

MICROBYTE

TODO COMPUTACION

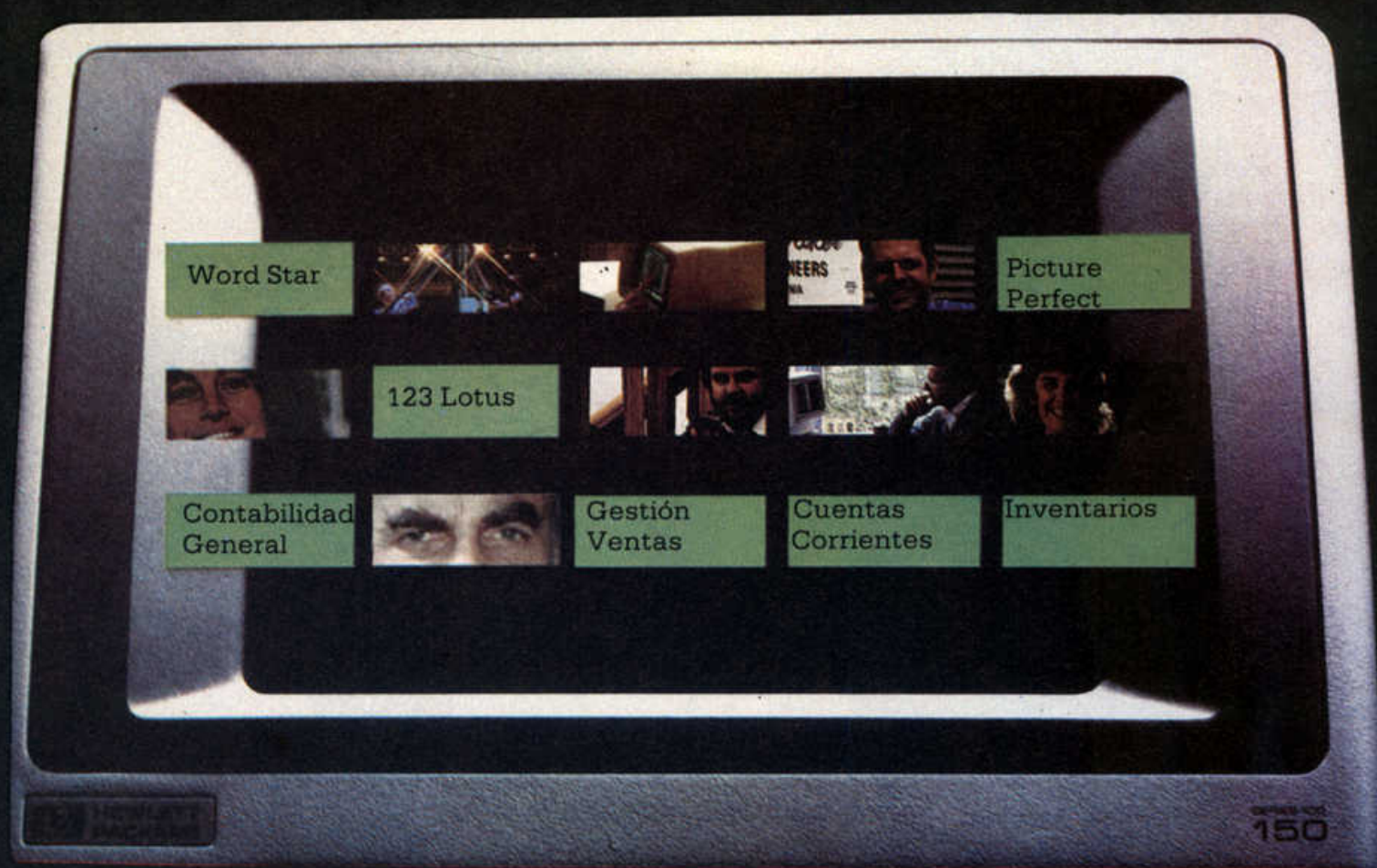
ENERO 1985.
Nº 9 \$ 150



dBase II: Conozca un administrador de bases de datos

LINC: Una herramienta de 4.^a generación Biblioteca de rutinas

HEWLETT-PACKARD



OLYMPIA

OLYMPIA (CHILE) Ltda. - Avda. Rodrigo de Araya 1045 ☎ 225 50 44 - 39 22 43 - Santiago.

| | | | | | | | | | | | |
|---------|---------|-------------|-----------|------------|-----------|----------|---------|------------|---------|--------|-------------|
| ARICA | IQUIQUE | ANTOFAGASTA | V DEL MAR | VALPARAISO | SANTIAGO | RANCAGUA | TALCA | CONCEPCION | TEMUCO | OSORNO | PUNTA ARENA |
| 3 28 74 | 2 66 56 | 22 23 25 | 68 07 32 | 25 67 72 | 225 50 44 | 2 42 15 | 3 34 11 | 2 17 02 | 3 17 82 | 49 56 | 2 15 37 |
| | 2 31 84 | | | | 39 22 43 | | | | | | 2 51 34 |



Foto Portada

Feliz Año les desea el compañero de todos los días, Salud!!!

Director Responsable
Jorge Carrera R.
Coordinador Técnico
José Kaffman T.
Director Publicidad y RR.PP.
Ariel Leporatti P.
Ventas
Orietta Bustamante
Redacción Periodística
Myniam Pinto M.
Directora Arte
Paz Barba
Fotografía
González y Geoffroy
Cuerpo Editorial
Jaime Aravena
Jorge Cea
Carlos Contreras.
Corresponsales en el exterior
Luis Kaffman T. (Londres)
Alfredo Zarowsky (Paris)
Victor Kahan (Ohio)
Fotocomposición
Laser Ltda.
Representante Legal
Jorge Carrera R.
Dirección: Merced 346 - Of. F
Fono: 393866
Distribución
Antarctica S.A.
Impresión
Tamarcos, quien sólo actúa como impresor

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 suscripciones semestral:

Correo Certif. Stgo. y Prov. \$ 850
Entrega por mano Stgo. \$ 800

Valor suscripciones anual:

Correo Certif. Stgo. y Prov. \$ 1.600
Entrega por mano Stgo. \$ 1.500

Solicite un representante al fono 393866, en Merced N° 346, Of. F. Santiago - Chile.

3
Editorial

21
dBase II:
Conozca un eficiente administrador de bases de datos.

4
Noticias
Novedades

27
Bienvenidos
al Basic

9
Evaluación
de Proyectos

29
Sonda:
Los 10
primeros años

14
Programando el Z-80
8ª Parte

35
OPEN FILE
Cartas del lector

18
Biblioteca
de rutinas

38
Sección por
Marcas
Atari -
Timex Sinclair

Ante las innumerables consultas llegadas a nuestra redacción referentes a retribuciones por colaboraciones publicadas, éstas se pueden dividir en dos categorías. Por programas publicados en las secciones por marca o como apoyo en otras secciones, se pagará \$ 2.500. Los artículos de mayor contenido, serán pagados a \$ 6.000 sin importar la cantidad de páginas, sólo tomando en cuenta su calidad. En principio, nuestras intenciones eran poder retribuir mejor a nuestros colaboradores, sin embargo la dura realidad económica nos ha impedido hacerlo.

Por otro lado, nos hemos visto inundados de correspondencia de nuestros lectores y un sinnúmero de colaboraciones a las que aún no hemos podido dar curso por lo que les rogamos disculparnos.

Rainbow 100

Doble capacidad, doble versatilidad y doble simplicidad para un mismo precio... o casi.

El nuevo computador personal DEC Rainbow 100 de DIGITAL es, sencillamente, sorprendente.

Sorprendente en su diseño ergonómico, largamente estudiado para facilitarle y ordenarle a usted su operación, acuciosamente construido para adaptarse a cualquier lugar de trabajo.

Sorprendente en su ingeniería.

El Rainbow 100 incorpora en forma standard 2 procesadores, de 8 y 16 BITS respectivamente. Esto le permite a usted multiplicar las aplicaciones disponibles, haciendo uso del experimentado y abundante software de 8 BITS para CPM/-80, incorporando todo el emergente software de la nueva y revolucionaria arquitectura de 16 BITS en CPM/-80 o en MS-DOS.

Pero lo que quizás a usted más llame la atención sea su sorprendente versatilidad y facilidad de uso. El Rainbow 100 le instruye a usted todo lo que necesita saber de su operación, mediante programas de instrucción especialmente incorporados a su sistema, evitándole la lectura de tediosos y voluminosos manuales.

Su impresionante capacidad le permitirá abordar y resolver en él sus problemas de administración de información, de contabilidad, de finanzas, de control de producción, de cuentas corrientes, de planificación, etc.

Finalmente, el versátil Rainbow 100 puede además transformarse en un terminal de los computadores centrales de su empresa, o multiplicar enormemente sus tareas, mediante la incorporación de la más completa gama de periféricos y equipos auxiliares.

Reconocemos que el Rainbow 100 tardó en aparecer en el nuevo y sorprendente mercado de los "personal computers" ...pero pensamos que valió la pena esperar.

Entrega inmediata.


Rainbow 100
Personal Computer



 **SONDA**

Sociedad Nacional de
Procesamiento de Datos Ltda.
CASA MATRIZ: Teatinos 574
Fono: 62277 Santiago - Chile.

digital

Digital
Equipment
Corporation
U.S.A.

Para todo aquel que se ha interesado en la dramática invasión de los computadores personales a prácticamente todos los aspectos de la vida cotidiana, 1985 representará sin duda uno de los hitos importantes en esa campaña.

Si quisiéramos resumir cómo se desarrolló ésta, bastaría con mencionar tan sólo unos pocos elementos y fechas.

- 1966 ve nacer los primeros clubes y asociaciones de entusiastas que prácticamente con desechos se dedican a armar primitivos computadores personales, capaces de procesar un set mínimo de instrucciones binarias.

- 1975 es el año en que estos pioneros comienzan a comercializar sus primeros productos, fundamentalmente kits para armar sin memoria ni interfaces. Rápidamente, estos kits se desarrollan dando lugar a los conocidos Apple, Morrow, Cromemco, Texas TI 99/4, Atari y otros. El éxito es fenomenal convirtiéndose empresas de garage en enormes consorcios.

- 1981 es el principio del fin de las empresas de garage. Los gigantes de la computación, con IBM a la cabeza, finalmente reconocen en el computador personal a uno de los más importantes componentes de lo que será la sociedad informática. La introducción del PC de IBM, comienza a arrasar con una gran cantidad de sus más pequeños competidores, dando inicio al fin de los pioneros.

- 1984 es testigo de la más desigual competencia. Apple Computers, símbolo de las empresas de garage, que ha logrado aumentar sus ventas de cero en 1975 a mil millones de dólares en 1984, se enfrenta con el símbolo de los gigantes, IBM, con ventas anuales de 40.000 millones de dólares. Finalmente, para sobrevivir, Apple ha debido buscar alianzas con otro coloso de la buromática, Xerox. El primer paso, ha sido que Apple ha pasado a Xerox la distribución de sus equipos en Canadá, Francia y Latinoamérica.

- 1985 es pues el comienzo de una nueva era. La iniciativa en el desarrollo de la microcomputación ha quedado en manos de los gigantes. El fin de la creatividad para algunos, el comienzo de un desarrollo armónico y racional para otros. Sólo el futuro dará la razón a unos sobre los otros.

La elección fundamental para nosotros, como revista técnica eminentemente especializada, es que la introducción de los computadores personales a todos los aspectos de la vida cotidiana, está sólo en sus primeros pasos. En un plazo, que creemos breve, no habrá empresa en la cual sus ejecutivos no cuenten cada uno con un computador. Conocer su manejo y poder sacarle provecho serán elementos que incidirán directamente en la productividad de los ejecutivos y sus empresas.

Por otro lado, el futuro de los computadores personales está íntimamente ligado a su capacidad para interconectarse en un ambiente de comunicaciones integral. Esto, que es innegable en el ámbito de las empresas, lo es también en el campo del hogar y de la educación. La creciente intromisión de empresas tradicionales del ámbito de las comunicaciones como AT&T e ITT en el campo de la computación y el continuo esfuerzo de IBM por captar un espacio en el terreno de las comunicaciones mediante la compra de empresas dedicadas a esa área como Rolm y Satellite Business Systems son claves importantes para predecir el próximo desarrollo en nuestra área.

Así suponemos será este nuevo año y no nos queda más que desear a todos nuestros lectores, nuestros mejores deseos y que pasen unas felices vacaciones.

Volveremos a estar con ustedes con el número de marzo, el que como ya es tradicional estará en los kioscos a partir de la última semana de febrero. Felicidades.

NOTICIAS

NOVEDADES

Las vueltas de un diskette.

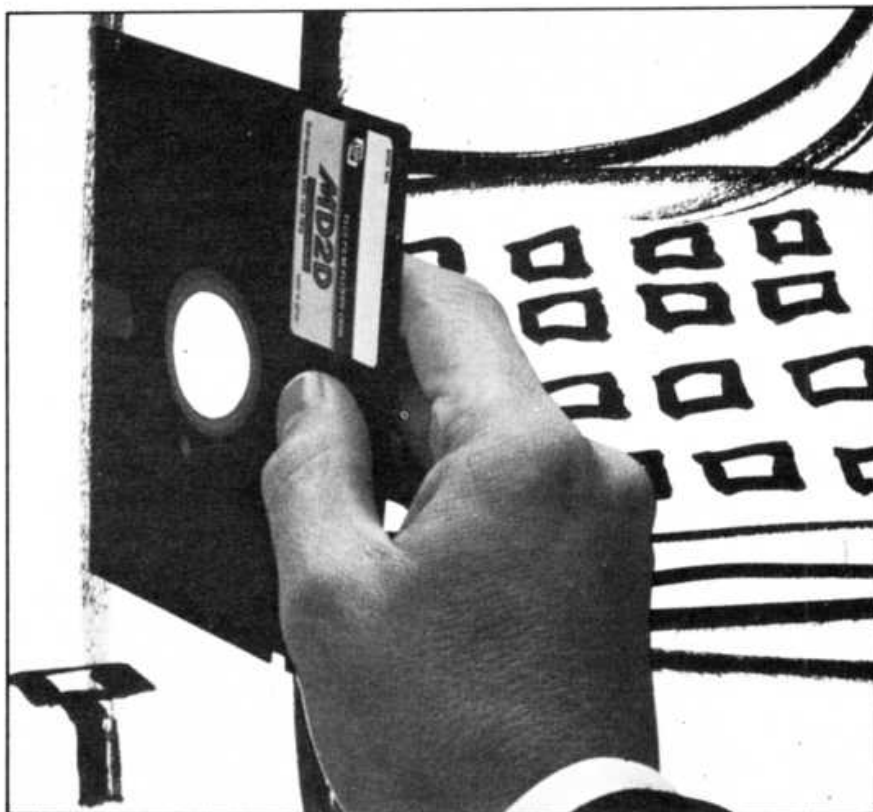
El auge que han tenido los computadores personales, se está traduciendo finalmente en un alto crecimiento en el mercado de los suministros, especialmente diskettes.

En efecto, el crecimiento en la venta de diskettes ha sido geométrico desde 1978, en que se vendieron diskettes por un valor de US\$ 74 millones, mientras que en 1983 esta suma ascendía a los US\$ 600 millones.

Dataquest, una firma consultora norteamericana, estima que de una venta de 400.000 unidades en 1983, ésta aumentará a 600.000 en 1984 y a un billón en 1986.

Sin embargo, no todas las empresas dedicadas a esta área han profitado de este aumento de ventas. Verbatim, que controla la mayor parte de este mercado, anunció que cerrará con pérdidas el tercer trimestre de este año, mientras que Dysan que ya lleva tres trimestres con pérdidas anunció que traspasaría sus activos a Xidex. La creciente competencia ha hecho bajar los precios por mayor de los diskettes de 5,25 pulgadas a aproximadamente un dólar la unidad, un 25% menos que el año pasado. Al detalle, en Estados Unidos el valor de los diskettes fluctúa entre 2 y 8 dólares de acuerdo a la marca.

A la ya larga lista de fabricantes de diskettes, recientemente se sumó también Kodak, la que produce internamente un sólo tipo de diskette con capacidad para más de tres millones de caracteres. Para completar su línea, Kodak comprará diskettes a otros fabricantes, especialmente Dysan y Xidex, los que comercializará posteriormente con su propia etiqueta, Kodak confía que su marca, de reconocido prestigio en el área fotográfica, le permita ganar también un importante segmento en el área de los medios magnéticos. Este fue el caso al menos con 3-M, quien ya ocupa el segundo lugar como proveedor de este tipo de medios.



IBM instalaría nueva planta en México

La legislación mexicana hasta ahora, sólo permitía la instalación de industrias microelectrónicas que contaran con una participación mayoritaria de capitales mexicanos.

Este es el caso de Apple Computers y Hewlett Packard, quienes al instalar plantas en México, habían visto limitada su participación a sólo un 49% de éstas.

El caso de IBM, que está aún en discusión, le permitiría a éste controlar el 100% de las acciones de la subsidiaria mexicana.

IBM produce en Guadalajara el Minicomputador Sistema 36 desde 1980 y esto hasta ahora no había significado ningún reclamo de sus competidores pues la legislación mexicana autorizaba la posesión del 100% de las acciones de las empresas que fabrican minis o mainframes.

Esta nueva política del gobierno mexicano, que en principio estaría aceptando las condiciones de IBM, responde sobre todo a la urgente necesidad de atraer un mayor volumen de inversiones extranjeras.

A pesar de que la planta proyectada en México, es diminuta comparada con el volumen global de producción de IBM en el mundo, tendría la capacidad de producir alrededor de 100.000 computadores personales al año, incluyendo el nuevo PC AT, cantidad suficiente para abastecer el mercado latinoamericano y parte del asiático. En Estados Unidos, IBM piensa introducir este año alrededor de 1,2 millones de equipos.

Hasta ahora, entre las veinte diferentes empresas que fabrican computadores en México, su producción anual no subía de las 20.000 unidades.

Computadores con oídos

Desde hace varios años que se conoce la tecnología necesaria para instruir a un computador para reconocer lenguaje hablado. Sin embargo, hasta ahora estos sistemas eran poco poderosos, reconocían tan sólo unas pocas palabras y debían ser instruidos palabra por palabra por la persona que lo usaría posteriormente. Como resultado, el uso de esta tecnología derivó principalmente hacia juegos y uso de aficionados.

Marconi Space & Defense Systems, encontró un mejor uso y ya hace dos años que comercializa un sistema que reconoce hasta 240 palabras para ser usado por pilotos, quienes durante el vuelo pueden cambiar la frecuencia de sus comunicaciones mediante instrucciones orales.

Texas Instruments, en su modelo Professional, introdujo el concepto de reconocimiento de voz para aplicaciones comerciales y ahora IBM, reconociendo las posibilidades de esta tecnología anunció que había desarrollado un sistema capaz de reconocer hasta 5.000 palabras con un 95% de precisión.

El sistema de IBM, que por el momento corre en un mainframe mediano, el 4341, requiere

de que las palabras sean dictadas de a una con una pequeña pausa entre ellas. El computador revisa unos veinte parámetros diferentes cada centésima de segundo, comparando cada set de éstos con una serie de sonidos que él conoce. Luego, de acuerdo al contexto, va eligiendo las palabras posibles. De este modo puede diferenciar palabras que suenan igual en inglés tales como "to", "two" y "too". En la medida que son ingresados más sonidos y palabras, el computador va reevaluando sus resultados anteriores.

Los usos de esta tecnología pueden llegar a ser de los más variados. ITT y Ericsson ya han anunciado entre sus productos un aparato telefónico al cual se le dicta el número con el que se quiere establecer la comunicación sin necesidad de discar. NEC, por su parte exhibió, recientemente el sistema DP-200 que cuenta con un vocabulario de 150 palabras que permite traducir directamente del japonés al inglés.

IBM se vuelca al campo de software

Sin duda, hasta hace unos años, aún era posible para cualquier persona con una buena dosis de suerte e ingenio, hacerse de una pequeña o gran fortuna en un mínimo de tiempo escribiendo algún programa novedoso para computadores personales.

Los tiempos sin embargo han cambiado y los nombres de Gates de Microsoft, Kildall de Digital Research o Kapor de Lotus, lo más probable es que pasen a la historia como los últimos en hacerse una fortuna de ese modo.

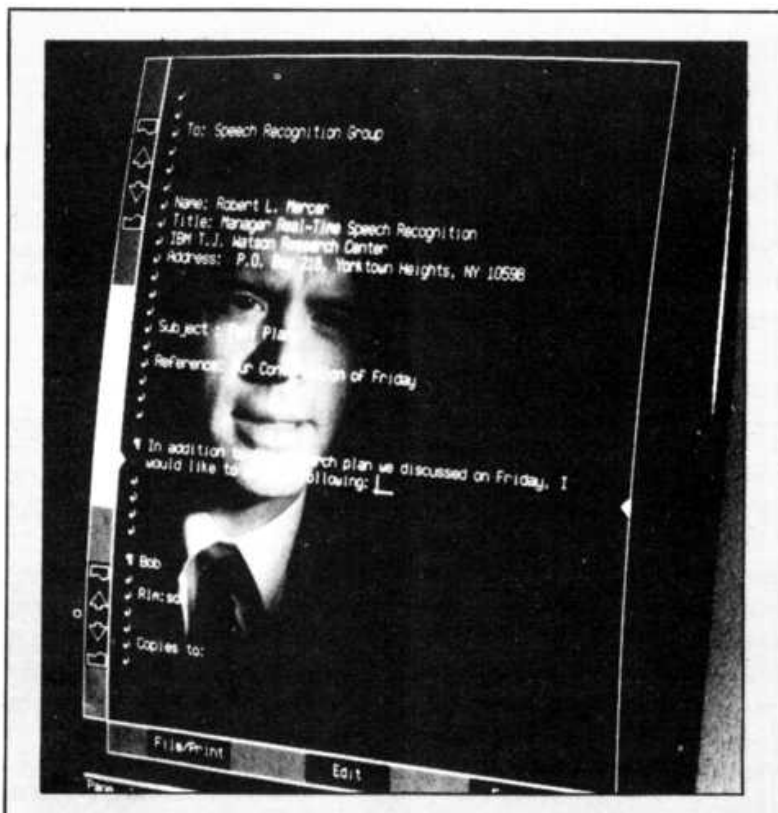
Hoy, además de un programa ingenioso, se requieren de millones de dólares para darlo a conocer y defenderse de los miles de programas que surgen y que se reclaman de hacer lo mismo si no mejor y más barato.

Con el ingreso de IBM a la arena, el terreno se le ha puesto aún más difícil a los productores de software. En efecto, IBM creó una unidad independiente encargada de la producción de software, la IBM Information Services y a un año de su formación ya ha dado a conocer dos series de programas para aplicaciones administrativas y comerciales.

Transtecnia.

Transtecnia Chile, es una nueva empresa cuyo objetivo es suplir algunas carencias en el mercado de la microcomputación, tanto en insumos específicos, cajas porta-diskettes, fundas para computadores, etc., como en apoyo a la elección de software e incluso desarrollo de Sistemas de Información Administrativos.

Formada por profesionales del área, Transtecnia cuenta con software para diversos sistemas operativos, destacando entre ellos, CP/M y MS-DOS. Mayores informaciones en Huérfanos 1022 of. 607 tel. 723036.



NOTICIAS NOVEDADES

Sperry y Control Data penetran en Brasil

Preocupada por la excesiva importancia de IBM en el sector mainframes en Brasil, la Secretaría Especial de Informática de ese país está incentivando la penetración de algunos de sus competidores para balancear el mercado.

En efecto, IBM controla entre un 85 y 90% de ese segmento, considerado estratégico por la SEI, por lo que está incentivando la participación de otras empresas en conjunto con industrias brasileñas.

Control Data está participando con Brasilinvest en la producción de mainframes de la línea Cyber 180 dirigidos a la banca y transmisión de datos. En la sociedad, Control Data participa con un 25%, Brasilinvest con un 20% y el resto serán aportes de tres bancos, los que en conjunto tienen una red de casi 1.600 sucursales, por lo que este mercado que se considera cautivo para esta sociedad por sí solo puede asegurar su viabilidad económica. La idea, es introducir a los bancos computadores Cyber 810, 830 y 830D como centros de la red y equipar a todas las sucursales con terminales remotos.

Por su parte, Sperry traspasó su división Univac en Brasil a la empresa Novadata, en Brasilia, la cual continuará bajo licencia produciendo los equipos de Sperry en Brasil.

AT & T también se encuentra en conversaciones con la SEI para licitar su sistema Unix a alguno de los fabricantes de mini-computadores brasileños.

Control de tránsito

Los problemas de congestión de tránsito en Tokio están siendo solucionados con un original sistema que redundará en menores costos y un mejor servicio a los pasajeros. Esa es al menos la intención de las autoridades japonesas, las que han puesto en práctica un sistema computarizado que permite regular el tráfico de buses urbanos utilizando avanzados sistemas de comunicaciones.

