

MICROHOBBY

AMSTRAD

Semana

AÑO II N.º 38

REVISTA INDEPENDIENTE PARA USUARIOS DE ORDENADORES AMSTRAD

160 Ptas.

Canarias 165 pts.

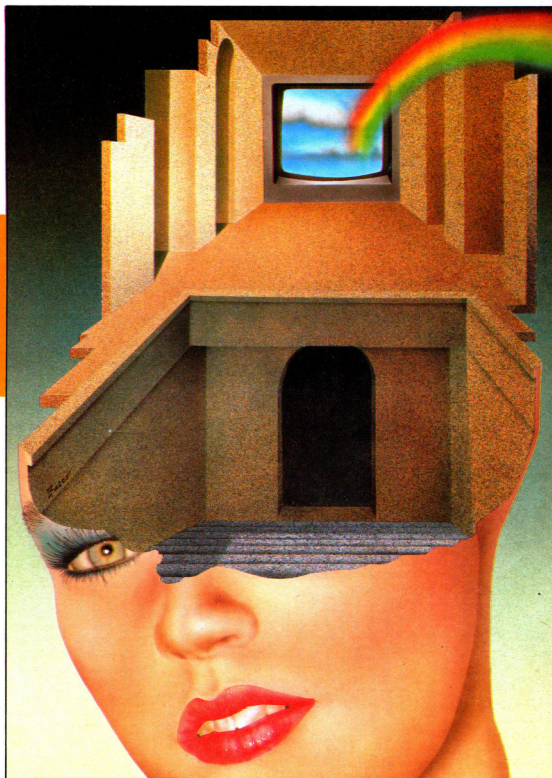
**Amscal: Hoja
de Cálculo
profesional para
Aplicaciones
de Gestión**

**¿PUEDE PENSAR
TU AMSTRAD?**

**LOS SECRETOS DE
LA INTELIGENCIA
ARTIFICIAL,
UNO A UNO**

**ANALIZAMOS
UN PROGRAMA
DE FACTURACION
PARA EL 8256
CREADO
INTEGRAMENTE
EN ESPAÑA**

**COMO DOMINAR
LAS MAS
SOFISTICADAS
FUNCIONES PARA
EL TRATAMIENTO
DE PALABRAS**



¡NOVEDAD MUNDIAL!

AMSTRAD

1^{er} JUEGO PCW-8256

3-D CLOCK CHESS

OPINA

Jon Speelman

Maestro Internacional (Campeón Británico - 1985)

"Con este programa de juego y los excelentes gráficos 3-D, puedo recomendar personalmente 3-D CLOCK CHESS a todos los amantes del Rey de los Juegos"



PRINCIPALES CARACTERÍSTICAS:

- POTENTE:** Reúne las últimas técnicas desarrolladas para los programas de ajedrez basadas en la inteligencia rápida, y puedes determinar los niveles de juego seleccionando el tiempo de respuesta en cualquiera de los cuatro modos de juego, lo que te da cientos de posibles niveles.
- ANALÍTICO:** Este programa tiene una fuerza insospechada: en los niveles de torneo el ordenador se plantea hasta siete movimientos consecutivos eligiendo literalmente entre decenas de miles de líneas de juego.
- AMISTOSO:** Posee todas las posibilidades que un verdadero jugador de ajedrez puede necesitar, llevándote totalmente informado acerca de su propio proceso pensante. Es un profesor ideal para el principiante y un tutor ideal para el experto.
- SUPERGRÁFICOS 3-D:** Los últimos desarrollos en visión tridimensional y bidireccional del tablero y piezas del ajedrez, así como de la exclusiva opción del Torneo Cronometrado de Ajedrez, con la interesante posibilidad de "jugadas en tiempo compensado".

PVP. 3950 pts
+ IVA

ACE DISTRIBUCION

Actividades Comerciales y Electrónicas, S.A.
C/ Tarragona, 110-112 - Tel. 325 10 58 - 08015 Barcelona. Telex 93133 ACE EE E

Producido en exclusiva para España por:

ACE SOFTWARE, S.A.

© CP SOFTWARE

AMSTRAD

sumario

Año II • Número 38 • 20 de Mayo de 1986
160 ptas. (incluido I.V.A.)
Canarias, 155 ptas. + 10 ptas. sobretasa aérea.
Ceuta y Melilla, 155 ptas.

Director Editorial
José I. Gómez-Centurión

Director Ejecutivo

José M.º Diaz

Redactor Jefe

Juan José Martínez

Diseño gráfico

José Flores

Maquetación

Fernando Chanmel

Colaboradores

Eduardo Ruiz

Javier Barceló

David Sepuerta

Robert Chatwin

Francisco Portalo

Pedro Sudón

Miguel Sepúlveda

Francisco Martín

Jesús Alonso

Pedro S. Pérez

Amalia Gómez

Secretaría Redacción

Carmen Santamaría

Fotografía

Carlos Candel

Portada

M. Barco

Ilustradores

J. Iguel, J. Pons, F. L. Frontán,

J. Septien, Pejo, J. J. Mora

Edita

HOBBY PRESS, S.A.

Presidente

Maria Andriño

Consejero Delegado

José I. Gómez-Centurión

Jefe de Producción

Carlos Peropadre

Marketing

Marta García

Jefe de Publicidad

Concha Gutiérrez

Publicidad Barcelona

José Galán Cortés

Tel: (93) 303 10 22/313 71 62

Secretaría de Dirección

Maria Cogorro

Suscripciones

M.º Rosa González

M.º del Mar Calzada

Redacción, Administración

y Publicidad

La Granja, 39

Polígono Industrial de Alcobendas

Tel.: 654 32 11

Telex: 49 480 HOPR

Dto. Circulación

Paulino Blanco

Distribución

Coedis, S. A. Valencia, 245

Barcelona

Imprime

ROTEDEC, S. A. Crta. de Irún,

Km. 12,450 (MADRID)

Fotocomposición

Novocomp, S.A.

Nicolás Morales, 38-40

Fotomecánica

GROF

Ezequiel Solana, 16

Depósito Legal:

M-28468-1985

Derechos exclusivos

de la revista

COMPUTING with

the AMSTRAD

Representante para Argentina, Chile,

Uruguay y Paraguay, Cia.

Americana de Ediciones, S.R.L. Sud

América 1.532. Tel.: 21 24 64. 1209

BUENOS AIRES (Argentina).

M. H. AMSTRAD no se hace

necesariamente solidaria de las

opiniones vertidas por sus

colaboradores en los artículos

firmados. Reservados todos los

derechos.

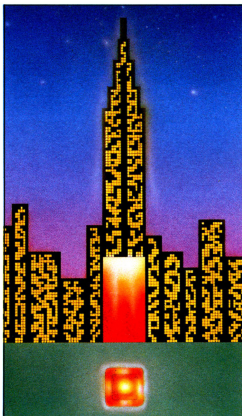
Se solicitará control OJD

5 Primera plana

Campeonato «Mercenario» de Master Computer. Música para tu Amstrad. Colossus Chess está aquí.

8 Primeros pasos

Continuamos analizando las técnicas esenciales de animación de figuras en pantalla, las cuales son un imprescindible preámbulo para pasar, posteriormente, a otras más sofisticadas, que permitirán dotar de vida a nuestros programas.

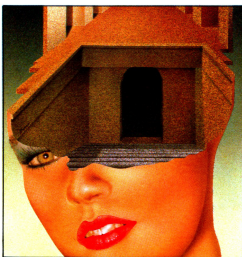


Serie Oro 18

Tenemos esta semana un programa verdaderamente excepcional; se trata de una completa hoja de cálculo, con las lógicas limitaciones que implica estar escrita en Basic, pero que resolverá, sin duda, los problemas de más de un usuario por muy poco dinero.

Inteligencia artificial 24

AMSTRAD Semanal inaugura, con este artículo, un curso sobre Inteligencia Artificial. Creemos ser la primera revista que aborda esta gigantesca tarea, y nos proponemos enseñar a todo el mundo cómo hacer programas inteligentes, dotándoles para ello de las herramientas adecuadas.



Para...PCW 30

Se acabaron los tediosos problemas de la facturación «artesanal», gracias al programa que mostramos esta semana, y que convierte a un 8256 en una herramienta de prestaciones comparables o superiores a ordenadores de mucho mayor precio.

34 ProgramAcción

Todas y cada una de las más sofisticadas funciones para el tratamiento de textos, unas existentes en el Basic y otras creadas por nosotros, junto con programas ejemplo sencillos y útiles.



Todo sobre AMSTRAD



1ª FERIA INFORMÁTICA

¡Ven a conocer el apasionante mundo de los ordenadores Amstrad!

Las más importantes empresas españolas y europeas del sector se dan cita en Madrid para presentar y ofrecer sus más recientes productos para **AMSTRAD**.

Programas de acción, juego, aventuras... Programas educativos, de utilidades, lenguajes... Programas de gestión y profesionales... Cientos de títulos inéditos...

Periféricos, ampliaciones de memoria, emuladores,

tabletas gráficas, digitalizadores, impresoras, lápices ópticos, redes de comunicación, discos duros, sintetizadores de voz, correo electrónico, tratamiento de imágenes...

Las últimas novedades editoriales... Todas las revistas...

Una ocasión única para conocer de "primera mano" los increíbles ordenadores personales **AMSTRAD** y todo cuanto para ellos se produce en el mundo.

- Patrocinada y organizada por **AMSTRAD ESPAÑA**
- Horario continuo de 10:00 a 19:30
- Entrada: **200 ptas.**
- Sorteo de Ordenadores **AMSTRAD** entre los visitantes.



23-24-25 MAYO
Palacio de Exposiciones y Congresos de Madrid

P.º Castellana, 99. 28046 MADRID

MIRRORSOFT ATACA DE NUEVO

Mirrorsoft ha lanzado un juego, denominado «Biggles», creado por el famoso Neil Dickson, en colaboración con un «programador musical» tres diseñadores de software más.

Biggles es un juego mezcla de arcade e inteligencia, que se desarrolla en varias partes, algunas de las cuales tienen mucho que ver con el film del mismo nombre.

Cada una de las diversas partes del juego debe ser completada antes de poder acceder a las demás. Sin embargo, cuando se alcanza un determinado estado crítico, podemos pasar a otra fase del juego completamente distinta, en la cual necesitamos unos recursos y una planificación para solventarla que no tienen nada que ver con los anteriores.

MONITORES A VIDEOS

MASTER-HARD presenta en España el CONVERTIDOR DE MONITORES.

Mediante este sistema MASTER-HARD ofrece al usuario de Amstrad o de cualquier otro tipo de monitor (incluido el sistema Amicano) la posibilidad de convertir su monitor tanto en COLOR como en FOSFORO VERDE en un completo y perfecto receptor de televisión o terminal de vídeo.

El sistema comercializado por MASTER-HARD incorpora un selector de canales y una conexión para aparato de vídeo.

Con una sencilla instalación, conseguirá ver en el monitor de su ordenador, tanto su programa como su película favorita, mejor que en su televisor normal gracias a la alta definición de los monitores.

**MASTER-HARD
C/ MAGDALENA, 213
FERROL**



COLOSSUS CHESS 4.0

Otro programa de ajedrez se incorpora a la ya larga lista de simulaciones por ordenador del inmortal juego; se trata de «Colossus Chess», creado por «CDS Software Ltd» y tridimensional, faltaría más.

Posee las habituales características de este tipo de programas, como librería de aperturas, distintos niveles de juego, repetición de partidas, relojes de tiempo transcurrido y la interesante posibilidad de jugar en modo supervisor, esto es, dos personas con el programa como árbitro.

«Colossus Chess» existe en versiones de disco y cassette, y corre en todos los CPCs.

Distribuye Serma
Bravo Murillo 377, 3-A
Tel. (91) - 733 73 11
Madrid

MUSICA PARA AMSTRAD

Suma Computers, una casa de software inglesa bien conocida en el mundo de Amstrad por su amplia gama de programas, acaba de lanzar en Inglaterra su programa Music Minstrel, compatible con toda la gama CPC.

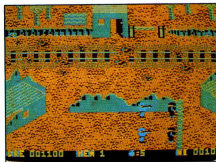
Se trata de un editor musical, que permite tocar música en el Amstrad usando la notación estándar, esto es, basada en el pentagrama. Dicha música, además de escucharse, puede modificarse y guardarse en disco para posterior uso.

Minstrel permitirá el uso de las tres

Primera PLANA

CAMPEONATO MERCENARIO

MASTER COMPUTER, con la colaboración de Micro-Byte, celebra el 2.º Campeonato MASTER COMPUTER en el juego «MERCENARIO». Tendrá lugar en los locales de MASTER COMPUTER en el Centro Comercial El Bulevar en La Moraleja (Alcobendas). Habrá premios para los mejores y regalos para todos los participantes. El precio de la inscripción será de 200 ptas. El campeonato se celebrará el sábado 14 de junio a las seis y media de la tarde.



Master computer está en:

**MASTER COMPUTER
CENTRO COMERCIAL, LOCAL 15.
URB. CIUDAD STO. DOMINGO.
CTRA. DE BURGOS, KM. 28. ALGETE
(MADRID).
TELEFONO (91) 622 12 89.**

vores simultáneamente, usando a tope las facultades polifónicas de los Amstrad.

Music Minstrel es un programa sencillo por menús, al parecer muy sencillo de usar y altamente recomendado para compositores musicales; por lo menos eso clama a los cuatro vientos Jon Day, director de ventas de Kuma Computers.

Existen versiones en disco y cassette, al precio de 19.95 \$ y 14.95 \$, respectivamente.

Kuma Computers Ltd. 12 Horsehoe Park.

Pangbourne. RG7 7 JW. teléfono 073 57-43 35.

Ofites

Presenta: el universo del software,

DELTA
+

La más moderna base de datos DELTA, superándose a sí misma, "DELTA +", desarrollada para CP/M por COMSOFT con todo en español.

Diseña sus propios ficheros; desde un simple fichero de nombres y direcciones hasta su propio sistema contable. El formato estándar DIF permite intercambiar datos en DELTA, desde las hojas de cálculo CRACKER II, etc... y viceversa. Intercambio de datos con la mayoría de los tratamientos de texto como NEWWORD para MAILING.

Incluye un sencillo y funcional sistema de impresión de etiquetas con: hasta 5 columnas de etiquetas, 65 caracteres por etiquetas, 20 líneas con 3 campos cada una.

- PROGRAMABLE Y RELACIONAL.
- FICHEROS INDEXADOS.
- HASTA 90 CAMPOS ó 2.000 CARACTERES.
- MULTIPLES SISTEMAS DE BÚSQUEDA, 8 CLAVES.
- FICHEROS DE HASTA 8 Mb.
- 8 GRUPOS DE TRANSACCION POR REGISTRO.

BASE
DE DATOS

17.850 pts.

NEWWORD

Programa de tratamiento de textos mejorando todo lo anterior. Manual y programa en español, que le enseñarán con facilidad y rapidez lo más avanzado en procesadores de textos. Compatibilidad funcional con WORDSTAR incluyendo muchas capacidades adicionales.

Tiene un potente MAIL-MERGE con opción de selección de destinatarios por criterios base de datos, creación de documentos, impresión de etiquetas. Utiliza todo el espacio de disco. Ensamblaje de textos, sustitución, etc., de la forma más fácil: autohace copias de seguridad. ¡NUNCA PERDERA UN TEXTO!

- Ñ, ACENTOS, DIERESIS, ETC...
- PRESENTACION EXACTA EN PANTALLA DEL FUTURO DOCUMENTO IMPRESO.
- INTERCAMBIOS DE FICHEROS CON CRACKER.
- VARIABLES SUSTITUIBLES EN IMPRESORA.
- POTENTE CALCULADORA.
- COMPROBADOR ORTOGRAFICO Y GRAN DICCIONARIO (45.000 TERMINOS AMPLIABLES).
- POSIBILIDAD DE LECTURA DE FICHEROS DE DELTA, CARD BOX, SUPERCALC, DBASE II, ETC...

TRATAMIENTO
DE TEXTOS

17.850 pts.

CRACKER II

El CRACK de las hojas de cálculo, la que deja atrás al resto. Funciones nunca vistas, formato de fechas, salvaguardia continua sobre un fichero. Realiza automáticamente copias de seguridad. Además de las tradicionales funciones, CRACKER II posee funciones lógicas, estadísticas y de alta matemática. Interchangea datos con NEWWORD, bases de datos y la mayoría de las hojas de cálculo.

- CELDAS PROGRAMABLES.
- FUNCIONES ESPECIALES:
Fecha, días; desde y hasta la fecha de la semana, del año, lapso de tiempo, retraso, beep entrada, saludo usuario.
- SISTEMA DE AYUDA ON-LINE.
- SUMA CONDICIONAL.
- TOMAR DECISIONES EN LA HOJA.
- 18 MODOS GRAFICOS DISTINTOS.
- TRADICIONALES FUNCIONES MATEMATICAS Y AMPLIACION, FUNCIONES ESTADISTICAS Y LOGICAS.
- GENERA GRAFICOS EN BASE A LOS DATOS.

HOJA
DE CALCULO

17.850 pts.

EDITOR Y DISTRIBUIDOR EXCLUSIVO PARA ESPAÑA

IVA
no
incluido

TOTALMENTE
EN
ESPAÑOL

Informática

estas son sus estrellas.

NUCLEUS

NUCLEUS más que una estrella una constelación; tres ESTRELLAS en un SUPERPROGRAMA, la solución a cualquier aplicación por compleja que sea, NUCLEUS es GENERADOR DE PROGRAMAS, BASE DE DATOS Y GENERADOR DE INFORMES.

Toda la información es multi-intercambiable y de libre acceso por cualquiera de los demás programas. Así los datos de la base los condicionamos y utilizamos en el generador de programas y los imprimimos a través del generador de informes.

- GENERADOR DE PROGRAMAS EN MALLARD BASIC.
- CREACION DE BASES DE DATOS RELACIONALES.
- GENERADOR DE INFORMES.
- DISEÑADOR DE FORMATOS.
- DISEÑADOR DE PANTALLAS.
- CODIGO FUENTE DE LIBRE ACCESO Y LIBRE DE ERROR.
- DISEÑA SU PROPIO SISTEMA.
- MAILMERGE.

GENERADOR
DE PROGRAMAS

26.780 pts.

BRAINSTORM

La revolución del pensamiento, BRAINSTORM es un programa que piensa con Vd.

El compañero ideal para el empresario, director o cualquier persona que tenga que planificarse o tomar decisiones.

BRAINSTORM es la ayuda necesaria para su organización. El programa que se ha estandarizado en Inglaterra, tan necesario, útil y popular como una base de datos o un tratamiento de textos.

- ORGANIZA POR RANGOS.
- ACCESO DESCENDENTE POR-MENORIZADO.
- PLANIFICACION A NIVEL DIA.
- DECISIONES A LARGO PLAZO.
- REVISION DE PROBLEMAS.
- SIMULTANEIZACION DE TAREAS.
- PROCESO TOP/DOWN.

ORGANIZADOR
DE IDEAS

17.850 pts.

STARCOM

Piii... su ordenador le comunica: La revolución de las comunicaciones, de la mano de OFITES INFORMÁTICA, llega a España. El nuevo mundo de las comunicaciones digitales lo tiene a su disposición, las redes de transmisión electrónica digitalizada, con su PCW 8256 o PCW 8512a través de un interface RS 232-C con otros ordenadores, redes de transmisión de datos, etc..., Vd. podrá enviar o recibir ficheros de texto o de datos, ASCII, etc..., creados por NEWWORD y otros...

- TRANSICIONES DIRECTAS EN RED.
- COMPATIBILIDAD CON NEWWORD.
- POSIBILIDADES DE TRANSMISIONES VIA MODEM, RED TELEFONICA.
- COMUNICACION INSTANTANEA.

COMUNICACIONES

17.850 pts.

DE VENTA EN LOS MEJORES COMERCIOS DE INFORMÁTICA

Si Vd. tiene alguna dificultad para obtener los programas, puede dirigirse a:



Avda. Isabel II, 16 - 8º
Tels. 455544 - 455533
Telex 36698
20011 SAN SEBASTIAN

CONDICIONES ESPECIALES PARA DISTRIBUIDORES

DELE "MARCHA" A SUS GRAFICOS

Si ha seguido nuestros artículos anteriores no le han de resultar desconocidos cada una de los caracteres que su Amstrad tiene incorporados.



Seguro que ya le resulta muy fácil poder dibujar en un determinado punto de la pantalla cualquier símbolo de los que ya están definidos en la memoria del ordenador. Don Sonrisitas—o el CHR\$ (224)—resulta ser un viejo conocido suyo.

Pues bien, vamos a intentar dar vida a todos estos gráficos, o por lo menos a alguno de ellos. Será como crear una pequeña película de dibujos animados poniendo «en movimiento» a Don Sonrisitas o al «alienígena» de turno y convertir nuestro monitor en un «cuadro vivo».

Por si acaso se había olvidado de nuestra amigo, leedle:

PRINT CHR\$(224)

pulse la tecla RETURN y de nuevo tendremos en acción (de momento está parado) una cara que nos sonríe desde la pantalla. ¡Hola!

El Amstrad incluye su juego de caracteres

Conviene recordar que dentro del ordenador hay ya definidos 256 caracteres distintos. Son todos los CHR\$(NUMERO) a los que podemos acceder.

Cada uno de ellos está perfectamente localizado por un número internamente y que hemos llamado su «código ASCII».

Por ejemplo, nuestro Don Sonrisitas tiene el número 224 como código ASCII; la letra «O» mayúscula tiene el 79, o al «1» le han asociado al código número 49.

Pero esto no es todo. También existen unos caracteres de «control» que son un poco diferentes de los anteriores. Escriba ahora:

PRINT CHR\$(7)

y observe lo que ocurre. ¿Qué aparece en la pantalla? A primera vista parece que se

debia visualizar el símbolo del carácter asociado al número 7.

Pero sin embargo, no vemos nada. Solamente airesmos un píxel producido por el ordenador. ¿Cómo es esto?

Caracteres de control

Muy sencillo. El carácter correspondiente al código 7 es de control. No produce ningún símbolo cuando se le envía a la pantalla, sino que el Amstrad los entiende como unas órdenes que debe cumplir. En este caso le estamos diciendo que tiene que generar un píxel, por eso no visualiza nada. Hemos cambiado un efecto óptico por un sonoro: ¡Qué maravilla!

Y no es el único. Disponemos de varios de estos caracteres de control con los que podemos hacer avanzar la posición del cursor un lugar—CHR\$(9), asignar un color al borde de la pantalla—CHR\$(28), o borrar cualquier texto mediante un sencillo:

PRINT CHR\$(12)

Los códigos que tiene asignados son los comprendidos dentro del margen de 0 a 31 y le recomendamos que tenga mucho cuidado al utilizarlos ya que los resultados obtenidos podrían ser nefastos.

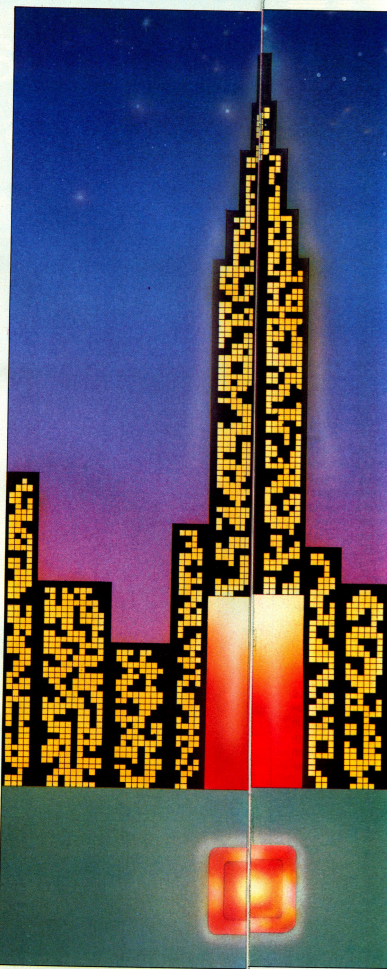
Tampoco debe resultarle nuevo o sonarle a chino si le decimos que no sólo podemos disponer de todos estos caracteres ya definidos sino que es posible crear, si nuestra «vena» artística da para tanto, todo un conjunto de ellos a nuestra medida.

