

AMSTRAD

CASSETTE

DE EDICIONES MANALI, S.A.

SERIE AMARILLA · AÑO 1 · N.º 1

P.V.P. 750 ptas. (I.V.A. incluido)

JUEGOS:

OPERACION NEPTUNO:

«Conquista el Peñón con el submarino fantástico».

Y'HTZZEE:

«El sorprendente juego de dados».

O'THELO:

«Nueve formas de practicar este clásico».



APRENDE A PROGRAMAR:

LENGUAJE BASIC
CODIGO MAQUINA

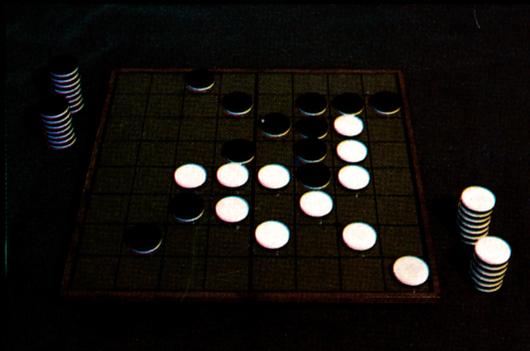
UTILIDADES:

FICHEROS

COLECCIONABLES:

«Diccionario de BASIC».

«Tablas de conversiones».



GANA DE 100.000 a 150.000 ptas. por tu juego

APLICACIONES PROFESIONALES PCW

AMSTRAD

CREADA PARA QUIENES DESCUBRIERON QUE:
LO INCREIBLE YA ES REALIDAD

CIERTAMENTE
lo increíble es realidad



AHORA,
que se ha atrevido
a descubrir que
los ordenadores increíbles
pueden ser realidad no se
conforme con cualquier software.

Exija la mejor calidad, un buen precio,
máxima facilidad de uso, servicio postventa,
comprobada profesionalidad y seriedad.

DEJESE SORPRENDER por nuestros productos
y descubrirá que los mejores programas no tienen
por que ser caros.

Descubra nuestros títulos DIMENSION NEW
para - **PCW** AMSTRAD

- **COMPRAS Y VENTAS.**
 - *Contabilidad Personal.*
 - *Control Stocks.*
 - *Facturación.*
 - *Gestión Comercial.*
 - *Plan Contable Nacional.*
 - *Vencimientos.*



IDEALOGIC

c/. Valencia, 85 - 08029 Barcelona
Pedidos: Tel. 253 74 00

Para mayor información:
DEPARTAMENTO DE M.K.
MARTA CORTILES
c/. Valencia, 85 - 08029 Barcelona
Tel. 253 86 93



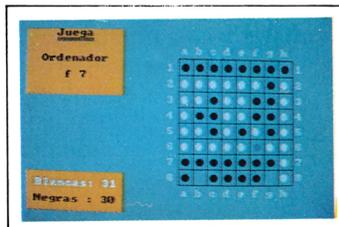
AMSTRAD

CASSETTE

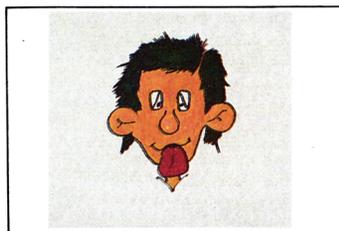
4 Y'AHTZEE

Juega: ENRIGUE		JUGADORES					
		1	2	3	4	5	6
A-Suma los	1	2	2	4	4	4	4
B-Suma los	2	5	4	6	8	12	12
C-Suma los	3	8	6	6	12	12	12
D-Suma los	4	10	8	12	15	18	20
E-Suma los	5	10	10	12	15	18	20
F-Suma los	6	12	12	12	15	18	20
Sub-Total		39	24	24	27	42	50
Bonus 35	1162						
G-3 iguales	101		21				
H-3 iguales	101			25			
I-Full (1/2)	25			25			
J-Escala 4	30		30		30		
K-Escala 5	40		40		40		
L-Y'AHTZEE	50		50		50		
M-Comodin	101					18	
		103	75	167	110	112	145
		1	2	3	4	5	6

6 O'THELO



8 PROGRAMANDO BASIC



12

UTILIDADES

Editorial

Hace ya algún tiempo empezamos a analizar las consecuencias que podría tener en la comercialización de revistas de software en cassette, la aparición de una revista en diskette.

De nuestras conclusiones, pensamos que directamente se produciría una fuerte caída de este mercado. Ello arrastraría a muchas revistas y editoriales, con lo que se crearía un gran vacío en este sector informático.

En consecuencia las víctimas serían todos los usuarios de ordenadores por cassette, entre ellos los poseedores del Amstrad CPC 464.

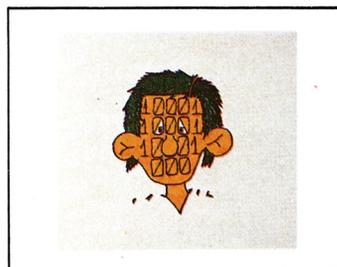
Este vacío vendría a sumarse al ya existente. La mediocre calidad de los programas que acompañaban a ciertas publicaciones hizo que el consumidor las rechazara.

De todo esto, nacerá un mercado mucho más restringido, que deberá poseer la calidad que los usuarios desde hace tiempo reclaman.

Nosotros, no podemos sino meternos de lleno en esta nueva ola, en la cresta, ofreciendo una relación precio-calidad que permita equiparar este mercado al del diskette.

He ahí nuestro objetivo; formar parte de esta nueva generación de revistas de software que ya nada tendrá que ver con las anteriores.

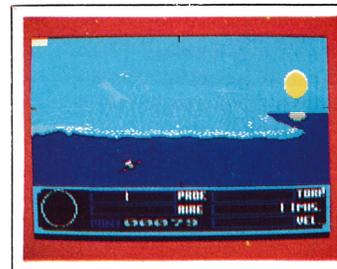
14 CODIGO MAQUINA

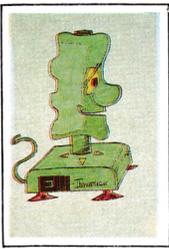


17

FICHEROS Y DICCIONARIO DE BASIC

20 NEPTUNO





JUEGOS

Y'AHITZEE

Y'AHITZEE es un juego de azar y estrategia que en esta versión para su AMSTRAD permite participar de 1 a 6 jugadores.

Instrucciones de carga del programa

—Programa en **CINTA CASSETTE** para ordenador AMSTRAD CPC 464. Conecte el ordenador y prepare la cinta cassette que contiene el programa desde el principio. Teclee en su ordenador **RUN "YAHT"** y pulse **ENTER**.

—Programa en **CINTA CASSETTE** para ordenador AMSTRAD CPC 664 y 6128. Conecte el ordenador y prepare su cassette para lectura, asegúrese de que el volumen se encuentra en una zona medio-alta, coloque la cinta conteniendo el programa desde el principio. Teclee: **TAPE** y pulse **ENTER**; el ordenador responderá con el mensaje **READY**. Teclee ahora **RUN "YAHT"** pulse **ENTER** y **PLAY** en su cassette.

—Programa en **DISKETTE** para ordenador CPC 664 y 6128. Conecte el ordenador, coloque el Diskette en

su alojamiento, teclee **RUN "YAHT"** y pulse **ENTER**.

En cualquiera de los casos anteriormente citados el juego se inicializará automáticamente una vez el programa haya sido cargado.

Desarrollo del juego

Cada jugador por turnos puede lanzar 5 dados hasta 3 veces consecutivas, pero antes de la segunda y tercera

tirada puede fijar los que crea que le ayudarán a conseguir la combinación adecuada; indicando con un número del 1 al 5, el dado que quiere volver a lanzar (1 es el dado de la parte izquierda de la pantalla).

Según la combinación obtenida después de las 3 tiradas, debe elegirse una de las 13 secciones (**A-M**) que aparecen en pantalla; (y que se detallan más adelante); intentando obtener el mayor número de puntos cada vez, pero **¡CUIDADO!** según las secciones elegidas usted puede llegar a las últimas rondas de la partida «obligado» a una determinada combinación; y si no la consigue realizar sumará 0 puntos.

Una vez realizadas las 13 combinaciones de cada jugador, el ganador será lógicamente aquel que haya obtenido más puntos; pasando su nombre a la lista de los 10 mejores **RECORDS** que aparecen al final de la partida.

Juega: ENRIQUE		JUGADORES					
		1	2	3	4	5	6
A-Suma los	1	1					
B-Suma los	2	2					
C-Suma los	3	6					
D-Suma los	4	6					
E-Suma los	5	10					
F-Suma los	6	24					
Sub-Total.....		43					
Bonus 35 si >62							
G-3 iguales	TOT	24					
H-4 iguales	TOT	22					
I-Full (3/2)	25	25					
J-Escala/4	30	30					
K-Escala/5	40	40					
L-Y'AHITZEE	50	50					
M-Comodin	TOT	25					
		259					
		1	2	3	4	5	6

Participa un solo jugador. El tanteo final es de 259 puntos.

Juega: ELISA		JUGADORES					
Primera tirada		1	2	3	4	5	6
A-Suma los	1	2	2		4	2	
B-Suma los	2	8	4	6	8	4	4
C-Suma los	3		3	6	0	12	12
D-Suma los	4	16	8	12	4	8	
E-Suma los	5	10	10		15		20
F-Suma los	6	18	12	18	18	18	18
Sub-Total.....		54	39	42	49	44	54
Bonus 35 si >62							
G-3 iguales	TOT	25	21	25	13	0	13
H-4 iguales	TOT		28	28			
I-Full (3/2)	25	25	25	25	25	25	25
J-Escala/4	30	30	30	30	30	30	30
K-Escala/5	40	40	40	40	40	40	40
L-Y'AHITZEE	50	50		50			
M-Comodin	TOT	23	19	23	18	12	18
		247	162	263	145	151	189
		1	2	3	4	5	6

Participan seis jugadores, los números tres y uno se afianzan en los primeros lugares.

Tabla de combinaciones

En pantalla aparece la tabla siguiente:

- A suma los 1 (ej. 41156 = 2 puntos)
- B suma los 2 (ej. 22.213 = 6 puntos)
- C suma los 3 (ej. 14.456 = 0 puntos)
- D suma los 4 (ej. 45.612 = 4 puntos)
- E suma los 5 (ej. 55.145 = 15 puntos)
- F suma los 6 (ej. 63.261 = 12 puntos)

Cuando la suma de las secciones A-F es mayor de 62 puntos, se obtiene un **BONUS** adicional de 35 puntos.

