

APRENDE A PROGRAMAR

EL BANCO DE DATOS

AVENTURAS



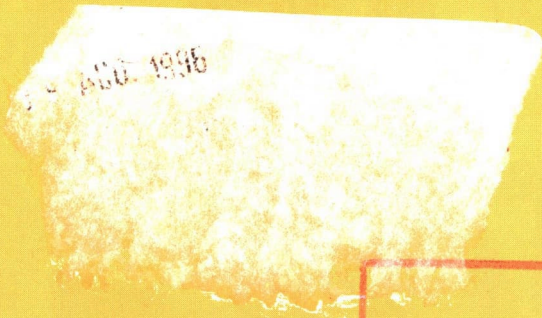
Ediciones Generales Anaya

PARA SPECTRUM
Y COMMODORE

172

Catedra
31-1-86

901



EXPURGO



**DONACION DEL MINISTERIO
DE CULTURA**

Diseño: Cooper West
Editor: James McCarter
Programas: Marcus Milton
Ilustraciones: Gerard Brown
Andrew Farmer

Traducción: Pedro Garre del Olmo

Agradecemos su colaboración a
J. David García Rodríguez

Queda prohibida la reproducción total o parcial de la presente obra bajo cualquiera de sus formas, gráfica o audiovisual, sin la autorización previa y escrita del editor, excepto citas en revistas, diarios o libros, siempre que se mencione la procedencia de las mismas.

Título original: CREATING A DATABASE

© 1985, Aladdin Books Ltd.

© 1985, de la edición española,

Ediciones Generales Anaya

Villafranca, 22. 28028 Madrid

I.S.B.N.: 84-7525-316-4

Depósito legal: M. 39.268/1985

Impreso por: Melsa. Ctra. de Fuenlabrada a Pinto, km. 21,800. Pinto (Madrid)

Impreso en España - Printed in Spain

681.3
Rod
bam

B. - 47.765

APRENDE A PROGRAMAR

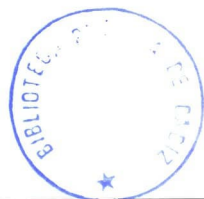
EL BANCO DE DATOS

AVENTURAS



Steve Rodgers y Marcus Miltón

Ediciones Generales Anaya



An illustration of school supplies on a blue grid background. In the top left, a spiral-bound notebook with a green cover is open. A wooden pencil lies diagonally across the notebook's pages. In the bottom left, a yellow folder holds a black floppy disk. On the right side, three pencils (blue, red, and orange) are standing upright. A yellow eraser is partially visible at the bottom right.

Introducción

Los juegos de aventuras transcurren en el mundo de tu imaginación. Son como un cuento, y pueden tener la forma que tú quieras. Puedes ser el único superviviente de una expedición espacial a un planeta lejano, o encontrarte atrapado dentro de los circuitos de un ordenador extraterrestre. El juego que se desarrolla en este libro tiene lugar en un mundo submarino, con tiburones, caballos de mar y esqueletos de exploradores como tú.

En términos informáticos, un juego de aventuras requiere una gran *base de datos* para almacenar todas las descripciones que se dan en el juego. Una base de datos es un conjunto de informaciones que el ordenador utiliza de acuerdo con las instrucciones que se le dan en el programa. El jugador utiliza verbos y nombres para indicar la acción que desea realizar, y el ordenador debe reconocer y procesar este mensaje correctamente. Los textos se guardan como *cadena*s de caracteres, y los juegos de aventuras aprovechan que tanto las cadenas como las partes de una cadena se pueden identificar y operar independientemente. En este libro, el juego está dividido en dos secciones. La primera sección desarrolla el programa principal y la parte de datos; la segunda explica cómo los verbos que usa el jugador llevan al ordenador a la parte correspondiente del programa y cómo dirigen el proceso del juego.

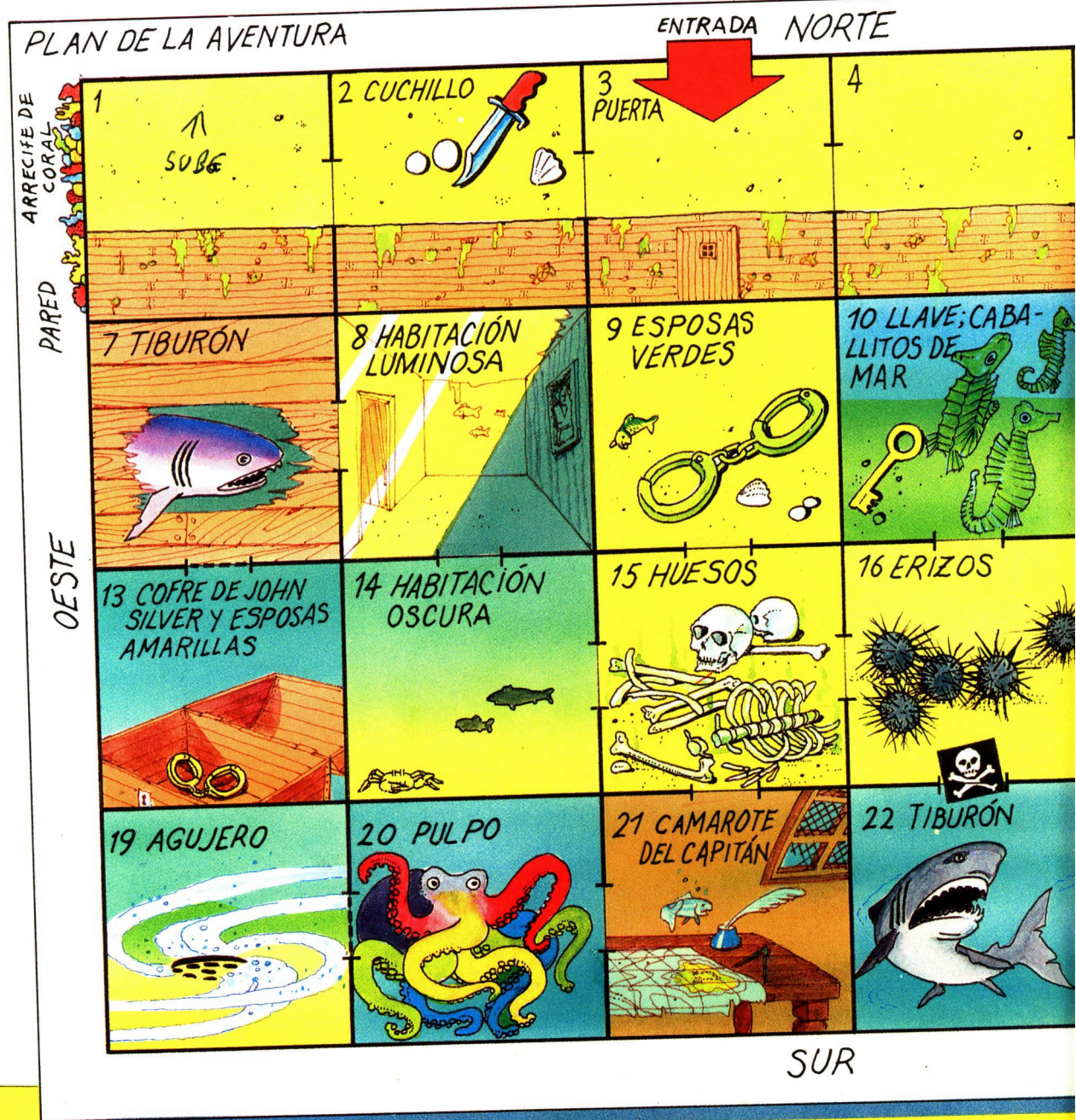
Contenido

El plano	8
El almacenamiento de datos	10
EL PROGRAMA PRINCIPAL	13
SPECTRUM	14
COMMODORE	19
LAS RUTINAS DE LOS VERBOS	25
SPECTRUM	26
COMMODORE	32
Listado de los programas	39

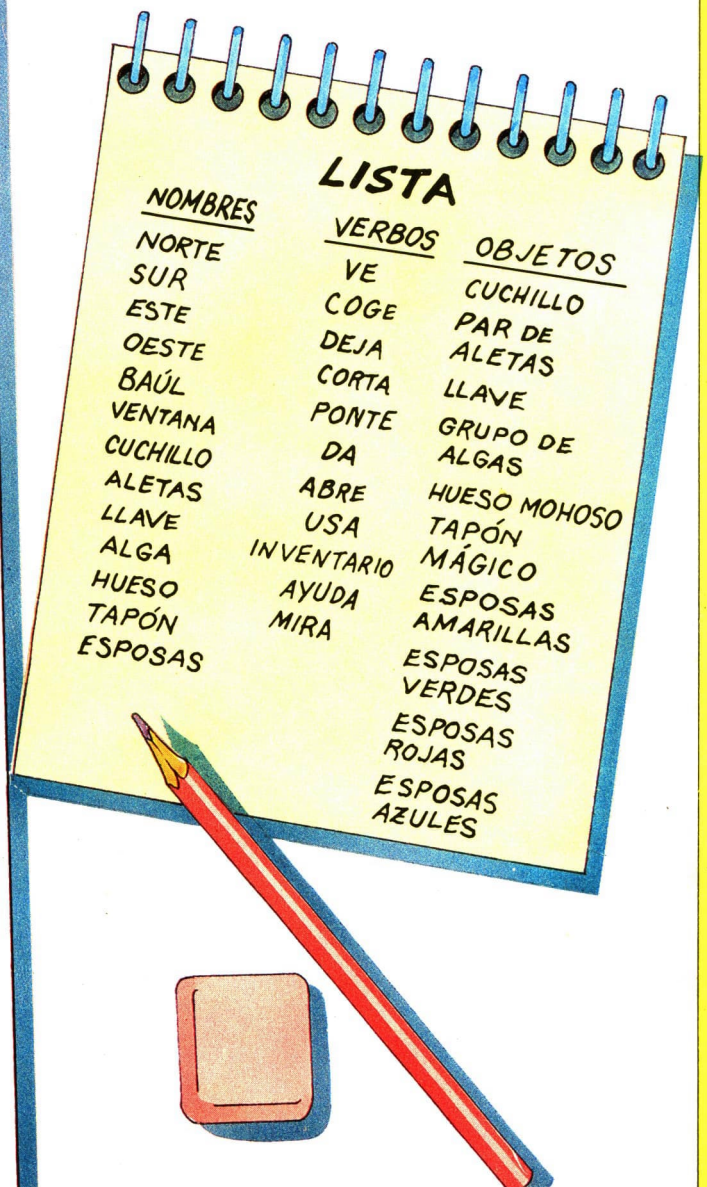
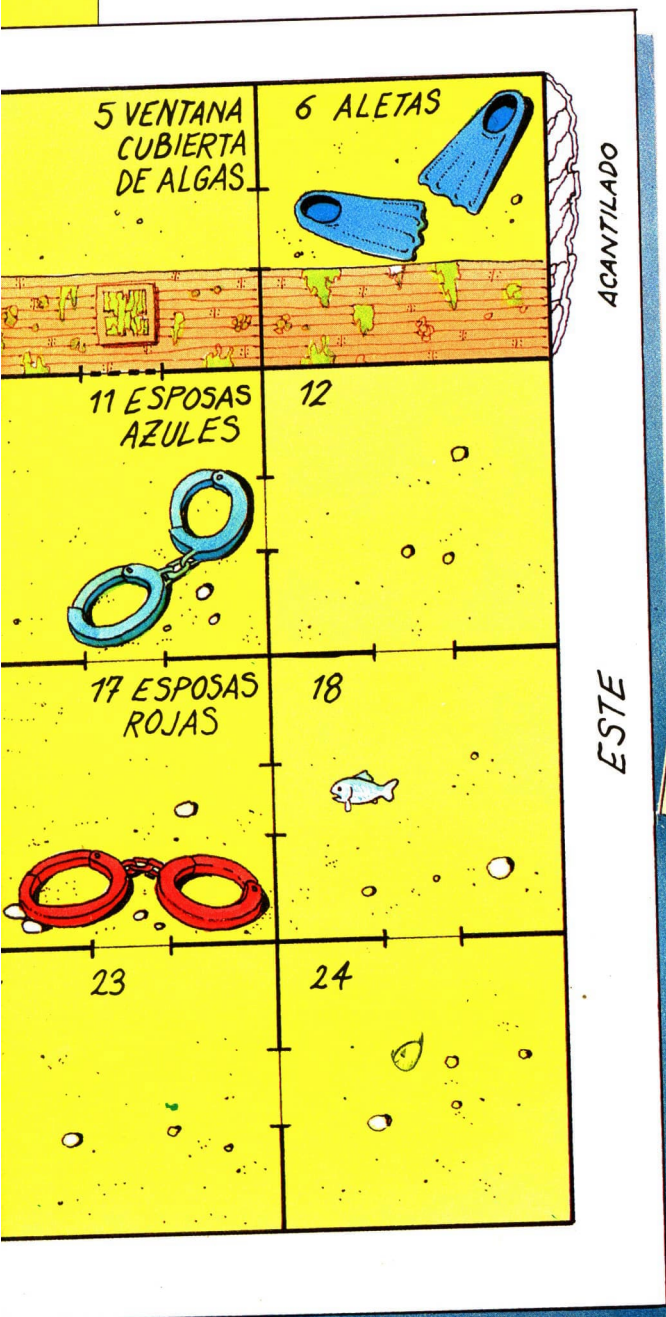
```
30 GOSUB 1000: REM ENTRADA
40 GOSUB 2000: REM SELECCION
50 IF FG<>1 AND FP<>1 THEN GOTO 30
60 GOSUB 5500: REM FIN
70 STOP
1000 REM *****ENTRADA*****
1010 VB$="": NO$="": R$=""
1020 INPUT "QUE HAGO AHORA ? ";R$
1030 FOR I=1 TO LEN(R$)
1040 IF MID$(R$,I,1)=" " THEN VB$=LEFT$(R$,3):
      NO$=RIGHT$(R$,I+1): GOSUB 1500: I=LEN(R$)
1050 NEXT I
1060 IF NO$<>"" THEN RETURN
1070 R$=LEFT$(R$,3)
1080 IF R$="NOR" OR R$="SUR" THEN
      R$="DES" THEN
1090 IF R$
```


El plano

El primer paso para escribir un juego de aventuras es hacer un plano del territorio que el jugador va a explorar. Suele consistir en una serie de habitaciones o posiciones. En nuestro juego, hay 24 habitaciones dispuestas en un rectángulo de 4×6 . Cada habitación tiene una o varias puertas que conducen a otras habitaciones, y a medida que se juega, se informa al jugador de las salidas disponibles. En algunas habitaciones hay objetos que pueden ser de utilidad más tarde, y en otras hay obstáculos que deben ser superados. El objeto del juego es colocar el tapón en el agujero de la habitación 19, pero antes debes pasar junto al pulpo de la habitación 20.



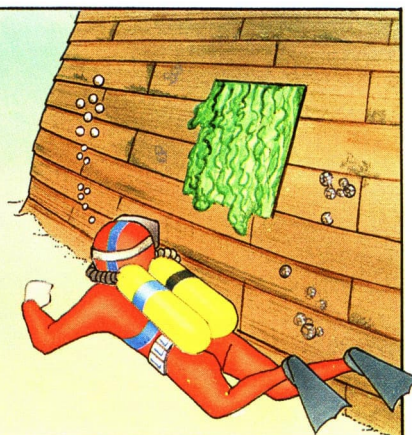
Para jugar, debes darle instrucciones al ordenador, usando un conjunto de verbos y palabras que éste entienda. Puedes ir de una habitación a otra tecleando "VE AL NORTE" (o a otra cualquiera de las cuatro direcciones). El verbo "MIRA" es útil, pues revelará cosas de la habitación que no se ven a primera vista. Si en cualquier momento tecleas "AYUDA", aparecerá la lista de verbos que puedes usar, por si los has olvidado. Para coger los objetos que te encuentres en tu camino, teclea "COGE EL CUCHILLO" (o cualquier otro objeto). Al planear el juego, plantéalo primero en un papel, para asegurarte de que funciona.



