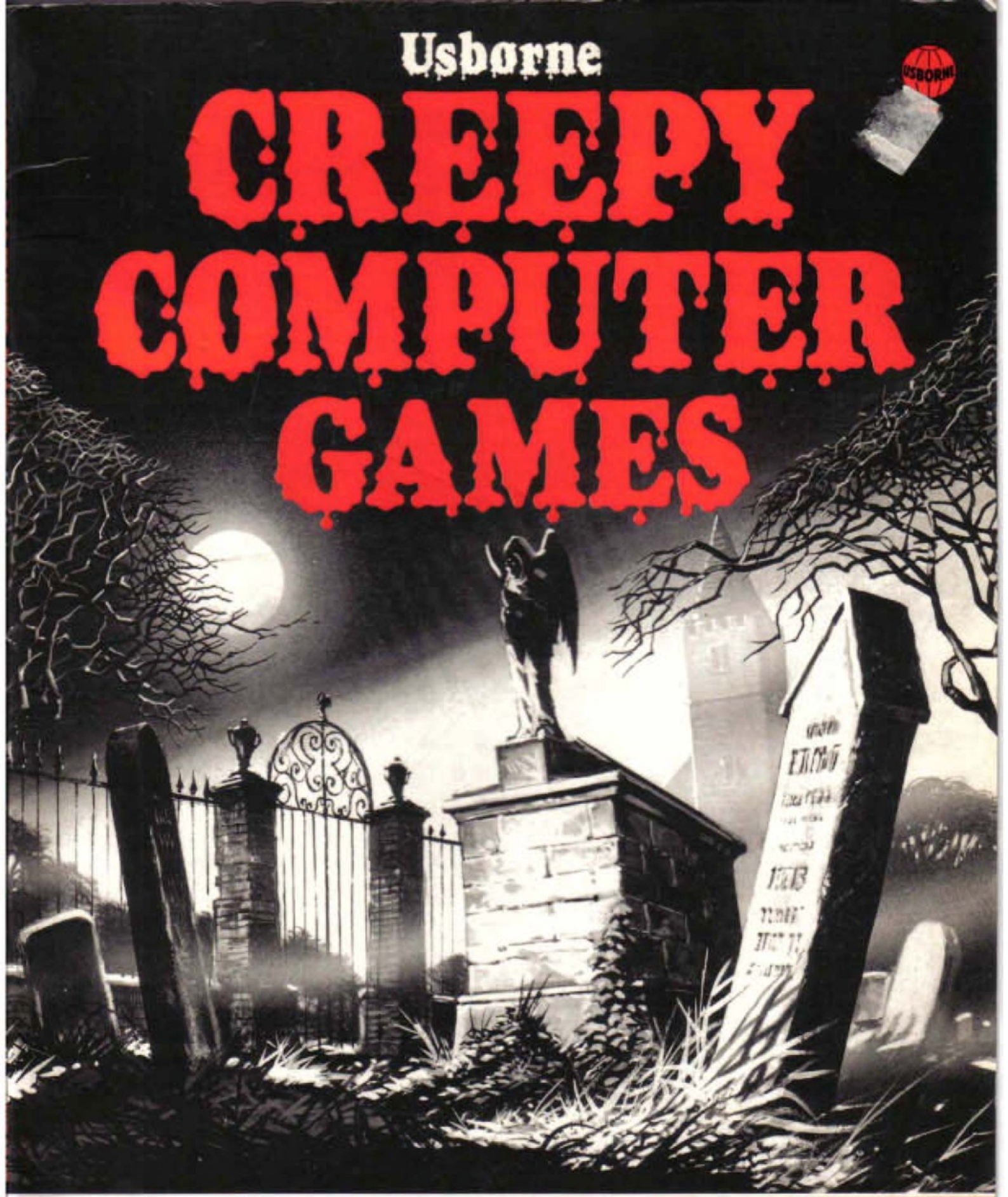
Usborne

CREEPY COMPUTER GAMES



.....FOR...SPECTRUM...ZX81...BBC.....

.....ORIC...VIC20...TRS-80...APPLE...DRAGON.....

Usborne

CREEPY COMPUTER GAMES



Contents

3 Computer Nightmare

4 The Number Wizard

6 Ghost Guzzler

8 Spiderwoman

10 Gravedigger

12 Mad House

14 Ghost Maze

16 Seance

17 Answers to puzzles

About this book

This book contains simple games programs to play on a microcomputer. They are written for use on ZX81 (with extra memory), ZX Spectrum, BBC, VIC 20, Dragon, Oric, TRS-80 and Apple. The main listing for each program works on the ZX81 and lines which need changing for the other computers are marked with these symbols:

▲ VIC and Pet
✱ BBC
■ TRS-80
● Apple
ǀ ZX Spectrum
▽ Dragon
○ Oric

Every time you see the symbol for the computer you are using, look below for the corresponding line number with the same symbol and type that in instead.

Running the programs

When you have typed in the listing, check it carefully. Remember that it is very easy to make mistakes when you are typing programs, even if you are quite experienced. To start the game, type RUN. If the program doesn't work properly, it is quite likely that there is still a mistake in it somewhere, so LIST the program and check again. When the game is over, the computer will say something like BREAK IN 200. To play again, you have to type RUN.

Experimenting with the programs

There are suggestions for changes you can make to the programs throughout the book, but don't be afraid to experiment with changes of your own. As the programs are written to fit so many different computers, they often do not make use of the special features available on some computers. In particular they do not make any use of graphics, colour or sound. You could try rewriting the programs for your computer using its graphics, sound and any other special features it may have.

Changing the speed

Next to some of the games you will find instructions for changing the speed. You may well find you need to do this as these games depend on the speed of both your computer and your reactions. You will almost certainly want to speed these games up after you have played them a few times in any case.

Computer Nightmare is based on an idea by Brendon Kavanagh
Number Wizard, Ghost Guzzler and **Ghost Maze** were written by Colin Reynolds
Spiderwoman was written by Val Robinson, **Gravedigger** by Alan Ramsey
Mad House by Keith Campbell and **Seance** by Chris Oxlade.

Illustrated by Rob McCaig
Edited by Jenny Tyler
Program editor: Chris Oxlade

First published in 1983 by Usborne Publishing Ltd, 20 Garrick Street, London WC2E 9BJ. © 1983 Usborne Publishing Ltd.

The name Usborne and the device are Trade Marks of Usborne Publishing Ltd.

Computer Nightmare

You are a late-night computer addict and you've fallen asleep at the keyboard. Suddenly your computer comes alive and starts hurling numbers and abuse at you. To beat it you have to match the numbers as they appear on the screen. Your starting score of 300 is increased if you hit the right number and decreased if you don't. If you can get your score up to 500 the computer will give up and you win, but if it goes down to zero, you will become a slave to your computer. It's a real nightmare! Can you stay sane?

```
 5 LET F$="0"
10 DIM C$(5,23)
20 LET S=300
30 LET C$(1)="** MICROS RULE! **"
40 LET C$(2)="*PEOPLE ARE STUPID*"
50 LET C$(3)="+A ROBOT FOR PRESIDENT!"
60 LET C$(4)="!COMPUTERS ARE GREAT!"
70 LET C$(5)="*I'M BETTER THAN YOU!*"
80 CLS
90 LET N=INT(RND*9)+1

100 PRINT TAB(5);N
110 PRINT TAB(15);S
120 IF RND>0.5 THEN GOTO 150
130 PRINT
140 PRINT C$(INT(S/100)+1)
150 IF S<60 THEN PRINT "
    <THERE'S NO HOPE>"
160 IF S>440 THEN PRINT "URK! HELP!!"
170 FOR I=1 TO 10
180 LET A$=INKEY$
190 IF A$<>"" THEN LET F$=A$
200 NEXT I
210 LET S=S-10
220 IF VAL(F$)<>N THEN GOTO 240
230 LET S=S+10+N*2

240 IF S<0 THEN GOTO 270
250 IF S>500 THEN GOTO 290

260 GOTO 80
270 PRINT "YOU'RE NOW MY SLAVE"
280 STOP
290 PRINT "OK. YOU WIN (THIS TIME)"
300 STOP
```

How it works

Sets starting value for score.