- G 3 iguales TOT (ej. 55125 = 18 puntos)
- H 4 iguales TOT (ej. 14444 = 17 puntos)
- I full (3 y 2) 25 (ej. 32233 = 25 puntos)
- J Escala de 4 30 (ej. 14236 = 30 puntos)
- K Escala de 5 40 (ej. 65324 = 40 puntos)
- L Y'AHTZEE 50** (ej. 111111 = 50 puntos)
- M Comodín TOT (ej. 13461 = 15 puntos)

Como se observa las secciones I a L tienen una puntuación fija y es indiferente el tipo de combinación para obtenerla, es decir, vale 50 puntos igual un **Y'AHTZEE** formado con cinco «1» que otro formado con cinco «6». La sección **M** debe reservarse como emergencia y se recomienda dejarla para el final de la partida cuando se empieza a ir «obligados»; nos puede sacar de una situación de apuro. No obstante la estrategia de la partida debe de marcarla el propio jugador.

Juega: ENRIQUE		JUGADORES					
		1	2	3	4	5	6
A-Suma los	1	2	2		4		
B-Suma los	2	8	4		8	4	
C-Suma los	3			6		12	12
D-Suma los	4		8			8	
E-Suma los	5	10	10		15		20
F-Suma los	6	18		18		18	18
Sub-Total . . .		38	24	24	27	42	50
Bonus 35 si >62							
G-3 iguales	TOT		21				
H-4 iguales	TOT			28			
I-Full (3/2)	25			25	25		25
J-Escala/4	30	30	30			30	30
K-Escala/5	40	40		40	40	40	40
L-Y'AHTZEE	50			50			
M-Comodín	TOT				18		
		108	75	167	110	112	145
		1	2	3	4	5	6

El participante n.º 1 ha conseguido Y'AHTZEE.

Juega: INMA		JUGADORES					
Primera tirada		1	2	3	4	5	6
A-Suma los	1	2			4		
B-Suma los	2		4		8	4	
C-Suma los	3			6		12	12
D-Suma los	4		8				
E-Suma los	5	10	10		15		
F-Suma los	6			18			18
Sub-Total . . .		12	22	24	27	16	30
Bonus 35 si >62							
G-3 iguales	TOT		21				
H-4 iguales	TOT			28			
I-Full (3/2)	25						25
J-Escala/4	30	30					
K-Escala/5	40	40		40	40	40	
L-Y'AHTZEE	50						
M-Comodín	TOT						
		82	43	92	67	56	55
		1	2	3	4	5	6

El participante n.º 5 ha conseguido una escalera.

Juega: MANUEL		JUGADORES					
NO VALIDO		1	2	3	4	5	6
A-Suma los	1	2	2		4	2	0
B-Suma los	2	8	4	6	8	4	4
C-Suma los	3	3	3	6	0	12	12
D-Suma los	4	16	8	12	4	8	
E-Suma los	5	10	10		15	15	20
F-Suma los	6	18	12	18	18	18	18
Sub-Total . . .		57	39	42	49	59	54
Bonus 35 si >62							
G-3 iguales	TOT	25	21	25	13	0	13
H-4 iguales	TOT		28	28	0		9
I-Full (3/2)	25	25	25	25	25	25	25
J-Escala/4	30	30	30	30	30	30	30
K-Escala/5	40	40		40	40	40	40
L-Y'AHTZEE	50	50		50			
M-Comodín	TOT	23	19	23	18	12	18
		250	162	263	145	166	189
		1	2	3	4	5	6

El participante n.º 2 ha pulsado una sección ocupada. Observar el mensaje «No valido».

Cómo tirar los dados

Una vez aparecida la primera combinación en pantalla y para volver a tirar los dados, debemos seleccionar los que queremos volver a lanzar, pulsando su número de orden (1 a 5) quedando indicados en pantalla con una señal debajo de los mismos. Una vez seleccionados pulsar **ENTER** y volverán a aparecer los nuevos valores para cada dado. Caso de que antes de la 3. tirada, apareciese una combinación que nos interese; pulsar **ENTER** sin marcar ningún dado y ésta quedará fijada. Para incrementar la fuerza de la tirada del dado se puede pulsar varias veces el número de orden de dado a tirar (normalmente con una vez es suficiente), antes de pulsar **ENTER**.

Cómo marcar la sección

El ordenador le preguntará: Sección (A-M?) y usted debe pulsar la letra de la sección que interese. Si la combinación es correcta, se sumarán los puntos obtenidos, en caso contrario se obtienen 0 puntos y se pasa el turno al siguiente jugador. Caso de pulsar una sección ya ocupada, aparecerá el mensaje «NO VALIDO» y el ordenador esperará una nueva entrada correcta. **BUENA SUERTE.**



JUEGOS

O'THELO

Es el *O'THELO* un fascinante juego de estrategia que ha pasado rápidamente de ser «el juego de moda», a convertirse en un *CLASICO MUY ESTIMADO POR LOS AFICIONADOS*.

Aunque muy fácil de aprender, requiere una buena dosis de habilidad y experiencia, pues las tácticas y estrategias empleadas han de ir variando de forma casi continua a lo largo de la partida.

Instrucciones de carga del programa

—Programa en **CINTA CASSETTE** para ordenador **AMSTRAD CPC 464**. Conecte el ordenador y prepare la cinta cassette que contiene el programa desde el principio. Teclee en su ordenador **RUN "OTHELO"** y pulse **ENTER**.

—Programa en cinta cassette para ordenador **AMSTRAD CPC 664 y 6128**.

Conecte el ordenador y prepare su cassette para lectura, asegúrese de que el volumen se encuentra en una zona medio-alta, coloque la cinta conteniendo el programa desde el principio. Teclee: **TAPE** y pulse **ENTER**; el ordenador responderá con el mensaje **READY**.

Teclee ahora **RUN "OTHELO"**, pulse **ENTER** y **PLAY** en su cassette.

—Programa en **disco** para ordenador **AMSTRAD CPC 664 y 6128**. Conecte

el ordenador e introduzca el disco en la unidad. Teclee **RUN "OTHELO"** y aparecerá en pantalla un **MENU** de opciones, seleccione la que desee y pulse **ENTER**.

Desarrollo del juego

En la primera opción,

tiene Ud. ante sí un poderoso rival que se lo va a poner muy difícil; no obstante, y como nadie «nace enseñado», el ordenador le va a ofrecer 9 niveles de juego; para que al principio disponga de alguna ventaja.

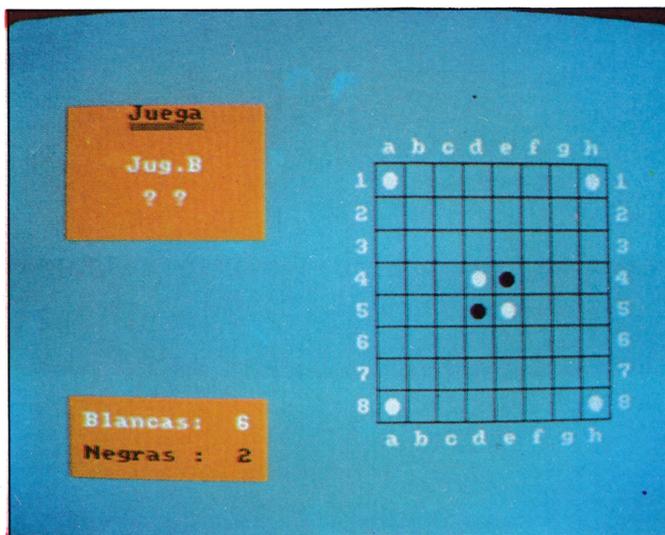
De todos modos no se confíe aunque aunque al

principio de la partida parezca que va ganando, está jugando contra un **CAMPEON** y como tal buscará ante todo un buen posicionamiento de sus fichas aunque tenga que sacrificar alguna de ellas, luego le atacará implacablemente hasta rematarlo. En la opción 2, será Ud. quien tendrá que demostrar su técnica contra otro jugador. Si uno de los dos es más experto que el otro, podrá (como en el caso del ordenador), dar cierta ventaja (distintos niveles).

En todos los casos el ordenador realizará las funciones de árbitro, indicando e impidiendo la realización de jugadas «no legales».

Cómo jugar

Cada jugador comienza con dos fichas ya colocadas en el centro del tablero que aparecen en pantalla (si juega contra el ordena-



INICIO DEL JUEGO... personas muy inexpertas, el ordenador le concede 4 fichas.

dor las suyas son las blancas).

En el caso de que no sea Ud. un jugador muy experto, dispondrá además de alguna ficha de refuerzo, que el ordenador le concede como ventaja según el nivel solicitado. **ESTO QUIERE DECIR QUE EL ORDENADOR SIEMPRE JUEGA A MAXIMA POTENCIA INDEPENDIENTEMENTE DEL NIVEL DE JUEGO**, y lo único que hace en los niveles bajos es dejar al contrincante fichas de ventaja en lugares estratégicos. Creemos que ésta es la mejor manera (pedagógicamente hablando), para llegar a convertirle a Ud. en un gran campeón (casi como su ordenador).

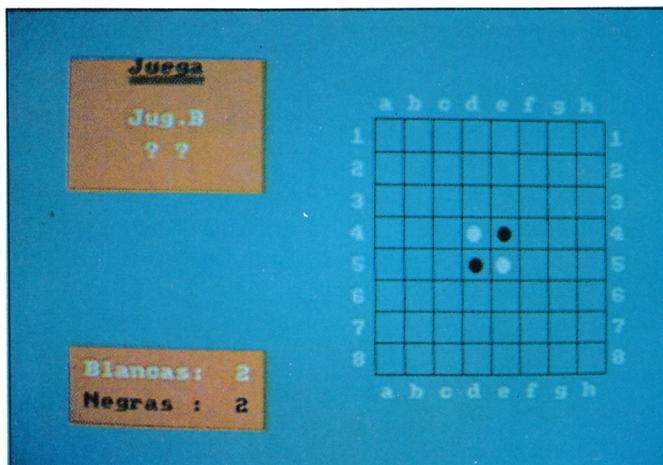
En caso de que se considere Ud. de nivel superior al normal; será Ud. el que le concederá fichas de refuerzo al ordenador.