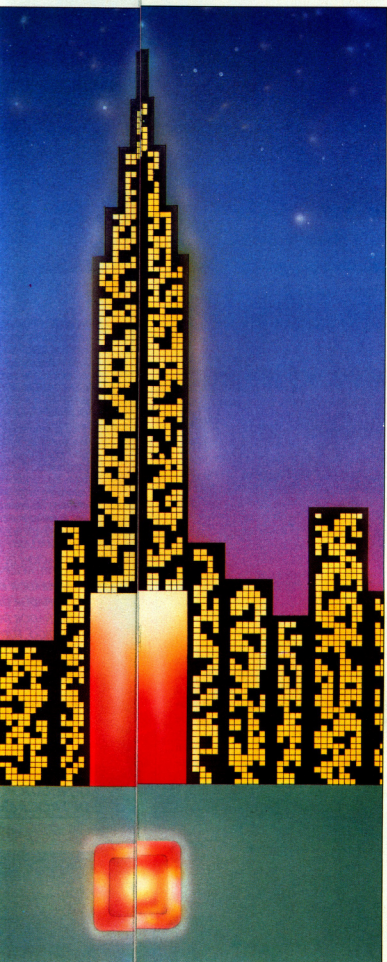
¿Se siente capaz de ir dando forma a cada uno de los gráficos que compondrán ese juego que no hemos visto todavía en ningún sitio? Va a ser el número «uno» en las listas de los «más maravillosos» y nosotros sus autores. ¡De acuerdo?

Y ya que hemos recordado, a vuelo de pluma y muy por encima, que poseemos un amplio grupo de símbolos para visualizar, pues vamos a ponerlos «en marcha».

Una vez elegido el carácter, recuerde que era:

PRINT CHR\$(224)





que podemos hacer para conseguir que se mueva por la pantalla? Póngase a pensarlo un momento y se sorprenderá de la cantidad de formas distintas que se le ocurren.

La más sencilla es colocar el cursor en diferentes puntos de la pantalla y en ellos imprimir el símbolo que queramos. ¿Se acuerda cómo se escribía en una determinada posición?

Sencillo. Colocábamos el cursor de textos en el lugar elegido por medio de una instrucción:

LOCATE x,y

donde «x» era la columna e «y» la fila de la parte de la pantalla donde queríamos visualizar un texto.

Y después sólo nos quedaba imprimir lo que quisiéramos. Teclee:

LOCATE 5,2:PRINT CHR\$(224)

y si no falla nada aparecerá Don Sonrisitas en el punto de cruce de la columna 5 y la fila 2. Bueno. Si vamos cambiando esta posición a lo largo y ancho de la pantalla parecerá que nuestro amigo se está moviendo.

Introduzca en la memoria de su ordenador el Programa I, procurando no equivocarse al teclearlo, y ¿que ocurrirá?

Programa uno

Antes de ejecutarle vamos a analizar despacio qué es lo que hace. En la línea 30 coloca el cursor de textos en la posición que tiene par coordenadas 1,1 y allí aparece nuestro carácter.

En la 50 hace lo mismo, pero en esta ocasión el punto elegido es el 3,6.

Las líneas 70 a 100 repiten el proceso situando e imprimiendo el símbolo que tiene código 224 en otros dos lugares diferentes dentro de la pantalla.

```
10 REM PROGRAMA I
20 CLR
30 LOCATE 1,1
40 PRINT CHR$(224)
50 LOCATE 3,6
60 PRINT CHR$(224)
70 LOCATE 4,5
80 PRINT CHR$(224)
90 LOCATE 18,20
100 PRINT CHR$(224)
```

Con esto no vamos a obtener ninguna sensación de movimiento ya que todos ellos van a permanecer en el lugar donde los hemos escrito. Busquemos la solución.

El fallo está, entonces, en que una vez que han aparecido en la pantalla se quedan allí. Si conseguimos hacerlos desaparecer después que los hemos visualizado, parece que nuestro problema se irá con ellos.

Y, ¡hay algo mejor para borrar un carácter que escribir un espacio en blanco encima de él! Intentemos hacerlo.

Basta con volver a situar el cursor en la posición donde aparece Don Sonrisitas y allí imprimir un espacio. Eche un vistazo al Programa II y verá en la práctica esta posible solución.

Primeros PASOS

Programa dos

Si lo ejecuta constatará que no lo hemos conseguido del todo. Lo hace con tanta rapidez que casi no podemos apreciar cómo se desliza por la pantalla.

El único resultado que podemos observar es que aparece fugazmente en la primera posición y casi inmediatamente va le encontramos en la última. Luego, tampoco conseguimos lo que queremos.

Piense ahora un poquito. ¿Cómo podríamos hacer que este movimiento fuera un poco más lento?

¿Se acuerda de lo que es un bucle retardador? Se trata sencillamente de un bucle FOR...NEXT en el que hemos suprimido las instrucciones que tendrían que repetirse. Hemos eliminado su cuerpo.

```
10 REM PROGRAMA II
20 CLR
30 LOCATE 1,1
40 PRINT CHR$(224)
50 LOCATE 1,1
60 PRINT CHR$(224)
70 LOCATE 3,6
80 PRINT CHR$(224)
90 LOCATE 3,6
100 PRINT " "
110 LOCATE 4,5
120 PRINT CHR$(224)
130 LOCATE 4,5
140 PRINT " "
150 LOCATE 18,20
160 PRINT CHR$(224)
```

Este bucle girará sobre sí mismo repitiéndose tantas veces como determine su variable de control. No hará nada, simplemente tardará un cierto tiempo en ejecutarse, tiempo que nos va a retrasar la marcha del programa.

Si colocamos un bucle de este tipo en el sitio adecuado, conseguiremos mantener en la pantalla el carácter durante un **eratio** un poco más largo para luego hacerlo desaparecer y visualizarlo en otro punto.

Pues, manos a la obra. Situemos este bucle inmediatamente detrás de cada instrucción PRINT y veamos lo que ocurre. En el Programa III así lo hemos hecho y los resultados ahora ya son otros.

Programa tres

En realidad colocamos una llamada a la subrutina de retardado (**GOSUB**) en las líneas que siguen a cada uno de los PRINTS (líneas 40, 70, 100, 130, 160, 190 y 220).

La subrutina está formada únicamente por el bucle de retardado —líneas 240 y 250— y la instrucción RETURN de la 260 cuyo misión es devolver control al programa principal exactamente a la línea o sentencia que va detrás de la llamada).

El resto del programa es exactamente igual que el anterior con la única diferencia del END que nos encontramos en la línea 230 y que hace que la ejecución, una vez que ha terminado de realizar lo que nosotros queremos, no continúe por la subrutina. Separa la misma del programa principal.

No podemos decir que ahora Don Sonrisitas se pase por la pantalla de un modo continuo y «armonioso», pero reconozca que poco a poco lo vamos ya consiguiendo. Intentemos mejorarlo.

¿Qué le parece si tratamos de crear un movimiento en sentido horizontal? La característica común a todos los puntos de una línea paralela al eje de coordenadas «X» es que están en la misma fila. O dicho en otras palabras, la coordenada «Y» es igual para todos.

```
10 REM PROGRAMA III
20 CLS
30 LOCATE 1,1
40 PRINT CHR$(224)
50 GOSUB 240
60 LOCATE 1,2
70 PRINT " "
90 GOSUB 240
90 LOCATE 3,6
100 PRINT CHR$(224)
110 GOSUB 240
120 LOCATE 3,6
130 PRINT " "
140 GOSUB 240
150 LOCATE 4,5
160 PRINT CHR$(224)
170 GOSUB 240
180 LOCATE 4,5
190 PRINT " "
200 GOSUB 240
210 LOCATE 18,20
220 PRINT CHR$(224)
230 END
240 FOR retardo=1 TO 500
250 NEXT retardo
260 RETURN
```

Entonces, si hacemos que la coordenada «X» cambie de valor actuando como variable de control de un bucle FOR... NEXT y una de las instrucciones del cuerpo del mismo es:

```
LOCATE X,12
```

quizá hasta consigamos que nuestro símbolo se mueva en sentido horizontal. Con ese fin hemos escrito el Programa IV. Tecléalo y ja ver que pasa!

Programa cuatro

En cada vuelta del bucle vamos a situar el cursor en la posición fijada por la instrucción LOCATE X,12 —líneas 60 y 100. Pero, por ser «X» la variable de control, su valor se va a ir incrementando en una unidad cada vez que lo repetimos.

Y, ¿entre qué valores va a oscilar? A esta pregunta nos contestará la línea 30: 1 a 39.

Con esto correremos todos los puntos de la línea excepto el último de la derecha, que estará situado en la columna 40.

El último punto donde situamos el cursor dentro del bucle es el 39,12. ¿de acuerdo?

```
10 REM PROGRAMA IV
20 CLS
30 FOR X=1 TO 39
40 LOCATE X,12
50 PRINT CHR$(224)
60 GOSUB 130
70 LOCATE X,12
80 PRINT " "
90 NEXT X
100 LOCATE X,12
110 PRINT CHR$(224)
120 END
130 FOR retardo=1 TO 500
140 NEXT retardo
150 RETURN
```

Allí escribimos primero nuestro carácter y después un espacio en blanco, o sea, lo borramos. Es decir, que si no hubiéramos añadido las líneas 100 y 110 a este Programa no nos quedaría ningún resto de su ejecución en la pantalla.

Con ellas lo único que conseguimos es situar el símbolo elegido en la última columna de la fila 12 y dejarlo allí escrito definitivamente.

El valor contenido en la variable «X» en la línea 100 será el que tenga a la salida del FOR... NEXT. ¿Cuándo dejamos de repetir las instrucciones contenidas dentro de este bucle? En el momento que la variable de control se salga de los límites establecidos: 1 y 39.

Como el paso (o STEP) es 1, el primer valor fuera de rango será el 40, así que la línea 100 nos colocará el cursor de textos en la posición 40,12, que es la que nosotros queríamos.

No pase por alto la llamada a la subrutina de «retardo» que hemos colocado entre las instrucciones de escritura y borrado —línea 60.

Con ella conseguimos, como antes dijimos, que el movimiento sea un poco más lento y por lo menos podamos apreciarlo. Quite del Programa dicha línea y verá cómo nuestro carácter se desplaza mucho más rápido por la pantalla.

Si nos lo permite vémos a sugerirle una pequeña idea. ¿Por qué no intenta cambiar el programa para que usted mismo pueda determinar la velocidad del movimiento?

Añada la línea 25:

```
25 INPUT "VELOCIDAD DE 1 A 10":
    velocidad
```

y cambie la 130 por:

```
130 FOR retardo=1 TO 500—50*velocidad
    y lo habremos conseguido. Cuanto más pequeño sea dicho límite superior, el bucle girará menos veces, el retardo será menor y por tanto el móvil se desplazará más rápidamente.
```

Otra forma de hacer que «algo» se mueva horizontalmente por la pantalla sin tener que escribir un espacio encima del carácter cada vez, es imprimir uno de los dos símbolos a la vez y uno a continuación del otro.

Programa cinco

Analícemos su labor con un determinado contenido de «X», por ejemplo el 5.

Cuando la variable «X» vale 5, el cursor de textos se sitúa en la posición 5,12 —columna

5, fila 12— y allí escribe un espacio en blanco. El punto y como existente nos está diciendo que a continuación y totalmente «pegado» a él el coloque a Don Sonrisitas. ¿En qué posición lo hará?

La respuesta es bien sencilla: en la misma fila pero una columna más a la derecha, o sea, en el punto 6,12.

Pasemos a la siguiente vuelta del bucle. Ahora la variable del control «X» tendrá un valor igual a 6. Por tanto, la instrucción LOCATE de la línea 40 colocará el cursor en el punto de coordenadas 6,12, que es donde estaba escrito nuestro CHR\$(224) con la anterior pasada del bucle.

```
10 REM PROGRAMA V
20 CLS
30 FOR X=1 TO 39
40 LOCATE X,12
50 PRINT " ";CHR$(224)
60 GOSUB 90
70 NEXT X
80 END
90 FOR retardo=1 TO 500
100 NEXT retardo
110 RETURN
```

Y allí justamente imprime un espacio en blanco, borrando lo que anteriormente se estuviera visualizando en ese punto. ¿Comprende?

A continuación coloca el carácter que hemos elegido en la siguiente columna y continúa repitiendo el bucle.

Una advertencia. Hay que tener mucho cuidado a la hora de componer la línea que contenga la instrucción PRINT. La que hemos usado en el Programa V solamente vale para este caso, cuando el objeto se mueve en una línea recta horizontal y de izquierda a derecha. Para otro diferente habría que analizar la manera de hacerlo, no lo olvide.

Programa seis

Vamos a emplear, como dijimos antes, «bucle» como variable de control del FOR... NEXT formado por las líneas 30 y 110.

En la 40 y la 50 obtenemos los valores que va a tener el punto donde vamos a escribir el carácter en cada repetición.

A continuación situaremos el cursor —línea 60— y allí visualizaremos «la sonrisa móvil» para, después de llamar a la rutina de borrado, volver a colocar allí el cursor y escribir ahora en el mismo punto un espacio en blanco que nos borra lo que anteriormente hubiera escrito —línea 100.

```
10 REM PROGRAMA VI
20 CLS
30 FOR bucle=1 TO 20
40 X=bucle+5
50 Y=bucle
60 LOCATE X,Y
70 PRINT CHR$(224)
80 GOSUB 150
90 LOCATE X,Y
100 PRINT " "
110 NEXT bucle
120 LOCATE X,Y
130 PRINT CHR$(224)
140 END
150 FOR retardo=1 TO 500
160 NEXT retardo
170 RETURN
```

ERBE *dades*

SABOTEUR

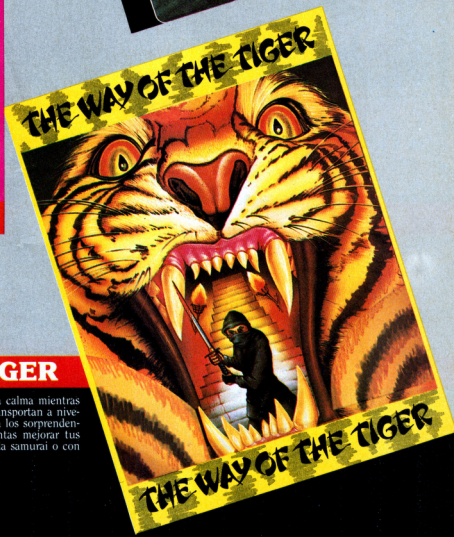
Como experimentado mercenario cuidadosamente entrenado en las artes marciales, debes cumplir la misión que te ha sido encomendada: robar el disco que con la información de los rebeldes tiene el Gran Dictador.



El primer juego de lucha libre hecho para ordenador. Más de 25 movimientos diferentes te permitirán hacer todo tipo de llaves: desde la sujeción de espaldas hasta la voltereta de hombros, pasando por los mismos programadores del legendario "Exploding Fist".

THE WAY OF THE TIGER

Entra en el mudo de los samurais. Mantén la calma mientras el movimiento y las rutinas de combate te transportan a niveles que nunca pensaste posibles. Experimenta los sorprendentes efectos del "Triple Scroll" mientras intentas mejorar tus técnicas de lucha cuerpo a cuerpo, con espada samurai o con mil posibilidades más.



Para que juegues ERBE

PING-PONG

La gran sorpresa. Gráficos increíbles, movimiento super-rápido, podrás efectuar las mismas jugadas que si tuvieras la paleta en tu mano.

Botes, rebotes, efectos, dejadas, saques, cortadas, mates... todo es posible con esta maravilla llamada "Ping-Pong"



AMSTRAD
(Schneider)
464
664
6128

KNIGHT GAMES

Un desafío medieval en tu ordenador. Transportáate a la Edad Media y conviértete en caballero de la Mesa Redonda demostrando tus habilidades en el torneo. Lucha a espada, ballesta, lanza, mazas, arco, hachas y con todas las armas propias de aquella fantástica época.

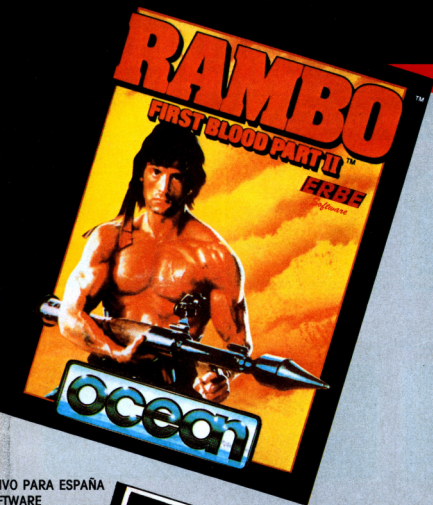
FRANKIE GOES TO HOLLYWOOD

El juego de los juegos. El que fuera numero 1 indiscutible en Spectrum y Commodore, ahora disponible para Amstrad. Más de 10 juegos diferentes se encierran en "Frankie goes to Hollywood", el programa más original que hayas visto.



DISTRIBUIDOR EXCLUSIVO
ERBE S.A.
C/ STA. ENGRACIADA, 10
28010 MADRID
TFNO. (91) 411 11 11

DELEGACION
AVDA. MISTRIAL, 10
TFNO. (93) 411 11 11

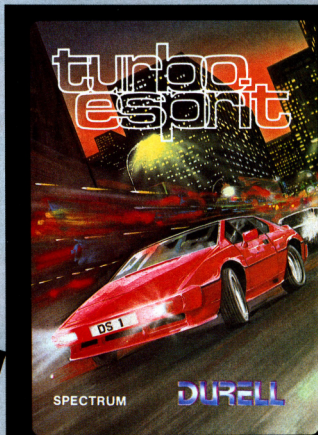
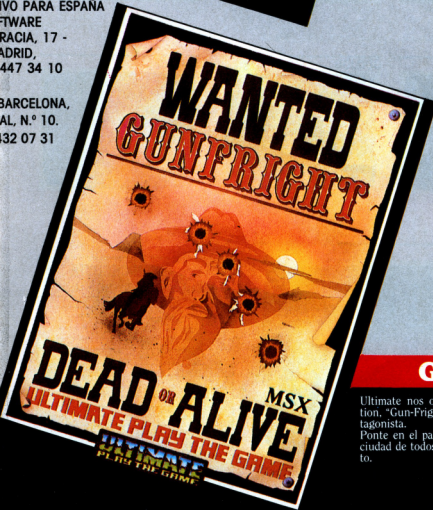


RAMBO

Toda la emoción del film, en tu ordenador. Siéntete como John Rambo en la jungla vietnamita e intenta salvar a tus compañeros prisioneros en el campo de concentración.

IVO PARA ESPAÑA
FTWARE
RACIA, 17 -
ADRID,
447 34 10

BARCELONA,
AL, N.º 10.
32 07 31



TURBO ESPRIT

Tu misión: vigilar y cuidar el cumplimiento de la ley que se ve amenazada por una terrible banda de delincuentes que han hecho del tráfico de narcóticos su negocio más rentable. Tus medios: un Lotus Turbo Sprit dotado de uno de los máximos adelantos técnicos y con el que deberás patrullar por calles y avenidas.

GUN-FRIGHT

Ultimate nos ofrece para Amstrad, usando su técnica Filmation, "Gun-Fright", el juego en el que el lejano oeste es el protagonista.

Ponte en el papel de Quikdraw, el sheriff que piensa librar la ciudad de todos los pistoleros a lomos de su buen caballo Panto.

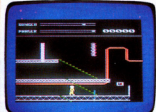
MicroByte

PRESENTA...

AMSTRAD

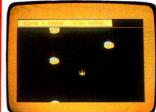
NUEVOS PROGRAMAS EN CASSETTE Y DISCO

ARGO NAVIS



El comandante de nave AMSTRAD-1 se encuentra atrapado en las profundidades de una central nuclear y debe salir con vida. Excelentes gráficos y sonido. P.V.P.: CASSETTE 2.200 pts. DISCO 2.900 pts.

ROCK RAID



Debes pilotar con acierto la nave que a lo largo de su viaje galáctico sufrirá encuentros con meteoritos, residuos planetarios, etc. Gran movilidad y excelentes efectos. P.V.P.: CASSETTE 1.900 pts. DISCO 2.600 pts.

WIZARD'S LAIR



Te encuentras atrapado en las profundidades de una caverna, llena de obstáculos, adversidades, etc. ¿Serás capaz de salir con vida? P.V.P.: CASSETTE 1.900 pts. DISCO 2.600 pts.

MACADAM FLIPPER



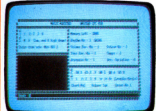
Al activo programa que nos trasladó al mundo de la máquina flipper del mejor casino de Las Vegas. Posibilidad de creación del tablero, puntuaciones, etc. P.V.P.: CASSETTE 2.200 pts. DISCO 2.900 pts.

JUMP JET



Te encuentras a los mandos de la nave "Air craft". En una perfecta maniobra debes despegar del portaviones. (Excelente versión simulador vuelo-combate). P.V.P.: CASSETTE 2.200 pts. DISCO 2.900 pts.

MUSIC MAESTRO



El más completo programa de música creado para el AMSTRAD. Permite crear sonidos, melodías y convertir tu ordenador en la mejor "caja de música". P.V.P.: CASSETTE 2.200 pts. DISCO 2.900 pts.

PAZAZZ



Programa que permite de una manera sana la creación de partituras con gráficos, dotadas de movimiento, acompañadas de música. P.V.P.: DISCO 2.900 pts.

SYCLONE 2



Programa de utilidad que permite realizar copias de seguridad (back-ups) a distintas velocidades (baudios). P.V.P.: CASSETTE 1.800 pts. DISCO 2.500 pts.

ZEDIS II



Editor-desensamblador del Z-80, para el programador más avanzado. P.V.P.: CASSETTE 1.900 pts. DISCO 2.600 pts.

SYSTEM X



Ampliación del lenguaje Basic. Conjunto de 30 nuevas instrucciones (fill, circle, protect) para ayudar en la programación. P.V.P.: CASSETTE 2.200 pts. DISCO 2.900 pts.

ODDJOB



La mejor utilidad para el mejor conocimiento del disco. (Copias de disco, Disk map, Disk track, sector, etc.). P.V.P.: DISCO 2.600 pts.

TRANSMAT

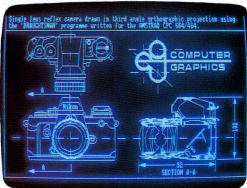


Pasar los mejores programas de cinta a disco ya no es problema. Con Transmat este proceso será fácil y sencillo. P.V.P.: DISCO 2.600 pts.

OTROS PROGRAMAS EN STOCK

- MINI OFFICE P.V.P. CASS. 3.200 pts.
P.V.P. DIS. 3.900 pts.
- WORLD CUP FOOTBALL P.V.P. CASS. 1.800 pts.
- BATTLE FOR MIDWAY P.V.P. CASS. 1.800 pts.
- FIGHTER PILOT P.V.P. CASS. 2.200 pts.
- SURVIVOR P.V.P. CASS. 1.800 pts.
- MOON BUGGY P.V.P. CASS. 1.800 pts.
- TECHNICIAN TED P.V.P. CASS. 1.800 pts.
- FRUITY FRANK P.V.P. CASS. 1.800 pts.
- DATABASE P.V.P. CASS. 2.100 pts.
- LOGO TURTLE GRAPHICS P.V.P. CASS. 2.400 pts.
- TASCOPY Y TASPRIPT P.V.P. CASS. 2.600 pts.
- FONT EDITOR P.V.P. CASS. 1.900 pts.

DRAUGHTSMAN



Sofisticado programa de dibujo que permite tratar la pantalla del AMSTRAD como un sencillo tablero de dibujo, sus resultados son espectaculares. P.V.P.: CASSETTE 4.500 pts. DISCO 5.200 pts.

ENVÍOS A MICROBYTE ^{AS.}
P.º Castellana, 179, 1.º - 28046 Madrid

Nombre _____
Apellidos _____
Dirección _____
Población _____
D.P. _____ Teléfono _____

ENVÍOS GRATIS

JUEGO	C	D	Precio	TOTAL

PRECIO TOTAL PESETAS

Incluyo salm nominativa
Contra-Reembolso

Pedidos por teléfono 91 - 442 54 33 / 44

De todos es conocido que una parte importante de las matemáticas es la representación de funciones. Realizarla sería sencillo si no se presentaran algunas muy complejas, y por consiguiente los «malditos» problemas a los que todos tememos (que se lo pregunten a los estudiantes).

