Here's a collection of 49 games guaranteed to make you glad you invested in a Spectrum.

Tim Hartnell, the world's most widely published computer author (and recently described by Personal Computer World magazine as 'Mr Sinclair') has drawn on the work of some of the UK's most talented young programmers to bring together this incredible collection of explosive games for your Spectrum.

The programs, many of which feature machine code subroutines, include:

PROTECTOR
WIPE-OUT
CHOPPER SQUAD
DODGEMS
SNAKE
ZOMBIES
DOORS OF DOOM (Adventure)
GOLD RUSH
SPACE TREK
3-D MAZE (with full graphics)

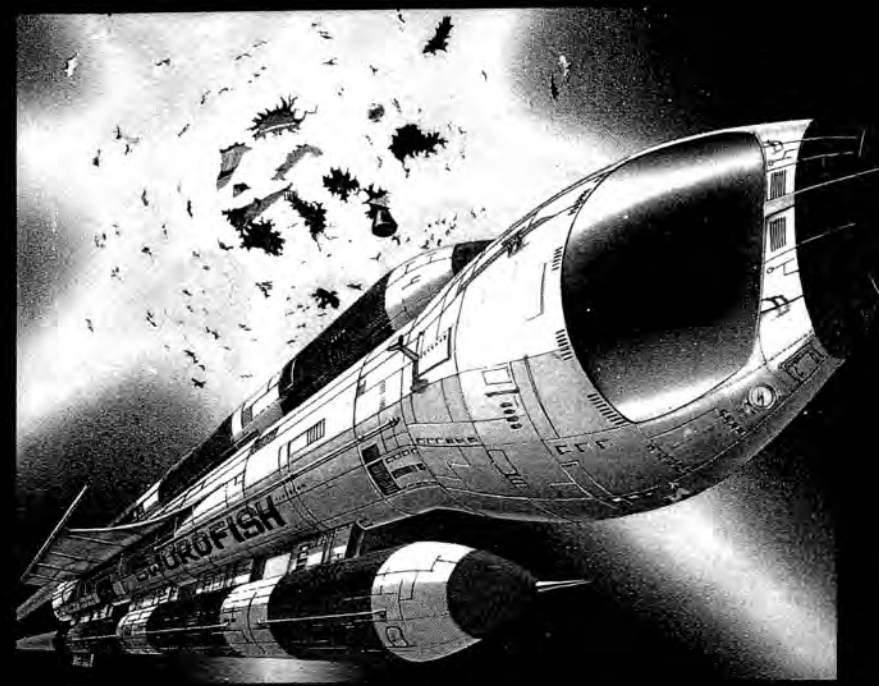
Don't wast a minute longer reading the back of this book in the shop. Buy it now, and rush home, and start your Spectrum exploding.

Another great book from INTERFACE PUBLICATIONS

49 EXPLOSIVE GAMES FOR THE ZX SPECTRUM

TIM HARTNELL

49 Explosive Games for the ZX Spectrum



Tim Hartnell

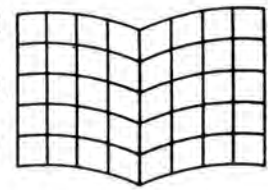
With David Perry, Graham Charlton, Neil Pellinacci and Malcolm Young

49 Explosive Games for the ZX Spectrum

Tim Hartnell

with David Perry, Graham Charlton, Neil Pellinacci and Malcolm Young

INTERFACE PUBLICATIONS



Interface Publications

49 Explosive Games for the ZX Spectrum

Tim Hartnell

Tim Hartnell is the most widely published author of computer books in the world. Founder of the UK's National ZX Users' Club, and founding editor of *ZX Computing* magazine, Tim has been at the forefront of developments of Sinclair computers and their application since the beginning. This book is a sequel to the highly successful work *49 Explosive Games for the ZX81*. Other work by Tim Hartnell include *The Giant Book of Computer Games* (Fontana/Interface, 1983), *Mastering the Timex Sinclair 1000* (Bantam Books, 1983) and *The Book of Listings* (BBC Publications, 1983). Tim is editor of the Virgin Books computer games series, wrote the master text and edited the 'Getting Started' series for Future and acts as computer consultant to Fontana Paperbacks.

First published in the UK by:

Interface Publications,
9–11 Kensington High Street,
London, W8 5NP

First printing: November, 1983

ISBN 0 907563 53 8

Artwork reproduced, with permission, from *Marine Art Clipbook* (Spectre, Peter H. and Putz, George; Van Nostrand Reinhold Company, New York, 1980)

Books published by INTERFACE PUBLICATIONS are distributed in the UK by THE COMPUTER BOOKSHOP (021-706 5511) and in Australia by PITMAN PUBLISHING PTY LTD.

Printed and bound by Short Run Press Ltd., Exeter

Contents

ARCADE GAMES

Protector 3
Hi-Rize 8
Wipe-Out 16
Birds 19
Chopper Squad 31
Snake 37
Dodgems 41
Zombies 47

ADVENTURES

TROLL and THE CITY 59
Doors of Doom 77

MOVING GRAPHICS GAMES

Gold Rush 103
Tarantula 111
Jungle Job 118
Dodge 124
Duel Cabbageway 126
Ghost Chase 128
Frog on a Log 132

LEISURE LINES

Card Pairs 141
Sub Search 145
Hangman 155

SPACE GAMES

Stellar Evade 161
Space Trek 162
Stellar Probe 170
Lunar Storm 178

THE LURE OF THE MAZE

Maze-maker 185
Mangled Mazes 187
Walls of Suspense 191
Wall of Suspense 193
Dual Level 3-D Maze 194
Scrolling Maze 196
Rollermaze 202
Three-D Maze 212
Three-D Maze II 219

TWO PLAYER GAMES

Squares 227
Four in a Row 236
Tanx 239

MACHINE CODE UTILITIES

Memory Monitor 245
Machine Code Screensave 250
BASIC Screensave 251
Ireland 252
Master Copier 255

UTILITIES AND DEMONSTRATIONS

Biorhythms 261
Paint Pot 263
Timothy Leary 264
Color Test 266
Logic Gate Emulator 267
Aural Assault 269
Rainbird 270
Line of Best Fit 273
Hall of Fame 276
Spec File 278

STRUCTURED PROGRAMMING

Sketching an outline 287

FOREWORD

Books as solid as this one don't just spring out of the Spectrum without a serious investment of time and effort.

And there has been plenty of both time and effort spent making sure this collection of games for your Spectrum is as good as we could make it.

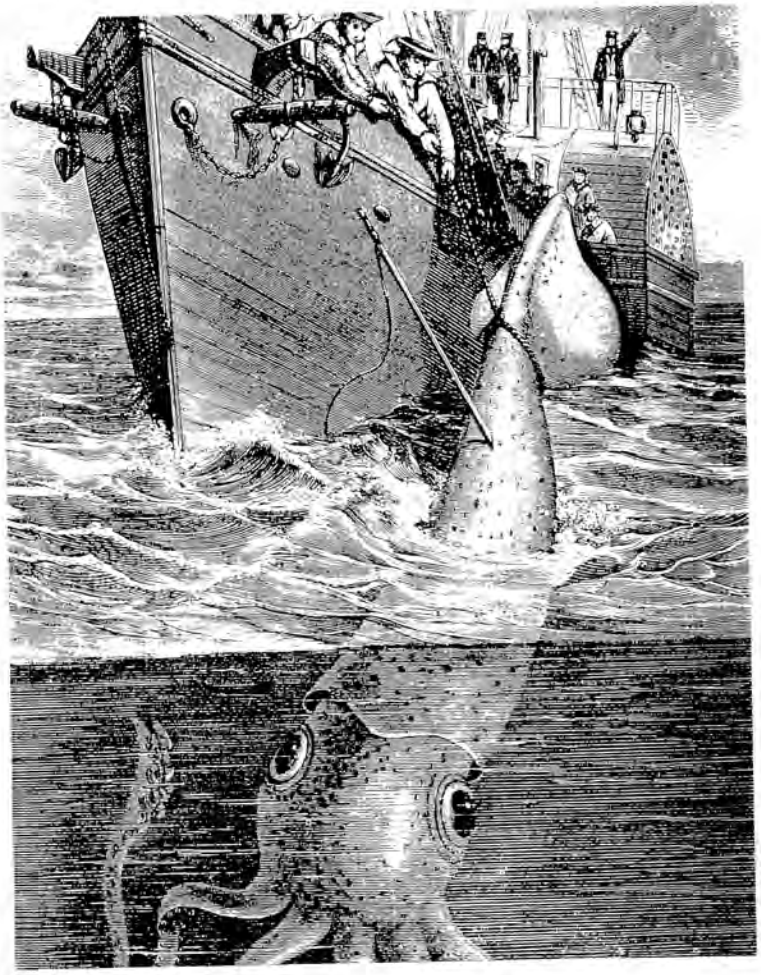
Drawing on the work of some of the UK's best young computer programmers, I've brought together a series of major programs for the Spectrum which deserve the label 'software quality'. The major contributors were Malcolm Young, David Perry, Graham Charlton and Neil Pellinacci, whose work forms the bulk of this book.

Others I contracted to produce programs for this collection include Scott Vincent, Raymond Blake, Neal Cavalier-Smith, Paul Toland, Andrew Sweetland, Martin Jones, Graham White, Tim Rogers, Damian Steele, Neville Predebon and Michael Briggs. They also deserve congratulations for the quality and 'play-ability' of the programs they've contributed.

Time to start your Spectrum exploding.

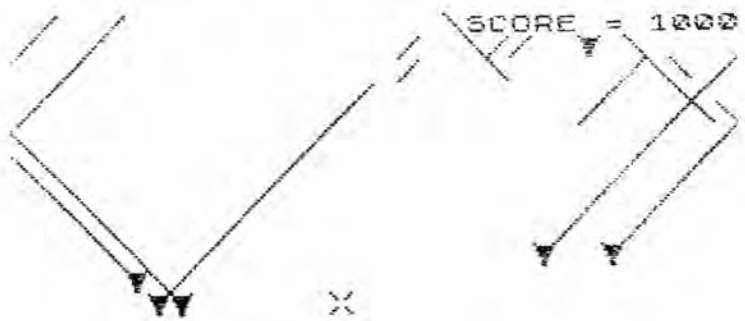
Tim Hartnell,
London, 1983.

ARCADE GAMES



Protector

No need now to spend your money in the arcade. PROTECTOR gives you the chance to save the world (or at least a few cities) as this sample run shows:



Your controls are "M", "Q", "A", "O" and "P". Time to start protecting...

```
 10 REM RUN WITH CAPS LOCK ON
 20 CLS
 30 FOR F=1 TO 49
 40 BORDER F/7
 50 BEEP 0.05,F
 60 PRINT AT 5,10; INK 0; PAPER
4; FLASH 1;"PROTECTOR"
 70 NEXT F
 80 LET SC=0
 90 LET HS=0
100 RESTORE
```

```
 110 LET N=0
 120 PAPER 1: BORDER 1: INK 6: C
LS
 130 LET Q$=" ": REM G ABC
 140 LET R$=Q$
 150 LET W$=+" ": REM G DEBG
 160 LET E$=" ": REM G 36 666
 170 FOR X=1 TO 12
 180 READ A$
 190 FOR Y=0 TO 7
 200 READ A
 210 POKE USR A$+Y,A
 220 NEXT Y: NEXT X
 230 DIM Z$(6,2,4): DIM L(15,2):
 DIM N(15): DIM X$(15)
 240 DATA "A",28,20,28,28,20,20,
60,126,"B",16,16,56,56,56,56,56,
124,"C",128,128,128,192,192,224,
224,240,"D",7,7,15,15,31,63,63,6
3,"E",254,254,254,255,255,255,25
5,255,"F",255,255,231,195,195,23
1,255,255,"G",240,248,248,252,25
2,254,254,255,"H",48,48,48,112,1
20,120,120,124
 250 DATA "^",129,66,36,0,0,36,6
6,129,"▼",129,126,126,60,60,24,2
4,24,"\",128,64,32,16,8,4,2,1,"/
",1,2,4,8,16,32,64,128
 260 FOR B=1 TO 6: LET Z$(B,1)=Q
$: LET Z$(B,2)=W$: NEXT B
 270 CLS
 280 LET V=10: LET D=V: LET H=1:
 LET W=6: GO SUB 0970: GO SUB 09
20
 290 FOR B=1 TO V: LET L(B,1)=IN
T (RND*H): LET L(B,2)=INT (RND*3
0): LET N(B)=INT (RND*2)*2-1: LE
T X$(B)=CHR$ (156.5-N(B)/2): NEX
T B: GO SUB 0950
 300 LET A1=17: LET A2=15
 310 LET B=B+1: IF B>D THEN LET
B=1
 320 PRINT AT 0,20; INK 0;"SCORE
= ";SC
 330 PRINT AT L(B,1),L(B,2);X$(B
): LET L(B,1)=L(B,1)+1: LET L(B,
2)=L(B,2)+N(B): IF L(B,2)>31 THE
N LET L(B,2)=0
 340 IF L(B,2)<0 THEN LET L(B,2)
=31
 350 IF SCREEN$ (L(B,1),L(B,2))<
>" " THEN GO TO 0710
```

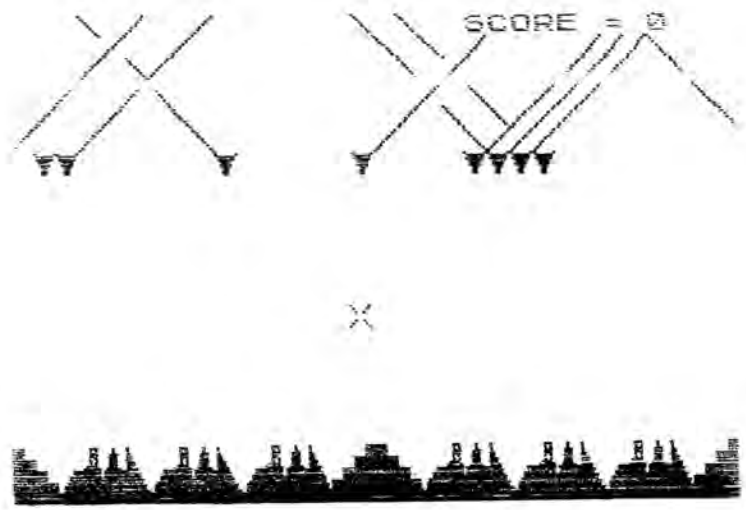


```
 360 PRINT AT L(B,1),L(B,2);"▼"
 370 LET L1=0: LET L2=0
 380 IF INKEY$="" THEN GO TO 050
0
 390 IF INKEY$="M" THEN GO TO 05
10
 400 IF INKEY$="Q" THEN LET L1=-
1
 410 IF INKEY$="A" THEN LET L1=1
 420 IF INKEY$="O" THEN LET L2=-
1
 430 IF INKEY$="P" THEN LET L2=1
 440 PRINT AT A1,A2;" "
 450 LET A1=A1+L1: LET A2=A2+L2
 460 IF A1<1 THEN LET A1=1
 470 IF A1>17 THEN LET A1=17
 480 IF A2<1 THEN LET A2=30
 490 IF A2>30 THEN LET A2=1
 500 PRINT AT A1,A2;"X": GO TO 0
310
 510 PRINT INK 2;AT A1-1,A2-1;"
 ";AT A1,A2-1;" ";AT A1+1,A2-
1;" "
 520 FOR Y=1 TO D: IF ABS (L(Y,1
)-A1)>1 OR ABS (L(Y,2)-A2)>1 THE
N GO TO 0540
 530 GO TO 0570
 540 NEXT Y
 550 PRINT AT A1-1,A2-1;"    ";AT
 A1,A2-1;"    ";AT A1+1,A2-1;"
"
 560 GO TO 0310
 570 LET L(Y,1)=L(D,1): LET L(Y,
2)=L(D,2): LET N(Y)=N(D): LET X$
(Y)=X$(D): LET D=D-1: LET SC=SC+
10
 580 BEEP .15,16
 590 IF D>0 THEN GO TO 0540
 600 LET H=H+1
 610 PAUSE 200
 620 CLS : PRINT AT 5,5; FLASH 1
;"YOUR SCORE IS ";SC: PAUSE 100
 630 LET W=0: FOR X=1 TO 6: IF Z
$(X,2)<>E$ THEN LET W=W+1
 640 NEXT X
 650 IF W=0 THEN GO TO 0850
 660 PRINT AT 10,0;: FOR X=1 TO
W: PRINT AT 10,4*(X-1);R$;AT 11,
4*(X-1);W$: NEXT X
 670 LET T=W*100: PRINT AT 11,(W
)*4;"= BONUS ";T: LET SC=SC+T
```

```
 680 GO SUB 0970
 690 GO SUB 0920: LET V=V+1: IF
V>15 THEN LET V=15
 700 LET D=V: GO TO 0290
 710 IF L(B,1)<18 THEN GO TO 035
0
 720 IF SCREEN$ (L(B,1),L(B,2))<
>"" THEN GO TO 0800
 730 LET Z=L(B,2)
 740 IF Z>1 AND Z<6 THEN LET Z$(
1,1)="     ": LET Z$(1,2)=E$: GO
TO 0840
 750 IF Z>5 AND Z<10 THEN LET Z$
(2,1)="     ": LET Z$(2,2)=E$: GO
 TO 0840
 760 IF Z>9 AND Z<14 THEN LET Z$
(3,1)="     ": LET Z$(3,2)=E$: GO
 TO 0840
 770 IF Z>17 AND Z<22 THEN LET Z
$(4,1)="     ": LET Z$(4,2)=E$: G
O TO 0840
 780 IF Z>21 AND Z<26 THEN LET Z
$(5,1)="     ": LET Z$(5,2)=E$: G
O TO 0840
 790 IF Z>25 AND Z<30 THEN LET Z
$(6,1)="     ": LET Z$(6,2)=E$: G
O TO 0840
 800 LET L(B,1)=L(D,1): LET L(B,
2)=L(D,2): LET N(B)=N(D): LET X$
(B)=X$(D): LET D=D-1
 810 IF D>0 THEN GO TO 0360
 820 IF W=0 THEN GO TO 0850
 830 GO TO 0600
 840 BEEP .15,2: GO SUB 0920: GO
 SUB 0950: GO TO 0800
 850 CLS : FLASH 1: FOR X=1 TO 2
2: PRINT "====================
==========": NEXT X: PRINT AT 10
,12;"GAME OVER": FLASH 0
 860 PRINT
 870 FLASH 0: IF SC>=HS THEN PRI
NT "WELL DONE YOU HAVE THE NEW H
IGH                SCORE
    ": LET HS=SC: INPUT "ENTER Y
OUR NAME ";N$
 880 PRINT : PRINT PAPER 0; INK
7;N$;" HAS THE HIGH SCORE"
 890 PRINT : PRINT FLASH 1;"PRES
S B TO PLAY AGAIN"
 900 IF INKEY$<>"B" THEN BEEP 0.
3,1: GO TO 0900
```



```
 910 LET SC=0: GO TO 0020
 920 LET Q$="   "+Z$(1,1)+Z$(2,1)
+Z$(3,1)+"   "+Z$(4,1)+Z$(5,1)+
Z$(6,1)+"   "
 930 LET Q$=Q$+"   "+Z$(1,2)+Z$(2
,2)+Z$(3,2)+"   "+Z$(4,2)+Z$(5,
2)+Z$(6,2)+"   "
 940 RETURN
 950 PRINT AT 19,0; INK 4;Q$( TO
 2); INK 3;Q$(3 TO 14); INK 4;Q$
(15 TO 18); INK 3;Q$(19 TO 30);
INK 4;Q$(31 TO 34); INK 3;Q$(35
TO 46); INK 4;Q$(47 TO 50); INK
3;Q$(51 TO 62); INK 4;Q$(63 TO )
 960 RETURN
 970 PRINT AT 20,13;"LEVEL ";H
 980 FOR F=1 TO 200: NEXT F: CLS

 990 RETURN
```

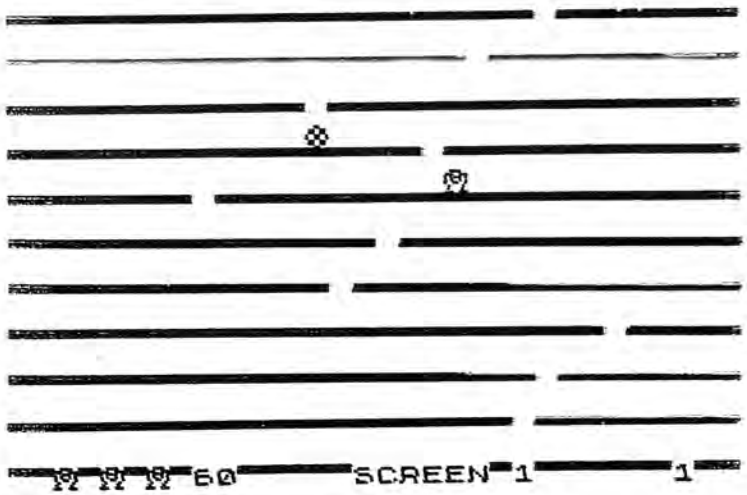


Hi-Rize

From Paul Toland comes this exciting arcade game, which involves jumping through many screens of lines with moving gaps. There are five screen variations to be played through, and you can see them all in these screen printouts:

HI-RIZE

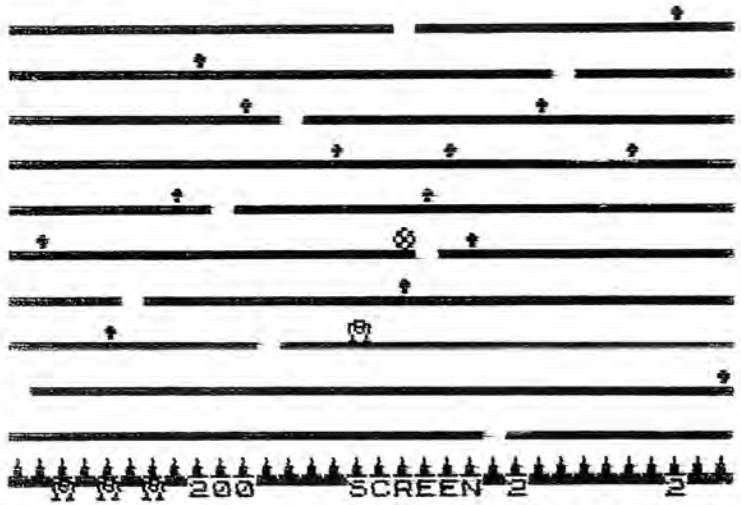
SCREEN 1



HI-RIZE

SCREEN 2

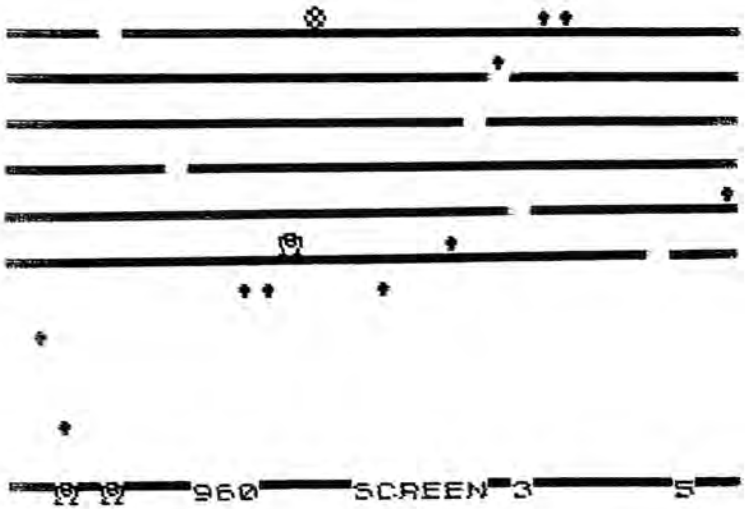
From this screen on, red dots appear on the screen and you can gain points by going over these.
In this screen you may not return to the base line.



HI-RIZE

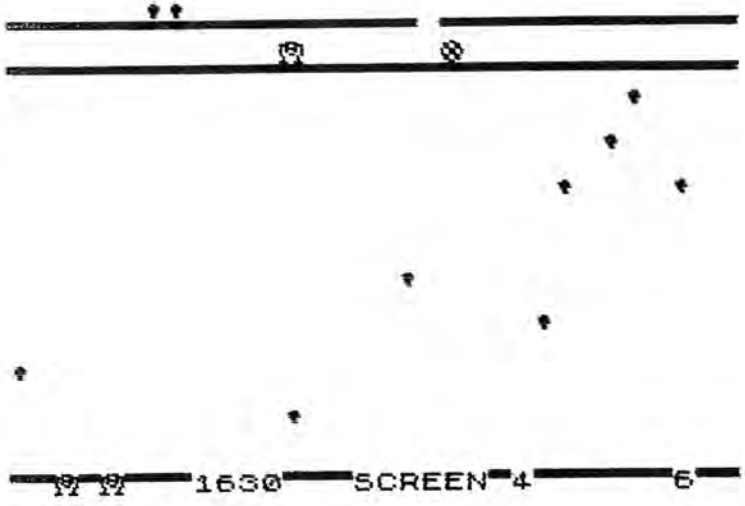
SCREEN 3

As you move up the screen the lines below you vanish, so you will lose a life if you fall.



HI-RIZE

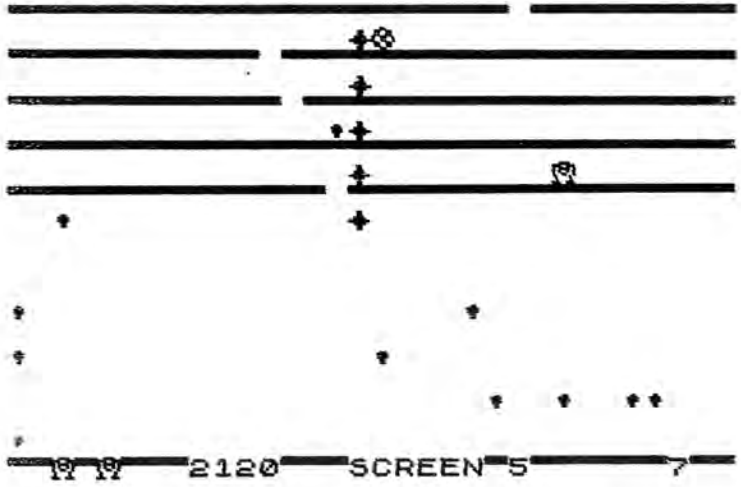
SCREEN 4



HI-RIZE

SCREEN 5

Spikes appear at the top of the screen and their touch is lethal.



And here's the listing for HI-RIZE:

```
  1 REM HI-RIZE © P. TOLAND
 80 GO SUB 800: GO SUB 2500
 90 LET TOT=TOT+1: LET UN=UN+2:
GO SUB 920
 100 FOR I=H TO L: PRINT AT I*2-
1,H(I,1);CHR$ 144: LET N=H(I,1)+
H(I,2): IF N<0 OR N>31 THEN LET
A=INT (RND+.5): LET H(I,2)=A-(A=
0): LET N=(H(I,2)=-1)*31
 110 PRINT AT I*2-1,N;" ": LET H
(I,1)=N: NEXT I
```

```
 120 PRINT AT Y,X;" ": IF SCREEN
$ (Y+1,X)=" " THEN LET SC=SC-20*
G: PRINT AT 21,8;SC;CHR$ 144: BE
EP .1,-20: LET Y=Y+2: LET C=146:
 LET T=0: IF L<10 THEN LET L=L+1
: IF G>2 THEN GO TO 300
 130 LET T=T+1: IF T=UN THEN LET
 C=145: BEEP .05,30
 140 IF C=146 THEN GO TO 155
 145 LET I$=INKEY$: IF I$<>"1" T
HEN GO TO 150
 146 IF SCREEN$ (Y-1,X)="" THEN
BEEP .05,0: LET C=146: LET T=0:
GO TO 160
 147 IF Y<20 THEN LET L=L-1: IF
L<10 AND G>2 THEN PRINT AT Y+1,0
;B$
 148 LET SC=SC+10*G: PRINT AT 21
,6;SC;CHR$ 144: BEEP .05,20: LET
 Y=Y-2: LET X=X-H(Y/2+1,2)
 150 LET X=X+(I$="8")-(I$="5"):
LET X=X+(X=-1)*32-(X=32)*32
 154 IF SCREEN$ (Y+1,X)=" " THEN
 GO TO 120
 155 LET N=ATTR (Y,X): IF N=50 T
HEN LET SC=SC+10*G: BEEP .05,5:
PRINT AT 21,8;SC;CHR$ 144
 157 IF N=48 OR N=51 THEN GO TO
310
 160 IF G>3 THEN LET H=Y/2: Y=2-
0)
 162 PRINT INK 0;AT Y,X;CHR$ C:
IF Y=0 THEN LET G=G+1: RESTORE 2
000: FOR I=1 TO 13: READ N: BEEP
 .1,N: NEXT I: GO TO 90
 170 PRINT AT BY,BX;" ": LET BX=
BX+D: LET BX=BX+(BX=-1)*32-(BX=3
2)*32: LET A$=SCREEN$ (BY+1,BX):
 IF SCREEN$ (BY+1,BX-D)=" " OR A
$=" " THEN LET BY=BY+2: LET D=-H
(BY/2+1,2): IF BY>11 OR BY>Y THE
N LET BY=0: LET BX=16: LET D=-H(
1,2)
 175 LET N=ATTR (BY,BX): IF N=48
 THEN GO TO 310
 177 IF N=51 THEN LET D=-D: LET
BX=BX+D
 180 PRINT INK 0;AT BY,BX;CHR$ 1
47
 185 IF G=2 AND Y<20 THEN PRINT
INK 0;AT 20,0;N$
 190 GO TO 100
```



```
 300 FOR I=L*2 TO 20: PRINT AT I
,X;CHR$ 146;AT I-1,X;" ": BEEP .
1,20-I: NEXT I
 310 BEEP 2,-10: LET M=M-1: IF M
=0 THEN INPUT "  TRY AGAIN ?";A$
: GO TO 1+(A$="N")*9998+(A$="n")
*9998
 320 LET G=G-1+(G=1)
 330 GO TO 90
 600 RESTORE 600: LET T$="": FOR
 I=1 TO 63: READ N: LET T$=T$+CH
R$ (N+128): NEXT I
 610 DATA 0,14,13,0,7,11,0,12,0,
11,6,0,7,11,0,3,3,11,0,11,3,0,10
,5,0,5,10,0,0,0,10,9,0,5,10,0,0,
0,0,0,11,3,0,10,5,0,13,14,0,0,0,
10,6,0,13,14,0,13,12,12,0,14,12
 620 LET T$="          "+T$(  TO 22)+"
          "+T$(23 TO 42)+"
      "+T$(43 TO )
 900 RESTORE 900: FOR I=0 TO 55:
READ N: POKE USR "A"+I,N: NEXT
I
 905 LET N=0: DIM B$(32): RANDOM
IZE : LET TOT=0: LET UN=5: LET S
C=0: LET M=4: LET G=1: RETURN
 910 DATA 255,255,255,0,0,0,0,0
 911 DATA 60,231,189,165,153,66,
 912 DATA 195,165,129,189,189,18
9,231,24
 913 DATA 24,102,102,153,153,102
,102,24
 914 DATA 0,0,24,60,60,24,24,0,0
 915 DATA 16,16,16,56,56,56,254,
0
 916 DATA 24,24,60,255,60,24,24,
24
 920 LET N$="": LET L$="": FOR I
=1 TO 32: LET N$=N$+CHR$ 149: LE
T L$=L$+CHR$ 144: NEXT I
 930 IF P$="Y" THEN GO SUB 2550
 940 LET H=1: LET L=10: LET T=10
0: LET X=16: LET Y=20: LET C=145
 950 INK 1: PAPER 6: BORDER 6: C
LS
 960 LET BX=16: LET BY=0
 970 IF G>1 THEN FOR I=1 TO 20:
PRINT INK 2;AT INT (RND*10)*2,RN
D*31;CHR$ 148: NEXT I
 980 IF G>4 THEN FOR I=2 TO 10 S
TEP 2: PRINT AT I,15; INK 3;CHR$
 150: NEXT I
```

```
1000 DIM H(10,2)
1010 FOR I=1 TO 21 STEP 2: PRINT
 AT I,0;L$: NEXT I
1020 FOR I=1 TO 10: LET H(I,1)=I
NT (RND*31): LET A=INT (RND+.5):
 LET H(I,2)=A-(A=0): NEXT I
1030 LET D=-H(1,2)
1040 PRINT AT 21,8;SC;AT 21,15;"
SCREEN";CHR$ 144;G;AT 21,29;TOT:
 FOR I=1 TO M-1: PRINT AT 21,I*2
;CHR$ 145: NEXT I
2000 RETURN
2010 DATA 2,2,2,14,14,14,11,11,1
1,7,11,7,2
2500 PAPER 1: INK 7: BORDER 1: C
LS : INPUT ;: PRINT AT 10,0;"Do
you want instructions ?"
2505 LET I=0
2510 LET I$=INKEY$: PRINT AT 0,0
; INK I;T$: IF I$="" THEN LET I=
I+1-(I=8)*8: GO TO 2510
2520 LET P$="N": IF I$="Y" OR I$
="y" THEN LET P$="Y"
2530 RETURN
2550 LET I=0: INK 7: PAPER 1: CL
S : BORDER 1: INPUT ;: PRINT AT
4,10; INK 6;"SCREEN ";G: GO SUB
G*100+2500
2560 LET I$=INKEY$: PRINT AT 0,0
; INK I;T$: IF I$="" THEN LET I=
I+1-(I=8)*8: GO TO 2560
2570 BEEP .5,10: RETURN
2600 PRINT AT 6,0;"Your aim is t
o reach the top of  each of the
screens in order  to advance t
o the next.  you move up b
y jumping through  moving gaps
in the lines above  whilst avoid
ing falling down  the gap in t
he line you are  standing on.
  You also have to  avoid the ba
rrel which rolls  about on the
 higher lines; you  will lose a
life if it hits  you."
2610 PRINT "Move left&right with
 keys 5&8,  jump with key 1": R
ETURN
2700 PRINT AT 6,0;"From this scr
een on, red dots  appear on th
e screen and you  can gain poi
nts by going over  these."
2710 PRINT "In this screen you m
ay not  return to the base
```



```
line.": RETURN
2800 PRINT AT 6,0;"As you move u
p the screen the  lines below
you vanish, so you  will lose a
life if you fall.": RETURN
2900 PRINT AT 10,0;"Things speed
 up a bit from now  on.": RETUR
N
3000 PRINT AT 10,0;"Spikes appea
r at the top of the  screen and
their touch is  lethal.": R
ETURN
9900 RETURN
9999 INK 0: PAPER 7: BORDER 7
```

Wipe-Out

No collection of computer games would be complete without a 'city bomber' type program. This is a much stronger version of the game than many I've seen, and provides a real challenge to the player. Here it is in action (at the end of a run):

The scenario is familiar. You are in a space ship which is coming down. You do not have any means of ascending, so must blast a landing strip for yourself by dropping bombs on the buildings. Use the "H" key to release your bombs.

```
  10 PAPER 0: BORDER 0: CLS
  20 DATA 0,0,0,63,255,0,2,0,1,3
,7,255,255,63,252,28,0,0,0,0,255
,255,255,255,255,76,255,255,102,
255,255,2,8,84,71,116,39,120,2
```



```
  30 FOR c=111 TO 116: FOR n=0 T
O 7
  40 READ x: POKE USR CHR$ c+n,x
  50 NEXT n: NEXT c
  60 LET be=0
  70 PRINT AT 9,5; INK 5;"***WIP
E-OUT***- H to drop        bomb.
Wipe the city off the      earth
with H-bombs to give       yourse
lf a landing strip.";AT 15,8; BR
IGHT 1; INK 3;"DIFFICULTY ?(1 TO
 7)": INPUT D: LET D=7-D
  80 PAPER 0: CLS : BORDER 0
  90 LET s=-1
 100 FOR a=5 TO 25
 110 INK 1+INT (RND*7): PAPER 1+
INT (RND*7): LET q=11+INT (RND*9
)
 120 PRINT AT q,a; PAPER 0;"▄";
FOR b=q+1 TO 21: PRINT AT b,a;"▒
": NEXT b
 130 NEXT a
 140 FOR a=1 TO 21
 150 PAPER 0: INK 1+INT (RND*7)
 160 LET bp=(21-a)
 170 FOR b=28 TO 1 STEP -1
 180 PRINT AT a,b;"▬▶": BEEP .0
1,bp
 190 FOR t=0 TO d
 200 LET s=s+2*(INKEY$="h")+(s=1
): GO SUB 320
 210 NEXT t
 220 IF SCREEN$ (a,b-1)<>" " THE
N GO TO 450
 230 PRINT AT a,b;"   "
 240 NEXT b: PRINT AT a,1;"  "
 250 NEXT a
 260 DATA 4,4,4,0
 270 RESTORE 260: FOR a=1 TO 4:
READ x: BEEP .25*(1+a*(a=4)),x:
NEXT a
 280 PRINT AT 21,0; FLASH 1; INK
 3; PAPER 5;"SCORE:";be;"standby
 for take-off": PAUSE 300
 290 PAPER 0: FOR a=20 TO 0 STEP
 -1
 300 PRINT AT a+1,a+1;"  ";AT a,
a; INK 1+INT (RND*6);"▬▶"
 310 NEXT a: GO TO 70
 320 IF a=21 THEN RETURN : IF s=
1 THEN LET e=a+1
 330 IF s=1 THEN LET e=a+2-(a=20
)
```

```
 340 IF s=1 THEN LET h=b+2: IF s
=1 THEN LET d=d/2
 350 IF s<0 THEN RETURN
 360 PRINT AT e-1,h;" ";AT e,h;"
*": BEEP .006,21-e: LET e=e+1
 370 IF SCREEN$ (e,h)<>" " THEN
GO TO 390
 380 IF e<21 THEN RETURN : IF e>
21 THEN LET e=21: GO TO 400
 390 LET be=be+10
 400 IF e>21 THEN LET e=21
 410 PRINT AT e-1,h; PAPER 0;" "
: FOR n=7 TO 0 STEP -1
 420 PRINT AT e,h; INK n;"■": BE
EP .02,n-12
 430 NEXT n
 440 LET s=-1: LET d=d*2: RETURN

 450 PRINT AT a,b-2;"   "
 460 DATA 0,0,0,-5,-5,-5,0
 470 RESTORE 460: INK 2
 480 FOR n=10 TO 90 STEP 10: PLO
T b*8,(21-a)*8: DRAW 20,n: NEXT
n
 490 PRINT AT 11,10; PAPER 6; IN
K 1;"SCORE:";be
 500 STOP
 510 GO SUB 410: GO TO 510
```


Birds

Written by Neil Pellinacci, this is an arcade-type game, in which you have to defend your planet from the lines of alien birdmen above.

Once by one the aliens swoop down on you, dropping large white crates in the process The crates are used as bombs, but later in the game, the crates can smother your planet, completely covering it. If this happens, the game ends, no matter how many lives you have left. The crates have a third role, but you'll have to play the game to discover what it is.

You control your laser base using the "Z" and "X" keys, with the space key to fire. You score 50 points for each swooping birdman you hit. If you hit a birdman on the bottom row, you get just 25 points. However, a birdman hit here will not be killed, but merely stunned, and will soon return.

You start with three laser bases and lose one each time it is hit by a birdman or a crate. The number of bases is shown on the top row of the screen, together with your score (on the left) and the current high score. You gain an extra base when you've destroyed three attack waves.

At this point, you'll meet mutant aliens. These swoop, but may then rise, dropping more crates.

To give you something to aim at, you might like to know that the best score that Neil has achieved - and only after a great deal of practice - is 12,000.

The program is written entirely in BASIC with the exception of a single machine code routine located at 32500. Here is a disassembled listing of the code:

```
32500 LD    HL,704
32503 LD    DE,22528
32506 LD    BC,32
32509 PUSH  HL
32510 ADD   HL,DE
32511 DEC   HL
32512 LD    A,(HL)
32513 CP    126
32515 JR    Z ,32539
32517 POP   HL
32518 DEC   HL
32519 LD    A,H
32520 OR    L
32521 JR    NZ,32509
32523 LD    HL,23263
32526 LD    B,32
32528 LD    A,(HL)
32529 CP    126
32531 JR    NZ,32536
32533 DEC   HL
32534 DJNZ  32528
32536 LD    C,0
32538 RET
32539 LD    (HL),70
32541 ADD   HL,BC
32542 LD    (HL),126
32544 JR    32517
```


And here is the key to the user-defined graphics:

AB =
CD =
EFG =
HIJ =
KL =

After you have the program working, you may wish to relocate the code. If you do, remember to check lines 90, 317, 412 and 7020. The subroutine locates the position of each crate and moves it down one position. It also checks to see if the planet is completely covered by crates. The result of this test is detected on return to BASIC by line 317.

You should type the program in, and then save it, before trying to run in, in case you've made a mistake with the machine code. It is also advisable to save the program so that it starts automatically on loading.

```
  10 RANDOMIZE : GO TO 90
  20 PRINT AT 0,6-LEN STR$ 5; IN
K 5;S: RETURN
  30 LET BC=BC+(INKEY$="x" AND B
C<27)-(INKEY$="z" AND BC>0): PRI
NT AT 21,BC; PAPER 6; INK 6;"
  "
  35 IF INKEY$<>" " THEN RETURN
  40 GO TO 520
  50 FOR X=1 TO 3: FOR Y=1 TO 10
  60 IF L(X,Y)<>-1 THEN LET L1=L
(X,Y): LET C1=C(X,Y): LET C2=Y:
LET L2=X: RETURN
  70 NEXT Y: NEXT X: LET WF=1: R
ETURN
```

```
  60 FOR X=20 TO 7+INT (RND*10)
STEP -1: PRINT AT X,BC+2; PAPER
7;" ": BEEP .001,40-X: PRINT AT
X,BC+2; PAPER 0;" ": NEXT X: PRI
NT AT X,BC+2; INK 6; PAPER 7;" "
: RETURN
  90 CLEAR 32499
  95 BORDER 6: PAPER 0: CLS : BO
RDER 0
 100 INK 4: BRIGHT 1: PRINT AT 0
,26; INK 5;"000000"
 105 GO SUB 7000: GO SUB 9000
 107 LET HS=0
 110 GO SUB 3000: LET BC=13: LET
 S=0: LET LB=3: LET WW=0
 120 LET FF=0
 130 LET HF=0
 140 LET BFF=0
 160 LET WF=0
 200 DIM L(3,10): DIM C(3,10)
 205 FOR B=0 TO 2
 210 FOR A=0 TO 9
 220 LET L(B+1,A+1)=B*2+1: LET C
(B+1,A+1)=A*3+1
 230 NEXT A
 235 NEXT B
 237 PRINT AT 0,0; INK 5;"000000
                  ";AT 0,17;LB
 240 FOR B=1 TO 3
 250 FOR A=1 TO 10: PRINT AT L(B
,A),C(B,A);"▲": NEXT A
 255 NEXT B
 260 LET BH=0
 290 FOR K=1 TO INT (RND*15+1)
 300 LET B=INT (RND*3+1): LET A=
INT (RND*10+1)
 305 IF L(B,A)=-1 THEN GO TO 315
 310 PRINT AT L(B,A),C(B,A);"▲"
: BEEP .01,-10: PRINT AT L(B,A),
C(B,A);"▲": BEEP .01,5
 312 IF RND>.6 THEN IF ATTR (22,
BC+2)=126 THEN GO SUB 80
 315 IF INKEY$<>"" THEN GO SUB 3
0
 317 IF NOT USR 32500 THEN GO TO
 4000
 318 IF ATTR (21,BC+1)=126 OR AT
TR (21,BC+2)=126 OR ATTR (21,BC+
3)=126 THEN GO TO 1000
 320 NEXT K
```



```
 330 IF BH>20 THEN GO SUB 50: IF
 WF=0 THEN GO TO 405
 340 IF WF=1 THEN GO TO 2000
 400 LET L2=INT (RND*3+1): LET C
2=INT (RND*10+1)
 402 LET L1=L(L2,C2): LET C1=C(L
2,C2)
 404 IF L1=-1 THEN GO TO 400
 405 PRINT AT L1,C1;" "
 406 LET CD=INT (RND*3-1)
 410 PRINT AT L1,C1; OVER 1;"▲"
: BEEP .001,L1+20: PRINT OVER 1;
AT L1,C1;"▲"
 411 IF RND>.6 THEN IF L1>6 THEN
 PRINT AT L1+1,C1; INK 6; PAPER
7;" "
 412 LET ZZ=USR 32500: IF INKEY$
<>"" THEN GO SUB 30: IF HF=1 THE
N LET HF=0: GO TO 290
 413 IF ATTR (21,BC+1)=126 OR AT
TR (21,BC+2)=126 OR ATTR (21,BC+
3)=126 THEN GO TO 1000
 415 LET C1=C1+CD: IF C1=32 THEN
 LET C1=0
 416 IF C1=-1 THEN LET C1=30
 418 IF RND>.65 THEN LET CD=INT
(RND*3-1)
 419 IF WW>=3 THEN IF RND>.6 THE
N LET L1=L1+1: LET C1=C1+CD
 420 LET L1=L1+1: IF L1<>21 THEN
 GO TO 410
 425 IF C1=BC+1 OR C1=BC+2 THEN
LET BFF=1: GO TO 1000
 430 PRINT AT L(L2,C2),C(L2,C2);
"▲": GO TO 290
 520 PLOT BC*8+20,6: DRAW INK 3,
 PAPER 0,0,128: PLOT OVER 1,BC*8
+20,8: DRAW OVER 1,0,128
 530 IF BC+2=C1 OR BC+1=C1 THEN
GO TO 610
 540 IF SCREEN$ (5,BC+2)<>" " TH
EN PRINT AT 5,BC+1;"    ": LET S=
S+25: GO SUB 20
 600 RETURN
 610 PRINT AT L1,C1; INK 7;"▒▒▒";
AT L1,C1;"   ": LET C1=-1: LET L1
=-1: BEEP .05,-30: LET L(L2,C2)=
-1: LET S=S+50: GO SUB 20
 620 LET HF=1: LET BH=BH+1: RETU
RN
```

```
1000 REM BANG
1010 FOR Y=1 TO 4: FOR Z=1 TO 10
: BEEP .015,20-Z: PRINT AT 21,BC
; INK RND*7;"▒▒▒": NEXT Z
1020 FOR Z=10 TO 1 STEP -1: BEEP
 .015,20-Z: NEXT Z: NEXT Y
1030 LET LB=LB-1: PRINT AT 0,17;
 INK 5;LB
1040 IF LB=0 THEN GO TO 1060
1045 IF BFF=1 THEN LET BFF=0: GO
 TO 430
1050 GO TO 415
1060 PRINT AT 0,17; INK 1; PAPER
 7; FLASH 1;LB
1070 FOR X=1 TO 20: PRINT AT X,0
;"
  ": NEXT X
1080 BEEP .2,50: BEEP .25,30
1090 FOR X=1 TO 7: PRINT AT X+7,
0; INK X;" GAME OVER
GAME OVER ": NEXT X
1100 BEEP .2,50: BEEP .25,30
1270 IF S>HS THEN LET HS=S
1280 PRINT AT 0,32-LEN STR$ HS;
INK 5;HS
1290 IF S=HS THEN FOR Z=1 TO 5:
FOR X=20 TO -20 STEP -2: BEEP .0
1,X: NEXT X: NEXT Z
1300 FOR X=20 TO 50: BEEP .005,20
: NEXT X
1305 FOR X=0 TO 20: BEEP .005,0:
 NEXT X
1310 FOR X=1 TO 300: NEXT X: BOR
DER 6: CLS : BORDER 0: PRINT AT
0,26; INK 5;"000000";AT 0,32-LEN
 STR$ HS; INK 5;HS: GO TO 110
1400 STOP
2000 REM WAVE OVER
2010 FOR X=1 TO 20: PRINT AT X,0
;"
  ": NEXT X
2020 FOR X=1 TO 7: PRINT AT 3+X*
2,2; INK X;"    ATTACK WAVE DESTR
OYED"
2030 NEXT X
2040 PAUSE 150: BEEP .2,50: FOR
X=-30 TO 35: BEEP .01,X: BEEP .0
15,20: NEXT X: BEEP .2,50
2050 FOR X=1 TO 20: PRINT AT X,0
;"
  ": NEXT X
```



```
2060 LET WW=WW+1: IF WW=3 THEN B
ORDER 6: PAPER 0: CLS : BORDER 0
: PRINT AT 0,26; INK 5;"000000";
AT 0,32-LEN STR$ HS; INK 5;HS: B
EEP .2,10: BEEP .2,-10: LET LB=L
B+1
2070 GO TO 120
3000 REM TITLE PAGE
3010 RESTORE 3500
3020 READ X: IF X=999 THEN GO TO
 3100
3040 READ Y: PLOT X,Y
3050 READ X: IF X=-999 THEN GO T
O 3020
3060 READ Y: DRAW X,Y: GO TO 305
0
3100 PRINT AT 10,4; INK 5;"© NEI
L PELLINACCI 1983"
3110 PRINT AT 13,7; INK 2; PAPER
 7;"C O N T R O L S"
3120 PRINT AT 15,2; INK 6;"Z....
.....................LEFT"
3130 PRINT AT 17,2; INK 6;"X....
.....................RIGHT"
3140 PRINT AT 19,2; INK 6;"(SPAC
E)...................FIRE"
3150 PRINT AT 21,5; INK 2; PAPER
 7;"PRESS ANY KEY TO PLAY"
3160 INK 7: PLOT 0,48: DRAW 0,60:
 DRAW 53,0: PLOT 176,70: DRAW 77
,0: DRAW 0,-68: DRAW -45,0: PLOT
 2,2: DRAW 37,0
3165 INK 2: PLOT 0,63: DRAW 255,
0: DRAW 0,16: DRAW -255,0: DRAW
0,-16
3170 INK 4
3200 FOR X=30 TO -30 STEP -1: BE
EP .01,X: BEEP .01,-X: NEXT X: P
AUSE 0
3210 FOR X=1 TO 21: PRINT AT X,0
;"
  ": NEXT X
3220 RETURN
3400 RETURN
3500 DATA 35,110,0,56,30,-14,-30
,-14,30,-14,-30,-14,-999
3510 DATA 86,110,0,56,-999
3520 DATA 106,110,0,56,30,-14,-3
0,-14,30,-28,-999
3530 DATA 151,110,0,56,30,-28,-3
0,-28,-999
```

```
3540 DATA 168,110,30,28,-30,0,30
,28,-999
3700 DATA 999
4000 REM LANDED
4010 BEEP .2,3: BEEP .15,-1: BEE
P .25,5
4020 FOR Z=1 TO 23
4030 FOR X=1 TO 3: FOR Y=1 TO 10
4040 IF L(X,Y)<0 OR L(X,Y)>20 TH
EN GO TO 4055
4050 PRINT AT L(X,Y),C(X,Y); INK
 3;"  ": LET L(X,Y)=L(X,Y)+1: PR
INT AT L(X,Y),C(X,Y); INK 3;"A"
4060 NEXT Y: NEXT X
4070 NEXT Z
4080 FOR Z=20 TO -30 STEP -3: BE
EP .01,Z: BEEP .01,Z+1.5: BEEP .
01,-Z+10: NEXT Z
4090 BEEP .15,-5: BEEP .15,-10:
BEEP .13,1
4095 FOR A=1 TO 5
4100 FOR Z=1 TO 19 STEP 2: PRINT
 AT Z,0; INK RND*6+1;"          BI
RDS HAVE LANDED          ": BEEP .0
1,Z*2: NEXT Z
4105 NEXT A
4200 BEEP 2,-10
4300 GO TO 1060
4900 STOP
7000 REM m/c
7020 RESTORE 7030: FOR A=1 TO 46
: READ B: POKE 32499+A,B: NEXT A
7030 DATA 33,192,2,17,0,88,1,32,
0,229,25,43,126,254,126,40,22
7040 DATA 225,43,124,181,32,242
7050 DATA 33,223,90,6,32,126,254
,126,32,5,43,16,248,14,0
7060 DATA 201,54,70,9,54,126,24,
227
8000 RETURN
9000 FOR C=0 TO 95: READ B: POKE
 USR "A"+C,B: NEXT C
9010 DATA 192,113,26,7,1,2,2,0
9020 DATA 3,142,88,224,126,64,64
,0
9030 DATA 0,1,2,7,25,114,196,0
9040 DATA 0,128,64,224,152,78,35
,0
9050 DATA 0,0,0,0,4,14,31,255
```



```
9060 DATA 8,8,28,28,62,127,255,2
55
9070 DATA 0,0,0,0,16,56,252,255
9080 DATA 0,2,0,0,16,8,0,255
9090 DATA 8,0,42,126,42,201,34,2
55
9100 DATA 0,32,0,126,36,136,0,25
5
9110 DATA 129,72,2,16,130,8,64,1
33
9120 DATA 129,16,64,9,64,16,2,16
1
9300 RETURN
```

We'll now have a look at what each part of the program does:

10: this skips over the following subroutines, which are at the beginning for maximum speed

20: subroutine to print score

30: this moves the laser base. If the space key is pressed, control is sent to line 520

50: this is the beginning of a subroutine which is called when only a few birds are left. It locates the first living birdman, and selects it to swoop down. If there are no birds left, WF is set to 1 to tell the main loop what has happened

80: an 'additional hazard' subroutine

90: makes room for the machine code

110: game initialisation

120 - 160: wave initialisation

200 - 235: sets up the birds

240 - 255: display birds

290 - 320: makes a few birds flap at random

317: checks for a whole line of crates; also scrolls them down

318: checks for a collision between the base and a crate

330: selects next bird without using the random selection routine, see subroutine 50

340: checks for end of wave

400 - 404: selects a bird at random, and sees if it exists

410 - 420: the main loop

411: drops a crate

419: if a bird is a mutant, it may move upward

520: fire routine

610 - 620: exploding bird

1000 - 1050: base hit routine

1060 - 1310: game over routine

2000 - 2070: 'attack wave over' routine

2060: start of mutant waves

3000 - 3210: title page routine

3500 - 3700: data for title page

4000 - 4300: this routine is called when the planet has been completely covered by crates; it causes the birdmen to change color and slowly land

7020: POKEs the machine code

7030 - 7060: data for machine code

9000: sets up UDG characters

9010 - 9120: data for graphics characters

Finally, here is a list of the main variables used in the program:

HS - high score
BC - base position
S - score

LB - bases left
WW - wave number
HF - hit flag (1 - yes, 0 - no)
WF - wave over flag (1 - yes, 0 - no)
L(3,10), C(3,10) - positions of the 30 birdmen
BH - number of birdmen hit
L1,C1 - position of swooping birdman
CD - horizontal direction of birdman

As with most programs in this book, there are many things which can be changed to suit your own needs. For instance, the random numbers can be changed, thus changing the probability of various events taking place (for example, lines 411 and 419). The screen displays can also be changed at various stages.