El almacenamiento de datos

Una vez tengas preparada la información, el siguiente paso es ponerla de forma que el ordenador la pueda entender. Esto se hace en las instrucciones **DATA** que se indican aquí. En las líneas 6000 a 6200 (*ver la página contigua*) se describe la situación de cada una de las habitaciones. De hecho sólo hay 20 situaciones diferentes —si observas el mapa, verás que las habitaciones 12, 18, 23 y 24 comparten la misma descripción (línea de **DATA** 6120). Los **DATA** se pueden teclear ya, pues el ordenador coloca los números de línea en el orden correcto. Puedes introducir el resto del programa en el orden que se indica en el libro.

En los juegos únicamente de texto, como éste, tienes que imaginarte las escenas; por ello, tu aventura será más entretenida si tus **DATA** son extensos y describen bien el paisaje. Las declaraciones **DATA** pueden ser mucho más largas de las dadas en la página contigua, pero debes recordar que el número de caracteres que uses depende de las limitaciones particulares impuestas por tu ordenador. Tus sentencias **DATA** podrían ser como ésta: 6050 **DATA** «Estás en el fondo del mar. Delante de ti, hacia el sur, hay un viejo barco pirata con un extraño parche de algas en él».



El Commodore permite teclear líneas de 80 caracteres como máximo. Los usuarios de un Spectrum podrán teclear líneas de programa muy largas. No obstante, el introducir más de 200 caracteres se hace aburrido, ya que el ordenador opera muy lentamente. Si tus **DATA** son mayores que los nuestros, tienes que contar el máximo número de caracteres que has utilizado dentro de las comillas (incluyendo los espacios en blanco) y reemplazar el número 176 que se encuentra entre paréntesis en el primer **DIM** de la línea 5050 (*véase página 20*).

Cuando teclees los **DATA** debes tener mucho cuidado en formatearlos, de forma que las palabras no aparezcan partidas por la mitad en la pantalla. El mejor modo de hacerlo es contar el número de caracteres que has introducido en cada línea. Si ves que no vas a poder colocar una palabra al final de una línea sin partirla por la mitad, teclea espacios en blanco hasta terminar la línea y pon la palabra en la siguiente. El Spectrum permite un máximo de 32 caracteres por línea, mientras que el Commodore permite 40.

6000 REM *****descripciones*****

6010 DATA "Estas en el fondo del mar. El camino hacia el oeste esta bloqueado por un arrecife de coral."

6020 DATA "Estas en el fondo del mar. En el sur, una pared se alza enfrente de ti."

6030 DATA "Estas enfrente de una puerta de madera que esta en la pared. No ves el picaporte."

6040 DATA "Estas en el fondo del mar. En el sur, una pared se alza enfrente de ti."

6050 DATA "Estas en el fondo del mar. En el sur hay una pared con algas marinas formando un cuadrado sobre ella."

6060 DATA "Estas en el fondo del mar. Hacia el sur hay una pared. Un precipicio bloquea el camino al este."

6070 DATA "Estas en una caverna larga y estrecha. Al fondo ves un tollo nadando en circulo."

6080 DATA "Estas en una habitacion llena deluz. Las paredes, el techo y el suelo brillan."

6090 DATA "Estas en una oscura caverna con una puerta enorme al fondo. No ves el picaporte."

6100 DATA "Estas en una habitacion llena dehambrientos caballitos de mar.Tehusmean la mano amistosamente."

6110 DATA "Estas en una pequena sala cuadrada. La pared del norte tiene una ventana, a traves de la que se ve el fondo del mar."

6120 DATA "Estas en una extrana sala cuadrada. Las paredes, el suelo,y el techo son cuadrados, asi como las puertas."

6130 DATA "Estas en una diminuta habitacionque esta casi acupada por un baul con las iniciales D.J."

6140 DATA "Estas en una habitacion muy fria. El fango te rodea y notas una pequena corriente hacia el este."

6150 DATA "Estas en un lugar realmente tetrico ! Esta todo lleno de loshuesos de exploradores anteriores."

6160 DATA "Estas en una sala cuadrada. Encima de la salida sur pone NO ENTRES. Las puertas del nortey del oeste estan llenas de hormigueantes erizos de mar."

6170 DATA "Estas en una habitacion circularcon una fuerte corriente que gira en remolino y se cuela por el agujero del fondo."

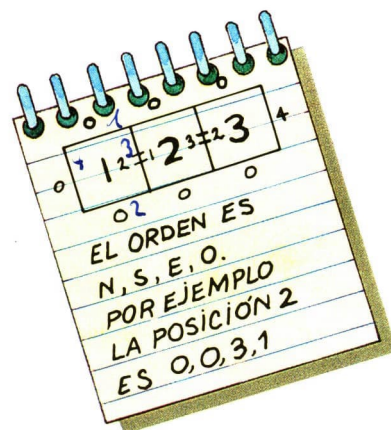
6180 DATA "Estas en un pasillo con una fuerte corriente hacia el oeste.El camino esta bloqueado por los brazos de un tremendo pulpo."

6190 DATA "Estas en el camarote del capitande un barco. Sientes el flujo del agua hacia el oeste."

6200 DATA "Ves un torbellino de agua, de donde sale un gran tiburón blanco."

Continuación de la relación de datos

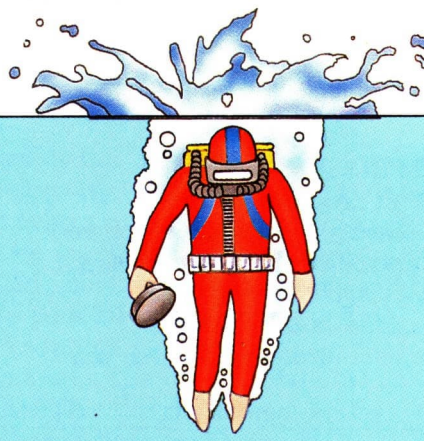
En las líneas 6210 a 6270 se almacenan las salidas disponibles inicialmente en cada posición, como una serie de cuatro números que corresponden al norte, sur, este y oeste. Si un número es cero, no hay salida en la dirección correspondiente. Por ejemplo, la posición 8 (la SALA LUMINOSA) tiene como salidas 0,14,0,7. Esto significa que no hay salida hacia el norte y al este, pero el sur conduce a la habitación 14 y el oeste a la 7. Las líneas siguientes contienen una lista con todos los objetos que el jugador puede coger. Tras cada objeto hay un número que representa la habitación en que se encuentra éste. Si el número es cero, el objeto aparecerá más tarde en el juego. Si es 99 significa que el jugador lleva el objeto.



La línea 6510 es una lista de los nombres que el ordenador reconoce (se trata de las direcciones y los nombres de los objetos). Sólo se almacenan las tres primeras letras, ya que es así como trabajará el programa. Por último, la línea 6530 tiene los verbos que reconocerá el ordenador. Se almacenan con todas las letras, pues tienen que aparecer en la pantalla cuando el jugador teclee AYUDA.

```
6210 REM *****SALIDAS*****
6220 DATA 0,0,2,0,0,0,3,1,0,0,4,2,0,0,5,3
6230 DATA 0,0,6,4,0,0,0,5,0,0,8,0,0,14,0,7
6240 DATA 0,15,0,0,0,16,0,0,5,17,12,0,0,18,0,11
6250 DATA 7,0,0,0,8,0,15,0,9,21,16,14,10,22,17,15
6260 DATA 11,23,18,16,12,24,0,17,0,0,20,0,0,0,21,0
6270 DATA 15,0,0,20,0,0,0,17,0,24,0,18,0,0,23
6300 REM *****OBJETOS*****
6310 DATA UN CUCHILLO,2
6320 DATA UN PAR DE ALETAS,6
6330 DATA UNA LLAVE,0
6340 DATA UN GRUPO DE ALGAS,0
6350 DATA UN HUESO MOHOSO,15
6360 DATA EL TAPON MAGICO,99
6370 DATA UNAS ESPOSAS AMARILLAS,0
6380 DATA UNAS ESPOSAS VERDES,9
6390 DATA UNAS ESPOSAS ROJAS,17
6400 DATA UNAS ESPOSAS AZULES,11
```

```
6500 REM *****NOMBRES*****
6510 DATA NOR,SUR,EST,OES,AUX,BAU,VEN,CUC,ALE,
        LLA,ALG,HUE,TAP,ESP
6520 REM *****VERBOS*****
6530 DATA VE,COJE,DEJA,CORTA,PONTE,DALE,ABRE,
        USA,INVENTARIO,AYUDA,MIRA,subir
```

EL PROGRAMA PRINCIPAL

Vas a penetrar en el reino submarino de las cavernas de Neptuno. En esta sección del programa, la pantalla de presentación indica al jugador cuál es el objeto del juego. Se inicializan también las variables que se van a usar en el juego. Estas operaciones son llamadas por el programa de control, ique sólo ocupa siete líneas!

LAS CAVERNAS DE NEPTUNO

Has encontrado el tapon magico que corresponde al desagüe del mar, y decides colocarlo antes de que el agua se salga.

Te sumerges en el oceano y comienza la aventura.

Si necesitas ayuda teclea AYUDA.

Estas frente a una puerta de madero que esta en la pared.No ves el picaporte.

SALIDAS : ESTE ; OESTE

Que hago ahora ?

La primera parte del listado es el PROGRAMA DE CONTROL, en las líneas 10 a 70. El programa está organizado en subrutinas, que se llaman en el orden necesario; así puedes ver fácilmente cómo funciona el programa. Cada subrutina se llama con la orden **GOSUB** seguida de un número de línea de programa. Cuando la subrutina ha terminado su trabajo, se devuelve el control al programa por medio de un **RETURN**.

```

10 CLS
20 GO SUB 5000: REM inicializacion
30 GO SUB 1000: REM entrada
40 GO SUB 2000: REM seleccion
50 IF fg<>1 AND fp<>1 THEN GO TO 30
60 GO SUB 5500: REM fin
70 STOP

```

La línea 20 llama a la rutina de **inicialización**, donde se preparan todas las variables y matrices necesarias para el juego. Las dos subrutinas siguientes, **entrada** y **selección**, hacen la mayor parte del trabajo del programa. La línea 50 comprueba si el juego se ha terminado, examinando el *flag* (indicador) de ganar (**fg**) y el *flag* de perder (**fp**). Si ninguno de ellos está activado, la instrucción **GOTO 30** hace saltar al programa a la línea 30, para repetir el proceso. En la línea 60 el juego se ha terminado, y la subrutina **fin** imprimirá el mensaje correspondiente en la pantalla, y te preguntará si quieres jugar otra vez. Si el jugador quiere probar de nuevo, el programa se ejecutará otra vez; en otro caso el **STOP** de la línea 70 parará el programa. Gracias al uso de subrutinas, el programa está básicamente escrito en siete líneas. Se ha indicado al ordenador qué es lo que tiene que hacer y en qué orden debe hacerlo. Todo lo que resta por hacer es escribir cada subrutina, para que el ordenador pueda saltar a la línea correspondiente cuando se llame a esta subrutina. La primera es **inicialización**. Las tres primeras líneas borran la pantalla con un **CLS** (línea 5010) e imprimen una introducción.

```

5000 REM inicializacion
5010 CLS : PRINT AT 0,4;"LAS CAVERNAS DE NEPTUNO"
5020 PRINT "Has encontrado el tapon magico""que corresponde al desague del""mar, y decides colocarlo antes"
    "de que el agua se salga.""Te sumerges en el oceano y""comienza la aventura."
5030 PRINT "ASEGURATE DE QUE TIENES PUESTAS LAS MAYUSCULAS""Si necesitas ayuda teclea AYUDA"

```

Es una buena idea inicializar las variables después de imprimir las instrucciones en la pantalla (líneas 5040 a 5120). Así el jugador puede leer las instrucciones mientras el ordenador está ocupado con los **DATA** (en juegos largos y complejos esto lleva cierto tiempo).

La línea 5040 da a las variables (**nn**, **vv**, **p**, **in**, **es**) sus valores iniciales, y pone a cero los *flags* (**fg**, **fp**, **fab**, **fa**, **fc**). La siguiente línea **DIM**ensiona cuatro matrices, reservando espacio para la descripción de las habitaciones, **l\$**; salidas, **s**; objetos, **o\$**; y posición de cada objeto, **o**. Por ejemplo, **DIM l\$(24,176)** reserva memoria para 24 descripciones con un máximo de 176 caracteres cada una —comprueba siempre que no sobrepasas este número—. Las líneas 5060 a 5110 leen (**READ**) los datos del **DATA** y los almacenan en las distintas matrices. Observa que la línea 5070 dice al ordenador que las posiciones 17,18,23 y 24 usan el mismo **DATA**. La línea 5120 ordena al ordenador que vaya (**GOTO**) a la línea 4000, pues **inicialización** tiene su fin en la sección **MIRA** de la subrutina **selecciona**.

```

5040 LET nn=14: LET vv=11: LET p=3: LET in=1: LET es=0:
    LET fg=0: LET fp=0: LET fab=0: LET fa=0: LET fc=0
5050 DIM l$(24,176): DIM s(24,4): DIM o$(10,24): DIM o(10)
5060 FOR i=1 TO 24
5070 IF i=17 OR i=18 OR i=23 OR i=24 THEN LET l$(i)=l$
    (12): GO TO 5090
5080 READ d$: LET l$(i)=d$
5090 NEXT i
5100 FOR i=1 TO 24: FOR j=1 TO 4: READ s(i,j): NEXT j:
    NEXT i
5110 FOR i=1 TO 10: READ o$(i): READ o(i): NEXT i
5120 GO TO 4000: REM mira

```

La siguiente subrutina es **entrada**. Acepta una frase de tres palabras de la forma verbo-artículo-nombre, una de dos de la forma verbo-nombre o una de una sola instrucción. Si no se reconoce ninguna de estas estructuras, el ordenador imprime un mensaje y vuelve al principio para leer otra frase. Las frases de una sola palabra que acepta el ordenador son "AYUDA", "INVENTARIO", "MIRA", y las cuatro direcciones.

VARIABLES

nn = NÚMERO DE NOMBRES

vv = NÚMERO DE VERBOS

p = POSICIÓN

in = NÚMERO DE ELEMENTOS DEL INVENTARIO

es = NÚMERO DE ESPOSAS

o\$(i) = OBJETO

n\$ = NOMBRE

o(i) = NÚMERO DE OBJETO

n = NÚMERO DE NOMBRE

l\$(i) = DESCRIPCIÓN DE POSICIÓN

v\$ = VERBO

s(p,i) = SALIDA DE UNA POSICIÓN

v = NÚMERO DE VERBO

FLAGS

fg = FLAG DE GANAR

fab = FLAG DE ABIERTO

fc = FLAG DE CORTADO

fp = FLAG DE PERDER

fa = FLAG DE ALETAS


```

1000 REM entrada
1010 LET v$="": LET n$=""
1020 INPUT "Que hago ahora ? ";r$
1030 IF LEN (r$)<3 THEN GO TO 1020
1040 FOR i=1 TO LEN (r$)
1050 IF r$(i TO i)=" " THEN LET v$=r$( TO 3): LET n$=r
$(i+1 TO ): GO SUB 1500: LET i=LEN (r$)
1060 NEXT i
1070 IF n$<>" " THEN RETURN
1080 LET r$=r$( TO 3)
1090 IF r$="NOR" OR r$="SUR" OR r$="EST" OR r$="OES" TH
EN LET v$="VE ": LET n$=r$: RETURN
1100 IF r$="AYU" OR r$="INV" OR r$="MIR" THEN LET n$="
AUX": LET v$=r$: RETURN
1110 PRINT "NO TE ENTIENDO !"
1120 GO TO 1010
1500 FOR j=1 TO LEN (n$)
1510 IF n$(j TO j)=" " THEN LET n$=n$(j+1 TO j+3): LET
j=j+2
1520 NEXT j
1530 IF LEN (n$)>3 THEN LET n$=n$( TO 3)
1540 RETURN

```

El objetivo es evitar que el jugador tenga que teclear "VE AL NORTE", etc. Es suficiente teclear "NORTE", o incluso "NOR". La subrutina pone "VE" en v\$ (verbo) y "NORTE" en n\$ (nombre) para uso posterior. "AYUDA", "INVENTARIO" y "MIRA" se tratan como si fueran verbos con un nombre auxiliar "AUX" que se inserta aunque no signifique nada. La línea 1010 hace que n\$ y v\$ estén vacíos antes de empezar, y la línea 1020 introduce la frase en r\$. Las líneas 1040 a 1060 contienen el bucle que divide r\$ en v\$ y n\$.