P

ero todo está solucionado! ya que ANALISIS presenta un programa que dibuja funciones de todo tipo.

Veámosle:

10-40 REMs que informan de su título.

50 Limpia la pantalla y crea un origen en la coordenada 0,0.

60 Se crea la ventana #1 con el comando WINDOW, situada y dimensionada según los parámetros indicados.

70 REM que nos indica la definición de la función.

80 Con el comando DEF FN se define la función a representar.

90 REM que nos dice que comienza la introducción de intervalos.

100-130 Pregunta los valores del intervalo y les asigna las variables «mínimo» y «máximo». Las líneas 110 y 120 borran los mensajes aparecidos en la ventana #1.

140 Especifica con la orden IF... THEN que si los valores del intervalo son menores que -16, mayores 16 o que si el valor mínimo es mayor que el valor máximo del intervalo, la ejecución del programa vuelve a la línea 100.

150 REM que informa del trazado del eje «X».

160 Comienza un bucle FOR... NEXT cuya variable de control es «x» y que interacciona 639 veces.

170 Dibuja puntos en las coordenadas x,200.

180 Define la variable «margen» como x/20.

190 Si la variable «margen» es igual su valor entero, INT (margin), se traza una línea desde el punto x,200 hasta el punto indicado por los incrementos.

200 Finaliza el ciclo FOR... NEXT.

210 REM que indica el trazado del eje «Y».

220 Se crea un nuevo ciclo FOR... NEXT cuya variable es «y» y que rota 399 veces.

230 Dibuja puntos en las coordenadas 320,y.

240 Asigna a la variable «margen» el valor y/20.

250 Si la variable «margen» es entera, se dibuja una línea con los incrementos indicados.

260 Acaba el ciclo FOR... NEXT.

270 REM que informa del comienzo del dibujo de la función.

280 Se crea un origen en las coordenadas 320,200.

290 Nuevo ciclo FOR... NEXT cuya variable es «x» y que rota un número de veces indicado por los valores del intervalo, pero incrementándose 0,01 en cada iteración. Este incremento es tan bajo para una mayor definición.

300 Con el comando FN se asigna a la variable «y» el valor de la función dada.

310 Se definen las variables «px» y «py».

320 Si la variable «py» pasa de los valores indicados, la ejecución del programa salta a la línea 340.

330 Pinta puntos en las coordenadas «px» y «py».

340 Finaliza el bucle FOR... NEXT.

350 Se detiene la ejecución del programa hasta que se pulse una tecla y luego salta a la línea 50.

NOTA DEL AUTOR: El programa realiza el dibujo de una función que ya viene incorporada. Para hacer la gráfica de cualquier otra, introduzca en la línea 80 la que desee representar.



```

10 REM *****
20 REM MICROHOBBY AMSTRAD
30 REM DIBUJO DE FUNCIONES
40 REM *****
50 CLS:ORIGIN 0,0
60 WINDOW #1,1,40,25,25
70 REM DEFINICION DE LA FUNCIO
N
80 DEF FN y(x)=3/(x^2-4)
90 REM INTRODUCCION DE INTERVAL
LOS
100 INPUT #1,"VALOR MINIMO DEL
INTERVALO ",minimo
110 CLS #1
120 INPUT #1,"VALOR MAXIMO DEL
INTERVALO ",maximo
130 CLS #1
140 IF minimo<-16 OR maximo>16
OR minimo>maximo THEN GOTO 10
150 REM TRAZADO EJE X
160 FOR x=1 TO 639
170 PLOT x,200
180 margen=y/20
190 IF margen=INT(margen) THEN
DRAWR 0,x:DRAWR 0,-8:DRAWR 0,
4
200 NEXT x
210 REM TRAZADO EJE Y
220 FOR y=1 TO 399
230 PLOT 320,y
240 margen=y/20
250 IF margen=INT(margen) THEN
DRAWR 2,0:DRAWR -4,0:DRAWR 2,
0
260 NEXT y
270 REM DIBUJO DE LA FUNCION
280 ORIGIN 320,200
290 FOR x=minimo TO maximo STE
P 0,01
300 y=FN y(x)
310 px=x:py=y:#20
320 IF py>199 OR py<-200 THEN
GOTO 340
330 PLOT px,py
340 NEXT x
350 CALL @BBO@:GOTO 50

```

TU PROGRAMA DE RADIO

claro!



A 11/80/2

- Entrevistas a fondo
- Exitos en Soft
- Noticias en Hard
- Concursos

Programátele: Sábados tarde de 5 a 7 horas.
En directo y con tu participación.

LA COPE A TOPE.

— RADIO POPULAR 54 EMISORAS O.M. —

En Barcelona Radio Miramar



**ATREVETE A ENFRENTAR EL COLOSSUS
A CUALQUIER OTRO JUEGO DE AJEDREZ**

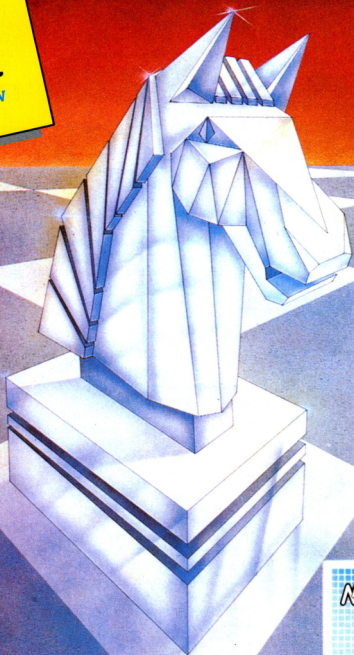
COLOSSUS 4 CHESS

AMSTRAD

cassette
2.500 pts.

disco
3.600 pts.


INSTRUCCIONES EN
ESPAÑOL



SERMA

MICROPOOL

*International
Limited*

RECORTA Y ENVIA ESTE CUPON A:  SERMA, C/ BRAVO MURILLO, N.º 377 - 3.º A. 28020 MADRID. TELEFONOS: 733 73 11 - 733 74 64.

CANTIDAD _____ SISTEMA _____ NOMBRE Y APELLIDOS: _____
DIRECCION: _____ POBLACION: _____
PROVINCIA: _____ CODIGO POSTAL: _____ FORMA DE PAGO: ENVIO TALON BANCARIO CONTRA REEMBOLSO

TODOS LOS PRODUCTOS EDITADOS POR EL GRUPO MICROPOOL SON UNA EXCLUSIVA DE SERMA.

TABLA DE SUBRUTINAS

10-480	Programa principal Dimensiona matriz Variable clave Título planilla Manejo de errores (lee) Opciones (manejo de subrutinas)
490-810	Dibuja pantalla principal Dimensiona Escribe subtítulos Recupera archivo (Si lo hay)
820-970	Imprime los mensajes del sistema
980-1050	(Puede suprimirse el sonido)
1060-1200	Scroll planilla abajo
1210-1340	Scroll planilla arriba
1350-1490	Scroll planilla derecha
1500-1640	Scroll planilla izquierda
1650-1940	Revisa si la celda está protegida Pide clave de entrada (si lo está) Entrada etiquetas Entrada variables Entrada de fórmulas
1950-2100	Establece el modo principal
2110-2160	Borrado de celda o sector de planilla
2170-2370	Cálculo de fórmula general
2380-2790	Imprime etiqueta en pantalla después de un cálculo
2800-2850	Imprime variable en pantalla después de un cálculo
2860-2910	Protege celda
2920-3030	Ventanas de ayuda (se puede suprimir si se requiere más memoria)
3040-3410	Reinicio de planilla Salida
3420-3470	Cálculo fórmulas particulares
3480-3530	Transfiere planilla a disco
3540-4050	Mueve cursor por planilla arriba
4060-4310	Abajo
4320-4380	Derecha
4390-4440	Izquierda
4450-4500	Rutinas de apoyo a movimiento cursor
4510-4560	Cálculo plantilla completa
4570-4610	
4620-4810	

1.2 Al volver de esta opción es necesario mover la «hoja» (planilla) hacia algún costado para que se escriban las variables en pantalla.

T Transferecia

Transfiere el disco la planilla en memoria.

1.1 La opción pide las líneas que se traspanan y las columnas.

1.2 Nombre (máximo 8/c).

1.3 Pide confirmación.

— Positiva, graba y vuelve a planilla.

— Negativa, vuelve a planilla.

S Salida

Salida del programa, borra el programa de la memoria del ordenador.

1.1 Pide confirmación.

CONTROL + CURSOR

Mueve cursor por planilla.

CURSOR

Mueve planilla.

Importante: Para mover la planilla es necesario que el cursor esté en el borde derecho, izquierdo, superior o inferior dependiendo de la dirección en que se quiera mover la planilla.

CONTROL + C

Recalcula toda la planilla

Importante: Es necesario que el cursor esté en la celda «AA» para ocupar esta opción.

— Visualiza los nuevos resultados en caso de producirse, en sus respectivas celdas.

Nota: Al cambiar la celda «D6» en el archivo «ejemplo» que acompaña junto al programa y usar la opción de «recalcular» produce el cambio de todas las celdas con sus nuevos valores.

Fórmulas y operaciones matemáticas

Las fórmulas pueden ser introducidas de dos formas.

1. Fórmulas generales.
2. Fórmulas particulares.

Fórmulas particulares

Estas fórmulas sólo pueden contener 1 solo operador y sólo de los siguientes.

- 1.—SEN.LC (seno de celda LC)
- 2.—COS.LC (coseno de celda LC)
- 3.—TAN.LC (tangente de celda LC)
- 4.—LOG.LC (LOG (base 10)) de celda LC)
- 5.—LON.LC (LON (natural)) de celda LC)
- 6.—PII.LC (producto de π por celda LC)
- 7.—RAZ.LC (? de celda LC)
- 8.—SUM.LC.LC2 (Sumatoria entre LC y LC2)
- 9.—PRO.LC.LC2 (Promedia entre LC y LC2)
- 10.—REP.LC.(xx) ó (yy) (copia repetitiva de LC)*

* Ver fórmulas generales.

Explicación

8: Suma una columna o línea o parte de ella pero siempre en forma vertical u horizontal (nunca en diagonal).

Ejemplo:

SUM.AB.HB

Suma la columna B desde A hasta H.

9: Promedia una columna o línea o parte de ellas pero siempre en forma vertical u horizontal (nunca en diagonal).

Ejemplo:

PRO.HC.HF

Promedia la línea H desde la columna C hasta F.

10: Copia una fórmula general de una celda en otra.

Ejemplo:

1.—REP.LC.XX

Repite celda LC con sentido horizontal.

2.REP.LC.YY

Repite celda LC con sentido vertical.

```

10 REM AMSALC
20 REM Patrici Lagos M./c/Valenci
a 235 2,27-08007),Barcelona
30 REM *** PROGRAMA PRINCIPAL ***
40 ON BREAK:CONT
50 MODE 2:CLERAR:CONT=0:mp=V
:=":i=":c=":cl=":i=":s=":c=":1
60 DIM matriz$(27,27),r$(12)
70 GOSUB 490:GOSUB 130:GOSUB 820
80 IF flag=1 THEN GOSUB 800:GOTO 18
0
90 frs="" / TITULO : ?:"GOSUB 980
100 LOCATE #3,1:INPUT #3,TITULO
110 IF LEN(TITULO)>60 THEN 90
120 GOSUB 800:GOTO 180
130 frs="" / ESPERE UN MOMENTO POR F
AVOR:GOSUB 980
140 FOR i=1 TO 26
150 FOR j=1 TO 26
160 mat:=(i,j)="DEN B
170 NEXT j:NEXT i:RETURN
180 frs="" / COMANDO : ?:"CLS #3:PRI
NT #3,frs:ee=0:flag=0
190 as=INKEY$:IF as="" THEN 190
200 as=LOWER$(as)
210 IF ASC(as)=248 THEN GOSUB 4320
220 IF ASC(as)=249 THEN GOSUB 4390
230 IF ASC(as)=251 THEN GOSUB 4450
240 IF ASC(as)=250 THEN GOSUB 4510
250 IF ASC(as)=240 THEN GOSUB 1210
260 IF ASC(as)=241 THEN GOSUB 1060
270 IF ASC(as)=242 THEN GOSUB 1500
280 IF ASC(as)=243 THEN GOSUB 1350
290 IF ASC(as)=101 THEN GOSUB 1650
300 IF ASC(as)=109 THEN GOSUB 2110
310 IF ASC(as)=112 THEN GOSUB 2920
320 IF ASC(as)=124 THEN GOSUB 2170
330 IF ASC(as)=114 THEN GOSUB 3430
340 IF ASC(as)=97 THEN GOSUB 3040
350 IF ASC(as)=116 THEN GOSUB 4060
360 IF ASC(as)=115 THEN GOSUB 3490
370 IF ASC(as)=3 THEN GOSUB 4620
380 IF ee=1 THEN 390
390 IF ee=5 THEN frs="" / OPERANDO C
ON ETIQUETAS 1
400 IF ee=5 THEN frs="" / OPERADOR I
NEXISTENTE 1
410 IF ee=7 THEN frs="" / CELDA INE
XISTENTE EN FORMULA 11
420 IF ee=7 THEN frs="" / INTENTO D
ENTRADA ILEGAL 11
430 IF ee=8 THEN frs="" / FORMULA M
Y LARGA 11
440 IF ee=9 THEN frs="" / CELDA SIN
CARACTER DE PROTECCION 11
450 IF ee=10 THEN frs="" / CELDA CO
N CARACTER DE PROTECCION MALO 11
460 IF ee=13 THEN frs="" / CELDA FRD
TEGIDA 11
470 GOSUB 980:FOR i=2000 TO 0 STEP
-1:NEXT
480 GOTO 180
490 REM ***** PANTALLA PRINCIPAL ***
*****
500 IN# 0,0:IN# 1,26: BORDER 0:GRAPH
ICS PEN #1:CLG
510 MOVE 3,7: DRAW 599,7: DRAW 599,36
: DRAW 2,26: DRAW 599,26
520 MOVE 3,42: DRAW 191,42: DRAW 91,84
: DRAW 3,84: DRAW 3,42
530 MOVE 19,103: DRAW 635,103: DRAW 6
35,308: DRAW 19,308: DRAW 19,103
540 MOVE 499,45: DRAW 587,45: DRAW 58
7,100: DRAW 499,100: DRAW 499,45
550 MOVE 599,103: DRAW 599,42: DRAW 9
,42: DRAW 99,103
560 MOVE 99,308: DRAW 99,314
570 MOVE 99,314: DRAW 599,378: DRAW 59
9,378: DRAW 599,340
580 MOVE 599,314: DRAW 599,308
590 MOVE 19,314: DRAW 635,314: DRAW 6
35,340: DRAW 19,340: DRAW 19,314
600 MOVE 91,103: DRAW 91,90: DRAW 3,9
0: DRAW 3,351: DRAW 91,351: DRAW 91,34
0
610 MOVE 91,314: DRAW 91,308
620 MOVE 19,104: DRAW 633,104: DRAW 6
33,307
630 MOVE 0,0: DRAW 639,0: DRAW 639,38
0: DRAW 0,380: DRAW 0,0
635 MOVE 620,10: FILL 1
640 WINDOW #1,4,79,12,18: WINDOW #1,
4,79,5,5: WINDOW #2,2,7,19
650 WINDOW #3,16,61,20,22: WINDOW #4
,2,11,21,22: WINDOW #5,4,79,7,18
660 WINDOW #6,14,74,3,3: WINDOW #7,1
,80,1,25
740 LOCATE #3,1:CLS #3
680 FOR i=1 TO 61:LOCATE #7,13+i,3:
PRINT #7,CHR$(207):NEXT
690 PEN #7,0:PAPER #7,1:LOCATE #7,2
,24:PRINT #7,"E":LOCATE #7,16,24:PR
INT #7,"M":LOCATE #7,21,24:PRINT #7
,"P":LOCATE #7,30,24:PRINT #7,"B"
700 LOCATE #7,37,24:PRINT #7,"R":LO
CATE #7,46,24:PRINT #7,"A":LOCATE #
7,52,24:PRINT #7,"T":LOCATE #7,68,2
4:PRINT #7,"SALIDA"
710 PEN #7,1:PAPER #7,0:LOCATE #7,3
,24:PRINT #7,"trada datos":LOCATE
#7,17,24:PRINT #7,"odo":LOCATE #7,2
,24:PRINT #7,"roteger"
720 LOCATE #7,31,24:PRINT #7,"orrar"
:LOCATE #7,38,24:PRINT #7,"eincia"
:LOCATE #7,47,24:PRINT #7,"yuda":L
OCATE #7,53,24:PRINT #7,"ransferenc
ia"
730 LOCATE #7,65,20:PRINT #7,CHR$(2
40):"CHR$(241):"CHR$(242):"
:CHR$(243):
740 LOCATE #7,64,21:PRINT #7,"MOVIM
IENTO":LOCATE #7,64,22:PRINT #7,"FO
R CURSOR":
750 PAPER #7,1:LOCATE #7,0:LOCATE #7,2
,1:PRINT #7,"***HOJA DE CALCULO#7,2
MENCIONAR MATRIZ DE 26 POR 26 ELEMEN
TOS***C:PIH-SDET***"
760 PAPER #7,1:FOR i=1 TO 10
12:LOCATE #7,2,1+i:PRINT #7,CHR$(64
+i):NEXT i
770 PAPER #1,0:PEN #1,1:PRINT #1,"A
B C D
E F G"
780 PAPER #4,0:PEN #4,1:PRINT #4,"M
O D 0"
790 RETURN
800 LOCATE #6,INT(62-LEN(TITULO$)
/2),1:PRINT #6,UPPER$(TITULO$)
810 RETURN
820 REM ***** INICIO *****
830 frs="" / RECUPERAR ARCHIVO s/n ?
:GOSUB 980
840 re=INKEY$:IF re="" THEN 840
850 IF re="s" OR re="S" THEN GOSU
B 870
860 RETURN
870 frs="" / NOMBRE DEL ARCHIVO M/B
:GOSUB 980
880 LOCATE #3,1:INPUT #3,nom$:IF no
m="" THEN 860
890 ar=LEN(nom$):IF lar>8 THEN frs
="" / NOMBRE MUY LARGO 1:"GOSUB 980:
GOTO 870
900 OPENIN nom$
910 INPUT #9,titulo$,a,b,c,d
920 FOR i=a TO b
930 FOR j=c TO d
940 INPUT #9,matriz$(i,j)
950 NEXT j:NEXT i:CLOSEIN
960 frs="" / ARCHIVO CARGADO:GOSUB
980:FOR i=200 TO 0 STEP -1:NEXT i:IF
1=1
970 RETURN
980 REM ***** IMPRESION INTERNA *****
990 PAPER #3,1:PEN #3,0:CLS #3:LOCA
TE #3,1,1
1000 LOC i=1 TO LEN(fr$)
1010 SOUND 1,65,3
1020 FOR j=20 TO 0 STEP -1:NEXT URN
1030 PRINT #3,MID$(fr$,j,1)
1040 NEXT i
1050 RETURN
1060 REM ***** SCROLL ABAJO *****
1070 IF c<12 THEN RETURN
1080 IF x<2 THEN RETURN
1090 LOCATE #2,1,12:PRINT #2," "
1100 LOCATE #2,1,1:PRINT #2,CHR$(11
)
1110 LOCATE #2,1,1:PRINT #2,CHR$(x+
63)
1120 LOCATE #5,1,1:PRINT #5,CHR$(11
)
1130 FOR i=0 TO 6
1140 ss=MID$(matriz$(x-1,y+1),2,1)
1150 mm=MID$(matriz$(x-1,y+1),4,9)
1160 LOCATE #5,1,11+i,1
1170 SOUND 4600
1180 NEXT i
1190 ss=Ilyy
1200 RETURN
1210 REM ***** SCROLL ARRIBA *****
1220 IF c<13 THEN RETURN
1230 IF x=14 THEN RETURN
1240 LOCATE #2,1,13:PRINT #2,CHR$(1
)
1250 LOCATE #2,1,12:PRINT #2,CHR$(x
+76)
1260 LOCATE #5,1,12:PRINT #5,CHR$(1
)
1270 FOR i=0 TO 6
1280 ss=MID$(matriz$(x+1,y+1),2,1)
1290 mm=MID$(matriz$(x+1,y+1),4,9)
1300 LOCATE #5,1,11+i,1
1310 SOUND 4600
1320 NEXT i
1330 ss=Ilyy
1340 RETURN
1350 REM ***** SCROLL DERECHA *****
1360 IF cc<6 THEN RETURN
1370 IF y<2 THEN RETURN
1380 FOR i=0 TO 6
1390 LOCATE #1,1,11+i,1:PRINT #1,CH
R$(y+1+i*64)
1400 NEXT
1410 FOR i=0 TO 6
1420 FOR j=0 TO 11
1430 ss=MID$(matriz$(x+i,y+1),2,1)
1440 mm=MID$(matriz$(x+i,y+1),4,9)
1450 LOCATE #5,1,11+i,j+1
1460 SOUND 4600
1470 NEXT: NEXT
1480 yy=Ilyx
1490 RETURN
1500 REM ***** SCROLL IZQUIERDA *****
1510 IF cc<0 THEN RETURN
1520 IF y>19 THEN RETURN
1530 FOR i=6 TO 0 STEP -1
1540 LOCATE #1,1,11+i,1:PRINT #1,CH
R$(y+1+i*64)
1550 NEXT
1560 FOR i=6 TO 0 STEP -1
1570 FOR j=0 TO 11
1580 ss=MID$(matriz$(x+j,y+1),2,1)
1590 mm=MID$(matriz$(x+j,y+1),4,9)
1600 LOCATE #5,1,11+i,j+1
1610 SOUND 4600
1620 NEXT: NEXT
1630 yy=Ilyx
1640 RETURN
1650 REM ***** INTRO.DATOS *****
1660 pro=LEFT$(matriz$(lin,col),1)
1670 IF pro="P" THEN frs="" / CELDA
PROTEGIDA 0 ESPERE...:GOSUB 980:FO
R i=1000 TO 0 STEP -1:NEXT i:GOSUB
1710:RETURN
1680 IF pro="D" THEN GOSUB 1750:RE
TURN
1690 IF pro="=" THEN ee=9:RETURN
1700 ee=10:RETURN
1710 frs="" / CLAVE DE ENTRADA : ?:"
GOSUB 980
1720 LOCATE #3,1,2:INPUT #3,ent$
1730 IF ent="" THEN RETURN
1740 IF ent#="clave" THEN ee=9:RET
URN
1750 IF mp="" THEN GOSUB 1860:RET
URN
1760 IF mp="E" THEN GOSUB 1790:RET
URN
1770 IF mp="F" THEN GOSUB 1950:RET
URN
1780 ee=11:RETURN
1790 frs="" / ETIQUETA m/9 : ?:"GOSU
B 980
1800 LOCATE #3,1,2:INPUT #3,et$:IF
LEN(et$)>9 THEN 1790
1810 et$=ROUND(SQR(VAL(var$)),2)
1820 MID$(matriz$(lin,col),4,10)=LE
FT(et$,10)
1830 MID$(matriz$(lin,col),2,1)="E"
1840 GOSUB 2800
1850 RETURN
1860 frs="" / VARIABLE 6/e:2/d : ?:"
GOSUB 980
1870 LOCATE #3,1,2:INPUT #3,var$:IF
var="" THEN RETURN
1880 IF var="9999999,9" THEN 1860
1890 IF var="9999999,9" THEN 1860
1900 IF var="9999999,9" THEN 1860
1910 IF var="0" THEN RETURN
1920 MID$(matriz$(lin,col),2,1)="V"
1930 MID$(matriz$(lin,col),4,10)=SP
ACES(10):MID$(matriz$(lin,col),4,10
)=LEFT$(STR$(var),10)