Fundamentalmente, el sistema consiste en llevar un permanente control de la localización de cada uno de los buses que recorren determinada ruta. Esta información es analizada en un computador central y en base a ésta es posible ir dando instrucciones a los choferes para que apuren o disminuyan la marcha, de acuerdo a las condiciones de tráfico y la distancia entre cada uno de los buses.

Uno de los pasos siguientes es coordinar esta información con el control de semáforos para hacer más expedito el avance de los buses. De acuerdo a la división de transportes, el costo de instalación de este sistema en una línea piloto fue de US\$ 5 millones, e incluso a ese costo el sistema ha sido rentable en términos de economías.

Los buses llevan unas pequeñas antenas por las que emiten señales que identifican al bus y al conductor. Estas señales son captadas en los paraderos y de ahí transmitidas a través de líneas telefónicas al computador central con una velocidad de 50 bits/sec. En los paraderos, por su parte, fueron instalados postes con pequeñas pantallas que además de proporcionar información a los conductores, le informa a los pasajeros cuánto tiempo deberán esperar al próximo bus.



Lo que dicen las encuestas

En una reciente encuesta, Yankee Group, una empresa consultora de Boston, descubrió que aquellos que compran computadores para la casa, los están usando para hacer trabajos de oficina en el hogar mas que para jugar con ellos.

Si bien la encuesta completa tiene un precio de US\$ 70.000, algunos de sus resultados fueron revelados a la prensa. Basados en entrevistas a recientes adquirentes de microcomputadores IBM, RadioShack,

Apple, Commodore y Atari; Yankee Group descubrió que:

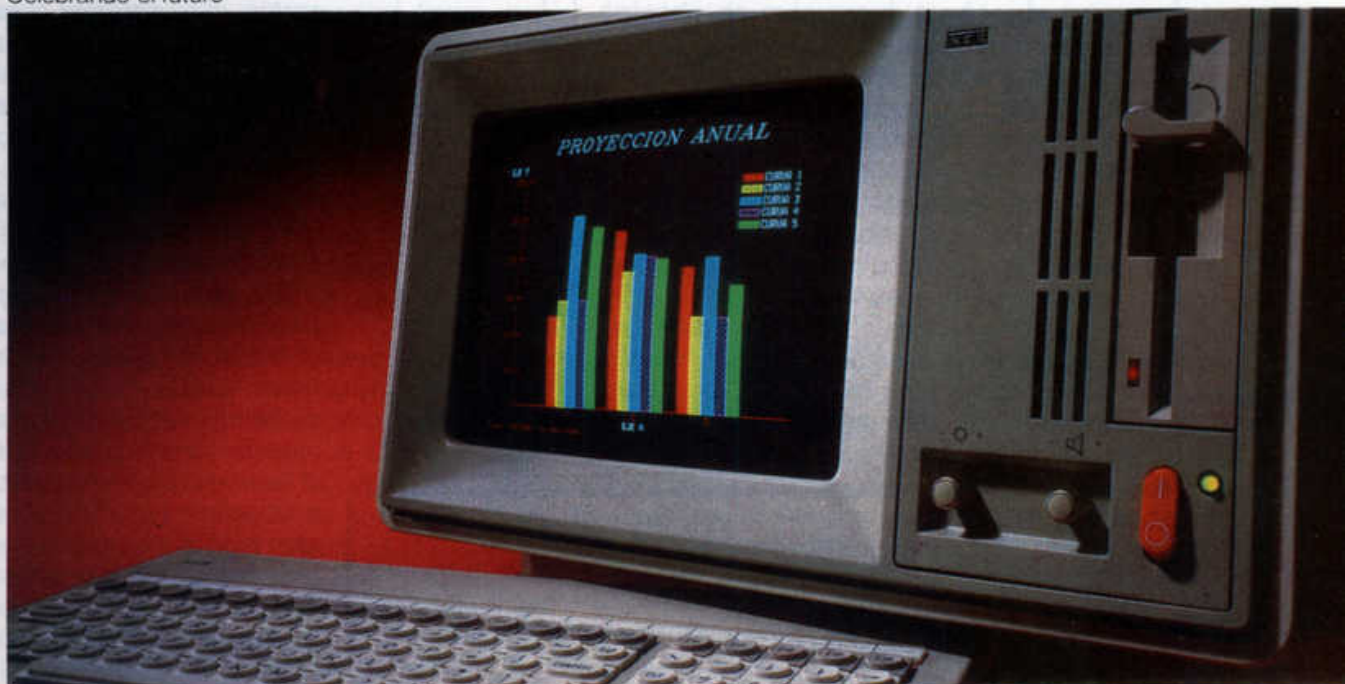
- Si bien tan sólo un 4% afirmó haber adquirido los equipos para hacer trabajos de oficina en el hogar, un 20% ya lo está haciendo y un 49% afirmó que lo hará dentro de los dos próximos años.

- Un 45,8% compró los equipos para aprender a usarlos, correr programas educacionales y juegos. Menos de un 1% lo hizo para aplicaciones tales como correo electrónico o co-

nectarse a su banco.

- Un 20% de quienes poseen un computador en la casa, poseen más de uno y un 25% piensa adquirir otro más dentro de un período próximo.

- Por último, en promedio, los dueños de computadores pasan junto al teclado más de 8 horas a la semana y comparten 3,5 horas menos con sus familiares que el resto de los mortales.

NCR1884-1984
Celebrando el futuro

ERA DE ESPERAR QUE EL COMPUTADOR PERSONAL NCR FUERA ASI.

Con todas las ventajas de la tecnología y experiencia de NCR.

- Procesador de datos de 8 bits o de 8/16 bits
- Procesador destinado a gráficos
- Pantalla con alto poder de resolución
- Teclado fácil de utilizar (numérico y alfanumérico)
- Memoria principal de hasta 512 KB.

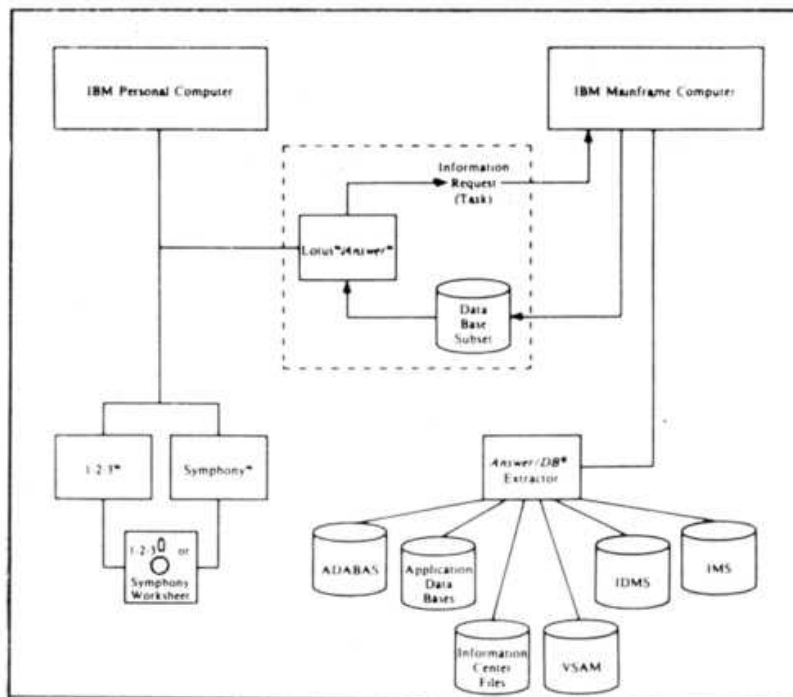
Fácil de manejar y con un alto rendimiento en el procesamiento de datos.

Era de esperar de NCR, una empresa que cumple 100 años innovando.

Lo esperamos con nuestra red de distribuidores.

NCR1884-1984
Celebrando el futuro

Mac-Iver 370 - Fono 380013 - Santiago



Lotus/Answer

La invasión de computadores personales a terrenos hasta ahora exclusivos de minis y mainframes le ha causado más de un dolor de cabeza a los gerentes de sistemas, quienes además de tener que lidiar con sus tradicionales problemas ahora deben satisfacer los deseos de algunos ejecutivos que quisieran extraer alguna información de un computador central para procesarla en su Lotus 123 o Simphony.

El problema menor en ese caso es encontrar un canal para conectar al mainframe otro terminal remoto. Lo terrible luego, es tener que manipular la información para hacerla accesible al formato del microcomputador y luego instruir al ejecutivo para usar todo el sistema.

Con Lotus/Answer, el único trabajo que queda es precisamente el más fácil; encontrar el canal disponible. Todo el resto lo hace este ingenioso paquete de software.

Fibras ópticas

American Telephone & Telegraph Co. (AT & T) anunció que invertirá US\$ 2 billones hasta fines de esta década, para expandir su red de comunicaciones de larga distancia utilizando fibras ópticas.

Con este presupuesto, AT & T tenderá alrededor de 21.000 millas de cable de fibra óptica, constituyéndose en la mayor red de este tipo en el mundo.

Esta inversión refleja la explosiva demanda para transmitir mensajes computarizados producto del gigantesco crecimiento en el parque de computadores personales.

Si bien ésta será la mayor red instalada, AT & T hasta ahora estaba quedando atrás frente a sus pequeños competidores por no haber renovado sus líneas y equipos para adaptarlos a las nuevas necesidades de transmisión de información digitalizada (voz, data y video).

MCI Communications Corp. por su parte anunció que en 1988 ya habrán tendido 18.000 millas de fibras ópticas. Hasta ahora, sólo se encuentran en funcionamiento unas 600 millas, entre Washington y Nueva York.

Nuevo sistema VS-15 de Wang.

Sisteco anunció la incorporación de un nuevo equipo a su familia Wang VS. El VS 15, dirigido a la pequeña y mediana empresa viene con 256 Kb de memoria principal, expandible a un mega y puede manejar en disco fijo hasta 76 mega. Maneja en forma standard hasta seis estaciones de trabajo y esta capacidad puede opcionalmente ser ampliada a diez.

Por otro lado, en las próximas semanas se espera que Sisteco dé a conocer las características de un nuevo microcomputador.

Entre los nuevos clientes incorporados a esta empresa se cuentan la Cía. Minera Disputada de Las Condes, la Cía. de Seguros Allianz y el Chase Manhattan Bank.

Ecuador instala sistema de videotex francés.

En Ecuador fue anunciada la puesta en marcha de un servicio de videotex comercial en dos de sus principales centros urbanos, Quito y Guayaquil.

El sistema ecuatoriano fue instalado por el Banco Popular, uno de los mayores de ese país, el cual lo usará fundamentalmente para servicios bancarios y también para informaciones generales, juegos, horóscopos e incluso recetas de cocina.

El banco ecuatoriano se encuentra negociando en estos momentos con otros potenciales prestadores de servicios tales como periódicos, agencias de viajes, aerolíneas y tiendas de departamentos para expandir la gama de servicios a los abonados.

El costo para el usuario es de US\$ 30 mensuales, precio en el que se incluye la instalación del terminal.

El sistema instalado en Ecuador es de procedencia francesa, tanto el software de Teletel, y los terminales de Minitel. De acuerdo a fuentes de Intelmatique, la empresa estatal francesa encargada de marketing internacional, ya se han instalado 200 terminales de videotex y para fin de año se estima que estarán en funcionamiento otros 750.

El Banco Popular estima que el mercado potencial para sus servicios puede alcanzar a unos 4.000 en los próximos dos años.

Evaluación de proyectos

Guillermo Beuchat
Ing. Civil Industrial U. de Chile

Cuando se trata de invertir nuestro dinero, ganado con mucho esfuerzo, resulta muy importante analizar correctamente las diferentes alternativas que se nos presentan. El mundo moderno nos ofrece muchas opciones diferentes, algunas de las cuales parecen atractivas a simple vista pero no resisten un análisis más exhaustivo. Resulta especialmente importante realizar evaluaciones económicas correctas cuando el dinero a invertir no es propio, como es el caso de las empresas o instituciones financieras.

Existen muchas técnicas matemáticas para evaluar proyectos, que aportan diversos indicadores de la rentabilidad de una determinada inversión. Algunos de estos indicadores son absolutos, como el BNA o Beneficio Neto Actualizado, y otros son indicadores relativos a otros proyectos alternativos, como el TIR o Tasa Interna de Retorno.

El Programa BASIC adjunto es una herramienta de cálculo que permite obtener rápidamente los diversos indicadores financieros, que puede ser usado en cualquier tipo de proyecto e incluso para hacer un análisis de sensibilidad variando los parámetros de la evaluación. Antes de explicar el uso del programa, se exponen a continuación un resumen de algunos de los conceptos matemáticos involucrados y que es necesario conocer para plantear correctamente un problema de evaluación de proyectos.

En primer lugar, es necesario destacar que los indicadores que calcula el programa se usan generalmente en evaluación privada de proyectos, ya que la evaluación de proyectos de tipo social utiliza otras técnicas y parámetros.

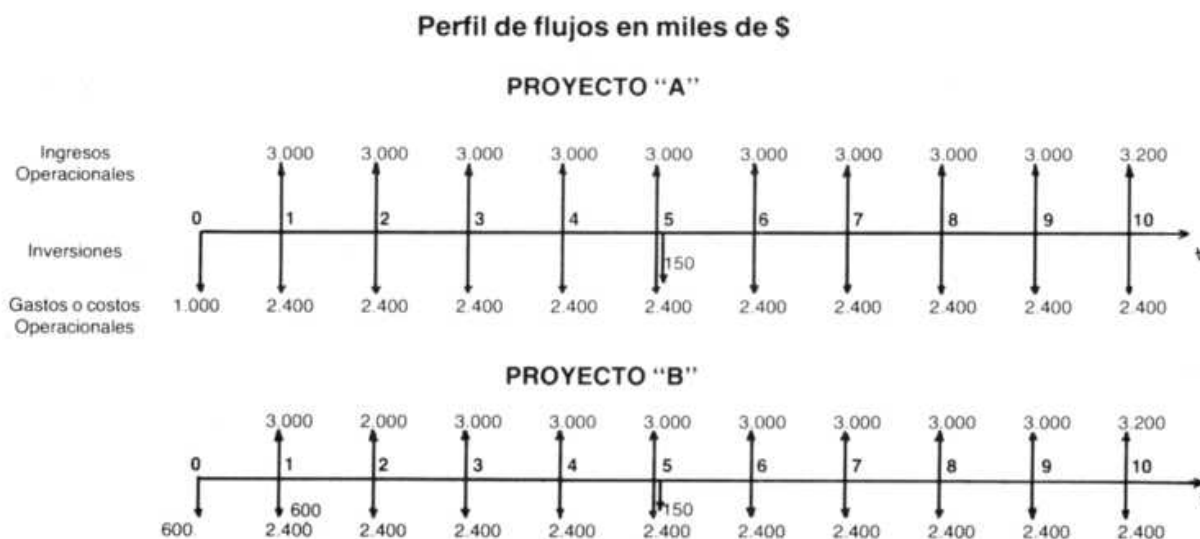
Un proyecto puede describirse matemáticamente como un conjunto de flujos monetarios, distribuidos a lo largo del tiempo en un perfil periódico. Generalmente, un proyecto se evalúa para un número finito de períodos, llamado también la "vida" del mismo. Por ejemplo, un proyecto de compra o

arriendo de maquinaria tendría un horizonte de tiempo de algunos meses a varios años. Existen esencialmente tres tipos de flujos de dinero: inversiones, ingresos y gastos o costos de operación. Es usual que las inversiones mayores se realicen en los primeros períodos de vida de un proyecto, y luego ocurren otras pequeñas inversiones de reposición de maquinarias, etc., a lo largo de todos los períodos. Por otra parte, muchas veces se empiezan a generar ingresos de explotación del proyecto varios períodos más adelante de la iniciación del mismo. Es el caso de los proyectos mineros y otros, en que el período de puesta en marcha dura varios años y las primeras ventas ocurren cuatro o cinco años después de realizadas las primeras inversiones. Es importante también considerar como ingresos, cuando corresponda, la venta de activos comprados durante la vida del proyecto, a un valor residual adecuado. Por ejemplo, si en el período cero se compra una máquina con una vida útil de diez años, y el proyecto tiene veinte años de duración, entonces en el período 10 será necesario considerar el ingreso adicional por venta de esa máquina, y el costo adicional de reemplazarla para los siguientes diez años.

Existen proyectos que no generan ingresos, en que se debe escoger aquella alternativa que signifique el menor costo. Por ejemplo, supongamos que una empresa constructora tiene dos alternativas para solucionar el problema de transporte de sus obreros a la zona de faenas: contratar un servicio de transporte, o comprar una flota de vehículos propios. Dadas todas las variables del problema, se deberá decidir por aquella alternativa que represente el costo mínimo para la empresa.

La figura 1 muestra los "perfiles" o flujos que ve un individuo que desea comprar un microbús, que tiene una vida útil de 10 años. Al final de ese período, es posible vender el vehículo a un 20% del valor original. El individuo utilizará el microbús

Figura 1



para viajes de turismo, que le significarán un ingreso promedio mensual de 250.000, con un costo promedio de 200.000. Además, el año 5 será necesario invertir 150.000 en un ajuste del motor. El individuo tiene 2 opciones para comprar el vehículo: el proyecto A implica pagarlo al contado, a un precio de 1 millón de pesos; el proyecto B implica pagarlo en 2 cuotas iguales de 600.000 pesos los años 0 y 1. Se supone que el inversionista dispone de fondos para comprar al contado si esto resultara conveniente, y la tasa de actualización relevante para él es del 12%. ¿Como podríamos determinar cuál alternativa es más conveniente para el inversionista?

Antes de resolver este problema, veamos qué significa cada uno de los indicadores que calcula el programa BASIC adjunto.

BNA, o Beneficio Neto Actualizado.

Este indicador, también llamado Valor Actualizado Neto (VAN), o Retorno Neto Actualizado (RNA), es uno de los más usados, y se define como:

$$BNA = \sum_{t=0}^n \frac{(Rt - It - Gt)}{(1 + i)^t}$$

en que Rt = retorno o ingreso en el período t .

It = inversión realizada en el período t .

Gt = gasto o costo de operación en el período t .

i = tasa de actualización relevante para el inversionista.

En otras palabras, el BNA puede definirse como la suma de todos los flujos netos (Ingreso-inversión-gasto) de un proyecto, actualizados al período 0, es decir, expresados en pesos recibidos hoy (valor actual del proyecto).

El parámetro i , correspondiente a la tasa de actualización relevante para el inversionista, merece una mayor explicación. Es importante notar que i no es una tasa de inflación, para expresar todos los valores en moneda de hoy. De hecho, para que una evaluación esté correcta, la inflación debe considerarse previamente al calcular cada uno de los flujos. La tasa i es una tasa que refleja adecuadamente el "costo de oportunidad" para el inversionista, de invertir su dinero en este proyecto.

Para aclarar más aún este concepto, pensemos en un inversionista que puede interactuar en el mercado de capitales, y que dispone de una cantidad de dinero para invertir. Si no realiza ninguno de los proyectos productivos que está evaluando, siempre podrá depositar sus fondos en su banco, y obtener por ellos una ganancia generada por la tasa de interés real de captación que paga el banco. En este caso, la tasa de actualización relevante para el empresario, al evaluar los otros proyectos productivos, será la tasa de interés del banco, puesto que para realizar cualquier proyecto, éste deberá tener una rentabilidad superior a esa tasa. Para el indicador que estamos estudiando, el BNA deberá ser mayor que cero a esa tasa i .

El criterio para seleccionar una o varias alternativas usando el BNA, consiste en jerarquizar los proyectos de mayor a menor valor de BNA a la misma tasa i , y realizarlos en ese orden mientras existan fondos para invertir. Los proyectos con $BNA < 0$ no deberán realizarse, puesto que generarán una rentabilidad menor que la tasa de actualización relevante.

TIR, o Tasa Interna de Retorno.

Este indicador, también usado extensamente, se define como aquella tasa de actualización i que hace que el BNA de un proyecto sea nulo, es decir, a aquella tasa a la cual es indiferente para el inversionista si realiza o no un proyecto. Se define entonces como i tal que:

$$\sum_{t=0}^n \frac{(Rt - It - Gt)}{(1 + i)^t} = 0$$

El TIR corresponde a la rentabilidad propia del proyecto, y es un indicador relativo, ya que no es posible jerarquizar una serie de proyectos usándolo como referencia. Por ejemplo, la figura 2 ilustra el caso de dos proyectos A y B. Como vemos, no es posible afirmar que un proyecto con TIR mayor que otro será siempre más rentable. En este caso, ocurre justamente lo contrario: para tasas i menores que i_q (i de quiebre), el proyecto R es más rentable; y para tasas i mayores que i_q , el proyecto B pasa a ser más rentable.

Relatividad por indicador "TIR"

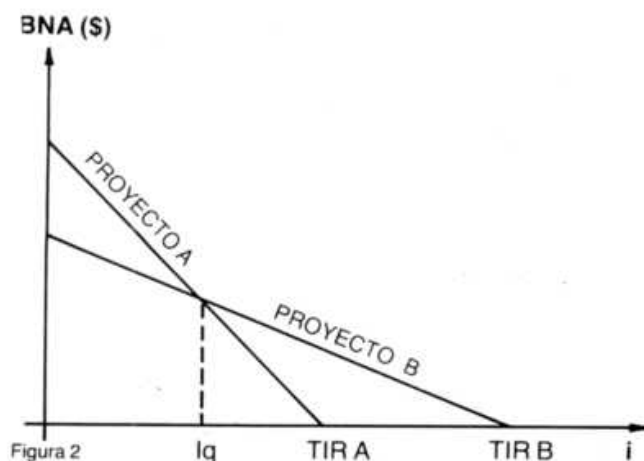


Figura 2

Otro problema de este indicador es que existe la posibilidad de que el TIR no sea único para un determinado perfil de flujos. En efecto, la ecuación para calcularlo puede tener más de una raíz o tasa i tal que $BNA(i) = 0$. El programa adjunto calcula el TIR de un perfil suponiendo que existe una sola raíz en el rango 0-200%. En general, es posible afirmar que si el proyecto tiene un perfil con inversiones iniciales bastante mayores que las de reposición efectuadas posteriormente, el perfil tendrá una sola raíz i tal que $BNA(i) = 0$.

El TIR se utiliza como una medida de comparación de tipo relativo, que se puede utilizar para comparar la rentabilidad de un proyecto con la tasa de interés relevante para el inversionista.

TREC, o Tiempo de Recuperación del Capital.

Este indicador es muy usado cuando existe una gran incertidumbre respecto del futuro. Como su nombre lo indica, corresponde al número de períodos necesarios para recuperar la inversión realizada, es decir, encontrar el x tal que:

$$\sum_{t=0}^x \frac{(R_t - G_t)}{(1+i)^t} > = \sum_{t=1}^n \frac{I_t}{(1+i)^t}$$

Este indicador puede tener una importancia clave cuando se trata de evaluar proyectos en condiciones inciertas en política económica o alto riesgo para el inversionista, ya que mientras antes se recupere el capital invertido, mejor.

ALFA y BETA

Estos dos indicadores permiten evaluar proyectos cuando existen restricciones de diversos tipos a los recursos disponibles. ALFA y BETA se utilizan cuando existe una restricción de capital para realizar las inversiones tanto en el período 0 como las inversiones de reposición. Se definen como:

$$\text{ALFA} = \frac{\text{BNA}}{I_0} \quad \text{y} \quad \text{BETA} = \text{ALFA} + 1$$

en que I_0 es la inversión total del proyecto actualizada al período 0, en la tasa i de actualización que sea relevante:

$$I_0 = \sum_{t=0}^n \frac{I_t}{(1+i)^t}$$

Estos dos indicadores permiten jerarquizar las alternativas en consideración, y se realizarán aquellas que tengan el mayor valor de BETA para aquellos con $\text{ALFA} > 0$. Si $\text{ALFA} < 0$, el proyecto no debe realizarse.

RBC y RBNC.

Estos indicadores corresponden a la Razón Beneficio-Costo, y a la Razón Beneficio Neto-Costo. Se define:

$$\text{RA} = \sum_{t=0}^n \frac{R_t}{(1+i)^t} \quad \text{y} \quad \text{CR} = \sum_{t=0}^n \frac{I_t + G_t}{(1+i)^t}$$

y tenemos que los indicadores son:

$$\text{RBC} = \frac{\text{RA}}{\text{CA}} \quad \text{y} \quad \text{RBNC} = \frac{\text{BNA}}{\text{CA}}$$

El indicador RBC se usa especialmente cuando existe alguna restricción de costos operacionales o se prevén problemas para cubrir esos costos. De todos los proyectos con $\text{RBC} > 1$, se elegirán los de mayor valor para el coeficiente.

Para concluir este resumen teórico, diremos que el uso correcto de estos indicadores dependerá de cada evaluador. El propósito de este artículo, sin embargo, es dar una visión general del problema, y proporcionar una herramienta computacional que permita calcular rápidamente los indicadores propuestos, además de poder realizar un análisis de sensibilidad con todas las variables o parámetros en juego. Para una exposición más completa de estos temas, conviene leer algún texto de evaluación económica de proyectos.

