

MICROBYTE

Vol. III N° 8

TODO COMPUTACION Y TELECOMUNICACIONES

DICIEMBRE 1986
N° 30 \$ 280



Redes Locales

Nuevo Apple II GS

Códigos de Barra

Técnicas de ordenamiento

ARMOR

INDUSTRIAL TERMOFIL S.A.



ARMOR- LA MARCA PARA TODAS LAS MARCAS

Con la inauguración reciente de su planta, Industrial Termofil S.A. -fabricante de productos ARMOR en Chile- consolida el máximo respaldo en términos de calidad, confiabilidad y garantía de sus artículos.

Cintas para impresoras y máquinas de escribir:	Margaritas para:
Panasonic KXP 1090	Olivetti
NCR 2140	Olympia
Lorenz 3000	Diablo
Adler 1010	

Más de 500 modelos de cintas y margaritas para impresoras y máquinas de escribir electrónicas, y el mejor servicio.

ARMOR - TECNOLOGIA FRANCESA EN LA PRODUCCION NACIONAL

Carmen 1985 - Santiago. Teléfonos: 5558324 - 515696 - 518365



Distribuidores de ARMOR en sus diferentes áreas:

Abatte - Chiloé 1811 - T. 5552588 / Cammelli y Cia. - Bandera 236 Local 5 - T. 6985627 / Guillermo von Chrismar y Cia. - Bandera 98 - T. 6980052, y Mac-Iver 101 - T. 384306 / Librería La Mercantil Agustinas 1121 - T. 6961019 / Librería Pachin Ltda. - Lira 709-A - T. 2222702 / Librería Rey-Ser - Merced 655 / Olimáquinas - Providencia 1649 - 2286 - T. 740068 y Nataniel 651 - T. 723475.

Asicom - Mac-Iver 115 - T. 383621 y Las Violetas 2099 - T. 43014 / Cienteq - Antonio Varas 754 - T. 743508 / Computerland - La Concepción 80 Local 2 - T. 740085 - Disas-Pío Nono 109 / Engatel - Ahumada 6 Of. 95 - T. 723507 / ST Computación - Génova 2086 - T. 2514571 / Suma - Edo. Castillo Velasco 532 - T. 490706 / Teorema - Agustinas 1169 - T. 721367 y Parque Arauco Local 247-A - T. 2420596 / Xerox - Alcántara 30 - T. 2460306.

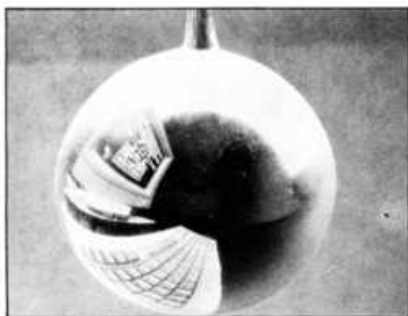


Foto Portada

Coordinador General
 José Kallman T.
 Director Publicidad y RR.PP.
 Ariel Leporatti P.
 Ventas
 Orlando Zepeda
 Directora de Arte
 Paz Barba
 Montaje
 Rodolfo Hillmer
 Cuerpo Editorial
 Jaime Aravena
 Guillermo Beuchat
 Carlos Contreras
 Héctor Miranda
 Humberto Silun
 Corresponsales en el exterior
 Luis Kallman T. (Londres)
 Alfredo Zarowsky (Paris)
 Victor Kahan (Ohio)
 Fotocomposición
 LASER
 Distribución
 Antártica S.A.
 Impresión

Tintazul Publicidad

Microbyte es una publicación mensual de KVC Asociados. Ninguna parte de esta revista puede ser reproducida, archivada en sistemas de clasificación o recuperación de datos, transmitida en modo alguno, electrónico o químico, mecánico, óptico, fotográfico o cualquier otro sin el permiso previo de KVC Asociados.

Microbyte no puede asumir ninguna responsabilidad por errores en artículos, programas o avisos publicitarios. Las opiniones expresadas en estas páginas corresponden a sus autores y no representan necesariamente el pensamiento de sus editores. Colaboraciones de los lectores son bienvenidas y serán publicadas previa revisión, con un pago de acuerdo a tipo de colaboración y calidad. Las colaboraciones deben venir tipeadas o impresas a doble espacio, y, si es posible, acompañadas de material gráfico. En el caso de listados de programas mayores de 15 líneas, es preferible enviar cassette o disco y una explicación de su contenido.

SUBSCRIPCIONES

Valor subscripciones anual (12 Ejs.)

Correo Certif. Stgo. y Prov. \$ 3.000

Entrega por mano Stgo. \$ 2.800

Solicite un representante al fono

2231530 en Huelén 164 2º Piso

Providencia - Santiago.

MICROBYTE

INDICE

Vol. III N° 8

3

Editorial:

A pesar de su escasez relativa, los profesionales informáticos quedan rápidamente obsoletos.

4

Noticias Novedades

18

Apple II GS:

Nuevo modelo fusiona al Apple II con el Macintosh.



23

Sorts:

Presentamos la versión completa de esta "biblia" de los métodos de ordenamiento.

48

Códigos de Barra:

Una tecnología que se impone.



52

Prototipos:

Tercera parte: el cuándo, el cómo, el cuánto y el con quién.



60

Redes Locales:

Un profesional argentino describe los fundamentos básicos de las redes de área local.



PRINTRONIX®

DEJAR UNA BUENA IMPRESION ES IMPORTANTE

Si en la vida cotidiana dejar una buena impresión es importante, en computación lo es aún más puesto que por su sofisticada tecnología y su multiplicidad de uso, se requiere imprimir con gran calidad, nitidez y exactitud en el trazo, textos, gráficos o códigos de barra.

Por eso RIMPEXCHILE ha traído al país las afamadas impresoras norteamericanas PRINTRONIX, compatibles con cualquier computador, capaces de reproducir con una insuperable calidad los más exi-

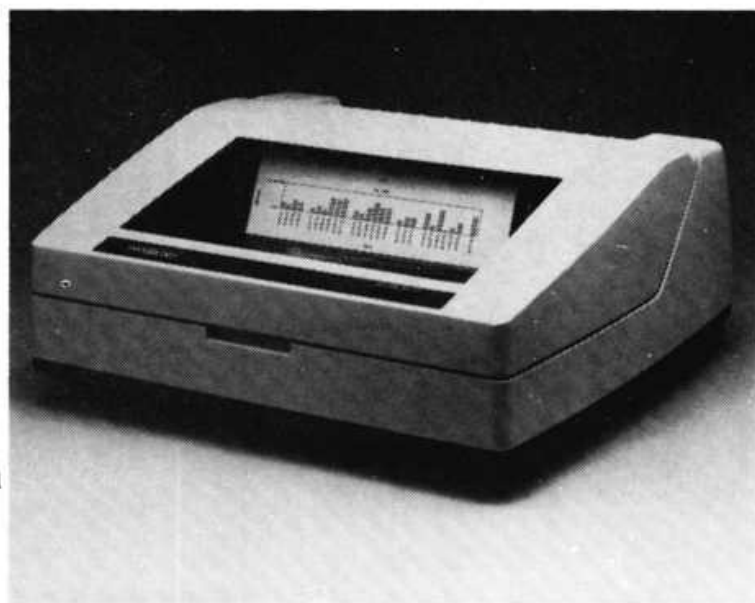
gentes requerimientos en forma rápida, eficiente y lo que es muy

importante, en forma económica. PRINTRONIX, imprime e impresiona... maravillosamente.

PINTRONIX ES EN EL PAIS RIMPEXCHILE

Características técnicas:

- Alta velocidad y calidad de impresión. Desde 47 LPM hasta 2.000 LPM.



- Múltiples modos de impresión.
- Trabajo continuo.
- Capacidad gráfica: códigos de barra, logos, formularios, etc.
- Operación silenciosa: gabinete acústico en modelos de alta velocidad.
- Compatible con todos los sistemas de computación existentes en el mercado.
- Durabilidad y economía por su alta tecnología de impresión.
- Modelos: P-300, P-600, P-1013, S7024, MVP.

EN COMPUTACION
RIMPEXCHILE
La elección de expertos

EDITORIAL

Una de las paradojas de la profesión informática es que a pesar de la continua escasez de especialistas, se produce permanentemente un fenómeno de obsolescencia prematura.

Cada año egresan de las universidades e instituciones de educación superior una gran cantidad de profesionales. Sin embargo, al cabo de un breve período, un número importante de éstos no están en condiciones de postular a nuevos empleos pues carecen de la formación necesaria para administrar las herramientas informáticas más modernas.

Naturalmente, uno de los factores que inciden en esta situación es el desarrollo vertiginoso de nuevas tecnologías que hace difícil una permanente actualización. Sin embargo, ésta es sólo una de las razones. Existe otra, de mucho peso también y es que el mercado ocupacional influye en el hecho de que los cerebros creativos y mejor preparados se transforman rápidamente en meros mantenedores de sistemas.

Nuestros profesionales dominan un arte, lo aplican y son finalmente absorbidos por su primera gran obra maestra, su primer sistema de peso. Su presencia en las empresas deja de ser valiosa por su capacidad creadora sino por su conocimiento a fondo del sistema que viene de crear. En el futuro sólo deberá mantenerlo y para eso no se requiere mantenerse actualizado con los avances de la ciencia computacional.

El problema surge cuando las empresas que si son entes dinámicos en permanente evolución descubren que sus sistemas están quedando obsoletos y corresponde su puesta al día. Obviamente, el profesional queda obsoleto junto con su sistema a menos que haya existido una planificación previa que le permite mantenerse actualizado en su formación.

Empresas como IBM mantienen dentro de sus esquemas una permanente capacitación de sus empleados y estamos seguros que si para IBM es una política rentable, lo es también para la gran mayoría de las empresas. Nuestros profesionales son un recurso caro que no debemos dilapidar.

No podemos terminar esta página sin dejar de agradecer a todos quienes nos han acompañado durante este año y desearles felices fiestas y un mejor próximo año.

A handwritten signature in black ink, reading "Hoffman." with a horizontal line underneath.

NOTICIAS

NOVEDADES

Microsoft rompe con Ascii en Japón

Luego de ocho años de fructíferas relaciones, Microsoft Corp., la empresa que se ha hecho multimillonaria licenciando el sistema operativo MS-DOS a IBM y todos los fabricantes de compatibles, rompió sus lazos con Ascii, la empresa japonesa que participó en el crecimiento de Microsoft abriéndole el camino al mercado japonés.

La relación comercial entre ambas empresas estaba basada en la amistad personal entre sus dos fundadores, Allan Gates y Kazuhiko Nishi, dos personajes de características asombrosamente similares para dos personas de continentes y culturas tan alejadas.

En efecto, tanto Gates como Nishi tienen la misma edad (30 años) y ambos abandonaron sus estudios formales en la universidad, para dedicarse al

tema que los apasionaba: los primeros microcomputadores y el potencial crecimiento de ese sector.

Entre otras cosas, Ascii con Microsoft son los creadores de la norma MSX sobre la que diversos fabricantes japoneses y europeos han comenzado a fabricar equipos que son compatibles en software y periféricos, un sistema que aún no llega a nuestro país.



Epson introdujo un sucesor a su impresora de matriz de punto LQ-1500

Epson introdujo un sucesor a su impresora de matriz de puntos LQ-1500.

La nueva LQ-2500 viene con una pantalla de cristal líquido y teclas de control programables que le permite almacenar hasta cuatro configuraciones de impresoras distintas.

Estas configuraciones son almacenadas en EEPROM, lo que significa que no se borran cuando se apaga la impresora. La cabeza de impresión es de 24 agujas y posee un buffer de 8K expandible a 40K.

Nueve Usuarios para un AT

Specialix Systems Ltd. desarrolló una tarjeta, la AT-8, que permite conectar hasta 8 terminales a un procesador PC/AT, lo que sumado a la propia consola del AT da como resultado nueve usuarios simultáneos.

La tarjeta, que va inserta en una de las ranuras de expansión del AT, se conecta a una caja exterior que contiene 8 puertos RS 232C.

La AT-8 contiene su propio procesador 80186, que se encarga de distribuir la información a los terminales, liberando de esta tarea al 80286 del AT, el que puede dedicarse a sus propias funciones. La tarjeta contiene rutinas para trabajar en Xenix y DOS Concurrente.

Robot Parlante

SYLKON es el primer robot de vigilancia que habla. Construido por una sociedad francesa, está dotado de un módulo electrónico de sintetización y un transmisor telefónico que le permite dar discretamente la voz de alerta a distancia.

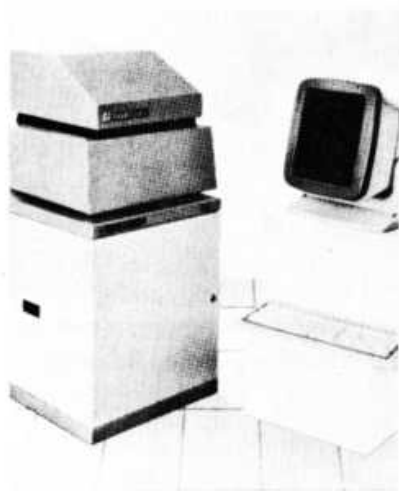
Dada su concepción modular, el SYLKON puede ser adaptado a distintas misiones: detectar ingresos no autorizados, manejo de instalaciones domésticas o industriales, etc.

Si el aparato detecta una anomalía, llama por teléfono e informa el tipo de situación que está ocurriendo. Puede memorizar tres números telefónicos y tiene un vocabulario programable en diferentes idiomas. La vigilancia la realiza sea mediante sensores conectados a la tarjeta principal o a través de un radar volumétrico y un receptor de ultrasonido. También es posible que un humano llame a SYLKON para interrogarlo respecto a sus actividades de vigilancia.

Mayores informaciones en SYLKON S.A.R.L. - 6, rue Maraldi - 06300 - Nice.



Sistema de archivo



Se ha creado en el Reino Unido un avanzado sistema videomicrográfico que hace posible un registro de documentación "sin papel", con la consiguiente y enorme economía en espacio y acceso de documentos, junto con otras e importantes ventajas.

El sistema "Retriever" ha sido diseñado para integrarse a nuevos o ya existentes sistemas de ordenadores, para proporcionar una notable capacidad de manipulación de grandes volúmenes de documentos. Cada Retriever Image Server, puede contener hasta un millón de páginas microfilmadas tamaño A4, equivalentes a 77 archivadores de oficina o 500 Gbytes de almacenamiento en discos. Hasta un total de 15 unidades pueden ser enlazadas y cualquiera de 15 millones de documentos puede ser examinado en un

tiempo típico de 6 segundos a distancias de hasta 2 km por varios usuarios simultáneamente, caso necesario.

Los Image Servers almacenan documentos en micropelícula, con 2000 páginas tamaño A4 en una tira de película de 35 mm de 200 mm de longitud. Cuando se necesita, selecciona el documento correspondiente y lo explora electrónicamente mediante un avanzado fotosensor CCD. La imagen electrónica es tratada digitalmente para garantizar su claridad, siendo después transmitida a 10 Mbaudios.

Los documentos explorados se presentan en Retriever Workstations en su tamaño completo A4 y con muy alta resolución, NPFB: 160786.

La Workstation o estación de trabajo puede conectarse a impresora y es compatible con la mayoría de las máquinas de facsímil.

Por comparación con los medios actuales de almacenamiento por discos ópticos, la técnica de almacenaje Ultratrip ha sido más totalmente probada y ofrece una duración de imágenes archivadas superior a 50 años, frente a 10 años de otros sistemas. Es además judicialmente aceptable en los tribunales de justicia como prueba válida, a semejanza de sus alternativas de almacenamiento digital, y representa un respaldo económico de seguridad.

Nueva línea mini de IBM

En uno de los anuncios más importantes en el presente año, IBM presentó una nueva familia de minicomputadores compatibles en software con sus equipos mayores.

Precisamente en el área de los mini, IBM ha resentido una fuerte competencia de parte de Digital Equipment, la que ha logrado consolidarse como una de las mayores empresas de computación en el mundo.

La ventaja de Digital estribaba en tener una línea completa de equipos compatibles entre sí, lo cual permite a sus clientes integrar sistemas y crecer armónicamente.

El debut de la familia 9370 de IBM marca una nueva tendencia en IBM que se asemeja más a la política de Digital de asegurar la migración de software entre los distintos tamaños de equipo. Los nuevos equipos que recién estarán disponibles el próximo año tendrán un valor entre US\$ 31.000 a US\$ 210.000 dólares.

EQUIPOS
COMPUTADORES
ATARI
TODA LA LINEA DE
IMPRESORAS
OKIDATA - PANASONIC

MULTIMATICA

COMPUTADORES - EQUIPOS Y MAQUINAS DE OFICINAS - ACCESORIOS
San Antonio 73 Galerías Nacionales - Teléfono 382663
Oficina Central: San Diego 318 - Teléfono 6964596

Guerra de los supercomputadores

Una situación insólita pero llena de proyecciones se está dando en el alejado y multimillonario mundo de las supercomputadoras. En efecto, la competencia que se le ha presentado a los computadores más veloces del planeta, los Cray, proviene del modesto Transputer, el computador en una pastilla producido por Inmos en Inglaterra.

Lo insólito se presenta cuando vemos el Cray 2, un computador enorme refrigerado en helio líquido y lo comparamos con una pequeña pastilla, pero la gran ventaja del Transputer está en su arquitectura modular que permite hacer trabajar en paralelo varios procesadores simultáneamente.

Por ejemplo, la computadora T/40000 desarrollada por Floating Point Systems que utiliza la tecnología del Transputer puede procesar hasta 200 veces más rápido que un Cray 2 y cuesta un décimo de su valor y todo esto gracias al procesamiento paralelo y al lenguaje especial de programación OCCAM.

Datasystem participa de un proyecto de automatización de avanzada.

Junto a Epic Data de Canadá, C&M de Italia y Multitech de Estados Unidos, las empresas Data System y Startin de Argentina participarán del fabuloso proyecto de automatización integral de la empresa automotriz SEVEL.

El sistema a aplicarse ya es utilizado en otros países por empresas como Boeing, General Dynamics, Fiat Gruppo Auto y Olivetti, entre otras, y abarcará las cuatro plantas que SEVEL posee en El Palomar, Córdoba, Brandsen y Uruguay, e incluirá a sus 360 concesionarios, 400 proveedores, y otras firmas asociadas al Grupo Fiat.

Esto le permitirá a la terminal automovilística reducir entre un 30 y un 50 por ciento sus costos de producción, y a Data System poner una planta en funcionamiento con una inversión del orden de los seis millones de dólares.

El sistema digital de alta velo-

cidad para transmisión de datos, voz y facsimil, funcionará con una antena de microondas que se instalará en el edificio de SEVEL que se levanta en Retiro, y unirá las cuatro plantas mencionadas con Francia, Italia y Brasil.

Otra novedad interesante es que a partir de la línea de producción cada vehículo estará acompañado por una tarjeta Coblin, con un código inalterable, que permitirá seguir el proceso de producción de cada auto, y en la cual el concesionario podrá incluir los servicios que se le vayan realizando con posterioridad.

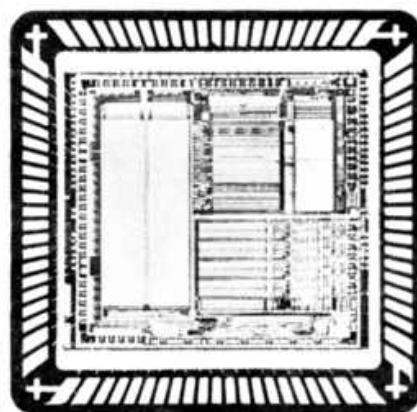
Según se explicó, el anuncio tiene relación con el lanzamiento del auto económico que producirá SEVEL, que ante la necesidad de abaratar los costos de su producción, se vio necesitado de automatizar el funcionamiento de la empresa en todos sus campos.

Burroughs vendió Memorex en US\$ 550 millones

Siguiendo con su política de deshacerse de activos prescindibles, Burroughs, ahora Unisys, anunció la venta de Memorex, la empresa que había comprado en 1981 en US\$ 117 millones.

Originalmente la compra había sido hecha para a través de ésta adquirir la tecnología de disqueteras y discos magnéticos para los minicomputadores Burroughs. Sin embargo, una nueva serie de modelos con diversos problemas técnicos provocaron serias pérdidas para Burroughs.

Los nuevos dueños de Memorex son antiguos funcionarios de ésta junto a inversionistas en Estados Unidos e Inglaterra.



Computación en países del Este

De acuerdo a informaciones publicadas por el "Washington Post" recientemente, en los países del bloque socialista los computadores habrían entrado más entre los usuarios privados que en instituciones gubernativas.

Citando el caso polaco, el matutino informaba que mientras en los ministerios e incluso en la línea aérea local se hace todo el trabajo manualmente, los activistas de la oposición utilizan PC Compatibles para editar sus diarios clandestinos.

El atraso que presentan estos países en comparación con Estados Unidos y otros países europeos es tal, que ha llevado a la formulación de vastos planes para cerrar la brecha. De hecho, la carencia de herramientas modernas de apoyo en las industrias están haciendo cada vez menos competitivos a los productos de esos países en los mercados internacionales.

En números, se estima que en Hungría hay 90.000 computadores y en Polonia, unos 500.000, aunque en su gran mayoría del tipo Sinclair, Commodore o Atari. La producción local es muy pequeña, siendo Alemania Oriental el principal fabricante de computadores personales, con una producción de 20.000 equipos anuales, una cantidad muy pobre incluso para nuestros estándares.

Amstrad apunta a nuevos segmentos

Alan Sugar, presidente de Amstrad, anunció que su compañía seguía trabajando en el desarrollo de nuevos y más económicos modelos de computadores personales. Para ello comenzaría a producir sus propios componentes, tales como discos duros e impresoras.

En impresoras precisamente es donde más avanzados están, pues ya se están produciendo entre 20 y 40.000 impresoras mensuales en Hong Kong que están siendo vendidas a OEMs.

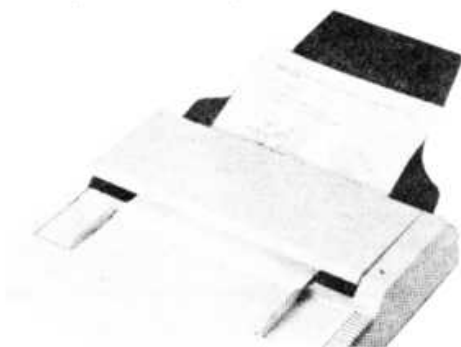
De tener éxito en sus planes, Sugar podría provocar una nueva conmoción en el mercado de los computadores personales.

El año de los scanners

De acuerdo a todos los pronósticos, 1987 será el año de los scanners, aquellos equipos que digitalizan la información textual o gráficos de un papel y la transmiten a un computador.

El auge de estos periféricos se debe principalmente al propio desarrollo que han tenido las aplicaciones de desktop publishing, publicaciones electrónicas que mediante software de composición y diagramación de textos y el uso de impresoras láser permite obtener impresos de calidad casi profesional.

Entre las compañías que van a la cabeza en el desarrollo de estos equipos está Canon, la que acaba de presentar dos nuevos modelos, la IX-12 y la IX-40. La resolución de la IX-12 es de 300 puntos por pulgada, la misma que es capaz de imprimir una impresora láser.



Olivetti pospone Introducción del M-22

Debido a problemas en la línea de producción en Hong Kong, Olivetti pospuso el lanzamiento oficial de su PC portátil, el M-22.

De acuerdo a trascendidos, el diseño del M-22, que fue presentado en prototipo a comienzos de este año no sería apto para ser producido económicamente, por lo que probablemente deba ser modificado.

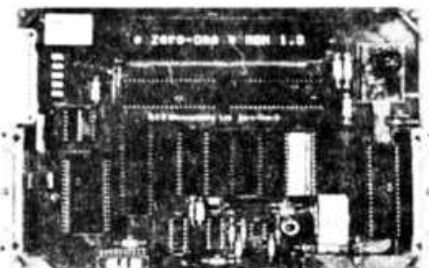
Computador en tarjeta

Se ha perfeccionado en el Reino Unido para aplicaciones de otros fabricantes de equipos en control de maquinaria, distribución de control y monitorización, seguridad en control de edificios y gerencia de la energía un ordenador potente, de bajo coste y adaptable, de un solo tablero, que tiene integrados pantalla de presentación y acceso de comunicaciones, así como extensas salidas y entradas.

El Zero-One-Q tiene una velocidad típica de operación superior a 600.000 instrucciones por segundo a una frecuencia de base de 2 MHz. Consta asimismo de una capacidad de memoria de acceso aleatorio, respaldada por batería, de 8 Mbytes y de una capacidad de lectura solamente de 32 kbytes. Los medios de entrada y salida integrados comprenden un acceso serial RS232 y 42 líneas paralelas, mientras que pueden

añadirse adicionales entradas y salidas en sistema externo empleando el conector de expansión que proporciona tres selecciones decodificadas de chip.

Se dispone igualmente de un programa Monitor que facilita medios de puesta a punto y desarrollo utilizando bien el teclado optativo o conectando un terminal RS232. Por añadidura proporciona numerosos y útiles servicios de entrada y salida y se suministra completo con pleno listado de base. El código de base de monitor también se ofrece en disco.



La respuesta americana

Mucho se ha hablado en la prensa respecto al proyecto japonés de quinta generación, a la iniciativa europea Esprit y casi nada se ha sabido respecto al consorcio de empresas norteamericanas que unieron sus multimillonarios esfuerzos de investigación y desarrollo.

La Microelectronics and Computer Technology Corporation, MCC, está formada por algunas de las principales empresas del área que decidieron unir sus esfuerzos de investigación, promoviendo la transferencia de tecnología entre ellas para enfrentar la competencia de Japón principalmente.

El primer resultado que se ha dado a conocer está relacionado con un nuevo método para fabricar los circuitos que unen a los distintos integrados en una tarjeta.

El problema que se atacaba es que si bien la tecnología se ha desarrollado velozmente en lo que se refiere a la construcción de integrados VLSI con una densidad de circuitos por centímetro que aumenta al doble cada tres años, la densidad en que se interconectan se duplica sólo cada diez años, creándose un cuello de botella difícil de resolver.

La solución encontrada está en la utilización de una cinta a la cual se adhiere la pastilla y en la cual luego son insertados pequeños filamentos metálicos que aseguran la interconexión de las pastillas. El producto será desarrollado y comercializado por 3M.

Procesamiento paralelo

Un grupo de trabajo en los Laboratorios Bell en Estados Unidos estaría cercano a fabricar un nuevo tipo de integrado semejante a la estructura del sistema nervioso animal.

A diferencia de la arquitectura convencional de los microprocesadores, el sistema nervioso animal no opera un paso a la vez sino lo hace en paralelo, vale decir, son muchas las neuronas las que se activan simultáneamente para converger en un resultado.

Así, un animal puede reconocer un objeto en milisegundos mientras que un poderoso computador provisto de una cámara y un analizador de imágenes tardaría varios segundos en hacerlo.

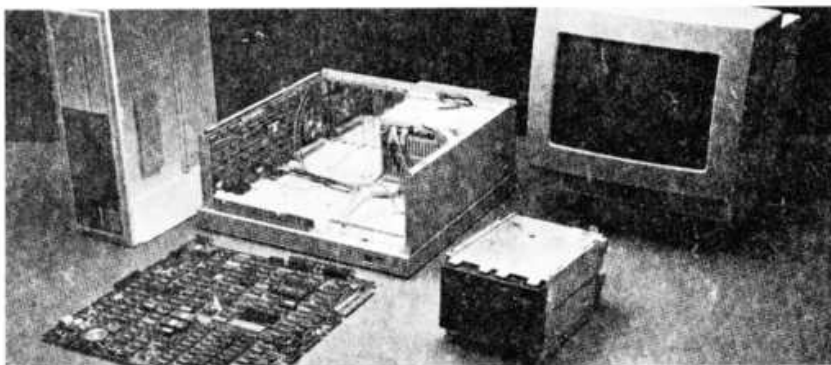
El circuito creado por el equipo de Laboratorios Bell, llamado Electronic Neural Network (ENN) imita la estructura animal con amplificadores en lugar de neuronas y resistencias en lugar de las sinapsis.

El modelo más poderoso hasta ahora contiene 256 neuronas compuestas por 25.000 transistores y 100.000 resistencias.

Olivetti profundiza acuerdo con la AT & T

Una nueva negociación entre Olivetti y AT&T culminó con una extensión del acuerdo entre ambas a diez años más, con la diferencia de que Olivetti asume el grueso de la producción y comercialización de la división computadores personales de la AT&T.

De acuerdo a los trascendidos, la gestión de AT&T en la venta de los modelos 6300 y 7300 no fue todo lo exitosa que se esperaba y llevó al gigante de las telecomunicaciones norteamericano a volcar sus esfuerzos en su propia área.

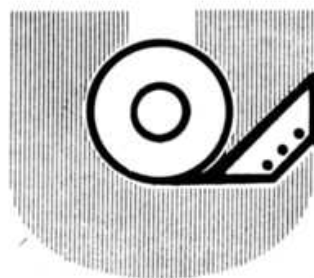


US\$ 5,000 para creativo de Unisys

En un concurso en el que participaron alrededor de 31.000 de los cien mil empleados del conglomerado Burroughs-Sperry, un programador de 36 años, Lee Machen fue quien propuso el nuevo nombre de este conglomerado.

Entre las proposiciones rechazadas figuraron Buros por su significado en español, Burrovac en honor a los mainframes de Sperry y otras más cómicas como SQUABB por "sure quality alternative to big blue" refiriéndose a IBM o TOC por "the other choice".

El nombre Unisys que significa United Information Systems fue presentado en una campaña de publicidad dirigida a nivel mundial por la agencia Young & Rubicam. El costo de esta campaña que duró tan sólo seis semanas está estimado en unos US\$ 16 millones de dólares.