```

```

1940 GOSUB 2860:RETURN
1950 REM ***** FORMULAS *****
1960 fr=""/ FORMULA m/11g m/9/p
1 ?:"GOSUB 980
1970 LOCATE #3,1,2:INPUT #3,form#;I
F form#="" THEN RETURN
1980 IF LEN(form#)<5 THEN THEN 1960
1990 IF MID$(form#,4,1)="" THEN 1960
2000 IF MID$(form#,4,1)="" THEN 1960
2010 IF MID$(form#,4,1)="" THEN 1960
2020 IF MID$(form#,4,1)="" THEN 1960
2030 IF MID$(form#,4,1)="" THEN 1960
2040 IF MID$(form#,4,1)="" THEN 1960
2050 IF MID$(form#,4,1)="" THEN 1960
2060 IF MID$(form#,4,1)="" THEN 1960
2070 IF MID$(form#,4,1)="" THEN 1960
2080 IF MID$(form#,4,1)="" THEN 1960
2090 IF MID$(form#,4,1)="" THEN 1960
2100 IF MID$(form#,4,1)="" THEN 1960
2110 IF MID$(form#,4,1)="" THEN 1960
2120 IF MID$(form#,4,1)="" THEN 1960
2130 IF MID$(form#,4,1)="" THEN 1960
2140 IF MID$(form#,4,1)="" THEN 1960
2150 IF MID$(form#,4,1)="" THEN 1960
2160 IF MID$(form#,4,1)="" THEN 1960
2170 IF MID$(form#,4,1)="" THEN 1960
2180 IF MID$(form#,4,1)="" THEN 1960
2190 IF MID$(form#,4,1)="" THEN 1960
2200 IF MID$(form#,4,1)="" THEN 1960
2210 IF MID$(form#,4,1)="" THEN 1960
2220 IF MID$(form#,4,1)="" THEN 1960
2230 IF MID$(form#,4,1)="" THEN 1960
2240 IF MID$(form#,4,1)="" THEN 1960
2250 IF MID$(form#,4,1)="" THEN 1960
2260 IF MID$(form#,4,1)="" THEN 1960
2270 IF MID$(form#,4,1)="" THEN 1960
2280 IF MID$(form#,4,1)="" THEN 1960
2290 IF MID$(form#,4,1)="" THEN 1960
2300 IF MID$(form#,4,1)="" THEN 1960
2310 IF MID$(form#,4,1)="" THEN 1960
2320 IF MID$(form#,4,1)="" THEN 1960
2330 IF MID$(form#,4,1)="" THEN 1960
2340 IF MID$(form#,4,1)="" THEN 1960
2350 IF MID$(form#,4,1)="" THEN 1960
2360 IF MID$(form#,4,1)="" THEN 1960
2370 IF MID$(form#,4,1)="" THEN 1960
2380 IF MID$(form#,4,1)="" THEN 1960
2390 IF MID$(form#,4,1)="" THEN 1960
2400 IF MID$(form#,4,1)="" THEN 1960
2410 IF MID$(form#,4,1)="" THEN 1960
2420 IF MID$(form#,4,1)="" THEN 1960
2430 IF MID$(form#,4,1)="" THEN 1960
2440 IF MID$(form#,4,1)="" THEN 1960
2450 IF MID$(form#,4,1)="" THEN 1960
2460 IF MID$(form#,4,1)="" THEN 1960
2470 IF MID$(form#,4,1)="" THEN 1960
2480 IF MID$(form#,4,1)="" THEN 1960
2490 IF MID$(form#,4,1)="" THEN 1960
2500 IF MID$(form#,4,1)="" THEN 1960
2510 IF MID$(form#,4,1)="" THEN 1960
2520 IF MID$(form#,4,1)="" THEN 1960
2530 IF MID$(form#,4,1)="" THEN 1960
2540 IF MID$(form#,4,1)="" THEN 1960
2550 IF MID$(form#,4,1)="" THEN 1960
2560 IF MID$(form#,4,1)="" THEN 1960
2570 IF MID$(form#,4,1)="" THEN 1960
2580 IF MID$(form#,4,1)="" THEN 1960
2590 IF MID$(form#,4,1)="" THEN 1960
2600 IF MID$(form#,4,1)="" THEN 1960
2610 IF MID$(form#,4,1)="" THEN 1960
2620 IF MID$(form#,4,1)="" THEN 1960
2630 IF MID$(form#,4,1)="" THEN 1960
2640 IF MID$(form#,4,1)="" THEN 1960
2650 IF MID$(form#,4,1)="" THEN 1960
2660 IF MID$(form#,4,1)="" THEN 1960
2670 IF MID$(form#,4,1)="" THEN 1960
2680 IF MID$(form#,4,1)="" THEN 1960
2690 IF MID$(form#,4,1)="" THEN 1960
2700 IF MID$(form#,4,1)="" THEN 1960
2710 IF MID$(form#,4,1)="" THEN 1960
2720 IF MID$(form#,4,1)="" THEN 1960
2730 IF MID$(form#,4,1)="" THEN 1960
2740 IF MID$(form#,4,1)="" THEN 1960
2750 IF MID$(form#,4,1)="" THEN 1960
2760 IF MID$(form#,4,1)="" THEN 1960
2770 IF MID$(form#,4,1)="" THEN 1960
2780 IF MID$(form#,4,1)="" THEN 1960
2790 IF MID$(form#,4,1)="" THEN 1960
2800 IF MID$(form#,4,1)="" THEN 1960
2810 IF MID$(form#,4,1)="" THEN 1960
2820 IF MID$(form#,4,1)="" THEN 1960
2830 IF MID$(form#,4,1)="" THEN 1960
2840 IF MID$(form#,4,1)="" THEN 1960
2850 IF MID$(form#,4,1)="" THEN 1960
2860 IF MID$(form#,4,1)="" THEN 1960
2870 IF MID$(form#,4,1)="" THEN 1960
2880 IF MID$(form#,4,1)="" THEN 1960
2890 IF MID$(form#,4,1)="" THEN 1960
2900 IF MID$(form#,4,1)="" THEN 1960
2910 IF MID$(form#,4,1)="" THEN 1960
2920 IF MID$(form#,4,1)="" THEN 1960
2930 IF MID$(form#,4,1)="" THEN 1960
2940 IF MID$(form#,4,1)="" THEN 1960
2950 IF MID$(form#,4,1)="" THEN 1960
2960 IF MID$(form#,4,1)="" THEN 1960
2970 IF MID$(form#,4,1)="" THEN 1960
2980 IF MID$(form#,4,1)="" THEN 1960
2990 IF MID$(form#,4,1)="" THEN 1960
3000 IF MID$(form#,4,1)="" THEN 1960
3010 IF MID$(form#,4,1)="" THEN 1960
3020 IF MID$(form#,4,1)="" THEN 1960
3030 IF MID$(form#,4,1)="" THEN 1960
3040 IF MID$(form#,4,1)="" THEN 1960
3050 IF MID$(form#,4,1)="" THEN 1960
3060 IF MID$(form#,4,1)="" THEN 1960
3070 IF MID$(form#,4,1)="" THEN 1960
3080 IF MID$(form#,4,1)="" THEN 1960
3090 IF MID$(form#,4,1)="" THEN 1960
3100 IF MID$(form#,4,1)="" THEN 1960
3110 IF MID$(form#,4,1)="" THEN 1960
3120 IF MID$(form#,4,1)="" THEN 1960
3130 IF MID$(form#,4,1)="" THEN 1960
3140 IF MID$(form#,4,1)="" THEN 1960
3150 IF MID$(form#,4,1)="" THEN 1960
3160 IF MID$(form#,4,1)="" THEN 1960
3170 IF MID$(form#,4,1)="" THEN 1960
3180 IF MID$(form#,4,1)="" THEN 1960
3190 IF MID$(form#,4,1)="" THEN 1960

```

```

HEN 2580 ELSE ee=3:RETURN
2580 n(b)=VAL(MID$(matriz$(aa,bb),4
,10))
2590 NEXT b
2600 l=i:i=1
2610 IF comando=0 THEN RETURN
2620 IF c$(i)="" THEN w=1
2630 IF c$(i)="" THEN w=2
2640 IF c$(i)="" THEN w=3
2650 IF c$(i)="" THEN w=4
2660 IF c$(i)="" THEN w=5
2670 IF c$(i)="" THEN w=6
2680 dn=4 GOSUB 2700,2710,2720,2730
,2740,2750
2690 GOTO 2610
2700 n(h)=n(h)+n(h+1):GOSUB 2790:RE
TURN
2710 n(h)=n(h)-n(h+1):GOSUB 2790:RE
TURN
2720 n(h)=n(h)*n(h+1):GOSUB 2790:RE
TURN
2730 n(h)=n(h)/n(h+1):GOSUB 2790:RE
TURN
2740 n(h)=n(h)*n(h+1):/100:GOSUB 2
790:RETURN
2750 n(h)=n(h)*n(h+1):GOSUB 2790:RE
TURN
2760 IF comando<0 THEN n(i)=n(i)+n
(i+1):GOSUB 2790:RETURN
2770 RETURN
2780 i=i+1:comando=comando-1:n(coma
ndo)=0:RETURN
2790 GOSUB 2780:GOSUB 2760:RETURN
2800 REM ***** IMPRESION (etiqueta)
*****
2810 IF lin<=1 AND lin<=12 THEN 2
820 ELSE RETURN
2820 IF col=y-1 AND col<=7 THEN 28
30 ELSE RETURN
2830 c=col-y+1:lin=x+i:m=MID$(matr
iz$(lin,col),4,9)
2840 LOCATE #5,c#1+1,1:PRINT #5,US
ING "###"m
2850 RETURN
2860 REM ***** IMPRESION (variable)
*****
2870 IF lin<=1 OR lin<=12 THEN 288
0 ELSE RETURN
2880 IF col=y-1 OR col<=7 THEN 289
0 ELSE RETURN
2890 c=col-y+1:lin=x+i:VAL(MID$(m
atriz$(lin,col),4,9))
2900 LOCATE #5,c#1+1,1:PRINT #5,US
ING "###"m
2910 RETURN
2920 REM ***** PROTECCION *****
2930 fr=""/ CELDA : ?:"GOSUB 980
2940 LOCATE #3,1,2:INPUT #3,nu#
2950 nu#=UPPER$(nu#)
2960 IF nu#="" THEN RETURN
2970 IF LEN(nu#)<2 THEN 2920
2980 a=ASC(LEFT$(nu#,1))-64
2990 b=ASC(RIGHT$(nu#,1))-64
3000 IF a<1 OR a>26 OR b<1 OR b>26
THEN ee=3:RETURN
3010 IF LEFT$(matriz$(a,b),1)="" THEN
HEN ee=3:RETURN
3020 MID$(matriz$(a,b),1,1)="" THEN
3030 RETURN
3040 RETURN
3050 WINDOW 10,30,8,18
3060 PAPER 1:OPEN 0:GRAPHICS:PEN 0:CL
S
3070 MOVE 74,120:DRAW 236,120:DRW
226,280:DRAW 74,280:DRAW 74,120
3080 PAPER 0:OPEN 1:LOCATE 2,2:PRIN
T "OPERADORES 01"
3090 PAPER 1:OPEN 0:LOCATE 2,3:PRIN
T "XY=1'CELDA'"
3100 LOCATE 2,4:PRINT"SEN.XY
COS.XY"
3110 LOCATE 2,5:PRINT"TAN.XY
LOG.XY"
3120 LOCATE 2,6:PRINT"LOG.XY
PI.XY"
3130 LOCATE 2,7:PRINT"LOG=(Log base
10)"
3140 LOCATE 2,8:PRINT"LOG=(Log Natu
ral)"
3150 LOCATE 2,9:PRINT"PI=(PI * Cel
da 3)"
3160 PAPER 0:OPEN 1:LOCATE 2,10:PRIN
T "SPACE PARA SEGUIR"
3170 IF INKEY(47)<0 THEN 3170
3180 WINDOW 15,35,7,17
3190 PAPER 1:OPEN 0:CLS

```



```

3200 MOVE 114,136:DRAW 276,136:DRW
276,296:DRAW 114,296:DRAW 114,136
3210 PAPER 0:OPEN 1:LOCATE 2,2:PRIN
T "OPERADORES 02"
3220 PAPER 1:OPEN 0:LOCATE 2,3:PRIN
T "XY=1'CELDA'"
3230 LOCATE 2,4:PRINT"X1Y1=2
'CELDA'"
3240 LOCATE 2,5:PRINT"RAZ.XY (cuadr
ada)"
3250 LOCATE 2,6:PRINT"PROX.XY.X1Y1"
3260 LOCATE 2,7:PRINT"SUN.XY.X1Y1"
3270 LOCATE 2,8:PRINT"REP.XY (x):o(x
:oy)"
3280 LOCATE 2,9:PRINT"x:thor. xy:
ver."
3290 PAPER 0:OPEN 1:LOCATE 2,10:PRIN
T "SPACE PARA SEGUIR"
3300 IF INKEY(47)<0 THEN 3300
3310 WINDOW 26,65,9,16
3320 PAPER 1:OPEN 0:CLS
3330 MOVE 204,152:DRAW 499,152:DRW
499,264:DRAW 204,264:DRAW 204,152
3340 PAPER 0:OPEN 1:LOCATE 2,2:PRIN
T "OPERADORES 03"
3350 PAPER 1:OPEN 0:LOCATE 2,3:PRIN
T "+,*,/,)Suma,resta,multiplica,di
visi6n"
3360 LOCATE 2,4:PRINT"vide la 1' Cel
da con la 2' Celda"
3370 LOCATE 2,5:PRINT"(X1):1' Celda
* 100 de la 2' Celda"
3380 LOCATE 2,6:PRINT"(X1):1' Celda
a la potencia de la 2'"
3390 PAPER 0:OPEN 1:LOCATE 2,7:PRIN
T "SPACE PARA PANTALLA PRINCIPAL"
3400 IF INKEY(47)<0 THEN 3400
3410 CLS #5:RETURN
3420 REM ***** REC-IMPRESION *****
3430 fr=""/ ESTA SEGURO : ?:"GOSUB
980
3440 LOCATE #3,1,2:INPUT #3,see#
3450 see=UPPER$(see)
3460 IF LEFT$(see,1)<0 THEN RETU
RN
3470 RUN
3480 REM ***** SALIDA *****
3490 fr=""/ ESTA SEGURO : ?:"GOSUB
980
3500 LOCATE #3,1,2:INPUT #3,see#
3510 see=UPPER$(see)
3520 IF LEFT$(see,1)<0 THEN RETU
RN
3530 CLEAR:WINDOW 1,80,1,25:CLS:NEW
END
3540 REM ***** CALCULO FORMULAS ***
***
3550 FOR aa=UPPER$(MID$(matriz$(lin,
col),15,LEN(form#)))
3560 IF LEN(form#)=9 THEN GOSUB 371
0
3570 IF LEN(form#)=6 THEN GOTO 3580
ELSE RETURN
3580 oper=LEFT$(form#,3):cd=RIGHT
$(form#,2)
3590 a=ASC(LEFT$(cd$,1))-64:b=ASC(R
IGHT$(cd$,1))-64
3600 IF a<1 OR a>26 OR b<1 OR b>26
THEN ee=3:RETURN
3610 IF MID$(matriz$(a,b),2,1)="" THEN
ee=3:RETURN
3620 a=VAL(MID$(matriz$(a,b),4,10))
3630 IF oper="SEN" THEN n(i)=SIN(a)
3640 IF oper="COS" THEN n(i)=COS(a)
3650 IF oper="TAN" THEN n(i)=TAN(a)
3660 IF oper="LOG" THEN n(i)=LOG10
(ABS(a))

```

```

3670 IF oper="LON" THEN n(1)=LOG(A
BS(a))
3680 IF oper="PI1" THEN n(1)=PI#a
3690 IF oper="RA2" THEN n(1)=SDR(a)
3700 RETURN
3710 oper="LEFT*(FORM,3):cd1=MSD
(FORM,5):2:cd2=RIGHT*(FORM,2)
3720 a1=ASC(LEFT*(cd1,1))-64:a2=ASC
C(LEFT*(cd2,1))-64:b1=ASC(RIGHT*(c
d1,1))-64:b2=ASC(RIGHT*(cd2,1))-6
4
3730 IF a1<1 OR a2<1 OR a1>26 OR a2
>26 OR b1<1 OR b2<1 OR b1>26 OR b2>
26 THEN oper=RETURN
3740 j=0:n(1)=0
3750 IF oper="SUM" THEN GOSUB 3790
3760 IF oper="PRO" THEN GOSUB 3830
3770 IF oper="REP" THEN GOSUB 3850
3780 RETURN
3790 GOSUB 3800:RETURN
3800 IF a1=2 j=0:FOR i=1 TO b1 TO
b2:n(1)=n(1)+VAL(MID$(MATRIZ*(a1,i)
,a,10)):j=j+1:NEXT:RETURN
3810 IF b1=7:FOR i=1 TO b2:n(1)=n(1)
+a2:VAL(MID$(MATRIZ*(a1,i),b1
,a,10)):j=j+1:NEXT:RETURN
3820 RETURN
3830 GOSUB 3800:IF j=0 THEN RETURN
3840 n(1)=n(1)+j:RETURN
3850 IF MID$(MATRIZ*(a1,b1),14,1)="
P" THEN RETURN
3860 q=INSTR(MATRIZ*(a1,b1),"*")
3870 longq=15-com=(long(1)/3)-1
3880 FOR i=1 TO longq/3+1:MID$(mat
r1z*(a1,b1),i+14,1):NEXT
3890 IF a2=b2=48 THEN GOSUB 3990
3900 IF a2=b2=50 THEN GOSUB 3920
3910 RETURN
3920 FOR i=2 TO long STEP 3:r#(i)=C
HR$(ASC(r#(i)+1)):NEXT
3930 r#(long+1)="*":nfn#=""
3940 FOR i=1 TO long+1:nfn#nfn#r#(
i):NEXT
3950 MID$(MATRIZ*(lin,col),15,10)=B
FACE(i)
3960 MID$(MATRIZ*(lin,col),15,10)=A
1:nfn#
3970 MID$(MATRIZ*(lin,col),14,1)="G"
3980 GOSUB 2080:RETURN
3990 FOR i=1 TO long+1 STEP 3:r#(i)=
CHR$(ASC(r#(i)+1)):NEXT
4000 r#(long+1)="*":nfn#=""
4010 FOR i=1 TO long+1:nfn#nfn#r#(
i):NEXT
4020 MID$(MATRIZ*(lin,col),15,10)=B
FACE(i)
4030 MID$(MATRIZ*(lin,col),15,10)=A
1:nfn#
4040 MID$(MATRIZ*(lin,col),14,1)="B"
4050 GOSUB 2080:RETURN
4060 REM ***** TRANSFERENCIA *****
4070 WINDOW #0,48,62,14,24:PAPER 1:
REN O:GRAPHIC PEN O
4080 MOVE 380,24:DRAW 491,24:DRAW 4
91,184:DRAW 380,184:DRAW 380,24
4090 PAPER 0:PEN 1:LOCATE 2,2:PRINT
"TRANSFERENCIA"
4100 PAPER 1:PEN 0:LOCATE 2,3:PRINT
"Lineas i"
4110 LOCATE 2,5:PRINT "Columnas i"
4120 LOCATE 2,7:PRINT "Nombre i"
4130 PAPER 0:PEN 1:LOCATE 2,9:PRINT
"CONTINUACION"
4140 LOCATE 2,10:PRINT " (s/
n)"
4150 PAPER 1:PEN 0
4160 LOCATE 2,4:INPUT #0,lines#1:11
n#="UPPER$(lines#)"
4170 LOCATE 2,6:INPUT #0,columns#
:columns#="UPPER$(columns#)"
4180 LOCATE 2,8:INPUT #0,nombres#
4190 PAPER 0:PEN 1
4200 a#="INKEY$: IF a#="" THEN 4200
4210 IF a#="*" OR a#="S" THEN 4230
4220 a#="1":GOSUB 490:RETURN
4230 a#="ASC(MID$(lines#,1))-64":b#="
ASC(MID$(lines#,2))-64":d#="ASC(MID$(co
lumnas#,2))-64":d#="ASC(MID$(co
lumnas#,2))-64"
4240 OPENUD nombres#
4250 WRITE #9,titulos,a,b,c,d
4260 FOR i=a TO b
4270 FOR j=c TO d

```

```

4280 WRITE #9,matriz*(i,1)
4290 NEXT:EXIT
4300 CLOSEOUT
4310 x=1:y=1:GOSUB 490:RETURN
4320 REM ***** CURSOR ARRIBA *****
4330 IF CLC2 THEN RETURN
4340 GOSUB 4570:LOCATE #5,cc#11+1,c
1:PAPER #5,OfPEN #5,1
4350 GOSUB 4600:c1=c1-1:GOSUB 4570:
PAPER #5,1:PEN #5,0
4360 LOCATE #5,CC#11+1,CL:GOSUB 460
0:PAPER #5,OfPEN #5,1
4370 lin=c1-1:RETURN
4380 RETURN
4390 REM ***** CURSOR ABAJO *****
4400 IF c1<1 THEN RETURN
4410 GOSUB 4570:LOCATE #5,cc#11+1,c
1:PAPER #5,OfPEN #5,1
4420 GOSUB 4600:c1=c1+1:GOSUB 4570:
PAPER #5,1:PEN #5,0
4430 LOCATE #5,cc#11+1,c1:GOSUB 460
0:PAPER #5,OfPEN #5,1
4440 lin=c1-1:RETURN
4450 REM ***** CURSOR DERECHA *****
4460 IF cc#5 THEN RETURN
4470 GOSUB 4570:LOCATE #5,cc#11+1,c
1:PAPER #5,OfPEN #5,1
4480 GOSUB 4600:cc=cc+1:GOSUB 4570:
PAPER #5,1:PEN #5,0
4490 LOCATE #5,cc#11+1,c1:GOSUB 460
0:PAPER #5,OfPEN #5,1
4500 col=cc+1:RETURN
4510 REM ***** CURSOR IZQUIERDA ***
**
4520 IF cc<1 THEN RETURN
4530 GOSUB 4570:LOCATE #5,cc#11+1,c
1:PAPER #5,OfPEN #5,1
4540 GOSUB 4600:cc=cc-1:GOSUB 4570:
PAPER #5,1:PEN #5,0
4550 LOCATE #5,cc#11+1,c1:GOSUB 460
0:PAPER #5,OfPEN #5,1
4560 col=cc+1:RETURN
4570 k#="MID$(MATRIZ*(c1+n-1,cc+y),2
,1)
4580 m#="MID$(MATRIZ*(c1+n-1,cc+y),4
,9)
4590 RETURN
4600 IF k#="E" THEN PRINT #5,USING
"5:1# ELSE PRINT #5,USING "*****
"1:VAL(m#)
4610 RETURN
4620 REM ***** CALCULO PLANILLA ***
**
4630 WINDOW #0,47,61,20,22
4640 IF c<1 OR y<1 THEN RETURN
4650 IF cc>0 OR c<1 THEN RETURN
4660 r#=""/:ESPERE, CALCULANDO PLA
NILLA (1)GOSUB 980
4670 FOR i=1 TO 26
4680 FOR j=1 TO 26
4690 lin=i:col=j
4700 IF MID$(MATRIZ*(i,j),2,1)<">
P" THEN 4810
4710 w=INSTR(MATRIZ*(lin,col),"*")
4720 for#="MID$(MATRIZ*(lin,col),15
,w-15)
4730 IF MID$(MATRIZ*(i,j),14,1)="
P" THEN GOSUB 3540
4740 IF MID$(MATRIZ*(i,j),14,1)="
G" THEN GOSUB 2380
4750 n(1)=ROUND(n(1),2)
4760 MID$(MATRIZ*(i,j),4,10)=SPAC
E(10)
4770 MID$(MATRIZ*(i,j),4,10)=RIGH
T$(STR$(n(1)),10)
4780 IF i%12 OR j%7 THEN BOTO 481
0
4790 c#="j-1":i#="i+VAL(MID$(MATRIZ*(
i,j),4,9))
4800 LOCATE #5,cc#11+1,1:PRINT #5,US
ING "*****"#:3:u
4810 NEXT:EXIT:cc=0:c1=1:lin=1:col=
1:RETURN

```

GANA 100.000 PESETAS CON MICROHOBBY AMSTRAD SEMANAL

Porque pretendemos que **AMSTRAD SEMANAL** sea también vuestra revista, hemos abierto una sección en la que se publicarán los mejores programas originales recibidos en nuestra redacción. Vosotros seréis los encargados de realizar estas páginas, en las que podréis aportar ideas y programas interesantes para otros lectores.

Las condiciones son sencillas:
— Los programas se enviarán a **AMSTRAD SEMANAL** en una cinta de cassette, sin protección en el software, de forma que sea posible obtener un listado de los mismos.

