION DE UNA INSTRUCCION (FETCH, DECODE, EXECUTE).

SECUENCIA DE EJECUCION E INSTRUCTION POINTER.

USO DE LOS CODIGOS PARA LA REPRESENTACION DE LETRAS Y CARACTERES:

CODIGO ASCII PARA LETRAS, NUMEROS Y CARACTERES DESPLEGABLES

CODIGO ASCII PARA CARACTERES DE CONTROL, CARACTERES DE GRAFICAS Y CARACTERES ESPECIALES (LETRAS EN OTROS IDIOMAS, SIMBOLOS MATEMATICOS, ETC.)

BITS DE PARIDAD, CODIGO EBCDIC, OTROS CODIGOS

USO DE LOS CODIGOS PARA LA REPRESENTACION DE NUMEROS:

REPRESENTACION DE NUMEROS ENTEROS

REPRESENTACION DE NUMEROS DE PUNTO FLOTANTE

NUMEROS DE DOBLE PRECISION (ENTEROS Y DE PUNTO FLOTANTE)

OTROS USOS DE LOS CODIGOS BINARIOS

CONFIGURACION DE HARDWARE CON MICRO SWITCHES

CODIFICACION DE PALABRAS DE USO FRECUENTE (INTERPRETES BASIC, TEORIA DE CODIFICACION DE HUFFMAN)

UNA VEZ COMPRENDIDA LA IDEA DEL USO DE LOS CODIGOS Y SU USO EN MEMORIA, INMEDIATAMENTE SE DESPRENDE LA IDEA DE DATOS DENTRO DE UNA COMPUTADORA. LOS DATOS DENTRO DE LA COMPUTADORA NO SON MAS QUE SECUENCIAS DE BITS "VISTAS" A TRAVES DE CRITERIOS PREESTABLECIDOS DE DECODIFICACION. SIENDO ESTRICTOS, DATA UNA SECUENCIA DE BITS CUALQUIERA, NO ES POSIBLE SABER SI ESTA REPRESENTA UNA INSTRUCCION EN LENGUAJE D MAQUINA O UN NOMBRE DE UNA PERSONA O UN NUMERO, O UNA COMBINACION DE ESTOS, ETC.

YA QUE SE HA LOGRADO REPRESENTAR SOLO CON BITS LOS DIFERENTES DATOS, SE ANTOJA LA IDEA DE ASOCIAR GRUPOS DE DATOS PARA FORMAR LO QUE SE CONOCE CON EL NOMBRE DE ARCHIVOS. UN ARCHIVO ES UNA COLECCION DE DATOS, NORMALEMENTE RELACIONADOS ENTRE SI, QUE ESTAN ALMACENADOS COMO UN CONJUNTO.

EL MEDIO MAS COMUN PARA ALMACENAR ARCHIVOS EN MICROCOMPUTADORAS ES EL DISKETTE, PERO EN MAQUINAS MAS GRANDES SE UTILIZAN CON FRECUENCIA LOS DISCOS DUROS, LAS CINTAS MAGNETICAS, CARACTERES MAGENTICOS, CARACTERES OPTICOS, Y BANDAS MAGNETICAS . TAMBIEN ES POSIBLE ENCONTRAR ARCHIVOS "ALMACENADOS" EN LISTADOS, CINTAS DE PAPEL PERFORADA, TARJETAS PERFORADAS, BARRAS OPTICAS Y OTROS MEDIOS DE SOPORTE.

APLICANDO LA MISMA IDEA MENCIONADA PARA LOS DATOS, UN ARCHIVO EN SI NO TIENE NINGUN SIGNIFICADO ESPECIFICO EXCLUSIVAMENTE POR SU CONTENIDO, SINO QUE REQUIERE DE UN CRITERIO PARA SER "VISTO" E INTERPRETADO.

DE ENTRE LOS ARCHIVOS MAS COMUNES SE ENCUENTRAN LOS ARCHIVOS DE TEXTO, LOS CUALES SON VISTOS A TRAVES DE UN DECODIFICADOR ASCII, Y LOS ARCHIVOS DE INSTRUCCIONES DE MAQUINA (TAMBIEN LLAMADOS ARCHIVOS EJECUTABLES), LOS CUALES REQUIEREN SER VISTOS A TRAVEZ DE UN DECODIFICADOR DE INSTRUCCIONES DE MAQUINA, O SEA A TRAVEZ DE UNA UNIDADDE CONTROL O PROCESADOS. EXISTEAN ADEMAS OTROS TIPOS DE ARCHIVOS, COMO LO ARHIVOS DE DATOS NUMERICOS CODIFICADOS, O LOS ARCHIVOS UTILIZADOS POR LA MAYORIA DE LOS INTERPRESTES BASIC, O BIEN CUALQUIER COMBINACION DE DATOS.

PARA DISTNGUIR LOS DIFERENTESARCHIVOS ALMACENADOS EN UN MEDIO DE SOPORTE SE ACOSTUMBRA DARLES UN NOMBRE, LO QUE CONSTITUYE EL NOMBRE DEL ARCHIVO. PARA DISTINGUIR ENTRE LOS DIFERENTES TIPOS DE

10
ARCHIVOS SE ACOSTUMBRA ASOCIAR UNA EXTENSION AL NOMBRE DEL ARCHIVO. ASI, ES COMUN ENCONTRARLOS TIPOS DE ARCHIVOS EXE, COM, PAS, PAS, COB, FOR, BAT, DAT, TXT, DOC, SYS, DBF, WKS, INC, ASM, HLP, Y EN GENERAL CUALQUIER OTRO QUE DESEE ASIGNARSE A UN ARCHIVO EN PARTICULAR.

CONVIENE CONCLUIR COMENTANDO ALGUNOS ASPECTOS GENERALES SOBRE EL SOFTWARE EL HARDWARE Y SU INTERRELACION.

DEFINICION DE SOFTWARE Y HARDWARE

COPMO INTERACTUAN EL SOFTWARE Y EL HARDWARE DESDE EL ENCENDIDO DE LA MAQUINA. EL FIRMWARE.

LOS TIPOS DE SOFTWARE MAS IMPORTANTES: SISTEMAS OPERATIVOS Y SUS UTILERIAS , INTERPRETES, COMPILADORES, PAQUETES, PROGRAMAS ESCRITOS POR EL USUARIO.

IA FUNCION DE CADA UNO DE ESTOS TIPOS DE SOFTWARE PARA LA SOLUCION DE LOS DIFERENTES TIPOS DE PROBLEMAS.

ARCHIVO

ART. 1	ART. 2	ART. 3	ART. 4	ART. 5	ART. 6	
-----------	-----------	-----------	-----------	-----------	-----------	--

REGISTRO

CODIGO	DESCRIPCION	UNIDAD	EXISTENCIA	MAXIMO	MINIMO
--------	-------------	--------	------------	--------	--------

CAMPO

L	I	J	A		G	R	A	N	A	T	E	
---	---	---	---	--	---	---	---	---	---	---	---	--

CARACTER

R

BYTE

I	O	I	I	I	O	O	I
---	---	---	---	---	---	---	---

BIT

I

000023

Tabla B-1. Códigos ASCII para el PC

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
8	08	BS	□
9	09	HT	Tabulador
10	0A	LF	Avance de línea
11	0B	VT	Cursor a reposo
12	0C	FF	Avance de página
13	0D	CR	Introducir
14	0E	SO	♪
15	0F	SI	⚙
16	10	DLE	▶
17	11	DC1	◀
18	12	DC2	↕
19	13	DC3	!!
20	14	DC4	π
21	15	NAK	§
22	16	SYN	—
23	17	ETB	⊥
24	18	CAN	↑
25	19	EM	↓
26	1A	SUB	→
27	1B	ESC	←
28	1C	FS	C. hacia la dcha.
29	1D	GS	C. hacia la izq.
30	1E	RS	Cursor hacia arriba
31	1F	US	Cursor hacia abajo
32	20		Espacio
33	21		!
34	22		-

14

Tabla B-1. (Continuación)

Valor decimal	Valor hexadecimal	Carácter de control	Carácter
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

14

Tabla B-1. (Continuación)

Valor decimal	Valor hexadecimal	Carácter de control	Carácter
70	46		F
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		a
97	61		b
98	62		c
99	63		d
100	64		e
101	65		f
102	66		g
103	67		h
104	68		

Tabla B-1. (Continuación)

Valor decimal	Valor hexadecimal	Carácter de control	Carácter
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
123	7B		{
124	7C		
125	7D		}
126	7E		~
127	7F		
128	80		¡
129	81		¢
130	82		£
131	83		¤
132	84		¥
133	85		¦
134	86		§
135	87		¨
136	88		©
137	89		ª
138	8A		«
139	8B		»

000024

13-

Tabla B-1. (Continuación)

Valor decimal	Valor hexadecimal	Carácter de control	Carácter
140	8C		ì
141	8D		í
142	8E		Ā
143	8F		Ă
144	90		Ĕ
145	91		ı
146	92		Æ
147	93		ø
148	94		õ
149	95		ö
150	96		û
151	97		ü
152	98		ÿ
153	99		Œ
154	9A		Ÿ
155	9B		Œ
156	9C		Ÿ
157	9D		Ÿ
158	9E		Ÿ
159	9F		Ÿ
160	A0		á
161	A1		â
162	A2		ã
163	A3		ä
164	A4		å
165	A5		Ä
166	A6		Å
167	A7		æ
168	A8		ç
169	A9		¸
170	AA		¸
171	AB		¼
172	AC		½
173	AD		¾
174	AE		«

Tabla B-1. (Continuación)

Valor decimal	Valor hexadecimal	Carácter de control	Carácter
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		

000025

(12)

Tabla B-1. (Continuación)

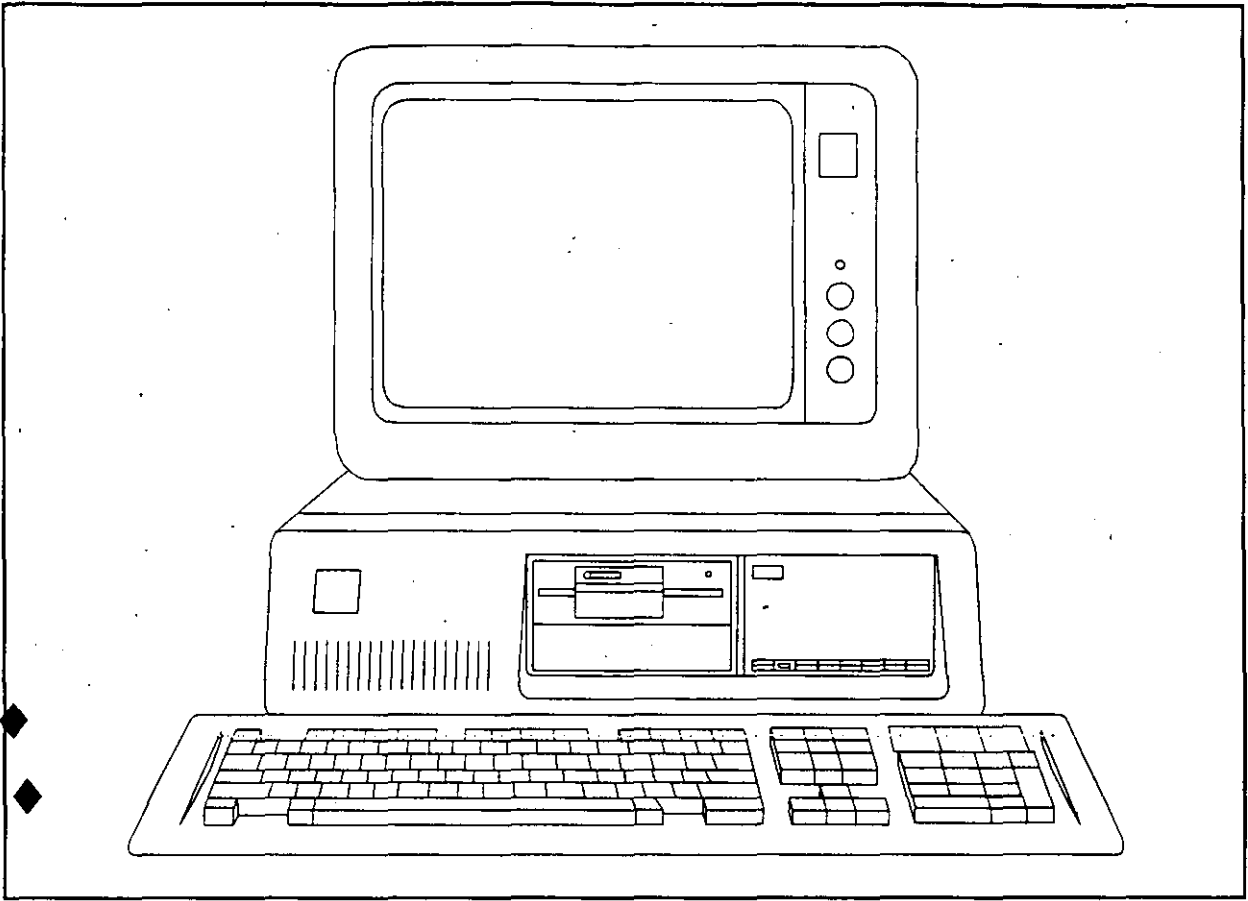
Valor decimal	Valor hexadecimal	Carácter de control	Carácter
210	D2		Ɔ
211	D3		Ɔ
212	D4		Ɔ
213	D5		Ɔ
214	D6		Ɔ
215	D7		Ɔ
216	D8		Ɔ
217	D9		Ɔ
218	DA		Ɔ
219	DB		Ɔ
220	DC		Ɔ
221	DD		Ɔ
222	DE		Ɔ
223	DF		Ɔ
224	E0		Ɔ
225	E1		Ɔ
226	E2		Ɔ
227	E3		Ɔ
228	E4		Ɔ
229	E5		Ɔ
230	E6		Ɔ
231	E7		Ɔ
232	E8		Ɔ
233	E9		Ɔ
234	EA		Ɔ
235	EB		Ɔ
236	EC		Ɔ
237	ED		Ɔ
238	EE		Ɔ
239	EF		Ɔ
240	F0		Ɔ
241	F1		Ɔ
242	F2		Ɔ
243	F3		Ɔ
244	F4		Ɔ

Tabla B-1. (Continuación)

Valor decimal	Valor hexadecimal	Carácter de control	Carácter
245	F5		Ɔ
246	F6		Ɔ
247	F7		Ɔ
248	F8		Ɔ
249	F9		Ɔ
250	FA		Ɔ
251	FB		Ɔ
252	FC		Ɔ
253	FD		Ɔ
254	FE		Ɔ
255	FF		Ɔ

000026

-11-



**¿ QUE SON Y POR QUE UTILIZAR
LAS COMPUTADORAS PERSONALES ?**

PRACTICANDO CON NUESTRA COMPUTADORA

INTRODUCCION

Hace apenas 30 años se inició la era de la computación, pero ya ha ejercido un profundo efecto sobre nuestras vidas. Y así las computadoras se han convertido en herramienta primordial en la oficina, en la fábrica, etc. En los últimos tres o cuatro años han penetrado también en los hogares, a medida que la gente ha ido comprando multitud de juegos por computadora y de computadoras personales. Las computadoras son tan comunes en la actualidad, que difícilmente transcurre un día sin que tengamos contacto con ellas.

Pese al auge que estas máquinas tienen en el mundo moderno, la mayoría de las personas saben poco sobre ellas. Las consideran una especie de "Cerebro Electrónico"; no saben cómo funcionan, cómo se usan ni cuánto pueden simplificar muchas de las tareas cotidianas. Ello no obedece a falta de interés. Casi todos se dan cuenta de que las computadoras ya forman parte de nuestra vida y cultura.

QUE ES LA COMPUTADORA PERSONAL

En los albores de la computación (el período comprendido entre la década de 1940 y la de 1950), la computadora típica era una enorme masa de partes electrónicas que ocupaban varios cuartos. Por esa época se necesitaba reforzar el piso de la pieza donde se hallaba y se instalaba aire acondicionado especial para que el equipo funcionara satisfactoriamente. Más aún, el precio de esas máquinas ascendía a varios millones de dólares. Con el tiempo el costo disminuyó de modo considerable y, gracias a la microminiaturización, el tamaño de ellas se ha reducido a un ritmo mayor que su precio.

A fines del decenio de 1970 hicieron su aparición en el mercado las primeras computadoras "personales". Esas máquinas tuvieron excelente acogida entre el público y han despertado el interés de todo mundo en todos los ámbitos. No exageramos al decir que está en marcha una verdadera revolución, pues miles de personas están aprendiendo a incorporarlas a su existencia.

Una computadora personal posee suficiente capacidad para realizar una serie de tareas, sería imposible proporcionar una lista completa de las múltiples aplicaciones de la computadora personal. A continuación nos limitamos a transcribir un repertorio de algunas de sus posibles aplicaciones :

Para el hombre de negocios

Contabilidad
Archivos
Trabajos de oficina
Inventario
Administración de efectivo
Nóminas
Gráficas y preparación de diagramas
Procesamiento de palabras
Análisis de datos
Redes

Para el Hogar

Archivos
Administración del presupuesto familiar
Análisis de inversiones
Correspondencia
Conservación de energía
Seguridad en el hogar
Recuperación de información
Preparación de la declaración de impuestos

Para el estudiante

Cultura general en computación
Preparación de exámenes escolares
Análisis de experimentos
Preparación de gráficas y diagramas
Programas de proyectos

Para el profesional

Facturación
Análisis de datos
Generación de informes
Correspondencia
Acceso a los datos sobre la bolsa de valores
Cálculos científicos y de ingeniería

Para recreación
Juegos por computadora
Gráficas por computadora
Arte por computadora

QUE ES UNA COMPUTADORA

La parte fundamental de toda computadora es la unidad central de proceso (CPU), que cumple las órdenes que le damos. Entre otras cosas, realiza operaciones aritméticas y toma decisiones lógicas. En esencia, es el "Cerebro" de la computadora. La memoria le permite a esta última "Recordar" números, palabras y párrafos, lo mismo que la lista de comandos que le comunicamos. La unidad de entrada nos permite enviar información a la computadora; La unidad de salida le permite a ella enviarnosla a nosotros.

En una computadora personal la unidad central de proceso está alojada en un diminuto chip (pastilla) electrónico, denominado microprocesador. (En las computadoras utilizadas en este curso el procesador es un 8088, pero comercialmente existen procesadores de Intel, Motorola etc.)

El principal dispositivo de entrada de la computadora personal es el teclado. Sus características se describirán posteriormente. Por ahora pensaremos que se trata de un teclado como el de las máquinas de escribir. (Al teclear los símbolos, vas introduciéndolos en la computadora).

Estos equipos también cuentan con varios dispositivos de salida, siendo el más importante el Monitor de Video. También se puede usar una impresora para obtener una salida en papel.

La computadora personal cuenta con los siguientes tipos de memoria: ROM, RAM, Casette, Diskette y Disco duro. Cada uno tiene sus ventajas pero también sus limitaciones.

ROM

Las siglas ROM (Read Only Memory) significan memoria sólo de lectura, es decir, la computadora puede leer esta memoria pero no puede escribir nada en ella. La ROM está reservada a ciertos programas muy importantes que se necesitan para que opere la máquina. Esos programas los incorpora el fabricante en ROM y no se pueden cambiar.

RAM

RAM (Random Access Memory) es una abreviatura que significa memoria de acceso aleatorio; en ella podemos leer y escribir. Si se pulsa caracteres de tipo en el teclado, se guardan en RAM; lo mismo sucede con los cálculos, en espera de ser utilizados. Esta memoria posee un rasgo de importancia que no se debe olvidar : Si se apaga la computadora, RAM se borra.

En consecuencia, RAM no debe emplearse para almacenar datos en forma permanente. Sin embargo, por su gran velocidad se usa como el depósito principal de trabajo. (Se tarda cerca de una millonésima de segundo en almacenar información en ella o en recuperarla)

El tamaño de memoria de acceso aleatorio se mide en bytes. en esencia, un byte es un carácter individual (por ejemplo "A" o "!"). ustedes escucharán a menudo frases como ésta : "La PC/AT viene con 1 Mb. en RAM", lo cual indica que su computadora tiene 1024 Kbytes en memoria RAM.

UNIDAD DE CARTUCHO

Se trata de un dispositivo el cual nos permite salvar o guardar información en cartuchos.

UNIDAD DE DISKETTE

Esta unidad registra información en discos flexibles, y cada uno puede contener miles de caracteres. Posteriormente se vera las características de estos discos.

DISCO DURO

Con este tipo de disco se consigue el acceso mas rápido a los datos, pudiendo almacenar allí millones de caracteres (esto depende de la capacidad del disco : 20 MB, 30 Mb, 40 Mb, 80 Mb, 100 Mb, 300 Mb etc).

ASPECTOS PRINCIPALES DE LA COMPUTADORA PERSONAL

Antes de encender la computadora, vamos a familiarizarnos con las partes que consta.

Una Computadora Personal típica consta de los componentes siguientes : Unidad Central de Proceso, Teclado y Monitor.

- UNIDAD CENTRAL DE PROCESO

las computadoras utilizadas en este curso son sistemas Printaform, la cual es compatible con IBM. Este equipo emplea un procesador Intel 8088 en Bus de 8 bits y 16 bits interno. El equipo puede tener hasta 640 Kb. en memoria RAM. (Para poder generalizar se tratará de dar los componentes de una PC Tipo)

Como se menciona anteriormente la Unidad Central es donde reside el "Cerebro" de la computadora, ya que se realizan cálculos, comparaciones, interpretaciones y procesamiento de información.

En el gabinete podemos distinguir los siguientes componentes :

En la parte Frontal de una computadora se puede observar :

- Unidad de diskette o drive "A", con capacidad de 360 Kb ó 1.2 Mb para discos de 5¼" y 720 Kb ó 1.44 Mb para discos de 3½".
- Area para instalar unidad de diskette o unidad de cartucho.
- Indicadores de funcionamiento (se encienden cuando el drive o disco duro esta en funcionamiento).
- Pulsador de "RESET" del hardware.
- Interruptor de encendido.
- Orificios de entrada de aire.

En la parte Posterior :

- Rejilla de ventilación.
- Ranuras de interfaz para tarjetas de expansión.
- Conector de salida de video.
- Conector de interfaz Puerto Paralelo
- Conector de interfaz Puerto Serie.
- Conector de interfaz Teclado.
- Conector de entrada de corriente.

- TECLADO

El teclado consta de las siguientes partes :

- Conector de la interfaz del teclado.
- Interfaz del teclado enrollado en espiral.
- Teclas numéricas.
- Teclas de control del cursor.
- Teclas de función.
- Teclas alfanuméricas.

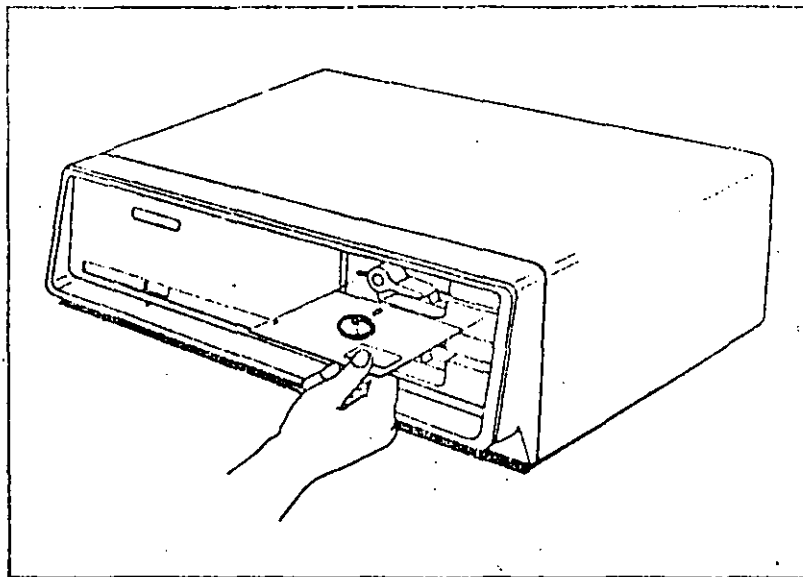
- MONITOR

Las partes que lo conforman son las siguientes :

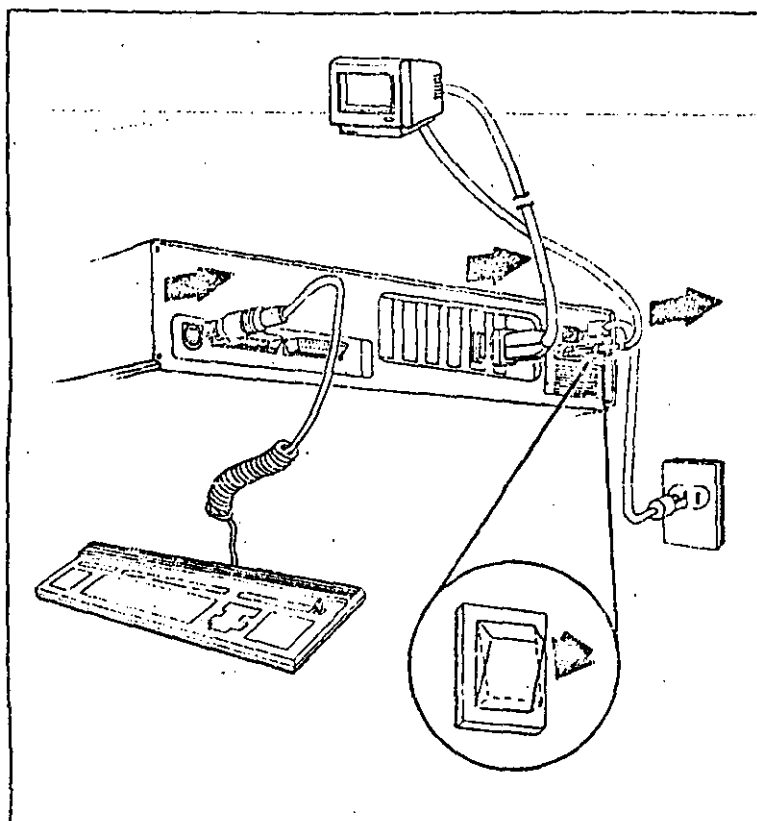
- Pantalla donde se visualiza la información.
- Interfaz de alimentación del monitor.
- Interfaz de señal de video.
- Control de contraste.
- Control de brillo.
- Base del monitor.

COMO COLOCAR EL DISKETTE

- TOMANDO DE LA ETIQUETA EL DISKETTE SAQUELO DE SU FUNDA CON CUIDADO DE NO DOBLARLO Y NO TOCAR LA SUPERFICIE BRILLANTE DEL MISMO, LA ETIQUETA DEBERÁ ESTAR HACIA ARRIBA Y LA MUESCA DE PROTECCIÓN DEL LADO IZQUIERDO.
- LEVANTE LA PUERTA DEL IMPULSOR DE DISCOS
- INTRODUZCA EL DISKETTE HASTA EL TOPE
- CIERRE LA PUERTA, BAJÁNDOLA CON PRESIÓN HASTA QUE QUEDE SUJETA POR EL MECANISMO.
- PARA SACAR EL DISKETTE, BASTARÁ GIRAR LA PUERTA Y ESTE SERÁ LANZADO HACIA AFUERA DE LA UNIDAD.

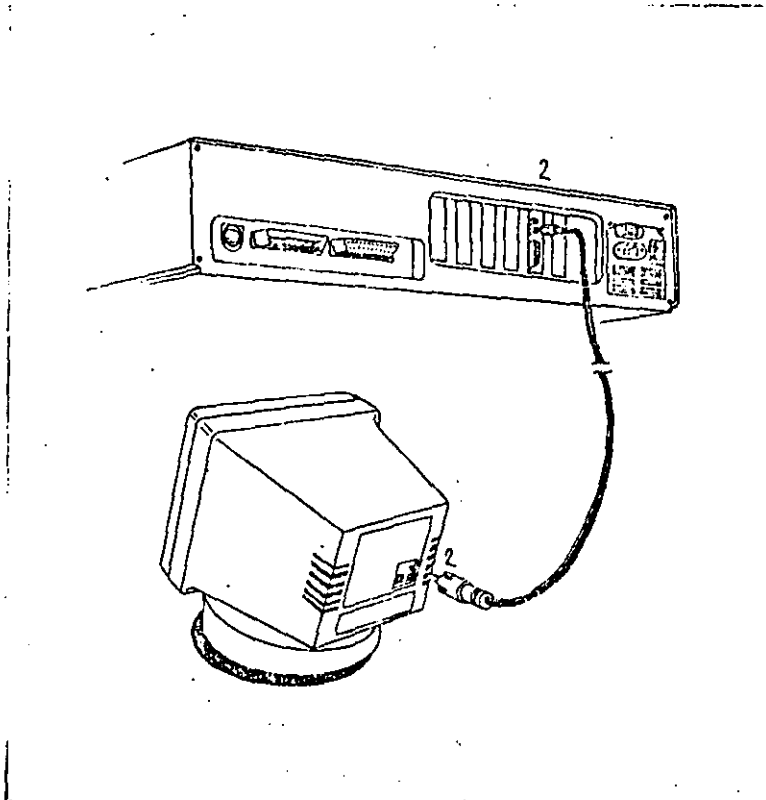
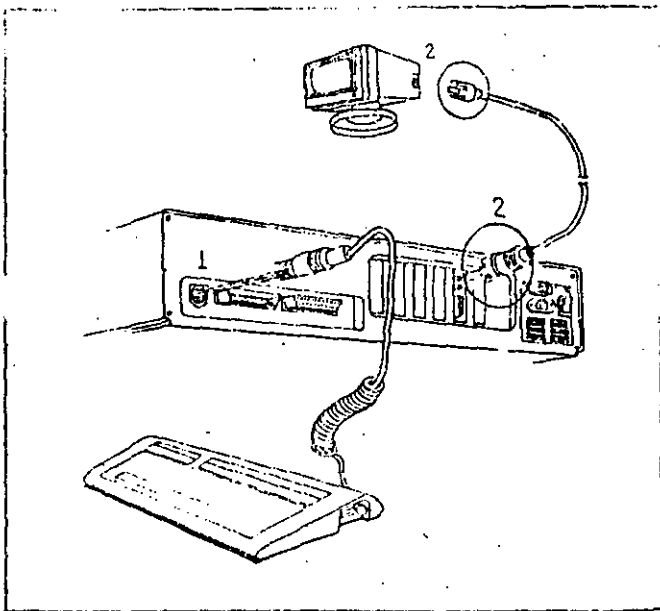
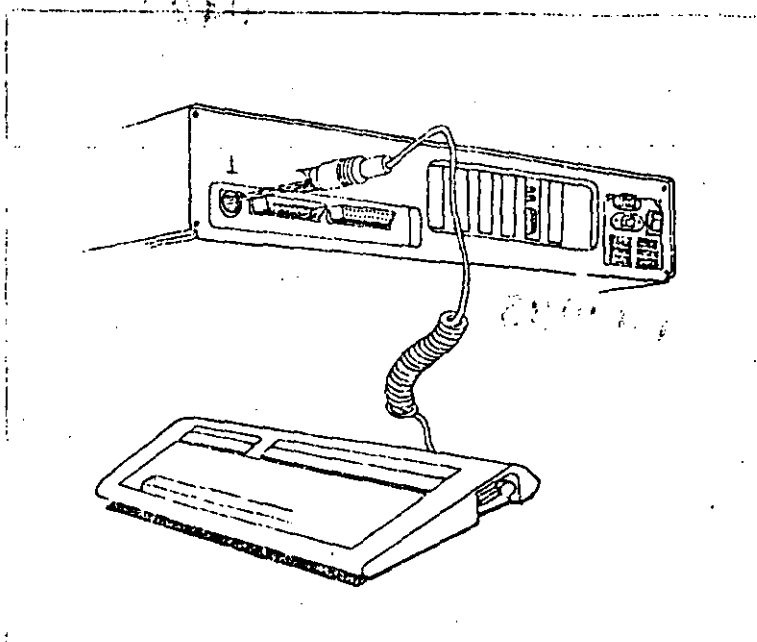


CONEXION

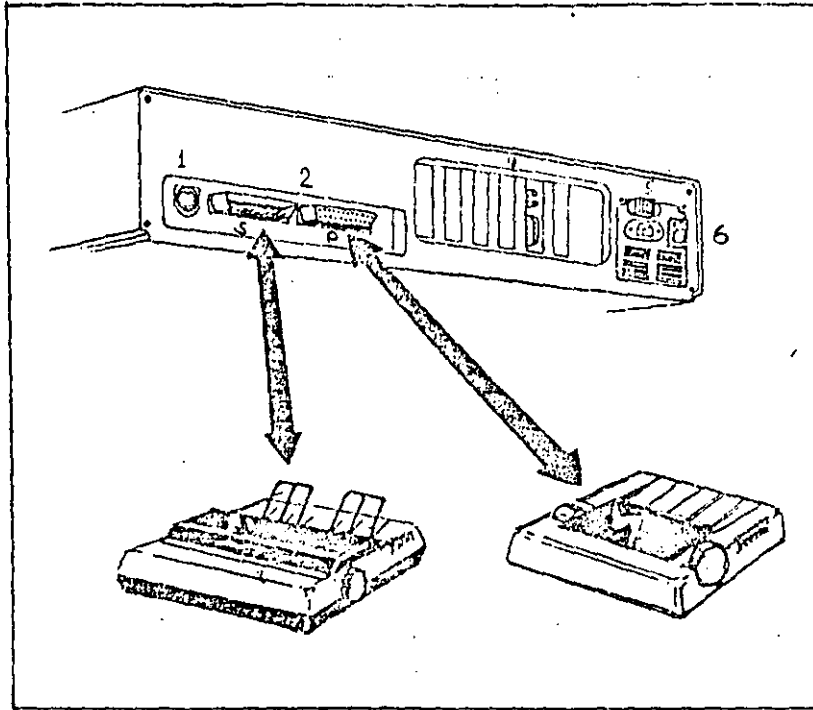


DESCONEXION

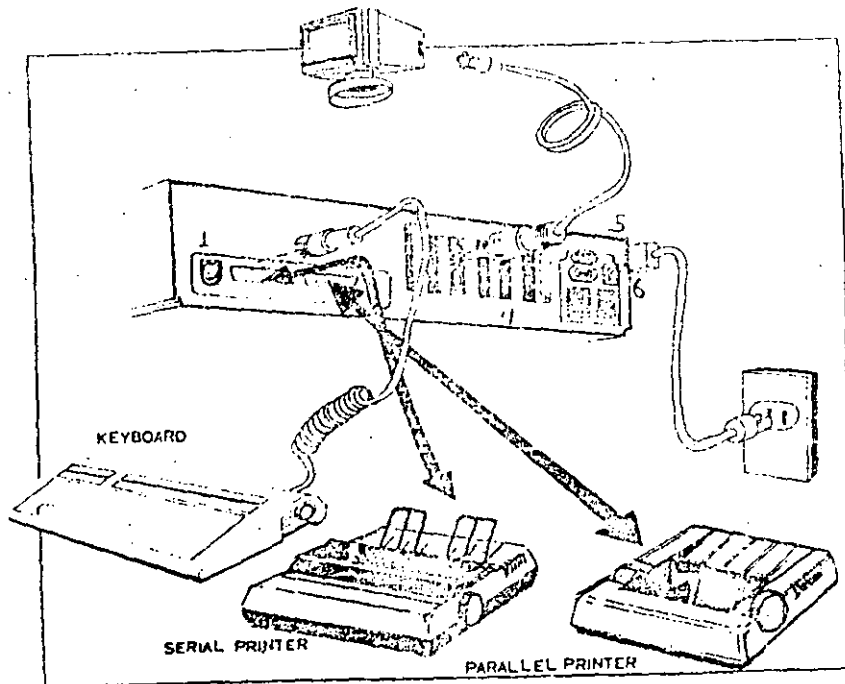
CONEXION



CONEXION.



GABINETE CENTRAL INTERRUPTORES Y CONECTORES



ENCENDIDO Y APAGADO

Para controlar el flujo de información necesitamos de un programa denominado sistema operativo (MS-DOS), el cual coordina las actividades que se llevan a cabo dentro de la computadora.

ENCENDIDO

- Coloque el diskette que contiene el sistema operativo en la unidad "A".
- Encienda el monitor.
- Encienda el interruptor de la computadora.
- Se efectuará pruebas de diagnóstico para comprobar que todos sus componentes estén en buenas condiciones de funcionamiento. Al terminar las pruebas desplegará:

Enter today's date (mm-dd-yy) :

- Se tecléa la fecha (según el formato mes:día:año) se oprime ENTER.
- Se desplegará lo siguiente :

Enter correct time (mm:hh:ss) :

- Se escribe la hora correcta (según formato hora:minuto:segundo) se oprime ENTER.
- se observará A > _ El cual es el indicador de DOS y nos señala que el sistema operativo ya está cargado y listo para recibir comandos.

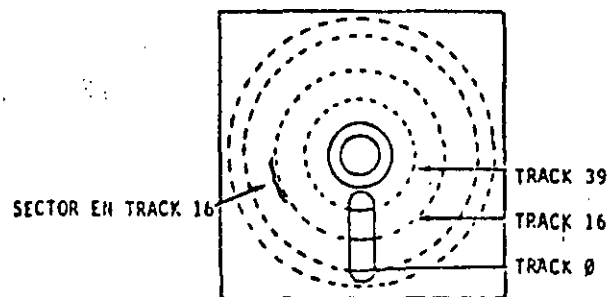
Nota : en computadoras con disco duro no hay que seguir estos pasos, debido a que el sistema se encuentra residente en el disco y se carga automáticamente.

APAGADO

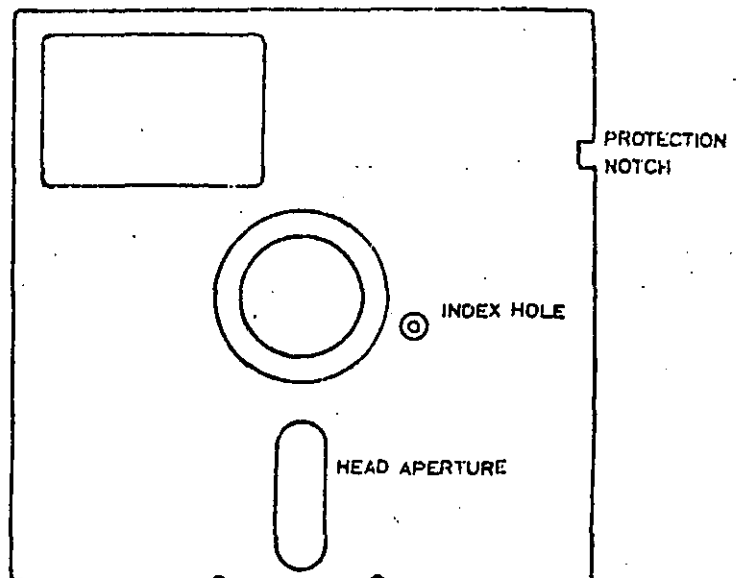
- Se apaga la unidad central.
- Se apaga el monitor.
- Se sacan el diskette de la unidad.

MANEJO DE DISKETTES

- EL EQUIPO IBM-PC UTILIZA DISKETTES DE DOBLE LADO (DOUBLE - SIDED) Y DOBLE DENSIDAD.
- EL FORMATO DE GRABACIÓN ES EL MISMO DE IBM-PC.
- ES NECESARIO FORMATEARLOS ANTES DE UTILIZARLOS
- EL SISTEMA OPERATIVO MS-DOS VERSIÓN 2.X MANEJA 40 TRACKS POR LADO.
- 8 ó 9 SECTORES POR TRACK DE 512 BYTES.
- CAPACIDAD NETA POR DISKETTE 360K

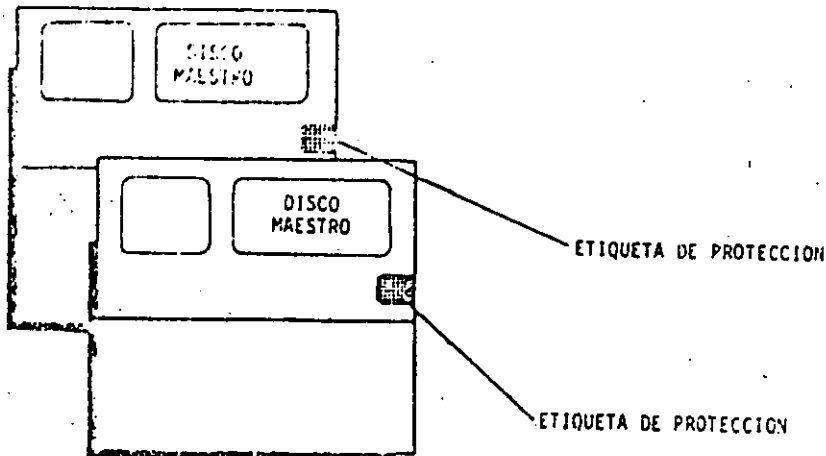


- LAS UNIDADES DE DISCO CUENTAN CON UN INDICADOR LUMINOSO QUE SE ENCENDERÁ AL ACTIVARSE LA UNIDAD Y SE APAGARÁ AL TERMINAR SU ACCESO.



PROTECCION DE DISKETTES

- COLOCANDO UNA ETIQUETA CUBRIENDO LA MUESCA DE PROTECCION EN EL DISKETTE ES COMO SE PUEDEN PROTEGER CONTRA BORRADO ACCIDENTAL.
- COMO PRECAUCION PARA LOS DISCOS MAESTROS HAY QUE PROTEGERLOS ANTES DE PROCEDER A COPIARLOS.



SIMULACION DE DOS DISKETTOS EN EQUIPOS
DE DISCO RIGIDO

- SI SE CUENTA CON DISCO RÍGIDO ES POSIBLE UTILIZAR LA UNIDAD DE DISKETTOS "A" TANTO COMO "A" Y COMO "B" PARA EFECTOS DE REALIZAR COPIAS DE DISKETTOS Ó TRANSFERENCIA DEL SISTEMA OPERATIVO.
- CADA VEZ QUE SE INVOQUE LA UNIDAD "B" EL SISTEMA INDICARÁ QUE SE CAMBIE EL DISKETTE DE LA UNIDAD "A" Y EN SU LUGAR SE INSERTE EL DISKETTE QUE EFECTUARÁ LAS FUNCIONES DEL DRIVE "B"

INSERT DISKETTE FOR DRIVE B: AND STRIKE
 ANY KEY WHEN READY.

- CUANDO EL SISTEMA REQUIERA HACER USO DE LA UNIDAD DE DISCO MOSTRARÁ EL SIGUIENTE MENSAJE:

INSERT DISKETTE FOR DRIVE A: AND STRIKE
 ANY KEY WHEN READY.

- CON EL SIGUIENTE COMANDO LEERÁ EL SISTEMA OPERATIVO DEL DISKETTE Y LO COLOCARÁ EN MEMORIA

SYS B:

INSERT DISKETTE FOR DRIVE B: AND STRIKE
 ANY KEY WHEN READY.

PARA FINALIZAR ESTE PROCEDIMIENTO COPIE EL ARCHIVO COMMAND.COM DEL DISCO "A" AL "B" CON EL COMANDO:

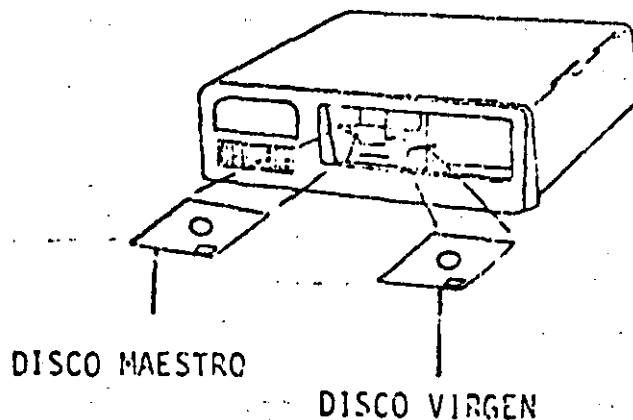
COPY A: COMMAND.COM B:

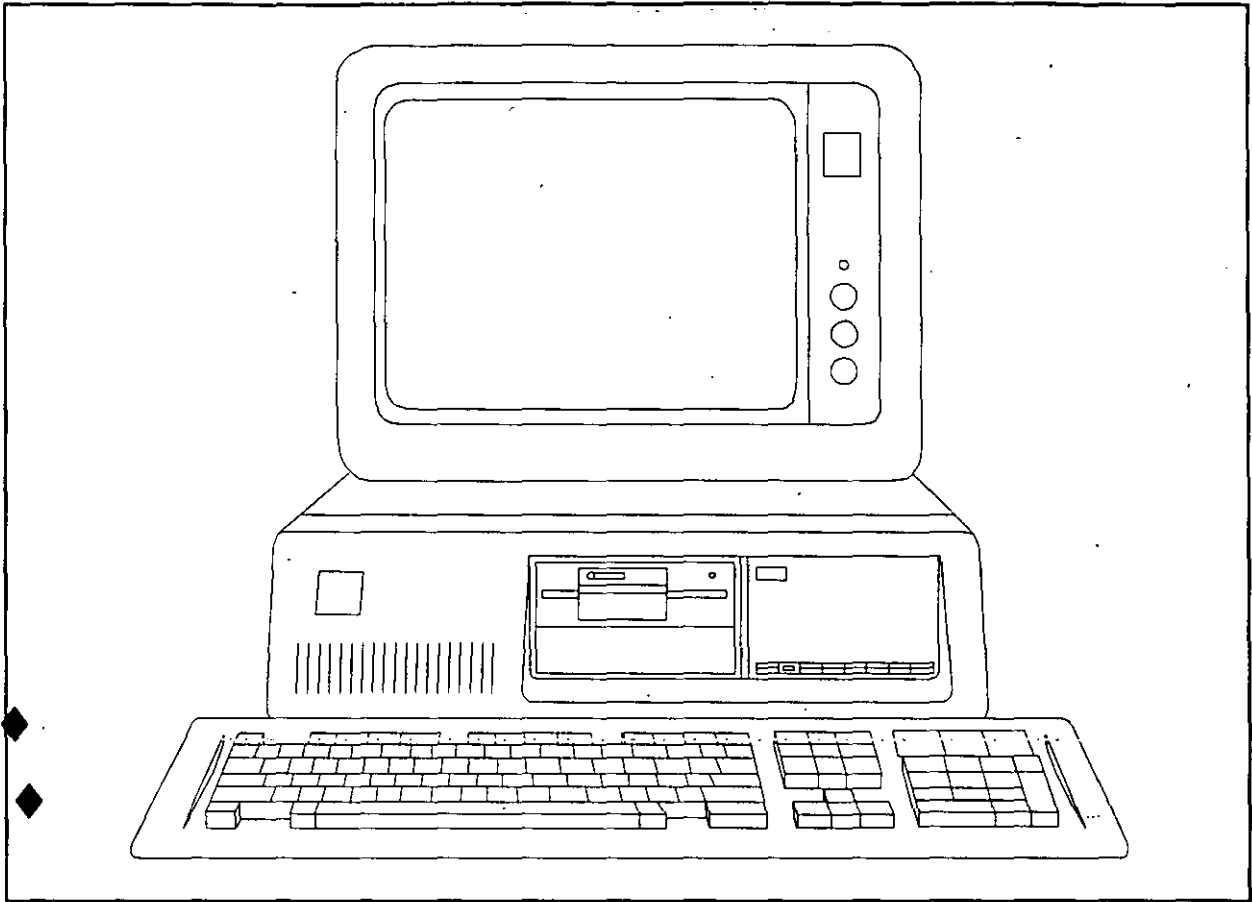
- OTRA FORMA DE SIMULAR UNIDADES DE DISKETTE ES A TRAVÉS DEL COMANDO ASSIGN.COM

POR EJEMPLO:

ASSIGN B=C

- ASOCIARÁ LA UNIDAD "B" CON LA "C" DE TAL FORMA QUE CUALQUIER ACCESO A LA UNIDAD "B" SE DIRIGIRÁ A LA UNIDAD "C"
- TODAS LAS UNIDADES DE DISKETTE PUEDEN ASIGNARSE CON ASSIGN.
- SI SE REQUIERE HACER LAS ASIGNACIONES DE DISCOS EN SU FORMA NORMAL, SIMPLEMENTE ESCRIBA ASSIGN SIN PARAMETROS.





INTRODUCCION A MS-DOS

SISTEMA OPERATIVO

ES EL CONJUNTO DE PROGRAMAS QUE CONTROLA LA UTILIZACIÓN DE LOS RECURSOS DE UN SISTEMA DE PROCESAMIENTO ELECTRÓNICO COMO SON: LA MEMORIA, LAS UNIDADES DE DISCO, LA IMPRESORA, LA PANTALLA, EL TECLADO, ETC. SUS DOS PRINCIPALES COMPONENTES SON UN PROGRAMA SUPERVISOR LLAMADO MONITOR Y UN CONJUNTO DE PROGRAMAS DE SOPORTE LLAMADOS UTILERÍAS.

EL PROGRAMA MONITOR ES LA INTERFACE DE COMUNICACIONES ENTRE LOS PROGRAMAS DE APLICACIÓN (SOFTWARE) Y LOS COMPONENTES FÍSICOS (HARDWARE), SIENDO SUS TRES FUNCIONES BÁSICAS: INICIALIZAR EL SISTEMA CON LOS PARÁMETROS NECESARIOS, CONTROLAR LA EJECUCIÓN DE LOS PROGRAMAS DE APLICACIONES Y DAR LAS PRIORIDADES A LAS OPERACIONES DE ENTRADA Y SALIDA.

LAS UTILERÍAS SON LA PARTE DEL SISTEMA OPERATIVO QUE BAJO EL CONTROL DEL PROGRAMA MONITOR REALIZAN FUNCIONES COMO COPIAR, IMPRIMIR Y EDITAR ARCHIVOS. ENTRE LOS PROGRAMAS DE UTILERÍA SE ENCUENTRAN LOS ENSAMBLADORES, EDITORES, INTÉRPRETES, COMPILADORES Y LIGADORES.

CARGA DE MS-DOS

CARGAR EL SISTEMA OPERATIVO MS-DOS SIGNIFICA ALMACENAR EN LA MEMORIA DEL SISTEMA COMPATIBLE. LOS PROGRAMAS QUE LO COMPONEN.

PRIMERO HAY QUE IDENTIFICAR EL DISKETTE QUE TIENE GRABADO MS-DOS Y QUE ACOMPAÑA AL SISTEMA SPERRY-PC CUANDO ES ENTREGADO. DICHO DISKETTE TIENE UNA ETIQUETA CON LA SIGUIENTE LEYENDA:

```

-----
VERSION 2.11  MICROSOFT (R) MS-DOS (R) VERSION 3.30
RELEASE 4      (C) COPYRIGHT MICROSOFT CORP 1981-1987
P/N 700500
  
```

HAY DOS FORMAS DE CARGAR MS-DOS:

- 1 - ENCENDER EL SISTEMA COMPATIBLE CON EL DISKETTE MS-DOS EN LA UNIDAD DE DISCO A.
- 2 - MIENTRAS EL SISTEMA COMPATIBLE SE ENCUENTRA ENCENDIDO INSERTAR EN LA UNIDAD DE DISCO A EL DISKETTE MS-DOS Y OPRIMIR SIMULTANEAMENTE LAS TECLAS DE CTRL, ALT Y DEL.

EN EL PRIMER PUNTO UNA VEZ QUE EL SISTEMA TERMINA DE REALIZAR UNA AUTOVERIFICACIÓN DE SUS COMPONENTES SE ESCUCHARÁ UN CHILLIDO DE LA UNIDAD DE DISCO QUE ESTÁ INICIANDO LA TRANSFERENCIA A MEMORIA DE MS-DOS.

ESCRIBA LA FECHA CON EL SIGUIENTE FORMATO:
MM-DD-AA Y OPRIMA LA TECLA DE <RETURN>

A LO QUE EL SISTEMA RESPONDE:

```
CURRENT DATE IS  SUN  1-01-1984
ENTER NEW DATE:  06-25-85
CURRENT TIME IS  0:01:06.78
ENTER NEW TIME:  _
```

ESCRIBA LA HORA EN EL SIGUIENTE FORMATO HH:MM Y
OPRIMA LA TECLA DE <RETURN>

Y EL SISTEMA CONTESTARÁ:

```
CURRENT TIME IS  0:01:06.78
ENTER NEW TIME   19:30:00
```

```
A>_
```

MS-DOS UTILIZA LOS SÍMBOLOS "A>" COMO MENSAJE DE PETICIÓN (PROMPT) E INDICA QUE MS-DOS ESTÁ UTILIZANDO POR OMISIÓN (DEFAULT) LA UNIDAD DE DISCO A. ASÍ DE ESTA MANERA EL SISTEMA COMPATIBLE SE ENCUENTRA DISPUESTO A RECIBIR LA SOLICITUD DE ALGÚN COMANDO.

CUESTIONARIO - 1

MODULO - 1

1.- ¿QUÉ ES UN SISTEMA OPERATIVO?

2.- ¿CUÁL ES LA FUNCIÓN DEL PROGRAMA MONITOR?

3.- MENCIONE UNA CARACTERÍSTICA DE LOS PROGRAMAS DE UTILERIAS EN UN SISTEMA OPERATIVO.

4.- ¿QUÉ ES MS-DOS?

5.- MENCIONE CINCO PROGRAMAS DEL SISTEMA OPERATIVO MS-DOS.

6.- ¿QUÉ SIGNIFICA CARGAR MS-DOS?

TIPOS DE COMANDOS

EXISTEN DOS TIPOS DE COMANDOS:

- INTERNOS
- EXTERNOS

LOS COMANDOS INTERNOS SON LOS MÁS SIMPLES Y LOS MÁS USADOS. ESTOS NO SE ENCUENTRAN EN EL DIRECTORIO DE LOS ARCHIVOS DE MS-DOS YA QUE FORMAN PARTE DEL ARCHIVO COMMAND.COM QUE TIENE COMO FUNCIÓN PROCESAR TODOS LOS COMANDOS DE MS-DOS, POR ESTO LOS COMANDOS INTERNOS SON EJECUTADOS DE MANERA INMEDIATA.

SEGUIDAMENTE SE PRESENTA UNA LISTA DE LOS COMANDOS INTERNOS QUE SE DESCRIBIRÁN DE FORMA MÁS DETALLADA MÁS ADELANTE:

BREAK	DEL(ERASE)	MKDIR(MD)	SET
CHDIR(CD)	DIR	PATH	SHIFT
CLS	ECHO	PAUSE	TIME
COPY	EXIT	PROMPT	TYPE
CTTY	FOR	REN	VER
DATE	GOTO	REN(RENAME)	VERIFY
	IF	RMDIR(RD)	VOL

LOS COMANDOS EXTERNOS RESIDEN EN DISCO COMO ARCHIVOS DE PROGRAMAS, POR LO QUE TIENEN QUE SER LEIDOS ANTES DE EJECUTARSE.

CUALQUIER ARCHIVO CON EXTENSIÓN DEL NOMBRE .COM, .EXT O .BAT ES CONSIDERADO COMANDO EXTERNO, COMO POR EJEMPLO FORMAT.COM Y COMP.COM.

POSTERIORMENTE SE DESCRIBIRÁN A DETALLE LAS SIGUIENTES COMANDOS EXTERNOS.

CHKDSK

MORE

DISKCOPY

PRINT

FIND

RECOVER

FORMAT

SORT

EXE22IN

SYS

ARCHIVOS

UN ARCHIVO ES UN CONJUNTO DE INFORMACIÓN A LA QUE SE ASIGNA UN NOMBRE Y SE ENCUENTRA ALMACENADA EN UN DISPOSITIVO SECUNDARIO, COMO ES, UN DISCO, UN DISKETTE O UN CASETE. ESTA INFORMACIÓN PUEDE SER PROGRAMAS, DATOS O TEXTOS.

ASÍ COMO LA INFORMACIÓN DE UN LIBRO ESTÁ AGRUPADA EN CAPÍTULOS; LA INFORMACIÓN EN UN DISCO ESTÁ AGRUPADA EN ARCHIVOS. CADA ARCHIVO TIENE UN NOMBRE Y SI SE DESEA LOCALIZAR UN ARCHIVO ENVIANDO A MS-DOS EL COMANDO DIR EL SISTEMA - COMPATIBLE. CONTESTARÁ SI SE ENCUENTRA RESIDENTE EN UN DISKETTE O NO.

UN DISKETTE DE UN SOLO LADO (SINGLE-SIDED) PUEDE ALMACENAR HASTA 64 ARCHIVOS Y HASTA 112 ARCHIVOS SI ES UN DISKETTE DE DOS LADOS (DUAL-SIDED).

EL NOMBRE DE UN ARCHIVO EN UN DISKETTE TIENE QUE SER ÚNICO, SIN EMBARGO, PUEDEN TENER EL MISMO NOMBRE DOS ARCHIVOS EN DIFERENTES DISKETTES.

EL NOMBRE DE UN ARCHIVO CONSISTE EN DOS PARTES, LA PRIMERA QUE ES PROPIAMENTE EL NOMBRE DEL ARCHIVO QUE PUEDE TENER HASTA 8 CARACTERES ALFANUMÉRICOS Y LA SEGUNDA QUE PUEDE CONTENER HASTA 3 CARACTERES Y DEFINE A QUE TIPO PERTENECE.

LA SIGUIENTE TABLA MUESTRA ALGUNOS DE LOS TIPOS DE ARCHIVOS MÁS COMUNES.

ASM	PROGRAMA FUENTE EN ENSAMBLADOR.
LIB	BIBLIOTECA DE SUB-RUTINAS DEL COMPILADOR.
COB	ARCHIVO FUENTE EN COBOL.
FOR	ARCHIVO FUENTE EN FORTRAN.
BAK	ARCHIVO DE RESPALDO.
COM	ARCHIVO PARA EJECUCIÓN.
BIN	ARCHIVO BINARIO.
ZBA	ARCHIVO BASIC EN PAQUETES MBA.
BAS	ARCHIVO FUENTE EN BASIC.
BAT	ARCHIVO TIPO "BATCH"
SYS	ARCHIVO CON DATOS DEL SISTEMA OPERATIVO.
OBJ	ARCHIVO OBJETO.
EXE	ARCHIVO DE EJECUCIÓN (SIMILAR A COM).

MODULO 2

18

LOS ARCHIVOS

NOTACION PARA EL NOMBRE DEL ARCHIVO

[< NOMBRE DE LA UNIDAD DE DISCO >]

< NOMBRE DEL ARCHIVO >

[< .EXTENSIÓN DEL NOMBRE DE ARCHIVO >]

EJ.: B: MIPROG.COB

A: TUPROG.EXT

A: NVOARCH.

TEXTO

LA SIGUIENTE ES UNA LISTA DE PALABRAS RESERVADAS QUE NO SE PUEDEN USAR COMO NOMBRES DE ARCHIVO:

AUX, CON, LST, PRN, NUL.

19

NOTACION UTILIZADA

- INTERPRETACIÓN DE LA NOTACIÓN UTILIZADA EN LA DESCRIPCIÓN DE LOS COMANDOS Y ENUNCIADOS.

[] : PARÉNTESIS DE CORCHETE INDICAN QUE EL CONTENIDO ES OPCIONAL.

< > PARÉNTESIS DIAGONALES PUEDEN INDICAR DOS OPCIONES.

A) SI EL CONTENIDO SE ENCUENTRA EN LETRAS MINÚSCULAS SIGNIFICA QUE SU CONTENIDO ES UN TEXTO,

EJ.: <NOMARCH>

B) SI EL CONTENIDO SE ENCUENTRA EN LETRAS MAYÚSCULAS SIGNIFICA QUE SE DEBE OPRIMIR LA TECLA QUE INDICA EL TEXTO.

EJ.: <RETURN>

{ } PARÉNTESIS DE LLAVE INDICA QUE SE TIENE QUE ESCOGER ALGUNA OPCIÓN A MENOS QUE LOS PARÉNTESIS DE LLAVE SE ENCUENTREN ENTRE PARÉNTESIS CUADRADOS.

... PUNTOS SUSPENSIVOS INDICAN QUE LAS INSTRUCCIONES PUEDEN SER REPETIDAS TANTAS VECES SE NECESITE.

| BARRA SIGNIFICA EL "0" LÓGICO.

MAYS LETRAS MAYÚSCULAS INDICAN QUE EL ENUNCIADO DEBE SER ESCRITO EXACTAMENTE DE ESA MANERA.

TODOS LOS SÍMBOLOS DE PUNTUACIÓN COMO SON: DOS PUNTOS, COMA, DIAGONALES, IGUAL DEBEN SER ENUNCIADOS COMO SE INDICA.

MANEJO DE ARCHIVOS

UN ARCHIVO EN UN SISTEMA OPERATIVO ES UNA COLECCIÓN DE DATOS ORGANIZADOS.

EXISTEN DOS FORMAS PARA ACCESAR ARCHIVOS:

- SECUENCIAL
- RANDOM

EN EL NOMBRE QUE SE LES ASIGNE SE PERMITEN HASTA 8 CARACTERES Y TRES MÁS PARA DEFINIR EL TIPO.

- MS-DOS MANEJA LOS PERIFÉRICOS COMO SI FUERAN ARCHIVOS. ES POSIBLE ENVIAR Ó RECIBIR DATOS A TRAVÉS DE CUALQUIER DISPOSITIVO.

- HAY QUE CONSIDERAR QUE EXISTEN PERIFÉRICOS QUE SOLO PUEDEN RECIBIR INFORMACIÓN (IMPRESORA), MIENTRAS HAY OTRAS QUE PUEDEN ENVIAR O RECIBIR.

LOS DIFERENTES PERIFERICOS QUE SE PUEDEN MANEJAR A

TRAVES DEL SISTEMA OPERATIVO SON:

DESCRIPCION	NOMBRE EN MS-DOS
DISKETE A Y B	A: Y B:
DISCO FIJO C	C:
TECLADO Y VIDEO	CON:
PRIMER PUERTO SERIE	AUX O COM1: IMPRESORA SECUNDA RIA.
PUERTO PARALELO	PRN, O LPT1 IMPRESORA PRIMARIA
FALTA DE EQUIPO (SIMULA UN NULO)	NUL:

COPIADO O RESPALDO DE ARCHIVO

SE ENTIENDE POR COPIA O RESPALDO DE UN ARCHIVO EL CREAR UN DUPLICADO, ESTO SE REALIZA POR MEDIO DEL COMANDO COPY. TAMBIÉN ES POSIBLE OBTENER ARCHIVOS PRODUCTO DE LA CONCATENACIÓN DE VARIOS ARCHIVOS FUENTE.