SI TECLEAS:

DEJA ALGA



DEJ

VERBO

ALG

NOMBRE

OESTE



VE

VERBO

OES

NOMBRE

AYUDA



AYU

VERBO

AUX

NOMBRE AUXILIAR

EL ORDENADOR REDUCIRÁ LOS NOMBRES Y LOS VERBOS A SUS TRES PRIMERAS LETRAS Y TRABAJARÁ CON ELLOS COMO CORRESPONDA.

El bucle empieza en la línea 1040, y su límite se fija al valor de la longitud (**LEN**) de la frase. La línea 1050 comprueba si (**IF**) el siguiente carácter de la frase es un espacio en blanco; si es así, entonces (**THEN**) se meten en **v\$** los tres primeros caracteres, y en **n\$** el resto de la frase. Tras haberse detectado un espacio en blanco, hay dos posibilidades: que después venga un nombre o que después venga un artículo y un nombre. En este último caso es necesario eliminar el artículo, y para ello se llama a la subrutina que empieza en la línea 1500, cuya estructura es prácticamente igual a ésta. En ambas subrutinas se pone en el contador del bucle un valor máximo, para hacer que se pare y no siga contando. Si **n\$** contiene algo, en la línea 1070 termina la subrutina, ya que se ha reconocido una frase de dos o tres palabras —recuerda que los artículos se eliminan, luego ahora sólo tenemos un verbo y un nombre. Si el programa llega a la línea 1080, quiere decir que en la frase no hay ningún espacio en blanco,

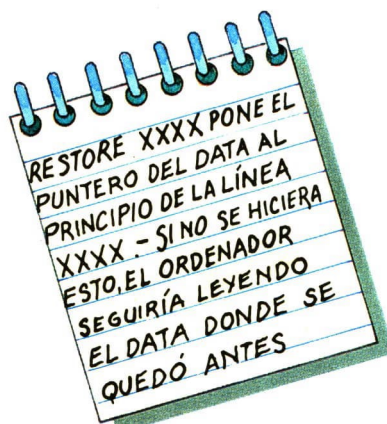
```

2000 REM seleccion
2010 LET v=0: LET n=0
2020 RESTORE 6530
2030 FOR i=1 TO vv
2040 READ d$: IF d$( TO 3)=v$ THEN LET v=i: LET i=vv
2050 NEXT i
2060 RESTORE 6510
2070 FOR i=1 TO nn
2080 READ d$: IF d$( TO 3)=n$ THEN LET n=i: LET i=nn
2090 NEXT i
2100 IF n=0 OR v=0 THEN PRINT "NO TE ENTIENDO !":
RETURN
2110 GO TO 2900+(100*v)

```

pero todavía puede ser una frase aceptable con una sola palabra. Entonces se reduce a sus tres primeros caracteres. Las líneas 1090 y 1100 comprueban si es una dirección o una orden. Si la palabra es válida, se ponen los valores adecuados en **v\$** y **n\$** y la subrutina termina. Si se llega a la línea 1110, la frase no tiene sentido y se imprime el mensaje. La línea 1120 hace volver al principio para leer otra frase.

La subrutina **selección** decide si las palabras de tres letras son válidas. La línea 2010 pone a cero el número de nombre y el de verbo. Con la instrucción **RESTORE** se coloca el puntero del **DATA** al principio de la línea 6530, y los datos de esta línea se leen (**READ**) en **d\$** usando un bucle. A continuación se comparan los tres primeros caracteres de **d\$** con **v\$** y, si coinciden, el verbo se reconoce como válido, y el número de verbo **v** toma el valor actual de la variable **i** y se sale del bucle al poner en **i** su valor máximo. Las líneas 2070 a 2090 comprueban si el nombre es válido de una manera similar, con la diferencia de que los nombres estaban ya almacenados con tres letras, luego pueden ser comparados directamente.




```

5500 REM fin
5510 IF p=22 THEN PRINT "DE UNA DENTELLADA, EL TIBURON
""TE ARRANCA LA CABEZA.": GO TO 5550
5520 IF p=16 THEN PRINT "LOS PINCHOS DE LOS ERIZOS SON
""MUY VENENOSOS !""SUFRES UNA MUERTE TERRIBLE. ""LA P
ROXIMA VEZ TOMA""MAS PRECAUCIONES": GO TO 5550
5530 PRINT "EL TAPON SE METER EN EL AGUJERO""Y EL AGUA
DEJA DE SALIR. ""FELICITACIONES!""HAS SALVADO LOS OCEA
NOS"
5550 INPUT "QUIERES JUGAR OTRA VEZ ? ":r$
5560 IF r$(1)="S" THEN RUN
5570 PRINT "ADIOS": RETURN

```

La línea 2100 comprueba si **(IF) n** o **(OR) v** son aún cero. Si es así, se imprime el mensaje y la subrutina termina aquí. Si el nombre y el verbo pasan todas estas pruebas, son palabras válidas y se elige el número de línea donde hay que saltar (**GOTO**). Por ejemplo, si el valor de **v** es cuatro (el verbo es "CORTA") el ordenador saltará (**GOTO**) a la línea 3300, $2900 + (100 \cdot 4) = 3300$.

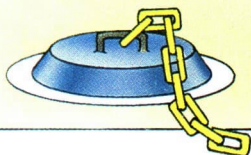
La última sección que queda por definir es la subrutina **fin**. Si el **flag** de ganar o el de perder están activados, de la línea 50 se pasa a la 60, donde se hace un **GOSUB 5500**. La línea 5510 comprueba si el jugador está en la habitación del TIBURON (posición 22). Si es así, se imprime un mensaje de muerte y pasa a la línea 5550 para otra partida.

Si la posición no es la 22, la línea 5520 comprueba si **(IF) p** es igual a 16 (la habitación de los ERIZOS), e imprime un mensaje. Si ninguna de estas condiciones se cumple el jugador ha ganado el juego. Se imprime (**PRINT**) un mensaje y las líneas 5550 y 5560 te preguntan si quieres jugar otra vez.

GANAS SI PONES EL TAPÓN
EN EL AGUJERO DE LA
SALA 19

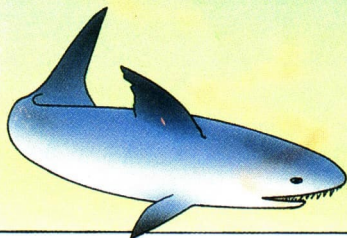
EL TAPÓN SE METER EN EL
AGUJERO Y EL AGUA
DEJA DE SALIR

ENHORABUENA
HAS SALVADO
LOS OCÉANOS



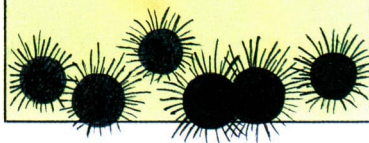
PIERDES SI ENTRAS EN
LA HABITACIÓN 22

DE UNA DENTELLADA,
EL TIBURÓN TE ARRANCA
LA CABEZA



PIERDES SI SALES DE LA
HABITACIÓN 16 HACIA EL NORTE
O EL OESTE SIN LLEVAR
LAS ALETAS

¡LOS PINCHOS DE LOS ERIZOS
SON MUY VENENOSOS!
SUFRES UNA MUERTE
TERRIBLE. LA PRÓXIMA
VEZ TOMA MÁS
PRECAUCIONES



La primera parte del listado es el PROGRAMA DE CONTROL, en las líneas 10 a 70. El programa está agrupado en subrutinas que se llaman en el orden necesario —permitiéndote ver cómo trabaja el programa. Cada subrutina se llama con una instrucción **GOSUB** seguida por un número de línea de programa. Cuando termina la subrutina, se devuelve el control al programa por medio de un **RETURN**.

EXPURG

```

10 REM LAS CAVERNAS DE NEPTUNO
20 GOSUB 5000: REM INICIALIZACION
30 GOSUB 1000: REM ENTRADA
40 GOSUB 2000: REM SELECCION
50 IF FG<>1 AND FP<>1 THEN GOTO 30
60 GOSUB 5500: REM FIN
70 STOP

```

La línea 20 llama a la subrutina **INICIALIZACION**, donde se preparan todas las variables y matrices necesarias para el juego. Las dos subrutinas siguientes, **ENTRADA** y **SELECCION**, hacen la mayor parte del trabajo del programa. La línea 50 comprueba si el juego se ha terminado, examinando el *flag* (indicador) de ganar **FG** y el de perder **FP**. Si ninguno de ellos está activado, la instrucción **GOTO 30** hace saltar el programa a la línea 30, para repetir el proceso. En la línea 60 el programa se ha terminado, y la subrutina **FIN** imprimirá el mensaje correspondiente en la pantalla y te preguntará si quieres jugar otra partida. Si el jugador quiere probar de nuevo el programa se ejecutará otra vez; en caso contrario, el **STOP** de la línea 70 lo parará. La primera subrutina es **INICIALIZACION**. La línea 5010 limpia la pantalla utilizando el símbolo que representa un corazón en inverso, y se imprimen las instrucciones del juego.

```

5000 REM *****INICIALIZACION*****
5010 PRINT " "
5020 PRINTSPC(12)"LAS CAVERNAS DE NEPTUNO"
5030 PRINT "HAS ENCONTRADO EL TAPON MAGICO QUE"
5040 PRINT "CORRESPONDE AL FONDO DEL MAR, Y"
5050 PRINT "DECIDES COLOCARLO ANTES DE QUE EL"
5060 PRINT "AGUA SE SALGA. TE SUMERGES EN EL"
5070 PRINT "OCEANO Y COMIENZA LA AVENTURA."
5080 PRINT "SI NECESITAS AYUDA TECLEA 'AYUDA'"

```

Es una buena idea inicializar las variables después de imprimir las instrucciones en la pantalla (líneas 5100 a 5210). Esto permite al jugador leer las instrucciones mientras el ordenador está ocupado con los **DATA**.

La línea 5100 da a las variables (NN, VV, P, IN, ES) sus valores iniciales, y pone los *flags* a cero (FG, FP, FAB, FA, FC). Las siguientes líneas **DIM**ensionan siete matrices, reservando espacio para las descripciones de las habitaciones, **T\$** y **D\$**; salidas **SA**; **NO\$** y **VB\$** (nombres y verbos); objetos, **OB\$**; y posición inicial de cada uno de ellos, **OB**. Por ejemplo, **DIM OB\$(10)** reserva memoria para la descripción de 10 objetos. Las líneas 5130 a 5200 leen (**READ**) los datos del **DATA** y los almacenan en las distintas matrices. Observa que la línea 5140 dice al ordenador que las posiciones 17, 18, 23 y 24 usan el mismo **DATA**. La línea 5210 hace que el ordenador vaya a la línea 4000, pues **INICIALIZACION** tiene su fin en la sección **MIRA** de la rutina **SELECCIONA**.

```

5100 NN=14: VV=11: P=3: IN=1: ES=015: FG=0: FP=0:
    FAB=0: FA=0: FC=0
5110 DIM T$(24): DIM D$(24): DIM SA(24,4):
    DIM NO$(NN): DIM VB$(VV)
5120 DIM OB$(10): DIM OB(10)
5130 FOR I=1 TO 25
5140 IF I=17 OR I=18 OR I=23 OR I=24 THEN T$(I)=T$(12):
    D$(I)=D$(12): GOTO 5160
5150 READ T$,D$: T$(I)=T$: D$(I)=D$
5160 NEXT I
5170 FOR I=1 TO 24: FOR J=1 TO 4: READ SA(I,J): NEXT J,I
5180 FOR I=1 TO 10: READ OB$(I),OB(I): NEXT I
5190 FOR I=1 TO NN: READ NO$(I): NEXT I
5200 FOR I=1 TO VV: READ VB$(I): NEXT I
5210 GOSUB 4000: RETURN: REM MIRA

```

La siguiente subrutina es **ENTRADA**. Acepta frases de tres palabras de la forma verbo-artículo-nombre, de dos palabras de la forma verbo-nombre o incluso frases con una sola instrucción. Si no se reconoce ninguna de estas estructuras, el ordenador imprime un mensaje y vuelve al principio para leer

VARIABLES

nn = NÚMERO DE NOMBRES 8

vv = NÚMERO DE VERBOS 8

p = POSICIÓN 1

in = NÚMERO DE ELEMENTOS DEL INVENTARIO 0

es = NÚMERO DE ESPOSAS ✓

ob\$(i) = OBJETO

no\$(i) = NOMBRE

ob(i) = NÚMERO DE OBJETO

no(i) = NÚMERO DE NOMBRE

t\$(i), d\$(i) = DESCRIPCIÓN DE POSICIÓN

vb\$(i) = VERBO

sa(p,no) = SALIDA DE UNA POSICIÓN

vb(i) = NÚMERO DE VERBO

FLAGS

fg = FLAG DE GANAR

fab = FLAG DE ABIERTO

fc = FLAG DE CORTADO

fp = FLAG DE PERDER

fa = FLAG DE ALETAS ✓


```

1000 REM *****ENTRADA*****
1010 VB$="": NO$="": R$=""
1020 INPUT "QUE HAGO AHORA ? ";R$
1030 FOR I=1 TO LEN(R$)
1040 IF MID$(R$,I,1)=" " THEN VB$=LEFT$(R$,3):
    NO$=RIGHT$(R$,I+1): GOSUB 1500: I=LEN(R$)
1050 NEXT I
1060 IF NO$<>"" THEN RETURN
1070 R$=LEFT$(R$,3)
1080 IF R$="NOR" OR R$="SUR" OR R$="EST" OR
    R$="OES" THEN VB$="VE ": NO$=R$: RETURN
1090 IF R$="AYU" OR R$="INV" OR R$="MIR"
    THEN NO$="AUX": VB$=R$: RETURN
1100 PRINT "NO TE ENTIENDO !"
1110 GOTO 1010
1500 FOR J=1 TO LEN(NO$)
1510 IF MID$(NO$,J,1)=" " THEN NO$=MID$(NO$,J+1,3):
    J=20
1520 NEXT J
1530 IF LEN(NO$)>3 THEN NO$=LEFT$(NO$,3)
1540 RETURN
    
```

otra frase. Las frases de una sola palabra que acepta el ordenador son "AYUDA", "INVENTARIO", "MIRA" y las cuatro direcciones. El objetivo es evitar al jugador el tener que teclear "VE AL NORTE", etc. Es suficiente teclear "NORTE", o incluso "NOR". La subrutina pone "VE" en VB\$ (verbo) y "NORTE" en NO\$ (nombre) para uso posterior. "AYUDA", "INVENTARIO" y "MIRA" se tratan como si fueran verbos con un nombre auxiliar "AUX", que se inserta aunque no signifique nada. La línea 1010 se asegura de que las variables VB\$, NO\$ y R\$ estén vacías antes de empezar, y la línea 1020 introduce la frase en R\$. Las líneas 1030 a 1050 contienen el bucle que divide R\$ en VB\$ y NO\$. El resto de la subrutina se explica en la siguiente página.

SI TECLEAS:

DEJA ALGA



SE TRANSFORMA EN :

DEJ

VERBO

ALG

NOMBRE

OESTE



VE

VERBO

OES

NOMBRE

AYUDA



AYU

VERBO

AUX

NOMBRE AUXILIAR

EL ORDENADOR REDUCIRÁ LOS NOMBRES Y LOS VERBOS A SUS TRES PRIMERAS LETRAS Y TRABAJARÁ CON ELLOS COMO CORRESPONDA.