Uso de programa BASIC.

El programa presenta inicialmente un menú con las siguientes opciones:

1. Ingreso de datos
2. BNA
3. TIR
4. TREC
5. ALFA, BETA
6. RBC, RBNC
7. FIN

Haciendo uso de la opción 1 se ingresa el perfil de flujos del proyecto. Para cada período, incluyendo el período 0, se introducen las inversiones, los gastos o costos y los ingresos. Cuando alguno de ellos no exista, es necesario digitar explícitamente un 0.

Las demás opciones se usan para calcular los distintos indicadores, usando diferentes tasas de actualización. Como el programa vuelve al menú cada vez que finaliza un cálculo, es posible realizar muchos cálculos sobre el mismo conjunto de datos, variando las tasas de actualización para efectuar el análisis de sensibilidad.

Este programa fue desarrollado en un Commodore C-64, por lo que algunas líneas tienen instrucciones particulares. La línea 110 sirve para borrar la pantalla (CLS, HOME, etc.), y la línea 305 además usa la instrucción CLR (CLEAR, etc.) para borrar las variables definidas por el programa antes de efectuar el DIM en la línea 315*. Lo demás puede ser transcrito sin problemas a cualquier otro computador.

El algoritmo usado para calcular el TIR es del tipo "búsqueda binaria", y encuentra una raíz entre 0 y 2 para la tasa i . Es posible variar estos límites aunque será difícil encontrar proyectos con un TIR mayor que 200% o negativo. Por otra parte, la precisión del valor encontrado está dada por la comparación entre el valor de la iteración actual y la anterior. Si la diferencia en el valor del BNA es menor que 0.002, entonces se detiene el algoritmo y muestra el resultado con 3 cifras decimales correctas.

La figura 3 muestra los valores que se obtienen para cada una de las alternativas propuestas para el proyecto de comprar un microbús que habíamos enunciado anteriormente. Según estos resultados es más conveniente comprar el bús al contado, ya que el BNA es mayor, y los indicadores ALFA, BETA, RBC y RBNC son mejores para el proyecto A.

Resultado de Evaluación

| TASA = 12% | PROYECTOS | |
|------------|------------|------------|
| | A | B |
| BNA | 2369414.43 | 2233700.14 |
| TIR % | 58.64 | 60.25 |
| TREC | 3 | 3 |
| ALFA | 2.1835626 | 1.82966 |
| BETA | 3.1835626 | 2.82966 |
| RBC | 1.161783 | 1.151116 |
| RBNC | 0.161783 | 0.151116 |

Figura 3

Para concluir este artículo, diremos que la evaluación de proyectos es hoy una necesidad imperiosa. Las condiciones tan impredecibles y variables de la economía nacional, incluso a nivel de los individuos o pequeños inversionistas, hace necesario realizar siempre una evaluación adecuada. Ya no es posible invertir basado solamente en la intuición pues se corre grandes riesgos. Por otra parte, la disponibilidad de equipos computacionales es cada vez mayor, lo que permite realizar cálculos y evaluaciones que antes no se hacían sencillamente porque no se tenía el tiempo ni las facilidades para calcular manualmente estos indicadores. Como los lectores podrán apreciar, hoy en día es posible crear un programa similar al propuesto incluso en una pequeña calculadora programable de bolsillo.

```

10 REM *****
20 REM #
30 REM # EVALUACION DE PROYECTOS #
40 REM #
50 REM # GUILLEMO BEUCHAT S. #
60 REM #
70 REM *****
80 #
90 REM ***** MENU *****
100 #
110 PRINT CHR$(26) : REM BORRA PANTALLA
120 PRINT:PRINT
130 PRINT "1. INGRESA DATOS"
140 PRINT "2. BNA"
150 PRINT "3. TIR"
160 PRINT "4. TREC"
170 PRINT "5. ALFA , BETA"
180 PRINT "6. RBC , RBNC"
190 PRINT "7. FIN"
200 PRINT
210 INPUT "OPCION =":OP
220 IF OP=1 OR OP=7 THEN 110
230 IF OP=7 THEN END
240 ON OP GOTO 260,420,550,750,940,1100
250 #
260 REM ***** INGRESA DATOS *****
270 #
280 CLEAR : PRINT CHR$(126)
290 INPUT "CUANTOS PERIODOS SON =":P
300 DIM INV(P),EG(P),IG(P)
310 PRINT
320 PRINT "INGRESE DATOS PARA CADA PERIODO:"
330 PRINT:PRINT
340 FOR K=0 TO P
350 PRINT "INVERSION PERIODO "K":=" :INPUT INV(K)
360 PRINT "COSTOS PERIODO "K":=" :INPUT EG(K)
370 PRINT "INGRESOS PERIODO "K":=" :INPUT IG(K)
380 PRINT
390 NEXT K

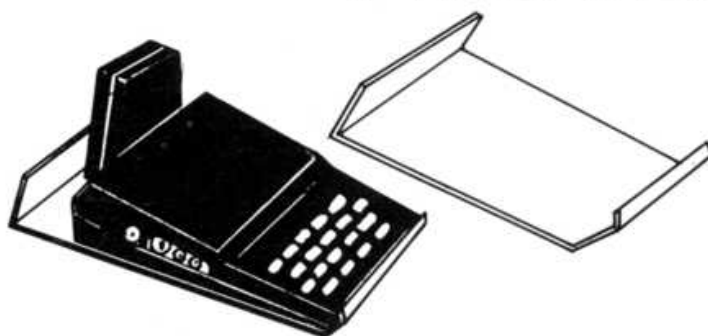
```

```

400 GOTO 110
410 #
420 REM ***** BNA *****
430 #
440 PRINT:PRINT
450 INPUT "TASA DE ACTUALIZACION ( % ) =":I
460 I=I/100
470 GOSUB 1260
480 BNA=S2-S1-S3
490 PRINT:PRINT
500 PRINT "BNA =" :BNA
510 PRINT
520 INPUT "DIGITE UNA TECLA PARA CONTINUAR ":A#
530 GOTO 110
540 #
550 REM ***** TIR *****
560 #
570 PRINT:PRINT
580 I=S1/S2-S1-S3
590 GOSUB 1260
600 S=S2-S1-S3
610 IF (S-T1)<.001 AND (S-T1)>-.001 THEN 700
620 T1=S
630 IF S=0 THEN 670
640 T1=I
650 I=(I+1)/2
660 GOTO 590
670 I=I
680 I=(I+1)/2
690 GOTO 590
700 PRINT "TIR =" :INT(I*10000)/100
710 PRINT
720 INPUT "DIGITE UNA TECLA PARA CONTINUAR ":A#
730 GOTO 110
740 #
750 REM ***** TREC *****
760 #
770 PRINT:PRINT
780 INPUT "TASA DE ACTUALIZACION ( % ) =":I
790 I=I/100
800 S=IG(0)-EG(0)
810 GOSUB 1260
820 S1=S
830 Y=0
840 IF S>S1 THEN 880
850 Y=Y+1 : IF Y/P THEN PRINT "TREC > VIDA DEL PROYECTO":GOTO 920
860 S=S-IG(Y)/(1+I)^K-EG(Y)/(1+I)^K
870 GOTO 840
880 PRINT
890 PRINT "TREC = " :Y : " PERIODOS"
900 PRINT
910 INPUT "DIGITE UNA TECLA PARA CONTINUAR ":A#
920 GOTO 110
930 #
940 REM ***** ALFA , BETA *****
950 #
960 PRINT:PRINT
970 INPUT "TASA DE ACTUALIZACION ( % ) =":I
980 PRINT
990 I=I/100
1000 GOSUB 1260
1010 BNA=S2-S1-S3
1020 ALFA=BNA/S1
1030 BETA=ALFA+1
1040 PRINT "ALFA =" :ALFA
1050 PRINT "BETA =" :BETA
1060 PRINT
1070 INPUT "DIGITE UNA TECLA PARA CONTINUAR ":A#
1080 GOTO 110
1090 #
1100 REM ***** RBC , RBNC *****
1110 #
1120 PRINT:PRINT
1130 INPUT "TASA DE ACTUALIZACION ( % ) =":I
1140 I=I/100
1150 PRINT
1160 GOSUB 1260
1170 BNA=S2-S1-S3
1180 RB=S2/(S1+S3)
1190 RN=BNA/(S1+S3)
1200 PRINT "RBC =" :RB
1210 PRINT "RBNC =" :RN
1220 PRINT
1230 INPUT "DIGITE UNA TECLA PARA CONTINUAR ":A#
1240 GOTO 110
1250 #
1260 REM ***** SUBROUTINA BNA *****
1270 #
1280 S1=S2-S1-S3=0
1290 FOR Y=0 TO P
1300 S1=S1-INV(Y)/(1+I)^K
1310 S2=S2-IG(Y)/(1+I)^K
1320 S3=S3-EG(Y)/(1+I)^K
1330 NEXT Y
1240 RETURN

```

PROTEKTOR™



No más pérdidas de Programas o riesgos de quemar su expansión de memoria de 16 K en el ZX81 y Timex-Sinclair 1000. Adaptable para expansión de memoria de 64K Mantenga su computador y memoria firmemente unidos y no tema mover el computador.

Adquiéralo por \$ 960 en Microbyte

Merced 346 Of. F. Pedidos a provincias agregar \$ 100 para gastos de franqueo.



DISTRIBUIDORES RESPALDADOS POR CIENTEC

| | | |
|--------------|--------------------|-------------|
| SANTIAGO | : ADCOM | F: 2237426 |
| SANTIAGO | : COMPUTER MARKET | F: 2243474 |
| SANTIAGO | : E. CHILENA COMP. | Moneda 673 |
| SANTIAGO | : ING. SER. ELECT. | F: 776991 |
| RANCAGUA | : ASCOMING LTDA. | F: 21869 |
| ANTOFAGASTA | : INFOCOM LTDA. | F: 225915 |
| LA SERENA | : E. CHILENA COMP. | F: 213222 |
| VIÑA DEL MAR | : VECOM LTDA. | F: 882490 |
| TALCA | : ABECAR LTDA. | F: 35837 |
| CONCEPCIÓN | : E. CHILENA COMP. | Caupol. 567 |
| TEMUCO | : STG LTDA. | Prat 837 |
| OSORNO | : STG LTDA. | F: 5354 |

MPF III

El Microcomputador compatible con APPLE^{MR}

| | |
|---|--------------------|
| Memoria RAM | : 64 KB |
| Memoria ROM | : 24 KB |
| Microprocesadores | : 6502 Z-80 A |
| Sistemas Operativos | : DOS CP/M |
| Lenguaje Standard | : Intérprete BASIC |
| Teclado profesional | |
| Tarjeta 80 columnas | |
| Disketeras de 140 KB | |
| Diskettes compatibles con APPLE ^{MR} | |
| ... y a un precio muy conveniente desde US\$ 1.176 + IVA. | |

Lenguajes, LOGO en español, Programas administrativos, educacionales y para diversas aplicaciones.



CIENTEC

INSTRUMENTOS CIENTIFICOS LTDA.

Departamento Computación

Antonio Varas 754 - Fonos: 225 7350 - 74 7028 - Santiago

Programando el Z80

(8.ª Parte)

Jorge Cea Silva

Grupo de Llamado (CALL) y Retorno (RETURN) de Subrutinas.

Al igual que en Basic u otros lenguajes de alto nivel, existe la posibilidad de trabajar con subrutinas, las que pueden ubicarse en cualquier lugar de memoria disponible. Las instrucciones utilizables son:

CALL nn: Equivale al GOSUB del Basic, con la diferencia que "nn" es una dirección de memoria que corresponde a la primera celda donde comienza la subrutina. Estos dos bytes se colocan después del código de operación con dichos bytes invertidos. Por ejemplo:

CALL 8570: CD 70 85

↑ Dirección con bytes invertidos
Código de operación

Antes de efectuarse el salto, la CPU almacena en el stack la dirección de retorno que corresponde al tercer byte después del código de operación y carga el Program Counter con la dirección a saltar.

Existen 8 versiones más de esta instrucción y que son saltos condicionales a los flags Carry, Cero, Paridad y Signo. Estas son:

| | |
|-------------|----------------------------------|
| CALL CY, nn | : Va a subrutina si flag CY = 1 |
| CALL NC, nn | : Va a subrutina si flag CY = 0 |
| CALL Z, nn | : Va a subrutina si flag Z = 1 |
| CALL NZ, nn | : Va a subrutina si flag Z = 0 |
| CALL PE, nn | : Va a subrutina si flag P/V = 1 |
| CALL PO, nn | : Va a subrutina si flag P/V = 0 |
| CALL M, nn | : Va a subrutina si flag S = 1 |
| CALL P, nn | : Va a subrutina si flag S = 0 |

Estas instrucciones equivalen a la línea Basic:
IF condición THEN GOSUB nn

RET:

Equivale a RETURN del Basic y al igual que este, es la última instrucción en la subrutina y permite volver al programa principal, colocando en el Program Counter los valores guardados en el stack por la instrucción CALL, ejecutando así la instrucción que sigue a esta. En el ZX-81 y otros equipos similares, esta instrucción permite retornar al Basic desde un programa en lenguaje de máquina.

Al igual que CALL, la instrucción RET tiene 8 versiones de retorno condicional de acuerdo a los flags Carry, Cero, Paridad y Signo. Además, hay dos instrucciones especiales que actúan con interrupciones:

| | |
|-----------|--|
| RET CY,nn | : Retorna de subrutina si flag CY = 1 |
| RET NC,nn | : Retorna de subrutina si flag CY = 0 |
| RET Z,nn | : Retorna de subrutina si flag Z = 1 |
| RET NZ,nn | : Retorna de subrutina si flag Z = 0 |
| RET PE,nn | : Retorna de subrutina si flag P/V = 1 |
| RET PO,nn | : Retorna de subrutina si flag P/V = 0 |
| RET M,nn | : Retorna de subrutina si flag S = 1 |
| RET P,nn | : Retorna de subrutina si flag S = 0 |

RET I: Esta instrucción permite retornar desde una subrutina de interrupción cuando esta ha sido del tipo enmascarado (INT) en cualquiera de sus modos. Su forma de operar es igual a una instrucción RET, es decir saca la dirección de retorno desde el stack.

RET N: Esta instrucción permite el retorno desde una rutina de interrupción cuando esta ha sido del tipo no-enmascarada (NMI). Además de retornar usando la dirección guardada en el stack, esta instrucción repone en IFF1 su estado previo a la interrupción, rescatándolo de IFF2 donde fue almacenado al ocurrir una interrupción NMI.

RST n (Restart) Cumple la misma función que CALL excepto que usa un solo byte. Esto se debe a que la dirección a la que salta está implícita en el código de operación y está ubicada en la Página Cero. Existen 8 posibles saltos Restart y cada uno posee 8 bytes para ejecutar la subrutina. La siguiente tabla indica estas posibilidades.

Nombre OP CODE Uso en ZX-81

| | | |
|--------|----|---|
| RST 0 | C7 | Inicialización del sistema |
| RST 8 | CF | Manejo de los reportes de error |
| RST 16 | D7 | Imprime carácter del acumulador |
| RST 24 | DF | Toma un carácter desde una línea Basic. |
| RST 32 | E7 | Toma el siguiente carácter desde una línea Basic. |
| RST 40 | EF | Salta a cálculo en punto flotante |
| RST 48 | F7 | Hace espacio en memoria. |
| RST 56 | FF | Interrupción para cada línea en pantalla. |

Grupo de Entrada y Salida (INPUT - OUTPUT)

Este grupo permite con dos instrucciones básicas recibir información de un dispositivo externo (teclado, cassette, conversores, diskettes, etc.), y guardarlo en un registro interno, como también enviar hacia afuera (pantalla, cinta, etc.), un byte cualquiera.

El Z-80 tiene un set muy completo de este gru-

po, el cual facilita mucho el manejo de la información. Las puertas van numeradas de 00h a FFh (255 puertas I/O).

Las instrucciones de entrada (IN) se pueden comparar a INKEY\$ o INPUT del Basic y la de salida (OUT) a un PRINT o un LPRINT en el sentido de que mueven información de un dispositivo a otro. (Teclado a memoria los primeros y memoria a pantalla o impresora los otros).

El siguiente es el detalle de estas instrucciones:

IN A,(n): La dirección "n" (un byte) del dispositivo de entrada/salida es puesto en las líneas de direccionamiento A0 a A7 del bus. En seguida el dispositivo coloca la información en el bus de data el cual es puesto en el acumulador.

OUT A,(n): La CPU coloca la dirección "n" en las líneas A0 a A7 del BUS, para luego sacar por el bus de data el contenido que lleva el acumulador.

IN r,(C): Esta instrucción puede llevar un dato desde la puerta cuya dirección está en el registro C a cualquiera de los 6 restantes registros principales (A, B, D, E, H, L). Antes de recibir el dato, la CPU coloca en A0 a A7 lo que hay en C y en A8 a A15 lo que hay en B.

OUT (C),r: En este caso la transferencia se hace desde cualquiera de los registros principales a la puerta cuya dirección está en el registro C.

En las siguientes instrucciones, el registro B cumple una doble función. Primero como contador de bytes transferidos (1 a 256 bytes) y otra la ya mencionada en las instrucciones IN r, (C) y OUT (C),r. Además, en las instrucciones INI, IND, OUTI y OUTD el flag Z detectara cuándo el registro B sea cero colocándose en uno.

INI: Operación $(HL) \leftarrow (C)$; $B \leftarrow B-1$; $HL \leftarrow HL + 1$

Esta instrucción es similar a la de transferencia de bloque de memoria, excepto que usa el registro par HL para señalar la memoria de destino de la información, mientras que el registro B es usado como un contador de bytes. El registro C guarda la dirección de la puerta a utilizar. Debido a que el registro B es de 8 bits de longitud, puede trasladarse hasta 256 bytes.

INIR: Opera igual que INI pero repite hasta que el registro B es cero.

IN D: Operación $(HL) \leftarrow (C)$; $B \leftarrow B-1$; $HL \leftarrow HL - 1$

Funciona en forma similar a INI con la diferencia que una vez trasladado el dato desde la puerta a la memoria, el puntero HL señalará la celda de memoria inferior.

INDR: Opera igual que IND pero repite hasta que el registro B sea cero.

OUTI: Operación $(C) \leftarrow (HL)$; $B \leftarrow B-1$; $HL \leftarrow HL + 1$

Instrucción de salida que envía el byte de la celda de memoria señalada por HL a la puerta indicada por el registro C. Los registros B y HL operan igual que en la instrucción INI.

OTIR: Opera igual que OUTI pero repite el proceso hasta que el registro B sea cero.

OUTD: Opera igual que OUTI con la diferencia que una vez trasladado el byte de la memoria a la puerta de entrada/salida, el puntero HL señalará la celda de memoria inferior como destino para el próximo byte a trasladar.

OTDR: Opera igual que OUTD pero repite el proceso hasta que el registro B sea cero.

El siguiente programa, ejemplo de las instrucciones mencionadas, renumera las líneas de un programa Basic, de 10 en 10. No cambia las direcciones de salto en los GOTO y GOSUBs, pero los marca para una fácil identificación y posterior modificación manual.

El programa ocupa 75 bytes y se debe llevar sobre la RAMTOP, por lo tanto antes de cargarlo hay que ejecutar los siguientes comandos para 1K.

POKE 16388,180

POKE 16389,67 (127 para 16K)

NEW

En la línea REM se deben reservar 88 caracteres, ya que al final hay una rutina de 13 bytes que traslada el programa sobre la RAMTOP. Si tiene 16K entonces debe cambiar el valor en la dirección 16595 (43) por (7F).

Con RAND USR 16589, ejecute el programa que traslada los bytes sobre la RAMTOP. Después de esto, un NEW dejará la zona de Basic libre.

Para ejecutar el programa de remuneración escriba PRINT USR 17332 (para 1K) o 32692 (para 16K).

En el número de marzo, Microbyte viene mejor que nunca. La Banca Electrónica. Comienza un nuevo curso: CP/M, qué es y cómo usarlo.

Construya su propio compilador para el procesador 6502.

Chilenos desarrollan original sistema de control de calidad.

Teoría de Colas.

Nuevos Equipos.

Y como siempre, programas para Atari, Sinclair, Commodore, Texas, etc.

A la venta en los kioscos a partir de la última semana de febrero.

| Dirección | Data | Assembler | | | |
|-----------|----------|----------------|-------|-------------|----------------|
| 16514 | 01 07 40 | LD BC, 407F | 16568 | 18 F3 | JR X6 |
| 16517 | 11 00 00 | LD DE, 0000 | 16570 | 23 | X4: INC HL |
| 16520 | 2A 00 40 | LD HL, 400C | 16571 | 08 | DEC BC |
| 16523 | ED 42 | X3: SBC HL, BC | 16572 | 3E 25 | X5: LD A, 25 |
| 16525 | E5 | PUSH HL | 16574 | BE | CP (HL) |
| 16526 | 44 | 'LD B, H | 16575 | 38 EC | JR X6 |
| 16527 | 4D | LD C, L | 16577 | 7E | X7: LD A, (HL) |
| 16528 | 21 7D 40 | LD HL, 407D | 16578 | CB FF | SET 7, A |
| 16531 | E5 | PUSH HL | 16580 | 77 | LD (HL), A |
| 16532 | 18 08 | JR X1 | 16581 | 23 | INC HL |
| 16534 | 3E 76 | LD A, 76 | 16582 | 3E 7E | LD A, 7E |
| 16536 | ED B1 | CPIR | 16584 | BE | CP (HL) |
| 16538 | 28 02 | JRZ X1 | 16585 | 20 F6 | JRNZ X7 |
| 16540 | 18 02 | JR X2 | 16587 | 18 E0 | JR X6 |
| 16542 | E5 | X1: PUSH HL | 16589 | 2A 82 40 | LD HL, 4082 |
| 16543 | 21 0A 00 | LD HL, 000A | 16592 | ED 5B B4 43 | LD DE, 43B4 |
| 16546 | 19 | ADD HL, DE | 16596 | 01 4E 00 | LD BC, 75 |
| 16547 | EB | EX DE, HL | 16599 | ED B0 | LDIR |
| 16548 | E1 | POP HL | 16601 | C9 | RET |
| 16549 | 72 | LD (HL), D | | | |
| 16550 | 23 | INC HL | | | |
| 16551 | 73 | LD (HL), E | | | |
| 16552 | 2B | DEC HL | | | |
| 16553 | 18 EB | JR X3 | | | |
| 16555 | E1 | X2: POP HL | | | |
| 16556 | C1 | POP BC | | | |
| 16557 | 3E EC | X6: LD A, EC | | | |
| 16559 | BE | CP (HL) | | | |
| 16560 | 28 08 | JRZ X4 | | | |
| 16562 | 3C | INC A | | | |
| 16563 | ED A1 | CPI | | | |
| 16565 | 28 05 | JRZ X5 | | | |
| 16567 | E0 | RET PO | | | |

Computadores Commodore

VIC 20, C64, SUPER PET

El mejor software para el mejor computador

Administración, finanzas.

Multiplan
Inventory
General Ledger
Account Payable
Account Receivable
Easy Finance
Easy Calc 64
Data Manager 2

Ajedrez:

Colossus Chess
Sargon III

Procesamiento de texto:

Wordpro 3 Plus
Easy Script
Easy Mail
Script 64

Lenguajes

Logo
Pilot
Assembler

Educativos:

EasyMath / EasyCount
Kinder Konzept
A little much/or less
Hodge Podge
Monkey Math
Sword of Sagoal
Easy quiz Easy Lesson
The Word Machine
The Name Machine

Juegos

Flight Simulator II
Audio Video Catálogo
River Chase
Zaxxon
Frogger
Ms Pacman
Jumpman Junior
The Factory
Galaxy
Miner 2049'er
Juice

Choplifter
Moon Buggy (Dr. Gamon)
Falcon Patrol
Bat Attack
Space - Pilot
Pogo Joe
Oil's Well
Buck Rogers
Dig Dug
Pooyan
Paka cuda
Puc Man
Camels
Hover Bover
Wall Street
Jammin
Saucer Attack!
Donkey Kong
Kong
Cosmic Tunnels
Moon Shuttle
Gangbusters

Cyclons
Geisterhaus
Slots
Axtex Challenge
Pole Position
M.U.L.E.
Galaxions
Defender
Perplexian Challenger
Maze Man
Skier
Le mans Man Knacker
Dragons Den
Pool

ELECTROQUIN

Estos y muchos otros programas para su Commodore 64 y además gran variedad de cartridges para el VIC 20
Garantía y Servicio Técnico. Despacho a Provincia.

Para servicio técnico, confíe su equipo a buenas manos.  **commodore**

Esto es lo que hace al Televideo TS-1605 tan atractivo:

Cómodo para su vista. La pantalla no reflectante de gran resolución lo hacen fácil de leer tablas, gráficos, caracteres y números.