Stores comments in memory.

Clears screen.

Chooses a number from 1 to 9 and puts it in N.

Prints N and your score (S) at different points on the screen.

Decides whether to go to 150.

Prints messages depending on your score.

Looks to see if you are pressing a key. If you are, it stores it in F$.

Decreases score by 10.

If you pressed the wrong key, computer jumps to line 240 to miss out score increase at line 230.

Checks score to see if you have won or lost and jumps down to end game if you have.

Goes back for another turn.

The speed of the game depends on this number. You will probably need to adjust it to fit the speed of your computer and your reactions. A higher number makes the game slower.

The above listing will work on a ZX81. For other computers make the changes below.

```
10 DIM C$(5)
80 HOME
80 PRINT CHR$(147)
90 LET N=INT(RND(1)*9)+1
90 LET N=INT(RND(0)*9)+1
120 IF RND(1)>0.5 THEN GOTO 150
120 IF RND(0)>0.5 THEN GOTO 150
170 FOR I=1 TO 400
175 A$=""
180 IF PEEK(-16384)>127 THEN GET A$
180 LET A$=INKEY$(0)
180 GET A$
180 LET A$=KEY$
```

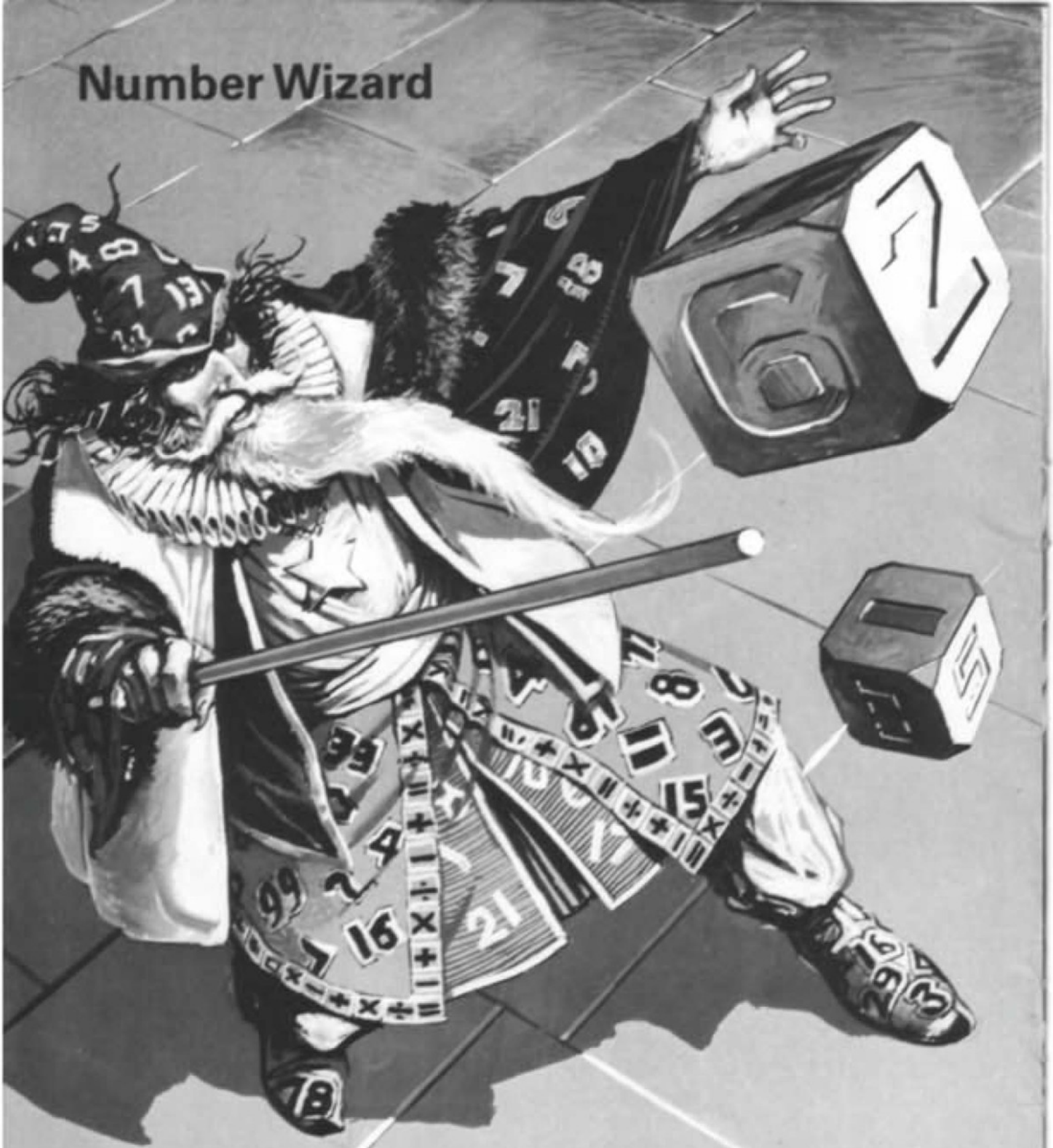
Changes to try

Lines 20, 210 and 230 control the scoring. Change the numbers in these lines and see what happens.

Puzzle

Can you think of a way of changing the program to make it produce letters on the screen instead of numbers?

Number Wizard



This is the game the Number Wizard plays with all his visitors. He conjures the numbers 1 to 9 in the air and then rolls two dice. You must give him two numbers which both appear in the air *and*, when added together, give the same total as the two numbers on the dice. Once you have used a number, its image disappears from the air and you cannot use it again. You win if all the numbers have disappeared before all your turns are used up. (You get an extra turn for every double the Wizard throws.)

You are allowed to use zero as many times as you like as one of your numbers. If you can't go, move on to the next turn by using two zeros. See how many times you can beat the Wizard.

How the program works

```
10 PRINT "THE NUMBER WIZARD"
20 DIM A(10)
30 LET T=8
40 FOR I=1 TO 10
50 LET A(I)=1
60 NEXT I
▲●70 CLS
80 LET A(1)=1

90 LET V=0
100 FOR I=2 TO 10
110 IF A(I)<>0 THEN GOTO 150
120 PRINT " ";
130 LET V=V+1
140 GOTO 160
150 PRINT I-1;
160 NEXT I
170 PRINT
180 IF V=9 THEN GOTO 370

190 PRINT "YOU'VE ";T;" TURNS LEFT"
■▲●200 LET C=INT(RND*6+1)
✱○▽210 LET B=INT(RND*6+1)
220 PRINT "THE DICE THROW IS ";C;",";B
230 PRINT "WHAT ARE YOUR NUMBERS?"
240 IF B=C THEN LET T=T+2

250 LET T=T-1
260 IF T<0 THEN GOTO 350
270 INPUT N
280 INPUT M
290 IF M>9 OR N>9 THEN PRINT
    "TOO BIG - TRY AGAIN"
300 IF M>9 OR N>9 THEN GOTO 230
310 IF M+N<>B+C OR A(N+1)=0 OR
    A(M+1)=0 THEN GOTO 70

320 LET A(M+1)=0
330 LET A(N+1)=0
340 GOTO 70
350 PRINT "THE WIZARD WON"
360 STOP
370 PRINT "YOU WON"
380 STOP
```

Sets up an array, **A**.