Chopper Squad

Action is the name of this game from Malcolm Young. You are in charge of a rescue helicopter flying above the desert.

You have to save as many soldiers as you can before you run out of fuel. You can either pick up a soldier by landing on him for 50 points, or for a faster and easier pickup, use the helicopter skyhook by pressing "1" when you are exactly two spaces above the man. This will get you 25 points.

You can also use the skyhook for clearing trees. However, in doing so you reduce the helicopter's cover. This cover is important as it helps shield you from the people who come along in jeeps from time to time to shoot at you. They can only shoot straight up, and you can destroy the jeeps by simply dropping bombs on them. Hardly seems fair, does it. You can only bomb them while in the sky, and you do so by pressing "1" again. (The program will know whether you want the bomb or the skyhook, which is why "1" is use to trigger both.)

After you pick up a man, the ambulance turns black and you can then gain a bonus of 100 points if you can get the man to the black ambulance.

To add to the things you have to keep track of in this program, you'll see (from time to time) a fuel truck somewhere along the bottom of the screen. If you manage to bomb this, you'll gain 20 fuel units as well as 10 points.

CHOPPER SQUAD ends when you either run out of fuel or when your lives have all been used up. There is a 'hall of fame' at the end of the program, which you can easily extend to five places.

```
  5 GO SUB 550
  10 REM a= b= c= d= e= f=
g= h= i= j= l= m= n= o=
p=
  20 DIM s$(5,10): DIM h(5)
  30 LET tr=0: LET sf=0: LET dp=
0: LET sc=0: LET pk=0: LET lives
=3
  40 BORDER 7: PAPER 7: INK 0: B
RIGHT 0: CLS
  110 LET jf=0: LET ln=100: LET f
uel=200
  120 BORDER 6: PAPER 5: INK 0: C
LS
  130 LET h$="": LET r$="": L
ET a$="": LET j$="": LET p$=
"": LET t$=""
  140 FOR g=11 TO 21: PRINT PAPER
 5;AT g,0;"
                ": NEXT g
  150 FOR t=0 TO 31: PAPER 6: LET
 r=INT (RND*4): PRINT INK 2;AT 1
5-r,t;")"; INK 4;AT 15-r,t;"*":
NEXT t
  160 LET y=INT (RND*10)+1: LET x
=INT (RND*30): LET a=1
  170 PRINT PAPER 6;AT 11,a-1;" "
; INK 2-2*pk;AT 11,a;a$
  175 LET fuel=fuel-1: PRINT PAPE
R 0; INK 7; BRIGHT 1;AT 0,0;"FUE
L=";fuel;" ";AT 0,10;"SCORE=";sc
;AT 0,20;"LIVES=";lives
```



```
  178 IF fuel<0 THEN GO TO 1000
  180 LET a=a+0.5 AND (a<30): IF
NOT a THEN LET pk=a
  190 PRINT OVER 0;AT y,x;" "
  191 IF y=11 AND x+1=a THEN LET
sc=sc+50: LET pk=0: BEEP .5,-5
  200 LET y=y+(INKEY$="6")-(INKEY
$="7"): LET x=x+(INKEY$="8")-(IN
KEY$="5")
  205 IF y<11 THEN LET x=x+(INKEY
$="8")-(INKEY$="5")
  210 LET y=y+(y<0)-(y>21): LET x
=x+(x<0)-(x>30)
  220 LET c=ATTR (y,x): IF c<>40
AND c<>48 THEN GO TO 1000
  230 LET c=ATTR (y,x+1): IF c<>4
0 AND c<>48 THEN GO TO 1000
  240 PRINT OVER 0;AT y,x;h$: BEE
P .1,-10: PRINT OVER 1;AT y,x;r$
  245 IF RND<.3 AND NOT pk THEN L
ET sf=1
  246 IF sf THEN GO SUB 420+dp
  250 IF RND<.5 THEN LET jf=1
  260 GO TO jf*ln+170

  270 LET jy=16+INT (RND*4)+1
  280 LET ln=150
  290 LET jd=RND: LET jd=(jd>.5)-
(jd<.5)
  300 LET q$=p$ AND jd=1: LET q$=
q$+(j$ AND jd=-1): IF RND<.333 T
HEN LET tr=1
  305 IF tr=1 THEN LET q$=(t$(2 T
O ) AND jd=1)+(t$( TO 2) AND jd=
-1)
  310 LET jx=30 AND jd=-1
  320 PRINT INK 0;AT jy,0;"
                "
  325 PRINT INK 1;AT jy,jx;q$
  330 LET jx=jx+jd

  340 IF jx=0 OR jx=31 THEN LET l
n=100: LET tr=0: LET jf=0: PRINT
 PAPER 6;AT jy,0;"
                ": GO TO 170
  350 IF y<10 AND INKEY$="1" THEN
 GO TO 510
  360 IF tr=1 OR NOT (jx>=INT x-1
 AND jx<=x+1) THEN GO TO 170
```

```
 370 LET e=y+1: FOR f=jy-1 TO e
STEP -1: LET b=ATTR (f,jx+1): IF
 b=40 OR b=48 THEN PRINT PAPER 6
; INK 2;AT f,jx+1;".": BEEP .1,f
: PRINT INK 0;AT f,jx+1;" ": NEX
T f
 380 PRINT INK 0;AT f,jx+1;" "
 390 IF (jx+1=x+1 OR jx+1=x) AND
 f=y THEN BEEP 1,-5: GO TO 1000
 400 GO TO 170
 410 REM Transport soldier
 420 LET sdx=INT (RND*30)+1: LET
 sdy=12+INT (RND*5): LET dp=10:
LET t=INT (RND*20)+10
 430 PRINT INK 1;AT sdy,sdx;"¥":
 BEEP .15,0
 440 LET t=t-1: PRINT AT sdy,sdx
;" ": IF NOT t THEN LET sf=0: LE
T dp=0: RETURN
 450 IF y=sdy AND x+1=sdx THEN L
ET sc=sc+50: BEEP 1,5: LET sf=0:
 LET dp=0: LET pk=1: RETURN
 460 IF INKEY$<>"1" THEN RETURN
 470 PRINT AT y+1,x+1;"↓";AT y+2
,x+1;"↓"
 480 IF sdy=y+2 AND sdx=x+1 THEN
 FOR m=y+2 TO y STEP -1: PRINT A
T m,x+1;"¥": BEEP .3,0: PRINT AT
 m,x+1;" ": NEXT m: LET pk=1: LE
T sf=0: LET dp=0: LET sc=sc+25
 490 PRINT AT y+1,x+1;" ";AT y+2
,x+1;" "
 500 RETURN
 510 LET e=jy: FOR f=y+1 TO e: L
ET b=ATTR (f,x+1): IF b=40 OR b=
48 THEN PRINT PAPER 6; INK 2;AT
f,x+1;"¥";CHR$ 8: BEEP .1,30-f:
PRINT INK 0;AT f,x+1;" ": NEXT f
: GO TO 530
 520 PRINT INK 0;AT f,x+1;" "
 530 IF b=49 THEN BEEP 1,-5: LET
 sc=sc+10: LET fuel=fuel+20*(tr=
1): PRINT AT jy,jx;"  ": LET j
x=0: GO TO 340
 540 GO TO 170
 560 REM
 570 RESTORE
 580 DATA "a",1,1,1,255,255,195,
219,24
 590 DATA "b",15,124,6,63,63,195
,219,24
```



```
 600 DATA "c",128,128,128,255,25
5,195,219,24
 610 DATA "d",48,12,10,252,252,1
95,219,24
 620 DATA "e",0,16,124,254,255,1
27,36,126
 630 DATA "f",0,0,192,199,255,1,
0,0
 640 DATA "g",255,0,0,0,0,0,0,0
 650 DATA "h",7,0,0,0,0,0,0,0
 660 DATA "i",126,97,97,255,255,
195,219,24
 670 DATA "j",255,239,199,239,25
5,195,219,24
 680 DATA "k",16,16,16,16,16,16,
84,56
 690 DATA "l",138,73,116,188,114
,82,69,178
 700 DATA "m",36,36,24,16,16,16,
16,16
 710 DATA "n",153,90,78,56,8,20,
20,54
 720 DATA "o",231,126,24,24,60,1
26,60,24
 730 DATA "p",126,255,255,255,25
5,195,90,24
 970 FOR c=1 TO 16: READ c$
 980 FOR e=0 TO 7: READ n: POKE
 USR c$+e,n: NEXT e
 990 NEXT c
 995 RETURN
1000 PRINT INK 2; FLASH 1;AT y,x
;"▒▒": BEEP 1,1
1010 PRINT INK 0; FLASH 0; PAPER
 8;AT y,x;"  ": LET y=INT (RND*1
0)+1: LET x=INT (RND*30)
1020 LET lives=lives-1
1025 IF fuel<1 THEN PRINT FLASH
1;"OUT OF FUEL! I'M AFRAID THIS
 IS THE END": GO TO 1040
1030 IF lives THEN GO TO 170
1040 IF sc<h(5) THEN GO TO 1200
1050 INPUT "Please enter your na
me for the score table",n$
1060 FOR r=5 TO 1 STEP -1: BEEP
.1,r: IF sc>h(r) THEN NEXT r
1070 FOR g=5 TO r+2 STEP -1: BEE
P .1,g: LET h(g)=h(g-1): LET s$(
g)=s$(g-1): NEXT g
```

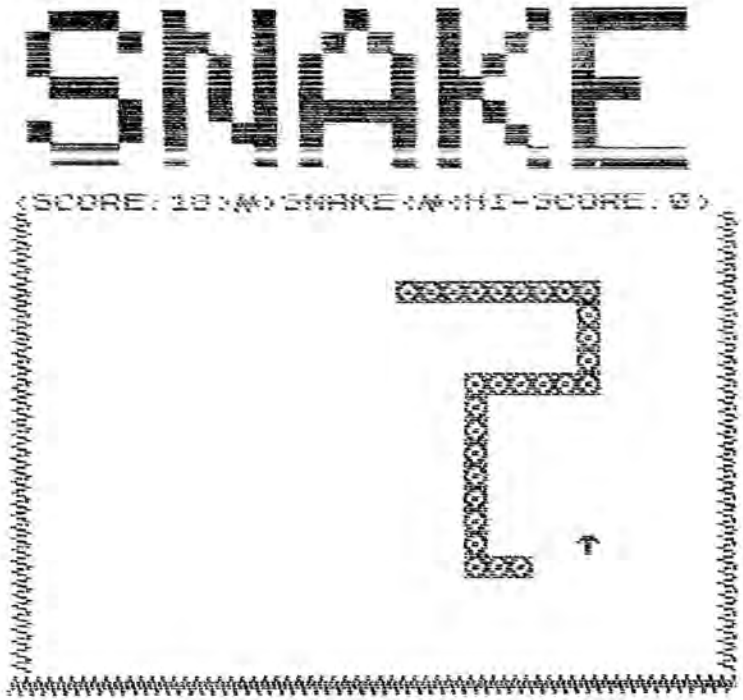
```
1080 LET h(g)=sc: LET s$(g)=n$
1090 BORDER 1: PAPER 2: INK 7: B
RIGHT 1: CLS
1110 PRINT INK 5;AT 3,2;"TODAYS
CHAMPIONS ARE-"
1120 FOR a=1 TO 5: BEEP .5,a: PR
INT INK 0;AT 7+a,8;s$(a);" ";h(a
): NEXT a
1200 PRINT FLASH 1; PAPER 4; INK
 0;AT 16,3;"DO YOU WANT TO PLAY
AGAIN?"
1210 IF INKEY$="n" OR INKEY$="N"
 THEN GO TO 1240
1220 IF INKEY$="" THEN GO TO 121
0
1230 GO TO 30
1240 BORDER 7: PAPER 7: INK 0: B
RIGHT 0: STOP
```


Snake

Written by David Perry, SNAKE is a game demanding a steady eye and concentration.

The object of the game is to lead your snake around a garden - surrounded by an electric fence - while getting the snake to eat mushrooms. Each mushroom makes the snake grow longer, and you must make sure you do not hit the snake's tail...or the game will end.

Here's what it looks like in action:



And here's the listing for SNAKE:

```
  20 GO SUB 210
  25 REM ***Start display
  30 LET I=0: INVERSE 0: BRIGHT
0: FLASH 0: BORDER 0: PAPER 0: I
NK 7: CLS
  40 PRINT "  ■■■■  ■   ■   ■   ■
 ■ ■■■■■"
  50 PRINT " ■    ■ ■■  ■  ■ ■  ■
 ■  ■"
  60 PRINT " ■      ■ ■ ■ ■    ■ ■
■  ■"
  70 PRINT "  ■■■■  ■ ■ ■ ■    ■ ■
■ ■■■■"
  80 PRINT "      ■ ■ ■ ■ ■■■■■ ■
 ■  ■"
  90 PRINT " ■    ■ ■ ■■■ ■    ■ ■
 ■  ■"
 100 PRINT "  ■■■■  ■   ■ ■    ■ ■
 ■ ■■■■■"
 110 PRINT AT 14,1;"Use keys Q,Z
,I&P for up,down,": PRINT " left
 and right. Each time you": PRIN
T " eat a toadstool the snake wi
ll": PRINT " grow longer.": PRIN
T " BE CAREFUL NOT TO BUMP INTO"
 120 PRINT " YOUR OWN BODY OR TH
E FENCE!!!"
 130 PRINT : PRINT INK 3;"
  GOOD LUCK  "
 140 PRINT AT 10,6; INK I;"ENTER
 LEVEL OF PLAY"
 150 PRINT AT 12,11; INK I/2;"(1
 TO 9)"
 155 REM ***Input Level
 160 LET A$=INKEY$: IF A$="" THE
N LET I=I+1: IF I>7 THEN LET I=0
 170 IF A$<"1" OR A$>"9" THEN GO
 TO 140
 180 LET L=VAL A$
 190 INK 0
 200 GO TO 350
 205 REM ***User defined Chrs
 210 FOR A=1 TO 5
 220 READ A$
 230 FOR N=0 TO 7
 240 READ B
 250 POKE USR A$+N,B
 260 NEXT N
 270 NEXT A
```



```
 280 DATA "A",189,102,195,153,15
3,195,102,189
 290 DATA "B",0,60,126,153,24,24
,24,24
 300 DATA "C",90,36,189,102,102,
189,36,90
 310 DATA "D",0,34,34,65,65,65,1
36,136
 320 DATA "E",28,96,28,3,28,96,2
8,3
 330 LET HS=0
 340 RETURN
 350 LET B$="p"
 360 BORDER 7
 370 PAPER 7
 380 LET A=0
 390 LET SC=0
 400 CLS
 405 REM ***Screen set up
 410 PRINT AT 0,0; INK 1;""; INK
 5;"<SCORE:0>"; INK 1;"#>"; INK
2;"SNAKE"; INK 1;"<#"; INK 4;("<
" AND HS<100);"HI-SCORE:";HS;(">
" AND HS<100)
 420 FOR N=1 TO 20
 430 PRINT AT N,0; INK 1;"$
                           $"
 440 NEXT N
 450 PRINT AT 21,0; INK 1;"######
################################"
 460 LET X=10
 470 LET Y=19
 480 LET A$=""
 490 FOR N=10 TO 19
 500 LET A$=A$+"10"+STR$ N
 510 PRINT BRIGHT 1; INK 2;AT 10
,N;"@"
 520 NEXT N
 530 GO SUB 800
 540 LET D$=INKEY$
 545 REM ***Key input
 550 IF D$="p" OR D$="q" OR D$="
i" OR D$="z" THEN LET B$=D$
 560 LET X=X+(B$="z")-(B$="q")
 570 LET Y=Y+(B$="p")-(B$="i")
 580 IF X=21 OR X=0 THEN GO TO 6
50
 590 IF Y=31 OR Y=0 THEN GO TO 6
50
 600 IF ATTR (X,Y)=122 OR X=22 O
R Y=32 THEN GO TO 830
 610 GO SUB 750
```

```
  620 LET C$=STR$ X
  630 IF X<10 THEN LET C$=" "+STR
$ X
  640 LET D$=STR$ Y
  650 IF Y<10 THEN LET D$=" "+STR
$ Y
  660 LET X1=VAL A$( TO 2)
  670 LET Y1=VAL A$(3 TO 4)
  680 IF ATTR (X1,Y1)<>122 THEN G
O TO 700
  690 PRINT AT X1,Y1;" "
  695 REM ***Print snake
  700 PRINT BRIGHT 1; INK 2;AT X,
Y;"O"
  710 LET A$=A$+C$+D$
  720 IF A=0 THEN LET A$=A$(5 TO
)
  730 LET A=A-(A>0)
  740 GO TO 540
  750 LET M=ATTR (X,Y)
  760 IF M<>60 THEN RETURN
  770 LET A=A+L
  780 LET SC=SC+L
  790 PRINT AT 0,0; INK 5;"<SCORE
:";SC;">"; INK 1;"(M)"; INK 2;"SN
AKE"; INK 1;"(M"; INK 4;"<HI-SCO
RE:";HS;(">" AND HS<100)
  800 PRINT PAPER 7; INK 4;AT INT
 (RND*18)+2,INT (RND*29)+2;"T"
  810 FOR n=50 TO -20 STEP -10: B
EEP .01,n: NEXT n
  820 RETURN
  830 LET X=X-(X=22)+(X=-1)
  840 LET Y=Y-(Y=32)+(Y=-1)
  850 FOR F=50 TO -20 STEP -5: BE
EP .02,F: NEXT F
  855 REM ***End of game
  860 IF SC>HS THEN LET HS=SC
  870 PRINT AT 10,10;"HIGH SCORE:
";HS
  880 PRINT AT 12,11;"Press a key
"
  890 LET I=0
  900 PRINT AT X,Y; INK I;"■": BE
EP .01,I*2
  905 REM ***Print scores
  910 PRINT AT 8,6; INK I; FLASH
1;">You scored ";SC;" points.<"
  920 LET 0$=INKEY$: IF 0$="" THE
N LET I=I+1: IF I>7 THEN LET I=0
  930 IF 0$="" THEN GO TO 900
  940 CLS
  950 GO TO 30
```


Dodgems

This fast-moving action game is by Raymond Blake. You steer your car around a track with four lanes, passing over flags to gain points. At the same time, you are trying to avoid the suicidal car controlled by the computer, which travels in the opposite direction to you, trying to spoil your fun.

You can change lanes in the four gaps in the track, and you can change tracks as many times as you like. The suicide car can also change lanes, but it is limited to moving a single lane at a time. Despite this, you'll find this a very difficult game.

Near the start of a run, you'll gain points for red flags, which turn yellow as you pass over them. When you've run out of red flags (or you just feel like a change) passing over the magenta flag will allow you to gain points from yellow flags, but will also allow your Kamikaze opponent a chance to catch up!

Another magenta flag will appear in another part of the track, and passing over it will once again change the color of the flags which allow you to gain points.

```
   20 FOR i=1 TO 5: READ z$: FOR
j=0 TO 7: READ z: POKE USR z$+j,
z: NEXT j: NEXT i
```

```
  30 LET hs=0
  35 LET c=6: LET c1=2: LET sc=0
  40 BORDER 0: PAPER 0: INK 7: B
RIGHT 1: CLS
  50 PRINT "##########################
#########"
  60 PRINT "##"; INK 2;"¶¶¶¶¶¶¶¶¶¶¶¶
¶¶¶¶      ¶¶¶¶¶¶¶¶¶¶¶¶"; INK 7;"##
"
  70 PRINT "##"; INK 2;"¶"; INK
7;"##########          ##########";
INK 2;"¶¶"; INK 7;"##"
  75 PRINT "##"; INK 2;"¶"; INK
7;"##"; INK 2;"¶¶¶¶¶¶¶¶¶¶      ¶¶¶
¶¶¶¶"; INK 7;"#"; INK 2;"¶"; INK 7
;"##"
  80 PRINT "##"; INK 2;"¶"; INK
7;"#"; INK 2;"¶"; INK 7;"########
##      ##########"; INK 2;"¶"; INK
7;"#"; INK 2;"¶"; INK 7;"##"
  85 PRINT "##"; INK 2;"¶"; INK
7;"#"; INK 2;"¶¶¶¶¶¶¶¶¶¶¶¶"; INK 7
;"#"; INK 2;"¶"; INK 7;"##"
  90 PRINT "##"; INK 2;"¶"; INK
7;"#"; INK 2;"¶"; INK 7;"#"; INK
7;"#"; INK 2;"¶"; INK 7;"########
##"; INK 2;"¶"; INK 7;"#"; INK 2;"¶
"; INK 7;"##"
  95 PRINT "##"; INK 2;"¶"; INK
7;"#"; INK 2;"¶"; INK 7;"#"; INK
7;"#"; INK 2;"¶¶¶¶¶¶¶¶¶¶¶¶"; INK 7
;"#"; INK 2;"¶"; INK 7;"##"
 100 PRINT "##"; INK 2;"¶"; INK
7;"#"; INK 2;"¶"; INK 7;"#"; INK
7;"#"; INK 2;"¶¶¶¶¶¶¶¶¶¶¶¶"; INK 7
;"#"; INK 2;"¶"; INK 7;"##"
 110 PRINT "##"; INK 2;"          "
; INK 7;"#          #"; INK 2;"¶¶
"; INK 7;"##"
 115 PRINT "##          #
#          ##"
 120 PRINT "##"; INK 2;"          "
; INK 7;"#          #"; INK 2;"¶¶
"; INK 7;"##"
```



```
 125 PRINT "##"; INK 2;"          "
; INK 7;"#          #"; INK 2;"¶¶
"; INK 7;"##"
 130 PRINT "##"; INK 2;"¶"; INK
7;"#"; INK 2;"¶"; INK 7;"#"; INK
2;"¶"; INK 7;"#"; INK 2;"¶"; IN
K 7;"##########"; INK 2;"¶"; "
INK 7;"#"; INK 2;"¶"; INK 7;"#"
; INK 2;"¶"; INK 7;"#"; INK 2;"¶
"; INK 7;"##"
 135 PRINT "##"; INK 2;"¶"; INK
7;"#"; INK 2;"¶"; INK 7;"#"; INK
2;"¶¶¶¶¶¶¶¶"; INK 7;"#"; INK 2;"¶¶
"; INK 7;"#"; INK 2;"¶"; INK 7;
"#"; INK 2;"¶"; INK 7;"##"
 140 PRINT "##"; INK 2;"¶"; INK
7;"#"; INK 2;"¶"; INK 7;"#"; INK
7;"#"; INK 2;"¶"; INK 7;"######  ######
#"; INK 2;"¶"; INK 7;"#"; INK 2;"¶
"; INK 7; "#"; INK 2;"¶"; INK 7
;"##"
 150 PRINT "##"; INK 2;"¶"; INK
7;"#"; INK 2;"¶"; INK 7;"#"; INK
7;"#"; INK 2;"¶¶¶¶¶¶¶¶"; INK 7;"#"
; INK 2;"¶"; INK 7;"#"; INK 2;"¶
"; INK 7;"##"
 160 PRINT "##"; INK 2;"¶"; INK
7;"#"; INK 2;"¶"; INK 7;"########
##"; INK 2;"¶"; INK 7;"#"; INK 2;"¶
"; INK 7;"#"; INK 2;"¶"; INK 7;"##"
 170 PRINT "##"; INK 2;"¶"; INK
7;"#"; INK 2;"¶¶¶¶¶¶¶¶¶¶¶¶"; INK 7
;"#"; INK 2;"¶"; INK 7;"#"; INK 2;"¶
"; INK 7;"##"
 180 PRINT "##"; INK 2;"¶"; INK
7;"##########          ##########";
INK 2;"¶¶"; INK 7;"##"
 190 PRINT "##"; INK 2;"¶¶¶¶¶¶¶¶¶¶
¶¶¶¶      ¶¶¶¶¶¶¶¶¶¶¶¶"; INK 7;"##
"

 200 PRINT "##########################
#########"
 210 LET x=20: LET y=2: LET x1=1
: LET y1=29: LET z1=66: LET flag
=0
 220 PRINT AT x,y; INK 4;"¶";AT
x1,y1; INK 5;"#";AT 0,22; INK 0;
 FLASH 1;"¶"
 230 LET lane=1: LET lane1=1
 240 PRINT AT 10,11;"Score";AT 1
1,10;"Hiscore ";hs
```

```
 250 LET a$="8": LET m$="■": LET
 b$="6": LET n$="■"
 300 IF flag>0 THEN GO TO 500
 310 IF SCREEN$ (x+(a$="6")-(a$=
"7"),y+(a$="8")-(a$="5"))<>"■" T
HEN GO TO 400
 320 IF a$="5" THEN LET a$="6":
LET m$="■": GO TO 400
 330 IF a$="6" THEN LET a$="8":
LET m$="■": GO TO 400
 340 IF a$="7" THEN LET a$="5":
LET m$="■": GO TO 400
 350 IF a$="8" THEN LET a$="7":
LET m$="■"
 401 IF (x>8 AND x<13) OR (y>13
AND y<18) THEN PRINT AT x,y; INK
 c1;" ": GO SUB 900: GO TO 410
 402 LET a=ATTR (x+(a$="6")-(a$=
"7"),y+(a$="8")-(a$="5"))
 403 PRINT AT x,y; INK c;"■"
 405 IF a=195 THEN GO SUB 850
 407 IF a=64+c1 THEN LET sc=sc+1
 410 LET x=x+(a$="6")-(a$="7"):
LET y=y+(a$="8")-(a$="5")
 420 PRINT AT x,y; INK 4;m$
 430 PRINT AT 10,17;sc: BEEP .00
1,20
 450 IF x=x1 AND y=y1 THEN GO TO
 800
 500 IF flag>0 THEN LET flag=fla
g-1: BEEP .02,20
 510 IF SCREEN$ (x1+(b$="6")-(b$
="7"),y1+(b$="8")-(b$="5"))<>"■"
 THEN GO TO 600
 520 IF b$="5" THEN LET b$="7":
LET n$="■": GO TO 600
 530 IF b$="6" THEN LET b$="5":
LET n$="■": GO TO 600
 540 IF b$="7" THEN LET b$="8":
LET n$="■": GO TO 600
 550 IF b$="8" THEN LET b$="6":
LET n$="■": GO TO 600
 580 IF (x>8 AND x<13) OR (y>13
AND y<18) THEN PRINT AT x,y;" ":
 GO SUB 900: GO TO 610
 601 IF (x1=9 AND b$="6") OR (y1
=14 AND b$="8") OR (x1=12 AND b$
="7") OR (y1=17 AND b$="5") THEN
 PRINT AT x1,y1; INK c1;" ": GO
SUB 700
```



```
 602 IF (x1>8 AND x1<13) OR (y1>
13 AND y1<18) THEN PRINT AT x1,y
1; INK c1;" ": GO TO 610
 603 LET z=ATTR (x1+(b$="6")-(b$
="7"),y1+(b$="8")-(b$="5"))
 605 PRINT AT x1,y1; INK ((z1-64
)-(128 AND z1-64>8)); FLASH 1 AN
D z1-64>8;"■"
 606 LET z1=z
 610 LET x1=x1+(b$="6")-(b$="7")
: LET y1=y1+(b$="8")-(b$="5")
 620 PRINT AT x1,y1; INK 5;n$
 635 IF x=x1 AND y=y1 THEN GO TO
 800
 650 GO TO 300
 700 IF lane=lane1 THEN RETURN
 705 PRINT AT x1,y1; INK c1;" "
 720 LET l=2*SGN (lane-lane1): L
ET lane1=lane1+l/2
 750 LET x1=x1+l*((b$="8")-(b$="
5"))
 760 LET y1=y1+l*((b$="7")-(b$="
6"))
 799 RETURN
 800 FOR i=25 TO -10 STEP -1: BE
EP .02,i: NEXT i
 810 PRINT AT 14,2; PAPER 4; INK
 0; FLASH 1;"Press any key for a
 new game"
 815 IF sc>hs THEN LET hs=sc
 820 IF INKEY$<>"" THEN GO TO 82
0
 830 IF INKEY$="" THEN GO TO 830
 845 GO TO 35
 850 LET c=8-c: LET c1=8-c1: LET
 sc=sc-10: BEEP .01,30: BEEP .01
,30
 855 LET flag=10
 860 LET sc=sc+10
 870 LET lin=INT (RND*20)+1: LET
 col=INT (RND*30)+1: IF CODE SCR
EEN$ (lin,col)<>0 THEN GO TO 870
 875 PRINT AT lin,col; INK 3; FL
ASH 1;"■"
 888 RETURN
 900 LET i$=INKEY$: IF y>13 AND
y<18 THEN GO TO 950
 905 IF y>22 THEN GO TO 925
 910 IF i$="5" AND y>2 THEN LET
y=y-2: LET lane=lane-1
 915 IF i$="8" AND y<8 THEN LET
y=y+2: LET lane=lane+1
```

```
  920 RETURN
  930 IF i$="5" AND y>23 THEN LET
  y=y-2: LET lane=lane+1
  935 IF i$="8" AND y<29 THEN LET
  y=y+2: LET lane=lane-1
  940 RETURN
  950 IF x>10 THEN GO TO 975
  960 IF i$="6" AND x<7 THEN LET
x=x+2: LET lane=lane+1
  965 IF i$="7" AND x>1 THEN LET
x=x-2: LET lane=lane-1
  970 RETURN
  980 IF i$="6" AND x<20 THEN LET
  x=x+2: LET lane=lane-1
  985 IF i$="7" AND x>14 THEN LET
  x=x-2: LET lane=lane+1
  990 RETURN
1000 DATA "a",12,60,124,60,12,4,
4,4
1010 DATA "b",24,90,126,90,24,15
0,255,153
1020 DATA "c",224,76,68,255,255,
68,76,224
1030 DATA "d",153,255,153,24,90,
126,90,24
1040 DATA "e",7,114,34,255,255,3
4,114,7
5000 IF flag>0 THEN LET flag=fla
g-1
```

```
1 - graphic A
2 - graphic B
3 - graphic C
4 - graphic D
5 - graphic E
```


Zombies

Zombies, from David Perry, is a game in which you must lure the Zombies into swamps. They are blind, and always move towards you.

Full instructions are provided within the program. Use the cursor keys to move around. Note that the pound signs in this listing should be entered as hash (#) signs.

```
  1 REM
  2 REM
  3 REM      > ZOMBIES <
  4 REM        ========
  5 REM
  6 REM      COPYRIGHT
  7 REM     DAVID PERRY
  8 REM        1983!
  9 REM
 10 REM
 11 LET NUM=10: DIM N(NUM+1): D
IM N$(NUM+1,8): FOR N=1 TO NUM:
LET N(N)=1100-(N*100): LET N$(N)
="SPECTRUM": NEXT N
 12 BORDER 7: PAPER 7: INK 0: B
RIGHT 0: CLS
 13 LET LEV=5: GO SUB 66
 14 LET SC=0
 15 BORDER 7: PAPER 7: INK 0: B
RIGHT 0: CLS : DIM A(LEV): DIM B
(LEV): LET KILL=0
```

```
  16 PRINT AT 0,3;"
          "
  17 FOR N=1 TO 20: PRINT AT N,3
;" "; INK 4;"
 "; INK 0;" ": NEXT n
  18 PRINT AT 21,3;"
          "
  19 BRIGHT 0: BORDER 7
  20 PRINT AT 2,28;"7";AT 3,27;"
5"; INK 2;"X"; INK 0;"8";AT 4,28
;"6";AT 0,26;"Move.";AT 1,26;"--
---";AT 6,27;"Key";AT 7,27;"---"
  21 PRINT AT 8,27;"You="; INK 6
;"a";AT 9,26; INK 0;"Zomb="; INK
 2;"b";AT 10,26; INK 0;"Swamp";
INK 3;"c"
  22 PRINT AT 12,27;"GOOD";AT 13
,27;"LUCK";AT 14,27;"     "
  23 PRINT AT 16,28;"";LEV-KILL;
AT 17,28;"TO";AT 18,27;"KILL!"
  24 FOR N=1 TO LEV
  25 LET A(N)=INT (RND*20): LET
B(N)=INT (RND*20): IF ATTR (A(N)
+1,B(N)+4)<>124 THEN  GO TO 25
  26 PRINT AT A(N)+1,B(N)+4; INK
 2; PAPER 4;"b": NEXT n
  27 FOR N=1 TO LEV*5
  28 LET A=INT (RND*20): LET B=I
NT (RND*20): IF ATTR (A+1,B+4)<>
124 THEN  GO TO 28
  29 PRINT AT A+1,B+4; INK 3; PA
PER 4;"c": NEXT n
  30 LET X=INT (RND*20): LET Y=I
NT (RND*20): IF ATTR (X+1,Y+4)<>
124 THEN  GO TO 30
```



```
  31 PRINT AT X+1,Y+4; INK 6; PA
PER 4;"a": FOR N=0 TO 50 STEP 5:
 BEEP .02,N: NEXT N
  32 PRINT AT X+1,Y+4; INK 6; PA
PER 4;"a"
  33 FOR N=1 TO LEV: IF A(N)=100
 AND N=LEV THEN  GO TO 49
  34 IF A(N)=100 THEN  NEXT N
  35 IF KILL=LEV THEN  GO TO 91
  36 IF N>LEV THEN  LET N=LEV
  37 PRINT AT A(N)+1,B(N)+4; BRI
GHT 1; INK 4;" "
  38 IF A(N)<X THEN  LET A(N)=A(
N)+1
  39 IF B(N)>Y THEN  LET B(N)=B(
N)-1
  40 IF A(N)>X THEN  LET A(N)=A(
N)-1
  41 IF B(N)<Y THEN  LET B(N)=B(
N)+1

  42 LET A=ATTR (A(N)+1,B(N)+4)
  43 IF A=124 THEN  PRINT AT A(N
)+1,B(N)+4; INK 2; PAPER 4; BRIG
HT 1;"b": BEEP .01,N: NEXT N
  44 IF A=99 THEN  LET SC=SC+10:
 FOR I=7 TO 0 STEP -1: PRINT AT
A(N)+1,B(N)+4; BRIGHT 1; PAPER 4
; INK I;"c": BEEP .02,I*7: NEXT
I: PRINT AT A(N)+1,B(N)+4; INK 3
; PAPER 4; BRIGHT 1;"c": LET KIL
L=KILL+1: PRINT AT 16,28;"";LEV-
KILL;"  ": LET A(N)=100: IF N<=L
EV THEN  NEXT N
  45 IF N>=LEV THEN  LET N=LEV
```

```
  46 IF N<=LEV AND A(N)=X AND B(
N)=Y THEN  FOR I=0 TO 7: PRINT A
T A(N)+1,B(N)+4; BRIGHT 1; PAPER
 4; INK I; FLASH 1;"a": BEEP .1,
I*7: NEXT I: GO TO 87
  47 IF KILL=LEV THEN  GO TO 91
  48 IF N<LEV THEN  NEXT N
  49 IF LEV=KILL THEN  PAUSE 30:
 GO TO 91
  50 FOR G=1 TO 2: PRINT AT X+1,
Y+4; INK 4; BRIGHT 1;" "
  51 LET X=X+(INKEY$="6")-(INKEY
$="7"): LET Y=Y+(INKEY$="8")-(IN
KEY$="5")
  52 LET A=ATTR (X+1,Y+4)
  53 IF A=124 THEN  PRINT AT X+1
,Y+4; INK 6; PAPER 4; BRIGHT 1;"
a": BEEP .04,20: NEXT G: GO TO 3
2
  54 IF A=98 THEN  GO TO 103
  55 LET LEV=5: CLS : PRINT AT 1
,11;"ZOMBIES";AT 2,11;"======="
  56 PRINT AT 3,1;"YOU ARE SLOWL
Y SINKING INTO THE"
  57 FOR I=0 TO 7: PRINT AT I+6,
8; PAPER 7-I; INK I;"  S W A M P
 !  ": NEXT I
  58 PRINT AT 16,3;"HOWEVER YOU
HAVE MANAGED": PRINT AT 19,5;"TO
 SCORE ";SC;" POINTS!"
  59 PRINT £1;"(c) Copyright DAV
ID PERRY 1983"
  60 LET I=0
  61 PRINT AT 21,9; INK I;"PRESS
 A KEY!"
```



```
  62 LET I=I+1: IF I>7 THEN  LET
 I=1
  63 IF INKEY$="" THEN  GO TO 61
  64 LET LEV=5: PAUSE 0
  65 GO TO 106
  66 CLS : PRINT AT 0,10;"ZOMBIE
S";AT 1,10;"========"
  67 PRINT : PRINT "  YOU HAVE J
UST LANDED ON AN"
  68 PRINT "    ISLAND POPULATED
 WITH?"
  69 RESTORE 70: FOR N=0 TO 31:
READ A: POKE USR "A"+N,A: NEXT N
  70 DATA 0,24,126,24,24,24,24,0
,129,195,165,165,102,24,36,24,12
9,126,66,90,90,66,126,129,153,90
,60,231,231,60,90,153

  71 FOR N=7 TO 13: PRINT AT N,1
0; INK N-7;"ZOMBIES": BEEP .5,N:
 NEXT N
  72 PRINT AT 14,10;"ZOMBIES": B
EEP .5,35
  73 PRINT AT 17,0;"(c) COPYRIGH
T DAVID PERRY 1983!"
  74 PAUSE 10: FOR N=0 TO 27: PR
INT AT 19,N; INK 2;" b "; INK 5;
"a": BEEP .02,N: NEXT N
  75 PRINT AT 19,N;"     ": PAUSE
 10: BEEP .5,40: PAUSE 5: FOR N=
27 TO 0 STEP -1: PRINT AT 19,N;
INK 2;"b "; INK 5;"a ": BEEP .02
,N: NEXT N
  76 CLS
  77 PRINT AT 0,10;"ZOMBIES!";AT
 1,10;"========="
```

```
  78 PRINT AT 2,1;"The zombies a
re flesh-eaters      however they
are blind, that      is your only
advantage. They      come in packs
 so you can use      yourself as b
ait to lure them     into the area
s of swamp.            Be careful,
 if you fall in      a pit then yo
u will die too.      You only have
 one life so-"
  79 PRINT AT 12,4;" B E  C A R
E F U L ! "
  80 PRINT AT 14,1;" The movemen
t keys are 5-8."
  81 PRINT AT 16,2;" YOU="; INK
5;"a"; INK 0;" ZOMBIE="; INK 2;"
b"; INK 0;" SWAMP="; INK 3;"c";
INK 0;","
  82 LET I=0: PRINT £1;"(c) Copy
right DAVID PERRY 1983"
  83 PRINT AT 19,8; INK I; PAPER
 2; BRIGHT 1;" PRESS A KEY! "
  84 LET I=I+1: BEEP .01,I*7: IF
 I>7 THEN  LET I=0
  85 IF INKEY$="" THEN  GO TO 83
  86 CLS : RETURN
  87 CLS : PRINT AT 1,11;"ZOMBIE
S";AT 2,11;"========="
  88 FOR N=0 TO 7: PRINT AT 4+N,
0; PAPER N; INK 7-N;" T H E Y  C
 A U G H T  Y O U ! ": NEXT N
  89 GO TO 58
  90 STOP
  91 CLS : PRINT AT 1,11;"ZOMBIE
S";AT 2,11;"========="
```



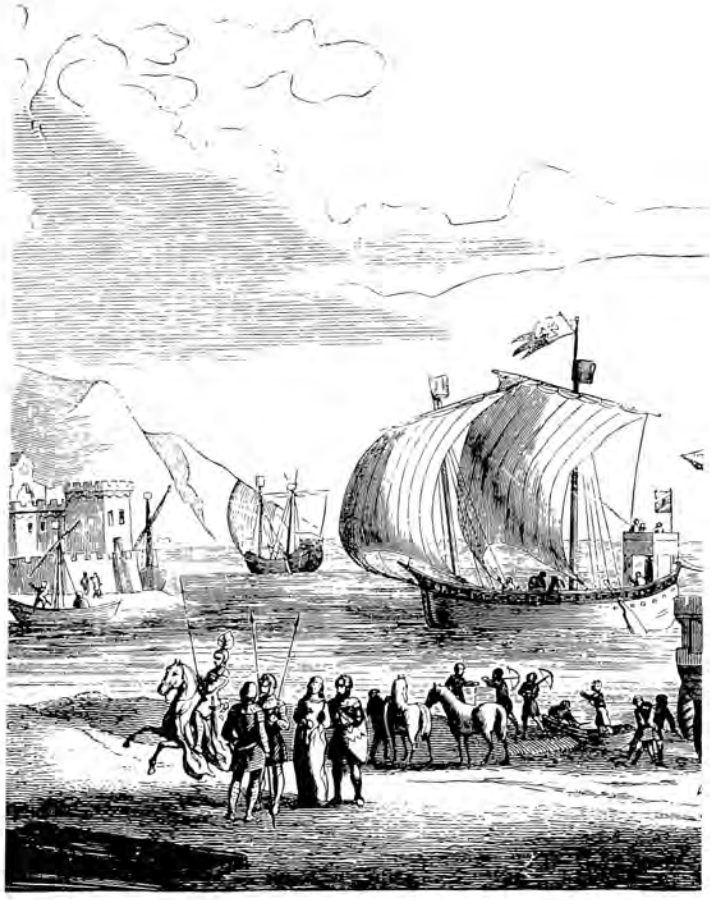
```
  92 FOR N=7 TO 0 STEP -1: PRINT
 AT 4+N,0; PAPER N; INK 7-N;" C
O N G R A T U L A T I O N S ": N
EXT N
  93 LET SC=SC+(LEV*20)
  94 PRINT AT 13,1;"YOU HAVE SCO
RED ";SC;" SO FAR!"
  95 LET LEV=LEV+5: FOR N=1 TO 1
0 STEP 2: BEEP .02,N: NEXT N: PR
INT AT 15,4;"NOW TRY THE NEXT LE
VEL": PRINT : PRINT "        WITH
 ";LEV;" ZOMBIES!"
  96 PRINT AT 19,7;"BONUS POINTS
=";(LEV-5)*20;""
  97 PRINT £1;"(C) Copyright DAV
ID PERRY 1983"
  98 LET I=0

  99 PRINT AT 21,8; INK I;"PRESS
 ANY KEY!"
 100 LET I=I+1: BEEP .02,I*5: IF
 I>7 THEN  LET I=0
 101 IF INKEY$="" THEN  GO TO 99
 102 GO TO 15
 103 CLS : PRINT AT 1,11;"ZOMBIE
S";AT 2,11;"========="
 104 FOR N=7 TO 0 STEP -1: PRINT
 AT 6+N,0; PAPER N; INK 7-N;" YO
U HAVE JUST COMMITED SUICIDE ":
NEXT N
 105 GO TO 58
 106 BORDER 0: PAPER 0: INK 7: B
RIGHT 1: CLS
 107 IF SC<=N(10) THEN  GO TO 12
3
```

```
 108 LET NUM=11: IF SC>=N(NUM) T
HEN  INPUT "ENTER 8 INITALS! ";P
$: IF LEN P$>8 THEN  GO TO 108
 109 PRINT AT 7,0; FLASH 1; BRIG
HT 0; INK 7; PAPER 2;"THIS WILL
ONLY TAKE A FEW SECS!"
 110 IF SC>=N(NUM) THEN  LET N(N
UM)=SC: LET N$(NUM)=P$
 111 FOR A=1 TO (NUM-1): LET B$=
N$(A): LET C$=N$(A+1): LET B=N(A
): LET C=N(A+1): IF B<C THEN  LE
T N(A)=C: LET N(A+1)=B: LET N$(A
)=C$: LET N$(A+1)=B$
 112 NEXT A: FOR N=1 TO NUM-1: I
F N(N)<N(N+1) THEN  GO TO 111
 113 NEXT N
 114 CLS
 115 PRINT AT 2,4;"H A L L  O F
 F A M E !"
 116 PRINT AT 3,4;"==========
==========="
 117 FOR N=1  TO NUM-1: PRINT AT
 N+5,7; INK 6;"("; INK 2;N; INK
6;")";AT N+5,12; INK 7;N(N): PRI
NT AT N+5,17; INK 5;N$(N): NEXT
N
 118 PLOT 0,0: DRAW 255,0: DRAW
0,175: DRAW -255,0: DRAW 0,-175:
 PRINT AT 17,8; PAPER 2; INK 7;"
 PRESS ANY KEY! ": LET I=0
 119 LET I=I+1: IF I>7 THEN  LET
 I=0
 120 PRINT AT 2,4; INK I;"H A L
L  O F  F A M E !"
 121 BEEP .01,I*7: PAUSE 2: IF I
NKEY$="" THEN  GO TO 119
```



```
 122 PAPER 7: INK 0: BORDER 7: B
RIGHT 0: CLS : GO TO 14
 123 PRINT AT 10,4;"SORRY SCORE
TO LOW!!!"
 124 PRINT AT 13,7;" PRESS A KEY
 "
 125 PAUSE 0: GO TO 114
```

ADVENTURES



TROLL and THE CITY

These two Adventure games, written by Neil Pellinacci, are played in similar ways, although the scenario and solution are - of course - different. It is worth trying to convince a friend to type the programs in for you, so you don't find out the solutions even before you run them.

THE CITY includes instructions in the program which precedes the main game. To complete the Adventure, you must say the password so your contact can hear it. You don't know who the contact is, although there are not many possibilities.

In the second Adventure, TROLL, you must try and escape from the Troll's lair. He's very mean and will let you escape only over his dead body. Therefore, you have to kill him to win.

In both programs, as in most Adventure games, you enter your commands as two-word phrases (verbs and nouns) such as GO WEST or DROP AXE. You'll soon discover which words the program will recognize.