IMPRESOS
UNIVERSAL Y CIA. LTDA.

GRAJALES 2948 FONO 97556 - SANTIAGO

Stock Blanco y Pautado
Papeles Autocopiativos y Calco One Time
Fabricación en Todas las Medidas
Asesoría en Diseño

Para todas las marcas y tipos de Impresoras.

Para su Personal Computer

FORMULARIOS CONTINUOS

PARA COMPUTADORES IBM SOLUCIONES ST



Si de computadores IBM se trata, trate con el especialista.

IBM es sinónimo de excelencia en computación y ST es experto en IBM. Regístrelo en su memoria. En ST Computación comercializamos exclusivamente sistemas y computadores IBM. Por lo tanto, somos la empresa con mayor experiencia y conocimiento en estos equipos. Nuestra ilimitada política de servicios, le garantiza que el computador IBM adquirido en ST Computación, le otorgará todos los beneficios esperados.


ST computación

Visítenos en nuestro Edificio S T:
Génova 2086 o llámenos al Fono 2514571.



Lanzan la MV/15000

La MV/15000 es la nueva serie de equipos que Data General puso, a partir del presente mes, a disposición de los usuarios chilenos. Con ella se ha lanzado una línea completa de máquinas con sus unidades de procesamiento completas (CPU, procesadores de comunicaciones y procesador de diagnóstico) integradas en sólo dos tarjetas electrónicas. Esto permite mejorar la relación precio/rendimiento, aumentar la disponibilidad del sistema y disminuir los costos de mantenimiento.

La nueva línea está constituida por el MV/15000 modelo 8, el modelo 10 y el modelo 20. La evolución del modelo 8 al 10 ó al 20 se efectúa por el solo cambio de la tarjeta del procesador. Lo mismo sucede con la evolución del modelo 10 al 20.

Entre las características técnicas de los nuevos equipos se pueden señalar: memoria máxima, 32 MB; direccionamiento virtual, 4 GB; máximo número de terminales/impresoras, 432; capacidad máxima en disco, 16,5 GB; número máximo cintas, 26; número máximo procesos, 1024; memoria cache datos 16 KB.



Los tres modelos del MV/15000 pueden compartir un mismo gabinete.

Commodore exhibe software educacional

En definitiva Commodore se ha propuesto convertirse en la más seria alternativa computacional para la educación y para ello sus representantes en el país han comenzado a liberar una amplia gama de programas dirigidos a ese sector.

Las dos principales líneas de software liberadas son la "Serie Educacional", con las materias escolares entre 4º básico y 4º medio de acuerdo a los planes del Ministerio de Educación y "Saber y Conocer" con pruebas de cultura general y recreacional.

Otro software que ha provocado favorables reacciones es "El Preguntón", un programa que permite generar programas en que se evalúan conocimientos de las distintas materias.



HP lanzó nuevos micros

Dos computadores para empresas medianas introdujo este mes en el mercado chileno Hewlett Packard. Son el Micro 3000 y el Micro 3000XE que permiten un mayor poder a un menor costo.

Ambos sistemas utilizan la tecnología de chip NMOS III de HP y Chips de memoria de 1 megabit. Estas características se presentan por primera vez en computadores comerciales de HP. Según sus representantes en el país permiten una mejora del 30 al 50 por ciento en la razón precio-rendimiento respecto a los sistemas actuales.

El Micro 3000 está dirigido a oficinas que cuentan con cuatro

a 16 usuarios. El micro 3000XE, por su parte, es útil a empresas con ocho a 56 usuarios.

El primero, para cuatro usuarios, cuesta US\$ 33.000 incluyendo una CPU con dos Mb de memoria principal, sistema operativo MPE V y utilerías. Además viene con sistema de manejo de base de datos HP TurboIMAGE, una unidad de disco de 81 Mb, cuatro terminales HP 2392A y una unidad de cartucho de cinta de seguridad.

El 3000XE para 12 usuarios, de configuración similar al anterior, con una unidad de disco de 130 Mb cuesta 72 mil dólares. Estos valores no incluyen el IVA.

Designan gerentes en ECOM

María Teresa Rosende resultó ratificada por el directorio de ECOM en el cargo de Gerente General de esa empresa.

Otras designaciones por área recayeron en: Olga Urrea M. en Administración y Finanzas; Agustín Paya C. como Gerente de Operaciones; Gerente Comercial, Sergio Ramírez; Gerente de Ingeniería, Fanny Pacheco y Gerente de Capacitación, Eduardo Beas.

A cargo de la sucursal de Valparaíso fue nombrado Lincoyán Latorre Caro.

Durante el mes de enero y en el verano, Ecom continuará desarrollando una larga serie de cursos y seminarios destinados a ejecutivos y profesionales. Entre los cursos destacan Computación en la Administración y Capacitación, Administración de cinotecas, DMS, Auditoría computacional y otros.

En otro ámbito, Ecom comenzará también a desarrollar cursos introductorios a la computación para escolares. Mayores informaciones en el 696 76 63.

¿ES UN TERMINAL? ¿ES UN PC?

Ambos. Es un PC Multitech conectable a su mainframe.

- En un momento usted está operando su PC con un programa específico. Al instante siguiente, usted está conectado al computador central, con una gran cantidad de información a su disposición.
- Y eso no es todo. Los PC Multitech además de ser conectables son compatibles: trabajan en la norma en que está hecha la mayoría del software existente. Es decir, en su PC Multitech usted puede usar los programas Lotus 1-2-3, dBASE III, Wordstar, Sistemas Administrativo/Contables, etc.
- Multitech está representado en Chile por CIENTEC, lo que significa respaldo sólido, servicio, asesoría técnica, garantía por 6 meses y capacitación gratis. Acérquese a CIENTEC o a algunos de sus distribuidores y conozca las tres familias Multitech: el Popular 500 (PC), el Plus 700 (XT) y el nuevo Accel 900 (AT). Establezca la relación entre las características de estos equipos y su precio, y comprenderá por qué Multitech se ha convertido en líder de su segmento en Chile.

MODELO	PC-POPULAR	PC-PLUS	PC-ACCEL
Microprocesador	Intel 8088	Intel 8088 2	Intel 8086
Coprocesador Opcional	-	8087 2	80287
Velocidad Proceso	4.77 MHz	4.77-8MHz	6.8 MHz
Memoria RAM	256 512 KB	256 640 KB	512 KB 3MB
Disquetera	12 x 360 KB	12 x 360 KB	12 x 12 MB 1 x 360 KB
Disco Fijo	10 20MB	10 20MB	20 30 40 MB
Conectores	4 tipo IBM PC	6 tipo IBM PC	8 tipo IBM AT
Reloj de tiempo real	-	si	si
Puertas Centronics	2	2	2
Puertas Seriales	1	2	2

Tarjetas Video

MGA Alta resolución monocromática 720 x 348 (compatible Hercules)
CGA resolución normal monocromática 640 x 200 o color 320 x 200
EGA Alta resolución color 640 x 380

Además, por la compra de cualquier PC Multitech, le obsequiaremos software original (costo US\$ 120 e q.m.n., cada uno).

Con el Popular: usted escoge entre el VP Planner y el VP Info.

Con el Plus o el Accel: le obsequiaremos ambos programas.

Oferta: Compre el nuevo Accel 900 (AT), con disco duro 20 MB, a precio de lanzamiento, desde US\$ 3,789
20% I.V.A. 758

OFERTA LIMITADA US\$ 4,547 (eq.m.n.)



CIENTEC

La Ciencia Aplicada

Antonio Varas 754 - Teléfono 743508 - Santiago

DISTRIBUIDORES CIENTEC

SANTIAGO : Adcom, Tel. 2237426; Asper, Tel. 2254775.
Computer Market, Tel. 2243474.
Graphic Photo, Tel. 461940
Ingeniería de Servicios Electrónicos, Tel. 776991.
ANTOFAGASTA : Infocom, Tel. 224762.
LA SERENA : Empresa Chilena de Computación, Tel. 213222.
VINA DEL MAR : Vecom Ltda, Tel. 882490.
RANCAGUA : Ascoming Ltda, Tel. 21869.
TALCA : Infoland, Tel. 35837.
CONCEPCION : Empresa Chilena de Computación, Caupolicán 567.
OSORNO : STG. Ltda, Ramirez 939, locales 7 y 8, Tel. 4243.

Multitech

MAI BASIC FOUR introduce nuevo MAI 3000

MAI BASIC FOUR anunció la disponibilidad del nuevo sistema de computación multiusuario comercial: el MAI 3000.

Utilizando el procesador Motorola 68020, este sistema de 32 bits permite llenar los requerimientos de negocios que necesitan un sistema de hasta 34 usuarios. El MAI 3000 está también posicionado como una expansión para los actuales usuarios del sistema MAI 2000.

Esta ampliación permite la conversión del sistema MAI 2000 con la CPU Motorola de 16 bits 68010 transformándolo en un computador MAI 3000 de 32 bits y al mismo tiempo protegiendo la inversión de los clientes en Software, dispositivos periféricos, unidades de discos y

controladores internos del computador.

Los usuarios del nuevo sistema MAI 3000 disponen de opciones de crecimiento incremental para ampliar las capacidades del sistema. Comenzando con un sistema de un solo usuario con 1 MB de memoria RAM y 44 MB de capacidad formateada en disco, hasta un sistema con 34 puertas seriales en total, 6 MB de memoria RAM y 360 MB de capacidad formateada en discos.

El sistema operativo del MAI 3000 es el BOSS/IX que siendo el mismo sistema operativo del MAI 2000 ha sido ampliado para mejorar la eficiencia del sistema y permitirle al usuario aprovechar las características superio-

res del sistema MAI 3000. BOSS/IX es un sistema operativo UNIX específicamente diseñado para el procesamiento de información comercial. El BOSS/IX en el MAI 3000 es totalmente compatible con el BOSS/IX del MAI 2000 y provee características adicionales que incluyen: rendimiento mejorado de sistema y soporte para una tercera unidad de discos Winchester y un segundo controlador de discos. Además permite soportar hasta 6 MB de memoria RAM, con chequeo y corrección de errores de memoria ECC, memoria cache y una unidad de administración de memoria controlada por hardware.

Nuevos equipos de Panasonic

Mellafe y Salas inició la venta de los dos últimos computadores personales para automatización de oficinas lanzados por Panasonic. Son el FX 800 y el FX 600, ambos con espacio para instalar dos discos duros.

El FX 800 viene en tres configuraciones básicas, todas con 512 KB en RAM expandibles a 1,1 MB. Trae disketteras de 360 MB y acepta discos duros de 20, 30 y 40 MB. Procesador 80286 (estándar AT), teclados en castellano y numérico separados. Diez slots de expansión. Su valor: US\$ 4.500 más IVA.

Por su parte el FX 600 trae 640 KB de memoria RAM; dos disketteras de 360 MB. También acepta discos duros de 20, 30 y 40 MB y viene con procesador 8086-2. Tiene seis slots de expansión y las mismas características en teclado que el anterior. Su precio es de 2.500 dólares más IVA.



Commodore viene con yapa

Desde octubre Commodore Chile vende su modelo C-64 C con el nuevo sistema GEOS incorporado. Por otra parte entrega el Commodore 64 con un programa tutorial que enseña el uso de ese equipo.

También para este último modelo un grupo de estudiantes egresados de la Universidad de Chile desarrollaron un sistema de cunicultura que algunos distribuidores están comercializando.

El GEOS es un sistema operativo intuitivo con iconos, menú de venta, basurero, calculadora y otros. Funciona en

todos los ocho millones de Commodore 64 que hay en el mundo.

En cuanto al tutorial, fue desarrollado en Chile por la empresa Softlogic y utiliza todos los recursos del computador: música, animación, gráficos y otros, en forma interactiva.

En cuanto al sistema de cunicultura desarrollado por los estudiantes sirve para controlar la producción de conejos (peso, dieta, reproducción, identificación del conejo, etc.).



1
7.000.000



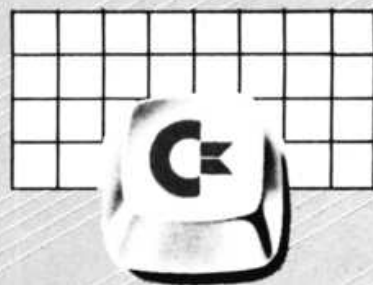
Esta navidad no piense en juguetes. Regale Commodore*.

Los niños crecen, pero la mayoría de los computadores nó.

¡No se equivoque!

Comprar un computador es una decisión importante.
Y tarde o temprano su hijo necesitará un Commodore.
7.000.000 de unidades vendidas en todo el mundo.
25.000 programas.

Y capaz de seguir creciendo con todo lo nuevo que
va creando el mundo de la computación.
Regáله un Commodore esta Navidad.
Cuanto antes, mejor. No regale nada menos que...



COMMODORE
Cuando los regalos son inteligentes.

(*) "Mientras otros deben ofrecer equipos discontinuados, nuestro compromiso es seguir fabricando y desarrollando el computador que, no en vano, es el más vendido de todos los tiempos". Tom Rattigan Presidente Commodore Business Machines

ACCA prepara congreso

Con bastante anticipación —como conviene— el Comité Ejecutivo del Tercer Congreso Latinoamericano de Control Automático fijó las fechas de recepción de resúmenes y de trabajos definitivos. En el primer caso el plazo vence el primero de noviembre de 1987. En el otro, la fecha tope es el primero de abril del 88. El evento tendrá lugar entre el tres y el ocho de octubre de este último año. La sede se la están disputando Santiago y Viña del Mar.

La organización corresponde a la Asociación Chilena de Control Automático (ACCA) quien —entretanto— proyecta realizar una serie de eventos en la capital y regiones a través de sus comités técnicos.

La presidencia del Comité Ejecutivo recayó en el profesor Gastón Lefranc. Los temas a tratar en el Congreso abarcan 15 áreas.

Integran además el Comité las siguientes personas: Jorge Yutronic, Guillermo González, Aldo Cipriano, Sebastián Ríos, Felisa Córdova y Ricardo Rojas.

Entre las áreas se incluyen automatización de procesos, control de proceso con computadora, robótica, visión artificial, sistemas flexibles de producción, diseño asistido por computador (CAD), manufactura asistida por computador (CAM), electrónica de potencia y otros.

Mientras tanto, en 1987 en agosto tendrá lugar el Tercer Taller de Robótica y en septiembre se efectuarán conjuntamente el Séptimo Seminario ACCA y el Segundo Seminario IEEE. Este último tratará sobre sistemas CAD/CAM, sistemas flexibles de producción y robótica industrial.

Inauguran laboratorio computarizado

Setenta y cinco mil exámenes atenderá mensualmente el Laboratorio Clínico del Sistema Nacional del Servicio de Salud (SNSS) que funciona en el Hospital Paula Jaraquemada. Esta tarea la efectuará gracias a sus nuevos equipos de computación inaugurados a comienzos del presente mes.

Al acto que marcó el inicio oficial de actividades del sistema computacional asistieron representantes de la Corporación de Desarrollo Social "Lautaro" que administra el nosocomio nombrado y de la empresa Datamerica que vendió los equipos.

Estos consisten en una unidad central y ocho terminales PC Corona, una impresora Genicom y dos impresoras Okidata.

La unidad central viene con 40 MB en disco duro, cinta de respaldo tipo streamer de 43 MB, una diskettera de 360 KB, memoria cache de 512 K. Los terminales vienen con 512 Kb RAM, alta resolución y compatibles con el estándar de IBM.

La inauguración del laboratorio computarizado constituye la culminación de la primera etapa del Proyecto de Laboratorio Centralizado.



Teorema presenta redes locales

La sucursal local de Teorema en Concepción dictó una conferencia a ejecutivos de las principales empresas de la zona respecto a redes de área local.

En la oportunidad se hizo una presentación de la red instalada por Teorema en CADE IDEPE, una empresa de ingeniería que tiene su sede en Santiago. Esta red cuenta con 25 estaciones y se la considera una de las más grandes que se han instalado en nuestro país.

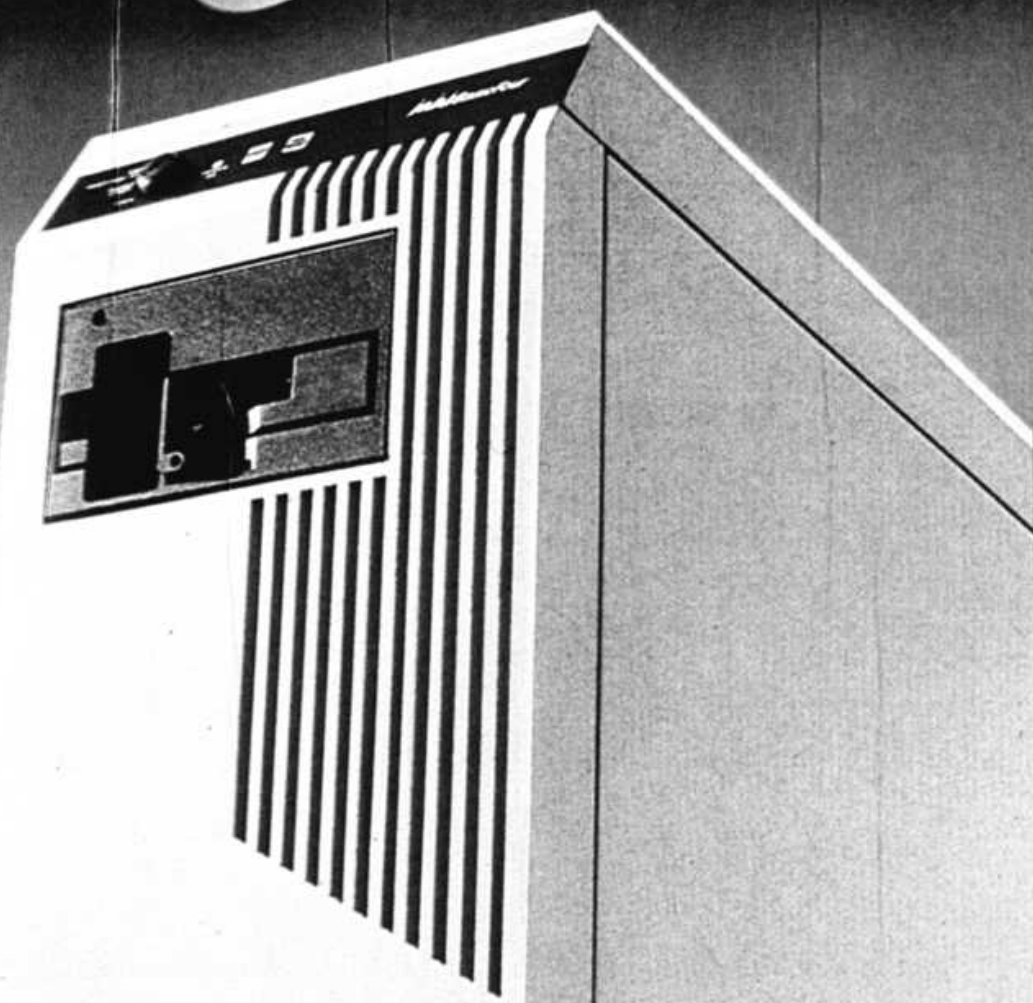
Estudiante premiado

El estudiante Hernán Méndez Salas de la Universidad de Chile obtuvo el primer premio del concurso establecido por la Asociación Chilena de Control Automático (ACCA) para las Memorias de Título en Control Automático y Areas Afines. Profesor guía del alumno fue Guillermo González.

El trabajo ganador fue elegido por una comisión a propósito entre seis memorias seleccionadas por los coordinadores de ACCA.

Para 1987 se establecerá, además, un premio para la mejor tesis de post grado. Las distinciones comprenden un galvano y una suma de dinero.

NUEVO MAI 3000



Con el nuevo computador MAI 3000 Usted puede planificar el futuro de su empresa sobre fundamentos sólidos.

¿Tiene Usted definido como será el crecimiento de sus necesidades computacionales en los próximos años?

Magnífico.

Entonces puede comprobar que el Sistema MAI 3000 podrá satisfacer plenamente sus requerimientos. Sin embargo lo más interesante es que este sistema también podrá hacerse cargo de las necesidades que no estaban previstas en esos planes.

Conozca la experiencia de los numerosos y felices usuarios de nuestro sistema MAI 2000, quienes ya pueden tener más de lo previsto... y mucho más, porque ahora también pueden transformar su sistema de 16 bit al nuevo super-microcomputador de 32 bit MAI 3000, sin siquiera moverlo de su oficina. Y sin cambiar programas pueden agregar terminales hasta completar 34 que operaran simultáneamente. Además aumentar la memoria RAM desde 1,5 MB hasta 6 MB.

CARACTERISTICAS SOBRESALIENTES

- Sistema Operativo BOSS/IX™ basado en UNIX™, con el lenguaje Business BASIC legítimo.
- CPU Motorola MC68020® de 32 bit.
- Memoria ECC con corrección automática de errores. Hasta 6 MB RAM.
- Memoria Cache hasta 64 instrucciones de CPU.
- Memoria Cache para datos en CPU.
- MAI ORIGIN™. Poderoso generador de informes y aplicaciones.
- MAI OFFICE™. Sistema integrado de automatización de oficinas.
- MAI MAGNET™. La clave para interconexión en redes locales y remotas.
- MAI TRACE™. Control automático del uso de teléfonos.
- RM/COBOL™.
- MAI PC-Link™.

MAI 3000 A LA ALTURA DE SUS REQUERIMIENTOS COMPUTACIONALES



MAI BasicFour
12 AÑOS EN CHILE

LOGICA

VECTINAL 61 LAS CONDES TELEFONO 2312626
CONCEPCION: O'HIGGINS 366 TELEFONO 225187

***No son los más caros,
ni los más baratos;
pero si los mejores!***

MICROCOMPUTADORES SANYO



EVALUACION ENTRE COMPATIBLES*					
Computador		Grabación Archivo	Lectura Archivo	Criba Eratóstenes	Cálculos
Epson QX-16	R	58	30	179	54
Hewlett-Packard	R	35	34	148	49
IBM PC	R	56	46	191	69
IBM PC XT	R	59	41	209	70
NCR Plus 4	R	57	30	182	56
Olivetti M 24	R	33	30	124	38
Zenith 158	R	56	29	132	40
Zenith 151	R	58	55	199	70
Sanyo MBC-885	R	31	30	112	34

Realizada por la Revista Computación Personal y publicada en su edición de Diciembre.

*Todas las medidas están en segundos.
Las pruebas fueron realizadas bajo MS-DOS 2.1 con GW-BASIC.

R = Son marcas registradas.

 **SANYO**
DE CHILE LTDA.

DISTRIBUIDORES AUTORIZADOS
INDES LTDA. Teléfono 392800;
INFORMATICA CHILENA LTDA.: Teléfono 2515836;
STUEDEMANN S.A.: Teléfono 2311923
ASSIN LTDA.: Teléfono 5550930

NOTICIAS NACIONALES

Aparece el MV/7800 DC

Data General lanzó en noviembre el MV/7800 DC. Este nuevo equipo computacional es una versión reducida del MV/7800, lanzado en agosto de este año.

El MV/7800 DC tiene la misma tarjeta electrónica base del MV/7800. En ella están dispuestos la CPU de más de 1 MIPS, coprocesador de punto flotante y dos o cuatro MB de memoria. También procesador de diagnóstico remoto y controladores de I/O.

La nueva versión, montada en un gabinete pequeño de oficina, permite la conexión de hasta 48 dispositivos asincrónicos tales como terminales, impresoras y otros. Puede alcanzar hasta 14 MB de memoria principal, 320 MB de almacenamiento en disco interno y 1,2 GB de almacenamiento en disco externo.

Adicionalmente se le puede conectar una cinta stream de 1600 BPI de 25 MB de capacidad o un cartridge de 15 MB. El sistema permite su conexión a redes de área local y comunicación sincrónica con otros sistemas.

Los Sistemas Operativos del MV/7800 DC son compatibles con todos los otros sistemas de la familia Eclipse MV. Esto significa que los periféricos son compatibles para toda la familia. Por este motivo el lanzamiento de un nuevo procesador no necesita esperar el lanzamiento de sus periféricos.



El MV/7800 DC. Las dos últimas letras significan Department Cluster.

Crece asociación de empresas

Nuevos estatutos aprobó la Asociación Chilena de Empresas de Informática AG que preside Víctor Celis. Los cambios permitieron el ingreso de IBM, NCR y Unisys.

El organismo que nació hace poco más de dos años como asociación de servicios informáticos nacionales se amplió luego a empresas chilenas vendedoras de equipo y hoy a empresas internacionales con filiales en el país.

Las modificaciones de documentos determinan también la reducción del directorio de 12 a

nueve miembros a elegir cada tres años. Además establecen dos estamentos de acuerdo al capital y reserva de las empresas. En el A quedarán las mayores y en el B las más pequeñas.

En la actualidad la Asociación está formada por 13 empresas, entre las que se encuentra ECOM que pudo ingresar al pasar al sector privado. Ellas representan, según Víctor Celis, el 70 por ciento de las ventas de servicios y más aún de las ventas de equipos.

UNA LINEA DIRECTA A SU COMPUTADOR

Digiman Ltda. pone a su disposición: **suministros computacionales** para todas las marcas (Discos magnéticos, Diskettes, Cintas Magnéticas, Cintas para Impresoras).

Equipos periféricos, computadores IBM PC, Microcomputadores DS-500, Plotters Gráficos e Impresoras.

Servicio Técnico con 12 años de experiencia en equipos MAI Basic Four e IBM PC.

DIGIMAN

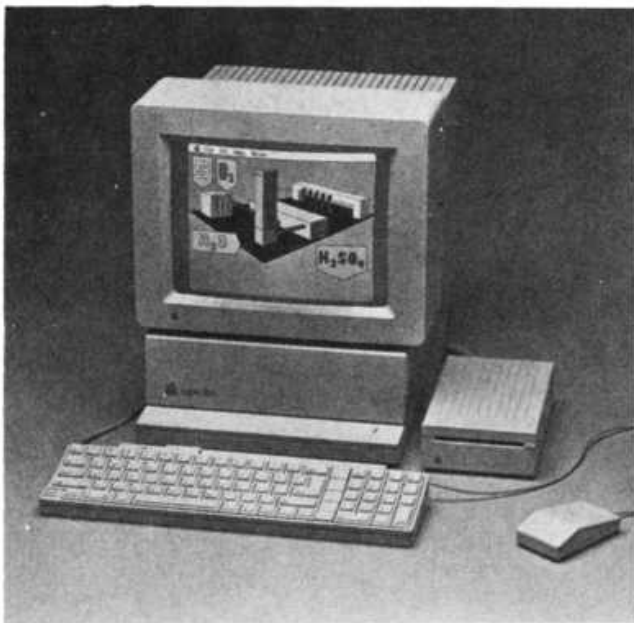
Monjitas 454 Of. 207
Teléfonos 337753 - 337764

337764

Nuevo modelo de Apple compatibiliza sus dos principales líneas.

APPLE II GS: UN PUENTE AL FUTURO

Eduardo Sabrovsky J.



El 19 de septiembre recién pasado, con toda la espectacularidad que se estila en estos casos, Apple Computer lanzó al mercado un nuevo integrante de la línea Apple II: el Apple II GS (Graphics & Sound). Si bien estos lanzamientos han ocurrido con cierta periodicidad a partir de 1978, fecha de aparición del primer Apple II, lo cierto es que, por primera vez, nos encontramos ante una variación radical en la arquitectura de la línea, arquitectura que los modelos anteriores —el II Plus ya discontinuado, y los aún vigentes IIe y IIfx— siempre conservaron con sólo modificaciones menores. Los cambios arquitectónicos son tan profundos que la mantención del nombre "Apple II" alude, no tanto al modo "nativo" de la máquina, similar a Macintosh, sino más bien a su modo de "emulación", que proporciona compatibilidad para una muy alta proporción del software, periféricos, tarjetas y accesorios preexistentes.

Si bien en este artículo nos proponemos describir la arquitectura del Apple II GS, se nos ocurre que una descripción técnica no estaría completa si no la insertáramos en un contexto, dado por la estrategia global de productos que Apple pareciera estar desarrollando, y al interior de la cual la mencionada arquitectura encuentra su razón de ser. Por lo tanto, primeramente haremos un recuento de la posición de Apple en el mercado, y de sus proyecciones.

El éxito de una política "aislacionista".

1986, un año difícil en general para los proveedores de computadores personales, ha sido sin embargo un buen año para Apple Computer, quizás el mejor desde que, en 1981, la entrada de IBM al mercado desplazó la marca de la manzana del primer lugar en las ventas. El terremoto que desencadenó el PC IBM hizo conjeturar que los días de Apple podían estar contados; el mismo lanzamiento del Macintosh en 1984, a pesar de la innovación tecnológica que el nuevo equipo representaba, fue recibido con reservas; si bien se elogiaba la audacia de Apple, su negativa a seguir la corriente representada por la arquitectura del PC IBM y el sistema operativo DOS parecían condenarla al aislamiento, confirmado por una relativa lentitud del Macintosh en penetrar en el mercado de la empresa, hacia el cual estaba orientado. Pero todo esto parece ser cosa del pasado, y las que parecían ser debilidades han resultado ser fortalezas. El "aislacionismo" de Apple respecto al "estándar" MS-DOS le ha permitido quedar al margen de las guerras de precios que se dan en torno a los "PC-compatibles"; así la revista especializada en temas económicos "Fortune", en su edición del 15 de septiembre, en un artículo titulado sugestivamente "Cloneless Macintosh" ("Macintosh sin 'clonos'") comenta cómo, para Apple, dada la política de innovación tecnológica que la caracteriza, no hay "clonos" en el horizonte, en cambio, los "PC-compatibles" superan ya en ventas al producto original. Macintosh Plus, la última versión de Macintosh lanzada a comienzos de año, parece haber superado las limitaciones del producto original: así parecen al menos percibirlo las empresas, que según "Fortune" compran un 60% de la producción de Macs, y las grandes casas de software —Microsoft, Lotus, Borland, Ashton Tate que acaba de liberar DBase Mac, una versión de DBase III— que invierten en desarrollar productos para él. Estos factores se reflejan en el valor de las acciones de Apple, que se ha duplicado durante el presente año, así como en las utilidades que han batido los récords históricos en dos de los cuatro trimestres del año. Finalmente, Apple mantiene una participación de mercado del orden del 50% en educación —aunque amenazada de erosión por productos competitivos de tecnología más actualizada— y domina sin contrapesos en el área de los siste-

mas electrónicos de publicación, en la cual es pionero con Macintosh y la impresora Laser Writer; según la misma fuente anteriormente citada, uno de cada cinco dólares invertidos en los EE.UU. en comprar impresoras es captado por Apple mediante la Laser Writer.