LA NOTACIÓN DEL COMANDO COPY ES LA SIGUIENTE:

COPY <ARCHFUE1>[+ARCHFUE2] [D:] [ARCHDES [.EXT]]

EJEMPLOS:

```

COPY  ARCHA  ARCHB
COPY  ARCHA  B:
COPY  *.BAS  B:
COPY  *.*    B:
COPY  A.XYZ+B.COM+B:C.TXT  BIGFILE.CRP

```

DIRECTORIOS

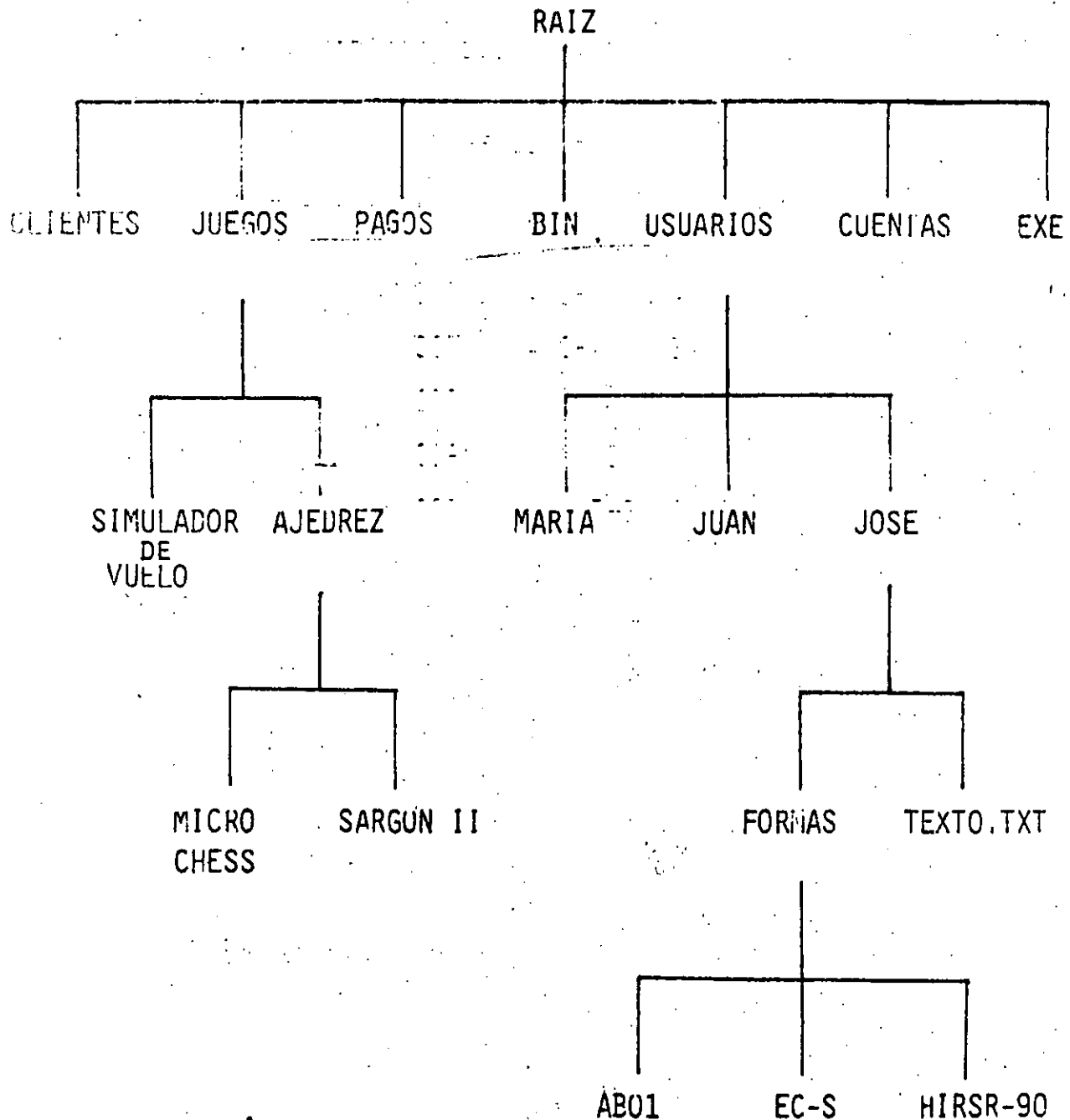
UN DIRECTORIO ES UN ARCHIVO QUE CONTIENE UNA LISTA DE INFORMACIÓN ACERCA DE LOS ARCHIVOS CONTENIDOS EN UN DISKETTE O DISCO. COMO INFORMACIÓN DE LOS ARCHIVOS SE ENCUENTRA EL NOMBRE, TAMAÑO, REFERENCIAS DE LOCALIZACIÓN Y FECHAS DE SU CREACIÓN, COMO DE LA ÚLTIMA ACTUALIZACIÓN.

CUANDO EN EL MISMO SISTEMA COMPATIBLE SE ESTÁ TRABAJANDO EN DISTINTOS PROYECTOS O CON VARIOS USUARIOS, EL NÚMERO DE ARCHIVOS QUE SE MANIPULAN PUEDE SER BASTANTE GRANDE, POR LO QUE, SE RECOMIENDA SEPARAR LA ORGANIZACIÓN EN DIFERENTES CATEGORÍAS. ASÍ DE LA MISMA MANERA QUE EN UNA OFICINA SE SEPARAN EN DIFERENTES GABINETES LA INFORMACIÓN ACERCA DE UNO U OTRO PROYECTO. MS-DOS TAMBIÉN REALIZA UNA ESTRUCTURA DE ORGANIZACIÓN JERÁRQUICA DONDE POR MEDIO DE DIRECTORIOS SE PUEDA DIVIDIR EN GRUPOS DE FORMA QUE MÁS CONVenga LOS ARCHIVOS.

CON UN DIRECTORIO DE ESTRUCTURA JERÁRQUICA SE PUEDE HACER LA ANALOGÍA A LA ESTRUCTURA DE UN ÁRBOL SÓLO QUE SE ENCUENTRA INVERTIDO O SEA LA RAÍZ ESTÁ EN LA PARTE ALTA O PRIMER NIVEL.

ESTE DIRECTORIO ES AUTOMATICAMENTE GENERADO CUANDO SE CREA EL FORMATO DE UN DISCO Y SE PROTEJEN ARCHIVOS.

LA SIGUIENTE FIGURA MUESTRA UN PROTOTIPO DE DIRECTORIO.



P A T H

ES UNA RUTA O TRAYECTORIA DENTRO DE UNA ESTRUCTURA DE ORGANIZACIÓN JERÁRGICA QUE PERMITE UN MANEJO EFICAZ DE LOS ARCHIVOS REFERENCIADOS EN UN DIRECTORIO.

CUANDO EN MS-DOS SE DESEA CREAR O BUSCAR UN ARCHIVO SE TIENE QUE CONOCER TRES ELEMENTOS:

- 1.- LA UNIDAD DE DISCO.
- 2.- EL NOMBRE DEL ARCHIVO.
- 3.- EL NOMBRE DEL DIRECTORIO.

SI EL ARCHIVO QUE SE DESEA BUSCAR SE ENCUENTRA EN EL DIRECTORIO QUE EN ESE MOMENTO SE ESTÁ TRABAJANDO NO ES NECESARIO ESPECIFICAR EL DIRECTORIO A MS-DOS AUTOMATICAMENTE TOMA EL QUE SE ESTÁ TRABAJANDO.

UN NOMBRE DE RUTA (PATHNAME) DENTRO DEL DIRECTORIO ES LA COMBINACIÓN DE NOMBRES DE DIRECTORIO SEGUIDOS POR EL NOMBRE DE UN ARCHIVO, TODOS SEPARADOS POR DIAGONALES INVERTIDAS.

NOTACIÓN DE UN NOMBRE DE RUTA (PATHNAME)

[D] \ [DIREC] \ [DIREC...] \ [FILENAME]

SI UN PATH INICIA CON UNA DIAGONAL INVERTIDA (\) MS-DOS INICIA LA BÚSQUEDA EN LA RAIZ DE DIRECTORIO.

EJEMPLOS:

\ INDICA LA RAÍZ DEL DIRECTORIO.
 \ PROGRAMAS DIRECTORIO DENTRO DE LA RAÍZ.
 \ USUARIOS \ MARIA \ FORMAS \ 1A
 NOMBRE COMPLETO DE RUTA
 \TEXT.TXT NOMBRE DE ARCHIVO EN UN DIRECTORIO DE
 TRABAJO.

MS-DOS TIENE UNA NOTACIÓN ESPECIAL TANTO PARA EL DIRECTORIO DE TRABAJO COMO PARA UN NIVEL SUPERIOR.

. INDICA EL NOMBRE DEL DIRECTORIO DE TRABAJO
 . . INDICA EL NOMBRE DEL DIRECTORIO EN UN NIVEL
 SUPERIOR.
 DIR . . LISTA LOS ARCHIVOS DE UN NIVEL SUPERIOR AL
 DIRECTORIO DE TRABAJO.
 DIR . . \ . . LISTA LOS ARCHIVOS DOS NIVELES SUPERIORES
 AL DIRECTORIO DE TRABAJO.

CUESTIONARIO - 2

MODULO - 2

1.- ¿QUÉ ES UN ARCHIVO?

2.- ¿CUÁNTOS ARCHIVOS SE PUEDEN ALMACENAR EN UN DISKETTE DE UN SOLO LADO Y EN UNO DE DOS LADOS?

3.- EN UN MISMO DISKETTE PUEDEN TENER DOS ARCHIVOS EL MISMO NOMBRE.

V () F ()

4.- ¿DE CUANTAS PARTES CONSISTE EL NOMBRE DE UN ARCHIVO?

5.- ESCRIBA LA NOTACIÓN PARA EL NOMBRE DE UN ARCHIVO.

6.- EN LA NOTACIÓN UTILIZADA QUE SIGNIFICAN:

A) [] _____

B) < > _____

C) { } _____

D) ... _____

E) | _____

F) LETRAS MAYÚSCULAS _____

7.- ¿QUÉ ES UN CARACTER TRANSPARENTE?

8.- ¿CUÁLES SON LOS CARACTERES TRANSPARENTES Y EN QUE SE DIFERENCIAN?

9.- ¿QUÉ FUNCIÓN REALIZA EL COMANDO COPY?

10.- ¿QUÉ ES UN DIRECTORIO?

11.- ¿CUÁL ES LA ESTRUCTURA DE UN DIRECTORIO?

12.- ¿QUÉ ES UN PATH?

PRACICA - 2

- 1.- ENCIENDA EL MONITOR.
- 2.- INSERTE EL DISCO MAESTRO EN LA UNIDAD A.
- 3.- ENCIENDA EL SISTEMA COMPATIBLE
- 4.- ENVIE LOS SIGUIENTES COMANDOS DE MS-DOS Y ESCRIBA ALGUN COMENTARIO DE LO DESPLEGADO EN LA PANTALLA POR EL SISTEMA COMPATIBLE

A) DIR <RETURN>

B) DIR / P <RETURN>

C) DIR / W <RETURN>

D) DIR *.COM <RETURN>

E) DIR *.EXE <RETURN>

F) DIR *.BAS <RETURN>

G) DIR CUES?.* <RETURN>

CARACTERES TRANSPARENTES

(WILD - CARDS)

LOS CARACTERES TRANSPARENTES SON AQUELLOS QUE PERMITEN ESPECIFICAR DE MANERA AMBIGUA UN ARCHIVO O UN GRUPO DE ARCHIVOS.

SI SE CUENTA CON UN GRUPO DE ARCHIVOS QUE COMIENCEN CON EL MISMO CARACTER O GRUPO DE CARACTERES SE PUEDE UTILIZAR LOS CARACTERES TRANSPARENTES PARA OBTENER UNA LISTA DE ESTOS ARCHIVOS.

MS-DOS ACEPTA DOS CARACTERES TRANSPARENTES, ESTOS SON EL SIGNO DE INTERROGACIÓN (?) Y EL ASTERISCO (*) Y SE INTERPRETAN DE LA SIGUIENTE MANERA:

SI AL PREGUNTAR POR UN ARCHIVO SE COLOCA EN EL NOMBRE UN (?) ÉSTE INDICA QUE CUALQUIER CARACTER PUEDE OCUPAR ESA POSICIÓN.

EJEMPLO:

```
A>DIR TEST?RUN.EXE
TEST1RUN.EXE
TEST2RUN.EXE
TEST3RUN.EXE
```

EL (*) ACTÚA DE MANERA SIMILAR AL (?) CON LA SALVEDAD QUE PUEDEN OCUPAR ESA POSICIÓN O A PARTIR DE ESA POSICIÓN HASTA EL FINAL DEL NOMBRE DEL ARCHIVO CUALQUIER CARACTER O CARACTERES.

EJEMPLO:

```
A)DIR TEST*.EXE
TEST1.EXE
TEST2.EXE
TEST3.EXE
```

EJEMPLO:

```
A)DIR TEST*.*
TEST1.EXE
TEST1.COB
TEST2.EXE
TEST2.BAS
TEST3.EXE
TEST3.FOR
```

OPCIONES PARA COMANDOS

LOS COMANDOS DE MS-DOS PUEDEN INCLUIR OPCIONES PARA ESPECIFICAR INFORMACIÓN ADICIONAL, SI ESTO NO OCURRE MS-DOS - ASUMIRÁ LA OPCIÓN POR OMISIÓN.

EN LA DESCRIPCIÓN DE LOS COMANDOS DE MS-DOS SE INDICA CUALES SON, EN SU CASO, LAS OPCIONES.

HAY QUE CONSIDERAR LOS SIGUIENTES PUNTOS COMO INFORMACIÓN GENERAL ACERCA DE LAS OPCIONES A LOS COMANDOS DE MS-DOS.

- 1 - EN LA MAYORÍA DE LOS CASOS LOS COMANDOS VAN SEGUIDOS DE UNA O MAS OPCIONES.
- 2 - LOS COMANDOS Y OPCIONES PUEDEN IR SEPARADOS POR LOS DELIMITADORES ESPACIO O COMA.

EJEMPLO: DEL ARCH.ANT NEWFILE.TXT

RENAME, ESTEARCH AGUELARC

TAMBIÉN SE PUEDE UTILIZAR EL SÍMBOLO PUNTO Y COMA (;) Y EL SÍMBOLO IGUAL (=) COMO DELIMITADORES.

- 3 - PARA ABORTAR LA EJECUCIÓN DE ALGÚN COMANDO OPRIMIR A LA VEZ LA TECLA DE <CTR> Y LA TECLA DE <C> REPRESENTADA CON LA SIGUIENTE NOTACIÓN QUE TIENE EL MISMO SIGNIFICADO <CONTROL - C> .
- 4 - LOS COMANDOS INICIAN SU EJECUCIÓN DESPUÉS DE OPRIMIR LA TECLA <RETURN>

- 5 - LOS CARACTERES TRANSPARENTES NO SE PUEDEN UTILIZAR EN LOS NOMBRES DE LOS COMANDOS.
- 6 - CUANDO LA EJECUCIÓN DE ALGÚN COMANDO PRODUCE UNA LARGA LISTA DE RESULTADOS EN LA PANTALLA OPRIMIENDO LAS TECLAS <CONTROL - S> SUSPENDE SU SALIDA Y DESPUÉS DE OPRIMIR CUALQUIER TECLA MS-DOS ENVÍA UN RESUMEN A PANTALLA.
- 7 - PARA LOS COMANDOS DE MÁS MANEJO ES POSIBLE UTILIZAR LAS TECLAS CON FUNCIONES ESPECIALES DE EDICIÓN.
- 8 - LOS SÍMBOLOS DE PETICIÓN (PROMPT) ES LA IDENTIFICACIÓN DE LA UNIDAD DE DISCO POR OMISIÓN SEGUIDO DEL SÍMBOLO "MAYOR-QUE" ,POR EJEMPLO A>

PROCESAMIENTO BATCH

CUANDO PARA DETERMINADA TAREA EN MÁS DE UNA OCASIÓN SE REQUIERE EJECUTAR LA MISMA SECUENCIA DE COMANDOS DE MS-DOS ES POSIBLE CREAR UN ARCHIVO QUE CONTenga ESA SECUENCIA DE COMAN DOS Y PROCESARSE EN EL MOMENTO QUE EL NOMBRE DEL ARCHIVO SEA TECLEADO Y ENVIADO A MS-DOS. ASÍ LO ANTERIORMENTE DESCRITO ES LLAMADO PROCESAMIENTO BATCH.

EL ARCHIVO TIENE QUE TENER DE EXTENSIÓN DE NOMBRE BAT AUNQUE PARA SER EJECUTADO NO SE REQUIERE TECLEAR LA EXTENSIÓN DEL NOMBRE.

EN LOS ARCHIVOS TIPO BAT SE PUEDE UTILIZAR LOS COMANDOS DE MS-DOS REM Y PAUSE. DONDE REM PERMITE COMENTARIZAR Y PAUSE HACER ESPERA, SI ES EL CASO, DE ALGUN PROCESO MANUAL.

EJ:

```
1: REM ESTE ARCHIVO VERIFICA DISKETTES NUEVOS
2: REM NOMBRE DE ARCHIVO NEWDISK.BAT
3: PAUSE INSERTE EL NUEVO DISCO EN LA
   UNIDAD B.
4: FORMAT B:
5: DIR B:
6: CHKDSK .B:
```

PARA EJECUTAR ESTE ARCHIVO TIPO .BAT BASTA TECLEAR Y ENVIAR A MS-DOS EL NOMBRE DEL ARCHIVO NEWDISK <RETURN> .

ARCHIVOS BAT

ES POSIBLE CREAR UN ARCHIVO TIPO BAT AL QUE SE PUEDAN DAR COMO PARÁMETROS LOS NOMBRES DE LOS ARCHIVOS QUE SE UTILIZARÁN EN EL PROCESAMIENTO BATCH.

LOS PARÁMETROS CON REFERENCIA DE %0 - %9 SERÁN REEMPLAZADOS POR SUS CORRESPONDIENTES VALORES CUANDO EL ARCHIVO TIPO BAT SE EJECUTE.

POR EJEMPLO CUANDO SE TECLEE LA LÍNEA DE COMANDO:

```
A) COPY CON ARCH.BAT
```

LAS SIGUIENTES LÍNEAS TECLEADAS SERÁN COPIADAS DE LA PANTALLA (CONSOLA) A EL ARCHIVO CON NOMBRE ARCH.BAT EN LA UNIDAD DE DISCO POR OMISIÓN.

```
A) COPY CON ARCH.BAT
COPY %1.MAC %2.MAC
TYPE %2.PRN
TYPE %0.BAT
<CONTROL-Z> <RETURN>
```

AL OPRIMIR <CONTROL-Z> Y <RETURN> MS-DOS RESPONDE CON ESTE MENSAJE:

```
1 FILE(S) COPIED
A) _
```

EL ARCHIVO ARCH.BAT CONSISTE DE TRES COMANDOS
Y RESIDE EN EL DISKETTE QUE SE ENCUENTRA EN LA UNIDAD A.

PARA EJECUTAR EL PROCESO BATCH ARCH HAY QUE
TECLEAR:

A) ARCH A:PROG1 B:PROG2

ARCH ES SUBSTITUIDO POR %0 , A:PROG1 POR %1
Y B:PROG2 POR %2.

EL RESULTADO ES IGUAL QUE SI SE HUBIERA TECLEADO
LOS SIGUIENTES PARÁMETROS:

A) COPY A:PROG1.MAC B:PROG2.MAC
TYPE B:PROG2.PRN
TYPE ARCH .BAT

REDIRECCIONAMIENTO DE LA SALIDA

LOS COMANDOS DE MS-DOS SIEMPRE DAN POR HECHO QUE LA ENTRADA ES EL TECLADO Y LA SALIDA ES LA PANTALLA, SIN EMBARGO SE PUEDEN REDIRECCIONAR, POR EJEMPLO, LA ENTRADA PUEDE SER UN ARCHIVO EN DISCO Y LA SALIDA UNA IMPRESORA.

PARA REDIRECCIONAR LA SALIDA DE UN COMANDO DE MS-DOS A UN ARCHIVO SE UTILIZA EL SÍMBOLO MAYOR-QUE (>).

EJEMPLO:

```
A> DIR> ARCH
```

SI EL ARCHIVO ARCH NO EXISTE MS-DOS LO CREA Y ALMACENA LA LISTA DEL DIRECTORIO.

SI SE REQUIERE AGREGAR EL DIRECTORIO A OTRO ARCHIVO UTILIZANDO DOS SÍMBOLOS SEGUIDOS DE MAYOR-QUE (>>) ES COMO SE INDICA A MS-DOS QUE AÑADA AL FINAL DE UN ARCHIVO OTRO ARCHIVO.

EJEMPLO:

```
A> DIR >> ARCH
```

PARA ASIGNAR UN ARCHIVO COMO ENTRADA DE UN COMANDO DE MS-DOS SE UTILIZA EL SÍMBOLO MENOR-QUE (<).

EJEMPLO:

A) SORT < NOMBRES > LISTA1

EL RESULTADO ES QUE CLASIFICA EL ARCHIVO NOMBRES DEJANDO EL RESULTADO EN EL ARCHIVO LISTA1.

FILTROS Y CONEXIONES

38

UN FILTRO ES UN COMANDO QUE LEE UNA ENTRADA Y DE ALGUNA MANERA LA TRANSFORMA GENERANDO UNA SALIDA, POR LO REGULAR UN ARCHIVO. ASÍ SE DICE QUE LOS DATOS DE ENTRADA SE HAN FILTRADO.

ENTRE ALGUNOS DE LOS COMANDOS DE MS-DOS QUE SON FILTROS SE CUENTAN LOS SIGUIENTES:

COMANDOFUNCION

FIND

BUSCA UNA CADENA DE CARACTERES EN UN TEXTO O ARCHIVOS.

SORT

CLASIFICA TEXTOS

EN MS-DOS SE EXTIENDE EL CONCEPTO DE REDIRECCIONAMIENTO POR MEDIO DEL DE CONEXIÓN (PIPIN) DEBIDO QUE SU FUNCIÓN ES TOMAR LA SALIDA DE UN COMANDO DE MS-DOS COMO LA ENTRADA DE UN SEGUNDO COMANDO. ESTO SE REALIZA POR MEDIO DE UN SEPARADOR QUE ES EL SÍMBOLO DE BARRA VERTICAL (|).

EJEMPLO:

```
A> DIR | SORT
```

DE ESTA MANERA SE OBTIENE COMO RESULTADO UNA LISTA DEL DIRECTORIO CLASIFICADA EN ORDEN ASCENDENTE.

EJEMPLO:

```
A> DIR SORT > DIREC.FIL
```

```
A> DIR SORT > B:DIREC.FIL
```

```
A> DIR | SORT | MORE
```

COMANDOS DE MS-DOS.

A CONTINUACIÓN SE INICIARÁ LA DESCRIPCIÓN DE LOS COMANDOS DE MS-DOS QUE DAN MANTENIMIENTO, PREPARAN, ORGANIZAN Y COPIAN LOS DISKETTES QUE SE UTILIZARON EN EL SISTEMA COMPATIBLE

FORMAT

DESCRIPCIÓN: PREPARA UN DISKETTE NUEVO PARA QUE ÉSTE SEA UTILIZADO POR MS-DOS, A TRAVÉS DE LO SIGUIENTE:

- ESCRIBE LA INFORMACIÓN NECESARIA QUE ORGANIZA EL ESPACIO EN PISTAS Y SECTORES.
- VERIFICA QUE NO EXISTAN ZONAS FÍSICAMENTE DAÑADAS EN EL DISKETTE.
- GRABA EL DIRECTORIO DE ASIGNACIÓN DE ESPACIO DEL DISKETTE.
- GRABA EL PROGRAMA DENOMINADO CARGADOR DEL SISTEMA.
- AGREGA, DE MANERA OPCIONAL, AL DISKETTE EL SISTEMA OPERATIVO MS-DOS.

NOTACIÓN:

FORMAT [D]:[/V][/S]

EL PARÁMETRO D REPRESENTA A LA UNIDAD DE DISCO Y LAS OPCIONES TIENEN EL SIGUIENTE SIGNIFICADO:

V - ES EL VOLUMEN DEL DISCO.

S - ES LA ALTERNATIVA PARA GRABAR MS-DOS.

EJEMPLOS:

A> FORMAT B:

A> FORMAT :/S

S Y S

DESCRIPCIÓN: REALIZA LA MISMA FUNCIÓN QUE EL COMANDO
FORMAT CON LA OPCIÓN /S, ES DECIR TRANSFIERE MS-DOS
DE LA MEMORIA DEL SISTEMA COMPATIBLE AL DISKETTE QUE
SE HALLA EN LA UNIDAD ESPECIFICADA EN EL COMANDO.

NOTACIÓN:

SYS <D> :

EJEMPLOS:

A> SYS :

A> SYS B:

DISKCOPY

DESCRIPCIÓN: REALIZA EL COPIADO DE UN DISCO COMPLETO.

NOTACIÓN:

DISKCOPY [D:] [E:]

DONDE LOS PARÁMETROS TIENEN EL SIGUIENTE SIGNIFICADO:

D - ES LA UNIDAD DE DISCO ORIGEN.

E - ES LA UNIDAD DE DISCO DESTINO.

EJEMPLOS:

A> DISKCOPY A: B:

A> DISKCOPY A:

EJEMPLCS:

45

```
A>DIR  
A>DIR E:  
A>DIR .EXT  
A>DIR NCMBARCH
```

AHORA SE EXAMINARÁN LOS COMANDOS CON LOS CUALES EN MS-DOS SE EFECTÚA EL MANEJO DE ARCHIVOS.

DIR

DESCRIPCIÓN: REALIZA EL LISTADO DE LOS NOMBRES DE LOS ARCHIVOS DE UN DIRECTORIO.

NOTACIÓN:

DIR [FILESPEC] [PATHNAME] [/P][/W]

DONDE LOS PARÁMETROS TIENEN EL SIGUIENTE SIGNIFICADO:

<FILESPEC> - NOMBRE ESPECÍFICO DE ARCHIVO.

<PATHNAME> - NOMBRE DE UNA RUTA O TRAYECTORIA DEL DIRECTORIO DEL DISKETTE.

Y LAS OPCIONES TIENEN EL SIGUIENTE SIGNIFICADO:

P - REALIZA PAUSA DESPUÉS DE HABER COMPLETADO UNA PANTALLA Y OPRIMIENDO CUALQUIER TECLA CONTINÚA ELISTANDO.

W - LISTA CINCO NOMBRES, DE ARCHIVO POR LÍNEA SIN ESPECIFICAR OTRA INFORMACIÓN.

50

CHKDSK

46

DESCRIPCIÓN: VERIFICA LA CONSISTENCIA DE LOS ARCHIVOS CONTENIDOS EN UN DISKETTE CON RESPECTO A SU DIRECTORIO, DANDO COMO RESULTADO ENTRE OTRA INFORMACIÓN LOS CARACTERES UTILIZADOS Y LOS LIBRES.

NOTACIÓN:

CHKDSK [D:]<FILESPEC> [/F] [/V]

DONDE LAS OPCIONES TIENEN EL SIGUIENTE SIGNIFICADO:

- F - CORRIGE LOS ERRORES ENCONTRADOS.
- V - DESPLIEGA EN PANTALLA POR MEDIO DE MENSAJES LO QUE ESTÁ OCURRIENDO EN LA EJECUCIÓN.

TYPE

DESCRIPCIÓN: LISTA EN LA PANTALLA ALGÚN ARCHIVO EN ESPECÍFICO.

NOTACIÓN: TYPE <FILESPEC>

EJEMPLO: A>TYPE NOMARCH

A>TYPE NOMARCH

COPY

DESCRIPCION: OBTIENE DUPLICADO O CONCATENACIONES DE ARCHIVOS.

NOTACIONES:

COPY <FILESPEC> [FILESPEC] [PATHNAME]
[PATHNAME] [/V]

PARA CONCATENACIONES SE TIENE LA SIGUIENTE NOTACIÓN:

COPY FILESPEC [+FILESPEC] [D:] [FILENAME [.EXT]]

EJEMPLOS:

A>COPY B: STAR.BAS

A>COPY STAR.BAS B:

A>COPY *.* B:

A>COPY *.LST ALL.LST

A>COPY ALL.LST + *.LST

A>COPY *.LST + *.REF COMBIN.PRN

DEL

DESCRIPCIÓN: BORRA UN ARCHIVO DEL DIRECTORIO Y POR ENDE DEL DISKETTE.

NOTACIÓN: DEL [FILESPEC] [PATHNAME]

EJEMPLO:

A>DEL ARCH1

A>DEL

RECOVER

DESCRIPCIÓN: RECUPERA UN ARCHIVO O DISKETTE COMPLETO - QUE CONTENGA SECTORES DAÑADOS.

NOTACIÓN: RECOVER <FILENAME [D:]>

EJEMPLO:

A>RECOVER ARCH1

A>RECOVER B:

MKDIR

DESCRIPCIÓN: CREA UN NUEVO DIRECTORIO.

NOTACIÓN: MKDIR <PATHNAME>

EJEMPLO:

```
A>MKDIR \USUARIOS
A>MKDIR \USUARIO\JOSE
```

CHDIR

ó

CD

DESCRIPCIÓN: CAMBIA UN DIRECTORIO A UNA DIFERENTE RUTA O TRAYECTORIA. (PATH).

NOTACIÓN: CHDIR [PATHNAME]

EJEMPLO:

```
A>CHDIR \BINARIO\USUARIO\JOSE\FORMAS
A>CHDIR . . .
```

RMDIR

DESCRIPCIÓN: BORRA UN DIRECTORIO DE UNA ESTRUCTURA JERÁRQUICA.

NOTACIÓN: RMDIR <PATHNAME>

EJEMPLO:

```
A>RMDIR \BIN\USUARIO\JOSE
```

P A T H

DESCRIPCIÓN: INDICA A MS-DOS EN QUE DIRECTORIO BUSCARÁ LOS COMANDOS EXTERNOS DESPUÉS DE HACERLO EN EL DIRECTORIO DE TRABAJO.

NOTACIÓN:

PATH [<PATHNAME> [; <PATHNAME>] ...]

EJEMPLO:

A>PATH \BIN\USUARIOS\JOSE

A>PATH \BIN\USUARIOS\JOSE;
 \BIN\USUARIOS\MARIA

R E N

DESCRIPCIÓN: CAMBIA EL NOMBRE DE UN ARCHIVO EN UN DISKETTE O DISCO,

NOTACIÓN: REN <FILESPEC><FILENAME>

EJEMPLOS:

A>REN *.LIST *.PRN

A>REN A AUX

FIND

DESCRIPCIÓN: BUSCA UNA CADENA DE CARACTERES EN ESPECIFICO EN UN ARCHIVO O ARCHIVOS.

NOTACIÓN:

FIND [\V\C\N] <STING> [<FILENAME...>]

DONDE LAS OPCIONES TIENEN EL SIGUIENTE SIGNIFICADO:

- V - DESPLIEGA EN PANTALLA TODAS LAS LÍNEAS QUE NO CONTIENEN LA CADENA DE CARACTERES BUSCADO.
- C - DESPLIEGA ÚNICAMENTE EL NÚMERO DE LÍNEAS DEL TOTAL QUE CONTIENEN LA CADENA BUSCADA.
- N - DESPLIEGA EL NÚMERO Y LA LÍNEA DEL ARCHIVO DONDE ENCONTRÓ LA CADENA DE CARACTERES.

EJEMPLOS:

```
A>FIND "HOLA" ARCH1
A>FIND \C "HOLA" ARCH1.*
A>FIND \N "HOLA" *.*
A>DIR B: | FIND \V "DAT"
```

DESCRIPCIÓN: BUSCA UNA CADENA DE CARACTERES EN ESPECIFICO EN UN ARCHIVO O ARCHIVOS.

NOTACIÓN: FIND [\V\C\N] <STING> [<FILENAME...>]

EJEMPLOS:

SET

DESCRIPCIÓN: EN UN PROGRAMA HACE EQUIVALENTE EL VALOR DE UNA CADENA DE CARACTERES POR EL DE OTRA CADENA DE CARACTERES.

NOTACIÓN:

SET [<STRING1> = <STRING2>]

EJEMPLOS:

A) SET TTY=UTS2

A) SET RE=READ

EXE2BIN

DESCRIPCIÓN: CONVIERTE ARCHIVOS DE TIPO EXE EJECUTABLE A FORMATO BINARIO.

NOTACIÓN:

EXE2BIN <FILESPEC> [D:] [<FILENAME>:<EXT>]

EJEMPLOS:

A) EXE2 BIN ARCH1.EXE ARCH1.BIN

SORT

DESCRIPCIÓN: CLASIFICA TEXTOS EN LA PANTALLA
O ARCHIVOS.

NOTACIÓN:

SORT [/R] [/+N]

DONDE LAS OPCIONES TIENEN EL SIGUIENTE SIGNIFICADO:

R - EL ORDEN DESEADO ES DESCENDENTE.

+N - INICIA EN LA COLUMNA N LA LLAVE DE LA CLASIFICACIÓN.

EJEMPLOS:

A>SORT /R < ARCH1.TXT > ARCH LA.TXT

A>DIR | SORT / + 14

DATE

DESCRIPCIÓN: PERMITE CARGAR LA FECHA.

NOTACIÓN:

DATE [<MM> - <DD> - <AA>]

EJEMPLO:

A)DATE 12-7-85

TIME

DESCRIPCIÓN: PERMITE CARGAR LA HORA.

NOTACIÓN:

TIME [<HH>[:<MM>]]

EJEMPLO:

A)TIME 8:20

VER

DESCRIPCIÓN: CUANDO SE ESTÁ GRABANDO EN UN DISKETTE ENCIENDE O APAGA LA OPCIÓN DE VERIFICACIÓN.

NOTACIÓN:

VERIFY [ON | OFF]

EJEMPLO:

A) VERIFY ON

A) VERIFY OFF

VOL

DESCRIPCIÓN: SI EXISTE DESPLIEGA EL VOLUMEN DE UN DISCO.

NOTACIÓN:

VOL [D:]

BREAK

DESCRIPCIÓN: ENCIENDE O APAGA LA OPCIÓN PARA INTERRUMPIR LA EJECUCIÓN DE COMANDOS DE MS-DOS CON LAS TECLAS DE <CONTROL-C>

NOTACIÓN:

BREAK ON | OFF

CLS

DESCRIPCIÓN: BORRA LA PANTALLA.

NOTACIÓN:

CLS

CTTY

DESCRIPCIÓN: CAMBIA EL PERIFÉRICO AL QUE SE ENVÍAN O
RECIBEN LOS COMANDOS DE MS-DOS.

NOTACIÓN:

CTTY / <DEVICE>

EJEMPLOS:

A) CTTY /AUX

A) CTTY /CON

PRINT

DESCRIPCIÓN: IMPRIME UN ARCHIVO DE TEXTO EN LA IMPRESORA MIENTRAS SE PROCESAN OTROS COMANDOS DE MS-DOS (BACKGROUND PRINTING).

NOTACIÓN:

PRINT [<FILESPEC>] [/T] [/C] [/P]...

DONDE LAS OPCIONES TIENEN EL SIGUIENTE SIGNIFICADO:

T - TERMINANTE
C - CANCEL
P - PRINT

EJEMPLOS:

A)PRINT /T *.ASM

A)PRINT A:TEMP1.TST/C A:TEMP2.TST

A)PRINT TEMP1.TST/C TEMP2.TST/P
TEMP3.TST

MORE

DESCRIPCIÓN: ENVÍA LA INFORMACIÓN SOLICITADA POR PANTALLAS COMPLETAS, PARA CONTINUAR CON LAS SIGUIENTES PANTALLAS HASTA OPRIMIR LA TECLA DE < RETURN >

NOTACIÓN: MORE

PROMPT

DESCRIPCIÓN: CAMBIA LOS SÍMBOLOS DE PETICIÓN (PROMPT) SEGÚN LA SIGUIENTE TABLA:

<u>CARACTER</u>		<u>PROMPT</u>
\$	-	\$
T	-	LA HORA
D	-	LA FECHA
P	-	EL DIRECTORIO DEL DISKETTE
V	-	LA VERSIÓN
N	-	LA UNIDAD DE DISCO POR OMISIÓN
G	-	EL CARACTER MAYOR-QUE (>)
L	-	EL CARACTER MENOS-QUE (<)
B	-	LA BARRA ()

NOTACIÓN:

PROPMT [< PROMPT-TEXT >]

CUESTIONARIO - 3

MODULO - 3

000000

1.- ¿CUÁL ES LA FUNCIÓN DE LOS COMANDOS DE MS-DOS?

2.- ¿CUÁNTOS TIPOS DE COMANDOS DE MS-DOS EXISTEN?

3.- ¿CUÁLES SON LOS TIPOS DE COMANDOS DE MS-DOS?

4.- ¿CUÁLES SON LAS TECLAS QUE HAY QUE OPRIMIR PARA CANCELAR LA EJECUCIÓN DE ALGUN COMANDO DE MS-DOS?

5.- ¿QUÉ ES UN PROCESAMIENTO BATCH?

6.- ¿QUÉ CARACTERÍSTICA TIENEN LOS ARCHIVOS TIPO BATCH?

7.- ¿POR MEDIO DE QUE SIMBOLO SE PUEDE DIRECCIONAR LA SALIDA DE UN COMANDO DE MS-DOS?

8.- ¿POR MEDIO DE QUE SIMBOLO SE PUEDE DIRECCIONAR LA ENTRADA A UN COMANDO DE MS-DOS?

9.- ¿QUÉ FUNCIÓN REALIZA EL SIMBOLO BARRA | ENTRE DOS COMANDOS DE MS-DOS?

10.- ¿QUÉ REALIZA EL COMANDO FORMAT?

11.- ¿EL COMANDO SYS ES EQUIVALENTE AL COMANDO FORMAT CON OPCIÓN /S?

V () F ()

12.- EL COMANDO DISKCOPY REALIZA EL COPIADO DE UN ARCHIVO COMPLETO.

V () F ()

13.- EL COMANDO DIR CON LA OPCIÓN /P DESPUÉS DE HABER COMPLETADO UNA PANTALLA REALIZA UNA PAUSA.

V () F ()

14.- ¿QUÉ FUNCIÓN REALIZA EL COMANDO CHKDSK?

15.- EL COMANDO DEL ELIMINA UN ARCHIVO DE UN DISKETTE.

V () F ()

16.- ¿QUÉ FUNCIÓN REALIZA EL COMANDO CD?

FUNCION DE LAS TECLAS ESPECIALES

DE EDICION

70

A LAS TECLAS CON FUNCIONES ESPECIALES DE EDICIÓN SE LES QUIERE DAR UN PARTICULAR ÉNFASIS, POR EL HECHO QUE NOS PERMITIRÁN HACER UN RÁPIDO ENVÍO DE LOS COMANDOS DE MS-DOS. NO SE TIENE QUE REPETIR LA MISMA SECUENCIA DE CARACTERES, DADO QUE LA ÚLTIMA LÍNEA DE COMANDOS ENVIADA A MS-DOS ES AUTOMATICAMENTE ALMACENADA EN UNA ÁREA DE MEMORIA LLAMADA PLANTILLA.

A CONTINUACIÓN SE PRESENTAN LAS VENTAJAS QUE SE OBTENDRAN SI SE UTILIZA LA PLANTILLA Y LAS TECLAS ESPECIALES DE EDICIÓN:

- 1 - UNA LÍNEA DE COMANDOS PUEDE SER REPETIDA OPRIMIENDO SOLAMENTE DOS TECLAS.
- 2 - SI SE COMETE UN ERROR EN UNA LÍNEA DE COMANDO SE PUEDE CORREGIR SIN NECESIDAD DE VOLVER A TECLEARLA COMPLETAMENTE.
- 3 - UNA LÍNEA DE COMANDO QUE ES SIMILAR A LA QUE LE ANTECEDIÓ PUEDE SER EDITADA Y EJECUTADA CON UN MÍNIMO DE TECLAS OPRIMIDAS.

71

7.- ENVIE CLS A MS-DOS Y COMENTARICE.

64

8.- ENVIE A MS-DOS Y COMENTARICE.

A) PROMPT] _____

B) PROMPT \$T] _____

C) PROMPT \$D> _____

D) PROMPT \$L _____

E) PROMPT \$N _____

9.- GENEERE UN ARCHIVO IGUAL AL DEL EJEMPLO DE LA PÁGINA 33 *ya* EJECÚTELO Y COMENTARICE. _____

10 - Genere un archivo igual al del ejemplo de la pagina 34 ejecútelolo y comentarice

000099

LA SIGUIENTE TABLA RESUME LAS FUNCIONES DE LAS TECLAS ESPACIALES DE EDICIÓN:

T A B L A 4-1

TECLADO SPERRY-PC	F U N C I Ó N	GUÍA DEL USUARIO
F 1	COPIA UN CARACTER DE LA PLANTILLA A LA LÍNEA DE COMANDO.	COPY 1
F 2	COPIA LOS CARACTERES DE LA PLANTILLA HASTA UN CARACTER ESPECIFICADO A LA LÍNEA DE COMANDO.	COPY UP
F 3	COPIA EL RESTO DE LOS CARACTERES DE LA PLANTILLA A LA LÍNEA DE COMANDO.	COPY ALL
F 4	SALTA EN LA PLANTILLA (NO COPIA) HASTA UN CARACTER ESPECIFICADO.	SKIP UP
F 5	REALIZA UNA NUEVA LÍNEA PARA LA PLANTILLA.	NEW LINE
F 6	COLOCA UN CARACTER DE FIN DE ARCHIVO (CTRL-Z Ó IAH) EN LA NUEVA PLANTILLA.	CONTROL-Z
DEL	SALTA (NO COPIA) UN CARACTER EN LA PLANTILLA.	SKIP 1
INS	ENTRA/SALE DEL MODO DE INSERCIÓN	INSERT
CTRL- C CTRL-BREAK	LIMPIA LA LÍNEA DE COMANDO Y ENVÍA LA PLANTILLA SIN CAMBIO.	VOID

EJEMPLO:

SI SE TECLEA EL SIGUIENTE COMANDO:

A) DIR. PROG.COM

MS-DOS MOSTRARÁ EN LA PANTALLA INFORMACIÓN ACERCA DEL ARCHIVO PROG.COM. EN ESE MOMENTO EL COMANDO QUEDA ALMACENADO EN LA PLANTILLA. PARA VOLVER A ENVIAR ESE MISMO COMANDO A MS-DOS, BASTA OPRIMIR DOS TECLAS:

<F3 ó COPY ALL> Y <RETURN>

AL OPRIMIR LA TECLA <F3 ó COPY ALL> EL CONTENIDO DE LA PLANTILLA ES COPIADO A LA LÍNEA DE COMANDO Y OPRIMIENDO LA TECLA <RETURN> LA LÍNEA DE COMANDO ES ENVIADA PARA SU EJECUCIÓN AL PROCESADOR DE COMANDOS DE MS-DOS

SI SE DESEA INFORMACIÓN ACERCA DEL ARCHIVO CON NOMBRE PROG.ASM, ES POSIBLE UTILIZAR EL CONTENIDO DE LA PLANTILLA, TECLEANDO:

<F2 ó COPY UP>C

DE ESTA FORMA Ó MS-DOS COPIA LOS CARACTERES DE LA PLANTILLA A LA LÍNEA DE COMANDO HASTA ENCONTRAR LA PRIMERA LETRA C. DESPLEGANDO EN LA PANTALLA LO SIGUIENTE:

A) DIR PROG._

NOTE EL CURSOR. AHORA TECLÉE ASM.
DANDO POR RESULTADO:

A) DIR PROG.ASM_

PARA SU EJECUCIÓN OPRIMIR < RETURN >

SUPÓNGASE QUE DESEA EJECUTAR EL SIGUIENTE COMANDO:

TYPE PROG.ASM.

PARA ÉSTO TECLEAR LO SIGUIENTE:

A>TYPE <INS> <F3=COPYALL>

A>TYPE PROG.ASM <RETURN>

NOTE QUE LOS CARACTERES "TYPE" REEMPLAZARON A LOS CARACTERES "DIR" ESTA CARACTERÍSTICA OCURRE CUANDO SE OPRIME LA TECLA DE <INS>. DESPUÉS PARA INSERTAR UN ESPACIO HAY QUE -- OPRIMIR EN UNA OCASIÓN LA BARRA ESPACIADORA . FINALMENTE PARA COPIAR EL RESTO DE LA PLANTILLA A LA LÍNEA DE COMANDO HAY QUE OPRIMIR LA TECLA <F3 = COPYALL> Y POR ÚLTIMO PARA ENVIAR A MS-DOS LA LÍNEA DE COMANDO HAY QUE OPRIMIR LA TECLA DE <RETURN>

UN CARACTER DE CONTROL TIENE UNA FUNCIÓN QUE AFECTA LA LÍNEA DE COMANDO. HAY QUE RECORDAR QUE CUANDO SE QUIERE CARGAR UN CARACTER DE CONTROL HAY QUE MANTENER OPRIMIDA LA TECLA DE --- <CONTROL> Y SUJETÁNDOLA OPRIMIR LA TECLA DE LA FUNCIÓN QUE SE DE SEA.

LA SIGUIENTE TABLA MUESTRA ALGUNAS DE LAS FUNCIONES DE LOS CARACTERES DE CONTROL.

CARÁCTER DE CONTROL	F U N C I Ó N
CONTROL-N	INTERRUMPE LA SALIDA A IMPRESIÓN.
CONTROL-C	ABORTA EL COMANDO QUE SE ENCUENTRA EJECUTANDO.
CONTROL-H	ELIMINA EL ÚLTIMO CARACTER DE LA LÍNEA DE COMANDO Y EL CARACTER DE LA PANTALLA.
CONTROL-Z	INSERTA UN CARACTER DE FIN DE ARCHIVO EN LA LÍNEA DE COMANDO.
CONTROL-P	INTERRUPTOR QUE HABILITA LA SALIDA A LA IMPRESORA.
CONTROL-S	SUSPENDE LOS MENSAJES ENVIADOS A LA PANTALLA Y OPRIMIENDO CUALQUIER TECLA CONTINÚA.

PRACTICA - 4

1.- DESPUÉS DE HABER CARGADO MS-DOS ENVIE Y COMENTARICE LO SIGUIENTE:

A) DIR <RETURN> <CTRL-C> _____

B) DIR <RETURN> <CTRL-S> _____

2.- REPETIR EL EJEMPLO DE LA PÁGINA 71,72 Y COMENTARIZAR.

MODULO 5

EL EDITOR (EDLIN)

EDLIN ES UN PROGRAMA DE UTILERÍA QUE REALIZA LA FUNCIÓN DE CREAR Y MODIFICAR ARCHIVOS QUE CONTIENEN CARACTERES ALFANUMÉRICOS O ESPECIALES.

LA SIGUIENTE TABLA MUESTRA ALGUNOS ARCHIVOS QUE ESTÁN DENTRO DE ESA CLASIFICACIÓN.

- 1 - PROGRAMAS EN ENSAMBLADOR.
- 2 - TEXTOS (CORRESPONDENCIA, MANUSCRITOS, FORMAS, MEMORÁNDUMS, OFICIOS, ETC.)
- 3 - PROGRAMAS EN BASIC.
- 4 - PROGRAMAS FUENTE DE LENGUAJES DE ALTO NIVEL (FORTRAN, PASCAL, ETC.)

LOS SIGUIENTES TIPOS DE ARCHIVOS NO SE PUEDEN MODIFICAR CON EL PROGRAMA EDLIN.

- 1 - CON NOMBRE DE EXTENSIÓN COM.
- 2 - GENERADOS POR UN PROGRAMA ENSAMBLADOR.

EDLIN EDITA LÍNEA POR LÍNEA UN PROGRAMA, PARA ÉSTO SE PUEDE UTILIZAR LAS TECLAS CON FUNCIONES ESPECIALES TENIENDO ÉSTAS LAS MISMAS FUNCIONES QUE LAS DE EDICIÓN DE COMANDOS MS-DOS EXPUESTAS EN EL MÓDULO ANTERIOR.

COMO INICIAR UNA SESION EN EDITOR

PARA EMPEZAR UNA SESIÓN DE EDITOR EDLIN BASTA TE-
CLEAR LO SIGUIENTE:

```
A>EDLIN <FILESPEC>
```

SI EL ARCHIVO ESPECIFICADO <FILESPEC> NO SE EN-
CUENTRA EN LOS DISKETTES QUE ESTÁN EN LAS UNIDADES DE DIS-
CO EDLIN CREA UN NUEVO ARCHIVO CON EL NOMBRE ESPECÍFICADO
Y EL SIGUIENTE MENSAJE ES ENVIADO A PANTALLA.

```
NEW FILE  
*  
—
```

NOTE QUE EL CARACTER DE PETCIÓN (PROMPT) DE
EDLIN ES UN ASTÉRISCO.

PARA INICIAR LA CARGA DE INFORMACIÓN, SE DEBE OPRI-
MIR LA TECLA <INS> QUE ES EL COMANDO PARA INSERTAR LÍNEAS.

SI SE DESEA EDITAR EN UN ARCHIVO QUE EXISTE, CUANDO EDLIN BUSCA EL ARCHIVO ESPECIFICADO EN LA UNIDAD DE DISCO SEÑALADA O EN LA DE OMISIÓN EL ARCHIVO ES ALMACENADO EN MEMORIA Y EDLIN MUESTRA EL SIGUIENTE MENSAJE EN LA PANTALLA:

```
END OF INPUT FILE
*
_
```

A PARTIR DE ESE MOMENTO SE PUEDE EDITAR UTILIZANDO LOS COMANDOS DE EDLIN.

SI EL ARCHIVO ES MUY GRANDE PARA SER ALMACENADO EN MEMORIA EDLIN GUARDA HASTA LAS 3/4 PARTES DE LA MEMORIA QUE SON LLENADAS Y ENSEGUIDA MUESTRA EL CARACTER DE PETICIÓN:

```
*
_
```

CUANDO SE TERMINA LA SESIÓN DE EDICIÓN SE PUEDE PROTEGER ESTA NUEVA VERSIÓN Y DEJAR INTACTA LA ANTERIOR UTILIZANDO EL COMANDO DE EDLIN END. EL ARCHIVO ORIGINAL TOMA EL MISMO NOMBRE SÓLO QUE CON NOMBRE DE EXTENSIÓN BAK. LOS COMANDOS DEL PROGRAMA EDLIN SERÁN EXPLICADOS CON MÁS DETALLE EN LAS PRÓXIMAS SECCIONES.

LA TECLAS CON FUNCIONES ESPECIALES ASÍ COMO EL CONCEPTO DE PLANTILLA QUE FUERON EXAMINADAS EN EL MÓDULO ANTERIOR SON LOS MISMOS PARA EL PROGRAMA EDLIN. EN ESTA SECCIÓN DE MANERA EJEMPLIFICADA, SE EXPLICARÁ CADA UNA DE ESTAS TECLAS, CONSIDERANDO QUE EN LA TABLA 4-1 SE ENCUENTRAN DEFINIDAS CADA UNA DE LAS FUNCIONES.

F1 ó COPY1

FUNCIÓN - COPIA UN CARACTER DE LA PLANTILLA A LA LÍNEA DE COMANDO.

EJEMPLO:

1:* ESTE ES UN EJEMPLO

<COPY1> 1:*E_

CADA VEZ QUE LA TECLA SE OPRIME LA TECLA F1=COPY1 APARECE UN CARACTER MAS EN LA PANTALLA

<COPY1> 1:*Es_

<COPY1> 1:*EST_

<COPY1> 1:*ESTE_

F4 - 0 SKIPUP

FUNCIÓN: SALTA EN LA PLANTILLA (NO COPIA) HASTA UN CARACTER ESPECIFICADO.

EJEMPLO:

```

1:* ESTE ES UN EJEMPLO
<SKIPUP> P1:*_
<COPYALL> 1:*?LO_

```

VOID

FUNCIÓN: LIMPIA LA LÍNEA DE EDICIÓN Y DEJA LA PLANTILLA SIN CAMBIO.

EJEMPLO:

```

1:* ESTE ES UN EJEMPLO
1:* ESTE ES OTRO TEXTO_
<VOID> 1:* ESTE ES OTRO TEXTO\
<RETURN> 1:*_
<COPYALL> 1:* ESTE ES UN EJEMPLO_

```

F2 ó COPYUP

FUNCIÓN: COPIA A LA LÍNEA DE EDICIÓN LOS CARACTERES DE LA PLANTILLA HASTA UN CARACTER EN ESPECÍFICO.

EJEMPLO:

1:*ESTE ES UN EJEMPLO
 <COPYUP> P 1:*ESTE ES UN EJEMPLO

F3 ó COPYALL

FUNCIÓN: COPIA A LA LÍNEA DE EDICIÓN EL RESTO DE LOS CARACTERES DE LA PLANTILLA.

EJEMPLO:

1:*ESTE ES UN EJEMPLO
 <COPYALL> 1:*ESTE ES UN EJEMPLO _

DEL ó SKIP1

FUNCIÓN: SALTA (NO COPIA) UN CARACTER EN LA PLANTILLA.

EJEMPLO:

1:*ESTE ES UN EJEMPLO
 <SKIP1> 1:_
 <COPYALL> 1:*STE ES UN EJEMPLO _

INSERT ó INS

FUNCIÓN: ENTRA/SALE AL MODO DE INSERCIÓN

EJEMPLO:

- 1:* ESTE ES UN SIMPLE COMENTARIO
- <COPYUP>C 1:* ESTE ES UN SIMPLE_
- <INS> EDITION 1:* ESTE ES UN SIMPLE ARCHIVO DE_
- <INS><COPYALL> 1:* ESTE ES UN SIMPLE ARCHIVO DE COMENTARIO_

PARA SALIR DE MODO DE INSERCIÓN BASTA OPRIMIR LA TECLA DE <INS>

EJEMPLO:

- 1:* ESTE ES UN SIMPLE ARCHIVO DE SALUDO
- <COPYUP>M 1:* ESTE ES UN SI_
- <INS> NCERO 1:* ESTE ES UN SINCERO_
- <SKIPUP> S 1:* ESTE ES UN SINCERO_
- <COPYALL> 1:* ESTE ES UN SINCERO SALUDO_

F5 ó NEWLINE

FUNCIÓN: CREA UNA NUEVA LÍNEA DE PLANTILLA.

EJEMPLO:

- <NEWLINE> 1:* ESTE ES UN SIMPLE EJEMPLO_

COMANDOS DE EDLIN

81

BÁSICAMENTE, HAY DOS MODOS DE FUNCIONAMIENTO AL USAR EDLIN. EL PRIMERO OCURRE AL DAR ALGÚN COMANDO A EDLIN EN RESPUESTA DE CARACTER DE PETICIÓN (*) SIN UN NÚMERO DE LÍNEA.

EL SEGUNDO MODO, EDLIN ACEPTA EL TEXTO QUE FORMARA PARTE DEL ARCHIVO QUE SE ESTÁ EDITANDO. COMO SE VIÓ EN LOS EJEMPLOS DE LA SECCIÓN ANTERIOR ESTE TIPO DE LÍNEA TIENE UN NÚMERO, DOS PUNTOS Y UN ASTERISCO.

EJEMPLO: 1:*_

CADA COMANDO EDLIN ESTÁ CONSTITUIDO POR UNA SOLA LETRA YA SEA MAYÚSCULA O MINÚSCULA, LA SIGUIENTE TABLA MUESTRA UN RESUMEN DE LOS COMANDOS DE EDLIN QUE A PARTIR DE LAS SIGUIENTES PÁGINAS SERÁN DESCRITOS CON MÁS DETALLE.

COMANDOFUNCION

LÍNEA

EDITA NÚM. DE LÍNEA

A

AÑADE LÍNEAS

C

COPIA LÍNEAS

D

ELIMINA LÍNEAS

E

FINALIZA LA EDICIÓN

I

INSERTA LÍNEAS

L

LISTA TEXTOS

M

MUEVE LÍNEAS

P

PÁGINA TEXTOS

Q

ABANDONA TEXTOS

R

REEMPLAZA LÍNEAS

S

BUSCA TEXTOS

T

TRANSFIERE TEXTOS

W

ESCRIBE LÍNEAS

LAS SIGUIENTES PÁGINAS DESCRIBEN LOS COMANDOS DE
EDICIÓN DE EDLIN :

I INSERT

DESCRIPCIÓN: INSERTA TEXTOS ANTES DE LA LÍNEA ESPECIFI-
CADA. PARA SALIR DE MODO DE INSERCIÓN TE=
CLEAR <CONTROL-Z >

NOTACIÓN: [<LÍNEAS>]

EJEMPLO:

- 1: ESTE ES UN EJEMPLO
- 2: QUE MUESTRA EL COMANDO
- 3: DE INSERCIÓN
- 4: PARA EDITAR TEXTOS
- 5: EN UN ARCHIVO

SI SE TECLEA:

5I

EDLIN CONTESTA

5: _

 AHORA SE TECLEA EL TEXTO EN LA LÍNEA 5:

5: Y RENUMERA LÍNEAS

PARA SALIR DEL MODO DE INSERCIÓN OPRIMIR

<CONTROL-Z >

8 OBSERVANDO EL ARCHIVO DESPUÉS DE TECLEAR EL COMANDO
LIST (L) QUEDA CON LA SIGUIENTE INFORMACIÓN:

- 1: ESTE ES UN EJEMPLO.
- 2: QUE MUESTRA EL COMANDO
- 3: DE INSERCIÓN
- 4: PARA EDITAR TEXTOS
- 5: Y RENUMERAR LÍNEAS
- 6: EN UN ARCHIVO

L LIST

DESCRIPCIÓN: LISTA UN GRUPO DE LÍNEAS, INCLUYENDO LAS
DOS LÍNEAS ESPECIFICADAS.

NOTACIÓN: [<LÍNEA>] [, <LÍNEA>] L

EJEMPLO: TOMANDO EL EJEMPLO ANTERIOR
2,4L

RESULTADO: 2: QUE MUESTRA EL COMANDO
3: DE INSERCIÓN
4: PARA EDITAR TEXTOS

000116

S SEARCH

DESCRIPCIÓN: BUSCA EN UN RANGO DE LÍNEAS UNA CADENA ESPECÍFICA DE TEXTO.

NOTACIÓN: - [<LÍNEA>] [, <LÍNEA>] [?] <CADENA> <RETURN>

EJEMPLO: CONTINUANDO CON NUESTRO TEXTO DE EJEMPLO:

2,4SLÍNEAS <RETURN>

RESULTADO: 5: Y RENUMERA LÍNEAS.

R REPLACE

DESCRIPCIÓN: SUBSTITUYE TODAS LAS OCURRENCIAS DE UNA CADENA DE TEXTO EN UN RANGO DE LÍNEAS, POR OTRA CADENA DE TEXTO DIFERENTE.

NOTACIÓN: - [<LÍNEA>] [, <LÍNEA>] [?]R<CADENA-1>
<CONTROL-Z> <CADENA 2> <RETURN>

EJEMPLO: 2,5R MUESTRA <CONTROL-Z> EXHIBE <RETURN>

RESULTADO: 2: QUE EXHIBE EL COMANDO

E END

DESCRIPCIÓN: FINALIZA LA SESIÓN DE EDICIÓN.

NOTACIÓN: E

EJEMPLO: E<RETURN>

Q QUIT

DESCRIPCIÓN: ABANDONA LA SESIÓN DE EDICIÓN Y ADEMÁS NO ES REALIZADO NINGÚN CAMBIO DE EDICIÓN.

NOTACIÓN: Q

EJEMPLO: Q<RETURN>
ABORT EDIT (Y/N)?Y<RETURN>
A)_

W WRITE

DESCRIPCIÓN: ESCRIBE UN NÚMERO ESPECIFICADO DE LÍNEAS DEL DISKETTE A LAS LÍNEAS QUE ESTÁN SIENDO EDITADAS EN MEMORIA. LAS LÍNEAS SON ESCRITAS DEL DISKETTE INICIANDO CON LA LÍNEA 1.

NOTACIÓN: [<N>]W

A - APPEN

DESCRIPCIÓN: AÑADE UN NÚMERO ESPECÍFICO DE LÍNEAS DE UN ARCHIVO EN DISKETTE AL ARCHIVO QUE SE ESTÁ EDITANDO EN MEMORIA. LAS LÍNEAS SON AÑADIDAS AL FINAL DE LAS LÍNEAS QUE EN ESE MOMENTO SE ENCUENTRAN EN MEMORIA.

NOTACIÓN: [<N>]A

C COPY

DESCRIPCIÓN: COPIA UN RANGO DE LÍNEAS A UN NÚMERO ESPECÍFICO DE LÍNEA. LAS LÍNEAS SERÁN COPIADAS TANTAS VECES COMO SE INDIQUE CON LA OPCIÓN COUNT.

NOTACIÓN: [<LÍNEA>],[<LÍNEA>], <LÍNEA>, [<COUNT>]C

EJEMPLO: 1,6,7C

D DELETE

DESCRIPCIÓN: ELIMINA UN RANGO DE LÍNEAS EN UN ARCHIVO.

NOTACIÓN: [<LÍNEA>][,<LÍNEA>]D

EJEMPLO: 2,4D

EDIT

DESCRIPCIÓN: EDITA UNA LÍNEA DE TEXTO.

NOTACIÓN: [<LÍNEA>]

EJEMPLO:

- 1: ESTE ARCHIVO ES USADO
- 2: PARA MOSTRAR LA
- 3: EDICIÓN DE LA LÍNEA
- 4: CUATRO

TECLEAR 4

RESULTADO: 4:*_

<INS> NÚMERO<CTR-Z>4: NÚMERO_
 <COPYALL> <RETURN> 4: NÚMERO CUATRO
 5:*_

MOVE

DESCRIPCIÓN: MUEVE UN RANGO DE LÍNEAS A UNA LÍNEA ESPECIFICADA.

NOTACIÓN. [<LÍNEA>],[<LÍNEA>],[<LÍNEA>]M

EJEMPLO: PARA MOVER LA LÍNEA 20 A LA 30 EN LA 100
 TECLEAR: 20,30,100M

CUESTIONARIO - 5

1.- ¿PARA INICIAR UNA SESIÓN CON EL PROGRAMA EDLIN QUE COMANDO HAY QUE ENVIAR A MS-DOS?

2.- ES POSIBLE EDITAR POR MEDIO DE EDLIN UN ARCHIVO QUE OCUPA MÁS DE LAS 3/4 PARTES DE LA MEMORIA.

V () F ()

3.- PARA SALIR DE MODO DE INSERCIÓN HAY QUE TECLAR CONTROL-S

V () F ()

4.- ¿QUÉ REALIZAN LOS SIGUIENTES COMANDOS DE EDLIN?

A) 2,8L _____

B) 5L _____

C) 2,10SHOLA _____

D) 5HOLA _____

E) 2,5RMUESTRA<CONTROL-Z>EXHIBE RETURN _____

F) E _____

G) Q _____

H) 1,6,7C _____

I) 2,4D _____

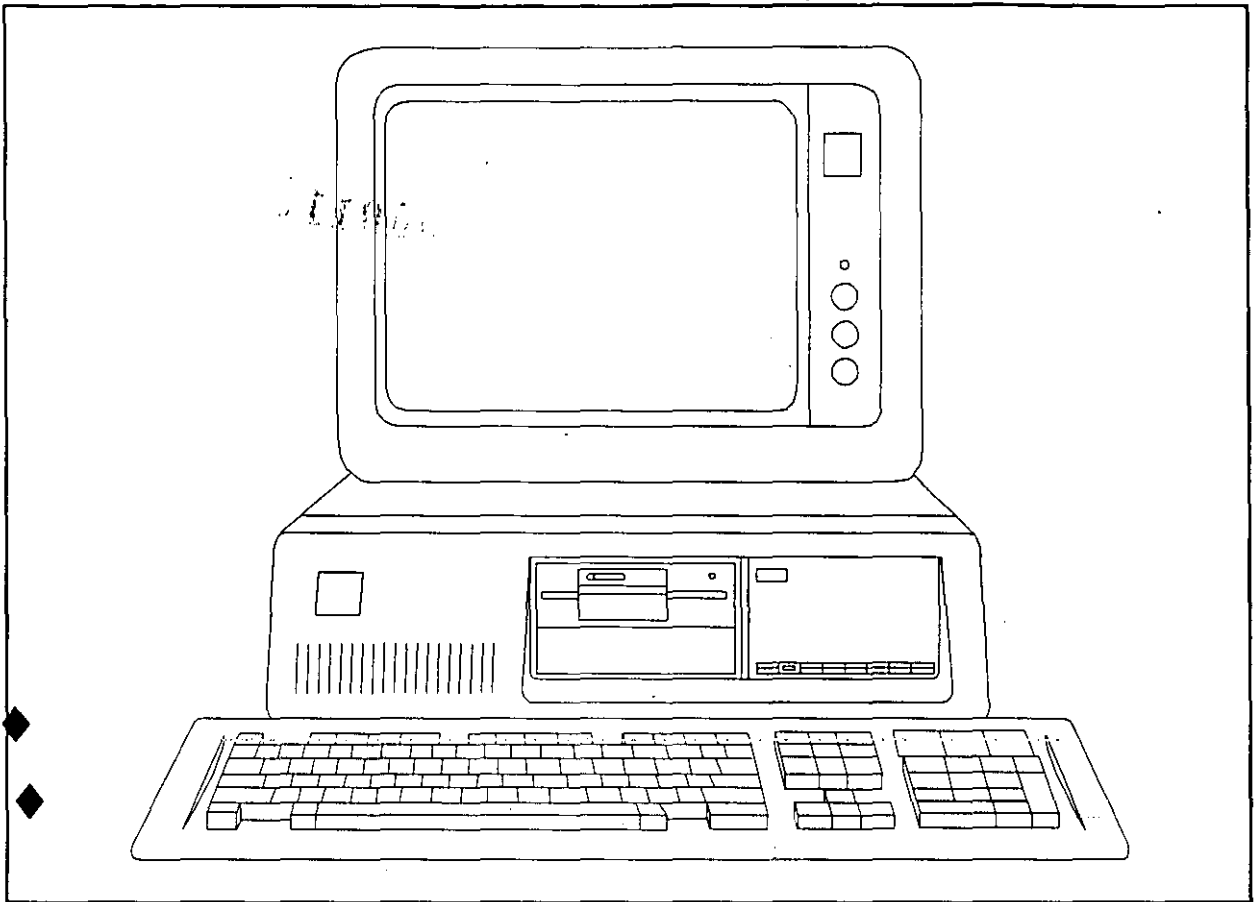
J) 4 _____

K) 20,30,100M _____

L) 54,75P _____

M) 25TRAIZ.LIB _____

5.- ¿CUÁL ES EL CARACTER DE PETICIÓN EN EDLIN?