Note that each program has a loader program (which for THE CITY doubles as an instruction manual). This sets up the screen and keyboard. 48K owners can leave out the CLEAR 32767 instruction, but the

CLS must be left in. If you use the in-built SAVE instructions, note that you'll have to set the colors and POKEs beforehand.

The loaders have been used to maximise memory space for the main program. To type the games in, enter and SAVE the loader program, then type in and SAVE the main game directly after it. When saving the programs, or re-running them, remember that the start line is 1 for the loaders and 900 for the main programs.

Here is the loader for TROLL:

```
  10 REM Adventure © Neil
         Pellinacci 1983
  20 BORDER 0: PAPER 0: INK 7: B
RIGHT 1: CLS
  30 CLEAR 32767
  40 POKE 23609,100: POKE 23658,
0: POKE 23624,71
  50 LOAD ""
```

And this is the main TROLL program:

```
  12 PRINT '"I AM AT THE TROLL'S
 GATE."
  13 IF O(9) THEN PRINT "I HAVE
NO WEAPON AND I MUST     LEAVE Q
UICKLY."
  14 LET S=P: RETURN
  16 PRINT '"I AM IN A SMALL SHO
P, WHICH     SELLS ONLY DOG-FOOD
.": LET S=P: RETURN
  33 PRINT '"I AM IN A SMALL, DA
RK CAVE.     THERE IS A GATE TO
THE NORTH,   AND A CLEARING TO T
HE EAST."
  34 LET N=P: LET S=P: LET E=P:
RETURN
```



```
  36 PRINT '"I AM IN A CLEARING.
 THE CAVE IS TO THE WEST.": LET
E=P: LET W=P: RETURN
  39 PRINT '"I AM ON A PATH, WHI
CH CONTINUES TO THE SOUTH. I CAN
 SEE A SHOP  TO THE NORTH, AND A
 FOUNTAIN TO THE EAST."
  40 LET N=P: LET S=P: LET E=P:
LET W=P: RETURN
  42 PRINT '"THERE IS A FOUNTAIN
 HERE. A ROADLEADS OFF TO THE SO
UTH.": LET S=P: LET W=P: RETURN
  54 PRINT '"I AM STANDING IN A
NARROW, ROCKYPASSAGE WHICH GENTL
Y RISES TO   THE NORTH. I CAN SE
E DAYLIGHT   ABOVE.": LET N=P: L
ET U=P: RETURN
  60 PRINT '"I AM STANDING ON A
GRAVEL PATH. A ROAD LIES TO THE
EAST.": LET N=P: LET E=P
  61 IF O(4)=LOC THEN PRINT "THE
RE IS A MOUND AT MY FEET."
  62 RETURN
  63 PRINT '"A ROAD RUNS NORTH-S
OUTH. A      FOUNTAIN LIES TO TH
E NORTH.": LET N=P: LET S=P: RET
URN
  69 PRINT '"I CAN SEE A SMALL C
LEARING WITH A LARGE TREE.": LET
 S=P: LET E=P: RETURN
  72 PRINT '"I AM IN A THICK FOR
EST. THERE   ARE CLEARINGS TO TH
E EAST AND   SOUTH."
  73 LET W=P: LET E=P: LET S=P:
RETURN
  75 PRINT '"I AM STANDING IN A
CLEARING.    THERE IS A DEEP SHA
FT IN THE    GROUND.": LET W=P:
LET D=P: RETURN
  84 PRINT '"THE ROAD ENDS HERE.
 THE RIVER ISTO THE SOUTH, WITH
MOUNTAINS    BEYOND.": LET N=P:
RETURN
  87 PRINT '"A LARGE WOLF LEAPT
OUT OF THE   TREES, AND KILLED M
E.": GO TO 9000
  90 PRINT '"I AM STANDING IN A
CLEARING. I  CAN HEAR RUSTLING N
OISES TO THE WEST.": LET N=P: LE
T E=P: LET W=P: RETURN
```

```
  93 PRINT '"THE FOREST ENDS HER
E. A RIVER    LIES TO THE SOUTH."
IT MAY BE     POSSIBLE TO CROSS."
: LET N=P: LET W=P: RETURN
 114 PRINT '"I AM STANDING ON TH
E RIVER BANK.I CAN SEE AN OPEN G
ATE GUARDED  BY A DOG, AND A MAN
SION BEYOND."
 115 LET S=DOG: IF NOT S THEN PR
INT "THE DOG WILL BE A PROBLEM..
."
 116 RETURN
 117 PRINT '"I AM NEAR A MANSION
. A DOOR IS   AHEAD OF ME.": LET
N=P: RETURN
 138 PRINT "I AM NOW INSIDE THE
 MANSION. I CERTAINLY WOULDN'T L
IVE HERE.    IT'S ALL DUSTY AND S
MELLY.": LET W=P: RETURN
 800 PRINT '"I CAN SEE...": LET
FL=0: FOR Z=P TO 9: IF O(Z)=LOC
THEN GO TO 830: REM THEN PRINT "
A ";O$(Z): LET FL=1
 810 NEXT Z: IF NOT FL THEN PRIN
T "NOTHING"
 820 RETURN
 830 IF Z=4 AND LOC=20 THEN GO T
O 810
 840 PRINT "A ";O$(Z): LET FL=1:
 GO TO 810
 850 IF A$(P)=" " THEN LET A$=A$
(2 TO ): GO TO 850
 860 IF A$(LEN A$)=" " THEN LET
A$=A$( TO LEN A$-P): GO TO 860
 870 LET B$="": FOR Z=P TO LEN A
$: IF A$(Z)=" " THEN LET B$=A$(Z
+P TO ): LET A$=A$( TO Z-P): RET
URN
 880 NEXT Z: RETURN
 900 GO SUB 9600: CLS
 998 LET N=0: LET S=0: LET E=0:
LET W=0: LET U=0: LET D=0
 999 GO SUB LOC*3: PRINT '"EXITS
: ";("NORTH " AND N);("SOUTH " A
ND S);("EAST " AND E);("WEST" AN
D W): GO SUB 800
1000 INPUT INK 5;"WHAT SHALL I D
O?"' LINE A$: IF A$="" THEN GO T
O C
1005 GO SUB 850: PRINT INK 6;A$;
" ";B$: PRINT
```



```
1007 IF LOC=4 AND RND>.88 THEN P
RINT "THE TROLL LEAPT OUT AND SM
ASHED MY SKULL. I AM DEAD.": GO
TO 9000
1008 IF LOC=39 AND NOT DOG AND R
ND>.9 THEN PRINT "THE DOG ATTACK
ED ME AND KILLED  ME.": GO TO 90
00
1010 IF A$="N" AND N THEN LET LO
C=LOC-F: GO TO C-P-P
1020 IF A$="S" AND S THEN LET LO
C=LOC+F: GO TO C-P-P
1030 IF A$="E" AND E THEN LET LO
C=LOC+P: GO TO C-P-P
1040 IF A$="W" AND W THEN LET LO
C=LOC-P: GO TO C-P-P
1045 IF A$="W" OR A$="E" OR A$="
N" OR A$="S" THEN PRINT "I CAN'T
 GO IN THAT DIRECTION.": GO TO C
1050 IF A$="I" THEN GO SUB 2000:
 GO TO C
1060 IF A$="LOOK" THEN CLS : GO
TO C-P-P
1070 IF A$="D" THEN GO TO 3000
1080 IF A$="U" THEN GO TO 3100
1090 IF A$="GET" OR A$="TAKE" TH
EN GO TO 2100
1100 IF A$="DROP" OR A$="PUT" TH
EN GO TO 2200
1110 IF A$="KILL" THEN GO TO 230
0
1120 IF A$="OPEN" THEN GO TO 240
0
1130 IF A$="CROSS" THEN GO TO 25
00
1140 IF A$="UNLOCK" THEN GO TO 2
500
1150 IF A$="FEED" THEN GO TO 270
0
1160 IF A$="BUY" THEN GO TO 2600
1170 IF A$="SAVE" THEN SAVE "AD"
 LINE 998: PRINT "VERIFY...": VE
RIFY "AD": GO TO C
1180 IF A$="QUIT" THEN GO TO 897
0
1190 IF A$="HELP" THEN PRINT "TH
E TROLL WON'T LET YOU GO EVEN IF
 YOU ASK NICELY...": GO TO C
1200 IF A$="DIG" THEN GO TO 2900
1300 PRINT "I DON'T UNDERSTAND."
: GO TO C
```

```
2000 PRINT '"I HAVE THE FOLLOWIN
G...": LET FL=0: FOR Z=1 TO 9: I
F O(Z)=0 THEN PRINT "A ";O$(Z):
LET FL=P
2010 NEXT Z: IF NOT FL THEN PRIN
T "NOTHING."
2020 RETURN
2100 IF B$="" THEN INPUT INK 5;(
A$);" WHAT ??"' LINE B$: GO TO 2
100
2105 IF OB=4 THEN PRINT "I CAN'T
 CARRY ANY MORE.": GO TO C
2110 FOR Z=P TO 9: IF O$(Z)( TO
LEN B$)=B$ AND O(Z)=LOC THEN GO
TO 2130
2120 NEXT Z: PRINT "I CAN'T DO T
HAT.": GO TO C
2130 IF B$="DOG BONE" AND LOC=6
THEN PRINT "THAT WOULD BE STEALI
NG!": GO TO C
2140 PRINT "OK..": LET O(Z)=0: L
ET OB=OB+P: GO TO C
2200 IF B$="" THEN INPUT INK 5;"
WHAT SHALL I ";(A$);" ??"' LINE
B$: GO TO 2200
2210 FOR Z=P TO 9: IF O$(Z)( TO
LEN B$)=B$ AND O(Z)=0 THEN PRINT
 "OK..": LET O(Z)=LOC: LET OB=OB
-P: GO TO C
2220 NEXT Z: PRINT "I HAVE NO ";
B$;"!": GO TO C
2300 IF B$="" THEN INPUT INK 5;"
KILL WHAT?"' LINE B$: GO TO 2300
2310 IF B$<>"TROLL" THEN PRINT "
I'M NOT THAT SORT OF PERSON!": G
O TO C
2315 IF LOC<>4 THEN PRINT "I CAN
'T SEE A TROLL!": GO TO C
2320 IF O(9) THEN PRINT "YOU HAV
E NOTHING TO KILL THE    TROLL W
ITH.": GO TO C
2330 PRINT '"WELL DONE, YOU HAV
E KILLED THE   TROLL AND EARNT YO
UR FREEDOM.": GO TO 9000
2400 IF B$="" THEN INPUT INK 5;"
WHAT SHALL I OPEN?"' LINE B$: GO
 TO 2400
2410 IF B$<>"DOOR" THEN PRINT "T
HAT I CANNOT DO.": GO TO C
2420 IF LOC<>39 THEN PRINT "WHAT
 DOOR?": GO TO C
```



```
2430 IF NOT DO THEN PRINT "IT'S
LOCKED.": GO TO C
2440 PRINT "OK..": LET LOC=LOC+P
: GO TO C-P-P
2500 IF B$="" THEN INPUT INK 5;"
WHAT DO YOU WANT TO CROSS?"' LIN
E B$: GO TO 2500
2505 IF B$<>"RIVER" THEN PRINT "
I'M NOT THAT CLEVER, MATE!": GO
TO C
2507 IF LOC<>31 AND LOC<>32 THEN
 PRINT "SHOW ME A RIVER, AND I'L
L TRY.": GO TO C
2510 IF O(2) THEN PRINT "I'M GOI
NG TO NEED SOMETHING TO  HELP ME
.": GO TO C
2520 PRINT "OK..": LET LOC=LOC+(
LOC=31)-(LOC=32): GO TO C-P-P
2600 IF B$="" THEN INPUT INK 5;"
PLEASE CONTINUE YOUR COMMAND."'
LINE B$: GO TO 2600
2605 IF B$<>"DOOR" THEN PRINT "T
HAT COULD TAKE ME A WHILE.": GO
TO C
2610 IF LOC<>39 THEN PRINT "I CA
N'T SEE A DOOR!": GO TO C
2620 IF DO THEN PRINT "YOU FOOL,
 IT'S ALREADY UNLOCKED!": GO TO
C
2625 IF O(4) THEN PRINT "YOU HAV
E NO KEY...": GO TO C
2630 PRINT "OK..": LET DO=P: GO
TO C
2700 IF B$<>"DOG" THEN PRINT "I
DON'T THINK IT'S WORTH IT.": GO
TO C
2710 IF LOC<>32 THEN PRINT "WHAT
 DOG?": GO TO C
2720 IF O(P) THEN PRINT "WHAT WI
TH, YOUR FINGERS?": GO TO C
2730 PRINT '"THE DOG HAS FALLEN
ASLEEP.": LET DOG=P: LET S=P: LE
T O(P)=LOC: LET OB=OB-P: GO TO C
2800 IF O(F) THEN PRINT "YOU HAV
E NO MONEY.": GO TO C
2810 IF B$="" THEN INPUT INK 5;"
ANYTHING IN PARTICULAR?"' LINE B
$: GO TO 2810
2820 IF B$="NO" THEN PRINT "WATC
H IT, MATE! I'LL LOSE MY    PATI
ENCE!": GO TO C
```

```
2830 IF B$<>"DOG BONE" THEN PRIN
T "WHY BOTHER IF IT'S FREE?": GO
 TO C
2840 IF LOC<>6 THEN PRINT "THERE
'S NO DOG BONE HERE.": GO TO C
2845 IF OB=4 THEN PRINT "I CAN'T
 CARRY ANY MORE.": GO TO C
2850 PRINT "OK.": LET O(P)=0: L
ET OB=OB+P: GO TO C
2900 IF LOC<>20 THEN PRINT "I CA
N'T DO THAT YET.": GO TO C
2910 IF O(4)<>LOC THEN PRINT "TH
ERE'S NOTHING THERE.": GO TO C
2920 PRINT "I HAVE FOUND A KEY."
: GO TO C
3000 IF LOC<>25 THEN PRINT "DOWN
 WHERE?": GO TO C
3010 IF O(5) THEN PRINT "I'M NOT
 JUMPING, I'LL BREAK MY  NECK!":
 GO TO C
3020 PRINT "OK.": LET LOC=LOC-F
: GO TO C-P-P
3100 IF LOC<>18 THEN PRINT "I CA
N'T DO THAT YET.": GO TO C
3110 IF O(5) THEN PRINT "HOW DID
 YOU GET ME DOWN IN THE  FIRST P
LACE?": GO TO C
3120 PRINT "OK.": LET LOC=LOC+F
: GO TO C-P-P
8970 CLS : INPUT "ARE YOU SURE -
 Y/N:"; LINE A$
8980 IF A$="Y" OR A$="YES" THEN
GO TO 9000
8990 GO TO C
9000 INPUT "DO YOU WANT ANOTHER
GO?     (Y/N)"; LINE A$
9010 IF A$="Y" OR A$="YES" THEN
RUN 900
9020 IF A$="N" OR A$="NO" THEN S
TOP
9030 GO TO 9000
9500 LET O=PI-PI: LET P=PI/PI
9510 LET F=VAL  "7": LET LOC=VAL
"24": LET OB=0
9520 LET DO=0: LET DOG=0: LET C=
VAL "1000": DIM O$(9,8): DIM O(9
)
9530 FOR Z=P TO INT (PI*PI): REA
D O$(Z),Y$: LET O(Z)=VAL Y$
9540 NEXT Z
```



```
9550 DATA "DOG BONE","6","PLANK"
,"11","RING","14","KEY","20","RO
PE","23","ROCK","28","WALLET","3
0","BOOK","39","GUN","40"
9600 RETURN
```

This next program is the loader for THE CITY:

```
  10 REM         The City
  20 REM   An Adventure Game
  30 REM By Neil Pellinacci 1983
  40 REM        16K Spectrum
  50 REM
  60 REM
  70 BORDER 0: PAPER 0: INK 7: B
RIGHT 1
  80 CLEAR 32767: CLS
  90 POKE 23609,100: POKE 23658,
8: POKE 23624,71
  95 GO SUB 200
 100 LOAD ""
 200 DIM A$(32): PAPER 6: INK 0:
 LET A$="                STOP THE TAPE
": LET L=9: GO SUB 900
 210 PAUSE 100: PAPER 0: CLS
 220 LET A$="                THE CIT
Y": LET L=3: INK 6: GO SUB 900
 230 LET A$="AN ADVENTURE FOR TH
E 2K SPECTRUM": LET L=5: GO SUB
900
 240 LET A$="WRITTEN BY NEIL PEL
LINACCI  1983": INK 5: LET L=8:
GO SUB 900
 250 LET A$="   PRESS ANY KEY TO
 CONTINUE": INK 7: LET L=16: GO
SUB 900
 260 LET L=1: PAUSE 1: PAUSE 0:
CLS
 265 FOR Z=-20 TO 30 STEP 5: BEE
P .01,Z: BEEP .01,Z/10: BEEP .01
,Z-10: BEEP .01,Z+10: NEXT Z
 270 RESTORE
 280 READ A$: IF A$(1)="*" THEN
RETURN
 290 IF A$(1)="+" THEN PRINT AT
21,0; INK 6;"   PRESS ANY KEY TO
 CONTINUE": PAUSE 1: PAUSE 0: CL
S : BEEP .1,0: BEEP .1,10: BEEP
```

```
.1,20: BEEP .3,30: LET L=1: GO T
O 280
 300 GO SUB 900: LET L=L+2: GO T
O 280
 400 DATA " IN THIS ADVENTURE GA
ME YOU MUST "
 410 DATA "RECOVER SOME STOLEN M
ONEY. THIS"
 420 DATA "MONEY IS, HOWEVER, ST
ORED IN THE"
 430 DATA "BANK VAULT. THE BANK
IS OWNED BY"
 440 DATA "THE GANGSTER WHO STOL
E THIS "
 450 DATA "MONEY,"
 460 DATA " YOU MUST BREAK INTO
THE BANK,"
 470 DATA "AND STEAL THE MONEY B
ACK. "
 480 DATA "HOWEVER, THIS WILL NO
T BE EASY,"
 490 DATA "AS YOU WILL APPEAR TO
 BE A THIEF"
 500 DATA "+"
 510 DATA " TO COMBAT THIS, YOU
HAVE "
 520 DATA "ARRANGED A BOMB SCARE
, AND SO"
 530 DATA "THE CITY IS NEARLY EM
PTY. YOU "
 540 DATA "MUST STILL BE CAREFUL
 THAT THE"
 550 DATA "POLICE DON'T GET THE
WRONG IDEA."
 560 DATA "THEY WILL NOT KNOW AB
OUT YOUR"
 570 DATA "SECRET MISSION."
 580 DATA " WHEN YOU HAVE COLLEC
TED THE"
 590 DATA "MONEY, YOU MUST FIND
YOUR"
 600 DATA "CONTACT. HE WILL TAKE
 YOU AWAY."
 610 DATA "+"
 620 DATA " TO PLAY, ENTER YOUR
WISHES IN"
 630 DATA "PLAIN ENGLISH, USING
VERBS AND"
 640 DATA "NOUNS. THE EXCEPTION
IS 'N' TO"
 650 DATA "GO NORTH, 'S' TO GO S
OUTH, ETC."
```



```
 660 DATA " IF YOU WANT TO STOP,
 TYPE "
 670 DATA "'STOP' OR 'QUIT'. 'SA
VE' WILL"
 680 DATA "SAVE THE CURRENT GAME
 ON TAPE,"
 690 DATA "SO THAT YOU CAN FINIS
H IT LATER."
 700 DATA "+"
 710 DATA "","","","","          S
TART THE TAPE"
 720 DATA "*"
 800 PAPER 0: INK 7: STOP
 900 FOR Z=32 TO 1 STEP -1: PRIN
T AT L,0;A$(Z TO ): NEXT Z: RETU
RN
```

Once you have that in place, you can enter and save THE CITY program itself:

```
   9 PRINT '"I AM IN A TUBE STAT
ION. THERE ISA TRAIN WAITING TO
LEAVE. THE    TAXI WAITS OUTSIDE.
": RETURN
  12 PRINT '"I AM STANDING BEHIN
D A BUTCHER'SSHOP. I CAN SEE A B
UILDING SITE TO THE EAST.": LET
E=P: LET S=P: RETURN
  15 PRINT '"A LARGE STEEL GIRDE
R FELL ON ME.I AM DEAD.": GO TO
9000
  21 PRINT '"I AM OUTSIDE THE BA
NK. THE DOOR IS BEFORE ME. THE H
IGH STREET ISTO THE EAST.": LET
E=P: LET D01=0: RETURN
  24 PRINT '"I AM IN THE HIGH ST
REET, WHICH   IS DESERTED, APART
FROM A TAXI. A MARKET IS TO THE
SOUTH.": LET E=P: LET W=P: LET S
=P: RETURN
  27 PRINT '"I AM STANDING IN A
LARGE SHOP.   I CAN SEE NO PEOPLE
. THERE IS A CAFE TO THE EAST.":
 LET N=P: LET S=P: LET W=P: LET
E=P: RETURN
  30 PRINT '"I AM STANDING IN A
SMALL,        DESERTED CAFE. THE
LARGE SHOP ISTO THE WEST.": LET
W=P: RETURN
```

```
  36 PRINT '"I AM INSIDE THE BAN
K. IT IS VERYQUIET. ";: IF GU TH
EN PRINT "I CAN SEE A GUARD WITH
   HIS BACK TO ME.";
  37 PRINT : LET N=P: LET S=NOT
GU: RETURN
  39 PRINT '"I AM IN A STREET MA
RKET. A       POLICE STATION IS T
O THE EAST,   AND A BUILDING SITE
 TO THE       SOUTH."
  40 LET N=P: LET S=P: LET E=P:
RETURN
  42 PRINT '"I HAVE ARRIVED AT T
HE POLICE     STATION. IT IS NOT
DESERTED..."
  43 IF NOT O(7) THEN PRINT "A P
OLICEMAN SEES THE MONEY AND  ARR
ESTS ME.": GO TO 9000
  44 PRINT "A POLICEMAN TAKES WH
AT I HAVE     COLLECTED AND RETURN
S THEM TO     WHERE I FOUND THEM,"
: LET N=P: LET W=P: GO TO 700
  45 PRINT '"I AM IN A SMALL GLO
OMY TUBE      STATION. THERE IS A
 HOLE IN THE WALL TO THE SOUTH."
: LET S=P: RETURN
  48 PRINT '"I AM STANDING IN FR
ONT OF THE    VAULT DOOR, WHICH I
S LOCKED."
  49 IF VC THEN PRINT "THERE IS
A HOLE IN IT."
  50 LET E=P: LET S=VC: RETURN
  51 PRINT '"I CAN SEE TWO DOORS
, ONE TO THE WEST, ONE TO THE SO
UTH. THE      VAULT ENTRANCE IS T
O THE NORTH.": LET N=P: LET S=P:
 LET W=P: RETURN
  54 PRINT '"I AM ON A BUILDING
SITE, WHICH   IS SURROUNDED BY HI
GH WALLS.": LET N=P: RETURN
  57 PRINT '"I'M NOW STANDING IN
 THE FOYER OFA LARGE HOTEL. BEFO
RE ME IS THE RECEPTION DESK.": L
ET E=P: RETURN
  60 PRINT '"I AM IN THE BASEMEN
T OF A LARGE BUILDING.": LET N=P
: LET W=P: RETURN
  63 PRINT '"I AM INSIDE THE VAU
LT. TO REACH THE MONEY, I HAVE T
O ENTER THE   KEYPANEL CODE. THER
E ARE 9       BUTTONS IN FRONT OF
```



```
ME. I MUST  PRESS THE RIGHT ONE
."
  64 LET N=P: RETURN
  65 PRINT '"A WALL HAS APPEARED
BEHIND ME.  I AM TRAPPED.": GO
TO 9000
  78 PRINT '"I AM INSIDE THE MAI
N VAULT ROOM.": LET N=P: RETURN
 600 IF A$(P)=" " THEN LET A$=A$
(2 TO ): GO TO 600
 610 IF A$(LEN A$)=" " THEN LET
A$=A$( TO LEN A$-P): GO TO 610
 620 LET B$="": FOR Z=P TO LEN A
$: IF A$(Z)=" " THEN LET B$=A$(Z
+P TO ): LET A$=A$( TO Z-P): RET
URN
 630 NEXT Z: RETURN
 700 RESTORE 6000: FOR Z=P TO 7:
 READ A$,B$: IF O(Z)=0 THEN LET
O(Z)=VAL B$
 710 NEXT Z: LET OB=0: RETURN
 800 PRINT '"EXITS: ";("NORTH "
AND N);("SOUTH " AND S);("EAST "
 AND E);("WEST" AND W)
 810 PRINT '"I CAN SEE.."
 820 LET FL=0: FOR Z=P TO 7: IF
O(Z)=LOC THEN PRINT "A ";O$(Z):
LET FL=P
 830 NEXT Z: IF NOT FL THEN PRIN
T "NOTHING INTERESTING."
 840 RETURN
 900 GO SUB 6000: CLS
 999 LET N=0: LET S=0: LET E=0:
LET W=0: GO SUB LOC*9: GO SUB 60
0
1000 INPUT INK 5;"WHAT SHALL I D
O NOW?"' LINE A$: IF A$="" THEN
GO TO C
1010 IF LOC=12 AND RND>.9 AND GU
 THEN PRINT '"THE GUARD TURNED,
SAW ME, THEN  SHOT ME. I AM DEAD
.": GO TO 9000
1050 GO SUB 600: PRINT ' INK 6;A
$;" ";B$
1060 IF A$="N" AND N THEN LET LO
C=LOC-F: GO TO C-P
1070 IF A$="S" AND S THEN LET LO
C=LOC+F: GO TO C-P
1080 IF A$="E" AND E THEN LET LO
C=LOC+P: GO TO C-P
1090 IF A$="W" AND W THEN LET LO
C=LOC-P: GO TO C-P
```

```
1100 IF A$="N" OR A$="S" OR A$="
E" OR A$="W" THEN PRINT "I CAN'T
 GO IN THAT DIRECTION.": GO TO C
1110 IF A$="GET" OR A$="TAKE" TH
EN GO TO 2000
1120 IF A$="DROP" OR A$="PUT" TH
EN GO TO 2100
1130 IF A$="I" THEN GO TO 2200
1140 IF A$="LOOK" THEN CLS : GO
TO C-P
1150 IF A$="HELP" THEN GO TO 230
0
1160 IF A$="BOARD" OR A$="CATCH"
 THEN GO TO 2400
1170 IF A$="UNLOCK" THEN GO TO 2
500
1180 IF A$="OPEN" THEN GO TO 260
0
1190 IF A$="D" THEN GO TO 2700
1200 IF A$="QUIT" OR A$="STOP" T
HEN GO TO 6900
1210 IF A$="READ" OR A$="EXAMINE
" THEN GO TO 2800
1220 IF A$="KILL" THEN GO TO 290
0
1230 IF A$="CUT" THEN GO TO 3000
1240 IF A$="PRESS" OR A$="PUSH"
THEN GO TO 3100
1250 IF A$="SAVE" THEN SAVE "CIT
Y" LINE 999: PRINT "VERIFY...":
VERIFY "": GO TO C
1260 IF A$="SAY" THEN GO TO 3200
1300 PRINT "I DON'T UNDERSTAND."
: GO TO C
2000 IF B$="" THEN INPUT INK 5;"
GET WHAT ??"' LINE B$: GO TO 200
0
2002 IF LEN B$>13 THEN PRINT "I
CAN'T": GO TO C
2005 IF OB=3 THEN PRINT "I CAN'T
 CARRY ANY MORE.": GO TO C
2010 FOR Z=P TO 7: IF O$(Z)( TO
LEN B$)=B$ AND O(Z)=LOC THEN LET
 O(Z)=0: LET OB=OB+1: PRINT "OK.
.": GO TO C
2020 NEXT Z: PRINT "I SEE NO ";B
$;"!": GO TO C
2100 IF B$="" THEN PRINT A$;" WH
AT?": GO TO C
2110 IF LEN B$>13 THEN PRINT "I
CAN'T.": GO TO C
```



```
2120 FOR Z=P TO 7: IF O$(Z)( TO
LEN B$)=B$ THEN PRINT "OK..": LE
T O(Z)=LOC: LET OB=OB-1: GO TO C
2130 NEXT Z: PRINT "I HAVE NO ";
B$;"!": GO TO C
2200 PRINT "I HAVE THE FOLLOWING
.": LET FL=0: FOR Z=P TO 7: IF
O(Z)=0 THEN PRINT "A ";O$(Z): LE
T FL=P
2210 NEXT Z: IF NOT FL THEN PRIN
T "NOTHING."
2220 GO TO C
2300 IF LOC=17 THEN PRINT "ASK T
HE VILLAGE PEOPLE.": GO TO C
2310 PRINT "FOLLOW THE OLD LONDO
N - BRISTOL ROAD!": GO TO C
2400 IF B$="" THEN PRINT A$;" WH
AT ?": GO TO C
2410 IF B$<>"TRAIN" AND B$<>"TAX
I" THEN PRINT "I CAN'T.": GO TO
C
2420 IF B$="TRAIN" AND LOC<>3 AN
D LOC<>15 THEN GO TO 2490
2430 IF B$="TAXI" AND LOC<>3 AND
 LOC<>8 THEN GO TO 2490
2440 IF B$="TRAIN" THEN GO TO 24
70
2450 IF O(2) THEN PRINT "I HAVE
NO MONEY FOR THE FARE.": GO TO C
2460 PRINT "THE TAXI DRIVES OFF
WITH ME      INSIDE.": LET LOC=LO
C+(F AND LOC=3)-(F AND LOC=8): G
O TO C-P
2470 IF O(3) THEN PRINT "I NEED
A TICKET!!": GO TO C
2480 PRINT "I GET IN AND THE TRA
IN MOVES     AWAY.": LET LOC=(3 A
ND LOC=15)+(15 AND LOC=3): GO TO
 C-P
2490 PRINT "I CAN'T DO THAT YET.
": GO TO C
2500 IF B$="" THEN PRINT "WHAT S
HALL I UNLOCK?": GO TO C
2510 IF B$<>"DOOR" THEN PRINT "I
 CAN'T DO THAT.": GO TO C
2520 IF LOC<>7 THEN PRINT "WHAT
DOOR?": GO TO C
2530 IF DO THEN PRINT "YOU IDIOT
, IT'S ALREADY          UNLOCKED!
!": GO TO C
2540 IF O(5) THEN PRINT "I NEED
THE KEY.": GO TO C
```

```
2550 PRINT "OK..": LET DO=P: GO
TO C
2600 IF B$="" THEN PRINT "OPEN W
HAT?": GO TO C
2610 IF B$<>"DOOR" THEN PRINT "I
 CAN'T.": GO TO C
2620 IF LOC<>7 THEN PRINT "I CAN
'T SEE A DOOR!": GO TO C
2630 IF NOT DO THEN PRINT "IT'S
LOCKED.": GO TO C
2640 LET DO1=P: PRINT "OK.."''"I
 CAN SEE SOME STAIRS GOING     D
OWN.": GO TO C
2700 IF LOC<>7 THEN GO TO 2490
2710 IF NOT DO1 THEN PRINT "THER
E'S A DOOR IN THE WAY!": GO TO C
2720 PRINT "OK..": LET LOC=LOC+F
: GO TO C-P
2800 IF B$="" THEN INPUT INK 5;(
A$);" WHAT?"' LINE B$: GO TO 280
0
2810 IF B$<>"CLUE" THEN PRINT "I
 CAN'T DO THAT!": GO TO C
2820 IF O(6) THEN PRINT "I'M NOT
 CARRYING A CLUE.": GO TO C
2830 PRINT "THE CLUE IS WRITTEN
ON A4-SIZED PAPER. IT SAYS: 'RUB
 A DUB       DUB...'": GO TO C
2900 IF B$="" THEN PRINT "KILL W
HAT?": GO TO C
2910 IF B$<>"GUARD" THEN PRINT "
I REFUSE ON MORAL GROUNDS.": GO
TO C
2920 IF LOC<>12 THEN PRINT "I CA
N'T SEE A GUARD!": GO TO C
2930 IF O(P) THEN PRINT "I HAVE
NO WEAPON.": GO TO C
2940 PRINT "OK..": LET GU=0: GO
TO C
3000 IF B$="" THEN INPUT INK 5;"
WHAT SHALL I CUT?"' LINE B$: GO
TO 3000
3010 IF B$<>"VAULT DOOR" THEN PR
INT "I CAN'T DO THAT.": GO TO C
3020 IF LOC<>16 THEN PRINT "I SE
E NO VAULT DOORS HERE!": GO TO C
3030 IF O(4) THEN PRINT "I HAVE
NOTHING TO CUT IT WITH.": GO TO
C
3040 PRINT "OK..": LET VC=P: GO
TO C-P
```



```
3100 IF LOC<>21 THEN PRINT "THER
E'S NOTHING HERE TO ";A$;"!": GO
 TO C
3110 IF LEN B$<>P THEN PRINT "I
NEED A NUMBER BETWEEN 1 AND 9.":
 GO TO C
3120 IF B$<"1" OR B$>"9" THEN PR
INT "THAT'S NOT A NUMBER!!": GO
TO C
3130 LET Z=VAL B$: IF Z<>3 THEN
LET LOC=LOC+P: GO TO C-P
3140 LET LOC=LOC+F: PRINT "THE I
RON DOOR SLIDES BACK          SLOWL
Y...": GO TO C-P
3200 IF B$="" THEN PRINT "SAY WH
AT?": GO TO C
3210 IF B$<>"PASSWORD" THEN PRIN
T "I CAN'T SAY THAT!": GO TO C
3220 IF LOC=15 AND NOT O(7) THEN
 PRINT '"I HAVE FOUND THE CONTA
CT, AND    HAVE SUCCESSFULLY COMP
LETED MY   MISSION. THE CONTACT W
AS THE     TRAIN DRIVER.": GO TO
9000
3225 IF LOC=3 OR LOC=6 THEN PRIN
T "THE TAXI DRIVER SAYS SOMETHIN
G   VERY RUDE TO ME.": GO TO C
3230 PRINT '"NOBODY APPEARS TO W
ANT TO KNOW.": GO TO C
8000 LET O=PI-PI: LET P=NOT O
8010 LET F=VAL "5": LET LOC=VAL
"13"
8020 LET OB=O: LET DO=O: LET GU=
P: LET VC=O: LET C=VAL "1000"
8030 DIM O(7): DIM O$(7,13)
8040 RESTORE 8000: FOR Z=P TO 7:
 READ O$(Z),A$: LET O(Z)=VAL A$:
 NEXT Z
8050 DATA "KNIFE","4","WALLET","
9","TUBE TICKET","10","BLOWTORCH
","16","KEY","13","WRITTEN CLUE"
,"20","CASE OF MONEY","26"
8060 PRINT '''TAB 11; INK 6;"THE
CITY"''"   WRITTEN BY NEIL PELLIN
ACCI"''"     PRESS ANY KEY TO STA
RT"
8070 PAUSE P: PAUSE O: FOR Z=50
TO -20 STEP -5: BEEP .01,Z: BEEP
 .01,Z-3: BEEP .01,Z+6: NEXT Z
8080 RETURN
```

```
8900 INPUT "ARE YOU SURE?
          Y/N"; LINE A$: IF A$=""
 THEN GO TO 8900
8910 IF A$(P)="Y" THEN GO TO 900
0
8920 GO TO 0
9000 INPUT "DO YOU WANT TO PLAY
AGAIN?    Y/N"; LINE A$: IF A$=""
 THEN GO TO 9000
9010 IF A$(P)="N" THEN STOP
9020 RUN 900
```


Doors of Doom

This is a masterpiece program by Malcolm Young which will well reward the time it will take you to type the whole thing in.

This graphic Adventure is based around the maze routine at lines 1000 to 1400. The routine starting at line 2000 handles the graphics.