— Cada programa debe ir acompañado de un texto explicativo en el cual se incluyan:

- Descripción general del programa.
- Tabla de subrutinas y variables utilizadas, explicando claramente la función de cada una de ellas.
- Instrucciones de manejo.
- Todos estos datos deberán ir escritos a máquina o con letra clara para mayor comprensión del programa.
- No se admitirán programas que contengan caracteres de control, debido a que no son correctamente interpretados por las impresoras.
- En una sola cinta puede introducirse más de un programa.
- Una vez publicado, **AMSTRAD SEMANAL** abonará al autor del programa de **15.000 a 100.000 pesetas**, en concepto de derechos de autor.

— Los autores de los programas seleccionados para su publicación, recibirán una comunicación escrita de ello en un plazo no superior a dos meses a partir de la fecha en que su programa llegue a nuestra redacción.

— **AMSTRAD SEMANAL** se reserva el derecho de publicación o no del programa.

— Todos los programas recibidos quedarán en poder de **AMSTRAD SEMANAL**.

— Los programas sospechosos de plagio serán eliminados inmediatamente.

¡ENVIAMOS TU PROGRAMA!

Adjuntando las siguientes datos:
Nombre y apellidos,
dirección y teléfono.
Indicando claramente en la sobre:

AMSTRAD SEMANAL
a HOBBY PRESS, S. A. La Granja, 39
Pol. Ind. Alcobendas (Madrid)





Aquí se habla de algo muy inteligente

*P*ara construir un edificio hacen falta un arquitecto, herramientas y los materiales más modernos.

Para crear un programa que pueda pensar y aprender por sí mismo es necesario un programador, tú, un gran ordenador, tu Amstrad, y una herramienta revolucionaria, el lenguaje de programación LISP.

AMSTRAD Semanal ha escrito una versión completa de LISP, y vamos a regalártelo y a enseñarte a utilizarlo.

Programas estúpidos no, gracias.

Más información en AMSTRAD Especial número 2.

LOS ORDENADORES YA PUEDEN PENSAR

por R.G. Bernal

Comenzamos con este artículo una serie dedicada a la Inteligencia Artificial, IA para los amigos y entendidos en el tema. Somos conscientes que abordamos un tema completamente nuevo, que nunca se había visto antes tratado en un medio informativo especializado, fuera de los claustros de las Universidades, de una forma continua y profunda, y que, tal vez por ello, está rodeado de un aura maléfica de misterio y esoterismo.

Pues bien, nada más lejos de la realidad: la IA es una técnica de programación como cualquier otra, y puede ser aprendida y ejecutada por cualquier persona que tenga un Amstrad y quiera aprender.

A pesar de «jugarnos el cuello» al abrir camino en el tema más puntero de la informática de los 80, estamos completamente convencidos de que este artículo, y los programas y herramientas que le seguirán, van a servir para desvelar el enigma que se esconde detrás de esta apasionante disciplina.

¿Para qué sirve? ¿Cómo se utiliza? Son preguntas a las que iremos dando respuesta de forma profunda, y a la vez sencilla y clara, accesible a todo el mundo.

Por hoy basta con una completa introducción que nos sirva para acercarnos a este nuevo mundo, que sentará sin la menor duda las bases de la informática del futuro, que ya es hoy.

Tal vez los ordenadores no puedan pensar, pero sí puede parecer que lo hacen, y vamos a enseñaros la forma de conseguirlo.



Desde la antigüedad el hombre ha soñado con construir máquinas inteligentes. La primera máquina que podía tener atributos de inteligencia fue construida en 1912 por Leonardo Torres Quevedo. El autómeta ajedrecístico que él construyó, jugaba las finales de rey y torre rey (siguiendo el método conocido que permite la victoria en cualquier situación), si en un acto de mal perder, por nuestra parte, realizábamos movimientos ilegales, no sólo se quejaba, sino que a la tercera vez, cansado de nuestra mala fe, dejaba automáticamente de jugar. Pero hubo que esperar a la aparición de los ordenadores digitales para dar los primeros pasos importantes. En la segunda mitad de los años cuarenta empezaron a salir a la luz los primeros programas de ajedrez. En aquella época aún se pensaba que era posible construir un programa

que, sólo con las reglas del juego, pudiese vencer. Esta idea se abandonó posteriormente debido a la gran cantidad de tiempo que consumiría en encontrar la solución (en algunos juegos es posible que ni siquiera la encontrase).

A principios de la década de los cincuenta aparecieron los primeros programas de álgebra simbólica, capaces de derivar funciones. Fue éste un logro muy importante, ya que se pasó de tratar con números a manipular símbolos. ¡Por fin estaban resueltos los problemas de aquellos estudiantes que se aburrían o sufrían intentando calcular derivadas de funciones, las mismas derivadas con las que tú te encontraste en el bachiller o con las que, tal vez, aún tropezarás!

El nacimiento de la IA

En 1956 aparece la Inteligencia Artificial (IA) como disciplina independiente. Hasta esta fecha se habían ido desarrollando las teorías de las que haría uso la IA. En un congreso celebrado en Dartmouth en ese año, Newell, y Simon presentan el

primer programa de inteligencia artificial. Era el «**Logic Theorist**», un demostrador de teoremas de la Lógica proposicional, basado en los trabajos publicados por Herbrand en 1930. El tipo de cosas que era capaz de demostrar no era excesivamente complicado (la negación de la negación de P es igual a P, o este otro: de P y Q se deriva P), pero su demostración puede representar un problema para cualquier persona no acostumbrada a este tipo de cosas. Fue en este mismo congreso donde McCarthy —el creador del Lisp— propuso el nombre de Inteligencia Artificial para esta nueva ciencia.

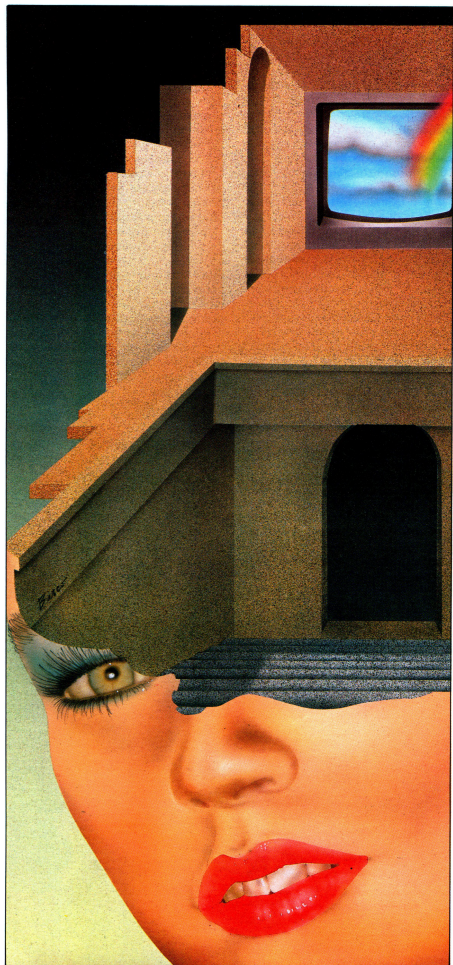
En los orígenes de la IA ya hay dos puntos de vista diferentes, provenientes de dos comunidades científicas distintas. Por un lado estaba el punto de vista de la Psicología, que pretendía utilizar el ordenador como una herramienta para comprobar teorías sobre el comportamiento humano. La Neurología había descubierto la organización interna de algunos sistemas nerviosos simples. Se intentó entonces simular estos sistemas mediante ciertas estructuras en red, apareciendo en 1958 el «**perceptron**» de Rosenblatt cuyo objeti-

Inteligencia ARTIFICIAL

vo era el reconocimiento de imágenes. Según su creador sería posible salir a la calle dotado de uno de tales aparatos a la búsqueda de objetos de un cierto tipo (gatos, casas, coches, etc.). Para ello bastaría que ajustásemos una rueda (como si se tratase del mando de una batidora) en la clase de objeto que queríamos hallar y presentásemos ante nuestro «preceptor» objetos de varios tipos. Entonces el mecanismo nos contestaría si lo que le mostrábamos pertenecía o no a la clase de objetos requeridos. Sin embargo, en 1969, M.L. Minsky y S. Papert —uno de los padres del Logo, «el lenguaje de la tortuga»— rompieron estas maravillosas fantasías. Demostraron que este tipo de sistemas era incapaz de distinguir formas conexas (de una sola pieza) de otras que no lo eran y se abandonó su estudio. Hoy parecen haber recobrado cierto interés.

Otro punto de vista ha sido el «informático», desde el que se pretenden hallar técnicas para simular el razonamiento y el comportamiento inteligente. Los investigadores en este campo no intentan reproducir el comportamiento humano, sino sus efectos. Es decir, no intentan construir programas que juegan al ajedrez en la misma forma en que lo haría un buen jugador —analizando sólo las jugadas que parecen mejores—, pero sí que juegan al nivel de maestro. Este es el campo más prometedor y en el que se están consiguiendo mayores logros. ¡De hecho, algunos programas de ajedrez ya juegan al nivel de expertos!

A partir de la década de los cincuenta comenzaron a aparecer gran número de programas de IA en numerosos campos: ajedrez, damas, geometría plana (la geometría plana es un juego que parece ser que inventaron unos señores en Grecia hace algunos años), traducción automática, etc. Los primeros grandes logros en algunos de estos campos desataron una euforia injustificada que llevaron a predicciones desatinadas como la que formularon en 1958 Newell y Simon: antes de 1968 será campeón de ajedrez un programa y se habrá demostrado «automáticamente» un teorema importante. Ninguna se cumplió.



Este fracaso se debió fundamentalmente a las técnicas utilizadas para construir estos programas, basados en la comprobación de todas las posibilidades de actuación en cada momento (soluciones combinatorias). Por ejemplo, en el caso del ajedrez, estos programas calculan, para cada posible movimiento de sus fichas, todas las respuestas posibles por parte de su adversario, y para cada una de éstas de nuevo todos sus posibles movimientos, etc. Se va creando de esta forma un árbol cuya raíz es la posición inicial; las ramas que salen de la raíz llevan a tableros a los que se puede llegar haciendo un único movimiento de una ficha propia; las ramas que salen de estos tableros llevan a tableros que representan posiciones a las que se llega tras hacer un movimiento de una ficha del oponente, etc. A cada tablero se le asigna después un valor según una cierta técnica llamada mínimas (porque unas veces se toma el máximo de unos valores y otras se toma el mínimo) que emplea una función de evaluación que hemos de fijar nosotros y que «mide» lo buena o mala que es una cierta posición. La jugada que nos lleve a un tablero de máxima puntuación es la que se elige. En ajedrez, el número medio de posibles jugadas es de unas 35, de modo que, para analizar hasta un nivel tres (propio-contrario-propio) era necesario revisar más de 40.000 tableros (35 x 35 x 35). Para lograr por estos métodos un nivel de juego aceptable era necesario profundizar hasta al menos seis niveles, lo que suponía analizar más de 1.800 millones de tableros para realizar un solo movimiento (¡las máquinas ajedrecísticas más rápidas de hoy en día tardarían más de tres horas en mover una ficha si utilizasen esta técnica tan grosera!).

Un programa que aprende

En 1959, A. Samuel presentó un programa que jugaba bien a las damas y además parecía «aprender», pues tras cada partida jugada mejoraba su rendimiento. Sin embargo, toda la «inteligencia» de este programa había sido colocada ya por el autor. El programa jugaba utilizando la técnica del minimax, sólo que la función de evaluación, que puntuaba determinadas características deseables del juego, no estaba

completamente fijada. Los factores que se medían (ocupación del centro del tablero, movilidad de las fichas, posibilidad de conseguir dama, etc.) estaban fijados a priori, pero los coeficientes que indicaban la importancia de estas características eran calculados por el programa según iba jugando, por lo que tras varias partidas estos coeficientes estaban bastante bien calibrados. Este era todo el aprendizaje que tenía el programa de Samuel. Se confió en gran manera en este método, que luego resultó inaplicable en otras áreas.

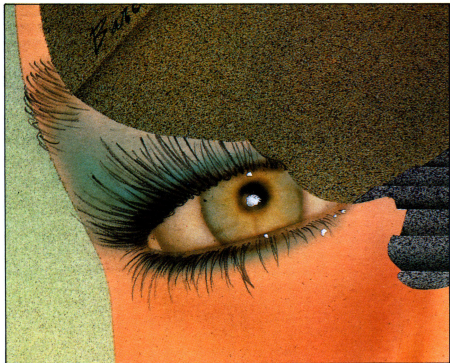
Otro campo de investigación en IA que despertó también grandes esperanzas es el de la traducción automática. Los primeros trabajos en este campo se proponían descifrar códigos militares. Se pensó entonces en utilizar estas técnicas para la traducción mecánica por ordenador de un idioma a otro (en particular del ruso al inglés, que para eso pagaban los americanos). Los métodos empleados eran muy simples. Consistían en traducir el texto palabra por palabra, utilizando enormes diccionarios. Después se cambiaba el orden de algunas palabras (sustantivo por adjetivo, por ejemplo) y se hacían concordar en género y número. Las traducciones obtenidas por aquellos métodos eran absolutamente ilegibles. No se hicieron progresos mayores y en 1964 apareció el informe ALPAC, tras el cual el Gobierno norteamericano, principal contribuidor monetario a la investigación en IA, decidió suspender su apoyo econó-

mico a estos estudios. Esto supuso un gran retraso en las investigaciones en este campo, que no se recuperó hasta mediados de los años setenta.

Entre 1965 y 1970 hubo un progreso importante en la resolución general de problemas mediante soluciones combinatorias debido al aumento en la potencia de las computadoras y a algunas técnicas más sofisticadas como el corte alfa-beta. Con este método, complementario del minimax, se evita revisar todas las ramas del árbol que se genera al analizar un problema (por ejemplo, el árbol que se produce al observar los posibles movimientos en ajedrez). En 1967, R. Greenblatt presentó el primer programa de ajedrez que ganaba sin dificultad a un jugador normal. Sin embargo, este tipo de programas no se pueden considerar «inteligentes», pues sólo usan la fuerza bruta de la computadora, es decir, su capacidad de cálculo rápido. (¿Se puede considerar inteligente a una persona que hace sumas muy deprisa?)

Robots e inteligencia artificial

A finales de los años sesenta se desarrollaron en gran manera los estudios sobre sistemas inteligentes para la robótica, en particular en tres campos: visión artificial, planificación de tareas y manipulación de objetos. Es curioso notar que el término «robot», que en checo significa trabajo



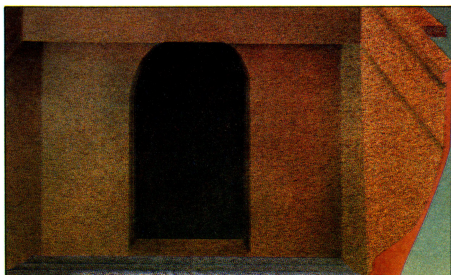


forzado, tuvo su origen en la comedia R.U.R. escrita por el checoslovaco K. Kapek, en 1920 (ya en aquella época, la ciencia ficción tenía más influencia que la propia ciencia). En IA, un robot es un mecanismo que recibe informaciones del mundo exterior mediante sensores (cámaras de vídeo, ultrasonidos, rayos infrarrojos, etc.) y tal que dadas unas órdenes (posiblemente en un lenguaje natural, como el español) es capaz de planificar las acciones que ha de llevar a cabo con el fin de ejecutar las órdenes que ha recibido, modificando en consecuencia el universo que le rodea. Por ejemplo, basta que le digamos a nuestro robot que nos traiga una cerveza helada a la terraza, donde nos hallamos tomando el sol, para que Federico, que así se llama, cruce la puerta entre la terraza y el salón, pase luego al pasillo, localice la puerta de la cocina, entre en ella, se dirija hacia el frigorífico, abra la puerta del mismo y extraiga una cerveza. Después de arrojarla con su justa espuma sobre una jarra de cerámica irá a buscar nuestros panchitos preferidos, lo colocará todo en una bandeja y nos lo traerá hasta la terraza sorteando cuantos obstáculos encuentre.

El problema más arduo con el que se enfrenta la robótica es, sin lugar a dudas, el de la visión artificial. Por otro lado, es una labor ineludible si queremos crear robots capaces de ejecutar complejas tareas de manipulación de objetos en ambientes diversos y cambiantes, es decir, capaces de adaptarse a su entorno. El problema se complica aún más cuando se intenta dotar a los computadores de una visión más real, una visión en tres dimensiones. No se encuentran con este tipo de problemas los sistemas de visión artificial empleados en la actualidad en la industria, ya que trabajan en unas condiciones ambientales bastante homogéneas: luz de igual intensidad en cada momento, focos luminosos colocados siempre en los mismos lugares, los objetos que ha de manipular no varían, etc.

Otro de los problemas fundamentales de la IA es la comprensión del lenguaje natural. Sería deseable poder conversar con una computadora en nuestro propio idioma, pero los lenguajes naturales son tan sumamente complicados y variables que todo intento por conseguir que una computadora pueda mantener un diálogo serio y general con nosotros

(o sea, los humanos) ha sido un fracaso. Se han logrado progresos significativos limitando el tema de conversación a campos muy específicos. En este sentido es digno citar el trabajo realizado por T. Winograd (mago de la informática) con su programa SHRDLU (llamado así por ser éstas las letras que en orden de máxima aparición ocupan en inglés los lugares del séptimo al duodécimo, ambos inclusive). Este programa, creado en 1970, conoce y manipula un mundo de bloques (cubos, pirámides y cajas) de diversos colores y tamaños situados sobre una mesa. (En realidad, el programa no maneja físicamente ningún mundo de este tipo, ni tan siquiera lo representa sobre una pantalla de ordenador, pero él debe saber lo que se hace). Se le pueden hacer preguntas sobre un universo y dar órdenes para que manipule los bloques, todo ello en inglés. El programa actúa de igual forma que lo debería hacer un robot, planificando las acciones que tiene que llevar a cabo en un orden preciso. Veamos un ejemplo de la actuación de SHRDLU: Supongamos que en el estado inicial hay sobre la mesa un bloque grande rojo y dos bloques verdes medianos, sobre uno de



los cuales descansa una pirámide roja mediana. Sobre la mesa también está un bloque rojo pequeño que tiene encima una pirámide verde pequeña. Un posible diálogo (naturalmente traducido, porque aún no entiendo de castellano) sería el siguiente:

OPERADOR: ¿Podrías apilar, por favor, los dos bloques rojos y también un cubo verde o una pirámide?

SHRDLU: Con gusto.

OPERADOR: ¿Qué cubo se halla sobre la mesa?

SHRDLU: El cubo verde grande que soporta a la pirámide roja.

OPERADOR: Pon un cubo pequeño sobre el cubo verde que soporta una pirámide.

SHRDLU: Muy bien.

OPERADOR: Coloca sobre él la pirámide más pequeña.

SHRDLU: De acuerdo.

OPERADOR: ¿Cuántas cosas están situadas por encima de los cubos verdes?

SHRDLU: No estoy seguro del significado de «por encima» en la frase «por encima de los cubos verdes». Quieres decir:

1. Directamente en contacto con su superficie.

2. En cualquier lugar más alto.

OPERADOR: 2.

SHRDLU: Hay tres.

OPERADOR: ¿Has tirado alguna pirámide antes de colocar la verde sobre el cubo pequeño?

SHRDLU: Sí, la verde.

OPERADOR: ¿Cuándo la cogiste?

SHRDLU: Mientras estaba apilando el cubo rojo, un bloque rojo grande y un cubo verde grande.

OPERADOR: ¿Por qué?

SHRDLU: Para desembarazarme de ella.

OPERADOR: ¿Por qué hiciste eso?
SHRDLU: Para despejar el cubo rojo.

OPERADOR: ¿Por qué despejaste ese cubo?

SHRDLU: Para colocarlo sobre un cubo verde grande.

OPERADOR: ¿Por qué hiciste eso?

SHRDLU: Para apilar el cubo rojo, un gran bloque rojo y el cubo verde grande.

OPERADOR: ¿Por qué hiciste eso?

SHRDLU: Porque tú me lo pediste.

Pensamiento y experiencia

De no ser que los programas sean capaces de ampliar sus «**capacidades**» haciendo uso de sus «**experiencias**», su rendimiento estará limitado por el conocimiento, previsión y tiempo libre del programador. De ahí que la capacidad de aprender adquiriera una tremenda importancia en IA. El mundo de bloques de Winograd está siendo utilizado por algunos investigadores de la IA como un magnífico laboratorio donde experimentar diversas estrategias de aprendizaje.

La conversación «**inteligente**» de SHRDLU y su comprensión del mundo con el que trata nos introduce en otro fructífero campo de la IA: el de los sistemas expertos.

En los primeros años setenta, un equipo de trabajo de la Universidad de Stanford intentaba crear un sistema capaz de determinar la estructura de un compuesto orgánico complejo a partir de ciertos datos experimentales como su espectro de masas. Dado que en este campo es inaplicable la combinatoria, los investigadores de Stanford dirigieron sus

trabajos por otros caminos. Se hacía preciso construir un sistema capaz de utilizar grandes cantidades de conocimientos y hacer deducciones a partir de ellos. Se plantearon por tanto dos nuevos problemas: manejar grandes bases de datos de forma eficaz y utilizar los conocimientos que ya poseía de forma que produjese datos completamente nuevos. Un problema paralelo era la obtención de los conocimientos iniciales, para lo cual se precisaba de la ayuda de los expertos en el campo del análisis químico. Como consecuencia de aquellos trabajos surgió el sistema DENDRAL. Este sistema evolucionó rápidamente y permitió crear nuevos métodos para la IA. La experiencia adquirida llevó a la aparición en 1974 MYCIN, el primer sistema experto, capaz de diagnosticar una infección bacteriana y recetar los antibióticos más apropiados para combatir la enfermedad. Además, MYCIN puede solicitar los informes clínicos que crea convenientes para realizar un pronóstico más fiable. Este sistema acierta sus predicciones en un 80 por 100 de los casos. ¡No estaría mal hacer un sistema experto para rellenar quinielas si fuese capaz de acertar los «catorce» con tanta facilidad como MYCIN cura catarros!

Una de las características más destacadas de los sistemas expertos es su separación entre los conocimientos que utilizan (la base conocimientos) y el programa que permite hacer deducciones (el motor de inferencia). Esta separación tiene dos grandes ventajas: la primera es que los datos pueden ser ampliados, suprimidos o modificados con gran facilidad; la segunda es que el motor de inferencia puede ser un programa general, válido para cualquier sistema experto. Esta validez general le hace ser un programa sumamente complejo, pero se construye de una vez por todas.

La programación de este tipo de sistemas se vio facilitada con la aparición de los lenguajes declarativos, como el PROLOG, creado por A Colmerauer en 1975. Con este tipo de lenguajes el programador puede olvidarse bastante de la forma de resolver un problema, dedicándose casi exclusivamente a describirlo. Este tipo de lenguajes junto con las nuevas arquitecturas de ordenadores, formadas por varios cientos de procesadores, seguramente supondrán un significativo avance en IA, cuyas consecuencias todos habremos de notar en no muchos años.

Ofites Informática

Presenta: la tableta gráfica

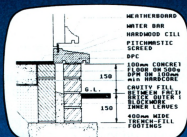
GRAFPAD II-

LO ÚLTIMO EN DISPOSITIVOS DE ENTRADA DE GRÁFICOS PARA AMSTRAD, COMMODORE Y BBC

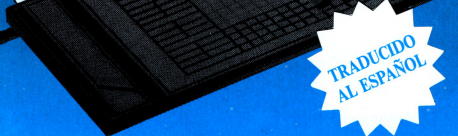
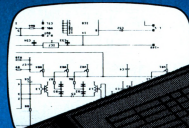
La primera tableta gráfica, de bajo costo, en ofrecer la duración y prestaciones requeridas por las aplicaciones de negocios, industria, hogar y educación. Es pequeña, exacta y segura. No necesita ajustes ni mantenimiento preventivo. GRAFPAD II es un producto único que pone la potencia de la tecnología moderna bajo el control del usuario.