El lugar del Apple II GS en la estrategia.

Antes de pasar a especular acerca de cómo esta posición se refleja estratégicamente para la línea Apple II, digamos en síntesis que la situación de Apple podría resumirse en la frase "renovarse o morir", puesto que sus ventajas competitivas radican en la reproducción continua de la innovación. Debe esforzarse en combinar una cultura de gran corporación, con el espíritu pionero que dio origen a la computación personal. Especulemos ahora respecto al significado del Apple II GS. Nos parece que Apple tiene tres objetivos para la línea Apple II:

1. Mantener su participación de mercado en mercados claves como educación, mediante tecnología "de punta", que resista la comparación con equipos competidores de reciente aparición.

2. Establecer un puente entre la tecnología Macintosh y la línea Apple II, en la perspectiva de

Las rutinas residentes en la Rom del GS son similares a sus correspondientes en el Macintosh

compatibilizarlas, a nivel periféricos, software, etc. Se trataría de incentivar en particular a los desarrolladores de software, garantizándoles que con un mismo producto —o introduciendo modificaciones menores— podrán acceder a un mercado de varios millones de computadores instalados, como es el que conforman conjuntamente Macintosh y Apple II.

3. Mantener la lealtad de la actual base de usuarios Apple II, ofreciéndoles un camino de migración hacia la tecnología de punta.

Veremos ahora cómo las características del Apple II GS implementarían esta estrategia.

Procesador y memoria.

El Apple II GS está construido en torno al procesador W65C816, de 16 bits, funcionando a 2,8 MHz, que reemplaza al legendario 6502, de 8 bits y 1 MHz. El 65C816 cuenta con registros de direccionamiento de 24 bits que permiten direccionar hasta 16 MB de memoria. La configuración básica del GS incluye 256 KB RAM y 128 KB ROM, expansibles hasta 8 MB y 1 MB, respectivamente.

Con estas características, uno pudiera preguntarse cómo es posible que se mantenga compatibilidad con el software y los periféricos preexistentes. La respuesta es que, sin embargo, esta compatibilidad es efectiva para un 95% del software —el cual corre a una velocidad de 2 a 3 ve-

ces mayor que en un Apple II corriente— y un 80% del hardware. Para conseguirla, los ingenieros de software debieron extremar su ingenio.

Por una parte, las instrucciones del W65C816 son un super-conjunto de las del 6502. Hasta ahí la cosa es sencilla. Los problemas empiezan cuando se considera la memoria de los Apple II preexistentes: en ésta, existía una serie de sectores reservados, que eran utilizados por el software, por ejemplo para la interacción con periféricos; ahora bien, estos sectores, dada la función que cumplen, son dependientes del tiempo, deben operar a 1 Mhz. La solución está en una técnica —cuya descripción detallada daría por sí sola para un artículo— denominada "shadowing". Un "chip" especial denominado FIP (Fast Processor Interfase) interpreta cualquier intento de escribir a las áreas reservadas, reduce —sólo en esos instantes— la velocidad a 1 Mhz, y duplica la operación en un banco de memoria (64 KB) lenta. De esta manera, se consigue que todo el software que no viola las convenciones de diseño del Apple II tradicional se ejecute —con excepción de algunas instrucciones— a 2,8 Mhz. El software diseñado especialmente para el GS, en cambio, puede deshabilitar el "shadowing", accediendo un "shadow register".

Gráficos y Sonido.

Estas características son tan relevantes como para constituir el distintivo del GS (Graphics & Sound). Además de los modos gráficos tradicionales del Apple II, el GS aporta dos novedades: super-alta resolución, de 200x320 pixels a 16 colores, o 200x640 a 4 colores, a elección en una paleta de 4096 colores.

En cuanto a sonido, hay un "chip" sintetizador incluido en el GS, el Ensoniq Digital Oscillator Chip, capaz de sintetizar hasta 15 instrumentos diferentes. Otros computadores —incluidos los Apple II tradicionales— han podido ser conectados a sintetizadores externos, pero es la primera vez que esta capacidad se incluye como parte de la configuración básica. Apple apunta hacia una nueva generación en el software educacional, que utilice integralmente el sonido. Con el "chip" Ensoniq será posible, por ejemplo, hacer programas que asistan en la enseñanza de un idioma extranjero sintetizando voz carente totalmente del monótono "acento computacional".

Interfaz Macintosh.

Apple ha implementado la tecnología Macintosh en el GS, completa, con "mouse", ventanas, iconos, menús, etc. Esta tecnología, además de presentar al usuario una interfaz gráfica que simula un escritorio —el "escritorio electrónico"— sobre el cual hay objetos que manipular intuitivamente mediante el "mouse", se caracteriza por estandarizar la interfaz con el usuario, que emplea elementos disponibles a todos los programas mediante

llamados a rutinas en ROM, las cuales en conjunto constituyen lo que Apple denomina el "toolbox", la "caja de herramientas". La memoria ROM del Apple II GS cumple precisamente esta función, y permite por tanto que los desarrolladores de software se concentren en los aspectos específicos de cada aplicación; se alcanza además un grado de consistencia entre programas que tiene un efecto sinérgico sobre las curvas de aprendizaje: aprender a usar bien un programa implica anticipar habilidades que serán requeridas por todos los demás programas que se llegue a utilizar.

Las rutinas residentes en la ROM del GS son en su mayoría muy similares a sus correspondientes en Macintosh. También lo son los lenguajes de programación anunciados junto con el producto, C y Pascal, los cuales dan acceso a las rutinas de la "caja de herramientas". De esta manera, se incentiva la migración de aplicaciones desde Mac hacia el GS, y la producción futura de software dirigido simultáneamente hacia ambas líneas.

Uno de los rasgos característicos de la tecnología de software Macintosh es la existencia de "accesorios de escritorio", pequeños programas que pueden ser invocados desde cualquier aplicación y que, en general, auxilian tareas que se presentan aleatoriamente durante una jornada de trabajo. Es el caso de calculadoras, agendas, libretas de anotaciones y objetos similares, que están habitualmente presentes sobre cualquier escritorio, y son simulados en la pantalla del Macintosh, estando disponibles para cualquier aplicación desde el omnipresente menú "de la manzana".

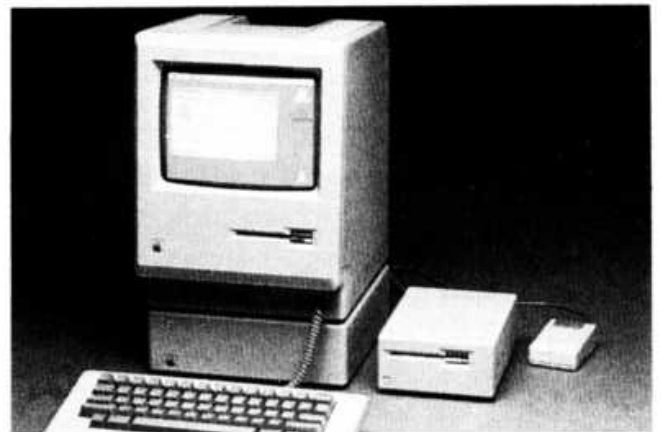
El Apple II GS ofrece también accesorios de escritorio, que son administrados por el "Desk Accessory Manager" presente en la ROM. Los accesorios pueden ser de dos tipos: "clásicos", que pueden ser invocados desde una aplicación Apple II tradicional mediante la depresión de alguna combinación de teclas, y los accesorios "nuevos", que se activan al estilo Macintosh, mediante el mouse y un menú ad-hoc. Hay un accesorio clásico que viene integrado al GS: es el Panel de Control, mediante el cual se escogen opciones de configuración, tales como asignación de las puertas seriales, características del despliegue (40 u 80 columnas, color), sonido; velocidad operacional (1 MHz o 2,8 MHz); uso de "slots" de expansión o de interfases integradas (ver sección siguiente para una explicación de estos conceptos); conjunto de caracteres para el teclado y el despliegue; fecha y hora.

Migración y puertas de expansión.

Hasta ahora, el mundo Apple ha estado dividido entre arquitecturas cerradas y abiertas. Estas últimas, como en el Apple II Plus y IIe, se caracterizan por "slots" de expansión, en los cuales los usuarios insertan tarjetas de interfaz diversas. Su fortaleza reside en que generan un mercado abierto de proveedores de accesorios de hardware

re que, tanto en el caso del Apple II como de los IBM PCs y compatibles —que también exhiben este tipo de arquitectura— ha demostrado ser un factor importante para el éxito de un equipo. En las arquitecturas cerradas, en cambio —como en el Apple IIc y Macintosh— el fabricante decide de antemano cuáles son las interfaces que los usuarios con mayor probabilidad utilizarán, y las incluye en la configuración básica. La ventaja reside en que la conexión de periféricos se facilita —se reduce a una operación tan simple como conectar un equipo modular de música— y, en general, se logra una mejor relación costo/rendimiento. Pero a la vez, se restringen las posibilidades de elección y de migración de los usuarios.

El Apple II GS soluciona el dilema ofreciendo lo mejor de ambos mundos: "puertas" integradas, como en el IIc, y 7 "slots" de expansión, como en el IIe. Las "puertas" incluidas son: 2 seriales (impresora y comunicaciones; la de comunicaciones sirve también para conexión con la red local Apple Talk); una puerta para conexión de unidades



de disco (permite hasta 4 unidades, 2 de 3,5" y 800 KB de capacidad, 2 de 5,25" y 143 KB); finalmente, están los conectores para monitor, TV y audífonos. De esta manera, se aseguran los beneficios de ambos tipos de arquitectura, y se da compatibilidad a todo el hardware existente, así como el software que lo utiliza. Finalmente, los usuarios de los actuales Apple IIe podrán actualizar sus equipos hasta incluir en ellos todas las características del GS, aunque aún no se conoce fecha de disponibilidad del "kit" correspondiente, ni la política concreta que se aplicará.

Conclusión: un camino a transitar.

Más allá de las características determinadas del Apple II GS, queremos terminar este artículo reflexionando acerca de las perspectivas que este anuncio ofrece, como la anticipación de un camino que tarde o temprano toda la industria de la computación personal debe recorrer, si no quiere ser víctima de la inercia generada por su propio éxito. Esta inercia surge de la aceptación y consagración por parte del mercado de ciertos estándares "de facto", como el Apple II o el IBM PC. En torno a ellos surgen miles de productos

APRENDA A HABLAR POLAROID.

CUANDO LAS PALABRAS
NO BASTAN, ESTA POLAROID.
AHORA, LOS COMPUTADORES
PUEDEN HABLAR EL LENGUAJE
UNIVERSAL EN COMPUTACION:
EL LENGUAJE POLAROID.
HOY, POLAROID ESTA
HACIENDO CON LOS FLOPPY
DISCS LO QUE HA HECHO POR
MAS DE CUATRO DECADAS CON
PELICULA INSTANTANEA,
EMPLEAR SU EXPERIENCIA
TECNOLOGICA DE
REVESTIMIENTO DE SUPERFICIES
PARA FABRICAR UN EXCELENTE
PRODUCTO CONSISTENTEMENTE.
APRENDA A HABLAR CON LOS
DISKETTES POLAROID Y USTED SE
EXPRESARA PERFECTAMENTE EN
UN NUEVO IDIOMA DE
PALABRAS VIVAS Y CLARAS.



 **Polaroid. El lenguaje universal.**

Polaroid es una marca registrada de Polaroid Corp., Cambridge, Mass., E.U.A.



complementarios de hardware y software, y se generan millonarios parques de equipos instalados: toda una masa, que llegado un cierto momento parece impedir toda innovación tecnológica, y que deja el mercado a merced de los fabricantes de "clonos", cuyos costos son menores en la medida en que no requieren de mayores inversiones en investigación y desarrollo. En el mundo de los "PC-compatibles", la inercia es tan fuerte que, a pesar de la introducción, en el modelo AT, del procesador Intel 80286, superior al Intel 8088 original, la verdad es que se ha desarrollado muy poco software que aproveche las características nativas del nuevo procesador —el modo "protegido", que ofrece entre otras cosas manejo de memoria virtual. Se da entonces la paradoja de que la mayor parte de los procesadores Intel 80286 instalados se sub-utilizan en la emulación de un procesador inferior.

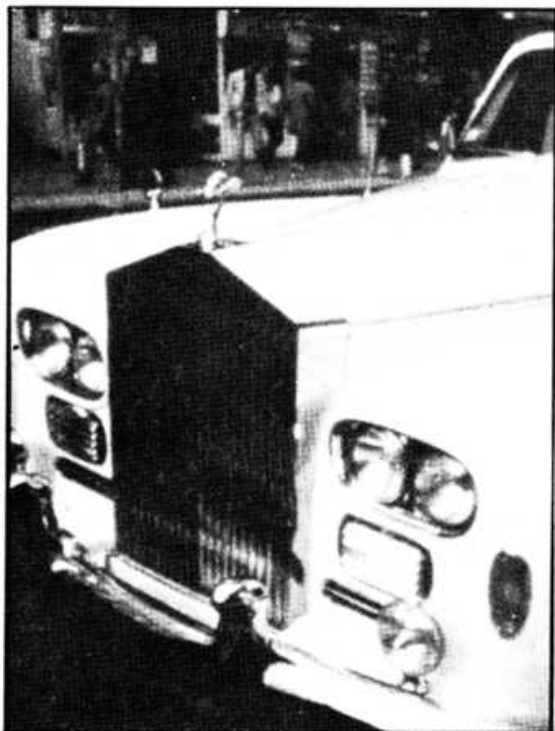
El W65C816 introducido por Apple en el GS podría correr la misma suerte, ser utilizado solamente como un acelerador del 6502 tradicional. Sin embargo, en el caso de Apple nos parece que las probabilidades de escapar del "campo inercial" de la antigua arquitectura son bastante más altas. La diferencia está determinada porque, en el caso de Apple, ya existe un equipo consoli-

do en el mercado —Macintosh— que constituye un polo de atracción alternativo para los "terceros", los numerosos productores independientes de hardware y software compatible de los cuales depende el éxito de un producto. En otras palabras, el salto de la línea Apple II hacia el futuro no es al vacío, sino hacia una cabeza de puente ya establecida: los centenares de programas existentes para Macintosh, cuya migración a Apple II GS es facilitada por la "caja de herramientas" que hemos mencionado. **M**

Eduardo Sabrovsky Jau-
neau se ha desempeñado
durante casi una década en
cargos técnicos y ejecutivos
en empresas del área infor-
mática. En la actualidad, tra-
baja como consultor inde-
pendiente, y desarrolla una
actividad como escritor
acerca de aspectos técni-
cos y culturales de la tecno-
logía computacional. Es
editor de Apple News, publi-
cación destinada a los usua-
rios de equipos Apple, y
consultor y columnista en la
revista Estrategia.



Cuando usted piensa en el automóvil más fino del mundo piensa en el Rolls Royce...



...Y SI USTED PIENSA EN LOS SUMINISTROS MAS FINOS DEL MUNDO, USTED TIENE QUE PENSAR EN INFORNA.

Representante exclusivo para Chile.

Graham Magnetics 

 **Dysan**
CORPORATION

Felikan 


FLOPPY DISKS

PRECISION
FLEXIBLE DISKS Manufactured by Kodak

"Un compromiso para siempre".

Teatinos 251 Of. 301 Tels.: 696 7968 - 699 4594 - 718922

Sucursales: Huérfanos 1052 Local 27

Agustinas 1035 (Galerías Crillón)



SORTS: UNA REFERENCIA COMPLETA Y DEFINITIVA

Héctor Miranda Riquelme

Por un lamentable incidente de última hora, en el número anterior de Microbyte se introdujeron diversos errores de composición de textos en la primera parte de este artículo sobre herramientas de ordenamiento. Dada la importancia de este tema, hemos decidido refundir las dos partes de que estaba compuesto y publicarlo en extenso en esta oportunidad.

Debido a algunas discrepancias que han surgido en cuanto a la designación y explicación de ciertas rutinas de sort aparecidas en números anteriores de MICROBYTE, hemos elaborado esta referencia sobre las técnicas más conocidas. Aunque los métodos que aquí se entregan han sido tratados en otras oportunidades, queremos despejar las dudas aparecidas, con el objeto de unificar criterios, contribuyendo de esta forma a un mejor y más acabado conocimiento de esta importante técnica computacional y, al mismo tiempo, contribuir al entendimiento entre nuestros lectores.

En este artículo investigaremos algunas técnicas de sort en una tabla singular, usando algoritmos standard y el lenguaje Basic. Las rutinas desarrolladas deberían trabajar en cualquier computador y requieren tan sólo mínimas modificaciones para su implementación en cualquier versión de Basic. Comenzaremos por revisar los métodos más simples —y a la vez más lentos— para introducir el primer método más perfeccionado, el Shell Sort, concluyendo con dos algoritmos más rápidos: el Heap Sort y el Quicksort.

Conceptos Generales

Cuando una lista de items es rearmada en otra secuencia, se dice que la lista ha sido "ordenada" (No es ésta una palabra que entregue cabalmente todo el significado que tiene el vocablo "sorted"; usaremos, con el perdón de los puristas, el verbo "sort" castellanizado, como por ejemplo "sorteada"). La necesidad de algún otro orden puede ser mostrada considerando una lista de nombres, direcciones y números de teléfono. Supongamos que la lista se ha confeccionado preguntando estos datos a 100 personas distintas. Dicha lista no estaría en ningún tipo de orden usable; sin embargo, podría ser re-armada en or-

den alfabético por apellido. Es importante notar que los items relacionados de dirección y números de teléfono también serán rearmados.

Para investigar el "sorting", es útil descartar cualquier parte o acción común y concentrarse en

Intercambiar dos elementos en un arreglo es análogo a intercambiar dos cajones en un escritorio.

aquellos elementos únicos al proceso de sorting. Dado que el proceso de sorting se basa en el campo de información que es ordenado (el nombre, por ejemplo), la presencia de otros campos no es importante. Un ordenamiento está basado en la comparación de dos cosas diferentes. Estas dos cosas pueden ser nombres (como en el ejemplo) o números. La lista podría ordenarse desde la Z hasta la A tan fácilmente como en la forma tradicional. En esta investigación consideraremos una lista de números a ser "sorteada" en secuencia ascendente. Los ejemplos de programación están escritos en Basic, pero pueden ser implementados fácilmente en otros lenguajes.

Una lista de números está en orden ascendente cuando cada par de números adyacentes lo está. Consideremos la siguiente lista:

4	10	16	22	28
---	----	----	----	----

Puesto que $4 < 10$ y $10 < 16$ y $16 < 22$ y $22 < 28$, la lista está en orden ascendente. Si cada par de elementos en la lista está desordenado, la lista lo está.

4	10	22	16	28
---	----	----	----	----

Aquí se ha dado el caso, puesto que $4 < 10$, $10 < 22$, pero $22 > 16$. Para estar en orden descendente, se requiere simplemente el reverso de la lista de más arriba y el reverso de los signos $<$ y $>$.

Para escribir ejemplos de programación significativos existe una tarea de soporte que debe ser definida. Para tener una lista a "sortear" es necesario tener una rutina escrita que proporcione dicha lista. En Basic esto puede ser de la siguiente forma:


```

10 N = 12
20 DIM A (N)
30 FOR I = 1 TO N
40 A (I) = INT (1 + 1000 * RND (1))
50 NEXT I

```

Esta rutina construirá una tabla de números con N elementos (en este caso N es igual a 12). Cada número estará en el rango de 1 a 1000. Los números estarán en una secuencia aleatoria. Se asumirá que esta rutina está presente en todos los programas Basic futuros.

Los Métodos Básicos – El Bubble Sort

La definición de orden sugiere el primer método de sorting: un programa escrito para "caminar" sobre la lista paso a paso desde el primer hasta el último elemento y corregir cualquier par de elementos que se encuentren fuera de orden.

```

120 FOR I = 1 TO N
140 IF A (I) ≤ A (I + 1) THEN 190
160 REM INTERCAMBIA LOS DOS ELEMENTOS
190 NEXT I

```

No es práctico permitir la comparación de A (N) con A (N + 1), así es que la línea 120 debe ser cambiada para permitir sólo N – 1 comparaciones.

Si la comparación en 140 es verdadera, los elementos están en orden, y no es necesaria ninguna acción. Pero si la comparación es falsa, la línea 160 debe ser la rutina que corrija el orden de los números.

Intercambiar dos elementos en un arreglo es análogo a intercambiar dos cajones en un escritorio. Uno de los cajones es sacado y puesto temporalmente sobre la mesa; se saca el segundo cajón y se coloca en el lugar vacante del primer cajón. Se recupera entonces el primer cajón de la mesa y se pone en el lugar vacante del segundo cajón. Ejecutando 3 movimientos, dos elementos son intercambiados. La línea 120 se cambia y la línea 160 es reemplazada con la rutina de intercambio.

```

120 FOR I = 1 TO N - 1
140 IF A (I) ≤ A (I + 1) THEN 190
160 T = A (I)
170 A (I) = A (I + 1)
180 A (I + 1) = T
190 NEXT I

```

Esto funciona para la lista:

4	10	22	16	28
---	----	----	----	----

pero falla en una lista como ésta:

20	14	28	22	16	10
----	----	----	----	----	----

Después de que nuestra rutina ha operado en esta lista, el resultado es:

14	20	22	16	10	28
----	----	----	----	----	----

La lista aún no está en orden. El 10 al final de la lista debería ser el primero de ella. El 10 se movió hacia allí sólo una posición, así es que si la rutina es ejecutada cuatro veces más, el 10 estará en su lugar apropiado. Ningún elemento en la lista puede estar a más de cinco movidas del lugar a donde pertenece. En general, una lista de N elementos no requerirá más de N – 1 ejecuciones de la rutina.

El primer método de sort queda entonces:

```

100 REM SORT – METODO 1
110 FOR J = 1 TO N - 1
120 FOR I = 1 TO N - 1
140 IF A (I) ≤ A (I + 1) THEN 190
160 T = A (I)
170 A (I) = A (I + 1)
180 A (I + 1) = T
190 NEXT I
200 NEXT J

```

Para comparar la performance de este método con listas de diferente tamaño y contra otros métodos, se debe idear una cuenta de actividades simple. El número de comparaciones en la línea 140 y el número de veces que dos elementos son permutados en 160 hasta 180, son ambos contados e informados al final de la rutina.

```

100 REM SORT – METODO 1
110 FOR J = 1 TO N - 1
120 FOR I = 1 TO N - 1
130 C = C + 1
140 IF A (I) ≤ A (I + 1) THEN 190
150 S = S + 1
160 T = A (I)
170 A (I) = A (I + 1)
180 A (I + 1) = T
190 NEXT I
200 NEXT J
900 PRINT "NUMERO DE COMPARACIONES: ";C,"NUMERO DE PERMUTACIONES: ";S

```

Esto llevará el número de comparaciones y permutaciones en cada método. También es útil tomar el tiempo de cada sort.

El análisis del método uno indica que el número de comparaciones C será (N – 1) (N – 1). El número de permutaciones S no puede ser más grande que C, pero el número exacto dependerá de la ubicación original de los elementos. En cada paso de comparación/permutación, un elemento fuera de orden es movido una posición más cerca de su posición ordenada. Durante cada paso de la lista, los números parecen ondear mientras se

mueven todos un paso más cerca de la posición a la que pertenecen.

Un ejemplo del sort método 1 se da para una lista corta de números.

Paso 1	16	30	86	10	56
	^	^			
		^	^		
			^	^	
			10	86	
				^	^
				56	86
Paso 2	16	30	10	56	86
	^	^			
		^	^		
		10	30		
			^	^	
				^	^
Paso 3	16	10	30	56	86
	^	^			
	10	16			
		^	^		
			^	^	
				^	^
Paso 4	10	16	30	56	86
	^	^			
		^	^		
			^	^	
				^	^
Final	10	16	30	56	86

El análisis del proceso anterior apunta hacia un interesante hecho. Al final del primer paso, el 86 es movido al fin de la lista. En el paso 2, el 56 es comparado con el 86 nuevamente; sin embargo, debe ser el número más grande en la lista, o si no el paso 1 habría empujado algún otro número al

Los distintos métodos de sort buscan reducir el número de comparaciones y permutaciones.

fondo. Esto significa que al final del paso 1, el número más grande en la lista es empujado hacia el fondo y no necesita ser chequeado nuevamente. El paso 2 podría haber considerado que la lista era un elemento más corta. Al final del paso 2, el elemento más grande de la lista restante habría sido empujado al final de esta lista más corta, esto es, a la penúltima posición en la lista original.

La lista de números podría ser tratada como una serie de listas sucesivamente más pequeñas, con el elemento más grande puesto al final de la lista en cada paso. Este efecto de desplazamiento puede ser visualizado como pequeños elementos livianos burbujeando hacia el tope de la lista. En consecuencia, este método es a veces llamado Sort de la Burbuja o Bubble Sort. Los cambios al método uno ocurren en las líneas 110 y 120:

```

100 REM SORT-METODO 2
110 FOR J = N - 1 TO 2 STEP - 1
120 FOR I = 1 TO J
130 C = C + 1
140 IF A(I) ≤ A(I + 1) THEN 190
150 S = S + 1
160 T = A(I)
170 A(I) = A(I + 1)
180 A(I + 1) = T
190 NEXT I
200 NEXT J

```

En el método 1, el número de comparaciones era calculado como $N - 1$ comparaciones cada $N - 1$ pasos. El método 2 requiere la sumatoria de $N - 1$ comparaciones con $N - 2$ comparaciones con $N - 3$ comparaciones hasta 1 comparación. Démos vuelta esta secuencia y obtendremos la suma de 1 más 2 más 3 más... más $N - 2$, más $N - 1$ comparaciones. La fórmula para la suma de los primeros m números naturales es $m(m + 1)/2$. El número de comparaciones en el método dos es entonces: $(N - 1)N/2$. Esto es aproximadamente la mitad del número de comparaciones requeridas por el método uno. Por lo tanto se acortará el tiempo requerido para sortear una lista.

Las permutaciones todavía ocurren entre elementos adyacentes, y no es posible ningún perfeccionamiento en el número de permutaciones. El número máximo ha sido reducido, puesto que las permutaciones no pueden exceder a las comparaciones.

Hay aún otra mejoría que puede efectuarse. Si la lista está en orden al comenzar, el número de comparaciones será el mismo, aún cuando el número de permutaciones será cero. Si una bandera se introduce para detectar el hecho de que no han ocurrido permutaciones durante el paso más reciente, la rutina puede ser terminada. Los cambios al método dos ocurren en las líneas 115, 185, 195 y 210.

```

100 REM SORT-METODO 3
110 FOR J = N - 1 TO 2 STEP - 1
115 F = 0
120 FOR I = 1 TO J
130 C = C + 1
140 IF A(I) ≤ A(I + 1) THEN 190
150 S = S + 1
160 T = A(I)
170 A(I) = A(I + 1)
180 A(I + 1) = T
185 F = 1
190 NEXT I
195 IF F = 0 THEN 210
200 NEXT J
210 REM SORT COMPLETO

```

El sort de inserción

El método tres ha sugerido que las comparaciones deben estar limitadas por los elementos fuera de orden. Consideremos la analogía de ordenar a mano un archivo índice de tarjetas. Las tarjetas no

sorteadas se colocan en una pila; la primera tarjeta se pone en el cajón. La segunda tarjeta es insertada en el cajón, ya sea detrás de la primera tarjeta o al frente de ella, como se requiera, para mantener la secuencia apropiada. La siguiente tarjeta es chequeada con la última tarjeta en el cajón. Si es más grande que dicha tarjeta, se coloca después de ella. Si no, es chequeada contra la siguiente tarjeta hacia el frente. Si cada tarjeta es puesta en el cajón encontrando su ubicación apropiada e insertándola entonces, el número de comparaciones debe ser mínimo.

Los pasos de programación necesarios para implementar este método requieren el concepto de agregar un elemento a la lista.

```
110 FOR J = 2 TO N
120 REM INSERTA EL J-ésimo ELEMENTO EN EL
    RANGO 1 A J-1
200 NEXT J
```

Al completar el loop FOR-NEXT que depende de J, cada elemento de la lista es insertado en la lista creciente hasta que todos los elementos están en la lista y en orden. La inserción puede ser llevada a cabo usando la técnica de intercambio desarrollada más arriba, excepto que el orden es desde atrás hacia adelante. El método de inserción comprende usar otra variable I como subíndice, para examinar hacia atrás la lista, comparando el I-ésimo elemento y el (I-1)-ésimo elemento, permutando el nuevo J-ésimo elemento hasta que se encuentre su lugar apropiado.

```
110 FOR J = 2 TO N
120 I = J
130 C = C + 1
140 IF A(I-1) ≤ A(I) THEN 210
150 S = S + 1
160 T = A(I)
170 A(I) = A(I-1)
180 A(I-1) = T
190 I = I - 1
200 IF I > 1 THEN 130
210 NEXT J
```

En tanto la variable I podría ser controlada con un loop FOR-NEXT, es más claro codificar la actividad en I directamente.