Más memoria. 256 Kbytes de memoria son incluidos como standard.

Mayor confiabilidad. Su torre vertical de enfriamiento mantiene al computador a una temperatura apropiada sin la existencia de ventilador. Esto lo hace silencioso, confiable y de alta performance.

Más espacio de trabajo. La pantalla de video es un 20% mayor que las pantallas standard, por lo que Ud. no necesita esforzar su mirada para ver su trabajo.

Gráficos. Una amplia variedad de elaborados gráficos comerciales pueden ser desplegados en la pantalla TeleVideo.

Fácil de leer. La inclinación de la pantalla es ajustable.

Más aplicaciones. Compatible con la vasta gama de software desarrollado para el IBM-PC.

Mayor capacidad de almacenamiento. Como standard, sus dos drives de 370 Kbytes c/u aseguran una amplia capacidad.

Mayor configuración standard. Gráficos en colores video-compuestos y RGB. Adaptadores de comunicaciones e impresora. Capacidad de ampliación compatible. Cinco programas de uso general...

Compacto. El Televideo TS 1605 ocupa muy poco espacio sobre su escritorio.

Fácil de usar. Usted puede mover el teclado en cualquier lugar de su escritorio e incluso sobre sus rodillas.

Menos cansador. El exclusivo diseño de su teclado incorpora un amplio espacio para descansar sus manos mientras escribe.

**IBM PC COMPATIBLE...
COMPATIBLE.**

Y esto es lo que lo hace irresistible

**US\$ 3.785 – MAS IVA
(precio promocional)**

 **TeleVideo Systems, Inc.**

PLETT

Visite nuestro nuevo local de ventas en MacIver 380.

Biblioteca de Rutinas

En general, al programar distintos sistemas en Basic, es usual encontrarse con rutinas que son aptas para varios de éstos. Este es precisamente el caso de las rutinas y que esperamos sean útiles para nuestros lectores.

Monto escrito. Asper Sarrás L. Luis Velásquez C.

Util en emisión de cheques, facturas, liquidaciones de sueldos y en general en toda aplicación en que es necesario traducir números a palabras. Esta rutina que puede ser insertada con toda facilidad en el contexto de cualquier otro programa, nos entrega en palabras, el monto

de cualquier número entre 0 y 999.999.999, suficiente para prácticamente todas las aplicaciones.

La rutina es bastante corta en relación a otros ejemplos que hemos visto, principalmente por un hábil uso de las instrucciones READ-DATA-RESTORE y ade-

más porque en el análisis del problema se enfrentó bien la problemática de la diferente traducción de un número de acuerdo al contexto en que se encuentra (once y dieciséis, veinte y veintiuno, etc.).

```
10 INPUT X1#:M2$=""
20 X$=STR$(INT(X1#)):X$=RIGHT$(X$,LEN(X$)-1):IF LEN(X$)=9 THEN 30: ELSE FOR X5=9
-LEN(X$) TO 1 STEP -1 :X$="0"+X$:NEXT X5
30 IF X1#=0 THEN M2$="CERO ":GOTO 290
40 X=VAL(MID$(X$,1,3))
50 IF X=1 THEN M2$="UN MILLON ":GOTO 70
60 IF X<>0 THEN GOSUB 120:M2$=M2$+" MILLONES "
70 X=VAL(MID$(X$,4,3))
80 IF X=1 THEN M2$=M2$+"MIL ":GOTO 100
90 IF X<>0 THEN GOSUB 120 :M2$=M2$+" MIL "
100 X=VAL(MID$(X$,7,3))
110 IF X<>0 THEN GOSUB 120:GOTO 290 ELSE GOTO 290
120 N1=INT(X/100):N2=INT(X/10)-N1*10:N3=INT(X)-N1*100-N2*10
130 IF N3=0 AND N2=0 THEN IF N1=0 THEN RETURN
140 IF N1=1 AND N2=0 AND N3=0 THEN M2$=M2$+"CIEN ":RETURN
150 IF N1<>0 THEN IF N1>4 THEN X3=N1+20:GOSUB 270:M2$=M2$+P$:GOTO 180
160 IF N1<>0 THEN IF N1=1 THEN M2$=M2$+"CIENTO ":GOTO 190 ELSE X3=N1:GOSUB 270:P
$=LEFT$(P$,LEN(P$)):M2$=M2$+P$+"CIENTOS ":GOTO 180
170 GOTO 190
180 IF N2=0 AND N3=0 THEN RETURN
190 IF N2=0 THEN X3=N3:GOSUB 270:M2$=M2$+P$:RETURN
200 X3=N2*10+N3
210 IF X3=20 THEN M2$=M2$+"VEINTE ":RETURN
220 IF X3>20 THEN X3=N2+15:GOSUB 270:M2$=M2$+P$:GOTO 250
230 IF X3>15 THEN M2$=M2$+"DIECI":N2=0:GOTO 190
240 GOSUB 270:M2$=M2$+P$:RETURN
250 IF N3<>0 AND X3>17 THEN M2$=M2$+" Y ":N2=0:GOTO 190
260 IF N3<>0 THEN N2=0:GOTO 190 ELSE RETURN
270 FOR X4=1 TO X3:READ P$:NEXT X4:RESTORE:RETURN
280 DATA UN,DOS,TRES,CUATRO,CINCO,SEIS,SIETE,OCHO,NUEVE,DIEZ,ONCE,DOCE,TRECE,CAT
ORCE,QUINCE,DIECI,VEINTI,TREINTA,CUARENTA,CINCUENTA,
SESENTA,SETENTA,OCHENTA,NOVENTA,"QUINIENTOS ","SEISCIENTOS ","SIETECIENTOS ","OC
HOCIENTOS ","NOVECIENTOS "
290 PRINT M2$+" PESOS"
300 GOTO 10
```


Validador de Campos

Iván Rojas A.

Prácticamente todos los programas consisten en ingresar datos al computador para que éste los procese de determinada manera, para luego entregar los resultados requeridos. Lo que varía entre programa y programa, en lo que a entrada de datos se refiere es la cantidad de datos o campos del registro, su tipo y su largo. Sin embargo, prácticamente siempre los datos se validan de la misma manera, de acuerdo al tipo (alfabético, numérico o alfanumérico) y el largo de cada uno. Además, siempre es necesario crear una pantalla para la entrada de datos en la que el operador pueda ir ingresando sus datos y pueda manejarla de un modo interactivo, permitiéndole a éste recorrer la pantalla ingresando y modificando a su entera comodidad.

En lugar de tener que repetir para prácticamente todos los programas los mismos pasos, es mejor insertar esta rutina. Con ella, usted puede definir el

número, nombre y características de los campos que necesita y el computador hará todo el resto.

El programa puede ser dividido en dos partes. A la primera, entre las líneas 10 y 220, la llamaremos Crea-Campos y en ella es posible definir hasta 20 campos de largo 25 cada uno. Al correr esta parte, el programa crea un archivo con las características de los campos que usted definió.

En la segunda parte, de las líneas 230 a 680, la que llamaremos Maneja-Pantalla, primero se lee el archivo anteriormente creado y de acuerdo a las características de los campos antes definidos, el computador procede a validar los datos que se le van entregando.

En la pantalla aparece el nombre de cada campo y a continuación de cada uno, una línea segmentada que informa del largo máximo que pueden tener éstos. Para recorrer la pantalla

se utilizan las siguientes teclas:

= > Avanzar
< = Retroceder
> Campo siguiente
< Campo anterior

Para grabar los datos o pasar al siguiente registro, se oprime la tecla Return cuando uno se encuentra en el último campo. La validación se realiza cada vez que el operador ingresa un carácter, por lo que es muy fácil detectar errores porque el computador no permite que éstos ocurran.

Con esta rutina, lo único que hay que modificar, de acuerdo a las necesidades específicas, es qué hacer con los datos obtenidos. En este ejemplo, los datos quedan todos en una tabla C\$ dimensionada para contener hasta 500 caracteres y los datos quedan como en el siguiente ejemplo:

| | |
|------------------|---------------|
| NOMBRE | TELEFONO |
| C\$ JULIAN PEREZ | 15167944 Etc. |

```

10 REM VALIDADOR DE CAMPOS
20 HOME : HTAB 6: PRINT "**** VALIDADOR DE CAMPOS ****":H = 30
30 HTAB 11: PRINT "NUMERO DE CAMPOS : --"
40 VTAB 2: HTAB H: GET A$: PRINT A$: IF ASC (A$) > 47 AND ASC (A$) < 58 THEN
AA$ = AA$ + A$: IF AA$ = "00" OR AA$ > "20" THEN RUN
50 H = H + 1: IF H = 32 THEN HTAB 10: PRINT "NOMBRE CAMPO TIPO LARGO":B = VAL
(AA$): POKE 34,4: GOTO 70
60 GOTO 40
70 FOR I = 1 TO B: PRINT "CAMPO "; RIGHT$ ("0" + STR$ (I),2);" .....
--":AA$ = "":H = 10
80 VTAB I + 3: HTAB H: GET A$: PRINT A$: IF ( ASC (A$) > 47 AND ASC (A$) < 58)
OR ( ASC (A$) > 64 AND ASC (A$) < 91) OR ASC (A$) = 32 THEN AA$ = AA$ + A$:H
= H + 1
90 IF ASC (A$) = 8 THEN AA$ = "":H = 10: VTAB I + 3: HTAB 10: PRINT ".....
": GOTO 80
100 IF H = 20 THEN H = H + 5: GOTO 120
110 GOTO 80
120 VTAB I + 3: HTAB H: GET A$: PRINT A$: IF ASC (A$) = 65 OR ASC (A$) = 78 O
R ASC (A$) = 35 THEN AA$ = LEFT$ (AA$ + ".....",10) + A$:H = H + 6:S = 0:
GOTO 140
130 GOTO 120
140 VTAB I + 3: HTAB H: GET A$: PRINT A$: IF ASC (A$) > 47 AND ASC (A$) < 58
THEN C$(I) = C$(I) + A$:H = H + 1: IF C$(I) = "00" OR C$(I) > "25" THEN C$(I) =
"":H = 31: GOTO 140
150 IF H = 33 THEN C$(I) = AA$ + C$(I):BB = VAL (C$(I)):K = 0:S = 0: NEXT I: T
EXT : HOME : GOTO 170
160 GOTO 140
170 TEXT :D$ = CHR$ (13) + CHR$ (4): PRINT D$"OPEN VALIDADOR,L20": FOR I = 1
TO B:AA$ = C$(I): PRINT D$"WRITE VALIDADOR,R";I: PRINT AA$: NEXT I:II$ = STR$ (
I - 1): PRINT D$"WRITE VALIDADOR,R";0: PRINT II$: PRINT D$"CLOSE VALIDADOR"
180 REM ****
190 REM ****
200 REM **** FIN CREADOR ****
210 REM ****
220 REM ****
230 CLEAR : DIM C$(500),R(23):K = 1: HOME
240 D$ = CHR$ (4): PRINT D$"OPEN VALIDADOR,L20": PRINT D$"READ VALIDADOR,R";0:
INPUT II$: FOR I = 1 TO VAL (II$): PRINT D$"READ VALIDADOR,R";I: INPUT AA$

```

```

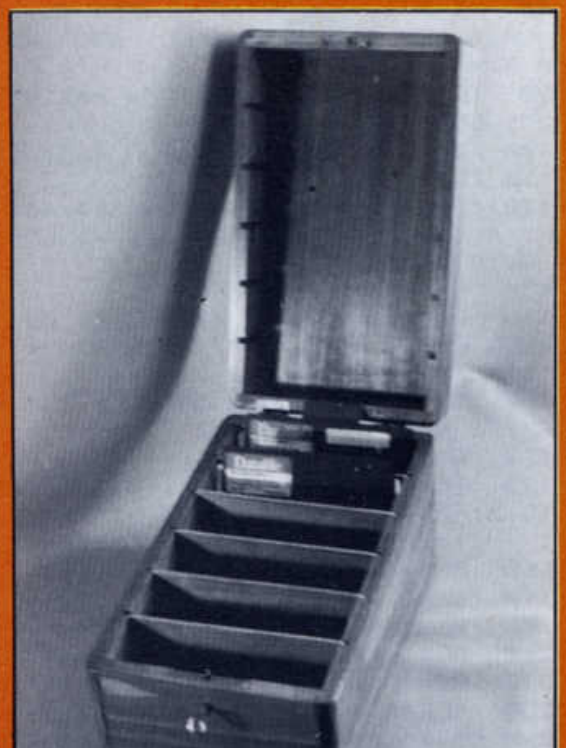
250 R(I) = VAL ( MID$ (AA$,12,2)) - 1:T = R(I) + T: IF MID$ (AA$,11,1) = "A" T
HEN L$ = L$ + STR$ (I): GOTO 280
260 IF MID$ (AA$,11,1) = "#" THEN L$ = L$ + "#": GOTO 280
270 L$ = L$ + " "
280 PRINT MID$ (AA$,1,10): FOR J = 1 TO VAL ( MID$ (AA$,12,2))
290 VTAB K: HTAB J + 11: PRINT "-": NEXT J:K = K + 1
300 NEXT I:T = T - R(I - 1)
310 PRINT D$"CLOSE VALIDADOR": GOSUB 670
320 R = 1:M = 1:N = R(R):K = 1
330 S = 1: FOR I = M TO M + N + 1
340 VTAB K: HTAB S + 11: GET A$:A = ASC (A$)
350 IF A = 13 AND R = VAL (II$) THEN END
360 IF I = M + N + 1 THEN 490
370 IF A = 32 AND MID$ (L$,K,1) < > " " THEN 460
380 IF A = 13 THEN 530
390 IF A = 60 THEN 560
400 IF A = 21 THEN 600
410 IF A = 8 THEN 650
420 IF MID$ (L$,K,1) = " " AND A > 47 AND A < 58 THEN 460
430 IF MID$ (L$,K,1) < > " " AND A > 64 AND A < 91 THEN 460
440 IF MID$ (L$,K,1) = "#" AND A < > 44 THEN 460
450 GOTO 470
460 PRINT A$:C$(I) = A$: GOTO 480
470 I = I - 1:S = S - 1
480 S = S + 1: NEXT I
490 IF A = 60 THEN 560
500 IF A = 13 THEN 530
510 IF A = 8 THEN 660
520 GOTO 470
530 IF A = 62 AND M > T THEN 470
540 M = 1: FOR J = 1 TO R:M = M + R(J) + 1: NEXT J
550 R = R + 1:N = R(R):K = K + 1: GOTO 330
560 IF A = 60 AND R = 1 THEN 470
570 K = K - 1:R = R - 1:N = R(R)
580 IF R = 1 THEN M = 1: GOTO 330
590 M = 1: FOR J = 1 TO R - 1:M = M + R(J) + 1: NEXT J: GOTO 330
600 FOR J = M TO M + N: IF C$(J) = " " THEN B = J - 1: GOTO 620
610 NEXT J: GOTO 480
620 IF I > B THEN GOTO 470
630 IF S < = B AND A < > 21 THEN C$(I) = A$: GOTO 480
640 PRINT C$(I): GOTO 480
650 IF S = 1 THEN 470
660 I = I - 2:S = S - 2: GOTO 480
670 REM
680 VTAB 23: HTAB 5: INVERSE : PRINT "<": NORMAL : PRINT "-->": INVERSE : PRI
NT "AVANZA CAMPO ANTERIOR": NORMAL : RETURN

```

LA INTEGRIDAD DE SU INFORMACION, es una inversión para su empresa.

Elegante gabinete de fina madera.
 Medidas: 29 x 17 x 16 cms. de alto.
 Terminaciones de lujo.
 Cierra con llave para mayor seguridad.
 Amplia capacidad: más de 90 diskettes en forma holgada.
 Seis divisiones ajustables.
 Placa metálica para identificación.
 Un producto indispensable junto a cualquier computador.

Adquiéralo por \$ 3.400 IVA incl. en MICROBYTE



DBASE II: Un administrador relacional de bases de datos

Héctor A. Miranda Riquelme

Hoy en día, la acción se está centrando en la administración o manipulación de bases de datos. Los sistemas de Administración de Bases de Datos (SABD) permiten al hombre de negocios utilizar rápidamente su computador personal y además hacerlo eficientemente, no sólo como un manejador de archivos sino como un administrador/informador de información. Dada la importancia del concepto de SABD, en esta oportunidad vamos a hablar de Dbase II, un SABD relacional de Ashton-Tate, Los Angeles, California.

Dbase II es una herramienta que permite una fácil manipulación de pequeñas y medianas (en tamaño) cantidades de información, ya sea interactivamente o con programas de comandos en Inglés.

Inicialmente, Dbase II se desarrolló para el sistema operativo CP/M, pero hoy en día con el auge del MS-DOS también existen versiones que corren bajo este sistema operativo. En este artículo me referiré principalmente a la versión CP/M.

Para comenzar, su computador debe cumplir con los siguientes requisitos mínimos:

- Un microprocesador Z80, 8080 u 8085.
- Un mínimo de 48 Kbytes de memoria (Dbase II usa las direcciones desde 5CH hasta A400H, pero trataremos al disco como una extensión del almacenamiento en memoria).
- Sistema operativo CP/M (versión 1.4 o 2.X).
- Uno, o más, dispositivos de almacenamiento masivo operando bajo CP/M (usualmente drives de disco floppy, pero los discos duros son ciertamente más satisfactorios).
- Un terminal CRT de 24 líneas por 80 columnas, con cursor direccionable, si se usan las operaciones de edición en toda la pantalla.
- Impresora opcional (para establecer algunos archivos de comandos).

Dbase II, como un sistema relacional, elimina los conjuntos, punteros de nexos, etc., los que pueden muy rápidamente tornar el trabajo de manejar el SABD en una pesadilla. Los datos se representan en la forma mostrada en la figura 1. Cada fila en la tabla se llama un registro, y cada columna en la tabla es referido como un campo. El orden en que los registros y filas se ingresan no interesa, pero cada elemento en la columna campo debe ser del mismo tipo, es decir, usted no puede mezclar PROVEEDORES con MONTOS.

| NUMERO ORDEN | PROVEEDOR | DESCRIPCION | MONTO | NUMERO |
|--------------|-------------------|--------------------|--------|---------|
| 2386 | Procesos Gráficos | Impresiones | 23.00 | BBQ-747 |
| 78622 | Grabados Pérez | Láminas Litografía | 397.42 | TFS-901 |
| M1883 | Carga Aérea Ltda. | Transporte | 97.00 | SPT-233 |
| | | | | |

Figura 1. Representación de datos bidimensional

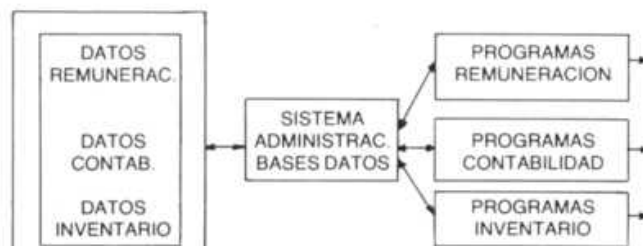
Ahora que ya tenemos alguna idea sobre la relación usada para almacenar ítems de datos, podemos revisar las especificaciones de este sistema para verificar si se acomodará a su aplicación. Aunque el sistema puede manejar una gran cantidad de información, su capacidad no es infinita y además estamos trabajando con un equipo microbasado. Las especificaciones principales incluyen: registros por archivo de Base de Datos: 65.535 máximo; campos por registro: 3 máximo; caracteres por registro: 1.000 máximo; caracteres por campo: 254 máximo; y precisión numérica: 10 dígitos.

Mirando el número de registros por archivo y el número máximo de caracteres por registro, podemos verificar que el sistema manejará hasta 65 Mbytes de información en un archivo, una cantidad bastante impresionante. Por supuesto, si usted está usando discos floppy, esto ocupará unos cuantos diskettes y algunas sofisticadas instrucciones para permitirle al programa acceder y manipular tantos datos. Pero en casi todas las situaciones de la vida real, no será necesario abusar tan severamente del sistema.

Antes de ahondar dentro de las características de operación, revisemos qué es la administración de bases de datos. Va más allá que el simple manejo de archivos y, de hecho, procesa, o administra, la información para los variados programas de manipulación de archivo. El SABD, por el hecho de tratar todos los datos en el mismo arreglo consistente, con compatibilidad archivo-a-archivo, permite a cada programa de operación trabajar con el banco de información completo. Esto elimina el forzar a los programas de remuneraciones solamente a acceder archivo de remuneraciones y a los programas de contabilidad sólo acceder archivos de contabilidad.



Manejo de Archivos



Bases de Datos

Figura 2. Arquitecturas de manejo de información.

Las dos diferentes arquitecturas de manejo de información se muestran en la figura 2. La principal ventaja es que toda la base de datos está disponible para cada programa operativo en lugar de tener que duplicar porciones del archivo de remuneraciones, si esta información es necesitada por los programas de contabilidad.

También, cuando se genera un nuevo sistema de procesamiento, aunque todavía es necesario escribir un nuevo acceso o programa de operación, no es necesario reestructurar o copiar ninguno de los datos. Están todos disponibles. Cuando las necesidades del negocio dictaminan agregar un tipo diferente de datos a un registro ya existente, no hay necesidad de cambiar ninguno de los programas operativos. Si estos programas no requieren los nuevos datos, ellos no lo verán. En un ambiente de manipulación de archivos, sería necesario reescribir los programas.

Dado que el principal uso de un SABD es manipular archivos de información, el primer paso es establecer los apropiados archivos de datos. En Dbase II, esto se hace con el comando CREATE. Cuando usted usa este comando, el sistema responde con la nota FILENAME, por lo que usted puede nombrar el archivo a ser creado. Las usuales limitaciones del CP/M se aplican aquí. El nombre debe comenzar con una letra, no puede tener más de ocho caracteres de largo, y no debe tener ":". Una vez que se ha nombrado el archivo, comienza el proceso de creación. Usted es requerido para describir su estructura de campos siguiendo la organización mostrada en la pantalla, simulada como sigue:

```
Create
ENTER FILENAME: Cualquier-nombre
ENTER RECORD STRUCTURE AS FOLLOWS.
FIELD NAME, TYPE, WIDTH, DECIMAL PLACES
001
```

Los nombres de campos pueden ser hasta de diez caracteres de largo, pueden ser en mayúsculas o minúsculas, pueden tener dos puntos ":", pero deben comenzar con una letra. Usted entonces describe el tipo de los datos, ya sea caracteres (C), numéricos (N), o lógicos (L). El ancho del campo puede ser hasta de 254 caracteres, pero, si es numérico, debe ser especificado el número de lugares decimales. Debe tenerse en cuenta el punto decimal, que ocupa un espacio de carácter.

La figura 3 muestra un ejemplo de una estructura de registro como la que se da en el manual de operación. En el primer intento de ingresar el campo 005, el nombre de este campo excedía la limitación de 10 caracteres y el programa indicó este error y permitió el reingreso. Habría sido ligeramente más fácil si sólo la estructura errónea necesitara ser corregida, pero esto está lejos de ser una dificultad. Cuando se finaliza con el proceso de establecer la estructura de registro, el operador presiona <RETURN> para terminar esta porción del comando de creación. El sistema entonces pregunta: INPUT DATA NOW? Dado que no es necesario entrar datos en este momento, abandonaremos esta función y miraremos las fun-

```
• USE Money Out
• DISPLAY STRUCTURE
STRUCTURE FOR FILE: MOEYOUT.DBF
NUMBER OF RECORDS: 00000
DATE OF LAST UPDATE: 00/00/00
PRIMARY USE DATABASE
```

| FLD | NAME | TYPE | WIDTH | DEC |
|-----|-------------|------|-------|-----|
| 001 | CLIENT | C | 004 | |
| 002 | JOBNUMBER | C | 003 | |
| 003 | BILL: DATE | C | 006 | |
| 004 | SUPPLIER | C | 020 | |
| 005 | DESCRIP | C | 010 | |
| 006 | HOURS | N | 006 | 002 |
| 007 | EMP: NMBR | C | 002 | |
| 008 | AMOUNT | N | 009 | 002 |
| 009 | BILL: NMBR | C | 006 | |
| 010 | CHECK: NMBR | C | 005 | |
| 011 | CHECK: DATE | C | 006 | |

Figura 3. Ejemplo de estructura de registro

Aunque el ingreso de datos a una estructura de archivo es obviamente el primer paso en un ambiente de SABD, la prueba real viene cuando es necesario cambiarla, manipularla, y exteriorizarla. En estas áreas, Dbase II realmente brilla. Usted puede trabajar interactivamente con su base de datos usando comandos conversacionales en inglés.

La lista de los comandos disponibles se muestra en la figura 4. Ha sido dividida en secciones bajo encabezamientos que relacionan la tarea particular que llevan a cabo. Por ejemplo, bajo el subconjunto creación de archivos hay comandos que permiten la creación, copia, modificación y varias otras órdenes de control. Algunas clases de comandos permiten varios métodos de desplegar o manipular los archivos. Una clase adicional de comandos asiste en el control y uso de los archivos de comandos, aquellos procedimientos que el usuario establece para manipular los archivos de datos.