When Malcolm started writing the program, he had intended it to be simply a 3D Maze program. However, once he had that up and running he decided he could make it a lot more interesting and the result is here for you to see. Full instructions are included. A word of warning: The program generates mazes which are extremely challenging, and difficult to solve. Do not run this program if you want a game you can master in an hour or so.

```
   2 REM ©1983 Malcolm Young
  10 DEF FN r(x)=INT (RND*x)+1
  15 LET set=0
  30 GO SUB 9000: REM instructio
ns
  40 BORDER 4: PAPER 5: INK 0: C
LS
  50 GO SUB 1000: REM generate M
aze
  60 GO SUB 8000: REM initialise
 Variables
  70 GO TO 230
  80 NEXT a
  90 REM main driving routine
 100 LET k$=INKEY$: IF k$="" OR
k$="a" THEN GO TO 100
 110 LET f=f+(k$="r")-(k$="l")+2
*(k$="b")
```

```
 120 LET f=f+4*(f<1)-4*(f>4)
 130 IF k$="m" THEN GO SUB 5000
 135 IF k$="f" THEN LET m$(py,px
)="""": LET l(11)=l(11)-l(13): I
F l(11)<0 THEN GO TO 500
 140 LET px=px+((k$="f")*((f=2)-
(f=4)))
 150 LET py=py+((k$="f")*((f=3)-
(f=1)))
 155 IF m$(py,px)="#" THEN LET s
w=1: LET px=px-(f=2)+(f=4): LET
py=py-(f=3)+(f=1)
 160 IF m$(py,px)=CHR$ 128 THEN
GO SUB 6000
 170 IF m$(py,px)>"." THEN GO TO
 2600
 175 IF g1=py AND g2=px THEN GO
TO 600
 180 IF k$="e" THEN LET re=1-re:
 LET k$="f": GO TO 140
 190 IF k$="y" THEN GO SUB 7000
 200 IF l(7) AND k$="d" THEN GO
SUB 7200
 210 IF l(8) AND k$="t" THEN GO
SUB 6500
 220 IF k$="s" THEN INPUT "Do yo
u want to save this game?"; LINE
 a$: IF a$="y" THEN GO SUB 7500:
 CLS
 230 CLS
 240 PRINT #0; INVERSE 1;"NORTH"
 AND f=1;"EAST" AND f=2;"SOUTH"
AND f=3;"WEST" AND f=4; INVERSE
0;" "; FLASH 1;"REPEAT" AND re;
 FLASH 0;TAB 15;"Energy=";l(11)
 250 BEEP .1,10: GO SUB 2000
 260 IF INKEY$="" AND (re*r) THE
N LET k$="f": GO TO 135
 270 IF sw THEN PRINT AT 10,11;
FLASH 1;"SOLID WALL": BEEP 1,-10
: LET sw=NOT sw
 280 GO TO 100
 500 REM end of game
 510 CLS
 520 PRINT AT 10,0;"You have sta
rved to death..."
 530 PRINT '"Your bones litter t
he maze with those of many other
s."
 540 GO TO 4638
 600 REM finish
 605 PAPER 0: INK 7: CLS
```



```
 610 GO SUB 5160: PRINT AT 4,0;
 620 FOR i=40 TO 45 STEP .5: BEE
P .25,i: PRINT "Well done! You m
ade it!": NEXT i
 625 FLASH 1
 630 PRINT INK 6;"You accomplish
ed the task with"
 635 INK 4
 640 IF l(10)>200 THEN PRINT "a
great amount of skill."
 650 IF l(10)<200 AND l(10)>100
THEN PRINT "well deserved merit"
 655 IF l(10)<100 THEN PRINT "a
considerable amount of luck"
 660 GO TO 4642
1000 REM Maze Generator
1002 PAPER 0: INK 5: CLS : PRINT
 "The maze can take a considerab
letime to construct so it is
  advisable to have the larger
  mazes  stored on cassette."
1004 PRINT '''"Please enter the d
imensions of  your maze so that
it has a widthbetween 30 AND 100
 and a length between 20 AND 100
"
1006 INPUT "Would you like to RE
STORE a gameanswer (y/n)? ;"; LI
NE a$
1008 IF a$="y" THEN GO SUB 7600:
 RETURN
1010 INPUT "Enter Width of Maze:
";w: LET w=INT (w/2): IF w<15 OR
 w>50 THEN GO TO 1010
1020 INPUT "Enter Length of Maze
:";l: LET l=INT (l/2): IF l<10 O
R l>50 THEN GO TO 1020
1030 LET s=INT (w*l/3): LET s=IN
T (s+s/10): DIM l(s): DIM d(s)
1032 LET s=s+INT (RND*s/3): LET
t=INT (s/60): LET t1=(s-t*60)/10
0: LET t=t+t1
1033 POKE 23674,0: POKE 23673,0:
 POKE 23672,0
1035 PRINT '''"Please expect to
wait at least  approximately ";t
;" minutes"
1040 LET m=w*2+1: LET n=l*2+1: D
IM m$(n,m)
```

```
1050 FOR i=1 TO m: LET m$(1,i)="
#": LET m$(n,i)="#": NEXT i
1060 FOR i=2 TO n-1: LET m$(i,1)
="#": LET m$(i,m)="#": NEXT i
1070 LET nn=2*INT (RND*(l-2))+3
1080 LET mm=2*INT (RND*(w-2))+3
1090 LET m$(nn,mm)="#": LET k=0
1100 IF m$(nn-2,mm)=" " THEN GO
TO 1210
1110 IF m$(nn+2,mm)=" " THEN GO
TO 1210
1120 IF m$(nn,mm-2)=" " THEN GO
TO 1210
1130 IF m$(nn,mm+2)=" " THEN GO
TO 1210
1140 IF k-1=0 THEN GO TO 1420
1150 IF d(k)=1 THEN LET nn=nn+l(
k)
1160 IF d(k)=2 THEN LET nn=nn-l(
k)
1170 IF d(k)=3 THEN LET mm=mm+l(
k)
1180 IF d(k)=4 THEN LET mm=mm-l(
k)
1190 LET k=k-1
1200 GO TO 1100
1210 LET l1=2*INT (RND*3)+2
1220 IF l1=8 THEN GO TO 1210
1230 LET d1=INT (RND*4)+1
1240 IF d1>2 THEN GO TO 1280
1250 LET s=-1+2*(d1=2)
1260 LET t=0
1270 GO TO 1300
1280 LET s=0
1290 LET t=-1+2*(d1=4)
1300 FOR i=2 TO l1 STEP 2
1310 IF m$(nn+s*i,mm+t*i)="#" TH
EN GO TO 1210
1320 NEXT i
1330 FOR i=1 TO l1
1340 LET m$(nn+s*i,mm+t*i)="#"
1350 NEXT i
1360 LET nn=nn+s*l1
1370 LET mm=mm+t*l1
1380 LET k=k+1
1390 LET l(k)=l1
1400 LET d(k)=d1
1410 GO TO 1100
1420 REM no. of doors
```



```
1425 LET nd=INT ((l(4)*l(5)/25):
LET st=nd/14
1430 LET l(4)=m: LET l(5)=n
1435 REM hide objects
1440 FOR o=1 TO 4
1450 FOR i=1 TO st
1460 LET l=FN r(l(5)-3)+1: LET c
=FN r(l(4)-3)+1: LET kn=FN r(5)*
10+o
1470 IF m$(l,c)<>"#" THEN GO TO
1460
1480 LET m$(l,c)=CHR$ (48+kn)
1490 NEXT i
1500 NEXT o
1510 REM hide food and villains
1520 FOR o=5 TO 7 STEP 2
1530 FOR i=1 TO st*3
1540 LET l=FN r(l(5)-3)+1: LET c
=FN r(l(4)-3)+1: LET kn=FN r(5)*
10+o
1550 IF m$(l,c)<>"#" THEN GO TO
1540
1560 LET m$(l,c)=CHR$ (48+kn)
1570 NEXT i: NEXT o
1580 REM hide treasure
1590 FOR i=1 TO st*4
1600 LET l=FN r(l(5)-3)+1: LET c
=FN r(l(4)-3)+1: LET kn=FN r(5)*
10+5
1610 IF m$(l,c)<>"#" THEN GO TO
1600
1620 LET m$(l,c)=CHR$ (48+kn)
1630 NEXT i
1640 LET m$(l(5)/2,l(4)/2)=CHR$
(48+FN r(5)*10+8)
1650 REM hide keys
1660 FOR i=1 TO 4*st
1670 LET l=FN r(l(5)-4)+2: LET c
=FN r(l(4)-4)+2
1680 IF m$(l,c)<>" " THEN GO TO
1670
1690 LET m$(l,c)=CHR$ 128
1700 NEXT i
1705 LET set=1
1710 LET time=(PEEK 23672+256*PE
EK 23673+65536*PEEK 23674)
1720 LET time=time/50: BEEP 2,10
```

```
1730 INPUT "This maze took ";(ti
me);" seconds to  generate,would
 you like to save it for future
 use?";a$
1740 IF a$(1)="y" THEN GO SUB 75
00
1750 CLS : RETURN
2000 REM display view
2010 BORDER 1: PAPER 0: BRIGHT 1
2020 LET x=24: LET y=16: REM wal
l angle gradiant
2030 FOR i=1 TO 6
2040 LET ly=py+((f=3)-(f=1))*i
2050 LET lx=px+((f=2)-(f=4))*i
2060 IF m$(ly,lx)<"#" OR m$(ly,l
x)=CHR$ 128 THEN NEXT i
2065 LET dt=(m$(ly,lx)=" ")
2070 LET vw=i-1-(i>6): REM maxim
um viewing distance
2075 IF NOT vw AND m$(ly,lx)>"#"
 THEN GO SUB 2500
2080 LET wx=vw*x: LET wy=vw*y
2090 FOR d=0 TO vw: REM dividing
 panels
2100 LET h=255-d*x*2
2110 PLOT d*x,d*y: DRAW 0,175-(y
*d)*2: IF d=vw THEN DRAW h,0
2115 PLOT d*x+1,d*y: DRAW 0,175-
(y*d)*2
2120 PLOT 255-d*x,175-d*y: DRAW
0,-(175-y*2*d): DRAW -h,0
2125 PLOT 254-d*x,175-d*y: DRAW
0,-(175-y*2*d)
2130 NEXT d
2140 FOR i=0 TO vw-1: REM draw w
alls
2150 LET ly=py+((f=3)-(f=1))*(i+
1)
2160 LET lx=px+((f=2)-(f=4))*(i+
1)
2170 LET d$=m$(ly-(f=2)+(f=4),lx
-(f=1)+(f=3))
2175 LET lwall=(d$="#"): LET ldr
=(d$>"#")
2180 LET d$=m$(ly+(f=2)-(f=4),lx
+(f=1)-(f=3))
2190 LET rwall=(d$="#"): LET rdr
=(d$>"#")
2195 IF (ldr+rdr) THEN GO SUB 23
00
```



```
2197 GO TO ldr*7+2200
2200 PLOT i*x,i*y+y*NOT lwall: D
RAW x,y*(lwall)
2205 PLOT i*x,175-(i*y+y*NOT lwa
ll): DRAW x,-(y*lwall)
2207 GO TO rdr*20+2210
2210 PLOT 255-i*x,i*y+y*NOT rwal
l: DRAW -x,y*(rwall)
2220 PLOT 255-i*x,175-(i*y+y*NOT
 rwall): DRAW -x,-(y*rwall)
2230 NEXT i
2240 INK 5: LET p=5
2250 LET v=255/p: LET e=(255-2*w
x)/p
2260 FOR l=1 TO p-1
2270 LET h=(wx+e*l)-v*l
2280 PLOT v*l,0: DRAW h,wy
2300 NEXT l
2310 IF dt THEN PRINT FLASH 1; O
VER 1;AT 10,15;"▒▒";AT 11,15;"▒▒
"
2320 IF NOT re THEN RETURN
2325 LET r=1
2330 REM check for repeat
2340 FOR i=0 TO 1
2350 LET ry=py+((f=3)-(f=1))*i:
LET rx=px+((f=2)-(f=4))*i
2360 LET r=r*(m$(ry+(f=2)-(f=4),
rx+(f=3)-(f=1))="#")
2370 LET r=r*(m$(ry-(f=2)+(f=4),
rx-(f=3)+(f=1))="#")
2380 NEXT i: RETURN
2390 REM draw door
2400 LET k=1/(i+1)*5
2410 FOR j=1 TO 7
2420 IF ldr THEN PLOT i*x+j*3,i*
y+j*2
2425 IF ldr THEN DRAW INK 6;0,17
5-(i*y+j*2)*2
2430 IF rdr THEN PLOT 255-(i*x+j
*3),i*y+j*2
2440 IF rdr THEN DRAW INK 6;0,17
5-(i*y+j*2)*2
2450 NEXT j
2460 RETURN
2480 PLOT 100,100+k: DRAW k,0,-P
I*1.75
2490 DRAW k/2.5,-k*1.5: DRAW -k/
2.5*2-k,0,-PI: DRAW k/2.5,k*1.5
2491 RETURN
```

```
2500 REM door
2510 PAPER 6: INK 0: CLS
2520 PRINT AT 10,4;"D O O R"
BEEP .5,10
2530 PAPER 0: INK 5
2540 RETURN
2600 REM enter
2610 IF l(12) THEN GO TO 3000
2620 PRINT AT 2,4; FLASH 1;"Sor
ry,no key-no entry!"
2630 LET px=px-(f=2)+(f=4): LET
py=py-(f=3)+(f=1)
2640 BEEP 1,-20: GO TO 100
2700 REM found centre
2710 IF py<>INT (l(5)/2) AND px<
>INT (l(4)/2) THEN RETURN
2720 CLS : RESTORE 2900
2730 PRINT "The door creaks slow
ly open..."
2740 FOR i=-30 TO -20 STEP .5: B
EEP .1,i: NEXT i
2750 FOR i=1 TO FN r(8): READ t$
: NEXT i
2760 PRINT "And there in the cen
tre of the room lies the ultima
te goal..."'TAB 2;t$;"!"
2770 PRINT '"And now you have to
 take this"'"back to an exit sit
uated"
2775 LET m$(py,px)=" "
2780 LET g1=FN r(l(5)): LET g2=F
N r(l(4))
2785 LET m$(g1,g2)=" "
2790 GO SUB 5500
2810 LET b=FN r(100): LET l(11)=
l(11)+b
2820 PRINT '"You have also found
 a vitamin"'"pill boosting your
energy by ";b
2840 BORDER 0: PAPER 0: INK 5
2850 PAUSE 0
2860 GO TO 230
2900 DATA "a slave for life","a
ticket to Timbuktu","an automati
c house cleaner","the LIONs rugb
y team beating the NZ ALL BLACKS
","the answer to life...42?"
2910 DATA "a free ZX83,4,5,6...",
,"an eternal supply of lager","1
2 million mars bars"
```



```
3000 REM open door
3010 LET j=CODE m$(py,px)-48: LE
T ln=INT (j/10)
3020 PRINT AT 0,3; INK 6;"The do
or has lock no.";ln
3025 IF l(9) THEN PRINT "Your sk
eleton keys opens the door!":
 PAUSE 100: GO TO 3070
3030 IF l(12)<>ln THEN PRINT AT
 1,0; INK 5;"Your key doesn't fi
t,keep going": GO TO 2630
3040 PRINT AT 20,0;"Your key fit
s! Do you want to unlock this
door? (y/n)": BEEP 1,10
3050 IF INKEY$="n" THEN GO TO 20
00
3060 IF INKEY$<>"y" THEN GO TO 3
050
3065 GO SUB 2700
3070 LET fate=j-ln*10: BORDER 4:
 PAPER 6: INK 1: CLS
3073 IF fate=8 THEN GO TO 2700
3075 PRINT "INSIDE,there is....
": BEEP .5,3: BEEP 1,10
3080 GO TO fate*200+3090
3090 REM empty
3100 PRINT AT 3,5;"NOTHING!";AT
5,0;"Some has been here before y
ou"
3110 GO TO (RND>.5)*80+3120
3120 REM lose key
3130 PRINT '"Your key is stuck i
n the lock!"'"Do you want to spe
nd some time trying to get it o
ut?"
3140 PAUSE 0: IF INKEY$<>"y" THE
N GO TO 3200
3150 IF l(6) THEN PRINT "OK,it s
houldn't take a second using y
our laser.": LET l(6)=l(6)-1: LE
T i=1: GO TO 3190
3160 PRINT "OK,I'll wait a momen
t while you try."
3170 PAUSE 70: FOR i=0 TO 10+FN
r(50): BEEP .01,RND*30-15: NEXT
i
3180 IF RND>.7 THEN LET l(13)=l(
13)-1: PRINT "It won't come out,
you'll have to leave it.": GO TO
 3190
```

```
3185 PRINT "You got it!"'"but ";
3190 PRINT "You wasted ";i;" ene
rgy units in"'"your attempt.": L
ET l(11)=l(11)-i
3200 GO TO 4900
3290 REM laser
3300 LET u=FN r(8)+2: PRINT AT 3
,6;q$(1)'"with enough EVERYREADY
 batteriesto last ";u;" uses!"
3310 IF l(6) THEN PRINT AT 3,0;"
Another "
3315 PRINT TAB 12;"▛▚▞▚▞▀▜"'TAB 1
6;"▚▜"
3320 PRINT AT 9,0;"A laser has a
n 'EC' factor of 3 Do you want t
o take it?"''
3330 IF INKEY$="n" THEN GO TO 48
90
3335 IF INKEY$<>"y" THEN GO TO 3
330
3340 IF NOT l(6) THEN LET l(13)=
l(13)+9: GO TO 3360
3350 IF l(6) THEN PRINT "Since y
ou already have a laser  you jus
t have to take the        batteri
es"
3360 LET l(6)=l(6)+u: BEEP .2,3:
 BEEP .2,10: BEEP .2,3: BEEP .7,
10
3370 PRINT '"You are now the pro
ud owner of adisposable 'LEPPY'
laser"'"guaranteed for ";l(6);"
uses!"
3380 GO TO 4900
3490 REM dematerialiser
3500 PRINT AT 3,5;q$(2);AT 3,0;"
Another " AND l(7)
3510 PRINT AT 5,11;"▌";TAB 5;"=▓
▓▓▓██▙"'TAB 12;"▜ ▚"
3520 PRINT '"This enables you to
 disintegratesolid walls in two
blasts and tocharcoal any villai
n you may      meet."
3530 LET u=FN r(6)+2
3540 PRINT '"This one has ";u;"
cartridges"'q$(2);"has an 'EC'"'
"factor of 14"
3550 PRINT '"Do you want to carr
y this along?"
```



```
3560 IF INKEY$="n" THEN GO TO 48
90
3565 IF INKEY$<>"y" THEN GO TO 3
560
3570 CLS : FOR i=0 TO 10: BEEP .
05,i: NEXT i
3580 PRINT FLASH 1;AT 2,9;"CONGR
ATULATIONS !"
3590 PRINT '"You didn't need muc
h convincing for this formidable
 weapon,as    it's the hottest th
ing around inarmourments."
3600 IF l(7) THEN PRINT '"You wo
n't want to carry around  two of
 these so you can have thecartri
dges for free."
3610 PRINT '"To use this weapon,
aim at your  target and squeeze
the key 'd'"
3620 PRINT '"You can use this we
apon ";l(7)+u;" times"
3630 PRINT ' INVERSE 1;"GOVERNME
NT WARNING:THIS WEAPON  CAN BE D
ANGEROUS FOR YOUR HEALTH"
3670 LET l(13)=l(13)+14: LET l(7
)=l(7)+u
3680 GO TO 4900
3690 REM transporter belt
3700 PRINT AT 2,6;q$(3);AT 2,0;"
Another " AND l(6)
3710 PRINT INK 4; PAPER 2;AT 4,1
5;"▛▀▀▜";AT 5,7; INVERSE 1;"+==:↑
:==▌"; FLASH 1;"T"; FLASH 0; INV
ERSE 1;"▌==:↑:==-";AT 6,15;"▀▀▀"
3720 PRINT "This enables you to
hyper-travelto a different part
of the maze but you have no cont
rol where   and if you stop in a
 wall you will be instantly suff
ocated"
3730 LET u=FN r(3)+2
3740 PRINT '"This is another mod
ern"'"disposable product and mus
t be  discarded after ";u;" uses
 or"'"transport failure may occu
r"
3750 PRINT 'q$(3);" has an 'EC'
 factor of 5"
3760 PRINT AT 20,3;"Do you want
to put this on?"
```

```
3770 IF INKEY$="n" THEN GO TO 49
00
3780 IF INKEY$<>"y" THEN GO TO 3
770
3790 CLS : LET l(8)=0: LET l(13)
=l(13)+5
3800 PRINT AT 1,0;"SOLD! To that
 daring person      sitting behin
d the ZX SPECTRUM."
3810 PRINT '"Use this belt no mo
re than ";u;"times."''"To jump t
o hyperdrive press the button ma
rked 't'";
3820 PRINT ' INVERSE 1;"THE MANU
FACTURER WILL NOT BE     RESPONSI
BLE FOR IMPROPER USE OF THIS TRA
NSPORTER BELT                "
3830 FOR i=1 TO 5: BEEP .75,i: N
EXT i
3840 GO TO 4900


3890 REM skeleton key
3900 PRINT AT 2,0;q$(4);"!"
3910 PRINT 'TAB 15;"[redacted]";TAB 11;
"[redacted]";TAB 15;"[redacted]"
3930 PRINT '''"This key gives you
 access to anydoor in the maze!"
3940 PRINT '"You can use it as m
any times as you like."''"This ke
y has an 'EC' factor of 3"
3950 PRINT FLASH 1;"Do you want
to carry this along?"
3960 IF INKEY$="n" THEN GO TO 48
90
3970 IF INKEY$<>"y" THEN GO TO 3
960
3980 FOR i=1 TO 10: BEEP .1,5: P
AUSE 10: NEXT i
3990 CLS
4010 IF NOT l(9) THEN LET l(13)=
l(13)+3
4020 LET l(9)=1
4030 GO TO 4900
4090 REM food(energy)
4100 PRINT AT 2,10;"Food!"
4110 PRINT '"You quickly rush to
 the table"'"and gobble up all t
he food."
4120 LET am=FN r(25)+10
```



```
4130 PRINT AT 10,3; INVERSE 1;"P
ress any key to find out";AT 7,0
;"The food has given you ": PAUS
E 0
4140 IF am>25+FN r(5) THEN PRINT
 "Indigestion! You don't get any
 energy points for that!": BEEP
 1,-20: GO TO 4900
4150 PRINT am;" valuable energy
points."
4160 LET l(11)=l(11)+am
4200 GO TO 4900
4290 REM treasure
4300 RESTORE 4480
4310 FOR i=1 TO FN r(5): READ t$
,am: NEXT i
4320 PRINT AT 2,3;q$(5);AT 2,17;
"!"''"You have ";t$
4330 LET am=FN r(am)+5: LET en=I
NT (am/5)
4340 PRINT "It comes to the valu
e of ";am'"treasure points and h
as an 'EC' factor of ";en
4350 PRINT '"The treasure may be
 useful to   bribe the evil beas
ts that hide in these passages."
4360 PRINT '"Do you want to carr
y this along?"
4370 IF INKEY$="n" THEN GO TO 48
90


4380 IF INKEY$<>"y" THEN GO TO 4
370
4390 LET l(13)=l(13)+en: LET l(1
0)=l(10)+am
4400 PRINT '"You have now gained
 ";l(10);" treasure points."
4410 PRINT '"Your 'EC' factor no
w adds up to ";l(13);". Each mov
e you make uses up "'l(13);" ene
rgy units."
4420 GO TO 4900
4480 DATA "struck gold!",60,"A c
hest of silver",30,"A sack of gl
eaming      diamonds",40,"A pla
num sword",20,"A jar of rubies",
20
4490 REM evil villain
4500 RESTORE 4500
```

```
4510 DATA "An angry","A crazy","
A fierce","An ugly","A horrible"
,"A giant","A smelly","A hungry"
4520 FOR i=1 TO FN r(8): READ a$
: NEXT i: RESTORE 4530
4530 DATA "gorilla","demon","dra
gon","troll","werewolf","android
",n$
4540 FOR i=1 TO FN r(7): READ t$
: NEXT i
4550 PRINT INK 3;AT 2,5;a$;" ";t
$
4560 PRINT INK 1;"that's going t
o rip your arms  off for waking
 him up!"
4570 PRINT INK 0;'"What are you
going to do?"'"Are you going to:
"
4580 PRINT INK 1;'"1/ fight"'"2/
bribe"'"3/ run"'"4/ pray he'll g
o away"
4590 IF INKEY$<"1" OR INKEY$>"4"
 THEN GO TO 4590
4595 CLS
4600 GO TO VAL INKEY$*50+4560
4610 PRINT "You have ";
4612 IF l(6) THEN PRINT "a laser
";
4614 IF l(7) THEN PRINT "an anti
matter ray,";
4616 IF NOT (l(6)+l(7)) THEN PRI
NT "nothing,";
4618 PRINT "for a weapon."
4620 LET pr=((l(7)>0)+.8*(l(6)>0
)): IF NOT pr THEN LET pr=RND/2
4622 PRINT AT 21,1;"Press a key
to begin the fight": PAUSE 0
4624 FOR i=1 TO 5*(1/pr)
4626 LET cl=FN r(8)-1: BORDER cl
: PAPER cl: INK 7-cl: LET mc=USR
 invert: BEEP .05,10-FN r(20)
4628 IF l(6) THEN BORDER 2: PAPE
R 6: BEEP .075,30: LET mc=USR in
vert
4630 IF l(7) THEN BORDER 0: PAPE
R 7: BEEP .1,0: LET mc=USR inver
t
4631 LET l(11)=l(11)-2
4632 NEXT i: BORDER 5: INK 4: PA
PER 0: BRIGHT 1: CLS
4634 IF RND<pr THEN GO TO 4648
```



```
4636 PRINT AT 5,1;"YOU LOST! The
 ";a$(3 TO );" ";t$;'"has thrown
 your armless body in the corner
 to join many others  who had ve
ntured in this maze"
4638 FOR i=0 TO -30 STEP -1: BEE
P .5,i: NEXT i
4640 FLASH 1: CLS : GO SUB 5160
4642 POKE 23693,56: PRINT AT 12,
2;"Do you want to try again?"
4644 IF INKEY$="n" THEN CLS : ST
OP
4646 IF INKEY$="y" THEN GO TO 60
4647 GO TO 4644
4648 PRINT AT 5,10;"You won!"
4650 IF l(7) THEN PRINT "but wit
h the antimatter ray you couldn'
t lose!": LET l(7)=l(7)-1: GO TO
 4900
4652 IF l(6) THEN PRINT "That wa
s handy work with the  laser!"
: LET l(6)=l(6)-1: GO TO 4900
4654 PRINT "That was a pretty to
cky blow,"'"the ";a$(3 TO );" ";
t$;" must be sick today."
4656 GO TO 4900
4660 PRINT "What are you going t
o give"'a$;" ";t$;"?"
4665 PRINT '"You have with you:"
: FOR a=1 TO 6: PRINT q$(a) AND
l(a+5)': NEXT a
4670 DIM g$(18): INPUT g$
4673 FOR i=1 TO 6: BEEP .1,5: IF
 g$=q$(i) AND l(5+i) THEN GO TO
4680
4676 FOR i=1 TO 6: BEEP .1,5: IF
 g$=q$(i) AND l(5+i) THEN GO TO
4680
4678 NEXT i: PRINT AT 12,0;"Sorr
y,I didn't understand that  Try
again!": BEEP .5,5: PAUSE 0: DIM
 g$(64): PRINT AT 12,0;g$: GO TO
 4670
4680 IF i=5 THEN GO TO 4690
4682 PRINT '"The not so ";a$(3 T
O );" ";t$;'"says thank you and t
akes the "'q$(i,3 TO )'"and sudd
enly disappears"
4684 LET l(i+6)=0: LET l(13)=l(1
3)-e(i)
4686 GO TO 4900
```

```
4690 INPUT "How much treasure ar
e you going to give the ";(t$);"
?";am
4693 IF am>0 THEN GO TO 4700
4694 PRINT "The ";t$;" is enrage
d that you tried to cheat him an
d has"'"taken everything,except
half"'"your treasure."
4695 FOR i=0 TO -20 STEP -1: BEE
P .05,i: NEXT i
4696 LET t=l(11)/2: LET l=l(10):
 LET m=l(4): LET n=l(5): DIM l(1
3)
4697 LET l(13)=1: LET l(4)=m: LE
T l(5)=n: LET l(10)=l: LET l(11)
=t
4698 GO TO 4900
4700 IF am>FN r(20)+10 THEN PRIN
T "The ";t$;" accepts your measl
y"'"offering and tells you he wi
ll  be more ";a$(3 TO );" to you
 next"'"time.": BEEP 1,10: GO TO
 4900
4702 PRINT "The ";t$'"insulted b
y what you have"'"offered,gets m
ore ";a$(3 TO )'"and steals your
";
4704 LET o=FN r(5): IF o=5 THEN
LET o=6
4706 LET l(13)=l(13)-e(o): LET l
(o+5+(o=6))=0
4708 PRINT q$(o,3 TO )
4709 GO TO 4900
4710 IF l(8) THEN PRINT "You act
ivate your transporter  and whoo
sh! You're away!": BEEP 1,50: G
O TO 6500
4711 FOR i=0 TO FN r(40)+10 STEP
 .5: BORDER FN r(7): BEEP .01,i:
 LET l(11)=l(11)-.5: NEXT i: FOR
 i=0 TO FN r(50): BEEP .01,-i: N
EXT i
4714 LET r=2-FN r(3)
4718 IF NOT r THEN GO TO 4730
4722 PRINT "Not fast enough!"'t$
;"'s are pretty fast"'"sometimes
;"'"you can only fight now."
4725 GO TO 4610
4735 PRINT "You made it! Why are
n't you"'"representing your coun
try in"'"running?"
```



```
4740 PRINT "After a quick 100 me
tre sprint  around the corner yo
u find"'"yourself back in the sa
me place"
4745 PRINT AT 20,0;"After you ha
ve had a rest,press a key"
4750 BEEP .5,-50: BEEP .3,-40
4755 IF INKEY$="" THEN GO TO 475
0
4758 GO TO 4900
4760 FOR i=0 TO 5: PAPER FN r(7)
: BEEP .1,50: CLS : BEEP .5,-50:
 NEXT i
4765 PAPER 6: INK 0: CLS
4770 PRINT "It didn't work,the "
;t$'"is still there!"
4773 LET w=FN r(10): LET l(11)=l
(11)-w
4775 PRINT "You wasted ";w;" ene
rgy units"'"trying though."
4780 GO TO 4570
4890 PRINT "OK,we'll leave it."
4900 GO SUB 8400: REM wait
4920 BEEP 1,5
4930 BORDER 0: PAPER 0: INK 5: B
RIGHT 1: CLS
4940 LET m$(py,px)=" "
4950 GO TO 230
5000 REM map
5005 IF l(10)<10 THEN PRINT INK
6;AT 0,0;"You don't have enough
treasure  to pay the fee,no mone
y,no map.": BEEP .5,-30: GO TO 1
00
5010 BORDER 5: BRIGHT 0: CLS : P
RINT AT 4,9; INK 6;"****NORTH***
*"
5020 IF INKEY$<>"" THEN GO TO 50
20
5030 LET mx=(px-6)*(px>5)+1: LET
 my=(py-6)*(py>5)+1
5040 IF mx+10>l(4) THEN LET mx=l
(4)-10
5050 IF my+10>l(5) THEN LET my=l
(5)-10
5060 LET d$=m$(py,px): LET m$(py
,px)="*"
5070 FOR i=my TO my+10: PRINT IN
K 6;TAB 9;"*";
5080 FOR c=mx TO mx+10
```

```
5090 IF CODE m$(l,c)>48 AND CODE
 m$(l,c)<128 THEN PRINT "■";: GO
 TO 5120: REM print door
5100 IF m$(l,c)>"5" THEN PRINT F
LASH 1; INK 6;"▒";: GO TO 5120
5110 PRINT m$(l,c);
5120 NEXT c: PRINT INK 6;"*"
5130 NEXT l: PRINT INK 6;TAB 3;"
****SOUTH****"
5140 LET l(13)=l(13)-2: LET l(10
)=l(10)-10: REM pay $10 fee for
map
5150 LET m$(py,px)=d$
5160 PRINT INK 4; BRIGHT 1;AT 0,
0;"Energy left=";l(11)'"Treasure
 points=";l(10)'"You energy cons
umption per move is ";l(13)
5170 PRINT INK 5;AT 17,0;"You ar
e carrying:"
5180 IF l(6) THEN PRINT INK 6;q$
(1);l(6);" shots left"
5190 IF l(7) THEN PRINT q$(2);l(
7);" shots left"
5200 IF l(8) THEN PRINT INK 5;q$
(3);" Life of ";l(8)
5210 IF l(9) THEN PRINT INK 4;q$
(4)
5215 IF l(12) THEN PRINT INK 5;"
A key"
5220 IF NOT (l(12)+l(6)+l(7)+l(8
)+l(9)) THEN PRINT FLASH 1;"Noth
ing"
5225 IF INKEY$<>"" THEN GO TO 52
25
5227 GO SUB 5500
5230 PAUSE 500: RETURN
5500 IF NOT g1 THEN RETURN
5505 PRINT "The EXIT is ";"NORTH
-" AND py>g1;"SOUTH-" AND py<g1;
5510 PRINT "WEST" AND px>g2;"EAS
T" AND px<g2;" of here."
5520 RETURN
5000 REM key
6010 PRINT AT 1,5;"You have foun
d a key!"
6020 BEEP .75,10
6030 IF NOT l(12) THEN PRINT AT
21,3;"Do you want to pick it up?
"
6040 IF l(12) THEN PRINT AT 21,3
;"Do you want to swap keys?"
```



```
6050 LET m$(py,px)=" ": REM spac
e
6060 IF INKEY$="n" THEN RETURN
6070 IF INKEY$<>"y" THEN GO TO 6
050
6075 IF l(13)<1 OR NOT l(12) THE
N LET l(13)=l(13)+1
6080 BORDER 2: PAPER 4: INK 0: C
LS
6090 LET key=FN r(5): IF key=l(1
2) THEN GO TO 6090
6095 LET l(12)=key
6100 BEEP .1,3: BEEP .4,10: BEEP
 .2,3: BEEP .5,10
6110 PRINT AT 10,2;"YOU NOW HAVE
 A KEY MARKED ";l(12)''" Guard i
t closely your life may        d
epend on it."
6120 GO SUB 8400
6140 BORDER 0: PAPER 0: INK 5: C
LS
6150 RETURN
6500 REM transporter
6510 LET px=FN r(l(4)): LET py=F
N r(l(5)): LET f=FN r(4)
6520 FOR i=0 TO 10: LET cl=FN r(
8)-1: BORDER cl: PAPER 7-cl: BEE
P .01,10-FN r(20): CLS : NEXT i
6530 FOR i=1 TO 5
6540 LET cl=FN r(8)-1: BEEP .1,4
0
6550 BORDER cl: PAPER 7-cl: CLS
6560 PAUSE 20: POKE 23693,56: CL
S
6570 NEXT i: BORDER 1
6580 PRINT AT 6,0;"You have now
been transported"'"to a differen
t part of the maze"
6590 LET l(8)=l(8)-1: IF NOT l(8
) THEN LET l(13)=l(13)-5
6600 GO SUB 8400: RETURN
7000 REM drop object
7005 PAPER 6: INK 0: BORDER 4: C
LS
7010 INPUT ("What do you want to
 leave"'n$;"? ");g$
7020 IF g$(1)=" " THEN RETURN
7030 FOR i=1 TO 6
7040 IF g$=q$(i) THEN GO TO 7070
7050 BEEP .1,10-FN r(20): NEXT i
```

```
7060 PRINT "Sorry,I don't know w
hat ";g$;"is,Try again.": GO TO
7010
7070 IF l(i+5) THEN GO TO 7090
7080 PRINT "But you don't have "
;g$'"Try again": GO TO 7010
7090 IF i=5 THEN GO TO 7140
7100 PRINT "OK,say goodbye to yo
ur "'g$(3 TO )
7110 LET l(13)=l(13)-e(i)
7120 LET l(i+5)=0
7130 BEEP 1,10: GO TO 7190
7140 INPUT "How much treasure do
 you want toleave? ";tr
7150 IF tr>l(10) THEN PRINT "You
 don't have that much"'"treasure
 ";n$': BEEP 1,-10: GO TO 7140
7160 LET l(13)=l(13)-tr/5: LET l
(10)=l(10)-tr
7170 PRINT "Say goodbye to ";tr;
" points of"'"treasure"
7180 IF l(13)<=0 THEN LET l(13)=
2
7190 BORDER 0: PAPER 0: INK 5: C
LS : RETURN
7200 REM dematerialisers
7210 LET dy=py+(f=3)-(f=1): LET
dx=px+(f=2)-(f=4)
7212 IF dy>l(5)-1 OR dy<2 OR dx<
2 OR dx>l(4)-1 THEN PRINT "Sorry
 you can dematerialise thisouter
 wall": PAUSE 0: RETURN
7215 LET m$(dy,dx)=" "
7220 FOR i=40 TO 55: LET ct=8-FN
 r(8)
7230 PAPER ct: BORDER 7-ct: CLS
7240 BEEP .05,i: NEXT i
7250 LET l(7)=l(7)-1
7260 BEEP .5,-20: RETURN
7500 REM save game
7505 LET l(1)=py: LET l(2)=px: L
ET l(3)=f
7510 PRINT FLASH 1;AT 16,7;"Savi
ng Maze Data"
7520 PRINT INVERSE 1'"Filename "
"maze""";AT 20,0;"Please keep pr
essing a key untilthe screen cle
ars"
7530 SAVE "maze" DATA l(): CLS :
 SAVE "maze" DATA m$()
7540 INPUT "Verify?";a$: IF a$<>
"y" THEN RETURN
```



```
7550 PRINT "Rewind cassette,then
 PLAY"'"If loading error then t
ype:GO TO 7500"
7560 VERIFY "maze" DATA l(): VER
IFY "maze" DATA m$()
7570 RETURN
7600 PRINT FLASH 1;AT 10,9;"Load
ing Data"
7620 PRINT #0; FLASH 1;AT 1,4;"S
tart cassette recorder"
7630 LOAD "maze" DATA l(): LOAD
"maze" DATA m$()
7640 LET py=l(1): LET px=l(2): L
ET f=l(3)
7650 CLS
7660 RETURN
8000 REM variables
8010 IF NOT set THEN GO TO 8070
8020 LET px=2: LET py=px: LET f=
2: REM player position in maze
8030 DIM l(13): LET l(4)=m: LET
l(5)=n
8040 LET l(10)=200: REM treasure
8050 LET l(13)=1: REM energy con
sumption
8060 LET l(11)=INT (l(4)*l(5)/5)
8070 LET r=0: LET re=r: REM repe
at
8080 LET g1=0: LET sw=0
8090 REM define graphics
8100 RESTORE 8250
8110 FOR i=1 TO 7: READ c$
8120 FOR e=0 TO 7: READ n: POKE
USR c$+e,n
8130 NEXT e: NEXT i
8140 FOR i=USR "s" TO USR "s"+16
8150 READ b: POKE i,b: NEXT i
8160 LET invert=USR "s"
8170 DIM q$(6,16): DIM e(6)
8180 FOR i=1 TO 6: READ a$,e: LE
T q$(i)=a$: LET e(i)=e: NEXT i
8190 BORDER 0: PAPER 0: INK 5: B
RIGHT 1: CLS
8200 RETURN
8250 DATA "a",0,0,3,15,12,12,12,
0
8260 DATA "b",0,0,128,224,96,48,
48,96
8270 DATA "c",1,1,1,1,0,1,1,0
8280 DATA "d",224,192,128,128,0,
128,128,0
```

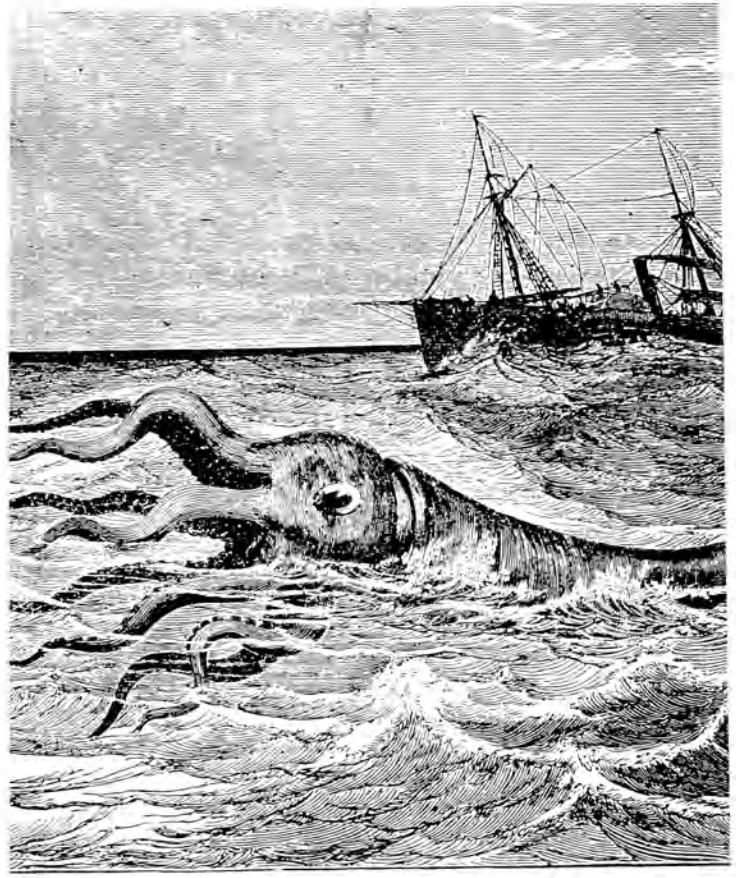
```
8290 DATA "e",255,129,129,129,12
9,129,129,255
8300 DATA "k",0,2,3,255,67,226,6
4,0
8310 DATA "h",24,60,126,60,24,24
,24,24
8320 DATA 33,0,64,6,24,197,6,0,1
26,238,255,119,35,16,249,193,16,
243,201
8330 DATA "a laser",3,"an antima
tter ray",14,"a transporter belt
",5,"a skeleton key",3,"some tre
asure",0,"a key",1
8400 PRINT AT 21,5; FLASH 1; INK
 9;"PRESS 'c' TO CONTINUE."
8410 IF INKEY$<>"c" THEN GO TO 8
410
8420 BEEP 1,10: RETURN
9000 REM instructions
9002 POKE 23609,100
9005 INPUT "Hullo! Before we sta
rt,please"'"enter your name:"; L
INE n$: IF n$="" OR n$="no" OR n
$="yes" THEN GO TO 9005
9007 IF n$(1)>"Z" THEN LET n$(1)
=CHR$ (CODE n$-32)
9010 BORDER 1: INK 1: PAPER 6: C
LS
9020 FOR b=0 TO 10
9030 PLOT 127,175: DRAW b*25-125
,-175
9040 PLOT 127,0: DRAW b*25-125,1
75
9050 NEXT b: POKE 23609,50
9060 PLOT 0,112: DRAW 255,0
9070 PLOT 0,70: DRAW 255,0
9080 PRINT AT 3,0;
9090 OVER 1: INK 2
9100 PRINT TAB 7;""
9101 PRINT TAB 7;""
9102 PRINT TAB 7;"
"
9103 PRINT TAB 7;"
"
9104 PRINT TAB 7;"
"
9105 PRINT
9110 PRINT TAB 10;""
9111 PRINT TAB 10;""
9112 PRINT TAB 10;""
9114 PRINT TAB 10;""
9115 PRINT
```



```
9120 PRINT TAB 7;""
9121 PRINT TAB 7;""
9122 PRINT TAB 7;"
"
9123 PRINT TAB 7;"
"
9124 PRINT TAB 7;"
"
9150 INK 1: OVER 0: BEEP .4,20:
BEEP .5,5: BEEP .5,10
9160 INPUT "Do You Need Instruct
ions(y/n)?"; LINE a$: IF a$="n"
THEN RETURN
9170 BORDER 5: PAPER 1: INK 4: B
RIGHT 1: CLS
9180 PRINT 'TAB 8; FLASH 1;"*DOO
RS OF DOOM*"''
9190 PRINT "GREETINGS to the DOO
RS OF DOOM"' FLASH 1;"?";CHR$ 8;
9220 PRINT n$;"."
9230 PRINT "You have come here i
n search of the Ultimate Goal wh
ich lies     beyond one of the DO
ORS OF DOOM at the heart of the
unwelcoming labyrinth."
9240 PRINT '"You must journey th
rough the"'"long deceiving passa
ges in"'"search of the right rou
te to theinterior."''
9250 PRINT "Throughout the maze
there are"'"doors leading to an
adjacent"'"passage,and behind th
ese doors  may lie DANGER...or h
ope?"'
9260 GO SUB 8400: PAPER 0: CLS
9270 PRINT "The doors are locked
 and can"'"only be opened if you
 possess a matching key,which ar
e littered around the labyrinth.
"
9280 PRINT INVERSE 1'"DIRECTIONS
->": INK 6
9290 PRINT '"You may use a map w
hich shows"'"the surrounding wal
ls and the"'"path you have taken
.It will alsoshow up any keys an
d doors"'"nearby"
9300 PRINT "Each time you ask fo
r a map you lose 10 treasure poi
nts"
9310 GO SUB 8400: BORDER 0: CLS
```

```
9320 PRINT TAB 8;"'KEY'words": P
LOT 70,167: DRAW INK 4;24,0
9330 PRINT '"Keys 'L' for left,'
R' for right and 'B' for back."
9340 PRINT "Key 'F' to step for"
ard"
9350 PRINT "Key 'S' to save game
"'"Key 'E' to repeat.(if you com
e to a long corridor?"
9360 PRINT INK 5'"Each object yo
u pick up has its own Energy Con
sumption or 'EC' factor.This is
 the amount of"'"energy used aft
er each step."
9370 PRINT INK 7'"You may specif
y the size of mazeto be used,the
 largest maze willtake up to 20
minutes to producewhile the smal
lest will take"'"about 1.2 minut
es"
9380 PRINT INK 5;"You may save t
he mazes for"'"future use."
9390 GO SUB 8400
9400 RETURN
```


MOVING GRAPHICS GAMES



Gold Rush

Your task in this game, written by Neil Pellinacci, is to carry six sacks of gold from a bank vault to a safe. The vault and safe are separated by a river. The river can be crossed via a network of small dams, which are built across it. Unfortunately, the future of this link is not certain, because occasional logs drifting down the river will destroy parts of the dam system. You must transfer all the sacks to the safe before your route is cut off.

You play the game using the following control keys:

Q - to move up
Z - to move down
I - to move left
P - to move right

Your current task is shown by the color of your man. White indicates that you have to pick up a sack, and yellow shows you are carrying a sack. To pick up a sack, you just move onto the same square as it. You put it in the safe by moving onto the safe. In either case, a beep will sound, and the man will change color. You have three men per game, and the number left is shown at the top of the screen.

A log will remove any section of dam in its path. However, sections above and below are also damaged, and if you move off one of these, and into part of the river where a log has been, you will lose one life. This is the only way you can lose a life, so be very careful after the first few logs have appeared.

When all six sacks have been put in the safe, you move onto the next level. On this level, there are two dam formations, and these alternate. As the game progresses, the number of logs increases, until finally you'll find the game is virtually impossible.

You'll see at this point the dam is almost nonexistent, and you'll have to resort to new tricks. You can 'hop' onto a log immediately above or below. It is difficult at first, but not impossible. Remember, you don't move with the log; you are just using it as a piece of dam. If you are very careful, you can even move right and left using the logs.

The end of the game is the hardest part, and it is the main reason for getting as high a score as you can before the dam is destroyed.

Scoring is simple. You get 100 points for going from the safe to the bank vault and another 100 for successfully bringing back a sack of gold. If you lose a life while

carrying the sack, you do not score. Neil's highest score is around 3300, which gives you a target to shoot for.

The program is written mainly in BASIC, but it uses a machine code routine located at 32530, which moves the logs right and left across the screen, stopping them after a collision with a dam wall. The data for this machine code is held in the last two lines of the program.

Here's the key to the user-defined graphics:

A = 

S = 

C = 

This is the program's structure:

```
      100: subroutine to print score
1000 - 1030: game preparation
1040 - 1045: initialisation for each game
1050 - 1100: initialisation for each level
             of the game
1510 - 1560: main loop
1600 - 1630: detects collisions
1700 - 1740: end of one level
1800 - 1820: lose one man routine
2000 - 2100: game over routine
6000 - 6650: draw dams across river
7000 - 7100: draw river and the banks
8000 - 8100: title page and instructions
9000 - 9120: POKE user graphics characters
9200 - 9310: POKE machine code into memory
```

And here's the listing:

```
  10 CLEAR 32529: RANDOMIZE : GO
 TO 1000
  99 REM SCORE
 100 PRINT AT 0,11-LEN STR$ S; I
NK 7; PAPER 2; BRIGHT 1;S: RETUR
N
1000 REM SETUP
1010 BORDER 2: PAPER 4: CLS : GO
 SUB 9000
1020 PRINT AT 0,0; INK 7; PAPER
2;"SCORE "; BRIGHT 1;"00000"; BR
IGHT 0;" MEN "; BRIGHT 1;"3"; BR
IGHT 0;" HI-SCORE "; BRIGHT 1;"0
0000"
1030 LET HS=0
1040 GO SUB 5000: PRINT AT 0,6;
PAPER 2; INK 7; BRIGHT 1;"00000"
;AT 0,18;"3"
1045 LET MM=0: LET MEN=3: LET S=
0: LET RN=.992
1050 LET SK=0: LET L=2: LET C=23
: LET G=7
1100 GO SUB 7000: GO SUB 6000
1500 REM LOOP
1510 LET L1=L: LET C1=C
1517 LET A$=INKEY$: IF A$="" THE
N GO TO 1555
1520 LET L=L+(A$="Z")-(A$="Q")
1530 LET C=C+(A$="P")-(A$="I")
1540 LET AT=ATTR (L,C): IF AT<>2
4 THEN GO TO 1600
1545 BEEP .0025,10
1550 PRINT AT L1,C1; PAPER 3; IN
K 0;" ";AT L,C; INK G; PAPER 3;"
*"
1555 IF RND>RN THEN PRINT AT 7+I
NT (RND*9),30; INK 0; PAPER 0;"
 ": LET RN=RN-.0005
1560 LET Z=USR 32530: GO TO 1510
1600 IF AT=30 THEN IF G=7 THEN L
ET G=6: BEEP .1,20: LET S=S+100:
 GO SUB 100: LET SK=SK+1: GO TO
1550
1610 IF AT=31 THEN IF G=6 THEN B
EEP .1,40: LET G=7: LET S=S+100:
 GO SUB 100: IF SK=6 THEN GO TO
1700
1620 IF AT=45 THEN GO TO 1600
1625 IF AT=0 THEN GO TO 1545
```



```
1630 LET L=L1: LET C=C1: GO TO 1
560
1700 FOR A=-10 TO 20 STEP 5: BEE
P .1,A: BEEP .1,A-3: NEXT A
1710 FOR A=15 TO -10 STEP -5: BE
EP .1,A: BEEP .1,A-3: NEXT A: BE
EP .1,-15
1720 FOR A=1 TO 21: PRINT AT A,0
; PAPER A/3;"
                ": NEXT A
1730 BEEP .1,20: BEEP .1,25: FOR
 A=1 TO 21: PRINT AT A,0; PAPER
0;"
   ": NEXT A
1740 GO TO 1050
1800 FOR A=10 TO -25 STEP -4: PR
INT AT L,C; PAPER 5; INK 0;"*";A
T L,C; PAPER 5; INK 5;" ": BEEP
.01,A: NEXT A
1805 PRINT AT L1,C1; PAPER 3; IN
K 0;" "
1810 LET MEN=MEN-1: PRINT AT 0,1
6; PAPER 2; INK 7; BRIGHT 1;MEN
1815 IF MEN=0 THEN GO TO 2000
1817 IF SK=6 THEN GO TO 1700
1820 LET L=2: LET C=23: LET G=7:
 GO TO 1500
2000 REM GAME OVER
2010 FOR A=0 TO 7: FOR B=1 TO 21
: PRINT AT B,0; PAPER A;"
                ": NEXT
 B: NEXT A
2020 FOR A=1 TO 12: BEEP .1,3: B
EEP .1,8: BEEP .1,13: NEXT A
2030 IF S>HS THEN LET HS=S: PRIN
T AT 0,32-LEN STR$ HS; PAPER 2;
INK 7; BRIGHT 1;HS: BEEP .1,2: B
EEP .2,10
2035 FOR A=20 TO 2 STEP -1: PRIN
T AT A,0; BRIGHT 1; PAPER RND*7;
 INK 9;" GAME OVER GAME OVER GAM
E OVER ": NEXT A
2040 FOR A=0 TO 7: BORDER A: BEE
P .2,7-A: NEXT A: FOR A=7 TO 0 S
TEP -1: BORDER A: BEEP .2,7-A-1:
 NEXT A
2100 GO TO 1040
5000 STOP
6000 REM BRIDGES
6010 IF MM=0 THEN GO TO 6600
6020 LET MM=0
```

```
6050 PAPER 3: INK 0: PRINT AT 7,
9;"      ";AT 7,18;"      "
6060 PRINT AT 8,9;"  ";AT 8,12;"
";AT 8,15;"
6070 PRINT AT 9,9;"      ";AT 9,15
;"      ";AT 10,9;"      ";AT 10,17;
"    "
6080 PRINT AT 11,9;" ";AT 11,12;
"  ";AT 12,6;"    ";AT
 12,14;"     ";AT 12,20;"   "
6090 PRINT AT 13,6;"      ";AT 13,14
;"  ";AT 13,21;"  "
6100 PRINT AT 14,6;"          ";A
T 14,18;"    ";AT 15,9;"      ";
AT 15,18;"    "
6500 RETURN
6600 PAPER 3: INK 0: PRINT AT 7,
12;"    ";AT 7,20;"    ";AT 8,10;"
6610 PRINT AT 9,10;" ";AT 9,13;"
";AT 9,16;" ";AT 9,20;"  ";A
T 10,10;" ";AT 10,13;" ";AT 10,1
6;" ";AT 10,20;"  "
6620 PRINT AT 11,10;"      ";AT
 11,20;"  ";AT 12,10;" ";AT 12,1
6;"    "
6630 PRINT AT 13,9;"    ";AT 13,1
6;" ";AT 13,20;" ";AT 14,9;"
    ";AT 14,20;"  "
6640 PRINT AT 15,9;"    ";AT 15,1
7;"  "
6650 LET MM=1: RETURN
7000 REM SCREEN
7010 FOR A=1 TO 21: PRINT AT A,0
; PAPER 4;"
                ": NEXT A
7015 FOR A=6 TO 16: PRINT AT A,0
; PAPER 5; INK 0;"
                ": NEXT A
7020 INK 0: PAPER 0: FOR A=1 TO
9: PRINT AT A,20;"  ": NEXT A
7030 PRINT AT 2,9;"
 ";AT 3,9;" ";AT 3,18;" ";AT 4,9
;"  ";AT 4,18;"  "
7040 PRINT AT 5,13;" ";AT 5,13;"
 ";AT 5,21;" ";AT 6,21;"  "
7050 PRINT AT 16,10;"  ";AT 16,18
;"  "
7060 PRINT AT 17,9;"      ";A
AT 18,18;"    ";AT 19,20;" ";A
T 20,14;"
```



```
7070 FOR A=19 TO 21: PRINT AT A,
6;"      ": NEXT A
7080 FOR A=19 TO 21: PRINT AT A,
6; INK 6;"▒▒": NEXT A: PRINT AT
2,25; INK 7;"Ø"
7100 RETURN
8000 REM TITLE PAGE
8005 BORDER 5
8010 FOR A=1 TO 21: PRINT AT A,0
; PAPER 6;"
                ": NEXT A
8012 DIM I(6): FOR A=1 TO 6: LET
 I(A)=A-1: NEXT A
8013 FOR A=1 TO 6: LET II=INT (R
ND*6)+1: LET AA=I(A): LET I(A)=I
(II): LET I(II)=AA: NEXT A
8015 GO TO 8030
8018 FOR A=1 TO 6: INK I(A): BOR
DER I(A): PRINT AT 4,7;"▒▒▒▒▒▒▒▒
▒▒▒▒▒▒▒▒▒▒"
8020 PRINT AT 6,7;"▒▒▒▒▒▒▒▒▒▒▒▒▒
▒▒▒▒▒"
8022 PRINT AT 5,7;"▒";AT 5,24;"▒
": BEEP .4,I(A)+10
8024 IF INKEY$<>"" THEN BORDER 2
: RETURN
8026 NEXT A: GO TO 8018
8030 PRINT AT 10,10; PAPER 0; IN
K 6; BRIGHT 1;" GOLD  RUSH "
8040 INK 0: PAPER 6: PRINT AT 12
,12;"CONTROLS": PLOT 0,75: DRAW
95,0: PLOT 255,75: DRAW -95,0
8050 INK 1: PRINT AT 14,9;"O....
.....UP";AT 15,9;"Z.........DO
WN"
8060 PRINT AT 16,9;"I.........LE
FT";AT 17,9;"P.........RIGHT"
8070 PAPER 6: INK 0: PLOT 0,24:
DRAW 255,0
8080 PRINT AT 19,6; PAPER 4; INK
 7; BRIGHT 1;"PRESS ANY KEY TO P
LAY"
8090 PRINT AT 21,0; PAPER 2; INK
 7;"Written by Neil Pellinacci
1983"
8100 GO TO 8016
9000 REM UGD
9005 RESTORE 9100
9010 FOR A=USR "A" TO USR "A"+23
: READ B: POKE A,B: NEXT A
```

```
9100 DATA 0,126,60,24,60,126,255
,126
9110 DATA 255,129,133,165,137,12
9,255,102
9120 DATA 16,56,16,124,16,40,68,
68
9200 REM MCODE
9210 FOR A=0 TO 47: READ B: POKE
 32530+A,B: NEXT A
9220 RETURN
9300 DATA 33,192,88,6,11,14,31,3
6,126,254,0,32,15,54,45,43,126,2
54,24
9310 DATA 32,4,54,45,24,2,54,0,3
5,13,32,232,35,16,227,33,192,88,
17,32,0,6,11,54,45,25,16,251,201
```


Tarantula

In Malcolm Young's great program TARANTULA you are in a dark pit about the size of a football field, fighting for your life against dreaded tarantula spiders.

The tarantulas slide down the sides of the pit looking for food, and human beings fit into the category of food. You're equipped with a torch and a laser, and you move around the pit in a fairly flimsy tank.