El primer turno de tirada siempre corresponde al jugador que lleva las fichas blancas. Si se desea cambiar el turno de tirada simplemente pulsar «I».

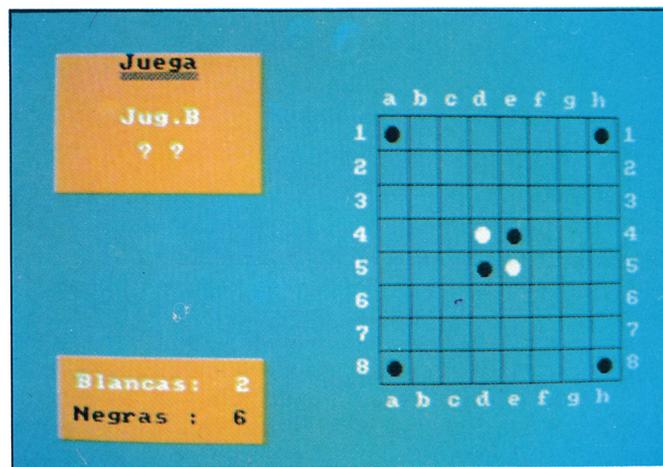
Para jugar (cuando le corresponda) ha de conseguir tirar una ficha nueva que de forma «**OBLIGATORIA**» debe encerrar otra u otras del adversario entre esa que Ud. ha puesto y otra de las que ya tenía (puede ser vertical, horizontal o diagonalmente), sin que exista un cuadro vacío entre ellas.

En ese momento, todas las fichas del contrario que Ud. haya encerrado se convertirán en fichas de su color y por tanto suyas. En el marcador se indicará ese cambio.

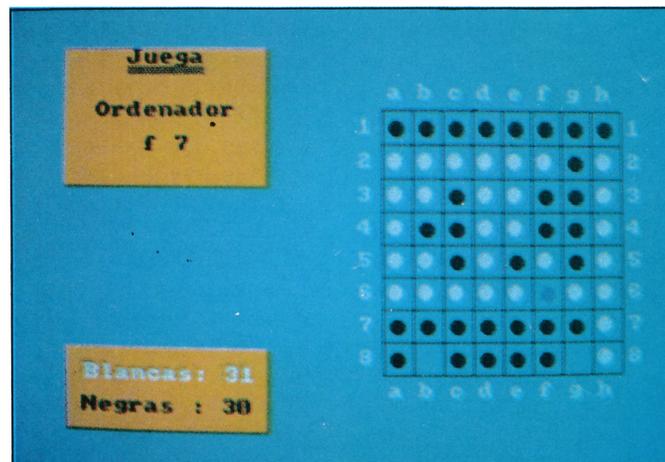
Para realizar la tirada, es



... personas de un nivel normal, tanto ordenador como jugador tienen las mismas posibilidades.



... personas de un nivel excepcional, el jugador concede 4 fichas al ordenador.



Está a punto de concluir la partida, el que más fichas tenga de un color (blancas o negras) es el ganador.

necesario indicarle al ordenador las coordenadas del cuadro donde quiere poner la nueva ficha (ej. **A6**), que aparecerá inmediatamente en ese punto, cambiando seguidamente el color de las encerradas.

Jugada ilegal

Se produce cuando se coloca una ficha en lugar inadecuado del tablero, que no permite encerrar fichas del contrario. Debe repetirse la jugada colocando la ficha correctamente.

Por todo lo expuesto, se observa que las reglas de este juego son muy simples, pero que una vez dentro de la partida, existen gran cantidad de estrategias, y es ahí donde se demuestra la calidad del jugador; y no es fácil a veces dar con la táctica más adecuada y que a la vez despiste al contrario.

Siempre es importante intentar ahogar al adversario, es decir, que no tenga posibilidad de tirar por no poder encerrar ninguna ficha. En este punto el adversario deberá ceder su turno al otro jugador pulsando «I» (el ordenador lo realiza automáticamente).

Cuando se han cubierto los 64 cuadros del tablero, aparecerá en una pantalla la puntuación final y el resultado de la partida.

El juego se presenta sobre un tablero dividido en 8 por 8 cuadros, numerados de A a H y de 1 a 8, para poder indicar la coordenada correspondiente.

Las fichas están representadas por discos negros y blancos.



LENGUAJE BASIC

En esta sección intentaremos daros una idea clara de lo que es la programación en lenguaje BASIC. Lo único que hace falta es fijarse un poco, algo de imaginación y práctica; que nadie espere aprender a programar sólo leyendo, hay que ponerse delante del ordenador e intentar hacer pequeños programas al principio, pero irán saliendo mejores a medida que practiquemos y aprendamos a programar mejor.

A lo largo de esta serie daremos una base firme para programar, enseñando eso que no vienen en los libros, por que son propias del ordenador que se está usando. Averiguar todo esto lleva mucho tiempo de práctica y de enredar con el aparato, algo que en la mayoría de los casos no dará ningún fruto. En definitiva os diremos cómo hacer vuestros programas más rápidos, vistosos y fáciles de usar. Esperamos contentar a todos, a los que empiezan y a los que ya saben algo de programación.

El ordenador

Lo primero que llama la atención en un Amstrad, es la unidad de disco o de cassette. Esta es la unidad de almacenamiento externa del ordenador, llamémosla su carpeta electrónica. En ella podemos almacenar programas, datos. Profundizaremos en esta cuestión más adelante.

La pantalla es una parte importantísima, ya que en ella veremos lo que decimos al ordenador, además de las contestaciones que nos dé el programa, sean texto o gráficos. Al encender el ordenador nos presenta un mensaje del fabricante y a continuación uno del aparato, **READY**, debajo de éste, tenemos un **cuadradito de color**, es lo que se llama **cursor** e in-

dica en qué posición se va a poner el carácter que tecléemos (llamamos carácter a cualquiera de los símbolos que puede imprimir el ordenador, ya sean letras, números o símbolos especiales). Más adelante veremos ampliamente la pantalla.

El teclado, es la parte del ordenador que más vamos a utilizar, ya que a través de él nos comunicaremos con el aparato para darle nuestras órdenes; por eso conviene conocerlo a fondo, ya que da muchas facilidades en la edición de programas. El teclado es igual al de una máquina de escribir, hay que tener en cuenta que las teclas tienen **autorepetición**, es decir, si se mantiene pulsada un tiempo repetirá el carácter correspondiente hasta que se levante el dedo.

Hay algunas teclas especiales que tienen una función muy importante. Vamos a explicar la utilidad de las mismas:

ENTER: Es probablemente la tecla más importante del ordenador, con ella le decimos cuando queremos que ejecute un orden o almacena un dato en memoria. **Se debe pulsar siempre que queramos que el ordenador se dé por enterado de algo.** Son las dos teclas grandes que hay a la derecha, una de ellas marcada como RETURN.

LAS TECLAS DE CURSOR: Son muy importantes en la edición de programas, conviene manejarlas muy bien, ya que agilizarán mucho la redacción de los mismos. Están situadas debajo del teclado numérico formando un T invertida (CPC6128) o enci-

ma del mismo formando una cruz (CPC464 y 664). **Nos permiten mover el cursor por la línea con la que trabajemos en ese momento.** Si no hay ningún carácter en la línea, el cursor se puede mover libremente por toda la pantalla (esto se puede utilizar para acostumbrarse a su manejo). Estas teclas son de autorepetición.

La opción de cursor de copia, se conecta cuando pulsamos la tecla MAYS junto con una de las de cursor, entonces se puede ver como sale una copia del mismo, la cual podemos mover por la pantalla con total libertad (siempre que mantengamos pulsada MAYS). Si paramos el cursor de copia encima de algún carácter y pulsamos la tecla COPIA (situada entre la barra espaciadora y CTRL en el 128 y entre las

teclas de cursor en los 64), veremos que el carácter situado debajo del cursor de copia se imprime en la posición en que se encuentra el cursor normal; si la mantenemos pulsada copia todo lo que encuentre a su paso.

Si las utilizamos **junto con la tecla CTRL podremos ir al principio o final de la línea en la que nos encontremos tanto física como lógica.**

MAYS: Pulsada con cualquiera de las teclas, producen mayúsculas o el símbolo dibujado en la parte de arriba de la tecla. Están situadas en los extremos inferiores del teclado. Es SHIFT en los ordenadores ingleses.

FIJA MAYS: Conecta o desconecta el modo mayúsculas, sólo afecta al alfabeto. Es CAPS LOCK en los teclados ingleses.

CTRL: Ya hemos visto una de las utilidades en el manejo del cursor: la otra es imprimir los caracteres de control (veremos su utilidad más adelante).

ESC: Pulsándola una vez detiene momentáneamente el programa que se ejecuta en memoria; si se pulsa otra tecla continúa de nuevo con la ejecución del programa, sino se detiene totalmente. Pulsándola una vez en modo auto o en el de edición se abandonan dichos modos sin modificar la línea. Está en la esquina superior izquierda.

TAB: Sólo sirve en modo programa. Salta un número determinado de espacios a cada pulsación. Situada debajo de ESC.

BORR: Borra el carácter situado a la izquierda del cursor. Está en el extremo



Teclado Amstrad CPC 464.



Teclado Amstrad CPC 664.



Teclado Amstrad CPC 6128.

LENGUAJE BASIC

superior derecho. DEL en los teclados ingleses.

CLR: **Borra el carácter situado debajo del cursor**, moviendo hacia la izquierda el texto ubicado en esa posición hasta el final de la línea. Situada a la derecha de BORR.

EL TECLADO NUMÉRICO: Se compone de las tres columnas de teclas situadas a la derecha al lado de la unidad de disco, se utiliza como un teclado de calculadora, aunque sus teclas se pueden redefinir más fácilmente que el resto de las mismas.

CTRL + FIJA MAYS: **Pone todo el teclado en mayúsculas**, incluyendo números y símbolos especiales.

CTRL + MAYS + ESC: **Reinicializa el ordenador**, es decir efectúa una desconexión de éste sin apagarle.

Practicad con el teclado hasta que estéis seguros de que sabéis cómo se maneja el cursor, las teclas de borrado y podáis imprimir en pantalla todos los caracteres que veis en él.