Se puede introducir una mejora comprendiendo la analogía de que la tarjeta no es insertada hasta que se haya encontrado su lugar apropiado. Esto puede ser llevado a cabo poniendo primero el J-ésimo elemento en T, luego moviendo A(I-1) a A(I) en lugar de permutar. Cuando se encuentra el lugar apropiado, T es movido hacia A(I). Este cambio comprende las líneas 125, 140, 160, 170, 180, 210 y 220.

```
110 FOR J = 2 TO N
120 I = J
125 T = A(I)
130 C = C + 1
140 IF A(I) ≤ T THEN 210
```

```
150 S = S + 1
180 A(I) = A(I-1)
190 I = I - 1
200 IF I > 1 THEN 130
210 A(I) = T
220 NEXT J
```

Nótese que la cuenta para el número de permutaciones S es ahora falseada cuando se compara con los métodos previos; sin embargo, dado que éste aún mueve los elementos solamente una ubicación más cerca en cada permutación, el número de permutaciones no variará con respecto a los métodos de más arriba.

Predecir el número de comparaciones ya no es un cálculo directo. Si la lista original está en orden, este método hará una pasada a través de la lista y acumulará N-1 comparaciones. El número máximo de comparaciones y permutaciones ocurrirá cuando cada elemento agregado a la lista deba "colarse" hasta el comienzo de ella. Esto ocurre cuando la lista está en orden inverso y el número de comparaciones será 1 + 2 + 3 + ... + (N-1). Esto es (N-1)N/2, lo mismo que para el método tres.

Un ejemplo del método cuatro, usando los mismos elementos que en el método uno, es el siguiente:

Paso 1	16	30			
	^	^			
Paso 2	16	30	86		
		^	^		
Paso 3	16	30	86	10	
			^	^	
			10	86	
			^	^	
			10	30	
	^	^			
	10	16			
Paso 4	10	16	30	86	56
				^	^
				56	86
				^	^
Final	10	16	30	56	86

El Shell Sort

Hasta este punto, todas las modificaciones y mejoras han sido dirigidas a reducir el número de comparaciones. El número de permutaciones ha permanecido invariado. El método cuatro ha reducido el esfuerzo de la permutación, pero el número de posiciones que un elemento se mueve es el mismo. Este es un resultado de los intercambios lado a lado. El siguiente conjunto de mejoras debe comprender el número de intercambios. Véase el siguiente ejemplo:

34	53	73	33	31	11	56	16	30	86	45	64	10
----	----	----	----	----	----	----	----	----	----	----	----	----

El 10 debe viajar a través de cada posición

Continúa en pag. 39

COMPUTER CLUB

Envíe sus colaboraciones a:
Computer Club
Revista Microbyte
Huelén 164 - 2º piso
Santiago

Computer Club es una sección escrita fundamentalmente por ustedes los lectores. En ésta se incluyen todos los aspectos de la microcomputación, desde programas de juegos, utilitarios a programas administrativos para todos los microcomputadores.

Los programas a publicar pueden ser en Basic, código de máquina o cualquier otro, pero al enviar su colaboración asegúrese de:

- acompañar un cassette o disco para verificar el buen funcionamiento de su programa.
- incluir una breve descripción de qué es lo que hace el programa y cómo.
- en lo posible incluir un listado por impresora. El listado debe ser claro como para reproducirlo; si su cinta no es nueva, imprima enfatizado.
- que los caracteres gráficos o en video inverso aparezcan claramente en el listado o de lo contrario incluya líneas REM describiéndolos.

Todas las colaboraciones publicadas serán pagadas a razón de \$ 2.500.

28 Atari:
30 Commodore:
33 CP/M.
35 Open File:

Buzo.
Procesador.
Instalaciones.
Cartas del lector.

El Buzo

Héctor Heredia, de Arica, juntó su pasión por los deportes náuticos con su hobby de coleccionar tesoros, tener aventuras y jugar con el computador. El resultado es este programa que para algunos resultará inquietantemente extenso de tipear pero pueden contar con nuestra garantía de que vale la pena.

Héctor nos cuenta que para el desarrollo de este programa se basó en un juego para construir aventuras por los usuarios llamado "Omniventure". El creó

nuevas escenas y produjo su propio juego.

El tema del juego es que un buzo que está en el fondo del mar debe llegar a la superficie recogiendo la mayor cantidad de ánforas con tesoros.

Para controlar al buzo se utiliza el joystick tratando de evitar a los monstruos marinos que lo acechan. Además, espesos arrecifes de coral le impiden el tránsito, por lo que debe ubicar hachas desechables (sólo sirven para un coral) y arpones

para defenderse de los monstruos.

El resto de la historia ya todos la conocen. El juego es tan difícil que a pesar de que el buzo tiene tres vidas, cuesta mucho llegar a la superficie intacto. En fin, el típico juego frustrante para la mayoría de nosotros que nunca pasamos de la tercera pantalla de PAcMan y envíanlo para los demás. Que se diviertan **M**

```

1 REM *****
2 REM **EL BUZO(AVENTURA MARINA) **
3 REM **creado por Hector Heredia **
4 REM ** para MICROBYTE **
5 REM **(basado en un juego de cons**
6 REM **truccion de aventuras de J.**
7 REM **Lemaitre "Omniventure") ** 8 REM ** fecha:5/5/86 **
9 REM *****
80 GOSUB 9000
90 GOTO 9270
100 S=PEEK(632):MOV=0:P=PEEK(POS+MOV(S)):IF NOT P THEN MOV=1:POKE 77,0:GOTO 300
110 IF P=141 THEN MOV=0:GOSUB 10040
120 IF P=2 OR P=3 THEN 6000
130 IF P=198 THEN POSITION 10,22: #6;A$:CARRY=1:MOV=1:SOUND 0,12,12,8
140 IF P=199 THEN POSITION 10,22: #6;B$:CARRY=2:MOV=1:SOUND 0,12,12,8
150 IF P=136 THEN SC=SC+100:POSITION 4,23: #6;SC:MOV=1:SOUND 0,10,10,10
300 IF MOV THEN POKE POS,0:POS=POS+MOV(S):POKE POS,MAN
310 F=PEEK(644):IF F OR CARRY<>1 THEN 400
320 FOR I=5 TO 14:SWG=POS+MOV(I)
330 SOUND 0,1+50,8,8:IF PEEK(SWG)=2 OR PEEK(SWG)=3 THEN POKE SWG,204:SC=SC+25:PO
SITION 4,22: #6;SC
340 FOR T=1 TO SK*2:IF POS(T)=SWG THEN FLAG(T)=0
350 NEXT T:NEXT I:POSITION 10,22: #6;" NADA ":CARRY=0:FOR I=5 TO 14:P=POS+MO
V(I):IF PEEK(P)=204 THEN POKE P,0
360 NEXT I:SOUND 0,0,0,0
400 SOUND 0,0,0,0:IF F OR CARRY<>2 THEN 500
410 FOR I=5 TO 14:SWG=POS+MOV(I)
420 SOUND 0,1+50,8,8:IF PEEK(SWG)=68 OR PEEK(SWG)=69 THEN POKE SWG,204
430 NEXT I:POSITION 10,22: #6;" NADA":CARRY=0:FOR I=5 TO 14:IF PEEK(POS+MOV(I))
=204 THEN POKE POS+MOV(I),0
440 NEXT I:SOUND 0,0,0,0
500 CH=CH+1:POKE 756,CH(CH):IF CH=1 THEN CH=-1
510 MST=MST+1:IF MST>SK*2 THEN MST=1
520 IF NOT FLAG(MST) THEN 600
530 M=POS(MST):M1=INT(M/20):P1=INT(POS/20):MOV=(POS-P1*20>M-M1*20)-(POS-P1*20<M-
M1*20)+20*(P1>M1)-20*(P1<M1)
540 P=PEEK(POS(MST)+MOV):IF P AND P<>MAN THEN 600
550 POKE POS(MST),0:POS(MST)=POS(MST)+MOV:POKE POS(MST),COL(MST)
570 IF POS=POS(MST) THEN 6000
600 GOTO 100
6000 P=0:SOUND 0,72,12,10:MEN=MEN-1:COLOR 0:PLOT 16+MEN,0:POKE POS,204:FOR I=1 T
O 111:NEXT I
6010 POKE POS,203:FOR I=1 TO 222:NEXT I:SOUND 0,0,0,0:IF MEN THEN 6200
6020 POSITION 5,4: #6;"FIN DE JUEGO":POSITION 4,6: #6;"PRES.FUEGO"
6030 IF PEEK(644) THEN 6030
6040 IF SC=HS THEN HS=SC
6050 GOTO 90

```

```

6200 POSITION 3,5:? #6;"MURIO EL BUZO!":FOR I=0 TO 10:FOR I1=0 TO 70:SOUND 0,I1,
10,10-I:NEXT I1:NEXT I:GOTO 10040
9000 DIM MOV$(39):RESTORE 9001:FOR I=1 TO 39:READ A:MOV$(I,I)=CHR$(A):NEXT I
9001 DATA 104,104,133,215,104,133,214,104,133,217,104,133,216,104,133,218,104,17
0,160,0,177,214,145,216
9002 DATA 200,208,4,230,215,230,217,202,208,242,198,218,16,238,96
9008 DIM AS$(9),BS$(9),CS$(7),CH(1),MOV(15),FLAG(20),POS(20),COL(20)
9010 AS$="ARPON":BS$="HACHA":CS$=" EL"
9020 GRAPHICS 18:POSITION 7,4:? #6;CS$:"":POSITION 7,6:? #6;"BUZO"
9030 CH(0)=PEEK(106)-8:CH(1)=CH(0)-8:D=USR(ADR(MOV$),57344,CH(0)*256,1023)
9040 FOR A=1 TO 13:FOR I=0 TO 7:READ B:POKE CH(0)*256+A*8+I,B:POKE 708,B:NEXT I:
NEXT A
9050 DATA 7,15,30,56,48,0,192,192
9060 DATA 24,60,61,61,85,146,72,132
9070 DATA 195,60,60,231,129,165,195,36
9080 DATA 28,60,62,115,231,62,127,238
9090 DATA 165,165,181,149,149,254,28,62
9100 DATA 1,2,4,8,24,40,200,192
9110 DATA 8,26,60,15,22,36,64,128
9120 DATA 128,57,16,58,184,57,56,146
9130 DATA 24,68,24,60,90,24,36,36
9140 DATA 255,255,255,255,255,255,255,255
9150 DATA 0,0,24,60,60,60,60,60
9160 DATA 137,74,36,192,3,36,82,145
9170 DATA 0,32,80,136,5,2,0,0
9200 D=USR(ADR(MOV$),CH(0)*256,CH(1)*256,1023)
9210 READ A:IF A=-1 THEN RETURN
9220 FOR I=0 TO 7:READ B:POKE CH(1)*256+A*8+I,B:POKE 708,B:NEXT I:GOTO 9210
9230 DATA 2,24,60,60,60,84,147,137,82
9240 DATA 3,66,189,60,231,129,227,34,85
9250 DATA 8,0,56,16,56,56,56,56,16
9260 DATA 9,24,60,24,126,24,24,36,66
9266 DATA 13,0,2,5,136,80,32,0,0,-1
9270 MOV(5)=21:MOV(6)=-19:MOV(7)=1:MOV(9)=19:MOV(10)=-21:MOV(11)=-1:MOV(13)=20:M
OV(14)=-20
9300 GRAPHICS 1:POKE 710,0:POKE 756,CH(0):POSITION 6,0:? #6;"BIENVENIDO":POSITIO
N 6,1:? #6;"*****"
9310 ? #6:? #6;")-este es el buzo"
9320 ? #6:? #6;"&-arpon p/animales"
9330 ? #6;"'-hacha p/alga/coral"
9340 ? #6:? #6;"(-tesoro-100 pts."
9350 ? #6:? #6;";CHR$(34);"-pulpo-25"
9360 ? #6:? #6;"#-cangrejo gigant-25"
9370 ? #6:? #6;"$-PAREDES DE CORAL"
9380 ? #6:? #6;"%-MATAS DE ALGAS"
9390 TRAP 9390:? CHR$(125);"SELECC.NIVEL.1-10 ";;INPUT SK:IF SK<1 OR SK>10 OR IN
T(SK)<>SK THEN 0
10000 TRAP 4444:GRAPHICS 17:POKE 756,CH(0):COLOR 173:PLOT 1,1:DRAWTO 18,1:COLOR
10:PLOT 0,20:DRAWTO 19,20
10010 PLOT 0,0:DRAWTO 0,23:DRAWTO 19,23:DRAWTO 19,0:POSITION 1,21:? #6;"hs:";HS:
POSITION 1,22:? #6;"sc:0"
10020 POSITION 1,0:? #6;CS$;"/BUZO":POSITION 10,21:? #6;"LLEVA: ":POSITION 11,22:
? #6;"NADA"
10030 POKE 708,103:POKE 709,215:POKE 710,31:POKE 711,40:POKE 712,192
10035 COLOR 137:PLOT 17,0:PLOT 18,0:MEN=3:SC=0
10040 COLOR 32:FOR I=2 TO 19:PLOT 1,I:DRAWTO 18,I:NEXT I:FOR I=0 TO 20:FLAG(I)=1
:POS(I)=77:NEXT I
10050 FOR I=0 TO 150:COLOR INT(2*RND(0)+4):PLOT INT(18*RND(0)+1),INT(18*RND(0)+2
):NEXT I:COLOR 32:PLOT 5,19
10060 DRAWTO 15,19:FOR I=0 TO 10-SK:COLOR INT(2*RND(0)+134):PLOT INT(18*RND(0)+1
),INT(18*RND(0)+2):NEXT I
10070 COLOR 168:FOR I=0 TO 6-SK/2:PLOT INT(18*RND(0)+1),INT(18*RND(0)+2):NEXT I
10080 SCR=PEEK(88)+256*PEEK(89):TOP=SCR+40:POS=SCR+389:MAN=201:POKE POS,MAN
10090 FOR I=1 TO SK*2:COL(I)=INT(2*RND(0)+2)
10100 POS(I)=INT(360*RND(0)+40+SCR):P=PEEK(POS(I)):IF P=74 OR P=2 OR P=3 OR P=MA
N THEN 10100
10110 POKE POS(I),COL(I):NEXT I:GOTO 100

```



```

250 goto 200
260 u=0:rem -----corregir-----
270 print "CORREGIR "
280 print " "
290 print "F1 regresa al menu C2 corregir "
300 rem--- loop ---
310 rem
320 print " "
330 for t=1 to 15:print " "
340 if u<0 then u=0
350 print " ":if t=u-3 to u+3
360 if t<0 then print:goto 350
370 if t=u then print "b$(t) " :goto 390
380 print b$(t)
390 next t
400 gets$:if $="" then 400
410 if $=" " then u=u-1:goto 320
420 if $=" " then u=u+1:goto 460
430 if $=" " then return
440 if $="c" then gosub 480:goto 270
450 goto 400
460 if u>k then u=u-1:goto 400
470 goto 320
480 print "b$(u) ":print " " :rem ---loop---
490 u=pos(0)
500 geta$:if a$="" then 500
510 if a$=" " then goto 500
520 if a$=" " then goto 500
530 if a$=" " then goto 500
540 if a$=" " then goto 500
550 if a$=" " then goto 500
560 if a$=" " then nnn=k:k=u:gosub 1230:k=nn:return
570 if a$=" " then goto 500
580 if a$=chr$(34) then goto 500
590 if a$=chr$(148) then goto 500
600 if b$(u)="" and a$=chr$(20) then goto 500
610 if a$=chr$(20) then b$(u)=left$(b$(u),len(b$(u))-1):u=u-ma*2:goto 650
620 if a$=chr$(13) then i=20:gosub 1350:print " ":return
630 if w>78 then goto 500
640 b$(u)=b$(u)+a$
650 print a$:" " :i=i+1:gosub 1350:w=w+ma:goto 500
660 rem -----cargar/grabar-----
670 print "CARGAR/GRABAR "
680 print " "
690 input "cargar o grabar c " :cc$:if cc$="c" then goto 720
700 if cc$="g" then goto 790
710 goto 690
720 input "nombre " :cc$
730 open 5,0,4,cc$+ ".s,r ":k=k+1:dc=0
740 get#5,a$:b$(k)=b$(k)+a$:if a$=chr$(13) then gosub 1390
750 if st=64 then goto 780
760 if len(b$(k))>80 then b$(k)="" :goto 780
770 goto 740
780 close 5:gosub 1420:return
790 input "nombre " :cc$
800 open 1,8,2,"00:" + cc$ + ".s,w"
810 for t=0 to k-1:print#1,b$(t):next t
820 close 1:gosub 1420:return
830 rem -----imprimir-----
840 print "IMPRIMIR "
850 print " "
860 input "Imprimir (S/N) s " :x$
870 if x$="s" then goto 900

```

```

880 ifx$="n"thenreturn
890 print "NO MAS MEMORIA!";goto860
900 open1,4,7:vx=0
910 fort=0tok
920 print#1,b$(t):vx=vx+1:ifvx=60thengosub1450
930 nextt
940 close1
950 return
960 rem -----pantalla-----
970 print "PANTALLA";
980 b$(K)="":u=0:ma=1:ifK>5999thengosub1410:goto980
990 geta$:ifa$=""then990
1000 ifa$=" "thengosub70:goto970
1010 ifa$="a"thengoto990
1020 ifa$="b"thengoto990
1030 ifa$="c"thengoto990
1040 ifa$="d"thengoto990
1050 ifa$="e"thengoto990
1060 ifa$="f"thengosub1230:goto980
1070 ifa$="g"thengosub1270:goto980
1080 ifa$="h"thengosub1170:goto980
1090 ifa$=chr$(148)thengoto990
1100 ifa$=chr$(34)thengoto990
1110 ifb$(K)=" "anda$=chr$(20)thengoto990
1120 ifa$=chr$(20)thenb$(K)=left$(b$(K),len(b$(K))-1):u=u-ma*2:goto1160
1130 ifa$=chr$(13)thenii=20:gosub1350:print " ";chr$(13):ifa$=" "K=K+1:goto980
1140 ifu>78thengoto990
1150 b$(K)=b$(K)+a$
1160 printa$;" ";ii=1:gosub1350:u=u+ma:goto990
1170 rem ---control l---
1180 print "L":fort=0tok-1:print "L":b$(t)
1190 gets$
1200 ifs$="*"thengoto1220
1210 nextt
1220 print "L":return
1230 rem ---control c---
1240 a$="":print " "
1250 forc=1to(80-len(b$(K)))/2:ia$=a$+" "inextc:ia$=a$+b$(K):b$(K)=a$
1260 printb$(K):print "C":K=K+1:return
1270 rem ---control s---
1280 K=K-1:j=1
1290 ifmid$(b$(K),j,1)<>" "thengoto1310
1300 b$(K+1)=b$(K+1)+" ":j=j+1:goto1290
1310 ifmid$(b$(K),j,1)=" "thengoto1330
1320 j=j+1:b$(K+1)=b$(K+1)+" ":goto1310
1330 printb$(K+1):K=K+2:print "S"
1340 return
1350 rem ---sonido---
1360 pokej1,15:pokej2,33:pokej3,15:pokej4,40:pokej5,200:fort=1to11:nextt
1370 pokej2,0:pokej3,0
1380 return
1390 rem sub
1400 b$(K)=left$(b$(K),len(b$(K))-1):K=K+1:return
1410 print "NO MAS MEMORIA!":fort=1to1000:nextt:gosub70:return
1420 rem error disketera
1430 open15,8,15:input#15,a$,b$,c$,d$:print "E"tab(10)a$,"b$","c$","d$
1440 close15:fort=1to1000:nextt:return
1450 rem ---proxima pagina---
1460 ford=1to8:print#1:nextd:vx=0:return
ready.

```


Instalación especial de su Wordstar

Héctor Saavedra.

Consideraciones sobre la instalación

El procedimiento de "instalación" es para que WORDSTAR funcione con su terminal e impresora específicas. Este procedimiento permite que una versión de Wordstar funcione con una gran variedad de equipos de computación. Para la mayor parte de los casos, la instalación se lleva a cabo haciendo selecciones en menus. Para terminales e impresoras excepcionales, y para obtener refinamientos en la instalación, es posible hacer todo tipo de "parches".

Modificación por medio de "parches"

La modificación por medio de parches se utiliza al instalar Wordstar para terminales que no están incluidos en los menus de instalación y para impresoras que necesitan cuerdas de control especiales y para lograr refinamientos en la instalación, tales como incluir características especiales de impresoras de margarita o de teletipo, o permitir utilizar la última columna en la última línea en la pantalla.

Dado que la instalación básica se lleva a cabo con el programa INSTALL, la mayor parte de los usuarios no necesitarán del parchado.

"Parchar" significa modificar Wordstar cambiando los contenidos de bytes específicos de WS.COM. El parchado puede llevarse a cabo con el parchador de INSTALL o con su propio debugger (DDT) que se encuentra en la mayor parte de los sistemas.

Herramienta de Depuración Dinámica (DDT)

La herramienta de depuración dinámica se utiliza para examinar y depurar programas en lenguaje de máquina al escribir

DDT d: WS.COM <cr>

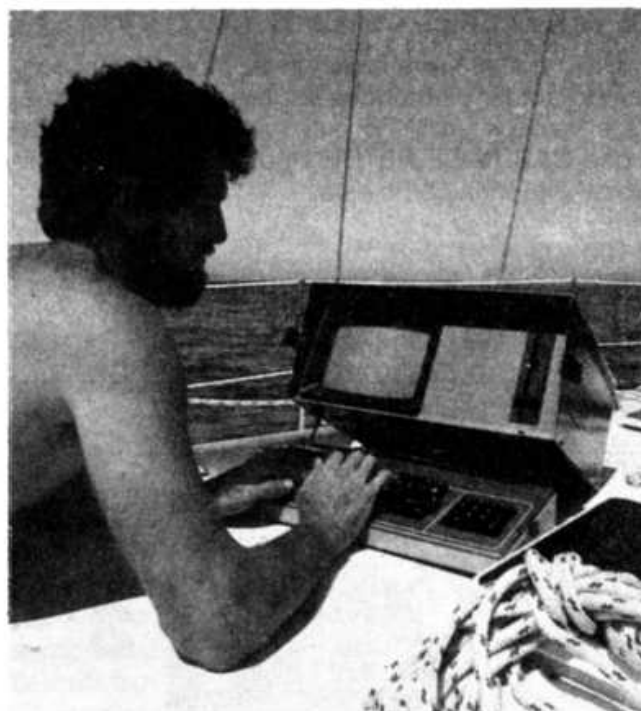
Se carga DDT en memoria y también WS.COM para examinarlo, modificarlo o extenderlo.

En este punto anote el valor de la 1ª dirección que aparece en pantalla, pues la necesitará más adelante. (Fig. 1)

```
B> DDT WS.COM
DDT VERS 2.2
NEXT PC
4500 0100
_
_
```

Figura 1

Cuando se ha cargado en memoria (con Wordstar), DDT está listo para aceptar órdenes.



La petición de orden consta de un guión. El guión indica que DDT está esperando órdenes DDT.

Función	Letra tecleada
Ensambla Instrucciones	A
Visualizar Memoria	D
Listar Instrucciones	L
Asignar Memoria al Valor	S

Instrucciones del Ensamblador A# <cr>

Introducir las instrucciones de lenguaje ensamblador comenzando en la dirección Hexadecimal (#) especificada. No habrá espacio entre la letra de la orden (A) y la dirección que se especifique.

```
-A 100 <cr>
0100 MOV A,C <cr>
```

Para finalizar la entrada de instrucciones del lenguaje ensamblador, simplemente digite un punto seguido por CARRIAGE RETURN en vez de una instrucción válida.

```
0101 . <cr>
```

Visualizar D# <cr>

Visualizar la memoria permite que se pueda especificar la dirección de comienzo.

```
-DO500 <cr>
```

Listar la memoria

L < cr >

Esta orden lista el contenido de la última dirección listada.

Inicializar memoria

S # < cr >

La orden de inicializar memoria permite visualizar y cambiar opcionalmente el contenido de la memoria, a partir de la posición especificada inmediatamente a continuación de S

DDT muestra una posición de memoria, su contenido actual y entonces espera que se introduzca el valor de un BYTE, o simplemente que se presione CARRIAGE RETURN para dejar el contenido de memoria como está.

S 100 < cr >

01000 C3 3D < cr > C3 se convierte en 3D

0101 30 < cr > lo deja igual

0102 20 • < cr > Termina

Tabla 1

DIRECCION	VALOR ON OFF	DESCRIPCION
02AA	FF, 00	ESCRIBE EN LA ULTIMA COLUMNA DE LA ULTIMA LINEA
02DD		NUMERO DE LINEAS A SALTAR NORMALMENTE 14H
0360		NIVEL INICIAL DE AYUDA, (0, 1, 2, 3) (H)
0363	FF, 00	EXHIBICION DEL DIRECTORIO (F)
0389	FF, 00	AYUDA GUION (CTRL-O H)
038A	FF, 00	EXHIBE CONTROL (CTRL-O D)
038B	FF, 00	LINEA TABS (CTRL-O T)
038C	FF, 00	GUION SUAVE (CTRL-O E)
038D	FF, 00	FIN PAGINA (CTRL-O P)
038E	FF, 00	ALINEA (CTRL-O J)
0422		CARACTERES CON <BS> AUTOMATICO 9 EN TOTAL
a		
042B		
0690		ESTA DIRECCION INDICA EL TIPO DE IMPRESORA CON LA CUAL WS. TRABAJA
	FF	CUALQUIER IMPRESORA DE TELETIPO STANDARD
	00	IMPRESORA CON <cr>
	01	IMPRESORA DE MARGARITA
0691		NUMERO DE GOLPES EN NEGRITAS (CTRL-P B)
0692		NUMERO DE DOBLE GOLPE (CTRL-B)

En las siguientes direcciones el primer Byte debe contener el largo de la cadena de caracteres a enviar:

Ejemplo:

Para activar el modo inverso en el microcomputador APPLE II se enviará la siguiente cadena de caracteres 1B, 01, 0F. Los que forman una cadena de 3 caracteres, por lo cual el primer Byte de-

De esta forma, la cadena quedará (03, 1B, 01, 0F). Todos los valores deben ser escritos en Hexadecimal.

En la mayor parte de los manuales se encontrarán las cadenas de comando en sistema decimal, los que deberán transformarse a hexadecimal. Para el ejemplo anterior, los valores de la cadena corresponden a: ESC + CTRL-A + CTRL-O o CHR\$ (27) + CHR\$ (01) + CHR\$ (15), donde 27 = 1B, 01 = 01, 0F = 15.

Para anular cualquier cadena de caracteres, es suficiente poner en 0 el primer byte (que indica la cantidad de caracteres a enviar).

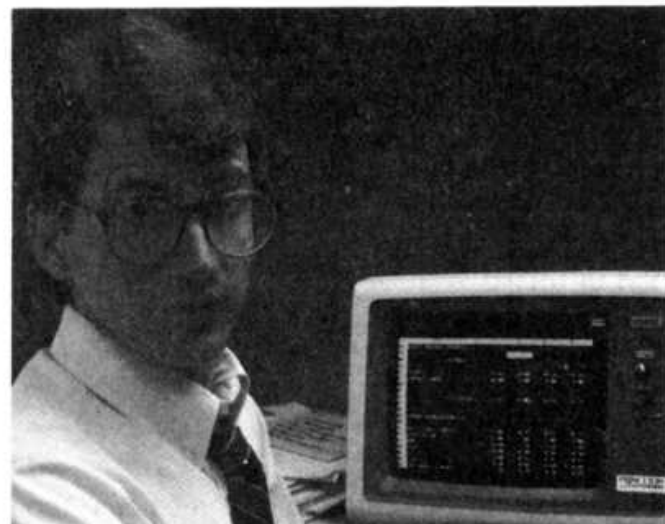
Tabla 2

DIRECCION	HEXADECIMAL Nº DE BYTE DISPONIB.	
06AB	07	CADENA PARA SALTAR 1/2 LINEA
06BF	05	CADENA PARA SUBIR 1 LINEA
06C4	05	CADENA PARA BAJAR 1 LINEA
06C9	05	FUNCION DE IMPRESION DEL USUARIO (CTAL-P Q)
06CE	05	FUNCION DE IMPRESION DEL USUARIO (CTAL-P W)
06D3	05	FUNCION DE IMPRESION DEL USUARIO (CTAL-P E)
06D8	05	FUNCION DE IMPRESION DEL USUARIO (CTAL-P R)
06DD	05	CAMBIO COLOR CINTA ON
06E2	05	CAMBIO COLOR CINTA OFF
06E7	10	CADENA PARA INICIALIZAR LA IMPRESORA
06F8	10	CADENA ENVIADA AL TERMINAR LA IMPRESION DE UN TEXTO

Una vez hecha todas las modificaciones, sólo nos quedará salvar éstas. Siga los siguientes pasos: transforme el valor inicial que nos dio DDT al comienzo en su correspondiente valor decimal - 1 y luego digite:

-G0 < cr >

A > SAVE 68 WS.COM < cr >



OPENFILE

Cartas del lector



EFFECTOS GRAFICOS

Sr. Director

Los siguientes programas muestran diferentes gráficos clásicos del computador ZX-SPECTRUM.

1. El efecto moire

Las posibilidades que los microordenadores ofrecen para la realización de gráficos constituyen un atractivo en la utilización de estos equipos, sólo hay que imprimir el listado y ejecutarlo.