DIBUJO A MANO ALZADA
SOFTWARE DE ICONOS



DISEÑO DE ARQUITECTURA
CON SOFTWARE DDX



TRADUCIDO
AL ESPAÑOL

COMBINA EN UN ÚNICO DISPOSITIVO TODAS LAS PRESTACIONES DE LOS INTENTOS PREVIOS DE MECANISMOS DE ENTRADA DE GRÁFICOS. LAS APLICACIONES SON MÁS NUMEROSAS QUE EN LOS DEMÁS DISPOSITIVOS COMUNES E INCLUYEN:

- selección de opciones
- entrada de modelos
- recogida de datos
- diseño lógico
- diseño de circuitos
- creación de imágenes
- almacenamiento de imágenes
- recuperación de imágenes
- diseño para construcción
- C.A.D. (diseño asistido por ordenador)
- ilustración de textos
- juegos
- diseño de muestras
- educación
- diseño PCB

ESPECIFICACIONES

RESOLUCION:

1.280 x 1.024 pixels.

PRECISION:

1 pixel.

TASA DE SALIDA:

2.000 pares de coordenadas por segundo.

INTERFACE:

paralelo.

ORIGEN:

borde superior izquierdo o seleccionable.

DIMENSIONES:

350 x 260 x 12 mm.

DISPONIBLE AMSTRAD:

CASSETTE 23.900 ptas.

DISCO 25.900 ptas.

(IVA NO INCLUIDO)

- FACIL DE USAR.
- TRAZADO PCB.
- C.A.D.
- AREA DE DISEÑO DIN A4.
- COLOR EN ALTA RESOLUCION.
- USO EN HOGAR Y NEGOCIOS.
- VARIEDAD DE PROGRAMAS DISPONIBLES.
- DIBUJO A MANO ALZADA.
- DIAGRAMAS DE CIRCUITOS.

DE VENTA EN LOS MEJORES COMERCIOS DE INFORMÁTICA
Si Vd. tiene alguna dificultad para obtener la tableta gráfica, puede dirigirse a:



Avda. Isabel II, 16 -8º
Tels. 455544 - 455533
Télex 36698
20011 SAN SEBASTIAN

CONDICIONES ESPECIALES PARA DISTRIBUIDORES

FACTURACION

Autor: **Fco. Javier Barceló T.**

Hasta la fecha, por la sección Para... PCW han pasado varios programas, la mayoría de los cuales ya eran conocidos en versiones para otros ordenadores, y casi todos realizados fuera de nuestras fronteras. Pero el Amstrad PCW 8256 cuenta cada día que pasa con más software realizado en España, y precisamente hoy pasamos por el banco de pruebas uno de los de más reciente aparición: FACTURACION.



Un programa de facturación es aquel que gestiona tanto la impresión de facturas, como la clasificación de dichas facturas, de manera ordenada. La entrada en vigor del Impuesto Sobre el Valor Añadido hace totalmente necesario controlar perfectamente tanto las facturas emitidas como las recibidas. Los programas de facturación, usualmente, solucionan el problema de las facturas emitidas, proporcionando un control más o menos detallado de las mismas, dependiendo del grado de control de las posibilidades del propio paquete.

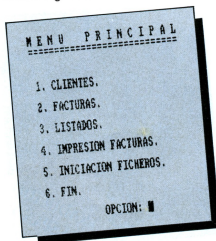
Para analizar este programa, hay que tener en cuenta varios puntos; versatilidad, facilidad de manejo y posibilidades son los que vamos a analizar a continuación.

Versatilidad

Un buen programa debe adaptarse fácilmente a una gran variedad de usuarios, en vez de ser éstos los que se adapten a él. Debe tener previstas la mayoría de las necesidades de los usuarios medios.

Este programa de facturación permite almacenar el nombre, DIF dirección, teléfono del usuario, y los sitúa en las facturas como cabecera del papel. Esto se hace la primera vez

que se utiliza el programa, y no se podrá modificar posteriormente. No es, pues, necesario disponer de papel especial para imprimir las facturas, aunque si se tiene, se pueden dejar en blanco los campos que figuren en los impresos, y así sólo imprimirá los que se necesiten. La incorporación del CIF resulta especialmente útil, dado que es imprescindible ponerlo en las facturas, dada la legislación vigente.

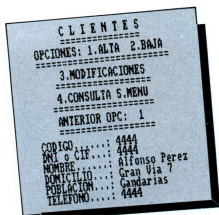


Además, este programa dispone de un fichero para tener almacenados los clientes principales, para evitar tener que teclear sus datos cada vez que se haga una factura. Dentro de estos datos, también está previsto almacenar su CIF o bien su DNI. Aparte de esto, a cada cliente se le asigna un código alfanumérico, que es el dato usado para localizarlos. El resto de los datos, nombre, domicilio y teléfono, son más que suficientes para tener clasificado a cada cliente. Para los no habituales, se pueden introducir los datos manual-

mente, sin necesidad de darle un código, ni almacenarlo en el disco.

El formato de las facturas es estándar, admitiendo hasta 30 artículos por factura, y de cada artículo la cantidad, descripción, PVP, descuento, IVA y calculando automáticamente el total en base a estos datos.

Resumiendo este apartado, el programa es versátil y se adapta sin problema a cualquier proceso normal de



facturación, llevando un control de facturas eficiente.

Manejo

El programa viene acompañado «simbólicamente» por un pequeño cuaderno de instrucciones, dada su filosofía de que las pantallas están muy claras. Con esto último es fácil estar de acuerdo, dado que efectivamente lo están, pero nunca viene mal explicar las cosas un poco más, particularmente el dar la clave de Pagado o Anulado a una factura, que no queda nada clara.

FACTURAS

FECHA : 01/05/86 NOMBRE : Alfonso Perez
 COD. CLIENTE : 4444 DOMICILIO : Gran Via 7
 FACT No... : 000002
 DNI o CIF : 4444
 PUB. : Gandarias

CANTIDAD	CONCEPTO	PVP	DTO	%IVA	TOTAL
10	Ordenadores Amstrad PCW 8256	129000	25	12	1083600
10	Cajas de Diskettes	11000	25	12	92400
10	Cartucho de Tinta Impresora	1900	12	12	18726
2	Papel continuo Facturas	2200	12	12	4336

00000

000000 00 00 00000000

La unidad es \$:

El disco del programa viene preparado para arrancar el ordenador, con el sistema incluido en él. No sólo es útil, sino que además es conveniente hacerlo así, dado que alguna vez que no se ha arrancado con él, sino con el disco original de CP/M, ha dado algún problema de impresora. Esto no es raro, dado que el programa de arranque cambia ciertos valores por defecto de la misma. Los programas vienen en la primera cara del disco, de la que —por cierto— no se puede hacer copia de seguridad, por estar protegida. Esto puede evitar el «pirateo», pero a costa de arriesgar al usuario a alguna incomodidad. La segunda cara es la destinada a los ficheros, y aquí naturalmente sí permite copias de seguridad. Aparte de esta cara del disco, se pueden utilizar los discos que sean necesarios para almacenar ficheros, y se puede optar además al cambiar de disco por poner el número de factura a cero, o bien continuar con el siguiente al último.

Opciones principales

El manejo del programa es fácil. Tiene un menú principal, donde podemos elegir una de las cinco opciones que posee. Estas son:

Clientes: Permite dar altas, modificaciones, bajas y consultas de clientes. De cada cliente pide un código, alfanumérico, que es el que luego sirve para localizarlo. Además, su nombre, domicilio, población, teléfono y DNI o Código de Identificación Fiscal.

Facturas: Permite introducir las facturas. Pide la fecha y el código de cliente. Si se le da, busca en el archivo y rellena los datos siguientes automáticamente. Si no se le da, pide que se introduzcan manualmente sus datos, para la factura. Después, se introducen las partidas que componen la factura, hasta un máximo de 30, calcula el Total bruto, el Descuento, la cantidad de IVA, y el Total Neto a cobrar. Permite imprimir la factura o continuación, o dejarlo para más adelante y seguir introduciendo facturas. En la ilustración de este artículo, se puede ver un ejemplo de factura realizado por este procedimiento.

Para... PCW

Listados: Permite hacer listados de clientes y facturas. En todos los casos los listados pueden ser totales o parciales, y en las facturas se pueden seleccionar por fecha, código de cliente, número y código de pago. **(A. Anuladas, P. Pagadas)** El listado se puede enviar a la impresora, o simplemente hacerlo por pantalla. El listado de facturas incluye al final del mismo, los totales de cada apartado.

Impresión de facturas: Opción pensada por si el proceso de hacer la factura y el de imprimirla se quieren hacer separados, y aunque no quede claro en el manual, también se puede usar para dar a la factura un código de pago. Después de pedir el intervalo de facturas que se desea imprimir, presenta la primera de ellas en pantalla. Si ésta tiene ya código, aparece en la pantalla PAGA-DA o ANULADA. Si no tiene código no aparece nada, pero en cualquier caso, después aparece el cursor, y ahí hay que poner el código de pago.

Iniciación: Esta opción se elige para iniciar una cara de un disco formateado, de manera que se pueda empezar a continuar con el proceso si se llena un disco de datos.

FACTURAS

FECHA : 01/05/86 NOMBRE : Alfonso Perez
 COD. CLIENTE : 4444 DOMICILIO : Gran Via 7
 FACT No... : 000002
 DNI o CIF : 4444
 PUB. : Gandarias

CANTIDAD	CONCEPTO	PVP	DTO	%IVA	TOTAL
10	Ordenadores Amstrad PCW 8256	129000	25	12	1083600
10	Cajas de Diskettes	11000	25	12	92400
10	Cartucho de Tinta Impresora	1900	12	12	18726
2	Papel continuo Facturas	2200	12	12	4336

TOTAL BRUTO..... *\$1,423,400
 TOTAL DESCUENTOS: ***352,888
 TOTAL NETO..... *\$1,070,512
 TOTAL IVA..... ***128,471
 TOTAL A COBRAR... *\$1,198,983

IMPRIMO FACTURA?(S/N)

La unidad es \$:

LISTADOS
=====

1. Clientes * 1. Pantalla * 1. Total
2. Facturas * 2. Impresora * 2. Parcial
3. Menu anterior * 3. Retorno list. * 3. Retorno OPC :
OPC : █ * OPC : █ * OPC : █

CLIENTES

Codigo de a * Codigo de a
* Fecha de a
* Clave(A/P): a

FACTURAS

FACT	COD	FECHA	T.VENTA	T.DESCUENTO	T.IVA	T.NETO
000001	1111	01/05/86	2,475,000	510,000	235,000	2,200,000
000002	2222	01/05/86	1,430,000	352,000	129,470	1,198,030
000003	3333	01/05/86	1,430,000	85,000	38,112	315,712
000004	4444	01/05/86	1,622,000	284,000	800	113,050
000005	5555	01/05/86	139,000	27,000	2,704	113,304
000006	6666	01/05/86	100,000	20,000	4,800	94,000
000007	5/A	06/05/86				

TOTAL BRUTO.....: ***5,507,450
TOTAL DESCUENTOS: **1,177,608
TOTAL NETO.....: *****3,329,842
TOTAL IVA.....: *****503,687

PARA SEGUIR:pulse enter La unidad es A:

Permite iniciar el fichero de clientes, de facturas sin poner el contador de éstas a cero y poner dicho contador a cero. De esta manera se puede continuar introduciendo facturas con número correlativo, pero habiendo borrado las anteriores, o simplemente disponer de varios discos de datos.

La presentación del programa es simple, pero resulta más que suficientemente efectivo. La localización de los clientes es rápida, y aunque la impresión de facturas no es muy veloz, las opciones proporcionan toda la in-

formación necesaria para llevar una gestión correcta y legal.

Posibilidades

Las posibilidades de este programa son más que suficientes. Permite imprimir las facturas clasificadas por el código de pago, saber la situación de determinado cliente, el detalle de los descuentos efectuado, el detalle del IVA repercutido, el total bruto y neto facturado, y todo esto separado por fechas, clientes o números de factura. La estructura del programa

permite disponer de varios discos con archivos de facturas, y el número de factura, que es automático, permite hasta 999.999 facturas.

Conclusiones

Los listados que proporcionan son completos, aunque cuando se pide el listado de facturas pagadas o anuladas quizá debería especificar en el papel qué tipo de facturas está listando. Pero es sólo un detalle. Otro detalle incómodo, es que en determinados campos desactiva la tecla de borrado y la de cursor a la izquierda, aunque siempre pide conformidad antes de introducir una información, por lo que siempre se puede remediar. No obstante, sería más fácil haber permitido el funcionamiento de dichas teclas siempre. De todas maneras son dos pequeños detalles negativos dentro de lo que es un buen programa. Y un consejo. Una vez inicializado el programa, poner la protección contra escritura, para evitar «accidentes».

INICIACION

1. Clientes.
2. Facturas.
3. Contador facturas a cero.
4. Menu principal.

OPC : █

NOTA: Recuerde que los datos anteriores serán borrados

FICHA DEL PROGRAMA FACTURACION

Ordenador: CPW 8256 y CPC6128

Sistema operativo: CPIM 3,0

Equipo mínimo:

Ordenador + impresora + 1 unidad de disco

Precio: 15.300

Distribuido por:

Informática Grotur, S. A.
Jaime el Conquistador, 27
28045 Madrid
Tel. 474 55 00

FECHA : 01/05/86

FACT No : 000003 DNI o CIF : 4444
COD. CLIENTE: 4444 NOMBRE: Alfonso Perez PCB : Gandarias
DOMICILIO: Gran Vía 7

CANTIDAD	CONCEPTO	PVP	DTO	IVA	TOTAL
2	Ordenador Amstrad CPC 6128	89000	20	12	111104
2	Impresora Astron 1400	62000	20	06	9528
1	Caja discos	11000	20	12	39200
1	Disco de Silicio	35000	00	12	

TOTAL BRUTO.....: *****348,000
TOTAL DESCUENTOS: *****402,600
TOTAL NETO.....: *****289,400
TOTAL IVA.....: *****435,720
TOTAL A COBRAR: *****319,120

RUN "TASWORD"

© 1984 TASMAN SOFTWARE LTD.

TASMAN
SOFTWARE

Tasman

SOFTWARE

por fin en España, software a precios británicos

TASWORD

Se imagina su ordenador convertido en una máquina de escribir? TASWORD es la mejor relación calidad-precio en tratamiento de texto profesional.

Totalmente en castellano, permitiendo realizar MAIL MERGE, trabajar en bloques sin ninguna interrupción incrementando su velocidad, etc... (en versión 6128 aprovecha las 128 K creando un disco virtual de 64 K).

- Acentos, ñ, ü, ¿, etc...
- Compatible Productos TASMAN.
- Adaptación impresoras.
- Configuración propia por usuario.
- Ensamblaje de textos.



9.900 pts.

AMSTRAD
COMMODORE
EINSTEIN
MSX



6.900 pts.

AMSTRAD
COMMODORE
MSX
SPECTRUM



7.900 pts.
SPECTRUM

TAS-SPELL

Primer auxiliar que corregirá la ortografía de sus escritos y pondrá los acentos olvidados no dando margen a ningún error. Contiene un potente diccionario con más de 20.000 vocablos pudiendo Vd. ampliarlos. Complemento ideal para su TASWORD con disco.



7.600 pts.

AMSTRAD

Próximamente
en versión PCW 8256
8512

TAS-PRINT

Con TAS-PRINT la escritura elevada a arte. Utiliza las grandes posibilidades gráficas de su ordenador. Las posibilidades tipográficas las explota al máximo al dar una doble pasada optimizando la calidad.

Los tipos de escritura son: **COMPACTA** **MEDIAN** **DATA-VALUE**
LECTURA LIGHT **POUNDER** **SCRIPSY**



7.600 pts.

AMSTRAD
EINSTEIN



5.900 pts.

AMSTRAD
SPECTRUM



6.900 pts.

QL
SPECTRUM

TASCOPI

Sin necesidad de un PLOTTER podrá obtener sus gráficos de pantalla a través de la impresora. Un increíble ZOOM le permite realizar sus gráficos en 4 hojas formando un póster de gran tamaño.



7.600 pts.

AMSTRAD



5.900 pts.

AMSTRAD
SPECTRUM



6.900 pts.

QL
SPECTRUM

GRAFMAN

Programa de E.G. Computer Graphics especialmente diseñado para trabajar conjuntamente con TASCOPI representando las funciones matemáticas en desarrollo de diagramas por coordenadas, permitiendo su efecto "ZOOM" ampliar sectores de dichos diagramas.



5.600 pts.

SOLO AMSTRAD



6.200 pts.

• IVA NO INCLUIDO

TOTALMENTE
EN
ESPAÑOL



DE VENTA EN LOS MEJORES COMERCIOS DE INFORMATICA
Si Vd. tiene alguna dificultad para obtener los programas, puede dirigirse a:

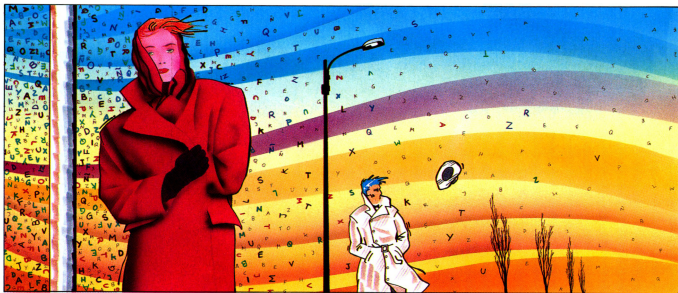
Ofites Informática

Avda. Isabel II, 16 - 8º
Tels. 455544 - 455533
Télex 36698
20011 SAN SEBASTIAN

CONDICIONES ESPECIALES PARA DISTRIBUIDORES
EDITOR Y DISTRIBUIDOR EXCLUSIVO PARA ESPAÑA Y PORTUGAL

NUEVAS FUNCIONES DE TRATAMIENTO DE PALABRAS

Recordará, que no ha mucho, acampábamos en el reino de las cadenas, quizá la más codiciada región de BASICLAND, para conseguir su dominio. Vamos a dar un gran paso en este sentido, estudiando el uso avanzado de las variables alfanuméricas.



H

abíamos visto casi todas las sentencias que se relacionaban de forma directa con el tratamiento de las cadenas. Sentencias que, por su particular actuación, reciben el nombre de sentencias de librería. Recordará que hablábamos de una instrucción muy especial, MID\$, que tiene una actuación bivalente, como función o como orden.

Diferencias entre órdenes y funciones

Lo primero sería establecer la diferencia existente entre órdenes y funciones. La cualidad que marca una distancia más clara entre ambas definiciones sería que la función precisa de un argumento, parte sobre la cual se ejerce ésta, provocando un

resultado, numérico o alfanumérico.

Ejemplos claros de función serían: LEN, LEFT\$ o COS. Observe que la función, acompañada de su argumento, no tiene mucho sentido por sí sola. Debemos hacer algo con su resultado, escribirlo, o asignarlo, o compararlo. Los órdenes por el contrario tienen entidad propia en sí mismas, la cualidad común es ¡HAZI, PRINT, RUN o GOTO serían claros ejemplos de este subconjunto de las sentencias del Basic.

El hecho de que una misma palabra del Basic pueda ser una función o una orden no debería incomodarlo, de hecho, en castellano no se confunde, a no ser en una situación carente de contexto, el verbo «haz» con el sustantivo «haza», de luz. Habíamos visto el uso de MID\$ como orden, recordará que:

MID\$(A\$, 3, 2) = «df»

Tenia como efecto la transformación de A\$ en una cadena idéntica a lo que era A\$, salvo en los elementos 3 y 4 (a partir de 3, 2 elementos), en los que tendría los caracteres «df» y «fa».

El uso de MID\$ como función es

muy sencillo: no es sino una sofisticación de la ya conocida función LEFT\$.

MID\$(A\$, n, m), lo que hace es generar una subcadena de A\$ tomando de esta m elementos a partir del elemento n.

¿Todo claro? Bien, pues hay va el primer reto, intente comprender la diferencia existente entre:

10 b\$ = "1234"

20 a\$ = "abcdefghi"

30 b\$ = MID\$(a\$, 3, 4)

Y:

10 b\$ = "1234"

20 a\$ = "abcdefghi"

30 MID\$(a\$, 3, 4) = b\$

Si ha comprendido esta sutil diferencia el MID\$ es suyo.

Funciones VAL y STR\$

Recordemos ahora, las otras dos instrucciones que, de la ya mencionada lista, nos quedaban por ver, VAL y STR\$.

Son dos funciones totalmente duales, con lo que la comprensión de una le llevará fácilmente al entendimiento de la otra. Comencemos por

VAL. La misión de esta función es transformarnos una variable alfanumérica en numérica, naturalmente, hasta donde sea posible; me explico: supongamos que tenemos una variable alfanumérica AS="123dsd12". Si ahora hiciésemos P=VAL(AS), el valor de P no sería sino: 123. Si el primer carácter encontrado en la cadena alfanumérica no fuera numérico, obtendríamos como resultado 0.

STR\$, como anunciábamos, realiza la operación contraria. Si tenemos P=123 y hacemos AS=STR\$(P), el valor de AS tras esta asignación sería, "123". Nótese la diferencia entre los contenidos de P y AS: P vale ciento veintitrés, y AS vale uno, dos, tres.

Como ejemplo práctico de utilización así como de utilidad, nos remitimos a los programas II y III.

En el programa II suponemos que intentamos llevar el control de ciertos productos vendidos, mediante sus facturas. Utilizaremos para ello una variable alfanumérica, en los primeros caracteres pondremos el importe de la factura y los tres últimos caracteres los usaremos para un código de producto al que hacen referencia. Así FACTS="109070ORD", sería la factura correspondiente a un ordenador cuyo importe asciende a

está perfectamente preparado para aprender todas las funciones nuevas que usted esté preparado para enseñarle, en un número no superior a cuarenta. A este punto nos vamos a referir ahora: cómo se definen funciones y cómo se utilizan. Aunque detallaremos el empleo de éstas para alfanuméricas no existe diferencia formal con la definición de funciones numéricas. DEF FN y FN

```
10 REM ** programa 2 **
20 REM ---DEMOSTRA EL USO DE VAL---
30 CLS
40 fact$="1090100RD"
50 PRINT fact$
60 importe=VAL(fact$)
70 REM--calculamos el 10%--
80 importe =importe+(importe/100)*1
90 PRINT "NUEVO IMPORTE (+10%):" ;a$
importe
10 REM ** programa 3 **
20 REM ---DEMOSTRA EL USO DE STR$---
30 IMPORTE=109911
40 PRINT importe
50 CODIGOS="ORD"
60 PRINT codigo$
70 IMP=STR$(importe)
80 fact$=imp$;codigo$
90 PRINT fact$
```

Como ya habrá intuido la primera de estas sentencias nos va a servir para definir nuestras funciones. Veamos cuál es su sintaxis:

DEF FNnombre [S] (arg1, arg2, ..., argN)=operaciones con estos argumentos.

Suponga que su CPC desconoce la función LEFTS y que va a utilizar muchas veces esta función. Nosotros disponemos de otra función MIDS, a partir de la cual podemos perfectamente sustituirla, utilizando como primer parámetro, 1, en forma de constante. Es decir:

MIDS (AS, 1, N)

Es lo mismo que:

LEFTS (AS, N)

Bien, definir ahora la función LEFTS, es tarea fácil:
DEF FN1efS (AS, N)=MIDS (AS, 1, N)

```
10 REM ** programa 4 **
20 CLS
30 REM ---USO DE STRING$---
40 FOR n= 33 TO 250
50 PRINT STRING$(120,n)
60 NEXT n
```

Observe que los argumentos de esta función, AS y N, 3 son alfanumérico y numérico respectivamente. Si ahora quisiésemos extraer los 4 primeros términos de PS y asignarlos a HS, nos bastaría teclear.

HS=FN1efS (PS, 4)

Es importante, resaltar que los nombres de los parámetros de entrada, AS y N en nuestro caso, no tienen por qué coincidir con el de las

PROGRAMACION

```
10 REM ** programa 5 **
20 REM ---USO DE MID$( ORDEN)---
30 AS="MISION:modificar este TROZO de la cadena"
40 PRINT AS
50 MID$(AS,1,6)="EFFECTO"
60 MID$(AS,23,5)="CACHO"
70 PRINT
80 PRINT AS
```

variables, sobre las que vayamos a aplicar nuestra función. La única condición es respetar el orden establecido en el interior del paréntesis.