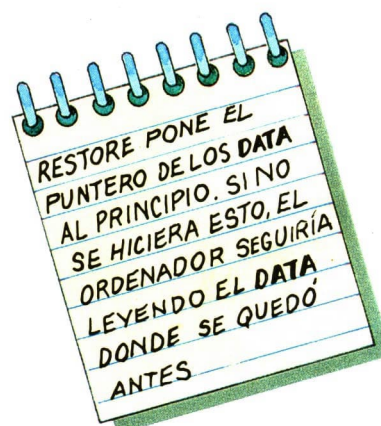
El bucle empieza en la línea 1030, y su límite se pone al valor de la longitud (**LEN**) de la frase. La línea 1040 comprueba si (**IF**) el siguiente carácter de la frase es un espacio en blanco —y si es así, entonces (**THEN**) se meten en **VB\$** los tres primeros caracteres y en **NO\$** el resto de la frase. Tras haberse detectado un espacio en blanco hay dos posibilidades: que después venga un nombre o que vengan un artículo y un nombre. En este último caso es necesario eliminar el artículo, y para ello se llama a la subrutina de la línea 1500, cuya estructura es prácticamente igual a ésta. En ambas subrutinas se pone en el contador del bucle un valor máximo, para hacer que se pare y no siga contando. Si **NO\$** contiene algo, en la línea 1060 termina la subrutina, ya que se ha reconocido una frase de dos o tres palabras —recuerda que los artículos se eliminan, luego ahora sólo tenemos un verbo y un nombre. Si el programa llega a la línea 1070, la frase no tiene espacios, pero todavía puede ser una frase aceptable de una sola palabra. Entonces se le reduce a sus tres primeros caracteres. Las líneas 1080 y 1090 comprueban si es una dirección o una orden. Si la palabra es válida, se ponen los valores correspondientes en **VB\$** y **NO\$** y la rutina termina. Si se alcanza la línea 1100, la frase no tiene sentido, y se imprime el mensaje correspondiente. La línea 1110 envía al principio para leer otra frase.

```

2000 REM *****SELECCION*****
2010 VB=0: NO=0
2020 FOR I=1 TO VV
2030 IF VB$=LEFT$(VB$(I),3) THEN VB=I: I=VV
2040 NEXT
2050 FOR I=1 TO NN
2060 IF NO$=NO$(I) THEN NO=I: I=NN
2070 NEXT
2080 IF NO=0 OR VB=0 THEN PRINT "NO TE
      ENTIENDO !": RETURN
2090 ON VB GOTO 3000,3100,3200,3300,3400,3500,3620,
      3700,3800,3900,4000

```

La rutina **SELECCION** decide si las palabras de tres letras son válidas (con una técnica llamada en inglés «parsing»). El bucle que está entre las líneas 2020 y 2040 intenta identificar el verbo. Si **VB\$** es igual a las tres primeras letras de alguno de los verbos almacenados en la matriz **VB\$**, la línea 2030 pone en **VB** el valor de la variable del bucle. Se sale del bucle poniendo un valor máximo en esta variable. De la misma forma, las líneas 2050 a 2070 comprueban si es un nombre válido, con la diferencia de que los nombres almacenados tienen ya tres letras, y se pueden comparar directamente. La línea 2080 comprueba si **NO** o **VB** son todavía cero. Si es así, se imprime un mensaje y termina la subrutina. Si el verbo y el nombre pasan todas estas pruebas son válidos, y se salta al




```

5500 REM *****FIN*****
5510 IF P=22 THEN PRINT "DE UNA DENTELLADA, EL TIBURON
TE ARRANCA LA CABEZA": GOTO 5570
5520 IF P=16 THEN PRINT "LOS PINCHOS DE LOS ERIZOS SON
MUY VENENOSOS. SUFRES UNA MUERTE TERRIBLE.":GOTO 5570
5540 PRINT "EL TAPON SE METE EN EL AGUJERO"
5550 PRINT "Y EL AGUA DEJA DE SALIR. HAS SALVADO"
5560 PRINT "LOS OCEANOS. FELICITACIONES !!"
5570 INPUT "QUIERES JUGAR OTRA VEZ ? ";R$
5580 IF LEFT$(R$,1)="S" THEN RUN
5590 PRINT "ADIOS": RETURN

```

lugar correspondiente en la línea 2090. Por ejemplo, si el valor de **VB** es cuatro (verbo "CORTA") el ordador saltará al cuarto número de línea de la lista (3300). Estas secciones de la rutina **SELECCION** se explican en el siguiente capítulo.

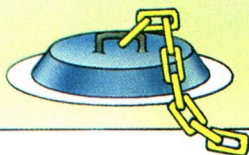
La última sección que queda por definir es la subrutina **FIN**. Si el *flag* de ganar o de perder está activado, de la línea 50 se pasa a la 60, donde se hace un **GOSUB 5500**. La línea 5510 comprueba si el jugador está en la habitación del TIBURON (posición 22). Si es así, se imprime un mensaje de muerte y el control pasa a la línea 5570, donde se pregunta si se quiere jugar otra partida.

Si la posición no es la 22, las líneas 5520 y 5530 comprueban si **P** es igual a 16 (la habitación de los ERIZOS), e imprimen un mensaje. Si ninguna de estas condiciones se cumple, el jugador ha ganado. Se imprime un mensaje en la línea 5540 y las líneas 5570 y 5580 llevan a cabo la rutina de jugar otra partida.

GANAS SI PONES EL TAPÓN
EN EL AGUJERO DE LA
SALA 19

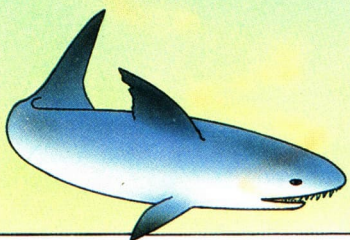
EL TAPÓN SE METE EN EL
AGUJERO Y EL AGUA
DEJA DE SALIR

ENHORABUENA
HAS SALVADO
LOS OCEANOS



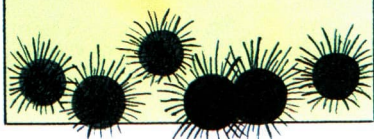
PIERDES SI ENTRAS EN
LA HABITACIÓN 22

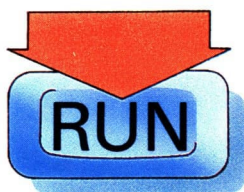
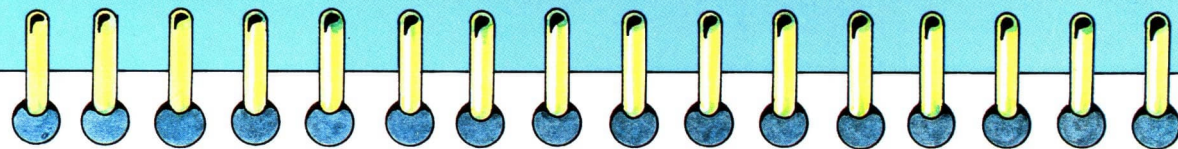
DE UNA DENTELLADA,
EL TIBURÓN TE ARRANCA
LA CABEZA



PIERDES SI SALES DE LA
HABITACIÓN 16 HACIA EL NORTE
O EL OESTE SIN LLEVAR
LAS ALETAS

¡LOS PINCHOS DE LOS ERIZOS
SON MUY VENENOSOS!
SUFRES UNA MUERTE
TERRIBLE. LA PRÓXIMA
VEZ TOMA MÁS
PRECAUCIONES





Comprobación del programa

Antes de que empieces con la siguiente sección del programa, es aconsejable que almacenes tu trabajo en una cinta. Encontrarás las instrucciones para hacerlo en tu manual de usuario. Para ver si el programa es correcto ve a la página 31 y teclea la sección "MIRA" que empieza en la línea 4000. Luego sigue las instrucciones que se indican a continuación. Podrás explorar las seis primeras habitaciones del juego.

COMMODORE: 1. TECLEA `2075 IF NO<=4 THEN P=SA (P,NO):GOTO 4000` PULSA `[RETURN]`. 2. TECLEA `2076 RETURN` PULSA `[RETURN]`. 3. PULSA `[RUN]`. CUANDO VEAS QUE TODO FUNCIONA COMO DEBE, BORRA LAS LÍNEAS 2075 Y 2076

SALIDAS: ESTE OESTE
ESTAS EN EL FONDO DEL MAR.
EN EL SUR HAY UNA PARED CON
ALGAS MARINAS FORMANDO UN
CUADRADO SOBRE ELLA.
SALIDAS: ESTE OESTE
ESTAS EN EL FONDO DEL MAR.
HACIA EL SUR HAY UNA PARED.
UN PRECIPICIO BLOQUEA EL CAMINO
AL ESTE.

ASEGURATE DE QUE TIENES PUESTAS
LAS MAYUSCULAS.
SI NECESITAS AYUDA TECLEA AYUDA.
ESTAS ENFRENTA DE UNA PUERTA
DE MADERA QUE ESTA EN LA PARED.
NO VES EL PICAFORTE.
SALIDAS: ESTE OESTE
ESTAS EN EL FONDO DEL MAR.
EN EL SUR, UNA PARED SE ALZA
ENFRENTA DE TI.

SPECTRUM: 1. TECLEA `2095 IF n=4 THEN LET p=s(p,n):GOTO 4000` PULSA `[ENTER]`. 2. TECLEA `2096 RETURN` PULSA `[ENTER]`. 3. PULSA `[RUN]`. CUANDO VEAS QUE TODO FUNCIONA COMO DEBE, BORRA LAS LÍNEAS QUE INTRODUIJISTE ANTES

SALIDAS: ESTE OESTE
ESTAS EN EL FONDO DEL MAR.
EN EL SUR HAY UNA PARED CON
ALGAS MARINAS FORMANDO UN
CUADRADO SOBRE ELLA.
SALIDAS: ESTE OESTE
ESTAS EN EL FONDO DEL MAR.
HACIA EL SUR HAY UNA PARED.
UN PRECIPICIO BLOQUEA EL CAMINO
AL ESTE.

ASEGURATE DE QUE TIENES PUESTAS
LAS MAYUSCULAS.
SI NECESITAS AYUDA TECLEA AYUDA.
ESTAS ENFRENTA DE UNA PUERTA
DE MADERA QUE ESTA EN LA PARED.
NO VES EL PICAFORTE.
SALIDAS: ESTE OESTE
ESTAS EN EL FONDO DEL MAR.
EN EL SUR, UNA PARED SE ALZA
ENFRENTA DE TI.



LAS RUTINAS DE LOS VERBOS

Esta sección del programa es parte de la subrutina que empieza en la línea 2000. Contiene cada uno de los verbos que el jugador puede usar para indicar sus acciones durante el juego, y hace que el ordenador dé la respuesta adecuada. Dado que el ordenador ordena automáticamente cada línea según su número, esta sección quedará tras la subrutina correspondiente en el listado del programa.

SALIDAS : ESTE OESTE

Que hago ahora? VE AL OESTE

Estas en el fondo del mar. En el sur, una pared se alza frente a ti.

SALIDAS : ESTE OESTE

Que hago ahora? VE AL ESTE

Estas en el fondo del mar. En el sur hay una pared con algas marinas formando un cuadrado sobre ella.

SALIDAS : ESTE OESTE

Que hago ahora? CORTAR LAS ALGAS

Las algas caen y aparece una ventana abierta


```

3000 REM *****ir*****
3010 IF n>4 THEN PRINT "IR A DONDE ?": RETURN
3020 IF s(p,n)=0 THEN PRINT "POR AHI NO HAY SALIDA !":
RETURN
3030 IF p=16 AND (n=1 OR n=4) AND fa=0 THEN LET fp=1:
RETURN
3040 LET p=s(p,n): GO TO 4000

```

La subrutina **selección** envía el programa a la línea 3000 si el verbo es "VE" (verbo 1). La línea 3010 imprime un mensaje si el nombre no es ninguna de las cuatro direcciones. La línea 3020 comprueba si se intenta hacer un movimiento en una dirección en la que no existe salida, comparando el número del nombre con el número correspondiente de la matriz de salidas **s(p,n)**. La línea 3030 comprueba si el jugador se ha pinchado con algún erizo y ha muerto. Para ello busca la posición actual (16=habitación de erizos), la dirección (norte y oeste son peligrosas) y el **flag** de aletas **fa** (que tendrá el valor 1 si el jugador las lleva puestas). Si estas condiciones se cumplen, el jugador ha muerto, luego se pone a uno **fp** y la rutina termina.

```

3100 REM *****coger*****
3110 IF n=6 AND p=13 THEN PRINT "ES DEMASIADO PESADO !
": RETURN
3120 IF n<8 THEN PRINT "NO SEAS TONTO !": RETURN
3130 IF o(n-7)=99 AND n$<>"ESP" THEN PRINT "YA LO HAS
COGIDO !": RETURN
3140 FOR i=7 TO 10
3150 IF o(i)=p AND n=14 THEN LET n=i+7: LET es=es+1: L
ET i=10
3160 NEXT i
3170 IF o(n-7)=p THEN PRINT "O.K.": LET o(n-7)=99: LET
in=in+1: RETURN
3180 PRINT "ESO NO ESTA AQUI !": RETURN

```

Si el número de verbo es 2, el control se pasa a la sección "COGER". Las líneas 3110 y 3120 cortan cualquier intento de hacer algo imposible. La línea 3130 comprueba si (**IF**) el objeto está ya cogido, es decir, tiene el valor 99... y (**AND**) si el nombre introducido no es "ESP". En las líneas 3140 a 3160 se comprueba si se coge un par de esposas, comparando la posición de cada par **o(i)** con la posición actual **p**. Si **o(i)=p** se incrementa el contador de esposas **es** y se pone en **i** el valor máximo para terminar el bucle. La línea 3170 compara la posición del objeto **o(n-7)** con la posición actual **p** e imprime "O.K." si son iguales. La posición del objeto se pone a 99 para indicar que lo lleva el jugador y el inventario **in** se incrementa en 1.

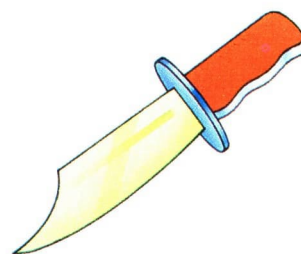

```

3200 REM *****dejar*****
3210 IF n<8 THEN PRINT "NO SEAS TONTO !": RETURN
3220 FOR i=7 TO 10
3230 IF o(i)=99 AND n=14 THEN LET n=i+7: LET es=es-1:
LET i=10
3240 NEXT i
3250 IF o(n-7)<>99 THEN PRINT "TU NO TIENES ESO !": RE
TURN
3260 PRINT "O.K.": LET o(n-7)=p: LET in=in-1
3270 IF p=19 AND n=13 THEN LET fg=1
3280 RETURN

```

En la sección "DEJAR", la línea 3210 evita que el jugador deje algo que en realidad no lleva. Si el elemento que se quiere dejar es un par de esposas, un bucle **FOR...NEXT** en las líneas 3220 a 3240 decrementa el valor de **es** en uno. Si la localización del objeto no es igual a 99, la línea 3250 imprime (**PRINT**) un mensaje para informar al jugador de que no está llevando ese objeto. Si el control llega hasta la línea 3260, se pone como localización del objeto la posición actual del jugador y se decrementa en uno el inventario. Por último, la línea 3270 comprueba si se ha ganado, es decir, si el jugador ha colocado el tapón mágico en el agujero.