INGENIERIA DE SOFTWARE

AREAS DE APLICACION DE LOS LENGUAJES DE PROGRAMACION

CIENTIFICA

Area encargada de resolver los problemas científicos, casi siempre relacionados con cálculos complicados, funciones trascendentes, vectores y matrices de varias dimensiones, números complejos, etc. La cantidad de datos generalmente es poca, pero la cantidad de operaciones aplicadas sobre ellos es elevada. Se cuenta con muchas facilidades para realizar operaciones y con pocas para la presentación ó formateo de los resultados en reportes impresos. Sus usuarios son científicos e ingenieros encargados de resolver problemas de sus áreas respectivas.

ADMINISTRATIVA

Area encargada de los sistemas administrativos en general, como serían la nómina, contabilidad, inventarios, control de producción, etc. Generalmente la cantidad de datos es elevada, mientras que las operaciones que se efectúan sobre ellos son escasas. Tiene gran importancia la confiabilidad y la protección de la información. Se cuenta con amplias facilidades para el manejo de archivos de datos y para el formateo de la información para su presentación en reportes impresos. Sus usuarios son las personas que están a cargo de los sistemas administrativos, como serían los contadores, jefes de ciertas áreas (personal, finanzas, producción, etc.) y administradores en general.

SISTEMAS

Area encargada de los programas especializados requeridos para el funcionamiento normal de la computadora, como serían los sistemas operativos, ensambladores, compiladores e intérpretes de los diversos lenguajes de programación, ligadores, etc. Esta área debe tener acceso a operaciones poderosas y poco usuales de la computadora, así como a información importante normalmente inaccesible a otras áreas. Es común manipular cantidades no numéricas (texto) que representan la información central de esta área. Las personas encargadas de esta área son expertos en computación, generalmente especializados en ciertos modelos ó marcas de computadoras, que trabajan con los fabricantes de ellas ó en casas especializadas de software.

ENSEÑANZA

Area destinada a la enseñanza del uso de computadoras en general, y a la enseñanza de los lenguajes de programación en particular. Los lenguajes orientados a esta área generalmente cuentan con ejecución inmediata de instrucciones, generación de gráficas y sonidos en forma sencilla, ayudas y detección de errores en forma extensiva, etc.; pero por la misma razón es frecuente que se carezca de mayores facilidades y poderío para un posterior desarrollo de sistemas a gran escala. Los usuarios de esta área son los alumnos involucrados en el aprendizaje de la programación de computadoras en general.

INTELIGENCIA ARTIFICIAL

Area destinada a manipular ciertos problemas con características especiales, que por su propia naturaleza pueden ser resueltos mediante computadora a través de técnicas sofisticadas, consistentes en deducir conclusiones en base a análisis de situaciones previas, como sería el caso de los juegos de salón clásicos (ajedrez, damas, etc.), ó de los llamados "sistemas expertos", los cuales alcanzan un objetivo mediante el seguimiento de una serie de pasos metódicamente establecidos (p.e., un diagnóstico preliminar dado por un médico en base a preguntas, sin análisis clínicos). Los lenguajes orientados a esta área generalmente son muy poco usuales en su estructura y manejo. Esta área en particular está limitada por la tecnología del equipo, ya que en muchos problemas de este tipo se requieren computadoras mas rápidas que las existentes, ó dispositivos de detección ó control que aún no se han diseñado.

CONTROL DE PROCESOS

Area destinada a controlar la ejecución de procesos en la computadora de tal forma que una serie de situaciones involucradas en la ejecución de tareas puedan ser detectadas por la computadora en forma automática y actuar en consecuencia. Por ejemplo: "si es viernes, ejecute la lista de raya", "si existen todos los archivos necesarios, ejecute el proceso; de lo contrario, copie los archivos faltantes", "si la hora actual es menor a las 12:00, muestre BUENOS DIAS; si es menor a las 19:00, muestre BUENAS TARDES; de lo contrario, muestre BUENAS NOCHES". Los usuarios de esta área son los operadores de la computadora encargados del proceso de los sistemas computarizados.

BREVE DESCRIPCION DE ALGUNOS LENGUAJES DE PROGRAMACION

1 SYMPLE

00124

1.1 Significado del nombre

SYMBOLIC Programming Language.
Lenguaje de programación simbólico.

1.2 Area de aplicación

Enseñanza. General de bajo nivel.

1.3 Algunas ventajas

Muy fácil de aprender. Es independiente del idioma inglés.

1.4 Algunos inconvenientes

Limitado para el desarrollo de sistemas grandes. El mantenimiento de programas es algo difícil.

1.5 Opinión general

Muy adecuado como una introducción previa a otros lenguajes estructurados.

2 BASIC

2.1 Significado del nombre

Begginer's All-purpose Symbolic Instruction Code.
Código de instrucciones simbólicas de propósito general para principiantes.

2.2 Area de aplicación

Enseñanza. General de nivel medio.

2.3 Algunas ventajas

Facilidad en el desarrollo de programas y en la corrección de errores. (Ventajas generales de todo lenguaje intérprete.)

000125

2.4 Algunos inconvenientes

Propicia hábitos de programación inadecuados. No es un lenguaje estructurado.

2.5 Opinión general

Adecuado para desarrollos de baja escala.

3 FORTRAN

3.1 Significado del nombre

FORMula TRANslation.
Traducción de fórmulas.

3.2 Area de aplicación

Científica.

3.3 Algunas ventajas

Grandes facilidades matemáticas. Amplísima difusión.

3.4 Algunos inconvenientes

Lenguaje anticuado "ajustado" a las normas modernas.

3.5 Opinión general

No es recomendable para nuevos desarrollos; sí lo es para utilizar la infraestructura ya existente.

4 COBOL

4.1 Significado del nombre

COmmon Business Oriented Language.
Lenguaje común orientado a los negocios.

4.2 Area de aplicación

Administrativa.

4.3 Algunas ventajas

Muy amplia difusión.

4.4 Algunos inconvenientes

Lenguaje anticuado y engorroso.

4.5 Opinión general

Adecuado en el medio de sistemas ya existentes en computadoras grandes. Para nuevos desarrollos, es preferible un lenguaje de cuarta generación.

5 Pascal

5.1 Significado del nombre

En honor del matemático del siglo XVII Blaise Pascal.

5.2 Area de aplicación

Enseñaza. General de gran escala.

5.3 Algunas ventajas

Completamente estructurado y modular. Extensas facilidades para el mantenimiento de sistemas.

5.4 Algunos inconvenientes

Algo limitado para ciertas aplicaciones avanzadas.

5.5 Opinión general

Excelente lenguaje.

6 C

6.1 Significado del nombre

Histórico: BCPL -> B -> C.

6.2 Area de aplicación

General de gran escala. Sistemas.

6.3 Algunas ventajas

Completamente estructurado y modular. Muy poderoso.

6.4 Algunos inconvenientes

Programas algo oscuros y crípticos. Un poco de dificultades para el mantenimiento.

6.5 Opinión general

Muy adecuado para todo tipo de desarrollos, en particular en microcomputadoras.

"EL USO DE LAS TECNICAS ESTRUCTURADAS PARA EL ANALISIS,
EL DISEÑO Y LA ELABORACION DE SISTEMAS DE INFORMACION"

CARLOS A. RAMOS LARIOS

OBJETIVO DE LA PLATICA:

PRESENTAR LOS PRINCIPALES CONCEPTOS RELACIONADOS CON LAS TECNICAS ESTRUCTURADAS DEL DESARROLLO DE SISTEMAS, ASI COMO SUS PRINCIPALES VENTAJAS Y LIMITACIONES.

MECANICA DE LA PRESENTACION:

EXPOSICION DE LOS TEMAS

PREGUNTAS, RESPUESTAS Y COMENTARIOS DURANTE LA EXPOSICION

TEMARIO:

- I) INTRODUCCION A LA INGENIERIA DE SOFTWARE
- II) EL ANALISIS ESTRUCTURADO
- III) EL DISEÑO ESTRUCTURADO
- IV) LA PROGRAMACION ESTRUCTURADA
- V) EL DESARROLLO DESCENDENTE
- VI) OTRAS TECNICAS ESTRUCTURADAS
- VII) CONCLUSIONES
- VIII) COMENTARIOS SOBRE LA BIBLIOGRAFIA DISPONIBLE

I) INTRODUCCION

PREGUNTAS Y RESPUESTAS INICIALES

¿QUE SON LAS TECNICAS ESTRUCTURADAS DEL DESARROLLO DE SISTEMAS?

UN CONJUNTO DE PROCEDIMIENTOS Y CONCEPTOS QUE, EN GENERAL, HAN DEMOSTRADO LOGRAR UN 100% DE AUMENTO TANTO EN LA PRODUCTIVIDAD COMO EN EFICACIA DE UN DEPARTAMENTO DE PROCESAMIENTO DE DATOS.

¿POR QUE SURGIO LA "REVOLUCION DE LAS TECNICAS ESTRUCTURADAS"?

DEBIDO AL SURGIMIENTO DE UNA PROBLEMATICA GENERALIZADA PRACTICAMENTE A TODAS LAS ORGANIZACIONES RELACIONADAS CON EL PROCESAMIENTO DE DATOS EN TODO EL MUNDO, EN DONDE SE REFLEJABA EL CONSTANTE AUMENTO EN EL COSTO DEL SOFTWARE Y SOBRE TODO AL COMPARAR LO CON EL COSTO DEL HARDWARE

"LA CRISIS DEL SOFTWARE", COMO SE LE CONOCIO Y RECONOCIO DURANTE LA DECADA DE LOS 70'S, HIZO QUE LA COMUNIDAD INFORMATICA VOLVIERA LOS OJOS HACIA LA NECESIDAD DE SENTAR LAS BASES PARA LO QUE HOY EN DIA SE CONOCE COMO LA "INGENIERIA DE SOFTWARE"

¿CUALES SON LOS OBJETIVOS ESPECIFICOS DE LA INGENIERIA DE SOFTWARE?

- CONTAR CON LA METODOLOGIA BIEN DEFINIDA QUE CONSIDERE EL CICLO DE VIDA DEL SOFTWARE: PLANEACION, ELABORACION Y MANTENIMIENTO.
- ESTABLECER UN CONJUNTO DE "COMPONENTES DE SOFTWARE", QUE DOCUMENTEN CADA PASO DEL CICLO DE VIDA Y PERMITAN SU SEGUIMIENTO PASO A PASO.
- DEFINIR UN CONJUNTO DE PUNTOS DE CONTROL PREDECIBLES, LOS CUALES PUEDAN SER REVISADOS A INTERVALOS REGULARES A LO LARGO DEL CICLO DE VIDA DEL SOFTWARE.

¿COMO SE RELACIONAN LA INGENIERIA DE SOFTWARE Y LAS TECNICAS ESTRUCTURADAS?

LAS TECNICAS ESTRUCTURADAS SON EL MEDIO ACTUAL A TRAVES DEL CUAL LA INGENIERIA DE SOFTWARE PUEDE CUMPLIR CON SUS OBJETIVOS.

LAS TECNICAS ESTRUCTURADAS SE MEJORAN CON EL TIEMPO, SE ADAPTAN A NECESIDADES ESPECIFICAS, SE ADOPTAN TOTAL O PARCIALMENTE EN PROYECTOS ESPECIFICOS: SON DINAMICAS

LA INGENIERIA DE SOFTWARE ES EL RECONOCIMIENTO A LA NECESIDAD DE VER A LA PRODUCCION DE SOFTWARE COMO UNA DISCIPLINA COMPARABLE A CUALQUIER OTRA RAMA DEL CONOCIMIENTO TECNOLOGICO: -- ES UN PRINCIPIO ESTATICO

¿ABARCAN LAS TECNICAS ESTRUCTURADAS TODOS LOS PROBLEMAS DE LA INFORMATICA?

LA I. DE S. ES SOLO PARTE DE UNA ACTIVIDAD MAS GENERAL:

"LA INGENIERIA DE SISTEMAS COMPUTACIONALES", LA CUAL ES UNA ESPECIALIDAD ORIENTADA HACIA LA SOLUCION DE UN PROBLEMA EN SU TOTALIDAD, MEDIANTE EL USO DE LA COMPUTADORA.

LA I. DE S.C. ESTUDIA, EVALUA Y DEFINE FUNCIONES ESPECIFICAS, AUNQUE GENERICAS, PARA LOS DOS COMPONENTES BASICOS DE UN SISTEMA DE INFORMACION: EL HARDWARE Y EL SOFTWARE.

EL CICLO DE VIDA DEL SOFTWARELA FASE DE PLANEACION DEL SOFTWARE:

DEBE EFECTUARSE POSTERIORMENTE O SIMULTANEAMENTE CON LA DEFINICION DEL SISTEMA (I.DES.C.)

DEBE GENERARSE:

- UNA DEFINICION DE ALCANCES, ESTIMACION DE COSTOS Y TIEMPOS
- UNA ESPECIFICACION DE LOS REQUERIMIENTOS DEL SOFTWARE
(¿QUE SE ESPERA QUE EL SOFTWARE REALICE? = ANALISIS)

LA FASE DE LA ELABORACION DEL SOFTWARE

- DEBE EFECTUARSE POSTERIORMENTE AL ANALISIS DE REQUERIMIENTOS
- TRADUCE UN CONJUNTO DE REQUERIMIENTOS A ELEMENTOS INTEGRANTES DEL SISTEMA, MUCHO ANTES DE SENTARSE FRENTE A UNA TERMINAL
(¿COMO VA A FUNCIONAR EL SISTEMA? = DISEÑO)
- PRODUCE UN PROGRAMA DE COMPUTADORA PARA LLEVAR A CABO CADA UNA DE LAS FUNCIONES DE LOS ELEMENTOS INTEGRANTES DEL SISTEMA
(ELABORACION DE PROGRAMAS)

LA FASE DE MANTENIMIENTO DEL SOFTWARE

- INVARIABLEMENTE EL SOFTWARE REQUERIRA MANTENIMIENTO
- LA FASE DE MANTENIMIENTO INICIA AUN ANTES QUE LA ELABORACION DEL SOFTWARE, RECONOCIENDO QUE EXISTIRA SIEMPRE
- EL MANTENIMIENTO DEBE CONTEMPLAR QUE LOS CAMBIOS AFECTAN NO SOLO A LOS PROGRAMAS SINO A LOS DOCUMENTOS GENERADOS DURANTE EL ANALISIS Y EL DISEÑO.

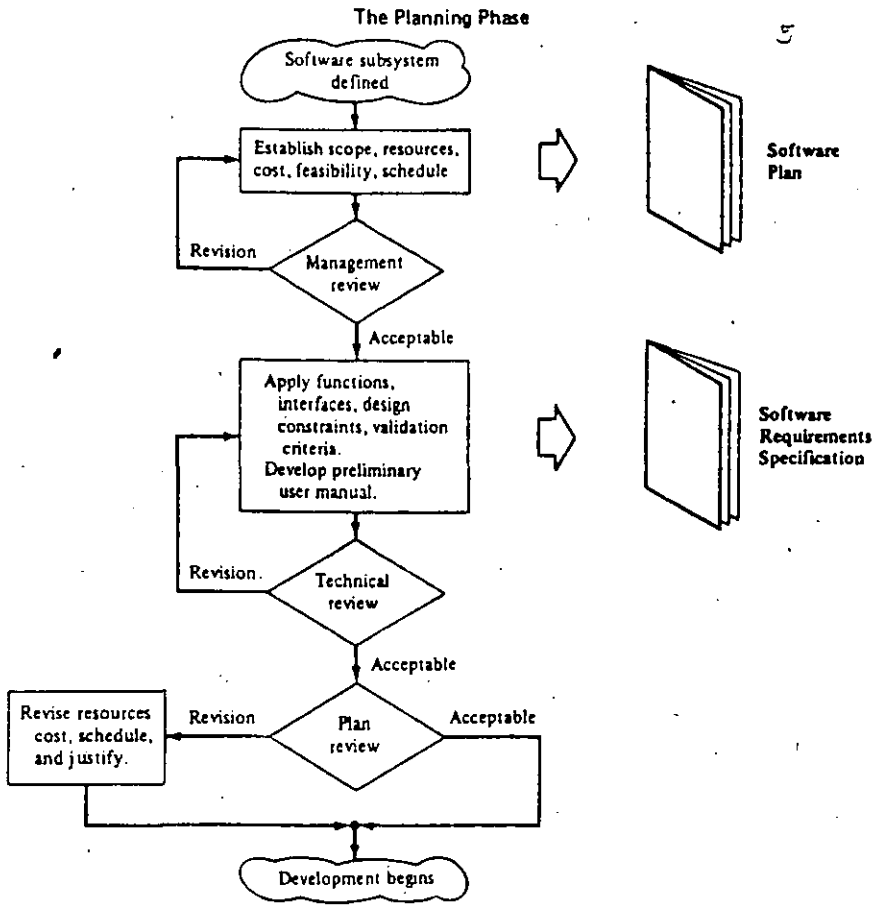


Figure 1.9

The Development Phase

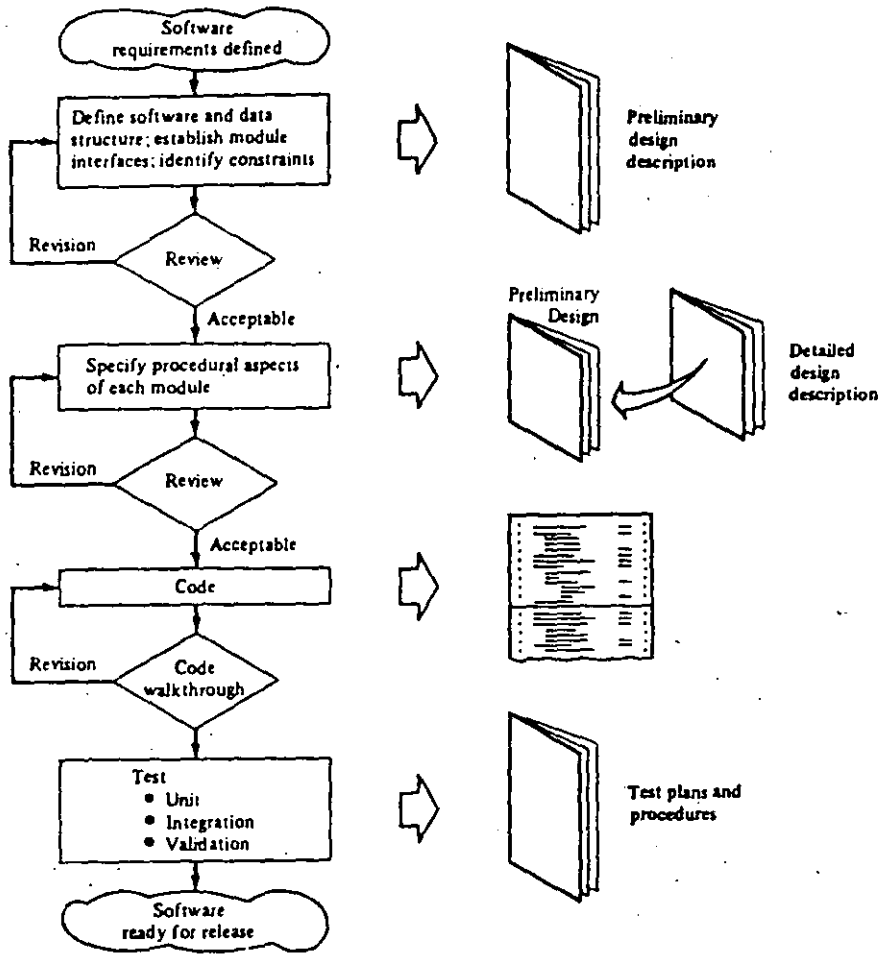


Figure 1.10

SC 11-10

The Maintenance Phase

7

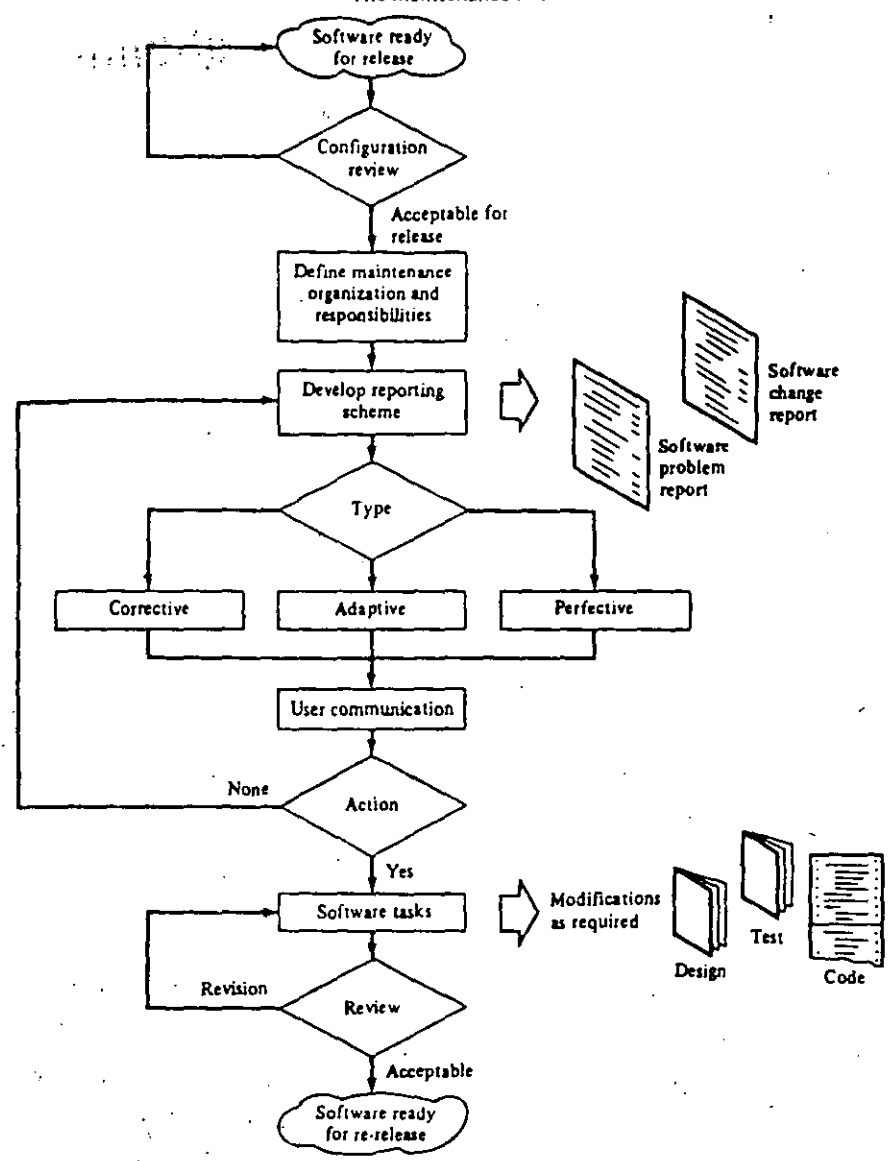


Figure 1.11

II). ANALISIS ESTRUCTURADO

EL ANALISIS SE UBICA EN EL INICIO DE LA FASE DE PLANEACION, CUANDO SE DESEAN DEFINIR LOS REQUERIMIENTOS DEL USUARIO. DEBE RESPONDER A LA PREGUNTA: ¿QUE SE REQUIERE HACER?

DURANTE MUCHO TIEMPO, EL ANALISIS SE HA BASADO EN LA PREPARACION DE UN DOCUMENTO NARRATIVO DE LAS NECESIDADES, EL CUAL PRESENTA LAS SIGUIENTES DIFICULTADES:

- AL SER MONOLITICO, DEBE LEERSE DE PRINCIPIO A FIN, AUN CUANDO SE NECESITE SOLO PARTE DE LA INFORMACION.
- EN GENERAL RESULTA REDUNDANTE, SIN AYUDA DE UN INDICE CRUZADO DE TEMAS AFINES.
- ES DIFICIL DE MODIFICAR Y POR LO TANTO DIFICIL DE RECIBIR RETROALIMENTACION.
- MUCHAS OCASIONES RESULTA UNA DESCRIPCION FISICA MAS QUE LOGICA. (HAY MAS DEL "COMO" QUE DEL "QUE").
- NO ES UN DOCUMENTO QUE PERMITA CONTINUAR FACILMENTE CON LAS SIGUIENTES ACTIVIDADES DEL DESARROLLO DEL SISTEMA (DISEÑO).

481540

IMPORTANCIA DEL ANALISIS

710820

EL ANALISIS ES DE LAS PRIMERAS ACTIVIDADES QUE SE LLEVAN A CABO DENTRO DEL CICLO DE VIDA DE UN SISTEMA, POR LO QUE SI UN REQUERIMIENTO QUEDA MAL DEFINIDO, PUEDE INVALIDAR EL DISEÑO, LA ELABORACION, LA DOCUMENTACION, LAS PRUEBAS Y AUN LOS RESULTADOS FINALES DE UN SISTEMA.

*10

DIFICULTADES INHERENTES AL ANALISIS:

EL ANALISIS ESTABLECE UN PROBLEMA DE COMUNICACION ENTRE EL USUARIO Y EL ANALISTA EN EL CUAL:

- EL USUARIO CONOCE SU PROBLEMÁTICA, PERO NO CONOCE SUFICIENTE EDP.
- EL ANALISTA CONOCE EDP, PERO NO CONOCE SUFICIENTE EL PROBLEMA
- EL ANALISTA RECIBE GRAN CANTIDAD DE DETALLES DESDE UN PRINCIPIO, MAS QUE UNA DESCRIPCION GENERAL COMPLEMENTADA CON LOS DETALLES.

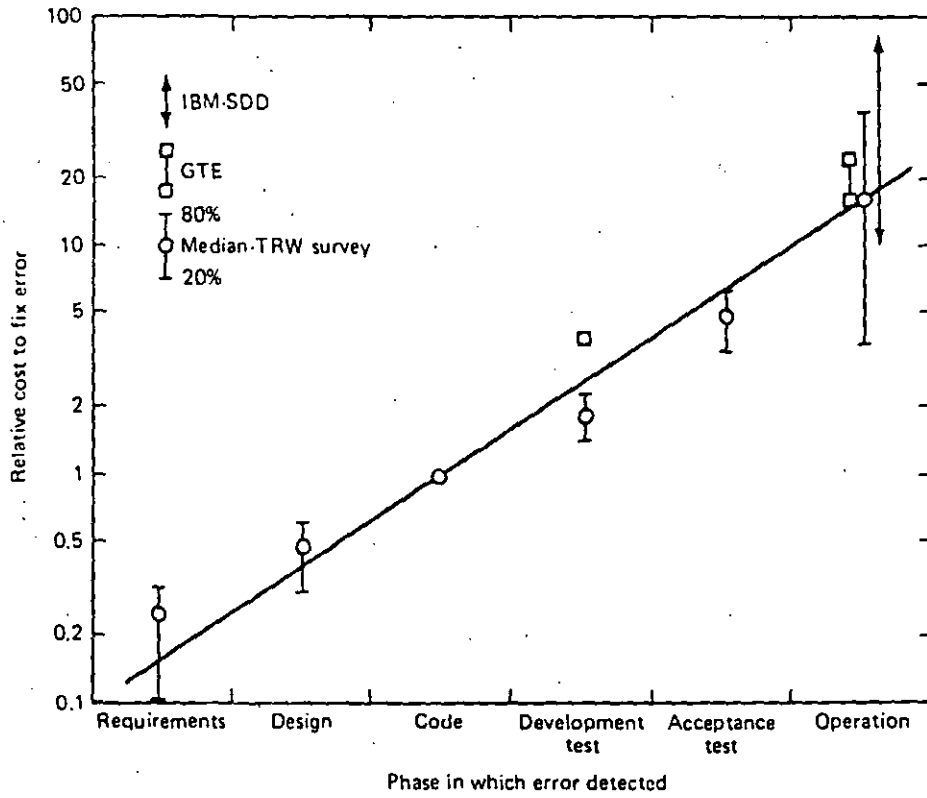


Figure 1.2 Relative cost to fix an error during system development

EL ANALISIS ESTRUCTURADO ES UNA TECNICA PARA DOCUMENTAR, EN FORMA GRAFICA, LOS REQUERIMIENTOS DE UN SISTEMA. ES UN MODELO GRAFICO - QUE DESCRIBE EL FLUJO DE INFORMACION A TRAVES DEL SISTEMA (DESCRIBE QUE HACE EL SISTEMA).

LAS HERRAMIENTAS DEL ANALISIS ESTRUCTURADO:

- DIAGRAMA DE FLUJO DE DATOS
- DICCIONARIO DE DATOS
- ESPANOL ESTRUCTURADO (PSEUDOCODIGO)

PRIMER PASO: TRAZO DE UN DIAGRAMA DE FLUJO DE DATOS A NIVEL GENERAL

- FUENTES Y DESTINOS DE LOS DATOS
- FLUJOS DE DATOS
- PROCESOS QUE TRANSFORMAN DATOS
- ALMACENAMIENTOS DE DATOS

*12

SEGUNDO PASO: CONSTRUCCION DE UN DICCIONARIO DE DATOS CON LOS DETALLES DE:

- ELEMENTOS DE DATOS
- ESTRUCTURAS DE DATOS
- FLUJOS DE DATOS
- ALMACENAMIENTOS DE DATOS
- PROCESOS (ESPANOL ESTRUCTURADO, ARBOLES, TABLAS,
- OTROS DETALLES (FUENTES Y DESTINOS, DEFINICIONES, SINONIMOS)

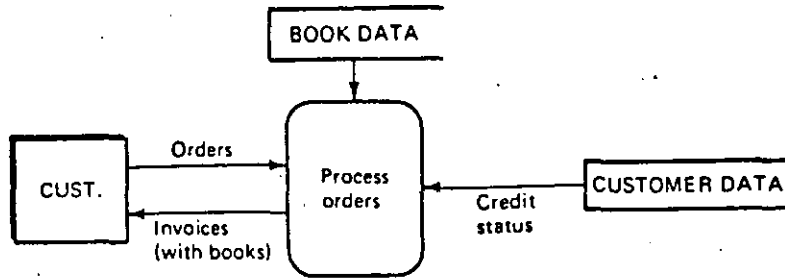


Figure 2.1 Logical data flow diagram

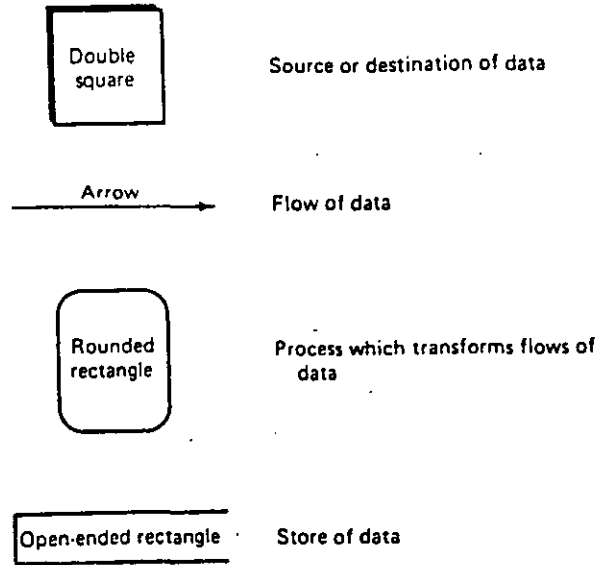


Figure 2.2 DFD symbols

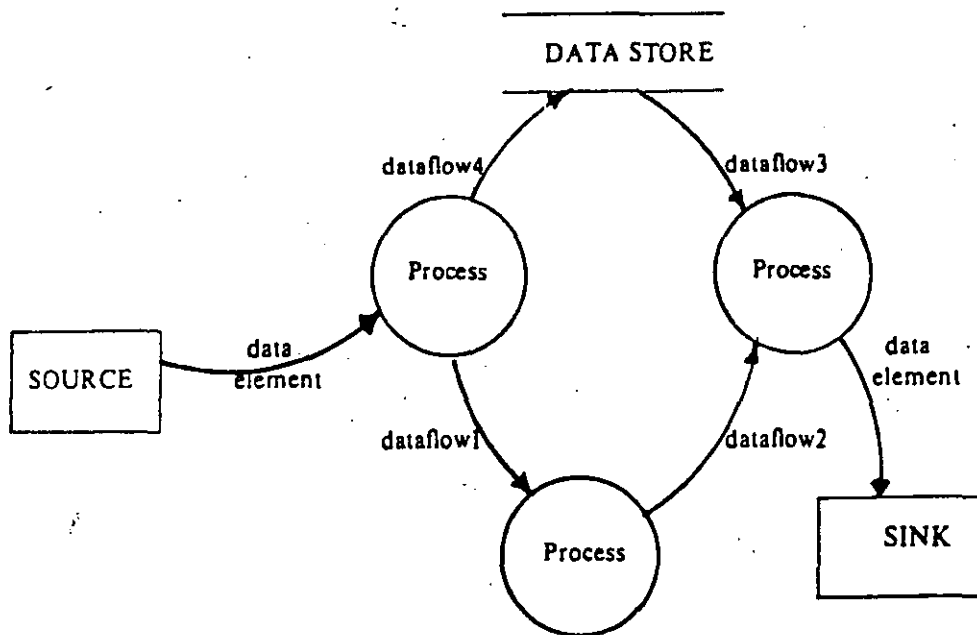


Figure 3.1. The elements of a data flow diagram.

ELEMENTO

13

NOMBRE :

DESCRIPCION :

TIPO A AN N

ALIAS :

SI ES DISCRETA

SI ES CONTINUA

VALOR

SIGNIFICADO

RANGO DE VALORES:

VALOR TIPICO:

LONGITUD:

REPRESENTACION INTERNA:

EN CASO DE MAS VALORES DAR REFERENCIA

INFORMACION ADICIONAL

ELEMENTOS O ESTRUCTURAS DE DATOS RELACIONADOS

000140

N O M B R E . : _____

DESCRIPCION : _____

FLUJO/ESTRUCTURAS DE DATOS RELACIONADOS:

VOLUMEN DE INFORMACION:

000141

5
000142

NOMBRE : _____

ORIGEN : _____ DESCRIPCION : _____

DESTINO: _____ DESCRIPCION : _____

DESCRIPCION _____

ESTRUCTURAS DE DATOS INCLUIDAS:

VOLUMEN DE INFORMACION:

000143

NOMBRE : _____

DESCRIPCION : _____

FLUJOS DE DATOS QUE ENTRAN:

FLUJOS DE DATOS QUE SALEN:

CONTENIDO:

ANALISIS DE ACCESO INMEDIATO EN:

ORGANIZACION FISICA:

NOMBRE : _____

DESCRIPCION : _____

ENTRADAS	RESUMEN DE LOGICA	SALIDAS

REFERENCIA FISICA : _____

DETALLES COMPLETOS DE LOGICA EN : _____

000144

TERCER PASO: EXPLISION DE LOS PROCESOS DEL NIVEL GENERAL

- REPETIR LA EXPLOSION UTILIZANDO LOS PASOS 1 Y 2
- TERMINAR HASTA EL NIVEL DE DETALLE DESEADO

*19

CUARTO PASO: DEFINICION DE LA FRONTERA DE IMPLANTACION

- PARA DESARROLLO PASO A PASO
- PARA CAMBIAR PROCESOS MANUALES CON AUTOMATICOS

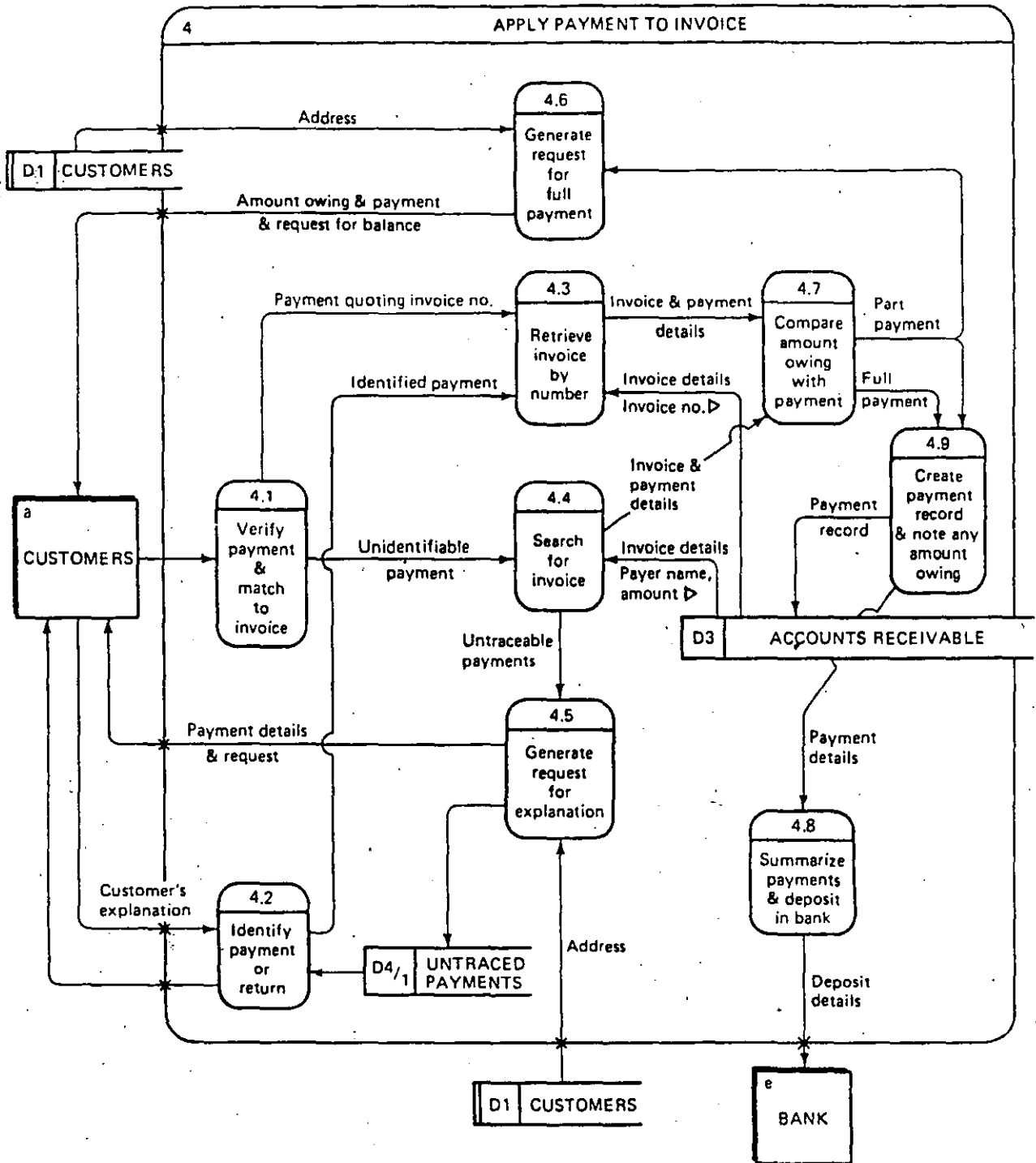


Figure 3.14 Process explosion

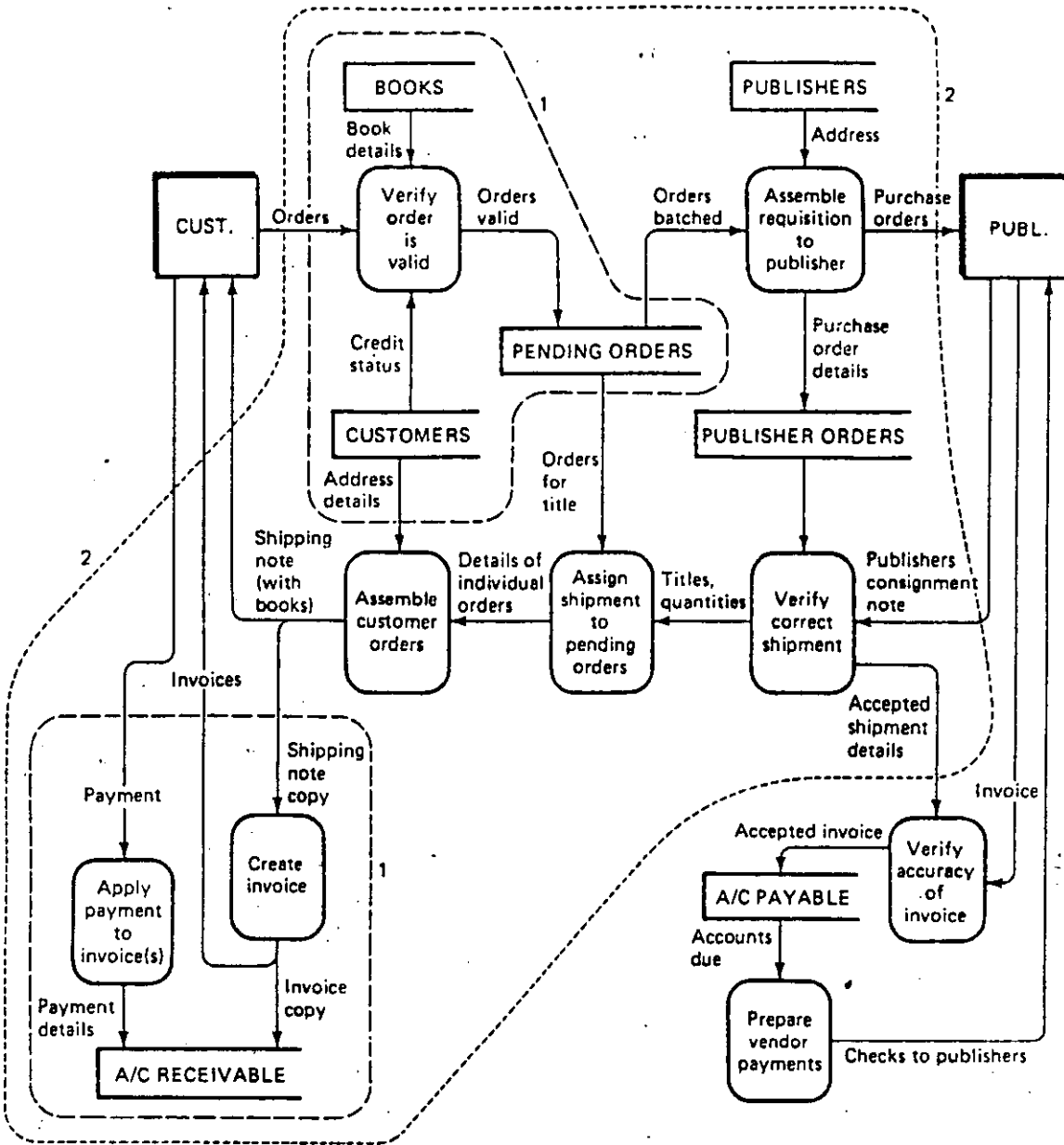


Figure 2.7 Possible automation boundaries

IIII) EL DISEÑO ESTRUCTURADO

EL DISEÑO SE UBICA EN EL INICIO DE LA FASE DE ELABORACION, UNA VEZ QUE SE HA TERMINADO EL ANALISIS Y SE PUEDE DEFINIR COMO:

EL PROCESO ITERATIVO MEDIANTE EL CUAL, A PARTIR DE UN MODELO LOGICO DEL SISTEMA, JUNTO CON UN CONJUNTO DE OBJETIVOS BIEN DE FINIDOS, SE PRODUCE LA ESPECIFICACION DE UN SISTEMA FISICO QUE CUMPLE CON LOS OBJETIVOS ENUNCIADOS.

DEBE RESPONDER A LA PREGUNTA: ¿COMO SE VA A HACER?

OBJETIVOS DEL SISTEMA QUE DEBEN DEFINIRSE ANTES DE INICIAR EL DISEÑO.

- QUE SE CUMPLA CON LAS FUNCIONES REQUERIDAS POR EL USUARIO
- TIEMPOS MAXIMOS PARA LAS TRANSACCIONES MAS IMPORTANTES
- CONTROL SOBRE POSIBLES FALLAS HUMANAS O DE LA MAQUINA
- FACILIDAD PARA SER MODIFICADO EN EL FUTURO

(EN GENERAL LOS TRES ULTIMOS FACTORES SIEMPRE SE CONTRAPONEN)

BROOKS "THE MYTHICAL MAN-MONTH", A. WESLEY, 1975 Y
BOHEM, IEEE TRANSACTIONS ON COMP., DIC 76, ENCONTRARON:

<u>ACTIVIDAD</u>	<u>PORCENTAJE DEL COSTO DE DESARROLLO</u>	<u>PORCENTAJE DEL COSTO TOTAL</u>
<u>DESARROLLO:</u>		
ANALISIS Y DISEÑO	35%	} 20%
CODIFICACION	15%	
PRUEBAS Y CORRECCIONES *	50%	
<u>MANTENIMIENTO:</u>		
CORRECCIONES DURANTE PRODUCCION *		} 80%
CAMBIOS POR ADAPTACION *		
MEJORAS *		

LOS COSTOS DE LAS ACTIVIDADES MARCADAS CON * PUEDEN PRODUCIRSE
SIGNIFICATIVAMENTE SI EL SISTEMA ES FACIL DE MODIFICAR

EL DISEÑO ESTRUCTURADO ES UNA TÉCNICA PARA CREAR SISTEMAS QUE SEAN, FACILES DE MODIFICAR. LOS SISTEMAS MAS FACILES DE MODIFICAR SON - AQUELLOS QUE ESTAN FORMADOS POR PEQUEÑOS MODULOS MANEJABLES, CADA UNO DE LOS CUALES ES HASTA DONDE SEA POSIBLE, INDEPENDIENTE DE LOS DEMAS, DE MANERA QUE SE LES PUEDA RETIRAR, MODIFICAR Y DEVOLVER SIN AFECTAR EL RESTO DEL SISTEMA.

UN MODELO QUE SATISFACE ESTOS REQUERIMIENTOS ES EL QUE TIENE LAS SIGUIENTES CARACTERISTICAS:

- EL SISTEMA ESTA COMPUESTO DE UNA JARARQUIA DE MODULOS (CAJAS NEGRAS)
- CADA MODULO ES SUFICIENTEMENTE PEQUEÑO PARA SE MANEJABLE
- CADA MODULO PUEDE SER MODIFICADO SIN PRODUCIR EFECTOS DE PROPAGACION DE CAMBIO
- LAS FUNCIONES DEL USUARIO SON AGRUPADAS EN EL MENOR NUMERO DE MODULOS COMO SEA POSIBLE

EL MODELO ANTERIOR ES SEMEJANTE A UNA JARARQUIA MILITAR U ORGANIZACIONAL, Y PUEDE REPRESENTARSE MEDIANTE UNA "CARTA ESTRUCTURADA"

*24 EL DISEÑO ESTRUCTURADO PROPORCIONA CUATRO CONCEPTOS MUY UTILES PARA LA EVALUACION DE UN DISEÑO:

- ACOPLAMIENTO ENTRE MODULOS
 - COHESION
 - ALCANCE DE EFECTO
 - ALCANCE DE CONTROL
- } DE UN MODULO

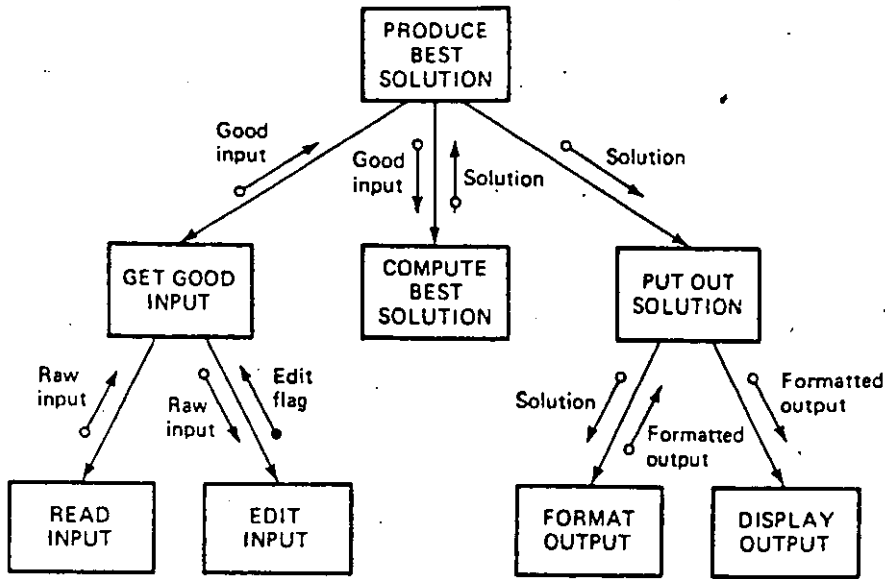


Figure 9.10 Structure chart for simple transform-centered system

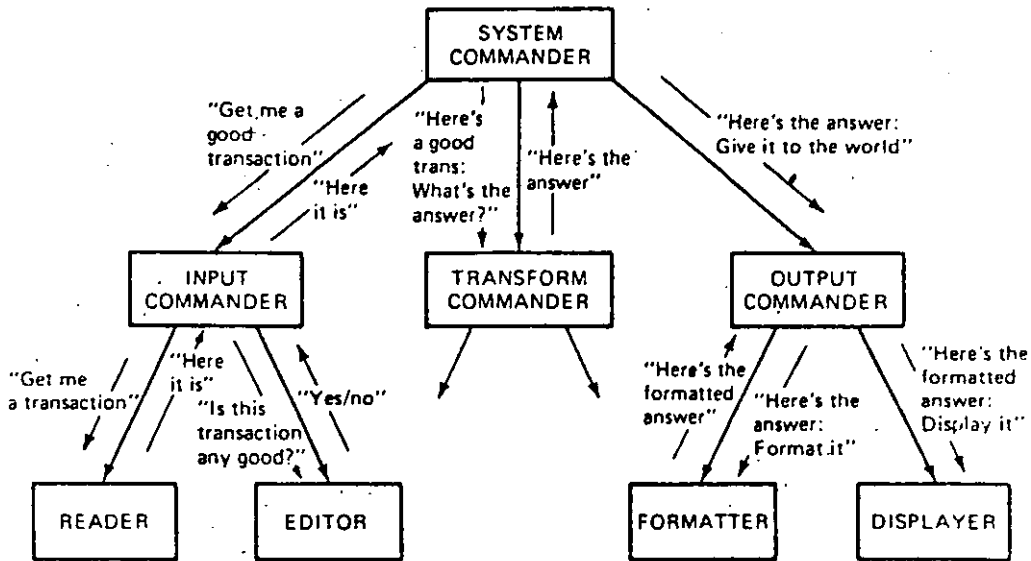


Figure 9.6 Military organization analogy

ACOPLAMIENTO ENTRE MODULOS:

SE DESEA MINIMIZAR EL ACOPLAMIENTO ENTRE MODULOS.

(MINIMIZAR LA RELACION EXISTENTE ENTRE ELLOS)

NO ES POSIBLE ANULARLO, YA QUE LOS MODULOS FORMAN PARTE DE UN TODO QUE ES EL SISTEMA:

TIPOS DE ACOPLAMIENTOS:

- A TRAVES DE DATOS: ES EL MAS DESEABLE. (PARAMETROS)
SE REPRESENTA CON UNA FLECHA CON COLA VACIA
- A TRAVES DE CONTROL: NO ES LO IDEAL, PERO EN OCASIONES RESULTA CONVENIENTE (BANDERAS COMO PARAMETROS) SE REPRESENTA CON UNA FLECHA DE COLA LLENA.
- EXTERNO O PATOLOGICO: ES LA RELACION DE UN MODULO CON OTRO EN LA CUAL UNO DE ELLOS HACE REFERENCIA AL ANTERIOR DEL OTRO. NUNCA ES ACEPTABLE (GOTO, COMMON)

COHESION DE UN MODULO:

SE DESEA MAXIMIZAR LA COHESION DE CADA MODULO.

(MAXIMIZAR EL GRADO DE PERTENENCIA DE TODAS LAS PARTES AL MODULO)

UN MODULO COHESIVO AUTOMATICAMENTE PRODUCE POCO ACOPLAMIENTO

TIPOS DE COHESION:

- ACCIDENTAL (LA PEOR). EJEM: MODULOS DE 60 LINEAS
- LOGICA (CERCANA A LA PEOR). EJEM: RUTINA GENERAL DE LECTURA
- TEMPORAL (MALA). EJEM: MODULO DE INICIALIZACION

- DE PROCEDIMIENTO (REGULAR). EJEM: MODULO DE TODO EL THEN DE UN IF.
- DE COMUNICACION (REGULAR A BUENA). SUBFUNCIONES AGRUPADAS.
EJEM: CALCULA E IMPRIME
- FUNCIONAL (IDEAL). LLEVA A CABO SOLAMENTE UNA FUNCION.

PRUEBAS PARA SABER SI UN MODULO TIENE COHESION FUNCIONAL

- ¿ SE PUEDE DESCRIBIR CON UN SOLO VERBO Y UN SOLO PREDICADO ?
- ¿ SE LE PUEDE "ACUSAR" DE POSEER ALGUN OTRO TIPO DE COHESION ?

ALCANCE DE EFECTO:

CUANDO EN UN MODULO SE TOMA UNA DECISION, Y DEPENDIENDO DEL RESULTADO DE ESA DECISION SE INVOCA A UNO O MAS MODULOS, AL CONJUNTO DE MODULOS SOBRE LOS CUALES EL EFECTO DE INVOCARLOS POR OTRO MODULO SE CONOCE - CON EL NOMBRE DE ALCANCE DE EFECTO.

EL TIPO DE DECISION PUEDE SER UNICA (ROMBO) O CICLICA (SEMIELIPSE)

*27

ALCENCE DE CONTROL:

ES EL CONJUNTO DE MODULOS A LOS CUALES PUEDE LLAMAR DIRECTAMENTE, MAS LOS MODULOS QUE A SU VEZ ESTOS MODULOS PUEDEN LLAMAR, ETC.

DE LA MISMA FORMA EN LA QUE UN OFICIAL NO DEBE DAR ORDENES A TROPAS QUE NO ESTEN BAJO SU MANDO, UN MODULO NO DEBE TOMAR UNA DECISION - QUE AFECTE A MODULOS QUE NO ESTEN DENTRO DE SU ALCANCE DE CONTROL.

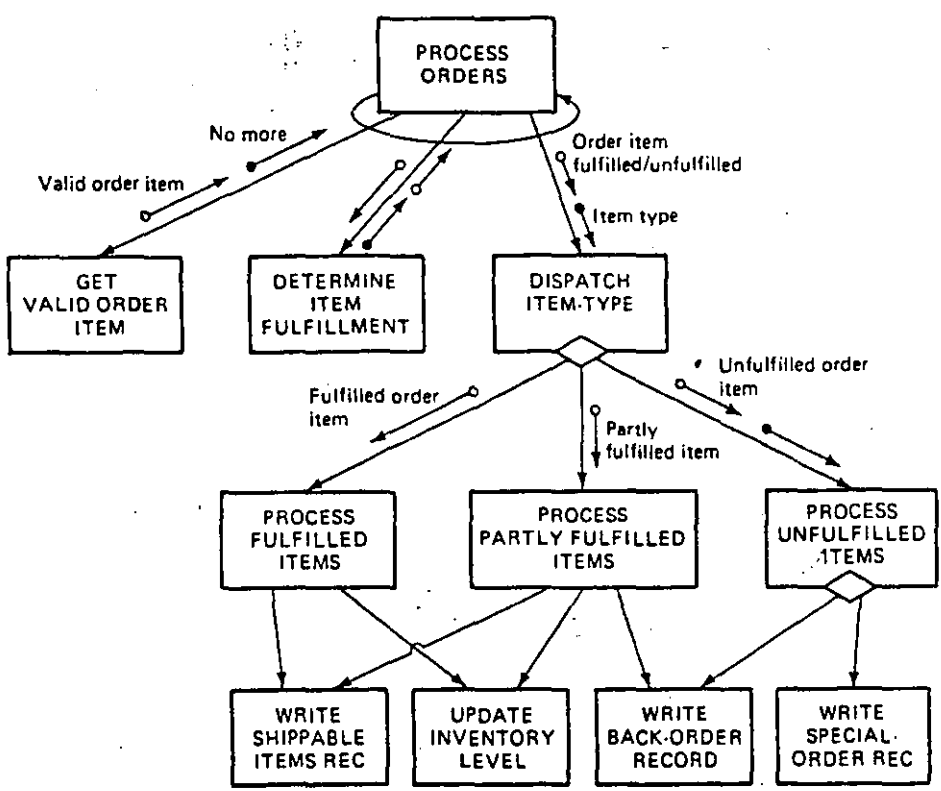


Figure 9.24 Refinement of design

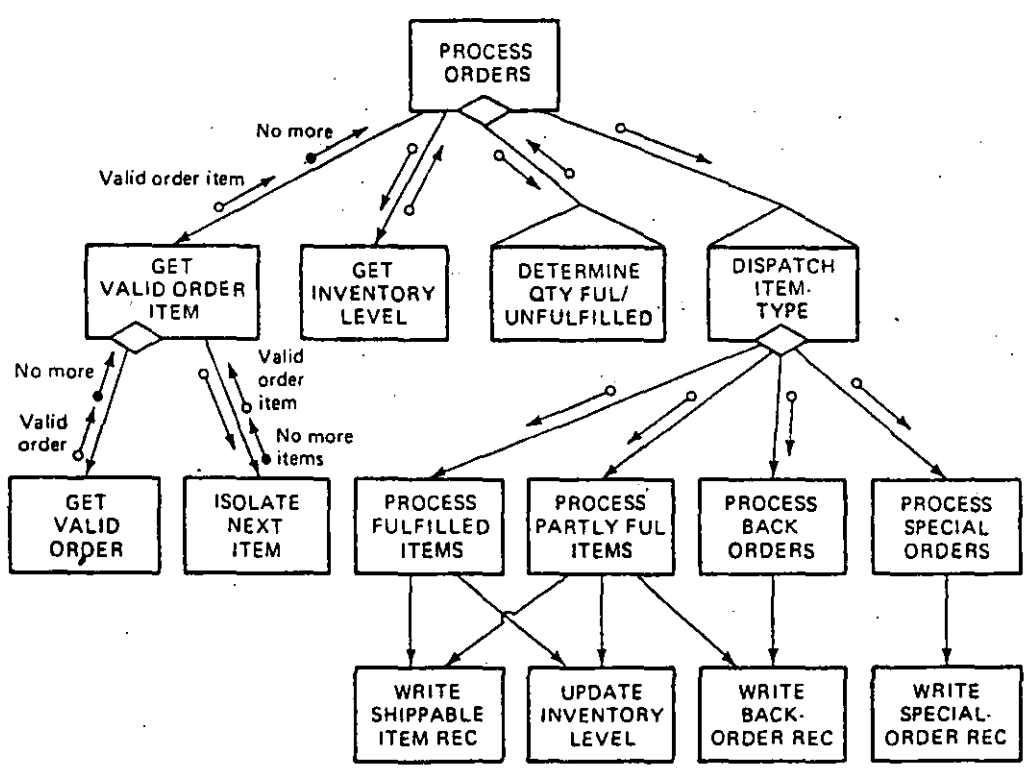


Figure 9.28 Start of design of input leg

UN DISEÑO PUEDE OBTENERSE A PARTIR DEL MODELO LOGICO DEL ANALISIS ESTRUCTURADO (EL DFD) A UN MODELO FISICO (LA CARTA ESTRUCTURADA) UTILIZANDO LA ESTRATEGIA CONOCIDA COMO "EL ANALISIS DE TRANSFORMACION", EL CUAL PUEDE RESUMIRSE EN:

- TRAZAR UN DFD

- IDENTIFICAR LA TRANSFORMACION CENTRAL (FUNCIONES ESENCIALES)
(NO INCLUYE E/S)

- TRAZAR UNA PRIMERA CARTA ESTRUCTURADA CON MODULOS ESENCIALES
(ITERAR CON LOS PROCESOS ESENCIALES PARA SELECCIONAR LA CAJA NEGRA DE MAYOR JERARQUIA)

- ANADIR MODULOS DE E/S, MANEJO DE ERRORES

- REVISAR CARTA ESTRUCTURADA
(BALANCEADA O REPETIR 2 PASOS ANTERIORES)

EXISTEN ALGUNAS ESTRATEGIAS ADICIONALES, ENTRE LAS CUALES DESTACA LA LLAMADA "EL ANALISIS DE TRANSFORMACIONES", LAS CUALES NO SE CUBRIRAN EN ESTA PLATICA.

CAPITULO V

PRUEBAS DEPURACION Y MANTENIMIENTO

V.1 CONCEPTOS DE VALIDACION, DEPURACION Y PRUEBA DE PROGRAMAS

La depuración, prueba y mantenimiento de programas es un área que ha sido ignorada por largo tiempo y poco se ha investigado, recomendado y/o concluido sobre estos puntos.

El objetivo principal de estos puntos es aumentar el nivel de comprensión así como apreciar la magnitud del esfuerzo que significa el Probar, Validar y Depurar programas; además, discutir y reconocer las técnicas y estrategias que se pueden aplicar para llevar a cabo los puntos arriba comentados.

V.1.1 DEFINICIONES Y CONCEPTOS:

Lo siguiente no pretende formalizar sino sólo familiarizar al lector con algunas de las palabras de uso común entre los programadores de cualquier área de trabajo.

1. Programa:

Un programa es una colección de instrucciones que, cuando se combinan con datos apropiados e información de control, puede llevar a cabo algunas funciones previamente definidas. Un programa es la mínima colección de instrucciones que puede funcionar sin interactuar con otros programas.

2. Módulo:

Un módulo se define de una manera similar a la de un programa; es decir, es una colección suficiente de instrucciones para desarrollar una función lógica. En algunos lugares se considera módulo como sinónimo de subrutina; también se lleva a definir con base en restricciones físicas, como por ejemplo, en el número de

instrucciones. Implícitamente se asume que un módulo no es útil si no se le combina con otros módulos para formar una unidad mayor (ejemplo un sistema).

3. Subsistemas:

Se asume que un subsistema es una colección de programas organizados de cierta manera, para llevar a cabo una función mayor y más compleja lo que no sería posible en un programa sencillo. También se habla de Jerarquías de subsistemas o niveles de subsistemas.

4. Sistema:

La palabra sistema se usa para describir una colección de programas, por ejemplo el sistema operativo o el sistema de nómina. En términos más generales un sistema se considera una colección de programas y/o subsistemas suficientes para llevar a cabo una aplicación o función coherente y significativa.

Nota:

Las anteriores definiciones de Programa, Módulo, Subsistema y Sistema se hicieron con base a la complejidad de su función pero si nos vamos a la parte atómica de cada una de las definiciones (colección de instrucciones) encontraremos que no existe diferencia alguna entre los conceptos arriba mencionados (para mayor detalle en la discusión se recomienda leer el capítulo de Diseño Estructurado).

5. Confiabilidad del Software:

Se ha llevado a definir como la probabilidad de obtener un programa operando por un periodo suficiente de tiempo, sin error de software, en la máquina para la que fue diseñado con todo y sus limitaciones. Confiabilidad se define en términos del tiempo promedio entre fallas MTEF (de sus siglas en inglés Mean Time Between Failures) y tiempo promedio de corrección MTTR (de sus siglas en inglés Mean Time To Repair "MTTR"), o en términos de integridad de la base de datos (si los datos manejados son actualizados y administrados correctamente, el sistema es confiable). Todos estos términos tienen un mayor significado cuando los utilizamos en el contexto de un sistema que cuando lo usamos en un programa sencillo o en un módulo.