CREACION DE ARCHIVOS

CREATE establece nuevas bases de datos estructuradas.

COPY copia bases de datos, sus estructuras, o sus datos a un archivo nuevo.

REPORT genera información según sus especificaciones, seleccionando sólo la información que usted desee, con o sin totales o subtotales.

SAVE copia las variables de memoria a un archivo para su uso posterior.

INDEX crea un archivo de índices para una localización más rápida de los datos.

AGREGACION DE DATOS

APPEND agrega registros al final de la base de dato.

CREATE permite la entrada de datos cuando el archivo ya está creado.

INSERT pone registros en un archivo.

EDICION DE DATOS

EDIT altera registros y campos específicos de la base de datos.

REPLACE cambia los contenidos de campos específicos.

CHANGE edita campos específicos en la base de datos.

DELETE marca registros para ser borrados.

RECALL borra las marcas de borrado.

DESPLIEGUE DE DATOS

DISPLAY y LIST muestran registros, campos y expresiones.

? muestra el valor de variables o expresiones.

REPORT formatea y despliega los datos de acuerdo a las especificaciones, con o sin totales.

READ muestra información y datos anotados.

SUM totaliza los campos que usted especifique en una base de datos.

TOTAL totaliza campos especificados a una nueva base de datos.

MANIPULACION DE ARCHIVOS

DO comienza la ejecución de un archivo de comandos secuencial.

APPEND agrega datos desde otros archivos (aunque no sean del dBASE II).

SELECT selecciona entre dos bases de datos en uso.

SORT organiza la base de datos ordenada por un campo.

COMANDOS VARIABLES DE MEMORIA

ACCEPT guarda caracteres de datos para uso posterior.

INPUT acepta datos en caracteres, numéricos y lógicos.

WAIT acepta el ingreso de un solo carácter.

GET trabaja como el comando INPUT, usado en el modo de ingreso interactivo a pantalla completa.

SUM guarda los totales a variables de memoria.

SAVE guarda las variables de memoria a un archivo en disco.

RESTORE recupera las variables de memoria almacenadas.

OTROS COMANDOS

FIND localiza un registro indexado, típicamente en menos de 2 segundos.

LOCATE encuentra registros que satisfagan condiciones específicas.

SKIP se mueve hacia adelante o hacia atrás en la base de datos.

DO WHILE permite operaciones repetitivas.


IF... ELSE permite hacer elecciones.

Figura 4. Comandos de dBASE II

El sistema permite operar a pantalla completa, con posicionamiento X-Y del cursor para determinar formatos propios para ya sea entrada o reporte de datos. Las funciones de las teclas de control del cursor son muy similares a las funciones usadas en otros populares programas con ligeras diferencias de significados para algunas de las teclas. Las funciones de las teclas de control del cursor se muestran en la figura 5.




Figura 5. Funciones de teclas de control de cursor


assin Ltda.
SOCIEDAD ASESORIAS
Y SERVICIOS DE
INFORMATICA LIMITADA

VICUÑA MACKENNA 1491
Fono: 5550930 Stgo.

MBC-550 Un drive 160 KB
US\$ 1.600
 + IVA



SANYO

Microcomputador con procesador 8088 (8/16 bits), 128 KB de memoria RAM expandible a 256 KB, diskettes de 5 1/4" con 160 ó 360 KB de capacidad, compatibles con IBM-PC, interfaz paralela Centronics, interfaz RS-232 optativa, capacidad de gráficos. Sistema operativo MS-DOS compatible con PC-DOS.

Precios no incluyen IVA, equivalente M/nac. Precio (US \$)

| | | |
|-----------|-------------------------------------|-------|
| MBC-550 | Microcomputador 1 Drive 160 KB | 1.600 |
| MBC-555 | Microcomputador 2 Drives 160 KB c/u | 1.800 |
| MBC-550-2 | Microcomputador 1 Drive 360 KB | 2.200 |
| MBC-555-2 | Microcomputador 2 Drives 360 KB c/u | 2.600 |
| FDD-1655 | Drive adicional 160 KB | 500 |
| FDD-3655 | Drive adicional 360 KB | 700 |
| CRT-30 | Monitor Monocromático | 285 |
| CRT-70 | Monitor en Color | 1.080 |
| MBC-232 | Interfaz RS-232 | 150 |
| MBC-64 | Memoria adicional 64 KB RAM | 195 |

Software incluido:

| | | | | |
|-------------|---|----------|---|-----------|
| SANYO-BASIC | — | MS-DOS | — | MAILMERGE |
| WORDSTAR | — | INFOSTAR | — | SPELLSTAR |

NEC-APC

Microcomputador con procesador 8086 (16/16 bits). Con 128 KB de memoria RAM expandible a 640 KB, diskettes de 8" con 1 MB c/u, interfaz serial RS-232, interfaz paralela Centronics y monitor de alta resolución (640 x 475) incluido. Sistema operativo MS-DOS.

Precio (US \$)

| | | |
|---------|-------------------------|-------|
| APC-H01 | Con un drive 1 MB | 4.314 |
| APC-H02 | Con dos drives 1 MB c/u | 5.265 |
| APC-H07 | Drive adicional | 1.024 |
| APC-H13 | Coprocador 8087 | 579 |
| APC-H26 | Disco fijo de 10 MB | 3.827 |

Software incluido:

MS-DOS — DBASE-II — WORDSTAR — MULTIPLAN

COSMO

Microcomputador con procesador 6502, compatible con APPLE-II. Incluye: 12 KB ROM con intérprete BASIC, monitor y autocarga, 48 KB de memoria RAM, conexión a cassette, teclado numérico separado y 10 teclas de función programables.

Precio (US \$)

| | |
|---------------------------------|-----|
| Microcomputador básico | 640 |
| Opciones: | |
| Monitor 12" color verde | 310 |
| Monitor 9" color ámbar | 220 |
| Drive diskette | 393 |
| Interfaz diskette | 80 |
| Interfaz impresora | 86 |
| Modulador RF para televisor | 26 |
| Tarjeta Z-80 | 86 |
| Tarjeta 80 columnas | 123 |
| Tarjeta 16 KB memoria adicional | 86 |

Software:

Puede utilizar todo el software disponible para computadores APPLE-II y programas para sistema operativo CP/M.

IMPRESORAS EPSON

Precio (US \$)

| | | |
|----------|--------------------------|-------|
| RX-80 | 10", 80 cols., 100 cps. | 540 |
| RX-80F/T | id. | 705 |
| MX-100 | 15", 136 cols., 100 cps. | 930 |
| FX-80 | 10", 80 cols., 160 cps. | 950 |
| FX-100 | 15", 136 cols., 160 cps. | 1.305 |

Aunque la mayoría de las funciones de los comandos son autodescriptivas, algunas no son fáciles de entender. INDEX se usa para generar un archivo de índices que permite un rápido ordenamiento y recuperación de datos. Utilizando la indexación, el operador establece un archivo usando solamente las llaves por las cuales desee buscar la información. (Las llaves son campos de la base de datos que describen al registro, es decir, en un archivo de personal, el nombre de los empleados y/o el código del empleado probablemente serían usados como las llaves en un archivo de índices).

La ventaja se produce dado que sólo las llaves son incluidas en el archivo de índices, en lugar que el contenido completo de la base de datos. Las llaves están organizadas con punteros al registro desde el cual pueden ser derivadas. El archivo de índices está organizado en una estructura llamada un B-tree, que permite una rutina de sort múltiple rápida y eficiente en memoria. Cuando se está usando un archivo de índices, las bases de datos de un tamaño típico pueden ser recorridas en aproximadamente 2 segundos, usando el comando FIND. También, una vez que usted ha establecido el archivo de índices, no es necesario agregar información desde ninguno de los nuevos registros agregados a la base de datos principal, el sistema lo hace automáticamente.

Además de permitir al operador crear un archivo de comandos para la manipulación de sus datos, el sistema provee un gran repertorio de operadores que pueden ser usados para generar nuevos resultados basados en los datos almacenados. La lista incluye las cuatro funciones aritméticas standard (adición, substracción, multiplicación y división), los tres operadores relacionales usuales (<, >, =), además del no igual (<>), y combinaciones de éstos. Los operadores lógicos y dos simples funciones de concatenación de strings, completan la lista de los operadores disponibles.

Al contrario de otros SABD micro-basados, Dbase II no limita la longitud de una operación de comandos a una sola línea. Para exceder este límite, se usa un punto y coma al final de la línea para unirla con otra, hasta el límite de 254 caracteres.

El sistema es sumamente fácil de echar a andar. Además existe la posibilidad de acomodarlo a las características del equipo utilizado. Para hacer esto, existe el utilitario INSTALL. Una vez que este comando ha sido ingresado, la pantalla desplegará un conjunto de mensajes que permiten al usuario ya sea seleccionar o no operaciones en pantalla completa y una caracterización específica de terminal.

El display de la pantalla completa se ilustra en la figura 6. Nótese que, ingresando una Z en SELECT TERMINAL TYPE, se le llevará a usted a través de otro conjunto de mensajes que permiten la instalación de un terminal con características propias. En seguida, el sistema quedará inicializado para su configuración particular. La rutina INSTALL no necesita ser corrida nuevamente, a no

ser que sea necesario cambiar el tipo de terminal o los "defaults" del sistema. Para correr el sistema para las pruebas iniciales, el usuario llama al programa tipeando dBASE.

A-> INSTALL

```
dBASE II INSTALLATION PROGRAM VER 1.9
ARE FULL SCREEN OPERATIONS WANTED (Y/N)? Y
SELECT TERMINAL TYPE
A - HAZELTINE 1500          B - SOROC 120, 140, TELEVIDEO
C - HEATH 89                D - PERKIN ELMER FOX 1100
E - ADM-3A                  F - ADM-31
G - VDP-80                  H - INTECOLOR
I - GNAT-SYSTEM 10          J - TRS-80- II PICKLES & TROUT
Z - USER SUPPLIED TERMINAL CHARACTERISTICS
```

A

ENTER A CHARACTER TO BE USED FOR INDICATING MACROS OR A RETURN FOR DEFAULT CHARACTER OF AMPERSANDS (&)

TYPE A RETURN IF THE ERROR CORRECTION DIALOGUE IS TO BE USED FOR ANY OTHER KEY IF NO DIALOGUE IS WANTED:

Figura 6. Ejemplo de display de install

La descripción fácil-de-seguir de arriba es característica del manual de 200 páginas proporcionado. Lo inusual es que se trata de dos manuales combinados. El primer manual aparentemente fue escrito para un usuario principiante con poca experiencia en este campo. Su estilo es conversacional y fácil de leer con interesantes consejos para mejorar sus resultados.

La segunda parte, escrita por el autor del programa Wayne Ratliff debe ser usada cuando esté generando sus programas de comandos, teniendo allí una guía de la sintaxis exacta. Sería difícil usar esta sección como una guía para personas que están recién comenzando con el sistema. Una versión anterior de Dbase fue realizada con sólo esta sección, ocasionando muchas llamadas de usuarios frustrados. La adición de la primera sección del manual solucionó este problema.

El sistema es compatible con archivos ASCII y probablemente puede leer sus archivos ya establecidos y agregar los datos a su base de datos.

DBASE II de Ashton-Tate puede no ser el SABD perfecto para los microcomputadores, pero lo será por un largo tiempo. Especialmente en aplicaciones de propuestas y costos de trabajo, Dbase prueba realmente lo que vale.

En el número de marzo, Microbyte viene mejor que nunca. La Banca Electrónica. Comienza un nuevo curso: CP/M, qué es y cómo usarlo.

Construya su propio compilador para el procesador 6502.

Chilenos desarrollan original sistema de control de calidad.

Teoría de Colas.

Nuevos Equipos.

Y como siempre, programas para Atari, Sinclair, Commodore, Texas, etc.

A la venta en los kioscos a partir de la última semana de febrero.

6502 Software

Instrucciones en mnemónico del ensamblador

6502

| | | | | | |
|------------|---------------------------|------------|---------------------|------------|------------------------|
| ADC | Add with carry | DEC | Decrement memory | ROL | Rotate left |
| AND | Logical AND | DEX | Decrement X | ROR | Rotate right |
| ASL | Arithmetic shift left | DEY | Decrement Y | RTI | Return from interrupt |
| BCC | Branch if carry clear | EOR | Exclusive OR | RTS | Return from subroutine |
| BCS | Branch if carry set | INC | Increment memory | SBC | Subtract with carry |
| BEQ | Branch if result = 0 | INX | Increment X | SEC | Set carry |
| BIT | Test bit | INY | Increment Y | SED | Set decimal mode |
| BMI | Branch if minus | JMP | Jump | SEI | Set interrupt disable |
| BNE | Branch if result \neq 0 | JSR | Jump to subroutine | STA | Store accumulator |
| BPL | Branch if plus | LDA | Load accumulator | STX | Store X |
| BRK | Break | LDX | Load X | STY | Store Y |
| BVC | Branch if overflow clear | LDY | Load Y | TAX | Transfer A to X |
| BVS | Branch if overflow set | LSR | Logical shift right | TAY | Transfer A to Y |
| CLC | Clear carry | NOP | No operation | TSX | Transfer SP to X |
| CLD | Clear decimal mode | ORA | Logical OR | TXA | Transfer X to A |
| CLI | Clear interrupt disable | PHA | Push A | TXS | Transfer X to SP |
| CLV | Clear overflow | PHP | Push P status | TYA | Transfer Y to A |
| CMP | Compare to accumulator | PLA | Pull A | | |
| CPX | Compare to X | PLP | Pull P Status | | |
| CPY | Compare to Y | | | | |

Ejemplos de sintaxis

Direccionamiento ejemplo

| | |
|----------------|-----------|
| Acumulador | ROL A |
| inmediato | LDA # dd |
| Página cero | LDA dd |
| Página o ind x | LDA dd, x |

| | |
|-----------|-----|
| Absoluto | LDA |
| Abs ind x | LDA |
| Abs ind y | LDA |
| Ind ind x | LDA |
| Ind ind y | LDA |
| Relativo | Bcc |
| Abs indir | JMP |

| | |
|---------|-----|
| dddd | LDA |
| dddd, x | LDA |
| dddd, y | LDA |
| (dd, x) | LDA |
| (dd), y | LDA |
| dd | Bcc |
| (dddd) | JMP |

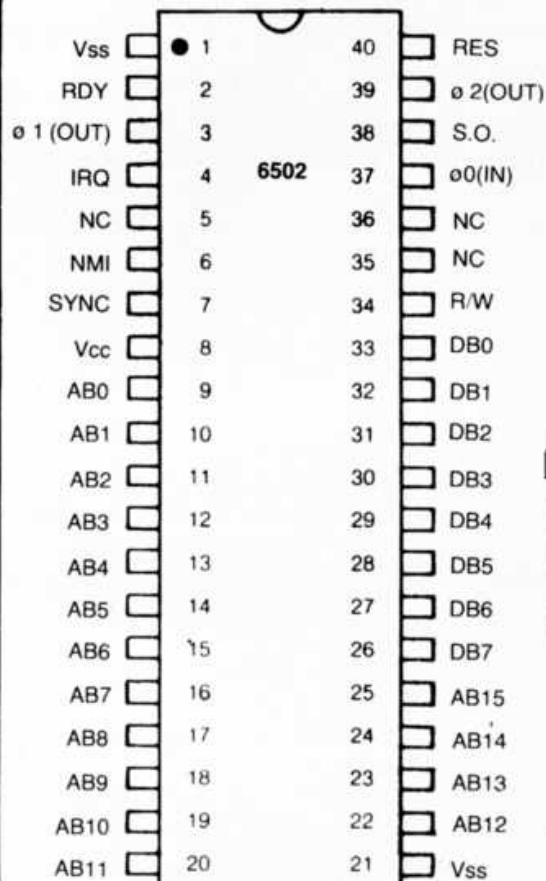
dd = N° de 8 bits
dddd = N° 16 bits

6502 Software

Dígito menos Significativo

| | 0 | 1 | 2 | 4 | 5 | 6 | 8 | 9 | A | C | D | E |
|---|------------|------------------|------------|--------------|--------------|--------------|-----|----------------|----------|----------------|----------------|----------------|
| 0 | BRK | ORA (nn, X) | | | ORA nn | ASL nn | PHP | ORA #nn | ASL A | | ORA nnnn | ASL nnnn |
| 1 | BPL | ORA (nn), Y | | | ORA nn, X | ASL nn, X | CLC | ORA nnnn, Y | | | ORA nnnn, X | ASL nnnn, X |
| 2 | JSR | AND - (nn, X) | | BIT nn | AND nn | ROL nnn | PLP | AND #nn | ROL A | BIT nnnn | AND nnnn | ROL nnnn |
| 3 | BMI | AND (nn), Y | | | AND nn, X | ROL nn, X | SEC | AND nnnn, Y | | | AND nnnn, X | ROL nnnn, X |
| 4 | RTI | EOR (nn, X) | | | EOR nn | LSR nn | PHA | EOR #nn | LSR A | JMP nnnn | EOR nnnn | LSR nnnn |
| 5 | BVC | EOR (nn), Y | | | EOR nn, X | LSR nn, X | CLI | EOR nnnn, Y | | | EOR nnnn, X | LSR nnnn, X |
| 6 | RTS | ADC (nn, X) | | | ADC nn | ROR nn | PLA | ADC #nn | ROR A | JMP (nnnn) | ADC nnnn | ROR nnnn |
| 7 | BVS | ADC (nn), Y | | | ADC nn, X | ROR nn, X | SEI | ADC nnnn, Y | | | ADC nnnn, X | ROR nnnn, X |
| 8 | | STA (nn, X) | | STY nn | STA nn | STX nn | DEY | | TXA | STY nnnn | STA nnnn | STX nnnn |
| 9 | BCC | STA (nn), Y | | STY nn, X | STA nn, X | STX nn, Y | TYA | STA nnnn, Y | TXS | | STA nnnn, X | |
| A | LDY #nn | LDA (nn, X) | LDX #nn | LDY nn | LDA nn | LDX nn | TAY | LDA #nn | TAX | LDY nnnn | LDA nnnn | LDX nnnn |
| B | BCS | LDA (nn), Y | | LDY nn, X | LDA nn, X | LDX nn, Y | CLV | LDA nnnn, Y | TSX | LDY nnnn, X | LDA nnnn, X | LDX nnnn, Y |
| C | CPY | CMP (nn, X) | | CPY nn | CMP nn | DEC nn | INY | CMP #nn | DEX | CPY nnnn | CMP nnnn | DEC nnnn |
| D | BNE | CMP (nn), Y | | | CMP nn, X | DEC nn, X | CLD | CMP nnnn, Y | | | CMP nnnn, X | DEC nnnn, X |
| E | CPX | SBC (nn, X) | | CPX nn | SBC nn | INC nn | INX | SBC #nn | NOP | CPX nnnn | SBC nnnn | INC nnnn |
| F | BEQ | SBC (nn), Y | | | SBC nn, X | INC nn, X | SED | SBC nnnn, Y | | | SBC nnnn, X | INC nnnn, X |

6502 Hardware



Regist

| | |
|---|-------------|
| | Accumulator |
| | Y Index Reg |
| | X Index Reg |
| | Counter |
| 1 | Stack PNTR |
| | Flags |

Mapa típico de memoria

| | |
|--------------|--------------|
| FFFF FFFE | Vector Reset |
| | ROM |
| | RAM |
| 01FF 0100 | STACK |
| 00FF 0000 | Datos |

| | | | | | | | |
|-----|---|---|---|-----|---|---|---|
| MSB | | | | LSB | | | |
| N | V | — | B | D | I | Z | C |

N — negativo
V — rebalse
B — instrucción BRK
D — modo decimal
I — deshabilita IRQ
Z — Cero
C — acarreo

6502 Software

| INST. | Acumulado | Inmediata | Página 0 | | Absoluto | | | Indirecto | | | Bandera |
|-------|-----------|-----------|----------|-------|----------|-------|-------|-----------|------|------|---------|
| | | | | Ind X | | Ind X | Ind Y | Indx | Indy | | |
| LDA | — | A9 | A5 | B5 | AD | BD | B9 | A1 | B1 | NZ | |
| STA | — | — | 85 | 95 | 8D | 9D | 99 | 81 | 91 | | |
| ADC | — | 69 | 65 | 75 | 6D | 7D | 79 | 61 | 71 | NZCV | |
| SBC | — | E9 | E5 | F5 | ED | FD | F9 | E1 | F1 | NZCV | |
| AND | — | 29 | 25 | 35 | 2D | 3D | 39 | 21 | 31 | NZ | |
| EOR | — | 49 | 45 | 55 | 4D | 5D | 59 | 41 | 51 | NZ | |
| ORA | — | 09 | 05 | 15 | 0D | 1D | 19 | 01 | 11 | NZ | |
| CMP | — | C9 | C5 | D5 | CD | DD | D9 | C1 | D1 | NZC | |
| ASL | 0A | — | 06 | 16 | 0E | 1E | — | — | — | NZC | |
| LSR | 4A | — | 46 | 56 | 4E | 5E | — | — | — | NZC | |
| ROL | 2A | — | 26 | 36 | 2E | 3E | — | — | — | NZC | |
| ROR | 6A | — | 66 | 76 | 6E | 7E | — | — | — | NZC | |
| LDX | — | A2 | A6 | B6† | AE | — | BE | — | — | NZ | |
| STX | — | — | 86 | 96† | 8E | — | — | — | — | | |
| CPX | — | E0 | E4 | — | EC | — | — | — | — | NZC | |
| DEX | CA* | — | — | — | — | — | — | — | — | NZ | |
| INX | E8* | — | — | — | — | — | — | — | — | NZ | |
| LDY | — | A0 | A4 | B4 | AC | BC | — | — | — | NZ | |
| STY | — | — | 84 | 94 | 8C | — | — | — | — | | |
| CPY | — | C0 | C4 | — | CC | — | — | — | — | NZC | |
| DEY | 88* | — | — | — | — | — | — | — | — | NZ | |
| INY | C8* | — | — | — | — | — | — | — | — | NZ | |
| DEC | — | — | C6 | D6 | CE | DE | — | — | — | NZ | |
| INC | — | — | E6 | F6 | EE | FE | — | — | — | NZ | |
| BIT | — | — | 24 | — | 2C | — | — | — | — | 7Z 6 | |

Comparaciones

| | | |
|------------|-----|-----|
| $A < M$ | BCC | yes |
| $A = M$ | BEQ | yes |
| $A > M$ | BCC | no |
| | BNE | yes |
| $A \geq M$ | BCS | yes |
| $A \neq M$ | BNE | yes |
| $A \leq M$ | BCC | yes |
| | BEQ | yes |

Transferencias y stack

| Ins. | Cod. | Bandera |
|------|------|----------|
| TAX | AA | NZ |
| TAY | A8 | NZ |
| TSX | BA | NZ |
| TXA | 8A | NZ |
| TXS | 9A | |
| TYA | 98 | NZ |
| PHA | 48 | |
| PHP | 08 | |
| PLA | 68 | NZ |
| PLP | 28 | Restored |

| Condicionales | | Saltos | | Clear/Set | |
|---------------|----|--------|----------------------|-----------|----|
| BCC | 90 | JMP | 4C Absolute | CLC | 18 |
| BCS | B0 | JMP | 6C Absolute Indirect | CLD | D8 |
| BEQ | F0 | JSR | 20 | CLI | 58 |
| BMI | 30 | RTS | 60 | CLV | B8 |
| BNE | D0 | RTI | 40 | SEC | 38 |
| BPL | 10 | BRK | 00 | SED | F8 |
| BVC | 50 | NOP | EA | SEI | 78 |
| BVS | 70 | | | | |

6 set V if bit 6
7 set N if bit 7

Gráficos

BIENVENIDOS AL BASIC (9ª Parte)

En el último número, nos despedimos con una tarea muy grata. En esa oportunidad, cuando hablábamos de la definición de funciones, dimos un ejemplo que además de ser útil, es extremadamente entretenido. En efecto, no hay nada más fascinante que observar en la pantalla una simulación de la vida misma.

En esa oportunidad, simulamos el comportamiento de dos especies animales distintas y cómo su interacción influía en el crecimiento de cada una de éstas. Si nuestros animales eran conejos y zorros, de acuerdo a la función que definimos, a medida que crece el número de conejos, aumenta el número de zorros por la simple razón de que hay más alimento. Sin embargo, a medida que son los zorros los que aumentan su población, entonces disminuye la población de conejos, porque como todos sabemos los zorros adoran comerse a los conejos. Al disminuir los conejos, no alcanza la comida para todos los zorros así que disminuye su número.

En realidad, esta simulación simplifica bastante a la realidad que es más compleja, pero para nuestros efectos, es suficientemente didáctica. Ahora, debemos aprender a graficar los resultados obtenidos. No en vano dicen que una imagen vale más de mil palabras.