Los programas

Fundamentalmente un programa es una serie de instrucciones que se dan al ordenador. Es una definición cierta; pero es como decir de un cuadro que es un montón de pintura. Un programa es algo más que eso. Es una herramienta, que como tal debe ser fuerte y no estropearse a la mínima, incluso debe poder soportar ataques y

aguantar funcionando cualquier adversidad. Esto se verá claro a medida que se conozca más cómo se debe programar.

El lenguaje basic

El lenguaje BASIC se desarrolló en el Dartmouth College en Estados Unidos, por Thomas Kurtz y Jhon Kemeny alrededor de 1964. El nombre de Instruction Code (código sim-

plificado) de instrucciones de tipo general para principiantes). Este lenguaje **es interpretado**, esto quiere decir que cada vez que se ejecuta el programa se traduce

la gran cantidad de versiones existentes, lo que ha hecho que cada fabricante de ordenadores la haya ampliado a su gusto. EL BASIC de Amstrad es una versión rápida y potente que incluye características de lenguajes más avanzados, y un tratamiento de gráficos y sonido bastante bueno.



ce a código máquina, como lo haría un traductor simultáneo en una conferencia. Aquí vamos a hacer una aclaración; **el ordenador sólo entiende el lenguaje binario**, llamado código máquina, este lenguaje es difícil de recordar, ya que está formado por sólo 2 números 0 y 1. Por eso se inventaron los lenguajes de programa-

ción que se aproximan algo más al lenguaje humano (inglés casi siempre). Estos lenguajes son un programa más para el ordenador.

Como todos los lenguajes **tiene una sintaxis, de la cual no podemos salir ya que el intérprete no nos entendería**, produciéndose el error. No se debe tener miedo a los errores de este tipo ya que siempre que aparecen se pueden solucionar fácilmente. El más común para los principiantes es "SYNTAX ERROR" y es como si nos dijera "NO ENTIENDO". Todos los errores están explicados en el manual junto a sus posibles causas.

Elementos del Basic

El basic se compone de una serie de elementos. En primer lugar tenemos las instrucciones propias del BASIC, llamadas también sentencias, comandos, palabras clave, etc... El programa puede contener subrutinas, que son pequeños trozos de programa que funcionan independientemente y que son utilizados por un programa principal. Este tema será visto en profundidad.

Las líneas de programa en BASIC **indican el orden en el que se van a ejecutar las instrucciones**; se puede poner más de una instrucción por línea, separadas por dos puntos, en este caso se ejecutarán de izquierda a derecha. No conviene utilizar esta opción por el momento, ya que una instrucción por

línea facilita mucho la detección de errores. Los números de línea son enteros positivos y no pueden exceder de 65535. Por convenio y para facilitar la inserción de líneas, éstas se numeran de diez en diez. Este número sirve para identificarlas y decirle al ordenador en qué orden tiene que realizar la ejecución de éstas.

Datos del Basic

Los datos en BASIC pueden ser de dos clases **numéricos y alfanuméricos**. Los numéricos son datos que sólo pueden contener los dígitos del uno al cero, el punto decimal, los signos más o menos y la letra E, está como símbolo que indica que ese número está en notación exponencial, y se traduce de la siguiente forma: 10E3 es como decir 10×10^3 .

Los datos alfanuméricos son conocidos también como cadenas de caracteres, y admiten cualquier carácter del que disponga el ordenador.

Dentro de los datos podemos tener **constantes o variables**. Una constante es un dato con un valor específico que **no se altera** durante la ejecución del programa. Las constantes pueden ser de cualquiera de los tipos mencionados. Las alfanuméricas deben encerrarse entre comillas. Para las numéricas rigen las normas expuestas en los datos. Ejemplos válidos son: "BASIC"; 123; "buenos días"; 3.23; 7.5E-2; "91/3246578".

Las variables son datos almacenados en la memoria a los cuales hacemos referencia mediante un nombre. Este nombre **debe empezar por una letra** y su longitud puede ser cualquiera, aunque el ordenador sólo tendrá en cuenta los ocho primeros caracteres.

Las variables numéricas pueden ser de dos tipos:

Enteras: Solamente pueden almacenar **números enteros**, es decir que

que da por supuesto el ordenador cuando detecta una variable sin símbolo de identificación, aunque se puede obligar a una variable a ser real añadiéndole al nombre el símbolo !.

Alfanuméricas: Pueden almacenar cualquier carácter, el número máximo de estos admitidos es 256. Deben ir siempre seguidas por el símbolo \$, los datos que se almacenen en esta variable deben estar encerrados entre comillas.

labras reservadas no pueden ser nombres de variables. El único símbolo especial, que no sea un distintivo de tipo, aceptado en un nombre de variable es el de la libra esterlina (£).

Tenemos dos formas de ejecutar las sentencias de BASIC, uno es el modo programa, en el cual introducimos las instrucciones precedidas por un número de línea almacenándose en memoria para su posterior ejecución (esta ejecución se produce al introducir la orden RUN y pulsar ENTER). Modo directo, en la cual se teclan las órdenes sin número de línea y se ejecutan al pulsar la tecla ENTER.

En nuestro próximo artículo, trataremos las distintas formas de comunicarnos con el ordenador, lo que se llama Entrada/salida. Además veremos como pueden nuestros programas almacenar datos para su utilización. Y algunas sentencias de inicio al tratamiento de la pantalla.

Mientras tanto y para que no os aburráis ahí os va un programa que os demostrará lo que Basic puede hacer en unas cuantas líneas. Teclado correctamente y cuando os canséis de él, paradlo (recordar cómo se para un programa).

```

10 CLS
20 PRINT "CUAL ES TU NOMBRE
(PULSA ENTER CUANDO
ACABES)":INPUT NOM$
30 CLS
40 LOCATE 10,12:PRINT NOM$
50 FOR N=1 TO 10
60 LOCATE 1,1:PRINT CHR$(11)
70 FOR A=1 TO 100:NEXT
80 NEXT
90 FOR N=1 TO 10
100 LOCATE 1,25:PRINT
CHR$(10)
110 FOR A=1 TO 100:NEXT
120 NEXT
130 GOTO 50

```



no contengan punto decimal, ni estén en forma exponencial. Se debe poner el símbolo % a continuación del nombre de la variable. El rango de números de este tipo va de -32768 a 32767.

Reales: Estas variables **pueden almacenar cualquier número comprendido entre $\pm 1.7E + 38$** . Es el **tipo implícito**, es decir, el

Algunos ejemplos válidos de nombres de variables son: A\$, c%, fecha\$, ROSAS!, SALDO, DIA, re34!...

Asignaciones válidas son:

A\$="hola"; SALDO=12.3; FECHA\$="27-08-80"

Los nombres que no lleven ningún distintivo de tipo (! % \$) son variables numéricas reales. Las pa-

UTILIDADES

Esta sección tratará de ir formando poco a poco esa *biblioteca de programas tan necesaria a cualquier programador*. Aquellos que estén empezando es probable que no entiendan cómo funcionan los módulos que presentamos aquí, pero deben conservarlos ya que cuando avancen en sus conocimientos le serán de gran utilidad.

Estas rutinas son módulos independientes que necesitan de un programa principal para su manejo, éste es muy fácil de realizar, razón por la cual se omite.

Clasificación

Vamos a ver dos formas de clasificar datos; una es la clasificación por el método de la burbuja, que hemos optimizado para hacerla más rápida. Este método se basa en que los elementos menores (o mayores, según el orden de clasificación), vayan flotando hacia el principio de la matriz en la que se encuentran, de ahí su nombre. Esto se consigue comparando cada elemento con el siguiente, intercambiándolos si el primer elemento es mayor. En esta subrutina utilizamos dos métodos de optimización. El primero de ellos consiste en saber si se han hecho intercambios en una pasada de la matriz; si éstos no se han realizado la matriz está clasificada. Esto lo realizamos con un indicador que se pondrá a cero cada vez que iniciemos una nueva pasada, si se produce algún intercambio éste se pone a 1.

La segunda modificación es hacer que la rutina sólo clasifique aquellos datos que no están ordenados. Este método en cada pasada nos deja los elementos mayores al final de la matriz. Así que cada vez que efectuemos una pasada tendremos un elemento ordenado, por lo que la matriz se nos redu-

ce en uno cada vez, esto lo realizamos con un sencillo contador.

Las variables utilizadas son las siguientes:

B: Matriz de datos, puede ser alfanumérica.

R: Es 1 si se han realizado intercambios.

M: Número máximo de elementos de la matriz.

I: Contador de parte ordenada.

El segundo método de ordenación es el conocido por rápido. Se basa en lo siguiente:

Se busca el elemento central de la matriz y situamos los elementos mayores a un lado y los menores a otro. El proceso se repite con las dos submatrices hasta que obtengamos submatrices de dos elementos, con lo cual tendremos la ordenación realizada.

Las variables utilizadas son las siguientes:

A: Matriz de datos, puede ser alfanumérica.

P: Matriz que almacena el comienzo y final de las divisiones que efectuamos para clasificar.

SP: Controla la posición de cada comienzo y final en la matriz P.

L, R, I, J: Variables de control de comienzo y final.

La clasificación rápida es aconsejable para ordenar más de 50 elementos,

mientras que la de burbuja es mejor para clasificar menos de este número.

Búsqueda

Tenemos dos tipos de búsqueda, llamados secuencial y binario.

La primera consiste en ir explorando la matriz empezando desde el primer elemento, hasta encontrar el dato deseado o el último elemento de la matriz, lo que significa que el dato no se encuentra en la misma. Es un método bastante lento.

La búsqueda binaria sólo se puede realizar con datos previamente ordenados.

Consiste en tomar el primer elemento de la matriz y compararlo con el que se busca, si son iguales termina la búsqueda, si es menor tomamos la parte inferior de la matriz y si es mayor la parte superior; repetimos las operaciones anteriores hasta encontrar el elemento o acabar la matriz.

Las variables utilizadas son las siguientes:

INF: Limite inferior de la matriz.

SUP: Limite superior.

IND: Contiene la posición del elemento a buscar, si éste se encuentra, si no es cero.

I: Elemento central de la parte en que se busca.

Rutina de introducción de datos

Esta rutina de entrada, es algo muy útil ya que nos permitirá hacer inputs, seleccionando el tipo de dato que queremos utilizar, y la longitud de los mismos.

Las variables utilizadas son las que siguen:

E\$: Almacena el carácter teclado.

W\$: Almacena la cadena producida.