Las líneas 3300 a 3340 de la sección "CORTAR" evitan que el jugador corte algo que no sean las algas de la habitación número cinco. Por cada intento no correcto, se imprime un mensaje y la subrutina termina. Sólo si se cumplen todas las condiciones aparecerá el mensaje de la línea 3350. A continuación se actualiza la información; en la habitación cinco se pone **UN GRUPO DE ALGAS**, aparece una nueva salida y el *flag* de cortar **fc** se activa a uno. Además, la descripción de posición **1\$(5)** también se modifica antes de que la subrutina termine.



```

3300 REM *****cortar*****
3310 IF o(1)<>99 THEN PRINT "NO TIENES NADA AFILADO !"
: RETURN
3320 IF n<>11 THEN PRINT "NO PUEDES CORTAR ESO !": RET
URN
3330 IF p<>5 THEN PRINT "NO PUEDES HACER ESO !": RETUR
N
3340 IF fc=1 THEN PRINT "YA LO HAS HECHO !": RETURN
3350 PRINT "LAS ALGAS CAEN Y APARECE" "UNA VENTANA ABIE
RTA"
3360 LET o(4)=5: LET s(5,2)=11: LET fc=1
3370 LET 1$(5)="Estas en el fondo del mar. Hacia el sur
hay una pared con una ventana abierta.": RETURN

```



```
3400 REM *****ponerse*****
```

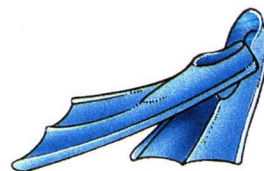
```
3410 IF fa=1 AND n$="ALE" THEN PRINT "YA LAS TIENES PU  
ESTAS !": RETURN
```

```
3420 IF o(2)=99 AND n$="ALE" THEN LET fa=1: LET o(2)=0  
: PRINT "TE QUEDAN MUY BIEN !": RETURN
```

```
3430 IF n=14 AND es>1 THEN PRINT "HACER ESO ES DE ESTU  
PIDOS !": RETURN
```

```
3440 PRINT "NO PUEDES PONERTE ESO !": RETURN
```

El verbo PONTE permite al jugador ponerse las ALETAS de la habitación sexta. Si estando el jugador en la habitación 16, va hacia el oeste o hacia el norte sin tener los pies protegidos, morirá instantáneamente debido a los pinchos venenosos de los erizos. La posibilidad de que las aletas ya estén puestas está prevista en la línea 3410. En la línea 3420 se permite al jugador que se ponga las aletas si las está llevando, es decir, si `o(2)=99` y (AND) si el nombre introducido es "ALE". El flag de aletas `fa` se activa a 1 y haciendo `o(2)=0` se eliminan las aletas de entre los elementos que se llevan. Las dos últimas líneas evitan que el jugador se ponga alguno de los otros objetos.



```
3500 REM *****dar*****
```

```
3510 IF p=10 AND n=11 THEN GO TO 3550
```

```
3520 IF p=7 AND n=12 THEN GO TO 3570
```

```
3530 IF p=20 THEN PRINT "ESO NO TE SERVIRA DE NADA !":  
RETURN
```

```
3540 PRINT "AQUI NADIE QUIERE ESO !": RETURN
```

```
3550 IF o(4)<>99 THEN PRINT "NO TIENES ESO !": RETURN
```

```
3560 LET o(4)=0: LET o(3)=10: PRINT "DAN VUELTAS ALREDE  
DOR TUYO" "Y VES ALGO EN LA ESQUINA": RETURN
```

```
3570 IF o(5)<>99 THEN PRINT "NO TIENES ESO !": RETURN
```

```
3580 LET o(5)=0: LET s(7,2)=13: PRINT "EL PEZ COGE EL H  
UESO" "Y SE RETIRA A UNA ESQUINA"
```

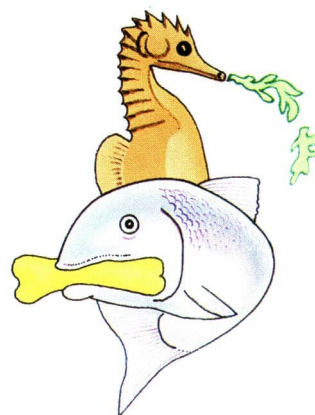
```
3590 LET l$(7)="Estas en una caverna.
```

En una

```
esquina hay un tollo con un hueso.": RETURN
```

La sección "DAR" permite que des algas a los caballitos de la habitación 10 y un hueso al pez de la 7. La línea 3510 examina la primera posibilidad y lleva a la línea 3550 si el jugador está en la habitación correcta (`p=10`) y si el nombre es "ALGAS". Una vez satisfechas todas las condiciones, la su rutina termina con un mensaje.

Si las condiciones de la línea 3510 no se cumplen, el programa sigue con la línea 3520, donde se hacen las pruebas necesarias para ver si el jugador está intentando dar un hueso al pez de la habitación número siete; si es así, se salta a la línea 3570. En cualquier otro caso se imprime un mensaje de error y la subrutina termina.



La línea 3570 comprueba si se tiene el hueso, y si no es así se imprime el mensaje correspondiente. Sin embargo, si **(IF)** **o(5)=99** el jugador cumple las dos condiciones y el programa sigue en la línea 3580. Se elimina el hueso, poniendo **o(5)=0**, y se abre una salida hacia el sur que conduce a la habitación 13, ya que se asigna el valor 13 a la variable **s(7,2)**.

En la línea 3580 se imprime **(PRINT)** un mensaje concerniente a la caverna y al pez con el hueso. Por último, se actualiza la descripción de la habitación para que esté lista cuando el jugador teclee "MIRA".

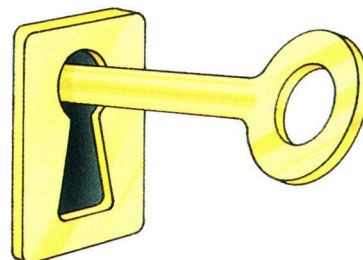
```

3600 REM *****abrir*****
3610 IF o(3)<>99 THEN PRINT "NO TIENES NINGUNA LLAVE !"
": RETURN
3620 IF n=5 AND (p=3 OR p=9) THEN PRINT "NO HAY NINGUN
A CERRADURA !": RETURN
3630 IF p<>13 OR n<>6 THEN PRINT "NO PUEDES HACER ESO
!": RETURN
3640 IF fab=1 THEN PRINT "YA ESTA ABIERTO !": RETURN
3650 LET fab=1: LET o(7)=13: PRINT "LA LLAVE LO ABRE FA
CILMENTE"
3660 LET l$(13)="Estas en una habitacion con un cofre
abierto en medio.": RETURN

```

Nada, excepto el cofre de John Silver, se puede abrir con una llave. La línea 3610 comprueba que, de hecho, el jugador lleva encima la llave. Es posible que el jugador intente abrir las puertas de las habitaciones tres y nueve. Sin embargo, estas puertas son sólo un engaño (¡Todos los buenos juegos de aventuras tienen trampas!) y nunca se podrán abrir. La línea 3620 controla estas situaciones e indica al jugador que no puede usar la llave.

El programa sólo llegará a la línea 3640 si el jugador está en la habitación del cofre y el nombre introducido es "COFRE". Si todo está correcto, se examina el *flag* de abierto, **fab**, para asegurarse de que el cofre no ha sido abierto anteriormente. La línea 3650 pone **fab** a 1, y se indica al ordenador que la llave se queda en esa habitación, es decir, se asigna a **o(7)** el valor 13. Por último, se imprime un mensaje para confirmar que la acción se ha realizado.



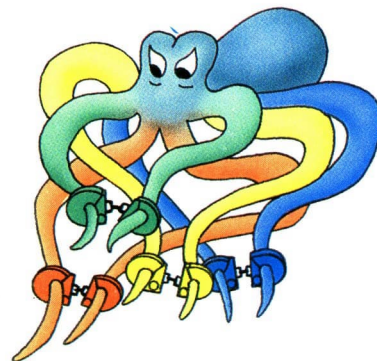

```

3700 REM *****usar*****
3710 IF p<>20 OR n<>14 THEN PRINT "NO PUEDES HACER ESO
      AQUÍ !": RETURN
3720 IF es<4 THEN PRINT "NO TIENES PARES SUFICIENTES !
      ": RETURN
3730 PRINT "EL PULPO NO SE PUEDE MOVER, Y ESO NO LE H
      ACE NINGUNA GRACIA !"
3740 FOR i=7 TO 10: LET o(i)=0: NEXT i
3750 LET es=0: LET s(20,4)=19: LET l$(20)="Estas en un
      pasillo con una corriente muy fuerte hacia el oes
      te. Hay sentado un pulpo atado y muy enfadado.": RE
      TURN

```

El verbo "USAR" tiene como misión el permitir al jugador inmovilizar al pulpo de la habitación 20 con cuatro pares de esposas, por ejemplo tecleando "USA LAS ESPOSAS".

La línea 3710 comprueba que la posición del jugador es correcta y que el número del nombre corresponde a las esposas ($n=14$). La línea 3720 hace que el jugador no pueda hacer nada en cuanto le falte una de las cuatro esposas que están escondidas en las distintas habitaciones del mapa. Si el programa llega a la línea 3730 aparece un mensaje para avisarnos de que el pulpo está inmovilizado. El jugador se libera de las esposas usando un bucle **FOR...NEXT** que pone la posición del objeto a cero en la línea 3740. Por último, el número de esposas se pone a cero, aparece una nueva salida disponible hacia el oeste y se modifica la descripción de la habitación **l\$(20)**.



Las últimas tres partes del programa corresponden a las órdenes "INVENTARIO", "AYUDA" y "MIRA", cuya utilidad es tal que sin ellas el juego sería extremadamente difícil.

```

3800 REM *****inventario*****
3810 PRINT "ESTAS LLEVANDO:—"
3820 IF in=0 THEN PRINT "NADA !": RETURN
3830 FOR i=1 TO 10
3840 IF o(i)=99 THEN PRINT o$(i)
3850 NEXT i: RETURN

```

La rutina INVENTARIO informa al jugador sobre los objetos que lleva. Si **in** contiene un cero, se imprimirán las palabras "ESTAS LLEVANDO:— NADA!", debido a las líneas 3810 y 3820.

Sin embargo, si **in** no es igual a cero, en la línea 3830 comienza un bucle que comprueba para cada objeto si el jugador lo está llevando o no, y lista los objetos que lleva por pantalla.


```

3900 REM *****ayuda*****
3910 PRINT "Estos son los verbos que puedes usar:"
3920 RESTORE 6530
3930 FOR i=1 TO vv: READ d$: PRINT d$,;: NEXT i
3940 PRINT "(Solo necesitas teclear las tres primeras l
      etras)": RETURN

```

En la orden "AYUDA", las cuatro líneas del programa desde la 3910 a la 3940 imprimen cada uno de los verbos disponibles para formar frases aceptables. Para ello se usa un bucle **FOR...NEXT** con la **i** variando hasta el máximo número de verbos (en este caso, el valor de **vv**).

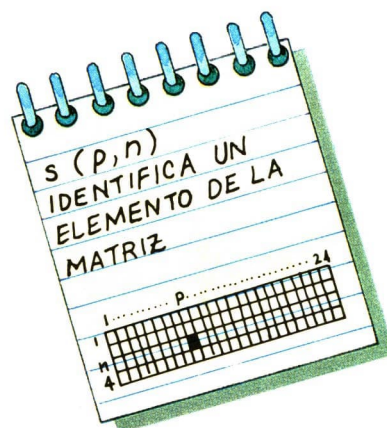
La última sección del programa se refiere al verbo "MIRA". Si has seguido las instrucciones de la página 24 (en el apartado **RUN**), ya habrás tecleado esta parte del programa.

Esta sección es un poco peculiar, pues sirve para el fin de la subrutina **inicialización**, unidas por una orden **GOTO** en la línea 5140.

La línea 4010 imprime (**PRINT**) la descripción de la habitación **l\$(p)**. El **flag** de perder, **fp**, se activa si el jugador ha ignorado la advertencia sobre la salida sur de la habitación 16 y ha entrado en la habitación del tiburón (posición 22).

Las líneas 4020 a 4050 imprimen los objetos que se encuentran en la habitación ocupada por el jugador. Observa que no se hace mención al **HUESO MOHOSO** en la línea 4030, para hacer el juego más difícil.

Por último, en las líneas 4060 a 4110 se indican las salidas disponibles para el jugador desde la habitación donde se encuentra.



```

4000 REM *****mira*****
4010 PRINT "l$(p): IF p=22 THEN LET fp=1: RETURN
4020 FOR i=1 TO 10
4030 IF i=5 AND p=15 THEN GO TO 4050
4040 IF o(i)=p THEN PRINT "AQUI HAY ";o$(i)
4050 NEXT i
4060 PRINT "SALIDAS:";
4070 IF s(p,1)>0 THEN PRINT "NORTE ";
4090 IF s(p,2)>0 THEN PRINT "SUR ";
4100 IF s(p,3)>0 THEN PRINT "ESTE ";
4110 IF s(p,4)>0 THEN PRINT "OESTE ";
4510 PRINT : RETURN

```



```

3000 REM *****IR*****
3010 IF NO>4 THEN PRINT "IR A DONDE ?": RETURN
3020 IF SA(P,NO)=0 THEN PRINT "POR AHI NO HAY
    SALIDA": RETURN
3030 IF P=16 AND (NO=1 OR NO=4) AND FA=0 THEN FP=1:
    RETURN
3040 P=SA(P,NO): GOSUB 4000
3050 RETURN

```

La subrutina **SELECCION** envía al programa a la línea 3000 si el verbo es "VE" (verbo 1). La línea 3010 imprime un mensaje si el nombre no es una de las cuatro direcciones. La línea 3020 comprueba si se intenta hacer un movimiento en una dirección para la que no existe salida, es decir, si **SA(P,NO)** es igual a cero. La línea 3030 comprueba si el jugador ha pisado algún erizo y ha muerto. Para ello mira la posición actual (16=habitación de los erizos), la dirección (norte y oeste son peligrosos) y el *flag* de aletas **FA** (que tendrá el valor uno si el jugador las lleva puestas). Si estas condiciones se cumplen, el jugador ha muerto, se pone **FP** a uno y la rutina termina.

```

3100 REM *****COGER*****
3110 IF NO=6 AND P=13 THEN PRINT "ES DEMASIADO
    PESADO !": RETURN
3120 IF NO<8 THEN PRINT "NO SEAS TONTO !":RETURN
3130 IF OB(NO-7)=99 AND NO<>"ESP" THEN PRINT "YA
    LO HAS COGIDO !": RETURN
3140 FOR I=7 TO 10
3150 IF (OB(I)=P AND NO=14) THEN NO=I+7: ES=ES+1:
    I=10: IN=IN+1
3160 NEXT I
3170 IF OB(NO-7)=P THEN PRINT "O.K.": OB(NO-7)=99:
    IN=IN+1: RETURN
3180 PRINT "ESO NO ESTA AQUI !": RETURN

```

Si el número del verbo es 2, el control llega a la rutina "COGER". Las líneas 3110 y 3120 cortan cualquier intento de hacer algo imposible. La línea 3130 comprueba si (**IF**) el objeto está ya cogido, es decir, tiene el valor 99... y (**AND**) si el nombre introducido no es "ALE". En las líneas 3140 a 3160 se comprueba si se coge un par de esposas, comparando la oposición de cada esposa **OB(I)** con la posición actual **P**. Si **OB(I)=P** se incrementa el contador de esposas **ES** y se pone en **I** el valor máximo para terminar el bucle. La línea 3170 compara la posición del objeto **OB(NO-7)** con la posición actual **P**, e imprime "O.K." si son iguales. La posición del objeto se pone a 99 para indicar que lo lleva el jugador y el inventario **IN** se incrementa en uno.

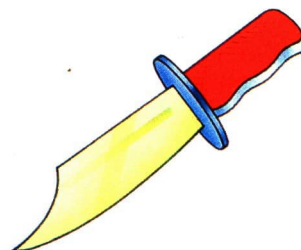

```

3200 REM *****DEJAR*****
3210 IF NO<8 THEN PRINT "NO SEAS TONTO !":RETURN
3220 FOR I=7 TO 10
3230 IF (OB(I)=99 AND NO=14) THEN NO=I+7:ES=ES-1:
    I=10: IN=IN-1
3240 NEXT I
3250 IF OB(NO-7)<>99 THEN PRINT "TU NO TIENES
    ESO": RETURN
3260 PRINT "O.K.": OB(NO-7)=P: IN=IN-1
3270 IF P=19 AND NO=13 THEN FG=1
3280 RETURN

```

En la sección "DEJAR", la línea 3210 evita que el jugador deje algo que en realidad no lleva. Si el elemento que se quiere dejar son un par de esposas, un bucle **FOR...NEXT** en las líneas 3220 a 3240 decrementa el valor de **ES** en uno. Si la localización del objeto no es igual a **99**, se imprime (**PRINT**) un mensaje para indicar al jugador que no está llevando ese objeto. Si el control llega hasta la línea 3260, se pone como localización del objeto la posición actual del jugador y se decrementa en uno el inventario. Por último, la línea 3270 comprueba si se ha ganado el juego, es decir, si el jugador ha dejado el tapón mágico en el agujero.