LISTADO:

```
10 OVER 1:PAPER 5:BORDER 1:
CLS
20 LET X1=RND*255
30 LET Y1=RND*175
40 FOR X=0 TO 255 STEP .8
50 PLOT X1,Y1:DRAW X-X1,-Y1
60 PLOT X1,Y1:DRAW X-X1,175-Y1
70 NEXT X
80 FOR Y=0 TO 175 STEP .8
90 PLOT X1,Y1:DRAW -X1,Y-Y1
100 PLOT X1,Y1:DRAW 255-X1,Y-Y1
110 NEXT Y
```

2. Efecto logrado por una sola línea

El gráfico que se presenta se consigue con el programa de una sola línea. Si realmente se desea saber por qué se logran estos resultados, es aconsejable eliminar OVER 1.

LISTADO:

```
1 PLOT 55,27: DRAW OVER
1:120,120,59 3*PI
```

3. Un gráfico tridimensional

Tiene la forma de una huevo en tres dimensiones, no puede decirse que esta figura se obtiene con rapidez sino hay que ser un poco paciente para obtener el gráfico en su pantalla.

LISTADO:

```
10 BORDER 0: PAPER 0: INK 7: CLS
20 FOR X=-100 TO 100
30 LET R=10: LET J=0: LET K=1
40 LET V=R*INT (SQR((10000)-(X*X)))/R)
50 FOR Y=V TO -V STEP -R
60 LET Z=INT (80+30*SIN ((SQR(X*
```

```
X+Y*Y))/12)-.7*Y)
70 IF Z<J THEN GOTO 110
80 LET J=Z
90 PLOT X+110,Z-15
100 LET K=0
110 NEXT Y: NEXT X
```

Ricardo Cortés Tornini
Vallejo 620 Depto. 42
Casilla 34
Copiapó

¿Algún lector conoce otras rutinas gráficas igualmente interesantes?

AVIS-RARA

Sr. Director:

En primer lugar permítame felicitarlo a Ud. y su equipo periodístico por su excelente revista.

He querido dirigirme a Ud. con el fin de darle a conocer que aunque su medio informativo es de muy alta calidad, no puede difundir información para todos los computadores y microcomputadores existentes en el mercado; le pido publicar mi intención de intercambiar información sobre los microcomputadores de Radio Shack específicamente del modelo MC-10 (el cual no es muy común en el país). A los interesados les ruego escribir a:
Salitrera Victoria 3268
Villa Magisterio
Iquique.

Sin otro particular, se despide y agradece su favor.

Marcelo Gálvez Moya
Estudiante

CONTABILIDAD-PC

Señor Director:

El objeto de la presente es agradecerles por la información que me enviaron acerca de los "Peeks" y "Pokes" de las computadoras ATARI y también por haber publicado el programa que les envié desde mi país, Bolivia.

Aprovechando la presente felicito a su persona y a su excelente plantel que compone su excelente revista por la colaboración que nos brindan mediante las diferentes ediciones de la misma.

Les insinúo contactarme con cualquier persona que esté interesada en intercambiar programas de contabilidad para la IBM PC.

Agradeciendo de antemano por su colaboración me despido reiterándoles mis felicitaciones.

Atte.

M. Murillo F.
Casilla 222
Castro, Isla Chiloé
Chile.

Desean contactarse

Juan Pablo Loreto V., de Parcela 7-I, Lo Arcaya, Colina, pregunta por el Computer Club Chile. Este funciona en Alameda 980, 5º piso, fono 382224. Además desea conseguir listados de programas para tipear en el C-16.

Danilo Flores Diaz, de Sub. Pequeñas 102-b
Cerro el Litre, Valparaíso
fono 253998.

Desea contactarse con usuarios de Commodore 64 con Datassette, para intercambio de Software y Cartridges de juego.

Representantes descuidados

He comprado su revista desde el N° 1 y la encuentro muy buena para el nivel nacional.

Soy entusiasta de los Sinclair Spectrum y eso me ha obligado a suscribirme a revistas extranjeras como el ZX, de España, o ZX Computing Monthly de Inglaterra, y al efectuar comparaciones con su Sección, "Aquí Sinclair", veo con pena el descuido que han tenido los representantes en Chile de no aprovechar el espacio que Uds. les brindan. Ese es el motivo por el cual les estoy remitiendo algunas colaboraciones sacadas de la revista inglesa.

- Programa Números Primos
Descompone factorialmente cualquier Número entre 1 y 4.294.967.296 e indica a los números que son primos.
- Copy Microdrive Data
Copia información de Microdrive a Microdrive.
- Microdrive File Analyser
Analiza los programas según sea Basic, Byte, Datat, etc., y longitud de los programas

Microdrive File Sorter

Permite ordenar o clasificar archivos que están en Micro-drive.

Le he dado en esta oportunidad más importancia a los programas para Microdrive, dado a que son poco conocidos en nuestro medio.

En el futuro, les seguiré enviando algunos programas para fortalecer la Sección "Aquí Sinclair", no teniendo ningún interés por el pago que Uds. efectúen por las colaboraciones publicadas.

Sin otro particular, les saluda
atentamente a Uds.,

E. Sergio Lorca Santelices
Casilla 3163 Correo Central
Fono Of. 6963049

una excepción esta vez por dos razones fundamentales. La primera es que los programas son muy buenos y sería poco amable de nuestra parte no compartirlos con ustedes.

La segunda y más importante quizás, es para mostrar que es posible hacer programas de muy buena factura en pocas líneas y que quienes en principio tienen el mayor acceso a rutinas y conocimiento respecto a los distintos equipos son los propios distribuidores de éstos.

Tal como dice nuestro lector, las empresas, y no hablamos tan sólo de Sinclair, han desaprovechado la oportunidad que nosotros les brindamos para, sin ningún costo para ellas, dar a conocer a sus usuarios información respecto a sus máquinas. Esperamos que este llamado los haga despertar de su inercia.

Si bien nuestra política expresa es no publicar programas que hayan aparecido en otras revistas, hemos hecho

```
PROGRAMA NUMEROS PRIMOS
*****
```

```

10 REM factorizacion
20 DATA 2,1,2,2,4
30 DATA 2,4,2,4,6,2,6,4,2,4,6,
6,2,6,4,2,6,4,6,8,4,2,4,2,4,8,6,
4,6,2,4,6,2,6,6,4,2,4,6,2,6,4,2,
4,2,10,2,10
40 RESTORE 20
50 INPUT "Ingrese un numero<=4
294967296:
;n
60 GO SUB 300
70 LET f:=0: LET d:=0
80 FOR b=0 TO 52: READ z: LET
d:=d+z
100 LET q=n/d: IF q<d THEN GO T
O 200
110 IF q=INT q THEN PRINT "=" A
ND NOT f;d;"*";: LET f:=1: LET n=
q: GO TO 100
120 NEXT b
130 RESTORE 30: LET b=4: NEXT b
200 IF NOT f THEN PRINT " is pr
ime";
210 IF f THEN GO SUB 300
220 PRINT : GO TO 40
300 LET h=INT (n/1e5)
310 IF NOT h THEN PRINT n:: RET
URN
320 PRINT h:(STR$ (n-1e5*(h-1)
)(2 TO 1):: RETURN

```

Calculo de Numeros

```

1 is prime
2 is prime
3 is prime
4=2*2
5 is prime
6=2*3
7 is prime
8=2*2*2
9=3*3
10=2*5
11 is prime
12=2*2*3
13 is prime
14=2*7
15=3*5
16=2*2*2*2
17 is prime
18=2*3*3
19 is prime
20 is prime

```

[illegible]

```

11 REM * Microdrive File Sorter
20 GO SUB 100: REM init
30 GO SUB userinput
40 GO SUB initfiles
50 GO SUB readlines
55 GO SUB shellsort
60 GO SUB openfiles
65 GO SUB merge
70 GO SUB closefiles
75 IF NOT end THEN GO TO 5080
85 GO SUB finishoff
90 GO TO 9999
100 REM *****
101 REM * Init *
102 REM *****
110 CLEAR #: CLS #
115 PRINT AT 0,5;"Microdrive File Sorter"
120 PRINT AT 1,8;"W.F.Barnard B.Sc."
127 PRINT AT 2,11;"April 1986"
128 REM limit=max no. lines read from i/p file & sorted
129 REM len=max line length
130 REM a$(limit, len+3)="limit lines of len chars (+3 to hold original length)"
135 DIM f$(3,11): REM 3 microdrive no.s + filenames
140 LET b$="": REM input line
145 LET TRUE=1: REM Boolean values
150 LET FALSE=0
155 LET swa=FALSE: REM used in sort routine
157 LET end=FALSE: REM end of input file.
160 LET userinput=250
165 LET initfiles=400
170 LET readlines=500
175 LET shellsort=600
180 LET openfiles=800
195 LET merge=900
200 LET closefiles=1100
202 LET filename=1300
205 LET finishoff=1200
206 LET old=5: LET new=6: REM stream no.s
210 RETURN
250 REM *****
251 REM * User input *
252 REM *****
255 PRINT "Please type in the name of your file to be sorted in the form: -"
260 PRINT "1fred"
265 PRINT "where 1 is the microdrive number and 'fred' is the filename."
270 GO SUB filename
271 PRINT "Main file = "; b$
272 LET f$(1)=b$
274 PRINT "How many lines to be read and sorted at a time?"
275 INPUT limit
276 PRINT "What is the length of the longest line in your file?"
277 INPUT len: IF len<1 THEN GO TO 275
278 DIM a$(limit, len+3)
280 PRINT "Please type in the names of 2 files, in the same format as above, that can be used for output."
285 GO SUB filename
286 PRINT "Temporary File 1 = "; b$
290 LET f$(2)=b$
295 GO SUB filename
296 PRINT "Temporary File 2 = "; b$
300 LET f$(3)=b$
310 IF f$(1)=f$(2) OR f$(2)=f$(3) OR f$(1)=f$(3) THEN PRINT "Sorry, you can't have the same filenames for input and output!": GO TO 250
350 LET o$f=f$(3): REM oldfile
355 LET n$f=f$(2): REM newfile
360 RETURN

```

```

400 REM *****
401 REM * Init files *
402 REM *****
410 PRINT "Initialising files"
415 PRINT "Opening main file"
420 OPEN #4;"m";VAL f$(1,1);f$(1,2 TO ): REM main file
425 PRINT "Creating a newfile"
430 OPEN #new;"m";VAL n$(1);n$(2 TO ): REM create newfile
440 REM #new;"****": REM with file terminator
450 CLOSE #new
460 RETURN
500 REM *****
501 REM * Read lines from main file *
502 REM *****
504 POKE 23692,0: REM auto scroll
505 PRINT "Reading main file"
510 LET count=0
520 INPUT #4: LINE b$
530 IF b$="****" THEN LET end=TRUE: RETURN
540 LET count=count+1
550 LET a$(count)=b$
560 LET a$(count, len+1 TO )=STR$ LEN b$
570 IF count<>limit THEN GO TO 520
580 RETURN
600 REM *****
601 REM * Shell sort a$( ) *
602 REM *****
605 PRINT "Sorting"
610 IF count<2 THEN RETURN
615 LET shl=1
620 IF shl<count THEN LET shl=shl*2: GO TO 620
630 LET shl=shl/2
640 LET swap=FALSE
650 FOR i=1 TO count-shl
660 IF a$(i)<a$(i+shl) THEN
GO TO 690
670 LET b$a$(i): LET a$(i)=a$(i+shl)
680 LET a$(i+shl)=b$: LET swap=TRUE
690 NEXT i
700 IF swap THEN GO TO 640
710 IF shl>1 THEN GO TO 630
720 RETURN
800 REM *****
801 REM * Open new & old files *
802 REM *****
810 IF count=0 THEN RETURN
815 REM swap new & old filename
820 LET b$o$: LET o$n$: LET n$b$
825 PRINT "Opening old file"
835 PRINT "Opening new file"
840 OPEN #new;"m";VAL n$(1);n$(2 TO )
850 RETURN
900 REM *****
905 PRINT "Merging"
910 IF count=0 THEN RETURN
920 LET ptr=1: REM pointer into a$( )
930 INPUT #old: LINE b$
940 IF b$="****" THEN GO TO 100
950 IF ptr>count THEN GO TO 105
960 IF b$<a$(ptr, TO len) THEN PRINT #new; b$: GO TO 930
970 PRINT #new; a$(ptr, TO VAL a$(ptr, len+1 TO ))
980 LET ptr=ptr+1
990 GO TO 950
1000 REM End of old file - write rest of a$( ) to newfile
1010 FOR i=ptr TO count
1020 PRINT #new; a$(i, TO VAL a

```

```

$(i, len+1 TO )
1030 NEXT i
1040 RETURN
1050 REM End of a$( ) - copy rest
of oldfile to newfile
1060 PRINT #new; b$
1070 INPUT #old; LINE b$
1080 IF b$(">") ***** THEN GO TO 10
60
1090 RETURN
1100 REM *****
**
1101 REM * Close new & old files
*
1102 REM *****
**
1110 IF count=0 THEN RETURN
1120 PRINT #new; *****: REM term
inator
1130 CLOSE #new
1140 CLOSE #old
1145 PRINT "Erasing old file"
1150 ERASE "m"; VAL o$(1); o$(2 TO
)
1160 RETURN
1200 REM *****
1201 REM * Finish off *
1202 REM *****
1210 CLOSE #4: REM main file
1220 CLEAR #1: CLS #
1230 PRINT "Your file" f$(1,2 T
O ) "on microdrive "; f$(1,1)
1240 PRINT "is now sorted in th
e file"
1250 PRINT n$(2 TO ) "on microdr
ive "; n$(1)
1260 RETURN
1300 REM *****
*****
1301 REM * Get filename in form1
fred *
1302 REM * where 1 is microdrive
no. *
1303 REM * and 'fred' is the fil
ename *
1304 REM *****
*****
1310 INPUT LINE b$
1320 IF LEN b$ < 2 OR LEN b$ > 11 TH
EN GO TO 1310
1330 IF b$(1) < "1" OR b$(1) > "6" T
HEN GO TO 1310
1340 RETURN
1400 ERASE "m"; 1; "Sorter"
1410 SAVE "m"; 1; "Sorter" LINE 1
0

```

```

2 REM * Microdrive File Analy
ser
10 GO SUB 100: REM init
20 GO SUB analyse
30 STOP
100 REM *****
101 REM * Init *
102 REM *****
110 CLS #1: CLEAR #
120 INPUT "What is the filename
? "; LINE f$
130 IF LEN f$ = 0 OR LEN f$ > 10 TH
EN GO TO 120
140 INPUT "Which microdrive num
ber is it on? "; md
150 IF md < 1 OR md > 8 THEN GO TO
140
160 LET line=0
165 LET total=0
170 LET maxlen=0
180 LET analyse=500
190 OPEN #4; "m"; md; f$
200 RETURN
500 REM *****

```

```

*****
501 REM * Analyse file. This ro
utine will end in EOF error *
502 REM * unless the last line
of the file is **** *
503 REM *****
*****
510 INPUT #4; LINE a$
520 LET line=line+1
530 LET len=LEN a$
535 LET total=total+len+1: REM
1=<CR>
540 IF len>maxlen THEN LET maxl
en=len
545 POKE 23692,0
550 PRINT INVERSE 1; "#"; line; "L
en="; len; " Max="; maxlen; " Tot=";
total
560 PRINT a$
565 IF INKEY$(">") THEN GO TO 56
5: REM wait if key pressed
570 IF a$(">") ***** THEN GO TO 51
0
580 CLOSE #4
590 RETURN
600 ERASE "m"; 1; "File an"
610 SAVE "m"; 1; "File an" LINE
10

```

```

11 REM * Copy Microdrive Data
File *
12 REM * using Only 1 Drive.
*
14 REM If 2 drives are availa
ble then use the MOVE command.
16 REM * MOVE "m"; 1; "Original"
TO "m"; 2; "Copy" *
20 CLS #1: CLEAR #
30 INPUT "What is the filename
? "; LINE f$
50 INPUT "How many lines in th
e file? "; lin
60 INPUT "What is length of th
e longest line? "; len
65 REM may fail here if file t
oo large to hold in memory
70 DIM a$(lin, len+3)
80 OPEN #4; "m"; 1; f$
90 FOR i=1 TO lin
100 INPUT #4; LINE b$
110 LET a$(i)=b$
120 LET a$(i, len+1 TO )=STR$ L
EN b$
130 NEXT i
140 CLOSE #4
150 INPUT "Place your new cartr
idge in your microdrive and press
ENTER"; LINE c$
160 OPEN #4; "m"; 1; f$
170 FOR i=1 TO lin
180 NEXT i
190 REM if last line in file no
t **** then add it.
200 IF b$(">") ***** THEN PRINT #4
; *****
210 CLOSE #4
220 INPUT "Would you like to ma
ke a backup copy of your file on
cassette (Y/N)? "; b$
230 IF b$="Y" OR b$="y" THEN SA
VE f$ DATA a$( )
240 STOP
1000 ERASE "m"; 1; "Copy file"
1010 SAVE "m"; 1; "Copy file" LIN
E 10

```


para alcanzar su lugar apropiado en la lista.

Consideremos una sub-lista hecha con el 1^{er}, 4^o, 7^o, 10^o y 13^{er} elementos.

34	53	73	33	31	11	56	16	30	86	45	64	10
34			33			56			86			10

Ordenemos esta lista, usando el método cuatro de sort de Inserción y volvamos a poner los elementos.

10			33					34				56			86
10	53	73	33	31	11	34	16	30	56	45	64	86			

Esto resulta en cuatro intercambios en lugar de los 12 originales, para mover el 10 a su correcto lugar en la lista. Este método permite que un elemento en la lista dé saltos a través de muchas posiciones.

Formemos una sub-lista con el 2^o, 5^o, 8^o y 11^{er} elementos, sorteemos la siguiente sub-lista usando el método cuatro y reemplacémosla en la lista completa:

10	53	73	33	31	11	34	16	30	56	45	64	86
	53			31			16			45		
	16			31			45			53		
10	16	73	33	31	11	34	45	30	56	53	64	86

Otra sub-lista de los elementos 3^o, 6^o, 9^o y 12^o:

10	16	73	33	31	11	34	45	30	56	53	64	86
		73			11			30			64	
		11			30			64			73	
10	16	11	33	31	30	34	45	64	56	53	73	86

Dado que una sub-lista conteniendo el cuarto elemento también contendría el 1^{er} elemento, no se necesita formar más sub-listas. Usando el método cuatro para sortear esta lista resultante, hace que sean necesarias menos permutaciones que con la lista original. El proceso intermedio movió la mayoría de los elementos más cerca de su posición final en la lista. La forma general de este método es encontrar un incremento conveniente para los elementos a ser sorteados, luego reducir el incremento y sortear nuevamente. El proceso se repite hasta que el incremento sea uno, en cuyo momento el proceso es el Sort de Inserción Directa del método cuatro. Este método fue llamado Sort del Incremento Decreciente por su autor, Donald Shell. Pero es más conocido como el Shell Sort.

La elección de los incrementos (3 y 1 en el ejemplo) no es arbitraria. Se obtienen excelentes resultados si se hacen las siguientes elecciones. El primer incremento es elegido para que sea la mitad de uno menos que una potencia de 2 que es justo menor que el número de elementos en la lista. Esto es, si N es el número de elementos en la lista, se debe encontrar un K tal que:

$$2^K < N < 2^{K+1}$$

A continuación, elegir el incremento inicial como:

$$L = ((2^K) - 1)/2$$

Este número puede ser calculado directamente usando esta fórmula:

$$L = (2^{\wedge} \text{INT}(\text{LOG}(N)/\text{LOG}(2))) - 1$$

LOG(N) / LOG (2) es necesario cuando el Basic no proporciona una función logaritmo en base 2.

Cada incremento sucesivo es la mitad del incremento previo con descarte de fracciones. Esto siempre resultará en elecciones de la lista siguiente:

1, 3, 7, 15, 31, 63, 127, 255, 511, 1023, etc.

El loop externo del método cinco es el que concierne al incremento descendente:

```
130 L = (2^INT(LOG(N)/LOG(2))) - 1
140 L = INT(L/2)
150 IF L < 1 THEN 300
160 REM SORT DE INSERCIÓN CON INCREMENTO L
290 GOTO 140
300 REM SORT LISTO
```

Después que la pasada con el incremento fijo en 1 se ha completado, el sort está listo. Dentro de este loop externo, un Sort de Inserción similar al método cuatro es usado para sortear la sub-lista:

```
160 FOR J = 1 TO L
170 FOR K = J + L TO N STEP L
180 I = K
190 T = A(I)
200 C = C + 1
210 IF A(I-L) > T THEN 260
220 A(I) = A(I-L)
230 S = S + 1
240 I = I - L
250 IF I > L THEN 200
260 A(I) = T
270 NEXT K
280 NEXT J
```

Combinando las dos fases, obtenemos el producto final:

```
100 REM METODO 5 - SHELL SORT
130 L = (2^INT(LOG(N)/LOG(2))) - 1
140 L = INT(L/2)
150 IF L < 1 THEN 300
160 FOR J = 1 TO L
170 FOR K = J + L TO N STEP L
180 I = K
190 T = A(I)
200 C = C + 1
210 IF A(I-L) > T THEN 260
220 A(I) = A(I-L)
230 S = S + 1
240 I = I - L
```

```

250 IF I > L THEN 200
260 A(I) = T
270 NEXT K
280 NEXT J
290 GOTO 140
300 REM SORT LISTO

```

La performance de este sort está basada en el peor caso. No puede funcionar en ningún caso inferior al método cuatro (Sort de Inserción). Debe mostrar la más alta mejoría sobre el método cuatro cuando los elementos están muy desordenados.

Las comparaciones de tiempo más adelante demostrarán que los primeros cuatro métodos presentados aquí no se consideran útiles para listas con un orden aleatorio. El Shell Sort funciona cerca de diez veces más rápido que el Sort de Inserción. Si la lista está ordenada originalmente, todavía es aproximadamente tan rápido como aquél. El Sort de Inserción es uno de los métodos más rápidos cuando la lista original está ya en orden o cuando se agrega un nuevo elemento a una lista ordenada.

Llegamos a un punto en el cual hemos investigado cinco métodos de sorting. Los primeros cuatro sufren el mismo problema: son muy lentos cuando se aplican a listas en orden aleatorio. El Sort de Inserción es considerado el mejor si la lista original está en relativamente buen orden. Los métodos restantes son más útiles en listas con orden aleatorio. Todos los métodos previos están basados en el Método Uno; por lo tanto, cualquier perfeccionamiento a realizar debe partir de conceptos enteramente nuevos.

El Heapsort

El sexto método es llamado Heapsort, que es un desarrollo de otra aproximación al problema de ordenamiento. La parte esencial de los primeros tres métodos se ha percibido de la siguiente forma: se selecciona el elemento más grande y se pone en la última posición de la lista. El último elemento es permutado con este elemento más grande y el tamaño efectivo de la lista es reducido en uno. El elemento más grande de la lista es localizado comparando sucesivos elementos de ella, reteniendo la dirección del elemento mayor.

Supongamos que el elemento más grande de la lista está en el tope de ella. Entonces ese elemento podría permutarse con el último elemento de la lista y la longitud efectiva de ésta podría reducirse en un elemento. Si el elemento mayor en esta nueva y más corta lista puede ser forzado fácilmente a ubicarse en el tope de ella, también podría ser permutado con el nuevo fin de la lista y la longitud es acortada nuevamente. Si el proceso puede repetirse tantas veces como se desee, la longitud efectiva de la lista se reducirá a un solo elemento y dicho elemento estará correctamente colocado.

Observar cada elemento de la lista para determinar cuál es el más grande requerirá $(N-1)(N-2)$

$(N-3) \dots 2 = (N-1)N/2$ o aproximadamente $N^2/2$ comparaciones. Esto no es mejor que el método dos.

Si fuera posible determinar el elemento mayor sin mirar cada uno de los elementos restantes cada vez, el número de comparaciones podría reducirse.

Esto sería posible si la lista original estuviese compuesta de tres sub-listas con la propiedad que la primera de ellas conste de un solo elemento en la posición 1; y que el primer elemento en las sub-listas dos y tres es el elemento más grande de la sub-lista respectiva. Esto se hace comparando el elemento único de la primera sub-lista con el mayor de los dos elementos tope de las sub-listas dos y tres y posiblemente permutando para asegurarse que el elemento más grande es ahora el elemento singular en el tope de la lista.

Elementos diferentes comparados:

Si la tarea es restringida a tres elementos en posiciones arbitrarias en la lista, se puede simplificar. El segundo elemento se compara con el tercero. El mayor de estos dos elementos se compara entonces con el primer elemento. Esto produce el elemento más grande de los tres. Para mantener el primer elemento como el más grande, simplemente permutamos el primero con el mayor entre el segundo y el tercero si fuere necesario.

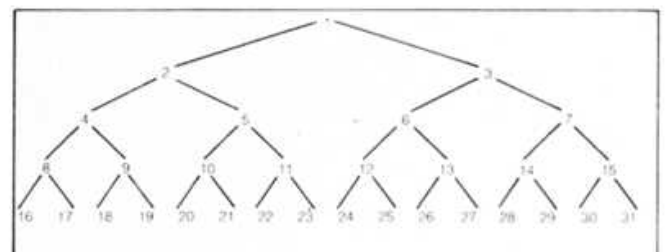
Supongamos que la lista está originalmente ordenada tal que el primer elemento de ella es mayor que el segundo y el tercero. El segundo elemento es mayor que el cuarto y el quinto. El tercer elemento es mayor que el elemento que está dos veces más abajo en la lista y que el elemento adyacente a éste. Esto es expresado por las instrucciones en Basic:

```
A(I) > A(2*I) y A(I) > A((2*I) + 1)
```

Ahora formemos las tres sub-listas de la forma establecida:

1. El elemento en la posición 1.
2. Los elementos en las posiciones 2, 4, 5, 8, 9, 10, 11, 16, ...
3. Los elementos en las posiciones 3, 6, 7, 12, 13, 14, 15, 24, ...

Estas sublistas están formadas desde el elemento raíz en 1 y las dos ramas del árbol:



El primer problema es hacer que la lista original se ajuste a este requerimiento. Esto se efectúa procediendo hacia adelante desde el final de la tabla y considerando cada subconjunto de tres elementos (15,30,31), (14,28,29), ... (1,2,3). En cada subconjunto, hay que asegurarse que el primer elemento de cada trío es el más grande de los tres. La rutina Basic aplicada a un solo trío es:

```
220 REM COMPARA TRIO ASUMIENDO QUE R0
    CONTIENE EL PRIMER ELEMENTO
230 REM EL 2º y 3º ELEMENTOS SON CALCULADOS
250 R1 = R0 + R0
280 IF A(R1) ≥ A(R1 + 1) THEN 300
290 R1 = R1 + 1
300 IF A(R0) ≥ A(R1) THEN 360
310 T = A(R0)
320 A(R0) = A(R1)
330 A(R1) = T
360
```

Si el elemento 7 es permutado con el elemento 14, el elemento 14 puede no ser mayor que el 28 o el 29. Esto significa que cuando ocurre una permutación, esa rama del árbol debe ser chequeada para mantener la organización correcta. Puede ser necesario forzar un elemento hacia las capas inferiores de las ramas hasta que alcance el fondo del "montón" (en inglés "heap"). Con esta corrección, la rutina es:

```
220 REM COMPARA TRIO ASUMIENDO QUE R0
    CONTIENE EL PRIMER ELEMENTO
230 REM EL 2º y 3º ELEMENTOS SON CALCULADOS
240 REM ENCAJA UN ELEMENTO PERMUTADO EN
    EL FONDO DE LA PILA
250 R1 = R0 + R0
260 IF R1 > N THEN 360
270 IF R1 = N THEN 300
280 IF A(R1) ≥ A(R1 + 1) THEN 300
290 R1 = R1 + 1
300 IF A(R0) ≥ A(R1) THEN 360
310 T = A(R0)
320 A(R0) = A(R1)
330 A(R1) = T
340 R0 = R1
350 GOTO 250
360
```

Es útil notar que no se deben chequear más de cuatro capas si el número de elementos es 31. En general, el número de capas a chequear es no más de aquella potencia de 2, justo mayor que el tamaño de la lista. Si el tamaño de la lista es N, el número máximo de capas a chequear es K, donde:

$$2^{K-1} < N \leq 2^K$$

Nótese también que una elección de R0 mayor que la mitad del tamaño de la lista será infructuosa, dado que no se puede formar un trío de elementos. La manipulación debe comenzar con R0 fijo en la mitad del tamaño de la tabla. El proceso

completo de llevar la lista a su necesario orden original es entonces:

```
200 FOR I = INT(N/2) TO 1 STEP -1
210 R0 = I
220 REM COMPARA TRIO ASUMIENDO QUE R0
    CONTIENE EL PRIMER ELEMENTO
230 REM EL 2º y 3º ELEMENTOS SON CALCULADOS
240 REM ENCAJA UN ELEMENTO PERMUTADO EN
    EL FONDO DE LA PILA
250 R1 = R0 + R0
260 IF R1 > N THEN 360
270 IF R1 = N THEN 300
280 IF A(R1) ≥ A(R1 + 1) THEN 300
290 R1 = R1 + 1
300 IF A(R0) ≥ A(R1) THEN 360
310 T = A(R0)
320 A(R0) = A(R1)
330 A(R1) = T
340 R0 = R1
350 GOTO 250
360 NEXT I
```

En cada uno de los elementos 1 a N/2, son necesarias 2 comparaciones, por lo que el número mínimo de comparaciones comprendidas en este orden inicial no es mayor que $2 \cdot N/2$ o N. Puesto que no se requiere más de $1 + \log_2(N)$ capas con dos comparaciones para llevar un elemento al fondo de la pila, el número máximo de comparaciones adicionales no es mayor que:

$$N \cdot 2 \cdot (1 + \log_2(N)) \text{ o } 2 \cdot N + 2 \cdot N \cdot \log_2(N)$$

Por lo tanto, el número máximo de comparaciones para poner la lista en la configuración necesaria es:

$$N + 2 \cdot N + 2 \cdot N \cdot \log_2(N) \text{ o } 3 \cdot N + 2 \cdot N \cdot \log_2(N)$$

Recordemos que después que la lista está en este orden, se intercambia el elemento del tope con el último elemento, puesto que se sabe que es el elemento más grande de la lista. La nueva longitud de la lista es N-1. El elemento tope es encajado en el fondo de la pila (N-1), asegurándose que el elemento tope es nuevamente el más grande de los elementos restantes. Este proceso se repite hasta que los elementos restantes se han contraído a un solo elemento.