Sets the number of goes allowed.

Loops round putting a 1 into each space in **A**.

Makes sure **A**(1) always contains 1 (this is so you can use 0 as many times as you like).

V keeps count of how many numbers have been used. (Set to 0 for start.)

Loops round printing numbers. If **A**(I) = 0 number has been used, so it leaves a space and adds 1 to **V**.

If all numbers have been used, jumps down to end game.

Tells you how many goes left.

Chooses numbers for dice throw and prints them.

If throw was a double, gives you extra go.

Reduces goes by 1, checks **T** is not zero.

Gets two numbers from you and checks they are valid.

If they don't add up or one of them has been used, goes back for another go.

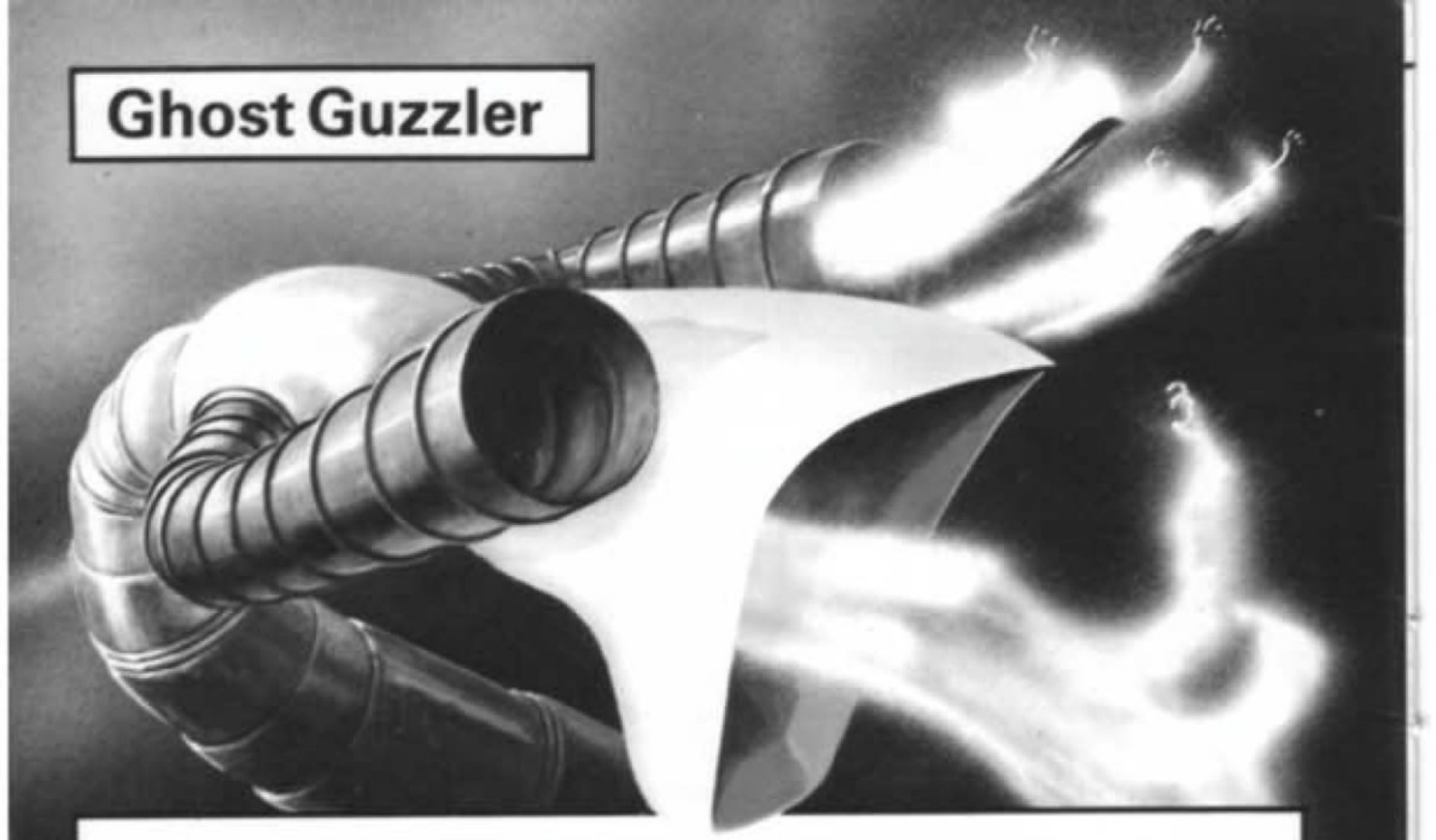
If numbers are OK, the 1s in their spaces in **A** are replaced with 0s to show they've been used.

The above listing will work on a ZX81. For other computers make the changes below.

```
   ●70 HOME
   ▲70 PRINT CHR$(147)
✱○▲●200 LET C=INT(RND(1)*6+1)
  ▽■200 LET C=INT(RND(0)*6+1)
✱○▲●210 LET B=INT(RND(1)*6+1)
  ▽■210 LET B=INT(RND(0)*6+1)
```

Perhaps you can work out a way of making the computer go back for another game automatically and build up a record of how many times you've won and lost. You could also work out a terrible fate for players who lose too many times . . .

Ghost Guzzler



The ghosts in this game are numbers rushing across the screen. To catch them, you activate your ghost guzzler by pressing key X, but it only works when the number on it is the same as that of the attacking ghost. You can increase the guzzler's number by pressing key M (when it gets to 9, it goes back to 0 again). If you fail to guzzle a ghost, it will snatch away one of your lives (shown as / top left of the screen). See how good *you* are at guzzling ghosts.

How the program works

```
10 PRINT "GHOST GUZZLER"
20 LET S=0
30 LET Y=0
40 LET L=3
50 LET N=INT(RND*10)
60 LET I=1
70 CLS
80 FOR J=1 TO L
90 PRINT "/";
100 NEXT J
110 PRINT
120 PRINT TAB(I);N;TAB(18);":";Y
```

- Lines 20–30: Set score and your number to zero for start of game.
- Line 40: Sets number of "lives" to 3.
- Line 50: Gets a number between 0 and 9 for the ghost.
- Line 60: Starts counter for distance ghost can travel before it reaches you.
- Lines 80–100: Loop which prints your "lives" as / top left of the screen.
- Line 120: Prints ghost number, barrier ":" and you at appropriate positions on screen.