Las líneas 3300 a 3340 de la sección "CORTAR" evitan que el jugador corte algo que no sean las algas de la habitación número cinco. Por cada intento no correcto, se imprime un mensaje y la subrutina termina. Sólo si se cumplen todas las condiciones aparecerá el mensaje de la línea 3350. A continuación se actualiza la información; en la habitación cinco se pone **UN GRUPO DE ALGAS**, aparece una nueva salida y el **flag** de cortado **FC** se activa a uno. Además, la descripción de posición **D\$(5)** se modifica antes de que la subrutina termine.



```

3300 REM *****CORTAR*****
3310 IF OB(1) <> 99 THEN PRINT "NO TIENES NADA
    AFILADO !": RETURN
3320 IF NO<>11 THEN PRINT "NO PUEDES CORTAR ESO !":
    RETURN
3330 IF P<>5 THEN PRINT "NO PUEDES HACER ESO !":
    RETURN
3340 IF FC=1 THEN PRINT "YA LO HAS HECHO !":RETURN
3350 PRINT "LAS ALGAS CAEN Y APARECE UNA VENTANA
    ABIERTA"
3360 OB(4)=5: SA(5,2)=11: FC=1
3370 D$(5)=LEFT$(D$(5),22)+"UNA VENTANA ABIERTA":
    RETURN

```



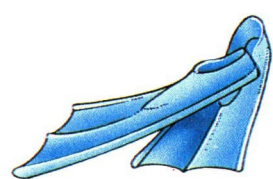
```

3400 REM *****PONERSE*****TRA
3410 IF FA=1 AND NO$="ALE" THEN PRINT "YA LAS TIENES  
PUESTAS !": RETURN
3420 IF OB(2)=99 AND NO$="ALE" THEN FA=1: OB(2)=0:  
PRINT "TE QUEDAN MUY BIEN !": RETURN
3430 IF NO=14 AND ES>0 THEN PRINT "HACER ESO ES DE  
ESTUPIDOS !": RETURN
3440 PRINT "NO PUEDES PONERTE ESO !": RETURN

```

YA TE LO VAS PUESTO

El verbo PONTE permite al jugador ponerse las ALETAS de la habitación número seis. Si estando el jugador en la habitación 16, va hacia el este o hacia el norte sin tener los pies protegidos, morirá instantáneamente debido a los pinchos venenosos de los erizos. La posibilidad de que las aletas ya estén puestas está prevista en la línea 3410. En la línea 3420 se permite al jugador que se ponga las aletas si las está llevando, es decir, si **OB(2)=99** y (**AND**) si el nombre introducido es "ALE". El *flag* de aletas **FA** se activa a **1** y se indica que las aletas ya no se están llevando al hacer **OB(2)=0**. Las dos últimas líneas evitan que el jugador se ponga alguno de los otros objetos.



```

3500 REM *****DAR*****NO VUEL
3510 IF P=10 AND NO=11 THEN 3550
3520 IF P=7 AND NO=12 THEN 3580
3530 IF P=20 THEN PRINT "ESO NO TE SERVIRA DE NADA":  
RETURN
3540 PRINT "AQUI NADIE QUIERE ESO !": RETURN
3550 IF OB(4) <> 99 THEN PRINT "NO TIENES ESO !":  
RETURN
3560 OB(4)=0: OB(3)=10
3570 PRINT "VES ALGO EN LA ESQUINA":RETURN
3580 IF OB(5)<>99 THEN PRINT "NO TIENES ESO !":RETURN
3590 OB(5)=0: SA(7,2)=13
3600 PRINT "EL PEZ COGE EL HUESO Y SE RETIRA A UNA  
ESQUINA"
3610 T$(7)="ESTAS EN UNA CAVERNA."
3615 D$(7)="EN UNA ESQUINA HAY UN TOLLO CON UN HUESO":  
RETURN

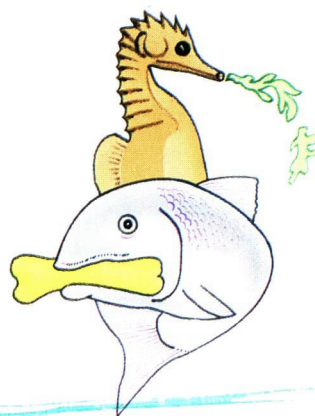
```

La sección "DAR" permite que des algas a los caballitos de la habitación diez y un hueso al pez de la siete. La línea 3510 examina la primera posibilidad y salta a la línea 3550 si **P=10** y el nombre es "ALGAS". Una vez satisfechas todas las condiciones, la subrutina termina con un mensaje.

Si las condiciones de la línea 3510 no se cumplen, el programa sigue con la línea 3520, donde se hacen las pruebas necesarias para ver si el jugador está intentando dar un hueso al pez de la habitación número siete —si es así, se salta a la línea 3580. En cualquier otro caso, se imprime un mensaje de error y la subrutina termina.

La línea 3580 comprueba si se tiene el hueso, y si no es así se imprime el mensaje correspondiente. Sin embargo, si (IF) **OB(5)=99**, el jugador cumple las dos condiciones y el programa sigue en la línea 3590. El hueso se quita de la habitación poniendo a cero **OB(5)** y se abre una salida hacia el sur al hacer **SA(7,2)** igual a 13.

En la línea 3590 se imprime (**PRINT**) un mensaje concerniente a la caverna y al pez con el hueso. Por último, se cambian **T\$** y **D\$**, para que estén listos cuando el jugador teclee "MIRA".



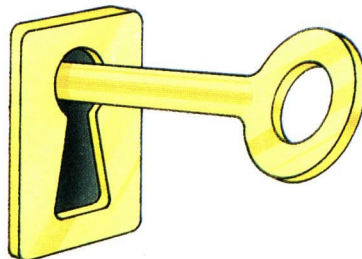
EXPURGO

```

3620 REM *****ABRIR*****
3630 IF OB(3)<>99 THEN PRINT "NO TIENES NINGUNA
      LLAVE !":RETURN
3640 IF NO=5 AND (P=3 OR P=9) THEN PRINT "NO HAY
      NINGUNA CERRADURA !": RETURN
3650 IF P<>13 OR NO<>6 THEN PRINT "NO PUEDES HACER
      ESO !: RETURN
3660 IF FAB=1 THEN PRINT "YA ESTA ABIERTO !": RETURN
3670 FAB=1: OB(7)=13: PRINT "LA LLAVE LO ABRE
      FACILMENTE"
3680 T$(13)="ESTAS EN UNA HABITACION CON UN COFRE
      ABIERTO EN MEDIO"
3690 D$(13)="": RETURN
  
```

Nada, excepto el cofre de John Silver, se puede abrir con una llave. La línea 3630 comprueba que, de hecho, el jugador lleva encima la llave. Es posible que el jugador intente abrir las puertas de las habitaciones tres y nueve. Sin embargo, estas puertas son sólo un engaño (¡todos los buenos juegos de aventuras tienen trampas!) y nunca se podrán abrir. La línea 3640 controla estas situaciones e indica al jugador que no puede usar la llave.

El programa sólo llegará a la línea 3650 si el jugador está en la habitación del cofre y el nombre introducido es "COFRE". Si todo está correcto, se examina el *flag* de abierto, **FAB**, para asegurarse de que el cofre no ha sido abierto anteriormente. La línea 3670 pone **FAB** a 1, y se indica al ordenador que la llave se queda en esta habitación, es decir, se asigna a **OB(7)** el valor 13. Por último, se imprime un mensaje para confirmar que la acción se ha realizado.

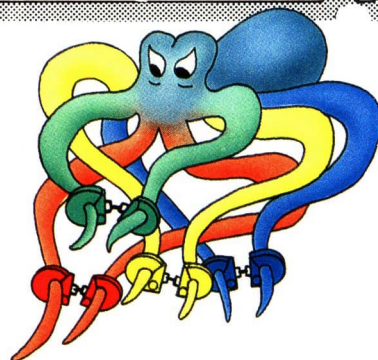



```

3700 REM *****USAR*****
3710 IF P<>20 OR NO<>14 THEN PRINT "NO PUEDES
      HACER ESO AQUI !": RETURN
3720 IF ES < 4 THEN PRINT "NO TIENES PARES
      SUFICIENTES !": RETURN
3730 PRINT "EL PULPO NO SE PUEDE MOVER, Y ESO
      NO LE HACE NINGUNA GRACIA !"
3740 FOR I=7 TO 10: OB(I)=0:NEXT
3750 ES=0: SA(20,4)=19
3760 T$(20)="ESTAS EN UN PASILLO CON UNA FUERTE
      CORRIENTE HACIA EL OESTE"
3770 D$(20)="HAY SENTADO UN PULPO ATADO Y
      MUY ENFADADO": RETURN
  
```

El verbo "USA" tiene como misión el permitir al jugador inmovilizar al pulpo de la habitación 20 con cuatro pares de esposas, por ejemplo tecleando "USA LAS ESPOSAS".

La línea 3710 comprueba que la posición del jugador es correcta y que el número del nombre corresponde a las esposas (**NO=14**). La línea 3720 se asegura de que el jugador tiene todos los pares de esposas. Si el programa llega a la línea 3730, aparece un mensaje para avisarnos de que el pulpo está inmovilizado. El jugador se libera de las esposas usando un bucle **FOR...NEXT** que pone la posición del objeto a cero en la línea 3740. Por último, el número de esposas se pone al cero, aparece una nueva salida hacia el oeste y se modifica la descripción de la habitación **T\$** y **D\$**.



Las últimas tres partes del programa corresponden a las órdenes "INVENTARIO", "AYUDA" y "MIRA", cuya utilidad es tal que sin ellas el juego sería extremadamente difícil.

```

3800 REM *****INVENTARIO*****
3810 PRINT: PRINT "ESTAS LLEVANDO:-"
3820 IF IN=0 THEN PRINT "NADA !": RETURN
3830 FOR I=1 TO 10
3840 IF OB(I)=99 THEN PRINT OB$(I)
3850 NEXT: RETURN
  
```

La rutina INVENTARIO informa al jugador sobre los objetos que está llevando. Si **IN** contiene un cero, se imprimirán las palabras "ESTAS LLEVANDO:- NADA!", debido a las líneas 3810 y 3820. Sin embargo, si **IN** no es igual a cero, en la línea 3830 comienza un bucle que comprueba para cada objeto si el jugador lo está llevando o no, y lista por pantalla los objetos que lleva.


```

3900 REM *****AYUDA*****
3910 PRINT: PRINT "ESTOS SON LOS VERBOS QUE
      PUEDES USAR:—"
3920 FOR I=1 TO VV: PRINT VB$(I),,;NEXT I
3930 PRINT: PRINT "(SOLO NECESITAS TECLEAR LAS TRES
      PRIMERAS LETRAS)": RETURN

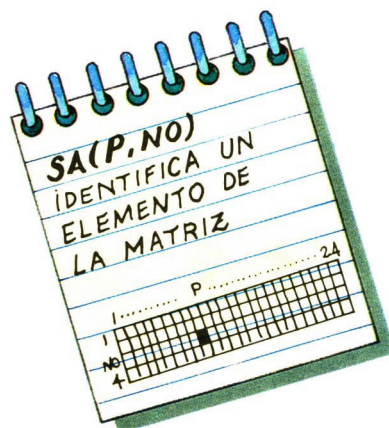
```

Cuando se teclaea "AYUDA", las tres líneas del programa desde la 3910 a la 3930 imprimen cada uno de los verbos disponibles para construir frases aceptables. Para ello se usa un bucle **FOR...NEXT** con la **I** variando hasta el máximo número de verbos (en este caso, el valor de **VV**).

La última sección del programa se refiere al verbo "MIRAR", que permite al jugador examinar la habitación donde se encuentra. Si has seguido las instrucciones de la página 24 (en el apartado **RUN**), ya habrás tecleado esta parte del programa. Esta sección es diferente de las otras rutinas de verbos, ya que se usa como el fin de la subrutina **INICIALIZACION**.

La línea 4010 imprime (**PRINT**) la descripción de la habitación **T\$(P)** y **D\$(P)**. El *flag* de perder, **FP**, se activa si el jugador ha ignorado la advertencia sobre la salida sur de la habitación 16 y ha entrado en la habitación del TIBURON (posición 22).

Las líneas 4020 a 4050 imprimen los objetos que se encuentran en la habitación ocupada por el jugador. Observa que no se hace mención del HUESO MOHOSO en la línea 4030, para hacer el juego más difícil. Por último, en las líneas 4060 a 4100 se indican las salidas disponibles para el jugador desde la habitación donde se encuentra.



```

4000 REM *****MIRA*****
4010 PRINT: PRINT T$(P): PRINT D$(P): IF P=22 THEN
      FP=1: RETURN
4020 FOR I=1 TO 10
4030 IF I=5 AND P=15 THEN 4050
4040 IF OB(I)=P THEN PRINT "AQUI HAY ";OB$(I)
4050 NEXT I
4060 PRINT: PRINT "SALIDAS:"
4070 IF SA(P,1)>0 THEN PRINT "NORTE  ";
4080 IF SA(P,2)>0 THEN PRINT "SUR    ";
4090 IF SA(P,3)>0 THEN PRINT "ESTE  ";
4100 IF SA(P,4)>0 THEN PRINT "OESTE  ";
4110 PRINT: PRINT: PRINT: RETURN

```


Mejora tu programa

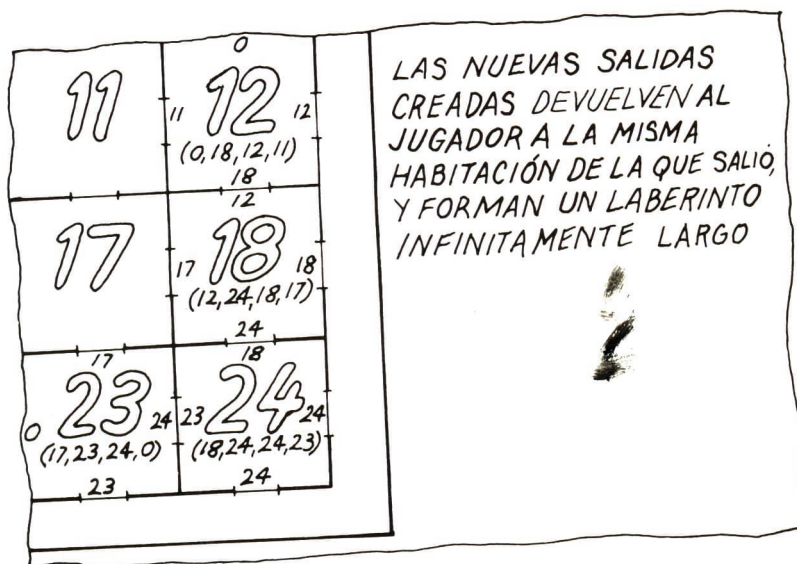
El programa que has teclado te ofrece un desafiante juego de aventuras. Si tú o tus amigos os perdéis, echa un vistazo al mapa de la página 8 para encontrar el camino. Una vez que hayas salvado el juego en una cinta, puedes hacer todo tipo de cambios en éste. Por ejemplo, cuando el juego se gane o se pierda puedes borrar la pantalla y hacer que aparezca un mensaje más vistoso. Las modificaciones que se dan más abajo, crean un laberinto distinto al que se dio inicialmente.



Creación de un laberinto

Muchos juegos de aventuras contienen un laberinto como un obstáculo más a superar por el jugador. En este programa, todo lo que tienes que hacer es poner salidas extras a las habitaciones 12, 18, 23 y 24. El **DATA** modificado que se da a continuación hace esto. En efecto, lo único que ocurre es que cuando tomas una salida equivocada desde una de estas habitaciones, llegas al mismo sitio donde empezaste. Si vas al mapa y añades estas salidas adicionales, serás capaz de descubrir el único camino válido. ¿Por qué no intentas crear un laberinto con más habitaciones?

```
6240 DATA 0,15,0,0, 0,16,0,0, 5,17,12,0, 0,18,12,11
6250 DATA 7,0,0,0, 8,0,15,0, 9,21,16,14, 10,22,17,15
6260 DATA 11,23,18,16, 12,24,18,17, 0,0,20,0, 0,0,21,0
6270 DATA 15,0,0,20, 16,0,0,0, 17,23,24,0, 18,24,24,23
```



El listado completo

El listado completo de nuestro juego de aventuras —las cavernas de Neptuno— está a continuación, tanto para el Spectrum como para el Commodore. Estos listados incluyen todas las secciones que has estudiado, es decir, constituyen la versión completa del juego.