En general, se puede hablar de la existencia de un Basic standard para prácticamente todas las marcas de microcomputadores, con algunas leves diferencias entre ellas. Sin embargo, distinta es la situación al referirnos a las instrucciones para graficar. El principio es el mismo en todo caso.

El computador divide su pantalla en un cierto número de puntos, también llamados píxeles. En algunos casos, los puntos son prácticamente del tamaño de una letra y caben por lo tanto en la pantalla un número bastante reducido de píxeles. En esos casos se habla de que el computador tiene una baja resolución. A mayor número de puntos, es mayor la resolución y por ende los dibujos aparecen mucho mejor definidos.

En algunos computadores, antes de poder graficar algo, debemos entrar en modo gráfico. Algunas instrucciones típicas para entrar en este modo son GR., HGR u otra que deberá revisar en el manual del equipo que esté usando. Algunos computadores pequeños tipo Sinclair, no tienen este tipo de instrucción. Si el computador maneja colores, podemos indicarle al computador el color con que queremos dibujar con instrucciones Color, Colour o HCOLOR seguido de un número.

Para dibujar, el computador no hace más que "encender" puntos en la pantalla. Graficar se trata por lo tanto de ir creando los algoritmos que le vayan indicando al computador cuáles puntos debe

encender. Para ubicar cada punto en la pantalla, el computador maneja un sistema de coordenadas, asignando un número a cada fila y a cada columna. En la figura 1, está representada una sección de una pantalla, con sus respectivos números de fila y columna,

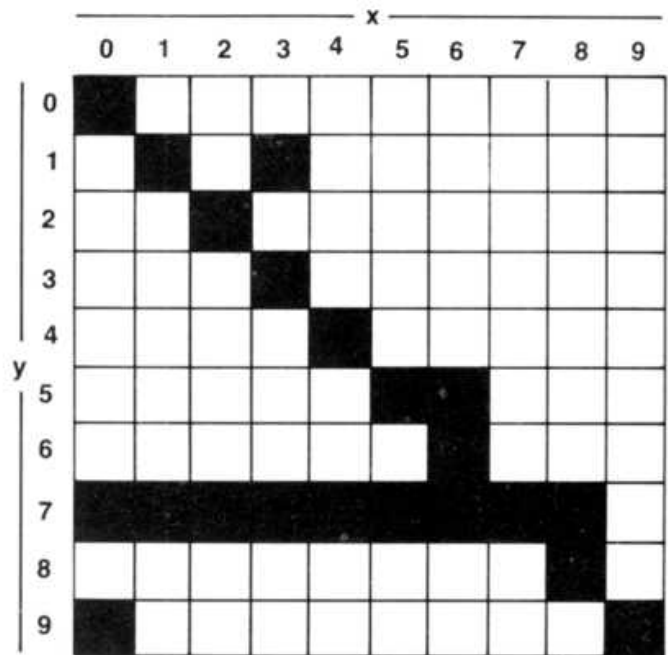


Figura 1

La instrucción que se utiliza para dibujar un punto, es PLOT X, Y. Esto significa dibujar un punto cuyas coordenadas son X e Y. En la figura 1, están dibujados los puntos (3,1), (6,5) y (9,9). Para dibujar todos los puntos en la línea 7 en la misma figura, se pueden utilizar dos métodos de acuerdo al equipo:

La primera es crear un ciclo FOR-NEXT como el siguiente:

```
FOR I = 0 TO 8  
PLOT I,7  
NEXT I
```

En otros equipos, basta dibujar el primer punto (origen) y luego indicar las coordenadas a las que queremos hacer llegar la recta. Para esto, las instrucciones son:

```
PLOT 0,7 TO 8,7  
y en otras versiones  
PLOT 0,7: DRAW TO 8,7
```

Lo mismo, para dibujar la diagonal de la figura 1, podemos escribir:

```
FOR I = 0 TO 9  
PLOT I,I  
NEXT I
```

o del otro modo, PLOT 0,0 TO 9,9
o por último, PLOT 0,0: DRAW TO 9,9

Por el momento, esto es todo lo que necesitamos para hacer nuestros primeros dibujos. Para ejercitarnos, volvamos al caso de las especies.

En el programa 1, dibujamos dos curvas que representan la población de cada especie en un período de tiempo determinado, en este caso representado por el ciclo en la línea 40.

```

10  REM  ESPECIES
20  DEF  FN A(X) = X + (2 * X - 3 * Y * X) * .01
30  DEF  FN B(Y) = Y + (2 * X * Y - 7 * Y) * .01
32  PRINT "INGRESE EL NUMERO DE MIEMBROS"
33  PRINT "PARA CADA ESPECIE"
35  INPUT X,Y
36  HGR : HCOLOR= 3
37  HPLLOT 200,150 TO 0,150 TO 0,0
40  FOR Z = 1 TO 1000
50  X =  FN A(X)
60  Y =  FN B(Y)
70  HPLLOT Z / 5,(15 - Y) * 10
75  HPLLOT Z / 5,(28 - X) * 5
76  PRINT X,Y
80  NEXT Z
90  PRINT "INGRESA OTROS VALORES "
100 INPUT X,Y: GOTO 40

```

Programa 1

En el programa 2, dibujamos tan sólo una curva en la que cada punto representa la población de ambas especies. La coordenada de las X representa a una y la coordenada de las Y a la otra.

Luego de correr los programas, modifiquen los coeficientes en las líneas 20 y 30 y obtendrán resultado sorprendente

```

10  REM  ESPECIES
20  DEF  FN A(X) = X + (2 * X - 3 * Y * X) * .01
30  DEF  FN B(Y) = Y + (2 * X * Y - 7 * Y) * .01
32  PRINT "INGRESE EL NUMERO DE MIEMBROS"
33  PRINT "PARA CADA ESPECIE"
35  INPUT X,Y
36  HGR : HCOLOR= 3
37  HPLLOT 200,150 TO 0,150 TO 0,0
40  FOR Z = 1 TO 200
50  X =  FN A(X)
60  Y =  FN B(Y)
70  HPLLOT 5 * X,150 - 10 * Y
76  PRINT X,Y
80  NEXT Z
90  PRINT "INGRESA OTROS VALORES "
100 INPUT X,Y: GOTO 40

```

Programa 2

Sonda celebra su primera década

ENTREVISTA

En tan sólo diez años de vida, Sonda Ltda. se ha convertido en la principal empresa de servicios de computación en Chile. Además, en solo seis años, ha logrado elevar a Digital Equipment al segundo lugar en el ranking de proveedores de equipos para el mercado nacional.

Para conocer más de cerca el balance que hacen de estos diez años y los futuros proyectos de esta empresa, entrevistamos en esta ocasión a su Gerente General, Andrés Navarro H.

Ingeniero Civil Industrial de la Universidad Católica, luego de dos años de docencia en esa misma casa de estudios, Andrés Navarro entra en sociedad con Copec para crear una empresa de servicios. Hoy, a los 35 años de edad, Navarro es padre de cinco hijos y está a la cabeza de una de las empresas más importantes en el área informática.

¿Cómo surgió Sonda?

En sus inicios, Sonda partió siendo una pequeña empresa de servicios. En 1974, al comenzar, trabajamos tan sólo 11 personas en ésta, aunque teníamos claro que las perspectivas de una empresa de esas características eran promisorias.

Nuestros esfuerzos apuntaban a dos grandes objetivos. Naturalmente, a que esta empresa fuese rentable y como segundo punto, considerábamos que en Chile no debía ocurrir lo que ha sido tan común en otros países en los que son empresas transnacionales las que dominan sin contrapeso el mercado informático, fijando en forma casi monopólica sus términos. Esto por un lado iría en desmedro de los propios usuarios de la informática y además significaría que el país mismo se vería privado de la posibilidad de ha-



cer suyas importantes herramientas tecnológicas.

La tecnología debe quedar en el país y no tan sólo eso sino que debemos ser capaces también de adaptarla a nuestras propias necesidades.

¿Qué ha significado para ustedes representar a Digital en Chile?

En primer lugar quisiera destacar que nosotros buscamos la representación de Digital fundamentalmente para satisfacer las propias necesidades de nuestros clientes. Al crecer éstos y con ellos sus necesidades, nos vimos obligados a buscar algún tipo de equipamiento cuando ya no les era suficiente con nuestros servicios tradicionales.

Elegimos en 1977 a Digital, una empresa que recién estaba surgiendo, luego de considerar lo avanzado de su tecnología.

Hoy, pienso que no nos equivocamos y somos afortunados de representar en Chile a una empresa que en pocos años se ha empujado hasta ubicarse en el segundo lugar a nivel mundial como proveedor de equipos.

Representar a Digital nos ha permitido ampliar nuestros horizontes en el área servicios. Pienso que es sobre todo nuestra capacidad para proporcionar un servicio integral lo que explica el acelerado crecimiento de Sonda. Nuestro personal alcanza a más de 350 personas, más de un 50% de éstos son profesionales y un 36% técnicos. De una facturación de U\$ 250.000 en el primer año, hemos pasado a facturar alrededor de U\$ 16 millones en 1984.

¿A qué se refiere con "servicio integral"?

Con integral, me refiero a que

estamos capacitados para ofrecer a un cliente el tipo de servicio que más se acomode a sus necesidades y estructura. Desde procesar su información en nuestros equipos, instalar terminales remotos en su propia oficina, venderle o arrendar equipos, proporcionarle el software e incluso instalarle un departamento completo de computación incluyendo el personal calificado necesario sin necesidad de que la empresa se preocupe de su funcionamiento en el más mínimo detalle.

Sonda cuenta con sucursales en Copiapó, Valparaíso y Concepción, además de centros técnicos en varias otras ciudades a lo largo del país. ¿No han pensado ampliar sus actividades al área de las telecomunicaciones o transmisión de datos?

Hemos jugado con la idea e incluso más allá de eso la hemos estudiado con detención pero a la postre lo hemos desechado. La razón es simple. Si bien contamos con la capacidad tecnológica y financiera como para abordar una empresa de esa envergadura, pensamos que tanto en Chile como en prácticamente todo Latinoamé-

rica, el campo de las comunicaciones ha estado tradicionalmente en manos del Estado. Pretender competir con el Estado cuando éste fija las normas y los términos podría convertirse en una aventura riesgosa para las empresas privadas, en la que estarían en juego inversiones de varios millones de dólares.

Pienso sin embargo que urge en Chile el establecimiento de una amplia red pública de transmisión de datos y ésta es precisamente una de las ideas que debería tomar la Asociación Gremial de Empresas de Informática a fin de apoyar a las empresas estatales para que la lleven a la práctica.

¿Cómo evalúa usted el estado actual de la informática en Chile, y qué proyectos tiene Sonda en ese contexto?

En los últimos años, considero que Chile ha alcanzado un nivel bastante alto, tanto en términos de capacidad de sus profesionales como en grado de introducción de las herramientas informáticas en las tareas administrativas. En este último campo, sin embargo, aún queda bastante por hacer principal-

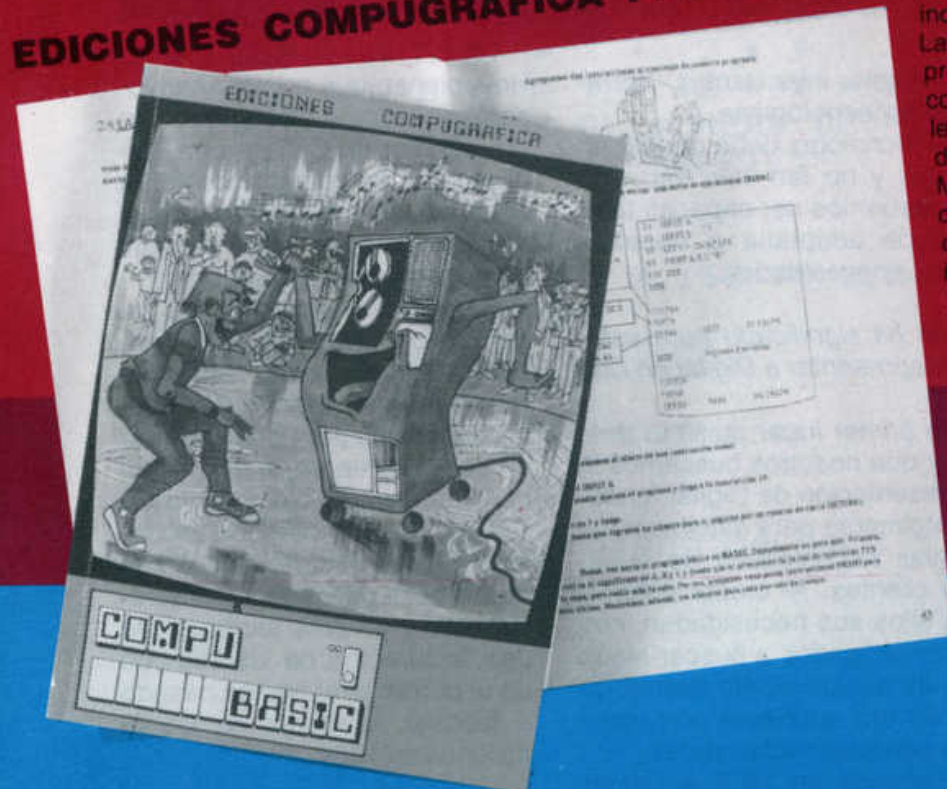
mente en la administración pública, salud y justicia.

En este contexto, Sonda tiene proyectado además de incrementar su participación en el área de servicios, exportar software e introducir con fuerza el uso de la computación en el área producción.

Exportar software de uso general e incluso a pedido porque en Chile estamos profesionalmente capacitados para hacerlo y tenemos la ventaja de tener menores costos que en los países más desarrollados. Ya hemos hecho los contactos necesarios y pienso que de aquí a un plazo prudente, podemos ganar un espacio interesante en este mercado a nivel mundial.

En relación al uso de la computación en los procesos productivos, pensamos que aún es muy poco lo que ha avanzado el país al respecto y es de urgente necesidad hacerlo. Por esto, a partir del primero de enero comenzará a funcionar en Sonda una división dedicada exclusivamente a esta área. Esta división estará encargada de estudiar y elaborar las herramientas computacionales que den respuesta a las necesidades concretas del país y de sus empresas.

EDICIONES COMPUGRAFICA PRESENTA



COMPU BASIC

Más de un 90% de los computadores traen incorporado el lenguaje Basic para programarlos. Las aplicaciones escritas en Basic van desde programas educativos y de juegos a los más complejos sistemas administrativos. Sin ser el mejor lenguaje, es sin duda el más necesario de conocer, dada su popularidad.

Manuales Basic ya conocíamos, pero ninguno tan completo como CompuBasic, el primero con más de 180 páginas de amena instrucción, con ilustraciones y numerosos ejemplos para poner en práctica de inmediato sus conocimientos. A solo \$ 980 CompuBasic es un manual que no debe faltar junto a todo computador.

DISTRIBUYE PARA TODO CHILE REVISTA MICROBYTE

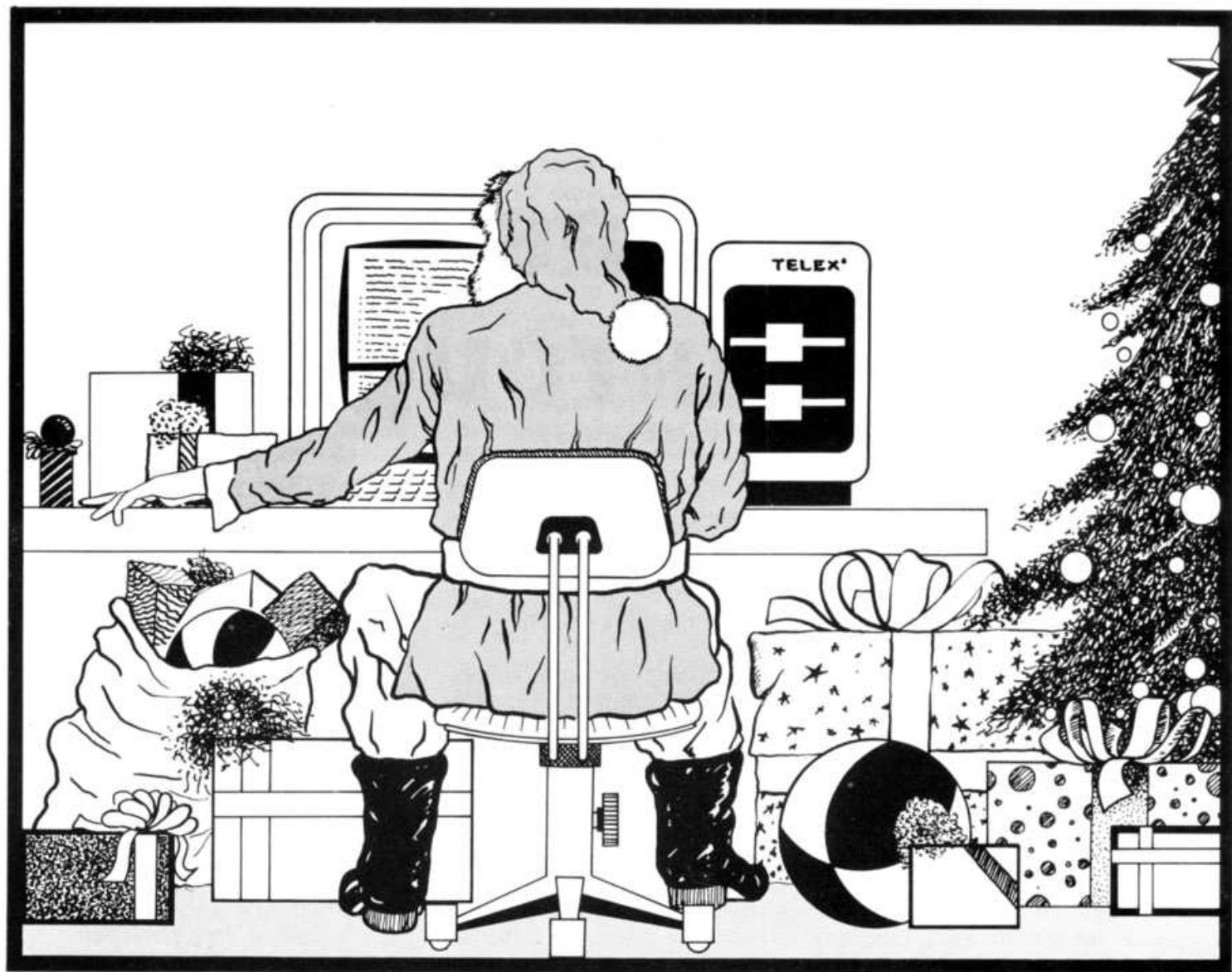
Solicite su ejemplar en Merced 346, Of. "F", Fono 393866, Santiago, y en las mejores tiendas del ramo.

DISTRIBUIDORES

AUTORIZADOS POR MICROBYTE:

Teorema Agustinas 1169 Stgo.
Sinclair Chile L. Thayer Ojeda 1234 Stgo.
Asicom Mac Iver 115 Stgo.
Latindata Nueva York 68 Stgo.
Infogroup Providencia 2623 Stgo.
Computer Market Pueblo del Inglés L. 66 Stgo.

Señores Microbyte, Merced 346, Of. "F"
Sírvanse enviar a mi dirección ...Ejemplar(es) de CompuBasic a \$ 980 c/u
Adjunto \$ 100 por ejemplar para gastos de franqueo por correo certificado.



GAD BARTOV S.P.

No sólo usted usa terminales **TELEX 370 COMPATIBLE**

COASIN es la única empresa en Chile que ofrece equipos compatibles con su sistema IBM, a través de la representación de las siguientes compañías norteamericanas:

TELEX COMPUTER, empresa dedicada al desarrollo de terminales compatibles con las series 370 y 4300 IBM.

DECISION DATA, especializada en la compatibilidad con sistemas 34, 36 y 38.

Usted ya puede disponer, en esta Navidad, de nuestros controladores remotos, pantallas e impresores.



... aporta soluciones!

El generador de sistemas Linc de Burroughs

Carlos Teixidó P.

El nivel de detalle en que debemos realizar cualquier descripción de un acontecimiento, es directamente proporcional al tiempo que es necesario invertir en ello y, de igual manera, a la cantidad de errores y omisiones que podamos cometer.

Esta realidad no es ajena al desarrollo de aplicaciones computacionales, cuya realización mediante metodologías y lenguajes tradicionales —como COBOL y RPG— exige de parte de los analistas y programadores, una especificación minuciosamente detallada de los pasos a seguir para un determinado proceso de información. De esta manera, resultan evidentes los riesgos en que se incurren si las etapas de diseño lógico o diseño físico, por ejemplo, arrojan como resultado especificaciones incompletas, poco claras o ambiguas.

Por otra parte, el diseño físico y construcción de un sistema, hace que analistas y programadores concentren sus esfuerzos en aspectos netamente técnicos (organización física de archivos, selección y uso de métodos de acceso a la información, estructuración de programas, etc.) lo que favorece una pérdida de compromiso con la esencia del problema aplicativo que se pretende resolver. Este entorno de trabajo demanda por último, que analistas y programadores sean sometidos a extensos ciclos de entrenamiento, para estar en condiciones de realizar su labor.

Estos son algunos de los factores que han llevado al desarrollo de aplicaciones computacionales a una situación que ha llegado a denominarse CRISIS

DEL SOFTWARE, y que señala la incapacidad de los departamentos de computación, para atender la creciente demanda por soluciones computacionales en sus empresas y organizaciones.

Burroughs, mediante su Generador de Sistemas LINC, ofrece una solución a esta crisis postulando una nueva metodología para el diseño y desarrollo de sistemas computacionales, un lenguaje formal de especificación de sistemas, y la generación automática de todos los programas requeridos para dar solución al problema computacional, a partir de una especificación del sistema.

La metodología LINC entrega 3 conceptos básicos con los cuales se podrá modelar un sistema:

Componentes: Corresponden a entidades lógicas constituidas por los datos básicos del problema a resolver. Para estas entidades lógicas LINC contempla que a partir de la definición del formato de pantalla, se proporcione automáticamente lógica de validación y cuatro operaciones básicas de mantención (agregar, cambiar, eliminar y consultar).

En un sistema de ventas, por ejemplo, los CLIENTES, PRODUCTOS y PROVEEDORES, serán componentes típicos.

Eventos: Son las transacciones cotidianas que alteran la información relacionada con los componentes, y desde la definición del o los formatos de pantalla, LINC infiere los datos y relaciones que participan en el evento. La VENTA de produc-

tos, el INGRESO de productos a bodega, el PAGO de una factura, son eventos propios del sistema de ventas.

Al definir los Componentes y Eventos de un sistema, se ha caracterizado por completo tanto la información que participa en él, como la interfaz que tendrá el usuario con el sistema.

Perfiles: Son perspectivas específicas sobre los datos del sistema, que al definirse como subconjuntos de componentes o eventos, ofrecen vías de acceso alternativas a la información, para satisfacer requerimientos de cálculo, validaciones lógicas, consultas, informes, etc. Por ejemplo, el control de stock de productos demanda una clara visión de los eventos de VENTA de productos, e INGRESO de productos a bodega; el control de cuentas corrientes, en cambio, requiere una perspectiva sobre los eventos de VENTA de productos a un cliente, y el PAGO de facturas que éste realice.

De esta manera, la especificación de un sistema corresponderá a la definición de los Componentes, Eventos y Perfiles que caracterizan al problema aplicativo, y a partir de esta especificación, el compilador LINC producirá la definición de la Base de Datos y todos los programas necesarios, que en un ambiente de desarrollo convencional, deberían diseñarse y construirse manualmente.

LINC permite entonces concentrar los esfuerzos en el análisis y especificación lógica del problema, realizada en un lenguaje simple y de muy alto nivel. Estas características favorecen

la incorporación activa del usuario final al proceso de desarrollo, aportando y participando en la solución computacional a su problema de información, lo que sumado a la capacidad de definir prototipos (modelos básicos del sistema final), garantiza una mayor coincidencia entre el problema aplicativo y el sistema computacional que lo resuelve.

La posterior mantención de los sistemas se simplifica en forma significativa, ya que los cambios que sufra la realidad del problema se incorporan a la especificación del sistema, independizándose de la tediosa tarea de modificar manualmente los programas y definición de base de datos, ya que esto será realizado por el compilador LINC. Resulta claro, entonces, que LINC es absolutamente autónomo y auto-suficiente en la generación y mantención de sistemas, puesto que bajo ninguna circunstancia el usuario deberá diseñar, construir o modificar programas o rutinas de los sistemas.

LINC genera sistemas orientados al proceso en línea y en tiempo real, incluyendo el ma-

nejo de bases de datos, consultas, transacciones e informes. Característica ineludible de un sistema generado por LINC es la total auditoría y recuperación ante interrupciones anormales, y la existencia de mecanismos de seguridad para restringir el acceso a usuarios no autorizados.

La gran rapidez y facilidad que se logra con LINC en el desarrollo y mantención de sistemas, redundando en un significativo aumento de la productividad del personal de procesamiento de datos, y en ahorros significativos a corto y largo plazo.