This is a good game for adding sounds to. Try adding sub-routines for something blood-curdling when you lose a life and something suitable for ghosts to make when they are guzzled.

```
    130 LET B$=INKEY$
    140 IF B$="X" THEN GOTO 220
    150 IF B$="M" THEN LET Y=Y+1

    160 IF Y=10 THEN LET Y=0

    170 LET I=I+1
    180 IF I<18 THEN GOTO 70

    190 LET L=L-1

    200 IF L>0 THEN GOTO 50
    210 GOTO 270

    220 IF Y<>N THEN GOTO 170
    230 PRINT "GOT IT"
    240 PRINT "*******"
    250 LET S=S+(18-I)
    260 GOTO 50
    270 CLS
    280 PRINT "YOUR GHOST GUZZLING"
    290 PRINT "SCORE IS ";S
    300 PRINT
    310 PRINT "ANOTHER GO ?
        (PRESS Y OR N)"
    320 INPUT A$
    330 IF A$="Y" THEN RUN
    340 IF A$="N" THEN STOP
```

Looks to see if you are pressing a key. If X (ie you activated ghost guzzler) computer jumps down program to check if you are allowed to. If M it increases your number by 1.

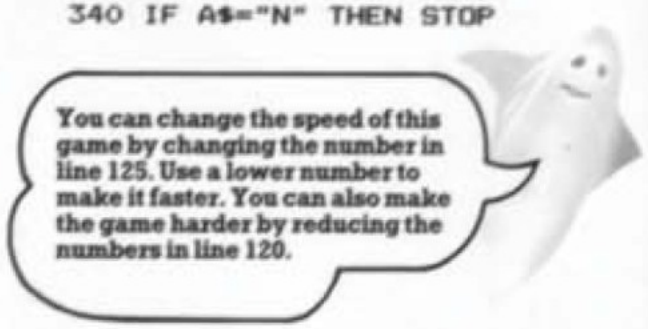
Sends your number back to zero when it reaches 10.

Increases ghost-distance counter. If ghost hasn't reached barrier and you haven't guzzled it, computer goes back to print ghost in next position.

If barrier reached, you lose a life.

Goes back for another ghost if you still have lives left.

Checks if your number equalled ghost's when you guzzled. If not goes back for another ghost.



You can change the speed of this game by changing the number in line 125. Use a lower number to make it faster. You can also make the game harder by reducing the numbers in line 120.

The above listing will work on a ZX81. For other computers make the changes below.

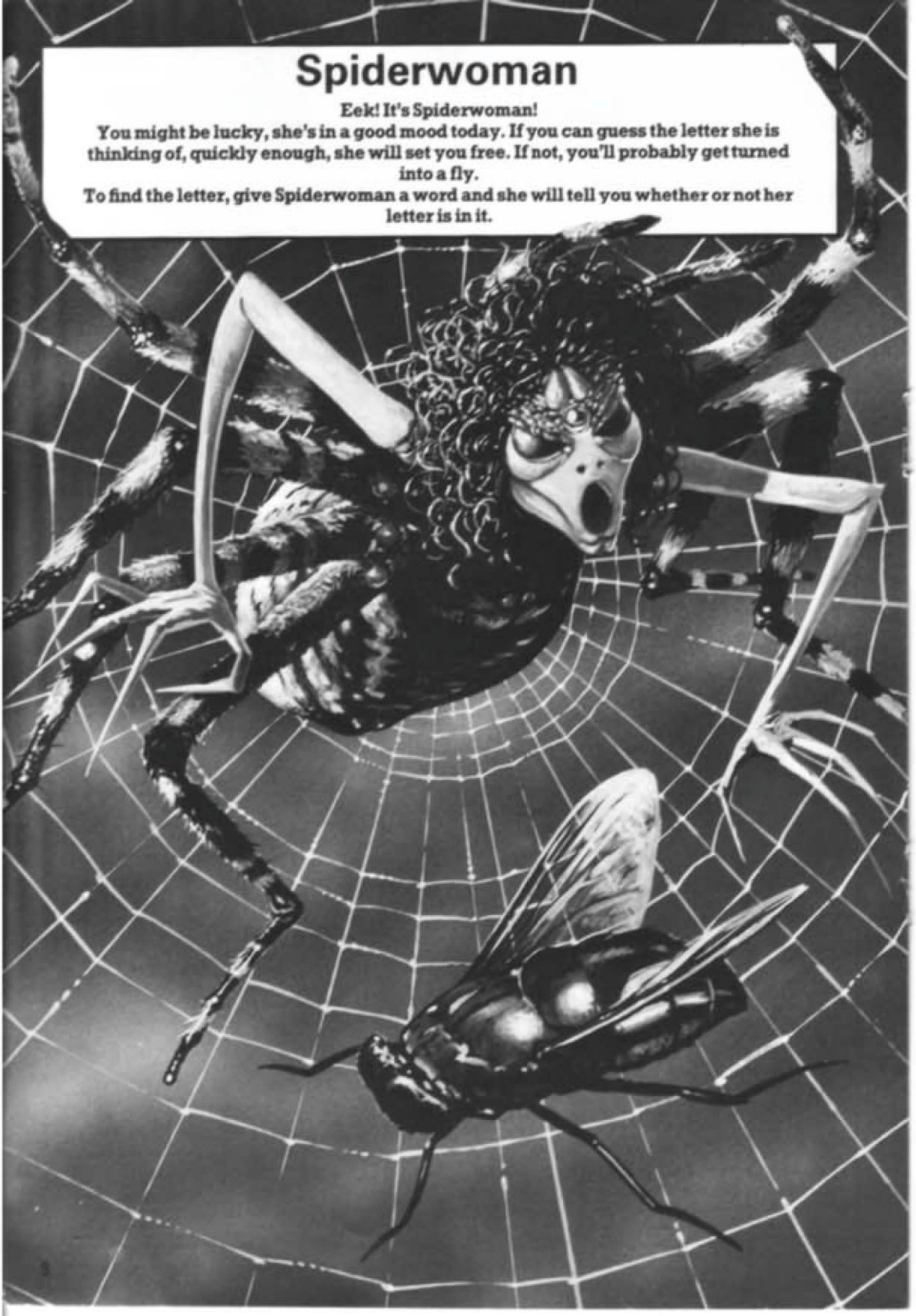
```
    50 LET N=INT(RND(1)*10)
    50 LET N=INT(RND(0)*10)
    70,270 HOME
    70,270 PRINT CHR$(147)
    120 PRINT SPC(I);N;SPC
        (18-I);":";Y
    125 FOR K=1 TO 50
    127 NEXT K
    128 LET B$=""
    130 IF PEEK(-16384)>127
        THEN GET B$
    130 LET B$=INKEY$(0)
    130 GET B$
    130 LET B$=KEY$
    245 FOR K=1 TO 300
    247 NEXT K
```

Spiderwoman

Eek! It's Spiderwoman!

You might be lucky, she's in a good mood today. If you can guess the letter she is thinking of, quickly enough, she will set you free. If not, you'll probably get turned into a fly.