Listado de los programas

SPECTRUM

```
10 CLS
20 GO SUB 5000: REM inicializacion
30 GO SUB 1000: REM entrada
40 GO SUB 2000: REM seleccion
50 IF fg<>1 AND fp<>1 THEN GO TO 30
60 GO SUB 5500: REM fin
70 STOP
1000 REM entrada
1010 LET v$="": LET n$=""
1020 INPUT "Que hago ahora ? ";r$
1030 IF LEN (r$)<3 THEN GO TO 1020
1040 FOR i=1 TO LEN (r$)
1050 IF r$(i TO i)=" " THEN LET v$=r$( TO 3): LET n$=r
$(i+1 TO ): GO SUB 1500: LET i=LEN (r$)
1060 NEXT i
1070 IF n$<>"" THEN RETURN
1080 LET r$=r$( TO 3)
1090 IF r$="NOR" OR r$="SUR" OR r$="EST" OR r$="OES" TH
EN LET v$="VE ": LET n$=r$: RETURN
1100 IF r$="AYU" OR r$="INV" OR r$="MIR" THEN LET n$="
AUX": LET v$=r$: RETURN
1110 PRINT "NO TE ENTIENDO !"
1120 GO TO 1010
1500 FOR j=1 TO LEN (n$)
1510 IF n$(j TO j)=" " THEN LET n$=n$(j+1 TO j+3): LET j=20
1520 NEXT j
1530 IF LEN (n$)>3 THEN LET n$=n$( TO 3)
1540 RETURN
2000 REM seleccion
2010 LET v=0: LET n=0
2020 RESTORE 6530
2030 FOR i=1 TO vv
2040 READ d$: IF d$( TO 3)=v$ THEN LET v=i: LET i=vv
2050 NEXT i
2060 RESTORE 6510
2070 FOR i=1 TO nn
2080 READ d$: IF d$( TO 3)=n$ THEN LET n=i: LET i=nn
2090 NEXT i
2100 IF n=0 OR v=0 THEN PRINT "NO TE ENTIENDO !": RETURN
2110 GO TO 2900+(100*v)
3000 REM *****ir*****
3010 IF n>4 THEN PRINT "IR A DONDE ?": RETURN
3020 IF s(p,n)=0 THEN PRINT "POR AHI NO HAY SALIDA !": RETURN
3030 IF p=16 AND (n=1 OR n=4) AND fa=0 THEN LET fp=1: RETURN
3040 LET p=s(p,n): GO TO 4000
3100 REM *****coger*****
3110 IF n=6 AND p=13 THEN PRINT "ES DEMASIADO PESADO !": RETURN
3120 IF n<8 THEN PRINT "NO SEAS TONTO !": RETURN
```


Continuación Spectrum

```

3130 IF o(n-7)=99 AND n$<>"ESP" THEN PRINT "YA LO HAS COGIDO !": RETURN
3140 FOR i=7 TO 10
3150 IF o(i)=p AND n=14 THEN LET n=i+7: LET es=es+1: LET i=10
3160 NEXT i
3170 IF o(n-7)=p THEN PRINT "O.K.": LET o(n-7)=99: LET in=in+1: RETURN
3180 PRINT "ESO NO ESTA AQUI !": RETURN
3200 REM *****dejar*****
3210 IF n<8 THEN PRINT "NO SEAS TONTO !": RETURN
3220 FOR i=7 TO 10
3230 IF o(i)=99 AND n=14 THEN LET n=i+7: LET es=es-1: LET i=10
3240 NEXT i
3250 IF o(n-7)<>99 THEN PRINT "TU NO TIENES ESO !": RETURN
3260 PRINT "O.K.": LET o(n-7)=p: LET in=in-1
3270 IF p=19 AND n=13 THEN LET fg=1
3280 RETURN
3300 REM *****cortar*****
3310 IF o(1)<>99 THEN PRINT "NO TIENES NADA AFILADO !": RETURN
3320 IF n<>11 THEN PRINT "NO PUEDES CORTAR ESO !": RETURN
3330 IF p<>5 THEN PRINT "NO PUEDES HACER ESO !": RETURN
3340 IF fc=1 THEN PRINT "YA LO HAS HECHO !": RETURN
3350 PRINT "LAS ALGAS CAEN Y APARECE" "UNA VENTANA ABIERTA"
3360 LET o(4)=5: LET s(5,2)=11: LET fc=1
3370 LET l$(5)="Estas en el fondo del mar. Hacia el sur
hay una pared con una ventana abierta.": RETURN
3400 REM *****ponerse*****
3410 IF fa=1 AND n$="ALE" THEN PRINT "YA LAS TIENES PUESTAS !": RETURN
3420 IF o(2)=99 AND n$="ALE" THEN LET fa=1: LET o(2)=0
: PRINT "TE QUEDAN MUY BIEN !": RETURN
3430 IF n=14 AND es>1 THEN PRINT "HACER ESO ES DE ESTUPIDOS !": RETURN
3440 PRINT "NO PUEDES PONERTE ESO !": RETURN
3500 REM *****dar*****
3510 IF p=10 AND n=11 THEN GO TO 3550
3520 IF p=7 AND n=12 THEN GO TO 3570
3530 IF p=20 THEN PRINT "ESO NO TE SERVIRA DE NADA !": RETURN
3540 PRINT "AQUI NADIE QUIERE ESO !": RETURN
3550 IF o(4)<>99 THEN PRINT "NO TIENES ESO !": RETURN
3560 LET o(4)=0: LET o(3)=10: PRINT "DAN VUELTAS ALREDE
DOR TUYO" "Y VES ALGO EN LA ESQUINA": RETURN
3570 IF o(5)<>99 THEN PRINT "NO TIENES ESO !": RETURN
3580 LET o(5)=0: LET s(7,2)=13: PRINT "EL PEZ COGE EL H
UESO" "Y SE RETIRA A UNA ESQUINA"
3590 LET l$(7)="Estas en una caverna. En una
esquina hay un tollo con un hueso.": RETURN
3600 REM *****abrir*****
3610 IF o(3)<>99 THEN PRINT "NO TIENES NINGUNA LLAVE !": RETURN
3620 IF n=5 AND (p=3 OR p=9) THEN PRINT "NO HAY NINGUN
A CERRADURA !": RETURN
3630 IF p<>13 OR n<>6 THEN PRINT "NO PUEDES HACER ESO !": RETURN
3640 IF fab=1 THEN PRINT "YA ESTA ABIERTO !": RETURN
3650 LET fab=1: LET o(7)=13: PRINT "LA LLAVE LO ABRE FACILMENTE"
3660 LET l$(13)="Estas en una habitacion con un cofre
abierto en medio.": RETURN
3700 REM *****usar*****
3710 IF p<>20 OR n<>14 THEN PRINT "NO PUEDES HACER ESO AQUI !": RETURN
3720 IF es<4 THEN PRINT "NO TIENES PARES SUFICIENTES !": RETURN
3730 PRINT "EL PULPO NO SE PUEDE MOVER, Y ESO NO LE HACE NINGUNA GRACIA !"
3740 FOR i=7 TO 10: LET o(i)=0: NEXT i
3750 LET es=0: LET s(20,4)=19: LET l$(20)="Estas en un
pasillo con una corriente muy fuerte hacia el oes
te. Hay sentado un pulpo atado y muy enfadado.": RETURN
3800 REM *****inventario*****
3810 PRINT "ESTAS LLEVANDO: -"
3820 IF in=0 THEN PRINT "NADA !": RETURN
3830 FOR i=1 TO 10
3840 IF o(i)=99 THEN PRINT o$(i)
3850 NEXT i: RETURN
3900 REM *****ayuda*****
3910 PRINT "Estos son los verbos que puedes usar:"

```


Continuación Spectrum

```

3920 RESTORE 6530
3930 FOR i=1 TO vv: READ d$: PRINT d$,: NEXT i
3940 PRINT "(Solo necesitas teclear las tresprimeras letras)": RETURN
4000 REM *****mira*****
4010 PRINT "l$(p): IF p=22 THEN LET fp=1: RETURN
4020 FOR i=1 TO 10
4030 IF i=5 AND p=15 THEN GO TO 4050
4040 IF o(i)=p THEN PRINT "AQUI HAY ";o$(i)
4050 NEXT i
4060 PRINT "SALIDAS:";
4070 IF s(p,1)>0 THEN PRINT "NORTE ";
4090 IF s(p,2)>0 THEN PRINT "SUR ";
4100 IF s(p,3)>0 THEN PRINT "ESTE ";
4110 IF s(p,4)>0 THEN PRINT "OESTE ";
4510 PRINT : RETURN
5000 REM inicializacion
5010 CLS : PRINT AT 0,4;"LAS CAVERNAS DE NEPTUNO"
5020 PRINT "Has encontrado el tapon magico""que corre
sponde al desague del""mar, y decides colocarlo antes"
""de que el agua se salga.""Te sumerges en el oceano y
""comienza la aventura."
5030 PRINT "ASEGURATE DE QUE TIENES PUESTAS LAS MAYUSC
ULAS""Si necesitas ayuda teclea AYUDA"
5040 LET nn=14: LET vv=11: LET p=3: LET in=1: LET es=0:
LET fg=0: LET fp=0: LET fab=0: LET fa=0: LET fc=0
5050>DIM l$(24,176):DIM s(24,4):DIM o$(10,24):DIM o(10)
5060 FOR i=1 TO 24
5070 IF i=17 OR i=18 OR i=23 OR i=24 THEN LET l$(i)=l$(12): GO TO 5090
5080 READ d$: LET l$(i)=d$
5090 NEXT i
5100 FOR i=1 TO 24: FOR j=1 TO 4: READ s(i,j): NEXT j: NEXT i
5110 FOR i=1 TO 10: READ o$(i): READ o(i): NEXT i
5120 GO TO 4000: REM mira
5500 REM fin
5510 IF p=22 THEN PRINT "DE UNA DENTELLADA, EL TIBURON
""TE ARRANCA LA CABEZA." : GO TO 5550
5520 IF p=16 THEN PRINT "LOS PINCHOS DE LOS ERIZOS SON
""MUY VENENOSOS !""SUFRES UNA MUERTE TERRIBLE.""LA P
ROXIMA VEZ TOMA""MAS PRECAUCIONES": GO TO 5550
5530 PRINT "EL TAPON SE METE EN EL AGUJERO""Y EL AGUA
DEJA DE SALIR.""FELICITACIONES!""HAS SALVADO LOS OCEANOS"
5550 INPUT "QUIERES JUGAR OTRA VEZ ? ";r$
5560 IF r$(1)="S" THEN RUN
5570 PRINT "ADIOS": RETURN
6000 REM *****descripciones*****
6010 DATA "Estas en el fondo del mar. El camino ha
cia el oeste esta bloqueado por un arrecife de coral."
6020 DATA "Estas en el fondo del mar. En el sur, u
na pared se alza enfrente de ti."
6030 DATA "Estas enfrente de una puerta de madera que e
sta en la pared. No ves el picaporte."
6040 DATA "Estas en el fondo del mar. En el sur, u
na pared se alza enfrente de ti."
6050 DATA "Estas en el fondo del mar. En el sur ha
y una pared con algas marinas formando un
cuadrado sobre ella."
6060 DATA "Estas en el fondo del mar. Hacia el sur
hay una pared. Un precipicio bloquea el camino al este."
6070 DATA "Estas en una caverna larga y estrecha. Al
fondo ves un tollo nadando en circulo."
6080 DATA "Estas en una habitacion llena deluz. Las par
edes, el techo y el suelo brillan."
6090 DATA "Estas en una oscura caverna con una puerta e
norme al fondo. No ves el picaporte."
6100 DATA "Estas en una habitacion llena dehambrientos
caballitos de mar.Tehusmean la mano amistosamente."
6110 DATA "Estas en una pequena sala cuadrada. La
pared del norte tiene una ventana, a traves de la

```


Continuación Spectrum

que se ve el fondo del mar."

6120 DATA "Estas en una extraña sala cuadrada. Las paredes, el suelo, y el techo son cuadrados, así como las puertas."

6130 DATA "Estas en una diminuta habitación que esta casi acupada por un baul con las iniciales D.J."

6140 DATA "Estas en una habitación muy fría. El fango te rodea y notas una pequeña corriente hacia el este."

6150 DATA "Estas en un lugar realmente tetrico ! Esta todo lleno de los huesos de exploradores anteriores."

6160 DATA "Estas en una sala cuadrada. Encima de la salida sur pone NO ENTRES. Las puertas del norte y del oeste estan llenas de hormigueantes erizos de mar."

6170 DATA "Estas en una habitación circular con una fuerte corriente que gira en remolino y se cuela por el agujero del fondo."

6180 DATA "Estas en un pasillo con una fuerte corriente hacia el oeste. El camino esta bloqueado por los brazos de un tremendo pulpo."