A: Longitud de la cadena.

B\$: Almacena los caracteres permitidos. Es muy importante añadir el carácter ENTER (CHR\$(13)) y el DEL (CHR\$(127)).

Refundición

Otra rutina bastante útil, ésta nos permite fusionar el contenido de dos matrices ordenadas en una. Debemos utilizar un número de fin en cada matriz (en nuestro ejemplo 999). Esta rutina se puede utilizar con el disco o cinta para refundir datos ya ordenados y almacenarlos juntos.

Variables utilizadas:

A, B, C: Matrices de datos, pueden ser alfanuméricas.

MA, MB: Almacenan los elementos a fundir.

I: Contador matriz A.

J: Contador matriz B.

K: Contador matriz C.

UTILIDADES

```
260 '-----BURBUJA-----
270 PRINT "CLASIFICANDO"
280 R=1:I=0:WHILE R=1
290 I=I+1:R=0:PRINT ">";
300 FOR N=1 TO M-I
310 IF B(N)>B(N+1) THEN R=1:T=B(N):B(N)=B(N+1)
:B(N+1)=T:PER1=PER1+1
320 COMP1=COMP1+1:NEXT
330 IF I>M-1 THEN R=0
340 WEND
350 PRINT
```

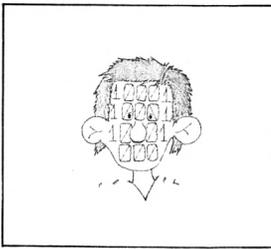
```
500 '-----BUSQUEDA SECUENCIAL-----
510 PRINT:IND=0:PRINT "BUSQUEDA SECUENCIAL"
520 I=0:T2=TIME:S=1:WHILE S
530 I=I+1
540 IF N=A(I) THEN S=0:IND=I
550 IF I>M THEN S=0
560 WEND
570 T1=TIME:IF IND=0 THEN PRINT "NO ENCONTRADO"
ELSE PRINT "ENCONTRADO EN POSICION "IND
580 PRINT "TIEMPO= "INT((T1-T2)/3)" CENTESIMAS
DE SEG.
```

```
600 '-----CLASIFICACION RAPIDA-----
610 PRINT :PRINT "CLASIFICANDO"
620 SP=1:PER=0:COMP=0
630 P(SP,0)=0:P(SP,1)=M
640 WHILE SP>0
650 PRINT ">";
660 L=P(SP,0):R=P(SP,1):SP=SP-1
670 WHILE L<R
680 I=L:J=R:S=1
690 WHILE I<J
700 COMP=COMP+1
710 IF A(I)>A(J) THEN T=A(I):A(I)=A(J):
A(J)=T:S=-S:PER=PER+1
720 IF S=1 THEN I=I+1 ELSE J=J-1
730 WEND
740 IF R-I>J-L THEN 780
750 IF J-1>L THEN SP=SP+1:P(SP,0)=L:P(SP,1)=J-1
760 L=I+1
770 GOTO 800
780 IF I+1<R THEN SP=SP+1:P(SP,0)=I:P(SP,1)=R
790 R=J-1
800 WEND
810 WEND
```

```
330 REM -----RUTINA DE ENTRADA-----
340 PRINT CHR$(24);" ";
350 e$="":w$="":A=0:S=1
360 WHILE S
370 W$=W$+E$
380 PRINT CHR$(24);CHR$(8);E$;CHR$(24);" ";
390 F=1:WHILE F
400 F=0:E$="":WHILE E$="":E$=UPPER$(INKEY$):WEND
410 IF INSTR(B$,E$)=0 THEN F=1:PRINT " ";
420 IF A=0 AND E$=CHR$(127) THEN A=0
:PRINT " ";:F=1
430 IF A>0 AND E$=CHR$(127) THEN PRINT CHR$(24)
:CHR$(8);CHR$(16);CHR$(8);CHR$(16);CHR$(24)
;" "":A=A-1:E$="":W$=LEFT$(W$,LEN(W$)-1):F=1
440 IF E$=CHR$(13) THEN S=0:A=A-1:E$=" "
450 IF A>L THEN PRINT " ";:F=1
460 WEND
470 A=A+1:WEND
480 PRINT CHR$(24);CHR$(8);CHR$(16);
```

```
370 '-----BUSQUEDA BINARIA-----
380 PRINT:INPUT "NUMERO A BUSCAR ",N
390 INF=1:SUP=M:IND=0
400 PRINT:PRINT "BUSQUEDA BINARIA"
410 T2=TIME:S=1:WHILE S
420 I=INT((INF+SUP)/2)
430 IF N=A(I) THEN S=0:IND=I
440 IF N<A(I) THEN SUP=I-1 ELSE INF=I+1
450 IF INF>SUP THEN S=0
460 WEND:T1=TIME
470 IF IND=0 THEN PRINT "NO ENCONTRADO" ELSE
PRINT "ENCONTRADO EN POSICION "IND
480 PRINT "TIEMPO= "INT((T1-T2)/3)"
CENTESIMAS DE SEG.
```

```
500 '-----REFUNDICION-----
510 I=1:J=1:K=1
520 MA=A(I):MB=B(J):S=1
530 WHILE S
540 IF MA=999 AND MB=999 THEN S=0:GOTO 580
550 IF MA=999 THEN GOSUB 630:GOTO 580
560 IF MB=999 THEN GOSUB 600:GOTO 580
570 IF MB>MA THEN GOSUB 600 ELSE GOSUB 630
580 WEND
590 C(K)=999:RETURN
600 '-----COPIA DE A-----
610 C(K)=MA:K=K+1:I=I+1:MA=A(I)
620 RETURN
630 '-----COPIA DE B-----
640 C(K)=MB:K=K+1:J=J+1:MB=B(J)
650 RETURN
```



CODIGO MAQUINA

Con este número iniciamos el curso de programación «LENGUAJE MAQUINA» invitando a los usuarios de AMSTRAD a descubrir las entrañas de su ordenador. Como sabéis es el único lenguaje que él entiende y ejecuta directamente.

La reciente revolución de los ordenadores personales ha acercado a sus usuarios a algo que antes estaba solamente reservado a un determinado sector especializado, «la programación».

Por todos es conocido la gran versatilidad de trabajo de los ordenadores, pero para que un ordenador realice una determinada tarea, antes tendremos que instruirle, indicándole de una forma organizada lo que ha de hacer en cada momento, esto es «hacer una programación adecuada de la tarea que le vamos a encomendar».

Pero el ordenador, como un dispositivo electrónico que es, no entiende ninguna de las palabras que nosotros utilizamos, sólo es capaz de trabajar con niveles de tensión y corriente, usualmente lo hace con niveles de 0 v y 5 voltios, diferenciándose así dos estados posibles para un determinado punto del sistema:

1. Que en dicho punto del sistema no haya tensión (0 v).

Este estado lo representamos por un «0».

2. *Que en dicho punto del sistema haya tensión (5 v).

Este estado lo representamos por un uno «1».

Así toda la información que un ordenador es capaz de manipular es una expresión de unos y ceros correspondientes a dos únicos niveles de tensión posible.

Surge entonces una pregunta:

¿Si un ordenador sólo es capaz de manejar unos y ceros, cómo se realiza la comunicación hombre/máquina?

Esta comunicación se realiza mediante los lenguajes de programación, que podemos dividir en dos grupos:

- 1) *Lenguaje máquina.*
- 2) *Lenguajes de alto nivel.*

Lenguaje máquina

Este es el único lenguaje que el ordenador entiende y es capaz de ejecutar directamente. Pero para hacer una programación adecuada en este lenguaje, nosotros tendremos que aprender a usar ese lenguaje constituido a base de unos y ceros; lo que en un principio nos puede parecer un inconveniente, más tarde veremos que no lo es tanto y que puede aportarnos muchas ventajas.

Lenguajes de alto nivel

Forman parte de este segundo grupo, los lenguajes Basic, Fortran, Logo, Cobol,... etc.

Estos lenguajes son en sí, unos programas que están diseñados para acercar la máquina al lenguaje humano.

Por ejemplo, el Basic utiliza palabras claves como PRINT, GOTO, END, que forman parte del vocabulario inglés y por tanto más fácil de comprender por nosotros. Pero no podemos olvidar en ningún momento que el único lenguaje que el microprocesador entiende es el lenguaje máquina.

Cuando nosotros escribimos una instrucción como PRINT, ésta es reconocida y traducida por el Basic de nuestro ordenador a lenguaje máquina y posteriormente ejecutada.

Conforme a todo lo expuesto veamos algunas de esas ventajas que nos puede aportar la programación en código máquina.

Código máquina

Ventajas que nos aporta:

El proceso de traducción que requieren los programas escritos en lenguajes de alto nivel necesitan de un tiempo, por eso los escritos en código máquina son mucho *más rápidos*.

Por otra parte, para realizar una misma tarea, se emplea *menos memoria* escribiendo en lenguaje máquina que si lo escribimos en lenguajes de alto nivel.

Además en lenguaje máquina podremos realizar tareas que escritas en Basic serían *imposibles de realizar*.

Esperamos que con estas aclaraciones, te animes lo suficiente como para querer introducirte en las profundidades de tu ordenador mediante la programación en código máquina.

Nosotros desde estas páginas pretendemos ayudarte dándote la información que necesitas para que dispongas de toda la potencia de tu ordenador.

Para iniciarnos hemos de formar una buena base. Aclaremos todo lo relacionado con los sistemas de numeración y sobre esto construiremos nuestro edificio.

Sistemas de numeración

El sistema binario y el sistema hexadecimal son dos sistemas de numeración que nos serán muy necesarios para introducirnos en la programación en lenguaje máquina. El conocerlos y habituarnos a trabajar con ellos nos resultará de gran utilidad.

Antes de tratar estos sistemas, vamos a estudiar el sistema decimal, pues al ser el que todos conocemos nos resultará de gran utilidad para comprender cómo se contruyen los demás sistemas.

Sistema decimal

El sistema decimal o *sistema de numeración en base de 10*, es al que todos estamos habituados. Se construye a partir de diez dígitos (0, 1, 2, 3, 4, 5, 6, 7, 8, 9).

Cuando contamos lo hacemos a partir del cero hasta llegar al nueve, que constituyen las unidades de este sistema, para continuar contando pasamos de nuevo al cero y sumamos uno en la columna de las decenas, así formamos el 10, este proceso se va repitiendo de forma tal que con diez dígitos formamos cualquier número.