To find the letter, give Spiderwoman a word and she will tell you whether or not her letter is in it.



```
  10 LET G=0
▲●20 CLS

○▽■▲●30 LET T=INT(RND*26+38)
✱!    40 LET T$=CHR$(T)
      50 PRINT "SPIDERWOMAN"
      60 PRINT "HAS CHOSEN"
      70 PRINT
      80 PRINT "TRY A WORD"
      90 PRINT
     100 PRINT
     110 INPUT W$
     120 LET G=G+1
     130 LET L=LEN(W$)
     140 IF L<4 OR L>8 THEN GOTO 70
     150 LET F=0
     160 FOR I=1 TO L
○▽■▲●170 LET A$=W$(I TO I)
✱    180 IF A$=T$ THEN LET F=1
     190 NEXT I
     200 IF F=1 THEN GOTO 280
     210 PRINT
     220 PRINT "IT'S NOT IN THAT WORD"
○▽■▲●230 FOR A=1 TO 20
✱    240 NEXT A
     250 IF G>15 THEN GOTO 400
  ▲●260 CLS
     270 GOTO 70
     280 PRINT "YES - IT'S ONE OF THOSE"
     290 PRINT
     300 PRINT "DO YOU WANT TO GUESS ?
         (Y OR N)"
     310 INPUT R$
     320 IF R$="N" THEN GOTO 250
     330 PRINT
     340 PRINT "WHAT IS YOUR GUESS
         THEN ? "
     350 INPUT G$
     360 IF G$<>T$ THEN GOTO 410
     370 PRINT "OK - YOU CAN GO"
     380 PRINT "(THIS TIME)"
     390 STOP
     400 PRINT "YOU ARE TOO LATE"
     410 PRINT "YOU ARE NOW A FLY"
     420 STOP
```

How it works

Chooses a number, converts it to a letter and puts it in T$.

Asks for a word.

Keeps count of number of goes.

Checks if word too long or short.

Checks through your word to see if letter is there. If it is goes to 280.

Prints message if letter not found in word.

Delay loop so you can read message.

Ends program if you have had 15 goes.

Clears screen. Goes back for another word.

Prints if letter found in your word. Asks if you want to guess. If "no", goes back for another word.

Prints message depending on whether your guess was right or not.

The above listing will work on a ZX81. For other computers make the changes below.

```
  ●20 HOME
  ▲20 PRINT CHR$(147)
   !30 LET T=INT(RND*26+65)
○✱▲●30 LET T=INT(RND(1)*26+65)
  ▽■30 LET T=INT(RND(0)*26+65)
■▲●  170 LET A$=MID$(W$,I,1)
✱○▽  230 FOR A=1 TO 500
  ●260 HOME
  ▲260 PRINT CHR$(147)
```

Experiments to try

See what happens if you change the 15 in line 250 to a smaller number. What about changing the 4 and the 8 in line 140?

Change the game

Spiderwoman is bored. She wants you to adjust the program to do the following:

If her letter is in your word, you are allowed *two* guesses at it, but if you guess wrong you forfeit five goes. Can you do it?

Gravedigger

It's dark and windy – not the kind of night to be lost in a graveyard, but that's where you are. You have until midnight to find your way out. Skeletons lurk in the shadows waiting to scare you to death should you come too close. You can dig holes to help keep them away but digging is tiring work and you cannot manage more than five in one game. You have to be careful not to fall down the holes you have dug too.

Grave stones (marked +) and the walls of the graveyard (marked :) block your way. The holes you dig are marked 0, you are * and the skeletons are X. See if you can escape.

How the program works

50–90	Fill the array A with empty spaces.
110	Sets the number of holes allowed.
120–170	Code the symbols to be used.
180–260	Put the walls into the array.
270–310	Choose random positions for the gravestones.
320–390	Set starting positions for you and the skeletons.
400–420	Put skeletons into the array.
440–640	Calculates your new position.
650–740	Looks to see what is in new position. If nothing, moves to it.
770–850	Print various ends to the game.
860–980	Print graveyard.
1030–1100	Put a hole in the array if you want one.
1100–1310	Move skeletons.

```
10 CLS
20 PRINT "GRAVE DIGGER"
30 DIM A(10,20)
40 DIM B(6)
50 FOR I=1 TO 10
60 FOR J=1 TO 20
70 LET A(I,J)=CODE(" ")
80 NEXT J
90 NEXT I
100 LET W=0
110 LET X=5
120 LET Y=CODE("*")
130 LET B=CODE("+")
140 LET C=CODE("O")
150 LET D=CODE(":")
160 LET E=CODE("X")
170 LET Z=CODE(" ")
180 FOR J=1 TO 10
190 LET A(J,1)=D
200 LET A(J,20)=D
210 NEXT J
220 FOR J=1 TO 20
230 LET A(1,J)=D
240 LET A(10,J)=D
250 NEXT J
260 LET A(9,20)=Z
270 FOR J=1 TO 20
280 LET F=INT(RND*7+2)
290 LET G=INT(RND*15+3)
300 LET A(F,G)=B
310 NEXT J
320 LET M=2
330 LET N=2
340 LET B(2)=19
350 LET B(4)=19
360 LET B(6)=19
370 LET B(1)=4
380 LET B(3)=3
390 LET B(5)=2
400 FOR J=1 TO 5 STEP 2
410 LET A(B(J),B(J+1))=E
420 NEXT J
430 GOSUB 860
440 PRINT
450 PRINT "ENTER MOVE ";W
460 PRINT "(YOU CAN GO N,S,
    E OR W)"
470 INPUT A$
480 IF A$="N" THEN GOTO 530
490 IF A$="E" THEN GOTO 560
500 IF A$="S" THEN GOTO 600
```



```
510 IF A$="W" THEN GOTO 630
520 GOTO 470
530 LET T=N-1
540 LET U=M
550 GOTO 650
560 LET T=N
570 LET U=M+1
580 IF A(T,U)=Z AND M=19 THEN
    GOTO 770
590 GOTO 650
600 LET T=N+1
610 LET U=M
620 GOTO 650
630 LET T=N
640 LET U=M-1
650 IF A(T,U)=Z THEN GOTO 710
660 IF A(T,U)=D OR A(T,U)=B
    THEN GOTO 690
670 IF A(T,U)=C THEN GOTO 800
680 IF A(T,U)=E THEN GOTO 830
690 PRINT "THAT WAY'S BLOCKED"
700 GOTO 440
710 LET A(N,M)=Z
720 LET N=T
730 LET M=U
740 LET A(N,M)=Y
750 GOSUB 1030
760 GOTO 400
770 PRINT "YOU'RE FREE**"
780 PRINT "YOUR PERFORMANCE
    RATING IS ";INT((60-W)/60*
    (96+X));"%"
790 STOP
800 PRINT "YOU'VE FALLEN INTO"
810 PRINT "ONE OF YOUR OWN
    HOLES"
820 STOP
830 PRINT "URK! YOU'VE BEEN
    SCARED"
840 PRINT "TO DEATH BY A
    SKELETON"
850 STOP
860 LET A(N,M)=Y
▲●870 CLS
880 LET R=N
890 LET S=M
900 LET W=W+1
910 IF W>60 THEN GOTO 990
920 FOR I=1 TO 10
930 FOR J=1 TO 20
940 PRINT CHR$(A(I,J));
950 NEXT J
960 PRINT
970 NEXT I
980 RETURN
990 PRINT "THE CLOCK'S STRUCK"
1000 PRINT "MIDNIGHT"
1010 PRINT "AGHHHHH!!!!"
1020 STOP
1030 IF X=0 THEN GOTO 1110
1040 PRINT
1050 PRINT "WOULD YOU LIKE
    TO DIG"
1060 PRINT "A HOLE (Y OR N)"
1070 INPUT B$
1080 IF B$<>"Y" THEN GOTO 1110
1090 LET X=X-1
1100 LET A(R,S)=C
1110 FOR J=1 TO 5 STEP 2
1120 LET T=B(J)
1130 LET U=B(J+1)
1140 IF A(T,U+1)=Y THEN GOTO 830
1150 IF A(T,U-1)=Y THEN GOTO 830
1160 IF A(T-1,U)=Y THEN GOTO 830
1170 IF A(T+1,U)=Y THEN GOTO 830
1180 IF A$="W" THEN GOTO 1300
1190 IF A$="S" AND A(T+1,U)=Z
    THEN GOTO 1270
1200 IF A$="N" AND A(T-1,U)=Z
    THEN GOTO 1280
1210 IF A$="E" AND A(T,U+1)=Z
    AND M>U THEN GOTO 1240
1220 IF A$="E" AND A(T,U-1)=Z
    THEN GOTO 1250
1230 GOTO 1300
1240 LET B(J+1)=B(J+1)+2
1250 LET B(J+1)=B(J+1)-1
1260 GOTO 1290
1270 LET B(J)=B(J)+2
1280 LET B(J)=B(J)-1
1290 LET A(T,U)=Z
1300 NEXT J
1310 RETURN
```