6. Error (Bug):

Se define como un defecto de software, puede ser sólido (consistente) o transitorio (pasajero).

a. Sólido:

Error que se produce invariablemente al ejecutar el programa, pueden producirse por equivocaciones en la codificación o debido a la lógica errónea del programador.

b. Transitorio:

Error que se produce esporádicamente con determinados datos de entrada o por problemas de insumos. Se diferencia del anterior en que la falla puede aparecer en determinadas ejecuciones y en otras no.

7. Imprevisto (Glitch):

Definiremos un imprevisto cuando se presentan características no pensadas en un programa (en cualquiera de las fases de su desarrollo) que se traduce en ineficiencia y pone de manifiesto una falta de estilo. Nótese la diferencia entre un "error" y un "imprevisto". Un error implica que el programa no trabaja como se especificó, mientras que un imprevisto implica que la especificación (de alguna propiedad del programa) sale del rango de la especificación original, es ineficiente, mal interpretada etcétera.

8. Probar:

Probar (Testing) se define como un conjunto de procedimientos y actividades que pretenden demostrar la correcta operación e interacción de un programa con su medio ambiente. Tal como, probar la existencia de errores o la ausencia de los mismos. Se distinguen tanto pruebas de noción como pruebas de depuración.

Las pruebas de noción son las que se ejecutan para detectar errores es decir para tener conciencia de la existencia de los mismos.

Las pruebas de depuración tienen como objetivo una vez detectado el error, buscar las causas que lo ocasionaron y cómo corregirlo.

9. Probar la Efectividad de un Programa:

En los últimos años, ha existido un interés creciente en el desarrollo de una prueba rigurosa de la efectividad de un programa. Esto se ha hecho tomando en cuenta el medio ambiente en el cual opera el programa, usando un conjunto de axiomas y teoremas. Se espera que eventualmente se desarrollen técnicas para automatizar el proceso.

Como se señaló en el capítulo de Implantación la dificultad para probar la efectividad de un programa está muy relacionado con su complejidad y el número de componentes que interactúan; por lo tanto la programación estructurada (que tiende a reducir la complejidad de los programas) será vista desde ahora como parte importante del proceso.

10. Validación, Verificación y Certificación:

No tan comunes como los términos "probar" y "depurar", validar, verificar y certificar son ocasionalmente usados en diálogos relacionados con pruebas de programas. Como la define Hetzel, "Validación es el conjunto de procedimientos y actividades que pretenden demostrar la lógica correcta de un programa en un medio ambiente externo al desarrollado". "Verificación es el conjunto de procedimientos y actividades que pretenden demostrar la correcta lógica de un programa con una prueba en un medio ambiente determinado de pruebas", el cual puede o no corresponder al medio ambiente en el cual eventualmente correrá. Finalmente, "Certificación se define como una aprobación de lo correcto y efectivo de un programa, basado tanto en experiencias, metodología como en un conjunto organizado de procedimientos y actividades".

11. Depuración:

Depurar (Debugging) es un conjunto de actividades (algunas de las cuales pueden relacionarse con la computadora y otras no) que empiezan en el descubrimiento de la existencia de un error y terminan cuando se localiza y concisamente la naturaleza del error. De acuerdo con algunos expertos, depurar también incluye el período de tiempo durante el cual el error se corrige, y no termina realmente hasta que se demuestra a todos satisfactoriamente que el error corrigió y se trabaja correctamente.

12. Mantenimiento:

La palabra mantenimiento se usa normalmente para describir todas las actividades relativas a la corrección de errores, actualizaciones para mantener la compatibilidad con nuevos sistemas operativos, compiladores y a la introducción de código para imprevistos de poca importancia. La distinción entre un cambio mínimo y un cambio mayor es absolutamente una característica propia del programa o sistema, pero debe ser fácil de hacer en la mayoría de los casos; ya que algunos casos se podrían considerar como parte del mantenimiento pero otros, según su dificultad, caerían en la categoría de desarrollo.

Otra manera de reforzar la definición adecuada de prueba es analizar los usos de las palabras exitoso y no-exitoso y particular su uso, por Jefes de proyecto, para clasificar los resultados en casos de prueba. La mayoría de ellos considera a un caso de prueba que no encuentra un nuevo error, como una pasada de prueba exitosa del programa mientras que generalmente se califica como no exitosa una prueba que descubre un nuevo error. De igual modo, un caso de prueba que descubre un error difícilmente puede ser considerado como no exitoso; antes bien ha probado ser una valiosa inversión. De aquí que otra manera de dar apoyo a la definición correcta de prueba sea invertir el uso corriente de estos términos. Se dice, entonces, que un caso de prueba exitoso es el que descubre un error y que un caso de prueba no exitoso es el que hace que el programa produzca un resultado correcto.

V.1.2 PRUEBA DE PROGRAMAS

V.1.2.1 INTRODUCCION

Es posible tratar el problema de la prueba de programas desde diversos puntos de vista; sin embargo, en este caso se parte de la idea de que se usa con gran frecuencia una definición totalmente incorrecta de la palabra probar y esta es la causa principal de que las pruebas resulten deficientes. Como ejemplo de tales definiciones se pueden dar las siguientes: 'probar es demostrar que no hay errores presentes en un programa', 'el propósito de probar es mostrar que el programa realiza correctamente las funciones esperadas' y 'probar es el proceso que lleva a confiar en que un programa hará lo que se supone que debe hacer'.

Todas estas definiciones son incorrectas porque describen casi lo opuesto de lo que debemos considerar como prueba de un programa. Dejando por un lado las definiciones, pensemos en que cuando se prueba un programa se desea agregarle algún valor (es decir, puesto que la prueba es una actividad costosa, se desea recuperar parte de este costo por medio de un incremento en el valor del programa). Agregar valor significa aumentar su calidad o su confiabilidad, lo que a su vez significa encontrar y eliminar errores. De aquí que no se debe probar un programa para mostrar que funciona; más bien es conveniente comenzar con la suposición de que el programa contiene errores y luego probar el programa para encontrar tantos como sea posible. Una definición más apropiada es:

Prueba es el proceso de ejecutar un programa con el fin de encontrar Errores

V.1.2.2 EL ALCANCE DEL PROBLEMA DE PROBAR PROGRAMAS

Casi todos los programadores (o administradores de grupos de programadores) han visto, oído, o experimentado las consecuencias del alcance de los errores en los programas de computadora; pero como la computación cada vez abarca más campos críticos para la sociedad, el efecto de chequeos y pruebas pobres o limitadas convierte a los programas y sistemas en algo cada vez más desastroso y con mayores consecuencias posteriores. Además de esto, muchos programadores, administradores de grupos de programadores y casi todos los Jefes no técnicos subestiman drásticamente el tiempo, energía, planeación y dinero que se deben dedicar a las pruebas. De la misma forma, se subestima la complejidad de depurar y ver operar correctamente un módulo; así mismo la magnitud y complejidad del mantenimiento de un sistema se dejan a un lado, siendo una consecuencia normal (pero no justificada) los problemas en las etapas posteriores.

Algunas estadísticas nos pueden ayudar a mostrar qué tan serio es el problema:

El tiempo requerido para probar un programa ha sido estimado entre un 30% y un 50% del total del tiempo requerido por el proyecto. Un sistema pequeño bien diseñado puede probarse en menos del 30%, pero por ejemplo un sistema planeado para dos años (diseño e implantación) Justifica plenamente de tres a nueve meses de pruebas.

El número de errores remanentes en programas grandes (después de que ha sido supuestamente probado) es aún muy grande; todo esto depende tanto del programador como de la aplicación.

12. Mantenimiento:

La palabra mantenimiento se usa normalmente para describir todas las actividades relativas a la corrección de errores, actualizaciones para mantener la compatibilidad con nuevos sistemas operativos, compiladores y a la introducción de código para imprevistos de poca importancia. La distinción entre un cambio mínimo y un cambio mayor es absolutamente una característica propia del programa o sistema, pero debe ser fácil de hacer en la mayoría de los casos; ya que algunos casos se podrían considerar como parte del mantenimiento pero otros, según su dificultad, caerían en la categoría de desarrollo.

Otra manera de reforzar la definición adecuada de prueba es analizar los usos de las palabras exitoso y no-exitoso y en particular su uso, por Jefes de Proyecto, para clasificar los resultados en casos de prueba. La mayoría de ellos considera a un caso de prueba que no encuentra un nuevo error, como una pasada de prueba exitosa del programa mientras que generalmente se califica como no exitosa una prueba que descubre un nuevo error. De igual modo, un caso de prueba que descubre un error difícilmente puede ser considerado como no exitoso; antes bien ha probado ser una valiosa inversión. De aquí que otra manera de dar apoyo a la definición correcta de prueba sea invertir el uso corriente de estos términos. Se dice, entonces, que un caso de prueba exitoso es el que descubre un error y que un caso de prueba no exitoso es el que hace que el programa produzca un resultado correcto.

V.1.2 PRUEBA DE PROGRAMAS

V.1.2.1 INTRODUCCION

Es posible tratar el problema de la prueba de programas desde diversos puntos de vista; sin embargo, en este caso se parte de la idea de que se usa con gran frecuencia una definición totalmente incorrecta de la palabra probar y esta es la causa principal de que las pruebas resulten deficientes. Como ejemplo de tales definiciones se pueden dar las siguientes: "probar es demostrar que no hay errores presentes en un programa", "el propósito de probar es mostrar que el programa realiza correctamente las funciones esperadas" y "probar es el proceso que lleva a confiar en que un programa hará lo que se supone que debe hacer".

Todas estas definiciones son incorrectas porque describen así lo opuesto de lo que debemos considerar como prueba de un programa. Dejando por un lado las definiciones, pensemos en que cuando se prueba un programa se desea agregarle algún valor (es decir, puesto que la prueba es una actividad costosa, se desea recuperar parte de este costo por medio de un incremento en el valor del programa). Agregar valor significa aumentar su calidad o su confiabilidad, lo que a su vez significa encontrar y eliminar errores. De aquí que no se debe probar un programa para mostrar que funciona; más bien es conveniente comenzar con la suposición de que el programa contiene errores y luego probar el programa para encontrar tantos como sea posible. Una definición más apropiada es:

Prueba es el proceso de ejecutar un programa con el fin de encontrar Errores

V.1.2.2 EL ALCANCE DEL PROBLEMA DE PROBAR PROGRAMAS

Casi todos los programadores (o administradores de grupos de programadores) han visto, oído, o experimentado las consecuencias del alcance de los errores en los programas de computadora; pero como la computación cada vez abarca más campos críticos para la sociedad, el efecto de chequeos y pruebas pobres o limitadas, convierte a los programas y sistemas en algo cada vez más desastroso y con mayores consecuencias posteriores. Además de esto, muchos programadores, administradores de grupos de programadores y casi todos los Jefes no técnicos subestiman drásticamente el tiempo, energía, planeación y dinero que se deben dedicar a las pruebas. De la misma forma, se subestima la complejidad de depurar y ver operar correctamente un módulo; así como la magnitud y complejidad del mantenimiento de un sistema se dejan a un lado, siendo una consecuencia normal (pero no justificada) las problemas en las etapas posteriores.

Algunas estadísticas nos pueden ayudar a mostrar qué tan serio es el problema:

El tiempo requerido para probar un programa ha sido estimado entre un 30% y un 50% del total del tiempo requerido por el proyecto. Un sistema pequeño bien diseñado puede probarse en menos del 30%, pero por ejemplo un sistema planeado para dos años (diseño e implantación) justifica plenamente de tres a nueve meses de pruebas.

El número de errores remanentes en programas grandes (después de que ha sido supuestamente probado) es aún muy grande; todo esto depende tanto del programador como de la aplicación.

Si probar es difícil, no debe de sorprender que el mantenimiento (que requiere su propio ciclo de prueba y depuración) causa serias dificultades en muchas organizaciones. Las estadísticas nos dicen que en promedio el 50% del presupuesto de los Centros de Cómputo se gasta en mantenimiento de programas existentes; en algunas de las organizaciones mayores, se alcanzó hasta un 80%.

V.1.2.3 NIVELES DE COMPLEJIDAD EN PRUEBAS DE PROGRAMAS

Hablar de pruebas de programas involucra hablar de la complejidad de los mismos. Es conveniente clasificarlos en pocas categorías; en este caso la clasificación está basada en un criterio que se inició en los grupos de desarrollo de aplicaciones en la industria de cómputo y se basa esencialmente en magnitudes.

1. Programas Sencillos

Características:

- Menos de 1000 instrucciones de código fuente
- Escrito generalmente por un programador en 6 meses o menos
- Generalmente no tiene interacciones con otros programas o sistemas.

En este grupo se incluirían: aplicaciones científicas pequeñas, escritas por ingenieros para resolver problemas numéricos sencillos; pequeñas rutinas de utilería para aplicaciones comerciales; programas de reportes escritos; y la gran mayoría de programas desarrollados por estudiantes de ciencias de la computación.

La observación más importante acerca de programas sencillos 'simples' es que usualmente no importa cómo se prueben de hecho no importa si son probados del todo.

No hay mucho interés en los programadores para probar programas de este grupo; generalmente depuran y prueban sus programas sin ninguna lógica y sobre la marcha; no tienen un plan ni una organización para probar; se supone que los errores que se encuentren después que se finalizó el programa, se arreglarán sobre la marcha durante la producción.

2. Programas de Complejidad Media

Características:

- Entre 1,000 y 10,000 instrucciones de código fuente,
- Escrito por entre 1 y 5 programadores en un periodo de 2 años
- Tiene poca, si es que la hay, interacción con otros sistemas
- Consiste de 10 a 100 módulos

La mayoría de programas caen en esta categoría. Aplicaciones comerciales (contabilidad, nómina, control de inventario, etc.) y aplicaciones científicas; los sistemas del equipo (ensambladores, compiladores, paquetes de manejo de datos, programas de control de telecomunicaciones etcétera.

Definitivamente no son programas triviales; si requieren de 1 a 2 años de tiempo de desarrollo implica que el esfuerzo de prueba puede llevar de seis a doce meses. Por otro lado cualquier problema que pueda encontrarse durante el proceso de prueba puede resolverse por el esfuerzo propio y la peor consecuencia es un poco de retraso en la fecha de instalación. Así mismo cuando el programa es puesto en producción el daño causado por cualquier error residual es perceptible pero no es crítico.

Es verdad que programas de 10,000 proposiciones son por sí mismos complejos y es verdad que los errores en estos programas a veces son catastróficos; aún así, parece ser que la mayoría de los programadores de este tipo de proyectos son muy confiados. La consecuencia de esto, desafortunadamente, es que el programador involucrado en tales proyectos raramente siente la necesidad de concentrarse en los buenos principios de diseño y programación, ya que de otra forma, los errores en proceso involucrarán retraso en la implantación.

Los efectos de fallas en esta categoría tienen mayor repercusión y teniendo en cuenta las típicas urgencias y necesidades de presentar resultados, pocas veces se cuenta con tiempo suficiente para trabajar con, por ejemplo, diseño descendente y programación estructurada.

3. Programas Complejos

Características:

- Entre 10,000 y 100,000 proposiciones de código fuente
- Escritos por entre 5 y 20 programadores en un periodo de 2 a 3 años
- Están compuestos por varios subsistemas
- Presentan frecuente interacción con otros programas
- Constituidos por entre 100 y 1000 módulos

Nótese que el criterio de clasificación ha sido el tamaño; los programadores ven con desagrado sistemas tan grandes. Los Jefes ven con recelo proyectos que requieren más de un equipo de programadores y un mayor nivel de supervisión. Frecuentemente cambian de trabajo uno o más programadores antes que el proyecto termine; el usuario no interactúa muy bien y cambia sus requerimientos muy frecuentemente; después que pasó cierto tiempo y comienzan los retrasos.

Los errores son más complejos y las técnicas de prueba clásicas no se ajustan en su totalidad. Los efectos son mucho más serios y los recursos desperdiciados son mayores. La presión es mucho mayor y las posibilidades de localizar un error son menores, debido a la complejidad. Si algunos errores se descubren poco antes de la producción oficial, ocasionarán retraso en la implementación y se caerá en el caso de los errores residuales (que se comentarán más adelante); en ocasiones se ha llegado a proponer que el usuario modifique su manera de trabajar para no modificar el sistema. La forma de trabajo detrás de esta explicación debe quedar clara:

Se puede algunas veces darse el lujo de ignorar las técnicas de pruebas (y buenas técnicas de diseño) cuando se desarrollan programas sencillos o de complejidad media; en el caso de programas complejos, utilizar una técnica de prueba muy pobre estará condenando al fracaso el sistema que se está desarrollando. Debido a que el crecimiento de un número de aplicaciones comerciales y científicas se aproximan a este nivel de complejidad se debe pensar que un nivel de pruebas más estricto es necesario y no solamente deseable como en el caso

anterior.

4. Programas muy Complejos

Características:

- Entre 100,000 y 1,000,000 de instrucciones de código fuente
- Escrito por grupos de 100 a 1000 programadores en varios años
- Requiere de desarrollo y mantenimiento continuos, por gente diferente a la del desarrollo original
- Formado por entre 1,000 y 10,000 módulos
- Varios subsistemas generales con interacción compleja y comunicación con otros sistemas desarrollados separadamente
- Generalmente consumen una gran cantidad de recursos, telecomunicaciones, multitareas y procesos en tiempo real.

Hasta hace pocos años estaban limitados a usuarios muy especiales como serían grandes bancos, gobiernos y proyectos militares; pero en la actualidad varias organizaciones industriales han comenzado a generar aplicaciones de este tamaño.

Tiene que quedar perfectamente claro que en estos proyectos hay una gran oportunidad de falla si no se insiste en estrategias rígidas de diseño y prueba.

Dada la magnitud y el tipo de usuarios, los grupos de trabajo encargados de desarrollar programas muy complejos, han contribuido enormemente al desarrollo de técnicas de depuración y prueba; desafortunadamente no todos los equipos de trabajo disponen de recursos (económicos, humanos, técnicos etcétera) suficientes para llegar a un término feliz; cabe comentar que una falla en un proyecto de este tipo se puede y debe evitar usando estrategias de análisis, diseño, pruebas, etcétera adecuadas.

5. Programas muy Especiales

Características:

1. Entre 1,000,000 a 10,000,000 de instrucciones de código fuente.
2. Más de 1,000 programadores en una década o más de desarrollo.
3. Uso de gran cantidad de recursos y en ocasiones tecnología especial
4. Involucran procesos críticos como pudieran ser: Control de tráfico aéreo, controles de tiempo real, controles espaciales, defensa terrestre y aérea etcétera.
5. Confiabilidad altísima.

Limitado a gobiernos, organizaciones militares o grupos económicamente muy poderosos

Para esta categoría, dado el tamaño y la importancia, algunos gobiernos han implantado grupos de estudio y técnicas especiales de prueba, operación, depuración y mantenimiento; pero su experiencia no se ha podido difundir debido a la importancia de la calidad de "secretos", de estos proyectos.

V.1.2.4. TIPO DE ERRORES QUE SE DETECTAN MEDIANTE PRUEBAS

Como en alguna ocasión comentó Dijkstra, "Las pruebas deben mostrar la presencia; no la ausencia de errores"

Clases de errores:

1. Errores de Lógica

Los errores de lógica son normalmente el tipo de error más común y mucho del esfuerzo estará enfocado a encontrar este tipo de error. Podemos considerar un error de lógica como un error sólido y repetitivo. Es decir, si una entrada determinada manifiesta la presencia de un error entonces la misma entrada, cuando se le dé al programa una segunda vez, debe dar el mismo error en la misma forma.

2. Errores de Documentación

Para algunas aplicaciones, un error de este tipo puede ser tan serio como un error de lógica; ¿qué ocurre si la documentación de operación o del usuario tiene errores? De aquí se desprende la importancia de la participación de todas las áreas involucradas en la prueba de programas.

3. Errores de Comunicación o Interfaz

Son los errores que encontramos cuando un módulo no entrega la información necesaria o cuando los datos que se pidieron son insuficientes. Es necesario verificar que la interfaz entre módulo y módulo se realice adecuadamente.

4. Errores de Exceso de Datos o Recursos

Este error se puede poner de manifiesto cuando se piense en los siguientes cuestionamientos: ¿Qué ocurre si varias tablas internas, colas u otras áreas de memoria se llenan a su máxima capacidad? ¿Cuál es la degradación del tiempo de respuesta? ¿Solo es degradación o pérdida de control?

Al notar este tipo de situaciones, será necesario verificar los dimensionamientos, capacidades y recursos consumidos.

5. Errores Función del Tiempo

Generalmente se presentan en sistemas en tiempo real, ya que existen funciones condicionadas al tiempo; o bien cuando la operatividad depende de ciertos eventos; además, debe tomarse en cuenta que las posibilidades y combinaciones función del tiempo son infinitas.

6. Errores de Sobrecapacidad

Consisten en rebasar los límites estimados, lo que trae como consecuencia el comportamiento del sistema no sea normal, ya sea por medio de errores específicos o no específicos.

Estos errores son críticos en tiempo real, ya que cuando no se prevén crecimientos suficientes o incrementos en los volúmenes manejados, en algo que, por ejemplo, ya esté operando y utilizando recursos, será necesario reiniciar, lo que involucra perder lo ya usado y repetir, en el mejor de los casos, parte del proceso y lo faltante. Para probar este tipo de errores, es necesario chequear rangos de uso normal y máximos

estimados así como tendencias de crecimiento.

7. Prevenir y Recuperar Procesos cuando se presentan errores

Quando se presentan fallas (tanto de hardware como de software) el no prevenir una recuperación (parcial o totalmente) de una o varias corridas puede causar la pérdida de varias horas (hombre y máquina) de trabajo o en el caso de un sistema en línea o de tiempo real, puede generar confusión y caos en los usuarios.

Probar la recuperación significa asegurarnos que los programas pueden continuar a partir de puntos de chequeo, que la protección sea sistemática, que los módulos recuperados se puedan reiniciar fácilmente, contemplar mensajes e instrucciones tanto oportunos como adecuados así como, contar con todos los insumos adecuados para el reinicio.

8. Errores de Software y Hardware

Se piensa que esto no es responsabilidad del usuario ni del programador pero las pruebas deben verificar entradas/salidas adecuadas y actualizadas, manejo de recursos disponibles y actualizados, garantía de continuidad en los cambios de versiones, sobre todo cuando los usuarios finales no son técnicos; cuantos trillados "fue error de la máquina" con el que se justifican muchos programadores, se hubieran podido prevenir y evitar.

9. Errores en los Estándares

Los programas deben ser supervisados para asegurarse que sean codificados de acuerdo con los estándares de programación de la organización. Entre otros estándares ver por ejemplo si son modulares, si están bien comentados, si tienen proposiciones no estándares, etcétera.

10. Errores de Codificación

No por comunes son menos importantes; cuantos sistemas se hubieran corregido rápidamente si los programadores le dieran mas importancia a este tipo de errores, el oportuno conocimiento y auxilio de un manual es indispensable para esto.

V.1.2.5 FILOSOFIAS DE PRUEBAS DE PROGRAMAS

Los resultados obtenidos de múltiples experiencias demuestran que los sistemas desarrollados deben ser sometidos a pruebas rígidas, de manera que muestren la presencia de todos los errores que pudieran contener (no sólo los más explícitos). Lo anterior no lo podremos lograr con pruebas triviales o rígidas pero que en su metodología apliquen mucho de intuición y no lleven un orden al realizarlas (muchos detalles importantes pueden pasar desapercibidos si saltamos de la prueba de un módulo a la prueba del sistema y luego probamos otro módulo).

La intuición y experiencia del programador jamás podrá ser suplida por ninguna técnica sofisticada de prueba pero si combinamos lo anterior con una técnica ordenada de pruebas lograremos mayor efectividad al realizarlas con el consiguiente ahorro económico.

Si queremos tener oportunidad de probar un sistema completamente debemos hacerlo por etapas ordenadas. Es interesante hacer notar que hay básicamente dos enfoques para el concepto de pruebas en etapas: La estrategia tradicional, llamada recientemente ascendente en contraste con el enfoque nuevo descendente.

Con estas técnicas no se garantiza que el sistema no tendrá ningún error al instalarlo pero los que existan serán de mínima importancia y podrán ser detectados y corregidos rápidamente con el consiguiente ahorro en dinero, tiempo y esfuerzo.

V.1.2.5.1 PRUEBA ASCENDENTE:

La prueba ascendente normalmente requiere tres etapas: prueba modular, prueba de subsistemas y prueba del sistema; una cuarta etapa, llamada de aceptación se usa en ciertas aplicaciones como las comerciales.

1. Prueba Modular:

Se considera el nivel básico de pruebas, como tal, involucra usualmente el trabajo de un solo programador por un periodo de tiempo menor a un mes.

La prueba modular establece que el módulo se prueba sin la presencia de otros módulos en el sistema. Sin embargo para hacer la prueba práctica y simple, a veces se tiene la necesidad de utilizar una base de datos y

otros elementos del medio ambiente en el cual el programa se ejecutará que podrían ser generados por alguna técnica automatizada de generación de datos.

Para que tenga objeto la prueba del módulo se debe de poner el máximo esfuerzo en probarlo exhaustivamente; más aún, se debe de poder demostrar que la propia prueba fue exhaustiva. Se debe hacer notar que una prueba exhaustiva de un módulo implica que el programa se diseñó para estar constituido de módulos pequeños y funcionalmente independientes.

Distintas organizaciones han hecho esfuerzos para apoyarse en estadísticas útiles para la prueba modular tales como:

- a. Estadísticas que nos determinen el número promedio de líneas en las cuáles se deberá obtener un vaciado de variables para verificar valores.
- b. En cuanto tiempo real se prueba un módulo completamente.
- c. Porcentaje de errores de nuestros codificadores de programas para saber cuántos errores debemos esperar.

2. Pruebas de Subsistemas:

El siguiente nivel prueba la interfaz entre varios módulos del programa. Es de esperarse encontrar errores de lógica y de comunicación en contraste con los errores de codificación descubiertos en el nivel anterior. Este punto se debe ejecutar en orden jerárquico ascendente y por niveles.

3. Prueba del Sistema:

En este nivel se prueba el sistema completo. Los errores que se debe esperar encontrar son los más sutiles tales como:

- a. Errores de comunicación entre módulos.
- b. Errores de control y lógica.
- c. Errores de recuperación.
- d. Errores de capacidad.

- e. Errores de secuencias (los módulos no se ejecutan en la secuencia debida).

Es importante establecer un criterio de fin de pruebas ya que es casi imposible probar todo el sistema completamente (dependiendo de la complejidad). Algunos de los criterios son:

- a. El número de transacciones o casos de prueba que han sido exitosos.
- b. El número de horas en operación sin problemas.
- c. Porcentaje de módulos o proposiciones de código fuente que se han ejecutado.

4. Prueba de Aceptación:

También llamada prueba de campo o prueba de usuario tiene como finalidad probar el sistema completo en el medio ambiente de uso (con datos de prueba) pero la prueba la realiza el usuario "en vivo" comparando lo que dicen sus manuales con lo que hace el sistema; involucra a los usuarios finales, administradores del sistema, al personal de operación y producción etcétera.

V.1.2.5.2 PRUEBA DESCENDENTE:

Busca probar el sistema de lo general a lo particular, es decir: a partir del programa principal con uno o dos niveles de subrutinas como esqueletos (el resto de los niveles se implantará con módulos simulados y se irán agregando siempre de manera descendente, ampliando la llave de acceso hasta cubrir la totalidad del sistema. Es muy importante tomar en cuenta que no es un procedimiento rígido, sobre todo por las limitantes de los módulos inferiores simulados (algunos vitales); como serían por ejemplo los de entrada/salida; para estos casos se recomiendan las siguientes opciones:

1. Los módulos inferiores críticos se pueden diseñar, codificar y probar solos, así se facilitan las pruebas del resto del sistema con estructura descendente.
2. Se pueden utilizar combinaciones de pruebas ascendentes y descendentes.

3. Se pueden implantar versiones primitivas o generales de ciertos módulos inferiores claves para probar los módulos superiores.
4. Se pueden probar porciones completas como si fueran árboles, es decir, probar una rama completa luego otra y así sucesivamente.
5. Esta técnica no es recomendable para programas de la categoría de los sencillos en un rango menor a las 500 instrucciones.

V.1.2.5.3 VENTAJAS DE LAS PRUEBAS DESCENDENTES:

1. Se va agregando sólo una nueva pieza de código a la vez; cuando se presenta un error es fácilmente localizable; esto como consecuencia directa de lo anterior.
2. Con pruebas descendentes, se chequean mejor las interfaces, el control y los errores de lógica (en ese orden) los sobrantes casi siempre son errores de codificación.
3. Si se combinan técnicas de prueba descendentes, con diseño y programación descendentes, se eliminan ambigüedades y se coincide en las tendencias, lo que origina una comunicación más clara.
4. Ver el sistema general trabajando estimula al usuario final, el cual contribuirá eliminando errores y haciendo observaciones con base en sus requerimientos; esto no ocurre cuando, pasado cierto tiempo, solo se presentan retrasos o etapas parciales.

Nota: Para más detalles sobre la prueba descendente referirse al capítulo de implantación.

V.1.2.6 OBSERVACIONES ADICIONALES PARA PROBAR PROGRAMAS:

1. Cheque continuamente en pruebas de escritorio Preliminares, recorridos y exámenes visuales, empiece por el código, la lógica, la comunicación y así

sucesivamente.

2. Diseñe el programa para que no tenga errores
3. No continúe si no ha corrido el error detectado en la etapa actual
4. Respete al máximo la organización y el plan de trabajo original; no quite etapas cuando aumente la presión.
5. No use sofisticaciones poco claras; no asuma que todos conocen las opciones por default disponibles.
6. No use cuatro o cinco técnicas de prueba simultáneas; defínase antes de iniciar; use una y auxíliase con otra más.
7. Trabaje colaborando con su equipo; aplique los estándares de programación de su instalación.

V.1.2.7 TECNICAS AUTOMATICAS DE PRUEBA DE PROGRAMAS:

El camino para eliminar errores de un programa es no permitir que se ocasionen desde la primera fase; esto es la pauta para generar técnicas automáticas de prueba de programas. La filosofía básica es crear datos prueba, almacenarlos y ejecutar el programa; verificar que la salida corresponda a los requerimientos. Todo esto se ha trabajado con base en las pruebas manuales y a factores como el error manual humano y el tiempo que lleva probar algo por los métodos usuales.

Las pruebas manuales son lentas, tediosas y generalmente la cantidad no es suficiente puesto que el tiempo siempre apremia. Los programadores manualmente construyen archivos de prueba; datos de prueba; examinan manualmente la salida y la comparan; chequean la documentación existente; la consecuencia normal es que se dejan incompletas y los errores surgen posteriormente. El propósito de estos paquetes de prueba es, como consecuencia de todo lo anterior, ayudar, ahorrar tiempo y validar o probar lo más posible los programas antes de su liberación al usuario final.

V.1.2.7.1 GENERACION AUTOMATICA DE DATOS DE PRUEBA:

Se pretende generar grandes volúmenes de datos de prueba en una forma mecánica, pero no se debe olvidar que 'el volumen no siempre es un buen criterio de prueba'.

En lo que se está interesado aquí es en el volumen mínimo de datos de prueba que ejercitarán adecuadamente a nuestro programa. Generalmente aun este mínimo va más allá de los que se pudieran construir manualmente de aquí el surgimiento de herramientas automáticas efectivas de generación de datos para prueba.

Las herramientas surgidas por el problema anterior son los generadores automáticos de datos prueba (Test Data Generator 'TDG') que deben permitir especificar formatos y rangos de entradas válidas para, con base en éstos, generar automáticamente grandes volúmenes de ese tipo de datos.

El problema del generador de datos es que generalmente no se puede dar el significado que tendrían los datos en conjunto y por lo tanto nos generará datos que en ocasiones carecerán de toda congruencia y no tendrán el valor que pudiese darse en la realidad.

Otro problema es que al crecer la complejidad de la prueba crecen más y más variadas entradas, por ejemplo, para un programa con 50 parámetros obtener los datos prueba, requiere de gran interacción usuario-generador, lo que en suma puede ser igual al tiempo usado en generar datos manualmente.

Una situación similar se presenta en la generación de archivos prueba, aunque en el mercado existen productos que a partir de una muestra, generan un archivo prueba del tamaño necesitado, es necesario considerar las relaciones y combinaciones de los datos de un programa, generarlos, bien relacionados, no es fácil ni siquiera manualmente, sin ciertos conocimientos.

Los paquetes comerciales desarrollados ayudan, pero distan mucho de automatizar las pruebas, más bien ayudan a probar por los métodos tradicionales logrando una mezcla manual-automática; la orientación actual es hacia programas escritos en Cobol y equipos muy comerciales como IBM.

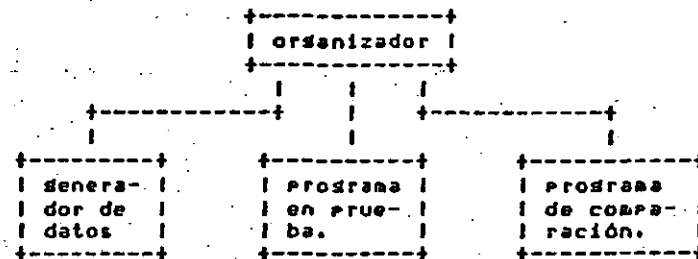
V.1.2.7.2 CHEQUEO AUTOMATICO DE SALIDAS:

Generalmente cuando se ejecuta una prueba, se debe saber cuál es la salida que se espera; con base en esto, se puede hacer que un programa pequeño compare una salida prevista previamente por métodos manuales y la obtenida con la ejecución, emitiendo una lista de discrepancias para su revisión.

Esto es factible para modelos individuales ya que si el sistema es muy grande la función de comparación se vuelve más compleja y no es posible efectuar las pruebas susodichas.

V.1.2.7.3 ORGANIZADOR AUTOMATICO DE PRUEBAS

La función del organizador es actuar como si fuese un programa principal. Se encarga de generar datos prueba de entrada, pasarlos al programa que está siendo probado, almacenar la salida, imprimirla y algunos de éstos chequean lo correcto de los resultados. Este tipo de paquetes se han desarrollado para automatizar las pruebas y más que automatizar estandarizar las pruebas.



V.1.2.7.4 RECHEQUEOS AUTOMATICOS:

La probabilidad de modificar adecuadamente un programa en el primer intento es menor al 50%. Entonces por qué no la siguiente metodología: la versión original del programa la ejecutamos con datos prueba; con los mismos datos ejecutamos la versión modificada; las salidas (datos iguales en dos versiones del programa) entran a un programa de comparación y generan un

000166

reporte de diferencias. Este método es Justificable sólo para casos de cambios substanciales y no para pequeños; además de que hay que generar un programa de comparación; para sistemas complejos y ciertas aplicaciones se Justifica el esfuerzo, ya que se contemplan varios puntos de prueba con base en los reportes finales.

V.1.2.7.5 PRUEBAS CON MONITOREOS AUTOMATICOS:

Con este tipo de herramienta se pretende seguir la ejecución del programa e interceptar cada instrucción y modificación de los registros que se han utilizado. Cuando el programa termina se emite un reporte de prueba para su estudio; Esta metodología no es nueva; muchas compañías tienen algo similar a disposición del usuario, lo difícil no es usarlo sino aprender a interpretarlo y entenderlo; además de emplear algunos recursos adicionales.

El monitor generalmente reporta:

1. Módulos ejecutados(útil en pruebas de sistemas)
2. Puntos de entrada/salida ejecutados
3. Condiciones y trayectorias seguidas
4. Instrucciones que no se ejecutan

Todo lo anterior nos permite hacer un seguimiento exhaustivo y de su interpretación correcta se deducen posibles errores o aciertos.

V.1.2.8 OTRAS TECNICAS DE PRUEBA:

1. PROGRAMACION REDUNDANTE:

Consiste en desarrollar dos o más programas independientes para la misma aplicación. El concepto de redundancia podría parecer extravagante, pero puede compararse con la idea de tener dos o más procesadores que respalden en caso de fallas de hardware. Hay una

analogía interesante: Algunos sistemas utilizan varios CPU's con un CPU que se usa para otro trabajo a menos que el sistema primario falle. Otros sistemas usan CPU's duales; con cada CPU ejecutan el mismo procesamiento y chequeando respuestas con los otros. La misma idea se puede extender a la redundancia en software con el gasto consecuente, pero en algunos casos se Justifica.

2. ESTANDARIZACION:

Esta sugerencia es bastante obvia: Los problemas de prueba disminuyen si utilizamos paquetes y rutinas de propósito general ya que éstas supuestamente tienen los estándares de la organización

3. SIMULACION:

En el desarrollo de sistemas complejos de computación es común utilizar la simulación analítica. Tal simulación involucra el desarrollo de modelos de un sistema de computación antes de su implantación ya que son muy útiles para encontrar, sobre todo, errores de capacidad.

Con este método se podrán encontrar algunos errores de lógica y no ayuda mucho a encontrar errores de codificación.

4. MODELOS MATEMATICOS DE LA CONFIABILIDAD DE PROGRAMAS

Los requerimientos de software confiable en muchos de los sistemas complejos actuales han encauzado a la investigación en el desarrollo de modelos matemáticos de la confiabilidad de un programa. El modelo utiliza a la estadística como arma principal para predecir lo siguiente:

1. El número de errores remanentes cuando va está en producción.
2. El tiempo promedio entre fallas (MTBF).
3. La probabilidad de que el programa corra exitosamente para un periodo determinado de tiempo.

5. INSPECCIONES Y RECORRIDOS

Las inspecciones y recorridos del programa son los métodos primarios de prueba humana; implican que la lectura o inspección visual de un programa han de ser

000167

efectuadas por un equipo de personas; involucran cierto trabajo preparatorio por parte de los participantes y el climax se alcanza en la "reunion de los cerebros" cuyo objeto es encontrar errores, pero no soluciones (es decir probar, no depurar).

Las inspecciones y recorridos ya tienen algún tiempo de empleo; el proceso se lleva a cabo por un grupo de trabajo (el número óptimo es 3 o 4 personas), una de las cuales es autora del programa. Se ha dicho también que las inspecciones o recorridos son sólo un nuevo nombre para lo que antes se denominó pruebas de escritorio (acto por el cual un programador lee su propio programa antes de comenzar a probarlo); pero no es así, aquí intervienen otras personas, además del autor. Ocurre también que estos procesos presentan costos menores de corrección, pues se determina perfectamente la naturaleza de un error al localizarlo y se descubren grupos de errores, lo cual permite una corrección masiva.

LOS RECORRIDOS (WALKTHROUGHS) de codificaciones, como la inspección son un conjunto de procedimientos y técnicas de detección de errores que consiste en la lectura del programa efectuada por un grupo de personas (Juntas de revisión). Tiene mucho en común con el proceso de inspección pero los procedimientos son ligeramente diferentes y se emplea una técnica distinta de detección de errores. El grupo de trabajo involucra un moderador, como en el proceso de inspección, otra tiene la función del secretario (registra todos los errores encontrados) y una tercera persona tiene el rol de probador.

El procedimiento inicial es idéntico al de inspección: los participantes reciben el material varios días antes para tener oportunidad de familiarizarse con el programa. Sin embargo el procedimiento usado durante la reunión es diferente. En lugar de leer simplemente el programa o usar listas "ayudamemoria", los participantes desempeñan la función de la computadora.

El tercer proceso humano de detección de errores es la Prueba de escritorio, esta puede ser vista como inspecciones o recorridos realizados por una sola persona, la cual lee el programa, lo controla con respecto a una lista de errores y/o pasa datos de prueba a través de él.

V.1.3 DEPURACION

Probar es un procedimiento organizado que intenta exponer la presencia de errores en un programa de computadora; como ya se mencionó es tedioso, difícil etcétera.

Una actividad paralela a probar, cuando se localiza la equivocación, es DEPURAR.

Depurar se considera un arte, el arte de encontrar la naturaleza y localización de un error que ya ha sido establecido con anterioridad. Mientras los procedimientos organizados y ayudas automáticas se siguen desarrollando e incrementándose (por ejemplo los vaciados de memoria) los programadores siguen encontrando errores gracias al sentimiento, la experiencia y la inspiración. Depurar, igual que probar, no es una actividad para cualquier programador.

A continuación se presentarán algunas estrategias y filosofías de depuración; así como algunos paquetes que desde nuestro punto de vista, son parte de los que han tenido mayor difusión.

V.1.3.1 FILOSOFIAS Y ESTRATEGIAS DE DEPURACION:

Para muchos programadores, depurar es sinónimo de ejecutar vaciados de memoria, seguimientos detallados, rastrear indicios, etc. Todo lo anterior es tan complicado en algunos equipos que los programadores piensan que esto es el último recurso a intentar.

El desconocimiento de las técnicas de depuración ha llevado al programador a creer que conociendo el paquete de depuración de cierto lenguaje de programación o rutina de software le resolverá cualquier problema de depuración. Esta es una verdad parcial ya que, en efecto, ayuda en gran medida pero no resuelve todos los problemas.

Los programadores más experimentados saben que depurar es mucho más que lo ya comentado, en muchas ocasiones todas las técnicas y paquetes existentes no servirán para encontrar y remediar errores difíciles de deducir, por la sencilla razón que no son obvios o bien por ser consecuencia de otros; muchos errores se pueden encontrar fácilmente (y con un considerable ahorro en tiempo de maquina) sólo con un poco de organización. Los errores realmente difíciles, en programas muy grandes, sólo se encuentran aplicando técnicas no orientadas a la computación sino a la administración y organización. El señor Hopkins comenta "un error es una equivocación; si nosotros conocemos un poco sobre el tipo de equivocaciones que cometemos, podremos

encontrarlos y corregirlos más fácilmente; así como posiblemente prevenir algunos que podrían presentarse en el futuro.

Con todo esto en la mente, a continuación se comentarán algunas estrategias para encontrar errores en un programa de computadores.

V.1.3.1.1 ESCRIBA MODULOS PEQUEÑOS

Uno de los caminos más sencillos para poder depurar adecuadamente un programa es dividirlo en módulos pequeños. Como ya se sabe, el promedio de errores se incrementa exponencialmente, cuando el número de instrucciones en un módulo crece; pero entonces qué ocurre cuando se sigue esta idea, si se hacen módulos pequeños, se incrementará el número de módulos y la interrelación entre módulos será más complicada, pero si el programa se desarrolla jerárquicamente y con metodologías descendentes, esto no se complicaría. Cuidado con caer en los extremos, ni muy grandes ni muy pequeños; no es práctico dar un número fijo de instrucciones para cada módulo, ya que esto depende de la aplicación.

V.1.3.1.2 TRATAR DE DETERMINAR LA NATURALEZA DEL ERROR

Si la actividad de pruebas se hace con suficiente cuidado puede ser posible encontrar la causa general del problema antes de que el programador se dé a la tarea de la búsqueda de la naturaleza específica del error. De tal forma que antes de tratar de encontrar el error con precisión hay que determinar con las pruebas si se está buscando un error de hardware, del sistema operativo, del compilador o del propio programa.

Puede ser útil contestarse las siguientes preguntas:

1. Puede ser un error de impresión?
2. El problema pudo haber sido causado por un error no usual en las tarjetas de entrada o el archivo maestro?
3. Se previeron mensajes indicativos, en este módulo, cuando la operación es anormal?
4. Se está ejecutando en una situación similar a la última vez o existen cambios significativos?, estos cambios pueden ocasionar el problema actual?

5. Se han hecho actualizaciones al hardware últimamente?

En un programa muy grande, la naturaleza de un error puede ser tan complicada como se quiera; el error lo causó la terminal, el usuario, la línea de comunicación, el multiplexor, el hardware, el sistema operativo o quizá su propio programa.

Esta situación se puede vencer "excluyendo del ámbito a buscar, lo que no produce error", por todo lo anterior, se sigue afirmando que depurar es un arte.

V.1.3.1.3 VER SI EL ERROR ES REPETITIVO Y CONSISTENTE

A menos que el error parezca demasiado simple, es buena idea ver si el error es consistente y repetitivo.

Si en la obtención de resultados observamos que el producto de un cálculo siempre es erróneo el error será consistente y si volvemos a correr el programa el error persistirá por lo que también es repetitivo. En un caso como el anterior resulta obvio el error puesto que todos los resultados son erróneos y fueron producto de determinado cálculo por lo tanto quizá en estos casos no sea necesario repetir la ejecución.

Existen casos no tan obvios como el anterior en que solo parte de los resultados están erróneos, aunque todos hayan sido producto del mismo cálculo este error no es consistente pero si repetitivo la ejecución del programa probablemente sea repetitivo aunque es posible que por las razones que exponeamos no se repita el error.

1. El operador pudo ocasionar el mal funcionamiento, un tropezón o un error de "dedo", ignorar algún error detectado por el sistema, o quizá alguna información requerida al operador y no contestada adecuadamente. Todos estos errores, y otros de la misma naturaleza, sólo se detectarán si se reejecuta el programa.
2. El programa se ejecuta en un medio ambiente de tiempo real y se originan errores de tiempo, qué ocurre si dos programas no lanzados adecuadamente tienen acceso al mismo tiempo a la misma base de datos? qué ocurre si una carga incorrecta del sistema operativo ocasiona errores aparentes.
3. Una falla en el hardware, como un falso contacto, o el daño a un cable, puede ocasionar fallas en ciertas instrucciones o comunicación con algún periférico.

Si un programa es muy grande y complejo, será muy difícil y caro reejecutarlo. Esto apoya nuestro argumento de escribir módulos pequeños; idealmente sería posible reejecutar sólo el módulo que involucra el error; pero si el error no es trascendente, no se puede echar a siempre al programador. Así como que el error no fue por causas humanas (ejemplo un error de operador), existirían varias cosas que pensar y hacer; esto sería una metodología secuencial muy sencilla:

1. Reejecutase el módulo que falló, si se obtienen los resultados correctos, no se debe olvidar el problema; nunca olvide ningún error; no acepte los errores inexplicables o misteriosos.
2. Ejecute algún programa de diagnóstico de hardware o solicite información a operación sobre el estado del mismo.
3. Asuma que el error está en su propio software y trate de encontrarlo.
4. Asuma que el error lo causó el operador y cómpelo; él probablemente tiene los elementos suficientes para desmentirlo; cuidado! en algunas organizaciones no pretenda recibir una respuesta antes de seis meses.
5. Si todo lo anterior falla, culpe al proveedor (ya sea hardware o software) probablemente él le ayudará "forzadamente"; pero es importante estar seguro que el error no está en su propio programa.

V.1.3.1.4 PROGRAMADOR, TE CONOCES BIEN?

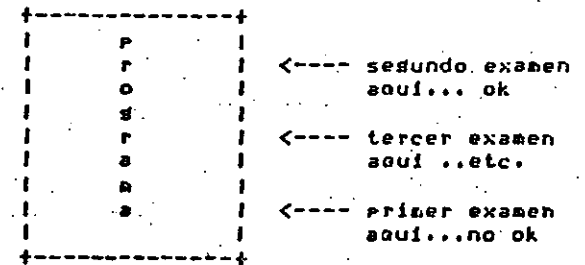
Los programadores que frecuentemente culpan al hardware, al sistema operativo o a cualquier cosa excepto a ellos mismos, tienen problemas para depurar sus programas. Es necesario que el programador reconozca que los errores son consecuencia de sus equivocaciones.

En muchas ocasiones los programadores cometen errores debido a malos hábitos como puede ser; al perforar o teclear sus propios programas; en otras ocasiones se tiene la costumbre de usar instrucciones que no se conocen bien y "suponen" sus funciones; otros olvidan salvar o reactualizar ciertos registros; algunos más olvidan pasos o destruyen valores útiles. Como último comentario, aproveche y reconozca los errores que cometió, sanará su experiencia; pero corrija la equivocación y entenderá por qué se generó.

V.1.3.1.5 SEA METODICO Y LOGICO

Cuestionese sobre los siguientes tópicos:

1. Cuál es la organización que sigue usted para buscar errores?
2. Tiene usted alguna organización?
3. Sigue usted caminos aleatorios o usa prueba y error?
4. Un ejemplo de enfoque para depurar programas puede ser la "búsqueda binaria", técnica muy común que es perfectamente aplicable a este caso.



Búsqueda binaria

La organización, la intuición y el sentimiento no son excluyentes entre sí, use todo lo que tenga a su alcance, pero ordenadamente y como complementos; si el error persiste, debe tomar en cuenta que probablemente el enfoque tomado no sea el correcto, reorganícese.

V.1.3.1.6 INVESTIGUE PRIMERO LO MAS OBVIO Y SENCILLO

No se complique la existencia, escriba una lista de "sospechosos" potenciales; antes de ahondarse y comprometerse con el problema. Examine en orden, vaya incrementando el nivel de dificultad; nunca al revés. No piense por dónde iniciar, por supuesto que por lo mas sencillo!

000170

V.1.3.1.7 NO DE NADA POR SEGURO

Se insiste en que, "La mayoría de los errores son consecuencia de nuestras propias equivocaciones". Un ejemplo claro de concesiones se puede ver en un programa grande, compuesto por muchas subrutinas. Cuando algún error ocurre, los programadores sin experiencia comentan: "no puede ser esta subrutina, ha funcionado por meses" y de esta forma se elimina la mitad de las rutinas (en donde probablemente sí se encuentre el error). Un programador más experimentado, diría: "Bueno, no creo que exista algún problema en estas rutinas, pero han tenido algunos pequeños cambios últimamente? ¿si hubo cambios, nunca dices que fueron insignificantes, tampoco afirmas que no son la causa del error, mejor cerciórese."

V.1.3.1.8 UTILICE LOS PEQUEÑOS DETALLES

En lugar de reejecutar el programa o hacer vaciados de memoria y seguimientos automatizados, practique las pruebas de escritorio, en muchas ocasiones el error está más cerca de lo que pensamos. Muchos errores se presentan esporádicamente y esporádicamente tenemos ocasión de detectarlos, inmediatamente vea cuáles fueron las circunstancias en las que se presentaron y céreuelos.

V.1.3.1.9 APOYESE EN SUS COMPANEROS DE TRABAJO

Si un error es muy difícil de encontrar, en ocasiones un compañero con la mente despejada puede apoyar bastante, solo necesita un concepto general del sistema y una gran intención. Describale el problema, su naturaleza y la secuencia de pasos involucrada, espere y escuche su opinión.

Tenga presente que su intención puede resultar contraproducente, no todos los compañeros pueden ser el apoyo esperado, acórruese a alguien en que usted confíe, una alternativa más es apoyarse en alguien para confirmar sus sospechas, le puedan ayudar o quizás aclarar algunos conceptos.

V.1.3.1.10 PRUEBE Y CONFIRME SUS SOSPECHAS

Quando una depuración se prolonga, uno se empieza a cuestionar sobre tópicos como el hardware, el lenguaje, el sistema operativo y duda de algunos de sus conocimientos, no lo haga! cheque sus preguntas, contéstelas, recurra a los manuales o a gente calificada, por lo menos conocerá más.

Aproveche el momento, depure otros aspectos no tan claros, esa documentación que se suponía perfecta y que no lo ayuda ahora, aplique y pruebe sus hipótesis, confirme.

V.1.3.1.11 OLVIDESE DEL PROBLEMA HASTA MAÑANA

Quando el tiempo se alarga y el caos mental comienza, lo mejor es olvidarse y descansar mentalmente, distráigase, haga algo diferente. Si su jefe tiene algo de experiencia lo entenderá, pero si no es así, no importa, mañana será otro día.

V.1.3.2 ERRORES COMUNES DE PROGRAMACION

1. Errores de perforación o codificación (confusiones entre 0 y O, proposiciones omitidas, etcétera.).
2. Inicializar variables
3. Compartir variables entre muchos módulos inadecuadamente
4. Errores de lógica y Control.
5. Subscriptores fuera de rango.
6. Mala terminación de módulos o entradas/salidas.
7. No prever manejos especiales de condiciones que se pudiesen presentar.
8. Mala comunicación o errores de interfaz.
9. Formatos inadecuados o equivocados de datos.
10. Paso de constantes como parámetros.
11. Mal uso de expresiones lógicas compuestas.
12. Anidamientos incorrectos de instrucciones de condición o ciclos.
13. Salidas incorrectas de módulos o subrutinas.
14. Uso excesivo de banderas.

15. Contadores muy pequeños
16. Direccionamientos incorrectos
17. Falta de capacidad en variables o arreglos.
18. Uso de versiones equivocadas de programas
19. Problemas con direccionamientos (invasión de áreas de código)
20. Errores con fórmulas para cálculos muy complicadas.
21. Problemas con el alineamiento de variables (mover campos de caracteres muy chicos a campos grandes sin prever la justificación).
22. Mal manejo de la recursividad.

V.1.3.3 TECNICAS CLASICAS DE DEPURACION

Después de discutir algunas filosofías de depuración discutiremos algunas técnicas clásicas de depuración.

No se hará mucho énfasis en los detalles del manejo de los paquetes ya que esa información puede ser conseguida con los proveedores. En lugar de eso se hará énfasis sobre las situaciones de programación en las cuales estos paquetes son útiles.

V.1.3.3.1 VACIADOS DE MEMORIA (CORE DUMP)

Uno de los paquetes más viejos y útiles de depuración es el simple vaciado de memoria. En cualquier momento durante la ejecución del programa, el programador puede, si piensa que es de utilidad, imprimir el contenido de los registros de memoria (o una porción de memoria) sobre una impresora, copiarlo a cinta magnética o a disco. El programador puede obtener una buena idea del error si examina cuidadosamente el vaciado.

El problema de este tipo de paquetes es que un 90% de la información obtenida no tendrá significado puesto que no entra en el contexto del error y por lo abundante de la información

incluso puede oscurecer el error. Otro punto importante es que la información sólo es el espejo en memoria de un determinado momento.

Por lo tanto se recomienda utilizar este tipo de paquetes cuando el tipo de error sea muy difícil de detectar y no se sospeche ni por dónde ocurrió. Cuando se decida utilizar este tipo de paquetes asegurarse de obtener el vaciado en un formato legible ya que no es raro encontrar una serie de números que espantan por su complejidad.

V.1.3.3.2 RASTREO (TRACE)

Otra ayuda muy popular para la depuración que es proporcionada en muchos lenguajes tanto de alto nivel como de bajo nivel es el rastreo.

Actualmente, un rastreo es una variación del vaciado de memoria descrito anteriormente. Hace que se impriman en determinados momentos, bajo condiciones especificadas por el programador, ciertas localidades de memoria (variables del programa, subrutinas, etc.). Virtualmente todas las recomendaciones hechas sobre los vaciados de memoria se aplican a los rastreos.

Si se usa indiscriminadamente, un rastreo wasterá quizá más papel que el vaciado de memoria ya que el rastreo es un proceso continuo a través de toda la ejecución del programa. Como en el caso de los vaciados de memoria algunos rastreos se imprimen en hexadecimal y a veces son imposibles de descifrar. Existen diversos tipos de rastreos:

1. Rastreo de Subrutinas:

El propósito es rastrear cada vez que se llama a una subrutina mediante la impresión de información concerniente a la subrutina (ejemplo valores de parámetros, nombre de la subrutina).

2. Rastreo de Variables:

Se usa imprimir el contenido de variables específicas, áreas de trabajo, arreglos, etcétera; conviene en muchos casos en el momento de un cambio de valor, condición o flujo.

3. Rastreo de Proposiciones:

Imprime el contenido de las variables que intervienen en determinada proposición cada vez que ésta se ejecuta.

4. Rastreo de Bases de Datos:

Es útil para sistemas que manejan bases de datos y que cada operación efectuada sobre ella es rastreada mediante la impresión de los cambios y actualizaciones efectuadas.

5. Rastros Condicionales:

En algunos casos el programador sabe cuándo no imprimir algo a menos que ocurran ciertas condiciones (ejemplo si A es menor que B; si Z es igual a cero, etc)

La técnica más aconsejable cuando se sospecha de la causa del error es la utilización de rastreos bien organizados. Si la naturaleza del error es completamente desconocida lo más recomendable es el vaciado de memoria.

V.1.4 MANTENIMIENTO

Partamos de la base de una frase clave, software mantenible, que es la facilidad de que el software pueda expandirse o contraerse para satisfacer nuevos requerimientos o bien se puedan detectar errores, deficiencias u obsolescencias.

Hagamos una comparación básica, mantenimiento en términos de hardware es restaurar una condición inicial. Mantenimiento en software implica un producto nuevo, probablemente mejor, mayor o más eficiente, pero nuevo, con la consecuente lista de acciones ligadas a esta situación: actualización de la documentación, adecuaciones, pruebas, implantaciones, nueva capacitación etcétera.

Mantenimiento de software involucra aspectos y casos como: Requerimientos o necesidades, correcciones, adaptaciones, reestructuraciones y engrandecimientos.

1. Requerir o Necesitar es la especificación que lleva al mantenimiento.
2. Corresidir es realizar todas aquellas actividades para que un módulo o conjunto de módulos que no efectúan la función que se pretende la realicen, modificando el software actual.

3. Adaptar es una modificación ocasionada por un cambio del hardware o software en que reside el sistema.
4. Engrandecimiento es una modificación para permitir al sistema realizar nuevas funciones con base en los nuevos requerimientos.
5. Reestructuración es una modificación que pretende mejorar la estructura interna del sistema conservando su comportamiento externo.

Entrando en materia, es común que el grupo encargado de dar mantenimiento a un sistema, se queje de que el grupo de desarrollo no previó la fase de mantenimiento de una manera adecuada. Para evitar lo anterior se recomiendan una serie de requisitos que pretenden obtener un producto mantenible, esto es, las condiciones mínimas para lograr sistemas estándares para que también el mantenimiento tienda a ser homogéneo.

1. Los requerimientos deben de presentarse (por escrito) bien definidos y detallados, de tal manera que se puedan comprender fácilmente. A los requerimientos del usuario final se deben de agregar los de operación y producción. De preferencia se desea tener bien diferenciados los requerimientos mínimos y los opcionales.
2. Las especificaciones deben de poder probarse
3. Se debe de analizar y diseñar, independientemente de la técnica usada, de tal manera que sea posible expandir, contraer y adaptar fácilmente; la explicación debe ser rica en ejemplos y prevenir cambios obvios.
4. La programación debe de usar versiones y características estándares del lenguaje seleccionado. Se debe de documentar lo más posible explicándolo al grupo de gente que lo va a usar y nunca a sí mismo.
5. Se debe de documentar total y ampliamente, entregando dicha documentación, tanto al usuario final como a las áreas involucradas, especialmente al encargado de mantenimiento.

Con estas condiciones mínimas, el mantenimiento deberá incluir las siguientes etapas:

1. Comprensión y compenetración del sistema actual.
2. Comprensión del objetivo de la modificación y enfoque.

3. Implantación de la actualización o modificación
4. Revalidación del software.

Quando se inicien las etapas del mantenimiento, se debe de tener presente la siguiente problemática:

1. La calidad de la documentación.
2. Las limitaciones de recursos (durante todas las etapas).
3. La calidad del software original.
4. La dificultad para entender el sistema debido a su complejidad.

Para comprender mejor el último aspecto, definamos calidad como la cercanía del sistema con respecto a los requerimientos del usuario así como la prevención de los efectos. Un software de calidad reúne las características de: confiabilidad, consistente, eficiente, conciso, mantenible, portable, entendible y económico.

V.1.4.1 EJECUCION DEL MANTENIMIENTO

Una vez familiarizados con la problemática que se presentara y pensando en las características anteriores, para ejecutar el mantenimiento de programas se necesita contar con los requerimientos siguientes.

1. Técnicos:
 - a. Medio ambiente de prueba muy semejante o al menos capaz de simular el ambiente operacional.
 - b. Recursos de cómputo suficientes con base en la situación real de operación.
 - c. Insumos suficientes (datos prueba, verificadores etcétera.).
 - d. Herramientas de diagnóstico, prueba (de los que se habló en su momento) y mantenimiento.
 - e. Personal de las áreas involucradas con perfecto conocimiento de causa.

2. Conocimientos y Practica: Dada la responsabilidad del mantenimiento, es necesario eliminar la tendencia de destinar a los programadores nuevos a esta actividad, mientras adquieren experiencia y conocen el equipo. Hay que recordar que un mantenimiento mal ejecutado ocasiona grandes costos y problemas con algo que ya está produciendo ya que se modifica su integridad. El personal de mantenimiento no sólo debe saber desarrollar sistemas, sino más que eso, mantener sistemas.
3. Elementos Técnico-Administrativos Buscando mantener la calidad y confiabilidad del producto, es necesario sujetarse a los estándares y verificar que se cumplan tomando en cuenta lo siguiente:
 - a. Las prioridades y objetivos pretendidos.
 - b. Verificación del cumplimiento de los estándares para mantener la calidad del producto
 - c. Existir documentación oportuna y actualizada
 - d. Establecer auditorias con enfoque en control de calidad.

V.1.4.2 DIRECTRICES A SEGUIR PARA EL MANTENIMIENTO DEL SOFTWARE:

1. Use en el análisis y diseño técnicas ascendentes y/o descendentes
2. Plantee desde un principio un plan de revalidación. Es decir un plan de pruebas del software que sea sujeto de mantenimiento.
3. Utilice todas las herramientas disponibles, tales como auditores de código, generadores de referencias cruzadas, generadores de documentación, generadores de datos de prueba, programas de diagnóstico en línea, etcétera.
4. Trate de lograr una independencia de la máquina así como un código lo más espaciado con el lenguaje estándar

5. Use estándares en el desarrollo del software
 6. Use técnicas adecuadas y actualizadas
 7. Opte por un producto mantenible sobre uno eficiente.
 8. Utilice técnicas de Ingeniería de Software adaptándolas para su aplicación en su propio medio ambiente del mantenimiento.
1. Directrices para el control del proyecto de mantenimiento de software:
 - a. Defina explícitamente las metas y prioridades del grupo de mantenimiento.
 - b. Prepare un plan de soporte o mantenimiento y úselo para administrar el proceso
 - c. Organice revisiones periódicas de requerimientos, directamente con el usuario
 - d. Trabaje con grupos de mantenimiento con poca gente de buen nivel
 - e. Lleve a cabo controles y contabilice lo efectuado por cada grupo de mantenimiento.
 - f. Intente desarrollar procedimientos ad-hoc para la administración del mantenimiento, dadas las condiciones de su instalación
 - g. Evalúe el mantenimiento con base en sus metas.
 2. Directrices para control de lo esperado en un proyecto de mantenimiento del software
 - a. Defina las tareas de Mantenimiento en términos de requerimientos ocasionales y futuros
 - b. Opte por la claridad y sencillez, aun sobre la integridad
 - c. Piense en un software mantenible aun después de modificarlo

- d. Revise periódicamente el control de calidad del producto
- e. Involucre al usuario en las revisiones
- f. Mantenga actualizada la documentación

3. Para llevar a cabo un plan de mantenimiento adecuado interactúe con el usuario final y contemple los siguientes puntos:
 - a. Un procedimiento para reporte y corrección de fallas en software
 - b. Un procedimiento de cambios y su implantación
 - c. Requerimientos de protección para el mantenimiento y control de calidad de la aplicación en general
 - d. Un plan de revalidación, que incluye pruebas.
 - e. Un plan de actualización de la documentación
 - f. Requerimientos de soporte técnico que incluyan, por ejemplo: herramientas técnicas disponibles, configuraciones, soporte e interfase con el usuario, facilidades para prueba del sistema, manuales de usuario y de operación y documentación en general.
 - g. Procedimientos para liberación de sistemas
4. Directrices a seguir durante el análisis de requerimientos para el mantenimiento
 - a. Defina el objetivo del requerimiento.
 - b. Defina el requerimiento en común acuerdo con el usuario.
 - c. Contemple posibles cambios y consecuencias.
 - d. Considere los efectos de las modificaciones.

- e. Determine compatibilidades presentes y futuras.
- f. Justifique las modificaciones en términos de costo, tiempo y riesgo.
- g. Es necesario que el usuario apruebe antes de proceder definitivamente.