Instructions are within the program, and you'll find you'll quickly work out how to play it once you get the program up and running.

```
   1 REM graphic characters: REM
 A=[graphic] B=[graphic] C=[graphic] D=[graphic] E=[graphic] F=[graphic] G=[graphic] H=[graphic]
 I=[graphic] J=[graphic] K=.
   5 DIM p$(15): DIM w$(5,15): D
IM q(5)
   8 PRINT "Please wait just a m
oment.": GO SUB 8615: REM define
 characters
  10 GO SUB 9000
  20 BORDER 5: PAPER 0: INK 7: C
LS
  30 PRINT AT y,x;t$
  35 PRINT INK 0; PAPER 5;AT 21,
2;"SCORE=";score;AT 21,12;"SPIDE
RS=";nb;"  LIVES=";lives
  40 LET a=x: LET b=y
  50 LET i$=INKEY$
  60 GO TO (i$<"1" OR i$>"8")*20
+70
  70 BEEP .05,0: GO SUB VAL i$*1
000
  80 PRINT AT b,a;" "; BRIGHT 1;
AT y,x; INK rk;T$
  90 GO SUB (RND<lev)*100+100
 100 GO TO (d$(y+1,x+1)="[graphic]")*390
+110
```

```
 110 LET cn=cn*(cn<n)+1
 120 GO TO (n>0)*85+40
 125 IF NOT s(cn,1) THEN GO TO (
n<1)*70+40
 130 LET sx=s(cn,1): LET sy=s(cn
,2)
 140 LET d$(sy,sx)=" "
 150 LET sy=sy+(sy<y+1)-(sy>y+1)
: LET s(cn,2)=sy
 160 LET sx=sx+(sx<x+1)-(sx>x+1)
: LET s(cn,1)=sx
 170 LET d$(sy,sx)="Ж"
 180 GO TO 40
 200 REM new tarantula
 210 IF n=15 THEN LET lev=.05: R
ETURN
 220 LET n=n+1: LET nb=nb+1
 230 LET sx=s(n,1): LET sy=s(n,2
)
 240 LET d$(sy,sx)="Ж"
 250 LET c=INT (RND*6)+2
 260 PRINT BRIGHT 1; INK c;AT sy
-1,sx-1;"Ж";CHR$ 8;: BEEP .4,1:
PRINT ;" "

 270 PRINT INK 0; PAPER 5;AT 21,
2;"SCORE=";SCORE;AT 21,12;"SPIDE
RS=";nb;"  LIVES=";LIVES
 280 RETURN
 500 PRINT PAPER 7; BRIGHT 1; FL
ASH 1; INK 2;AT 10,5;"arrgh!!  T
HEY GOT YOU.": LET lives=lives-1
 510 FOR t=1 TO n: IF s(t,1)=x+1
AND s(t,2)=y+1 THEN GO TO 530
 520 NEXT t: STOP
 530 LET d$(s(t,2),s(t,1))=" "
 540 LET s(t,1)=0: LET nb=nb-1
 550 FOR t=1 TO 10: PRINT INK 2;
AT y,x;"Ж";CHR$ 8;: BEEP .25,1:
PRINT "Ж": BEEP .25,0: NEXT t
 560 PAPER 0: INK 7: FLASH 0: CL
S
 565 IF n=15 AND nb=0 THEN GO SU
B 9250
 570 PRINT INK 0; PAPER 5;AT 21,
2;"SCORE=";SCORE;AT 21,12;"SPIDE
RS=";nb;"  LIVES=";LIVES
 575 LET rk=rk+(rk<7)
 580 IF lives THEN RETURN
 590 IF score<q(5) THEN GO TO 64
0
```



```
 600 INPUT PAPER 5; INK 1;AT 0,0
;("You have made the "; FLASH 1;
"TOP FIVE"; FLASH 0' PAPER 6;"Pl
ease enter your name. "); LINE p
$
 610 FOR r=5 TO 1 STEP -1: BEEP
.1,r: IF score>q(r) THEN NEXT r
 620 FOR g=5 TO r+2 STEP -1: BEE
P .1,g: LET q(g)=q(g-1): LET w$(
g)=w$(g-1): NEXT g
 630 LET q(g)=score: LET w$(g)=p
$
 640 BORDER 3: PAPER 5: INK 2: C
LS
 650 PRINT AT 3,2;"Today's Champ
ions are:"
 660 FOR r=1 TO 5: BEEP .5,r
 670 PRINT AT 7+r,4; FLASH (w$(r
)=p$ AND score=q(r));w$(r);q(r):
NEXT r
 680 PRINT AT 18,3; FLASH 1; INK
0;"DO YOU WANT TO PLAY AGAIN?":
PAUSE 0
 690 IF INKEY$="n" THEN STOP
 695 GO TO 10
 700 REM promote rank
 710 BORDER 2: PAPER 6: INK 1: C
LS
 720 PRINT AT 3,7;"CONGRATULATIO
NS"''TAB 1;"You have killed ";sc
ore/10;" tarantulas"
 730 PRINT 'TAB 5;"You have now
been"
 740 LET rk=rk-1
 750 PRINT 'TAB 3;"promoted to t
he ";rk;"th" AND rk>3;"rd" AND r
k=3;"nd" AND rk=2;"st" AND rk=1;
" RANK"
 760 RESTORE 8660: FOR m=1 TO 25
: READ n: BEEP .2,n: NEXT m
 770 PRINT 'TAB 2;"You will now
change tanks to"
 780 PRINT TAB 5;"signify your n
ew rank"
 790 PRINT '"You now control the
";
 800 FOR c=6 TO rk STEP -1: READ
c$: NEXT c
 810 PRINT BRIGHT 1; PAPER rk; I
NK 9;c$;
 820 PRINT " tank."
```

```
 830 DIM d$(22,32): LET s=7: LET
 r=s/3: LET n=0: LET nb=0
 840 GO SUB 9250: PRINT AT 20,10
;"Press a key": PAUSE 500
 850 BORDER 5: PAPER 0: INK rk:
CLS
 860 LET p=0: RETURN
1000 LET VD=(T$="▲")-(T$="▼")
1010 LET HD=(T$="►")-(T$="◄")
1020 LET V=Y+1: LET H=X+1: LET w
th=1
1025 PAPER 6: INK 0:
1030 GO TO ABS vd*110+1040
1040 LET dt=(s*hd)
1050 LET d=h+dt: LET d=d OR (d<1
): LET d=d-(d>32)*(d-32)
1070 FOR t=h+hd TO d STEP hd
1080 LET ul=v-wth OR v-wth<1
1090 LET ll=(v+wth)-(v+wth>21)*(
v+wth-21)
1100 PRINT AT ul-1,t-1;d$(ul,t);
CHR$ 8;: IF ul>1 THEN PRINT OVER
 1;"◤" AND hd=1;"◥" AND hd=-1
1110 FOR w=ul+1 TO ll-1: PRINT F
LASH (d$(w,t)<>" ");AT w-1,t-1;d
$(w,t);: NEXT w
1120 PRINT AT ll-1,t-1;d$(ll,t);
CHR$ 8;: IF ll<21 THEN PRINT OVE
R 1;"◣" AND hd=1;"◢" AND hd=-1
1130 LET wth=wth+1: NEXT t
1140 GO TO 1230
1150 LET dt=(s*vd)
1160 LET d=v+dt: LET d=d OR (d<1
): LET d=d-(d>21)*(d-21)
1170 FOR t=v+vd TO d STEP vd
1180 LET ll=(h-wth) OR (h-wth<1)
1190 LET ul=(h+wth)-((h+wth-32)*
(h+wth>32))
1200 PRINT AT t-1,ll-1;d$(t,ll T
O ul):
1202 IF ll>1 THEN PRINT OVER 1;A
T t-1,ll-1;"◤" AND vd=1;"◣" AND
vd=-1
1205 IF ul<32 THEN PRINT OVER 1;
AT t-1,ul-1;"◥" AND vd=1;"◢" AND
 vd=-1
1210 LET wth=wth+1
1220 NEXT t
1230 BEEP .2,5: PAPER 0: INK 7:
CLS : PRINT AT y,x;t$:
```



```
1240 PRINT INK 0; PAPER 5;AT 21,
2;"SCORE=";score;AT 21,12;"SPIDE
RS=";nb;"  LIVES=";lives
1250 LET s=s-RND/3: RETURN
2000 LET VD=(T$="▲")-(T$="▼"): L
ET HD=(T$="►")-(T$="◄")
2010 LET V=Y+1: LET H=X+1
2020 FOR T=1 TO R
2025 LET V=V+VD: LET H=H+HD
2030 LET V=V+(V<1)-(V>21): LET H
=H+(H<1)-(H>32)
2035 BEEP .05,10
2040 PRINT AT V-1,H-1;" "
2050 GO TO (D$(V,H)<>" ")*20+206
0
2060 PRINT INK 0;AT V-1,H-1;"■":
 NEXT T
2070 LET r=r-(r>2)/10: RETURN
2080 FOR T=0 TO 7: BEEP .1,t: PR
INT INK t;AT V-1,H-1;"▒";CHR$ 8;
: BEEP .1,t+7: PRINT INK 7-t;"▒"
;CHR$ 8;: BEEP .1,t-7: NEXT T
2090 LET SCORE=SCORE+10: LET S=S
+RND/4: LET r=INT (s/1.5)+1
2100 FOR t=1 TO n: IF v=s(t,2) A
ND h=s(t,1) THEN GO TO 2120
2110 NEXT T: STOP
2120 LET s(t,1)=0: LET nb=nb-1
2130 LET d$(v,h)=" "
2140 PRINT INK 0; PAPER 5;AT 21,
2;"SCORE=";SCORE;AT 21,12;"SPIDE
RS=";nb;"  LIVES=";LIVES
2150 LET lev=lev+0.01*(lev<2)
2160 LET p=p+1: IF p=15 THEN GO
TO 700
2170 IF n=15 AND nb=0 THEN PRINT
 BRIGHT 1; PAPER 4; INK 0;AT 15,
0;"Please wait 2 seconds for the
   next wave.": GO SUB 9250
2190 RETURN
5000 LET T$="◄"
5010 LET X=X-(X>0)
5020 RETURN
6000 LET T$="▲"
6010 LET Y=Y+(Y<20)
6020 RETURN
7000 LET T$="▼"
7010 LET Y=Y-(Y>0)
7020 RETURN
8000 LET T$="►"
8010 LET X=X+(X<31)
8020 RETURN
```

```
8500 REM DEFINE CHARACTERS
8510 DATA 66,90,126,60,126,153,6
6,36
8520 DATA 129,155,126,60,126,153
,153,129
8530 DATA 0,126,60,55,55,60,126,
0
8540 DATA 16,16,146,254,206,254,
254,130
8550 DATA 130,254,254,230,254,14
6,16,16
8560 DATA 0,126,60,236,236,60,12
6,0
8570 DATA 1,3,7,15,31,63,127,255
8580 DATA 128,192,224,240,248,25
4,255
8590 DATA 255,127,63,31,15,7,3,1
,0
8600 DATA 255,254,252,248,240,22
4,192,128
8610 DATA 0,0,0,24,24,0,0,0
8615 RESTORE
8620 FOR d=97 TO 107
8625 FOR e=0 TO 7
8630 READ n: POKE USR CHR$ d+e,n
8640 NEXT e: NEXT d
8650 RETURN
8660 DATA 0,-1,-2,-1,1,1,1,0,0,0
,1,1,1,1,0,-2,0,1,1,1,0,-1,1,0,-
1
8670 DATA "YELLOW","CYAN","GREEN
","MAGENTA","RED","BLUE","BLACK"
9000 REM HEADINGS
9010 RANDOMIZE RND*65535
9020 LET H$="▚▚▚▚▚▚▚▚▚▚▚▚▚▚▚▚▚▚
▚▚▚▚▚▚▚▚▚▚"
9030 INK 1: PAPER 5: BORDER 1: C
LS
9040 PRINT H$(2 TO 31);AT 0,0;
9050 FOR X=1 TO 11: PRINT OVER 1
;TAB 31;"▚▚": NEXT X
9060 PRINT AT 21,0;H$( TO 32)
9070 PRINT AT 10,10; BRIGHT 1; F
LASH 1;"TARANTULA"
9080 LET m$="You are in a dark p
it about the size of a football
field fighting for your life aga
inst the dreaded tarantulas.The
tarantulas slide down the side o
f the pit in search of food,you
must fight off these ravenous be
asts if you value your life."
```

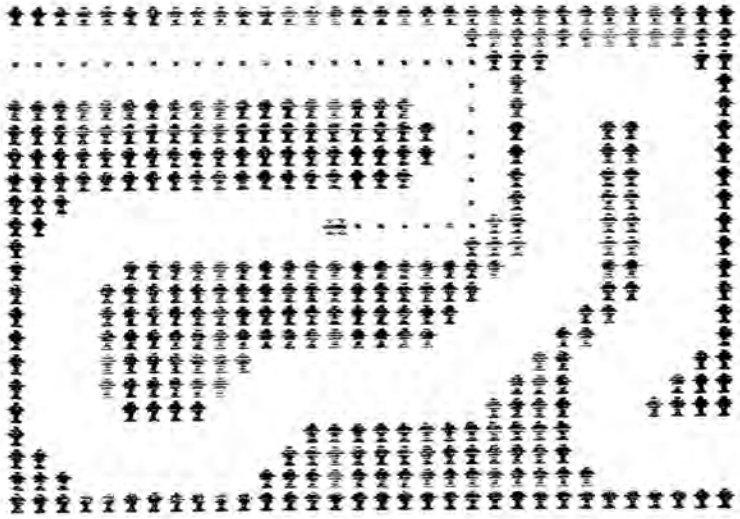


```
9090 LET m$=m$+" You are equiped
 with only a torch,a laser and y
ou move around in a lightly armo
red tank which is no match for
the crushing jaws of the tarantu
las."
9100 LET m$=m$+" Your weapons ru
n on tarantula juice and if they
 don't get enough they begin to
wear down and their range slowly
 decreases."
9110 LET m$="
       "+m$+"
                "
9120 PRINT INK 0; PAPER 6;AT 19,
1;"PRESS 'e' TO END INSTRUCTIONS
";AT 20,1;"or PRESS ANY KEY TO
SPEED UP. "
9125 PAUSE 500
9130 FOR L=1 TO LEN M$-27: PRINT
 AT 15,2;M$(L TO L+27)
9140 IF INKEY$="e" OR INKEY$="E"
 THEN GO TO 9160
9150 PAUSE 10: NEXT L
9160 BORDER 4: PAPER 5: BRIGHT 0
: CLS
9170 PRINT AT 5,2;"USE KEYS 5 TO
 8 TO MOVE YOUR    TANK."''"USE
KEY '1' TO LIGHT YOUR TORCH"''"U
SE KEY '2' TO FIRE."
9180 DIM d$(21,32): DIM s(15,2)
9190 LET s=7: LET r=s/1.5+1
9200 LET x=INT (RND*32): LET y=I
NT (RND*21)
9210 LET score=0: LET lives=3: L
ET nb=0
9220 LET rk=7: LET t$="▚"
9230 LET cn=n
9240 LET lev=.05: LET p=0
9245 GO SUB 9250: INPUT " Press
ENTER when you are ready",r$: RE
TURN
9250 FOR z=1 TO 15
9260 LET sx=INT (RND*32)+1: LET
sy=INT (RND*21)+1: LET o=RND
9270 IF o<0.5 THEN LET s(z,1)=sx
: LET s(z,2)=20*(RND>.5)+1
9280 IF o>0.5 THEN LET s(z,2)=sy
: LET s(z,1)=31*(RND>.5)+1
9290 NEXT z: LET n=nb
9300 RETURN
```

Jungle Job

This is a two-stage challenge from Andrew Sweetland and Martin Jones. In the first stage, you have to guide your canoe down a fast-flowing river. There are a number of rocks blocking your way, and you must avoid them. If you hit a rock, a man-eating fish (or a faint computer version of one) will leap out of the water to consume you.

In the second stage, you'll find yourself on a dirt roadway like this:



Your car has no brakes, so working your way around the track will call on all your driving skill.

Use the following keys to control your boat and your car:

Q – to move up
A – to move down
O – to move right
P – to move left

```
  1 REM  ***JUNGLE JOB***
©Martin Jones & Andrew Sweetland
  2 REM  EPISODE I
  3 GO SUB 9001: LET li=3
  4 POKE 23658,0
  5 FOR t=1 TO 3: BORDER 0: BRI
GHT 1: PAPER 5: INK 7: CLS : BRI
GHT 0
 10 FOR f=14 TO 21: PRINT AT f,
0; PAPER 1;: NEXT f
 15 PRINT #1;AT 0,0; PAPER 1;;;
;
 20 PRINT AT 8,0; BRIGHT 1; INK
 3;"
```

```
 30 FOR f=10 TO 13: PRINT BRIGH
T 1;AT f,0; PAPER 3;: NEXT f
 40 PRINT AT 1,29; BRIGHT 1; IN
K 6;"DE";AT 2,29;"FG"
 45 FOR f=15 TO 18: LET p=5+(RN
D*15): PRINT AT f,p; PAPER 1; IN
K 6;"HI";AT f,p+10;"HI": NEXT f:
 PRINT AT 20,RND*30; INK 6; PAPE
R 1;"HI"
 50 LET x=17: LET y=0
 55 IF y<=29 THEN IF ATTR (x,y)
=14 OR ATTR (x,y+1)=14 OR ATTR (
x,y+2)=14 THEN GO SUB 130
 60 PRINT AT x,y; INK 7; PAPER
1;".AB"
 70 LET y=y+1
 80 IF INKEY$="q" THEN IF (x>15
 AND y<29) THEN IF ATTR (x-1,y)<
>14 AND ATTR (x-1,y+1)<>14 AND A
TTR (x-1,y+2)<>14 THEN PRINT AT
x,y;  PAPER 1; INK 7;"   ": LET x
=x-1
```

```
  83 IF INKEY$="a" THEN IF (x<20
 AND y<29) THEN IF ATTR (x+1,y)<
>14 THEN IF ATTR (x+1,y+1)<>14 T
HEN IF ATTR (x+1,y+2)<>14 THEN P
RINT AT x,y; PAPER 1; INK 7;"
": LET x=x+1
  85 IF INKEY$="o" AND y<28 THEN
 LET y=y-.5
  86 OUT 254,0: OUT 254,16
  90 FOR z=1 TO 10: NEXT z: IF y
=30 THEN NEXT t: GO TO 105
  95 GO TO 55
 100 REM  END OF EPISODE I
 105 PRINT AT x,y; PAPER 1; INK
7;"AB": FOR f=50 TO 60: FOR g=0
TO 32: OUT 254,g+16: OUT 254,0:
NEXT g: NEXT f
 110 GO TO 500
 120 REM  OOOPS!
 130 PRINT AT x,y; PAPER 1; INK
7;"JK": FOR f=0 TO 50: OUT 254,1
6: OUT 254,0: NEXT f: PRINT AT x
+1,y+1; PAPER 1; INK 4;"CC";AT x
-1,y+1; INK 7;"LM": LET li=li-1:
 IF li=0 THEN GO TO 8000
 140 FOR r=1 TO 500: NEXT r: GO
TO 5
 500 REM  EPISODE II
 505 CLS : PAPER 1: INK 4: CLS :
 FOR f=0 TO 21: PRINT "NNNNNNNNNN
NNNNNNNNNNNNNNNNNNNNNNNNNN": NEXT f
 510 FOR f=-50 TO 50: BEEP .01,f
: NEXT f
 515 FOR f=1 TO 3: PRINT PAPER 4
; INK 1;AT f,0;"
    ": NEXT f
 520 FOR f=5 TO 6: PRINT AT f,19
; PAPER 4; INK 1;"    ";AT f,23;"
   ";AT f,28;"  ": NEXT f
 525 FOR f=7 TO 12: PRINT AT f,2
3; PAPER 4; INK 1;"    ";AT f,28;
"  ": NEXT f
 530 FOR f=3 TO 4: PRINT AT f,23
; PAPER 4; INK 1;"          ": NEX
T f
 535 FOR f=8 TO 10: PRINT AT f,3
; PAPER 4; INK 1;"
    ": NEXT f
 540 FOR f=10 TO 16: PRINT AT f,
1; PAPER 4; INK 1;"    ": NEXT f
```



```
 545 FOR f=15 TO 17: PRINT AT f,
11; PAPER 4; INK 1;"          ";
AT f,25;"  ": NEXT f
 550 FOR f=18 TO 20: PRINT AT f,
0; PAPER 4; INK 1;"          ";AT
f,26;"  ": NEXT f
 555 FOR f=13 TO 15: PRINT AT f,
20; PAPER 4; INK 1;"    ";AT f,27
;"  ": NEXT f
 560 PRINT PAPER 4; INK 1;AT 2,2
0;"  ";AT 2,24;"  ";AT 3,20;"
  ";AT 4,18;"  ";AT 7,18;"
  ";AT 8,20;"  ";AT 9,2;"  ";AT 9,0
;"  ";AT 9,20;"  ";AT 11,4;"
  ";AT 11,22;"  ";AT 12,21;"  ";AT
13,23;"  ";AT 13,30;"  "
 565 PRINT PAPER 4; INK 1;AT 14,
19;"  ";AT 14,23;"  ";AT 14,26;"
  ";AT 16,10;"  ";AT 16,21;"  ";A
T 16,28;"  ";AT 17,4;"  ";AT 17,9;
"  ";AT 18,11;"  ";AT 18,25;"  ";
AT 19,2;"  ";AT 19,11;"  ";AT 19,2
5;"  "
 570 LET x$="": LET x=2: LET y=0
: LET d$="O"
 580 IF x=(18 AND y=31) OR (x=19
 AND y=31) OR (x=20 AND y=31) TH
EN GO TO 7000
 581 IF ATTR (x,y)=12 THEN GO TO
 800
 583 PRINT AT x,y; PAPER 4; INK
0;d$
 585 OUT 254,0: OUT 254,16
 590 IF INKEY$="o" THEN LET x$="
o": LET d$="O"
 600 IF INKEY$="a" THEN LET x$="
a": LET d$="P"
 610 IF INKEY$="q" THEN LET x$="
q": LET d$="R"
 620 IF INKEY$="p" THEN LET x$="
p": LET d$="O"
 630 PRINT AT x,y; PAPER 4; INK
0;"T"
 640 IF x$="o" THEN LET y=y-1
 650 IF x$="p" THEN LET y=y+1
 660 IF x$="a" THEN LET x=x+1
 670 IF x$="q" THEN LET x=x-1
 680 GO TO 580
 700 GO TO 900
 800 REM  OOOPS!
```

```
 810 PRINT AT x,y; INK 2; FLASH
1;"S"; FLASH 0;AT x-1,y; PAPER 4
; INK 0;"LM": FOR f=0 TO 50: OUT
 254,16: OUT 254,0: NEXT f: LET
li=li-1: IF li>0 THEN FOR f=0 TO
 500: NEXT f: GO TO 500
 820 IF li<=0 THEN GO TO 8000
7000 REM  SUCCESS!!
7010 BORDER 0: PAPER 0: INK 7: C
LS : PRINT AT 10,8;"CONGRATULATI
ONS!";AT 12,0;"YOU HAVE BEATEN Y
OUR WAY THROUGH";AT 13,10;"THE J
UNGLE!!"
7020 FOR f=0 TO 30: BEEP .01,f:
BEEP .01,-f: NEXT f
7030 GO TO 8050
7040 STOP
8000 REM  FAILURE!!
8010 RESTORE : BORDER 0: PAPER 0
: INK 7: CLS
8020 PRINT AT 10,1;"HA! HA! FASC
IST! YOU'VE USED UP  EVERY CHANC
E THAT I GAVE YOU!"
8030 FOR f=1 TO 12: READ a,b: BE
EP a,b: NEXT f
8040 DATA .5,-5,.5,-5,.5,0,.5,0,
.5,2,.5,2,.5,5,.5,10,.5,15,.5,20
,.5,25,2,-20
8050 PRINT AT 15,0;"WOULD YOU LI
KE ANOTHER GO? (Y/N)"
8060 IF INKEY$="y" THEN LET li=3
: GO TO 5
8070 IF INKEY$="n" THEN PRINT US
R 0
8080 GO TO 8060
8090 STOP
8999 STOP
9000 REM  UDG DATA
9001 RESTORE 9000: FOR f=0 TO 15
9: READ a: POKE USR "a"+f,a: NEX
T f
9010 DATA 0,0,1,1,48,96,127,63
9020 DATA 0,0,192,192,134,131,25
5,254
9030 DATA 240,56,188,252,124,63,
6,4
9040 DATA 0,0,0,3,15,15,31,31
9050 DATA 0,0,0,192,240,240,248,
248
9060 DATA 31,31,15,15,3,0,0,0
```



```
9070 DATA 248,248,240,240,192,0,
0,0
9080 DATA 0,0,0,1,3,7,31,63
9090 DATA 0,0,0,192,192,224,240,
240
9100 DATA 0,24,48,56,28,14,7,2
9110 DATA 0,12,6,14,28,56,112,64
9120 DATA 0,234,170,238,0,0,0,0
9130 DATA 0,234,142,234,0,0,0,0
9140 DATA 24,60,126,60,24,24,60,
0
9150 DATA 0,238,68,255,255,68,23
8,0
9160 DATA 90,126,90,24,90,126,90
,24
9170 DATA 0,119,34,255,255,34,11
9,0
9180 DATA 24,90,126,90,24,90,126
,90
9190 DATA 0,6,70,0,0,48,248,254
9200 DATA 0,0,0,24,24,0,0,0
9210 RETURN
```

Dodge

In this fast-moving game of logic and skill, you are heavily disguised as an asterisk. You have to avoid the VBB's (Vicious Black Blobs) which appear at random around you to terrorise and trap you.

As you'll see when you run DODGE, you must enter a 'level of difficulty'. You need to enter a number between 1 and 100, with the higher numbers representing easier games. Level 100 is only for real experts.

You'll have to keep moving to ensure you are not trapped by the VBB's. Use the arrow keys to move around.

```
  20 INPUT "Use cursor keys to m
ove,          Input Difficulty
              1(tricky)-100(easy)
";p
  30 BORDER 4: LET x=10: LET y=1
5: LET s=0: LET a=53
  40 PRINT " ": POKE 23692,255
  50 IF SCREEN$ (x,y)="_" THEN G
O TO 160
  60 PRINT AT x,y;"*": LET s=s+1
  70 PRINT AT x+INT (RND*3)-1,y+
INT (RND*3)-1; INVERSE 1;"_"
  80 PAUSE p
  90 IF CODE INKEY$<53 OR CODE I
NKEY$>56 THEN GO TO 40
 100 LET a=CODE INKEY$
 110 IF a=53 AND y>0 THEN LET y=
y-1
 120 IF a=54 AND y<20 THEN LET x
=x+1
 130 IF a=55 AND x>0 THEN LET x=
x-1
 140 IF a=56 AND y<20 THEN LET y
=y+1
```



```
 150 GO TO 40
 160 PRINT "[illegible] you scored "
;s: PAUSE 600: INPUT "ENTER TO P
LAY AGAIN  ";s$: CLS : RUN
```

Duel Cabbageway

The fiendish mind of Neal Cavalier-Smith thought up this strange little program (and its 'witty' title). The aim of this game is to plant cabbages. You score one point for each cabbage you plant, and lose 10 if you attempt to plant a cabbage on top of a house. Three points will be deducted from your score if you plant them next to a house or under a tree. (I told you it was strange.)

```
   3 LET a=0
   4 PAPER 0: INK 7: BORDER 0
   5 PRINT "[illegible]
[illegible]DUEL -CABBAGE
WAY"
   6 PRINT "You score 1 Point fo
r each       cabbage you plant, b
ut lose  10 if you plant one on
top of a     post (it will not gr
ow there)    and 3 cabbages if yo
u plant themby a house-(beware o
f terraces) or under a tree."
   10 LET a$="▝   ▘": LET s=0: LE
T h=0
   20 DIM b$(12,32)
   30 LET c$="          "
   50 FOR x=1 TO 12
   60 LET c$=c$+" "
   70 LET b$(x)=c$+a$
   80 NEXT x
   90 LET y=13
  100 LET x=(RND*12+1)
  101 PRINT AT 20,12; FLASH 1;"ST
ART"
  110 PRINT AT 21,0: PRINT INK 5;
b$(x)
  120 LET x=x+(RND*2-1)
  125 LET s=s+1
```



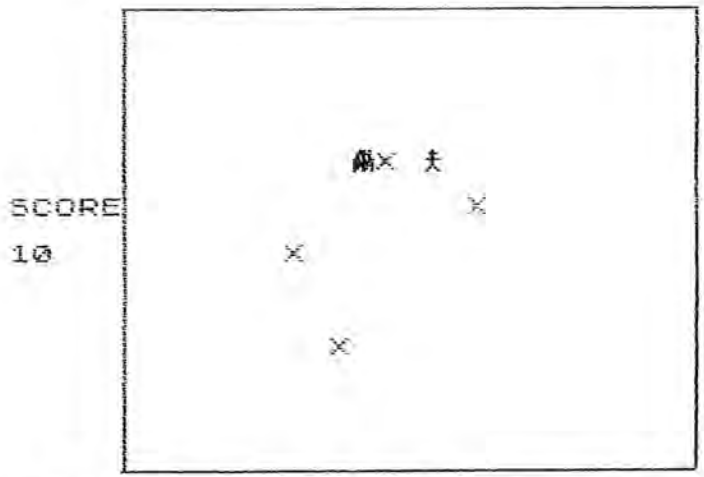
```
  130 IF x>12 THEN LET x=x-1
  140 IF x<1 THEN LET x=x+1
  145 IF INT x=6 AND y=12 OR y=10
 THEN PRINT AT 20,4; INK 2;"▟▙ "
: PRINT AT 21,4; INK 7;" ▙▟ ": LE
T s=s-2
  146 IF INT x=9 AND y=14 THEN PR
INT AT 20,20; INK 4;"▟▙": PRINT
AT 21,20; INK 4;" ▜▛". LET s=s-3
  150 IF INKEY$="5" THEN LET y=y-
1
  160 IF INKEY$="8" THEN LET y=y+
1
  170 IF SCREEN$ (11,y)<>" " THEN
 GO TO 220
  180 PRINT AT 11,y; INK 4;"@"
  190 POKE 23692,255
  200 GO TO 110
  220 BEEP 1,1: LET h=h+1: LET s=
s-10: IF h=3 THEN PRINT "Score="
;s: FOR x=1 TO 10: LET z=TAN x:
NEXT x: PAUSE 50: RUN
  230 GO TO 110
1459 IF INT x=6 AND y=12 OR y=10
 THEN PRINT AT 20,4; INK 2;"▟▙ "
: PRINT AT 21,4; INK 7;" ▙▟ ": LE
T s=s-2
```

Ghost Chase

The ghost is after you. The small playing grid has a number of mines (X) on it, and you have to try and tempt the ghost to come into contact with one of the mines. At this point, the ghost will be electrocuted.

The mines also pose a danger to you, so as you try to evade the evil little ghost, you have to make sure you don't touch mines or the ghost.

Written by Neal Cavalier-Smith and Graham White, this program demonstrates a fairly intelligent ghost in action. You certainly



won't be able to evade it for long. You'll see that the program runs very quickly indeed. If it is too fast for you the first few times you run it, insert a few dummy FOR/NEXT loops to slow it down a bit.

```
    1 GO SUB 600
    2 GO SUB 800
    3 LET q=0: LET s=0: LET p=6:
LET hs=0
    5 BORDER 0: PAPER 0: INK 7: C
LS
   10 LET a=INT (RND*18)+2
   15 PLOT 30,0: DRAW 205,0: DRAW
 0,160: DRAW -205,0: DRAW 0,-160
   20 LET b=INT (RND*12)+10
   30 LET x=2*(INT (RND*8))+2
   40 LET y=2*(INT (RND*10))+10
   43 LET e=2*(INT (RND*6))+6: LE
T f=2*(INT (RND*8))+10
   44 IF e=x OR e=a OR f=y OR f=b
 THEN GO TO 45
   45 PRINT AT e,f; PAPER 2;"X"
   46 IF q=p THEN GO TO 49
   47 LET q=p: FOR z=0 TO 7-p: GO
 TO 45: NEXT z
   49 IF y>29 THEN LET y=29
   50 PRINT AT x,y; INK 5;CHR$ 15
6; LET v=x: LET w=y
   55 REM CHR$(156) is a user
defined man
   60 PRINT AT a,b; INK 6;CHR$ 14
4: LET c=a: LET d=b
   65 REM CHR$(144) is a user
defined ghost
   70 LET a=a-(x<a)+(x>a)
   80 LET b=b-(y<b)+(y>b)
   88
   90 LET x=x+2*(INKEY$="6")-2*(I
NKEY$="7")
   95 LET x=x+2*(x<1)-2*(x>20)
  100 LET y=y+2*(INKEY$="8")-2*(I
NKEY$="5")
  105 LET y=y+2*(y<5)-2*(x>29)
  110 PRINT AT v,w;" ";AT c,d;" "
  115 IF a=x AND b=y THEN GO TO 4
00
  120 IF SCREEN$ (a,b)="X" THEN G
O TO 200
  130 IF SCREEN$ (x,y)="X" THEN G
O TO 300
```

```
140 GO TO 49
200 PRINT AT a,b; FLASH 1;CHR$ 145
205 REM CHR$(145) is a user defined POP
210 FOR d=2 TO 40 STEP 4
211 BEEP d/800,-15
212 NEXT d
215 PRINT AT a,b;" "
216 LET s=s+10
217 PRINT AT 10,0; PAPER 4; FLASH 1;"SCORE"
218 PRINT AT 12,0; FLASH 1; PAPER 0;s
220 GO TO 10
300 PRINT AT x,y; FLASH 1;"*"
310 FOR d=20 TO 1 STEP -1
311 BEEP d/800,25
312 NEXT d
315 PRINT AT x,y;" "
316 LET p=p-1: IF p=0 THEN GO TO 500
320 GO TO 10
400 PRINT AT x,y; FLASH 1;"*"
410 FOR d=1 TO 10
411 BEEP .01,INT (RND*60)-25
412 NEXT d
415 PRINT AT x,y;" ": PRINT AT e,f;" "
416 LET p=p-1: IF p=0 THEN GO TO 500
420 GO TO 10
500 PAPER 1: FLASH 1: CLS
510 PRINT AT 10,12;"O U T  O F ";AT 12,13;"M E N"
515 FOR N=40 TO -20 STEP -1
516 BEEP 0.01,N
517 NEXT N
520 FOR d=1 TO 200: NEXT d
530 FLASH 0: CLS
535 PAPER 0: CLS
540 IF hs<s THEN LET hs=s
550 PRINT AT 5,10; PAPER 3;"HIGH SCORE ";hs;AT 10,0; PAPER 5; INK 0;"PRESS ANY KEY FOR ANOTHER GO"
560 LET A$=INKEY$: IF A$="" THEN GO TO 560
585 LET p=5: LET s=0
590 GO TO 5
600 FOR n=0 TO 7
610 READ d: POKE USR "a"+n,d
620 NEXT n
```



```
630 DATA 28,62,42,107,127,127,109,73
640 FOR n=0 TO 7
650 READ d: POKE USR "b"+n,d
660 NEXT n
670 DATA 145,82,0,195,0,74,137,0
680 FOR n=0 TO 7
690 READ d: POKE USR "m"+n,d
700 NEXT n
710 DATA 8,28,8,62,8,8,20,34
720 RETURN
800 BORDER 0: PAPER 0: INK 7: CLS
805 PRINT TAB 5; INK 6; FLASH 1;" G H O S T  C H A S E "
810 PRINT ''"You have to move your man ("; INK 5;CHR$ 156; INK 7;")   around the screen to entice the ghost ("; INK 6;CHR$ 144; INK 7;")  which is chasing you, to run into the bombs("; PAPER 2;"X"; PAPER 0;").But be  careful you don't run into the bomb or let the ghost catch you. You have 5 lives.                    Good luck........"
820 PRINT '"PRESS KEYS ""5""-""8"" TO MOVE IN   DIRECTION SHOWN ON KEY"
830 PRINT ' FLASH 1;"Press any key to continue"
840 LET a$=INKEY$: IF INKEY$="" THEN GO TO 840
850 RETURN
```

Frog on a Log

Frog on a Log is another great program from Malcolm Young. He says he enjoyed creating this program as with it he had broken new ground in programming techniques. "For example," he wrote, "I have been experimenting with machine code routines and produced a scroll left routine of my own...The computer cannot usually recognize user-defined graphics using SCREEN$, and most people have got around this by using the ATTR function.

"I found that by pointing the system variables CHARS to the UDG memory, SCREEN$ can operate in the normal way, although you must use the letters that the character represents. I have used this technique in my program from lines 300 to 490. This technique has the advantage of recognising any UDG characters, including non-flashing, non-bright black ones.

"I have made the program so it can be typed into the 16K model as well as the 48K model without any modifications, i.e. line 30 CLEAR USR "2"-43 for the machine code."

The object of Malcolm's program is to catch as many insects as possible using keys I and P for left and right, and CAPS SHIFT to make the frog jump. You must avoid going off the screen or falling off

the log and encountering the spider, as this will result in a loss of life. Each fly starts off with 200 energy units and this slowly decreases to zero. When it reaches zero, the frog starves and dies.

Butterflies may appear, and if caught are worth 10 energy points plus 50 bonus points. You have only a 50% chance of catching a butterfly, no matter how accurate you are in your jump.

You'll find this program quite addictive and you may wish to modify it to include a skill level option, which changes the size and number of the logs. You should type the program in and then save it by using SAVE "frog" LINE 5000.

```
   1 REM a=   b=   c=   d=   e=   f=
         gh=    ij=    lm=
  10 REM frog on a log
  20 DEF FN r(x)=INT (RND*x)+1
  30 CLEAR USR "a"-43
  40 GO SUB 9000: REM udg
  45 GO SUB 8110
  50 GO SUB 8000: REM display
  60 GO TO 500
  70 POKE 23606,0: POKE 23607,60
  80 PRINT AT 0,0; BRIGHT 1; INK
 7; PAPER 2;"SCORE: ";sc;TAB 11;
"ENERGY: ";en;TAB 24;"FROGS: ";f
rog
  90 RETURN
 100 REM read keyboard
 110 LET k=PEEK 23556
 120 LET x=x+(k=80)-(k=73)
 130 IF x<1 OR x>31 THEN GO TO 3
30
 140 LET k=IN 65278
 145 GO TO (k=254 OR jp)*ds+150
 150 PRINT AT y,x-1;"   "
 160 IF NOT POINT (x*8,31) THEN
GO TO 450
```

```
 165 RETURN
 170 REM jump
 180 LET jp=1: LET bp=10: LET ds
=40: PRINT AT y,x-1;"   "
 190 POKE 23606,iw: POKE 23607,b
p
 200 FOR i=0 TO 1
 210 LET y=y-jp
 215 GO TO (y=17)*180+220
 220 IF SCREEN$ (y,x)<>"i" THEN G
O TO 300
 230 PRINT AT y-1,x;"a" AND jp>0
;AT y,x;"c";AT y+1,x;"d";AT y+2,
x;"a" AND (jp)0 AND y<15): BEEP
.005,bp
 240 NEXT i
 260 LET bp=bp+jp
 270 LET jp=jp-2*(y<3)+(y>15)
 280 POKE 23606,0: POKE 23607,60
 290 RETURN
 300 LET a$=SCREEN$ (y,x)
 310 IF a$="i" OR a$="j" THEN LE
T e=(y-1)*32+x-2: LET j$(e TO e+
2)="   ": GO TO 350
 320 IF a$="g" OR a$="h" THEN LE
T b=0: LET en=en+10: LET s(=s(+3
0: GO TO 350
 330 FOR z=20 TO 30: BEEP .05,-z
: BEEP .1,-z-.5: NEXT z
 340 LET frog=frog-1: LET s=0: B
EEP 1,-10: IF NOT frog THEN GO T
O 1000
 345 LET en=200: GO TO 6000
 350 LET sc=sc+10: LET jp=-1
 360 BEEP .5,10
 370 GO SUB 70
 380 RETURN
 400 IF SCREEN$ (13,x)<>"e" THEN
 GO TO 450
 410 GO SUB 8080
 420 PRINT AT y-1,x;"c";AT y,x;"
d"
 430 PRINT AT 16,x-1;"aaa";AT 17
,x;"b"
 440 RETURN
 450 BEEP .05,-5
 460 PRINT AT 16,x;" ";AT 17,x;"
";AT 16,x;"";
 470 BEEP .05,-5
 480 PRINT AT 17,x;" ";AT 18,x;"
": BEEP .1,-5: PRINT AT 18,x;"
```



```
 490 GO TO 330
 500 REM main routine
 520 IF RND>.05 THEN GO TO 560
 530 LET s1=32*(FN r(6)+9)-1
 540 LET i$(s1 TO s1+1)=""
 560 PRINT AT 1,0; INK 1;i$
 570 LET i$=i$(3 TO )
 580 LET x=x-NOT jp: LET l=USR a
$
 600 BEEP .005,10
 610 GO SUB 100
 620 GO TO (NOT b)*40+630
 630 LET d=FN r(4): LET b=b-1
 640 LET by=by+(d=1)-(d=2): LET
bx=bx+(d=3)-(d=4)
 650 LET by=by-(by>15)+(by<1): L
ET bx=bx+(bx<0)-(bx>30)
 660 PRINT AT by,bx; INK 2; OVER
 1;""
 670 IF NOT b THEN LET b=(RND>.9
)*(FN r(20)+5): IF b THEN LET by
=FN r(15)+1: LET bx=FN r(30)
 680 REM spider
 690 LET s=s+(RND>.4)*((s<x)-(s>
x))
 700 PRINT INK 0;AT 16,0;q$;AT 1
5,s;""
 710 IF y>15 AND (x=s OR x=s+1)
THEN GO TO 810
 720 LET en=en-1
 730 PRINT AT 0,19; BRIGHT 1; IN
K 7; PAPER 2;en;" "
 740 IF en<0 THEN PRINT PAPER 2;
AT 8,7; FLASH 1;"No More Energy!
": GO SUB 330
 790 GO SUB 100
 800 GO TO 500
 810 PRINT INK 0;AT 16,s;"  ";AT
 17,s;""
 820 FOR z=0 TO 1 STEP .1
 830 BEEP z,-z*10: NEXT z
 840 GO SUB 340
 850 GO TO 500
1000 REM score table
1010 GO SUB 5000: GO SUB 70
1020 IF sc<=hs THEN GO TO 1050
1030 INPUT "You have the new hig
h score!","Please type in your n
ame. "; LINE n$
1040 LET hs=sc
1050 DIM a$(1)
1060 INPUT "Do you want to play
again?";a$
```

```
1070 IF a$="n" OR a$="N" THEN PO
KE 23693,56: PRINT AT 20,10;"OK,
play again soon.": PAUSE 50: BOR
DER 7: RANDOMIZE USR 4750
1080 GO SUB 8120
1090 GO TO 50
5000 REM Title
5010 BORDER 5: PAPER 1: INK 4: C
LS : PRINT AT 2,0;
5020 PRINT TAB 4;"████";TAB 20;"█
█"
5030 PRINT TAB 4;"█";TAB 19;"█
█"
5040 PRINT TAB 4;"███ █▀▄ ▄▀▀▄ ▄▀▀▄
 █  █ █▀▄"
5050 PRINT TAB 4;"█  █  █ █ █
 █  █ █"
5060 PRINT TAB 4;"█  █  ▀▄▄▀ ▀▄▄▀
 ██  █ █"
5070 PRINT TAB 16;"█";TAB 14;"▀▄▄
▀"
5080 PRINT 'TAB 6;"██      █"
5090 PRINT TAB 5;"████    █"
5100 PRINT TAB 5;"████    █ ▄▀▀▄
▄▀▀▀"
5110 PRINT TAB 5;"█  █    █  █ █
█ █"
5120 PRINT TAB 5;"█  █   ███ ▀▄▄▀
▀▄▄▀"
5130 PRINT TAB 22;"█";TAB 20;"▀▄▄
▀"
5135 RESTORE
5140 FOR a=1 TO 10
5150 READ n: IF n=99 THEN PAUSE
5: GO TO 5170
5160 BEEP .15,n
5170 NEXT a
5180 LET x=PEEK 23730+256*PEEK 2
3731
5190 IF x=USR "a"-1 THEN RUN
5200 RETURN
5500 DATA 2,5,3,99,2,5,3,99,5,10
8000 REM set display
8010 BORDER 2: PAPER 5: INK 6: C
LS
8020 GO SUB 70
8040 PRINT PAPER 3; INK 4;AT 17,
0;TAB 31;" ";TAB 31;" "
8050 PRINT PAPER 1;TAB 31;" ";TA
B 31;" "
8053 PAPER 8
```



```
8055 PRINT AT 18,0;"▒▒▒  ▒▒▒ ▒▒▒▒▒
▒▒▒   ▒▒▒   ▒▒▒   ▒▒▒"
8057 PRINT AT 19,0;"▒▒▒  ▒▒▒ ▒▒▒▒▒
▒▒▒   ▒▒▒   ▒▒▒   ▒▒▒"
8060 PRINT AT 21,0; PAPER 2; INK
 7; BRIGHT 1;" HIGH SCORE ";hs;"
 ";n$
8070 LET y=17: LET x=19
8080 LET jp=0
8090 LET ds=20
8100 RETURN
8110 LET hs=0: LET n$=""
8120 DIM i$(480)
8160 LET b=0: LET w=0
8170 LET n=USR "a"-776: RANDOMIZ
E n: LET lw=PEEK 23670: LET hh=P
EEK 23671: RANDOMIZE
8180 LET b=0: LET w=0
8190 LET s=0
8200 LET frog=3
8210 LET en=200
8220 LET sc=0
8240 DIM q$(32)
8300 RETURN
9000 REM define graphics
9010 RESTORE 9000
9020 FOR c=1 TO 12: READ c$
9030 FOR b=0 TO 7: READ n
9040 POKE USR c$+b,n
9050 NEXT b: NEXT c
9060 LET mc=USR "a"-42
9070 FOR m=mc TO mc+41
9080 READ byte: POKE m,byte
9090 NEXT m
9190 RETURN
9195 DATA "a",0,0,0,0,0,0,0,0
9200 DATA "b",36,126,60,153,126,
126,60,99: REM frog
9210 DATA "c",165,231,66,129,153
,153,126,60: REM 1/2jumping frog
9220 DATA "d",126,126,60,90,129,
195,36,195: REM 1/2 jumping frog
9230 DATA "e",170,85,170,85,170,
85,170,85
9240 DATA "f",170,85,170,85,0,0,
0,0: REM log (bottom)
9250 DATA "g",17,109,86,130,170,
86,109,17: REM 1/2 butterfly
9260 DATA "h",16,108,212,130,170
,212,108,16: REM 1/2 butterfly
9270 DATA "i",1,2,4,73,253,56,86
,73: REM insect
```

```
9280 DATA "j",224,16,112,128,85,
0,0,0: REM insect tail
9300 DATA "l",0,34,83,159,166,17
1,144,80: REM spider
9310 DATA "m",0,68,202,242,98,21
0,10,10: REM spider
9320 REM machine code scroll lef
t
9330 DATA 33,32,80,62,3,50,129,9
2,6,8,197,229,126,245,84,93,35
9340 DATA 1,31,0,237,176,241,16,
225,36,193,16,237,167,17
9350 DATA 224,7,237,82,58,129,92
,61,32,220,201
```


LEISURE LINES

Card Pairs

This game - written by Raymond Blake - is based on Pangolines, in which you have to try to pick pairs of cards from a pack of 20 until no cards are left. The aim is to do this in as few moves as possible.

When prompted, enter the coordinates of the card to be turned, entering the column number first. The chosen card will then be turned over. You do this again, so two cards are showing. If the two cards are the same, they are removed. If not, they are turned face down again. The game ends when there are no cards left, at which point the score will be given. The lower the score the better. Raymond Blake challenges you to beat his best score of 17 turns.

```
  30 BORDER 0: PAPER 0: INK 7: C
LS
  50 DIM a(4,5): RANDOMIZE
  60 LET a$="1112131415212222324253132333435414243444 5"
  80 FOR i=1 TO 10
  90 FOR j=1 TO 2
 100 LET l=LEN a$: LET b=INT (RN
D*(l/2)*2+1: LET b$=a$(b TO b+1)
 120 LET x=VAL b$(1): LET y=VAL
b$(2)
 130 LET a$=a$( TO b-1)+(a$(b+2
TO ) AND b<>(l-1)
 150 LET a(x,y)=i: NEXT j: NEXT
i
 200 LET sc=0
 210 FOR i=1 TO 5: PRINT AT 0,4*
i-1; INK 2; INVERSE 1; BRIGHT 1;
i: NEXT i
```

```
 220 FOR j=0 TO 3: PRINT AT j*5+
3,0; INK 2; INVERSE 1; BRIGHT 1;
j+1: FOR i=0 TO 4: FOR k=2 TO 5:
 PRINT AT j*5+k,i*4+2; INK 6;"▓▓
▓": NEXT k: BEEP .03,30: NEXT i:
 NEXT j
 330 PRINT AT 6,23;"Turn over";A
T 8,25;" "
 340 IF INKEY$<>"" THEN GO TO 34
0
 350 LET q$=INKEY$: IF q$<"1" OR
 q$>"4" THEN GO TO 350
 360 PRINT AT 8,24;q$: BEEP .05,
20: LET q1=VAL q$
 370 IF INKEY$<>"" THEN GO TO 37
0
 380 LET q$=INKEY$: IF q$<"1" OR
 q$>"5" THEN GO TO 380
 390 PRINT AT 8,26;q$: BEEP .05,
20: LET q2=VAL q$
 400 IF a(q1,q2)<>0 THEN GO TO 4
50
 410 PRINT AT 10,23; INVERSE 1;"
Try again"
 420 BEEP 1,-10: PRINT AT 8,24;"
   ";AT 10,23;"
 430 GO TO 330
 450 PRINT AT 6,23;"          ";A
T 8,24;"  "
 470 LET x1=(q1-1)*5+2: LET y1=(
q2-1)*4+2: LET x=x1: LET y=y1: L
ET c=q1: LET d=q2: GO SUB 800
 490 LET q1=c: LET q2=d
 530 PRINT AT 6,23;"Turn over";A
T 8,25;" "
 540 IF INKEY$<>"" THEN GO TO 54
0
 550 LET q$=INKEY$: IF q$<"1" OR
 q$>"4" THEN GO TO 550
 560 PRINT AT 8,24;q$: BEEP .05,
20: LET p1=VAL q$
 570 IF INKEY$<>"" THEN GO TO 57
0
 580 LET q$=INKEY$: IF q$<"1" OR
 q$>"5" THEN GO TO 580
 590 PRINT AT 8,26;q$: BEEP .05,
20: LET p2=VAL q$
 600 IF a(p1,p2)<>0 THEN GO TO 6
50
 610 PRINT AT 10,23; INVERSE 1;"
Try again"
```



```
 620 BEEP 1,-10: PRINT AT 8,24;"
   ";AT 10,23;"
 630 GO TO 530
 650 PRINT AT 6,23;"          ";A
T 8,24;"  "
 670 LET x2=(p1-1)*5+2: LET y2=(
p2-1)*4+2: LET x=x2: LET y=y2: L
ET c=p1: LET d=p2: GO SUB 800
 690 LET p1=c: LET p2=d
 695 LET sc=sc+1
 700 IF a(q1,q2)<>a(p1,p2) THEN
GO TO 760
 705 FOR i=0 TO 12 STEP 6: BEEP
.1,i: NEXT i
 710 FOR i=0 TO 3: PRINT AT x1+i
,y1;"    ";AT x2+i,y2;"    ": NEXT
 i
 720 LET a(q1,q2)=0: LET a(p1,p2
)=0
 730 LET c=0: FOR i=1 TO 4: FOR
j=1 TO 5: LET c=c+a(i,j): NEXT j
: NEXT i
 740 IF c=0 THEN GO TO 900
 750 GO TO 300
 760 BEEP .6,0
 765 FOR f=1 TO 200: NEXT f: FOR
 i=0 TO 3: PRINT AT x1+i,y1; INK
 6;"▓▓▓▓";AT x2+i,y2;"▓▓▓▓": NEXT
 i
 770 GO TO 300
 800 RESTORE 1000: FOR z=1 TO a(
c,d): READ z$: NEXT z
 820 PRINT AT x,y;z$(1 TO 3);AT x
+1,y;z$(4 TO 6)
 830 PRINT AT x+2,y;z$(7 TO 9);A
T x+3,y;z$(10 TO 12)
 850 RETURN
 900 CLS
 920 PRINT AT 8,2;"You cleared t
he table after";AT 10,11;sc;" tu
rns"
 930 FOR i=1 TO 25: BEEP .01,i:
NEXT i
 940 LET a$="Press any key for a
 new game": PRINT AT 12,2;
 950 FOR i=1 TO LEN a$: PRINT IN
K 6;a$(i);: IF a$(i)<>" " THEN B
EEP .03,20:
 952 IF a$(i)=" " THEN PAUSE 2
```

```
955 NEXT i
960 IF INKEY$="" THEN GO TO 960
970 RUN
1000 DATA "[illegible]"
1010 DATA "[illegible]"
```


Sub Search

In this challenging game from Malcolm Young, you have to track down and destroy a submarine in the shortest possible time, by employing your fleet in a strategically sound manner.

You control a fleet of warships made up of a battle cruiser, two patrol vessels and three frigates. Your task is to hunt and destroy an enemy submarine which has been lurking around the shipping lanes near you. Your battle cruiser is equipped with a sonar system which rises in pitch as it gets closer to the sub.