The above listing will work on a ZX81. For other computers make the changes below.

```
●10,870 HOME
▲10,870 PRINT CHR$(147)
✱○▽■▲●70,120-170 CHANGE CODE TO ASC
✱○▲●280,290 CHANGE RND TO RND(1)
▽■280,290 CHANGE RND TO RND(0)
```



Mad House

You're trapped in a weird house where everything moves including the walls. If the doorways would line up, even for a moment, you could make a dash for freedom. You've found a console which appears to control some of the movements in the house. Keys X and C make the doorway nearest to you (top of the screen) change direction. Keys N and M have the same effect on the doorway furthest from you. There doesn't seem to be any way of controlling the centre doorway.

As you frantically wrestle with the knobs on the console, you can hear footsteps pounding down the corridor behind you. The number top left of the screen shows their progress towards you. If you can't escape before the phantom footsteps catch up with you...

...aaggghhhhh!

If your computer has sound, see if you can add sound effects for the footsteps and for the end of the game if you fail to escape.

```
 10 DIM P(3)
 20 DIM S(3)
 30 CLS
 40 PRINT "MAD HOUSE"
 50 LET L$=""
 60 LET W=31
 70 FOR I=1 TO W
 80 LET L$=L$+"*"
 90 NEXT I
100 LET CT=240
110 LET S(1)=1
120 LET S(3)=-1
130 FOR I=1 TO 3
140 LET P(I)=INT(RND*(W-4)+1)
150 NEXT I
160 CLS
170 FOR I=5 TO 15 STEP 5
180 PRINT AT I,0;L$
190 PRINT AT I,P(I/5);"> <"
200 NEXT I
210 PRINT AT 0,0;CT;"  "
220 IF CT<0 THEN GOTO 450
230 IF P(1)=P(2) AND P(2)=P(3)
    THEN GOTO 390
240 LET Z$=INKEY$
250 IF Z$="" THEN GOTO 300
260 IF Z$="X" THEN LET S(1)=-1
270 IF Z$="C" THEN LET S(1)=1
280 IF Z$="N" THEN LET S(3)=-1
290 IF Z$="M" THEN LET S(3)=1
300 IF CT/25=INT(CT/25) THEN LET
    P(2)=P(2)+INT(RND*20)-10
310 LET CT=CT-1
320 LET P(1)=P(1)+S(1)
330 LET P(3)=P(3)+S(3)
340 FOR I=1 TO 3
350 IF P(I)<1 THEN LET P(I)=1
360 IF P(I)>(W-4) THEN LET P(I)=W-4
370 NEXT I
380 GOTO 170
390 LET L=P(1)+1
400 FOR I=1 TO 15
410 PRINT AT I,L;"M"
420 NEXT I
430 PRINT "YOU ARE FREE !!"
440 STOP
450 PRINT "TOO LATE!  THE FOOT STEPS
    HAVE STOPPED."
460 PRINT "ARGHHH!!!"
470 STOP
```

How it works

Sets up 2 arrays. One is for the position (P) of each doorway, the other if for the direction (S) in which each is moving.

Make a row of stars the width (W) of the screen and puts it in L$.

Sets footstep counter to its starting value.

Sets direction of doorway movement at start of game.

Gets random starting positions for the three doorways.

Prints walls and doorways.

Prints footstep counter top left.

Checks if footsteps have reached you. If so, jumps down to end game.

Checks if you've won.

Checks keyboard for instructions. Changes direction of movement, if necessary, by changing + and − signs.

Moves centre doorway every 25 footsteps.

Decreases footstep count.

Finds new doorway positions. If doorway reaches edge of screen, computer keeps it there.

Goes back for next move.

Prints your dash through doorways with message.

Prints losing message.



The listing on the left will work on a ZX81. For other computers make these changes.

○ ORIC

```
140 LET P(I)=INT(RND(1)*(W-4)+1)
180 PLOT 1,I,L$
190 PLOT P(I/5),I,"> <"
210 PLOT 0,0,STR$(CT)+"  "
240 LET Z$=KEY$
300 IF CT/25=INT(CT/25) THEN LET
410 PLOT L,I,"M"
    P(2)=P(2)+INT(RND(1)*20)-10
```

✱ BBC

```
60 LET W=40
140 LET P(I)=INT(RND(1)*(W-4))+1
180 PRINT TAB(0,I);L$
190 PRINT TAB(P(I/5),I);"> <"
210 PRINT TAB(0,0);CT;"  "
240 LET Z$=INKEY$(10)
300 IF CT/25=INT(CT/25) THEN LET
    P(2)=P(2)+INT(RND(1)*20)-10
410 PRINT TAB(L,I);"M"
```

▲ VIC-20

```
30 PRINT CHR$(147)
60 LET W=21
140 LET P(I)=INT(RND(1)*(W-4))+1
160 PRINT CHR$(147)
170 FOR I=1 TO 3
172 FOR J=1 TO 5
174 PRINT
180 NEXT J
190 PRINT LEFT$(L$,P(I));"> <";
    RIGHT$(L$,(18-P(I)))
210 PRINT CHR$(19);CT;"  "
240 GET Z$
300 IF CT/25=INT(CT/25) THEN LET
    P(2)=P(2)+INT(RND(1)*20)-10
375 PRINT CHR$(19)
395 PRINT CHR$(19)
410 PRINT TAB(L)"M"
```

● APPLE

```
30 HOME
140 LET P(I)=INT(RND(1)*(W-4))+1
160 HOME
180 VTAB(I)
185 PRINT L$
190 VTAB(I)
195 HTAB(P(I/5))
197 PRINT "> <"
210 VTAB(1):HTAB(1):PRINT CT;"  "
240 Z$=""
245 IF PEEK(-16384)>127 THEN GET Z$
300 IF CT/25=INT(CT/25) THEN LET
    P(2)=P(2)+INT(RND(1)*20)-10
395 VTAB(1)
410 PRINT TAB(L)"M"
```

■ TRS-80

```
60 LET W=63
140 LET P(I)=RND(60)
170 FOR I=2 TO 12 STEP 5
180 PRINT@(I*64),L$
190 PRINT@(I*64:P((I+3)/5)),"> <"
210 PRINT@0,CT
300 IF CT/25=INT(CT/25) THEN LET
    P(2)=P(2)+RND(20)-10
410 PRINT@(I*64+L),"M"
```

▽ DRAGON

```
50,70,80,90 DELETE
140 LET P(I)=RND(28)
170 FOR I=2 TO 12 STEP 5
180 PRINT@32*I,STRING$(P((I+3)/5)-1,
    "*");"*> <*";STRING$(28-P
    ((I+3)/5),"*");
190 DELETE
210 PRINT@0,CT
300 IF CT/25=INT(CT/25) THEN LET
    P(2)=P(2)+RND(20)-10
410 PRINT@(I*32+L),"M"
```

What happens if you change the number in line 100?