Se han realizado observaciones empíricas, comparando el aumento de productividad en instituciones que usan LINC, versus organizaciones con un ambiente de desarrollo tradicional. Los resultados señalan un aumento de productividad de 20 veces en el desarrollo de sistemas pequeños; comprobándose productividades mayores, según aumenta la complejidad y tamaño de los sistemas.

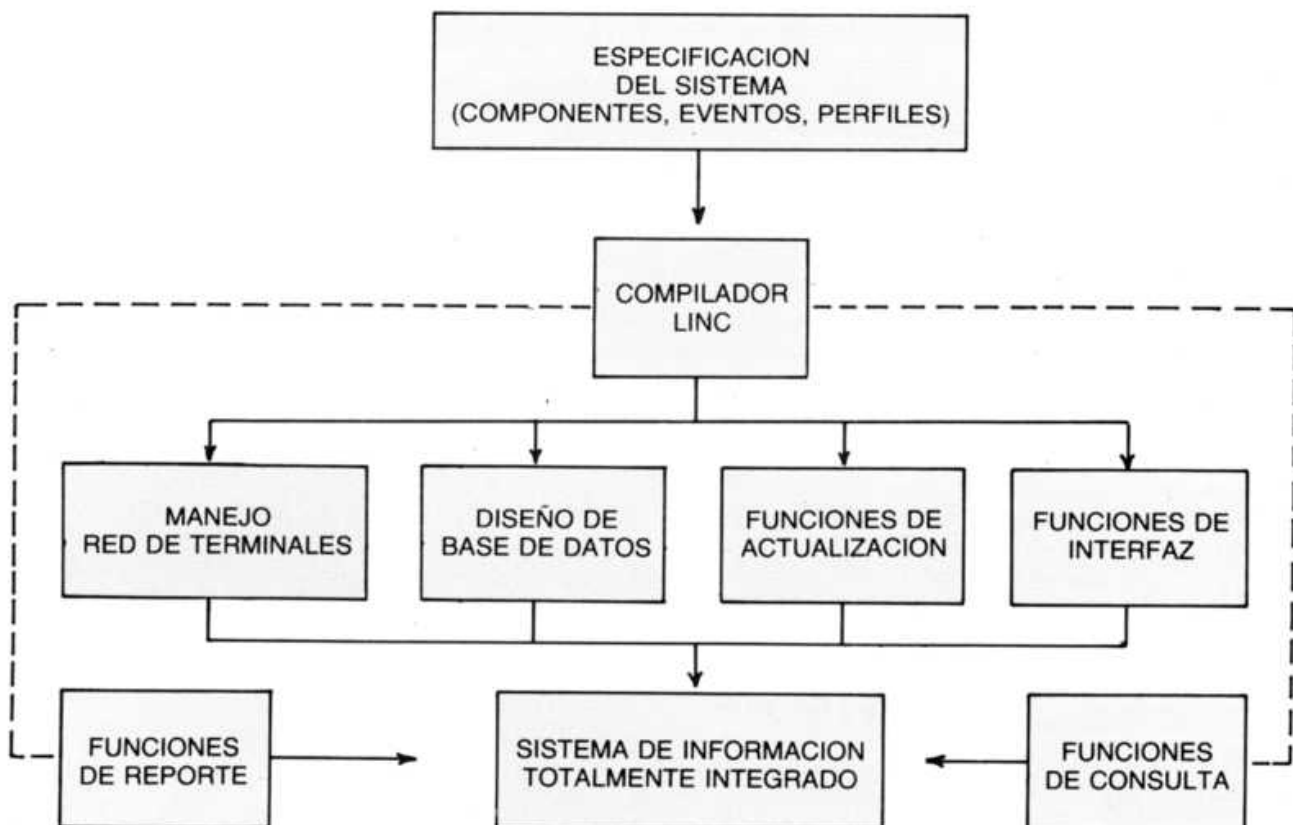
LINC se encuentra disponible en todos los computadores Burroughs, desde el equipo B1965

hasta el poderoso B7900. Una configuración básica de B1965 tiene 1 Megabyte de memoria principal, 130 Megabytes en disco, y está capacitado para manejar 32 terminales sin agregar componentes adicionales.

Contar con LINC en este amplio espectro de equipos, ofrece al usuario un importante nivel de protección sobre la inversión que realice en desarrollo de aplicaciones, dada la total compatibilidad de LINC en éstos equipos. Desaparece entonces, el trauma de la conversión.

La imperiosa necesidad de contar con herramientas de apoyo que permitan superar la denominada crisis del software, ha causado una notable aceptación de LINC a nivel mundial. En septiembre de este año la Corporación Burroughs informaba un total de 1000 usuarios LINC. En Chile, diversas empresas e instituciones, en las áreas de industria y comercio, gobierno, hospitales y servicios públicos, ya se cuentan entre los usuarios de este revolucionario producto.

LINC es marca registrada de Burroughs Corporation.



OPENFILE

Cartas del lector

JUVENTUD · DIVINO TESORO

Señores de Microbyte:

Primero debo felicitarlos por su excelente revista y preguntarles si hay un Club C-64 y pedirles hacer programas entretenidos para el C-64.

Saluda atentamente,

José Luis Arriagada
9 años

Habíamos recibido anteriormente cartas de niños, pero de nueve años es la primera.

Felicitaciones José Luis. Trataremos de entregarte el mayor número de programas e ideas para el C-64.

En cuanto al Club, éste funciona en Alameda 980, 5º piso.

PIDE INFORMACION

Muy señores míos:

Me llamo Alejandro Cifuentes Hurtado y estoy en 8º Básico.

Además de felicitarlos por su revista, les quería pedir dos favores. Yo tengo un computador Timex Sinclair 2068 y estoy interesado en la parte de su revista llamada "Programando el Z-80", pero las direcciones de memoria son diferentes; ya que para el 2068 no hay mucha información, les quería pedir que si pueden imprimieran el "CHR \$" de los códigos de caracteres y las direcciones de memoria. Esto también le serviría al computador ZX Spectrum, ya que es muy similar.

El otro favor es que me interesaría intercambiar software del 2068.

Les agradece y se despide de ustedes

Alejandro Cifuentes H.
Luis Pasteur 5248 - Las Condes

No hemos tenido acceso a un buen manual de estos equipos. Apenas podamos, publicaremos la información que necesitas.

ACLARACIÓN SOBRE EL SHARP PC-1500

Señores Microbyte:

Ante todo deseo felicitarlos por su excelente publicación.

En Microbyte N° 7, el señor Carlos Muñoz, en la sección Openfile, manifestó sus dudas acerca del SHARP PC-1500/A, por lo que me gustaría aclararle lo siguiente:

a) Sobre la diferencia entre el modelo PC-1500 y el modelo PC-1500/A, es sólo que el primero viene con 2 Kb de memoria standard, mientras que el segundo viene con 8 Kb de memoria.

b) En los modelos que se mencionan, existen los siguientes comandos para el lenguaje de máquina: PEEK, PEEK#, POKE, POKE# y CALL.

Con PEEK y POKE podemos entrar en la memoria del computador directamente, permitiéndonos operar con mayor rapidez en problemas de graficado y almacenamiento de datos. Los comandos PEEK y POKE fueron definidos y explicados en las revistas anteriores.

Existen dos memorias en estos modelos, siendo la capacidad de cada una de estas 65.536 bytes. La memoria primaria contiene los programas, ROM; el espacio para la expansión de memoria; y también el espacio para el cassette/printer interface. A la memoria primaria se llega utilizando los comandos PEEK y POKE (no tienen abreviación).

La memoria secundaria, solamente puede ser usada para el almacenamiento de datos, RAM. Para su uso se utilizan los comandos PEEK# (P. o PE.) y POKE# (PO.).

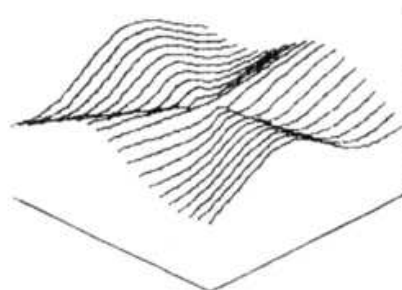
CALL, cuando es usada correctamente, envía la ejecución del programa a una subrutina específica del lenguaje de máquina, puede abreviarse como CA.

Me despido esperando haber solucionado gran parte de sus dudas.

Saluda atentamente,

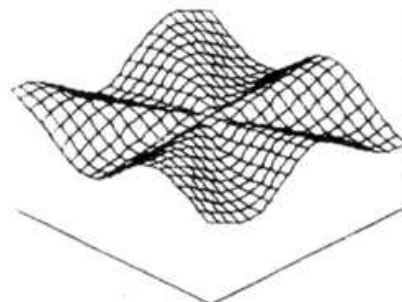
Christian Köhler R.
José T. Medina 094
Antofagasta

El lector nos envía además dos ejemplos de gráficos tridimensionales.



$$F(R, T) = R * \cos(4 * T + 45)$$

Radius = 1
ZL = -4.5
ZH = 4.5
Step X = 5
F(error) = 0



$$F(R, T) = R * \sin(5 * T + 60)$$

Radius = 1
ZL = -4.5
ZH = 4.5
Step X = 5
F(error) = 0

SUGERENCIA + CONSULTA + PETICION

De mi consideración:

Primero que nada quiero felicitarlos por este espacio que nos permite una comunicación con ustedes y los demás lectores de esta estupenda e instructiva revista.

Ahora me gustaría también, hacerles una pequeña sugerencia, una consulta y además, pedirles un gran favor.

—Sugerencia: A mi modo de pensar y creo representar el pensamiento de la mayoría de los lectores, por no decir de todos, que esta revista debería salir a lo menos dos veces al mes, puesto que el sufrimiento de la espera de la salida de Microbyte es demasiado largo.

—Consulta: Resulta que tengo un programa grabado en una cassette que por error fue borrado en su comienzo. ¿Habrá alguna forma de cargar lo que resta de programa en el Timex?

Peticion: ¿Podría transcribir al lenguaje de máquina del Timex 1000 este programa?

For i = 1 to 10

Print i

if i = 5 Then print "cinco"

Next i

De antemano, muchísimas gracias.

Francisco Lira
de la Parra.
P. Lynch 955
Stgo.

Respondiendo a su sugerencia: Ud. puede imaginar cuánto deseáramos conectarnos con nuestros lectores 2 veces al mes, pero aún, por el momento, no podemos realizar este sueño. Tal vez más adelante.

Consulta: Desgraciadamente, a nuestro conocer, Timex-Sinclair "empaqueta" los archivos a grabar de modo cerrado, no aceptando el conjunto si una parte se pierde. Sin embargo, es posible recuperar, al menos en teoría, lo que queda, aunque se obtendrá en forma de "dato" y no como un programa, pero puede ser útil para recuperar la información. Este camino re-

OPENFILE

Cartas del lector

quiere extensa experimentación.

Peticion: Sugerimos el siguiente programa:

| | | | | | |
|------|--------|------|--------|------|----------|
| 4082 | | 0001 | | ORG | 16514 |
| 4082 | 3E01 | 0002 | PART: | LD | A, 1 |
| 4084 | D7 | 0003 | REPI: | RST | 16 |
| 4085 | FE05 | 0004 | | CP | 5 |
| 4087 | CA9340 | 0005 | | JP | Z, CINCO |
| 408A | 3C | 0006 | | INC | A |
| 408B | FE0B | 0007 | | CP | 11 |
| 408D | C28440 | 0008 | | JP | NZ, REPI |
| 4090 | C3A640 | 0009 | | JP | FIN |
| 4093 | F5 | 0010 | CINCO: | PUSH | AF |
| 4094 | 3E43 | 0011 | | LD | A, C |
| 4096 | D7 | 0012 | | RST | 16 |
| 4097 | 3E49 | 0013 | | LD | A, T |
| 4099 | D7 | 0014 | | RST | 16 |
| 409A | 3E4E | 0015 | | LD | A, N |
| 409C | D7 | 0016 | | RST | 16 |
| 409D | 3E43 | 0017 | | LD | A, C |
| 409F | D7 | 0018 | | RST | 16 |
| 40A0 | 3E4F | 0019 | | LD | A, O |
| 40A2 | D7 | 0020 | | RST | 16 |
| 40A3 | C38440 | 0021 | | JP | REPI |
| 40A6 | C9 | 0022 | FIN: | JP | |

MORSE

Sres.
Microbyte
Pte.

Ref.: Open file

Les deseo manifestar que su revista llega a todos los interesados en computación, independiente del nivel de conocimiento sobre la materia.

Ante esta verdad, cabe consultar cómo se digita el programa de Microbyte N.º 6, página 40, tabla 1.

Su indicación, ayuda para novicios, constituirá un estímulo para todos.

Cordialmente,

Iván Melo Z.
Independencia 389
Stgo.

En el programa a que hace referencia, en la línea 4, hay que ingresar el contenido de la tabla 1 que representa a los distintos caracteres (números y letras) pero escritas en Morse, usando puntos, guiones y espacios. A cada carácter corresponde una combinación de 5 puntos, guiones o espacios. Para copiar esto en su programa digite 4 LET B\$ = " (etc)"

Donde aparece una "b" en la tabla, Ud. debe ingresar un espacio vacío.

ABURRIDO DEL BASIC

Señor Director:

En esta carta tengo el placer de felicitarlo por su magnífica revista, la cual ayuda mucho a quien posee un microcomputador.

Me dirijo a Ud. para preguntarle si es posible ponerle un compilador para algún lenguaje al Timex Sinclair 1000, y si es así publicarlo en su revista, ya que me aburre mucho utilizar sólo el basic, y ojalá que éste no sobrepase los 8 a 10 Kbytes, porque yo tengo una memoria de 18 Kbytes.

Se despide de Ud. y agradece de antemano su atención.

Roberto A. Rozas S.
Santiago.

A nuestro conocer no existe un auténtico compilador para el Timex 1000- Sinclair ZX81. Sin embargo, si Ud. desea programar en lenguaje de máquina, existen Ensambladores que facilitan mucho esta tarea. En el mercado hay dos ensambladores diferentes a la venta:

1. "Assembler Memopack", que viene en ROM, costo aproximado \$ 12.000, y
2. "ZXASM", cassette con ensamblador y desensamblador de precio aproximado de \$ 2.000.

Ambos se obtienen a través de los distribuidores Sinclair-Timex.

Además existe otro lenguaje disponible, aunque tampoco es un compilador: es una versión en español del lenguaje LOGO, que se usó en el curso de computación de Teleduc, Canal 13. Desafortunadamente, fuera de divisarlo en Fisa '84, no sabemos ni precio ni características. Tal vez algún lector pueda complementar esta información.

Con respecto a la posibilidad de publicar un compilador, no tenemos nada en carpetas, tal vez un lector audaz pueda emprender la tarea de inventar uno.

Señor Director:

Antes que nada, permítame agradecerle la publicación de mi artículo describiendo un Micro-Ensamblador para VIC-20, y al respecto deseo aclarar algunos puntos que pueden no haber quedado claros a algunos lectores:

- En primer lugar estos programas sólo funcionan en un VIC con 5K bytes de memoria RAM. Para usarlos en un VIC que tiene expansión de memoria es necesario desconectarla antes de cargarlos.
- Creo que el título del artículo no refleja realmente su contenido, pues mi intención no es enseñar a programar en Assembler, sino describir el uso de una herramienta que hace posible practicar lo que el lector puede aprender en un libro sobre Assembler 6502.

Finalmente deseo hacer un llamado a aquellos lectores que posean un computador Commodore (tanto VIC-20 como C-64, pues ambos comparten el mismo dialecto del lenguaje BASIC, lo que permite adaptar fácilmente programas de uno al otro) para intercambiar ideas y programas. Además cualquier lector que desee una copia del Ensamblador y/o Editor sólo tiene que enviarme un Cassette, el cual gustosamente devolveré grabado.

Saluda atentamente a Ud.
Eduardo Ahumada Mazuranich
Dominica 103 Depto. 34.
Santiago.

MATRICES

De mi más alta consideración:

Por intermedio de la presente me permito felicitarle por la excelente calidad de la revista que Ud. dirige.

Aprovecho la oportunidad para expresarle un par de observaciones que bien podrían optimizar, aún más, el nivel de la revista.

- a) Dado a la poca literatura disponible en nuestro país sobre lenguaje Z-80, considero que la sección "Programando el Z-80" es uno de los mejores aportes de la revista, empero, se podría concentrar toda la información de la citada sección, ya que, al parecer por problemas de coordinación dicho tópico se trata en tres partes de la revista (v.g. MICROBYTE N° 3, pág. 19, 37 y 38), si bien es cierto que bajo enfoques diferentes pero con un común denominador, el lenguaje del Z-80 quitándole espacio a la presentación de programas interesantes para el SINCLAIR del cual tampoco existe literatura disponible.

- b) Los programas en su gran mayoría apuntan hacia el área de los juegos (SIN-CLEAR), lo que a mi juicio, como profesor, es una peligrosa minimización del potencial de la computación, en mi ciudad NO EXISTE software apuntando al área de la educación, seguramente habrá que generarlo, he aquí una línea que podría promover la revista pensando en todas las asignaturas de la enseñanza preuniversitaria.

Adjunto, sírvase encontrar una nota explicativa de un completo programa de operaciones con MATRICES y desarrollo de DETERMINANTES útil para el cuarto medio e incluso como ejercitación todo adaptado al SINCLAIR y fácil de ajustarlo al BASIC STANDARD.

Estoy preparando programas para inecuaciones y programa-

ción lineal, materia de II MEDIO, pretendo continuar con simulaciones para Física, me gustaría que me escriban si tendría el apoyo de Uds. en el sentido de publicar con cierta regularidad mis aportes educacionales de programas SINCLAIR, dada la enorme cantidad de tiempo que hay que distraer para tales efectos.

Sin otro particular, saluda atte. a Ud.

Luis Manríquez Pantoja
Prof. Mat. y Física
V. de Saldías 780 int. Chillán

Respuesta: Extraordinariamente interesantes nos parecen sus aportes, y apuntan a un problema muy serio que sólo se soluciona con trabajos como el suyo. Por supuesto que tiene Ud. nuestro entusiasta apoyo para difundir sus aportes a la nueva educación.

Le rogamos disculpar nuestro retardo en responderle. Su programa lo publicaremos en la sección Sinclair.

CLUB TRS-80

Sr. Director:

Somos un grupo de entusiastas de la computación que tenemos TRS-80 de RADIO SHACK y estamos formando un Club de Usuarios y Amigos de TRS-80 en Valparaíso.

Creemos que será ventajoso para todos los que tenemos esta línea de computadores poderlos apoyar en diversas materias.

Rogamos a los lectores que estén interesados en ingresar a este Club contactarse con Rubén Vergara Morales en Condell 1443 L-11, Valparaíso.

Esperamos que esta invitación que les proponemos encuentre una acogida favorable entre los lectores de esta revista, entre los cuales nos contamos.

Saluda atentamente a Uds.

Rubén Vergara Morales.

Efectos visuales y sonoros en el TI-99/4A

(Tercera parte)

Gustavo Mery Camposano



Continuando con el tema de la definición de caracteres, es necesario tener presente que los caracteres 32 al 127 sólo pueden ser redefinidos en modo RUN, es decir, mientras se está ejecutando un programa. En cuanto el programa se detiene estos caracteres vuelven a tomar su forma estándar. No ocurre lo mismo con los caracteres 128 al 159 (que no tienen una forma estándar), éstos mantienen la forma definida aún en modo de comandos.

Si en algún programa se redefinen algunos de los caracteres entre el 32 y el 127 y en algún momento se ejecuta un BREAK POINT ya sea por programa o externamente mediante FCTN 4, al continuar su ejecución por medio del comando CONTINUE se habrá perdido la definición de los caracteres.

Es recomendable, por lo tanto, si el programa contiene un BREAK POINT, hacer la definición de caracteres en una subrutina, la que deberá ser llamada a continuación del BREAK POINT, de este modo se podrá continuar normalmente con la ejecución de dicho programa.

En la segunda parte de este artículo (aparecida en el número seis de esta revista) presentamos un programa de ejemplo que ilustraba la forma de hacer un laser que dispara vertical y horizontalmente. Preguntábamos entonces si habría forma de hacer un laser que disparara en diagonal.

A continuación se presenta un programa de ejemplo que ilustra una forma de lograr este efecto.

```

100 REM LASER DIAGONAL
110 CALL CLEAR
120 CALL SCREEN (2)
130 CALL COLOR (1, 2, 2)
132 REM DEFINICION DE CARACTERES
135 REM LINEA VERTICAL DE LA MIRA
140 CALL CHAR (96, "0808080808080808")
145 REM LINEA HORIZONTAL DE LA MIRA
150 CALL CHAR (97, "00000000FF")
155 REM LASER IZQUIERDO
160 CALL CHAR (104, "0102040810204080")
165 REM LASER DERECHO
170 CALL CHAR (105, "8040201008040201")
175 REM EXPLOSION
180 CALL CHAR (112, "144008884414")
190 CALL COLOR (9, 16, 1)
200 CALL COLOR (10, 1, 1)
210 CALL COLOR (11, 10, 1)
220 CALL COLOR (2, 16, 1)
225 REM SE DIBUJAN LAS ESTRELLAS
230 FOR I = 1 TO 40
240 X = INT (RND * 30) + 2
250 Y = INT (RND * 24) + 1
260 CALL HCHAR (Y, X, 46)
270 NEXT I
280 CALL HCHAR (11, 15, 96)
290 CALL HCHAR (13, 15, 96)
300 CALL HCHAR (12, 14, 97)
310 CALL HCHAR (12, 96, 97)
315 REM SE DIBUJAN LOS RAYOS
320 FOR I = 13 TO 24
330 X1 = Y + 23
340 X2 = 27 - Y
350 CALL HCHAR (Y, X1, 105)
360 CALL HCHAR (Y, X2, 104)
370 NEXT I
375 REM SE DISPARA EL LASER
380 CALL KEY (0, A, ST)
390 IF ST = 0 THEN 380
400 CALL COLOR (10, 4, 1)
410 CALL SOUND (50, -7, 0)
420 CALL COLOR (10, 1, 1)
430 CALL HCHAR (12, 15, 112)
440 FOR I = 0 TO 30 STEP 5
450 CALL SOUND (80, -6, 1)
460 NEXT I
470 CALL HCHAR (12, 15, 32)
480 GOTO 380

```

Una vez cargado el programa, entrar el comando RUN para iniciar su ejecución, luego oprima una tecla cualquiera y observe el efecto.

En este ejemplo hay dos cosas que vale la pena destacar.

Si se observa la línea 150, se verá que para definir el carácter 97 se ha usado un string que contiene solamente ocho elementos. Esto es debido a que los restantes ocho elementos deben ser ceros y el BASIC lo entiende de ese modo. Es decir, cuando el string que define la forma de un carácter tiene menos de dieciséis elementos se asume que el resto de ellos, a la derecha, son ceros.

Con respecto al laser, puede observarse que los caracteres correspondientes se ponen en la pantalla en las líneas 320 a la 370, quedando allí en forma permanente, sin embargo, como el color usado es 1 (transparente) tanto para el carácter propiamente tal como para su fondo (ver línea 200), los rayos permanecen invisibles. Luego, cuando se oprime una tecla, se cambia el color del laser (línea 400) con lo que éste se vuelve visible y a continuación se vuelve a cambiar al color primitivo, de este modo se logra el efecto de destello deseado.

Si este efecto se quiere usar en un programa en el que exista algún elemento que se mueva por la pantalla, deberá cuidarse que si éste pasa por sobre los caracteres que forman los rayos, estos últimos deben ser restituidos a continuación pues de otro modo al momento de disparar el laser aparecerá como una línea cortada.

Este truco, consistente en poner caracteres invisibles en la pantalla tiene múltiples aplicaciones. Por ejemplo, se puede usar para poner mensajes destellantes con el propósito de llamar la atención. Sirve también para marcar determinados puntos o regiones de la pantalla que luego pueden ser reconocidos con la instrucción CALL GCHAR. Permite hacer dibujos completos que luego aparecen en forma repentina, etc.

En el programa SLALOM, publicado en el número siete de esta revista, se usan caracteres invisibles en las puertas, entre las banderas, de esa manera se puede detectar el momento en que el esquiador pasa por una de ellas (ver línea 230, 970 y 1060).

Otro aspecto que conviene tener presente al programar, principalmente si hay animación de dibujos se refiere a la velocidad.

Debido a que el BASIC del TI-99/4A es relativamente lento, comparado con el de otros computadores similares, no se debe esperar gran velocidad en la acción.