6190 DATA "Estas en el camarote del capitane un barco. Sientes el flujo del agua hacia el oeste."

6200 DATA "Ves un torbellino de agua, de donde sale un gran tiburón blanco."

6210 REM *****salidas*****

6220 DATA 0,0,2,0, 0,0,3,1, 0,0,4,2, 0,0,5,3

6230 DATA 0,0,6,4, 0,0,0,5, 0,0,8,0, 0,14,0,7

6240 DATA 0,15,0,0, 0,16,0,0, 5,17,12,0, 0,18,12,11

6250 DATA 7,0,0,0, 8,0,15,0, 9,21,16,14, 10,22,17,15

6260 DATA 11,23,18,16, 12,24,18,17, 0,0,20,0, 0,0,21,0

6270 DATA 15,0,0,20, 16,0,0,0, 17,23,24,0, 18,24,24,23

6300 REM *****objetos*****

6310 DATA "UN CUCHILLO",2

6320 DATA "UN PAR DE ALETAS",6

6330 DATA "UNA LLAVE",0

6340 DATA "UN GRUPO DE ALGAS",0

6350 DATA "UN HUESO MOHOSO",15

6360 DATA "EL TAPON MAGICO",99

6370 DATA "UNAS ESPOSAS AMARILLAS",0

6380 DATA "UNAS ESPOSAS VERDES",9

6390 DATA "UNAS ESPOSAS ROJAS",17

6400 DATA "UNAS ESPOSAS AZULES",11

6500 REM *****nombres*****

6510 DATA "NOR", "SUR", "EST", "OES", "AUX", "BAU", "VEN", "CUC", "ALE", "LLA", "ALG", "HUE", "TAP", "ESP"

6520 REM *****verbos*****

6530 DATA "VE", "COGE", "DEJA", "CORTA", "PONTE", "DALE", "ABRE", "USA", "INVENTARIO", "AYUDA", "MIRA"

COMMODORE

10 REM LAS CAVERNAS DE NEPTUNO

20 GOSUB 5000: REM INICIALIZACION

30 GOSUB 1000: REM ENTRADA

40 GOSUB 2000: REM SELECCION

50 IF FG<>1 AND FP<>1 THEN GOTO 30

60 GOSUB 5500: REM FIN

70 STOP

1000 REM *****ENTRADA*****

1010 VB\$="": NO\$="": R\$=""

1020 INPUT "QUE HAGO AHORA ? ";R\$

1030 FOR I=1 TO LEN(R\$)

1040 IF MID\$(R\$,I,1)=" " THEN VB\$=LEFT\$(R\$,3):
NO\$=RIGHT\$(R\$,I+1): GOSUB 1500: I=LEN(R\$)

1050 NEXT I

1060 IF NO\$<>"" THEN RETURN

1070 R\$=LEFT\$(R\$,3)

1080 IF R\$="NOR" OR R\$="SUR" OR R\$="EST" OR
R\$="OES" THEN VB\$="VE ": NO\$=R\$: RETURN

Continuación Commodore

```

1090 IF R$="AYU" OR R$="INV" OR R$="MIR" THEN NO$="AUX": VB$=R$: RETURN
1100 PRINT "NO TE ENTIENDO !"
1110 GOTO 1010
1500 FOR J=1 TO LEN(NO$)
1510 IF MID$(NO$,J,1)=" " THEN NO$=MID$(NO$,J+1,3): J=20
1520 NEXT J
1530 IF LEN(NO$)>3 THEN NO$=LEFT$(NO$,3)
1540 RETURN
2000 REM *****SELECCION*****
2010 VB=0: NO=0
2020 FOR I=1 TO VV
2030 IF VB$=LEFT$(VB$(I),3) THEN VB=I: I=VV
2040 NEXT
2050 FOR I=1 TO NN
2060 IF NO$=NO$(I) THEN NO=I: I=NN
2070 NEXT
2080 IF NO=0 OR VB=0 THEN PRINT "NO TE ENTIENDO !": RETURN
2090 ON VB GOTO 3000,3100,3200,3300,3400,3500,3620,3700,3800,3900,4000
3000 REM *****IR*****
3010 IF NO>4 THEN PRINT "IR A DONDE ?": RETURN
3020 IF SA(P,NO)=0 THEN PRINT "POR AHI NO HAY SALIDA": RETURN
3030 IF P=16 AND (NO=1 OR NO=4) AND FA=0 THEN FP=1: RETURN
3040 P=SA(P,NO): GOSUB 4000
3050 RETURN
3100 REM *****COGER*****
3110 IF NO=6 AND P=13 THEN PRINT "ES DEMASIADO PESADO !": RETURN
3120 IF NO<8 THEN PRINT "NO SEAS TONTO !":RETURN
3130 IF OB(NO-7)=99 AND NO$<>"ESP" THEN PRINT "YA LO HAS COGIDO !": RETURN
3140 FOR I=7 TO 10
3150 IF (OB(I)=P AND NO=14) THEN NO=I+7: ES=ES+1: I=10: IN=IN+1
3160 NEXT I
3170 IF OB(NO-7)=P THEN PRINT "O.K.": OB(NO-7)=99: IN=IN+1: RETURN
3180 PRINT "ESO NO ESTA AQUI !": RETURN
3200 REM *****DEJAR*****
3210 IF NO<8 THEN PRINT "NO SEAS TONTO !":RETURN
3220 FOR I=7 TO 10
3230 IF (OB(I)=99 AND NO=14) THEN NO=I+7:ES=ES-1: I=10: IN=IN-1
3240 NEXT I
3250 IF OB(NO-7)<>99 THEN PRINT "TU NO TIENES ESO": RETURN
3260 PRINT "O.K.": OB(NO-7)=P: IN=IN-1
3270 IF P=19 AND NO=13 THEN FG=1
3280 RETURN
3300 REM *****CORTAR*****
3310 IF OB(1) <> 99 THEN PRINT "NO TIENES NADA AFILADO !": RETURN
3320 IF NO<>11 THEN PRINT "NO PUEDES CORTAR ESO !": RETURN
3330 IF P<>5 THEN PRINT "NO PUEDES HACER ESO !": RETURN
3340 IF FC=1 THEN PRINT "YA LO HAS HECHO !":RETURN
3350 PRINT "LAS ALGAS CAEN Y APARECE UNA VENTANA ABIERTA"
3360 OB(4)=5: SA(5,2)=11: FC=1
3370 D$(5)=LEFT$(D$(5),22)+"UNA VENTANA ABIERTA": RETURN
3400 REM *****PONERSE*****
3410 IF FA=1 AND NO$="ALE" THEN PRINT "YA LAS TIENES PUESTAS !": RETURN
3420 IF OB(2)=99 AND NO$="ALE" THEN FA=1: OB(2)=0:
PRINT "TE QUEDAN MUY BIEN !": RETURN
3430 IF NO=14 AND ES>0 THEN PRINT "HACER ESO ES DE ESTUPIDOS !": RETURN
3440 PRINT "NO PUEDES PONERTE ESO !": RETURN
3500 REM *****DAR*****
3510 IF P=10 AND NO=11 THEN 3550
3520 IF P=7 AND NO=12 THEN 3580
3530 IF P=20 THEN PRINT "ESO NO TE SERVIRA DE NADA": RETURN
3540 PRINT "AQUI NADIE QUIERE ESO !": RETURN
3550 IF OB(4) <> 99 THEN PRINT "NO TIENES ESO !": RETURN
3560 OB(4)=0: OB(3)=10
3570 PRINT "VES ALGO EN LA ESQUINA":RETURN
3580 IF OB(5)<>99 THEN PRINT "NO TIENES ESO !":RETURN
3590 OB(5)=0: SA(7,2)=13
3600 PRINT "EL PEZ COGE EL HUESO Y SE RETIRA A UNA ESQUINA"
3610 T$(7)="ESTAS EN UNA CAVERNA."
3615 D$(7)="EN UNA ESQUINA HAY UN TOLLO CON UN HUESO": RETURN

```


Continuación Commodore

```

3620 REM *****ABRIR*****
3630 IF OB(3)<>99 THEN PRINT "NO TIENES NINGUNA LLAVE !":RETURN
3640 IF NO=5 AND (P=3 OR P=9) THEN PRINT "NO HAY CERRADURA !": RETURN
3650 IF P<>13 OR NO<>6 THEN PRINT "NO PUEDES HACER ESO !: RETURN
3660 IF FAB=1 THEN PRINT "YA ESTA ABIERTO !": RETURN
3670 FAB=1: OB(7)=13: PRINT "LA LLAVE LO ABRE FACILMENTE"
3680 T$(13)="ESTAS EN UNA HABITACION CON UN COFRE ABIERTO EN MEDIO"
3690 D$(13)="": RETURN
3700 REM *****USAR*****
3710 IF P<>20 OR NO<>14 THEN PRINT "NO PUEDES HACER ESO AQUI !": RETURN
3720 IF ES < 4 THEN PRINT "NO TIENES PARES SUFICIENTES !": RETURN
3730 PRINT "EL PULPO NO SE PUEDE MOVER, Y ESO NO LE HACE NINGUNA GRACIA !"
3740 FOR I=7 TO 10: OB(I)=0:NEXT
3750 ES=0: SA(20,4)=19
3760 T$(20)="ESTAS EN UN PASILLO CON UNA FUERTE CORRIENTE HACIA EL OESTE"
3770 D$(20)="HAY SENTADO UN PULPO ATADO Y MUY ENFADADO": RETURN
3800 REM *****INVENTARIO*****
3810 PRINT: PRINT "ESTAS LLEVANDO:-"
3820 IF IN=0 THEN PRINT "NADA !": RETURN
3830 FOR I=1 TO 10
3840 IF OB(I)=99 THEN PRINT OB$(I)
3850 NEXT: RETURN
3900 REM *****AYUDA*****
3910 PRINT: PRINT "ESTOS SON LOS VERBOS QUE PUEDES USAR:-"
3920 FOR I=1 TO VV: PRINT VB$(I),:NEXT I
3930 PRINT: PRINT "(SOLO NECESITAS TECLEAR LAS TRES
PRIMERAS LETRAS)": RETURN
4000 REM *****MIRA*****
4010 PRINT: PRINT T$(P): PRINT D$(P): IF P=22 THEN FP=1: RETURN
4020 FOR I=1 TO 10
4030 IF I=5 AND P=15 THEN 4050
4040 IF OB(I)=P THEN PRINT "AQUI HAY ";OB$(I)
4050 NEXT I
4060 PRINT: PRINT "SALIDAS:"
4070 IF SA(P,1)>0 THEN PRINT "NORTE ";
4080 IF SA(P,2)>0 THEN PRINT "SUR ";
4090 IF SA(P,3)>0 THEN PRINT "ESTE ";
4100 IF SA(P,4)>0 THEN PRINT "OESTE ";
4110 PRINT: PRINT: RETURN
5000 REM *****INICIALIZACION*****
5010 PRINT " "
5020 PRINTSPC(12)"LAS CAVERNAS DE NEPTUNO"
5030 PRINT "HAS ENCONTRADO EL TAPON MAGICO QUE"
5040 PRINT "CORRESPONDE AL FONDO DEL MAR, Y"
5050 PRINT "DECIDES COLOCARLO ANTES DE QUE EL"
5060 PRINT "AGUA SE SALGA. TE SUMERGES EN EL"
5070 PRINT "OCEANO Y COMIENZA LA AVENTURA."
5080 PRINT "SI NECESITAS AYUDA TECLEA 'AYUDA'"
5100 NN=14: VV=11: P=3: IN=1: ES=0: FG=0: FP=0: FAB=0: FA=0: FC=0
5110 DIM T$(24): DIM D$(24): DIM SA(24,4): DIM NO$(NN): DIM VB$(VV)
5120 DIM OB$(10): DIM OB(10)
5130 FOR I=1 TO 24
5140 IF I=17 OR I=18 OR I=23 OR I=24 THEN T$(I)=T$(12):
D$(I)=D$(12): GOTO 5160
5150 READ T$,D$: T$(I)=T$: D$(I)=D$
5160 NEXT I
5170 FOR I=1 TO 24: FOR J=1 TO 4: READ SA(I,J): NEXT J,I
5180 FOR I=1 TO 10: READ OB$(I),OB(I): NEXT I
5190 FOR I=1 TO NN: READ NO$(I): NEXT I
5200 FOR I=1 TO VV: READ VB$(I): NEXT I
5210 GOSUB 4000: RETURN: REM MIRA
5500 REM *****FIN*****
5510 IF P=22 THEN PRINT "DE UNA DENTELLADA, EL TIBURON
TE ARRANCA LA CABEZA": GOTO 5570
5520 IF P=16 THEN PRINT "LOS PINCHOS DE LOS ERIZOS SON
MUY VENENOSOS. SUFRES UNA MUERTE TERRIBLE.":GOTO 5570
5540 PRINT "EL TAPON SE METE EN EL AGUJERO"
5550 PRINT "Y EL AGUA DEJA DE SALIR. HAS SALVADO"
5560 PRINT "LOS OCEANOS. FELICITACIONES !!"

```


Continuación Commodore

```

5570 INPUT "QUIERES JUGAR OTRA VEZ ? ";R$
5580 IF LEFT$(R$,1)="S" THEN RUN
5590 PRINT "ADIOS": RETURN
6000 REM *****DESCRIPCIONES*****
6010 DATA "ESTAS EN EL FONDO DEL MAR."
6011 DATA "EL CAMINO HACIA EL OESTE ESTA BLOQUEADO POR UN ARRECIFE DE CORAL"
6020 DATA "ESTAS EN EL FONDO DEL MAR."
6021 DATA "EN EL SUR, UNA PARED SE ALZA ENFRENTA DE TI"
6030 DATA "ESTAS ENFRENTA DE UNA PUERTA DE MADERA"
6031 DATA "NO VES EL PICAPORTE"
6040 DATA "ESTAS EN EL FONDO DEL MAR."
6041 DATA "EN EL SUR, UNA PARED SE ALZA ENFRENTA DE TI"
6050 DATA "ESTAS EN EL FONDO DEL MAR. EN EL"
6051 DATA "SUR HAY UNA PARED CON ALGAS FORMANDO UN CUADRADO SOBRE ELLA"
6060 DATA "ESTAS EN EL FONDO DEL MAR. HACIA EL"
6061 DATA "SUR HAY UNA PARED. UN PRECIPICIO BLOQUEA EL CAMINO AL ESTE"
6070 DATA "ESTAS EN UNA CAVERNA LARGA Y ESTRECHA"
6071 DATA "AL FONDO VES UN TOLLO NADANDO EN CIRCULO"
6080 DATA "ESTAS EN UNA HABITACION LLENA DE LUZ"
6081 DATA "LAS PAREDES, EL SUELO Y EL TECHO BRILLAN"
6090 DATA "ESTAS EN UNA OSCURA CAVERNA CON UNA PUERTA"
6091 DATA "ENORME AL FONDO. NO VES EL PICAPORTE."
6100 DATA "ESTAS EN UNA HABITACION LLENA DE HAMBRIENTOS"
6101 DATA "CABALLITOS DE MAR. TE HUSMEAN LA MANO AMISTOSAMENTE"
6110 DATA "ESTAS EN UNA PEQUENA SALA CUADRADA."
6111 DATA "LA PARED DEL NORTE TIENE UNA VENTANA,
A TRAVES DE LA QUE SE VE EL FONDO DEL MAR"
6120 DATA "ESTAS EN UNA ESTRANA SALA CUADRADA. LAS"
6121 DATA "PAREDES, EL TECHO Y EL SUELO SON CUADRADOS, ASI COMO LAS PUERTAS"
6130 DATA "ESTAS EN UNA DIMINUTA HABITACION OCUPADA"
6131 DATA "POR UN BAUL CON LAS INICIALES D.J."
6140 DATA "ESTAS EN UNA HABITACION MUY FRIA. EL"
6141 DATA "FANGO TE RODEA Y NOTAS UNA PEQUENA CORRIENTE HACIA EL ESTE"
6150 DATA "ESTAS EN UN LUGAR REALMENTE TETRICO !"
6151 DATA "ESTA LLENO DE HUESOS DE EXPLORADORES ANTERIORES !"
6160 DATA "ESTAS EN UNA SALA CUADRADA. SOBRE LA SALIDA SUR PONE 'NO ENTRES'"
6161 DATA "LAS PUERTAS DEL OESTE Y DEL NORTE ESTAN
LLENAS DE HORMIGUEANTES ERIZOS DE MAR"
6170 DATA "ESTAS EN UNA SALA CIRCULAR CON UNA FUERTE CORRIENTE"
6171 DATA "QUE GIRA EN REMOLINO Y SE CUELA POR EL AGUJERO DEL FONDO"
6180 DATA "ESTAS EN UN PASILLO CON UNA FUERTE CORRIENTE HACIA EL OESTE"
6181 DATA "EL CAMINO ESTA BLOQUEADO POR LOS BRAZOS DE UN TREMENDO PULPO"
6190 DATA "ESTAS EN EL CAMAROTE DEL CAPITAN DE UN BARCO"
6191 DATA "SIENTES EL FLUJO DEL AGUA HACIA EL OESTE"
6200 DATA "VES UN TORBELLINO DE AGUA, DE DONDE"
6201 DATA "SALE UN GRAN TIBURON BLANCO"
6210 REM *****SALIDAS*****
6220 DATA 0,0,2,0,0,0,3,1,0,0,4,2,0,0,5,3
6230 DATA 0,0,6,4,0,0,0,5,0,0,8,0,0,14,0,7
6240 DATA 0,15,0,0,0,16,0,0,5,17,12,0,0,18,0,11
6250 DATA 7,0,0,0,8,0,15,0,9,21,16,14,10,22,17,15
6260 DATA 11,23,18,16,12,24,0,17,0,0,20,0,0,0,21,0
6270 DATA 15,0,0,20,0,0,0,0,17,0,24,0,18,0,0,23
6300 REM *****OBJETOS*****
6310 DATA UN CUCHILLO,2
6320 DATA UN PAR DE ALETAS,6
6330 DATA UNA LLAVE,0
6340 DATA UN GRUPO DE ALGAS,0
6350 DATA UN HUESO MOHOSO,15
6360 DATA EL TAPON MAGICO,99
6370 DATA UNAS ESPOSAS AMARILLAS,0
6380 DATA UNAS ESPOSAS VERDES,9
6390 DATA UNAS ESPOSAS ROJAS,17
6400 DATA UNAS ESPOSAS AZULES,11
6500 REM *****NOMBRES*****
6510 DATA NOR,SUR,EST,OES,AUX,BAU,VEN,CUC,ALE, LLA,ALG,HUE,TAP,ESP
6520 REM *****VERBOS*****
6530 DATA VE ,COJE,DEJA,CORTA,PONTE,DALE,ABRE,USA,INVENTARIO,AYUDA,MIRA

```


Biblioteca Pública de Cádiz
Biblioteca Pública de Cádiz



1008560

JUV 681 ROD ban

ESTA NUEVA SERIE CONSTITUYE UNA INTRODUCCIÓN A
EL ARTE DE PROGRAMAR ORDENADORES. CADA LIBRO
ENSEÑA CÓMO ESTRUCTURAR UN PROGRAMA EN
RUTINAS, Y AL MISMO TIEMPO EXPLICA Y
ANALIZA LO QUE SE LE PIDE AL ORDENADOR
QUE HAGA Y POR QUÉ.

¡Y ADEMÁS ES DIVERTIDO! CADA LIBRO CONTIENE
COMO MÍNIMO UN JUEGO DE ORDENADOR,
APASIONANTE Y ORIGINAL, ESTRUCTURADO
DE TAL MODO QUE PUEDE AMPLIARSE
O REDUCIRSE, DE ACUERDO CON EL NIVEL
O LOS DESEOS DEL USUARIO.

TÍTULOS DE LA SERIE

Gráficos - Iniciación al basic

El banco de datos