If you take the minus sign out of line 120, what happens?

Change the two 25s in line 300 to another number and see if it makes any difference.

Ghost Maze

It's a creepy sort of place. The identical dark corridors don't seem to go anywhere. It might even be haunted. You can only see straight in front of you, and you can only move in the direction in which you are facing. You can turn left or right, but this won't actually move you anywhere, it will just show you another view. All you have to do is find the cross which marks the way out. Your position is marked with a Y and walls are marked #.

Gulp! It *is* haunted. Ghosts are shown by a letter G. If you get right up next to one you will be whisked away to another part of the maze, not knowing where you are or in which direction you are facing. Here are the keys you can use: X moves you forward, N turns you to the left (through 90°), M turns you to the right (through 90°)

```
10 DIM E(70)
20 DIM V$(4,3)
30 DIM F(3)
40 LET W$=""
50 LET W$=W$+"0000000000"
60 LET W$=W$+"0111100110"
70 LET W$=W$+"0010011100"
80 LET W$=W$+"0011010110"
90 LET W$=W$+"0110100100"
100 LET W$=W$+"0011111100"
110 LET W$=W$+"0000009000"
120 FOR I=1 TO 70
130 LET E(I)=VAL(W$(I TO I))
140 NEXT I
150 LET S=-1
160 LET G=12
170 LET X=INT(RND*50)+10
180 IF E(X)<>1 THEN GOTO 170
190 GOSUB 860
200 LET D=INT(RND*4)+1
210 IF X=G+10 OR X=G-10 THEN GOTO 170
220 IF X=G+1 OR X=G-1 THEN GOTO 170
230 LET H=H+1
240 IF H=5 THEN GOSUB 860
250 GOSUB 390
260 LET A$=INKEY$
270 IF A$="" THEN GOTO 260
280 IF A$="M" THEN LET D=D+1
290 IF A$="N" THEN LET D=D-1
300 IF D=5 THEN LET D=1
310 IF D=0 THEN LET D=4
320 IF A$<>"X" THEN GOTO 210
330 IF D=1 AND E(X-10)<>0 THEN LET X=X-10
340 IF D=3 AND E(X+10)<>0 THEN LET X=X+10
350 IF D=2 AND E(X+1)<>0 THEN LET X=X+1
360 IF D=4 AND E(X-1)<>0 THEN LET X=X-1
370 IF E(X)=9 THEN GOTO 930
380 GOTO 210
```

Sets up a storage area (array) for the maze.

Sets up a mini-grid within the maze, which is the part of it that you can see on the screen.

Storage space for data to fill one row of this mini-grid.

Contains the data for the maze 1 = corridor; 0 = wall; 9 = exit.

Loop which puts the data in the array, E.

Sets number of times the ghost has moved to −1 for start.

Sets number of ghosts to zero for start.

Chooses random starting position for you.

If this position is not in a corridor, goes back for another.

Goes to sub-routine to position the ghost.

Chooses a direction for you to face at start of game.

Checks if you are next to a ghost and if so moves you to new place.

Increases number of moves you've made.

Goes to sub-routine to move ghost every 5 moves.

Goes to sub-routine which prints out section of maze you can see.

Gets an instruction from you. Changes value of direction, D, if necessary.

Goes back for next move if you don't move forward.

Checks you can move.

Checks if you have reached exit.

Goes back for next move if you haven't.

The listing on the left will work on a ZX81. For other computers make the changes below.

```
▽■5 CLEAR
✱○▽■▲●130 LET E(I)=VAL(MID$(W$,I,1))
✱○▲●170 LET X=INT(RND(1)*50)+10
▽■170 LET X=INT(RND(0)*50)+10
✱○▲●200 LET D=INT(RND(1)*4)+1
▽■200 LET D=INT(RND(0)*4)+1
●255 LET A$=""
●260 IF PEEK(-16384)>127
     THEN GET A$
✱260 LET A$=INKEY$(0)
▲260 GET A$
○260 LET A$=KEY$
✱○▽■▲●410 ON D GOTO 420,460,500,540
●660 HOME
▲660 PRINT CHR$(147)
✱○▲●870 LET G=INT(RND(1)*50)+10
▽■870 LET G=INT(RND(0)*50)+10
```

Add noises

If your computer has sound, see if you can add a noise every time a ghost appears.

Add better symbols

Try replacing the symbols +, G, Y and # with graphics characters (and colours too). This will involve changes to lines 590-620 and 840.

```
390 FOR I=1 TO 4
400 LET T=I-1
✱○▽■▲●410 GOTO 380+40*D
420 LET F(1)=X-10*T+1
430 LET F(2)=X-10*T
440 LET F(3)=X-10*T-1
450 GOTO 570
460 LET F(1)=X+10+T
470 LET F(2)=X+T
480 LET F(3)=X-10+T
490 GOTO 570
500 LET F(1)=X+10*T-1
510 LET F(2)=X+10*T
520 LET F(3)=X+10*T+1
530 GOTO 570
540 LET F(1)=X-T-10
550 LET F(2)=X-T
560 LET F(3)=X-T+10
570 FOR J=1 TO 3
580 IF F(J)<1 OR F(J)>69 THEN GOTO 840
590 IF E(F(J))=0 THEN LET V$(I,J)="#"
600 IF E(F(J))=1 THEN LET V$(I,J)=" "
610 IF E(F(J))=9 THEN LET V$(I,J)="+"
620 IF E(F(J))=2 THEN LET V$(I,J)="G"
630 NEXT J
640 NEXT I
650 LET V$(1,2)="Y"
▲●660 CLS
670 PRINT
680 PRINT"**** GHOST MAZE ****"
690 PRINT
700 PRINT "FORWARD          X"
710 PRINT "TURN RIGHT     M"
720 PRINT "TURN LEFT      N"
730 FOR I=1 TO 5
740 PRINT
750 NEXT I
760 FOR I=4 TO 1 STEP-1
770 LET P$=""
780 FOR J=3 TO 1 STEP -1
790 LET P$=P$+V$(I,J)
800 NEXT J
810 PRINT TAB(7);P$
820 NEXT I
830 RETURN
840 LET V$(I,J)="#"
850 GOTO 630
860 LET E(G)=1
✱○▽■▲●870 LET G=INT(RND*50)+10
880 IF E(G)<>1 THEN GOTO 870
890 LET E(G)=2
900 LET H=0
910 LET S=S+1
920 RETURN
930 PRINT "YOU HAVE ESCAPED"
940 PRINT "IN ";S*5+H;" MOVES"
950 STOP
```

Sub-routine which calculates:
a) which squares you can see from where you are.
b) which characters to put in these squares.
and c) prints out instructions and the section of grid you can see.

Make the maze bigger

This maze is based on a grid containing 70 squares. To make it bigger you need to change the following lines, replacing *N* with the number of squares you want in your new grid. (*N* must be a multiple of 10.)
10: Replace 70 with your number, *N*.
50-110: Add extra lines of 1s and 0s at 55, 65 etc. so that the total number of characters in the grid is *N*.
120: Replace 70 with *N*.
170 and 870: Replace 50 with *N*-20.
580: Replace 69 with *N*-1.

Sub-routine for moving ghost every five goes.

Message for end of game.

Seance

Messages from the Spirits are coming through, letter by letter. They want you to remember the letters and type them into the computer in the correct order. If you make mistakes, they will be angry – very angry . . .