Si tomamos un número del sistema decimal siempre podremos desarrollarlo de la siguiente forma:

$$846 = 800 + 40 + 6 = (8 \times 100) + (4 \times 10) + (6 \times 1) = (8 \times 10^2) + (4 \times 10^1) + (6 \times 10^0).$$

$$1365 = 1000 + 300 + 60 + 5 = (1 \times 1000) + (3 \times 100) + (6 \times 10) + (5 \times 1) = (1 \times 10^3) + (3 \times 10^2) + (6 \times 10^1) + (5 \times 10^0).$$

En estos desarrollos, observamos que la base de nuestro sistema (10), aparece en cada uno de los sumandos elevada a un exponente, que partiendo de cero, crece de derecha a izquierda.

Sistema binario

El sistema binario utiliza solamente dos dígitos, el 0 y el 1 (base dos). Definiremos un *bit* como uno de esos dos dígitos en el sistema de numeración binaria y corresponde a la unidad de información.

A la agrupación de ocho bits se le conoce como *byte*, también se le suele llamar *octeto*.

Debido a la confusión que se puede producir al hablar de un número decimal o al hacerlo de un número binario, les vamos a diferenciar tal y como lo hace nuestro AMSTRAD; para ello los números binarios irán siempre precedidos de los símbolos «&X».

Así cuando, por ejemplo, digamos &X10110, nos daremos cuenta de que nos referimos a un número binario.

Una vez hechas estas aclaraciones veamos cómo se construye este sistema de numeración en el que sólo disponemos de dos dígitos.

Para contar en binario lo haremos comenzando por el cero, después vendrá el uno; para seguir contando volvemos de nuevo al cero y aumentamos en uno la columna inmediata a la izquierda y continuamos. De esta forma representamos cualquier número disponiendo solamente de dos dígitos.

En la TABLA 1 tienes las equivalencias de los 16 primeros números en los sistemas decimal-binario-hexadecimal; examínala cuidadosamente y después intenta crearla tú mismo.

TABLA 1

Decimal	Binario	Hexadecimal
0	0000	0
1	0001	1
2	0010	2
3	0011	3
4	0100	4
5	0101	5
6	0110	6
7	0111	7
8	1000	8
9	1001	9
10	1010	A
11	1011	B
12	1100	C
13	1101	D
14	1110	E
15	1111	F

Al igual que hacíamos con los números decimales, podemos desarrollar los números binarios, pero hemos de tener en cuenta que la base es 2. Por ejemplo, dado un número binario tal como &X101, lo descomponemos de la siguiente forma:

$$\&X101 = (1 \times 2^2) + (0 \times 2^1) + (1 \times 2^0) = 4 + 0 + 1 = 5 \text{ (en decimal).}$$

$$\&X10010111 = (1 \times 2^7) + (0 \times 2^6) + (0 \times 2^5) + (1 \times 2^4) + (0 \times 2^3) + (1 \times 2^2) + (1 \times 2^1) + (1 \times 2^0) = 128 + 0 + 0 + 16 + 0 + 4 + 2 + 1 = 151.$$

Como podemos comprobar el convertir un número binario en su correspondiente decimal es algo muy sencillo. Observamos que la base del sistema (2), aparece en todos los sumandos elevada a un exponente que comienza en cero para el bit menos significativo y aumenta de derecha a izquierda. De esta forma a cada bit le corresponde un valor. Por ejemplo:

- al bit n.º 5 (posición que ocupa) le corresponde un valor igual a $2^5 = 32$.
- al bit n.º 7 le corresponderá un valor igual a $2^7 = 128$.

Pues bien, a este valor que le corresponde a cada posición ocupada le llamaremos *peso*.

byte	&X 1	0	1	1	0	1	1	1
posición de bits	7	6	5	4	3	2	1	0
peso	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
valor	128	64	32	16	8	4	2	1

Como observamos al pasar de un bit a otro de derecha a izquierda el peso se dobla.

Del decimal al binario

Si lo que deseamos es convertir un número decimal al sistema binario, el proceso es muy sencillo. Para ello el número decimal lo dividimos entre dos, el cociente obtenido, lo volvemos a dividir entre dos y así sucesivamente hasta obtener un cociente cero. El número binario correspondiente se construye con los restos obteni-

CODIGO MAQUINA

dos de forma que, el primero de estos restos lo situamos en la columna de la derecha, el segundo resto una columna más a la izquierda y así sucesivamente con todos los demás. Para comprenderlo mejor, veámoslo con un ejemplo:

Dado el número decimal 43, convertirlo al sistema binario.

Posición del bit

	43 : 2	
bit N.º 0	_____ 1 21 : 2	
bit N.º 1	_____ 1 : 10 : 2	
bit N.º 2	_____ 0 5 : 2	
bit N.º 3	_____ 1 2 : 2	
bit N.º 4	_____ 0 1 : 2	
bit N.º 5	_____ 1 0 (cociente cero)	

43 = &X1 0 1 0 1 1
 5 4 3 2 1 0 N.º de bit (posición)

Podemos comprobar que efectivamente $43 = \&X101011$. Realizando el proceso contrario: el número binario &X101011, pasarlo al sistema decimal. Como ya vimos $\&X101011 = (1 \times 2^5) + (0 \times 2^4) + (1 \times 2^3) + (0 \times 2^2) + (1 \times 2^1) + (1 \times 2^0) = 32 + 0 + 8 + 0 + 2 + 1 = 43$, como todos podemos comprobar la conversión entre ambos sistemas es muy sencilla.

Sistema hexadecimal

El sistema hexadecimal o sistema de numeración en base 16, necesitará de dieciséis dígitos.

Para rerepresentarlos, utilizamos los 10 dígitos del sistema decimal (del 0 al 9) y las seis primeras letras del alfabeto (A, B, C, D, E, F), de forma que a la «A» le corresponda el número 10 en decimal, a la «B» el 11 y así sucesivamente hasta llegar a la «F» que sería el 15 (ver TABLA 1).

Anteriormente tomamos un convenio para distinguir a los números decimales de los binarios, ahora lo haremos para distinguir a los hexadecimales que irán siempre precedidos del símbolo «&».

Todo lo tratado anteriormente es válido para este sistema, sólo hemos de tener en cuenta que la base es ahora 16.

Por ejemplo, podremos desarrollar cualquier número hexadecimal:

$$\&FF = (F \times 16^1) + (F \times 16^0) = (15 \times 16^1) + (15 \times 16^0) = 240 + 15 = 255$$

$$\&AC3 = (A \times 16^2) + (C \times 16^1) + (3 \times 16^0) = (10 \times 16^2) + (12 \times 16^1) + (3 \times 16^0) = 2560 + 192 + 3 = 2755$$

La importancia de este sistema de numeración para la programación en código máquina, radica en que cada una de sus cifras equivalen a medio octeto en binario (un cuarteto), o lo que es lo mismo, con sólo dos cifras en hexadecimal representamos lo que con ocho bits en

el binario. Esto supone el no tener que manejar tantos 1 y 0. Nos resultará mucho más cómodo decir &8A que decir &X10001010; por otra parte cometeremos menos errores.

Para pasar del hexadecimal al binario es muy simple. El método consiste en sustituir cada cifra del número hexadecimal por los cuatro que le corresponden en el sistema binario. Veamos un ejemplo y quedará más claro: Pasemos &AC3 a binario:

En la TABLA 1 vemos que $A = \&X1010$, $C = \&X1100$, $3 = \&X0011$, luego $\&AC3 = \&X101011000011$.

Te aconsejamos que mientras no sepas de memoria la tabla de conversiones (TABLA 1), antes de usarla, intentes hacer tus deducciones; de esta forma tomarás más confianza. Así por ejemplo si no recuerdas que $\&A = \&X1010$ basta con aplicar el sistema que explicamos anteriormente.

&A = 10 (en decimal)

	10 : 2	
bit N.º 0	_____ 0 5 : 2	
bit N.º 1	_____ 1 2 : 2	
bit N.º 2	_____ 0 : 1 : 2	
bit N.º 3	_____ 1 0 (cociente cero)	

&A = &X1010 = 10

Del decimal al hexadecimal

El proceso es igual que el caso de la conversión del sistema decimal al binario, pero tendremos en cuenta que la base es 16, y estableceremos la correspondencia de los restos obtenidos con el sistema hexadecimal para expresar el resultado.

Ejemplo:

Pasemos el n.º 7852 al sistema hexadecimal.

7852 : 16		
&C _____ 12 490 : 16		12 = &C
&A _____ 10 30 : 16		10 = &A
&E _____ 14 1 16		14 = &E
&1 _____ 1 0		1 = &1

7852 = &1EAC

Y hasta aquí es todo lo que se refiere a la conversión entre los sistemas de numeración que vamos a necesitar.

Para que tu mismo te puedas proponer ejercicios de conversión de un sistema al otro, te ofrecemos un programa que va incluido con el nombre de «CONVER».

Este programa te preguntará el número que deseas convertir (en cualquier sistema) y te dará la notación correspondiente en los demás sistemas.

Para ello sólo has de recordar que los números binarios y hexadecimales han de ir precedidos de «&X» y «&» respectivamente.

**FICHERO
COLECCIONABLE
CODIGO MAQUINA**

CODIGO MAQUINA

TABLA DE CONVERSION DE HEXADECIMAL A DECIMAL PARA EL BYTE MENOS SIGNIFICATIVO

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
2	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47
3	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63
4	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79
5	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95
6	96	97	98	99	100	101	102	103	104	105	106	107	108	109	110	111
7	112	113	114	115	116	117	118	119	120	121	122	123	124	125	126	127
8	128	129	130	131	132	133	134	135	136	137	138	139	140	141	142	143
9	144	145	146	147	148	149	150	151	152	153	154	155	156	157	158	159
A	160	161	162	163	164	165	166	167	168	169	170	171	172	173	174	175
B	176	177	178	179	180	181	182	183	184	185	186	187	188	189	190	191
C	192	193	194	195	196	197	198	199	200	201	202	203	204	205	206	207
D	208	209	210	211	212	213	214	215	216	217	218	219	220	221	222	223
E	224	225	226	227	228	229	230	231	232	233	234	235	236	237	238	239
F	240	241	242	243	244	245	246	247	248	249	250	251	252	253	254	255

**FICHERO
COLECCIONABLE
«DICCIONARIO
DE BASIC»**

DICCIONARIO DE BASIC

USO DEL DICCIONARIO: Las palabras escritas con MAYUSCULAS son palabras reservadas de BASIC. Las escritas con minúscula son el tipo de datos que hay que escribir. Todo lo encerrado entre corchetes son parámetros opcionales. Los paréntesis forman parte de la sintaxis de la palabra.