5. Directrices para la especificación del proceso de mantenimiento

- a. Desarrolle especificaciones para las modificaciones, siguiendo los mismos estándares del desarrollo del software.
- b. Describa especificaciones factibles de probar.
- c. Identifique el ámbito a modificar.
- d. Estudie el impacto de la modificación y los recursos necesarios para soportarla.
- e. Antes de proceder a diseñar, es necesario que el usuario apruebe la propuesta.

6. Directrices para la fase de diseño del mantenimiento

- a. Analizar diseños compatibles con la filosofía original del diseño.
- b. Sea sencillo en el diseño, seleccione aquél que modifique el menor número de módulos y que éstos sean los menos complejos.
- c. Documente siguiendo el estándar del diseño original.
- d. Evalúe la flexibilidad del diseño para aislar funciones y proveer interfases pensando en futuros cambios.
- e. Es necesario que el usuario apruebe antes de implantar, para dicha implantación se debe pensar en:

- i. Interpretar correctamente el diseño de la modificación.
- ii. Minimizar el impacto de la modificación en el resto del software.

7. Directrices que pretenden conservar la correcta estructura del código

- a. Use los estándares de la instalación.
- b. Siga las técnicas usadas en el análisis y diseño.
- c. Documente todos los cambios en el código y conserve las últimas versiones antes de modificar.
- d. Duplique el código en lugar de crear rutinas comunes.
- e. Codifique pensando en no degradar la mantenibilidad del software.
- f. Actualice la documentación.
- g. Revise código para conservar la calidad y los estándares.

8. Directrices para la revalidación

- a. Revalide empleando pruebas unitarias y siguiendo el orden del desarrollo.
- b. Trabaje con cada módulo, compare los resultados con los del desarrollo y busque discrepancias.
- c. Pruebe pensando en evitar cualquier tipo de consecuencia directa o indirecta.

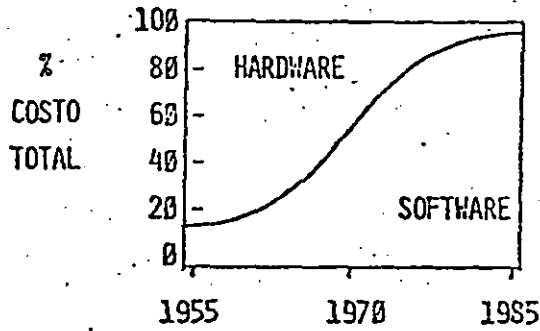
Es necesario recordar que el mantenimiento además de correcciones de problemas incluye cambios, crecimientos, adecuaciones, actualizaciones y revisiones de la aplicación para

'preservar' su función operativa y productiva de manera oportuna y veraz.

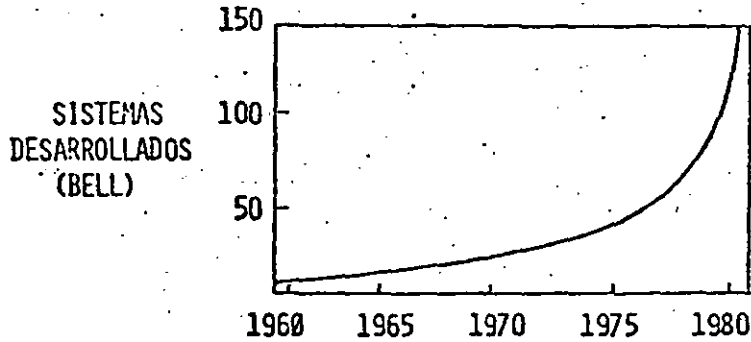
Una vez que una aplicación de software es liberada a operación, es necesario pensar que los reportes de fallas y modificaciones del sistema se presentarán como un resultado normal del uso del mismo, esto contribuirá a que el ciclo de vida del sistema sea estable y sobretodo controlable, todo esto como consecuencia de la dinámica y versatilidad de la utilización del computador, pero siempre con la idea de que es un medio y no un fin.

TENDENCIAS EN EL DESARROLLO DE SOFTWARE

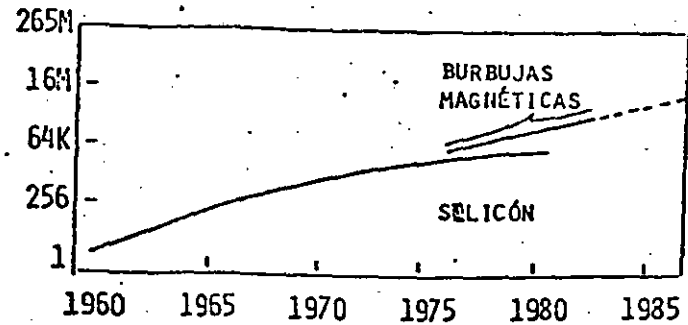
CRISIS EN EL DESARROLLO DE SOFTWARE?



\$ 40.2 x 10⁹ SOFTWARE EN EEUU (1980)



COMPONENTES



AVANCES TECNOLÓGICOS

.. REDES DE COMPUTADORAS

- 26 NODOS (4020 MINI-300 MAXI)

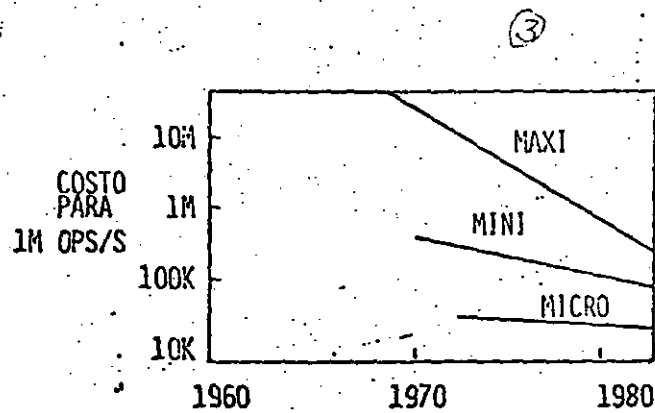
- 35~40 MILLONES DE PAQUETES/HORA

.. SISTEMAS OPERATIVOS

- 3 MILLONES DE LÍNEAS DE CÓDIGO

.. BASES DE DATOS

- 10 GBYTES



• ESTE AVANCE TECNOLÓGICO HA ORIGINADO LA CRISIS DEL SOFTWARE.

- LA REDUCCIÓN DEL PRECIO DE CÓMPUTO ORIGINA PRESIONES ECONÓMICAS HACIA UNA MAYOR EXPLOTACIÓN DEL HARDWARE POR MEDIO DE SOFTWARE.

- LA REDUCCIÓN DEL PRECIO HACE DISPONIBLE A UN NÚMERO MAYOR DE APLICACIONES EL EQUIPO DE CÓMPUTO.

• QUE SE REQUIERE DEL SOFTWARE? ④

- .. ALTA CONFIABILIDAD
- .. ALTA MODULARIDAD
- EXTENDIBLE
- MODIFICABLE

** SE HA LOGRADO EN EL HARDWARE **

• ES MÁS DIFÍCIL EL SOFTWARE QUE EL HARDWARE?

- .. NO, NO SE TRATA DE UN PROBLEMA ENTRE EL HARDWARE Y EL SOFTWARE
- .. EN AMBOS, EL PROBLEMA CENTRAL ES LA COMPLEJIDAD DEL SISTEMA TOTAL.

VARIABLES	COMPLEJIDAD	
	COMBINACIONES	FUNCIÓNES
1	2	4
2	4	16
3	8	256
4	16	65536
5	32	≈ 4265000000

000179

.. (S) HASTA AHORA EL ÚNICO ÉXITO PARA CONTROLAR LA COMPLEJIDAD HA SIDO LA PARTICIÓN DEL SISTEMA EN SUB-SISTEMAS DE MENOR -- COMPLEJIDAD: (S)

3

- DIVIDE Y VENCERÁS
- STEPWISE REFINEMENT

Y BAJO EL CONTROL DISCIPLINADO DEL DESARROLLO Y ACOPLAMIENTO DEL SISTEMA.

.. PROCESO DE DESARROLLO DE SOFTWARE INVOLUCRA:

- NECESIDADES DEL SISTEMA Y ESPECIFICACIÓN.
- ANÁLISIS DEL SISTEMA
- DISEÑO DE CADA SUBSISTEMA
- CODIFICACIÓN
- PRUEBAS PARCIALES DE LOS SUBSISTEMAS.
- INTEGRACIÓN DEL SISTEMA
- PRUEBAS DEL SISTEMA
- DOCUMENTACIÓN

.. HERRAMIENTAS PARA EL DESARROLLO DE SOFTWARE:

.. REQUERIMIENTOS Y DISEÑO

- LENGUAJES DE ESPECIFICACIÓN
- SIMULADORES

(6)
ESTADO: INCOMPLETO
REQUIERE ESPECIALIZACION

IMPLEMENTACION

- LENGUAJES DE ALTO NIVEL

ESTADO: SATISFACTORIO
REQUIERE ESPECIALIZACION

PRUEBAS Y DEPURACION

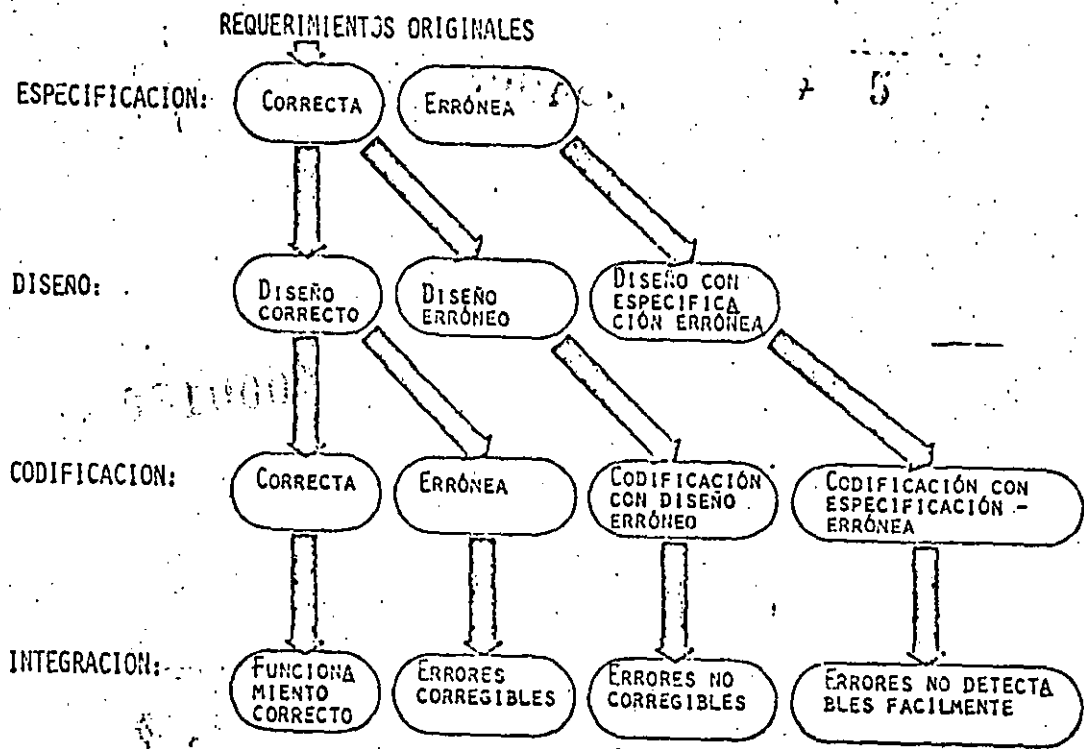
- SIMULADORES
- GENERADORES DE PRUEBAS
- DEPURADORES DE ALTO NIVEL

ESTADO: EN VIAS DE CONTROLARSE
REQUIERE ESPECIALIZACION

PAQUETES

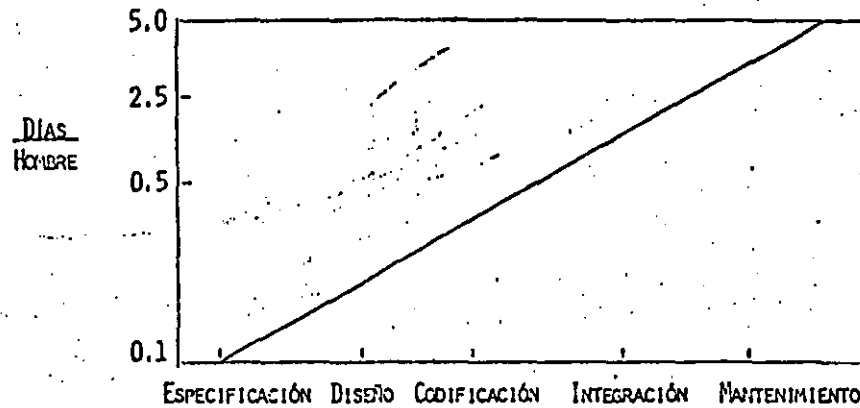
- PAQUETES DE USO GENERAL
- PAQUETES DE USO ESPECIFICO

*ESTADO: EN DIFUSION
MAS CERCANOS AL USUARIO FINAL

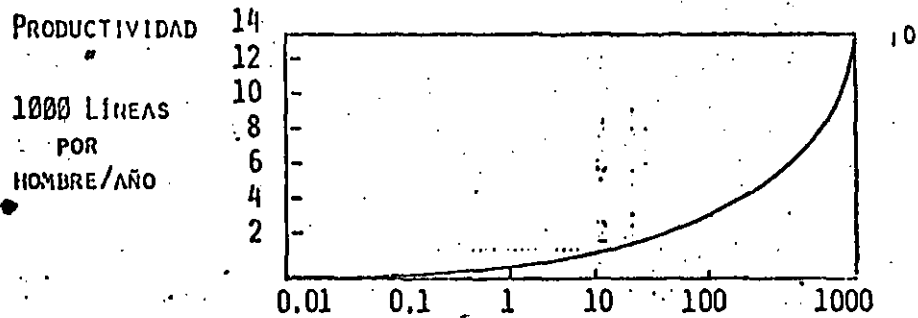


ACUMULACION DE ERRORES

13	50	60	70	80	90
					CURVA DE LA IN- BILIDAD HUMANAS
				SISTEMAS AVAN- ZADOS DE IN- FORMACION	SF
		PROCESAMIENT O DE DATOS	SISTEMAS DE INFORMACION		
	OPERACIONES ARITMETICAS				
BULBOS	TRANSISTOR	IC	LSI	VLSI	UNIONES DE JOSEPHSON



TIEMPO ESTIMADO PARA CORREGIR PROBLEMAS.



CALIDAD = ERRORES/1000 LÍNEAS

RELACION ENTRE PRODUCTIVIDAD: CALIDAD

MEJORAR PRODUCTIVIDAD:

- LENGUAJES DE ALTO NIVEL
- SISTEMAS OPERATIVOS AVANZADOS
- SISTEMAS AUTOMATIZADOS EN LA AYUDA AL DISEÑO
- USO DE MÓDULOS EXISTENTES
- DISCUTIR PROBLEMAS ABIERTAMENTE

MEJORAR CALIDAD:

- DEDICAR MÁS ESFUERZO A LAS ETAPAS DE ESPECIFICACIÓN Y DISEÑO.
- USAR MÓDULOS YA EXISTENTES

CONCLUSIONES

• NUEVA TECNOLOGIA DE SEMICONDUCTORES



• MENORES COSTOS DEL HARDWARE



• DEMANDA AL CONTROL POR SOFTWARE



• MAYOR CONFIABILIDAD Y USO MAS SENCILLO



• AUMENTAR PRODUCTIVIDAD Y CALIDAD DEL SOFTWARE

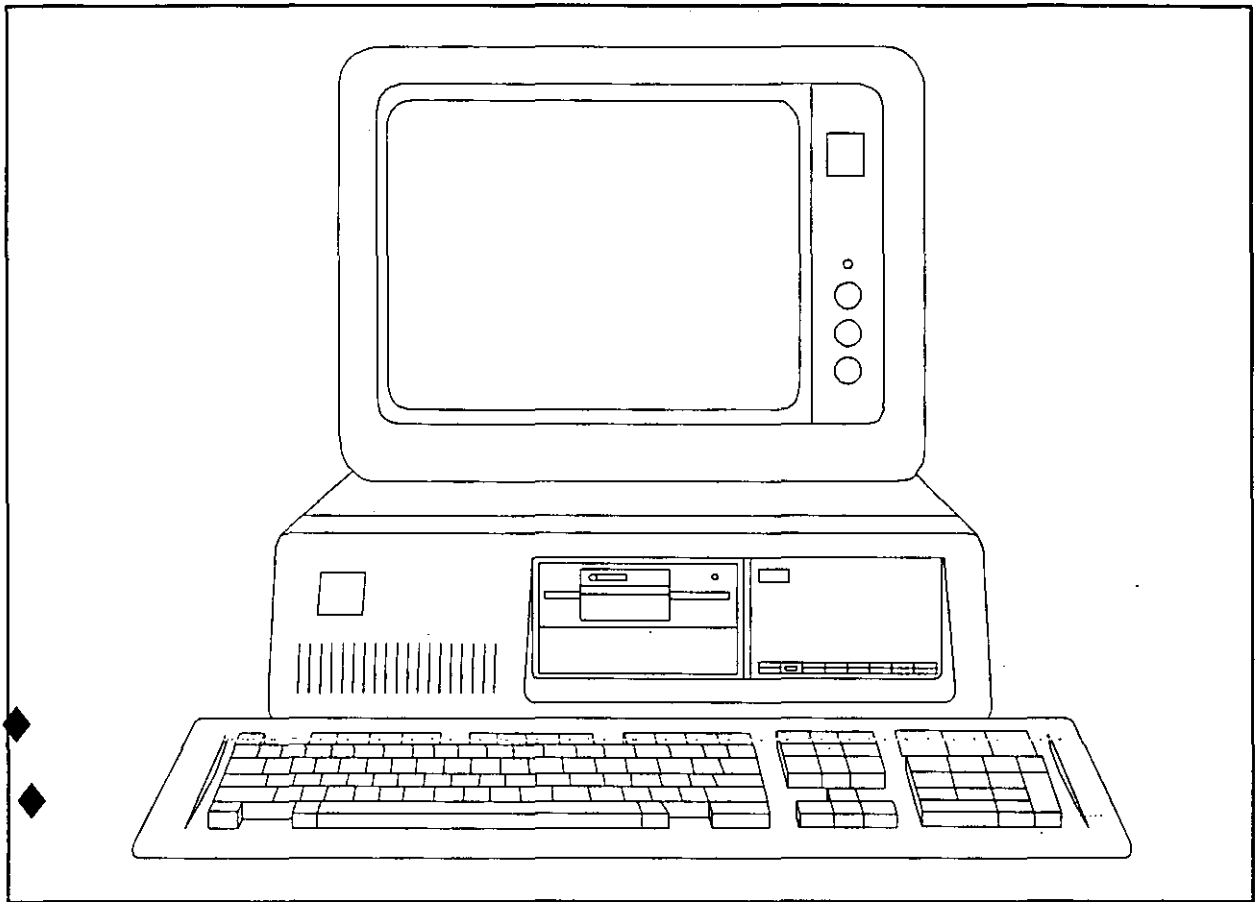
= RESPUESTA =

• LENGUAJES DE MAS ALTO NIVEL

• MAS AUTOMATIZACION

• BIBLIOTECAS DE MODULOS

• MAYOR ENFASIS EN LAS ETAPAS DE ESPECIFICACION Y DISERO.



EJEMPLOS Y PRACTICAS

(4)

EJEMPLO DE APLICACION UTILIZANDO EL PAQUETE
DE COMPUTACION LOTUS-123

Indicador de celda, panel de control e indicador de modo.

Señalamos que la celda A1 esta sobreiluminada en la hoja de trabajo. Una sobreiluminación móvil, conocida como indicador de celda, está actualmente en A1.

Si se pulsa la tecla con flecha derecha, el indicador se mueve una celda hacia la derecha, o B1.

Ingresos de Harapos-para-los-Rico.

Hace dos años Patricio y Craso Rico iniciaron el lucrativo negocio de fabricar bayetas para pintores. Las bayetas se producían grabando la etiqueta PCR en harapos de segunda mano. Las bayetas para pintores tuvieron mucho éxito, tanto que los innovadores denominaron su empresa Harapos-para-los-Rico y sus ingresos crecieron hasta 10 millones durante su primer año de operación.

Los Rico creían que la demanda de bayetas continuaría siendo fuerte durante varios años. Les gustaría tener una predicción (o pro forma) de ingresos para conocer con anticipación los niveles de ventas, gastos y beneficios durante los próximos cinco años. El modelo de pro forma debe permitir reflejar los cambios en las hipótesis con respecto a las ventas y a los gastos, dado que raramente se podrá conocer con absoluta certeza el futuro financiero de un negocio.

Introducir títulos de la hoja de trabajo.

1. Utilizar las teclas con flecha para llevar al indicador de celda hasta C1.
2. Pulsar la tecla CAPS LOCK cerca de la parte inferior derecha del teclado, para escribir con mayúsculas.
3. Teclar el título CUENTA DE RESULTADOS y pulse la tecla Enter.

Al pulsar la tecla Enter se almacena el título en la celda C1, y el 1-2-3 regresa al modo Activo (Ready).

4. Ir a la celda B2 pulsando la tecla F5 (Goto) y teclar B2.
6. Teclar HARAPOS-PARA-LOS RICO, S.L., y a continuación teclar Enter para almacenar el título en la celda.

Utilización de Modo de Edición.

Dado que la segunda línea del título es más larga que la primera, la empezamos una columna más a la izquierda (en columna B) esperando quede centrada debajo de la primera línea del título.

Necesitamos insertar dos espacios delante de HARAPOS-PARA-LOS-RICO, S.L., para que los títulos queden centrados. Esto se consigue utilizando el modo Edición.

1. Con el indicador aún en la celda B2, pulsar la tecla F2 (Edit) para pasar al modo Edit.
2. Pulsar la tecla Home.
3. Pulsar una vez la tecla con flecha derecha para mover el cursor, un espacio hacia la derecha del prefijo de título.
4. Pulsar seis veces la barra espaciadora para insertar seis espacios.

En la línea 2 del panel de control aparece el contenido revisado,

que es:

HARAPUS-PARA-LOS-RICO, S.A.

5. Pulsar la tecla Enter para introducir el título corregido.

Introducir ~~cabeceras~~ en las columnas.

1. Utilizar la tecla F5 (Goto) para desplazarnos a la celda B4.

Aquí introduciremos la cabecera del año 0.

2. Pulsar la tecla 0 (cero) situada en el renglón superior del teclado.

3. Pulsar la tecla Enter para completar la entrada de datos.

Fijémonos en el panel de control. Su primera línea es B4:0
Observemos que no hay prefijos para los valores, a diferencia de lo que ocurre con los títulos.

Una fórmula para encabezamientos numéricos de columnas.

1. Ir a la celda C4.

2. Pulsar la tecla 1 de la fila superior del teclado.

3. Teclar +B4 y pulsar la tecla Enter.

Cómo introducir la orden Copy.

La orden Copy duplica un rango de celdas desde una posición de la hoja de trabajo (la fuente) a otra (el destino). Con las órdenes y funciones del 1-2-3 se utilizan frecuentemente rangos de celdas, de modo que antes de pasar al experimento de copiar vamos a dedicar un momento a hablar de rangos.

Rangos de Celdas.

Un rango es un rectángulo de celdas. Este rectángulo puede ser una fila de dos o más celdas adyacentes, una columna de celdas adyacentes o un bloque que contenga dos o más filas y columnas. Una sola celda también se denomina rango, es un bloque formado por una sola fila y una sola columna.

El Lotus 1-2-3 identifica un rango por sus coordenadas primera y última. Hay varias formas de identificar un rango. La que utilizaremos aquí consiste en introducir las direcciones de los dos extremos del rango, separadas por un punto (.). La fila de celdas desde A1 hasta J1 se introduce como A1.J1. La columna desde A1 hasta A15 se introduce como A1.A15. El bloque rectangular de celdas A1, J1, J15, A15 se puede introducir como A1.J15 o como J1.A15.

Cuando el 1-2-3 ve cualquiera de estos conjuntos de coordenadas, se percata de que la anchura del rango se extiende desde las columnas A a J, y la longitud del rango va de 1 a 15.

Cómo copiar una sola celda.

1. Posicionar el indicador en la celda C4, la fuente.
2. Teclar /C para invocar la orden Copy.
3. Pulsar Enter para fijar el rango fuente.
4. Teclar D4.
5. Pulsar enter.

Copiado Relativo.

Para obtener las cabeceras para los años 3,4, y 5, simplemente utilizar la orden Copy para duplicar la fórmula de la celda D4 en las celdas desde E4 hasta U4.

1. Llevar el indicador hasta la celda D4.

A continuación, iniciar la orden.

2. Teclar /C

El 1-2-3 presupone D4..D4 como fuente, lo que es correcto.

3. Pulsar la tecla Enter.
4. Teclar E4..G4.
5. Pulsar la tecla Enter.

Las cabeceras de las columnas de los años a predecir ya se han completado.

Repetir títulos y subrayados.

1. Pulsar la tecla F5 (Goto) y, a continuación, teclar B5 y pulsar Enter.
2. Pulsar la tecla \
3. Pulsar la tecla - y a continuación Enter.
4. Teclar /C
5. Pulsar Enter para aceptar B5 como fuente.

El destino es el rango desde C5 a G5, por lo tanto

6. Teclar C5..G5 y pulsar Enter.

Guardar y recuperar Archivos :

1. Teclar /FS

Como respuesta a ello, el programa pide un nombre de archivo.

2. Teclar CR y pulsar la tecla Enter.

Escogemos CR como abreviatura de Cuenta de Resultados. Podríamos haber escogido cualquier nombre. Sin embargo, es mejor escoger un nombre relacionado con el contenido del archivo.

Para recuperar el archivo, lanzar el orden de File Retrieve.

1. Teclar /FR

A continuación, el 1-2-3 preguntará el nombre del archivo a recuperar.

2. Seleccionar el nombre del archivo en la línea del panel de control pulsando Enter o teclar explícitamente el nombre del archivo y pulsar Enter.

Construir la previsión de Ingresos de Harapos-para-los-Rico.

Los gestores predicen que el crecimiento de los ingresos, año a año, será el siguiente:

AÑO	CRECIMIENTO CON RESPECTO AL AÑO ANTERIOR
1	25%
2	30%
3	20%
4	10%
5	10%

1. Ir a la celda A6, e introduzca el título de fila de INGRESOS.
2. Llevar el indicador a los INGRESOS del año 0 (celda B6).
3. Introducir 10000 (sin punto), que representa 10 millones de dólares de ventas en el año anterior.
4. Ir a la celda F1.
5. Teclar (\$ EN MILES) y pulsar la tecla Enter.
6. Mover el indicador a la celda, C7, la celda que está debajo de INGRESOS del año 1.
7. Teclar .25 y pulsar Enter para almacenar un 25% de porcentaje de crecimiento.

Introducir fórmulas Señalándolas.

1. Ir a la celda de INGRESOS del año 1 (C6).

La fórmula es (1+ celda de abajo) * celda Izquierda.

2. Teclar (1+
3. Pulsar la tecla con flecha abajo una vez para mover el indicador hasta la celda de porcentaje de crecimiento anual (C7).
4. Teclar) para completar la primera expresión de la fórmula.
5. Teclar * para indicar multiplicación.
6. Pulsar una vez la tecla con flecha izquierda para señalar la celda INGRESOS del año 0 (B6).

El panel de control indica ahora $(1+ C7)* B6$.

7. Pulsar la tecla Enter.

El resultado que aparece en la hoja de trabajo es 12500, o unos ingresos de 12.5 millones en el año 1.

Copiar un rango señalado.

1. Introducir los porcentajes de crecimiento anual .30, .20, .10 y .10 en la celdas de porcentaje de crecimiento anual, (D7, E7, y G7) (respectivamente). En lugar de pulsar la tecla Enter tras cada entrada, se puede pulsar la tecla con flecha derecha, que tiene la doble función de almacenar los valores y de mover el indicador hacia la derecha.
2. Llevar el indicador a la celda de INGRESOS del año (C6), la fuente desde la que ha de copiar.
3. Teclar /C para iniciar la orden.
4. Pulsar Enter para designar la celda C6 como la fuente.
5. Llevar el indicador a la celda INGRESOS del año 2 (D6), la primera celda en la fila de celdas de destino.
6. Teclar un punto , para fijar este valor.
7. Llevar el indicador hasta la celda de INGRESOS del año 3 (G6), la última celda del rango.
8. Pulsar la tecla Enter para activar la copia.

Utilizar la orden de Range Format.

La orden Range Format asigna características de escritura (formato) a una o más celdas.

1. Ir a la primera celda a la que se haya de dar formato, el porcentaje de crecimiento del año 1 (C7).
2. Teclar /R
3. Pulsar F para escoger Format.

La orden Format invoca a su vez un menú.

4. Teclar P para seleccionar la opción Percent del menú y pulsar la tecla Enter.
5. Pulsar la tecla END.
6. Pulsar la tecla con flecha derecha.
7. Pulsar Enter para activar la orden.

Utilizar la orden de Column de la Hoja de trabajo.

1. Llevar el indicador hasta la celda debajo del rótulo INGRESOS (A7).
2. Pulsar 3 veces la barra espaciadora.
3. Teclar % CRECIMIENTO SOBRE EL AÑO ANTERIOR.
4. Pulsar Enter.

Para ampliar o contraer la anchura de cualquier columna de la hoja de trabajo, se utiliza la orden WORKSHEET Column. Con el indicador en la columna A cuya anchura se quiere cambiar.

5. Teclar /WC
6. Escoger la opción Set-Width (fijar ancho) pulsando la tecla S o la tecla Enter.
7. Pulsar la tecla con flecha derecha repetidamente hasta que el panel de control indique una anchura de 38; a continuación, pulsar la tecla Enter ó bien oprimir la tecla ESC y teclar valor igual: 38 y posteriormente pulsar la tecla Enter.

Recalcular Proyecciones automáticamente.

Una de las prestaciones principales de una hoja electrónica es el recálculo automático.

1. Llevar el indicador a la celda C7.
2. Teclar .2 y pulsar la tecla Enter. Con esto se cambia el porcentaje de crecimiento del año 1 a 20%.

Calcular el costo de la producción.

- Costo de las materias primas como porcentaje fijo.

1. Ir a la celda A9 e introducir el título GASTOS DE PRODUCCION.
2. Ir a la celda A10.
3. Pulsar la barra espaciadora tres veces, teclar MATERIALES y pulsar la tecla Enter.

La fórmula para MATERIALES del año 1 es $0.16 * \text{INGRESOS del año 1}$.

4. Ir a la celda C10.
5. Teclar .16% para iniciar la fórmula.

6. Llevar el indicador a la celda C6.

7. Pulsar la tecla Enter.

Esta fórmula se repetirá para los años previstos.

8. Teclee /C para iniciar la orden.

9. Pulsar la tecla Enter para aceptar la posición presente del indicador (C10) como fórmula fuente.

10. Llevar el indicador al comienzo del rango de destino (MATERIALES del año 2, en la celda D10).

11. Pulse la tecla . para fijar el rango.

12. Llevar el indicador al final del rango de destino (celda C10).

13. Pulsar la tecla Enter.

Utilización de Inserciones en la hoja de trabajo para crear espacio extra.

1. Llevar el indicador a una celda de la columna A pulsando la tecla Home.

2. Teclar /W

3. Seleccionar la opción Insert tecleando I.

4. Seleccionar la opción Column de este menú tecleando C.

5. Pulsar la tecla Enter para ejecutar la inserción.

6. Ir a la celda A4 e introducir el título PORCENT.

7. En A5, teclar ^- para subrayar el título.

Guardar la Hoja de trabajo.

1. Teclar /FS

El 1-2-3 escogerá automáticamente el último nombre de archivo utilizado para guardar la hoja de trabajo.

2. Pulsar Enter para aceptar CR como nombre de archivo.

3. Seleccionar Replace para sustituir la versión original.

Cálculo de los costos de producción.

Utilizar la tecla F4 (Abs) para la fórmula de los salarios.

1. Ir a la celda B11, debajo de las etiquetas MATERIALES, e introducir la etiqueta adentrada SALARIOS pulsando tres veces la barra espaciadora y tecleando SALARIOS.

2. Ir una vez más hacia la izquierda a la celda PORCENT (A11) e introducir .14 como porcentaje.
3. Ir a la celda SALARIOS del año 1 a la derecha (D11).
La fórmula a introducir aquí es porcentaje*ingresos.
4. Teclar + para iniciar la fórmula.
5. Llevar el indicador a la celda PORCENT (A11).
6. Pulsar una vez la tecla F4 (Abs).
7. Pulsar de nuevo la tecla F4 (Abs).
8. Pulsar otra vez la tecla F4 (Abs).
9. Teclar * para continuar la fórmula de multiplicación.
10. Llevar el indicador a la celda de INGRESOS del año 1 (D6).
11. Pulsar F4 tantas veces como lleve el llegar a D#6 (eso exige pulsar dos veces).
12. Pulsar la tecla Enter.

La fórmula para los salarios es ahora \$A11*D#6 que da como resultado 1680. Ahora se puede copiar la fórmula en el resto de la fila.

13. Teclar /C y Enter para activar la orden Copy y elegir la posición actual del indicador como fuente.
14. Ir hacia la derecha una vez para señalar la celda de SALARIOS del año 2 (E11), como comienzo del rango destino.
15. Pulsar la tecla . para fijar el principio del rango.
16. Llevar el indicador a la celda de SALARIOS del año 5 (H11), el final del rango destino.
17. Pulsar la tecla Enter.

Cálculo de los beneficios marginales.

1. Introducir la etiqueta adentrada BENEFICIOS MARGINALES debajo de la etiqueta SALARIOS. (Ir a la celda B12, pulsar 3 veces la barra espaciadora y teclar BENEFICIOS MARGINALES).
2. Ir a la izquierda e introducir .15 en la celda PORCENT (A12)
3. Ir a la celda BENEFICIOS MARGINALES del año 1 (D12).
4. Introducir + como inicio de la fórmula.
5. Llevar el indicador a la celda de PORCENT (A12).
6. Pulsar la tecla F4 (Abs) para convertir en absoluta la referencia a la celda (se convertirá en \$A12).

000195

7. Pulsar * para continuar la fórmula de multiplicación.
8. Señalar hacia la celda de SALARIOS del año 1 (D11) llevando el indicador un espacio hacia arriba.
9. Pulsar la tecla Enter.

Ahora se va a copiar la fórmula en los restantes años.

10. Teclar /C y pulsar la tecla Enter.
11. Llevar el indicador a la celda de BENEFICIOS MARGINALES del año 2 (E12).
12. Pulsar . para fijar el principio del rango destino.
13. Llevar el indicador a la celda de BENEFICIOS MARGINALES del año 5 (H12).
14. Pulsar la tecla Enter.

Cálculo de otros ingresos.

1. Ir a la celda debajo de la etiqueta BENEFICIOS MARGINALES (B13), pulsar tres veces la barra espaciadora e introducir OTROS.
2. Ir a la celda PORCENT (A13) e introducir .08 como porcentaje de crecimiento.
3. Llevar el indicador a la celda OTROS del año 1 (D13).
4. Introducir 100 como valor inicial.
5. Llevar el indicador a la celda OTROS del año 2 (E13).
6. Introducir (+ para iniciar la fórmula.
7. Llevar el indicador a la celda PORCENT (A13).
8. Pulsar la tecla F4 (Abs) tres veces, de manera que la fórmula que aparezca en el panel de control sea (1+\$A13).
9. Teclar)* para continuar la fórmula.
10. Llevar el indicador a la celda de OTROS gastos del año anterior (D13).
11. Pulsar la tecla Enter.
12. Con el indicador en E13, teclar /C y pulsar Enter.
13. Pulsar una vez la tecla con flecha derecha.
14. Pulsar la tecla . para indicar que es un rango.
15. Llevar el indicador a la celda OTROS del año 5 (H13).
16. Pulsar la tecla Enter.

Anotaciones en la hoja de trabajo.

1. Llevar el indicador a la celda J4.
2. Teclar /WCS 17.
3. Pulsar la tecla Enter.
4. Teclar ^NOTAS y pulsar la tecla Enter.
5. Descender una celda hasta J5 y teclar \- Enter.
6. Ir a la celda J10 e introducir el título de % DE LAS VENTAS; a continuación en la celda J12 introducir el título % DE SALARIOS.
8. Descender una celda de % DE SALARIOS (J13) e introducir % DE PORCENTAJE.
9. Ir a la celda I1 y teclar /WCS 2 y pulsar la tecla Enter.

Cálculo de la Compensación por la gestión empresarial.

1. Ir a la celda B15 e introducir el título GASTOS GENERALES Y ADMINISTRATIVOS.
2. Descender una posición a la celda E16, pulsar tres veces la barra espaciadora para adentrar el título y teclar COMPENSACION:OFICINA.

Copiar de un rango a otro.

1. Ir a la celda COMPENSACION:OFICINA del año 1 (D16) e introducir el valor 1200.
2. Ir a la celda PORCENT (A16) e introducir .1
3. Ir al comienzo del rango fuente, la celda OTROS del año 2 (E13).
4. Teclar /C para llamar a la orden Copiar.
5. Pulsar la tecla END, seguido de la tecla con flecha derecha para transferir el indicador hasta la celda OTROS del año 5 (H13).
6. Pulsar la tecla Enter para fijar el rango fuente.
7. Llevar el indicador a la celda COMPENSACION:OFICINA del año 2 (E16).
8. Pulsar la tecla Enter para activar el copiado.
9. Ir a la celda NOTAS (J16) e introducir % DE PORCENTAJE.

Cálculos de las compensaciones por ventas.

1. Ir a la celda B17 e introducir el título adentrado COMPENSACION:VENTAS.
2. Introducir .08 en la celda PORCENT (A17).
3. Ir a SALARIOS del año 1, el principio del rango fuente (celda B11).
4. Teclar /C.

El 1-2-3 pregunta por el final del rango fuente.

5. Pulsar END, seguido de la tecla con flecha derecha para llevar el indicador a la celda de SALARIOS del año 5 (H11).
6. Pulsar dos veces la tecla con flecha derecha para ampliar el rango fuente hasta incluir la celda NOTAS (J11).
7. Pulsar la tecla Enter para fijar el rango fuente.
8. Descender hasta la celda COMPENSACION:VENTAS del año 1 (D17), el comienzo del rango destino y teclar un punto.
9. Pulsar Enter.

Beneficios Marginales.

1. En la celda B18, introducir el título adentrado BENEFICIOS MARGINALES.
2. Ir a la columna PORCENT (celda A18) e introducir .17.
3. Ir a la celda de BENEFICIOS MARGINALES del año 1 (D18).

La fórmula es

porcentaje*(compensación por oficina+compensación por ventas):

4. Pulsar + para empezar la fórmula.
5. Ir hacia la izquierda hasta la celda PORCENT (A18).
6. Pulsar tres veces la tecla F4 (Abs), hasta obtener \$A18, que es la referencia adecuada a copiar.
7. Introducir *(
8. Llevar el indicador hasta la celda COMPENSACION:OFICINA del año 1 (D16).
9. Pulsar + para continuar la fórmula.
10. Llevar el indicador hasta la celda COMPENSACION:VENTAS del año 1 (D17).
11. Teclar) y pulsar la tecla Enter.

Publicidad y promoción, año 1.

1. Ir a la celda B19 y a continuación, introducir el título adentrado PUBLICIDAD Y PROMOCION.
2. Almacenar .025 en la celda de porcentaje (A19).
3. Ir a la celda SALARIOS del año 1 (D11), la celda fuente.
4. Teclar /C y pulsar la tecla Enter.
5. Descender hasta la celda PUBLICIDAD Y PROMOCION del año 1 (D19).
6. Pulsar la tecla Enter para iniciar la copia.

Depreciación, año 1.

1. Ir a la celda B20 e introducir el título adentrado DEPRECIACION.
2. Almacenar el valor 20 en la celda DEPRECIACION del año 1 (D20).

Gastos varios, año 1.

1. Ir a la celda B21 e introducir el título adentrado VARIOS.
2. Ir a la celda PORCENT (A21) y teclar 10.
3. Ir a la celda VARIOS del año 1 (D21).
4. Introducir + para empezar la fórmula.
5. Llevar el indicador a la celda PORCENT (A21).
6. Pulsar tres veces la tecla F4 (Abs) para hacer absoluta la referencia a la columna.
7. Teclar + para continuar la fórmula.
8. Llevar el indicador a la celda VARIOS del año 0 (D21).
9. Pulsar la tecla Enter.

Copiar de una columna a una fila.

1. Llevar el indicador a D18, el comienzo del rango fuente.
2. Teclar /C y agregar un punto.
3. Llevar el indicador a la celda VARIOS del año 1 (D21) para indicar el final del rango fuente.
4. Pulsar la tecla Enter.
5. Llevar el indicador a la celda BENEFICIOS MARGINALES del año 2 (E18) para indicar el inicio del rango destino.

6. Teclar . para indicar un rango.
7. Llevar el indicador a la celda de BENEFICIOS MARGINALES del año 5 (H18).
8. Pulsar la tecla Enter.
9. En las celdas J18, J19, J20 y J21, introducir consecutivamente los siguientes títulos : % DE COMP; % DE LAS VENTAS; ENT. DIRECTA y CRECIMIENTO CONST. .

Cálculo de los gastos de funcionamiento totales : @SUMA y Copiar.

1. Introducir un guión en la columna año 0 (celda C22) tecleando \- y pulsando la tecla enter.
2. En B23 introducir el título siguiente, adelantando tres espacios, GASTOS TOTALES DE FUNCIONAMIENTO.
3. Ir a la celda C23, la celda de GASTOS TOTALES DE FUNCIONAMIENTO del año 0 y teclar 6400.
4. Ir a la celda de GASTOS TOTALES DE FUNCIONAMIENTO del año 1 (D23).

Utilizar @SUM para obtener los gastos del año 1.

1. Con el indicador en la celda 23, teclar @SUM(
2. Señalar el comienzo del rango a sumar, la celda de MATERIALES del año 1 (D10).
3. Pulsar . para fijar el rango.
4. Llevar el indicador a la celda VARIOS del año 1 (D21).
5. Pulsar) para cerrar el rango, entre paréntesis, de la función @SUM.
6. Pulsar la tecla Enter.

Utilización de copias para los restantes gastos.

1. Llevar el indicador a la celda D22 e introducir un guión tecleando \-
2. Introducir /C y teclar un punto.
3. Ampliar el indicador hasta la celda GASTOS TOTALES DE FUNCIONAMIENTO del año 1 (D23), el final del rango fuente.
4. Pulsar Enter para acabar la introducción del rango fuente.
5. Señalar el comienzo del rango destino en la columna del año 2, celda E22.

6. Pulsar , para fijar el rango.

7. Ampliar el indicador hasta la columna del año 5 (celda H22).

8. Pulsar Enter para activar la orden.

Cálculo de los gastos por interés

1. Ir a la celda B25 e introducir el título GASTOS POR INTERESES.

2. Ir a la celda C25 e introducir el valor 10.

3. Ir a la celda J25 e introducir la nota ENT. DIRECTA.

Cálculo de los gastos pre-impuestos.

1. Ir a la celda B27 e introducir el título INGRESOS PRE-IMPUESTOS.

2. Ir al año 0 (celda C27).

3. Pulsar + para iniciar la introducción de la fórmula.

4. Llevar el indicador a la celda INGRESOS del año 0 (C6).

5. Pulsar - para continuar con la sustracción de los gastos de funcionamiento de los gastos brutos.

6. Llevar el indicador a la celda GASTOS TOTALES DE FUNCIONAMIENTO del año 0 (C23).

7. Pulsar - y llevar el indicador a la celda GASTOS POR INTERESES del año 0 (C25).

8. Pulsar la tecla Enter.

Cálculo de los ingresos netos.

1. Ir a la celda B29 e introducir el título adentrado IMPUESTOS.

2. Ir a la columna PORCENT de la línea IMPUESTOS (celda A29) y almacenar .52

3. Ir a la celda IMPUESTOS del año 0 (C29).

4. Pulsar + para empezar a introducir valores.

5. Señalar a la celda PORCENT (A29).

6. Pulsar la tecla F4 (Abs) para convertir en absoluta la referencia a PORCENT, para copiarla posteriormente.

7. Teclar * para multiplicar este porcentaje por los ingresos pre-impuestos.

8. Llevar el indicador a la celda INGRESOS PRE-IMPUESTOS del año 0 (C27).

000201

9. Pulsar la tecla Enter.
10. En la celda NOTAS (J29), teclear % DE INGR.PRE-IMP.
11. Ir a la celda B31 y teclear INGRESOS NETOS.
12. Ir a la celda INGRESOS NETOS del año 0 (C31) y teclear + para empezar la fórmula.
13. Señalar a la celda INGRESOS PRE-IMPUESTOS del año 0 (C27).
14. Teclear - para continuar.
15. Señalar a la celda IMPUESTOS del año 0 (C27).
16. Pulsar la tecla Enter.

Cálculo de los beneficios para los restantes años.

1. Llevar el indicador a la primera celda del rango fuente, la celda de GASTOS POR INTERESES del año 0 (C25).
2. Teclear /C y teclear un punto para iniciar la orden Copiar.
3. Ir a la celda INGRESOS NETOS del año 0 (C31).
4. Pulsar la tecla Enter para fijar el rango fuente.
5. Llevar el indicador a la celda GASTOS POR INTERESES del año 1 (D25), la primera celda del rango de destino.
6. Pulsar . para establecer el comienzo del rango de destino.
7. Llevar el indicador a la celda GASTOS POR INTERESES del año 5 (H25).
8. Pulsar la tecla Enter para activar la orden Copy.

Toque final : Modificar los formatos numéricos.

1. Teclear /WGF
2. Seleccionar la opción Fixed del menú Format.
3. Teclear 0 (cero) y pulsar la tecla Enter.
4. Llevar el indicador a la primera celda de la columna PORCENT (celda A6).
5. Teclear /RFP
6. Pulsar la tecla Enter.
7. Pulsar la tecla PgDn, seguido de la tecla con flecha abajo, saltando con ello hasta el final de la columna a continuación de la última entrada, INGRESOS NETOS, hasta la celda A31.

8. Pulsar la tecla Enter.

9. ~~Teclar~~ Pulsar un . e ir hasta la celda PORCENT de VARIOS (A21).

10. Teclar /RFG y pulsar la tecla Enter.
Guardar el archivo.

11. Teclar /FS pulsar la tecla Enter, teclar Replac para reemplazar el archivo CR por la hoja de trabajo actual.

Impresión de la hoja de trabajo.

1. Alinear el papel de la impresora en la parte superior de la página; prender la impresora.

2. Pulsar la tecla Home para llevar el indicador hasta la celda A1.

3. Teclar /P para lanzar la orden imprimir.

4. Seleccionar PRINTER para enviar la salida a la impresora.

5. Teclar R para seleccionar la opción Range.

6. Pulsar . o continuación de la tecla END y posteriormente, la tecla Home.

7. Pulsar la tecla Enter para aceptar este rango.

9. Escoger la opción Align Co.

10. Teclar /FS y pulsar Enter; a continuación, teclar R para sustituir la versión anterior del archivo CR que no contenía los valores fijados para la impresión.

Representar gráficamente la fila de ingresos globales.

1. Teclar /G para llamar al menú de la orden Graph.

El menú Graph es como sigue:

Type X A B C D E F Reset View Save Options Name Quit

2. Teclar A para designar un rango de datos como el rango A.

3. Llevar el indicador a la celda Ingresos del año 0 (C6).

4. Teclar . para fijar el rango.

5. Pulsar la tecla END y a continuación la tecla con flecha derecha para llevar el indicador hasta la celda INGRESOS del año 5 (H6).

6. Pulsar la tecla Enter.

7. Escoger la opción View del menú Graph.

8. Pulsar cualquier tecla para volver al menú Graph y a la hoja del trabajo.

9. Teclar X para designar el rango X.

10. Llevar el indicador al título de la columna 0 en la celda C4.

11. Pulsar la tecla . para fijar el rango.

12. Llevar el indicador al título de columna 5 en la celda H4.

13. Pulsar la tecla Enter.

14. Teclar V para visualizar la gráfica.

15. Pulsar cualquier tecla para volver al menú Graph.

Utilizar las opciones Gráficas.

1. Teclar la letra O para escoger Options.

2. Escoger la opción Titles tecleando T.

3. Teclar F para introducir la primera línea del título de una gráfica.

4. Teclar PREDICCION FINANCIERA y pulsar la tecla Enter.

5. Escoger de nuevo la opción Titles.

6. Escoger S para introducir el segundo título.

7. Teclar GLOBAL DE LA COMPANIA y pulsar la tecla Enter.

8. Escoger de nuevo Titles, y a continuación, la opción X-Avis.

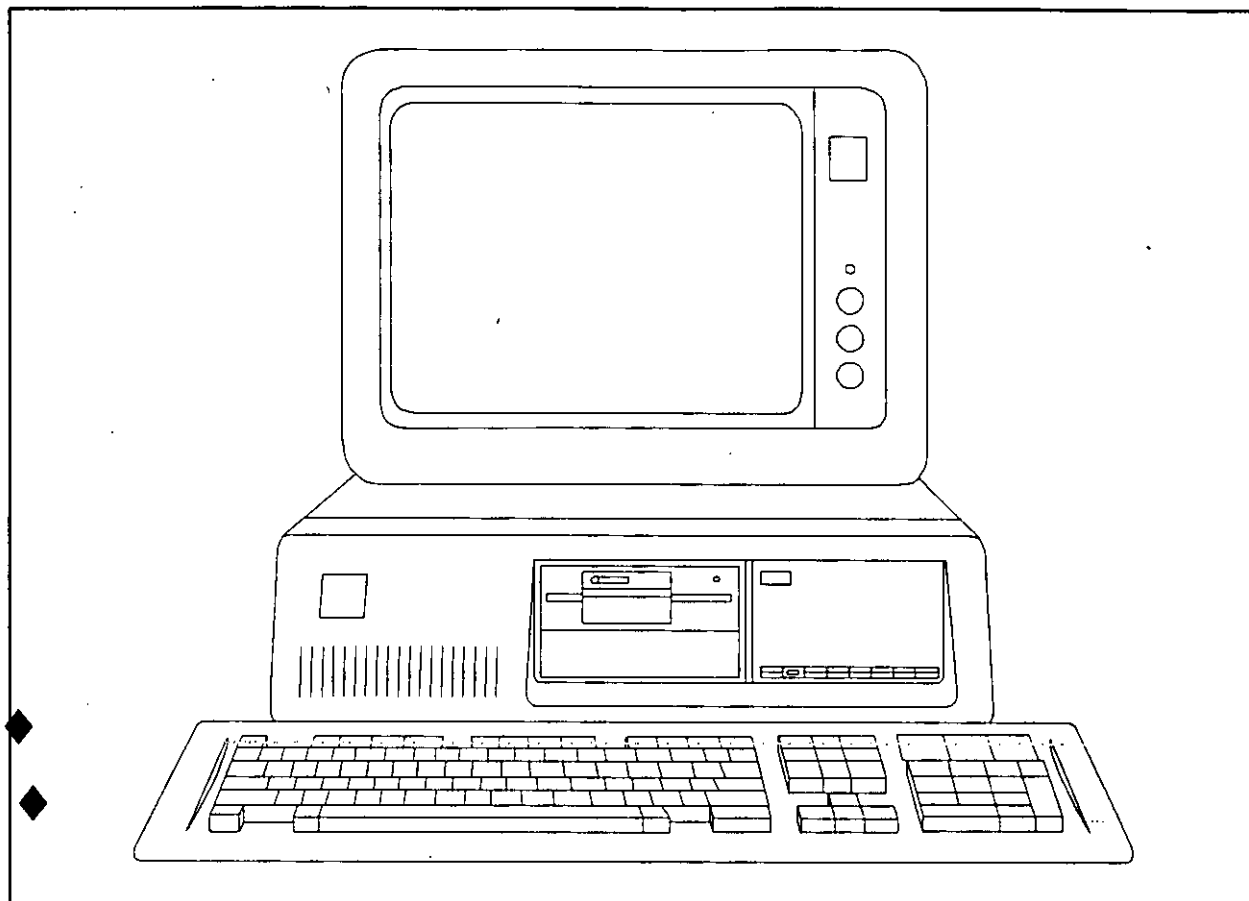
9. Teclar años y pulsar la tecla Enter.
10. Teclar T y, a continuación, y para escoger la opción Y-Axis.
11. Teclar INGRESOS y pulsar la tecla Enter.
12. Escoger la opción Quit
13. Teclar V para visualizar el gráfico.

5.000 Otras opciones gráficas.

1. Pulsar cualquier tecla para regresar al menú Gráfico.
2. Seleccionar Type a continuación Bar y a continuación visualizar.
3. Pulsar cualquier tecla para volver a la hoja de trabajo.
4. Pasar al modo Ready escogiendo la opción Quit del menú Graph.
5. Guardar la hoja de trabajo tecleando /FS Enter R y pulsar la tecla Enter.

CUENTA CON RESULTADOS (EN MILES)
HARAJOS-PARA-LOS-RICOS, S.L.

PORCENT.	0	1	2	3	4	5	NOTAS
INGRESOS	10000	12500	16250	19500	21450	23595	
% CRECIMIENTO SOBRE EL AÑO ANTERIOR		25.00%	30.00%	20.00%	10.00%	10.00%	
GASTOS DE PRODUCCION							
14.00%		2000	2600	3120	3432	3775	% DE LAS VENTAS
15.00%		1750	2275	2730	3003	3303	
8.00%		263	341	410	450	495	% DE SALARIOS
8.00%		100	108	117	126	136	% DE PORCENTAJE
GASTOS GENERALES Y ADMINISTRATIVOS							
10.00%		1200	1320	1452	1597	1757	% DE PORCENTAJE
8.00%		1000	1300	1560	1716	1888	
17.00%		374	445	512	563	620	% DE COMP
2.50%		313	406	488	536	590	% DE LAS VENTAS
		20	20	20	20	20	ENT. DIRECTA
		0	0	0	0	0	CRECIMIENTO CONS.
GASTOS TOTALES DE FUNCIONAMIENTO							
	6400	7019	8816	10408	11444	12584	
GASTOS POR INTERESES							
	10	10	10	10	10	10	ENT. DIRECTA
INGRESOS PRE-IMPUESTOS							
	3590	5471	7424	9082	9996	11001	
52.00%		1867	2845	3861	4723	5198	% DE ING. PRE-IMP.
INGRESOS NETOS							
	1723	2626	3564	4360	4798	5280	



LA COMPUTACION EN MEXICO

ARTICULOS VARIOS

E INFORMACION COMERCIAL

¿QUE ES LA INFORMATICA?

La Informática constituye la solución para controlar la información relevante, generada por circunstancias cada día más complejas en el mundo actual.

En la medida en que exista habilidad para adquirirla, organizarla y aprovechar los conocimientos y las experiencias que se han desarrollado, la humanidad progresará.

La información de la naturaleza y de la sociedad debe ser comprendida para que la humanidad evolucione.

No existe una sola concepción acerca de qué es la Informática; aquí se presentan las más empleadas por los especialistas.

Etimológicamente, la palabra INFORMATICA, deriva del francés "informatique". Este neologismo proviene de la conjunción de "information" (información), y "automatique" (automática). Su creación fue estimulada por la intención de dar una alternativa menos tecnocrática y menos mecanicista al concepto de "proceso de datos".

En 1966, la Academia Francesa reconoció este nuevo concepto, y lo definió del modo siguiente:

"Ciencia del tratamiento sistemático y eficaz, realizado especialmente mediante máquinas automáticas, de la información contemplada como vehículo del saber humano y de la comunicación en los ámbitos técnico, económico y social".

Hacia principios de los '70, ya eran claras las limitaciones de esta definición; sobre todo por el énfasis en el uso de las máquinas. El principal esfuerzo para definir el concepto de Informática, lo realizó en esa época el IBI (Oficina Intergubernamental de Informática, en aquel tiempo, órgano asociado a la UNESCO). Este organismo a través de los comités de expertos convocados para ello, generó en 1976 la definición:

"Aplicación racional sistemática de la información para el desarrollo económico, social y político".

El IBI también generó en esa época una descripción del concepto de informática, que aunque no constituye una definición formal, resulta muy descriptiva:

"Ciencia de la política de la información".

En 1977, con la intención de actualizar y afinar el concepto, la Academia Mexicana de Informática propuso la definición siguiente:

"Ciencia de los sistemas inteligentes de información".

Consideraciones Sobre las Definiciones.

Resulta claro que el concepto de informática, en la actualidad, debe separarse del uso de máquinas. Por este motivo es fundamental destacar que el concepto de informática es totalmente distinto del de "computación".

La computación fue un elemento fundamental para el origen de la informática, y sigue siendo uno de los puntos principales de apoyo para la realización de sistemas informáticos. Sin embargo resulta un error confundir ambos conceptos o tomarlos como sinónimos.

Las definiciones mencionadas tienen como común denominador una asociación del concepto de informática con el manejo de información. Esto nos permite contar con un firme punto de partida para entender el concepto de informática. Esto es, el concepto de información resulta central para desarrollar un concepto de informática.

N. Wiener destaca el hecho de que la información es una entidad con la misma categoría que la energía y la materia. Aún considerando la validez de esta consideración, cabe destacar que la información tiene características propias que la hacen muy distinta de la energía o de la materia. En particular, pueden mencionarse: no se gasta ni se acaba con el uso y puede duplicarse prácticamente sin costo.

Otra propiedad importante de la información es que no existe por sí misma, sino que debe reflejarse en un soporte: tarjeta perforada, hoja de papel, pulso eléctrico, neurona cerebral, etc. Esta distinción entre soporte y contenido resulta cada vez más relevante, sobre todo desde un punto de vista legal, en lo referente a derechos de autor, propiedad de la información, etc.

El concepto de información generalmente se considera erróneamente como sinónimo de dato. Por este motivo, resulta necesario distinguir entre ambos aspectos. El concepto de datos se refiere al significado de los símbolos. El concepto de información está más asociado al conocimiento y a la comprensión de los fenómenos que a la simbología con la que se reflejan las observaciones.

Algunos autores enfatizan en el concepto de información, el aspecto de su capacidad para reducir la incertidumbre del receptor. Este fue un enfoque muy productivo; C. Shannon fue uno de los exponentes más relevantes de este enfoque.

Otra forma de ver la información, es considerándola como una forma de comunicación. A este respecto cabe aclarar que la información y la comunicación pueden verse como dos caras de la misma moneda. Esto es, estamos estudiando el mismo fenómeno,

pero desde dos puntos de vista distintos. En el caso de la comunicación, el énfasis es en los mecanismos para transferir los mensajes; en el caso de la información, se trata más bien de estudiar el contenido de las transmisiones y los requerimientos de los destinatarios.

W. Weaver propone un esquema que permite integrar una perspectiva congruente de la información, en base a diversos niveles o perspectivas de estudio.

El primer nivel, el más simple, denominado nivel técnico, considera los aspectos de eficiencia y capacidad de los canales de transmisión. Considera la información desde un punto de vista de mensajes o signos que reducen la incertidumbre del receptor. Los trabajos de Shannon consideran esencialmente este nivel.

El segundo nivel, o nivel semántico, presupone que el anterior ha sido resuelto, y considera la información desde el punto de vista de su significado. Los trabajos más conocidos en este aspecto se deben a Chomski, aunque es claro que falta mucho para que este nivel tenga el desarrollo del anterior.

El tercer nivel conocido como nivel pragmático, da por supuesta la solución de los anteriores, y se orienta al efecto que la información tiene sobre el receptor en un contexto dado. Este es el nivel que más nos interesa desde el punto de vista de la informática.

además de los tres niveles mencionados por Weaver, cabe introducir un cuarto nivel de carácter normativo, en el cual se estudie la información desde un punto de vista ético, o sea, consideran cuándo, dónde y a quién afecta la obtención o difusión de la información.

la definición de un concepto como "Información", requiere considerar el contexto en el que nos estamos ubicando. En este sentido la informática se relaciona estrechamente con la toma de decisiones, y por lo tanto, el concepto de información que nos interesa debe verse bajo esta perspectiva.

Lo anterior nos permite introducir una de las definiciones de información que consideramos más adecuada para nuestra discusión:

"La información es una comunicación que modifica los parámetros de una situación de toma de decisiones."

Esta concepción es congruente con el tercer nivel mencionado arriba, y pone de manifiesto la trascendencia que tiene el receptor de la información desde un punto de vista informático.

El concepto de información desarrollado en los párrafos precedentes, enmarca el campo de acción de la informática. Sin embargo la percepción social de la información ha variado enormemente durante el transcurso de la historia.

En los orígenes de la humanidad, los mecanismos para manejar la información estaban basados exclusivamente en información oral. Este tipo de información no cuenta con un soporte permanente y por lo tanto es de corta duración, ya que la información desaparece al terminar la comunicación. Aún cuando el cerebro del receptor podía almacenar durante cierto tiempo los conocimientos captados al morir el sujeto se perdía toda la información que acumuló.

La información oral es una forma muy limitada para que una sociedad acumule conocimientos y experiencias. Su carácter informal además, resulta muy inadecuado en muchas circunstancias.

al inventarse la escritura, se eliminaron muchas de las desventajas de la información oral. Esto no significa que deba eliminarse sólo establece que la información escrita resulta más conveniente para ciertos aspectos de la comunicación.

La información escrita permitió el desarrollo de sistemas numéricos sin los que hubiera sido imposible avanzar en muchas áreas. Al mismo tiempo, gracias a su permanencia, permitió ampliar el rango de comunicación independientemente del espacio y del tiempo. Gracias a ello, fue posible acumular mayores conocimientos a velocidades cada vez más rápidas.

Con la invención de la imprenta, se contó con la posibilidad de difundir los conocimientos y las ideas a un amplio número de personas. Con ello se dio un impulso fundamental al desarrollo de nuevos conocimientos, llegándose a hablar en las últimas décadas de una "explosión de la información".

La introducción de la computadora establece nueva modalidad para el manejo de la información: en forma electrónica. En esencia esto introduce dos variaciones respecto a los métodos anteriores.

Por una parte, la información electrónica es fácilmente modificable y adaptable a las características y necesidades de cada receptor. Esta flexibilidad ofrece posibilidades que nunca antes había existido.

Por otro lado, la información electrónica permite manejar los datos de que está constituida en forma extraordinariamente rápida y en grandes volúmenes. Esto permite generar, localizar, analizar duplicar y distribuir información de modo que no se hubieran podido siquiera sospechar hace unas décadas.

Una modalidad que actualmente se está desarrollando es la de la información distribuida. En este caso, gracias a la microcomputadoras y a nuevas redes de comunicación, dicha información se encuentra electrónicamente desde muchas fuentes, y está al alcance de cada usuario para que éste la maneje de acuerdo con sus necesidades.

la interacción directa del usuario o receptor con los mecanismos y fuentes de información, conjuntamente con un acceso cada vez más fácil, parecen estar entre los avances con mayor impacto potencial en la sociedad en mucho tiempo.

podemos decir que la actividad informática ha existido desde que el hombre empezó a planear el manejo de la información que requería. Sin embargo, la conformación de esta actividad con carácter formal es muy reciente.

El esquema de cosas que dio origen a la necesidad de considerar la informática como una ciencia autónoma, se empieza a concretar con la invención de la computadora. Esto da origen a la computación, como ciencia que estudia las características, el diseño y el aprovechamiento de esta nueva herramienta.

Al introducirse la computadora en los ámbitos administrativos se origina una nueva área de actividad denominada "proceso de datos". Con esta orientación, se desarrollan aplicaciones administrativas de nómina, contabilidad, facturación, etc., lo cual resultó de gran utilidad para el mundo empresarial y para la administración gubernamental.

Desafortunadamente, el proceso de datos fue una adaptación de muchas técnicas y métodos que venían usando en la automatización de labores rutinarias en las organizaciones. Por tanto, heredó una serie de deficiencias conceptuales.

Por una parte, las aplicaciones resultaban generalmente independientes e incompatibles entre sí, con lo cual era muy difícil aprovechar la información de manera integrada. Además de inconveniente esto resultaba ineficiente por la gran duplicidad que resultaba en la información almacenada.