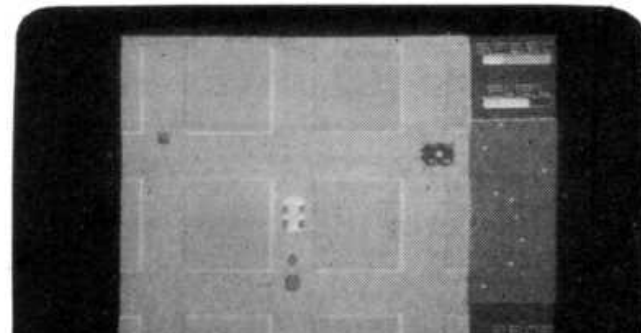
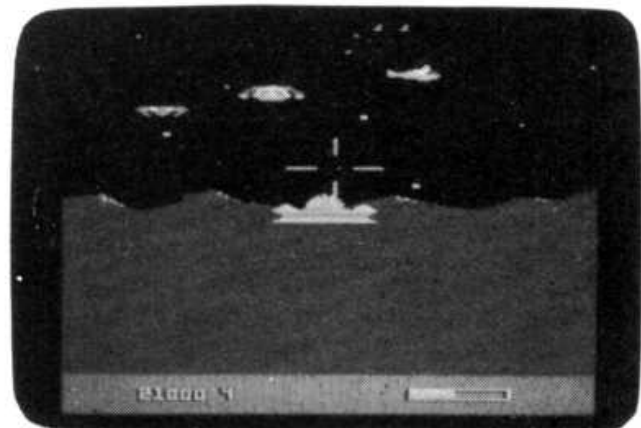
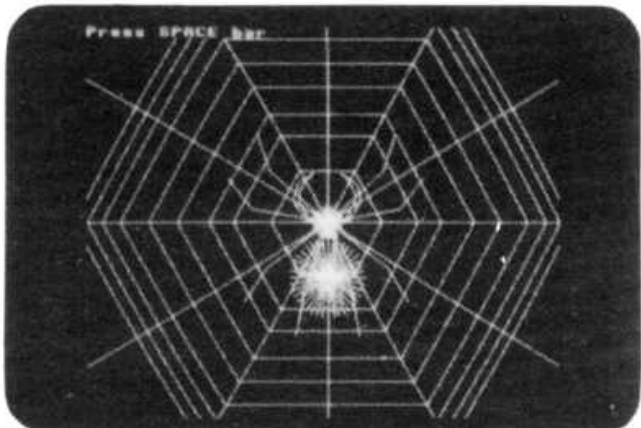
Por otra parte, la memoria disponible para el programa, es capaz de contener programas bastante largos, por lo que normalmente no es un factor que deba preocupar. De allí que a menudo sea preferible sacrificar memoria con el propósito de ganar velocidad.

Por ejemplo, es más veloz usar varias secuencias de instrucciones muy parecidas que hacer una sola secuencia con varias consultas que resuelven las diversas situaciones.

Una característica importante de la instrucción CALL CHAR es lo que podríamos llamar su efecto

retroactivo. Por medio de esta instrucción podemos definir la forma de un carácter, pero esta definición no sólo será válida para los caracteres que imprimamos a continuación, sino para aquellos que ya se encuentran en la pantalla. Por ejemplo, si tenemos impresas varias letras A y luego redefinimos el carácter 65 dándole una nueva forma, todas las A que teníamos desplegadas adoptarán la nueva forma.

Esta técnica puede ser muy útil cuando se desee hacer cambios rápidos en los dibujos que se tiene en la pantalla.



Operando con matrices

El profesor Luis Manríquez P. nos escribe de Chillán para hacernos llegar una muy digna colaboración. Un programa que resuelve todo lo que siempre quiso hacer con matrices y no sabía cómo.

Las matrices se estudian actualmente en cuarto medio y desde el próximo año figuran en los planes electivos de tercero medio. Para el nivel de enseñanza media, este programa permite comprender la teoría con facilidad mientras que en la enseñanza universitaria constituye una gran herramienta de ejercitación sobre todo en la inversa que es realmente difícil de obtener "a mano" y prácticamente todos los libros omiten los resultados por falta de espacio.

Con este programa, se puede sumar, restar, multiplicar, obtener la matriz inversa, ponderar una matriz y desarrollar un determinante de orden 4. Aquellos que sólo deseen utilizar una parte del programa, pueden omitir tipear el resto guiándose por los IF... que siguen al menú. Otra modificación interesante puede ser cambiar los dimensionamientos de acuerdo a la capacidad de memoria del equipo que esté utilizando. Originalmente, el programa sólo acepta matrices de 5 por 5.

```

10 REM PROF. L. MANRIQUEZ
20 REM CHILLAN JULIO 84
30 REM MATRICES
40 REM *****
41 DIM A(5,5)
42 DIM B(5,5)
43 DIM T(5,5)
44 DIM M(5,5)
45 DIM R(5,5)
46 DIM P(5,5)
47 DIM K(5,5)
48 DIM C(5,5)
50 PRINT AT 10,3,"OPERACIONES CON MATRICES"
60 PAUSE 100
70 CLS
80 SLOW
90 PRINT TAB 10,"*****MENU*****"
91 PRINT TAB 10,"*****"
95 PRINT "1.- SUMA"
97 PRINT "2.- RESTA"
99 PRINT "3.- MULTIPLICACION"
100 PRINT "4.- MATRIZ TRASPUESTA"
102 PRINT "5.- MATRIZ INVERSA"
104 PRINT "6.- PONDERACION"
106 PRINT "7.- DESARROLLO DETERMINANTES." (data: ORDEN 4)"
107 PRINT "*****"
108 PRINT AT 15,5,"TRABAJAR CON MATRICES HASTA DE 5X5"
110 PRINT AT 20,1,"PRESIONE NUMERO SEGUN ELECCION"
120 IF INKEY="" THEN GOTO 120
125 LET Q=INKEY
130 IF Q="1" THEN GOTO 1000
140 IF Q="2" THEN GOTO 1000
150 IF Q="3" THEN GOTO 2000
160 IF Q="4" THEN GOTO 1000
170 IF Q="5" THEN GOTO 3000
180 IF Q="6" THEN GOTO 3500
185 IF Q="7" THEN GOTO 4000
190 GOTO 120
500 REM ORDEN
510 CLS
515 IF VAL Q>3 THEN GOTO 530
520 PRINT TAB 5,"INGRESE DATOS SE PIDEN"
521 PRINT
525 PRINT "PRIMERA MATRIZ"
530 PRINT "NUMERO DE FILAS "
535 INPUT F1
540 PRINT "NUMERO DE COLUMNAS"
545 INPUT C1
546 IF F1>5 OR C1>5 THEN GOTO 577
547 IF VAL Q>3 THEN GOTO 590
548 CLS
550 PRINT "SEGUNDA MATRIZ"
560 PRINT "NUMERO DE FILAS "
565 INPUT F2
570 PRINT "NUMERO DE COLUMNAS"
575 INPUT C2
576 IF F2>5 OR C2>5 THEN GOTO 590
577 CLS
580 PRINT "MATRIZ SOBRE DIMENSIONADA"
590 GOTO 100
590 RETURN
600 REM INGRESO INFORMACION
601 CLS
610 IF VAL Q>3 THEN GOTO 640
620 PRINT "INGRESE ELEMENTOS PRIMERA MATRIZ "
630 GOTO 647
640 PRINT "INGRESE ELEMENTOS DE LA MATRIZ O DETERMINANTE"
645 PRINT AT 3,5,"ELEMENTO"

```

```

647 FAST
650 FOR I=1 TO F1
660 FOR J=1 TO C1
670 PRINT AT 3,10,I," ",J
680 INPUT A(I,J)
695 LET B(I,J)=0
696 LET C(I,J)=A(I,J)
699 LET T(I,J)=A(I,J)
700 NEXT J
705 LET B(1,1)=1
710 NEXT I
720 PAUSE 50
730 CLS
740 IF VAL Q>3 THEN GOTO 900
750 PRINT "INGRESE ELEMENTOS SEGUNDA MATRIZ"
760 PRINT AT 3,5,"ELEMENTO"
770 FOR I=1 TO F2
780 FOR J=1 TO C2
790 PRINT AT 3,10,I," ",J
800 INPUT B(I,J)
810 LET S(I,J)=A(I,J)+B(I,J)
820 LET R(I,J)=A(I,J)-B(I,J)
830 NEXT J
840 NEXT I
900 RETURN
1000 GOSUB 500
1001 IF Q="1" OR Q="2" THEN GOTO 1100
1002 GOSUB 400
1004 FOR I=1 TO C1
1005 FOR J=1 TO F1
1010 LET M(I,J)=T(I,J)
1020 NEXT J
1030 NEXT I
1040 LET H1=C1
1050 LET H2=F1
1060 GOTO 5000
1100 IF F1=F2 AND C1=C2 THEN GOTO 1150
1105 CLS
1110 PRINT "MATRICES DE DISTINTO ORDEN"
1120 GOTO 90
1150 GOSUB 600
1151 IF Q="2" THEN GOTO 1250
1159 FOR I=1 TO F1
1160 FOR J=1 TO C1
1170 LET M(I,J)=S(I,J)
1180 NEXT J
1190 NEXT I
1200 GOTO 1300
1250 FOR I=1 TO F1
1260 FOR J=1 TO C1
1270 LET M(I,J)=R(I,J)
1280 NEXT J
1290 NEXT I
1300 LET H1=F1
1310 LET H2=C1
1320 GOTO 5000
2000 REM MULTIPLICACION
2010 GOSUB 500
2020 IF C1=F2 THEN GOTO 2060
2030 CLS
2040 PRINT "PRODUCTO IMPOSIBLE"
2050 GOTO 90
2060 GOSUB 400
2100 LET SUM=0
2110 FOR K=1 TO F1
2120 FOR Y=1 TO C2
2130 GOSUB 2300
2140 LET M(K,Y)=SUM
2150 LET SUM=0
2160 NEXT Y
2170 NEXT K
2180 LET H1=F1
2190 LET H2=C2
2200 GOTO 5000
2300 FOR K=1 TO C1
2310 LET S=H1*K+H2*B(K,Y)
2320 LET SUM=SUM+S
2325 NEXT K
2330 RETURN
3000 REM M. INVERSA
3010 GOSUB 500
3020 IF F1=C1 THEN GOTO 3055
3030 CLS
3040 PRINT "NO ES M. CUADRADA"
3050 GOTO 90
3055 GOSUB 400
3060 FOR I=1 TO F1
3070 FOR J=1 TO F1
3080 IF A(I,J)=0 THEN GOTO 3120
3090 NEXT J
3095 CLS
3096 SLOW
3100 PRINT "MATRIZ SINGULAR"
3110 GOTO 90
3115 FAST
3120 FOR K=1 TO F1
3130 LET S=A(I,J)
3140 LET A(I,K)=A(I,K)+S
3150 LET A(I,K)=S
3160 LET S=A(I,K)
3165 LET B(I,K)=B(I,K)+S
3170 LET B(I,K)=S
3180 NEXT K
3190 LET T=1/A(I,J)
3200 FOR K=1 TO F1
3210 LET R(I,K)=T*A(I,K)
3220 LET B(I,K)=T*B(I,K)
3230 NEXT K
3240 FOR L=1 TO F1
3250 IF L=J THEN GOTO 3310
3260 LET T=-1*A(L,J)
3270 FOR K=1 TO F1
3280 LET A(L,K)=A(L,K)+T*A(J,K)
3290 LET B(L,K)=B(L,K)+T*B(J,K)
3300 NEXT K
3310 NEXT L
3320 NEXT J
3330 SLOW
3340 CLS
3350 PRINT TAB 10,"M. INVERSA"
3360 PRINT
3370 FOR I=1 TO F1
3380 FOR J=1 TO F1
3390 PRINT INT (A(I,J)/1000+0,5)/1000
3400 PRINT " "
3410 NEXT J

```




Cronómetro de segundos.

Hace algún tiempo, al hacer un pequeño recorrido por algunas direcciones de memoria interesantes del Atari, mencionamos las direcciones 18, 19 y 20 como apropiadas para llevar un control de tiempo real. Jorge Alegría nos ha enviado un programa que permite utilizar estas direcciones.

En las direcciones mencionadas, el computador va contando los pulsos que envía a la pantalla.

En los equipos que se comercializan en Chile (sistema americano), el computador envía 60 pulsos por segundo a la pantalla. Primero, se va llenando la dirección 20 (byte menos significativo). Al alcanzar el valor máximo que puede contener un byte, 255, esta dirección vuelve a 0 y se incrementa la dirección 19 en 1. Cuando esta dirección a su vez alcanza su valor máximo, vuelve a 0 y se in-

crementa en 1 la dirección 18. De este modo, el Atari puede llegar a contar más de 16 millones de pulsos, lo que equivale a más de tres días a razón de 60 pulsos por segundo.

La fórmula para traducir los ciclos a segundos aparece en la línea 30 del listado, y al ejecutar el programa, aparece en el ángulo superior izquierdo de la pantalla el segundero.

```
0 REM CRONOMETRO DE SEGUNDOS
1 REM *****
2 REM * PARA MICROBYTE *
3 REM * JORGE ALEGRIA *
4 REM * JULIO 1984 *
5 REM *****
6 GRAPHICS 2
9 REM * BORRA PANTALLA: BORRA CURSOR
10 PRINT CHR$(125):POKE 752,1
19 REM EL RELOJ USA DIRECCIONES DE MEMORIA 18,19,20. LAS DEJO EN CERO
20 POKE 18,0:POKE 19,0:POKE 20,0
30 TIME=INT((PEEK(18)*65536+PEEK(19)*256+PEEK(20))/60)
40 POSITION 0,0:PRINT #6;TIME
50 GOTO 30
```

Rutina de edición.

Para quienes resulta tedioso editar un programa teniendo que listar, parar el listado con CTR-1, detenerse con Break y luego moverse con el cursor por la pantalla hasta ubicar la línea a editar, la rutina que les presentamos les será de gran utilidad.

Para utilizarla, primero tipeela y guárdela con LIST. Luego car-

gue el programa que desea editar con LOAD y luego ENTER la rutina. Ponga atención que en su programa no haya instrucciones con número de línea 0 o superiores a 32000, de otro modo las instrucciones de la rutina borrarán las de su listado.

Al ejecutar a continuación el programa, el listado de éste irá apareciendo línea por línea. Si

desea editar, tipee una E y entrará en modo edición. Para continuar luego, tipee CONT y continuarán apareciendo las siguientes líneas de su programa.

Al finalizar la rutina, ésta se borra automáticamente, quedando en memoria sólo su programa principal. La rutina de borrado automático es explicada a continuación.

```
0 GOTO 32000:REM Rutina Edición
32000 L=PEEK(136)+PEEK(137)*256
32010 N=PEEK(L)+256*PEEK(L+1)
32020 L=L+PEEK(L+2)
32030 IF N>32000 THEN STOP
32040 IF N=0 THEN 32010
32050 LIST N
32060 K=PEEK(764)
32070 IF K=42 THEN POKE 764,255:STOP
32080 IF K=33 THEN POKE 764,255:PRINT "I":GOTO 32010
32085 IF PEEK(764)=12 THEN 32100
32090 GOTO 32060
32100 END:GRAPHICS 0:POSITION 2,4:FOR I=32000 TO 32130 STEP 10:PRINT I:NEXT I
32110 PRINT 32085:PRINT "POKE 842,12"
32120 POSITION 2,0
32130 POKE 842,13
```


ATARI[®] COMPUTADORES

La línea más completa en computadores, periféricos y software.



ATARI 600 XL:
COMPUTADOR CON 16KB MEMORIA

Expandible a 64KB, mediante módulo externo.
24KB en ROM, incluyendo lenguaje BASIC.
Teclado profesional con 62 teclas, 16 modos gráficos distintos. Alta resolución en pantalla (320 x 192).
256 colores disponibles, 4 sintetizadores de sonido.
Bus de expansión exterior y 2 puertas para controladores.

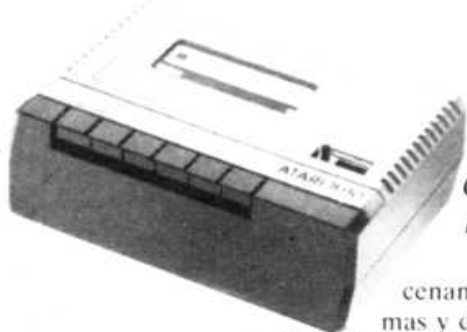
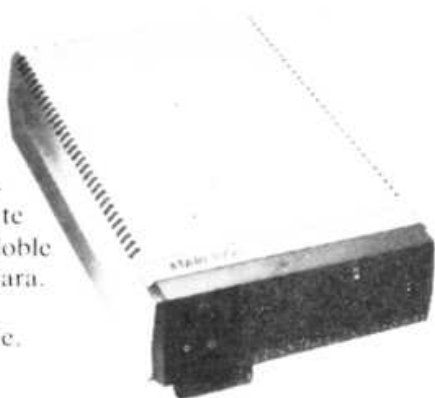


ATARI 1027:
IMPRESORA DE CALIDAD

Impresora de 80 caracteres por línea, con caracteres de calidad de correspondencia.
Imprime sobre hojas de papel corriente a razón de 20 caracteres por segundo. Interfase directa al computador.

ATARI 1050:
DISKETTERA

Unidad de almacenamiento en mini diskette de 5 1/4" pulgadas de doble densidad y una sola cara. Capacidad de 127KB por diskette.



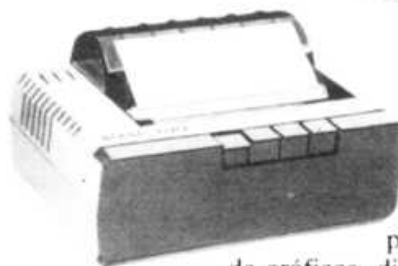
ATARI 1010:
GRABADORA DE CASSETTE

Unidad de almacenamiento de programas y datos en cintas de cassette normal. Capacidad de 100 KB en cinta de 60 minutos. Dispone de canal de audio controlable por software.

ATARI 800 XL:

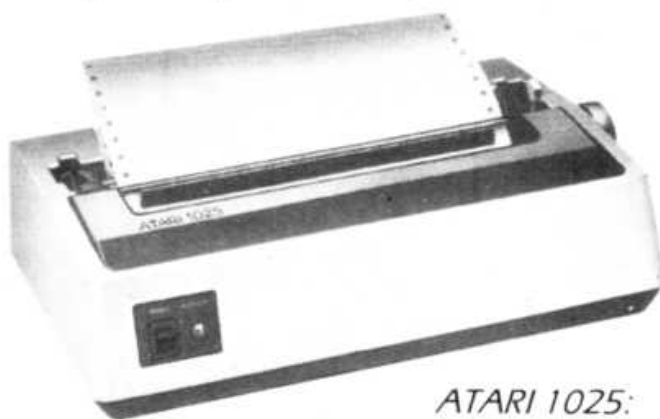
COMPUTADOR CON 64KB MEMORIA

24KB en ROM, incluyendo lenguaje BASIC.
Teclado profesional con 62 teclas, 16 modos gráficos distintos. Alta resolución en pantalla (320 x 192).
256 colores disponibles, 4 sintetizadores de sonido.
Bus de expansión exterior y 2 puertas para controladores. Salida a monitor de video.



ATARI 1020:
IMPRESORA A COLORES

Impresora gráfica para elaboración de gráficos, diagramas o cualquier forma de arte por computadora. Hace uso de todas las capacidades gráficas del computador ATARI.



ATARI 1025:
IMPRESORA 80 COLUMNAS

Impresora de matriz de puntos por impacto, imprime hasta 80 caracteres por línea a razón de 40 caracteres por segundo, en papel corriente. Interfase directa al computador.

Adquiéralas en la más selecta red de distribuidores, a lo largo del país.

**Computadores con
respaldo y
garantía de Coelsa.**





Lectura forzada.

En la rutina de edición, al final veíamos que ésta se borraba automáticamente. Para esto se utiliza la dirección de memoria 842. Con un POKE 842,13 se obliga al computador a recorrer la pantalla con Returns. Por el modo de edición del Atari, cada uno de estos Returns equivale a leer la línea de la pantalla donde está ubicado el cursor. Si en esta línea hay escrito un número, el Atari entiende que es una línea de instrucción. Si a continuación del número no hay nada más, esto equivale a borrar una instrucción con ese número de línea. POKE 842,12 devuelve la situación a la normalidad.

En el ejemplo agregamos nuevas líneas a nuestro listado. Primero tipéelo, córralo y luego lístelo de nuevo para asegurarse de que efectivamente agregó nuevas líneas. A continuación escriba un REM en la línea 20 y saque el REM de la línea 25 para activarla y corra nueva-

mente el programa y comprobará que se borraron las líneas que había agregado antes. Esta es la teoría. En la práctica, usted puede utilizarla en sus programas, sea para protegerlos, sea para ahorrar memoria o para lo que usted se le ocurra.

```
10 FOR I = 40 TO 140 STEP 10
20 PRINT I;" PRINT ";I
25 REM PRINT I
30 NEXT I
34 PRINT "POKE 842,12"
36 POS .2,0
145 POKE 842,13
```

Estrella perfecta

Sres. Microbyte:

Soy un fiel lector de vuestra revista, tengo 15 años, poseo un Atari 600XL e integro, en mi colegio, una academia de computación de la cual soy monitor y también alumno.

He observado que en dos de vuestros números se publican banderas chilenas cuyas estrellas, o no corresponden, o sólo se acercan a la verdadera.

En una clase de geometría pasamos la distancia exacta que debe existir entre las puntas de la estrella de la bandera; este resultado es:

$$r \sqrt{\frac{5 - \sqrt{5}}{2}}$$

La misma distancia se obtiene al inscribir en una circunferencia un pentágono regular, cuyos ángulos deben medir 72 grados; luego, prolongado sus lados, se obtiene la ESTRELLA PERFECTA.

Este programa, además de dibujar la bandera, dibuja la estrella desde 1 punto y luego agranda el radio de 1 en 1 para producir la pintura de la misma.

Sin otro particular, me despido felicitándoles por vuestra revista.

Gonzalo Concha-Laborde

```
0 REM *****
1 REM * BANDERA CON *
2 REM * ESTRELLA PERFECTA *
3 REM * *****
4 REM * GONZALO CONCHA-LABORDE *
5 REM * *
6 REM *****
8 GOTO 200
10 XC=30:YC=24
20 RD=1:YS=0.78
25 REM AHORA LA ESTRELLA
30 COLOR 2
40 DEG : PLOT XC,YC-RD*YS
50 DRAWTO XC-SIN(-144)*RD,YC-COS(-144)*RD*YS
60 DRAWTO XC-SIN(-288)*RD,YC-COS(-288)*RD*YS
70 DRAWTO XC-SIN(-72)*RD,YC-COS(-72)*RD*YS
80 DRAWTO XC-SIN(-216)*RD,YC-COS(-216)*RD*YS
90 DRAWTO XC-SIN(-360)*RD,YC-COS(-360)*RD*YS
100 RD=RD+1: IF RD=18 THEN 300
110 GOTO 30
190 REM LA BANDERA
200 GRAPHICS 7+16: SETCOLOR 1,0,14:C=2
210 FOR I=0 TO 12: COLOR C
220 FOR J=0 TO 6:PLOT 0,I*7+J
230 DRAWTO 159,I*7+J
240 NEXT J
250 IF I>5 THEN C=1
260 NEXT I
270 COLOR 3
280 FOR I=0 TO 48:PLOT 0,I:DRAWTO 60,I:NEXT I
290 GOTO 10
300 GOTO 300
310 REM *****
320 REM * TABLA DE VARIABLES *
330 REM * XC : COORDENADA CENTRO ESTRELLA *
340 REM * YC : " " " " *
350 REM * RD : RADIO DEL PENTAGONO *
360 REM * YS : FACTOR DE ESCALA DEL DIBUJO *
370 REM *****
```


SMITH-CORONA®

La respuesta americana en impresoras.

| MODELO | Características | | VELOC. | BIDIREC. |
|--------|-----------------|-----------------|--------|----------|
| | NOR. COL. | ANCHO MAX. COL. | | |
| F-80 | 80 | 132 | 80 | S |
| D-200 | 80 | 132 | 160 | S |
| D-300 | 80 | 132 | 180 | S |
| TP-II | 132 | 233 | 12 | N |
| | 100 | - | | |



Exíjalas en los mejores negocios de computación.

Exíjalas en los mejores negocios de computación.
 TUCAN INGENIERIA Y CIA. LTDA. Chile. Santiago.
 Representante oficial para Ojeda 2125, 494085 Telex: 240177 VOAG-CL.
 Luis Thayer Ojeda 742453, Correo Central, Santiago.
 Fonos: Casilla 1261.



- Multiusuario
- Multitarea
- Multiprogramación
- Software hecho a la medida de sus necesidades
- Amistosidad: Relación conversacional entre usuario y computador
- Alpha Micro le ofrece equipos adaptables a su empresa, con nosotros no necesita que su empresa se adapte a un computador

PARA VER Y CREER

alpha micro



TUCAN Ingeniería y Cía. Ltda.
 Representante oficial para Chile.
 Luis Thayer Ojeda 2125, Santiago.
 Fonos: 742453, 494085.
 Télex: 240177 VOAG-CL
 Casilla 1261, Correo Central, Santiago.