ABS: Esta función calcula el valor absoluto de una expresión numérica, es decir, no tiene en cuenta el signo dejando la expresión como positiva.

SINTAXIS: ABS (exp. numérica) TOKEN: 255.

Ej. ABS(-100*4)=400

AFTER: Es una orden que invoca a una subrutina BASIC después de transcurrido un tiempo especificado por nosotros. El retardo del temporizador viene expresado en cincuentavos de segundo (un segundo = 50). Hay cuatro temporizadores numerados de 10 al 3.

SINTAXIS: AFTER ret. tem. / ,num. tem./ GOSUB num. lin. TOKEN: 128.

**BOLETIN DE
SUSCRIPCION**

SEIS NUMEROS

- Deseo suscribirme a la revista AMSTRAD CASSETTE de EDICIONES MANALI.
- Recibiré en cada número, un cassette grabado con programas de juegos.
- Con el primer ejemplar me remitirán el **OBSEQUIO de un RELOJ DIGITAL CALCULADORA.**

- COMIENZA LA SUSCRIPCION CON EL N.º _____ (inclusive)

NOMBRE Y APELLIDOS _____

DOMICILIO _____ N.º _____ PISO _____

CIUDAD _____ PROVINCIA _____

CODIGO POSTAL _____ EDAD _____ TELEFONO _____

- FORMA DE PAGO (señale con una X)

Talón bancario

Giro Postal n.º _____

Contra reembolso, más gastos de envío

FIRMA: _____

- Remitir a: EDICIONES MANALI, S.A. P.º de la Esperanza, 22 · 28005 MADRID.

**OBSEQUIO
RELOJ DIGITAL
CALCULADORA**

PRECIO DE SUSCRIPCION
3.850 PTAS. IVA INCLUIDO



CODIGO MAQUINA

TABLA DE CONVERSION DE HEXADECIMAL A DECIMAL PARA EL BYTE MAS SIGNIFICATIVO

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	0	256	512	768	1024	1280	1536	1792	2048	2304	2560	2816	3072	3328	3584	3840
1	4096	4352	4608	4864	5120	5376	5632	5888	6144	6400	6656	6912	7168	7424	7680	7936
2	8192	8448	8704	8960	9216	9472	9728	9984	10240	10496	10752	11008	11264	11520	11776	12032
3	12288	12544	12800	13056	13312	13568	13824	14080	14336	14592	14848	15104	15360	15616	15872	16128
4	16384	16640	16896	17152	17408	17664	17920	18176	18432	18688	18944	19200	19456	19712	19968	20224
5	20480	20736	20992	21248	21504	21760	22016	22272	22528	22784	23040	23296	23552	23808	24064	24320
6	24576	24832	25088	25344	25600	25856	26112	26368	26624	26880	27136	27392	27648	27904	28160	28416
7	28672	28928	29184	29440	29696	29952	30208	30464	30720	30976	31232	31488	31744	32000	32256	32512
8	32768	33024	33280	33536	33792	34048	34304	34560	34816	35072	35328	35584	35840	36096	36352	36608
9	36864	37120	37376	37632	37888	38144	38400	38656	38912	39168	39424	39680	39936	40192	40448	40704
A	40960	41216	41472	41728	41984	42240	42496	42752	43008	43264	43520	43776	44032	44288	44544	44800
B	45056	45312	45568	45824	46080	46336	46592	46848	47104	47360	47616	47872	48128	48384	48640	48896
C	49152	49408	49664	49920	50176	50432	50688	50944	51200	51456	51712	51968	52224	52480	52736	52992
D	53248	53504	53760	54016	54272	54528	54784	55040	55296	55552	55808	56064	56320	56576	56832	57088
E	57344	57600	57856	58112	58368	58624	58880	59136	59392	59648	59904	60160	60416	60672	60928	61184
F	61440	61696	61952	62208	62464	62720	62976	63232	63488	63744	64000	64256	64512	64768	65024	65280

**FICHERO
COLECCIONABLE
CODIGO MAQUINA**

DICCIONARIO DE BASIC

AND: Operador lógico que equivale a realizar una multiplicación en binario; es decir, si uno de los operadores es falso el resultado de la operación es falso. El *argumento* puede ser cualquier expresión.

SINTAXIS: arg. AND arg.

ASC: Esta función calcula el código ASCII del primer carácter de una expresión alfanumérica.

SINTAXIS: ASC (expr.)

TOKEN:

ATN: Es una función que calcula el arco tangente de una expresión numérica. Puede venir dada en grados o radianes.

SINTAXIS: ATN (expr.)

TOKEN:

**FICHERO
COLECCIONABLE
«DICCIONARIO
DE BASIC»**

DOCE NUMEROS

- Deseo suscribirme a la revista AMSTRAD CASSETTE de EDICIONES MANALI.
- Recibiré en cada número, un cassette grabado con programas de juegos.
- Con el primer ejemplar me remitirán el **OBSEQUIO de un JOYSTICKS y un RELOJ DIGITAL CALCULADORA.**

- COMIENZA LA SUSCRIPCION CON EL N.º _____ (inclusive)

NOMBRE Y APELLIDOS _____

DOMICILIO _____ N.º _____ PISO _____

CIUDAD _____ PROVINCIA _____

CODIGO POSTAL _____ EDAD _____ TELEFONO _____

- FORMA DE PAGO (señale con una X)

Talón bancario

Giro Postal n.º _____

Contra reembolso, más gastos de envío

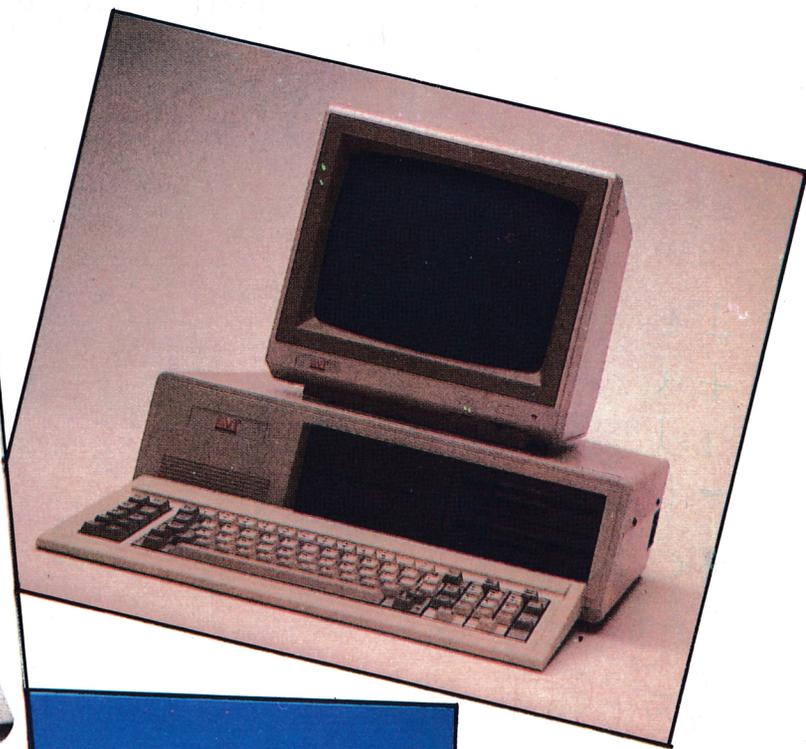
FIRMA: _____

- Remitir a: EDICIONES MANALI, S.A. P.º de la Esperanza, 22 · 28005 MADRID.

**OBSEQUIO
JOYSTICKS y
RELOJ DIGITAL
CALCULADORA**

PRECIO DE SUSCRIPCION
7.650 PTAS. IVA INCLUIDO

**BOLETIN DE
SUSCRIPCION**



MAYORISTA NACIONAL DE PRODUCTOS INFORMATICOS

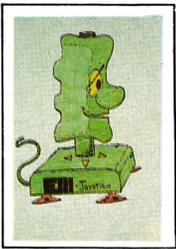
**DISTRIBUCION
INTROLINE, S.A.**

LE GARANTIZAMOS:

- **SERVICIO CONSTANTE**
- **RAPIDEZ EN LA ENTREGA**
- **APOYO EN SU GESTION COMERCIAL**

¡NO PIERDA ESTA OPORTUNIDAD!

**LLAMENOS A LOS TELEFONOS: 435 25 49 ó 470 13 62
DE MADRID**



JUEGOS

NEPTUNO

«Aquella noche el peñón parecía embrujado; el cielo completamente encapotado escondía una luna que presentía los acontecimientos, el silencio nocturno huía de la bahía mientras el mar dormía inquieto. Y nosotros sumergidos a una veintena de pies, esperábamos los primeros rayos de luz, la operación *NEPTUNO* iba a comenzar...»

Nos encontramos ante un juego de simulación bélica. Esta vez la misión puede parecernos un tanto familiar; la acción se desarrolla en Gibraltar; nuestro objetivo, recuperar el *Peñón*. El programa está basado en este tema; la recuperación de Gibraltar. Por supuesto, una maniobra militar con tal objetivo tan solo es racional de este modo, dentro de una pantalla de ordenador, un juego.

Desarrollo

Lo primero que debemos hacer es entrar con el cassette o disco en su unidad correspondiente, teclear RUN "NEPTUNO" y pulsar ENTER. Tras la rutinaria introducción de nuestro nombre, tenemos que descifrar una clave que nos es dada. Contiene seis dígitos y podemos tener la buena suerte de encontrarla ya resuelta. La clave corresponde a un texto, que escrito en la parte superior de la pantalla nos ayudará a descifrarla.

Ojo, no te entretengas mucho en este punto dado que cuanto más pronto la resuelvas, más puntos obtendrás. Cuando consigas la clave, pulsa ENTER, tecléala y vuelve a pulsar ENTER. Si la clave es correcta pasarás a otra fase del juego, si no lo es po-

drás seguir intentándolo hasta que el tiempo se agote.