La rutina para efectuar el orden final de la lista es:

```
400 FORM = N-1 TO 1 STEP -1
410 T = A(M+1)
420 A(M+1) = A(1)
430 A(1) = T
440 R0 = 1
470 REM ENCAJA EL ELEMENTO TOPE AL FONDO DE
    LA PILA
480 R1 = R0 + R0
490 IF R1 > M THEN 590
500 IF R1 = M THEN 530
510 IF A(R1) ≥ A(R1 + 1) THEN 530
520 R1 = R1 + 1
```



```

530 IF A(R0) ≥ A(R1) THEN 590
540 T = A(R0)
550 A(R0) = A(R1)
560 A(R1) = T
570 R0 = R1
580 GOTO 480
590 NEXT M

```

Este proceso no puede sumar más de $N \cdot \text{LOG}_2(N)$ comparaciones al total; así, el número máximo de comparaciones para el sort completo debe ser menor que:

$$(3 \cdot N + 2 \cdot N \cdot \text{LOG}_2(N) + N \cdot \text{LOG}_2(N)) \text{ o } 3 \cdot N + 3 \cdot N \cdot \text{LOG}_2(N).$$

Para un tamaño 100 nominal de la tabla, esto no implica más de 2.400 comparaciones. En la práctica, el número de comparaciones es mucho menor que esta cantidad.

Dado que la primera y segunda parte de este método requieren una rutina para llevar un elemento al fondo de la pila, parece que lo mejor es escribir dicha rutina como una subrutina Basic.

La versión Basic final queda:

```

100 REM HEAP SORT
200 REM PARTE 1 - SE ESTABLECE LA PILA
210 M = N
220 FOR I = INT(N/2) TO 1 STEP -1
230 R0 = I
240 GOSUB 500
250 NEXT I
260 REM PARTE 2 - ORDENAMIENTO FINAL
270 FOR M = N-1 TO 1 STEP -1
280 T = A(M+1)
290 A(M+1) = A(1)
300 A(1) = T
310 R0 = 1
320 GOSUB 500
330 NEXT M
340 STOP
350 REM SORT COMPLETO
360 REM
500 REM LLEVA UN ELEMENTO AL FONDO DE LA
    PILA (LARGO M)
510 REM COMPARA TRIO ASUMIENDO QUE R0
    CONTIENE EL PRIMER ELEMENTO
520 REM EL 2º Y 3º ELEMENTOS SON CALCULADOS
530 R1 = R0 + R0
540 IF R1 > M THEN 640
550 IF R1 = M THEN 580
560 IF A(R1) ≥ A(R1+1) THEN 580
570 R1 = R1 + 1
580 IF A(R0) ≥ A(R1) THEN 640
590 T = A(R0)
600 A(R0) = A(R1)
610 A(R1) = T
620 R0 = R1
630 GOTO 530
640 RETURN

```

Quicksort

Todos los métodos previos han tratado de resolver el problema de sorting desde adentro ha-

cia afuera. Atacan primero el orden, luego el problema del rearreglo. Una alternativa (método siete - Quicksort 1) es solucionar el problema desde afuera hacia adentro. La metodología de resolución basada en el lema "dividir para conquistar" puede ser aplicada al problema de sortear tablas de elementos. Aplicada a una lista de elementos, el concepto es: 1) Cualquier lista de elementos que contenga más de 1 elemento es difícil de poner en orden; por consiguiente, hay que dividir la lista en dos más pequeñas. 2) Una tabla de 1 elemento es fácil de poner en orden.

A continuación, encontramos un proceso de partición que dividirá una lista de elementos en tres sublistas con las siguientes características: 1) La sublista del medio consiste en un solo elemento, ubicado donde pertenece en la lista finalmente sorteada. 2) Ningún elemento de la sublista de la izquierda es mayor que el elemento de la sublista central. 3) Ningún elemento de la sublista de la derecha es menor que el elemento de la sublista del medio.

Consideremos la lista de tres elementos: 20 30 10. Debemos poner el 20 donde pertenece y en seguida llevar los elementos más pequeños a la izquierda del 20 y los elementos más grandes a la derecha del 20: 10 20 30. Esto resulta en tres sublistas, cada una de un elemento de largo. Todas las sublistas están en orden dentro de ellas y además están en orden con respecto a cada una de las otras; por lo tanto, la lista completa está ordenada. La clave para el éxito de este método dependerá de la rutina de partición utilizada para separar la lista en sublistas.

El primer problema es decidir cuál elemento será movido hacia su lugar correcto como la sublista central. Una elección arbitraria del primer elemento de la lista es satisfactoria. Nos referiremos a este elemento como el elemento clave, o simplemente la clave.

Se forman las particiones preliminares: las particiones izquierda y derecha están vacías y la partición del medio contiene todos los elementos. La clave es comparada con el elemento en el otro extremo de la partición central. Si estos dos elementos están desordenados, son permutados. El elemento del extremo opuesto de la clave es empujado fuera de la partición central dentro de la partición extrema que está próxima a él.

La clave es comparada nuevamente con el elemento en el otro extremo de la partición central y permutado con él si están fuera de orden. Nuevamente, el elemento opuesto a la clave es empujado fuera de la partición central. Continuando el proceso de los últimos dos pasos, la partición del medio debe eventualmente contener sólo la clave.

Los elementos que han sido llevados fuera de la partición central están siempre en orden con respecto a la clave. Esto fuerza a que las tres particiones resultantes, o sublistas, tengan las carac-

terísticas deseadas. (Debe notarse que las sub-listas izquierda y derecha resultantes no necesitan contener ningún elemento. No obstante, una sub-lista vacía no molesta a las relaciones globales de orden.)

Después que la lista original se ha partido en tres sub-listas, cada sub-lista es tratada como una lista y se usa el proceso indicado para subdividir cada una de ellas hasta que existan solamente listas vacías o con un solo elemento. Cuando se ha llevado a cabo esto, la lista original ha sido sorteada. Este método de sorting por partición es generalmente llamado Quicksort y se le atribuye a C. A. R. Hoare.

Las características de performance de este método se basan en tests con verdaderos conjuntos aleatorios de elementos, pero es instructivo hacer un pequeño análisis del proceso.

Nótese que si las sub-listas son divididas en forma par, el número de sub-listas consideradas será cercano a la suma de las potencias de 2 que son menores que el tamaño de la tabla. Por ejemplo, consideremos una tabla de 100 elementos. En el primer nivel, 1 sub-lista de 100 elementos es procesada. En el segundo nivel, se procesan 2 sub-listas de 50 elementos. El tercer nivel tiene 4 sub-listas de 25 elementos cada una. El número de sub-listas consideradas es $1 + 2 + 4 + 8 + 16 + 32 + 64 = 127$. Este número es uno menos que una potencia de 2 justo mayor que el tamaño de la tabla.

Este es probablemente el número mínimo de comparaciones y permutaciones posibles con este método. Cada elemento se ve comprometido en comparaciones aproximadamente siete veces (basado en los siete niveles de sub-listas). Es lo más probable que cerca de $100 \cdot 7$ o 700 comparaciones se lleven a cabo, con el número de permutaciones entre 127 y 700.

El método siete requerirá prestarle atención a los puntos inicial y final de las sub-listas. La técnica usada generalmente es un stack. En Basic, el stack es simulado por dos tablas extras y un puntero para indicar cuántos elementos hay en las tablas stack.

```
120 DIM L (20), R (20)
130 S1 = 1
```

El stack contendrá originalmente la lista completa, puesto que va a ser particionada. Cuando el stack no contenga más sub-listas, el proceso está completo y el sort terminado.

```
130 S1 = 1
140 L (1) = 1
150 R (1) = N
160 IF S1 < 1 THEN 430
```

El tamaño de las tablas stack L y R se determina por el tamaño y orden original de la tabla A.

Asumiendo que cada partición corta la sub-lista previa por la mitad, el número de elementos necesarios en L y R sería LOG base 2 de N. Para una tabla de 100 elementos, L y R deben ser de al menos 7 elementos de largo. Si la tabla original estuviera en orden aproximadamente inverso, L y R podrían ser requeridos a que contuvieran unas 33 sub-listas. Una sugerencia sería dimensionar L y R en un tercio de A para seguridad o incluir chequeos para asegurarse que S1 no llegue a ser más grande que el tamaño de L y R.

Para particionar una lista en sub-listas, recuperamos el próximo par de límites izquierdo y derecho. Esta sub-lista es tratada como una lista y el proceso es repetido hasta que el stack está vacío.

```
160 IF S1 < 1 THEN 430
170 L1 = L (S1)
180 R1 = R (S1)
190 S1 = S1 - 1
200 REM PARTICION DE ESTA LISTA
340 REM PONER LIMITES DE SUB-LISTAS EN EL STACK
420 GOTO 160
430 REM SORT COMPLETO
```

El proceso de partición es efectuado guardando los índices izquierdo y derecho originales (L1, R1), levantando una bandera F para indicar en cuál lado de la sub-lista central está la clave y continuando el loop comparación-permutación-empuje hasta que los índices izquierdo y derecho de la sub-lista central sean el mismo.

```
200 L2 = L1
210 R2 = R1
220 F = -1
230 IF L2 ≥ R2 THEN 340
240 IF A (L2) ≤ A (R2) THEN 290
250 T = A (L2)
260 A (L2) = A (R2)
270 A (R2) = T
280 F = -F
290 IF F < 0 THEN 320
300 L2 = L2 + 1
310 GOTO 230
320 R2 = R2 - 1
330 GOTO 230
```

Los límites de las particiones recientemente creadas son puestos en el stack solamente si la sub-lista es de más de 1 elemento de largo.

Las sub-listas vacías y las sub-listas singleton no necesitan ser particionadas más adelante.

```
340 IF (L2 - L1) < 2 THEN 380
350 S1 = S1 + 1
360 L (S1) = L1
370 R (S1) = L2 - 1
380 IF (R1 - R2) < 2 THEN 160
390 S1 = S1 + 1
400 L (S1) = R2 + 1
410 R (S1) = R1
```

Luego, la rutina completa del método siete es:

```

100 REM QUICKSORT - VERSION 1
120 DIM L (20), R (20)
130 S1 = 1
140 L (1) = 1
150 R (1) = N
160 IF S1 < 1 THEN 430
170 L1 = L (S1)
180 R1 = R (S1)
190 S1 = S1 - 1
200 L2 = L1
210 R2 = R1
220 F = -1
230 IF L2 ≥ R2 THEN 340
235 C = C + 1
240 IF A (L2) ≤ A (R2) THEN 290
245 S = S + 1
250 T = A (L2)
260 A (L2) = A (R2)
270 A (R2) = T
280 F = -F
290 IF F < 0 THEN 320
300 L2 = L2 + 1
310 GOTO 230
320 R2 = R2 - 1
330 GOTO 230
340 IF (L2 - L1) < 2 THEN 380
350 S1 = S1 + 1
360 L (S1) = L1
370 R (S1) = L2 - 1
380 IF (R1 - R2) < 2 THEN 160
390 S1 = S1 + 1
400 L (S1) = R2 + 1
410 R (S1) = R1
420 GOTO 160
430 REM SORT COMPLETO

```

Cambiando los requerimientos para el proceso de partición del método siete, se deriva una versión diferente del quicksort (método ocho - Quicksort 2). El criterio de particionamiento es modificado para dividir una lista de elementos en dos sub-listas con estas características: ningún elemento de la sub-lista de la izquierda es mayor que el elemento más pequeño de la sub-lista de la derecha.

En el lugar del primer elemento de la lista, elegimos el elemento en el centro aproximado de la lista como el elemento con el cual comparar. Este elemento es el elemento clave. El proceso comienza formando tres particiones. Las particiones izquierda y derecha están vacías y la partición central contiene todos los elementos de la lista.

Los elementos de más a la izquierda son chequeados contra el elemento clave y empujados fuera de la partición central en cuanto no son mayores que el elemento clave. Los elementos de más a la derecha son chequeados contra el elemento clave y empujados fuera de la partición central en cuanto no son más pequeños que el elemento clave. Si la partición del medio no está vacía, los elementos de los extremos izquierdo y derecho son permutados. Estos elementos están fuera de orden por los dos pasos previos.

Los elementos de más a la izquierda y de más a la derecha son nuevamente chequeados contra

el elemento clave y empujados fuera de la partición central. Los extremos son permutados y el proceso continúa hasta que la partición central está vacía.

Cuando la partición central está vacía, la lista original ha sido dividida en dos sub-listas con la propiedad que ningún elemento de la sub-lista de la izquierda puede ser mayor que el elemento más pequeño de la sub-lista de la derecha.

Los puntos extremos de la sub-lista de la derecha son guardados en el stack y el proceso se repite en la sub-lista de la izquierda. Cuando ésta está agotada, el stack proporciona la siguiente sub-lista con la cual trabajar y se repite el proceso de arriba. Cuando ya no hayan más sub-listas guardadas en el stack, el proceso está completo y la lista ha sido sorteada.

Este método requiere una operación menos de stack que el método siete y hace una prometedora mejor elección del elemento clave.

La rutina completa es:

```

100 REM METODO OCHO - QUICKSORT 2
120 S1 = 1
140 L(1) = 1
160 R(1) = N
190 L1 = L(S1)
200 R1 = R(S1)
210 S1 = S1 - 1
220 L2 = L1
230 R2 = R1
240 X = A(INT((L1 + R1)/2))
250 C = C + 1
255 IF A(L2) ≥ X THEN 280
260 L2 = L2 + 1
270 GOTO 250
280 C = C + 1
285 IF X ≥ A(R2) THEN 310
290 R2 = R2 - 1
300 GOTO 280
310 IF L2 > R2 THEN 340
315 S = S + 1
320 T = A(L2)
322 A(L2) = A(R2)
324 A(R2) = T
330 L2 = L2 + 1
335 R2 = R2 - 1
340 IF L2 ≤ R2 THEN 250
350 IF L2 ≥ R1 THEN 390
360 S1 = S1 + 1
370 L(S1) = L2
380 R(S1) = R1
390 R1 = R2
400 IF L1 < R1 THEN 220
410 IF S1 > 0 THEN 190
420 REM SORT COMPLETO

```

La Tabla N° 1 está basada en una secuencia de números generada aleatoriamente puesta en el arreglo A(N). Cada línea consiste de un par de números. El número de arriba es la cuenta de las comparaciones, el número de abajo es la cuenta de los intercambios. La columna de la izquierda representa el número de elementos en los diferen-

COMO ESCRIBIR PARA MICROBYTE

Nuestra revista es una publicación viva que se nutre de las vivencias y experiencias de ustedes, nuestros lectores y colaboradores. El objetivo de Microbyte es informar en forma seria y didáctica a todas las personas que de uno u otro modo están relacionadas con la informática y computación, divulgando a través de estas páginas las herramien-

tas y metodologías relacionadas con el tema en una forma práctica, amena y comprensible para un amplio espectro de lectores.

A fin de uniformar la presentación y calidad de los artículos de fondo de Microbyte, presentamos a continuación las pautas para la preparación de trabajos que rigen para este tipo de artículos:

a) Temática:

- * Artículos descriptivos sobre software, lenguajes, compiladores y productos de cuarta generación.
- * Metodología de diseño y análisis de sistemas.
- * Técnicas de programación.
- * Usos y experiencias prácticas en áreas no tradicionales de la computación (medicina, ciencias sociales, etc.).
- * Herramientas de gestión de empresas con uso de computadores.
- * Inteligencia artificial, sistemas expertos y robótica.
- * Administración de departamentos de informática.

b) Presentación:

- * Los artículos presentados deberán tener una longitud máxima de doce páginas escritas a máquina a doble espacio.
- * Todos los gráficos, figuras y facsímiles de pantallas deberán ir separados, numerados y en hojas blancas individuales para cada una.
- * Si se acompañan listados de programas, éstos deben venir impresos con cinta nueva o en modo enfatizado (negrita), en formulario continuo color blanco sin rayado.
- * La bibliografía citada deberá numerarse, colocando los datos en el siguiente formato: título de la obra o artículo, autor, publicación y año.

c) Contenido:

- * Todos los artículos que presenten herramientas computacionales de cualquier tipo deberán incluir, además de una descripción teórica de la herramienta, un ejemplo resuelto mediante el programa correspondiente.
- * Los trabajos deben ser originales. En caso de ya haber sido publicados en otro medio, indicar nombre y fecha de aparición.
- * Los artículos deberán incluir al final un párrafo de conclusiones, que resuma lo más relevante del trabajo y ofrezca sugerencias para la aplicación práctica o posibles usos de los temas tratados.

Todos los artículos recibidos serán sometidos a revisión por nuestro comité editorial y un panel de expertos en los temas tratados antes de ser publicados. Las colaboraciones aceptadas tendrán un pago que fijará periódicamente la revista.

Las colaboraciones deben enviarse acompañadas de una breve reseña biográfica y fotografía tamaño pasaporte del autor a:

At. Sr. José Kaffman
Huelén 164-2º piso
Providencia - Santiago



tes tamaños del arreglo. De la tabla, se hace evidente que los métodos Shell, Quicksort y Heapsort son lejos superiores a los otros métodos para secuencias aleatorias.

El número de comparaciones e intercambios es importante; sin embargo, la comparación de tiempos de ejecución para cada método de sorting es mucho más dramática. Hemos provisto una tabla de tiempos (Tabla N° 2) para su comparación. Estos tiempos están basados en la ejecución de las rutinas en un microcomputador IBM-PC compatible con CPU Intel 8088 a 4.77 MHz y Basic intérprete. Cada línea de la tabla representa el tiempo de ejecución para sortear en minutos y segundos (MM:SS). La columna de la izquierda representa el número de elementos en el arreglo A. El tiempo de sort para los primeros tres métodos se incrementa muy rápidamente así como crece el tamaño del arreglo. Recordemos que el número de comparaciones es proporcional al cuadrado del número de elementos en la tabla. No debe ser sorprendente que el tiempo para sortear sea también proporcional al cuadrado del número de elementos sorteados.

La mejoría esperada entre el método dos y el método tres no se presenta; de hecho, el método tres parece ser más lento. Las instrucciones extras, necesarias para chequear la condición de un arreglo ordenado toma más tiempo que las pequeñas mejoras en el número de comparaciones.

El método cuatro (Inserción) trabaja generalmente mejor que lo esperado, debido a los ahorros de tiempo ocasionados por las mejorías en el método de permutación. La gran mejoría en tiempo aparece entre los primeros cuatro métodos y los últimos cuatro. Los métodos Shell, Heapsort y Quicksort son lejos los métodos más rápidos.

El número de comparaciones para los últimos cuatro métodos es proporcional al número de elementos en el arreglo por el LOG (base 2) del número de elementos. Pero esto no representa un incremento lineal en tiempo ni tampoco incrementa tan rápidamente como las funciones en 2a. potencia de los primeros cuatro métodos.

Los métodos Quicksort requieren instrucciones extra para manejar las actividades asociadas con el stack de punteros de límite. Para tamaños de tabla pequeños, este gasto fijo es consumidor de tiempo. A medida que aumenta el número de elementos en el arreglo, los mejores algoritmos de los métodos siete y ocho salvarán finalmente este esfuerzo extra.

Los métodos Quicksort son los ganadores sobre la base del número de comparaciones y permutaciones y también son los más cortos en términos de tiempo de sort. Los métodos Shell y Heapsort son también francamente superiores que los primeros cuatro métodos. La elección final debe basarse en cuál método trabaja mejor en la situación particular para la que se requiera el sort.

Tabla N° 1
Métodos de sort – Comparación de secuencia aleatoria.

Largo (N)	1. Bruto	2. Bubble	3. Bubble2	4. Insc.	5. Shell	6. Heap	7. Quick1	8. Quick2
10	81	45	35	18	24	33	21	44
	10	10	10	10	11	28	13	11
20	361	190	175	93	73	85	76	114
	77	77	77	77	33	74	27	29
50	2401	1225	1147	675	291	266	270	358
	629	629	629	629	141	242	97	78
100	9801	4950	4944	2723	800	627	604	917
	2627	2627	2627	2627	405	569	225	191
200	39601	19900	19795	10475	1980	1457	1593	2036
	10279	10279	10279	10279	991	1355	534	420
300	89401	44850	44184	22005	3224	2362	2905	3643
	21709	21709	21709	21709	1517	2199	827	653
400	159201	79800	79610	39336	4320	3327	3995	4249
	38940	38940	38940	38940	1947	3104	1211	952
500	249001	124750	124597	61439	6179	4311	4923	6034
	60944	60944	60944	60944	3139	4042	1630	1176

Tabla N° 2
Métodos de sort – Comparación de Tiempos.

Largo (N)	1. Bruto	2. Bubble	3. Bubble2	4. Insc.	5. Shell	6. Heap	7. Quick1	8. Quick2
10	:01.5	:01	:01	:00.5	:00.5	:01	:00.5	:01
20	:05	:04	:04	:02	:01	:03	:02	:02
50	:33	:22	:22	:12	:05	:10	:07	:06
100	2:18	1:30	1:34	:54	:14	:22	:16	:15
200	9:10	5:59	6:15	3:32	:36	:51	:39	:32
300	20:10	12:43	13:18	7:07	:59	1:22	1:07	:55
400	35:30	22:16	23:10	12:22	1:18	1:57	1:32	1:08
500	56:09	35:32	37:20	19:33	1:51	2:31	1:56	1:32

Héctor Miranda Riquelme, se convirtió en 1978 al apostolado informático con una TI 59. Su devoción por estas máquinas lo llevó a peregrinar por las H.P., Casio y otras hasta ser iluminado por los computadores personales.

Frente a la tentación sibarítica de los mainframes y terminales opuso su pasión por la libertad del computador personal especializándose en todas las nuevas tecnologías, microprocesadores, lenguajes, compiladores y paquetes de software que aparecen para éstos.

Se interesa en divulgar el verbo informático de un modo accesible a toda la grey.



Ejerce su ministerio como asesor consultor en empresas tales como Vidrios Lirquén y Shell Chile, participando también en el consejo editorial de Revista Microbyte.

LA SOLUCION EFICAZ...

...A LOS REQUERIMIENTOS DE SU EMPRESA

I.C.S. Ingenieros Consultores de Sistemas.
Le proveemos de soluciones adecuadas, rápidas y económicas
para sus necesidades de información.

Más de 150 empresas del país cuentan con nuestros sistemas
funcionando con éxito. Nuestra amplia experiencia le asegura la
mejor solución para optimizar su gestión empresarial. Ponemos
a su disposición:

- Sistema de Contabilidad General.
- Sistema de Remuneraciones.
- Sistema de Control de Existencias.
- Sistema de Cuentas Corrientes Clientes.
- Sistema de Cuentas Corrientes Proveedores.
- Sistema de Facturación y Estadísticas de Ventas.
- Sistema de Activo Fijo.
- Sistema de Cálculo de Costo.
- Sistema de Correo Directo.

Si usted es usuario de un microcomputador

IBM PC, XT, AT
Burroughs B-25
NCR Decisión Mate
Texas Instrument
Hewlett Packard HP-150

Multitech
Radio Shack
Olivetti M-24
IBM Compatibles.

Con sistemas operativos **MS-DOS / XENIX - Multiusuario.**
Contáctese con nosotros. Solicite una demostración en:



INGENIEROS CONSULTORES DE SISTEMAS

Un método para mejorar la eficiencia operacional y el control interno.

CODIGO DE BARRA

Juan Enrique Spencer Risopatrón



Códigos de Barra

Figura N° 1

En esta era de comunicaciones y tecnología, todo lo referente a transferencia de datos y eficiencia operacional parece estar resuelto. Sin embargo, los seres humanos todavía "hablamos" al computador por medio del tradicional papel y lápiz. Así, cada vez que necesitamos juntar una cierta cantidad de información, para un posterior procesamiento, se procede primero a llenar un formulario con los datos de interés y luego se llevan a un pool de digitadores que procederán a ingresar los datos al computador central. La pérdida de tiempo y dinero relacionada con las dos etapas anteriormente mencionadas, por concepto de impresión de formularios (escritura) y luego por digitación, hace que la adquisición de grandes sistemas computacionales sea poco me-

Este problema fue resuelto desarrollando un sistema de lectura óptico consistente en un símbolo formado por barras y espacios (o códigos de barra) que son fácilmente leídos por dispositivos ópticos de bajo costo, con una probabilidad de error cercana a cero. La ventaja de esta técnica gráfica es la de permitir leer fácilmente el código impreso, con un lector de bajo costo, sensible a las variaciones de luz y sombra producidas por el ancho de las barras y el fondo blanco. Existen otros métodos de identificación automática de productos como el OCR (Optical Code Recognition), Visión Computarizada, Reconocimiento de formas, etc. Sin embargo la codificación con barras ha sido ampliamente aceptada por su bajo costo de generación (generalmente por impresora) y a su elevada precisión.



Telxon
Figura N° 2

¿Qué es un código de barra?

Un código de Barra (CB) es

sentado a un código, número de identificación, o número de parte de un determinado producto. Esto significa que no es necesario cambiar la actual identificación (o codificación) de los productos en una empresa, sino que sólo es preciso generar el símbolo o CB a la codificación existente.

Un CB se constituye de seis partes, que son:

- Zona de silencio
- Carácter o secuencia de partida
- Código a representar
- Dígito verificador
- Carácter o secuencia de parada
- Zona de silencio

De los seis componentes mencionados sólo el tercero es representativo del criterio de codificación de la empresa, los restantes están dados por la norma del CB empleado (Cod. 39, 12 de 5, UPC, ...etc.). La aparente complejidad observada nos garantiza que el código de formato de barras, si es leído y aceptado por el decodificador (con un beep, generalmente) este es correcto. El usuario normal de CB no requiere conocer los detalles ni de la norma utilizada, ni de su estructura, pero es útil conocerla para fortalecer la confianza en su utilización.

En la figura N° 3 se muestra la definición del Código 39, en ella se puede apreciar el criterio que permite a la norma tener implícitamente un método de detección de errores de lectura. Cada carácter es represen-



SISTECO OFRECE:

IMPRIMIR... mayor calidad por más tiempo.

Para imprimir con gran nitidez y alto rendimiento, las Cintas de Impresión de SISTECO tienen la ventaja adicional: Mejor Costo/Beneficio.

SISTECO, sólo suministros de probada calidad para las exigentes tareas del área computacional.

Cintas para todo tipo de impresoras, especialmente para aquellas que deben trabajar tiempo extra.



SISTECO

Vicuña Mackenna 152
teléfono 222 55 33



cuatro blancas (o fondo, que pertenece al código), de los cuales tres son anchas. De ahí viene el nombre de código 3 de 9 o código 39. Este tipo de codificación disminuye asombrosamente la probabilidad de una lectura errónea.

(Cuadro con la definición del código 39)



Figura N° 3

¿Para qué sirven los códigos de barra?

Su utilidad práctica está ligada a la velocidad y precisión con que es posible ingresar los datos de mayor trascendencia para un sistema computacional. Si se reemplaza la impresión manual de Códigos en formularios o planillas por CB, es posible eliminar casi totalmente los errores de digitación y/o transcripción. Si junto con los CB se utiliza un sistema de captadores de datos portátiles que permitan el ingreso de datos en formato de barra (ver figura N° 2), su utilidad queda doblemente justificada.

¿Qué código de barra usar?

Existen varias normas de Códigos de Barra, las cuales han sido desarrolladas por diferentes empresas. En general no existe ninguna limitación para usar cualquier norma de CB en cualquier aplicación, sin embargo, algunas aplicaciones han tomado una norma como su standard, ya sea porque ha sido normalizado por una institución reguladora o bien porque se ha preferido emplear la experiencia anterior. En la tabla N° 1 se presenta un resumen de las normas de CB más utilizadas actualmente y las aplicaciones con las cuales más frecuentemente se les relaciona.

Tabla N° 1: Comparación de los códigos de barra más conocidos.

CODIGO	DESCRIPCION	TIPO	FMTD	DETECCION			APLICACIONES
				BID	IMP	EXP	
2 de 5	B. Angosta = 0 B. Ancha = 1	N	Disc.	S	X	X	Organización Bodegas Tickets Aerolíneas Id. Cubiertas.
1-2/5	2 de 5 en el cual los caract. pares se rep. en los espacios de los caract. impares.	N	Cont.	S	X	X	Ap. Industrial Ind. Automotriz Bodegas Flejes Supermercados
UPC/EAN	2 barras y 2 espacios ocupando 7 módulos.	N	Cont.*	S	X	X	Id. productos al por menor
Cód. 39	Código con 3 de los 9 elementos son anchos.	A	Cont.	S	X	X	Fabricación Hospitales Librerías Universidades Agencias de Gobierno.
Codabar	Barra con 3 espacios	N	Disc.	N	X	X	Empaquetamiento productos al por mayor.
Cód. 11	Alta densidad	N	Disc.	S	X	X	Id. productos de telecomunicaciones.

Donde el TIPO indica si el CB es Numérico o Alfanumérico, FMTD indica si el formato es Continuo o discreto, BID indica si es de lectura bidireccional o no, y si la DETECCION de errores es IMPlícita (en la defini-

ción) o EXPlícita (dígito verificador).