Por otra parte, las aplicaciones se veían sólo como el manejo mecánico de una serie de reportes rígidamente estructurados. Los cambios y modificaciones eran pocas veces simples, y la obtención de información colateral normalmente muy difícil.

Dadas las limitaciones conceptuales del proceso de datos, muy pronto se inició una corriente que incorporaba aspectos de la teoría de sistemas, metodologías nuevas de trabajo, conceptos de bases de datos, y otras herramientas conceptuales. Esta nueva forma de conceptualizar el manejo de la información se le conoce como "sistemas de información".

La concepción que caracteriza el diseño de los sistemas de información tiene también una serie de deficiencias. Entre otras, el reducido énfasis a los aspectos humanos del uso de los sistemas, la poca flexibilidad que presentan los sistemas ante cambios ambientales, la falta de mecanismos "inteligentes" que permitan al propio sistema detectar situaciones anormales, etc.

La perspectiva informática pretende incidir en estos problemas incorporando herramientas como la inteligencia artificial, la

investigación de operaciones, la ergonometría, las ciencias sociales y del comportamiento, la cibernética y otras.

Evidentemente aún no hemos logrado integrar en forma satisfactoria un enfoque informático completo o siquiera bien es estructurado. Sin embargo, los avances son continuos y segura mente podremos contar en el futuro con metodologías y enfoques que nos permitan diseñar sistemas informáticos flexibles eficientes, características e idiosincrasias de los usuarios.

ING. ENZO MOLINO

SEMANA DE INFORMATICA

UNIVERSIDAD ANAHUAC

20 - MAYO - 1983.

BY
ROBERT A. STILLMAN JR.

34

TESTING FOR IBM PC COMPATIBILITY

Determining how compatible a computer is compared to the IBM PC has now been simplified with two new programs

The IBM PCs and IBM XT^s have set an industry standard that has generated numerous IBM PC clones. The question always asked about a clone is, How compatible is it with the IBM PC? The answer, in turn, determines what hardware and software will be available for the clone. We at Award Software have developed a series of tests aimed at confirming a computer's compatibility with the IBM PC.

Figure 1 shows an architectural model of the IBM PC and IBM XT. It consists of four levels: hardware, low-level support routines (ROM BIOS), PC-DOS I/O (Input/output) device drivers (IBM-BIO.COM), and the PC-DOS (IBM-DOS.COM) operating system. If the hardware were completely compatible, then IBM ROMs (read-only memory) and DOS (disk operating system) could be used on the computer. However,

copyright laws prohibit this approach, and a ROM BIOS (basic input/output system) must be implemented by the vendor (and must be sufficiently distinct from IBM's as some vendors have found out) to provide interfaces identical to the IBM ROM BIOS. Last, the MS-DOS IO.SYS module must be implemented (and again must be distinct enough to avoid copyright infringement) to provide a PC-DOS interface.

The three areas of compatibility testing addressed in this article are hardware, ROM BIOS, and PC-DOS interfaces. The hardware testing covers the major LSI (large-scale integration) chips (Intel 8088, 8259, 8255, NEC 765, etc.), I/O ports, and how the chips are inter-

(continued)

Robert A. Stillman Jr. is an executive vice-president of Award Software Inc. (236 N. Santa Cruz Ave., Los Gatos, CA 95030).

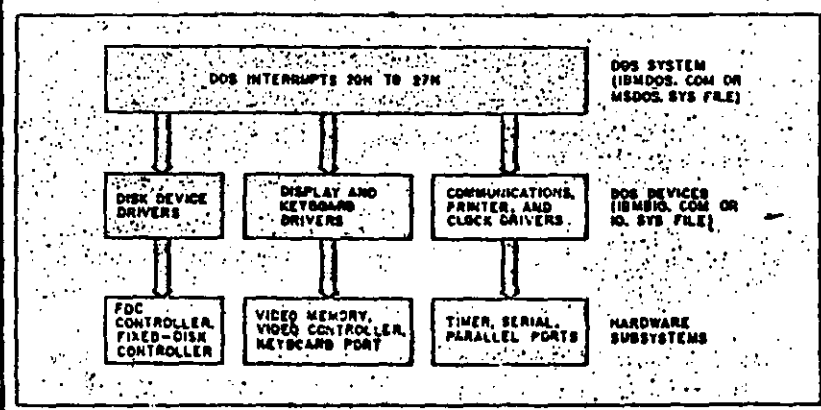


Figure 1: An architectural model of the IBM PC and XT.

connected. The internal bus structure and timings are not addressed. The ROM BIOS is the set of primitive I/O software routines used to organize the hardware into a somewhat manageable form. The IBM PC and IBM XT ROM BIOS is a single-task/single-user I/O scheme. Even though interrupts are used, only one process is running at a time. (One exception does occur with the timer interrupt.) The PC-DOS is an IBM version of Microsoft's MS-DOS. This is the major operating system used on the IBM PC and XT.

Microcomputers found to have an exceptional degree of hardware and software compatibility with the IBM PC are Columbia, Compaq, Corona, Eagle, Mitsubishi, Stearns, and Seequa.

GENERAL TEST ASPECTS

PC-TEST includes 50 programs written in C to investigate the hardware and ROM BIOS interfaces. Each individual test is designed to look at a particular software interface and/or hardware subsystem. Listing 1 is an example listing, which checks the BIOS.

DOS-TEST has 40 C programs used to verify the PC-DOS implementation. It exercises all MS-DOS user-documented interrupts and function calls. It verifies IBM PC keyboard and video interfaces. It also verifies that all the IBM PC supported disk organizations are correct.

The motivation for developing PC-TEST and DOS-TEST was to validate implementations of IBM PC ROM BIOS and DOS BIOS developed by Award Software for IBM PC clones. Also one of our utilities, Crossdata, which transfers CP/M-80, CP/M-86, and Concurrent CP/M-86 files to PC-DOS files and back again, makes extensive use of the ROM BIOS interfaces. So to aid our porting of Crossdata and validating of IBM PC compatibility, PC-TEST and DOS-TEST were written. PC-TEST and DOS-TEST are currently available to interested OEMs (original equipment manufacturers).

PC-TEST and DOS-TEST were written to be run against an IBM PC or IBM XT. Some tests check whether the system conforms or not and draw a conclusion. Some tests require visual confirmation by the tester. All tests identify themselves and explain what the test is about and areas of testing. All tests report register clobbering when detected and

Listing 1: Example test case for IBM PC compatibility included in PC-TEST.

```

.....
--TEST ID-- BIOS.DISPLAYIO.16 - BL_10_16
--PURPOSE-- Test characters and attributes for black/white display
--VERSION-- A
--TEST PROCEDURE-- Program writes attributes for all characters. Tester verifies visually for
each character. Shown is each character for all b/w attributes and modes and the ordinal value
of the character for cross reference. Set 40 or 80 columns via mode command for test of 25
by 40 and 25 by 80 screen sizes.
.....

#include "srvrv.c"

unsigned char curpage = 0;
int poscurs(y,x)
unsigned char x,y;
{
    srvah = 2; srvdh = y; srvid = x; stvbh = curpage;
    sint(0x10,0x7ff);
}

int clrscr()
{
    curpage = 0;
    srvah = 6; srval = 1; srvbh = 7; srvc = 0; srvid = 0;
    srvdh = 24; srvid = 79; sysint(0x10, &srvc, &srvid);
    poscurs(24,0);
}

main ()
{
    unsigned char cont,c;
    unsigned int i,j,k;
    printf("TEST BIOSLOWINTIO.16 - Test of char/attribute video I/O
Interrupt n");
    timestampl();
    printf("\nHit key to continue ");
    if ((bdos(0x8) & 0xf) == 'c') cont = 1;
    else cont = 0;
    getkey();
    srvah = 6; srval = 0; srvbh = 7; srvc = 0; srvid = 0;
    srvdh = 24; srvid = 79; sysint(0x10, &srvc, &srvid);
    poscurs(0,0);

    printf("Hit key to continue ");
    poscurs(2,0);
    printf("The character is shown with \n");
    printf("intensity off/ blink off, then \n");
    printf("intensity on/ blink off, then \n");
    printf("intensity off/ blink on, then \n");
    printf("intensity on/ blink on. ");
    poscurs(9,0);
    printf("BLACK CHAR ON BLACK \n");
    printf("UNDERLINE \n");
    printf("WHITE CHAR ON BLACK \n");
    printf("BLACK CHAR ON WHITE \n");
    printf("WHITE CHAR ON WHITE \n");

    for (i=0; i<256; i++) {
        for (j=0; j<5; j++) {
            switch (i) {
                case 0: c = 0; break;
                case 1: c = 1; break;
                case 2: c = 7; break;
                case 3: c = 0x70; break;
                case 4: c = 0x77; break;
            }
        }
    }
}

```

TESTING COMPATIBILITY

```

for (k=0; k<4; k++) {
  switch (k) {
    case 0: srv.bl = c; break;
    case 1: srv.bl = c | 8; break;
    case 2: srv.bl = c | 0x80; break;
    case 3: srv.bl = c | 0x88; break;
  };
  poscur(9 + j, 20 + k + 2);
  srv.bh = 0; srv.ch = 1; srv.al = 1; srv.ah = 9;
  sint(0x10, 0x7f);
};
srv.ah = 11; srv.bh = 0; srv.bl = 0; sysint(0x10, &srv.frrv);
poscur(15, 0); printf("VAL %xH\n");
poscur(16, 0); printf("VAL %d\n");
if (cont) {
  if ((bdos(0xb) & 0xf)) break;
}
else if ((bdos(0x8) & 0xf) != ' ') break;
};
clrscr();
}

```

report error values when results differ from those generated by an IBM PC. These tests are designed to be run by a systems programmer or technician who understands the internal structure of the IBM PC. The goal of the tests was to aid in determining where any incompatibility existed, not just that an incompatibility existed.

HARDWARE COMPATIBILITY TESTING

IBM PC and IBM XT hardware compatibility testing consists of verifying that the LSI chips and I/O addresses given in table 1 are the same. Also hardware compatibility requires that the monochrome and graphics memories exist at the proper addresses and that

the character and attribute (or pixel) memories have identical organization. PC-TEST checks hardware compatibility by looking for correct operations from the integrated chips that identify that the right chip is present at the proper I/O address. Display memory use is tested by putting patterns directly into the video memory and having the tester verify the pattern.

To run all IBM PC versions of interrupt-driven operating systems, such as QNX, Concurrent CP/M-86, etc., the above tests must pass. Otherwise you must obtain special versions of those operating systems. (Many times they do not exist.)

You can check I/O bus and timing compatibility by plugging in commercially available boards and verifying

their operation. Another method would be to take timing pictures using logic analyzers and verify them against an IBM PC or IBM XT.

ROM BIOS COMPATIBILITY TESTING

The ROM BIOS has two types of interface. The most commonly used and recommended one is via software interrupts. The other type is direct addressing of RAM (random-access read/write memory) used by the ROM BIOS (not recommended).

Using just software interrupts permits a wider range of computers upon which a particular software package will run. More computers are compatible in this area than the hardware area. Also, a number of computers use a different I/O and chip set to implement the system.

Table 2 lists the software interrupts provided by the IBM PC ROM BIOS. These software interrupts are passed parameters in registers. Some of the software interrupts use other interrupt vectors to obtain other parameters.

The major ROM BIOS functions used by software packages are the Video, Disk, Printer, Communications, Keyboard, Timer, and Equipment Interrupts. These interfaces are described in the IBM Personal Computer and XT *Technical Reference Manuals* ROM BIOS listings, Appendix A.

The Video interrupt (10 hexadecimal) provides 15 functions that allow for mode of screen display, cursor addressing, lightpen feedback, scrolling, teletype output, character and attribute read/write, graphics read/write, and

(continued)

GRAPHS WITHOUT GRAPHICS?

Disk Drives 11.95

Printers 29.45

Software 19.95

Modems 27.05

SALES IN 1983

No need for color monitor or graphics board.
Make graphs on dot matrix printers.

Easy to Use on PCs and Compatibles.
Excellent Manual.
PC-1005, MS-1005, or CP/M.

DataPlotter™

Line Graphs & Scatterplots . . . \$69
Bar Graphs & Pie Charts . . . \$69
Both for \$99

(MS-1005 includes manual)
Add \$3 shipping.
\$4 outside US and Canada.

Specify type of Printer.
(201) 276-7552 Visa, MC

Lark Software™
7 Cedars Road
Caldwell, NJ 07006

TESTING COMPATIBILITY

37

palette and color selection. PC-TEST verifies all these uses in a series of 17 tests. Attribute values have been a particular problem on some implementations. However, all of the manufacturers of microcomputers we have examined have cleared up these problems.

The Disk interrupt (13 hexadecimal) provides five functions for floppy disks and 14 functions for fixed disks. Fixed-disk functions 9 through 14 are used by IBM for manufacturing and diagnostic testing and are not generally incorporated in software packages. Both the floppy and fixed disks use parameter tables at interrupt vector 1E hexadecimal and 41 hexadecimal, respectively. These are pointers to tables in ROM or RAM that describe motor timings, sectors per track, tracks per disk, and so on.

PC-TEST extensively tests these parameters because many software packages, like our Crossdata program, alter these parameters to read foreign disks. Also, these parameters are altered to implement many of the copy-protection schemes used to protect software from duplication on the IBM PC. This is the major area of incompatibility found on computers because a number of vendors have implemented higher-storage floppy-disk systems. These systems, in turn, exhibit internal timings and responses different from those of the IBM PC, and the copy-protection subsequently fails. Another major area of incompatibility is in error reporting. Some copy-protection schemes rely on certain error codes being reported by the IBM PC, and some systems do not

Table 1: Major hardware components of the IBM PC.

Identification	I/O addresses	Use
Intel 8088, 8086	—	Processor
NEC 765	3F4H, 3F5H	Floppy-disk controller
74LS Latch	3F7H	Floppy-disk secondary control
Intel 8255	60H, 61H, 62H, 63H	I/O ports for speaker, keyboard, equipment configuration, RAM parity enable
Intel 8253	40H - 43H	Real-time clock, speaker timer, DMA (direct memory access) refresh
Intel 8237	0 - 0FH	DMA controller
74LS Latches	80H - 83H	DMA page registers
Intel 8259	20H, 21H	Interrupt controller
74LS Latch	0A0 - 0AFH	NMI control
Fixed Disk	320H - 32FH	Fixed disk controller
Intel 8255	378H - 37AH 38CH - 38EH 278H - 27AH	Printer controllers
National 8250	278H - 27EH 3F8H - 3FEH	Serial communications chip
Motorola 6845 Latches	3D0H - 3DFH	Color/graphics controller
Motorola 6845 Latches	3B0H - 3BFH	Monochrome display controller

initially report the error codes the same way as the PC.

The Printer interrupt (17 hexadecimal) provides three functions for as many as three parallel ports. PC-TEST verifies these functions for the specified logical parallel port and checks for compatibility in error-code reporting.

The communications interrupt (14 hexadecimal) provides four functions for as many as two serial communications ports. PC-TEST verifies these functions for the specified logical serial ports and checks for compatibility in error code reporting.

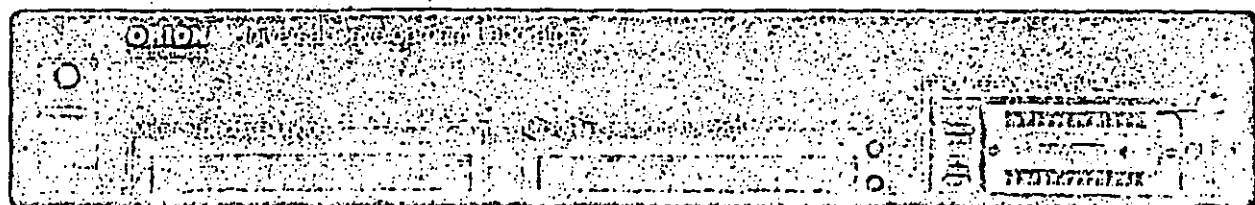
The major area of divergence for

microcomputers is in the keyboard-handling interrupt (16 hexadecimal). PC-TEST does an exhaustive keystroke test, with input from the tester, to ensure all shift states and keycode combinations occur. The test looks at the actual internal shift states as well as keycodes returned. Most of the computers listed earlier in the article as compatible pass these tests. Some initially failed to report proper internal shift states, but the failure has since been corrected. All reported the proper keycodes that are used by the vast majority of the software packages.

The Equipment interrupts (11 hexa-

THE \$2395 DEVELOPMENT SYSTEM

Turns any personal computer into a complete micro-computer DEVELOPMENT SYSTEM. Our integrated control/display program runs under PCIMS-DOS, CP/M, or TRS-DOS, and controls the UDL via an RS-232 port.



Up to 128K bytes of ROM EMULATION (8K standard) allows you to make program patches instantly. Since the target ROM socket connects data and address lines to both the analyzer and the emulator, no expensive adapters or personality modules are needed.

The powerful BUS STATE ANALYZER features four-step sequential triggering, selective trace, and pass and delay counters. \$99 symbolic trace disassemblers are available for Z-80, 8048, 6500, 6800, 8031, 8085, 3870, Z-8, 1802, 8088, & 8086.

The PROM PROGRAMMER also doubles as a STIMULUS GENERATOR. For a brochure and list of cross assemblers call or write:

ORION Instruments
172 Old Ave., Woodside, CA 94095
(415) 851-1172

TESTING COMPATIBILITY

38

Table 2: ROM BIOS interrupt use in the IBM PC.

Interrupt	Use
0	Divide by zero from 8086/8088
1	Single step (used by DEBUG)
2	Nonmaskable (parity error)
3	Breakpoint (used by debugger)
4	Overflow
5	Print screen
6-7	Reserved
8-0FH	Hardware Interrupts from 8259
10H	Video interrupt—15 functions available
11H	Equipment map
12H	Memory size
13H	Disk interrupt—6 floppy- and 14 fixed-disk functions
14H	Communications interrupt—4 functions
15H	Cassette
16H	Keyboard interrupt—3 functions
17H	Printer interrupt—3 functions
18H	ROM basic entry
19H	Bootstrap loader
1AH	Time-of-day interrupt—2 functions
1BH	Keyboard break interrupt
1CH	Timer tick interrupt
1DH	Video initialization parameter list pointer
1EH	Disk parameter list pointer
1FH	Graphics character pattern pointer
20H-3FH	DOS interrupts
40H	Disk interrupt reroute for fixed-disk systems
41H	Fixed-disk parameter list pointer
42H-FFH	Reserved or used by DOS, BASIC or user-supplied interrupts

decimal and 12 hexadecimal) tell the type of equipment and amount of memory available. PC-TEST reports the equipment list and the memory used. No problems were seen on the computers listed earlier.

The Timer interrupts (1A hexadecimal and 1C hexadecimal) yield a free running clock and a timer tick. PC-TEST programs verify the accuracy of the clock and the implementation of the timer.

Other interrupt vectors used are the 8259 interrupt chip vectors, trap and single-step interrupts, divide-by-zero interrupts, and print-screen interrupts. PC-TEST checks all of the above except the trap and single-step interrupts. They are most easily verified by using DEBUG. Again the microcomputers listed earlier provided no problems. However, the print-screen interrupt interferes with the Intel 188 processor chips, and systems that use the more advanced Intel processors may be incompatible.

PC-DOS COMPATIBILITY TESTING

Most software packages that do not

directly address the screen or use the ANSI (American National Standards Institute) escape sequences to address the screen do use the standard MS-DOS function calls. If the package does not use function keys, then it probably runs on any MS-DOS machine. However, most packages for the PC require screen addressing and function key codes. They use the Video software interrupt as the only lower-level machine interface (with perhaps an Equipment check) and use only standard MS-DOS calls. If the computer supports the Video interface, the screen character set, and the function key codes, then a large set of software packages will run on the machine.

One potential area of incompatibility is the inability to read and write all supported IBM PC floppy-disk layouts. These layouts are 48-tpi (tracks per inch), single- and double-sided, eight- and nine-sectored disks. This inability may be caused by having a 96-tpi drive and not being able to write on a 48-tpi disk or by not reading eight-sectored

(continued)

Personal Computer SMARTWARE™

Spreadlink™

IBM PC/XT, PCjr

\$195

Mainframe Data to Your PC...

Instead of entering data by hand from external sources into your spreadsheets or databases, Spreadlink reformats your data and allows you to load your information directly into 1-2-3, VisiCalc, MultiPlan, dBase II, and many other packages for direct use. Your files can be from mainframes, commercial databases, mini-computers, other personal computers, or any application packages that create ASCII standard text files (such as word processors).

You get to preview your data, edit rows and columns with Spreadlink, and then Spreadlink automatically converts it into the right format for your application. Since Spreadlink lets you save your edit format as a command file, you can automate your requests for the next time. Spreadlink allows any mix of labels and values without any special formatting or separators... it also adjusts to column variations caused by stacking several dissimilar reports in one file. We use Spreadlink to move data from our Mini to our controller's PC for monthly reports... and it only took our controller 15 minutes to learn and start using.

FEATURES

- Converts standard ASCII text files to native file for Lotus 1-2-3, VisiCalc, Multiplan, dBase II, and DIF.
- Screen preview of your data allows you to edit out unwanted rows or columns or data. Convert only the data you need, or just take all of it automatically.
- File size is only limited by the disk space available.
- No pre-editing or programming required. Reports do not need to be in columns and rows.
- Requires no installation or adaptation on the host computer.
- Requires 64K memory, one disk drive, any version of MS or PC DOS.

APPLICATIONS

Download the last year's financials from your IBM 3033. Using Spreadlink, load the data to your 1-2-3 model for analysis to determine next year's plan. Take expense figures out of a memo in a word processor text file and use it as input to your VisiCalc expense tracking model. All completely automated, no technical knowledge required.

© 1983 Smartware, Inc.

SMARTWARE, INC.

557 Howard Street San Francisco, California 94105

DIAL: 800-SMARTWARE Visa and MC accepted (In California: 974-1500)

Send me a catalog!

BY1

NAME _____
 COMPANY _____
 ADDRESS _____
 CITY _____ STATE _____ ZIP _____

000217

SAVE BIG ON COMPUTER PRODUCTS

MODEMS

Smartmodem 300	200
Smartmodem 1200	425
Smartmodem 1200B	425
Micromodem II E	240
American B 8000 (for IBM PC)	80
Cable to Hayes Modems	CALL
NOVATION	
SmartCat Plus (NEW)	CALL
1 Cat 3000 Fax	99
103 SmartCat 3000 Smart	129
103 242 Smart 300/1200 B	379
AppleCat II 300 baud for Apple	200
212 AppleCat 300/1200 for Apple	350
Access 1-8-3 1200 for IBM	379

SAVE 31%-43%
OFF MFR. SUGG. RETAIL PRICES ON
PRINTERS
EPSON • OKIDATA • DIACLO
SCM • DELTA • GEMINI • TTX
RADIX • COMPUTE-MATE
MAGNESMANN TALLY



CALL FOR PRICES
Wabash
Maxell
3M/Janit

MONITORS

V3000 12" green	122
V3000 12" amber	95
V310A 12" amber (for IBM)	165
Color 12" composite	220
Color 12" RGB	425
Color 12" RGB video	740
Monitor Cable	
CB 5890 Apple II to Monitor	7
CB 9291 IBM to RGB	19
CB 5692 for TI-99 4A or Commodore	15

QUADRAM MICROFAX

QRMP-B Per/Par	135
QRMP-C Per/Par	145
QRMP-BB Per/Par	145
QRMP-B Per/Par	145
QUADRAD (for IBM PC)	
QR 5210 (no mem. installed)	210
QR 5364 64K (mem. installed)	270
QR 4064 64K (mem. installed)	270
QR 8201 Quadstar-1	200
QR 8202 Quadstar-2 (upgrade kit)	200
QUADLINK	
QR 3000 for IBM	475
QR 3010 for Compaq	475
QR 3020 for Columbia	475

AST

Sis Pak Plus	230
Nege Plus II	230
1/0 Plus II	112

DISKETTES

10 - 5 1/4" Floppy Diskettes	
Wabash 55/50	\$5.00
Maxell NA	\$6.00
Dyan NA	\$3.00
Dyan 10 - 5 1/4" Rugged diskettes	37.00

CALL FOR QUANTITY PRICING ON 10 OR MORE BOXES OF DISKETTES

HEWLETT-PACKARD CALCULATORS

HP-11C	85
HP-11C	90
HP-11C	90
HP-41C	145
HP-41CX	145
HP-41CV	165
HP-97	360

PORTABLE COMPUTERS

HP-71B	309
HP-75D	709
series 70 software & peripherals	discounted too
HP-9225B ThinJet Printer (HP11)	375

SHARP CALCULATORS

SL 5100	43
SL 5000 F	70
SL 512 F	70

SHARP HAND HELD COMPUTERS

pc 1250A	80
pc 1260	CALL
pc 1261	CALL
pc 1500A	180

CABLES - INTERFACES

GRAPPLER PLUS for Computer Printers	105
16K BUFFERED GRAPPLER	165
Apple Dumping CK	25
Corde G	55
CP100 10 ft. Par. Cable for IBM	25
CP122 10 ft. 36-pin Parallel	25
CP122 10 ft. 25-pin RS-232	25
CP122 8 ft. 11-50/44 parallel cable	25
CR500 2 ft. par. 1150 mod. 44-1122	CALL
ALL OTHER CABLES	CALL

CALL TOLL FREE 800-621-1269 EXCEPT Illinois, Alaska, Hawaii

BETA-TEK INC.

BASIC Discovers Shorthand.

Programming in BASIC just got smarter, easier, faster... and five great reviews. Before you write another program in BASIC, you might be wise to start with a little reading.

Five major computer publications have something good to tell you. About the BASIC Development System (BDS). And about making your programming in BASIC more efficient than you ever dreamed.

BDS is a powerful, integrated set of software tools that quickly gets you beyond the BASIC basics. It gives you everything from Cross-Referencing to Scrolling Keys that let you scroll by page or line. It gives you Compress and Uncompress commands, Variable Dump, Single Step Trace and more.

BDS also gives you something else. A 30-day money back guarantee.

Finally, BDS gives you great references. From reviewers who've lived with it and loved it. For example: "I wonder how I ever got along without it." John M. Woram, PC Magazine, September 1983.

We'll send you the reviews just for writing us. Or, for only \$79, we'll send you BDS itself. And a 30-day money back guarantee.

BETA TOOL SYSTEMS
BTS

872 E Hampden Ave. - Suite 179
Denver, Colorado 80231 • (303) 793-0145

BDS is available for IBM PC and COMPAQ computers. VISA/MC accepted. Please add \$3 for shipping.

TESTING COMPATIBILITY

339

formats at all.

Another area of incompatibility arises from the manufacturer not partitioning the fixed disk in the same manner as the IBM XT does. The IBM XT allows as many as four different bootable disk partitions. Each operating system may use one partition only. Thus a fixed disk may contain a DOS, CP/M-86, and/or Concurrent CP/M-86 partition.

The DOS-TEST programs check four areas of MS-DOS: the display, communications, printer, and keyboard characteristics; the disk organization; DOS interrupts; and the remainder of DOS functions that are invariant Microsoft functions.

The display testing consists of displaying the characters as shown in the IBM Technical Reference Manual, Appendix C. These tests use DOS interrupt 21 hexadecimal functions 2 and 6. The keyboard tests in DOS-TEST verify that the extended keycodes are identical. The physical keyboard layout may be different, but the ASCII (American National Standard Code for Information Interchange) and extended codes must be identical to pass the tests. The keyboard tests use DOS interrupt 21 hexadecimal functions 1, 6, 7, 8, A, B, and C. The other character device operations are tested via DOS interrupt 21 hexadecimal functions 3, 4, 5, and 9.

DOS-TEST includes programs for the testing of disk organization as noted above. Also, one program checks the fixed-disk partitioning. This test uses Disk ROM BIOS interrupts because the partition table exists outside of the system.

The remainder of the DOS-TEST programs exercise DOS interrupts 20 hexadecimal (program terminate), 22 hexadecimal (terminate address), 23 hexadecimal (control break exit address), 24 hexadecimal (critical error handling vector), 25 hexadecimal (absolute disk read), 26 hexadecimal (absolute disk write), 27 hexadecimal (terminate process and remain resident), and 21 hexadecimal (DOS function calls not included above).

Checking for IBM PC and IBM XT compatibility is tedious at best. PC-TEST and DOS-TEST were designed to be specific engineering test tools to pinpoint areas of compatibility and incompatibility.

Un nuevo significado a la computación personal...

Presentando el Sistema Personal/2 de IBM: una familia completa de productos con adelantos tecnológicos que ponen la computación personal en una nueva ruta hacia el futuro. El Sistema Personal/2 de IBM significa nuevas unidades de sistema, pantallas e impresoras... lo último en gráficos... un sistema operativo superior... impresionantes capacidades de conexión... y mucho más.

El Sistema Personal/2 cubre toda la gama de soporte que usted podría necesitar para implantar las estrategias de su empresa y mantenerse adelante de la competencia. Es mucho más que una herramienta de productividad personal—es una solución total para mejorar la eficiencia de toda la empresa.

Y eso es precisamente lo que estos tiempos necesitan. El procesamiento de palabras se ha convertido en la edición de publicaciones completas, sobre el escritorio mismo.

Donde simples hojas electrónicas bastaban para los cálculos de contabilidad, hoy se requieren modelos financieros muy complejos. Donde se corría una aplicación a la vez, ahora el procesamiento simultáneo de varias aplicaciones es lo común. Ya el procesamiento aislado está siendo reemplazado por redes de computadores, conectados a computadores mayores y a otras fuentes externas de información.

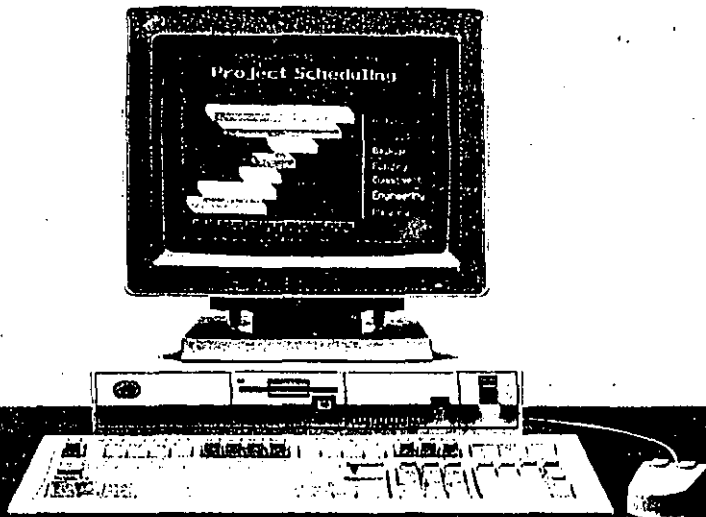
El Sistema Personal/2 tiene casi todo lo que usted necesita para poder enfrentar estas demandas... potencia... rendimiento... capacidad... conectividad... fiabilidad... facilidad de uso. Ponga al Sistema Personal/2 a trabajar dentro de una gran variedad de ambientes—ya sea como estación de trabajo conectada a un sistema mayor, como sistema departamental, o para las comunicaciones a lo largo de la empresa. Equipe a todos sus empleados con estaciones de trabajo que se adaptan a las tareas que ellos hacen.

Con las unidades del Sistema Personal/2 usted tiene un rango de selecciones para satisfacer cualquier necesidad de su empresa. Dos modelos para el piso y dos para el escritorio, equipados con los poderosos microprocesadores 80386, 80286 u 8086—y suficiente memoria y almacenamiento para respaldarlos. Añada a esto las diferentes opciones disponibles, y usted encuentra que casi no hay límite en el número de oportunidades que el Sistema Personal/2 ofrece a su empresa, para mejorar la eficiencia y capacidad de competir.

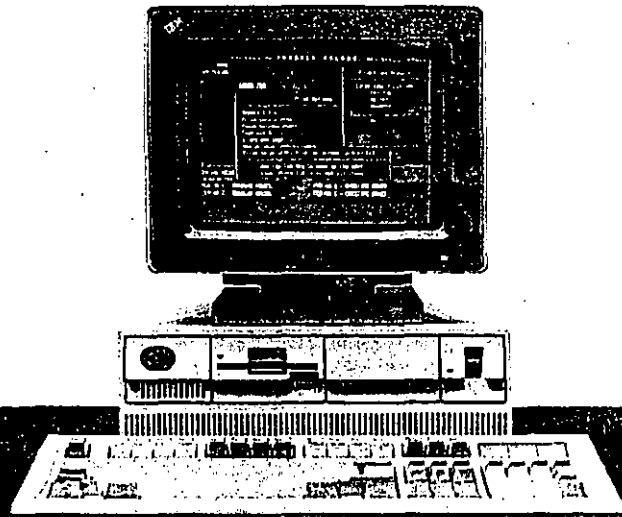


000219

Sistema Personal/2 IBM 8512
Pantalla de Color



Sistema Personal/2 IBM 8503
Pantalla Monocromática



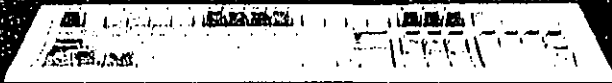
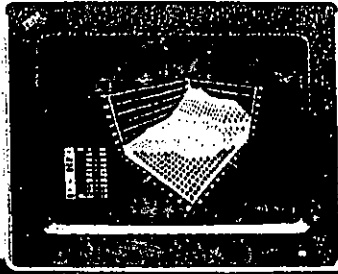
El Sistema Personal/2 Modelo 30 ofrece un rendimiento excelente a un precio muy cómodo—convirtiéndolo en la selección ideal para cualquier oficina. Las dos versiones de este modelo vienen con:

- un microprocesador 8086
- memoria de usuario de 640KB
- dos unidades para diskettes de 720KB, o una unidad para diskettes de 720KB y un disco fijo de 20MB

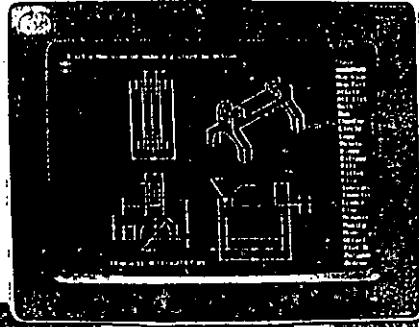
El Sistema Personal/2 Modelo 50 ofrece alto rendimiento en un estilo moderno. Es el sistema de escritorio ideal para ayudarle a alcanzar nuevos niveles de productividad. Las características de este modelo incluyen:

- un microprocesador 80286
- memoria de usuario de 1MB, con expansiones a 7MB
- un disco fijo de 20MB
- una unidad para diskettes de 1,44 MB

Sistema Personal/2 IBM 8513
Pantalla de Color

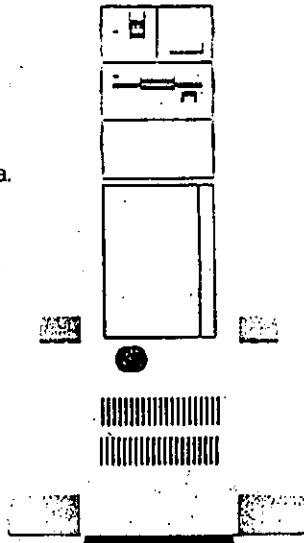


Sistema Personal/2 IBM 8514
Pantalla de Color



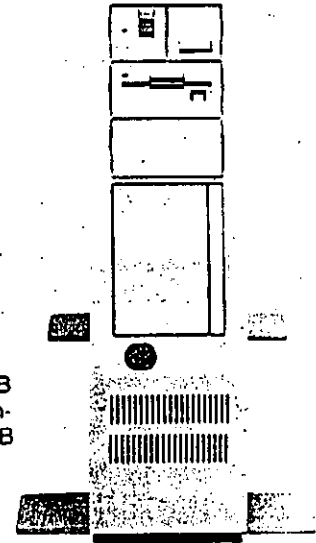
El Sistema Personal/2 Modelo 60 ofrece funciones avanzadas, y la capacidad para ayudarlo a correr su oficina con la eficiencia más alta. Las dos versiones de este modelo ofrecen:

- un microprocesador 80286
- memoria de usuario de 1MB, con expansiones a 15MB
- disco fijo de 44MB ó 70MB—y un segundo disco opcional de 44MB, 70MB o hasta 115MB fijo
- unidades para diskettes de 1,44 MB



El Sistema Personal/2 Modelo 80 es el miembro más poderoso de la familia Sistema Personal/2. Ofrece suficiente poder, velocidad y capacidad para procesar las aplicaciones más complejas en empresas, ingeniería y ciencias. Las tres versiones de este modelo ofrecen:

- microprocesador 80386
- memoria de usuario de 1MB ó 2MB—con expansiones a 16MB
- capacidad para almacenamiento en disco fijo de 44MB a 70MB a 115MB—se puede doblar con un segundo disco
- unidades para diskettes de 1,44MB



Sistema Personal/2: Una inversión inteligente para hoy y mañana

El Sistema Personal/2 es más que una solución ingeniosa para mejorar la eficiencia de su organización—es también una inversión muy inteligente. Eso se debe a que el Sistema Personal/2 ofrece avances tecnológicos de tanto alcance y profundidad, que hacen resaltar su valor como inversión.

Soluciones totales al precio de un microcomputador
Utilizando el modelo más poderoso del Sistema Personal/2 como servidor, usted puede mejorar la relación entre costo y eficiencia a través de toda su red. Utilícelo como puerto de acceso al sistema central, y

mejore las comunicaciones para todos sus usuarios. Esto se debe a que el Sistema Personal/2 se diseñó para que trabaje como parte de un sistema, y por lo tanto, el valor de cada modelo se extiende mucho más allá del modelo mismo.

Y su valor aumenta a medida que el nuevo Sistema Operativo/2 le permite ejecutar un creciente número de diversos trabajos. En una empresa típica, por ejemplo, un empleado utilizando cualquier modelo del Sistema Personal/2 podría:

- Trabajar simultáneamente con varias aplicaciones
- Trabajar simultáneamente con varias aplicaciones y sesiones con sistemas centrales
- Inclusive, trabajar simultáneamente con aplicaciones y sesiones con sistemas centrales que dan soporte a todos los usuarios de la red

Los beneficios que usted obtiene con el Sistema Personal/2 se extienden a una gran variedad de recursos. Como todos los modelos dentro de la familia son compatibles, la mayoría de las aplicaciones² se pueden transportar de un modelo a otro.¹ Y los equipos periféricos que usted compra para una estación de trabajo, se pueden también utilizar con cualquiera otra.

Una inversión muy sólida para el futuro
Equipos compatibles y operaciones consistentes, constituyen la base de un plan que IBM tiene para el futuro, llamado: Arquitectura para Sistemas de Aplicaciones (SAA). Es un plan para aumentar el nivel de uniformidad en la programación, y así poder integrar más los distintos productos

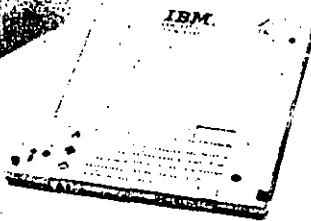
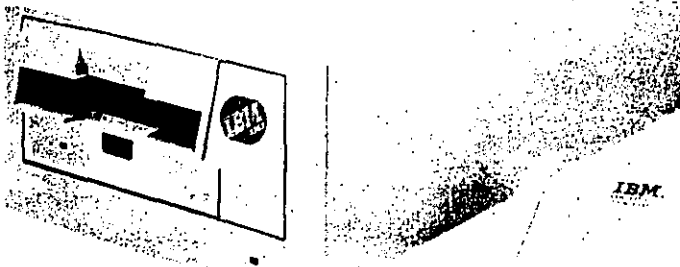
de computación, a cada nivel. Como primera etapa en este plan, el nuevo Sistema Operativo/2 pone al Sistema Personal/2, a usted y a su empresa, en la ruta hacia el futuro.

La Edición Ampliada del Sistema Operativo/2 ofrece capacidades internas para bases de datos relacionales—y facilita la creación, mantenimiento y acceso a bases de datos en sistemas mayores. Y como el Sistema Operativo/2 también ofrece interfaces uniformes al usuario, una vez que alguien conoce las operaciones de bases de datos en un sistema, sus conocimientos se pueden utilizar en cualquier otro. Así su inversión en educación lo lleva más lejos que nunca.

Trabajando en conjunto el Sistema Operativo/2 y el Sistema Personal/2 le ofrecen una base muy firme para su futuro crecimiento.

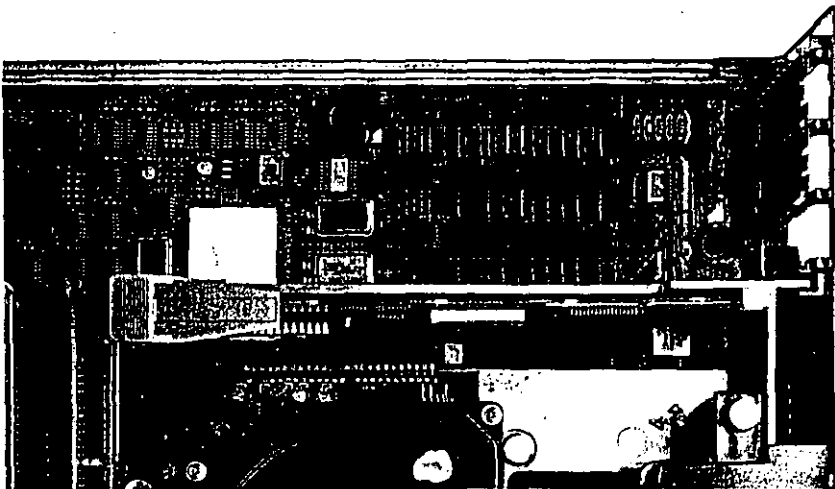
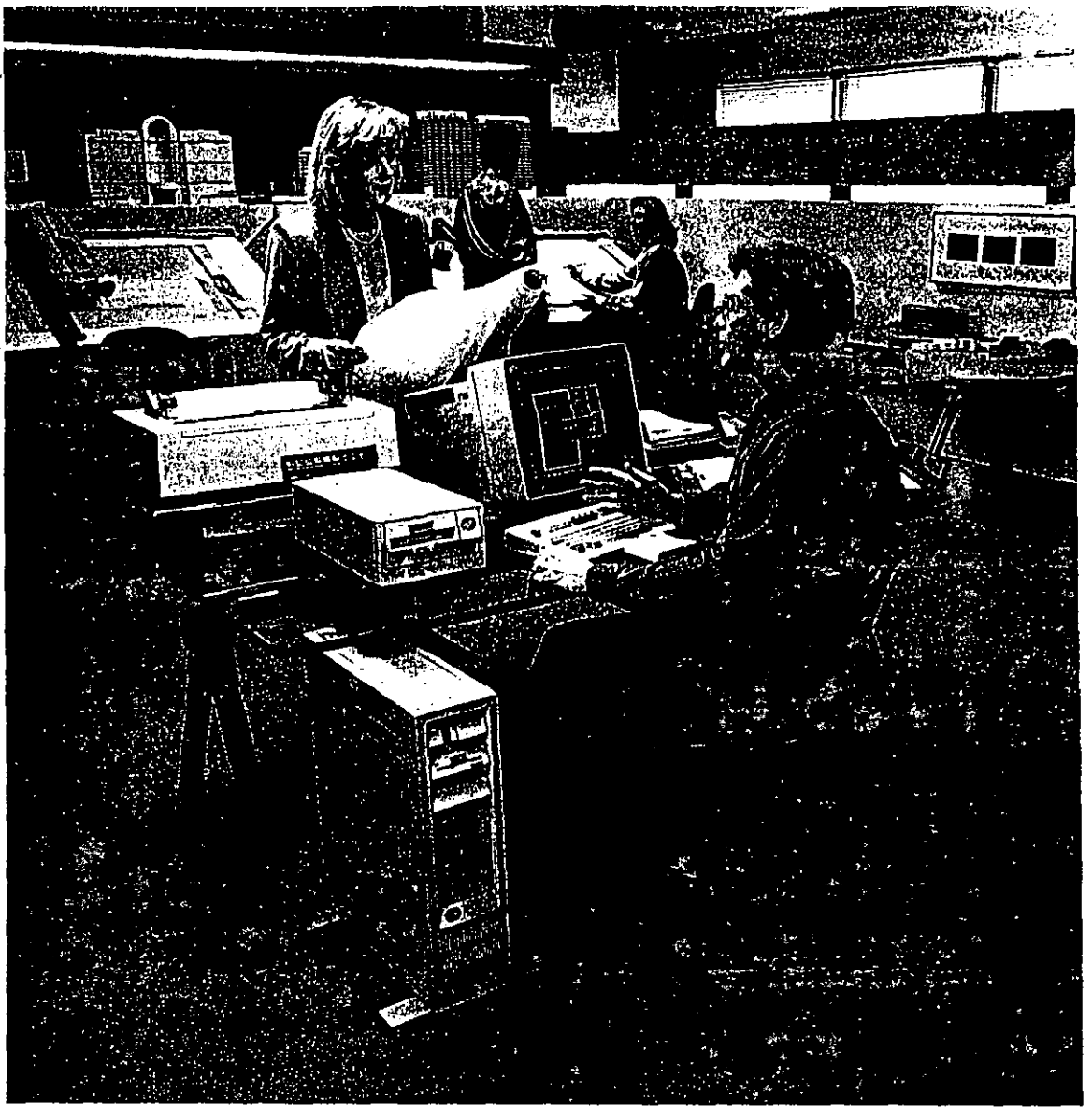
La nueva arquitectura de Micro Canal es clave a este crecimiento. Esta arquitectura prepara a la familia¹ para futuras expansiones—a nuevos niveles de capacidad y habilidad, que la mayoría de los microcomputadores actuales no han podido alcanzar hasta ahora. Más equipos periféricos se pueden conectar y operar simultáneamente. Usando los adaptadores de expansión opcionales, la memoria se puede duplicar directamente en la placa del sistema, hasta alcanzar 15MB ó 16MB.





Con un disco
óptico opcional,
un solo modelo
del Sistema
Personal/2 de IBM
puede respaldar
los archivos de
todos los
usuarios en
la red.

000223



La nueva
tecnología de
montaje de
circuitos direc-
tamente sobre
la superficie
requiere menos
alambres y
menos conexio-
nes frágiles.
Por esta razón
el Sistema
Personal/2 es
tan confiable.

6

Lo distinto del Sistema Personal/2

En el modelo más potente, la expansión de memoria se hace con "chips" de 1MB. Esto se debe a que la arquitectura acepta los mismos "chips" que se utilizan en la línea de sistemas mayores de IBM. Así las ranuras de expansión quedan libres para añadir más memoria, u otras funciones opcionales.

Existe un amplio rango de tamaños de almacenamiento permanente para que usted pueda seleccionar el que mejor trabaja con la memoria de su Sistema Personal/2. Por ejemplo, tanto los modelos de piso, como los de escritorio, vienen con discos fijos de 44MB, 70MB, ó 115MB—y con los discos opcionales, esta capacidad se puede doblar, como mínimo. Para aún mayor comodidad, en casi todos los Sistemas Personales/2 se puede añadir una segunda unidad de diskettes y dos nuevas opciones—un disco óptico de 200MB y una cinta de modalidad continua.

El Sistema Personal/2 lo acompañará durante el crecimiento de su empresa—inclusive cuando este es muy rápido. Es tecnología para edificar—no para reemplazar. Su inversión en esta generación de productos de computación personal estará protegida durante muchos años, así decida usted aumentar el número de sistemas, o añadir más capacidades a aquellos que ya tiene.

Protección a sus pasadas inversiones

Con el Sistema Personal/2, usted puede también proteger su inversión en sus Computadores Personales IBM—ya que la compatibilidad no se queda dentro de la familia. Las nuevas unidades del sistema se pueden conectar a las redes locales actuales, para que trabajen al lado de los Computadores Personales IBM, los Computadores Personales IBM XT, o los Computadores Personales IBM AT, que usted ya tiene. También son compatibles con el Computador Personal IBM Convertible que usted lleva a todas partes.

Y a pesar de que los modelos del Sistema Personal/2 utilizan un nuevo medio de almacenamiento—diskettes de 3,5 pulgadas, de alta capacidad—los programas y archivos que usted tiene almacenados en diskettes de 5,25 pulgadas no están en peligro. Existen varias opciones de conversión para facilitar la migración de datos y programas entre el PC de IBM y el Sistema Personal/2.

Gracias a sus características tan sobresalientes, el Sistema Personal/2 de IBM es un sistema distinto—y mucho mejor—que cualquier sistema del pasado—por fuera y por dentro. Estas son las razones por las cuales IBM llama al Sistema Personal/2 "la próxima generación en computación personal":

Arquitectura avanzada de Micro Canal

Hasta ahora, la mayoría de los sistemas de computación personal han tenido dificultades cumpliendo con las estrictas demandas de rendimiento, comunicaciones y capacidad de almacenamiento, que los usuarios están imponiendo.

El Sistema Personal/2 viene a ofrecer nuevos niveles, tan altos, que inclusive muchos de los sistemas de computación personal ya existentes no pueden todavía ofrecer, ni siquiera con las opciones y ampliaciones a su disposición.

Los modelos del Sistema Personal/2 se caracterizan por sus canales más rápidos y amplios, una nueva arquitectura, soporte de microcódigo ampliado, y canales para acceso a memoria directa. Debido a que los datos fluyen, a través de estas unidades, directamente y con más rapidez, la capacidad y velocidad de procesamiento son también mayores. Los beneficios que usted obtiene de estas características son: cálculos y comunicaciones a velocidades muy altas, y la capacidad de ejecutar múltiples tareas y operaciones de entrada/salida, sin sacrificar rendimiento, o correr riesgos de conflictos dentro del sistema.

Funciones integradas

El Sistema Personal/2 utiliza circuitos de muy alta integración (VLSI) y una nueva tecnología para colocar los circuitos directamente sobre la placa del sistema. De esta manera se pueden integrar más funciones en un espacio menor. Como resultado, los modelos del Sistema Personal/2 contienen muchas más funciones internas, a pesar de que tienen menos componentes separadas. Por ejemplo, las funciones para gráficos avanzados y colores, así como el soporte a pantallas, impresoras y teclados—inclusive el dispositivo "ratón"—vienen como funciones estándar.

Más aún... La construcción integrada significa también mayor confiabilidad, ya que con menos componentes, existen menos posibilidades de problemas. Lo que quiere decir, rendimiento óptimo, año tras año.

Color, gráficos y nuevas pantallas analógicas

Una nueva tecnología de pantallas, es responsable por los nuevos estándares de rendimiento que ofrece la familia Sistema Personal/2. Se pueden contemplar hasta 256 colores al mismo tiempo, de una paleta con más de 256.000 colores. Y en las pantallas monocromáticas, los colores se convierten automáticamente a hasta 64 tonos de gris.

Ya sea en color o en monocromo, las funciones internas para color y gráficos en el Sistema Personal/2, se combinan con el alto direccionamiento de las nuevas pantallas, para producir imágenes más nítidas, claras, y con gran cantidad de detalle. Y más natural también. Inclusive fotografías digitalizadas de productos, propiedades o gente, aparecen completamente natural.

Mejor y mayor almacenamiento

La gran noticia sobre el almacenamiento de datos en el Sistema Personal/2 es acerca de los discos fijos internos, con capacidad que varía entre 20MB y 115MB. Estos discos proveen la capacidad necesaria para trabajar con bases de datos y programas de contabilidad de gran tamaño y complejidad—ya sea en cada estación de trabajo, o a lo largo de la red. Y el que trabaja con estos discos está seguro de que puede obtener información al instante. De hecho, los discos en algunos modelos procesan datos a velocidades cuatro veces mayor que los de los computadores personales del pasado.

Y la noticia más pequeña sobre el Sistema Personal/2 es acerca de sus diskettes de 3,5 pulgadas, muy pequeños en tamaño, pero inmensos en capacidad—una combinación que significa más comodidad y menos intercambio de diskettes. Dependiendo del modelo, un diskette puede almacenar entre 360 y 720 páginas¹ de información—con unidades de 720KB ó 1,44MB.

Diseño y documentación elegantes

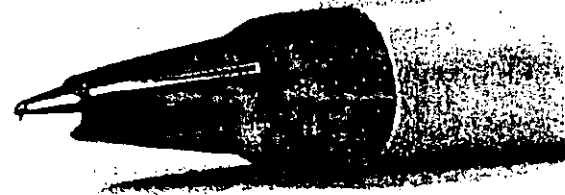
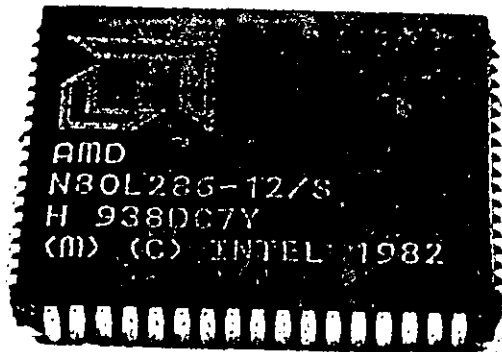
A pesar de todas sus capacidades, el Sistema Personal/2 no ocupa todo el espacio en su escritorio. Los modelos de mesa son más pequeños y compactos que antes. Y en los modelos para uso sobre el piso, la pantalla y el teclado se pueden colocar sobre el escritorio.

La documentación para todos los modelos del Sistema Personal/2 es tan compacta y elegante como el sistema mismo. Y el sistema es tan fácil de usar, que uno puede aprender a operarlo, siguiendo un libro guía muy delgado, y un programa tutelar en diskette.

Instalación sin interruptores, y expansión rápida

Las interfaces físicas y enchufes en el Sistema Personal/2 son tan simples, que su instalación se hace en minutos. No hay que seleccionar ningún interruptor, ni añadir ninguna placa, ni levantar ninguna cubierta, a menos que usted decida ampliar las capacidades del sistema. Inclusive, esta tarea es tan simple, que cualquier persona puede añadir tarjetas adaptadoras para ampliar las funciones y dispositivos deseados.





80286

En el año de 1984 IBM lanzó al mercado la primera computadora basada en el microprocesador INTEL 80286, la IBM PC AT. Esta computadora forma parte de la familia de computadoras llamadas PC's, las cuales están inundando los escritorios de millones de usuarios en todo el mundo.

Actualmente existen numerosos fabricantes de computadoras compatibles con la computadora IBM PC AT, a las cuales se les denomina en forma genérica AT's, y se espera que este tipo de máquinas dominen el mercado mundial en los años 1990 y 1991 en forma importante.

La presente reseña pretende explicar las generalidades de este tipo de computadoras, así como ofrecer una guía de compra de los equipos más importantes disponibles en el mercado mexicano.

Sabía usted...

A las computadoras con procesador 80286 compatibles con la IBM PC AT se les conoce como AT's. El concepto de compatibilidad abarca dos aspectos: la compatibilidad en hardware y en software. La compatibilidad en hardware se refiere a la posibilidad de intercambiar componentes o subensambles de una computadora a otra. Esto significa que si dos computadoras son 100% compatibles en hardware entre sí, estaríamos en posibilidad de intercambiar los teclados, las fuentes de poder, los monitores o cualquier otro subensamblable. Por otro lado, la compatibilidad en software se refiere a la posibilidad de poder utilizar los mismos programas en computadoras compatibles entre sí. Este tipo de compatibilidad resulta la más importante, pues lo que hace productivo un equipo de cómputo son los programas que se puede utilizar en él.

A continuación se describen los diferentes componentes que integran una computadora AT, tratando de ilustrar lo más posible esta explicación con las fotografías que se muestran en las páginas siguientes.

La tarjeta principal

Una computadora AT, al igual que la mayoría de las micro y minicomputadoras que se fabrican hoy en día, se compone de diversos subensambles que funcionan alrededor de una tarjeta principal o también denominada tarjeta madre (ver ilustración de las páginas 12 y 13). Gran parte de los componentes de la microcomputadora a excepción de la fuente de poder, el teclado o las

unidades de disco se encuentran incorporados en la tarjeta principal o en tarjetas que se conectan a ésta en las ranuras de expansión.

En una tarjeta principal típica AT podemos encontrar los siguientes componentes:

El procesador

El procesador es, por decirlo de alguna forma, el cerebro de la computadora. Es el lugar donde se ejecutan todas las instrucciones aritméticas, lógicas y de control, y por lo tanto determina en primera instancia la capacidad de la computadora.

Las computadoras AT cuentan con un procesador 80286, que consiste en una pastilla de 68 patitas de muy alta integración (esto significa que tiene incorporado miles de transistores y otros componentes electrónicos por centímetro cuadrado). Este microprocesador maneja 16 bits tanto para el procesamiento de instrucciones y operaciones internas, como para la comunica-

ción con los dispositivos de entrada y salida.

El reloj maestro

Una computadora requiere de un reloj maestro que coordine la ejecución de todas las operaciones que realizan el microprocesador, la memoria y los dispositivos de entrada y de salida. La IBM PC AT se lanzó al mercado con un reloj maestro operando a 6 MHz, es decir, una computadora con un procesador 80286 capaz de realizar 6 millones de instrucciones por segundo. INTEL ha desarrollado nuevas versiones del procesador 80286 funcionando a 10 y 12 MHz, mientras que Harris fabrica procesadores incluso a velocidades de 16 y 20 MHz. Esto ha permitido el desarrollo de computadoras compatibles más veloces que los fabricantes han llamado "turbo", e incluso "super turbo". Un factor fundamental al comprar una computadora AT (especialmente si se va a destinar como un servidor de red o de un sistema multiusuario) es la velocidad del reloj.

La memoria ROM (BIOS)

Las computadoras tiene dos tipos de memoria: la memoria permanente (exclusivamente para lectura) y la memoria de lectura y escritura. La memoria de lectura (ROM Read Only Memory) sirve para almacenar el BIOS, que consiste en el conjunto básico de programas que necesita el procesador para diagnosticar y ejecutar programas de más alto nivel. La ROM está formada por 2 pastillas de 28 patitas, alimentadas permanentemente por baterías alcalinas o de litio. El BIOS entra en operación al encenderse la computadora y su primera función es la del reconocimiento de los dispositivos (incluyendo memoria, controladores y puertos) con los que cuenta. Una vez realizada esta operación, el trabajo del BIOS es totalmente internos y totalmente transparente (invisible) para el usuario. Existen varios fabricantes de BIOS, entre los más importantes se encuentran Phoenix, Award, American Megatrends (AMI).

La memoria RAM

La memoria RAM (Random Access Memory) es una memoria de lectura y escritura, que tiene la característica de ser volátil, esto es que sólo opera mientras esté funcionando la alimentación de corriente. Esta memoria es el lugar donde el procesador guarda la información que va utilizando durante la ejecución de un programa. Con el desarrollo de aplicaciones y sistemas operativos más complejos, los requerimientos de memoria RAM se han visto incrementados sensiblemente. Baste recordar que la pri-

mera PC venía configurada con 128 Kb (Kb es la abreviación de Kilo bytes, que significa mil palabras computacionales) cuando en la actualidad es posible configurar una computadora hasta con 8 Mb (Mb es la abreviación de Mega bytes, que significa un millón de palabras computacionales) en la tarjeta principal, esto es, 63 veces más.

En términos, la memoria RAM consta de pastillas que pueden almacenar 64 Kb, 256 Kb o 1 Mb de memoria. Es importante conocer la capacidad máxima de memoria que puede ser instalada en la computadora sin la necesidad de tarjetas de expansión adicionales. Una computadora AT necesita de un mínimo de 512 Kb de memoria para funcionar, aunque en la práctica 640 Kb es lo mínimo necesario para ejecutar la mayoría de las aplicaciones disponibles para estos equipos.

Puertos

La computadora puede comunicarse con dispositivos y equipos periféricos (tal es el caso de las impresoras, graficadores, ratón, etc.) o con otras computadoras por medio de los puertos. Normalmente viene configurada una AT con un puerto serial y un puerto paralelo. El puerto paralelo se utiliza normalmente para conectar la impresora, mientras que el puerto serial se puede utilizar para el ratón, para la comunicación con otras computadoras o para la conexión de un modem.

Chip set AT

A partir de 1986 han aparecido

en el mercado los llamados "chip set", que constan de 3 a 6 circuitos en los que se encuentran contenidos los circuitos lógicos que componen una computadora AT tradicional (la cual requiere de más de 100 circuitos y componentes discretos). Esto ha permitido una sensible reducción en el espacio requerido para fabricar una tarjeta principal. A las tarjetas de tamaño reducido que cuentan con un "chip set" se les llama "Baby AT".

Controladores de disco duro y disco flexible

Para la utilización de unidades lectoras de disco duro o flexible, la microcomputadora cuenta con un controlador que por lo común viene integrado en una tarjeta que se inserta en una ranura de la tarjeta principal. De este controlador se desprenden dos cables para cada unidad lectora: uno de datos y otro de control. En fecha reciente algunos fabricantes han incorporado el controlador de discos en la tarjeta principal.