Each vessel can carry only a limited quantity of depth charges. Once they've been fired, your ship must return to port. Full instructions are within the program.

```
  10 REM Sub Search
  20 REM © 1983 Malcolm Young
  30 REM a=[illegible] b=[illegible] c=[illegible] d=[illegible] e=[illegible] f=[illegible] g=[illegible] h=[illegible] i=[illegible] j=[illegible] k=[illegible] l=[illegible] m=[illegible] n=[illegible]
  40 DEF FN b()=SQR (FN s(sx,s(1,1))+FN s(sy,s(1,2)))
  50 DEF FN s(a,b)=ABS (a-b)*ABS (a-b)
  60 DEF FN r(x)=INT (RND*x)+1
  70 DEF FN t(j,k,l,m)=(ABS (j-k)<7)*(l=m)
  80 GO SUB 9000: REM define graphics
  90 GO SUB 8000: REM instructions
 100 GO SUB 7000: REM set up screen
 110 REM vvvvvvvvvvvvvvvvvvvvvvvv
 120 REM move submarine
```

```
130 LET sm=FN r(6)
140 LET sx=sx+(sm=1)-(sm=2)
150 LET sy=sy+(sm=3)-(sm=4)
160 LET sd=sd+(sm=5)-(sm=6)
170 LET sx=sx+(sx<1)-(sx>19)
180 LET sy=sy+(sy<1)-(sy>19)
190 LET sd=sd+(sd<1)-(sd>9)
210 FOR c=1 TO 6
220 IF FN t(sx,s(c,1),sy,s(c,2)
) THEN GO TO 5000
230 IF FN t(sy,s(c,2),sx,s(c,1)
) THEN GO TO 5000
240 NEXT c
250 REM player's move
260 PRINT #0;" Press any key to
throw dice."
270 IF s(1,1)>=0 THEN PRINT AT
15,21;"Sonar";AT 16,21;"reads:"
; INK 3; FLASH 1;INT FN b(); FLA
SH 0; INK 5;" "
280 IF s(1,1)>=0 THEN BEEP .1,1
0-FN b(): PAUSE 10
290 IF INKEY$="" THEN GO TO 280
300 REM dice
305 LET nb=nb+1
310 LET d=FN r(20)+10: LET m=FN
r(6)
320 FOR i=1 TO d
330 LET m=m*(m<6)+1: BEEP .1,m
340 PRINT AT 18,21;"You have ";
AT 19,21;CHR$ (151+m);" moves."
350 NEXT i
360 PRINT BRIGHT 1; FLASH 1; IN
K m; PAPER 9;AT 19,21;CHR$ (151+
m)
370 BEEP 1,10: LET n=0
380 LET n=n*(n<6)+1
390 LET p=s(n,2): LET r=s(n,1)
395 IF r<0 THEN GO TO 380
400 PRINT FLASH 1; INK n-1; PAP
ER 7;AT p,r;d$(n)
410 PRINT PAPER n-1; INK 9;AT 2
0,8;s$(n);"(";s(n,1);",";s(n,2);
")"
420 PRINT AT 21,8;"depth charge
s left:";s(n,3)
430 INPUT AT 0,0;"Enter move. "
, LINE m$: IF m$="" THEN PRINT I
NK n-1; PAPER 5+(n=6);AT p,r;d$(
n): GO TO 380
440 LET l=LEN m$: IF l>m THEN B
EEP 1,-10: GO TO 430
450 FOR c=1 TO l
```



```
460 IF m$(c)<>"n" AND m$(c)<>"s
" AND m$(c)<>"w" AND m$(c)<>"e"
AND m$(c)<>"d" THEN INPUT AT 0,0
;"Invalid entry.Try again. "; LI
NE m$: GO TO 440
470 IF m$(c)="d" THEN GO TO 670
480 NEXT c
490 REM move ship
500 FOR q=1 TO l
510 PRINT AT p,r; PAPER 5; BRIG
HT 1; INK 0;"L"
520 BEEP .1,15
530 LET p=p+(m$(q)="s")-(m$(q)=
"n")+(p<0)-(p>20)
535 IF p=20 AND r<6 THEN LET s(
n,3)=3: BEEP 1,20: PRINT AT 21,0
;"3 Depth Charges loaded.": LET
p=19
540 LET r=r+(m$(q)="e")-(m$(q)=
"w")+(r<0)-(r>20)
550 IF ATTR (p,r)<64 THEN GO TO
3000
560 PRINT AT p,r; PAPER 5; INK
n-(n<>6); OVER 1;d$(n)
570 BEEP .1,20
580 LET m=m-1
590 PRINT BRIGHT 1; FLASH 1; IN
K m; PAPER 9;AT 19,21;CHR$ (151+
m)
600 NEXT q
610 LET s(n,1)=r: LET s(n,2)=p:
IF NOT u THEN GO TO 650
620 INPUT "Do you want to fire
a charge?"; LINE i$
630 IF i$<>"y" AND i$<>"n" THEN
INPUT "Fire a depth charge? ";
LINE i$: GO TO 630
640 IF i$="y" THEN GO TO 670
650 IF m>0 THEN GO TO 380
660 GO TO 120
670 REM fire depth charge
675 IF s(n,3)<1 THEN PRINT FLAS
H 1;AT 21,0;"No depth charges le
ft!": GO TO 650
680 PRINT AT 2,21; PAPER 6; INK
0;"YELLOW"
690 FOR a=1 TO 7: BORDER 7: BEE
P .3,-1: BORDER 6: PAUSE 5: NEXT
a
700 PRINT FLASH 1; INK 7; PAPER
3;AT 21,3;"Depth charges ready!
"
```

```
 710 INPUT "Enter depth, (1 TO 1
0) ";dp
 720 IF dp<1 OR dp>10 THEN INPUT
 (dp;" is out of range,try again
");dp: GO TO 720
 730 LET dp=INT dp
 740 BEEP .5,-10: BEEP .2,20
 750 FOR f=19 TO -dp STEP -1
 760 BEEP .02,f
 770 NEXT f
 780 BEEP .8,-10: PAUSE 20
 790 IF sx=r AND sy=p THEN BEEP
.5,1: IF sd=dp THEN GO TO 1000
 800 LET s(n,3)=s(n,3)-1
 810 PRINT FLASH 1;AT 21,0;"Miss
ed!": IF sx=r AND sy=p THEN PRIN
T AT 21,6;"The sub's right under
 you!"
 820 PAUSE 100
 830 PRINT AT 21,0;"
                    "
 840 GO TO 110
1000 REM sub destroyed
1010 BORDER 1: PAPER 1: BRIGHT 1
: INK 7: CLS
1020 PRINT AT 2,3;"CONGRATULATIO
NS!"
1030 PRINT AT 4,3;"You manage to
 destroy the"
1040 PRINT AT 6,3;"sub in ";nb;"
 rounds."
1050 PRINT "That's ";pc/nb*100;"
% efficiency."
1060 INPUT "Would you like to pl
ay again? "; LINE i$
1070 IF i$="y" THEN RUN
1080 BORDER 7: PAPER 7: INK 0: B
RIGHT 0: CLS : STOP
3000 REM collision
3010 BEEP 1,-12
3020 PRINT AT 21,0;"COLLISION!";
3030 IF ATTR (p,r)=40 THEN PRINT
 AT 20,0; FLASH 1; INK 0; PAPER
7;s$(n);" has sunk!": LET s(n,1)
=-1
3040 IF ATTR (p,r)<>n*8+n-1 THEN
 GO TO 3100
3050 PRINT AT 20,0; FLASH 1; INK
 0; PAPER 7;s$(n);" has sunk!":
LET s(n,1)=-1
3060 FOR a=1 TO 0 STEP -.1: BEEP
 a,a*5-10: NEXT a
```



```
3070 PAUSE 100
3080 PRINT AT 20,0;TAB 31;" ";TA
B 31;" "
3090 GO TO 110
3100 LET s=ATTR (p,r)-40
3110 STOP
4999 REM ↑↑↑↑↑↑↑↑↑↑↑↑↑↑↑↑↑↑↑↑↑↑↑↑
5000 REM submarine attack
5005 IF RND>.7 THEN GO TO 250
5010 BORDER 2: LET u=1: PRINT AT
 2,21; PAPER 2; INK 7;" RED "
5020 FOR a=1 TO 7: BORDER 4: BEE
P .5,-5: BORDER 2: PAUSE 5: NEXT
 a
5030 IF RND<0.5 THEN GO TO 5500
5040 PRINT AT 21,0; INK 7; BRIGH
T 1;s$(c);" under attack!"
5045 PAUSE 50
5050 LET p=5
5060 PRINT AT 21,0; PAPER 6; FLA
SH 1;"Press 'F' and hope for the
 best."
5070 FOR t=1 TO 2
5080 LET t1=FN r(10): LET pf=(t1
*8+164)-208
5090 PRINT AT 8,20+p; PAPER 5; I
NK 1;"+"
5100 LET p=p+(p<10)+1
5110 PAUSE 10
5120 PRINT AT 8,20+p; PAPER 5; I
NK c-(c<>6);d$(c)
5130 PAUSE 10
5140 IF INKEY$<>"f" THEN GO TO 5
090
5150 FOR i=7 TO 1 STEP -1
5160 PLOT 208,62: DRAW INVERSE 1
; INK i;pf,40
5170 BEEP .05,a: NEXT i
5180 PLOT 208,62: DRAW INK 1;pf,
40: PAUSE 50
5190 IF t1=p THEN GO TO 5300
5200 PRINT AT 21,0; PAPER 2; INK
 7;"  That was good evasive acti
on! "
5210 PAUSE 100: PRINT AT 21,3;"W
atch out!Another torpedo!": FOR
 a=1 TO 7: BEEP .05,a: NEXT a
5220 PAUSE 50: PRINT AT 21,0;TAB
 31;" "
5230 NEXT t
5240 PRINT AT 20,0;"False Alarm!
 Damage ";u-1;" Hits"
```

```
5250 PAUSE 50: PRINT AT 20,0;TAB
 31;" ";TAB 31;" "
5260 GO TO 250
5300 REM hit
5310 IF u=1 THEN PRINT AT 21,0;"
a HIT!": BEEP .8,-5: PAUSE 50: P
RINT AT 20,6;"The ";s$(c);" has
  ejected all it's depth charges
.": LET s(c,3)=0
5320 IF u=2 THEN PRINT FLASH 1;
INK 7; PAPER 0; BRIGHT 1;AT 21,0
;s$(c);" has been lost!"
5330 LET u=u+1: PAUSE 100
5340 IF u=2 THEN NEXT t
5350 PRINT BRIGHT 1;AT s(c,2),s(
c,1); INK 0; PAPER 5;"┌"
5360 LET s(c,1)=-1
5370 FOR a=1 TO 0 STEP -0.1
5380 BEEP a,a*5-10
5390 NEXT a
5400 FOR a=1 TO 20: BEEP 0.0025,
20: NEXT a
5410 GO TO 250
5500 REM submarine in close prox
imity
5510 PRINT AT 21,0; FLASH 1; BRI
GHT 1; INK 7;"Alert!"
5520 PAUSE 100
5530 PRINT AT 21,0;s$(c);" has s
potted sub!"
5550 PAUSE 100
5560 GO TO 250
6999 GO TO 9999
7000 REM display
7010 BORDER 4: PAPER 5: INK 1: B
RIGHT 0: CLS
7020 FOR i=0 TO 19: PRINT BRIGHT
 1;"┌┬┬┬┬┬┬┬┬┬┬┬┬┬┬┬┬┬┐": REM 2
0 graphic 'f'
7030 NEXT i
7040 PLOT 0,15: DRAW 160,0: DRAW
 0,160
7050 REM mines
7060 LET m=FN r(10)+10
7070 FOR i=1 TO m: LET mx=FN r(1
9): LET my=FN r(19): IF ATTR (my
,mx)>64 THEN PRINT AT my,mx;"$"
7075 NEXT i
7080 REM ships
7090 FOR a=1 TO 6: PRINT AT s(a,
2),s(a,1); OVER 1; BRIGHT 1; PAP
ER 7; INK a-1;d$(a): NEXT a
```



```
7100 PRINT AT 0,21; FLASH 1;"SUB
 SEARCH"; FLASH 0;AT 1,21;"CONDI
TION";AT 2,21; PAPER 4;"GREEN"
7110 REM periscope
7120 CIRCLE 208,102,40
7130 FOR p=40 TO 0 STEP -1
7140 PLOT 208-p,102: DRAW p*2,0,
PI
7150 NEXT p
7160 FOR i=0 TO 9: PRINT OVER 1;
AT 4+i,25;" + ": LET i=i+(i=3): N
EXT i
7170 PRINT OVER 1;AT 8,21;"+++++
+++++"
7180 RETURN
8000 REM instructions
8010 BORDER 1: PAPER 5: INK 1: B
RIGHT 1: CLS
8020 DIM s$(6,15): DIM s(6,3): D
IM i$(1): LET u=0: LET i=1: LET
d$="▚▚▚▚▚▚▚▚": REM graphic b,c,c,c
,a,a
8030 FOR a=1 TO 5: BEEP .1,-a: P
RINT AT 1,9; INK 2; INVERSE i;"S
UB SEARCH": BEEP .25,a: LET i=1-
i: NEXT a
8040 INPUT BRIGHT 1;"Do you want
 to read instructions(y/n)?  ";i
$
8050 IF i$<>"y" THEN GO TO 8210
8060 FOR a=1 TO 6: READ p$: READ
 a$
8070 PRINT AT 3,0;p$''''a$
8080 DIM b$(LEN p$): DIM t$(LEN
a$)
8090 INPUT "Press "; FLASH 1;"EN
TER"; FLASH 0;" to continue. ";
 LINE p$
8100 PRINT AT 3,0;b$''''t$: NEXT
 a
8110 IF p$="n" THEN GO TO 8210
8120 IF p$<>"y" THEN INPUT "DO y
ou want to name your own     flee
t? Answer (y/n) ";p$: GO TO 8110
8130 REM rename ships
8140 POKE 23658,8: RESTORE 9300:
 BORDER 4: PAPER 5: BRIGHT 0: CL
S
8150 FOR a=1 TO 6: READ c$
8160 PRINT AT 0,0;"What would yo
u like to call"'c$;"
: INPUT n$
```

```
8170 IF LEN n$>15 OR n$="" THEN
INPUT "No you can't have that,tr
y again";n$: GO TO 8170
8180 LET s$(a)=n$
8190 PRINT PAPER 8-a; INK a-1;AT
 a+5,0;n$;" ";d$(a): NEXT a
8200 POKE 23658,0: GO TO 8230
8210 REM name ships
8220 RESTORE 9310: FOR a=1 TO 6:
 READ c$: LET s$(a)=c$: NEXT a
8230 FOR i=1 TO 6: READ d: LET s
(i,1)=i-1: LET s(i,2)=20: LET s(
i,3)=d: NEXT i
8240 LET sx=FN r(20)-1: LET sy=F
N r(20)-1: LET sd=FN r(10)
8250 LET pc=FN b(): LET nb=0: RE
TURN
9000 REM Data
9010 REM Define graphics
9020 FOR c=1 TO 14: READ c$
9030 FOR b=0 TO 7: READ bt: POKE
 USR c$+b,bt: NEXT b
9040 NEXT c: RETURN
9050 DATA "a",0,8,8,12,28,127,12
8,0: REM patrol boat
9060 DATA "b",16,16,59,186,255,2
55,126,0: REM battle cruiser
9070 DATA "c",0,8,8,204,92,255,2
04,0: REM frigate
9080 DATA "d",137,74,44,31,248,5
2,82,145: REM explosion
9090 DATA "e",0,40,46,120,30,116
,20,0: REM mine
9100 DATA "f",255,128,128,128,12
8,128,128,128: REM grid
9110 DATA "g",0,0,1,1,7,1,1,0: R
EM 4
9120 DATA "h",0,0,0,0,224,0,0,0:
 REM -
9130 DATA "i",0,0,0,24,24,0,0,0:
 REM dice no.1
9140 DATA "j",0,96,96,0,0,6,6,0:
 REM dice no.2
9150 DATA "k",3,3,0,24,24,0,192,
192: REM dice no.3
9160 DATA "l",0,102,102,0,0,102,
102,0: REM dice no.4
9170 DATA "m",195,195,0,24,24,0,
195,195: REM dice no.5
9180 DATA "n",102,102,0,102,102,
0,102,102: REM dice no.6
```



```
9190 DATA "You control a fleet o
f warships consisting of:1 Battl
e cruiser, 2 Patrol vessels and
3 frigates."
9200 DATA "Your task is to hunt
and destroyan enemy submarine wh
ich has    been lurking around t
he shippinglanes near you."
9210 DATA "Your battlecruiser is
 equipped  with a sonar system
that rises in pitch as it gets c
loser to   the sub."
9220 DATA "Each vessel can carry
 only a    limited amount of dep
th charges and once they are fir
ed the shipmust return to port.T
he limits  are 4 to the battle c
ruiser          3 to each of the
frigates   and 2 to the patrol b
oats."


9230 DATA "The submarine may tor
pedo any   ship that is within a
 range of  5 units and is in lin
e either   horizontally or verti
cally."
9240 DATA "The submarine moves o
ne unit    each round and may hi
de up to a depth of 10 units.If
you move 1 of your ships directl
y above thesubmarine an alarm wi
ll sound."


9250 DATA "The number of moves y
ou get each round depends on a
dice',you  may move any number o
f ships as long as it is within
the limits."
9260 DATA "The computer will seq
uence       through each ship and
 ask you ifyou want to move it.A
nswer with 'ENTER' if you don't
or use the directions n,s,e,w or
 'd' for   depth charge."
9270 DATA "If you navigate your
ship into  another ship then the
 smaller   ship will be sunk.If
you crash  into mine your ship w
ill sink."
```

```
9280 DATA " ","The ships under y
our command are-HMS INVINCIBLE (
your flagship)  -HMS SHEFFIELD (f
rigate 1)       -RNZS OTAGO (frig
ate 2)          -RNZS CANTERBURY
(frigate 3)     -HMS BRITANNIA (p
atrol boat 1)   -USS TITANIC (pat
rol boat 2)"
9290 DATA "Would you like to nam
e your own fleet? Answer (y/n)."
9300 DATA "your Battlecruiser (f
lagship)","Frigate 1","Frigate 2
","Frigate 3","Patrol boat 1","P
atrol boat 2"
9310 DATA "HMS INVINCIBLE","HMS
SHEFFIELD","RNZS OTAGO","RNZS CA
NTERBURY","HMS BRITANNIA","USS T
ITANIC"
9320 DATA 4,3,3,3,2,2
9999 STOP : BORDER 7: PAPER 7: I
NK 0: BRIGHT 0: INVERSE 0: CLS :
 STOP
```


Hangman

This is a superior version (by Raymond Blake) of the ever-popular hangman game. It has been designed so you can customise it to your own requirements and the word list has been left small deliberately to encourage this.

To alter the words, simple change the 30 in line 70 to the total number of words in the list, and enter the words into DATA statements at the end of the program in the same fashion as lines 1500 to 1520. Witty comments and the scope for alterations can make running this game a long-lasting enjoyable experience. It is even better if you can get a friend to change the word list for you, so you don't know what the words are. As stated in line 20, CAPS LOCK must be engaged before running the program.

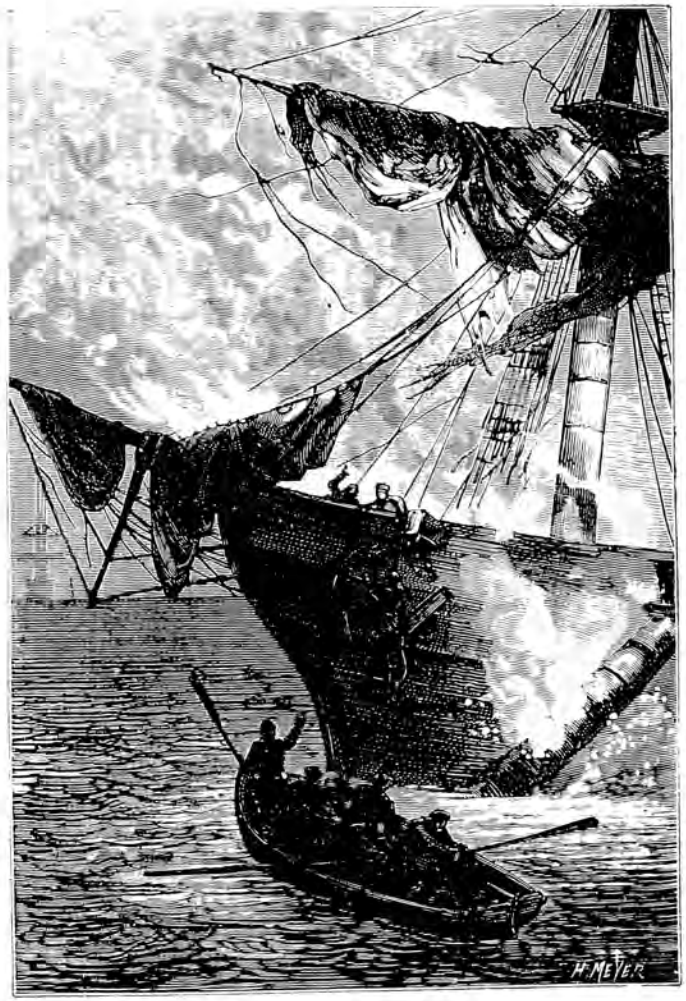
```
  20 REM     ENGAGE CAPS LOCK
             BEFORE RUNNING
  25 LET F=0: RANDOMIZE
  30 BORDER 5: PAPER 7: INK 0: C
LS : RESTORE
  40 PRINT AT 0,7;"SPECTRUM HANG
MAN"; INK 2;AT 1,7;"--------- ---
-----"
  50 IF F=0 THEN PRINT AT 3,0;"H
elp! These madmen will hang me";
AT 5,1;"if you can't guess their
 word."
  55 IF F=1 THEN PRINT AT 3,0;"O
h no, now they went to hang my"
;"friend! You've got to help him
."
  60 LET W=0: LET W$="": LET C=0
```

```
  70 FOR I=1 TO INT (RND*30+1):
READ A$: NEXT I
  75 FOR I=-20 TO 30: BEEP .02,I
: NEXT I
  80 LET L=LEN A$: LET B$="-----
-----"( TO L)
  90 PRINT AT 12,3; INK 2;B$
 100 PRINT AT 8,0; INK 1;"Enter
your guess"
 120 INPUT C$: IF LEN C$<>1 AND
LEN C$<>L THEN GO TO 120
 130 IF C$<"A" OR C$>"Z" THEN GO
 TO 120
 140 PRINT AT 8,0;"
    "
 150 IF LEN C$=1 THEN GO TO 180
 160 IF C$=A$ THEN LET B$=C$
 170 IF C$<>A$ THEN LET W=L
 175 GO TO 200
 180 LET W=0: FOR I=1 TO L: IF A
$(I)=C$ THEN LET B$(I)=C$
 190 IF C$<>A$(I) THEN LET W=W+1
 195 NEXT I
 200 PRINT AT 12,3; INK 2;B$: IF
 W=L THEN LET C=C+1: GO SUB 700+
C*20: BEEP .6,5*(10-C)-25: LET W
$=W$+(C$ AND LEN C$=1)
 210 IF W<>L THEN FOR I=10 TO 30
: BEEP .02,I: NEXT I
 220 PRINT AT 17,0;"Wrong letter
s": PRINT AT 19,2; INK 4;W$
 240 IF A$=B$ THEN GO TO 500
 250 IF C=10 THEN GO TO 500
 260 GO TO 100

 500 PRINT AT 3,0; INK 3;" Ouch!
 Next time YOU want help,"'"Don
't come running to me, moron!"
 520 FOR I=20 TO -30 STEP -1: BE
EP .02,I: NEXT I: GO TO 650
 600 PRINT AT 3,0; INK 3;" Thank
s very much. Remind me I "'"Owe
 you a large drink, partner!"
 620 FOR I=-20 TO 30: BEEP .02,I
: NEXT I
 650 PRINT AT 10,2; INK 1;"The w
ord was:";AT 12,3;A$
 660 PAUSE 120
 690 LET F=1: GO TO 30
 800 FOR J=10 TO 12: PLOT 150,J:
 DRAW 80,0: NEXT J
```



```
 810 FOR J=212 TO 214: PLOT J,10
: DRAW 0,100: NEXT J: RETURN
 820 FOR J=201 TO 203: PLOT J,13
: DRAW 10,10: NEXT J
 830 FOR J=21 TO 23: PLOT 215,J:
 DRAW 10,-10: NEXT J: RETURN
 840 FOR J=111 TO 113: PLOT 211,
J: DRAW -45,0: NEXT J
 850 FOR J=201 TO 203: PLOT J,11
0: DRAW 10,-10: NEXT J: RETURN
 860 PLOT 171,110: DRAW 0,-20: R
ETURN
 880 CIRCLE 171,83.7: PLOT 169,8
5: PLOT 173,85
 890 PLOT 169,80: DRAW 4,0,.5*-P
I: RETURN
 900 FOR J=170 TO 172: PLOT J,75
: DRAW 0,-14: NEXT J: RETURN
 920 FOR J=70 TO 72: PLOT 169,J:
 DRAW -10,4: NEXT J: RETURN
 940 FOR J=70 TO 72: PLOT 173,J:
 DRAW 10,4: NEXT J: RETURN
 960 FOR J=59 TO 62: PLOT 171,J:
 DRAW -7,-10: NEXT J: RETURN
 980 FOR J=59 TO 62: PLOT 171,J:
 DRAW 7,-10: NEXT J: RETURN
1500 DATA "ISOTOPE","POKER","PLA
NET","SPHERE","FALLACY","TRIUMPH
","KANGAROO","ATTRIBUTE","DEADLI
NE","ROUTE"
1510 DATA "HOTEL","GALAXY","UNIV
ERSE","MIRAGE","BRIMSTONE","QUIV
ER","COLOUR","RAINBOW","LAUGH","
VISION"
1520 DATA "SLATE","USHERETTE","F
ELLOW","PRINTER","COMPUTER","FOL
DER","DIVISION","MAGAZINE","NUGE
T","JUMPER"
```

SPACE GAMES



Stellar Evade

This tiny program, written by Graham Charlton, produces a surprisingly entertaining game. You use the "z" and "m" keys to move left and right to dodge the asterisks coming up at you from below. (Note that the odd thing after the second equals sign in line 90 is an "m".)

```
  10 BORDER 2: PAPER 2: CLS
  20 LET a=10: LET b=15: LET c=2
0: LET t=0
  30 POKE 23692,0
  40 LET t=t+1
  50 PRINT AT 21,31;"  "
  60 FOR z=1 TO 2
  70 INK 7: PRINT AT c,RND*31;"*
"
  80 NEXT z
  90 LET b=b+(INKEY$="m" AND b<3
0)-(INKEY$="z" AND b>1)
 100 IF SCREEN$ (a,b)="*" THEN B
EEP 2,6: PRINT "You scored ";t:
FOR x=1 TO 200: NEXT x: RUN
 110 PRINT AT a,b;"@"
 120 GO TO 30
```

Space Trek

Skilled programmer David Perry wrote this program for the book. In this game, you fly through each galaxy, avoiding the stars, aiming to shoot all the enemy alien battle cruisers. Once you've finished, you must leave the galaxy via the Black Hole to penetrate into deeper space.

Full instructions are included within the program. You move around using the cursor keys, with "0" to fire. The pound signs should be entered as hash (#) symbols.

```
  1 REM
  2 REM    SPACE TREK!
  3 REM    ££££££££££
  4 REM
  5 REM     COPYRIGHT
  6 REM    DAVID PERRY
  7 REM        1983
  8 REM
  9 POKE 23658,8: LET NUM=10: D
IM N$(NUM+1,8): DIM N(NUM+1): FO
R N=1 TO NUM: LET N(N)=(1000-(10
0*n)): LET N$(N)="SPECTRUM": NEX
T N: GO SUB 2000: LET N$(1)="DAV
E OK!": LET N(1)=5000
 10 BORDER 1: PAPER 0: INK 7: B
RIGHT 1: CLS
 20 LET GAL=1: LET BON=3: LET S
C=0: LET LEV=3: LET MEN=3
```



```
 30 PRINT AT 0,0; INK 5; BRIGHT
 0;"
   ";AT 21,0;"
                "
 35 FOR N=1 TO 3: PRINT  INK 5:
AT N,0;" ";AT N,31;" ": NEXT N:
FOR N=4 TO 20: PRINT  INK 5;AT N
,0;" ";AT N,31;" ": NEXT N
 40 IF MEN=0 THEN  GO TO 1000
 42 FOR n=1 TO LEV
 45 LET X=INT (RND*14)+6: LET Y
=INT (RND*29)+1: IF SCREEN$ (X,Y
)<>" " THEN  GO TO 45
 50 PRINT AT X,Y; INK 2;"K"
 51 NEXT N
 54 FOR J=1 TO BON*2
 55 LET X=INT (RND*14)+6: LET Y
=INT (RND*30)+1: IF SCREEN$ (X,Y
)<>" " THEN  GO TO 55
 60 PRINT AT X,Y; INK 6;"*"
 70 NEXT J
 75 LET X=INT (RND*14)+6: LET Y
=INT (RND*29)+2: IF SCREEN$ (X,Y
)<>" " THEN  GO TO 75
 80 PRINT AT X,Y; INK 7;"O"
 90 PRINT AT 1,3; INK 4;"SHIPS
LEFT:";: FOR N=1 TO MEN: PRINT
INK 7;"a";: NEXT N: PRINT AT 1,1
9; INK 6;"SCORE:";SC
 100 PRINT AT 2,8; INK 3;"HIGHES
T SCORE:";N(1): PRINT AT 4,0; IN
K 5;"
    "
 105 PRINT AT 3,4; INK 2;"GALAXY
:";GAL: PRINT AT 3,14; INK 6;"KL
INGONS LEFT:";LEV
```

```
 110 LET D=2: LET A=5: LET B=1
 120 PRINT AT A,B;" "
 130 IF D=1 THEN  LET A=A-1
 140 IF D=2 THEN  LET B=B+1
 150 IF D=3 THEN  LET A=A+1
 160 IF D=4 THEN  LET B=B-1
 163 IF B<1 THEN  LET B=30
 164 IF B>30 THEN  LET B=1
 165 IF A<5 THEN  LET A=20
 166 IF A>20 THEN  LET A=5
 167 IF LEV=0 THEN  BEEP .01,50:
 IF SCREEN$ (A,B)="0" THEN  LET
BON=BON+2: LET GAL=GAL+1: PRINT
AT 3,11;GAL: GO SUB 800: LET LEV
=BON: GO TO 40
 170 IF SCREEN$ (A,B)<>" " THEN
 FOR I=7 TO 0 STEP -1: PRINT AT
A,B; INK I;"X": BEEP .02,I*7: NE
XT I: GO SUB 800: PRINT AT 1,13;
"    ": LET MEN=MEN-1: FOR N=1 T
O MEN: PRINT AT 1,13+N;"a": NEXT
 N: GO TO 40
 180 IF INKEY$="7" THEN  LET D=1
 190 IF INKEY$="8" THEN  LET D=2
 200 IF INKEY$="6" THEN  LET D=3
 210 IF INKEY$="5" THEN  LET D=4
 220 IF D=1 THEN  PRINT AT A,B;
INK 6;"a"
 230 IF D=2 THEN  PRINT AT A,B;
INK 6;"b"
 240 IF D=3 THEN  PRINT AT A,B;
INK 6;"c"
 250 IF D=4 THEN  PRINT AT A,B;
INK 6;"d"
 255 IF INKEY$="0" THEN  GO SUB
300
```



```
 260 BEEP .01,D*10: GO TO 120
 300 IF D=1 THEN  GO TO 400
 310 IF D=2 THEN  GO TO 500
 320 IF D=3 THEN  GO TO 600
 330 IF D=4 THEN  GO TO 700
 340 STOP
 400 IF A=5 THEN  RETURN
 405 FOR N=A-1 TO 5 STEP -1
 410 LET A$=SCREEN$ (N,B)
 420 IF A$="K" THEN  LET LEV=LEV
-1: PRINT AT 3,28;LEV;"  ": LET
SC=SC+100: PRINT AT 1,25;SC: FOR
 I=7 TO 0 STEP -1: PRINT AT N,B;
 INK I;"K": BEEP .01,I*7: NEXT I
: PRINT AT N,B;" ": NEXT N
 430 IF A$="*" OR A$="0" THEN  B
EEP .03,10: LET N=6: RETURN
 440 PRINT AT N,B;("f" AND N>5):
 BEEP .01,N
 450 PRINT AT N,B;(" " AND N>5):
 NEXT N: RETURN
 500 IF B=30 THEN  RETURN
 505 FOR N=B+1 TO 30
 510 LET A$=SCREEN$ (A,N)
 520 IF A$="K" THEN  LET LEV=LEV
-1: PRINT AT 3,28;LEV;"  ": LET
SC=SC+100: PRINT AT 1,25;SC: FOR
 I=7 TO 0 STEP -1: PRINT AT A,N;
 INK I;"K": BEEP .01,I*7: NEXT I
: PRINT AT A,N;" ": NEXT N
 530 IF A$="*" OR A$="0" THEN  B
EEP .03,10: LET N=30: RETURN
 540 PRINT AT A,N;("e" AND N<31)
: BEEP .01,N
 550 PRINT AT A,N;(" " AND N<31)
: NEXT N: RETURN
```

```
 600 IF A=20 THEN  RETURN
 605 FOR N=A+1 TO 20
 610 LET A$=SCREEN$ (N,B)
 620 IF A$="K" THEN  LET LEV=LEV
-1: PRINT AT 3,28;LEV;"  ": LET
SC=SC+100: PRINT AT 1,25;SC: FOR
 I=7 TO 0 STEP -1: PRINT AT N,B;
 INK I;"K": BEEP .01,I*7: NEXT I
: PRINT AT N,B;" ": NEXT N
 630 IF A$="*" OR A$="0" THEN  B
EEP .03,10: LET N=20: RETURN
 640 PRINT AT N,B;("f" AND N<21)
: BEEP .01,N
 650 PRINT AT N,B;(" " AND N<21)
: NEXT N: RETURN
 700 IF B=1 THEN  RETURN
 705 FOR N=B-1 TO 1 STEP -1
 710 LET A$=SCREEN$ (A,N)
 720 IF A$="K" THEN  LET LEV=LEV
-1: PRINT AT 3,28;LEV;"  ": LET
SC=SC+100: PRINT AT 1,25;SC: FOR
 I=7 TO 0 STEP -1: PRINT AT A,N;
 INK I;"K": BEEP .01,I*7: NEXT I
: PRINT AT A,N;" ": NEXT N
 730 IF A$="*" OR A$="0" THEN  B
EEP .03,10: LET N=1: RETURN
 740 PRINT AT A,N;("@" AND N>0):
 BEEP .01,N
 750 PRINT AT A,N;(" " AND N>0):
 NEXT N: RETURN
 801 LET B=2: FOR N=20 TO 5 STEP
 -1
 810 BEEP .02,N*2: PRINT AT N,0;
 INK 5;"
       "
 820 NEXT N
```



```
 830 PRINT AT 3,31; INK 5;" ": P
RINT AT 21,1; INK 5;"
                     ": RETURN
1000 PRINT AT 10,4; PAPER 7; INK
 0; BRIGHT 0;" YOU HAVE SCORED "
;SC;". "
1010 BEEP .1,24: BEEP .1,23: BEE
P .84,24: BEEP .5,21: BEEP .5,20
: BEEP .85,21
1500 IF SC<=N(10) THEN  PAUSE 50
: GO TO 1540
1502 LET NUM=11: IF SC>=N(NUM) T
HEN  INPUT "ENTER 8 INITALS! ";P
$: IF LEN P$>8 THEN  GO TO 1500
1505 IF SC>=N(NUM) THEN  LET N(N
UM)=SC: LET N$(NUM)=P$
1510 FOR A=1 TO (NUM-1): LET B$=
N$(A): LET C$=N$(A+1): LET B=N(A
): LET C=N(A+1): IF B<C THEN  LE
T N(A)=C: LET N(A+1)=B: LET N$(A
)=C$: LET N$(A+1)=B$
1520 NEXT A: FOR N=1 TO NUM-1: I
F N(N)<N(N+1) THEN  GO TO 1510
1530 NEXT N
1540 CLS

1550 PRINT AT 2,4;"H A L L  O F
 F A M E !"
1560 PRINT AT 3,4;"============
============"
1580 FOR N=1  TO NUM-1: PRINT AT
 N+5,7; INK 6;"("; INK 2;N; INK
6;")";AT N+5,12; INK 7;N(N): PRI
NT AT N+5,17; INK 5;N$(N): NEXT
N
```

```
1590 PLOT 0,0: DRAW 255,0: DRAW
0,175: DRAW -255,0: DRAW 0,-175:
 PRINT AT 17,8; PAPER 2; INK 7;"
 PRESS ANY KEY! ": LET I=0
1600 LET I=I+1: IF I>7 THEN  LET
 I=0
1610 PRINT AT 2,4; INK I;"H A L
L  O F  F A M E !"
1620 BEEP .01,I*7: PAUSE 2: IF I
NKEY$="" THEN  GO TO 1600
1630 GO TO 10
2000 BORDER 1: PAPER 0: INK 7: B
RIGHT 1: CLS
2030 PRINT AT 2,6;"S P A C E  T
R E K !"
2040 PRINT AT 3,6;"£££££££££££
£££££"
2050 PRINT AT 6,1;"CONTROL YOUR
SPACESHIP USING     THE CURSOR KE
YS AND 'O' TO       FIRE. THE OBJ
ECT OF THE GAME     IS TO PASS TH
ROUGH ONE GALAXY    AFTER ANOTHER
 WEAVING BETWEEN    THE STARS TO
SHOOT THE ENEMY"
2060 PRINT " KLINGON BATTLE CRUI
SERS.WHEN     THEY ARE ALL DESTRO
YED YOU       CAN LEAVE VIA THE B
LACK HOLE."
2065 RESTORE 2085: FOR C=0 TO 7:
 READ HAT: POKE USR "A"+C,HAT: N
EXT C
2070 PRINT : PRINT "     YOU 'a'
 BLACK HOLE '"; INK 7;"O'"
2080 PRINT : PRINT "      STAR '"
; INK 6;"*"; INK 7;"'  KLINGON '
"; INK 2;"K"; INK 7;"'"
```



```
2081 PRINT : PRINT TAB 2; PAPER
2; INK 7;" PRESS A KEY TO COMMEN
CE! "
2082 PLOT 0,0: DRAW 255,0: DRAW
0,175: DRAW -255,0: DRAW 0,-175
2084 FOR C=0 TO 39: READ HAT: PO
KE USR "B"+C,HAT: NEXT C
2085 DATA 24,36,36,36,60,90,102,
66
2086 DATA 0,224,94,49,49,94,224,
0
2087 DATA 66,102,90,60,36,36,36,
24
2088 DATA 0,7,122,140,140,122,7,
0
2089 DATA 0,0,0,255,255,0,0,0
2090 DATA 24,24,24,24,24,24,24,2
4
2110 LET I=0
2120 LET I=I+1: IF I>7 THEN  LET
 I=0
2130 PRINT AT 2,6; INK I;"S P A
C E  T R E K !"
2140 BEEP .01,50-(I*7): PAUSE 2:
 IF INKEY$="" THEN  GO TO 2120
2150 RETURN
9998 FOR N=0 TO 7: PRINT  INK N;
" ";: NEXT N
9999 GO TO 9998
```

Stellar Probe

This is a fascinating game in which, once again, you patrol a sector of the galaxy. As you might have guessed, there are aliens at the bottom of the galaxy, and it is your task to find them, and blast hell out of them.

The program shows a map of the current Galactic Sector, along with its important features. You'll quickly learn how to read the map once you run the game a few times.

At any time you can move, scan or fire. Your scanners operate in two ways. The short range scanner, which looks into the eight squares immediately surrounding you, consumes little energy. The long range scanners look two squares in a single direction, and use up more energy.

The game ends if you land on top of an alien ship. The aliens do not move around during a single game. You have limited reserves in your energy bank and must try and kill as many aliens as you can before your energy is totally depleted.

An alien (and the program invents names for the aliens for each game) can only shoot back after you have fired your laser at it (thus revealing your position) and if the alien is within a single square of

you. Damage to your ship from an accurate alien shot is shown in energy terms (that is, energy is drained from your bank). The game continues until you land on an alien ship, or run out of energy. Tim Rogers converted this program from an original program of mine.

```
  10 BORDER 0: RANDOMIZE
  20 INK 6
  30 PAPER 0
  40 CLS
  50 GO SUB 2000: GO SUB 1000
  70 GO SUB 3000
 110 GO TO 70
1010 DIM g(10,10)
1020 DIM s(10,10)
1050 FOR a=1 TO 20
1060 LET g(FN a(x),FN a(x))=1
1100 NEXT a
1110 LET g(u,p)=2
1120 LET s(u,p)=2
1125 LET rn=FN a(8)
1130 LET rn1=FN a(8)
1135 LET z$=FN a$()
1140 RETURN
1510 LET bl=FN a(20)
1520 BEEP .1,bl
1530 BEEP .1,bl-12
1540 RETURN
2005 DEF FN a(x)=INT (RND*x)+1
2010 DEF FN a$()=x$(rn, TO t(rn,
1))+y$(rn1, TO t(rn1,2))
2015 LET x=10
2020 LET u=5: LET p=5
2040 LET a$="
2100 DIM c$(5,9)
2110 FOR a=1 TO 5
2120 READ c$(a)
2130 NEXT a
2140 LET at=0
2150 DIM b(6): DIM c(6)
2160 FOR a=1 TO 8
2170 READ b(a): READ c(a)
2180 NEXT a
2190 LET e=1000+2000*RND
2200 DIM x$(10,8)
2210 DIM y$(10,8)
2215 DIM t(8,2)
```

```
2220 FOR a=1 TO 8
2225 READ o$: LET t(a,1)=LEN o$
2230 LET x$(a)=o$
2235 READ o$: LET t(a,2)=LEN o$
2240 LET y$(a)=o$
2250 NEXT a
2260 DIM s$(10,11)
2270 FOR a=1 TO 9
2280 LET rn=FN a(8)
2290 LET rn1=FN a(8)
2300 LET s$(a)=FN a$()
2310 NEXT a
2320 FOR a=1 TO 3
2330 FOR b=0 TO 7
2340 READ c
2350 POKE USR CHR$ (143+a)+b,c
2360 NEXT b: NEXT a
2490 RETURN
3010 CLS
3020 PRINT AT 0,8;"STAR ";
3030 GO SUB 1500
3050 PRINT "MAP ";
3060 GO SUB 1500
3080 PRINT "READ-OUT"
3090 GO SUB 1500
3100 FOR a=1 TO 32
3110 PRINT "*";
3130 NEXT a
3140 GO SUB 1500
3150 PRINT AT 3,0;" 1234567890"
3160 FOR a=1 TO 10
3170 IF a<10 THEN PRINT a;: GO T
O 3190
3180 PRINT "0";
3190 FOR b=1 TO 10
3195 LET b$=a$(s(a,b)+1)
3200 PRINT INVERSE 1;b$;
3205 IF b$<>" " THEN GO SUB 1500
3210 NEXT b
3215 IF a<10 THEN PRINT a: GO TO
 3230
3220 PRINT "0"
3230 NEXT a
3240 PRINT " 1234567890"
3510 PRINT AT 5,14; INK 2; PAPER
 7; FLASH 1;"Energy Banks ";INT
e
3520 PRINT AT 6,14;
3530 FOR a=1 TO e/200
3540 PRINT INK 5;"■";
3550 NEXT a
3560 GO SUB 1500
```



```
3570 IF e<1 THEN GO TO 4000
3580 IF at>0 THEN PRINT AT 8,14;
"alien ships";AT 9,14;"destroyed
 ";at: GO SUB 1500
3590 PRINT AT 10,14;"Ship is loc
ated"
3600 PRINT AT 11,14;"at sub-quad
rant"
3610 PRINT AT 12,14; FLASH 1;"("
;u;",";p;")"
3620 GO SUB 1500
3630 PRINT AT 13,14;"in ";s$(INT
 (u*p/10+.5));AT 14,14;"sector"
3640 PRINT AT 16,0;
3645 GO SUB 1500
3650 GO TO 4500
4010 GO SUB 5000
4020 PRINT FLASH 1;"Energy banks
 completely drained"
4030 PRINT "We killed ";at;" ali
ens"
4110 PRINT ,,c$(FN a(5));": Do y
ou want to be","captain again, s
ir?"
4120 IF INKEY$<>"" THEN GO TO 41
20
4130 IF INKEY$="" THEN GO TO 413
0
4140 IF INKEY$<>"n" AND INKEY$<>
"N" THEN RUN

4160 PRINT ,,c$(FN a(5));". You
always were"
4170 PRINT "a bit of a ";
4180 LET rn=FN a(8)
4190 LET rn1=FN a(8)
4200 PRINT FN a$()
4210 PRINT END
4500 GO SUB 5000
4510 PRINT AT 16,0;c$(FN a(5));"
:  Your order, Sir?"
4515 BEEP .2,RND*50
4520 PRINT AT 17,12;"1 Scan"
4530 GO SUB 1500
4540 PRINT TAB 12;"2 Move"
4550 GO SUB 1500
4560 PRINT TAB 12;"3 Fire"
4570 GO SUB 1500
4580 PRINT AT 21,0;"Press 1 2 or
 3 on your console"
4590 LET i$=INKEY$
```

```
4592 IF CODE i$<49 OR CODE i$>51
 THEN GO TO 4590
4595 GO SUB 1500
4600 GO SUB 5000
4610 IF i$="1" THEN GO TO 6000
4620 GO SUB 5000+1000*VAL i$
4630 FOR a=0 TO 26
4640 PRINT AT 21,a;"■■■■■■"
4650 BEEP .01,a+12
4660 BEEP .01,a
4670 PRINT AT 21,a;" "
4680 NEXT a
4690 PRINT AT 21,a;"      "
4700 RETURN
5005 PRINT AT 16,0;
5010 FOR a=1 TO 6
5020 PRINT "
                "
5030 NEXT a
5040 PRINT AT 16,0;
5050 RETURN
6005 PRINT TAB 11; FLASH 1; INK
7;"■■■■ SCANNER ■■■■"
6010 PRINT c$(FN a(5));"-"
6020 PRINT "(long 2) or short 1)
range, sir?"
6030 LET i$=INKEY$
6040 BEEP .01,0
6050 IF CODE i$<49 OR CODE i$>50
 THEN GO TO 6030
6060 LET k=VAL i$
6070 LET e=e-10*k
6080 IF k=2 THEN GO TO 6500
6090 PRINT AT 16,0;"  scanning,.
.."
6095 LET f=0
6100 FOR a=-(u>1) TO (u<10)
6110 FOR b=-(p>1) TO (p<10)
6115 BEEP .01,0
6120 IF g(u+a,p+b)=1 THEN LET f=
1
6130 NEXT b: NEXT a
6140 IF f=1 THEN PRINT z$;" ship
 in vicinity, sir.": GO TO 6160
6150 PRINT "Nothing in vicinity.
"
6155 GO SUB 1500
6160 PRINT AT 21,0;"Scanner off,
 sir?"
6165 BEEP .1,RND*50
6170 IF INKEY$="" THEN GO TO 617
0
```



```
6175 GO SUB 1500
6180 IF INKEY$="n" OR INKEY$="N"
 THEN GO SUB 5000: GO TO 6010
6190 GO TO 4500
6500 PRINT AT 18,0;c$(FN a(5));"
: Direction?                   "
6510 PRINT "north:1 east:2 south
:3 west:4"
6515 IF INKEY$<>"" THEN GO TO 65
15
6520 LET i$=INKEY$
6530 BEEP .01,0
6540 IF CODE i$<49 OR CODE i$>52
 THEN GO TO 6520
6550 LET k=(VAL i$-1)*2+1
6560 LET z=(g(u+b(k)*2,p+c(k)*2)
=1)
6570 IF z=1 THEN PRINT FLASH 1;"
positive": GO TO 6590
6580 PRINT "negative"
6590 GO SUB 1500
6595 IF INKEY$<>"" THEN GO TO 65
95
6600 GO TO 6160
7002 LET fl=0
7005 LET e=e-50
7010 PRINT "Bearings:"
7015 GO SUB 1500
7020 CIRCLE 40,24,16
7030 PLOT 40,24
7040 DRAW 0,16
7050 GO SUB 1500
7060 PRINT AT 16,16;"000"
7080 PLOT 40,24
7090 DRAW 16,0
7100 GO SUB 1500
7110 PRINT AT 18,18;"090"
7130 PLOT 40,24
7140 DRAW 0,-16
7150 GO SUB 1500
7160 PRINT AT 20,16;"180"
7180 PLOT 40,24
7190 DRAW -16,0
7200 GO SUB 1500
7210 PRINT AT 18,14;"270"
7230 INPUT "Degrees? ";r
7240 IF r>315 OR r<0 THEN GO TO
7230
7250 LET r=INT (r/45)+1
7255 IF fl=1 THEN RETURN
7260 LET g(u,p)=0
7270 LET s(u,p)=3
```

```
7280 LET u=u+b(r)
7290 LET p=p+c(r)
7300 IF g(u,p)<>1 THEN GO TO 111
0
7310 PRINT AT RND*21,0; FLASH 1;
 INK 2; PAPER 6;"You have landed
 on a ";z$;" ship!!!!!!!!!!!!!!!
!!!!!!!!!!!!"
7320 BEEP 1,3
7325 IF RND>.95 THEN GO TO 7310
7330 FOR a=1 TO RND*5
7340 PLOT 128,87
7350 DRAW 128-RND*255,87-RND*175
7360 NEXT a
7370 BEEP .3,-23
7380 GO TO 7320
8005 LET fl=1
8010 PRINT AT 14,12; INK 6; PAPE
R 2; FLASH 1;"RED ALERT"
8020 GO SUB 1500: GO SUB 1500
8030 PRINT AT 15,0;c$(FN a(5));"
 - Direction of fire:"
8040 GO SUB 7015
8045 LET e=e-100
8050 IF g(u+b(r),p+c(r))<>1 THEN
 GO TO 8300
8070 BEEP 1,20
8080 GO SUB 5000
8090 PRINT c$(FN a(5));" - You h
it the ";z$;" ship, sir!"
8095 LET s(u+b(r),p+c(r))=4
8097 LET g(u+b(r),p+c(r))=4
8100 LET at=at+1
8120 PRINT ,,c$(FN a(5));" - Sha
ll I turn","the red alert off, s
ir?"
8130 IF INKEY$<>"" THEN GO TO 81
40
8135 IF INKEY$="" THEN GO TO 813
5
8140 GO TO 1110
8300 GO SUB 5000
8310 PRINT c$(FN a(5));";";
8320 LET s(u+b(r),p+c(r))=3
8330 PRINT " We missed, sir."
8340 FOR a=0 TO 30
8350 BEEP .1,a
8360 NEXT a
8370 BEEP 1,35
8380 FOR a=1 TO 50
8390 BORDER RND*7
8400 NEXT a
```



```
8410 BORDER 0
8420 PRINT ,,c$(FN a(5));": The
";z$;" are","shooting back!!"
8430 GO SUB 1500
8450 IF RND>.6 THEN GO TO 8700
8460 PRINT "They HIT US SIR!!!!"
8470 LET e=e-100*RND
8480 GO TO 8110
8700 PRINT c$(FN a(5));": But th
ank the stars-"
8710 PRINT "The ";z$;" missed us
!"
8720 GO TO 8110
9020 DATA "Dr Sock","Spottie","L
t. Loohu","Checkout","Zulu"
9030 DATA -1,0,-1,1,0,1,1,1,1,0,
1,-1,0,-1,-1,-1
9040 DATA "Glob","ulian","Frax",
"tloid","Mesh","nik","Ro#","ulan
","Grup","tish","Krell","on","Qu
es","tar","Coch","rane"
9050 DATA 255,165,195,153,153,19
5,165,255
9060 DATA 255,189,219,255,255,21
9,189,255
9070 DATA 248,152,152,255,47,57,
249,223
```

Lunar Storm

David Perry's LUNAR STORM uses sophisticated string-handling to provide a mass of meteors and spaceships moving sideways on the screen.

Below this is the surface of the planet McClariana (named after the intrepid explorer O. J. McClaren who discovered the planet in 2068). You have to try and weave your way between these obstacles, and land on one of the three landing pads.