¿Entendido? Bien, naturalmente en nuestro caso la función es tan corta que no tiene mucho sentido el definirla, aparte de estar ya definida en el Basic del Amstrad. Sin embargo, basta que tenga dos sumandos para que sea más que probada su utilidad.

```
10 REM ** programa 6 **
20 CLS
30 REM ---USO DE RIGHT$ Y LEFTS---
40 AS="izquierdaderecha"
50 PRINT AS
60 PRINT
70 PRINT "9 caracteres contando desde la izquierda"
80 PRINT LEFT$(AS,9)
90 PRINT
100 PRINT "7 caracteres contando desde la derecha"
110 PRINT RIGHT$(AS,7)
```

Nótese que la \$ detrás del nombre asignado a la función deberá añadirse, siempre y cuando el resultado obtenido mediante la aplicación de esa función sea alfanumérico; si no fuese así, no sólo es innecesario sino causa de error. (Type Mismatch). Esto es igual en las funciones ya definidas en su CPC. Aquellas que entregan un resultado numérico no lo llevan, LEN, INSTR, COS; en contraposición con las que lo dan alfanumérico que sí van etiquetadas con \$, LEFTS, MIDS, STRS.

La utilidad de las funciones

A estas alturas se habrá preguntado ya varias veces para qué diablos puede servirle esta cualidad del Basic de su CPC. Evidentemente, no po-

```
10 REM ** programa 7 **
20 CLS
30 REM ---USO DE LEN---
40 INPUT "Introduzca texto:" ;a$
50 PRINT
60 PRINT "La longitud de su texto e n de"
70 PRINT LEN (a$)
```

demos darle una lista de todas las ocasiones en las que el uso de FNames sea aconsejable. Sin embargo, si po-

```
10 REM ** programa 1 **
20 REM ---USO DE MID$( FUNCION)---
30 CLS
40 AS="vamos a coger este TROZO de la cadena"
50 PRINT AS
60 PRINT
70 PRINT MID$(AS,20,5)
```

109.070 pesetas. Bien, el problema ahora es que ha llegado el tan comentado IVA y a todas nuestras facturas hemos de incrementarlas un tanto por ciento, supongamos un 10 por 100. El programa II realiza esta operación mediante la orden VAL. Ahora deberemos unir nuevamente el importe de la factura, IMP, con el código del producto, COD\$. Labor que realiza el programa III.

Creando funciones

Sin embargo, aún queda una, no mencionada, por explicar, se llama IMAGINACION. No intente teclearla pues se encontrará irremediablemente con el fatídico Sintax error. Hemos hablado de ciertas funciones que el ordenador es capaz de utilizar en el tratamiento de cadenas. Sin embargo, cómo no, su fiel CPC

```

10 REM ** programa 8 **
20 CLS
30 REM ----USO DE SPACE-----
40 FOR n=1 TO 21
50 PRINT STRING$(40,100);
60 NEXT n
70 LOCATE 1,22
80 INPUT "Due línea desas barras:(
1-22)";b
90 LOCATE 1,b;PRINT SPACE$(40)
100 GOTO 70

```

demos ponerle unos ejemplos, donde el uso de ciertas funciones definidas desde el teclado puede llevarnos a un resultado tan elegante como potente.

Existen lenguajes, no es el caso del Basic, en el que es posible definir nuestras propias estructuras de datos, Pascal, Fortran... conforme a las necesidades de nuestro programa. Esto, que tan sumamente complejo puede parecer a simple vista (de hecho su empleo puede serlo) es algo sumamente sencillo de entender.

Una estructura de datos no es sino un conjunto de datos relacionados entre ellos de forma determinada. También dentro de la estructura de datos se incluyen las operaciones que con estos datos vamos a poder realizar. Un ejemplo de estructura de datos que ya conocemos, y que ya está definido en el Basic, son las matrices o arrays. A (1, 1), A (1, 2), A (2, 1) y A (2, 2) formarían una estructura de datos junto a las operaciones +, -, *, /.

Existe un tipo de estructura de datos, que por su utilidad suele creerse, implementarse sería la palabra adecuada, frecuentemente en distintos programas. La cola. Su estructura

```

1 REM ** programa 9 **
10 REM ----FNID$(col$a,n$)-----
20 REM ----INCORPORA UN NUEVO-----
30 REM ----ELEMENTO POR LA DERECHA---
40 DEF FNID$(col$a,n$,n$)=col$a+n$*n$
50 AS=FNID$(a$,C$)
60 INPUT C$
70 AS=FNID$(a$,C$)
80 PRINT AS
90 GOTO 60

```

es muy sencilla, podría compararse a una fila de automóviles en un túnel estrecho. Podemos sacar por un lado y meter por el otro, lo que no podemos es ni eliminar ni meter por el centro. En el caso de que nuestros datos dispusieran de «marcha atrás» podríamos sacar y meter por ambos lados indistintamente.

Dicho esto, seguramente se figurará que la cola que vamos a crear, no va a ser sino una variable alfanumérica que llamaremos cola\$.

Naturalmente, este planteamiento conllevará ciertas limitaciones. La primera será la longitud de la cola, 255 caracteres. La segunda la longitud de cada uno de los datos que conforman la cola, deberá tener una longitud prefijada. En el caso de que esta longitud sea de un carácter, po-

drems almacenar hasta 255 datos, para dos caracteres por dato tendremos una cola de 127 datos como máximo, 83 para tres caracteres y, en general, 255/n para n caracteres por dato.

Por ejemplo si nosotros quisiésemos almacenar los números 32, 45, 56, 1, 23 y 12, podríamos emplear para ello seis variables distintas, pero si nuestras necesidades no nos lo impidiesen podríamos utilizar para ello, una variable alfanumérica que quedaría, si la longitud de nuestros números fuera de dos cifras: COLA\$="324556012312".

Una vez definida la base de nuestro almacenamiento, precisamos crear ciertas funciones, que faciliten el manejo de cualquier cola.

```

1 REM ** programa 10 **
10 REM ----FNID$(col$a,n$)----
20 REM ----INCORPORA UN NUEVO-----
30 REM ----ELEMENTO POR LA IZQUIERDA-
40 DEF FNID$(col$a,n$,n$)=n$+col$a
50 AS=""
60 INPUT C$
70 AS=FNID$(a$,C$)
80 PRINT AS
90 GOTO 60

```

Estas funciones están definidas en la línea 40 de los programas VIII, IX, X y XI, respectivamente. Aunque el uso de estos programas puede ser más que suficiente para ver la utilidad de las funciones, que en ellos se definen, describamos éstos con más detalle.

Los programas

El programa VIII demuestra la utilización de una de las funciones que precisamos en el tratamiento de nuestros datos. La operación que realiza es muy simple, añade un nuevo dato a la cola. Si tenemos definida una cola cuyo nombre es a\$ y hemos de incorporar un nuevo elemento n\$, nos bastará con hacer a\$=fnid\$(a\$,n\$). El programa, al igual que los siguientes, demuestra el uso de ésta, mediante un ejemplo.

La función definida en la línea 40 del programa IX, realiza la misma tarea, con la salvedad de incorporar los nuevos datos por la izquierda.

En los programas X y XI, nos encontramos con una sofisticación de estas dos funciones. La función del programa X aparte de incorporar un nuevo dato por la derecha, elimina uno por la izquierda. Y la del programa XI realiza la operación inversa, incorpora por la izquierda a la vez que elimina por la derecha. Al principio de estos programas se nos solicita la longitud de la cola, la cual,

una vez llena, captará y perderá elementos.

Mediante estas cuatro funciones más la sentencia INSTR, ya conocida y comentada en el artículo anterior, es mucho el juego que podemos dar a estas estructuras. Existirán aún, algunas funciones que podrían permitirnos un tratamiento más cómodo de las colas. Intente definir dos funciones que eliminen el último dato, el de la derecha, y-el primero, el de la izquierda.

Habría observado que la estructura que hemos creado, no permite el uso de datos con más de dos caracteres. Es decir, si los datos fueran numéricos, no podríamos almacenar números superiores a 9. Esta condición, a la altura que nos encontramos no debiera suponerle ninguna tragedia irreparable, intente modificarlo para que sirva para datos de dos caracteres. Naturalmente, si no desea que todo se le vaya al fraste, respete este nuevo formato y si desea almacenar un 3, almacene 03, o si desea almacenar un A almacene junto a la A un espacio ("A").

Un ejemplo clásico de este tipo de estructuras es el BUFFER, un buffer no es sino una cola de espera. Su CPC dispone de varios buffer, el del sonido por ejemplo; si usted exige a su ordenador la ejecución de 7 notas éste carga sus valores en el buffer de sonido y las va enviando al generador de sonido, PSG, a medida que las va ejecutando. De este modo, puede ir realizando otra cosa, no teniendo que estar detenido esperando a que finalice la ejecución de éstas. Existen otros buffer, para el teclado, para la impresora...

Después de toda esta sintaxis teórica se preguntará qué utilidad puede dar a todos estos nuevos conocimientos que se incorporan a su nuevo bagaje informático. Nosotros por nuestra parte poco es lo que podemos ofrecerle en este aspecto. Es usted, junto a su CPC, quien debe decidir como solucionar los problemas que se le vayan planteando, no obstante y para abrirle el apetito, vaya a continuación un pequeño programa que aprovecha este tipo de almacenamiento de datos.

```

1 REM ** programa 11 **
10 REM ----FNID$(col$a,n$)-----
20 REM ----INCORPORA POR LA DERECHA-
30 REM Y EXTRAE POR LA IZQUIERDA---
40 DEF FNID$(col$a,n$,n$)=RIGHT$(c$)
50 AS=FNID$(a$,n$)
60 INPUT "longitud de la lista ";l
70
80 AS=SPACE$(l*n)
90 INPUT "NUEVO ELEMENTO:";n;n$=LEFT$(n$,l)
100 AS=FNID$(a$,n$)
110 PRINT AS
120 CLS
130 GOTO 90

```

Uso de «colas» como estructuras de datos (queues)

Amnesia

Para explicar la utilidad de este tipo de estructuras vamos a imaginar un sencillo juego.

Un jugador dice una letra del alfabeto y el otro le contesta con ella, el juego termina cuando uno de ellos repite una letra, lo cual le descalifica.

La labor de nuestro **Amstrad** va a ser, en esta ocasión, doble. Ejercerá como es natural la función de jugador y por otro la de árbitro llevando el control de las letras ya dichas y sancionando las repeticiones.

Tenemos tres formas para abordar esta empresa. La primera sería que fuese sacando una letra al azar cada vez que fuese su turno.

Sin embargo, esto no sería muy justo por nuestra parte. Fácilmente lograríamos alzarnos con la victoria ante la incommensurable estupidez de nuestro **Amstrad**. La segunda estrategia sería aprovechar el hecho de que en todo momento, nuestro CPC, conoce las letras ya utilizadas, impidiéndole, previa consulta, las repeticiones. En esta ocasión seríamos nosotros los que estaríamos totalmente fuera de juego, nuestro programa sería un verdadero experto, permitiéndonos como mucho, si su memoria es superior, las tablas.

La tercera forma, la que vamos a elegir en el desarrollo de nuestro programa, consiste en basarnos en una ley que nosotros supondremos axiomática: «**Olvido primero lo primero que aprendo.**» Es decir, aplicada a nuestro caso, si se dicen las letras P, D, N, T, W, K, L y R, por este orden la primera que olvidaré, si he de olvidar alguna, es la P.

La forma en que crearemos esta memoria pseudohumana, será apoyándonos en las cola\$.

Definiremos dos de estas estructuras, una en la que iremos anotando todas las letras que se digan, tanto por parte nuestra, como por parte de nuestro **Amstrad**. Mediante esta cola podremos en todo momento saber si las letras que se vayan diciendo en el transcurso del juego, han aparecido ya en algún momento.

La segunda cola que definiremos será utilizada como la memoria que queremos simular. Esta tendrá una longitud máxima de forma que podremos controlar el número de letras que nuestro CPC va a poder recordar en cada momento del juego. Dada la corta longitud del programa, tecléelo y úselo unas cuantas veces antes de intentar comprenderlo. Para hacer que la cola, «memoria» de nuestro ordenador, utilizaremos la función, antes definida, que realiza la inserción por la derecha, perdiéndose un elemento por la izquierda. Veamos de forma somera y rápida el desarrollo del programa.

En las líneas 30 y 40 se definen las funciones que vamos a necesitar. Recuerdese que FNid\$ (cola\$, neo\$) incorpora el elemento neo\$ a la cola, mientras que FNder\$ (cola\$, neo\$) incorpora el elemento neo\$ a la cola, a la vez que le hace perder su primer elemento.

En las líneas 60 a 90 se prepara el juego, se pone la pantalla en modo 0, caracteres grandes, y se nos solicita el número de elementos que nuestro CPC va a «poder recordar». En la línea 90 se crea la variable memoria\$, donde se guardarán estos elementos, con una longitud igual al número de elementos elegido. Inicialmente y como era de esperar, ésta se encuentra vacía (formada por espacios blancos).

En la línea 100 y 110 se nos solicita la letra que queremos, decir. La misión de la línea 110 es convertir el carácter en mayúsculas y comprobar que éste es una letra, mediante los códigos ASCII, en caso de no serlo, la jugada será invalidada solicitándonos una nueva letra.

Mediante la línea 130 comprobamos que la palabra que decimos no se encuentre en dicha\$, segunda de las anunciadas colas. De ser así la variable mensaje se cargará con «**TU PIERDES**», enviando la ejecución a la línea 230, donde se dará el juego por finalizado tras presentar en pantalla el contenido de mensaje\$.

En la línea 140 incorporamos esta nueva letra, elegida por nosotros, a la memoria de nuestro CPC, eliminando el elemento más antiguo.

En la línea 150 añadimos a dicha\$ la nueva letra.

Ahora le toca jugar a nuestro **Amstrad**, en la línea 160 elige una letra, realmente elige un número al azar entre 65 y 90, transformándolo posteriormente en el símbolo, CHR\$, al que representa.

Mediante la línea 170 nuestro CPC consulta si «**recuerdo**» que esta letra estuviese ya dicha. Es decir, inspecciona si está contenida en memoria\$, de ser así volverá a la línea 160 para elegir una nueva letra.

En el caso de no «**recordar**» que esta letra hubiese sido dicha se arriesga a decirlo (línea 180), mirando posteriormente si ésta se había o no dicho (línea 200), antes en la línea 190 habrá incorporado esta nueva letra en su «**memoria**», «olvidando» otra de las letras dichas.

Una vez verificado si está en dicha\$, de estarlo la ejecución se envía a la línea 230, con la variable mensaje\$ cargada con «**YO PIERDO**»; apunta esta nueva letra en la cola dicha\$ mediante la función FNid\$.

En la línea 220 la ejecución es enviada a la línea 100 para proceder a la siguiente jugada.

Como ven el desarrollo del juego es bastante fácil de seguir, pueden incorporar en el programa dos órdenes para imprimir las variables dicha\$ y memoria\$, pudiendo observar de este modo su evolución.

Al final

Hasta aquí todo lo que puede decirse sobre las cadenas alfanuméricas y su tratamiento, el resto ya no es hablar sino hacer.

Naturalmente aquellas cosas que pueden hacerse con funciones definidas por el usuario también pueden hacerse de otras formas. En programación aunque no todos, si muchos caminos conducen a Roma, el caso está en no tener que pasar por Nueva Delhi.

Por otra parte, la elegancia, rapidez y economía deben ser puntos siempre presentes en la programación y sin duda el uso de funciones definidas desde el teclado es una buena arma para el logro de este objetivo.

Si ha llegado a esta línea, enhorabuena, el reino de las cadenas es suyo.

```
10 REM--AMNESIA--
20 REM-definición de funciones
30 DEF FNid$(cola$,neo$):cola$=neo$
40 DEF FNder$(cola$,neo$):col$=RIGHT$(cola$,LEN(cola$)-1)+neo$
50 REM--inicio--
60 MODE 0
70 INPUT "¿CUANTAS LETRAS QUIERE RES O LE RECUERDE (0-20) ?":n
80 memoria$=""SPACE$(neo)
90 CLR
100 LOCATE 7,10:INPUT "TU JUGADA":
110 tu$=UPPER$(tu$):IF ASC(tu$)<65 OR ASC(tu$)>90 THEN 100
120 tu$=LEFT$(tu$,1)
130 IF INSTR(dicha$,tu$)>0 THEN mensaje$="TU PIERDES":GOTO 230
140 memoria$=FNder$(memoria$,tu$)
150 dicha$=FNid$(dicha$,tu$)
160 yo$=CHR$(ASC("A")+RND(26))
170 IF INSTR(memoria$,yo$) THEN 100
180 LOCATE 7,12:PRINT "MI JUGO :";yo$
190 memoria$=FNder$(memoria$,yo$)
200 IF INSTR(dicha$,yo$) THEN mensaje$="YO PIERDES":GOTO 230
210 dicha$=FNid$(dicha$,yo$)
220 GOTO 100
230 REM--fin del juego--
240 CLR
250 LOCATE 12,12:PRINT mensaje$
```

MERCENARIO

© ALLIGATA SOFTWARE

Vive una gran aventura.

Cada pantalla es un nuevo desafío al peligro, una misión suicida, una dura batalla. Te encontrarás solo frente a compañías de asalto, aviones, morteros, bombas de mano, etc., que emplearán toda su fuerza y material bélico para frenar tu avance endemoniado. Hace falta mucha sangre fría mucha agilidad mental para librarse del ataque más furioso que jamás hayas visto.

Cada paso es un cuerpo a cuerpo.

para

AMSTRAD

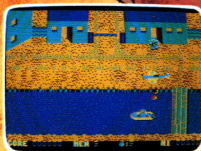
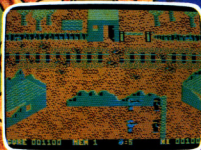
cassette: 2.000,- Ptas. (+I.V.A.)

disco: 2.900,- Ptas. (+I.V.A.)

PRODUCE Y
DISTRIBUYE

microSTE

P. CASTELLANA, 179-1.º - 28046 MADRID Tel. 442 54 32/44



SOFTWARE de muchos rombos, para mayores

TOTALMENTE EN ESPAÑOL

C Compilador C

Versión completa del famoso C-Hisoft para CP/M. Capacidades de E/S, ficheros aleatorios y modos de acceso binario y ASCII. Incluye editor ED 80 compatible WORDSTAR.

**15.000
pts.**

PASCAL 80 Compilador Pascal

Especial para Z-80. Deja el programa fuente en un programa directamente ejecutable. Incluye ED 80, editor compatible con WORDSTAR.

**15.000
pts.**

KNIFE Editor sectores

Permite trabajo directo sobre disco, bien en hexadecimal o ASCII, recuperar ficheros perdidos o borrados, alterar y/o proteger directorios, todo bajo AMSDOS y CP/M.

**7.900
pts.**

DEVPAK 80 Ensamblador/des

ED 80: Editor Configurable GEN 80: Macros, inclusión en disco, ensamblador condicional, manipulación bit a bit. MON 80: Monitor y debugger, puntos de ruptura y presentación de memoria.

**15.000
pts.**

MODULA-2 Comp. Modula-2

Implementación total del lenguaje MODULA-2 para CP/M. Compilador en un único paso, listo para ser linkado.

**19.900
pts.**

TORCH Tutor de CP/M

Diseñado específicamente para AMSTRAD. Incluye THE WAND, creador de menús de programas.

**7.900
pts.**

POLYPRINT Multitipos

Transforme su impresora en una imprenta. Permite la impresión en 8 tipos distintos de letras; configurable para cualquier impresora.

**19.900
pts.**

POLY TYPEFACES Multitipos

Añade a la potencia del programa POLYPRINT 8 juegos adicionales de impresión a los ya existentes.

**7.900
pts.**

WRITE HAND MAN Sidekick en CP/M

Residente en memoria, sin interferir en su programa principal le ofrece: Calculadora (Hex-Dec), Block de notas y teléfonos, Calendario, Directorios, etc...

**11.900
pts.**

POLYPLOT Impresora/Plotter

Permite realizar gráficos sofisticados en su impresora. Gráficos de pastel, histogramas comparativos, gráficos de líneas, imágenes de 980 PIXELS de densidad.

***11.900
pts.**

POLYMAIL Mailing

Sencillo sistema de MAIL-MERGE. Idóneo para producir circulares. Incluye editor. Permite la realización de etiquetas autoadhesivas.

***9.900
pts.**

CATALOG Clasificador

Asigna a cada disco un número de serie y además indexa y cataloga los ficheros en ese disco.

**8.900
pts.**

MULTI-TEXT Módulo de textos

Módulo de textos, preparado para ser empleado con nuestro lápiz óptico ESP o con las teclas de cursor.

***10.900
pts.**

FIRST STEPS Tutor de Newword

Explore las enormes capacidades del procesador de textos NEWWORD; guiado desde los fundamentos del proceso de textos.

**7.000
pts.**

MASTER LOCOSCRIP

Dos cintas audio con instrucciones claras para aprendizaje y apoyo al manual del tratamiento de textos LOSOSCRIP.

**3.000
pts.**

DRAUGHTS- MAN II

Nueva versión mejorada y compatible con nuestra tableta GRAFPAD II: Gran capacidad en gráficos.

**6.200
pts.**

*** los 4 juntos
23.800 pts.**

TYPING CRASH COURSE Inicia a telear

Curso de iniciación a los teclados, recomendado para personas no acostumbradas a su uso.

**9.900
pts.**

TWO FINGERS Curso mecanográfico

Conozca a fondo las posibilidades del teclado, escribiendo con sus diez dedos en lugar de sólo dos.

**9.900
pts.**

**IVA
no
incluido**



DE VENTA EN LOS MEJORES COMERCIOS DE INFORMÁTICA
Si Vd. tiene alguna dificultad para obtener los programas, puede dirigirse a:

**Ofites
Informática**

Avda. Isabel II, 16 - 8º
Tels. 455544 - 455533
Télex 36698

CONDICIONES ESPECIALES PARA DISTRIBUIDORES
EDITOR Y DISTRIBUIDOR EXCLUSIVO PARA ESPAÑA

SINTETIZADOR DE VOZ EN CASTELLANO MHT...



PORQUE A SU AMSTRAD SOLO LE FALTABA HABLAR

Utilizable con los modelos AMSTRAD, CPC 464, CPC 664 y CPC 6128, el programa que controla este sintetizador, contiene las reglas básicas de pronunciación en castellano y permite su funcionamiento, tanto en modo directo, como bajo el control de un programa.

Software de manejo:

Presentado en cinta, tiene la posibilidad de copiarlo en disco o en otra cinta para poder realizar copia de seguridad.

Viene preparado para poder usarlo desde:

- Basic: Genera cuatro comandos de muy fácil uso, cubriendo todas las necesidades de manejo.
- Código máquina: Esta forma de utilización está orientada a aquellos usuarios que poseen un amplio conocimiento del código máquina. Incluye amplificador de sonido con mando de volumen.

MHT Ingenieros

Le esperamos en nuestros stands 9 y 10 de la 1.ª Feria Amstrad, desde el 23 al 25 de Mayo

Sin duda alguna

A través de esta sección se pretende resolver, en la medida de lo posible, todas las posibles dudas que «atormenten» a todas las personas interesadas en el mundo del AMSTRAD, sean o no poseedores de uno y, si lo son, se encuentren en cualquier nivel de destreza en su manejo.

Semanalmente, aparecen en estas páginas las consultas de la mayor cantidad de usuarios posible; ello redundará en un mejor servicio y en un contacto más estrecho entre todos nosotros a través de la revista.

SIN DUDA ALGUNA está abierta a todos.

Soy poseedor de un CPC-664 y me gustaría acoplar un cable para cassette. Aunque me informaron en el establecimiento donde lo compré, veo que los cassettes tienen dos clavijas (micphones) y para el cable, hace falta otro (REM). ¿Dónde se acopla éste? También quería que me informaran sobre «las teclas programables».

Muchas gracias.

J.F.A.B. (Valladolid)

La mayoría de los cassettes normales, no necesariamente adaptados a ordenador, poseen un conector RE-Moto que permite al ordenador en cuestión controlar el motor de la grabadora, como es el caso del **Amstrad**. El cable que vende Indescomp u otras tiendas debe tener tres conectores y la persona que le vendió el cassette debiera estar informado de ello, y, consecuentemente, proporcionarle uno que se adapte a esa especificación.

Aprovecho la ocasión para decirlos que habéis tenido una gran idea al

invertir en el **Amstrad** pues tiene gran futuro dentro de lo que cabe.