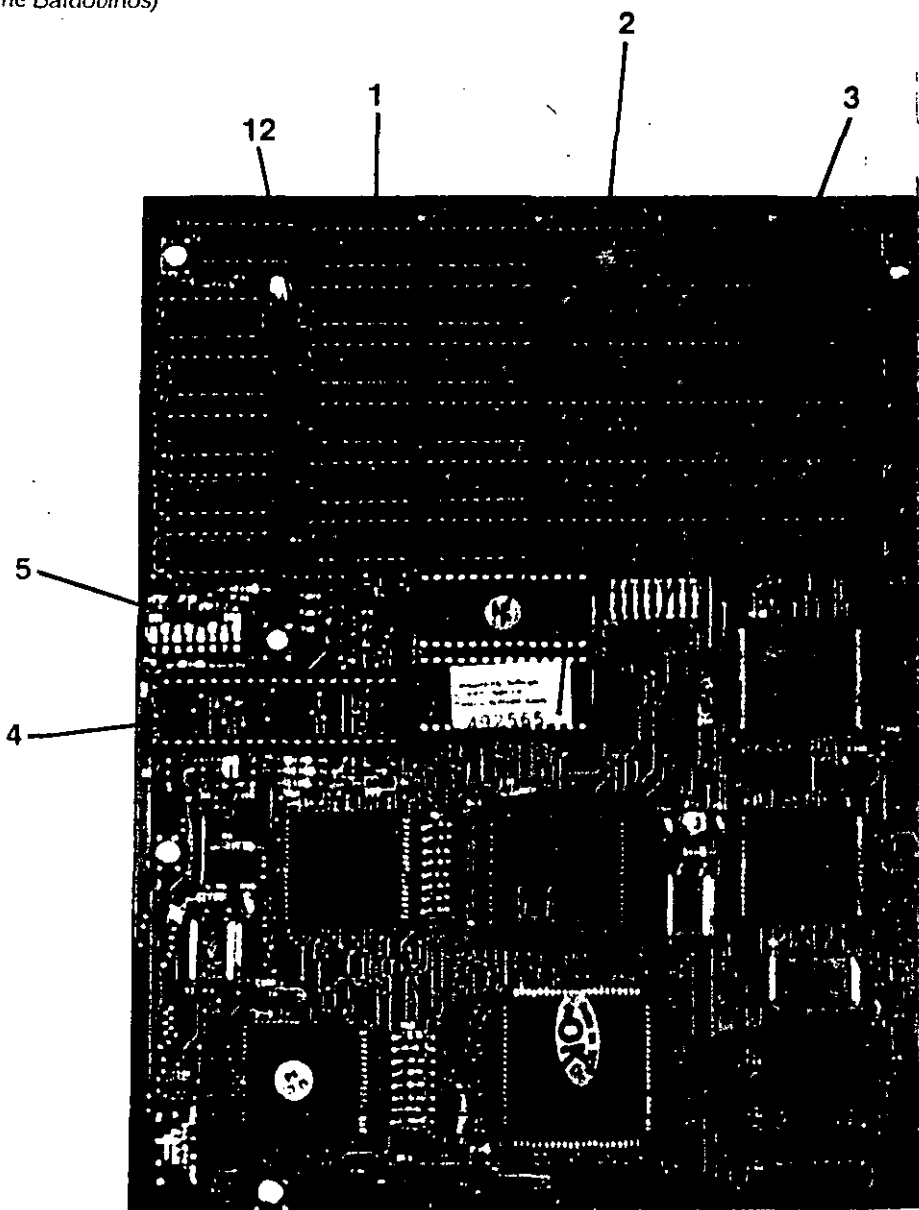
Unidades de disco flexible

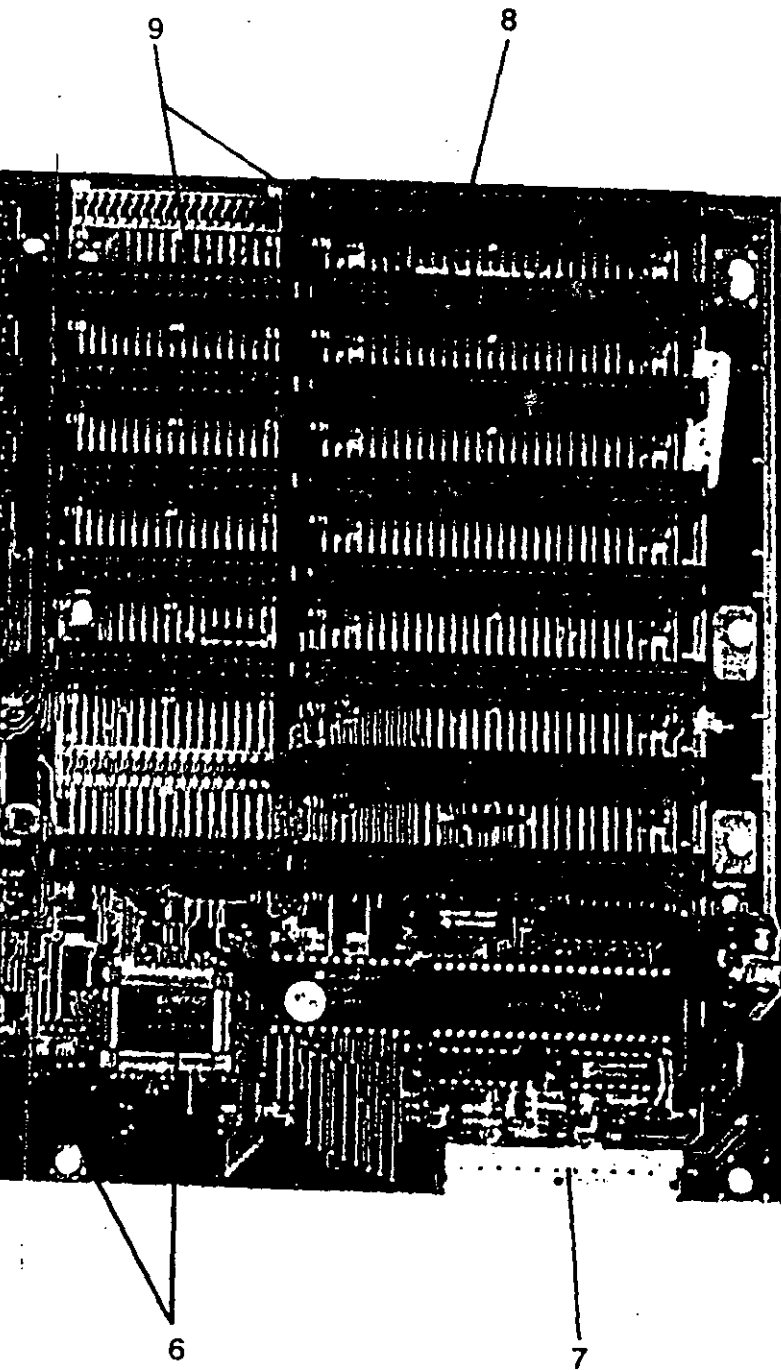
Las unidades de disco flexible que que por lo general se emplean en las ATs son de 1.2 Mb en 5 1/4" y 1.4 Mb en 3 1/2", a diferencia de 360 Kb y 720 Kb, respectivamente que se pueden utilizar en las PC XT. Las unidades de 3 1/2" están haciéndose cada vez más populares debido a su tamaño más reducido, la mayor durabilidad y resistencia de los diskettes de esta dimensión, y a que la nueva generación

continúa en la página 26

La tarjeta principal

La fotografía muestra una tarjeta principal AT Baby de 12 MHz. (Fotografía de Jaime Balduino)





- 1 DRAM de 1 Mb
[4 carrilleras de 9 chips 41256]
- 2 BIOS [ROM]
- 3 Procesador 80286
- 4 Ranura para procesador
numérico 80287 80287
- 5 Cristal de reloj maestro
- 6 Chipset AT
- 7 Conector fuente de poder
- 8 Ranura de expansión de 8 bits
- 9 Ranura de expansión de 16 bits
- 10 Conector de teclado
- 11 Batería para BIOS
- 12 Conectores para los interruptores
del gabinete [reset, turbo y llave]

de computadoras de IBM (el Sistema Personal 2) sólo incluyen este tipo de unidades lectoras. Lo deseable para cualquier usuario sería el poder contar con ambos tipos de unidades hasta que no exista un estándar generalizado, lo que implica una inversión adicional de 150 a 200 dólares.

Unidades de disco duro

Los discos duros se han convertido en la forma más económica de poder almacenar grandes cantidades de información con alta velocidad de acceso. Las capacidades tradicionales de almacenamiento son de 20, 30, 40, 60, 80 y 110 Mb; incluso hay modelos que pueden aceptar hasta gigabytes (miles de millones de bytes). Evidentemente a mayor capacidad mayor costo. Pueden estar disponibles en tamaños de 5 1/4" o 3 1/2" como en los discos flexibles, pero a menos que se trate de computadoras portables o muy compactas, esto no es muy trascendente puesto que el disco duro no se saca del gabinete de la computadora.

Una característica que es de suma importancia en un disco duro es la velocidad de lectura y escritura. La velocidad de lectura y escritura está determinada por el tiempo de acceso del disco: a menor tiempo de acceso mayor velocidad de lectura y escritura. Los tiempos de acceso más comunes van de 80 a 16ms (milisegundos), siendo estos últimos los utilizados en aplicaciones de uso pesado y en el que se requiere de gran velocidad de acceso.

El gabinete de la computadora cumple dos funciones: uno estético y otro práctico. La mayoría de los fabricantes de computadoras están tendiendo a reducir al mínimo el espacio que ocupa una máquina, pues a nadie le gusta tener un armastote encima del escritorio. Aquellos que requieren de mucho espacio para la conexión de varios discos, cinta de respaldo, tarjetas de expansión, etc., lo más recomendable es optar por una configuración en torre, la cual por estar diseñada para colocarse en el piso resulta en la mayoría de los casos poco estorposa. En el gabinete se incluyen normalmente un botón de reset (restablecimiento de la computadora), selector de velocidad turbo, "llave de seguridad" (que por cierto, la misma llave sirve para casi todos los gabinetes) y, en algunos casos, un indicador de la velocidad a la que está funcionando el reloj de la computadora.

Teclado

El teclado es una parte que si bien no se considera como un aspecto crítico en la elección de un equipo, es el medio por el cual nos comunicamos con la computadora. Un buen teclado de "click" puede hacer una enorme diferencia en productividad, en particular para usuarios que tienen que capturar una gran cantidad de datos. Tal vez usted ha de haber visto algunos teclados de mala calidad que con el tiempo se ponen "aguados". Antes de comprar una computadora sienta el teclado y evite a toda costa teclados corrientes.

La elección de un buen monitor y especialmente uno que se ajuste a los requerimientos de las aplicaciones que habrán de utilizarse, es uno de los aspectos más importantes en el momento de comprar de una computadora.

El primer punto a considerar es la resolución. Si se va a hacer uso de aplicaciones como diseño, edición o graficación, una alta resolución es indispensable. El color puede ser una ayuda muy buena en este caso, pero la resolución es el factor más importante. Existen aplicaciones como las presentaciones por computadora y los juegos donde el color es más importante que la resolución. Con el descenso en los precios de los monitores VGA (muy alta resolución y 256 colores disponibles) es posible tener ambas cosas: alta resolución y color a un precio razonable. Debe tenerse cuidado con los monitores CGA (Color Graphics), pues son los monitores de menor resolución tanto en monocromático como en color, y están discontinuados en la mayor parte del mundo.

Antes de comprar una computadora es necesario hacer una evaluación de diversos factores: tecnología, capacidad de la configuración que se desea comprar, precio, garantía y soporte.

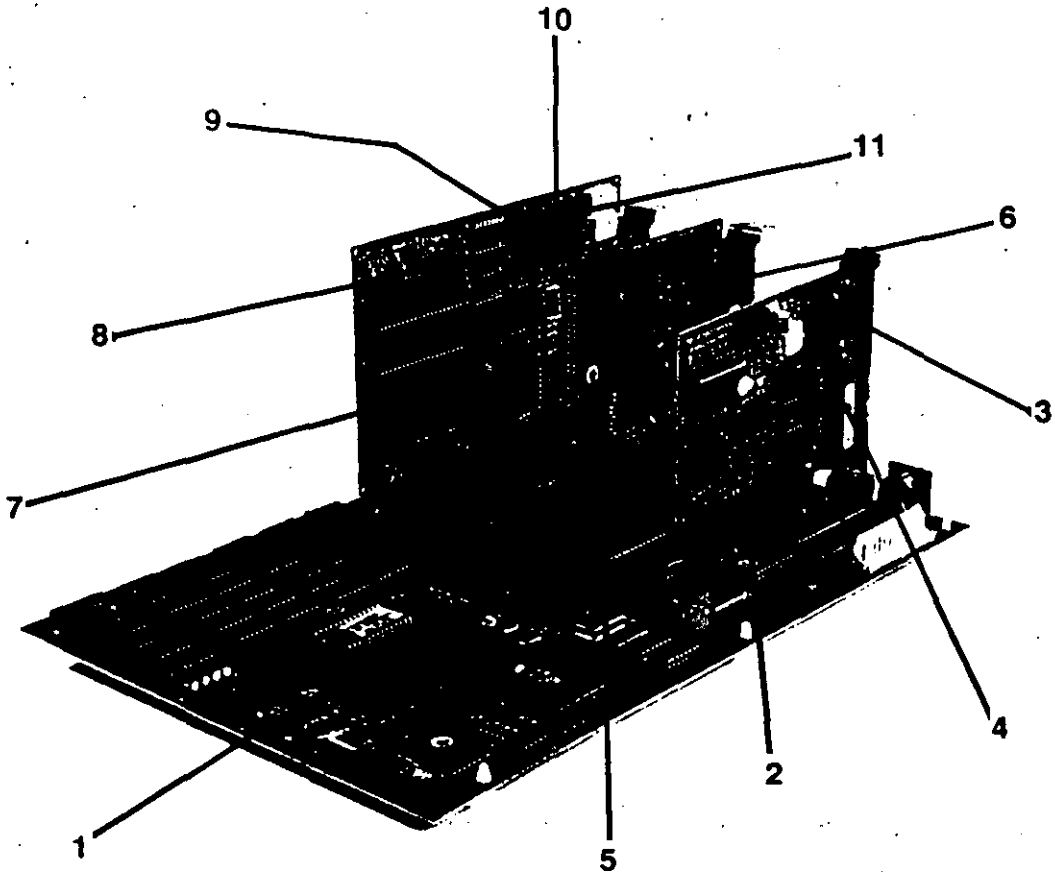
En este número se incluye una tabla comparativa que muestra diversas características de los equipos más importantes en nuestro país.

Elija usted.

Las tarjetas de una AT

La fotografía muestra las tarjetas que componen una computadora AT convencional: la tarjeta principal, la tarjeta de video, la tarjeta de puertos y la tarjeta controladora de discos. Al agregar una fuente de poder, un gabinete, un monitor y un teclado tenemos una computadora AT. (Fotografía de Jaime Baldwinos)

- | | | | |
|---|----------------------------------|----|------------------------------------|
| 1 | Tarjeta principal | 7 | Tarjeta controladora de disco duro |
| 2 | Tarjeta de video | 8 | Conector de disco duro 1 [datos] |
| 3 | Conector de video | 9 | Conector de disco duro 2 [datos] |
| 4 | Conector segundo puerto paralelo | 10 | Conector de discos duros [control] |
| 5 | Tarjeta serial/paralelo | 11 | Conector discos flexibles |
| 6 | Conector puerto paralelo | | |



Una AT con todos sus componentes

Monitor VGA

Cabinete

Pedestal giratorio

Unidad de disco flexible de 3.5"

Llave de seguridad

Tarjeta de bus de dos ranuras (tamaño completo, 16 bits, ISA)

Indicador de encendido

Controlador de memoria

Indicador de operación de la unidad

Discos 5.25"

Unidad de disco duro

256Kb de RAM en el video

Fuente de poder

Controlador super VGA

Ampliación del RAM de video

RAM estándar IBM (modo de página)

Entradas SIMM hasta para 4Mb

Batería

Interfase de unidad AT

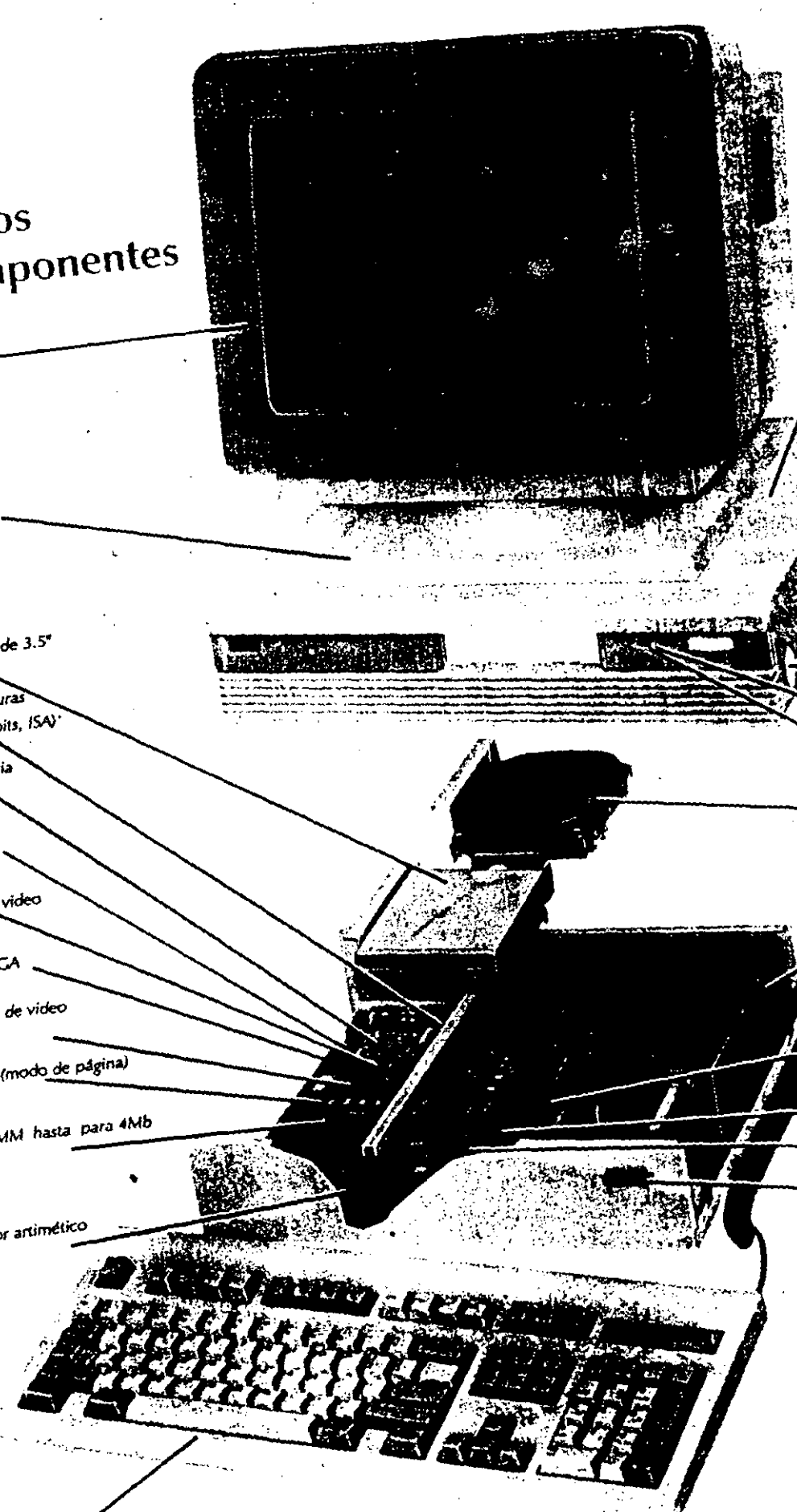
Co-procesador aritmético

Tarjeta plana

botón de encendido/apagado

Teclado

Cortesía de Siga Micros



El predominio del procesador 80286 en el campo de las microcomputadoras compatibles llegará a su nivel más alto en el presente año. Según las estimaciones de la firma Dataquest, las ventas de este tipo de máquinas llegarán a los 8 millones de unidades y no será sino hasta mediados de 1991 en que la tecnología del 80386 empezará a dominar el mercado. Una de las razones más importantes que han impulsado las ventas del 80286 ha sido el surgimiento de software que requiere de este procesador y que es inoperante en las máquinas del tipo XT (con procesador 8088). Tal es el caso de todos los programas que funcionan en el ambiente Windows (interfase gráfica desarrollada por Microsoft), las redes locales y los desarrollos en el sistema operativo Xenix.

En las páginas siguientes se muestra una tabla comparativa con las características básicas de los equipos 80286 más importantes en México con el objeto de facilitar al usuario la elección del equipo que más se ajuste a sus necesidades. Esta tabla también puede servir de referencia en el caso de encontrarse con un equipo que no aparezca en la presente reseña. A continuación se comentan los aspectos considerados en la tabla.

Velocidad

La velocidad es uno de los criterios fundamentales en la elección de un equipo, especialmente si se trata de una máquina de trabajo pesado como puede ser un servidor de red local, una estación de edición o una estación de diseño asistido por computadora. En la tabla comparativa se indica la velocidad a la que trabaja el procesador, complementándose con las gráficas 1 y 3, que indican la velocidad de procesamiento de la computadora.

Dispositivos integrados en la tarjeta principal

Algunos fabricantes de computadoras han optado por integrar en la tarjeta principal algunas de las funciones que normalmente se encuentran en tarjetas separadas. Tal es el caso de los puertos serial y paralelo, el controlador de discos flexibles, el controlador de disco duro, la interfase de video y el puerto para ratón. Esta integración permite un ahorro importante en el costo de fabricación y la posibilidad de reducir las dimensiones del gabinete. Sin embargo, esta integración trae como desventaja el hecho de que alguna falla en alguna de las funcio-

nes de la tarjeta principal requiere en muchos casos el reemplazamiento completo de esta. En México ya se han vivido amargas experiencias con fabricantes que no cuentan con las refacciones necesarias para dar mantenimiento a este tipo de equipos, que se traducen en molestias continuas para los usuarios. La ventaja de contar con las funciones en tarjetas separadas es por otro lado el menor costo de mantenimiento y la posibilidad de configurar el equipo con opciones diferentes a las proporcionadas por el fabricante sin tener que pagar por las funciones ya integradas.

Capacidad de memoria

La columna del lado izquierdo, correspondiente a la capacidad de memoria, indica la memoria básica con la cual se vende la computadora. La columna de la derecha indica la capacidad de memoria que se puede instalar en la tarjeta principal sin la necesidad de utilizar una tarjeta de expansión. Esto no tiene nada que ver la memoria direccionable por el procesador, que en la mayoría de los casos es de 16 Mb. Es importante considerar que la velocidad de acceso a la memoria que se encuentra instalada en la tarjeta principal es superior a aquella que se encuentra en tarjetas de expansión.

Ranuras de expansión

Las ranuras disponibles para expansión pueden ser un factor importante, para aquellos que deseen conectar múltiples tarjetas adicionales como es el caso de un controlador de láser, un scanner, un modem (interno), un fax de tarjeta, una interfase MIDI, una tarjeta multipuertos u otra de las tantas posibilidades que están disponibles en la actualidad. Estas tarjetas pueden tener un contacto de 8 o

16 bits (las tarjetas de 8 bits son las que se utilizan en las PC's XT).

Unidades de disco

En la tabla se indica el tipo de disco flexible que viene integrado en la configuración básica. Es posible en todos los casos incorporar unidades adicionales de 3 1/2", lo cual no sucede con las unidades de 5 1/4". En el caso de IBM por ejemplo, es necesario incorporar un controlador especial para instalar una unidad de 5 1/4" y en el caso de NCR simplemente no hay espacio para este tipo de unidad.

Gabinete

En la tabla comparativa se podrá ver algunos fabricantes han optado por un nuevo tipo de gabinete llamado 'slim' (delgado), el cual, como su nombre lo dice, es especialmente delgado y práctico para utilizarse sobre el escritorio.

Precios

Los precios que se indican en la tabla comparativa son precios válidos a la fecha de cierre de la presente publicación y son precios de lista al público sugerido por los fabricantes. Los precios finales al usuario final variará según el distribuidor. Cabe aclarar que las configuraciones se indican en la tabla sólo corresponden a una comparación de configuraciones similares. En la mayoría de los casos cada fabricante cuenta con diversos modelos y configuraciones.

Antes de comprar una computadora asegúrese de no olvidar los siguientes puntos:

- poliza escrita de garantía
- servicio
- capacidad técnica del distribuidor

Buena suerte.

Tablas comparativas

Tabla de computadoras 80286

Marca	Fabricante	Modelo	Velocidad	Integrado en tarjeta principal				Capacidad de memoria		Bios
				Puerto serial/paralelo	Cart. Unidad de disco	Controlador de disco	Puerto para mouse			
Acer	CompuTec	915P	10 MHz	2/1	S	S	S	N	640Kb 8Mb	Award
Acer	CompuTec	915V	12	2/1	S	S	S	S	1Mb 8Mb	Award
Ashton Inter.	Ashton Inter.	AI-616	16	2/1	S	S	N	N	1Mb 4Mb	Phoenix
Denki	Denki	Ctel-286	12	0/0	N	N	N	N	640Kb 1Mb	No ident.
Digitel	Prodigit	Digitel AT/16	16	0/0	N	N	N	N	1Mb 8Mb	Phoenix
Digitel	Prodigit	Digitel AT/20	20	0/0	N	N	N	N	1Mb 8Mb	Phoenix
Elektra	Flujo de Datos	Elektra AT	12	0/0	N	N	N	N	1Mb 1Mb	Phoenix
IBM PS/2	IBM de México	Mod. 50z	10	1/1	S	S	S	S	1Mb 2Mb	IBM
IBM PS/2	IBM de México	Mod. 60	10	1/1	S	S	S	S	1Mb 2Mb	IBM
Memorex-Telex	Memorex-Telex	7405	10	1/1	S	N	S	N	1Mb 1Mb	Phoenix
Numen	Intelecsis	Numen 100	10	0/0	N	N	N	N	640Kb 1Mb	Phoenix
Numen	Intelecsis	Numen 400	16	0/0	N	N	N	N	1Mb 1Mb	Phoenix
NCR	Siga Micros	NCR PC286	16	2/1	S	S	S	N	1Mb 2Mb	Phoenix
Onyx	Microológica	Onyx 286	12	0/0	N	N	N	N	1Mb 1Mb	Phoenix
Pine	PB Computación	Super Set 400	16	1/1	S	N	S	S	1Mb 8Mb	Phoenix
Soltec	Mexel	Paragon 286SL	10	1/1	S	N	S	N	1Mb 1Mb	Phoenix
Sprint	Sahuaro	Sprint 286	12	0/0	N	N	N	N	512Kb 4Mb	Award
Vectra	HP	Vectra ES/12	12	0/0	N	N	N	S	640Kb 1Mb	Phoenix
Televideo	Ind. Televideo	TeleOAS IIA	12	2/1	N	N	N	N	1Mb 1Mb	Phoenix
Wang	Wang	PC-AT-260-40G12		2/1	S	N	N	N	640Kb 1Mb	Phoenix

Nota: Los precios mencionados son precios de lista sugeridos al público y están sujetos a cambio. El precio final puede variar según el distribuidor. En todos los casos, los precios indicados incluyen las características indicadas en la tabla.

S es indicación afirmativa

N es indicación negativa.

Ranuras de expansión

8 bits
16 bits

5 1/4"
3 1/2"

Chip Set

Puerto paralelo
Puerto serial
Cort. video
Cort. disco duro

En tarjetas adicionales

5 1/4"
3 1/2"

Espacio libre

Gabinete

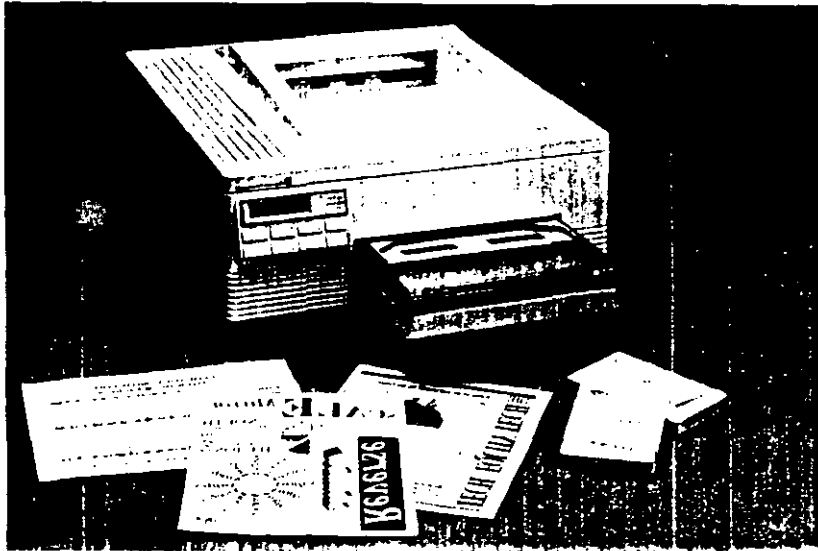
Disco Duro

Precio

	0	4	S	N	Chips & Tec.	N	N	N	N	0	1	Escritorio	40Mb	\$2,298 (c/Monitor TTL 14")
	0	4	S	N	Chips & Tec.	N	N	N	N	0	1	Escritorio	40Mb	\$3,120 (c/Monitor mono VGA y mouse)
	1	3	S	N	Chips & Tec.	N	N	S	S	1	1	Mini Torre	40Mb	\$3,526 (c/Monitor TTL 14" mouse)
	1	3	S	N	----	S	S	S	S	1	1	Escritorio	40Mb	\$2,960 (c/Monitor TTL 14" mouse)
	1	3	S	N	VLSI	S	S	S	S	1	0	Escritorio Slim	40Mb	\$2,344 (c/Monitor TTL 14" blanco)
	1	4	S	N	Chips & Tec.	S	S	S	S	1	1	Torre	40Mb	\$2,899 (c/Monitor monocromático)
	1	3	S	N	Suntac	S	S	S	S	1	0	Escritorio	40Mb	\$1,843 (c/Monitor TTL 12" am...)
	0	6 (MCA)	N	S	VLSI	N	N	N	N	0	1	Escritorio	30Mb	\$4,396+cargos únicos (c/Mon. mono)
	0	6 (MCA)	N	S	VLSI	N	N	N	N	1	1	Torre	30Mb	\$6,549+cargos únicos (c/Mon. mono)
	1	3	S	S	Chips & Tec.	N	N	N	Y	0	1	Escritorio	20Mb	\$2,850 (c/Monitor mono VGA)
	1	4	S	N	----	S	S	S	S	1	1	Escritorio	30Mb	\$2,055 (c/Monitor TTL 14")
	1	4	N	S	Chips & Tec.	S	S	S	S	1	1	Escritorio	50Mb	\$4,037 (c/Monitor EGA 14")
	0	2	N	S	VLSI	N	N	N	N	0	1	Escritorio Slim	40Mb	\$4,440 (c/Monitor monocromático)
	0	2	S	N	Suntac	S	S	S	S	1	0	Escritorio	40Mb	\$3,040 (c/Monitor TTL 12")
	1	3	S	N	Chips & Tec.	N	N	N	N	0	1	Escritorio Slim	40Mb	\$2,895 (c/Monitor TTL 12")
	1	3	S	N	Chips & Tec.	N	N	N	S	0	2	Escritorio	40Mb	\$3,120 (c/Monitor mono VGA y mouse)
	1	4	S	N	Suntac	S	S	S	S	1	1	Escritorio	40Mb	\$1,890 (c/Monitor TTL 12")
	0	4	S	N	----	S	S	S	S	1	0	Escritorio	40Mb	\$4,890 (c/Monitor mono VGA)
	1	3	S	N	VLSI	N	N	S	S	1	2	Escritorio	40Mb	\$3,125 (c/Monitor TTL 14")
	1	3	S	N	Chips & Tec.	N	N	S	S	0	2	Escritorio	40Mb	\$3,673 (c/Monitor EGA)

Impresora HP LaserJet III

Hoja Técnica



Una revolución en resolución

- Tecnología de Resolución Mejorada que marca un nuevo estándar en calidad de impresión láser de escritorio.
- Tipografía escalable y un mayor número de fuentes; todo interconstruido.
- Impresión más rápida de gráficas con el sistema vectorial de gráficas HP-GL/2 integrado.
- Lenguaje de impresión PCL 5 con características mejoradas para el formateo de páginas.
- Compatibilidad con la impresora LaserJet Series II de HP.
- Impresión rápida y silenciosa de 8 páginas por minuto.
- Cartucho PostScript y cartucho de personalidad Epson FX/IBM Proprinter.
- Expansión de memoria a un costo reducido.

La revolución en resolución: calidad de impresión insuperable de Hewlett-Packard.

La exclusiva tecnología de Resolución Mejorada de Hewlett-Packard elimina la distorsión en líneas y ángulos, suaviza las curvas y agudiza los ángulos y las intersecciones. El resultado es una calidad insuperable de impresión en textos y gráficas.

El poder para hacer lo que nunca antes había hecho.

El lenguaje de impresora HP PCL 5 pone al alcance características más poderosas y prácticas. Combina la escalabilidad de tipografía y el formateo mejorado de las páginas, para proporcionar una gran variedad de formas nuevas en la presentación de textos y gráficas.

Tipografía escalable, flexible y rápida.

Pueden crearse miles de fuentes a partir de una sola tipografía escalable, hasta llegar a 999.75

puntos ocupando una página. Debido a que el lenguaje de impresión PCL 5 presenta la tecnología de escalamiento de tipografía Intellifont, ya no tendrán que adquirirse fuentes adicionales para obtener el tamaño de letra que se desea.

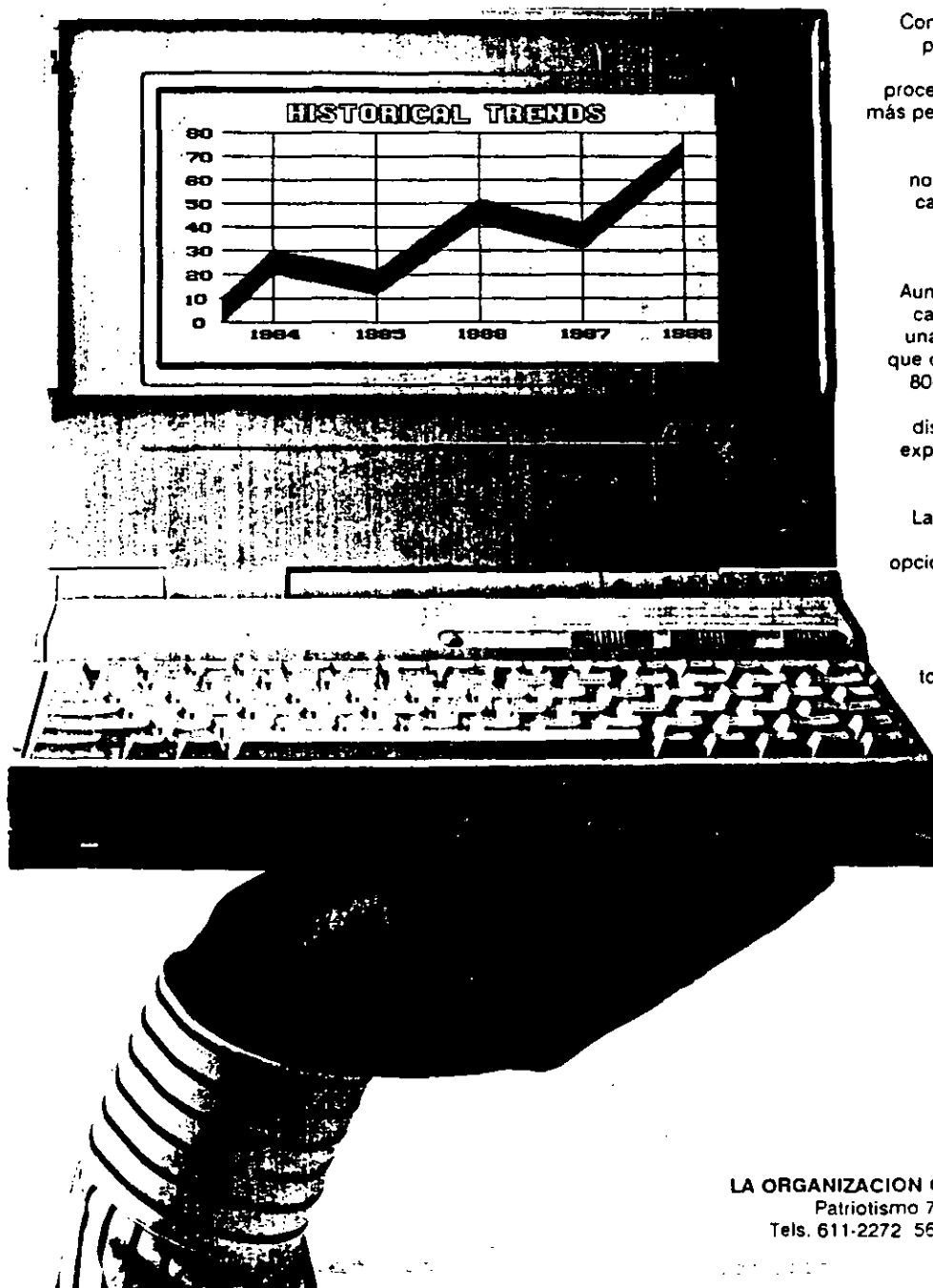
Esta flexibilidad se encuentra interconstruida, con ocho tipos escalables (familia CG Times y Universe) y 14 fuentes "bitmapped" (Courier y Line Printer), requiriéndose menos tiempo para transferir las fuentes a la memoria de la impresora; Adicionalmente, se puede seleccionar entre una gran variedad de fuentes y productos de tipografía escalable que ofrece HP Master Type Library.

Mayor rapidez en la impresión de gráficas.

HP ha integrado su lenguaje estándar de graficación HP-GL/2 en PCL 5; el resultado: impresión más rápida de gráficas ocupando menos espacio en el disco.

El lenguaje de impresión PCL 5, permite integrar el tipo de calidad de una publicación en gráficas de negocios. Se puede imprimir en orientación vertical (portrait) u horizontal (landscape) en la misma página, o girar el texto y las gráficas en intervalos de un grado. Imprime letras blancas en fondos negros, o letras delineadas o sombreadas, y rellena el texto con sombras o patrones. Imprime imágenes de espejo, y da al texto o a las gráficas un escalamiento uniforme o en un solo eje, para condensar o expandir las letras o imágenes.

Todo el poder de la computación.. en sus manos.



Conozca las nuevas computadoras portátiles COMPAQ, a la medida de sus necesidades de procesamiento de datos y al tamaño más pequeño hasta ahora imaginable.

La nueva COMPAQ LTE-286 no mide más de una hoja tamaño carta y sólo pesa 3 kg. La puede llevar a donde quiera, inclusive en su portafolios.

Aunque su tamaño es pequeño, su capacidad es tan poderosa como una computadora de escritorio, ya que cuenta con un microprocesador 80C286 a 12 MHz, con disco duro de 20 ó 40 Mb, soportando diskettes de 3.5". Su memoria se expande de 640Kb hasta 2.6 Mb, y cuenta con monitor CGA.

La nueva COMPAQ LTE-286 tiene la capacidad para un modem opcional y batería recargable que la mantiene funcionando por más de 3 horas y media.

Venga a GRUPO TEA y llévase todo el poder de la computación en sus manos.

COMPAQ
LTE/286



GRUPO TEA

LA ORGANIZACION COMPLETA EN INFORMATICA
Patriotismo 767 Mixcoac, 03910 México, D.F.
Tels. 611-2272 563-3567 660-2393 Fax 611-1471

BOLETIN TECNICO COLECCIONABLE

¿QUE ES UNA RED?

Una red simplemente es interconectar computadoras de forma que se puedan comunicar entre sí y compartir datos y recursos periféricos. A diferencia de otros enfoques de procesamiento, la Red es un sistema de procesamiento distribuido.

¿QUIEN NECESITA UNA RED?

- ¿Estoy compartiendo discos?
- ¿Necesito impresoras o capacidad en disco adicional?
- ¿Muchas personas necesitan acceder la misma información?
- ¿Requiero proteger datos sensibles?
- ¿Necesito comunicarme a una Minicomputadora o Mainframe?

Si la respuesta es "SI" a cualquiera de estos proyectos, una red:

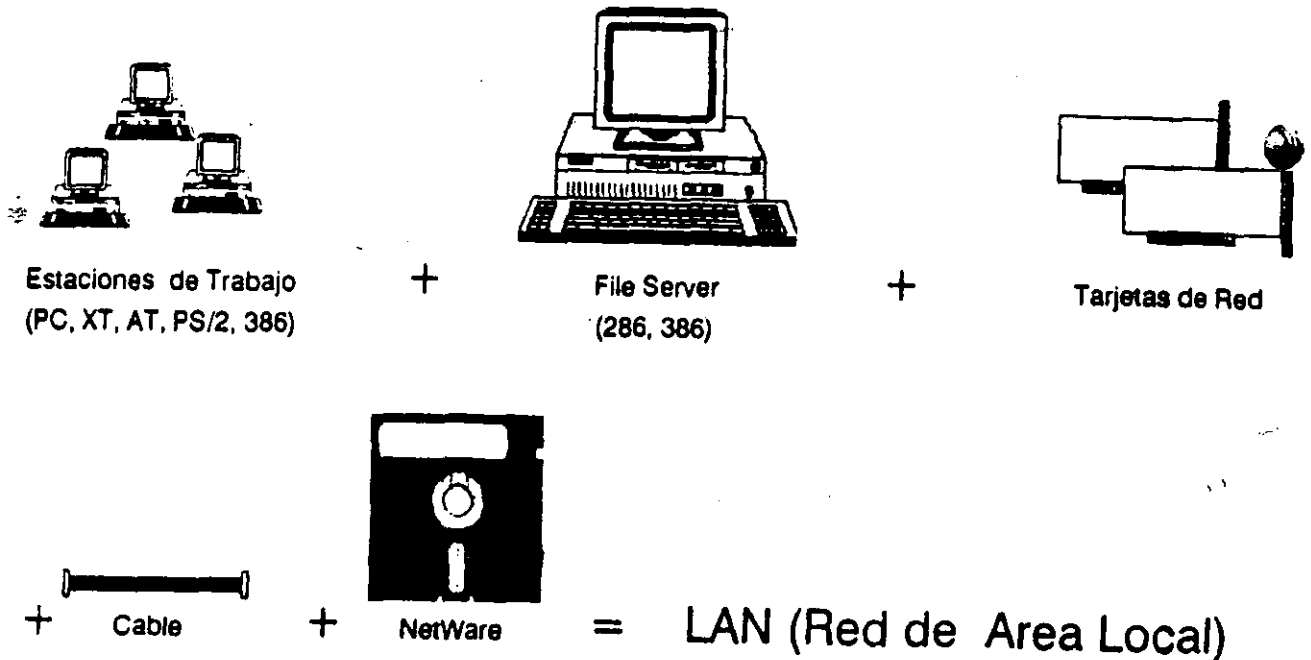
- Ahorra dinero
- Permite mayor utilización de recursos y evita gastos adicionales futuros
- Incrementa la productividad
- Mejora la comunicación
- Protege los datos contra fallas en el sistema
- Restringe el acceso a usuarios

¿COMO SE COMPARAN LAS REDES CON LAS MINIS?

	MINI	LAN
RENDIMIENTO (PERFORMANCE)	Proceso Central	Proceso Distribuido
EXPANSION	Límite establecido por el HW y el SW	Opción Limitada
COSTO	Muy Caro	Poco Caro
TECNOLOGIA DE APLICACION	Maduro no Expandiéndose	Muy rápida Evolución

PROCESAMIENTO DISTRIBUIDO. En un sistema Multiusuario, Minicomputadora o Mainframe, el sistema central tiene conectadas Terminales Tontas y el procesamiento de información ocurre en un sólo punto (el sistema central), a esto se le llama Procesamiento Centralizado. En contraste el Procesamiento Distribuido es mucho más flexible ya que dicho procesamiento de información se lleva a cabo en la estación de trabajo conectada a la red. Por lo que la red completa puede ser vista como un sólo dispositivo de cómputo. Es por este tipo de proceso que en México existen redes de más de 100 microcomputadoras.

¿QUE ELEMENTOS COMPONEN UNA RED?



¿POR QUE NETWARE?

- NetWare, el Sistema Operativo de NOVELL es el estándar "de facto" de la Industria con una participación de Mercado en USA del 65% y de más del 85% en México.
- NetWare protege la inversión existente y reconoce el derecho del usuario de escoger apropiadamente el Hardware y Software a utilizar.
- Corren todos los niveles de DOS y OS/2 y sus aplicaciones.
- Soporta múltiples tarjetas de red y protocolos de Comunicación.
- Capacidad ilimitada en disco (2 GB).
- Interconexión local y remota a PCs y Redes.
- Conexión a minis y mainframes.
- Contabilidad por los Sistemas de Tolerancia a fallas incluidos.
- Los más avanzados niveles de seguridad y administración del sistema.

N E T W A R E

Existen cinco tipos básicos de Netware, ELS I y II, Advanced, SFT y 386. Estas categorías difieren en el número de usuarios, esquemas de red, así como en el nivel de rendimiento y seguridad que proveen.

ELS (ENTRY LEVEL SOLUTION) NETWARE NIVEL I. VERSION 2.0a

Enfocado a muy pequeños grupos de trabajo que requieren compartir información e impresoras.

- * File Server NO-dedicado (Servidor y Estación de Trabajo al mismo tiempo). Requiere 1.6 MB RAM, AT,386 y PS/2, 30/286 en adelante.
- * Cuatro usuarios máximo (4 PC's).

ELS NETWARE NIVEL II. VERSION 2.15

Enfocado a grupos de trabajo de 8 usuarios máximo.

- * File Server dedicado (1 MB) o NO dedicado (2 MB).
- * Seguridad avanzada, contabilidad de recursos.

ADVANCED NETWARE VERSION 2.15

Ideal para grupos de más de 8 usuarios que no requieren la sofisticada Tolerancia a Fallas del SFT.

- * 100 usuarios.
- * Soporte a comunicaciones locales y remotas.
- * Soporte a discos externos e internos de alta capacidad (mayor a 100 MB).
- * Dedicado (1 MB) y NO dedicado (2 MB).

SFT (SYSTEM FAULT TOLERANCE) NETWARE VERSION 2.15

Enfocado a Corporaciones. Tiene las características de los Sistemas anteriores más:

- * Transaction Tracking (Registro de Transacciones para aplicaciones de Base de Datos).
- * Discos en espejo.
- * Dedicado.

NETWARE 386

Enfocado a instalaciones en donde el rendimiento de seguridad y funcionalidad son altas prioridades.

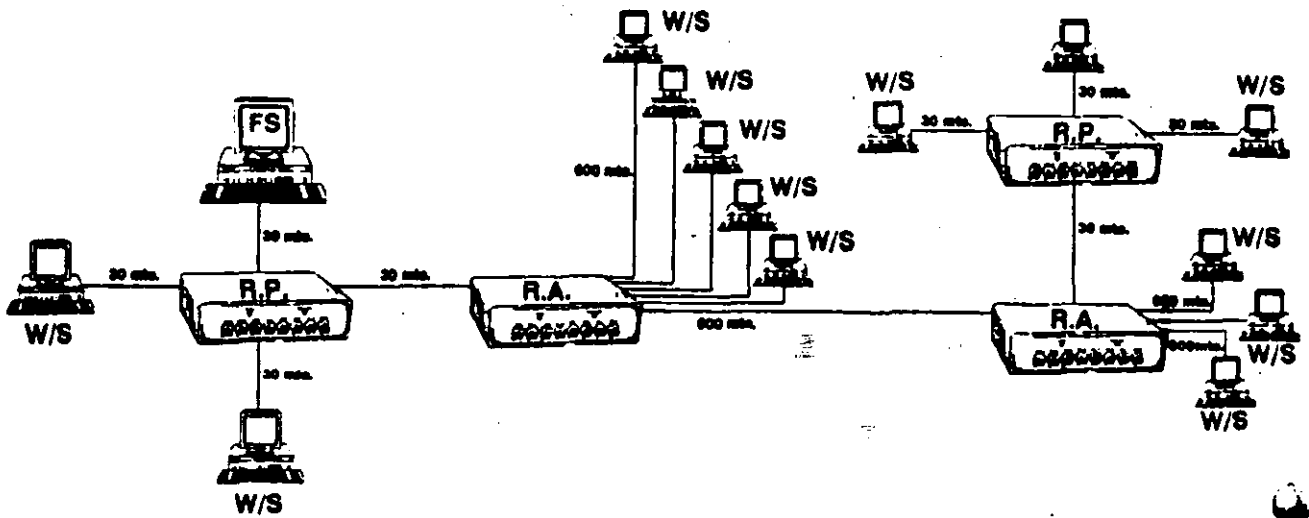
- * Tres veces más rápido.
- * Plataforma para los 90's.
- * Para procesadores 386 (32 bits).
- * 250 usuarios.

A R C N E T

- Esta Red es muy popular en México y de bajo costo. Tiene componentes básicos, tarjetas de red y repetidores.
- Tarjeta Arcnet (8 bits, para PC, XT, AT, 386 y PS/2 25 y 30).
- Tarjeta Arcnet PS/2 (para PS/2 50 en adelante).
- Repetidor Pasivo (4 nodos, 30 metros por nodo).
- Repetidor Activo (8 nodos, 600 metros por nodo).
- Cable Coaxial RG-62 (tipo videocassettera).

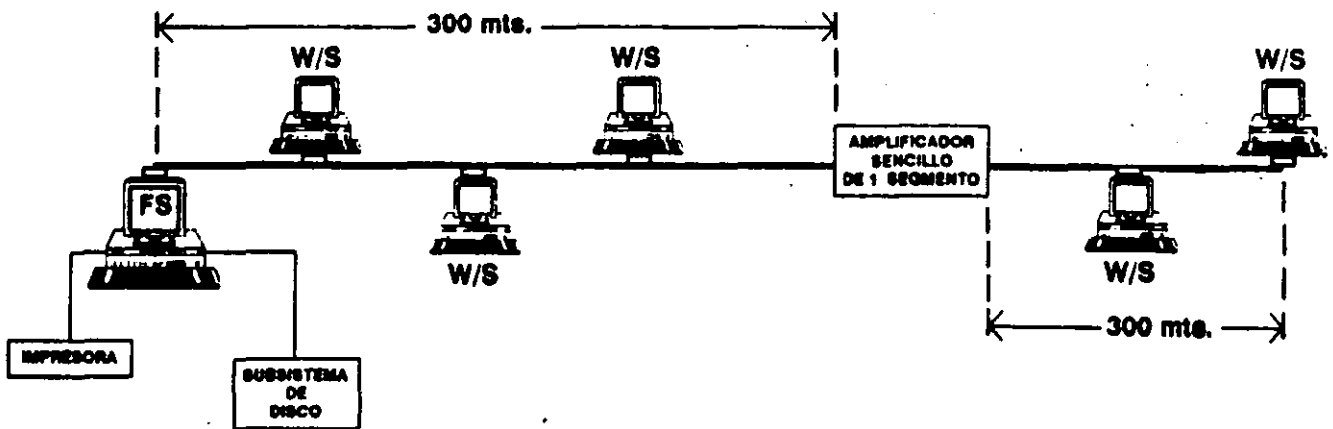
REGLA

- Distancia máxima entre A y P 30 metros.
- De 1 pasivo a cualquier cosa (PC, File Server o RA) la distancia máxima es de 30 metros.
- De 1 activo a todo es 600 metros.



ETHERNET

- Esta red se recomienda para trabajos pesados con mucho tráfico en el canal de comunicaciones y con accesos a disco constantes. La velocidad de transferencia de los datos en el cable de comunicación es de 10 Megabits por segundo.
- Esta topología de red es en Bus Lineal, las estaciones de trabajo se van anexando al troncal de cable coaxial con conectores tipo "T". El único componente son las tarjetas de red.
- Tarjeta EtherNet (F/S y W/S PC-XT,AT y 386).
- Tarjeta EtherNet PS/2 (F/S y W/S PS/2).
- Tarjeta EtherNet Plus (F/S, AT y 386).
- Cable coaxial RG-58.
- Soporta 300 metros en el Bus sin amplificador.
- Amplificador sencillo de 1 segmento para ampliar el Bus 300 metros.



Sistema Operativo NetWare 286 y 2.0A (Advanced NetWare, SFT NetWare y ELS I) utilizando como Servidor una PS/2 50Z, 55SX ó 70.

COMENTARIOS:

Los modelos de las máquinas mencionadas anteriormente, tienen problemas con la Versión 2.0A de NetWare para que funcionen como Server, el problema radica en el Controlador de Disco Duro. Para solucionar este problema se tiene una serie de archivos, los cuales "parchan" los Programas de Utilerías (INSTALL, COMPSURF, PREPARE) y el archivo que contiene las características de la Red que se va a instalar (NET\$OS.EXE.).

Como la Versión 2.0A sólo se está comercializando con el Sistema Operativo ELS I se necesita un diskette de 3.5" con tres archivos

- * 50ZNDDED.EXE - Para Server No Dedicado
- * 50ZUTILS.EXE - Para utilerías y File Server Dedicado
- * READ.ME - Documentos

PROCESO DE PARCHE:

Para ELS I o Advanced NetWare 286 No-Dedicado

- 1) Generar Sistema Operativo (NET\$OS.EXE) con el proceso normal.
- 2) Parchar programas de Utilerías y Sistema Operativo de la forma siguiente:
A>50ZNDDED NET\$OS.EXE
A>50ZUTILS INSTALL.EXE
A>50ZUTILS PREPARE.EXE (ELS I)
A>50ZUTILS COMPSURF.EXE
- 3) Preparar Disco Duro para NetWare.
- 4) Realizar proceso de instalación normal.

Para Advanced NetWare 286 Dedicado

- 1) Generar Sistema Operativo con el proceso normal.
- 2) Parchar los programas de la siguiente forma:
A>50ZUTILS NET\$OS.EXE
A>50ZUTILS INSTALL.EXE
A>50ZUTILS COMPSURF.EXE
A>50ZUTILS PREPARE.EXE (Si es SFT)
- 3) Preparar Disco Duro para NetWare.
- 4) Instalar el Sistema Operativo con el proceso normal.

Tipo de Controladora de Disco Duro para NetWare ELS I y ELS II.

CARACTERISTICAS:

En este tipo de Software no es posible seleccionar el tipo de Controladora de

DOS 5



PHOTOGRAPHY: JOHN WILKES

From fast formatting to memory management, from an undelete command to a user-friendly Shell, DOS 5.0 has the features you've been waiting for and a price tag that makes upgrading irresistible.

What's in It for You?

by Jeff Prosize

Cut through all the hype surrounding the release of MS-DOS 5.0 and the big question remains: What's in it for you? Is DOS 5.0 worth the cost of an upgrade when DOS 3.3 or 4.0 does everything you need? More importantly, if you're an IS manager looking at the cost of 100 or perhaps 1,000 upgrades, is the increase in productivity going to offset the cost of retrofitting all your PCs with DOS 5.0?

000245

FoxGraph™



The Most Powerful Microcomputer
Graphics System Ever!

Fox Software

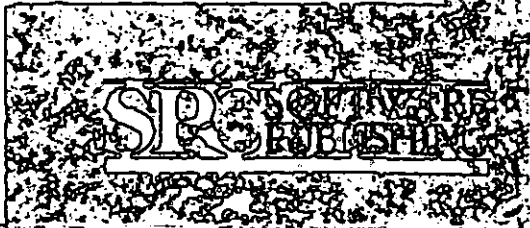
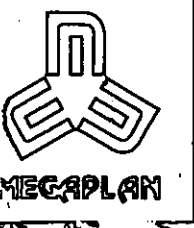
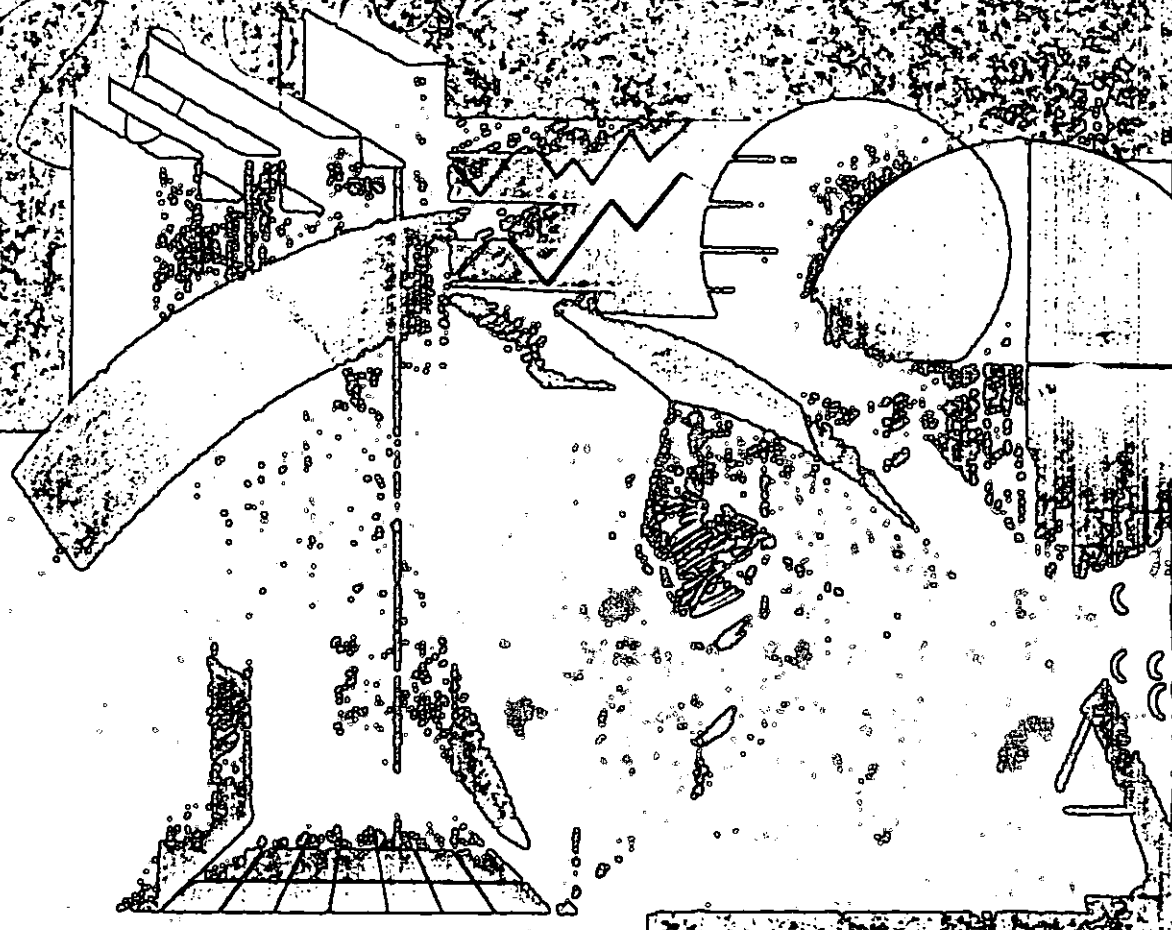
Nothing Runs Like a Fox.

NEW VERSION
Includes HyperShow™,
Chart Gallery, and Draw Partner™

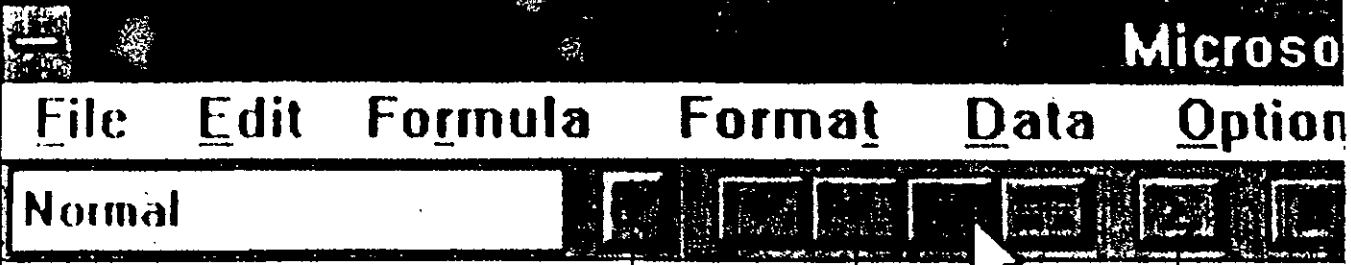
For IBM compatibles, DOS
000246

Harvard Graphics 2.3

Powerful
presentations
for the
business user



Fortunately it'll



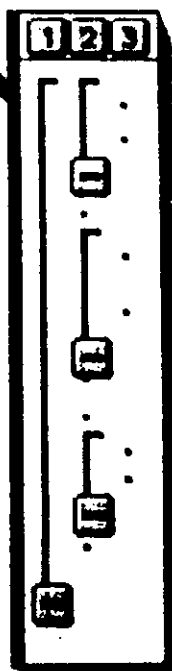
Normal

One step outlining. Outline worksheets to see only the detail you need.

One step bold and click here.

One step styles. Save and apply your own style. It's possible to mix up to 256 fonts, 16 of 16 million colors and 18 patterns.

One step formulas. Use Autosum™ to add a range of cells by double clicking.



Our exclusive Outlining feature lets you expand and collapse your worksheet. So you're able to display and print only the data you need to view or share with others.

Eliminating the need to create separate documents for your boss and your boss's boss. Furthermore, if you don't select specific parameters, the program will automatically outline your worksheet. You can also combine virtually hundreds of worksheets through the consolidation feature. Regardless of format or structure.

	1991	1992	1993
Canada	184,945	203,330	223,663
Mexico	49,252	49,768	50,763
United States	1,232,300	1,355,530	1,219,977
North America	1,466,437	1,608,645	1,491,973
France	184,945	194,087	203,791
Germany	308,075	369,630	373,357
Other European	67,819	92,423	104,846
United Kingdom		67,777	74,935
Europe		723,977	836,579
Australia		53,232	58,955
Japan	433,371	500,862	1,733,724
Korea	43,383	65,366	98,985
Taiwan	63,388	82,467	103,189
Far East	540,149	631,715	1,936,003
Total Revenue	2,888,889	3,474,183	4,388,874

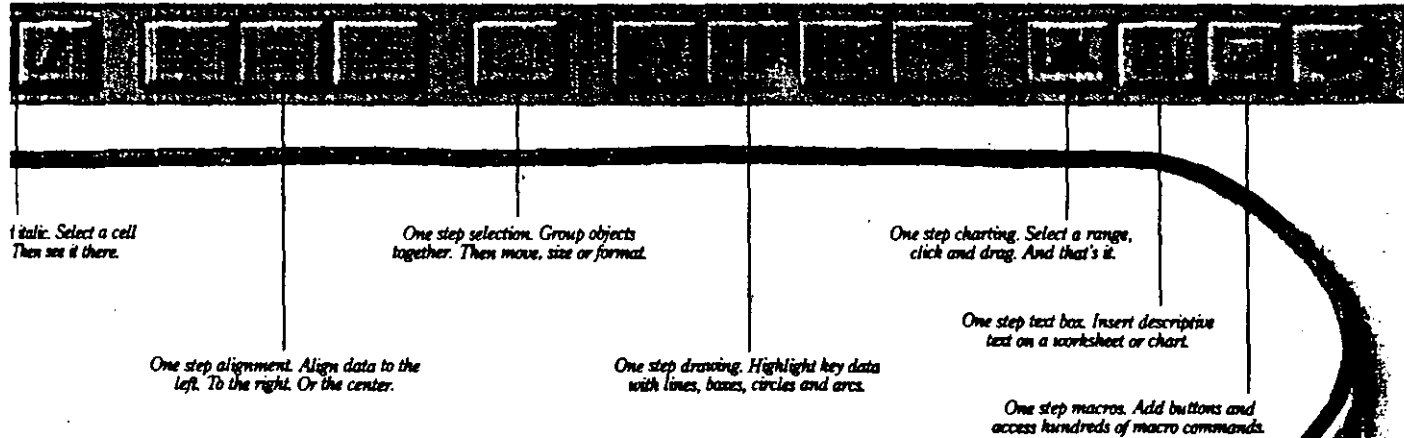
Select a range of cells and transform them into a chart on a worksheet. With one simple click. Or why not create a 3-D chart. Then rotate it 360°! How about a pie? With Microsoft Excel, you can choose from 68 different chart types, 24 of which are 3-D. Or even create your own. And because this is a graphical program, it's easy to combine text, data and graphics on a single page. Of course, it's WYSIWYG too. Which means you can actually see what you're working on.

*Offer good for current owners of Lotus 1-2-3, SuperCalc, Quattro Pro and VP Planner. Please allow 2-4 weeks for delivery upon receipt of order by Microsoft. Offer expires 9/30/91. Limit one per customer. Retailer price (416) 568-3503; outside the United States and Canada, call (206) 936-8661. Microsoft and the Microsoft logo are registered trademarks and Making It All Make Sense, Autosum, ThruLine and Windows are trademarks of Microsoft Corporation.

only take you one.

Microsoft Excel

File Edit Macro Window Help



Out of the many, many steps it took us to create Microsoft Excel for Windows, there were two that really had the most impact. The first was to listen to our users and to listen carefully. The second was to take that knowledge and implement it intelligently.

The result is outrageous power coupled with ease-of-use. It is this idea that drives all of our efforts. And has since we introduced the first graphical spreadsheet back in 1985.

A recent example of this is the Toolbar. This addition lets you do everything from formatting, to formulas, to outlining with one click of a button.

Best of all, it's easy to make the

move. Microsoft Excel comes complete with online help for 1-2-3[®] users. So you can actually learn by applying what you already know. And if you're currently using another spreadsheet, you can upgrade on a limited basis for only \$129.* Why not take the next step. See your reseller or call (800) 323-3577, Department 08.

Microsoft

1.
2.
3.



What The Experts Think.

Recently, *VARBUSINESS* conducted a survey of some people who know more about spreadsheets than anyone.

The people who sell, install, and support them for a living. The survey asked hundreds of spreadsheet dealers and resellers all kinds of questions about the top-rated spreadsheets. They were trying to find out what the experts rate as the best spreadsheet. And the experts told them.

SuperCalc®5.

In virtually every category from ease-of-use to technical support, SuperCalc5 was rated higher than Lotus® 1-2-3® and Microsoft® Excel.

We're not surprised. While Lotus and Microsoft have been fussing over their many different spreadsheet versions, we've been stuffing one spreadsheet, SuperCalc5,

with everything you would ever want. From spreadsheet linking to 3-D graphics and beautiful, presentation-quality output, SuperCalc5 has it all. And it runs on just about any PC.

But the best part is something they forget to ask about. The price.

At \$149.00, SuperCalc5 Costs Less Than Half The Price Of Lotus 1-2-3 And Excel.

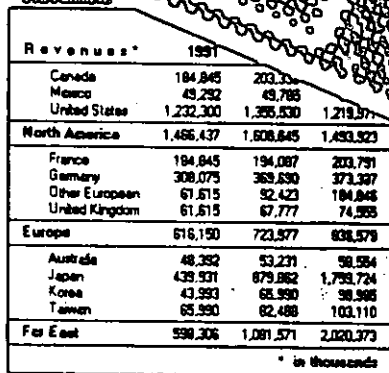
Heard enough? Call 1-800-645-3003 today for the name of the nearest dealer where you can buy the best spreadsheet there is.

If, you're willing to spend a couple hundred dollars less.