Juan Enrique Spencer
Risopatrón
Gerente Depto. Ing. y Desarrollo
Rimpex Chile.

AGRICULTORES E INDUSTRIALES

AGRICULTORES E INDUSTRIALES disponen ahora de dos excelentes programas indispensables en la aguda competencia actual:

DIETAS: permite calcular la fórmula adecuada para aves, cerdos, vacunos, etc. que cumplan con los requisitos de nutrición al costo mínimo posible con los productos de que Ud. dispone (método SIMPLEX de programación lineal).

COSTOS Y PRODUCCION: Mantiene la estructura de costos (costo estándar) de hasta 10.000 productos y la actualiza cada vez que se modifican los insumos. Además evalúa el costo y requerimientos de un programa de producción y entrega informe estadístico de los márgenes de comercialización.



Solicite programas y desquite por \$15.000 con ambos programas para demostración, enviando cheque o giro por \$1.000 a Carlos Contreras Mezzano, El Vergel 2475, Providencia, Santiago. Consultas telefónicas: 748770-2260038.



A UD. QUE NECESITA UN COMPUTADOR PROFESIONAL DATAMERICA LE OFRECE SU CORONA.

CORONA PC de Corona DATA SYSTEMS-CORDATA, California. La más alta resolución; sólida arquitectura; chips de primera selección; mayor capacidad de crecimiento; alta compatibilidad y facilidad de comunicación con todos los computadores de otras marcas.

La más grande biblioteca de software disponible y en definitiva un mejor y permanente servicio.

El Computador CORONA es el único que goza de garantía DATAMERICA.

DATAMERICA

**CORONA, SU EMPRESA CORONA.
VAYA A DATAMERICA Y OBTenga SU CORONA.**

PROTOTIPOS

INTRODUCCION A UNA METODOLOGIA FORMAL DE DESARROLLO DE SISTEMA

Guillermo Beuchat S.

Toda metodología formal de desarrollo de sistemas computacionales intenta resolver problemas básicos asociados a cualquier proyecto de software, tratando de minimizar los costos de la aplicación y de mejorar el rendimiento y productividad de las personas involucradas, tanto en el área usuaria como en el área de sistemas. En un artículo anterior sobre este tema, se abordaron las características generales y variables claves que debe considerar una metodología de desarrollo que utilice los prototipos como técnica fundamental de apoyo. Ahora, se introducen los aspectos generales de un enfoque de desarrollo propuesto por el autor, la que será explicada en detalle en el próximo artículo de esta serie.

Definición de los usuarios de un sistema

En un artículo anterior de MICROBYTE [1], se hizo una definición formal de los usuarios en un proceso de desarrollo de sistemas y los mecanismos de participación específicos que tienen en el proceso, la que recordaremos aquí pues se ha utilizado para definir el rol de los usuarios en la metodología de diseño por prototipos.

Casi todos los enfoques de diseño propuestos en la bibliografía, incluyendo por supuesto el método de prototipos, se preocupan de la manera de involucrar a los usuarios en el proceso de diseño, y de las formas de obtener su colaboración para construir un buen sistema. Se sostiene que la participación de los usuarios es fundamental. Incluso, se propone un enfoque de desarrollo directamente por los usuarios, usando software de cuarta generación. Sin embargo, existe una notoria deficiencia en cuanto a la definición misma de usuario, lo que se traduce en diversos problemas y malentendidos durante la fase de desarrollo de un sistema de información.

Esta ambigüedad en la definición se produce por cuanto no se especifica claramente la relación del usuario con el sistema. Obviamente, diferentes usuarios tienen diferentes necesidades de información o diferentes tareas que realizar dentro del sistema computacional, por lo que es necesario considerar la función de cada uno al diseñar los programas y especialmente las interfaces visuales o impresas.

Se ha considerado que la causa principal del "cuello de botella" en el desarrollo de software es el uso de enfoques de diseño imperfectos. Por ello, la investigación se ha centrado en proponer

nuevos enfoques, entre ellos el de prototipos. Sin embargo, las experiencias del autor indican que éste no es el camino adecuado. Por ello, se plantea que la definición adecuada de "niveles" de usuario y sus respectivos mecanismos de participación constituye un paso previo y necesario que contribuirá mucho a mejorar el proceso de desarrollo de sistemas.



Para definir los tipos y roles de los usuarios, enfocaremos el problema a través de dos aspectos fundamentales:

a) Definición de usuario

GREMILLION y PYBURN [2] sostienen que un usuario es "cualquier persona que no sea analista de sistemas, cuya misión fundamental es el desarrollo, administración y/o explotación de sistemas computacionales". Nosotros ampliaremos esa definición, diciendo que un usuario es cualquier persona dentro de la organización, excluyendo a los analistas encargados del desarrollo, que entrará de algún modo en contacto con el sistema computacional, ya sea durante la fase de análisis, desarrollo u operación del mismo. Esta definición amplia contempla entonces a diversos niveles o tipos de usuarios, desde los ejecutivos máximos de la organización hasta las digitadoras de entrada de datos.

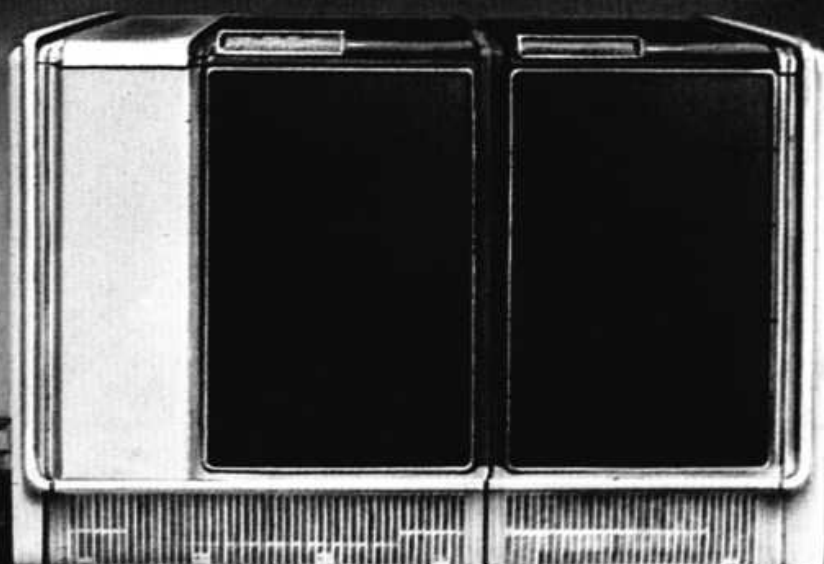
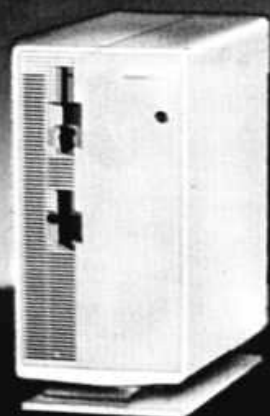
b) Definición de niveles

Se propone dividir a los usuarios así definidos, en tres niveles: ejecutivo, intermedio y operativo.

Los usuarios de nivel ejecutivo corresponden a

Continúa en pág. 54

De MINI a SUPERMINI compatibilidad total



ECLIPSE	MV 2000 DC	MV 4000	MV 7800 DC	MV 7800	MV 8000 II	MV 10000	MV 10000 SX	MV/20000 I	MV/20000 II
Terminales	24	48	48	128	128	192	192	1008	1008

Desde el más pequeño hasta el más poderoso, todos los computadores DATA GENERAL le aseguran una COMPATIBILIDAD TOTAL, tanto en Hardware como en Software. Esto significa un costo más bajo si Ud. desea aumentar la capacidad de su equipo. Es también una de las razones por las cuales Data General le puede ofrecer la mejor Relación PRECIO/RENDIMIENTO en cada uno de sus productos. Cuando considere la compra de su equipo computacional tenga presente a Data General. Estamos seguros de poder facilitar su decisión.

 **Data General**
una Generación adelante

los ejecutivos de la plana gerencial de la organización. Su preparación académica y experiencia es amplia, y poseen sólidos conocimientos de técnicas y herramientas de gestión empresarial. En sus manos están las decisiones de nivel estratégico de mediano y largo plazo, que requieren de información agregada y de buena calidad en forma oportuna y confiable.

El usuario ejecutivo, dada su alta posición en la organización, dispone normalmente de poco tiempo para intervenir en el proceso de desarrollo de un SIA, interesándose más bien en los resultados finales que éste entregue, desde el punto de vista de su efectividad y utilidad para la toma de decisiones.

Los usuarios de nivel intermedio corresponden al personal administrativo (mandos medios) de la organización. Muchas veces, este tipo de usuario origina la necesidad de desarrollar un SIA en primer lugar, ya que debe tomar decisiones de nivel táctico y operativo que le exigen contar con mecanismos adecuados de control y generación de información para apoyar esas decisiones. Este tipo de usuario generalmente proporciona la mayor ayuda al analista de sistemas, dado su interés

Los programas no efectúan ningún tipo de validación.

directo en hacer más eficiente su trabajo y su disponibilidad de tiempo es bastante mayor que los usuarios de tipo ejecutivo. Además, este tipo de usuario se preocupa por la eficiencia de los procesos del sistema, ya sean estos manuales o mecanizados.

El nivel inferior de usuario corresponde al nivel operativo. Estos usuarios se relacionan directamente con la operación rutinaria del sistema, ya sea en la generación o input mecanizado de datos (llenado de formularios, digitación), o con el uso de menús interactivos de control de procesos. Su preparación académica es generalmente de bajo nivel, y su grado de comprensión de los conceptos generales y estratégicos del sistema que se diseña es bajo. El nivel decisional es bajo o nulo, y tienen un alto grado de interacción con los aspectos netamente computacionales del sistema. Esto último los hace especialmente importantes en el proceso de diseño de programas y procedimientos.

Mecanismos de participación.

Dado que los diferentes niveles de usuario que se han definido tienen diferentes formas de participación en el desarrollo de un sistema, los abordaremos por separado. Los mecanismos de participación expuestos no son una lista exhaustiva, pudiendo encontrarse diferentes formas de participación de acuerdo al tipo de sistema y otros

Participación de los diferentes niveles de usuario en el proceso de desarrollo de sistemas.

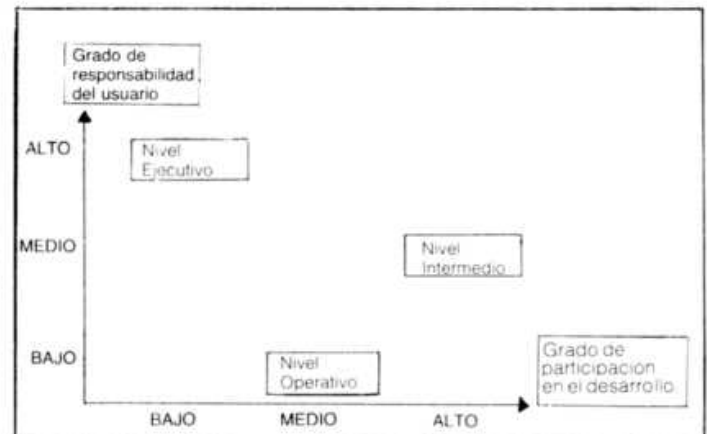


Figura 1.

factores. La figura 1 muestra gráficamente la importancia de cada tipo de usuario en el proceso de diseño del sistema.

a) Nivel ejecutivo.

A este tipo de usuario le corresponden, entre otras, las siguientes formas de participación:

- * Definición clara y exhaustiva de los requerimientos de un sistema, sin importar la forma en que éste se implemente (manual o computacionalmente). Es decir, se debe especificar los objetivos del sistema, sin entrar en detalles de cómo lograrlos.

- * Evaluación y crítica al diseño propuesto, en términos de las metas de efectividad planteadas para el sistema más que su eficiencia en términos computacionales.

- * Evaluación y definición de políticas de costos y asignación de recursos al proyecto, autorizando o negando la continuación del proceso de desarrollo.

b) Nivel intermedio.

Su participación con el grupo de desarrollo del SIA se manifiesta a través de los siguientes aspectos:

- * Definición de procedimientos administrativos, computacionales y manuales, necesarios para alcanzar los objetivos propuestos.

- * Especificación de metas de eficiencia computacional y administrativa, comparando con el sistema actual si es que existe.

- * Definición de requerimientos de datos y almacenamiento, para poder generar información de salida del sistema. Ayuda para la confección de formularios y métodos de obtención de datos de entrada al sistema, especialmente durante la construcción de prototipos.

- * Diseño y especificación de salidas del sistema, tales como listados, informes, pantallas de consulta interactiva y otros, validándolas a través del uso de los prototipos.

COMPRUEBE LA POTENCIA DE **ALTOS** EN SUPER-MICROS



Se sorprenderá lo que el poderío de los Super-Microcomputadores multiusuarios ALTOS pueden hacer por el éxito de su negocio. Desde 2 hasta 30 usuarios por equipo, cientos de programas administrativos y de comunicaciones. Todo a un menor costo por usuario que un computador personal.

Decídase por un ALTOS y compruebe las ventajas de un multiusuario de verdad con la mejor relación Costo/Rendimiento del mercado y el respaldo efectivo de ELCA- COMPUTACION.

ELCA
COMPUTACION

ALTOS

THE POWER IN MULTI-USER MICROS

Comuníquese con ELCA

Representante exclusivo de ALTOS COMPUTER SYSTEMS Inc. de USA. Casa matriz: Amunátegui 669, fono 722583, Santiago.

Sucursales en: Arica • Iquique • Antofagasta • La Serena • Viña del Mar • Rancagua • Talca • Chillán



c) Nivel operativo.

Su interacción con el grupo de desarrollo de SIA se manifiesta a través de los siguientes mecanismos:

- * Evaluación del tiempo de respuesta de los programas de consulta interactiva y entrada masiva de datos, durante la etapa de construcción del SIA.

- * Evaluar la calidad general y facilidad de uso de las interfaces usuario-sistema (pantallas, menús), incluyendo el grado de estandarización en el uso de comandos del sistema a través de todos los programas.

- * Calidad y facilidad de lectura de los formularios de ingreso de datos y compatibilidad de formatos con las pantallas respectivas de ingreso. La evaluación de estos aspectos es crítica para lograr una eficiencia global del sistema.

- * Evaluación y crítica a los manuales del usuario y guías de operación, ofreciendo sugerencias de redacción e indicando las omisiones y redundancias que se detecten.

El enfoque generador.

El enfoque generador consiste en la utilización de un prototipo del sistema para generar un diseño completo y definitivo a partir de un breve análisis inicial. Previo al desarrollo del prototipo, es necesario haber realizado la etapa formal del estudio de factibilidad. Los datos contenidos en este estudio servirán en una primera instancia, para el diseño del prototipo, que incluirá pantallas de entrada/salida y el modelo de datos básicos del sistema.

El prototipo generador posee dos características fundamentales: la modularidad del diseño y el trabajo con archivos.

El concepto de modularidad se refiere al hecho de que el prototipo es una unidad autosuficiente, compuesta por módulos funcionales independientes. En cualquier momento, es posible agregar o eliminar módulos completos, de acuerdo a los cambios en el diseño y los requerimientos que necesariamente van ocurriendo durante el proceso de construcción, evaluación y modificación del prototipo. Por otra parte, un prototipo generador trabaja directamente con archivos de datos, por lo que los diferentes módulos funcionales de-

berán contener todos los programas de mantenimiento de la base de datos.

Otros aspectos importantes que es necesario considerar son los siguientes:

- * El prototipo generador constituye, en la práctica, un modelo "vivo" o "real" del sistema computacional, compuesto por interfaces usuario-sistema (pantallas, menús y listados) además de programas de mantención de archivos.

- * Los programas no efectúan ningún tipo de validación de los datos que se ingresan al sistema durante la simulación realizada en conjunto con el usuario. El objetivo primario del prototipo, en este caso, es generar un diseño lógico y físico útil, que solucione realmente las necesidades de los usuarios, en un tiempo mínimo y con el menor esfuerzo de programación. Sólo en la etapa posterior de conversión prototipo-sistema, se introduce a los programas las rutinas de validación que sean necesarias.

- * El sistema opera en forma "real", funcionan todos los comandos de ejecución, pantallas y menús, con los que el usuario puede interactuar como si estuviese operando el sistema definitivo.

El prototipo generador permite aclarar los requerimientos de un sistema de información.

- * La concepción modular del prototipo debe completarse con una adecuada visión general del sistema, con el fin de no perder de vista el objetivo central del mismo durante las iteraciones en el desarrollo. De otra manera se corre el riesgo de seguir agregando módulos funcionales indefinidamente y perder el control y coherencia en el sistema, creando un diseño inorgánico.

- * Todos los procesos "batch" del sistema, tales como la emisión de listados y cierres mensuales, deberán programarse como parte del prototipo durante el desarrollo del mismo. En el caso particular de los listados, el diseño del formato de impresión deberá validarse a partir del resultado impreso de estos programas, a fin de otorgar una sensación de realidad al usuario.

- * Si durante el desarrollo del prototipo se detecta la necesidad o conveniencia de usar o transmitir datos con otros sistemas, la interface con dichos sistemas deberá enfocarse como un módulo más del prototipo, cuyo desempeño también será validado por el usuario.

¿Cuándo usar el enfoque?

El enfoque generador, por sus características, se presta especialmente para resolver el diseño de sistemas con los siguientes problemas:

- a) El o los usuarios no conocen claramente sus propios requerimientos de información. Existe una

idea vaga de la necesidad de contar con un sistema computacional, pero no se sabe exactamente para qué. En este caso, el prototipo generador proporciona una manera de aclarar esos requerimientos, tanto a los usuarios como a los analistas.

b) El o los usuarios conocen sus requerimientos, pero no saben exactamente qué información es necesario mantener almacenada para satisfacerlos, o dónde obtenerla. En este caso, el prototipo permite definir un modelo de datos completo del sistema, transformando una idea vaga del usuario en un conjunto de datos y atributos específicos. La labor del analista, en este caso, consiste en diseñar una estructura de datos o archivos adecuada.

c) Aunque se reconoce una necesidad bien definida, y se sabe exactamente qué datos es necesario manipular, los usuarios se resisten al desarrollo del sistema por razones personales, organizacionales o psicológicas, en circunstancias que el desarrollo del sistema es aconsejable desde el punto de vista económico y de los objetivos estratégicos de la organización. La construcción del prototipo es un proceso que requiere de mucha intervención de los usuarios, por lo que se produce un compromiso y una motivación para realizar el proyecto que sería imposible analizar mediante el método tradicional. Los usuarios se sentirán responsables del resultado y calidad de "su" sistema, y prestarán por lo tanto su máxima colaboración a los analistas.



d) El tema del sistema en desarrollo es altamente complejo, y el analista no conoce o no domina los aspectos técnicos o matemáticos del sistema. En este caso, la utilización del prototipo facilita la intervención del usuario en la definición de requerimientos y permite que el mismo diseñe las partes más complejas del sistema.

Como parte de este análisis, es importante destacar aquellas situaciones en que no es recomendable el uso de este enfoque de diseño, debido a potenciales problemas en la administración del proyecto, o bien a la falta de una justificación económica suficiente. Estas situaciones son las siguientes:

a) Cuando un sistema tiene muchos usuarios con diferentes requerimientos o muchos subsistemas de características opuestas, se produce una confusión al intentar aplicar el enfoque generador. En efecto, cada usuario tiene una visión particular del sistema, y por lo tanto solicitará al analista que el prototipo satisfaga sus propias necesidades. Ello implicará un prototipo excesivamente grande y desordenado, el que será muy difícil de administrar.

b) Si un sistema está claramente definido, se conocen todos los datos a manipular y no se prevén problemas técnicos durante el desarrollo, el costo de aplicar el modelo de prototipos puede ser demasiado elevado, debido al uso intensivo de tiempo de computador y recursos humanos. Este factor es crítico aun en sistemas pequeños.

Los usuarios en el enfoque generador.

Los diferentes niveles de usuario intervienen en el enfoque generador de acuerdo a las siguientes pautas:

1. Nivel ejecutivo.

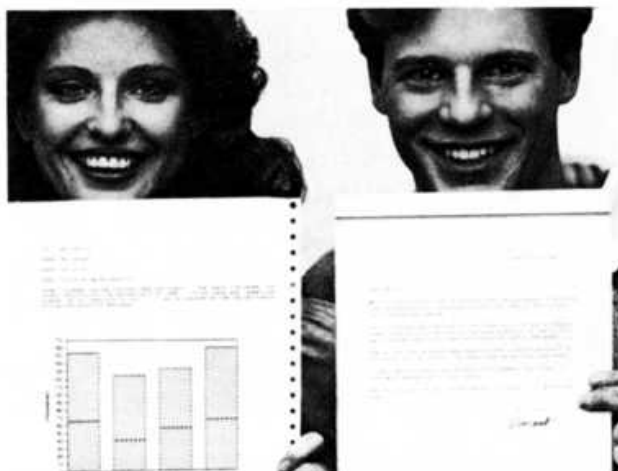
- a) Etapas de participación.
 - estudio de factibilidad.
 - construcción prototipo generador.
- b) Modo de participación.
 - definición clara y exhaustiva de los requerimientos lógicos y de información del sistema.
 - señalar y cuantificar los beneficios directos e indirectos del sistema propuesto.
 - especificar áreas de injerencia del sistema en la organización.
 - evaluar los resultados del funcionamiento del prototipo en términos del cumplimiento de objetivos, utilidad para la toma de decisiones, cumplimiento de normas de auditoría y seguridad, y facilidad de uso de los resultados generados por el sistema.
 - sugerir modificaciones de fondo al sistema, en cuanto a los resultados producidos por el prototipo.
 - sugerir la modificación o eliminación total de uno o más módulos funcionales del prototipo, de acuerdo a su percepción de la utilidad que prestan al sistema en general.

2. Nivel intermedio.

- a) Etapas de participación.
 - construcción del prototipo generador.
 - especificación funcional y diseño físico.
- b) Modos de participación.
 - definición y especificación de formularios de ingreso y salida de datos.
 - definición de los datos a almacenar, ne-

cesarios para cumplir los objetivos planteados por los usuarios de nivel ejecutivo.

- especificación de prioridades para la construcción de módulos funcionales del prototipo de acuerdo a la importancia que tenga.
- *evaluar cada módulo del prototipo, en términos de la redundancia o falta de datos, la secuencia de procedimientos, la presentación y calidad de los resultados impresos y la utilidad que prestan los diversos listados.*
- construir y/o diseñar pantallas de consulta interactiva, de haberlas, usando el software adecuado. De ser necesario, recibir la capacitación requerida para usar este tipo de herramienta.
- especificar rangos y valores válidos para los datos que almacena el sistema, los que serán incluidos en la Guía de Conversión Prototipo-sistema.



3. Nivel Operativo

a) Etapas de participación

- construcción prototipo generador
- conversión prototipo-sistema

b) Modos de participación

- construir y diseñar el layout de pantallas de ingreso de datos y menús del sistema. De ser necesario, recibir la capacitación necesaria para usar las herramientas de software adecuadas

- evaluar cada módulo del prototipo en términos de la facilidad de uso de los programas, calidad y comprensión de los comandos de procesamiento en línea, y el tiempo de respuesta en los terminales

- sugerir modificaciones en la secuencia de procesamiento, cuando ésta sea demasiado compleja para el operador

- evaluar la correspondencia entre formularios de ingreso de datos y las pantallas asociadas en el sistema, indicando posibles mejoras

- evaluar el tratamiento dado a los errores en los programas interactivos, sugiriendo mejoras cuando corresponda.

Controles y administración

Un proyecto desarrollado mediante el enfoque generador resulta bastante difícil de administrar, dada la naturaleza esencialmente iterativa y la duración incierta de cada actividad, además de

El analista debe preocuparse de que el usuario sea consistente en sus definiciones.

la dificultad para definir claramente dichas etapas.

En general, es posible establecer puntos de control (check-points) formales entre cada etapa del enfoque, pero el problema se presenta más bien dentro de cada etapa. En particular, la etapa de construcción del prototipo presenta graves dificultades, pues el proceso de diseño, evaluación y modificación del prototipo es de tipo iterativo y resulta difícil fijar metas intermedias.

Existen dos tipos de controles administrativos durante el desarrollo de un sistema mediante el enfoque generador: controles de tiempo de desarrollo y controles para evitar un diseño personalizado.

1. Controles de tiempo de desarrollo

Para evitar que el desarrollo del proyecto se prolongue innecesariamente, será necesario establecer los siguientes controles:

a) Fijación de metas "modulares", asimiladas al concepto general de modularidad del enfoque. Esto implica considerar al prototipo como una suma de módulos funcionales autosuficientes, en que la secuencia de desarrollo de módulos deberá ser planificada de antemano. Estas metas parciales se insertan dentro de la etapa "Construcción del Prototipo Generador".

b) Controlar y evaluar el desempeño del personal asignado a cada módulo del prototipo, incluyendo a los usuarios, estableciendo medidas de productividad tales como el cumplimiento de horas presupuestadas por módulo. Por otra parte, asignar personal a cada módulo de acuerdo a una planificación previa.

c) Requerir de los usuarios involucrados el máximo de colaboración, controlando su asistencia y participación en las sesiones de construcción, prueba, evaluación y crítica del prototipo. Se debe hacer hincapié en la importancia y necesidad de la participación de los usuarios, en todos sus niveles.

2. Controles de diseño personalizado

Para impedir el diseño de un sistema personalizado, apto sólo para el usuario que intervino en su desarrollo y con pérdida de generalidad, se hace necesario establecer los siguientes controles:

a) Controlar y verificar la consistencia de los datos del sistema. El analista debe preocuparse de que el usuario sea consistente en sus definiciones y requerimientos, señalando aquellos datos que no pertenezcan al contexto del sistema, o se definan de diferente manera en diferentes módulos del prototipo.

b) Si existen muchos usuarios para el mismo módulo, el analista deberá preocuparse de obtener un criterio común de diseño, que satisfaga los requerimientos de todos.

c) Si el usuario es único, ya sea para todo el sistema o algún módulo, se podrá solicitar asesoría externa a expertos en el área, de tal manera de dar generalidad al diseño. Esta asesoría podría incluso traducirse en la participación de terceros en las actividades de evaluación del prototipo. El uso de este tipo de controles estará sujeto al grado de confidencialidad del sistema y al criterio del jefe de proyecto encargado del desarrollo.

Conclusiones

En este artículo, el tercero de la serie dedicada al tema de los prototipos como herramienta para el desarrollo de software, hemos introducido una nueva metodología llamada "Enfoque Generador", que pretende resolver de alguna manera los problemas del uso de esta técnica en el desarrollo de sistemas reales. Falta ahora la definición formal de cada una de las etapas de la metodología propuesta, la que realizaremos en el siguiente artículo.

Como se habrá podido apreciar, se abarcaron aquí los principales temas y variables claves de

un proceso de diseño de software usando prototipos. En especial, se dedicó suficiente espacio al tratamiento del rol de los usuarios y se mostraron las características propias del enfoque propuesto. **M**

Referencias bibliográficas

[1] LOS USUARIOS EN EL DESARROLLO DE UN SIA. MECANISMOS DE PARTICIPACION.

Guillermo Beuchat S.

MICROBYTE, julio 1985.

[2] BREAKING THE SYSTEMS DEVELOPMENT BOTTLENECK.

Lee L. Gremillion & Philip Pyburn.

HARVARD BUSINESS REVIEW, March-April 1983.

Guillermo Beuchat S. es Ingeniero Civil Industrial de la U. de Chile, habiéndose especializado en el área de Informática y Sistemas. Su principal interés está en el uso de la computación como herramienta estratégica en la gestión de empresas y en la administración eficiente de los recursos de información. Se ha desempeñado como consultor independiente en microcomputadores, consultor de Auditoría Computacional y relator de seminarios de microcomputación para ejecutivos de Price Waterhouse y profesor auxiliar de computación en la Universidad Gabriela Mistral. Actualmente trabaja

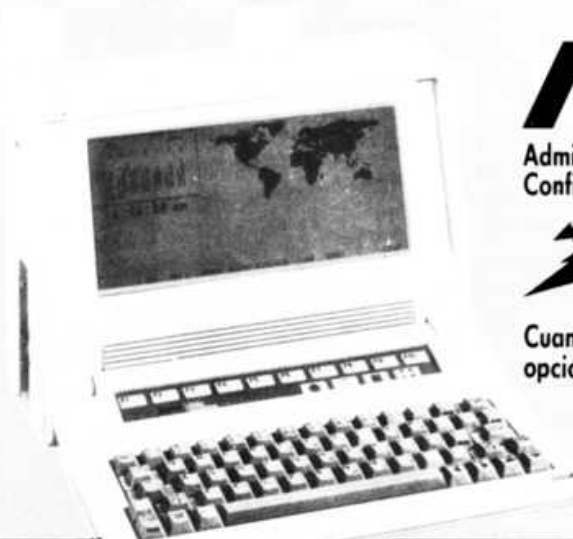


como Analista en Shell Chile SACI y es miembro del comité editorial de MICROBYTE.

UN GRAN ENCUENTRO TECNOLÓGICO ADINF Y ZENITH • DATA • SYSTEMS.

ADINF representa en Chile a una de las mejores marcas a nivel mundial: ZENITH DATA SYSTEMS. Equipos de alta calidad, de origen norteamericano, superiores a los asiáticos, y a menor costo. ZENITH tiene 65 años de permanencia en la industria electrónica. Uno de los 10 primeros proveedores mundiales de microcomputadores. El PC ZENITH es compatible con IBM® y aún así usted paga un menor costo para dar soluciones eficaces, acordes con la realidad de su empresa. Compruébelo. Conéctese con ADINF.

(*) Configuración básica.



ADINF

Administración e Informática Ltda.
Confiabilidad y experiencia.