Bueno, mi problema es el siguiente:

No hace mucho compré la unidad de disco externa, pues yo tengo el CPC 464 y mi sorpresa fue mayúscula al comprobar que no acepta hacer MERGE trabajando con discos. Mi única salvación es pasar uno de los dos programas que quiero unir a cassette y cargarlos a continuación del primero, que lo hago desde disco.

Tengo entendido y comprobado que no existe este problema en el 664. Y también he oído comentarios en revistas inglesas de que el sistema operativo AMSDOS, en principio fue una chapuza de nuestro amigo ALAN SUGAR.

¿Es esto cierto? y si lo es, ¿cómo puedo solucionar mi problema? Sin más que decirlos, os saluda atentamente.

Joaquín Bueno Caro (Málaga)

Efectivamente tienes razón. El Amsdos del 464 es un poco «chapuza». Sin embargo, tu problema tiene fácil solución, si es que te ocurre lo que nosotros suponemos.

Probablemente suceda que, al darle al disco la orden MERGE, te encuentres con un mensaje de error del tipo «EOF MET», o algo parecido, obteniendo como resultado final de tu intento nada de nada.

En el caso del 464, debes salvar en el disco el programa que quieras «mergear» en formato ASCII, como se especifica en el Manual de Usuario, esto es, con la opción «A». Por ejemplo:

SAVE «PROGRAMA», A

Un programa grabado en disco de esta manera, realizará el MERGE con otro presente en memoria perfectamente, o al menos a nosotros siempre nos ha funcionado hasta ahora.



Corta y pega este cupón en la casilla correspondiente de la página 16 del número 31 de **AMSTRAD Semanal**, una vez completada la página, envíanosla junto con tus datos. ¡SUERTE!

FUNDAS PARA TU «AMSTRAD»

464-472-664 y 6128	2.262
8256	3.250
Joystick Quickshot II	1.975

Pago reembolso, más 250 ptas. de gastos de envío.
Indicar modelo y monitor (verde o color).

Pedidos a: **BAZAR POPULAR**
Apartado 27.040
08080 BARCELONA

OPERACION CAMBIO

Pásate a monitor color por
25.000 ptas.

Valoramos:

Tu Amstrad 464 en 50.000 ptas.

Amstrad 664 en 60.000 ptas.

En la compra de un Amstrad
CPC 6128, PCW8256, PCW8512

Consulte para monitor color
☎ (91) 270 34 97 de 4,30 a 8,30

PUBLICIDAD



New Line

GABINETE DE INFORMÁTICA

- **Clases de Informática sobre AMSTRAD**
En grupos e individuales.
- **Ordenadores AMSTRAD y periféricos**
Las mejores precios
- **Software:** Estándar y a la medida

ZURBANO, 4 ☎ 410 47 63
28010 MADRID

MECA-SCRIB

El Curso de Mecanografía para el AMSTRAD PCW 8256.

¡¡IMPORTANTE PARA ACADEMIAS!!

- Gestión de alumnos.
- Capacidad para 60 alumnos.
en un solo diskette.

Pedidos a:

EDUCOMP, S.A.
C/ Molina de Aragón, 1.
Tel. (911) 22 32 12
19003 GUADALAJARA

Ofites Informática

Presenta:
 el lápiz al que gusta decir **SI**
 mientras nuestros competidores dicen no
UNICO PARA AMSTRAD, CON PRECISION PIXEL

FUNCIONES	ESP	dt trónica	OTROS
UNICO MENU DE PANTALLA	SI	NO	
ARRASTRE OBJETOS PANTALLA	SI	NO	
TRASLADO OBJETOS PANTALLA	SI	NO	
TRASLADO DE CURSOR	SI	SI	
CAJAS ELASTICAS	SI	SI	
LINEA ELASTICA	SI	SI	
TRIANGULO ELASTICO	SI	NO	
ELIPSE ELASTICO	SI	NO	
DIAMANTE ELASTICO	SI	NO	
POLIGONO ELASTICO	SI	NO	
HEXAGONO ELASTICO	SI	NO	
OCTOGONO ELASTICO	SI	NO	
CUBO ELASTICO	SI	NO	
PIRAMIDE ELASTICA	SI	NO	
CIRCUNFERENCIAS	SI	SI	
CIRCULOS RELLENOS	SI	NO	
CAJAS RELLENAS	SI	NO	
ELIPSES RELLENAS	SI	NO	
CUNAS	SI	NO	
SIMULADOR DE CORTES	SI	NO	
DISEÑO DE ZOOM	SI	NO	
IMAGEN ESPEJO E INVERTIDA	SI	SI	
FONDO DE REFERENCIA	SI	NO	
REJILLA DE FONDO	SI	NO	
OPCION DISPLAY X, Y	SI	NO	
RELLENADO CON COLOR	SI	SI	
LAVADO DE COLOR	SI	NO	
VOLCADO PANTALLA RESIDENTE	SI	NO	
DIBUJO DE BORDES EN 3 D	SI	NO	
TEXTO	SI	SI	
9 TAMANOS DE BROCHA	SI	NO	
18 TOBERAS MOSTRADORAS	SI	NO	
4 MEZCLAS BASICAS	SI	NO	
VARIADOR DE MEZCLAS	SI	NO	
SOMBREADO DE MEZCLAS XOR	SI	NO	
FICHERO ICONOS RESIDENTES	SI	NO	
FICHERO RELLENOS RESIDENTES	SI	NO	
26 COLORES DE PAPEL	SI	NO	
PALETA DE 15 TONOS DE COLOR	SI	NO	
POSICIONAMIENTO DE PUNTO	SI	SI	
RAYOS DESDE UN PUNTO FLUJO	SI	NO	
DIBUJO REFLEJADO (ESPEJO)	SI	NO	
FUNCION HOME	SI	NO	
CONTROL DESDE TECLADO	SI	SI	
CONTROL CON JOYSTICK	SI	NO	
DISPONIBLES MODOS 1 Y 2	SI	?	

Compare con otros lápices



**ESTOS SON
 ALGUNOS EJEMPLOS
 DE LOS GRAFICOS QUE VD.
 PODRA REALIZAR CON NUESTRO
 LAPIZ OPTICO**



**DE VENTA EN LOS MEJORES COMERCIOS
 DE INFORMÁTICA**

Si Vd. tiene alguna dificultad para obtener el lápiz óptico, puede dirigirse a:

DISPONIBLE PARA:

- CPC 464 CASSETTE 4.900 Ptas.
- CPC 464-664 DISCO 6.900 Ptas.
- CPC 6128 DISCO 6.900 Ptas.

(IVA no incluido)

CONDICIONES ESPECIALES PARA DISTRIBUIDORES



Avda. Isabel II, 16 -8º
 Tels. 455544 - 455533
 Télex 36698
 20011 SAN SEBASTIAN

Mercado común

Con el objeto de fomentar las relaciones entre los usuarios de AMSTRAD, **MERCADO COMÚN** te ofrece sus páginas para publicar los pequeños anuncios que relacionados con el ordenador y su mundo se ajusten al formato indicado a continuación.

En **MERCADO COMÚN** tienen cabida, anuncios de ventas, compras, clubs de usuarios de AMSTRAD, programadores, y en general cualquier clase de anuncio que pueda servir de utilidad a nuestros lectores.

Envíanos tu anuncio mecanografiado a: **HOBBY PRESS, S.A. AMSTRAD SEMANAL.** Apartado de correos 54.062 28080 MADRID ¡**ABSTENERSE PIRATAS!**

Vendo 1.—Amstrad CPC-664 con pantalla color. 2.—Impresora **PRINTER-80 INDESCOMP** (adaptada a Amstrad) matricial, 40-80-142 col; 80 cps; 640 puntos gráficos. 3.—Juego regalo: **COMBAT LINX** y otros. Precio total: 160.000 ptas. Tel. (93) 788 07 53 (21,30-22,30).

Me gustaría en primer lugar contactar con usuarios de CPC 464 para intercambiar ideas, juego, etc. (a ser posible de Avila aunque no necesariamente). En segundo lugar me gustaría obtener el Exploding Fist. Bien por dinero o bien por otros programas. También estoy interesado en: Beach Head, Hypers Sports, Decathlon, Fighter Pilot. Con las mismas condiciones que Exploding Fist. Estoy abierto a otras ofertas. Los interesados llamar al (918) 22 74 99 desde las 6 a las 6,30 o bien de 8,30 en adelante. Fines de semana a cualquier hora.

Cambio juegos excelentes de todo tipo (knight Lore, Alien 8, G. P. Rally II, Exploding Fist, etc.) También utilidades. Escribir a Jesús Alonso, c/ Ruiz 9 4.º C, Madrid 28004. También llamar desde las 22,00 al 448 98 05.

Vendo Amstrad CPC-464, monitor color en perfecto estado, incluyendo lote de 50 programas a escoger entre más de 100: **DEVPAC/AMSWORD I/PASCAL/AMSBASE SCREEN DESIGNER/AMSCALC...**, últimos juegos, 7 libros. Todo por 85.000 ptas. Más información: Eliseo González Real c/Pereu, 4. Viveda (Cantabria). Tel.: (942) 88 48 24.

Estoy interesado en comprar un monitor color, o cambiarlo por uno GT/65 con garantía Indescamp, comprada hace 2 meses, pagando la diferencia justa. También cambio juegos, programas, utilidades, etc., tengo unos 100; mandaré lista.

Interesados dirigirse a: Urb. Bahía de Algeciras Bloque 2G-2; 2.º C. Algeciras (Cádiz)... Manolo Benitez.

Deseo contactar con usuarios de Amstrad, de Gandia y alrededores, para ofrecermelo como profesor de programación (200 ptas/h). O para intercambiar listados e ideas. Abstenerse los manicidos de los juegos. Santiago Torrego Calabuig, c/ Abad Sala, 92, Gandia (Valencia).

Desearía contactar con usuarios 464 para cambio de utilidades y programas. Interesados escribir a: Juan Carlos Martínez Lizán. C/ Pedro Díaz, 12, 1.º A. Cartagena (Murcia).

Vendo o cambio programas de Amstrad. Usuarios de toda España interesados, escribir a Carlos Ribo Vilaseca, Urbanización Mirasol, 1, Solana (Lérida) o llamar al tel. (93) 811 28 35 para llamadas de fuera de la provincia de Barcelona. Llamar de 1,30 a 3,00 y de 8 a 9,30.

Vendo Amstrad CPC 464 con monitor en fósforo verde. Manual y 100 programas (juegos, aplicaciones...) incluidos. Interesados llamar al (91) 260 79 94. Preguntar por Juan Carlos de 3 a 5.

Cambio juegos excelentes de todo tipo (knight Lore, Alien 8, G. P. Rally II, Exploding Fist, etc.) También utilidades. Escribir a Jesús Alonso, c/ Ruiz 9 4.º C, Madrid 28004. También llamar desde las 22,00 al 448 98 05.

Vendo consola de videojuegos Atari-2.600, un joystick, 5 cartuchos, Defender, Berzeck, dog'em, Space Invaders, y Missile Command. Más 2 mangos de paleta, más instrucciones en castellano, cables, etc.; todo ello casi nuevo y en perfecto estado de funcionamiento. Además 5 revistas de Microhobby (Spectrum). Todo ello por 19.500 ptas. (negociables). Interesados llamar al tel.: 23 51 14 de Murcia. Preguntar por Javier (llamar a cualquier hora de la tarde de los sábados).

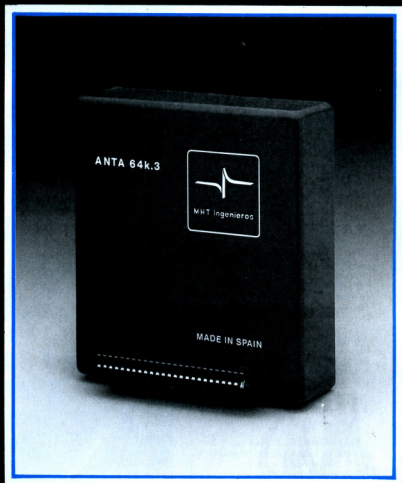
LE CONCEDEMOS TRES DESEOS



ANTA 64K.3

Los 64K de memoria que esperaba su Amstrad

Ampliación de memoria, buffer de impresora y ram disk*



Si tiene un AMSTRAD CPC 464, CPC 664 o CPC 6128 conéctele el ANTA 64K.3 y seleccione la opción que necesite:

64K de Memoria

Para leer y escribir datos, cadenas y bloques de caracteres, así como copiar o trasladar pantallas.

64K de Buffer de Impresora

Permite seguir trabajando

con el ordenador mientras la impresora funciona.

64K de Ram Disk/Basic

La memoria simula el funcionamiento de un disco con mejor tiempo de acceso.

* Software de manejo contenido en ROM.

MHT Ingenieros

MEMORIA DE PANTALLA



Les envío una pequeña subrutina que pretende ayudar a conocer de manera rápida el mapa de pantalla de un locate en modo cero.

ESTRUCTURA DEL PROGRAMA

- 1-2** Cabecera.
 - 20** Dimensionamiento de variables.
 - 30-100** Mediante bucles se introducen en la variable PIN (N, N) las direcciones de memoria de los ocho primeros pixel verticales de cada una de las cuadrículas de la primera columna de pantalla.
 - 110** Establece modo de pantalla.
 - 120-140** Pregunta por el locate que ha de analizar.
 - 150** Resta a la coordenada x una unidad, e imprime los números 1 al 4 indicativos de los pixel.
 - 160** Bucle que contabiliza los ocho pixel verticales.
 - 170** Introduce en la variable pin1 (1, n) los valores de las direcciones de memoria de la cuadrícula de coordenada (1, y) y los multiplica por el factor (x*4).
 - 180** Imprime primero el número de pixel y después sus valores en hexadecimal.
 - 190** Cierra bucle.
 - 200-210** Espera pulsación de una tecla (CALL&BB18) y vuelve a ejecutar el proceso completo.
- Rafael Morales Navarro

```

1 *-----RAFAEL MORALES NAVARRO-----
2 *-----MAPA DE PANTALLA EN MODO CERO-----
30 DIM pin (26,8),pin1(1,8)
30 primero=&C000:ultimo=&FB00:punto=0
40 FOR y=1 TO 26
50   FOR asprimero TO ultimo STEP 500
60     punto=punto+1
70     pin(y,punto)=a
80   NEXT a
90 primero=primero+&50:ultimo=ultimo+&50:punto=0
100 NEXT y
110 MODE 1
120 PRINT "DE QUE <LOCATE> QUIERES SABER SU MAPA DE" STR
ING$(40," ") "MEMORIA EN MODO CERO"
130 LOCATE 1,6:INPUT "X":X:IF X<1 OR X>20 THEN LOCATE 3,
6:PRINT " ":GOTO 130
140 LOCATE 1,8:INPUT "Y":Y:IF Y<1 OR Y>25 THEN LOCATE 3,
8:PRINT " ":GOTO 140
150 X=X-1:PAPER 1:PEN 0:LOCATE 15,10:PRINT "LOCATE" (X+1)
",Y":PAPER 0:PEN 1:LOCATE 8,12:PRINT "1 2 3
4":LOCATE 7,13:PRINT STRING$(25,"-"):PRINT
160 FOR A=1 TO 8
170 PIN1 (1,A)=PIN1(Y,A)+(X*4)
180 PRINT A SPC(3) HEX$(PIN1(1,A))SPC(3) HEX$(PIN1(1,A)
+1) SPC(3) HEX$(PIN1(1,A)+2) SPC(3) HEX$(PIN1(1,A)+3)
190 NEXT
200 LOCATE 1,25:PRINT "pulsar una tecla para continuar":C
ALL &BB18
210 GOTO 110
    
```

4 297318

A usted, minorista, y con sólo marcar este teléfono, le concedemos lo que siempre ha esperado de su mayorista informático.

- 1 Todas las marcas.** Amstrad, Spectrum, Commodore... para que con una sola llamada, usted tenga todo lo que necesita.
- 2 Rapidez en el servicio.** Le entregamos su pedido en 24 horas, sin demoras y en cualquier punto de España.
- 3 Trato directo.** Mantenemos un contacto continuo con usted, nos preocupamos por sus problemas y le ayudamos a solucionarlos. Queremos que usted sea algo más que un cliente.

Si es esto lo que pide a su mayorista, LLAMENOS



CUMPLIMOS SUS DESEOS

MICRO-1

el IVA lo paga
MICRO-1

C/ Duque de Sesto, 50. 28009 Madrid
Tel.: (91) 275 96 16/274 53 80
(Metro O'Donell o Goya)
Aparcamiento gratuito en Felipe II

SOFTWARE: ¡¡2 PROGRAMAS POR EL PRECIO DE 1!!
Y además, completamente gratis, un magnífico reloj de cuarzo. Increíble ¿verdad?

	Ptas.
PING PONG	2.295
SABOTEUR	2.295
RAMBO	2.295
YEAR KUNG FU	2.295
WORLD SERIES BASEBALL	2.095
MAPGAME	2.750
RAID	2.295
HYPERSPORTS	2.295
HIGHWAY ENCOUNTER	1.750
HIGHWAY ENCOUNTER DISCO	3.300
ALIEN B	1.750

	Ptas.
DYNAMITE DAN	2.100
SABRE WULF	1.650
THEY SOLD A MILLION	2.500
FIGHTER PILOT	1.975
MASTER OF T. LAMP	1.950
NIGHT SHADE	1.950
HACKER	1.950
SUPER TEST	2.300
TORNADO LOW LEVEL DISCO	3.300
TORNADO LOW LEVEL	1.750
KNIGHT LORE	1.750

SOFTWARE DE REGALO: ¡¡OFERTA 2 x 1!!

Beach Head Decathlon Dummy Run Beach Head Southern Belle

*Fabulosos
precios para tu Amstrad
CPC-464 CPC6128
PCW-8256 y
PCW-512*

SOFTWARE DE GESTION PROFESIONAL

DBA II	17.800	DR. GRAPH	15.100
CBASIC	15.100	CONTABILIDAD	
DR DRAW	15.100	Y VTOS.	16.600

IMPRESORAS ¡¡20% DTO. SOBRE P.V.P.!!

COMPATIBLE IBM PC-XT 256 K
Y DOS DISKETTES DE 360 K
229.900 PTAS.

UNIDAD DE DISCO 5¼"
PARA AMSTRAD
34.900 PTAS.

LAPIZ OPTICO+INTERFACE
3.495 PTAS.

CINTA VIRGEN ESPECIAL ORDENADORES
69 PTAS.

SINTETIZADOR DE VOZ EN
CASTELLANO
15% DTO.
CASSETTE ESPECIAL ORDENADOR
5.295 PTAS.

JOSTICK QUICK SHOT II
1.995 PTAS.
JOSTICK QUICK SHOT V
2.295 PTAS.
con la compra de un joystick
¡¡GRATIS 1 RELOJ DE CUARZO!!

DISKETTE 5¼"
295 PTAS.

DISKETTE 3"
990 PTAS.

LO NUESTRO ES HACER BUENAS GESTIONES



PARA
AMSTRAD, PC Y COMPATIBLES

¡No estamos para juegos!

DEMOSTRACIONES DE NUESTRO SOFTWARE
**COMERCIAL Y DE GESTION
EN NUESTRO STAND**



les esperamos en Palacio de Exposiciones y Congresos de Madrid
23, 24 y 25 de Mayo 1986

PRECIOS ESPECIALES 1.ª FERIA AMSTRAD

FACTURACION - Sólo teclee un código y salen todos los datos del cliente. Numeración correlativa automática. Admite 30 productos distintos por factura. Admite 30 productos distintos por factura. Automáticos, descuentos, cargos, IVA. Proporciona 5 totales por factura. (PVP 15.300 incl. IVA)

PRESUPUESTOS - Guarda en memoria los presupuestos y extiende los facturas. Conceptos de 200 caracteres cada uno (3 renglones de escritura) (PVP 18.300 incl. IVA)

CLIENTES - PROVEEDORES, BANCOS, CLIENTES - 3 ficheros separados. Resúmenes totales, unitarios o parciales. El mejor auxiliar de CONTABILIDAD al día. (PVP 8.600 incl. IVA)

CONTROL DE ALMACEN IVA - Código de 9 dígitos alfanuméricos. 25 dígitos denominación. Una sola pantalla entradas y salidas, con visión de asientos anteriores. Stocks máximo, mínimo y aviso para reaprovisionamiento. Totales entradas y salidas cada pantalla (PVP 15.300 incl. IVA)

CLIENTES (con etiquetas) - 11 campos distintos para localización. Etiquetas 4 modelos distintos en salida de dos. El más fiel auxiliar ahorrador de tiempo. (PVP 8.600 incl. IVA)

RECIBOS - Resuelve el problema interminable a asociaciones, comunidades, colegios. Fijos los campos del normalizado y 12 campos libres (4 numéricos con cálculos automáticos). Liquidaciones bancos. (PVP 18.300 incl. IVA) Con numeración automática (21.200 incl. IVA)

RESTAURANTES - Tratamiento de minuta y facturas. Resúmenes por grupos. Mesas abiertas permanentemente, correcciones, cambios, etc. Hasta emisión fra. final. (PVP 35.000 incl. IVA)

IVA POR ALMACEN - Rellena liquidaciones Hacienda. Introduce cuentas IVA gastos. (PVP 18.900 incl. IVA)

URBANIZACIONES - Lectura y tratamiento de contadores consumos. (agua, gas, luz, etc) Extensión recibos y totalizaciones bancos. Emisión etiquetas.

LIBROS DEL IVA - Controles de repercutido y soportado orden numérico. Resúmenes estudios comparativos. Relleno liquidación Hacienda. (PVP 16.800 incl. IVA)

ADMINISTRACION DE FINCAS - Gestión completa profesionales. Sencillo manejo cualquier persona (PVP 40.000 incl. IVA)

FACTURACION Y ALMACEN - Gestión unido. Ficheros clientes, producto, descuentos y cargos. Todos los resúmenes. (PVP 18.900 incl. IVA)

1 AÑO DE GARANTIA

NUESTRO EQUIPO PROFESIONAL
PARA CUALQUIER MODIFICACION QUE
UD. INDIQUE EN LOS PROGRAMAS, A
UN PRECIO MODICO

Llamar o contactar con Juan Luis Ruiz

PEDIDOS, TELEFONO, CARTA O TELEX
REEMBOLSO SIN GASTOS.

ESPECIAL A COLABORADORES
RESTO DE ESPAÑA



**informática
GROTUR, S.A.**

C/ JAIME EL CONQUISTADOR, 27
28045 MADRID Tno. 474 55 00

474 55 32

Télex: IGSA 48452



RPA[®]

Systems Inc.

GESTION DE EMPRESA



LA SENCILLEZ ESTA EN EL PROGRAMA

El programa de gestión de Empresa RPA Systems es un claro ejemplo de lo que debe ser una solución informática.

Una herramienta eficaz que hace más fácil la tarea de la Gestión Empresarial, huyendo de innecesarias complicaciones. Porque para obtener el máximo rendimiento de los programas RPA Systems **no es necesario saber programar**. Con un lenguaje compilado de alto nivel y continuas ayudas en pantalla, son muy fáciles de usar. Como el programa de Gestión de Empresas que diseñado para trabajar con los Amstrad 8256, 6128 y 8512 incluye los subprogramas de: Contabilidad General, Nóminas, Facturación, Fichero de clientes y proveedores.

Si es Usted pequeño o mediano empresario en RPA Systems encontrará una eficaz ayuda para la clasificación y control de clientes, realización de facturas, totalización de cobros y pagos, generación de nóminas y contabilidad ajustada al plan general contable.

Además, el programa de gestión de empresa de RPA Systems permite llevar un perfecto control de la aplicación del IVA.

RPA Systems es la respuesta eficaz a sus necesidades de informatización. **Así de sencillo.**

SOLICITE INFORMACION EN:
División Informática de GALERIAS . División On-line de GALERIAS
Tiendas especializadas en informática y Equipos de oficina.

RPA

Distribuidor exclusivo en España: **ACE DISTRIBUCION S.A.**
Galileo, 25. Entrepiso A. Tel. 447 97 51 / 98 09. 28015 Madrid.
Distribuidor exclusivo en Catalunya: **ACE DISTRIBUCION S.A.**
Tarragona, 112. Tel. (83) 925 15 12. 08015 Barcelona. Telex: 531333 AGEE E

*Su programa sólo
por 76.500 ptas
Así de sencillo!*