**COMPUTER
ASSOCIATES**

Microsoft Excel for Windows

New!
Version 3.0
with
Toolbar...



The screenshot shows the Microsoft Excel for Windows interface. A spreadsheet window is open, displaying revenue data for various regions from 1991 to 1993. The data is presented in a table format. To the right of the spreadsheet, a 3-D bar chart is visible, representing the same data. The chart has three bars for each region, corresponding to the years 1991, 1992, and 1993. The Excel window title bar shows 'Microsoft Excel' and 'File Edit View Window Help'. The spreadsheet has a menu bar with 'File', 'Edit', 'View', 'Window', and 'Help'. The spreadsheet content is as follows:

Revenues*	1991	1992	1993
Canada	184,845	203,300	203,791
Mexico	49,292	49,786	49,786
United States	1,232,300	1,355,530	1,219,371
North America	1,466,437	1,608,616	1,493,923
France	184,845	194,087	203,791
Germany	308,075	369,630	373,327
Other European	61,615	92,423	184,848
United Kingdom	61,615	67,777	74,958
Europe	616,150	723,977	838,579
Australia	48,392	53,221	58,584
Japan	439,931	679,862	1,793,724
Korea	43,993	65,990	58,998
Taiwan	65,990	62,488	103,110
Far East	598,306	1,081,571	2,020,373

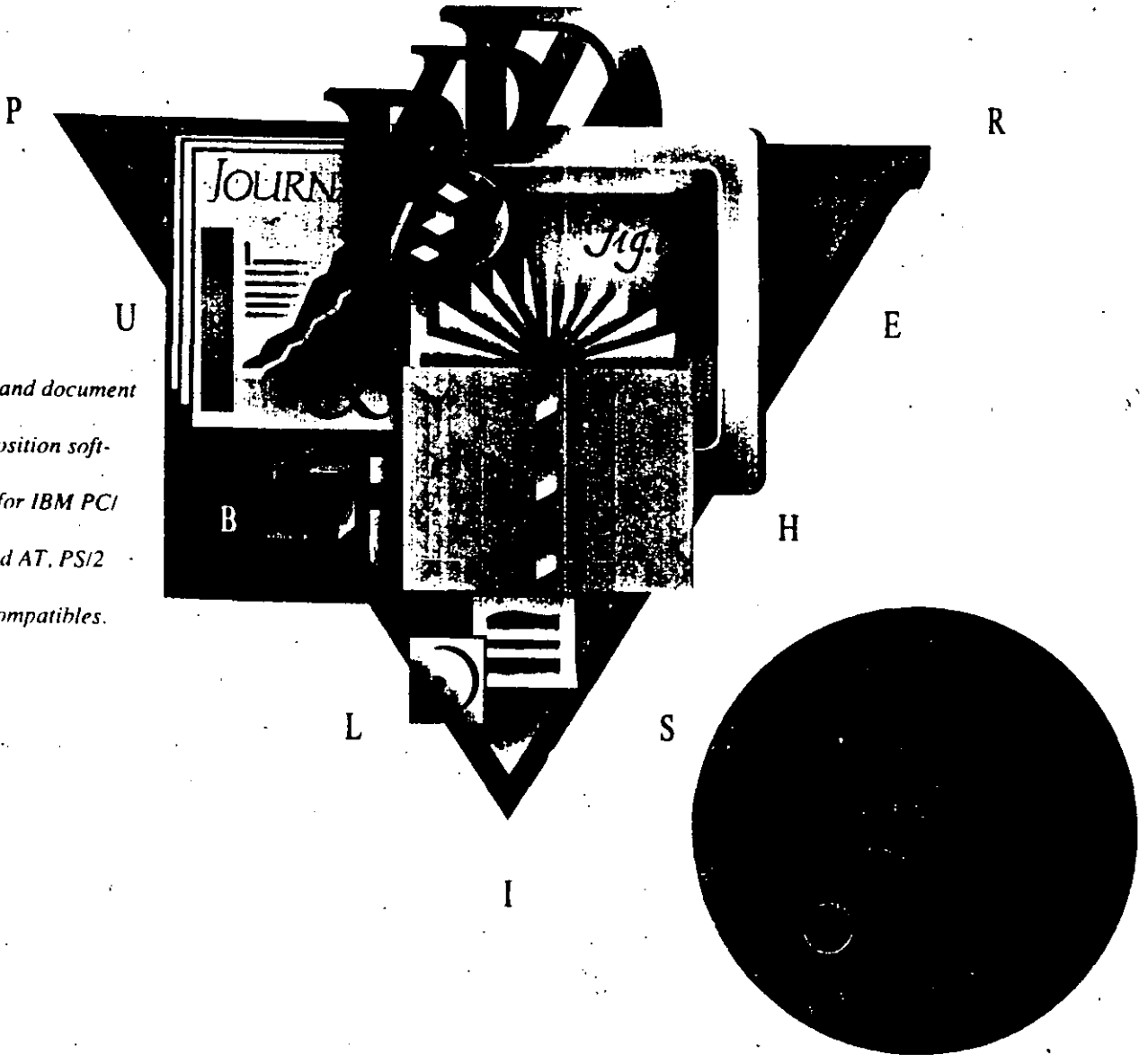
* in thousands

Now with 3-D charting!

Microsoft®

Ventura Publisher

DOS/GEM Edition



Page and document
composition soft-
ware for IBM PC/
XT and AT, PS/2
and compatibles.

Microsoft Word

Word Processing Program

NEW!
Version 5.5
Now easier than
ever to get the
best looking
documents!

Version 5.5 makes it dramatically
easy to access even the most
powerful features—With just a
couple of simple keystrokes!

Microsoft®

WordPerfect™

para IBM® PC y compatibles
y para redes locales



EMERALD GEMS, LTD.

Tokyo • New York • Rio de Janeiro • London

Abstract. ... is trimming its conglomeration by selling ... Beryline Mining Group and Beryline Laboratories. Of the two Beryline industries, the Beryline Mining Group (see table 1.1) is the safest investment because it possesses some of the most productive emerald mines in the world.

Mine/ Site	Emerald Production (Carats)	Value (\$1,000)	Remarks
London Mine (Site A-1)	4.10	12,500	No more than 1000 carats of one
London Mine (Site A-2)	1.00	400	Spangled crystals; many gem quality
London Mine (Site A-3)	1.00	1,500	Very distinctive crystals
London Mine (Site A-4)	1.00	1,000	Clear crystals
London Mine (Site A-5)	1.00	1,000	Clear crystals

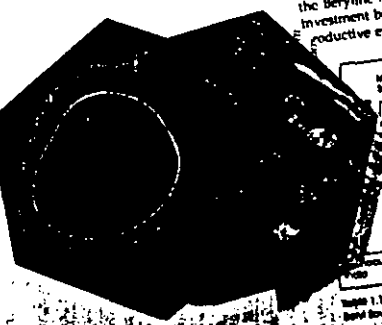
Because of inaccuracies which often occur when measuring the crystallization interval of synthetic emeralds, the emerald's growth rate (B) was also studied.

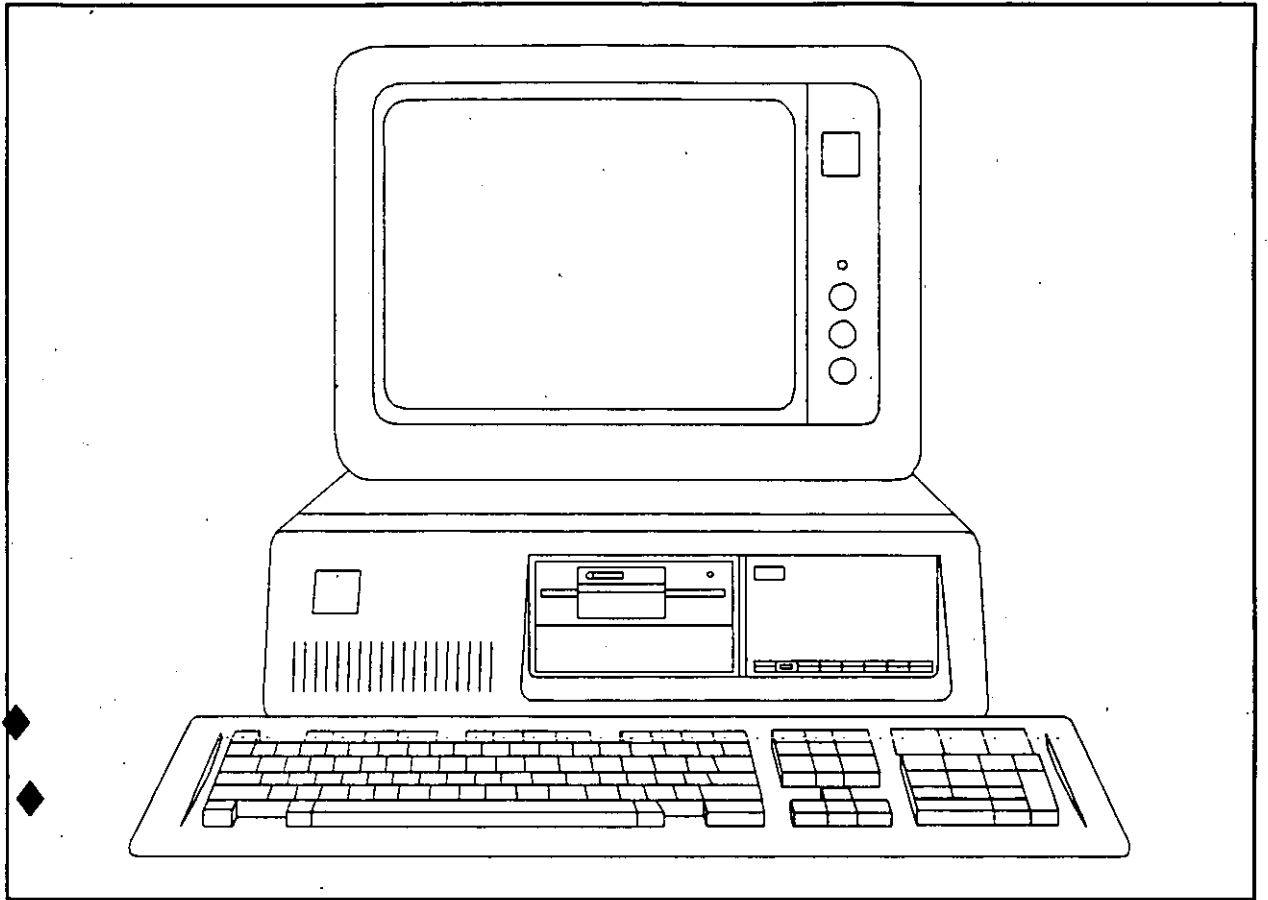
$$\frac{dN_s}{dt} = \frac{dN_c}{dt} \cdot \exp(\alpha t) \cdot \left[\frac{\sum_{i=1}^{m(t)} \sigma_i^{m(t)}(t)}{\sigma_m} \right]$$

B Synthetic emerald growth rate

$$\frac{dR}{dt} = \frac{dR_c}{dt} \cdot \exp(\alpha t)$$

Bullock Gems, Inc.
Wellington House, New Zealand Ave.
Wotton-Thames, Surrey KT18 3JY
UNITED KINGDOM





GLOSARIO

Avance de formulario: Una operación de impresora que hace avanzar el papel de la impresora a la parte superior de una nueva hoja (si el papel está situado de forma correcta). El carácter de control que una computadora envía a una impresora para hacerle realizar un avance del formulario se denomina también "carácter de avance de papel" o "avance del papel".

Baudio: Una unidad de medida de la rapidez con que los datos se transmiten entre una computadora y un periférico. Para convertir una velocidad de transmisión expresada en baudios en un número aproximado de caracteres por segundo habrá que dividirlo por 10. Por ejemplo, un modem de 300 baudios transmitirá y recibirá a una velocidad de 30 caracteres por segundo aproximadamente.

Byte (Octeto): La unidad de almacenamiento (en RAM, ROM o disco) que puede retener un solo carácter de datos. La capacidad de un dispositivo de almacenamiento se suele expresar en bytes. Véase también "RAM", "ROM", "K" y "M".

Cabeza de lectura/escritura: El sensor magnético en una unidad de discos o en una unidad de cinta que escribe información en disco o cinta y la recupera por lectura.

Cable cruzado: Véase "Cable de modem nulo".

Cable de modem nulo: Un cable del RS-232 C en el que algunos pares de conductores están cruzados en su conexión. El usuario debe utilizar un cable de modem nulo para conectar un interfaz en serie del PC a la mayor parte de las impresoras de interfaz en serie.

Cadena: Una secuencia de caracteres. Muchos programas pueden buscar ficheros por las ocurrencias de una cadena de una manera u otra. Ejemplos de cadenas son la orden de búsqueda en EDLIN y la orden FIND en el sistema operativo DOS.

Cadena de ambiente: Una cadena de texto en el entorno de ambiente que contiene un elemento de información para un programa particular.

Camino de búsqueda: Un grupo de uno o más reconocidos llevan a directorios que el sistema operativo DOS ha de buscar cuando el usuario introduce una orden y DOS no puede encontrar dicha orden en el directorio que indica la línea de la orden.

Camino, rama: La secuencia de directorios que van desde un directorio base o raíz de disco, o de un directorio actual, a algún otro directorio o fichero particular. Véase también "Camino de búsqueda".

Carácter de control: Un carácter introducido manteniendo oprimida la tecla CTRL y pulsando una tecla de caracteres. Muchos pro-

gramas de aplicación permiten al usuario introducir caracteres de control para funciones tales como el desplazamiento del cursor y la manipulación de los datos. Los caracteres de control no suelen utilizarse como datos en ficheros de texto y no representan caracteres susceptibles de visualización en el conjunto de caracteres ASCII. Cuando han de representarse por escrito llevan el prefijo de un circunflejo (^) o con la palabra CTRL.

Carácter de nombrefichero global: Un símbolo que puede utilizarse en un nombre de fichero para representar a uno o más caracteres. El sistema operativo DOS identifica dos caracteres de nombrefichero globales: ? representa cualquier carácter y * representa a cualquier número de caracteres.

Carga inicial ("Boot"): Puesta en servicio del sistema operativo de una computadora. El término "boot" procede de la noción de que una computadora "realiza por sí misma su 'bootstraps'" porque una pequeña parte del sistema operativo que se almacena en memoria ROM carga el resto del sistema operativo desde el disco.

Cilindro: La parte de un disco que se puede leer y escribir con las cabezas de lectura/escritura de la unidad de disco en una sola posición. El número de pistas por cilindro depende del número de superficies de grabación que tenga un disco. Un disquete tiene una o dos pistas por cilindro y un disco rígido o duro puede tener un número mayor, puesto que puede estar constituido por una pila de platos de disco en un solo husillo, teniendo cada uno su propio par de cabezas de lectura/escritura.

Cola: Un lugar para mantener una lista de elementos que están esperando que suceda algo. En el sistema operativo DOS 2.0, la orden PRINT pone en cola a los ficheros que han de imprimirse. Mientras se estén imprimiendo los ficheros, el usuario puede ejecutar otras órdenes en su PC. Los ficheros se imprimen en el orden en que están colocados en la cola: cada fichero debe esperar a que se acabe la impresión del fichero precedente. La cola de impresión se suele denominar el "carrete" ("spool").

Compilador: Un programa que lee un programa fuente que ha escrito el usuario y lo traduce al lenguaje máquina de su PC. Dicho programa se traduce finalmente a un fichero COM o EXE y se ejecuta como una orden.

Consola: Está constituida por un teclado y un visualizador. Tomado en conjunto estos dispositivos forman el interface principal entre el usuario y su computadora.

Copia de seguridad o de reserva ("Backup"): Una copia de un fichero o de un disquete que se mantiene en un lugar seguro en caso

de que el original resulte deteriorado o se pierda. Véase también "Reserva completa" y "Reserva parcial".

Correo electrónico: Un servicio que permite al usuario enviar un mensaje a otra persona anunciando el comienzo de la emisión en un servicio de información, indicando el nombre del destinatario y tecleando el mensaje. Cuando el destinatario está dispuesto, el servicio visualiza un mensaje que el correo está esperando y presenta visualmente el texto de cada mensaje a petición.

Corrupción: Lo que sucede a los datos de un disco que se han deteriorado por un error de E/S, un error del usuario u otras anomalías. Los síntomas de datos en corrupción comprenden los datos que desaparecen de un fichero de texto, los datos "extraños" que aparecen espontáneamente en un fichero de texto, un fichero de programa que no se puede ejecutar o un directorio con nombres de ficheros que no tienen sentido o que faltan o ambas cosas a la vez.

CPU: Un chip de circuito integrado que contiene la parte de una computadora que efectúa realmente las operaciones de cálculo. Otros chips en una computadora realizan funciones de apoyo tales como almacenar datos y controlar periféricos. CPU es la abreviatura de "Central Processing unit" (Unidad central de procesamiento). La CPU de su PC es un microprocesador 8088 fabricado por Intel Corporation.

Cursor: Un símbolo en una presentación visual de computadora que marca el lugar en donde aparecerá lo que el usuario teclee a continuación. El cursor de su PC aparece como un subrayado intermitente.

Diagnóstico de encendido: Procedimiento de comprobación de que el PC funciona de forma automática cuando se le aplica la tensión. Estos procedimientos de comprobación sirven para verificar que la totalidad del hardware del PC está funcionando de forma adecuada. Si el diagnóstico detecta anomalías, su PC visualizará códigos de error antes (o en lugar de) la carga inicial del sistema operativo DOS. Los códigos de error pueden ayudar a un técnico de servicio a determinar qué está defectuoso en su computadora personal PC.

Directorio: Una zona en un disco que registra los nombres y las posiciones de los ficheros en el disco. Véase también "Directorio de base", "Subdirectorio" y "Directorio en curso".

Directorio base (Raíz): El primer directorio definido en un disco. Todos los ficheros y subdirectorios en un disco deben almacenarse en el directorio base o en subdirectorios que pueden ser chips de seguimiento de nuevo en el directorio base.

Directorio en curso: El directorio al que se refiere el sistema operativo DOS cuando una especificación de fichero o un nombre de unidad de disco se introduce sin ningún nombre del directorio. En cualquier momento el disco en cada unidad tiene su propio directorio en curso. Cuando se pone en servicio el sistema operativo DOS o se monta un nuevo disco, el directorio de base del disco está constituido por el directorio en curso o directorio corriente.

Directorio con estructura de árbol: El tipo del directorio que el sistema operativo DOS 2.0 utiliza para el control de los ficheros de disco. El directorio base, o directorio raíz, contiene subdirectorios ("ramas") y ficheros ("hojas"), y cada subdirectorio puede contener por sí mismo otros subdirectorios y ficheros.

Directorio matriz o principal: El directorio en el que se define un subdirectorio. Un directorio matriz puede ser el directorio base del disco u otro directorio secundario. Véase también "Directorio con estructura de árbol".

Disco: Un medio utilizado para guardar datos de computadora. Un disco gira a través de una cabeza de lectura/escritura, que registra (escribe) y lee información bajo el control de la computadora. Véase también "Unidad de disco", "Unidad de discos rígidos" y "Disquete".

Disco fijo: El disco rígido y la unidad de discos rígidos suministrados por IBM para el IBM PC y que se instalan en la unidad PC XT.

Disco flexible: Un disco constituido por material plástico flexible que está revestido con una superficie magnética conductiva y protegido mediante una envoltura de plástico cuadrada. Véase también "Disquete".

Disco flexible ("Floppy disk"): Un nombre que se le da a un disco flexible o disquete.

Disco de RAM: Un programa que permite al sistema operativo DOS utilizar parte de la memoria RAM de su PC para almacenar ficheros como si la memoria RAM fuera un disco.

Disco Winchester: El tipo de disco rígido más frecuentemente utilizado en computadoras pequeñas como la computadora personal PC. Está sellado en una caja hermética al polvo y fijado de forma permanente a la unidad de discos.

Dispositivo: Un accesorio de computadora que se utiliza para entrada/salida, tal como una unidad de disco, una impresora o un teclado. Este tipo de dispositivo se denomina "dispositivo de E/S".

Dispositivo E/S: Véase "Dispositivo".

000256

Disquete: Un disco flexible (o "floppy"), de 5¼ pulgadas de diámetro. Los disquetes son un medio de soporte estándar de su PC para almacenamiento de discos y copias de seguridad.

Disquete de distribución: El disquete en el que un proveedor de software vende un producto de programas.

División activa: La división de un disco fijo que el PC carga inicialmente.

DOS: Una abreviatura de "Disk Operating Systems" (Sistema operativo de disco). El término "DOS" se suele referir de manera genérica a muchos sistemas operativos diferentes en computadoras distintas. En las referencias al IBM PC, el término DOS se suele referir al PC DOS, que es el sistema operativo más frecuentemente utilizado por su PC.

Edición de líneas: Una característica funcional del sistema operativo DOS que permite al usuario la edición de líneas con las teclas de función del programa y las teclas INS y DEL. Esta característica funcional puede utilizarse en las órdenes de DOS, en EDLIN y en muchos programas de aplicación.

Editor de texto: Un programa para crear y modificar ficheros de texto.

Elemento de imagen ("Pixel"): El elemento de imagen más pequeño en una imagen de gráficos. Un punto que puede ser luminoso u oscuro. El término "pixel" es una contracción de "Picture element" (elemento de imagen).

En línea: Significa que el dispositivo correspondiente está preparado para su uso inmediato o que está preparado para comunicarse con una computadora. Se utiliza para describir un dispositivo periférico tal como una impresora. Su significado es opuesto al de "Fuera de línea".

Ergonómica: El estudio de cómo las propiedades del cuerpo humano y de la mente afectan al diseño de herramientas y equipos utilizados en un trabajo.

Error E/S: Cualquier error asociado con una operación de entrada/salida. Un ejemplo de un error de E/S es un error de disco, que se produce cuando el PC está tratando de leer o de escribir datos en un disco.

Esclavizado, subordinamiento: La sección para conseguir que un sistema operativo copie su salida de visualización en una impresora se denomina "subordinamiento o esclavizado de la impresora a la presentación visual". En el sistema operativo DOS, ha de subordinarse la impresora a la presentación visual introduciendo CTRL.PRTSC.

3

Espaciamiento proporcional: Una característica de impresora que hace que algunas letras (como *m*) sean más anchas que otras (como *i*). La impresión de espaciado proporcional se parece en su aspecto a la composición tipográfica. Véase también "Microespaciamiento".

Especificación de fichero: Véase "Nombre de fichero".

Especificador de unidad de discos: Véase "Nombre de unidad de disco".

Etiqueta, rótulo: Un nombre dado a un punto en un fichero. En el sistema DOS 2.0, el usuario puede poner una etiqueta, o rótulo, en una línea de un fichero de lotes y utilizar la orden GOTO para hacer que el sistema DOS "vaya a" la etiqueta.

Etiqueta de volumen, rótulo de volumen: Un nombre que identifica un disco. El sistema operativo DOS 2.0 le permite asignar a cualquier disco una etiqueta de volumen de hasta 11 caracteres de longitud.

Extensión de nombrefichero: La segunda parte de un nombre de fichero que describe el tipo de datos existentes en el fichero. En el nombre EJEMPLO.TXT, ".TXT" es la extensión de nombrefichero.

Fecha hora ("Time stamp"): Un registro de la fecha y de la hora en que un fichero fue creado o modificado por última vez. Cada fechador horario de fichero del sistema operativo DOS se mantiene en la entrada de directorio del fichero.

Fichero, archivo: Un conjunto de datos similares que se almacena en un disco y que puede ser objeto de referencia por su nombre.

Fichero ASCII: Un fichero o archivo, que está constituido por datos susceptibles de impresión en la anotación ASCII. Véase también "Fichero de texto".

Fichero AUTOEXEC: Un fichero denominado AUTOEXEC.BAT que contiene órdenes DOS. Cuando se carga inicialmente el sistema operativo de disco DOS, busca un fichero AUTOEXEC en el directorio base del disco DOS. Si encuentra dicho disco, ejecuta el fichero como un fichero de lotes en lugar de solicitarle la introducción de la fecha y de la hora.

Fichero de lotes: Un fichero de órdenes que el sistema operativo DOS puede ejecutar como si el usuario lo hubiera introducido mediante el teclado. Un fichero de lotes debe tener la extensión de nombrefichero BAT. El usuario puede ejecutarlo introduciendo su nombrefichero como si fuera una orden.

Fichero oculto: El fichero que se almacena en un disco pero que no aparece en un listado de directorio del disco. Un disco de DOS

000258

susceptible de carga inicial tiene dos ficheros ocultos, que contienen partes de DOS que se mantienen en memoria RAM siempre que esté funcionando el PC.

Fichero de recubrimiento ("Overlay file"): Un fichero que contiene uno o más recubrimientos.

Fichero de texto: Un fichero que contiene texto tal como una carta, papel para investigaciones o listas de clientes.

Filtro: Un programa que copia datos desde un fichero de texto a otro, realizando algún tipo de cambio sistemático en los datos a medida que los copia. Un ejemplo es un programa que copia de un fichero solamente aquellas líneas de texto que contienen una determinada secuencia de caracteres. El sistema operativo DOS 2.0 tiene tres órdenes de filtro: SORT, FIND y MORE.

Formateado: El proceso de escribir una configuración especial de datos a través de la superficie de grabación de un disco. El usuario debe formatear un disco antes de utilizarlo para almacenar datos. En el sistema operativo DOS, ha de utilizarse la orden FORMAT para realizar el formateado de los discos.

Formularios continuos: Véase "Papel de perforaciones marginales" o "Papel alimentado por rodillo dentado" (papel de perforaciones marginales).

Fragmentación: La división de un fichero de disco en dos o más elementos no contiguos como un efecto secundario de crear y borrar ficheros. Una fragmentación excesiva puede retardar notablemente las operaciones de entrada/salida de discos.

Fuente: El punto desde "donde" se realiza una operación de copia o de desplazamiento. Se suele utilizar en un término combinado tal como "fichero fuente", "disco fuente", etc.

Fuente de alimentación ininterrumpible (UPS): Un dispositivo que protege a una computadora, o a otro dispositivo eléctrico, contra todas las clases de anomalías en la alimentación eléctrica, incluyendo las caídas de tensión. Cuando la tensión en la alimentación de la red cae por debajo de un nivel admisible, la UPS proporciona, de forma automática, una alimentación de corriente alterna generada por una batería. La mayor parte de estas fuentes UPS para computadoras pequeñas suministran alimentación solamente el tiempo suficiente para que el usuario interrumpa su trabajo y desconecte su computadora de una forma adecuada. Véase también "Sistema de alimentación de reserva".

Fuera de línea: Significa que el dispositivo correspondiente no está en condiciones de utilización ni está preparado para comunicarse

con una computadora. Se utiliza para describir un dispositivo periférico tal como una impresora. Es el término opuesto a "En línea".

Gráficos: La técnica de formar una imagen a partir de una configuración de puntos o de líneas en un dispositivo de salida de computadora, tal como una presentación visual o una impresora.

Gráficos de trama: La técnica de construir una imagen a partir de una configuración de líneas de exploración paralelas. Esta técnica se utiliza por los aparatos de televisión y por la mayor parte de las pantallas de visualización de computadoras.

Grupo de usuarios, club de usuarios: Una organización de usuarios de computadoras con intereses comunes. Actualmente existen muchos grupos de usuarios constituidos por propietarios del IBM PC.

Impresora de calidad de correspondencia: Una impresora de matriz de puntos que imprime caracteres constituidos por una excelente configuración de puntos, que proporciona una calidad comparable con la salida de una máquina de escribir provista de una cinta de seda.

Impresora de gran calidad de escritura: Una impresora que forma caracteres golpeando un elemento de tipo en relieve contra el papel a través de una cinta. Las impresoras de gran calidad de escritura proporcionan una salida de alta calidad pero trabajan a unas velocidades pequeñas o moderadas.

Impresora de matriz de puntos: Una clase de impresora cuya cabeza de impresión contiene una hilera vertical de elementos de percusión que inciden en el papel a través de cinta entintada. La impresora forma caracteres impresos golpeando el papel con los elementos de percusión a través de la cinta a medida que la cabeza de impresión se desplaza a través del papel.

Interface: Un conjunto de normas que define la forma en que se pueden comunicar entre sí dos dispositivos de computadora.

Interface Centronics: Un tipo de interface en paralelo utilizado por muchos periféricos y computadoras, incluyendo el PC. Su nombre completo sería "interface compatible con Centronics" puesto que realmente se trata de un producto utilizado inicialmente por la Centronics Corporation. Los términos "interface Centronics" e "interface en paralelo" se suelen utilizar de forma indistinta, aunque un interface Centronics es realmente un tipo específico de interface en paralelo.

Interface en paralelo: Un tipo de interface entre una computadora y un periférico. Es el interface idóneo para conectar una impresora al PC. Véase también "Interface Centronics".

Interface de RGB: Un tipo de interface utilizado entre computadoras y algunos monitores de color. Utiliza un cable independiente para enviar la información de imagen para cada color (rojo, verde y azul). Tiende a proporcionar una imagen de calidad más alta que un interface de video compuesto.

Interface RS-232C: Una clase de interface en serie que se utiliza por muchos periféricos y computadoras, incluyendo la computadora PC. Los términos "interface RS-232C" e "interface en serie" se suelen utilizar de forma intercambiable, aunque un interface RS-232C es realmente un tipo específico de interface en serie.

Interface en serie: Una clase interface entre una computadora y un periférico. En el PC, se suele utilizar para establecer la comunicación con un modem u otra computadora. Muchas impresoras utilizan también interfaces en serie. Véase también "Interface RS-232C".

Interface de video compuesto: Una clase de interface que se utiliza entre una computadora y un monitor y que combina la totalidad de la información de imagen y de color en una sola señal. Véase también "Interface RGB".

Intérprete: Un programa que lee un programa que el usuario ha escrito y que realiza las acciones que proponen las sentencias de dicho programa.

I/O (E/S): Una abreviatura de ("entrada/salida"), que son las dos operaciones básicas que una computadora realiza en sus periféricos.

Joystick: Un dispositivo de entrada con una palanca de mando montada en una junta de articulación universal. El usuario puede indicar el movimiento (que suele ser un movimiento del cursor) desplazando la palanca en cualquier dirección.

K: Una unidad utilizada para medir la capacidad de la memoria de una computadora. 1K equivale a 1.024 bytes.

LAN: Una abreviatura de "local area network" (red de área local). LAN es una combinación de hardware y de software que conecta dos o más computadoras, permitiéndoles intercambiar datos con gran rapidez y compartir periféricos.

Lápiz óptico: Lápiz que se utiliza con un monitor para la entrada de computadora. El PC puede detectar cuándo el lápiz óptico toca la pantalla del monitor a través de un conmutador situado en la punta del lápiz. Si el monitor está visualizando cualquier información en el punto de contacto, el PC puede determinar la posición aproximada del lápiz óptico.

Lenguaje de alto nivel: Un lenguaje de programación en el que el

usuario escribe programas en una notación que es muy similar a su forma natural de expresar el problema que quiere resolver. Véase también "Lenguaje de bajo nivel".

Lenguaje de bajo nivel: Un lenguaje de programación en el que el usuario escribe programas en una notación que es muy similar a los pasos que la computadora debe realizar para ejecutar el programa. Véase también "Lenguaje de alto nivel".

Lenguaje ensamblador: Un lenguaje de programación de bajo nivel. Se trata realmente de una notación cómoda para escribir programas en lenguaje de máquina.

Lenguaje máquina: El único tipo de lenguaje de programación que una computadora puede comprender de forma directa. Un programa escrito en lenguaje máquina está constituido por una cadena de números y resulta muy difícil de comprender para los usuarios. En consecuencia se utilizan compiladores, intérpretes y ensambladores para permitir a los usuarios tratar los programas en una forma más legible.

Lenguaje de programación: Una forma de notación para escribir programas de computadora. Muchos lenguajes de programación diferentes con puntos fuertes y débiles distintos son utilizables por su PC. Véase también "Lenguaje de alto nivel" y "Lenguaje de bajo nivel".

M: Una unidad utilizada para medir la capacidad de la memoria de una computadora. 1M es equivalente a 1.024 K o 1.048.576 bytes.

Memoria: Cualquier elemento de hardware que una computadora utiliza para almacenar datos. Véase también "RAM", "ROM" y "Unidad de discos".

Memoria de acceso aleatorio: Véase "RAM".

Memoria de sólo lectura: Véase "ROM".

Mensaje de solicitud ("Prompt"): Un mensaje procedente de una computadora que invita al usuario a introducir alguna información. El mensaje de solicitud de la línea de orden estándar de DOS está constituido por el nombre del disco por defecto seguido por el carácter indicativo de "mayor que" (>).

Menú: Una lista de lo que un programa puede realizar a continuación y que el programa visualiza para solicitar del usuario sus instrucciones. El usuario debe responder a esta solicitud seleccionando uno de los elementos o artículos del menú.

Método de margen derecho: La técnica de obtener el margen derecho correcto de un documento impreso incluso insertando magnitudes

000259

iguales de espacio entre todas las palabras en una lista. La salida microespaciada se parece en su aspecto a la composición tipográfica. Véase también "Espaciamiento proporcional".

Modem: Un dispositivo para establecer una comunicación entre computadoras a través del sistema telefónico. Un modem actúa convirtiendo datos entre la forma digital utilizada para una computadora y una forma de audio (un "tono de portadora") que puede transmitirse a través de las líneas telefónicas. "Modem" es una contracción de "modulador/demodulador".

Modem de conexión directa: Un tipo de modem que se conecta al sistema telefónico mediante enchufe directo en una toma de pared modular.

Modulador de RF: Un dispositivo que adapta un aparato de televisión para su utilización como una pantalla de visualización de computadora.

Monitor: Un dispositivo similar a un aparato de televisión que una computadora puede utilizar para visualizar palabras o imágenes. Un monitor puede visualizar imágenes monocromáticas o en color. En el IBM PC, debe utilizarse un monitor con un adaptador "IBM Color/Graphics Monitor Adapter" o una tarjeta adaptadora equivalente. Véase también "Visualización monocromática".

Mouse (Ratón): Un dispositivo de entrada que el usuario puede desplazar a través de la superficie de una mesa o un soporte especialmente adecuado. Muchos programas de aplicación permiten al usuario el empleo de un "ratón" para controlar el movimiento del cursor.

Nivel de error: En el sistema operativo DOS 2.0, un valor establecido por una orden exterior para indicar si la orden finalizó de forma normal o con un error. Una orden IF puede comprobar el nivel de error y tomar diferentes cursos de acción dependiendo de si se ejecutó de forma satisfactoria, o no, una orden precedente.

Nodo: Una computadora conectada a una red de área local (LAN), considerada desde el punto de vista de la LAN.

Nombre de dispositivo: Un nombre utilizado por el sistema operativo DOS para identificar un dispositivo periférico como una consola o un interface en serie. Siempre finaliza con dos puntos. Por ejemplo, en la orden copy con: bexample.txt {ENTER}, "con:" es un nombre de dispositivo que se refiere a la consola.

Nombre de fichero: El nombre utilizado para identificar un fichero. Se suele denominar una "especificación de fichero". Es diferente de "nombrefichero", que es la parte de un nombre de fichero que describe el contenido del fichero.

Nombre de unidad de discos: Se le conoce formalmente como un "especificador de unidad de disco" y se trata del nombre de una unidad de discos que se utiliza solo o con un prefijo para la especificación de un fichero. Por ejemplo, en la orden copy b:example.txt a: {ENTER}, "b" y "a" son nombres de unidades de discos.

Nombrefichero: La primera parte de un nombre de fichero, que describe el contenido del fichero. En el nombre EJEMPLO.TXT, "EJEMPLO" es el nombrefichero.

Número de línea: Un número que identifica una línea en un fichero de texto.

Objeto, destino: El punto "a donde" se destina una operación de copia o de desplazamiento. Se suele utilizar en un término combinado tal como "fichero objeto", "disco objeto", etc.

Opción: Una letra precedida por una barra inclinada que proporciona una información suplementaria de orden DOS acerca de lo que se quiere hacer. Por ejemplo, en la orden copy example, txt h:/v {ENTER}, la opción /V indica a la orden COPY que efectúe una lectura, con el fin de verificar que se está escribiendo de forma correcta. Cada orden DOS reconoce su propio conjunto de opciones y las interpreta a su propia manera.

Orden: Una pulsación de teclas o una línea de entrada que introduce el usuario para hacer que un sistema operativo o un programa de aplicación realice su tarea. Por ejemplo, la orden DIR es una orden de sistema operativo DOS que efectúa el listado del contenido de un directorio de disco. Un programa de procesamiento de palabras aceptaría órdenes que hiciera tareas tales como suprimir una palabra, introducir un bloque de texto, o comenzar una nueva página. Véase también "Orden exterior" y "Orden interna".

Orden exterior: Una orden de sistema operativo DOS que se almacena en un fichero de disco con la extensión de nombrefichero COM o EXE y que se carga en memoria RAM cuando ha de ejecutarse.

Orden interna (interior): Una orden que forma parte del sistema operativo DOS y que, por consiguiente, se mantiene en memoria RAM siempre que esté funcionando el sistema operativo DOS. Véase también "Orden exterior".

Papel de avance por rodillo dentado ("Pin-feed"): Papel con una hilera de orificios de rueda dentada a lo largo de cada uno de sus bordes. También se le denomina "formulario continuo". Se suele vender en cajas de varios miles de hojas, con dichas hojas unidas extremo a extremo en una banda plegada continua.

Parámetro: Una palabra en una línea de orden DOS que sigue el nombre de la orden y proporciona a la orden información acerca

de lo que se quiere que realice. Por ejemplo, en la orden `diskcopy a:b {ENTER}`, "a:" y "b:" son parámetros. Véase también "Opción".

Partición, fraccionamiento: Una parte de un disco fijo que está dedicada a almacenamiento de disco para un sistema operativo particular. Cada partición solamente es visible para el sistema operativo para el que está destinada.

PC I: Una versión primitiva de la computadora personal de IBM que puede tener hasta 64K de memoria RAM en la placa del sistema.

PC II: Una versión de la computadora personal de IBM, introducida como una sustitución de la primitiva versión PC I, que puede tener hasta 256K de memoria RAM en la placa del sistema.

PCDOS: El sistema operativo más frecuentemente utilizado en el PC. Véase también "DOS".

PC XT: Una versión de la computadora personal de IBM que puede tener hasta 256K de memoria RAM en la placa del sistema y que tiene un disco fijo incorporado.

Periférico: Un dispositivo que una computadora utiliza para transferir información, en ambos sentidos, entre ella misma y el mundo exterior. Algunos ejemplos de periféricos son una pantalla de visualización, una unidad de discos y una impresora.

Pipe (Tubo): Una conexión directa entre la salida de un programa y la entrada a otro. El sistema operativo DOS 2.0 puede crear uno de estos conductos entre una salida de visualización de programa y una entrada de teclado de otro programa.

Pista: Una zona anular de datos en un disco que pasa a través de una cabeza de lectura/escritura a medida que gira el disco.

Placa del sistema ("System Board"): La placa de circuito impreso grande situada en el interior de la unidad del sistema ("System Unit") con muchos componentes electrónicos montados en ella. Contiene la mayoría de los elementos electrónicos que constituyen el fundamento de las operaciones de la computadora PC.

Plantilla: Una línea de texto que la característica de edición correctora de líneas del sistema operativo DOS utiliza como la base para crear una línea de texto nueva o modificada. Con el empleo de las teclas de función de programa, la tecla `INS` y la tecla `DEL`, el usuario puede copiar caracteres desde la plantilla a la línea nueva, saltar a través de caracteres de plantilla e insertar nuevos caracteres en la línea nueva.

Procesador de palabras, procesador de textos: Un editor de textos concebido especialmente para la corrección de documentos tales

como cartas, memorándums y manuscritos, a diferencia con los ficheros de datos tales como listas de envío por correo y registros financieros.

Programa de aplicación: Cualquier programa de computadora que realiza una tarea tal como la edición correctora de una carta, el cálculo de las nóminas de su compañía o el mantenimiento de las anotaciones en su agenda. Véase también "Sistema operativo".

Programas de diagnóstico: Programas para diagnosticar problemas del hardware en el PC. Están incluidos en un disquete suministrado con el PC. El usuario ejecutará dichos programas cargándoles inicialmente desde el disquete (es decir colocando el disquete de diagnóstico en la unidad A y encendiendo su PC) y realizando elecciones a partir de los menús que el programa le presentará. Véase también "Diagnóstico de encendido".

Protección de copia: La práctica de realizar un disquete no copiable almacenando datos en el mismo de una forma no normalizada. Los proveedores de software suelen proteger contra la copia a sus disquetes de distribución para evitar que un comprador copie los disquetes y los revenda a terceros.

Pulsación de tecla ("Keystroke"): La cantidad de datos que introduce el usuario a través de un teclado al pulsar una tecla.

Punto (.) flotante: Un método de almacenar datos numéricos que permite a una computadora representar valores no enteros tales como 1.25, 3.1415926 y 0.0000001.

RAM: Una abreviatura de "random-access memory" (memoria de acceso aleatorio), que es un tipo de memoria que se utiliza para almacenar un programa de computadora y los datos que el programa somete a tratamiento mientras que se está ejecutando el programa. La memoria RAM se encuentra en la placa del sistema PC y en las tarjetas de adaptadores.

Ranura de expansión del sistema: Un conector para retener una tarjeta adaptadora. La unidad del sistema ("System Unit") y la unidad de expansión del sistema ("System Expansion Unit") tienen ambas unas ranuras de expansión del sistema.

Ranura de protección de la escritura: Una muesca en el borde derecho de la envoltura de plástico de un disquete, que si está cubierta por un trozo de cinta, protegerá el disco contra cualquier tentativa de escritura.

Recubrimiento ("Overlay"): Una parte de un programa de computadora que se almacena en un fichero separado del fichero COM o EXE que contiene la parte principal del programa. El programa car-

000261

000262

ga el "overlay" en memoria RAM solamente cuando se necesita. Los programas grandes pueden reducir sus necesidades de memoria RAM cargando varios "overlays" en la misma parte de RAM, pero en momentos diferentes.

Red de área local: Véase "LAN".

Redirección: Envío de la salida prevista para un destino a otro diferente o la búsqueda de la entrada prevista como proveniente de un origen en otro diferente. El sistema operativo DOS 2.0 permite al usuario redireccionar la salida de visualización y la entrada por el teclado de la mayor parte de los programas para ficheros o dispositivos.

Reserva completa: Una reserva ("backup") de todos los ficheros en un disco. Véase también "Reserva parcial".

Reserva parcial ("Partial backup"): Una reserva de ficheros seleccionados en un disco. Una reserva parcial suele incluir todos los ficheros cambiados a partir de la más reciente reserva completa del disco o todos los ficheros cambiados desde la más reciente reserva de cualquier clase.

Resolución: La más pequeña separación entre puntos en una imagen que un dispositivo puede representar como puntos separados en su salida. Por ejemplo, una impresora de matriz de puntos que imprime 72 puntos por pulgada lineal tiene una resolución de 1/72 pulgadas. La resolución es una medida de cuantos detalles puede imprimir o visualizar un dispositivo de gráficos.

ROM: Una abreviatura de "read-only memory" (memoria de sólo lectura), que es un tipo de memoria que retiene, de forma permanente, los datos almacenados cuando se fabrica la memoria ROM. La computadora puede leer los datos pero no puede cambiarlos. La memoria ROM se utiliza para retener algunas partes del sistema operativo del PC, el intérprete de BASIC y otros elementos fundamentales de software operativo.

Rueda de impresión, disco impresor: El nombre que se suele dar a un tipo de elemento de impresión utilizado por muchas impresoras de gran calidad de escritura. Esta rueda impresora se parece en su aspecto al cubo y los radios de una rueda de vehículo, con un solo carácter imprimible sobresaliendo en el extremo de cada "radio". Esta impresora posiciona los diferentes caracteres enfrente del papel girando la rueda alrededor de su eje. Véase también "Thimble".

Sector: Una unidad física de información grabada en una pista de disco. En el sistema operativo DOS, un sector de disquete tiene una longitud de 512 bytes y cada pista contiene ocho o nueve sectores.

Semigráficos: La técnica de construir una imagen a partir de caracteres, con la frecuente utilización de símbolos especiales tales como cajas y triángulos.

Símbolo de parámetro: Un signo del tanto por ciento (%) seguido por un número, que se utiliza en un fichero de lotes para representar un parámetro. Cuando se ejecuta el fichero de lotes, cada "%1" en el fichero de lotes se sustituye por el primer parámetro de la orden que ejecutó el fichero, mientras que cada "%2" se sustituye por el segundo parámetro y así sucesivamente. Cada "%0" se sustituye por el propio nombre de la orden.

Síncrona: Describe un tipo de transmisión de datos que se utiliza para establecer la comunicación entre una computadora y otra a través de modems. La transmisión sincrona se suele utilizar entre dos computadoras grandes o entre un terminal o computadora pequeña y la clase de computadora grande que muchas sociedades comerciales explotan para su propio uso.

Sistema de alimentación de reserva: Un dispositivo que protege a una computadora o a otro dispositivo eléctrico contra anomalías en la alimentación. Es semejante a una fuente de alimentación ininterrumpible, pero no proporciona protección contra perturbaciones de la alimentación tales como sobrecargas y altas demandas.

Sistema de gestión de base de datos (DBMS): Un programa de aplicación que procesa información organizada como un conjunto de registros, constituido cada uno por campos de datos. Un DBMS puede visualizar e imprimir informes obtenidos a partir de una base de datos y puede realizar varias clases de procesamiento, tales como clasificación de registros, totalización de campos a partir de grupos de registros y selección de registros sobre la base de los valores de los campos.

Sistema operativo: Un programa de computadora que controla los detalles técnicos del funcionamiento de una computadora, dejando libres a los programas de aplicación para realizar una tarea de utilidad. Ejemplos de tareas de sistemas operativos son la lectura de programas de aplicación desde un disco, la captación de las pulsaciones del teclado y la presentación de la salida de un programa de aplicación en el dispositivo de visualización.

Sistema operativo de disco: Véase "DOS".

Sistema de tiempo compartido: Una computadora grande que puede ser compartida por muchos usuarios que realizan tareas no afines. Muchos sistemas de tiempo compartido se explotan como servicios comerciales.

Software: La totalidad de los programas de computadora que hacen

que ésta realice tareas de utilidad. Asimismo, el conjunto de programas de computadoras que hacen que la misma realice una tarea de utilidad particular.

"Spooling" (Operación simultánea de periféricos en línea): El proceso de enviar ficheros de texto a una cola, desde la cual puede ser objeto de impresión, con posterioridad. Véase también "Cola". El término "spool" es una abreviatura de "system peripheral output, on-line" (salida en línea de periféricos del sistema).

Subdirectorio: Un directorio que se almacena en otro directorio como si fuera un fichero. El sistema operativo DOS 2.0 permite al usuario definir muchos niveles de subdirectorios en un disco.

Tarjeta de adaptador: Una tarjeta de circuito impreso que proporciona a su PC más memoria, le permite controlar un nuevo dispositivo o darle alguna otra capacidad nueva. Debe enchufarla en una de las ranuras de expansión del sistema en la unidad del sistema, o en la unidad de expansión del sistema.

Tarjeta de co-procesador: Una tarjeta de adaptador que contiene una CPU. Puede permitir a su PC ejecutar programas concebidos para la utilización de una CPU distinta a la del microprocesador 8088 del Intel incorporado en el PC.

Tarjeta de RAM: Una tarjeta de adaptador que amplía la capacidad de memoria RAM de su PC.

Tarjeta de opción: Véase "Tarjeta de adaptador".

Tarjeta multifunción: Una tarjeta adaptadora que proporciona varias funciones y que le permite añadir más capacidades funcionales a su PC sin agotar las ranuras de expansión del sistema.

Tecla ALT: Una tecla del teclado del PC etiquetada "Alt". Se pronuncia "alt" (una sola sílaba) o "alternate". Algunos programas de aplicación le permiten realizar funciones de control manteniendo oprimida la tecla ALT y pulsando una tecla normal. Observe que la tecla ALT no es equivalente a la tecla CTRL o a la tecla ESC.

Tecla de caracteres: Cualquier tecla en el teclado del PC que, cuando se pulsa, introduce un carácter.

Tecla de control: Véase "Tecla CTRL".

Tecla CTRL: Una tecla en el teclado de su PC etiquetada "Ctrl". Se pronuncia "control". Muchos programas de aplicación permiten al usuario introducir órdenes de programas manteniendo oprimida la tecla CTRL y pulsando una tecla normal.

Tecla ENTER: La tecla situada en el teclado de su computadora personal PC está etiquetada ". Se le denomina también "tecla de re-

torno", puesto que su función se parece mucho a la de la tecla de retorno del carro de una máquina de escribir.

Tecla ESC: Una tecla en el teclado de su PC etiquetada "Esc". Es la abreviatura de "escape". Muchos programas de aplicación permiten al usuario utilizar esta tecla para escapar de una condición de error o para cambiar el programa desde un modo operativo a otro.

Tecla de retorno: Véase "Tecla ENTER".

Teclado numérico: Un grupo de teclas situadas cerca del lado derecho del teclado del PC. Estas teclas tienen dos funciones: pueden utilizarse para introducir números como el teclado de una calculadora o bien se pueden emplear para realizar funciones de programa tales como posicionamiento del cursor. En este último caso, se suelen denominar "teclas de control del cursor". Para conmutar las teclas entre estas dos funciones hay que pulsar la tecla NUM LOCK, que está situada inmediatamente por encima del teclado numérico.

Teclas de control del cursor: Véase "Teclado numérico".

Teclas de función de programas (PFK): Se trata de diez teclas situadas en el extremo izquierdo del teclado del PC. Muchos programas de aplicación permiten al usuario emplear estas teclas de función para realizar operaciones de control. Por ejemplo, un procesador de palabras podría permitir su utilización para desplazar el cursor, introducir o borrar texto y buscar el texto correspondiente a determinadas palabras o frases.

Thimble (Dedal): El nombre que se suele dar a una clase de elemento impresor utilizado por algunas impresoras de gran calidad de escritura. Los caracteres susceptibles de impresión con este nombre sobresalen en relieve en brazos que salen del borde de un disco de plástico plano, de modo que el elemento de impresión tiene el aspecto de un "dedal", con los brazos constituyendo los lados del dedal. La impresora posiciona caracteres diferentes enfrente del papel girando el elemento impreso alrededor de su eje. Véase también "Rueda de impresión".

Tono de portadora: Una señal de audio generada por un modo. El modem transmite datos modulando el tono, de forma muy similar a una emisora de radio que transmite sonidos modulando una señal radioeléctrica.

Tractor: Un mecanismo que ejerce una tracción sobre papel con perforaciones marginales a través de una impresora. Engrana los orificios de rueda dentada del papel con un par de correas o ruedas dentadas. Un tractor puede estar incorporado en una impresora o añadirse cuando sea necesario.

000263

Trazador ("Plotter"): Un dispositivo de salida que dibuja gráficos desplazando un lápiz sobre una lámina de papel bajo control de computadora.

Unidad de discos ("disk drive"): Un dispositivo que lee y escribe datos en un disco magnético, de forma muy similar a como una grabadora de cinta lee y escribe grabaciones sonoras en cinta magnética. Véase también "Unidad de disquetes" y "Unidad de discos rígidos".

Unidad de discos por defecto u omisión: La unidad de disco que un sistema operativo DOS o una orden utiliza para leer o escribir un fichero cuando se ha omitido el nombre de la unidad en una especificación de fichero. El nombre de la unidad por defecto corriente se visualiza en el mensaje de solicitud de la línea de órdenes del sistema operativo DOS estándar.

Unidad de discos rígidos: Un tipo de unidad de discos que utiliza un disco rígido (duro). El disco duro suele estar permanentemente sellado en la unidad de discos para protegerle contra la contaminación del polvo. Las unidades de discos rígidos tienden a tener una mayor capacidad, velocidad y fiabilidad que las unidades de discos flexibles y también suelen ser más caros.

Unidad de disquetes: Un tipo de unidad de disco que utiliza un disquete.

Unidad de expansión: Véase "Unidad de expansión del sistema".

Unidad de expansión del sistema: Un componente que se une a la unidad del sistema ("System Unit") a través de un cable de transmisión de datos. Proporciona ranuras para tarjetas adaptadoras adicionales y espacio para unidades de discos suplementarias.

Unidad del sistema ("System Unit"): El componente central de IBM PC. Contiene una o más unidades de discos, una fuente de alimentación, la placa del sistema y un número variable de tarjetas adaptadoras.

UPS: Véase "Fuente de alimentación ininterrumpible".

Utilidad de información: Un negocio que utiliza una computadora grande para ofrecer información y servicios basados en la información al público.

Valor por defecto o por omisión: Un valor (por ejemplo, correspondiente a un parámetro) que el sistema operativo DOS o una orden supondrá cuando no se da ningún valor.

Verificación: El proceso de lectura hacia atrás de datos de discos después de su escritura para cerciorarse de que se escribió de forma co-

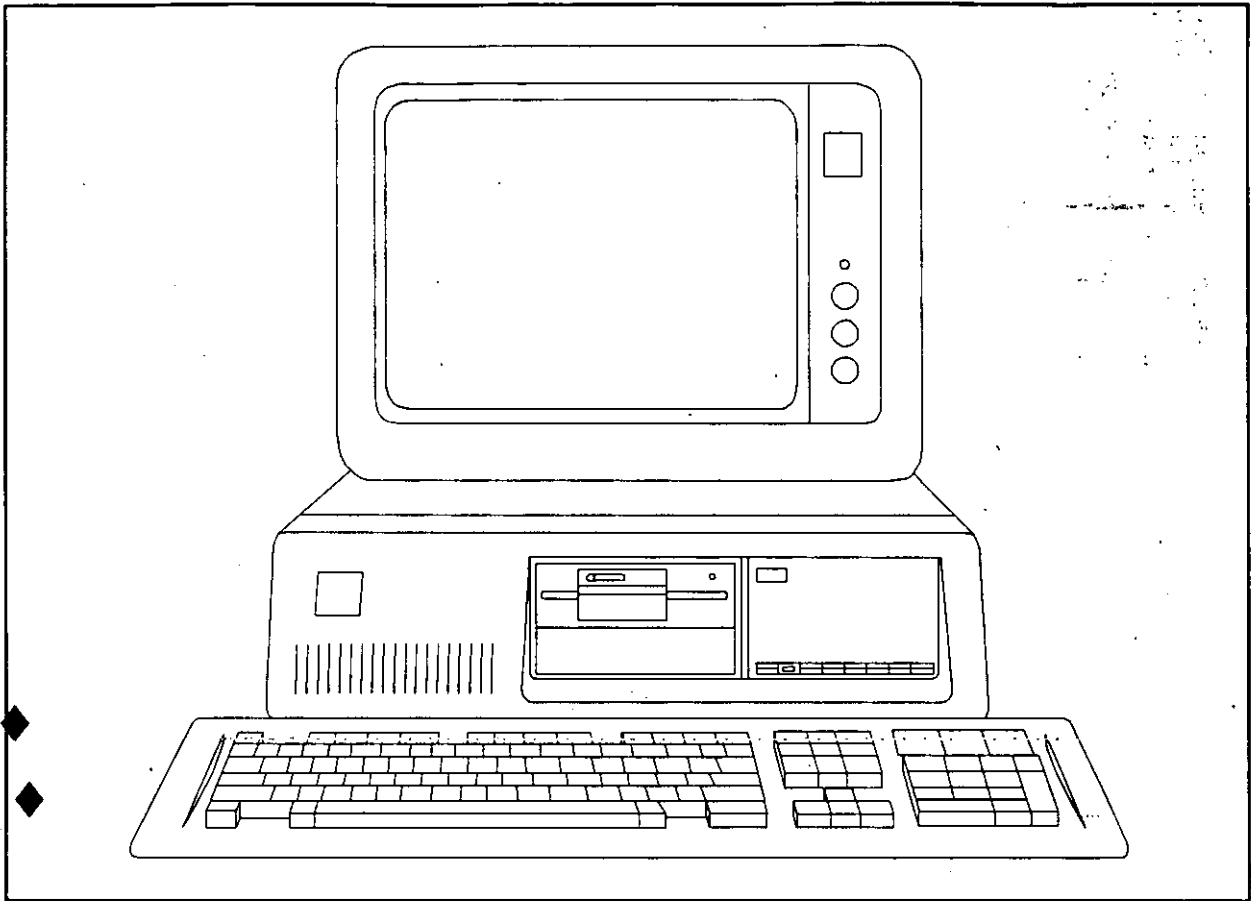
recta. Para hacer que la orden COPY verifique su salida, utilice la opción /V. Para hacer que el sistema operativo DOS 2.0 verifique todas las salidas de disco, ha de ejecutar la orden VERIFY.

Visualización monocromática: Un dispositivo de presentación visual similar a un monitor, que debe utilizarse con un adaptador de visualización monocromática de IBM o una tarjeta adaptadora equivalente. En esta visualización monocromática se presenta una imagen de color verde sobre fondo negro con mejor definición y menos fluctuación que en la mayor parte de los monitores.

000264

(19)

(10)



BIBLIOGRAFIA

TEMAS DE ESTE CURSO
TRATADOS

- SISTEMA OPERATIVO MS - DOS 1, 2, 3, 4, 6
 GUIA DE USUARIO
 INCLUYE IBM PC - DOS
 PAUL HOFFMAN - TAMARA NICOLOFF
 OSBORNE, MC. GRAW - HILL 1985

- EL IBM - PC 1, 2, 3, 4, 6
 INCLUYE IBM - PC (DOS 2.0) Y PC - XT
 JONATHAN SACHS
 OSBORNE / MC GRAW - HILL 1985

- BYTE 2, 3, 4, 6
 GUIDE TO THE IBM - PC
 FALL 1984

- UPWARD MIGRATION PART II 2, 3, 4
 A COMPARISON OF CP/M - 86
 AND MS - DOS
 TAYLOR, ROGERS
 BYTE
 JULY, 1982 PAG. 330

- 8086/8088 16-BIT MICROPROCESSOR 2, 4
 PRIME BYTE BOOKS
 MORGAN, CHRISTOPHER
 BYTE, 1982

- 8086/8088 16 - BIT MICROPROCESSOR PRIMER 2, 4, 6
 MORGAN, CHRISTOPHER
 PETERBOROUGH BYTE BOOKS, 1982

- THE IBM - PC USER'S GUIDE 3, 4
 THE MC. HILLAN EASY HOME
 COMPUTER SERIES 1985

- PUBLICACIONES IBM 3, 4
 - . IBM PERSONAL COMPUTER BASIC MANUAL
 PN 6025010
 - . IBM PERSONAL COMPUTER DISK
 OPERATING SYSTEM
 PN 6024001
 - . IBM PERSONAL COMPUTER GUIDE
 TO OPERATIONS
 PN 6025117
 - . IBM PERSONAL COMPUTER TECHNICAL
 REFERENCE MANUAL
 PN 6025005

INTRODUCCION A LAS COMPUTADORAS PERSONALES

BIBLIOGRAFIA

TEMAS DE ESTE CURSO
TRATADOS

- AN INTRODUCTION TO MICROCOMPUTERS
VOLUME I, OSBORNE, ADAM
OSBORNE/MC. GRAW - HILL 1977 1, 2
- COMPUTER DICTIONARY
THE GUIDE TO COMPUTING TERMS
AND CONCEPTS
THE POCKET PROFESSOR
COLLEGE LANE PUBLISHERS
1984 CANADIAN CATALOGUING PUBLICATIONS 1, 2
- THE SILICON JUNGLE
DAVID H. ROTHMAN
BALLANTINE BOOKS - NEW YORK
JULY, 1985 1, 2
- PROGRAMANDO EN FORTRAN
PROGRAMACION ESTRUCTURADA CON
FORTRAN IV Y FORTRAN 77
VLADIMIR ZWASS
CECSA, 1985
CAPITULOS 1, 2, 4, 7, 8 1, 2, 5
- INTRODUCCION A LA COMPUTACION Y A LA
PROGRAMACION ESTRUCTURADA
G. LEVINE
MC. GRAW - HILL, 1984 1, 2, 5
- INTRODUCCION AL IBM - PC
VERSION DOS 2.0 Y BASIC 2.0
COMUNES AL PC Y AL XT
WALTER SIKONOWIZ
MC. GRAW - HILL 1984 1, 2, 3, 4, 6

TEMAS DE ESTE CURSO
TRATADOS

- PUBLICACIONES MULTITECH INDUSTRIAL CORP. 3, 4
 - . MPF - PC USER'S GUIDE (MS - DOS VERSION) 1985
 - . MICROSOFT MS - DOS OPERATING SYSTEM USER'S GUIDE 1983 MICROSOFT CORPORATION

- HOME COMPUTER SOFTWARE GUIDE STEVE DITLEA OSBORNE / MC GRAW - HILL 1984 2, 5, 6

- BYTE JUNE 1985, VOL 10 No. 6 PROGRAMMING TECHNIQUES 5, 6
 - . CHOOSING A PROGRAMMING LANGUAGE
 - . STRUCTURING BASIC
 - . USING DATA FLOW FOR APPLICATION DEVELOPMENT
 - . DEBUGGING TECHNIQUES

- DICCIONARIO DE TERMINOS BASIC IBM - PC E. ADAMS MC. GRAW - HILL 1984 2, 6

- SOFTWARE ENGINEERING A PRACTITIONER'S APPROACH ROGER S. PRESSMAN MC. GRAW - HILL 1982 5, 6

- SOFTWARE ENGINEERING MARTIN L. SHOOMAN MC. GRAW - HILL 1983 5, 6

- STRUCTURED DESIGN FUNDAMENTALS OF A DISCIPLINE OF COMPUTER PROGRAM AND SYSTEM DESIGN YOURDON, CONSTANTINE PRENTICE HALL, 1979 5, 6

- STRUCTURED ANALYSIS/DESIGN WORKSHOP 5, 6
YOURDON
YOURDON INC, 1981
- SISTEMA DE INFORMACION DE LA SOCIEDAD 2, 5, 6
DE EX-ALUMNOS DE LA FACULTAD DE
INGENIERIA
TESIS PROFESIONAL
ARRONA, MUÑIZ, PEREZ Y PIERCE
U.N.A.M.
- THE ART OF SOFTWARE TESTING 5, 6
GLENFORD J. MYERS
JOHN WILEY & SONS, INC
1979
- STRUCTURED PROGRAMMING A FIRST COURSE 5, 6
FOR STUDENTS AND HOBBYIST
PETER E. GOSLING
MC GRAW-HILL, 1983
- LA AUTOMATIZACION DE LA ADMINISTRACION 2, 6
PUBLICA FEDERAL USANDO MICROCOMPUTADORAS
HECTOR ARRONA
BOLETIN DE POLITICA INFORMATICA
INSTITUTO NACIONAL DE ESTADISTICA,
GEOGRAFIA E INFORMATICA
ENERO 1985, NUM. 1 AÑO VIII
- SERIOUS PROGRAMMING 2, 3, 4, 5, 6
FOR THE IBM PC/XT/AT
HENRY SIMPSON
THE COMPUTER BOOK CLUB, 1986.
- 1001 THING TO DO WITH 2, 3, 6
YOUR PERSONAL COMPUTER
MARK SAWUSCH
THE COMPUTER BOOK CLUB, 1985.
- START POWER MASTERING 2, 6
SOFTWARE
PAUL GARRISON
THE COMPUTER BOOK CLUB, 1985
- NETWORKING WITH THE IBM NETWORK AND CLUSTER
MICHAEL HURWICZ
THE COMPUTER BOOK CLUB, 1986
- ENCYCLOPEDIA OF COMPUTER SCIENCE AND ENGINEERING Todos
ANTHONY RALSTON
EDWIND REJLLY
SECOND EDITION 1985
THE LIBRARY OF COMPUTER AND INFORMATION SCIENCES

REVISTAS

- BYTE
- PC MAGAZINE
- PC TIPS
- PC WORLD
- MAC USER
- I-EEE
- INFO WORLD

800010

DATE 1-18-81

100010

100010

100010

100010

100010

100010

100010

100010

100010

100010

100010

100010

100010

100010

100010

100010

100010

100010

100010

100010

100010

100010

100010

100010

100010

100010

100010

100010

100010

100010

100010