To simplify an extremely difficult task, you'll see you've been provided with a thrust button, to allow you to high tail it out of there when the going gets too rough. The game uses lots of sound and color, plus some rather clever UDG's.


```
    1 REM
    2 REM   LUNAR STORM
    3 REM    Copyright
    4 REM   DAVID PERRY
    5 REM
    6 REM
    7 REM
    8 BORDER 0: PAPER 0: INK 7: C
LS
    9 LET SC=0: LET BO=100: LET i
=7: GO TO 45
   10 CLS : RESTORE : GO SUB 29
   11 LET x=0: LET y=INT (RND*31)
+1
   12 PRINT AT x,y;" ": PRINT AT
0,0; INK 5;a$;AT 7,0;a$;AT 11,0;
a$;AT 15,0;a$;AT 18,0;a$
   13 PRINT AT 5,0; INK 5;b$;AT 9
,0;b$;AT 13,0;b$;AT 17,0;b$
   14 IF INT X>18 THEN GO TO 79
   15 IF SCREEN$ (x,y)<>" " THEN
GO TO 36: PRINT AT x,y;"@"
   16 BEEP .01,X*2
   17 PRINT AT x,y;" "
   18 LET x=x+.5: IF IN 61438=254
 THEN LET x=x-.5: LET sc=sc-1
   19 LET Y=Y+(INKEY$="8")-(INKEY
$="5")
   20 IF y<0 THEN LET y=31
   21 IF y>31 THEN LET y=0
   22 IF X>18 THEN GO TO 79
   23 IF SCREEN$ (x,y)<>" " THEN
GO TO 36
   24 LET SC=SC+1
   25 PRINT AT x,y; INK 7; BRIGHT
 0;"@"
   26 LET a$=a$(2 TO )+a$(1)
   27 LET b$=b$(31)+b$( TO 30)
   28 GO TO 12
   29 PRINT AT 20,0; INK 2;"[redacted]
[redacted]"
   30 PRINT AT 19,0; INK 3;"[redacted]";
 FLASH 1;"[redacted]"; FLASH 0; INK 7;"[redacted]
"; INK 3; FLASH 1;"[redacted]"; FLASH 0;"[redacted]
[redacted]"; FLASH 1;"[redacted]"; INK 7; F
LASH 0;"[redacted]"; INK 3; FLASH 1;"[redacted]";
FLASH 0;"[redacted]"; FLASH 1;"[redacted]"
; FLASH 0; INK 7;"[redacted]"; INK 3; FLA
SH 1;"[redacted]"; FLASH 0;"[redacted]"
   31 FOR n=1 TO 3: READ c$: FOR
a=0 TO 7: READ b: POKE USR c$+a,
b: NEXT a: NEXT n
```

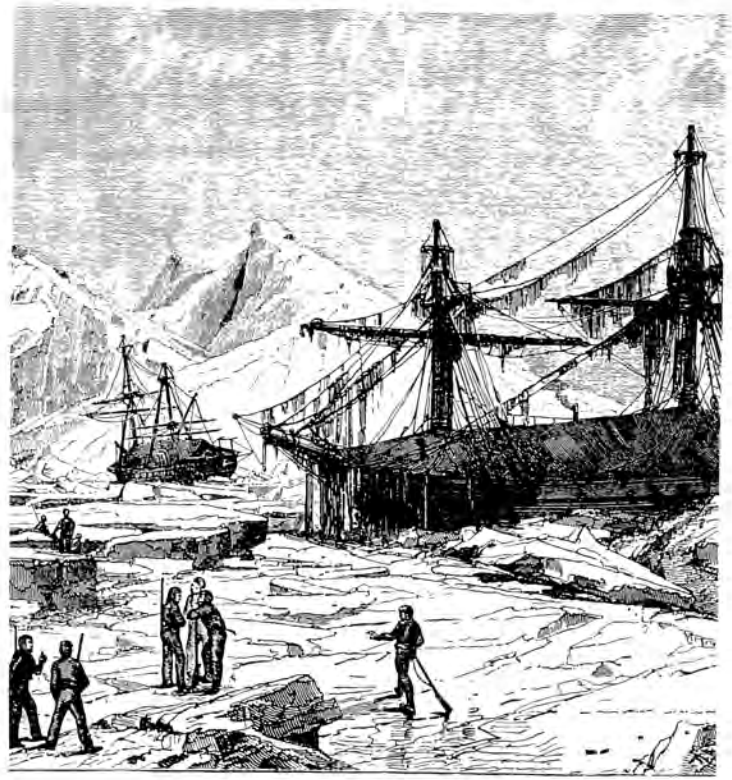
```
  32 DATA "c",24,36,90,165,219,2
19,189,195
  33 DATA "a",0,60,126,171,213,1
26,60,0
  34 DATA "b",195,189,90,126,126
,90,189,195
  35 RETURN
  36 FOR n=7 TO 0 STEP -1
  37 PRINT AT x,y; INK n;"▣"
  38 BEEP .01,n*3: BEEP .01,50-n
*3
  39 NEXT n
  40 LET I=7
  41 PRINT AT 2,8; INK 5;"S C O
R E :";SC
  42 PRINT INK I; BRIGHT 8;AT 0,
1;"P R E S S  A N Y  K E Y ! ! !
"
  43 LET I=I-1: IF I<0 THEN LET
I=7
  44 IF INKEY$="" THEN GO TO 42
  45 LET SC=0: CLS : LET d=0: BR
IGHT 1: INK 6
  46 PRINT "
  47 PRINT "
  48 PRINT "
  49 PRINT "
  50 PRINT "
  51 PRINT "
  52 PRINT ; INK 5: BRIGHT 0
  53 PRINT "
  54 PRINT "
  55 PRINT "
  56 PRINT "
  57 PRINT "
  58 PRINT "
  59 PRINT "
  60 PRINT AT 15,5; INK i;"ENTER
 LEVEL(1 TO 3)"
```



```
  61 PRINT AT 16,5; INK 1/2;"===
==============="
  62 PRINT AT 18,0; INK 4; INK T
;"5-SHIP LEFT  8-SHIP RIGHT"
  63 PRINT AT 19,11; INK 4;"0-TH
RUST"
  64 PRINT AT 20,0; INK 6;"TRY T
O LAND SHIP ON LANDING PADS"
  65 PRINT #1; BRIGHT 8; INK 7;"
 LUNAR STORM © DAVID PERRY 1983"
  66 PRINT AT 15,6; INK i;"ENTER
 LEVEL(1 TO 3)"

  67 LET a$=INKEY$: IF a$<"1" OR
 a$>"3" THEN GO TO 75
  68 IF a$="" THEN GO TO 75
  69 IF a$="1" THEN LET a$="
@
ET d=1
  70 IF d=1 THEN LET b$="
  71 IF a$="2" THEN LET a$="
@
ET d=2
  72 IF d=2 THEN LET b$="
@
  73 IF a$="3" THEN LET a$="
  74 IF d=3 THEN LET b$="
  75 GO TO 10
  76 LET i=i-1: IF i<0 THEN LET
i=7
  77 GO TO 65
  78 STOP
  79 PRINT AT x,y; INK 7; BRIGHT
 8;"@": IF Y=4 THEN LET BO=150:
GO TO 83
  80 IF Y=16 THEN LET BO=100: GO
 TO 83
  81 IF Y=28 THEN LET BO=150: GO
 TO 83
  82 GO TO 36
  83 LET SC=SC+BO: FOR N=1 TO 3:
 FOR I=0 TO 7
  84 PRINT AT 1,0; INK I;"C O N
G R A T U L A T I O N S !"
  85 PRINT AT 3,7;"BONUS POINTS
:";BO;"."
```

```
   86 PRINT AT 5,11;"SCORE :";SC
   87 NEXT I: NEXT N
   88 FOR N=-20 TO 50 STEP 5: BEE
P .005,N: BEEP .001,50-ABS (N):
BEEP .006,N/2: NEXT N
   89 CLS : GO TO 10
```


THE LURE OF THE MAZE

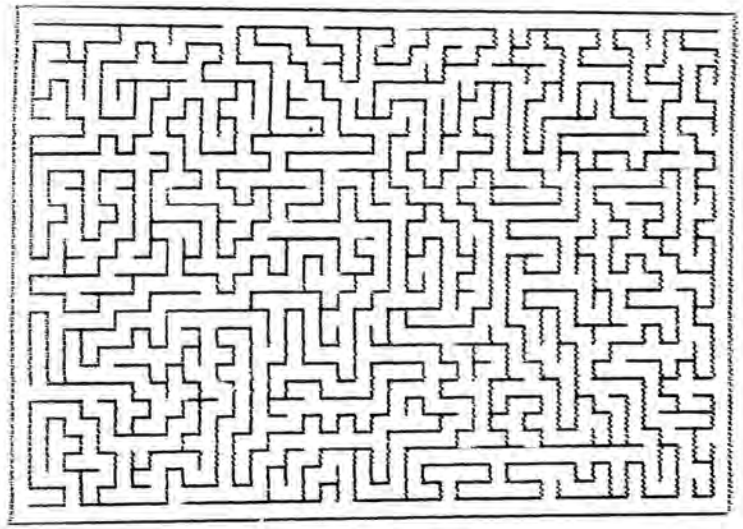


ABSOLUTELY AMAZING

We now look at a number of different, outstanding maze programs, most of which were written by Graham Charlton.

Maze-maker

We start with a program which draws a maze of any size, such as this one:



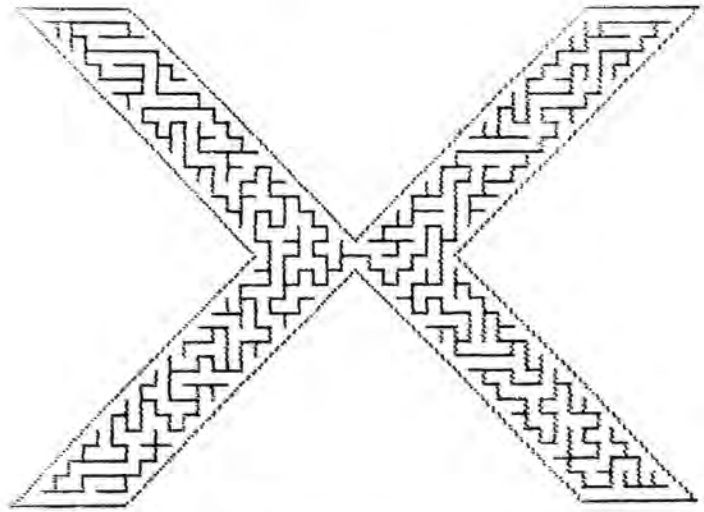
When prompted to do so, enter any number between two and 23 to see the program in action.

```
10 DIM a$(9999)
20 DIM b$(9999)
30 INPUT s
40 LET t=INT (255/s)*s
```

```
  45 LET w=INT (175/s)*s
  50 PLOT 0,0
  60 DRAW t,0
  70 DRAW 0,w
  80 DRAW -t,0
  90 DRAW 0,-w
 100 LET x=s
 110 LET y=s
 120 LET z=1
 130 PLOT x,y
 140 IF (POINT (x+s,y)+POINT (x,
y-s)+POINT (x,y+s)+POINT (x-s,y)
)<>4 THEN GO TO 200
 150 LET z=z-1
 160 IF z=0 THEN STOP
 170 LET x=CODE a$(z)
 180 LET y=CODE b$(z)
 190 GO TO 130
 200 LET a$(z)=CHR$ x
 210 LET b$(z)=CHR$ y
 220 LET z=z+1
 230 LET r=INT (RND*4)
 240 LET c=s*((r=0)-(r=1))
 250 LET d=s*((r=2)-(r=3))
 260 IF POINT (x+c,y+d) THEN GO
TO 230
 270 DRAW c,d
 280 LET x=x+c
 290 LET y=y+d
 300 GO TO 130
```

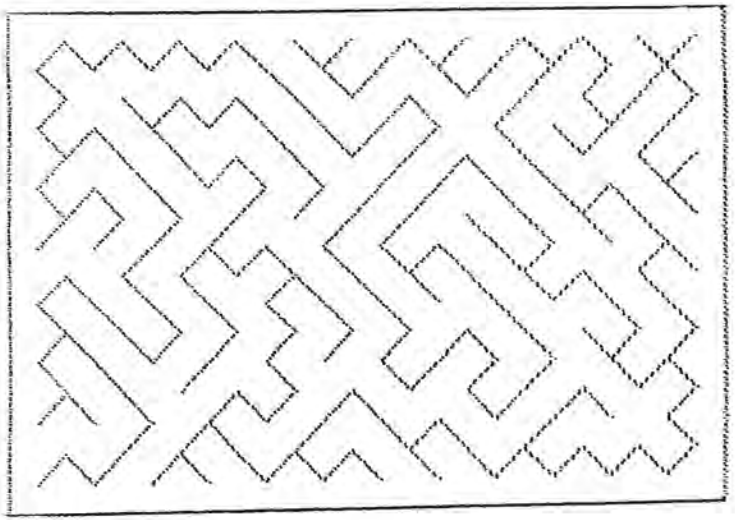

Mangled Mazes

There is no reason why the walls of the maze should be at right angles. Here are four MANGLED MAZES programs, which use walls at odd angles:



```
 10 DIM a$(9999)
 20 DIM b$(9999)
 30 INPUT s
 40 LET t=INT (254/s)*s
 50 LET w=INT (174/s)*s
 60 PLOT 0,0
 70 DRAW w/2,w/2
 72 DRAW -w/2,w/2
 74 DRAW (t-w)/2,0
 76 DRAW w/2-s,-w/2+s
 78 DRAW w/2-s,w/2-s
 80 DRAW (t-w)/2,0
 81 DRAW -w/2,-w/2
 82 DRAW w/2,-w/2
 84 DRAW -(t-w)/2,0
 86 DRAW -w/2+s,w/2-s
 88 DRAW -w/2+s,-w/2+s
```

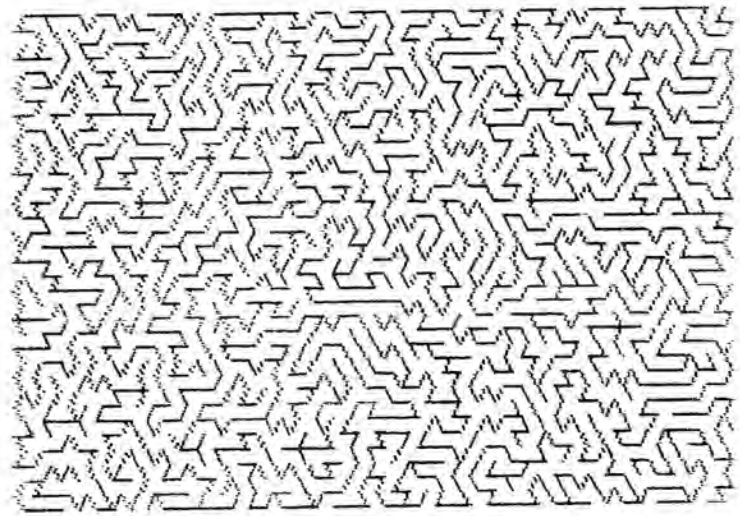
```
  90 DRAW -(t-w)/2,0
 110 LET x=s*12
 120 LET y=s*10
 130 LET z=1
 140 PLOT x,y
 150 IF (POINT (x+s,y)+POINT (x-
s,y)+POINT (x,y+s)+POINT (x,y-s)
```



```
  10 DIM a$(9999)
  20 DIM b$(9999)
  30 INPUT s
  40 LET t=INT (255/s)*s
  45 LET w=INT (175/s)*s
  50 PLOT 0,0
  60 DRAW t,0
  70 DRAW 0,w
  80 DRAW -t,0
  90 DRAW 0,-w
 100 LET x=s
 110 LET y=s
 120 LET z=1
 130 PLOT x,y
 140 IF (POINT (x+s,y+s)+POINT (x
+s,y-s)+POINT (x-s,y+s)+POINT (x
-s,y-s))<>4 THEN GO TO 200
 150 LET z=z-1
```

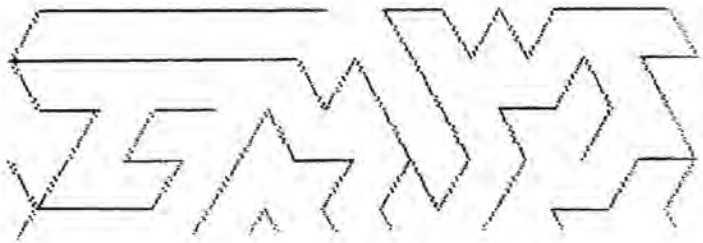


```
 160 IF z=0 THEN STOP
 170 LET x=CODE a$(z)
 180 LET y=CODE b$(z)
 190 GO TO 130
 200 LET a$(z)=CHR$ x
 210 LET b$(z)=CHR$ y
 220 LET z=z+1
 230 LET r=INT (RND*4)
 240 LET c=s*((r<2)-(r>1))
 250 LET d=s*((r=0)+(r=3)-(r=1)-
(r=2))
 260 IF POINT (x+c,y+d) THEN GO
TO 230
 270 DRAW c,d
 280 LET x=x+c
 290 LET y=y+d
 300 GO TO 130
```



```
  10 DIM a$(3000)
  20 DIM b$(3000)
  30 LET x=128
  40 LET y=88
  50 LET z=1
  60 INPUT s
  70 LET t=s*2
```

```
  60 LET q=INT (SQR (t*t-s*s))
 100 PLOT x,y
 200 FOR c=-t TO t STEP t*2
 210 IF x+c>255 OR x+c<0 THEN GO
TO 230
 220 IF POINT (x+c,y)=0 THEN GO
TO 3000
 230 NEXT c
 250 FOR c=-s TO s STEP t
 260 IF x+c>255 OR x+c<0 THEN GO
TO 310
 270 FOR d=-q TO q STEP q*2
 280 IF y+d>175 OR y+d<0 THEN GO
TO 300
 290 IF POINT (x+c,y+d)=0 THEN G
O TO 3000
 300 NEXT d
 310 NEXT c
2000 LET z=z-1
2010 IF z=0 THEN STOP
2020 LET x=CODE a$(z)
2030 LET y=CODE b$(z)
2040 GO TO 100
3000 LET a$(z)=CHR$ x
3010 LET b$(z)=CHR$ y
3020 LET z=z+1
3030 LET r=INT (RND*6)
3040 LET c=s*((r=1)+(r=5)-(r=0)-
(r=4))+t*((r=3)-(r=2))
3050 LET d=q*((r<2)-(r>3))
3060 IF x+c>255 OR x+c<0 THEN GO
TO 3030
3070 IF y+d>175 OR y+d<0 THEN GO
TO 3030
3080 IF POINT (x+c,y+d) THEN GO
TO 3030
3090 DRAW c,d
3100 LET x=x+c
3110 LET y=y+d
3120 GO TO 100
```



Walls of Suspense

In this program, you have to race two computer opponents (both shown as "O") in trying to solve a maze. You are the "*". You start at the top left hand corner and are trying to get to the bottom right hand one.

The first one to solve the maze is the winner. Delete line ~~3110~~ 3120 to make it a real time game.

```
  10 DIM c$(23,33)
  12 DIM a$(1000)
  14 DIM b$(1000)
  20 LET x=13
  30 LET y=17
  40 LET z=1
  50 FOR w=2 TO 22
  60 LET c$(w)=" ################
################ "
  70 NEXT w
 100 LET c$(x,y)=" "
 110 IF c$(x+2,y)="#" OR c$(x-2,
y)="#" OR c$(x,y+2)="#" OR c$(x,
y-2)="#" THEN GO TO 1000
 120 LET z=z-1
 130 IF z=0 THEN GO TO 2000
 140 LET x=CODE a$(z)
 150 LET y=CODE b$(z)
 160 GO TO 100
1000 LET a$(z)=CHR$ x
1010 LET b$(z)=CHR$ y
1020 LET z=z+1
1030 LET r=INT (RND*4)
1040 LET c=(r=0)-(r=1)
1050 LET d=(r=2)-(r=3)
1060 IF c$(x+2*c,y+2*d)=" " THEN
GO TO 1030
1070 LET c$(x+c,y+d)=" "
1080 LET c$(x+2*c,y+2*d)=" "
1090 LET x=x+2*c
1100 LET y=y+2*d
1110 GO TO 100
2000 FOR w=2 TO 22
```

```
2010 PRINT INK 7;c$(w,2 TO )
2020 NEXT w
2030 PRINT AT 19,29;"X"
3000 LET x=1
3005 LET p=1
3006 LET u=1
3010 LET y=1
3015 LET q=1
3016 LET v=1
3020 LET s=0
3025 LET t=0
3100 PRINT AT x,y;"#"
3110 LET z$=INKEY$
3120 IF z$="" THEN GO TO 3110
3130 LET c=(z$="6")-(z$="7")
3140 LET d=(z$="8")-(z$="5")
3160 IF SCREEN$ (x+c,y+d)="#" TH
EN BEEP .1,0: GO TO 3600
3170 PRINT AT x,y;"■"
3180 LET x=x+c
3190 LET y=y+d
3200 PRINT AT x,y;"#"
3210 IF x=19 AND y=29 THEN STOP
3600 LET s=s-1
3700 IF s=-1 THEN LET s=3
3900 PRINT AT p,q;"■"
4000 IF s=4 THEN LET s=0
4010 LET c=(s=1)-(s=3)
4020 LET d=(s=0)-(s=2)
4030 IF SCREEN$ (p+c,q+d)="#" TH
EN LET s=s+2: GO TO 4065
4050 LET p=p+c
4060 LET q=q+d
4065 PRINT AT p,q;"0"
4070 IF p=19 AND q=29 THEN STOP
5600 LET t=t-1
5700 IF t=-1 THEN LET t=3
5900 PRINT AT u,v;"■"
6000 IF t=4 THEN LET t=0
6010 LET c=(t=0)-(t=2)
6020 LET d=(t=1)-(t=3)
6030 IF SCREEN$ (u+c,v+d)="#" TH
EN LET t=t+2: GO TO 6065
6050 LET u=u+c
6060 LET v=v+d
6065 PRINT AT u,v;"0"
6070 IF u=19 AND v=29 THEN STOP
6080 GO TO 3100
```


Wall of Suspense

In this version of the game, you are only racing a single computer opponent. However, it is far more intelligent than the double opponents and can race as fast as you can.

```
  10 DIM c$(23,33)
  12 DIM a$(1000)
  14 DIM b$(1000)
  20 LET x=13
  30 LET y=17
  40 LET z=1
  50 FOR w=2 TO 22
  60 LET c$(w)="  ##################
##################  "
  70 NEXT w
 100 LET c$(x,y)="  "
 110 IF c$(x+2,y)="#" OR c$(x-2,
y)="#" OR c$(x,y+2)="#" OR c$(x,
y-2)="#" THEN GO TO 1000
 120 LET z=z-1
 130 IF z=0 THEN GO TO 2000
 140 LET x=CODE a$(z)
 150 LET y=CODE b$(z)
 160 GO TO 100
1000 LET a$(z)=CHR$ x
1010 LET b$(z)=CHR$ y
1020 LET z=z+1
1030 LET r=INT (RND*4)
1040 LET c=(r=0)-(r=1)
1050 LET d=(r=2)-(r=3)
1060 IF c$(x+2*c,y+2*d)="  " THEN
 GO TO 1030
1070 LET c$(x+c,y+d)="  "
1080 LET c$(x+2*c,y+2*d)="  "
1090 LET x=x+2*c
1100 LET y=y+2*d
1110 GO TO 100
2000 FOR w=2 TO 22
2010 PRINT INK 7;c$(w,2 TO )
2020 NEXT w
2030 PRINT AT 19,29;"X"
3000 LET x=1
3005 LET p=1
3010 LET y=1
3015 LET q=1
```

```
3020 LET s=0
3100 PRINT AT x,y;"*"
3110 LET z$=INKEY$
3120 IF z$="" THEN GO TO 3110
3130 LET c=(z$="6")-(z$="7")
3140 LET d=(z$="8")-(z$="5")
3160 IF SCREEN$ (x+c,y+d)="#" TH
EN BEEP .1,0: GO TO 3600
3170 PRINT AT x,y;" "
3180 LET x=x+c
3190 LET y=y+d
3200 PRINT AT x,y;"*"
3210 IF x=19 AND y=29 THEN STOP
3600 LET s=s-1
3700 IF s=-1 THEN LET s=3
3900 PRINT AT p,q;" "
4000 IF s=4 THEN LET s=0
4010 LET c=(s=1)-(s=3)
4020 LET d=(s=0)-(s=2)
4030 IF SCREEN$ (p+c,q+d)="#" TH
EN LET s=s+1: GO TO 4000
4050 LET p=p+c
4060 LET q=q+d
4065 PRINT AT p,q;"O"
4070 IF p=19 AND q=29 THEN STOP
4080 GO TO 3100
```

Dual Level 3-D Maze

In this program, the maze is on two levels connected by a number of tunnels.

The display shows the top and bottom floors, along with the interconnecting tunnels. You move through the maze using the cursor keys "5" to "8", with "9" to go up a level and "4" to descend. The maze is invisible, but you can delete line 2000 if you want to see it.

You have to get from the top left hand corner of the left hand maze to the top right of the right hand one.


```
##########   ##########   ##########
#*#    # #   # ### # #   #    #    #
# # ### # #   ########## ##########
#     # # #   ######## #  #        #
########## #  ########## # ######  #
#          #  # ######## # #       #
####### #  #  ########## ### ######
#     #    #  # ### ### #  # #     #
### #####  #  ########## #  #####  #
# #   #    #  ### ###### #  #   #  #
# ### # #  #  ########## #  ### #  #
# #   # # #  # # ###### #  #   #  #
# ##### # #  ########## ######## #
# #   # # #  ### # ###  #    #    #
# ##### # #  ########## #  # ###  #
#     #   #  ### # ###  #  # # #  #
########## #  ########## #  ###  ###
# #        #  ### ### #  #  #    # #
# ######## #  ########## #  #####  #
#          #  ##### ###  #        #X#
##########   ##########   ##########
```

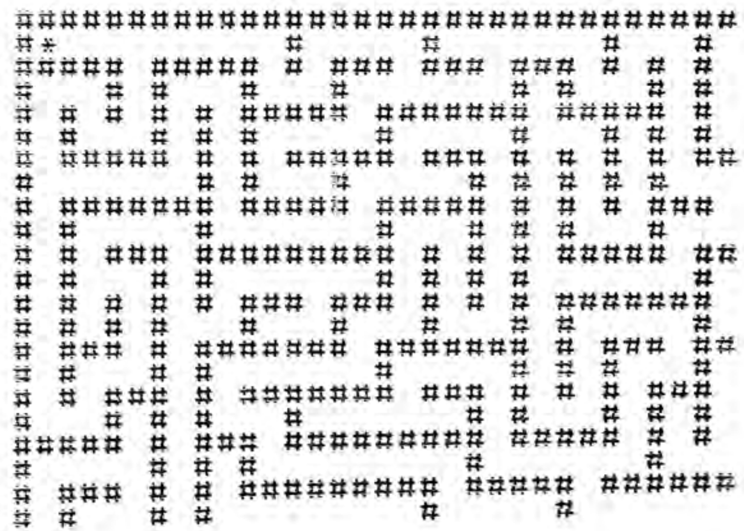
```
  10 DIM d$(7,23,11)
  20 DIM a$(1000)
  30 DIM b$(1000)
  40 DIM c$(1000)
  50 LET x=5
  60 LET y=21
  70 LET z=9
  80 LET w=1
  90 FOR a=2 TO 6
 100 FOR b=2 TO 22
 110 LET d$(a,b)="  ##########  "
 120 NEXT b
 130 NEXT a
 140 LET d$(x,y,z)=" "
 150 IF d$(x+2,y,z)="#" OR d$(x-
2,y,z)="#" OR d$(x,y+2,z)="#" OR
 d$(x,y-2,z)="#" OR d$(x,y,z+2)=
"#" OR d$(x,y,z-2)="#" THEN GO T
O 1000
 160 LET w=w-1
 170 IF w=0 THEN GO TO 2000
 180 LET x=CODE a$(w)
 190 LET y=CODE b$(w)
 200 LET z=CODE c$(w)
 210 GO TO 140
1000 LET a$(w)=CHR$ x
1010 LET b$(w)=CHR$ y
```

```
1020 LET c$(w)=CHR$ z
1030 LET w=w+1
1040 LET r=INT (RND*6)
1050 LET c=(r=0)-(r=1)
1060 LET d=(r=2)-(r=3)
1070 LET e=(r=4)-(r=5)
1080 IF d$(x+2*c,y+2*d,z+2*e)="
" THEN GO TO 1040
1090 LET d$(x+c,y+d,z+e)=" "
1100 LET d$(x+2*c,y+2*d,z+2*e)="
"
1110 LET x=x+2*c
1120 LET y=y+2*d
1130 LET z=z+2*e
1140 GO TO 140
2000 BORDER 0: INK 0: PAPER 0: C
LS
2010 FOR b=2 TO 23
2020 PRINT d$(3,b);
2030 PRINT d$(4,b);
2040 PRINT d$(5,b, TO 10)
2050 NEXT b
2060 INK 7
2100 LET x=1
2110 LET y=2
2120 PRINT AT 19,30;"X"
3000 PRINT FLASH 1;AT x,y;"*"
3010 IF x=19 AND y=30 THEN STOP
3020 LET z$=INKEY$
3030 IF z$="" THEN GO TO 3020
3040 LET c=(z$="6")-(z$="7")
3050 LET d=(z$="8")-(z$="5")+11*
((z$="9")-(z$="4"))
3055 IF y+d<0 THEN GO TO 3020
3060 IF SCREEN$ (x+c,y+d)="#" TH
EN GO TO 3020
3070 PRINT AT x,y;"■"
3080 LET x=x+c
3090 LET y=y+d
3100 GO TO 3000
```

Scrolling Maze

In the next program in this section of the book, the maze is four times bigger than the screen, which scrolls across when you hit the sides. There are two versions given. The first scrolls using a machine code routine, and the second version does much the same thing in BASIC.

The object of the game is to move from the top left hand corner of the maze to the bottom right hand one using the cursor keys. Delete lines 3070 and 3075 if you don't want to leave a trail.



```
 2 CLEAR 59999
 4 FOR a=60000 TO 60183
 6 READ n
 8 POKE a,n
10 NEXT a
15 DIM c$(45,65)
20 DIM a$(1000)
25 RANDOMIZE
30 DIM b$(1000)
40 LET x=43
```

```
   50 LET y=63
   60 LET z=1
   70 FOR w=2 TO 44
   80 LET c$(w)="  ##################
##########################################
##################  "
   90 NEXT w
  100 LET c$(x,y)="  "
  110 IF c$(x+2,y)="#" OR c$(x-2,
y)="#" OR c$(x,y+2)="#" OR c$(x,
y-2)="#" THEN GO TO 1000
  120 LET z=z-1
  130 IF z=0 THEN GO TO 2000
  140 LET x=CODE a$(z)
  150 LET y=CODE b$(z)
  160 GO TO 100
1000 LET a$(z)=CHR$ x
1010 LET b$(z)=CHR$ y
1020 LET z=z+1
1030 LET r=INT (RND*4)
1040 LET c=(r=0)-(r=1)
1050 LET d=(r=2)-(r=3)
1060 IF c$(x+2*c,y+2*d)="  " THEN
 GO TO 1030
1070 LET c$(x+c,y+d)="  "
1080 LET c$(x+2*c,y+2*d)="  "
1090 LET x=x+2*c
1100 LET y=y+2*d
1110 GO TO 100
2000 FOR w=2 TO 23
2010 PRINT c$(w,2 TO 33)
2020 NEXT w
2100 LET x=1
2110 LET y=1
2120 LET p=2
2130 LET q=2
3000 PRINT FLASH 1;AT x,y;"*"
3010 IF p=23 AND q=33 THEN PRINT
 AT 20,30;"X"
3020 LET z$=INKEY$
3030 IF z$="" THEN GO TO 3020
3040 LET c=(z$="6")-(z$="7")
3050 LET d=(z$="8")-(z$="5")
3060 IF SCREEN$ (x+c,y+d)="#" TH
EN GO TO 3020
3070 PRINT AT x,y;"■"
3080 LET x=x+c
3090 LET y=y+d
3120 IF y=0 THEN GO SUB 5000
3130 IF y=31 THEN GO SUB 5100
3140 IF x=0 THEN GO SUB 5200
```



```
3150 IF x=21 THEN GO SUB 5300
3160 PRINT FLASH 1;AT x,y;"*"
3170 IF x=20 AND y=30 AND p=23 A
ND q=33 THEN STOP
3180 GO TO 3010
50000 RANDOMIZE USR 60021
50010 LET y=y+1
50015 LET q=q-1
50020 FOR w=0 TO 21
50030 PRINT AT w,0;c$(p+w,q)
50040 NEXT w
50090 RETURN
51000 RANDOMIZE USR 60000
51110 LET y=y-1
51115 LET q=q+1
51120 FOR w=0 TO 21
51130 PRINT AT w,31;c$(p+w,q+31)
51140 NEXT w
51190 RETURN
52000 RANDOMIZE USR 60111
52010 LET x=x+1
52015 LET p=p-1
52020 PRINT AT 0,0;c$(p,q TO q+51
)
52090 RETURN
53000 RANDOMIZE USR 60043
53010 LET x=x-1
53015 LET p=p+1
53020 PRINT AT 21,0;c$(p+21,q TO
q+31)
53090 RETURN
90000 DATA 33,0,64,85,62,192,6,31
90010 DATA 35,94,43,115,35,16,249
90020 DATA 114,35,61,32,242,201
91000 DATA 33,255,87,22,0,62,192
91110 DATA 6,31,43,94,35,115,43
91120 DATA 16,249,114,43,61,32
91130 DATA 242,201
92000 DATA 33,0,64,17,32,64,229
92010 DATA 213,14,23,6,32,26,119
92020 DATA 121,230,7,254,1,32,2
92030 DATA 151,18,35,19,16,241,13
92040 DATA 40,19,121,230,7,254,0
92050 DATA 40,22,254,7,32,225,213
92060 DATA 17,0,7,25,209,24,217
92070 DATA 209,225,20,36,124,254
92080 DATA 72,32,204,201,229,33,0
92090 DATA 7,25,235,225,24,198
93000 DATA 33,255,87,17,223,87,22
9
93010 DATA 213,14,23,6,32,26,119
```

```
9320 DATA 121,230,7,254,1,32,2
9330 DATA 151,16,43,27,16,241,13
9340 DATA 40,21,121,230,7,254,0
9350 DATA 40,24,254,7,32,225,213
9360 DATA 17,0,7,167,237,82,209
9370 DATA 24,215,209,225,21,37
9380 DATA 124,254,79,200,24,201
9390 DATA 229,33,0,7,235,167,237
9400 DATA 82,235,225,24,193
```

This is the BASIC version:

```
  10 DIM c$(45,65)
  20 DIM a$(1000): DIM b$(1000)
  40 LET x=43: LET y=63: LET z=1
  70 FOR w=2 TO 44
  80 LET c$(w)="  ##################
##############################
################  ": NEXT w
 100 LET c$(x,y)=" "
 110 IF c$(x+2,y)="#" OR c$(x-2,
y)="#" OR c$(x,y+2)="#" OR c$(x,
y-2)="#" THEN GO TO 1000
 130 LET z=z-1: IF z=0 THEN GO T
O 2000
 140 LET x=CODE a$(z)
 150 LET y=CODE b$(z): GO TO 100
1000 LET a$(z)=CHR$ x
1010 LET b$(z)=CHR$ y: LET z=z+1
1030 LET r=INT (RND*4)
1040 LET c=(r=0)-(r=1)
1050 LET d=(r=2)-(r=3)
1060 IF c$(x+2*c,y+2*d)=" " THEN
 GO TO 1030
1070 LET c$(x+c,y+d)=" "
1080 LET c$(x+2*c,y+2*d)=" "
1090 LET x=x+2*c: LET y=y+2*d
1110 GO TO 100
2000 LET x=1: LET y=1: LET p=2
2010 LET q=2: LET c$(43,63)="X"
2090 PRINT AT 0,0;
2100 FOR w=p TO p+21
2110 PRINT c$(w,q TO q+31)
2120 NEXT w
3000 PRINT FLASH 1;AT x,y;"+"
3010 IF x=20 AND y=30 AND p=23 A
ND q=33 THEN STOP
3020 LET z$=INKEY$
3030 IF z$="" THEN GO TO 3020
```



```
3040 LET c=(z$="6")-(z$="7")
3050 LET d=(z$="8")-(z$="5")
3060 IF SCREEN$ (x+c,y+d)="#" TH
EN GO TO 3020
3070 PRINT AT x,y;"■"
3075 LET c$(x+p,y+q)="■"
3080 LET x=x+c: LET y=y+d
3120 IF y=0 THEN LET y=y+1: LET
q=q-1: GO TO 2090
3130 IF y=31 THEN LET y=y-1: LET
 q=q+1: GO TO 2090
3140 IF x=0 THEN LET x=x+1: LET
p=p-1: GO TO 2090
3150 IF x=21 THEN LET x=x-1: LET
 p=p+1: GO TO 2090
3180 GO TO 3000
```

Rollermaze

This maze program was written by Neil Pellinacci. It was written for the 16K computer. It will run on the 48K machine but may be difficult to modify because the program's machine code will have to be altered.

The aim of the game is to travel across a randomly-generated maze, first from top to bottom and then back to the top again in the shortest possible time. To enliven the game, bits of the maze roll left and the rest rolls right. You, however, don't roll anywhere.

You have three lives and you lose one every time you collide with a chunk of maze to your left or to your right. A siren noise sounds when this happens. The number of lives, together with the time taken and the best time so far, is displayed at the top of the screen.

The title page sequence begins when you start the program. This tells you which keys to press and also the top ten times and the names of the players which achieved them.

Presssing "K" at any time in the sequence will allow you to change the control keys. Just follow the instructions. "S" will leave the title pages and start a game.

ROLLERMAZE has five levels of difficulty. I suggest you start with level one, then look at level five in action to see the difference. The clock starts as soon as the maze appears. If your final time is in the top ten, you'll be asked for your name, which you will then see in the best scores list.

To put the game into your machine, carry out the following, and remembering to VERIFY after SAVE:

1/ Type in program one, the machine-code loader program, and run it. You should get the message '9 STOP statement' if the program is correct. Now save this program on tape as you may need it later.

PROGRAM 1...Machine Code

```
  5 REM byte poker
 10 POKE 23609,100
 20 CLEAR 31999
 30 READ add
 40 READ byte: IF byte=-1 THEN
GO TO 30
 50 IF byte=999 THEN STOP
 60 POKE add,byte
 70 LET add=add+1
 80 GO TO 40
100 DATA 32000,33,96,64,205,150
,125,33,150,64,205,150,125,33,200
4,64,205,150,125,33,100,64,205,200
,125,33,192,64,205,200,125,33,
0,72,17,32,0
110 DATA 205,200,125,25,205,150
,125,25,205,200,125,25,205,150,1
25
120 DATA 25,205,200,125,25,205,
150,125,25,205,200,125,25,205,15
0,125
```

```
  130 DATA 33,0,80,205,200,125,25
,205,150,125,25,205,200,125,25,2
05,150,125,25,205,200,125,25,201,-1
  140 DATA 32150,197,213,229,245,
43,17,32,0,6,8,197,35,126,203,23
,43,245,25,241,6,32,203,22,43,16
,251,36,193,15,236,241,225,209,1
93,201,-1
  150 DATA 32200,197,213,229,245,
17,32,0,6,8,197,43,25,126,203,31
,245,167,237,82,35,241,6,32,203,
30,35,16,251,167,237,82,35,193,1
6,230,241,225,209,193,201,-1
  160 DATA 32250,33,0,64,6,192,19
7,6,31,35,126,43,119,35,16,249,1
93,54,0,35,16,240,201,-1
  170 DATA 32300,6,1,205,0,125,16
,251,58,4,92,254,61,32,4,1,1,0,2
01,254,90,32,4,1,2,0,201,254,73,
32,4,1,3,0,201,254,80,32,4,1,4,0
,201,254,71,32,4,1,5,0,201,1,0,0
,201,-1
  180 DATA 32400,33,0,88,6,24,14,
31,35,126,43,119,35,13,32,246,54
,0,35,16,241,201,-1
  190 DATA 32500,6,1,197,33,15,0,
17,40,0,229,205,161,3,225,17,6,0
,167,237,90,125,254,255,32,237,1
93,16,230,201,-1
  195 DATA 32550,6,1,197,33,0,3,1
7,1,0,229,205,161,3,225,17,16,0,
167,237,82,32,240,193,16,233,201
,999
```

2/Type RUN again. When the program stops, type NEW to remove the program, and nothing else, as you still need the machine code.

3/ Type in the main program, program two which follows. Then save it directly after program one. To save it, use SAVE "ROLLERMAZE" - do not use line 9600 in the program.

PROGRAM 2...Main Game

```
   1 REM  ** Rolling Maze **
            Neil Pellinacci

  800 CLS : LET 0$="ENTER LEVEL O
F DIFFICULTY (1-5)"
  810 INPUT (0$);'LEV
  820 IF LEV<1 OR LEV>5 OR LEV<>I
NT LEV THEN LET 0$="ENTER A WHOL
E NUMBER, 1 TO 5": GO TO 810
  830 POKE 32551,6: POKE 32555,1:
 POKE 32557,1: POKE 32565,1
  900 BORDER 2: PAPER 7: CLS
  910 PRINT AT 0,0; PAPER 2; INK
7;"            THE ROLLING MAZE
  "
  920 PRINT AT 1,0; INK 6; PAPER
0;"TIME> 0      LIVES> 3    BEST>
000"
  930 PRINT INK 6; PAPER 0; OVER
1;AT 1,4;"<";AT 1,17;"<";AT 1,27
;"<"
  940 PRINT AT 1,29; PAPER 0; INK
 6;H(1)
  950 PRINT AT 2,0; INK 1;S$;AT 2
1,0;S$
  960 GO SUB 8000
1005 INK 7
1010 FOR A=4 TO 20 STEP 2: PRINT
 AT A,0;"##############################
##########": NEXT A
1030 FOR A=1 TO LEV: FOR B=3 TO
20 STEP 2
1040 LET B1=INT (RND*30)+1: PRIN
T AT B,B1;"##"
1050 NEXT B: NEXT A
1060 FOR A=1 TO LEV: FOR B=4 TO
20 STEP 2
1070 LET B1=INT (RND*30)+1: PRIN
T AT B,B1; INK 2;"  "
1080 NEXT B: NEXT A
1090 FOR A=3 TO 20: PRINT AT A,0
; OVER 1; INK 2;S$: NEXT A
1100 INK 0: RETURN
1900 GO SUB 800
1910 LET LIVES=3: LET T=0
2000 LET FIN=21: LET L=2: LET C=
15
2010 PRINT AT 1,6; PAPER 0; INK
6;T;AT L,C; INK 2; PAPER 7;" "
```

```
2020 LET A=USR 32300: IF A=5 THE
N GO TO 2120
2025 IF A=1 THEN IF L>2 THEN IF
SCREEN$ (L-1,C)<>"" THEN LET L=L
-1
2030 IF A=2 THEN IF L<21 THEN IF
 SCREEN$ (L+1,C)<>"" THEN LET L=
L+1
2035 LET C=C+(A=4 AND C<31)-(A=3
 AND C>0)
2040 IF SCREEN$ (L,C)="" THEN LE
T A=USR 32550: GO TO 2100
2050 IF L<>FIN THEN PRINT AT L,C
; INK 1;"*": LET T=T+1: GO TO 20
10
2060 IF FIN=21 THEN LET FIN=2: G
O TO 2010
2070 FOR B=1 TO 2: FOR A=40 TO -
20 STEP -2: BEEP .004,A: BEEP .0
04,A-15: BEEP .004,39-A: BEEP .0
04,39-A-15: NEXT A: NEXT B
2080 GO TO 2200
2100 LET LIVES=LIVES-1: PRINT AT
 1,19; PAPER 0; INK 6;LIVES
2110 IF LIVES<>0 THEN GO TO 2000
2120 PRINT AT 6,0; INK 1;S$;AT 1
5,0;S$
2130 PRINT AT 6,11; INK 8;"GAME
 OVER";AT 15,11;"GAME  OVER"
2140 FOR A=1 TO 256: LET B=USR 3
2300: NEXT A
2150 FOR A=1 TO 150: NEXT A: GO
TO 4000
2200 FOR A=1 TO 10: IF T<H(A) TH
EN GO TO 2230
2210 NEXT A: GO TO 4000
2230 FOR B=9 TO A STEP -1: LET H
(B+1)=H(B): LET H$(B+1)=H$(B): N
EXT B
2240 LET H(A)=T: INPUT "PLEASE T
YPE YOUR NAME:"; LINE N$: LET H$
(A)=D$: LET H$(A, TO LEN N$+1)=N
$
2250 GO SUB 5000: GO TO 4070
4000 BORDER 0: PAPER 0: CLS : RE
M TITLE PAGES
4010 GO SUB 5100: LET INK=6: GO
SUB 4500: REM TITLE
4020 FOR C=1 TO 20: GO SUB 4500:
 IF CHR$ PEEK 23556="S" OR CHR$
PEEK 23556="K" THEN GO TO 4100
4030 NEXT C
```



```
4040 GO SUB 5000: REM OPTIONS
4050 FOR C=1 TO 20: GO SUB 4500:
 IF CHR$ PEEK 23556="S" OR CHR$
PEEK 23556="K" THEN GO TO 4100
4060 NEXT C
4070 GO SUB 5500: REM HI-SCORES
4080 FOR A=1 TO 600: IF CHR$ PEE
K 23556="S" OR CHR$ PEEK 23556="
K" THEN GO TO 4100
4090 NEXT A: GO TO 4000
4100 LET A=USR 32500: IF CHR$ PE
EK 23556="S" THEN GO SUB 4200: G
O TO 1900
4110 GO SUB 4200: GO TO 4300
4200 FOR A=1 TO 34: LET B=USR 32
250+USR 32400: NEXT A: RETURN
4300 REM CHANGE KEYS
4310 INK 0: BORDER 6: PAPER 6: C
LS : BORDER 3
4315 POKE 32557,1: POKE 32551,10
: POKE 32555,1: POKE 32565,16
4320 PRINT AT 1,10; INK 0;"REDEF
INE KEYS"
4330 PRINT '' INK 1;"  PRESS YOU
R NEW KEY WHEN EACH"'TAB 9;"PROM
PT APPEARS"
4340 PRINT ''' INK 2;TAB 11; PAP
ER 7;"UP";TAB 21;
4350 LET K=PEEK 23556: IF K=255
THEN GO TO 4350
4360 POKE 32311,K: PRINT PAPER 7
;CHR$ K
4365 LET A=USR 32550: FOR A=1 TO
 150: NEXT A
4370 PRINT ' INK 2;TAB 11; PAPER
 7;"DOWN";TAB 21;
4380 LET K=PEEK 23556: IF K=255
THEN GO TO 4380
4390 POKE 32319,K: PRINT PAPER 7
;CHR$ K
4395 LET A=USR 32550: FOR A=1 TO
 150: NEXT A
4400 PRINT ' INK 2;TAB 11; PAPER
 7;"LEFT";TAB 21;
4410 LET K=PEEK 23556: IF K=255
THEN GO TO 4410
4420 POKE 32327,K: PRINT PAPER 7
;CHR$ K
4425 LET A=USR 32550: FOR A=1 TO
 150: NEXT A
```

```
4430 PRINT ' INK 2;TAB 11; PAPER
 7;"RIGHT";TAB 21;
4440 LET K=PEEK 23556: IF K=255
THEN GO TO 4440
4450 POKE 32335,K: PRINT PAPER 7
;CHR$ K
4455 LET A=USR 32550: FOR A=1 TO
 150: NEXT A
4460 PRINT ' INK 2;TAB 11; PAPER
 7;"ABORT";TAB 21;
4470 LET K=PEEK 23556: IF K=255
THEN GO TO 4470
4480 POKE 32343,K: PRINT PAPER 7
;CHR$ K
4490 LET A=USR 32550: GO TO 4000
4500 INK INK: RESTORE 4600: FOR
A=0 TO 2: READ A$: PRINT AT A,1;
A$: NEXT A
4510 FOR A=3 TO 5: READ A$: PRIN
T AT A,5;A$: NEXT A
4520 FOR A=7 TO 11: READ A$: PRI
NT AT A,14;A$: NEXT A
4530 PRINT INK 7;AT 8,1;"Written
 by";AT 10,4;"Neil";AT 12,1;"Pel
linacci"
4540 LET INK=INK+1: IF INK=8 THE
N LET INK=1
4595 INK 9: RETURN
4600 DATA "
4610 DATA "
4620 DATA "
4630 DATA "
"
4640 DATA "
"
4650 DATA "
"
4660 DATA "
"
4670 DATA "
4680 DATA "
4690 DATA "
4700 DATA "
4710 DATA "
5000 REM OPTIONS
5010 FOR A=10 TO 21: PRINT AT A,
0;S$: NEXT A
5020 INK 5: PRINT AT 15,10;"PRES
S:";AT 17,10; BRIGHT 1;"S"; BRIG
HT 0;"  TO START";AT 19,7; BRIGH
T 1;"K"; BRIGHT 0;"  TO CHANGE K
EYS"
```



```
5030 RETURN
5100 INK 6: PRINT AT 15,9;"GAME
CONTROLS";AT 17,6;"- LEFT";AT 17
,22;"- UP";AT 19,6;"- RIGHT";AT
19,21;"- DOWN"
5110 PRINT BRIGHT 1;AT 17,4;CHR$
 PEEK 32327;AT 17,20;CHR$ PEEK 3
2311;AT 19,4;CHR$ PEEK 32335;AT
19,20;CHR$ PEEK 32319
5120 PRINT AT 21,11; BRIGHT 1;CH
R$ PEEK 32343; BRIGHT 0;" - ABOR
T"
5130 INK 0: RETURN
5500 BORDER 0: PAPER 0: CLS : IN
K 7: PRINT AT 0,6; INK 6; BRIGHT
 1;"TODAY'S BEST SCORES"
5510 FOR A=1 TO 10: PRINT '" " A
ND A<>10;A;"  ";H$(A);"..";H(A):
 NEXT A
5515 PRINT #0; INK 5;"
RESS "; BRIGHT 1;"S"; BRIGHT 0;"
 OR "; BRIGHT 1;"K"
5520 RETURN
6000 REM JINGLE
6010 RESTORE 6400
6020 READ A,B: IF A=99 THEN RETU
RN
6025 IF A=88 THEN PAUSE B*5: GO
TO 6020
6030 BEEP A/3,B-3+LEV: GO TO 602
0
6400 DATA .3,4,.3,4,.6,10,.3,4,.
3,4,.6,10,.3,4,.3,4,.9,16.1
6410 DATA 88,19
6420 DATA .4,16,.4,12,.4,6,.4,0,
.8,4.5
6430 DATA 88,20,.5,0,.6,4
6500 DATA 99,0
9000 REM FORM UDG'S
9010 RESTORE 9000: FOR A=USR "A"
 TO USR "B"+7: READ B: POKE A,B:
 NEXT A
9030 DATA 16,56,16,124,16,40,68,
68,24,126,126,255,255,126,126,24
9050 LET S$="

9060 DIM H(10): DIM H$(10,20)
9070 FOR A=1 TO 10: LET H$(A)="N
EIL PELLINACCI ...": NEXT A
9080 FOR A=1 TO 10: READ B: LET
H(A)=B: NEXT A
```

```
9090 DATA 350,390,430,470,510,55
0,590,630,720,1000
9100 LET D$="....................
.."
9200 GO TO 4000
9500 FOR Z=32550 TO 32560: PRINT
 Z;" ";PEEK Z: NEXT Z
9510 STOP
9600 SAVE "ROLLERMAZE" LINE 9700
: SAVE "ROLLERCODE"CODE 32000,60
0
9610 STOP
9700 CLEAR 31999: LOAD ""CODE :
GO TO 9000
```

User Defined Graphics
A...A B...B

4/ Type RUN 9000. When the title page appears, press "S" to play. If the program crashes, you've made a mistake somewhere. If the program runs properly, try redefining the keys. Test everything.