Watch for stars on your screen – they show the letters in the Spirits' messages.

```
10 LET S=0
20 LET G=0
30 LET CS=37
40 CLS
50 PRINT
60 PRINT TAB(8);"SEANCE"
70 FOR I=1 TO 8
80 LET X=6+I
90 LET Y=5
100 LET A$=CHR$(CS+I)
110 GOSUB 710
120 LET Y=11
130 LET A$=CHR$(CS+22-I)
140 GOSUB 710
150 NEXT I
160 FOR I=1 TO 5
170 LET X=5
180 LET Y=5+I
190 LET A$=CHR$(CS+27-I)
200 GOSUB 710
210 LET X=16
220 LET A$=CHR$(CS+8+I)
230 GOSUB 710
240 NEXT I
250 LET P$=""
260 LET N=INT(RND*4+3)
270 FOR I=1 TO N
280 LET A$="*"
290 LET L=INT(RND*26+1)
300 LET S$=CHR$(CS+L)
310 LET P$=P$+S$
320 LET D=4
330 IF L<22 THEN LET D=3
340 IF L<14 THEN LET D=2
350 IF L<9 THEN LET D=1
360 GOTO 340+30*D
370 LET Y=6
380 LET X=L+6
390 GOTO 480
400 LET X=15
410 LET Y=L-3
420 GOTO 480
430 LET Y=10
440 LET X=28-L
450 GOTO 480
460 LET X=6
470 LET Y=32-L
480 GOSUB 710
490 FOR T=1 TO 20
500 NEXT T
510 LET A$=" "
520 GOSUB 710
530 NEXT I
540 LET A$=""
550 LET X=0
560 LET Y=13
570 GOSUB 710
580 INPUT R$
590 IF R$=P$ THEN GOTO 670
600 LET G=G+1
610 IF G=1 THEN PRINT "THE TABLE
    BEGINS TO SHAKE"
620 IF G=2 THEN PRINT "THE LIGHT
    BULB SHATTERS"
630 IF G=3 THEN GOTO 730
640 FOR T=1 TO 100
650 NEXT T
660 GOTO 40
670 LET S=S+N
680 IF S<50 THEN GOTO 40
690 PRINT "THE SPIRITS HAVE GONE"
700 STOP
710 PRINT AT Y,X;A$;
720 RETURN
730 PRINT "A PAIR OF CLAMMY HANDS
    GRASPS YOUR NECK!"
740 STOP
```

How the program works

After you have typed in the program and played it a few times, see if you can work through the program lines and see what they all do. (Hint: D is a variable which tells the computer on which side of the square – 1, 2, 3 or 4 – the next letter in the message is.)

Line 490 controls the speed at which the stars appear on the screen. Change it to a lower number to speed up the game.

The above listing will work on a ZX81. For other computers make the changes below.

```
30 LET CS=64
40 HOME
40 PRINT CHR$(147)
260 LET N=INT(RND(1)*4+3)
260 LET N=INT(RND(0)*4+3)
290 LET L=INT(RND(1)*26+1)
290 LET L=INT(RND(0)*26+1)
360 ON D GOTO 370,400,430,460
490 FOR T=1 TO 100
490 FOR T=1 TO 600
640 FOR T=1 TO 300
640 FOR T=1 TO 1000
710 PRINT TAB(X,Y);A$;
710 VTAB(Y):HTAB(X+1):PRINT A$
710 PRINT CHR$(19);:FOR K=1 TO Y:PRINT:
    NEXT:PRINT TAB(X);A$
710 PRINT@32*Y+X,A$;
710 PRINT@64*Y+2*X,A$;
710 PLOT X,Y,A$
```

Answers to puzzles

Here are some suggested solutions to the puzzles set in this book. You may find that your answers are different but if they work on your computer this doesn't matter. Check, though, that your answers are as neat and straightforward as the ones given here.

Computer Nightmare (page 3)

```
!✱○▽■▲●90 LET N$=CHR$(INT(RND*26+38))
      100 PRINT TAB(5);N$
      220 IF F$<>N$ THEN GOTO 240
!✱○▽■▲●230 LET S=S+10+(CODE(N$)-38)

    !90 LET N$=CHR$(INT(RND*26+65))
  ✱○▲●90 LET N$=CHR$(INT(RND(1)*26+65))
   ▽■90 LET N$=CHR$(INT(RND(0)*26+65))
    !230 LET S=S+10+(CODE(N$)-65)
✱○▽■▲●230 LET S=S+10+(ASC(N$)-65)
```

Chooses a letter and prints it onto the screen, instead of a number.

Checks the key pressed against the letter chosen.

Increases score. (This depends on the letter – Z scores more than A for instance.)

Note the different versions for different computers.

Number Wizard (page 5)

```
5 LET L=0
6 LET W=0
355 LET L=L+1
360 GOTO 390
380 LET W=W+1
390 PRINT
400 PRINT "THE WIZARD   YOU"
410 PRINT
420 PRINT "      ";L;"         ";W
430 FOR Q=1 TO 1000
440 NEXT Q
450 GOTO 30
```

Set number of wins and loses at start of game.

Increases number of times you have lost.

Increases number of times you have won.

Prints scoreboard.

Pauses so you can see the score.

Goes back for another game.

Change number here for the faster computers.

Spiderwoman (page 9)

```
5 LET NG=0
355 LET NG=NG+1
360 IF G$=T$ THEN GOTO 370
362 PRINT "NO! THAT'S WRONG"
364 LET G=G+5
366 IF NG=2 THEN GOTO 410
368 GOTO 230
```

Sets number of guesses to zero.

Increases number of guesses.

If guess is correct, stop the game.

Prints message to say guess wrong.

Increases number of goes by 5 as forfeit for wrong guess.

If you have used two guesses, then game finishes.

Returns for next go (via pause so you can see message on screen).

Usborne Computer Books

Usborne Computer Books are colourful, straightforward and easy-to-understand guides to the world of home computing for beginners of all ages.

Usborne Guide to Computers A colourful introduction to the world of computers. *"Without question the best general introduction to computing I have ever seen."* Personal Computer World

Understanding the Micro A beginner's guide to microcomputers, how to use them and how they work. *"This introduction to the subject seems to get everything right."* Guardian

Computer Programming A simple introduction to BASIC for absolute beginners. *"... lucid and entertaining ..."* Guardian

Computer and Video Games All about electronic games and how they work, with expert's tips on how to win. *"The ideal book to convert the arcade games freak to real computing."* Computing Today

Computer Spacegames, Computer Battlegames Listings to run on the ZX81, Spectrum, BBC, TRS-80, Apple, VIC 20 and PET. *"Highly recommended to anyone of any age."* Computing Today

Practical Things to do with a Microcomputer Lots of programs to run and a robot to build which will work with most micros.

Computer Jargon An illustrated guide to all the jargon.

Computer Graphics Superbly illustrated introduction to computer graphics with programs and a graphics conversion chart for most micros.

Write Your Own Adventure Programs Step-by-step guide to writing adventure games programs, with lots of expert's tips.

Machine Code for Beginners A really simple introduction to machine code for the Z80 and 6502.

Better BASIC A beginner's guide to writing programs in BASIC.

Inside the Chip A simple and colourful account of how the chip works and what it can do.

+000·99

ISBN 0-86020-780-3

00099

9 780860 207801

ISBN 0 86020 780 3

10 11 12/83 99p