ZENITH | data systems

Cuando la performance total es la única opción

Z-171 PC
Portátil
Precio:
US\$ 2.495 (*)
(eq. moneda nacional)

Importante aporte de un colega argentino en el área comunicaciones entre PCs.

COMUNICACIONES: LAS REDES DEL AREA LOCAL

Sergio Omar Cabello.

1. Introducción

Las redes de área local comúnmente conocidas como LANS (Local Area Networks) han proliferado rápidamente en nuestro medio.

Si bien existen en las grandes organizaciones, también están siendo adoptadas por empresas medianas, universidades y oficinas de la administración pública.

Pero aunque muchas veces hayan escuchado el término de LANS, poca gente conoce realmente lo que involucra las posibilidades de explotación de una red local, como así también desconocen las proyecciones que ofrece esta nueva tecnología.

¿Qué es una LAN?

Estas como su nombre lo indica son redes de un área geográfica pequeña (Local), las cuales facilitan la comunicación de varios computadores conectados entre sí y la explotación de dispositivos periféricos comunes a ellos.

Usualmente la explotación de una red local se limita a un radio de acción relativamente pequeño, tal es el caso de un departamento financiero por ejemplo, o tal vez deba cubrir las necesidades de todo un edificio. Pero en ambos casos sigue considerándose local.

De todos modos las limitaciones en cuanto a la distancia máxima soportada por la red, los dispositivos que puedan conectarse a la misma, la velocidad de transferencia de la información, el protocolo utilizado y el sistema operativo soportado entre otras cosas, son específicas de cada red y dependen de la compañía que manufacture ésta.

Por el momento podríamos imaginar que una LAN es algo

así como una red telefónica donde cada usuario una vez conectado a ésta puede enviar y recibir información, lo cual si bien a menudo es solamente datos, la tecnología actual le permite a estas redes transportar señales de video y de voz.

terminal le parecía que el computador lo atendía solamente a él.

En la medida que aumentaba la cantidad de usuarios, disminuía el intervalo de tiempo de atención, y la performance del sistema dependía en general

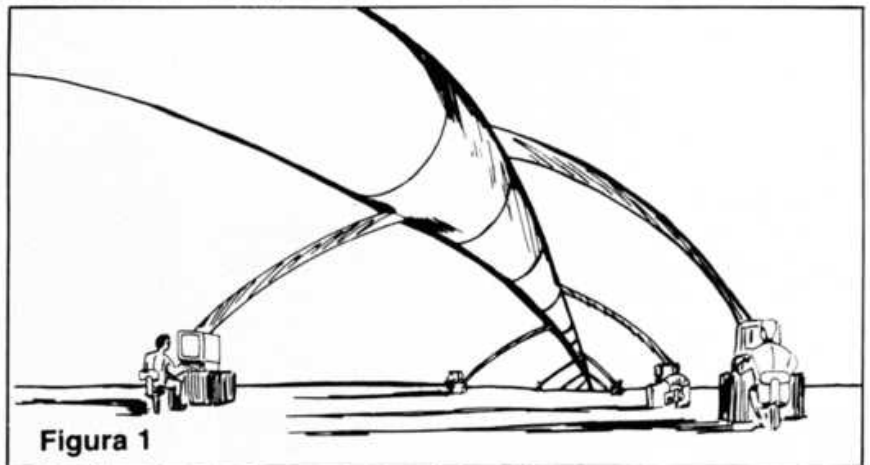


Figura 1

¿Por qué una LAN?

Creo que los argumentos para el uso de LANS que se plantean a diario, son diversos y en algunos casos contradictorios.

Aunque existen algunos puntos que son indiscutibles y que es necesario analizar.

Uno de ellos se reduce a la conocida expresión de "Time Sharing" (Tiempo Compartido), el cual dio lugar a los procesos interactivos, lo que provocó un cambio radical a los modelos computacionales por la década del sesenta, los cuales se manejaban dentro de un contexto en cuanto a procesamiento denominado "BATCH".

Ese tiempo compartido consistía en una serie de terminales no inteligentes ("bobas") conectadas a un computador central. Este atendía a cada terminal durante un intervalo de tiempo muy pequeño, pero dada la velocidad a que el procesador efectuaba este ciclo de atención, al usuario de una

casi exclusivamente del hardware del procesador central.

Poco tiempo después aparecería en el mercado el microprocesador, y con él, los microcomputadores.

Estos presentaban una gran versatilidad y un costo bastante accesible, lo que provocó una gran aceptación masiva en el mercado. Pero si bien eran muy versátiles, con capacidad de manejar poderosos utilitarios que se diseñan a diario para éstos, la modalidad de trabajo era de monousuario.

Comenzó así a surgir la necesidad de establecer una comunicación fluida entre ellos, de manera que les permitiera a los usuarios básicamente centralizar la información, compartir los recursos (caso de los periféricos de alto costo), y aumentar la velocidad de proceso basada en la idea de un procesamiento distribuido.

Si bien éstas son algunas de las pautas que marcaron la necesidad del diseño de redes,

en su mayoría son las mismas, que se plantea hoy en día el usuario de computadores personales para adoptar una LAN.

Componentes de una LAN

Las redes de área local mantienen en general un diseño estructurado, sobre el cual uno puede ir conformando la red según las necesidades específicas de cada caso.

Uno de estos componentes es el cable, el cual permite la interconexión de cada componente de la red y le da soporte a los canales de comunicación permitiendo la recepción y transmisión de la información.

El tipo de cable depende de cada necesidad y de cada red, los cuales los veremos más en detalle como medios de transporte.

Como interfaz, una pieza de hardware es localizada entre el cable y los microcomputadores (PC). Esta placa es llamada generalmente NIC (Network Interface Card).

Además se requiere de un medio de soporte para almacenamiento masivo de información (archivos y programas a ser compartidos por los usuarios).

Generalmente este medio es un disco duro (Hard Disk), el cual permite una alta capacidad de almacenamiento. Aunque algunas arquitecturas permiten el uso aleatorio (RAM-DOM) de cualquier dispositivo de almacenamiento de la red, incluyendo "Floppy disk drive".

Debido a que una red es un sistema multiusuario, ya que más de una persona puede enviar o requerir información sobre un simple PC, es requisito de la red contar con un administrador que soporte los medios a compartir, este componente es conocido como SERVER (servidor), ya que da servicio a la red permitiéndole a los usuarios compartir los recursos y mantener centralizada la información.

Este dispositivo suele ser específico de cada fabricante y pensado para ofrecer una alta performance en la explotación

de la red. Aunque suele ser posible también que esta función la cumpla cualquier integrante de la red que en su configuración incluye los elementos anteriormente nombrados (Cable, Nic, Hard disk).

De todos modos este último punto es manejado por otro componente importante de la red, y es el que da "inteligencia" al sistema, el software. Este software es llamado NOS (Network Operating System).

2. Topología de la red y protocolos de comunicación

Si bien no podemos establecer una simple solución de diseño para LAN, si podemos hablar sobre coincidencias que deben existir sobre la estructura.

La mayoría de los fabricantes de LAN han aceptado la importancia de establecer una estructura, y ellos han debido seguir el esquema para redes de ISO (International Standard Organizations), con referencia al modelo OSI (Open Systems Interconnection).

El modelo OSI no promueve o establece una norma en particular. Sus definiciones son lo suficientemente extensas para incluir muchas normas.

El modelo OSI contiene los

siguientes siete niveles o capas (Layers):

- LAYER 7 – Aplicaciones
- LAYER 6 – Presentación
- LAYER 5 – Sesión
- LAYER 4 – Transporte
- LAYER 3 – Network
- LAYER 2 – Enlace de datos
- LAYER 1 – Física



Figura 2

Estos niveles son independientes, y donde cada uno de ellos es concebido como interfaz del nivel adyacente. Por ejemplo del nivel 2 se pueden pasar datos al nivel 3 o al nivel 1, pero del 1 no se puede establecer una comunicación directa con el 3.

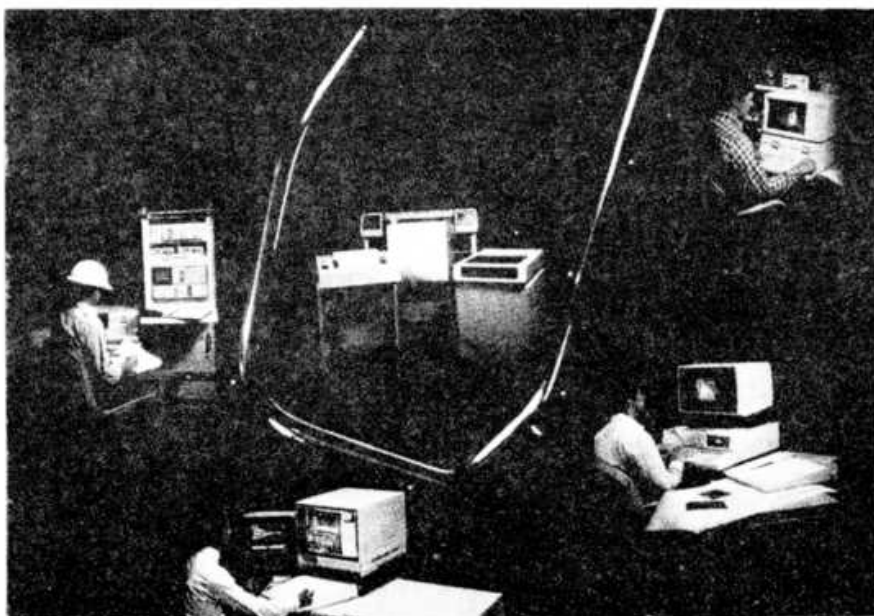
El modelo es jerárquico y la función que cumplen los distintos niveles podemos dividirlos básicamente en 3:

Debido a que las definiciones de cada uno de estos niveles son bastante áridas y teóricas, trataremos de trazar un paralelo con sus correspondientes aplicaciones.

Las etapas 1 y 2 son niveles de hardware. Ellos proveen la conexión fundamental para permitir los más sofisticados servicios. La topología y el ancho de banda (velocidad) de la red son determinadas en estos niveles.

A nivel hardware las dos arquitecturas que más predominan para trabajos en redes con PCs hoy en día son Ethernet y ARCnet. En EE.UU. el Comité del Instituto de Ingenieros Eléctricos y Electrónicos 802 (IEEE)





es el primer grupo que establece los modelos para networks.

Si bien el hecho de tener ciertos modelos en los niveles de hardware, muchas compañías han elegido modificar estos modelos y experimentar sistemas con una performance superior.

Algunas de estas compañías han trabajado para que su sistema sea aceptado como modelo IEEE, caso de IBM cuando liberó la "IBM PC Network", y ahora con su nueva red "IBM Token Ring" (IEEE 802.5).

Debido a la creación de nuevos modelos sobre los mismos niveles de hardware, es conveniente referirse a ellos como esquemas en vez de modelos.

La situación no es menos diversa en los niveles de software (Layers 3 a 7) del modelo OSI.

El nivel (layer) 4 que define la distribución de direcciones sobre la red, estableciendo el mecanismo de transferencia de información, es el comúnmente llamado nivel de protocolo.

Uno de los protocolos comúnmente usado en LAN es el XNS, Xerox Network Systems, propio para comunicaciones "peer-to-peer" (de igual a igual), que se dan por excelencia en una red de microcomputadores.

El protocolo en el nivel 4 para IBM es el SNA (System Network Architecture), el cual es esencialmente un protocolo "host-

to-terminal", pero que gradualmente ha sido adaptado por IBM para darle servicio a las estaciones de trabajo en instalaciones de inteligencia distribuida.

Este alcanza hoy en día su uso en LANs, caso instalaciones con dispositivos "gateways" que permiten la comunicación de una LAN con un computador principal (mainframe) trabajando con protocolo SNA.

La ECMA (European Computer Manufacturers Association) está trabajando con la ISO (International Standards Organization) y la oficina de modelos de EE.UU. para desarrollar su propio protocolo.

Por el momento podemos quedarnos con la idea de que en estas capas (3 y 4) administramos todo lo referente al NOS (Network Operating System), lo cual desde el punto de vista del usuario es muy importante debido a que en el NOS encuentra el control de acceso a los utilitarios de la red como así también acceso a procedimientos, y organización de datos.

En cuanto a los niveles superiores del modelo de OSI (layers 5, 6 y 7) aún se encuentran en un estado teórico, inclusive algunas confusiones se han dado a lugar con fabricantes como Xerox al presentar publicaciones de protocolos de alto nivel en estas capas.

En cuanto a las topologías, son las que provienen de las definiciones de los niveles (layers) del hardware de modelo OSI.

El costo y la flexibilidad de la instalación de la red son afectadas en parte por la topología, como así también la confiabilidad.

Muchas topologías son usadas para LAN, pero todas ellas tienen cierta similitud. Cada red usa un cable para transportar la información, este cable debe controlar el movimiento de la información sobre la red en forma confiable.

En la mayoría de las topologías las señales son transportadas en todas las direcciones desde el PC transmisor. Cada dispositivo tiene su propia dirección asignada, y el software usado sobre los dispositivos de la red acepta los mensajes con su propia dirección e ignora los demás.

Las topologías comúnmente usadas por las LANs son básicamente Star (estrella), Ring (anillo), Distributed Bus (barra distribuidora) y Token Bus.

La topología Star (figura Nº 3) tiene un cable por separado para cada PC de la red donde cada cable es conectado al procesador central de éste (server). Esta topología es muy usada en redes "host-to-terminal", y en sistemas PBX, pero poco usada en Local Area Network (LAN).

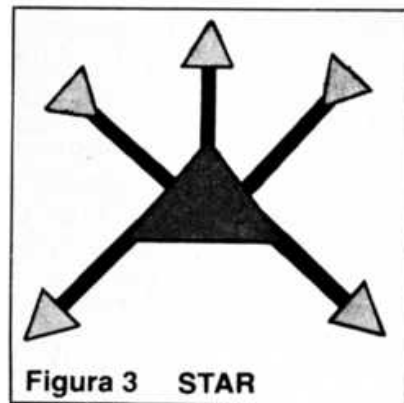


Figura 3 STAR

En la topología estrella (Star), el procesador central envía la información desde sus fuentes a sus destinos (cualquier otro PC o dispositivo periférico).

El mensaje o información llega al PC a través del Server, y el PC reconoce el mensaje. Si el PC no reconoce el mensaje, el Server lo envía nuevamente.

Las ventajas de esta topología son que la conexión del hardware es simple y que a veces existiendo cables telefónicos éstos pueden ser usados como medios de transmisión.

Asimismo, esta topología tiene varias desventajas al ser usada en "local area networks".

Entre las desventajas podemos decir que bajo esta topología cada PC debe ser conectado al procesador central con un cable dedicado, por lo que se utiliza mucho más cable que en otras topologías.

La instalación de un PC adicional a la red es más dificultosa debido a que un nuevo cable debe ser instalado desde el procesador central al PC.

Otra desventaja que existe (punto muy vulnerable en la red) es el procesador central (server), ya que si éste falla toda la red queda inoperante. En el caso de que las estaciones de trabajo sean inteligentes, al fallar el server éstos pueden seguir trabajando con operaciones "stand-alone", pero generalmente no pueden asumir el rol del server.

La topología Ring (figura N° 4) es básicamente un sistema cerrado, el cable pasa a través de cada PC y periférico de la red, y las terminaciones se unen formando un anillo.

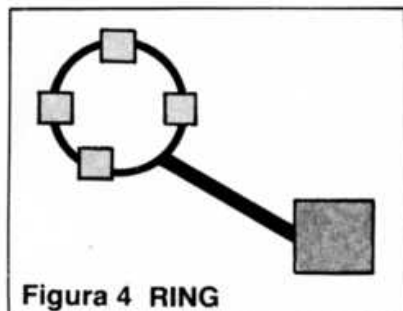


Figura 4 RING

Con esta topología la red está propensa a "caídas", ya que si un PC de la red falla, toda la red cae y la operatoria bajo la red se detiene.

Este problema puede ser solucionado haciendo trabajar

dos anillos en paralelo, pudiendo efectuar un "by-pass" de la máquina que se cae enlazando la apertura del otro anillo.

La topología de barra distribuidora (Distributed Bus) es la más frecuentemente usada en LANs.

El Bus es un cable simple que hace de "camino" para la información a través del área de trabajo. Esta topología es altamente segura y flexible, cualquier falla en un dispositivo de la red no afecta a la operación de la misma, pero fallas del cable provocarían la caída de la red.

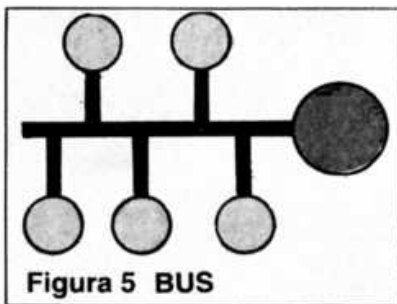


Figura 5 BUS

La topología "Token Bus", también llamada "Distributed Star", se asemeja a la "Distributed Bus" en que ambas usan un simple bus. Sin embargo, la Token Bus usa unos dispositivos de interfaz especiales en el Bus llamados HUBs, los cuales son conectados al bus en los puntos convenientes.

Cables dedicados corren desde estos HUBs a los PCs de la network como muestra la figura N° 6.

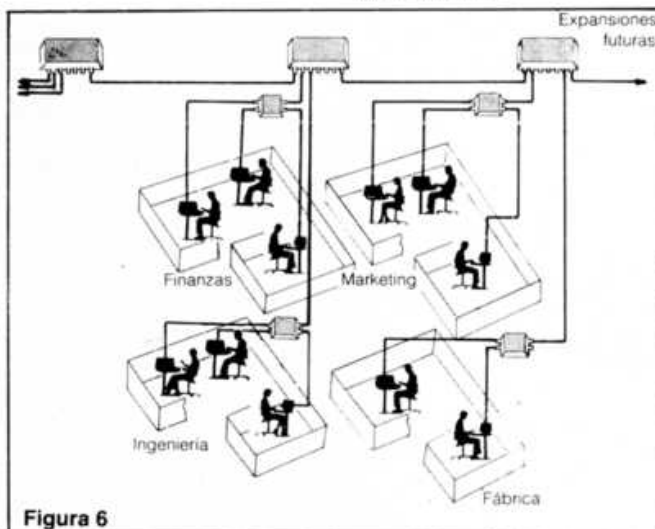


Figura 6

3. Esquemas de acceso

Como se mencionó anteriormente, cualquier topología que usa un BUS común debe de alguna manera regular el acceso a este BUS. Dicho de otra forma, debemos prevenir que dos PCs efectúen transmisiones simultáneas o hagan uso del BUS indiscriminadamente.

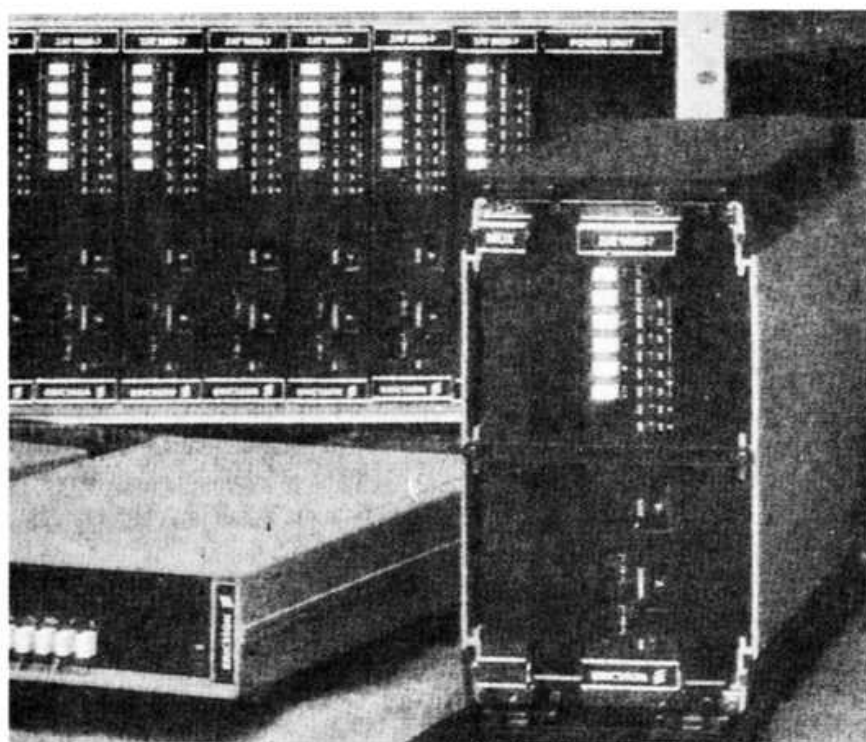
Actualmente dos métodos de acceso son utilizados:

- * Carrier Sense Multiple Access (IEEE 802.3 CSMA/CD)
- * Token Passing (IEEE 802.4 Token Bus) (IEEE 802.5 Token Ring)

El CSMA es usado por Ethernet y el "Token Passing" por ARCnet.

En el caso de CSMA (Acceso Múltiple por senseo de Portadora), una estación que desea transmitir "escucha" el medio para determinar si otra transmisión se está efectuando. Si el medio se encuentra libre, la estación transmite. De lo contrario la estación espera un período de tiempo e intenta nuevamente.

Posteriormente a la transmisión, la estación espera un tiempo razonable para recibir de la estación receptora una señal de reconocimiento, en caso de no recibirla, la estación transmisora presume que el paquete no llegó a destino (tal vez por una colisión) y vuelve a transmitirlo.



La mayoría de las redes CSMA incluyen la Detección de Colisiones (CD) como parte del método de acceso (CSMA/CD). Esta técnica pretende solucionar el problema que se presenta al chocar dos paquetes de información en el medio de transporte, ya que éste permanece inerte durante la transmisión de estos paquetes.

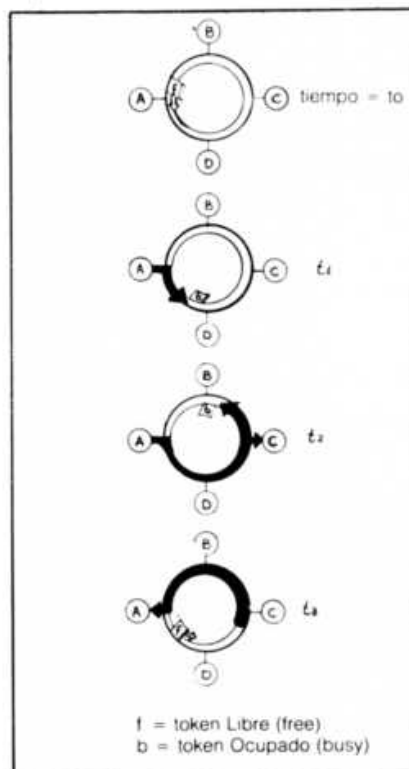
El CSMA/CD reduce esta pérdida, ya que mientras un PC transmite, éste continúa escuchando al medio para verificar que la transmisión no sea interrumpida. Si otro PC transmitió durante este tiempo, el resultado de la colisión será detectado. Cualquier PC que detecta la colisión envía una señal de alerta para que se retransmita después de un intervalo predeterminado.

En cuanto al "Token Passing" es un método de acceso que se relaciona estrechamente con algunas topologías cerradas del tipo de anillo (ring), ya que esta estructura tiene implícito un ordenamiento cíclico.

Sin embargo se adapta también en topologías BUS ya que se puede hacer que el BUS se convierta en un anillo lógico (Token Bus).

Este método de acceso se puede ver claramente en una

configuración del tipo "Token Ring", en la cual la técnica de acceso está basada en el uso de un pequeño paquete conocido como "token", que circula alrededor del anillo. Cuando todas las estaciones están ocupadas y ninguna desea transmitir, el paquete "token" es reconocido como libre ("free"), resultado del reseteo de un bit de status (figura N° 7 tiempo = t_0).



Si una estación desea transmitir, detecta el paso del "token", modifica la estructura de los bits de status y lo pasa de libre ("free") a ocupado ("busy") transmitiendo inmediatamente siguiendo el paso del "token ocupado" (tiempo = t_1).

Debido a ello en tiempo = t_2 no hay token libre ("free") y si otra estación desea transmitir, deberá esperar hasta que el token dé una vuelta completa en el anillo y sea sacado por la estación transmisora, quien insertará un nuevo "token libre" en el anillo (tiempo = t_3).

Bibliografía

1. Using Netware: Michael Durr & Bill Lawrence, Que Co., 1986.
2. Networking IBM PCs: Michael Dur, Que Co., 1985.
3. Local Area Network: James Harry Green, Scott Foresman & Co., 1985.
4. PC - Volumen 4 N° 3, febrero 1985.
5. IDM (International Distributor Meeting), Multitech Co., 1985.
6. PC Week Volumen 3 N° 16, abril 1986.
7. Telecommunications Vol. 20 N° 3, marzo 1986.
8. Mundo Electrónico, N° 162/3, junio-julio 1986.



Sergio Omar Cabello. Cursó sus estudios de Ingeniería Electrónica y Electricidad en la Universidad de Mendoza (1982). Realizó diversos trabajos de automatización con microprocesadores, especializándose más tarde en comunicaciones y teleprocesamiento de datos. Actualmente se encuentra a cargo del Departamento de Investigación y Desarrollo de LATINDATA S.A. en Argentina, realizando tareas en el área de comunicaciones, para el enlace de los computadores personales LATINDATA PC.

– Aportamos Soluciones Integrales



En comunicación de datos, en sistemas periféricos, en software y equipos complementarios, definimos nuestra gestión como un aporte de soluciones integrales. Porque analizamos cada necesidad en forma rigurosa, porque diseñamos y evaluamos con criterios objetivos los sistemas adecuados, porque nos hacemos responsables de la implementación total de estos sistemas y porque ofrecemos el respaldo permanente de nuestros ingenieros especializados. En COASIN operamos con las empresas líderes mundiales en módems, redes de conmutación de paquetes y terminales directamente compatibles con IBM. En COASIN contamos con la ingeniería y los recursos necesarios para desarrollar equipos y software que permiten integrar computadores a redes de télex y sistemas de control. Sea con tecnología propia o importada, en COASIN aportamos soluciones integrales.

Comunicación de datos: RACAL MILGO - RACAL VADIC
 Equipos y sistemas para conmutación de paquetes: TELENET
 Terminales directamente compatibles con IBM: TELEX COMPUTER - DECISION DATA
 Automatización de oficinas: CPT CORPORATION
 Sistemas ininterrumpidos de energía UPS: EMERSON - TOPAZ
 Interfases para redes télex: COASIN.


SOLUCIONES INTEGRALES
 Holanda 1292 • Fono 2250643 • Santiago

IMPRESORA EPSON

EX-800

EX-1000

Porcentaje de ventas por local

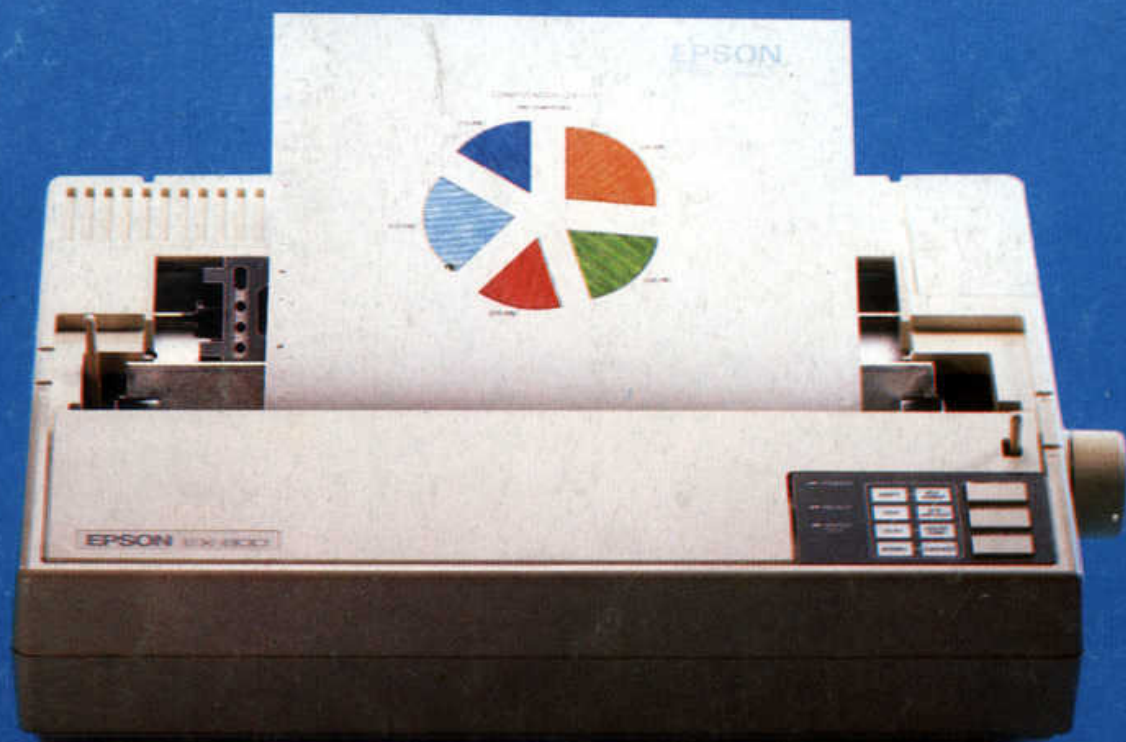


- **MATRIZ DE PUNTOS**

- 300 c.p.s. impresión normal
- 60 c.p.s. calidad de letra
- ancho en columnas

	EX-800	EX-1000
pica	80	136
comprimida	132	233

- tractor bidireccional
- opcional dispensador hojas sueltas
- interfaces CENTRONIC y RS-232
- opción buffer 32 Kb
- opción de impresión normal o en colores utilizando el COLOR OPTION KIT



EPSON

EPSON Chile S.A.