Fase primera

Habiendo superado es-

He aquí una breve descripción de los mandos del submarino:

Para aumentar o disminuir la velocidad cuentas con las teclas **Q** y **A**. Es lo primero que debes hacer

go sin estar sumergido, correrás un alto riesgo de ser alcanzado por sus defensas.

Para movernos por la bahía tenemos la ayuda de la computadora central del submarino. Esta nos muestra un mapa del escenario geográfico de la misión, en el que debemos mover nuestro barco sobre la posición de cualquier unidad enemiga. Tanto nuestro barco como los de la potencia opuesta están representados por medio de puntos. Puedes mover nuestro barco con las teclas del teclado numérico de la manera siguiente:

ARRIBA	8
ABAJO	2
DERECHA	6
IZQUIER.	4

Antes de nada, asegúrate de haber aumentado la velocidad del submarino. El número de barcos a



ta etapa nos adentraremos en las siguientes del juego: Dirigir un submarino nuclear hacia la bahía de Algeciras y destruir los buques enemigos que allí se encuentran.

al comenzar el juego; dar velocidad al submarino.

Otras teclas son **W** y **S**, su función es la de controlar la profundidad. Ten cuidado pues si tratas de perseguir un barco enemi-

destruir variará dependiendo del nivel en el que nos encontremos.

Hemos de prestar atención al aire, que irá disminuyendo poco a poco y su fin impondrá el de nuestra misión.

También el número de torpedos es limitado; procura no desperdiciarlos. Una vez situados en la zona de ataque al enemigo oiremos un sonido distinto al del sonar, que nos anuncia esta situación. En el radar situado en el extremo superior izquierdo, aparecerá un punto que representa la situación enemiga respecto a nuestra proa. Para poder ver al buque enemigo a través del periscopio, debemos situar tal punto en la parte superior del radar.

Esto lo conseguiremos mediante las teclas del **cursor** → ← o bien con el **joystick**.

Si nuestra velocidad es mayor que la del enemigo cada vez que este aparezca por la mirilla, lo hará a un tamaño mayor que la vez anterior.

Para efectuar disparos de torpedos pulsa la tecla «**copy**», o bien el **disparador de tu joystick**. Si el barco enemigo es alcanzado podremos ver cómo se hunde frente a nosotros.

Para concluir con éxito esta primera fase debemos «limpiar» de enemigos la bahía, es decir, acabar con todos los buques.

La puntuación que obtendremos por cada hundimiento será proporcional a la distancia a la que se encuentre el barco, a más distancia más puntos y viceversa.

Fase segunda

El submarino emerge y entre las olas del mar podemos observar el **Peñón de Gibraltar**. Nuestra misión aquí es acabar con todas las unidades aéreas enemigas que nos salgan al paso.

He aquí la forma de hacerlo: En los bordes de la pantalla aparece una raya horizontal y vertical, que nos indica una coordenada física del escenario.

Este es nuestro punto de mira, para abatir una aeronave enemiga, debemos encuadrarla entre los ejes y pulsar el disparador. Usa las teclas del **cursor** y «**copy**» o bien el **joystick**.

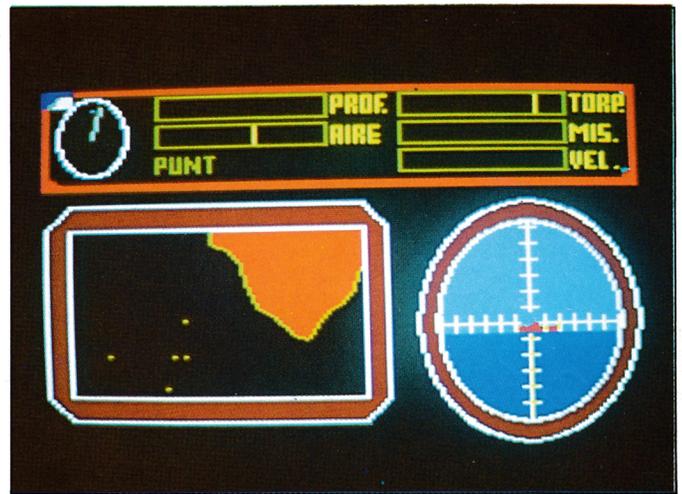
El número de aviones, al igual que el de barcos depende del nivel de dificultad anteriormente elegido.

Las fuerzas enemigas también tienen sus medios de ataque. Debes impedir que se acerquen demasiado, pues si lo hacen sus disparos alcanzan el submarino destruyéndolo.

Las trayectorias del ataque enemigo están determinadas al azar, por lo que tu atención tendrá que ser total a la hora de localizar y alcanzar el blanco enemigo.

Aunque nuestra munición es inagotable, no puedes mantener el disparo pulsado todo el tiempo. Tus disparos, aunque no continuos, si pueden ser muy frecuentes.

Y este es básicamente el juego, esperamos que sea de vuestro agrado y consigáis aunque sea en el ordenador, sentir os protagonistas de la polémica de Gibraltar. **¡Buena suerte!**



Hemos torpedeado un barco.



Avión aniquilado.



El Neptuno ha sido destruido.

RECOMPENSA



SE BUSCAN

PROGRAMAS DE JUEGOS

PAGAMOS de 100.000 a 150.000 ptas. por cada uno

CONDICIONES:

- El programa estará documentado por escrito, con las correspondientes instrucciones y funcionamiento del mismo.
- Indicar nombre, apellidos, edad, dirección, código postal, provincia y teléfono con letra muy clara.
- Abstenerse de enviar juegos que no sean versiones originales o no poseáis los derechos.
- Serán rechazados directamente los programas sospechosos de plagio o los de mala calidad.
- Si es el tuyo el seleccionado, te mandaremos una notificación. A partir de ese momento deberás ponerte en contacto con RECOMPENSA «SE BUSCAN», para efectuar los trámites oportunos.
- Inmediatamente recibirás un talón nominativo por valor de 100.000 a 150.000 ptas.
- Remitir la grabación en disco o cassette a EDICIONES MANALI, S.A. P.º de la Esperanza, 22. 28005 MADRID.

Información para distribuidores de **INDESCOMP, S.A.**



IDEALOGIC® SA

Calle Valencia, 85 - 08029 BARCELONA - Télex: 54554 DLGC
Teléfonos 253 86 93 - 253 89 09 - 253 90 45 - 253 74 00

La máquina de escribir revolucionaria



**AMSTRAD
PCW 8256***

UN COMPLETO PROCESADOR DE TEXTOS + UN POTENTE ORDENADOR

Prepárese a cambiar su máquina de escribir. A poner al día su empresa, su despacho o su consulta con el **AMSTRAD PCW 8256**: la nueva y revolucionaria máquina de escribir (procesador de textos + ordenador personal).

¿QUE PUEDE HACER EL PROCESADOR DE TEXTOS PCW 8256 POR USTED?

Sencillamente, mucho más de lo que hacen las máquinas de escribir convencionales. Fíjese: **Escribir** cualquier texto. **Corregir**: borrar, insertar o corregir desde una letra a párrafos enteros. **Componer páginas**: definir "pasos" entre letras y entre líneas; ajustar márgenes, numerar páginas, seleccionar tamaños de papel, cabeceras, pies, etc. **Conservar** lo escrito en discos con capacidad para cientos de folios para volver a localizarlo cuando Usted desee y trabajar sobre ello de nuevo. **Imprimir** con más de 400 tipos de escritura (normal, cursiva, negrita, subrayado, subíndices, superíndices...) en calidad normal o de alta calidad, en papel continuo u hojas sueltas. **Personalizar** cartas para sus mailings o envíos, etc.

¿QUE PUEDE HACER EL ORDENADOR PCW 8256 POR USTED?

Sencillamente, ayudarle a llevar el trabajo más pesado y rutinario sea cual sea su actividad profesional: previsiones y control de presupuestos; archivar, clasificar y localizar ficheros y datos; organizar su contabilidad, confeccionar su Declaración de la Renta, analizar resultados, programar sus objetivos, controlar su almacén y emitir facturas... etc. En definitiva, todo aquello que resuelve un ordenador de altas prestaciones, pero a un precio increíblemente más bajo.

¿QUE TIENE QUE HACER USTED?

Sencillamente, casi nada.

El PCW 8256 se maneja con tanta facilidad que sólo es cuestión de conectar el cable a la red (un sólo cable) y ponerse a trabajar.

A través de sus sencillos "menús" en castellano, aún sin conocimientos previos, Usted o su secretaria podrán crear minutos, informes, presupuestos, cartas, etc. Archivarlos en discos, recuperarlos en cuestión de segundos y hacer las correcciones precisas, etc., etc.

Con el PCW 8256, Usted, sin apenas esfuerzo entrará de lleno en el mundo de la informática personal.

No espere más y cambie a la máquina de escribir revolucionaria de Amstrad. Comprobará su ventaja automáticamente.

Configuración y características técnicas de la máquina de escribir revolucionaria.

Unidad central (256 K RAM)
Un teclado profesional en castellano.
Unidad de disco (180 K por cara)
Pantalla de alta resolución
Impresora de alta calidad (N.L.Q.)

Programas en discos:

- Procesador de textos Locoscript
- Sistema operativo CP/M Plus
- Mallard BASIC con sistema JETSAM para ficheros indexados
- Lenguaje DR LOGO
- Diversas utilidades
- Completa documentación y manuales en castellano.

Existe también la versión **PCW 8512** con 512 K RAM y la segunda Unidad de Disco de 1 Mbyte incorporada.
P.V.P. 149.900 Ptas.

Solicite demostración en el distribuidor AMSTRAD más próximo.
Descubrirá la máquina de escribir del futuro.

AMSTRAD

ESPAÑA

GRUPO INDESCOMP

C/ Aravaca, 22 - 28040 Madrid - Tel. 459 30 01. Télex 47660 INSC-E - FAX - 459 22 92
Delegación en Cataluña: C/ Tarragona, 110. Tel. 325 10 58 - 08015 Barcelona

PARA MAS INFORMACION, RUEGO:

- DEMOSTRACION EN MI EMPRESA/DOMICILIO
 ENVIO DOCUMENTACION POR CORREO

D./ EMPRESA _____
DOMICILIO _____ CP _____
CIUDAD _____ PROVINCIA _____
TELEFONO _____

ENVIAR A: INDESCOMP, Aravaca, 22 - 28040 MADRID