5/ When you're happy with the program, save your final copy, using RUN 9600. Then check it by using VERIFY "ROLLERMAZE" and then VERIFY "ROLLERCODE"CODE 32000,600

6/ Play the game, either loading it, or by typing RUN 9000

The program, as is pretty obvious, is written in BASIC and machine code. The machine code reads the keyboard, rolls the maze and handles sound and other special effects. Various POKEs in the program control these routines. Line 9500 can be used for checking machine code by changing

the addresses. There is enough space in the machine code memory map for your own special effects routines, if you want to add them.

If you own a 48K machine and you want to move the machine code higher up in memory, you must remember to:

i/ Change the USR addresses in the main programs

ii/ Change individual bytes within the machine code, as some routines call others. For those who know about such things, the two address bytes following CALL (205 decimal) will have to be changed

iii/ Make the necessary adjustments to the machine code loader program

Three-D Maze

This great program was written by Scott Vincent. You need a 48K computer to run it.

A ten by ten maze, with a single entrance/exit is stored in the computer's memory. The object of the game is to wander through the maze, find the treasure and then get back to where you started.

If you lose your sense of direction, you can press the "R" key for a repeat. This takes you through the maze from the beginning, showing you quickly all of your previous moves. When the repeat has finished, you continue on with your trek.

Use the following keys to move around the maze:

1 - to move left
2 - to move right
0 - to go forwards
L - to turn around

Note that you need to press L twice in order to turn through 180 degrees; you cannot press 1 or 2 twice.

If you manage to get out of the maze with the trasure, you'll be told how many moves you took. The maze is not generated at random, so it is the same from game to

game. However, the treasure is placed in a random location each time you run the program.

```
  5 BORDER 5: PAPER 5: INK 1: C
LS
 10 RANDOMIZE : DIM M(9): DIM N
(9): DIM M$(201,9): DIM N$(200,1
2)
 20 FOR x=0 TO 31: READ y: POKE
 USR "a"+x,y: NEXT x
 30 FOR x=1 TO 9: READ M(x): NE
XT x
 40 FOR x=1 TO 9: READ N(x): NE
XT x
 50 FOR x=1 TO 200: READ M$(x):
 NEXT x
 60 FOR x=1 TO 200: READ N$(x):
 NEXT x
 70 LET Q$="001": LET P$="": LE
T L=1: LET Tr=0: LET x=1+INT (RN
D*7)
 80 LET La=201*(x=1 OR x=4 OR x
=6 OR x=7)+81*(x=2)+127*(x=3)+16
9*(x=5)
 90 LET Lb=201*(x=4 OR x=6)+100
*(x=1)+84*(x=2)+122*(x=3)+172*(x
=5)+196*(x=7)
 100 LET Lc=94*(x=1)+96*(x=2)+12
4*(x=3)+131*(x=4)+117*(x=5)+191*
(x=6)+199*(x=7)
 110 CLS : FOR x=1 TO 9: IF M$(L
,x)="0" THEN GO SUB M(x)
 120 NEXT x
 130 FOR x=1 TO 9: IF M$(L,x)="1
" THEN GO SUB N(x)
 140 NEXT x
 150 IF L=2 OR L=La OR L=Lb OR L
=Lc THEN GO SUB 220: IF L=2 AND
Tr=1 THEN GO TO 70
 160 LET A$=INKEY$: IF A$="" THE
N GO TO 160
 170 LET P=VAL N$(L, TO 3)*(A$="
1")+VAL N$(L,4 TO 6)*(A$="0")+VA
L N$(L,7 TO 9)*(A$="2")+VAL N$(L
,10 TO )*(A$="l")
 180 IF A$="r" THEN GO TO 300
 190 IF P=0 THEN GO TO 160
```

```
 200 BEEP .008,0: LET L=P: LET Q
$=Q$+("0" AND L<100)+("0" AND L<
10)+STR$ L
 210 LET P$=P$+A$: GO TO 110
 220 IF L=2 AND Tr=0 THEN CLS :
PRINT AT 10,12; FLASH 1;"ENTRANC
E": RETURN
 230 IF L=2 THEN CLS : PRINT AT
8,4;"Well done, that took you";A
T 10,11;LEN P$;" moves.";AT 14,0
;"press any key for another game
.": PAUSE 4e4: RETURN
 240 IF L=La THEN PLOT 116,99: D
RAW OVER 1;22,0: PLOT 116,76: DR
AW OVER 1;22,0: RETURN
 250 IF L=Lb THEN PLOT 100,115:
DRAW OVER 1;54,0: PLOT 100,60: D
RAW OVER 1;54,0: RETURN
 260 IF L=Lc THEN PLOT 72,143: D
RAW OVER 1;110,0: PLOT 72,32: DR
AW OVER 1;110,0
 270 IF Tr=1 THEN PRINT AT 9,11;
"You already";AT 10,12;"have the
";AT 11,12;"treasure.": RETURN
 280 LET Tr=1: PRINT AT 9,11;"Yo
u have";AT 10,11;"found the";AT
11,11;"treasure."
 290 RETURN
 300 LET y=1
 310 FOR n=1 TO LEN Q$ STEP 3
 320 IF n=1 THEN GO TO 390
 330 IF P$(y)="1" THEN PRINT #1;
AT 0,6;"A"
 340 IF P$(y)="0" THEN PRINT #1;
AT 0,15;"B"
 350 IF P$(y)="2" THEN PRINT #1;
AT 0,25;"C"
 360 IF P$(y)="l" THEN PRINT #1;
AT 1,15;"D"
 370 LET y=y+1
 380 FOR x=1 TO 25 STEP 5: FOR s
=x TO x+4 STEP 1.5: BEEP .004,x:
 BEEP .004,x+s: NEXT s: NEXT x
 390 LET L=VAL Q$(n TO n+2)
 400 CLS : FOR x=1 TO 9: IF M$(L
,x)="0" THEN GO SUB M(x)
 410 NEXT x
 420 FOR x=1 TO 9: IF M$(L,x)="1
" THEN GO SUB N(x)
 430 NEXT x
```



```
 440 IF L=2 OR L=La OR L=Lb OR L
=Lc THEN GO SUB 220
 450 NEXT n: GO TO 160
5000 PLOT 40,175: DRAW 32,-32: P
LOT 40,0: DRAW 32,32: RETURN
5005 PLOT 24,175: DRAW 0,-175: D
RAW 0,32: DRAW 48,0: DRAW 0,111
 DRAW -48,0: RETURN

5010 PLOT 183,143: DRAW 32,32: P
LOT 183,32: DRAW 32,-32: RETURN
5015 PLOT 231,175: DRAW 0,-175:
DRAW 0,32: DRAW -48,0: DRAW 0,11
1: DRAW 48,0: RETURN
5020 PLOT 73,142: DRAW 27,-27: P
LOT 73,33: DRAW 27,27: RETURN
5025 PLOT 73,142: DRAW 11,-11: D
RAW 0,-87: DRAW -11,-11: PLOT 85
,60: DRAW 15,0: DRAW 0,55: DRAW
-15,0: RETURN
5030 PLOT 155,115: DRAW 27,27: P
LOT 182,33: DRAW -27,27: RETURN
5035 PLOT 182,33: DRAW -11,11: D
RAW 0,87: DRAW 11,11: PLOT 170,6
0: DRAW -15,0: DRAW 0,55: DRAW 1
5,0: RETURN
5040 PLOT 101,114: DRAW 19,-19:
PLOT 101,61: DRAW 19,19: RETURN
5045 PLOT 101,114: DRAW 8,-8: DR
AW 0,-37: DRAW -8,-8: PLOT 110,9
9: DRAW 6,0: DRAW 4,-4: DRAW -4,
4: DRAW 0,-23: DRAW -6,0: DRAW 6
,0: DRAW 4,4: RETURN
5050 PLOT 154,114: DRAW -19,-19:
 PLOT 154,61: DRAW -19,19: RETUR
N
5055 PLOT 154,114: DRAW -8,-8: D
RAW 0,-37: DRAW 8,-8: PLOT 145,9
9: DRAW -6,0: DRAW -4,-4: DRAW 4
,4: DRAW 0,-23: DRAW 6,0: DRAW -
6,0: DRAW -4,4: RETURN
5060 PLOT 73,143: DRAW 110,0: DR
AW OVER 1;0,-110: DRAW 0,-1: DRA
W -111,0: DRAW OVER 1;0,110: RET
URN
5065 PLOT 101,115: DRAW 54,0: DR
AW OVER 1;0,-54: DRAW 0,-1: DRAW
 -55,0: DRAW OVER 1;0,54: RETURN
```

```
5070 OVER 1: PLOT 117,77: DRAW 3
,3: PLOT 138,98: DRAW -3,-3: PLO
T 138,77: DRAW -3,3: PLOT 120,95
: DRAW -4,4: OVER 0: DRAW 23,0:
DRAW OVER 1;0,-22: DRAW 0,-1: DR
AW -23,0: DRAW OVER 1;0,23: RETU
RN
5980 DATA 16,48,96,255,255,96,48
,16,24,60,126,219,24,24,24,24
5990 DATA 8,12,6,255,255,6,12,8,
24,24,24,24,219,126,60,24
6000 DATA 5000,5010,290,5020,503
0,290,5040,5050,290
6010 DATA 5005,5015,5060,5025,50
35,5065,5045,5055,5070
6020 DATA "111","","0000000010","
101","0000010000","010101","01000
00011","0000010101","0000010111","0
00011","0000000010","011","100000
000","111","0001000000","101","10
1","001","010101","101","100001"
,"011","001","011","101"
,"010000000","011","010
6030 DATA "011","0001100011","011","011","10
1","100011","101","000111","000011","
,"011","111","000111","000011","
,100000111","010000101","101","11
1","000101","010101","101","1000
10101","001","0001000010"
6040 DATA "011","010111","111","
011","1000100000","011","001","10
1","110011","111","011","000111"
,"001","110001","1100000111
","001","101","110001","111","00
0010100","101","010100101","0101
01","111"
6050 DATA "100101","0000010101","
01000000000","101","1000000010","10
00000001","0101000000","011","0000
01","111","111","0000000111","111
","100111","000111","0001001111",
"111","0000000100","001","001","1
00001","010011","001","0000010011
","010001"
6060 DATA "0101100000","1000000010
","1100000000","011","0000000101",
"111","000101","000111","101","1
10101","0000000111","101","011","
0000000110","111","000101","001",
"1100000000","1000000101","111","0
11","000001","100011","001","000
100011"
```



```
5070 DATA "100011","0100000001","
011","101","101","001","111","01
1","010101","011","101","101","1
01","0000001110","011","0001100000
","0000011","1100000000","0000000011
1","100011","0000000011","0000001
10","1100000000","0001101111","000
110000"
6080 DATA "110111","110011","000
011","011","011","011","000011"
,"101","101","111","101","011","0
0010000","011","101","1000001000
","101","010001","000111","1101000
11","111"
6090 DATA "0001101000","0000001100
","0000001001","0000100000","0100
00010","1000001010","0000100000","
101","000101","0000000010","111","
"011","001","011","101","101","0
1000000","000001","0000100000","
011","001","000011"
6100 DATA "003000004002","000000
000001","000005000006","025000000
026","000007000008","000004000200
03","000010009011","000006000005
","000052000053","000012000013","
,"000008000007","000000014015"
6110 DATA "009011000010","018000
017016","000013000012","015000000
0014","020000000021","0000000000
19","000017016018","023000000022
","016018000017","000000021020","
,"000000000024","000000022023"
6120 DATA "028000000027","002003
000004","000000026025","000030020
9031","000059000060","000000003200
33","000000027028","035000000034
","029031000030","033000000032","
,"000036000037","039000038040"
6130 DATA "000000034035","042000
043041","000053000052","000037000
0036","040039000038","000045044000
46","054000000055","071000072070
","000047000048","000043041042","
,"049000000050","044046000045"
```

```
6140 DATA "000000000051","000048
000047","000000050049","00003804
0039","011000010009","0000000570
56","041042000043","000000055054
","000000000058","056000000057",
"063061064062","031000030029"
6150 DATA "000000067066","000060
000059","000000000069","00000000
0065","062063061064","0640620630
61","000000000068","066000000067
","051064062063","046000045044",
"000073000074","101000000102"
6160 DATA "000076075077","000072
070071","121000122120","07807900
0080","000074000073","0000870880
86","031000000082","075077000076
","085084000083","000080078079",
"000000082081","000098000097"
6170 DATA "066000067068","079000
080078","000090000089","084000088
3085","086086000087","000092000090
91","000089000090","095000094093
","000091000092","000000000096",
"000000000100","093095000094"
6180 DATA "000083085084","000000
000099","000097000098","00009409
3095","000103105104","070071000072
","146144147145","000000102101
","000107000106","104000103105",
"000109000108","000106000107"
6190 DATA "110000000111","114112
115113","000108000109","153000000
0152","000000111110","000150000151
","116000117118","000171000172
","000000000119","113114112115",
"118116000117","077000076075"

6200 DATA "000000126127","000124
000123","120121000122","00000000
0125","000123000124","130126000129
","000122120121","000000135134
","127000000126","131000000132",
"000000000133","128000129130"
6210 DATA "000000132131","000129
130128","000000136137","139000000
0138","134000000135","137000000136
","000141000140","000000138139
","000143000142","000140000141",
"144147145146","000173000174"
```



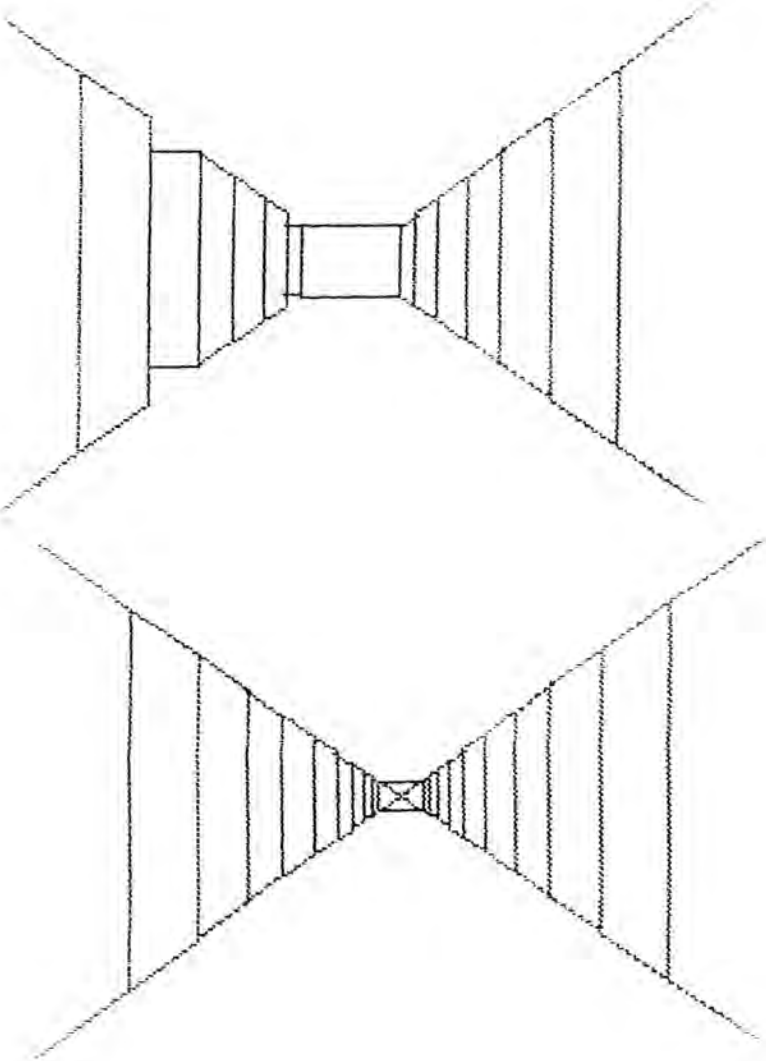
```
6220 DATA "105104000103","000142
000143","000149000148","14514514
4147","000151000150","00014800001
49","112115113114","115113114112
","000154000155","000000156157",
"000000152153","000000159158"
6230 DATA "000155000154","157000
000156","160000000161","163000016
4162","158000000159","0000001611
60","000180000181","000000165166
","168000000167","162163000164",
"166000000165","000000169170"
6240 DATA "000172000171","000000
167168","170000000169","00011711
6116","000175000176","147145145
44","178000179177","000174000173
","000176000175","000182000183",
"000181000180","177178000179"
6250 DATA "000164162163","000184
000185","000179177178","18618700
0188","000183000182","190000000
89","000194000195","000185000184
","187000188186","000000191192",
"000000000193","000000189190"
6260 DATA "192000000191","196000
000197","000188186187","00019900
0198","000195000194","0000001971
95","000000000200","000198000199
"
```

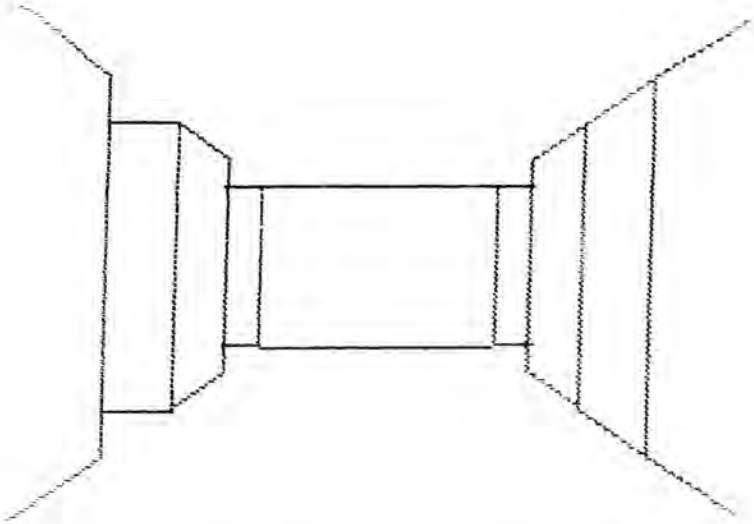
Three-D Maze II

Once you've mastered Scott Vincent's program, you can progress on to this one. In this 3-D maze program, written by Graham Charlton, you once again see the maze in three dimensions. However, unlike Scott's program, this one generates a new maze each time you run the game.

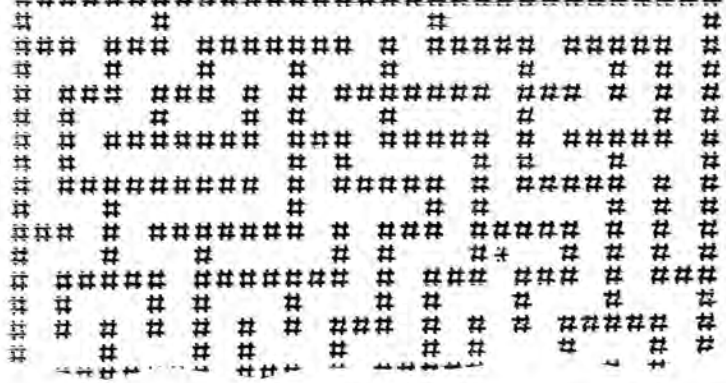
The maze is built on a 22 X 32 grid, and the computer takes about a minute to

create a full maze. Here are some shots of it in action:





Your object is to move from the top left hand corner of the screen to the bottom right hand one. You press the "R" to turn right, "L" to go left, and "F" to go forward. Pressing "H" [for 'help'] will show you the maze from above, and will show you where you are on it. You start facing south.

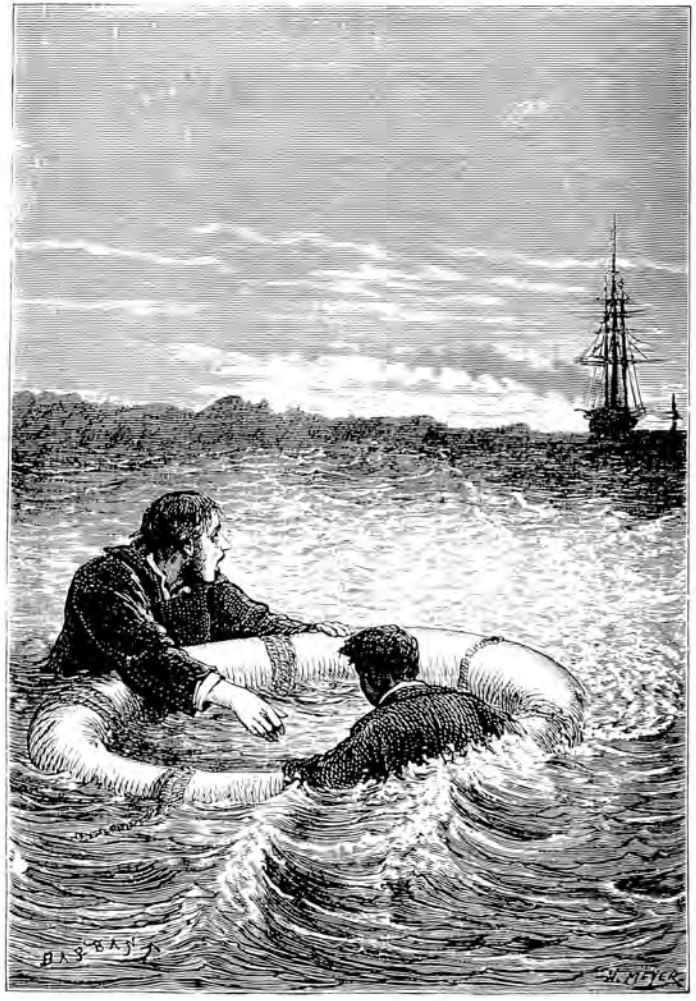


```
  10 DIM c$(23,33)
  20 DIM a$(1000)
  30 DIM b$(1000)
  40 LET x=21
  50 LET y=31
  60 LET z=1
  70 FOR w=2 TO 22
  80 LET c$(w)="  ################
################  "
  90 NEXT w
 100 LET c$(x,y)="  "
 110 IF c$(x+2,y)="#" OR c$(x-2,
y)="#" OR c$(x,y+2)="#" OR c$(x,
y-2)="#" THEN GO TO 1000
 120 LET z=z-1
 130 IF z=0 THEN GO TO 2000
 140 LET x=CODE a$(z)
 150 LET y=CODE b$(z)
 160 GO TO 100
1000 LET a$(z)=CHR$ x
1010 LET b$(z)=CHR$ y
1020 LET z=z+1
1030 LET r=INT (RND*4)
1040 LET c=(r=0)-(r=1)
1050 LET d=(r=2)-(r=3)
1060 IF c$(x+2*c,y+2*d)="  " THEN
 GO TO 1030
1070 LET c$(x+c,y+d)="  "
1080 LET c$(x+2*c,y+2*d)="  "
1090 LET x=x+2*c
1100 LET y=y+2*d
1110 GO TO 100
2000 LET x=3
2010 LET y=3
2020 LET r=1
2100 PRINT AT 0,0;
2110 FOR w=2 TO 22
2120 PRINT c$(w,2 TO );
2130 NEXT w
2140 PRINT FLASH 1;AT x-2,y-2;"#
"
2150 IF INKEY$="" THEN GO TO 214
0
2500 CLS
2510 LET k=127
2520 LET l=87
3000 LET p=(r=1)-(r=3)
3010 LET q=(r=2)-(r=4)
3100 LET j=0
```



```
3110 IF c$(x+j*p+p,y+j*q+q)="#"
THEN PLOT 128-k,88+l: DRAW 2*k,0
: PLOT 128-k,88-l: DRAW 2*k,0: G
O TO 3200
3120 LET j=j+1
3130 IF c$(x+j*p-q,y+j*q+p)="#"
THEN PLOT 128-k,88+l: DRAW k/4,-
l/4: DRAW 0,-l*3/2: DRAW -k/4,-l
/4: GO TO 3150
3140 PLOT 128-k,88+l*3/4: DRAW k
/4,0: DRAW 0,-l*3/2: DRAW -k/4,0
3150 IF c$(x+j*p+q,y+j*q-p)="#"
THEN PLOT 128+k,88+l: DRAW -k/4,
-l/4: DRAW 0,-l*3/2: DRAW k/4,-l
/4: GO TO 3170
3160 PLOT 128+k,88+l*3/4: DRAW -
k/4,0: DRAW 0,-l*3/2: DRAW k/4,0
3170 LET k=k*3/4
3180 LET l=l*3/4
3190 GO TO 3110
3200 IF x+j*p=21 AND y+j*q=31 TH
EN PLOT 128-k,88+l: DRAW 2*k,-2*
l: PLOT 128-k,88-l: DRAW 2*k,2*l
3210 IF x=21 AND y=31 THEN STOP
3220 LET z$=INKEY$
3230 IF z$="" THEN GO TO 3220
3240 IF z$="f" AND c$(x+p,y+q)="
 " THEN LET x=x+p: LET y=y+q
3250 IF z$="r" THEN LET r=r-1
3260 IF z$="l" THEN LET r=r+1
3270 IF r=5 THEN LET r=1
3280 IF r=0 THEN LET r=4
3290 IF z$="h" THEN GO TO 2100
3300 GO TO 2500
```

TWO PLAYER GAMES



Squares

This program, from Neil Pellinacci, includes versions for either one or two players. In both versions, the game is played on a ten by ten board.

You have to place counters on the board, following the rule that you cannot place a placing a piece on, or adjacent to, one of the computer's pieces.

The one-player game is called 'Area' and you play against the computer. The first player who cannot legally place a piece on the board is the loser. In this game, your move will be rejected if it involves placing a piece on or adjacent to one of the computer's moves.

The computer moves randomly unless it has problems finding a move, in which case it will search methodically. If it still cannot move, it will admit defeat.

The two-player game is called 'Lines' and the winner is the first person to get five squares of their color in a row, horizontally or vertically. A move is rejected if the square is already filled. However, if you move adjacent to [above, below, or to the side of] one of your opponent's squares, the piece becomes one of the opponent's. There are a number of winning techniques which can be used [as

there are in the one-player game). One slight mistake can cost you the game.

It is essential that CAPS LOCK is switched ON during this game. When the program starts, you will see the title page, and four options A to D. If you cannot select an option, check CAPS LOCK before you check your program.

The first two options, one and two player games, are fairly obvious. The third option, to change colors, allows you to redefine both players' colors, by pressing the appropriate key, 1 to 7, when the prompt appears. The two colors cannot be the same.

Asking for information will give you a brief rundown on the aims of the game.

When you play a game, you'll be presented with:

PLAYER ONE NAME ['ENTER' TO SKIP]

Type your name, or if the player's name is the same as before, simply press ENTER. The same goes for player two, when necessary.

The display shows the board when the game is in progress, together with a score display, which shows which player is which color and how many squares of that color

there are on the board. A prompt appears at the bottom right of the screen, indicating whose move it is. Below this is the last move entered, displayed in white.

Enter your move as a number and a letter, eg 7A, 1D in that order. Entering AA will terminate the game, and return you to the options page.

The program is written entirely in BASIC and is made up from several modules which are linked together by the main routine, lines 540 to 740.

```
  500 REM LINES
  510 REM *****
  530 GO SUB 9000: GO SUB 8000
  540 GO SUB 9500: BORDER 0: PAPE
R 0: CLS : INK 7
  600 GO SUB 9000: PAPER 0: INK 7
  700 GO SUB 6000: GO SUB 2000: R
EM PLAYER ONE
  705 IF UF OR AF THEN GO TO 5000
  707 GO SUB 6000
  710 IF NP=1 THEN GO SUB 3000
  720 IF NP=2 THEN GO SUB 4000
  725 IF UF OR AF THEN GO TO 5000
  740 GO TO 700
 1000 STOP
 2000 REM PLAYER ONE
 2010 BEEP .3,-10: PRINT INK 6;AT
 16,22;"PLAYER ONE";AT 17,22;P$
 2020 PRINT AT 18,22; INK 6;"TO M
OVE:";
 2030 GO SUB 2600: PRINT AT 19,23
;A$
 2031 IF AF THEN LET AF=1: RETURN
 2032 IF B(X,Y)<>0 THEN GO TO 203
0
 2034 IF NP=1 THEN GO TO 2100
 2035 LET LL=(Y-2)*2+1: LET CC=(X
-2)*2+1
```

```
2040 LET P1S=P1S+1: LET B(X,Y)=1
: PRINT AT LL,CC; OVER 1; PAPER
C1;"  ";AT LL+1,CC;"  "
2050 IF B(X,Y+1)=2 OR B(X,Y-1)=2
 OR B(X+1,Y)=2 OR B(X-1,Y)=2 THE
N LET B(X,Y)=2: PRINT AT LL,CC;
OVER 1; PAPER C2;"  ";AT LL+1,CC
;"  ": LET P2S=P2S+1: LET P1S=P1
S-1
2052 BEEP .2,10
2055 IF NP=1 THEN LET WF=0: RETU
RN
2060 LET WF=0: GO SUB 7500
2070 RETURN
2100 IF B(X,Y+1)=2 OR B(X,Y-1)=2
 OR B(X+1,Y)=2 OR B(X-1,Y)=2 THE
N GO TO 2030
2110 GO TO 2035
2599 REM MOVE
2500 INPUT LINE A$: IF LEN A$<>2
 THEN GO TO 2600
2505 IF A$="AA" THEN LET AF=1: R
ETURN
2510 IF CODE A$(1)<48 OR CODE A$
(1)>57 THEN GO TO 2600
2520 IF CODE A$(2)<65 OR CODE A$
(2)>74 THEN GO TO 2600
2530 LET X=CODE A$(1)-46
2540 LET Y=CODE A$(2)-63
2550 RETURN
2999 REM SPECTRUM
3000 REM SPECTRUM
3010 PRINT AT 16,22; INK 6;"PLAY
ER TWO";AT 17,22;0$;AT 18,22;"TO
 MOVE:"
3015 FOR A=1 TO 15
3020 LET X=INT (RND*10)+2: LET Y
=INT (RND*10)+2
3030 IF B(X,Y)<>0 THEN GO TO 302
0
3040 IF B(X+1,Y)=1 OR B(X-1,Y)=1
 OR B(X,Y+1)=1 OR B(X,Y-1)=1 THE
N NEXT A: GO SUB 3500: IF WF THE
N RETURN
3045 LET LL=(Y-2)*2+1: LET CC=(X
-2)*2+1: PRINT AT LL,CC; OVER 1;
 PAPER C2;"  ";AT LL+1,CC;"  "
3050 LET B(X,Y)=2: LET P2S=P2S+1
: RETURN
3500 REM SEARCH
```



```
3510 FOR X=2 TO 11: FOR Y=2 TO 1
1: IF B(X,Y)<>0 THEN GO TO 3540
3520 IF B(X+1,Y)=1 OR B(X-1,Y)=1
 OR B(X,Y+1)=1 OR B(X,Y-1)=1 THE
N GO TO 3540
3530 RETURN
3540 NEXT Y: NEXT X: LET WF=1: R
ETURN
3999 RETURN
4000 REM PLAYER TWO
4010 BEEP .3,-10: PRINT INK 6;AT
 16,22;"PLAYER TWO";AT 17,22;0$
4020 PRINT AT 18,22; INK 6;"TO M
OVE:";
4030 GO SUB 2600: PRINT AT 18,23
;0$
4031 IF AF THEN LET AF=2: RETURN

4032 IF B(X,Y)<>0 THEN GO TO 403
0
4035 LET LL=(Y-2)*2+1: LET CC=(X
-2)*2+1
4040 LET P2S=P2S+1: LET B(X,Y)=2
: PRINT AT LL,CC; OVER 1; PAPER
C2;"  ";AT LL+1,CC;"  "
4050 IF B(X,Y+1)=1 OR B(X,Y-1)=1
 OR B(X+1,Y)=1 OR B(X-1,Y)=1 THE
N LET B(X,Y)=1: PRINT AT LL,CC;
OVER 1; PAPER C1;"  ";AT LL+1,CC
;"  ": LET P1S=P1S+1: LET P2S=P2
S-1
4055 BEEP .2,20
4060 LET WF=0: GO SUB 7500
4070 RETURN
5000 REM GAME OVER
5010 BRIGHT 1: PAPER 1: FOR A=10
 TO 21: PRINT AT A,22;"
": NEXT A
5020 PRINT AT 13,23;"GAME OVER"
5030 IF AF THEN PRINT AT 15,24;"
STOPPED";AT 16,23;"BY PLAYER";AT
 17,27;AF
5040 IF WF THEN PRINT AT 15,23;"
PLAYER ";WF;AT 16,25;"WINS"
5050 PRINT AT 20,23; INK 6;"PRES
S ANY";AT 21,26;"KEY"
5060 BRIGHT 0: PAPER 0: GO SUB 6
000: FOR K=-20 TO 40 STEP 2: BEE
P .05,K: BEEP .01,30-K: NEXT K
5070 PAUSE 0: GO TO 540
6000 REM PRINT SCORES
```

```
6010 PRINT AT 2,22; INK 7;"PLAYE
R ONE";AT 3,22;P$;AT 4,23; INK 7
; PAPER C1;"O"; PAPER 0; INK 5;"
 ";P1$;"  "
6020 PRINT AT 9,22; INK 7;"PLAYE
R TWO";AT 10,22;0$;AT 11,23; INK
 7; PAPER C2;"O"; PAPER 0; INK 5
;" ";P2$;"  "
6030 RETURN
7000 RETURN
7500 REM CHECK FOR WIN
7510 FOR X=2 TO 11: FOR Y=2 TO 1
4
7520 IF B(X,Y)<>0 THEN LET A=B(X
,Y): GO SUB 7800: IF WF=1 THEN R
ETURN
7530 NEXT Y: NEXT X: RETURN
7800 LET X1=X: LET Y1=Y
7810 FOR Z=1 TO 4: LET X1=X1+1:
IF B(X1,Y1)<>A THEN GO TO 7830
7820 NEXT Z: LET WF=A: RETURN
7830 LET X1=X: LET Y1=Y
7840 FOR Z=1 TO 4: LET Y1=Y1+1:
IF B(X1,Y1)<>A THEN GO TO 7860
7850 NEXT Z: LET WF=A: RETURN
7860 RETURN
7999 RETURN
8000 REM VARIABLES
8010 DIM P$(10): DIM 0$(10): LET
 C1=4: LET C2=6
8020 DIM B(12,12)
8030 POKE 23609,100
8500 RETURN
8600 REM INFO
8610 BORDER 1: PAPER 1: INK 6: C
LS
8620 PRINT AT 1,13;"AREA";AT 2,1
3;"----"
8630 PRINT '"TAB 8;"A ONE-PLAYER
 GAME"''" YOU MUST BEAT THE COM
PUTER, YOU"''"PLACE SQUARES ON T
HE BOARD SO   "''"THAT NONE ARE
ADJACENT TO THOSE "''"OF THE COM
PUTER."
8640 PRINT '" THE COMPUTER WILL
PLAY LIKEWISE"''"AND WILL TELL Y
OU WHO HAS WON."
8650 PRINT '' INK 7;"   PRESS AN
Y KEY TO CONTINUE   "
8660 PAUSE 0: BORDER 6: PAPER 6:
 CLS : INK 0
```



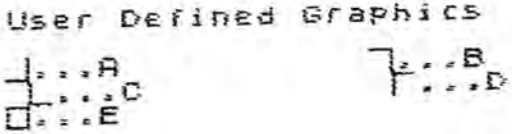
```
8670 PRINT AT 1,12;"LINES";AT 2,
12;"-----"
8680 PRINT '"TAB 7;"A TWO-PLAYER
 GAME"''" YOU MUST BEAT YOUR OP
PONENT BY"''"GETTING FIVE SQUARE
S IN A ROW"''"BEFORE HE DOES."
8690 PRINT '" THE LINES CAN BE H
ORIZONTAL OR"''"VERTICAL."
8700 PRINT '' INK 2;"   PRESS AN
Y KEY TO CONTINUE"
8710 PAUSE 0: BORDER 4: PAPER 4:
 CLS : INK 0
8715 PRINT '"TAB 7;"SQUARES con
tinued";TAB 7;"----------------
"
8720 PRINT '" IF YOU PLACE A SQ
UARE ON THE "''"BOARD ADJACENT T
O ONE OF YOUR"''"OPPONENT'S, IT
WILL IMMEDIATELY"''"BECOME ONE O
F HIS."
8730 PRINT '' INK 7;"   PRESS A
NY KEY TO SELECT GAME"
8740 PAUSE 0: GO TO 9500
9000 REM UDG'S
9010 FOR A=0 TO 6: POKE USR "A"+
A,1: POKE USR "G"+7-A,1: NEXT A
9020 POKE USR "A"+7,255: POKE US
R "B",255
9030 FOR A=0 TO 6: POKE USR "C"+
A,128: POKE USR "D"+7-A,128: NEX
T A
9040 POKE USR "C"+7,255: POKE US
R "D",255
9050 POKE USR "E",255: FOR A=1 T
O 6: POKE USR "E"+A,129: NEXT A:
 POKE USR "E"+7,255
9060 RETURN
9200 REM DRAW BOARD
9210 PAPER 0: CLS : INK 7
9220 FOR A=4 TO 20 STEP 2: PRINT
 AT A,4;"                ";AT A+1,4;"                "
: NEXT A
9230 PAPER 7: INK 2: FOR A=0 TO
9: PRINT AT 0,A*2+1;A;" ";AT 21,
A*2+1;A;" ": NEXT A
9240 FOR A=1 TO 10: LET A1=A*2
9250 PRINT AT A1,0;CHR$ (A+64);A
T A1,24;CHR$ (A+64);AT A1+1,0;"
";AT A1+1,24;" ": NEXT A
```

```
9260 PRINT AT 0,0;" ";AT 1,0;" "
;AT 1,21;" ";AT 0,21;" "
9300 RETURN
9500 REM OPTIONS
9505 LET AF=0: LET WF=0: LET P1S
=0: LET P2S=0: DIM S(12,12)
9510 BORDER 2: PAPER 2: INK 7: C
LS
9520 PRINT AT 1,5;"###SQUARES###
";AT 3,0;"WRITTEN BY NEIL PELLIN
ACCI  1983"
9530 PRINT '''TAB 12;"OPTIONS"
9540 PRINT 'TAB 6;"[A] ONE PLAY
ER GAME"
9550 PRINT 'TAB 6;"[B] TWO PLAYE
R GAME"
9560 PRINT 'TAB 6;"[C] CHANGE CO
LOURS"
9565 PRINT 'TAB 6;"[D] INFORMATI
ON"
9570 PRINT '''"    CHOOSE YOUR OP
TION NOW BY "'"  PRESSING THE AP
PROPRIATE KEY"
9580 IF INKEY$="A" THEN LET NP=1
: GO TO 9700
9590 IF INKEY$="B" THEN LET NP=2
: GO TO 9800
9595 IF INKEY$="D" THEN GO TO 85
00
9600 IF INKEY$<>"C" THEN GO TO 9
580
9610 BORDER 7: PAPER 7: CLS : IN
K 1: PRINT '''"PLAYER ONE COLOUR
.":"'" PRESS THE KEY FOR THE COLO
UR   YOU WANT NOW"
9620 LET A=CODE INKEY$
9630 IF A<49 OR A>55 THEN GO TO
9620
9640 LET C1=A-48: BEEP .5,C1
9650 PRINT '''"PLAYER TWO COLOUR
.":"'" PRESS THE KEY FOR THE COLO
UR   YOU WANT NOW"
9660 LET A=CODE INKEY$: IF A<49
OR A>55 THEN GO TO 9660
9670 LET C2=A-48: IF C2=C1 THEN
GO TO 9660
9680 BEEP .5,C2: GO TO 9500
9700 INPUT "TYPE YOUR NAME ['ENT
ER' TO SKIP]"; LINE A$
9710 IF A$="" THEN GO TO 9730
9720 LET P$=A$
```



```
9730 LET 0$="SPECTRUM": RETURN
9800 INPUT "PLAYER ONE NAME [ENT
ER TO SKIP]  "; LINE A$
9810 IF A$="" THEN GO TO 9830
9820 LET P$=A$
9830 INPUT "PLAYER TWO NAME [ENT
ER TO SKIP]  "; LINE A$
9840 IF A$="" THEN GO TO 9850
9850 LET 0$=A$
9860 RETURN
```

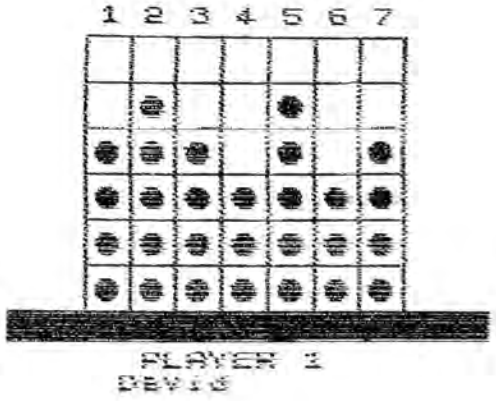
User Defined Graphics



Four in a Row

From the fertile mind of David Perry comes this version of FOUR IN A ROW. The program contains full instructions and has a hires load.

This is the program in action:



And here's the listing for it:

```
    2 FOR n=0 TO 7: READ a: POKE
USR "a"+n,a: NEXT n: GO SUB 5000
: DIM r(7,7)
    5 DATA 60,126,255,255,255,255
,126,60
   10 BORDER 0: PAPER 0: INK 7: B
RIGHT 1: CLS
   20 PRINT AT 3,7;"1 2 3 4 5 6 7
"
   40 FOR n=52 TO 166 STEP 16
   50 PLOT n,40
   60 DRAW 0,100
   80 NEXT n
```



```
   90 FOR n=44 TO 142 STEP 16
  100 PLOT 52,n: DRAW 110,0
  110 NEXT n
  120 FOR n=34 TO 44: PLOT 25,n:
DRAW 168,0: NEXT n
  130 LET play=1
  140 BRIGHT 0
  200 LET play=1: PRINT AT 18,8;
INK 7; PAPER 2; FLASH 1;" PLAYER
 1 ": PRINT AT 19,8;A$;"      "
  210 INPUT "MOVE PLAYER 1 ? ";mo
  220 IF mo>7 OR mo<1 THEN GO TO
210
  230 GO SUB 1000
  250 LET play=2: PRINT AT 18,8;
INK 7; PAPER 2; FLASH 1;" PLAYER
 2 ": PRINT AT 19,8;B$;"      "
  260 INPUT "MOVE PLAYER 2 ? ";mo
  270 IF mo>7 OR mo<1 THEN GO TO
260
  280 GO SUB 1000
  290 GO TO 200
1000 GO TO 1060
1060 FOR n=6 TO 1 STEP -1: IF r(
n,mo)=1 THEN NEXT n: RETURN
1070 LET r(n,mo)=1
1080 IF play=1 THEN PRINT AT 3+(
n*2),(mo*2)+5; INK 2; BRIGHT 1;"
●"
1081 IF play=2 THEN PRINT AT 3+(
n*2),(mo*2)+5; INK 5; BRIGHT 1;"
●"
1090 RETURN
5000 BORDER 1: CLS : PLOT 0,0: D
RAW 0,175: DRAW 255,0: DRAW 0,-1
75: DRAW -255,0
5010 PRINT AT 2,3; BRIGHT 1;"F O
 U R   I N   A   R O W ! !"
5020 PRINT AT 3,3;"
5030 PRINT AT 4,1;"In the game o
f "; FLASH 1; PAPER 2; INK 7;"FO
UR IN A ROW!"
5040 PRINT TAB 2;"you must try t
o defeat your ": PRINT TAB 1;"op
ponent by connecting four of": P
RINT TAB 1;"your counters in a r
ow either": PRINT TAB 1;"VERTICA
LLY,HORIZONTALLY or": PRINT TAB
1;"DIAGONALLY... eg."
```

```
5050 PLOT 0,0: DRAW 0,175
5060 PRINT AT 13,4;"■■■■■": PRINT
 AT 10,11;"■";AT 11,11;"■";AT 12
,11;"■";AT 13,11;"■"
5070 PRINT AT 10,15;"■";AT 11,16
;"■";AT 12,17;"■";AT 10,18;"■"
5080 PRINT AT 10,26;"■";AT 11,25
;"■";AT 12,24;"■";AT 13,23;"■"
5090 INPUT "Player 1's name ?";A
$: IF LEN A$>10 THEN GO TO 5090
5091 PRINT AT 15,1; INK 7; PAPER
 2; FLASH 1;"PLAYER 1:";A$
5100 INPUT "Player 2's name ?";b
$: IF LEN B$>10 THEN GO TO 5100
5101 PRINT AT 16,1; INK 7; PAPER
 2; FLASH 1;"PLAYER 2:";b$
5110 PRINT AT 18,1;"PLAYER 1=";
INK 2;"■"
5120 PRINT AT 19,1;"PLAYER 2=";
INK 5;"■"
5130 PRINT AT 20,1; FLASH 1; INK
 2;" PRESS ENTER TO COMMENCE!!!
": PAUSE 0: RETURN
```


Tanx

This is a two-person game, in which you have to try and shoot the turrent of your opponent's tank.

Here are the controls for player one:

Left - Z
Right - X
Up - 2nd row/left
Down - 3rd row/left
Fire - Top row/left

And these are the ones which player two should use:

Left - M
Right - Symbol Shift
Up - 2nd row/right
Down - 3rd row/right
Fire - Top row/right

You'll notice that the tanks disappear for a moment when a shell is fired, but they can still be hit, and can still move out of the way. You must not run over any of the mines dropped by the shells. You have three lives each and the first to die is, naturally enough, the loser. TANX was written by Neal Cavalier-Smith.

```
  10 REM TANX
  15 REM Neal Cavalier-Smith
  20 LET x1=INT (RND*20): LET y1
=5: LET x2=INT (RND*20): LET y2=
25
```

```
  30 LET y1b=1: LET x1b=0: LET y
2b=-1: LET x2b=0
  40 LET cc=0: LET c1=0: LET life
e2=3: LET life1=3
  50 GO SUB 1000
  60 GO SUB 100
  70 GO TO 60
 100 REM Movement
 101 PRINT AT x1,y1;" "
 102 IF x1b=0 THEN PRINT AT x1,y
1+y1b;" "
 103 IF y1b=0 THEN PRINT AT x1+x
1b,y1;" "
 110 IF IN 65022<255 AND x1<20 T
HEN LET x1=x1+1: LET x1b=1: LET
y1b=0
 120 IF IN 64510<255 AND x1>0 TH
EN LET x1=x1-1: LET x1b=-1: LET
y1b=0
 130 IF IN 65278=253 AND y1>0 TH
EN LET y1=y1-1: LET y1b=-1: LET
x1b=0
 140 IF IN 65278=251 AND y1<30 T
HEN LET y1=y1+1: LET y1b=1: LET
x1b=0
 150 PRINT AT x2,y2;" "
 152 IF x2b=0 THEN PRINT AT x2,y
2+y2b;" "
 153 IF y2b=0 THEN PRINT AT x2+x
2b,y2;" "
 160 IF IN 49150<255 AND x2<20 T
HEN LET x2=x2+1: LET x2b=1: LET
y2b=0
 170 IF IN 57342<255 AND x2>0 TH
EN LET x2=x2-1: LET x2b=-1: LET
y2b=0
 180 IF IN 32766=251 AND y2>0 TH
EN LET y2=y2-1: LET y2b=-1: LET
x2b=0
 190 IF IN 32766=253 AND y2<30 T
HEN LET y2=y2+1: LET y2b=1: LET
x2b=0
 200 REM Press Shift
 201 IF IN 61438<255 THEN LET cc
=3
 202 IF cc>0 THEN GO SUB 300
 210 IF SCREEN$ (x2+x2b,y2+y2b)=
"." THEN GO TO 2000: PRINT AT x1
,y1;CHR$ 144
 211 PRINT AT x1,y1;CHR$ 144
 212 IF IN 63486<255 THEN LET c1
=3
```



```
 213 IF c1>0 THEN GO SUB 400
 214 IF SCREEN$ (x1+x1b,y1+y1b)=
"." THEN GO TO 2200: PRINT AT x2
,y2;CHR$ 144
 220 IF x1b=0 THEN PRINT AT x1,y
1+y1b;CHR$ 146
 230 IF y1b=0 THEN PRINT AT x1+x
1b,y1;CHR$ 145
 235 IF SCREEN$ (x1+x1b,y1+y1b)=
"." THEN GO TO 2200: PRINT AT x1
,y1;CHR$ 144
 240 PRINT AT x2,y2;CHR$ 144
 250 IF x2b=0 THEN PRINT AT x2,y
2+y2b;CHR$ 146
 260 IF y2b=0 THEN PRINT AT x2+x
2b,y2;CHR$ 145
 270 RETURN
 300 REM Shell
 310 IF cc=3 THEN LET x2s=x2+x2b
: LET y2s=y2+y2b
 311 FOR c=0 TO 9
 312 IF c=0 THEN GO TO 335
 315 IF y2s<1 THEN LET y2b=1: LE
T x2b=INT (RND*3)-1
 316 IF y2s>29 THEN LET y2b=-1:
LET x2b=INT (RND*3)-1
 317 IF x2s>19 THEN LET x2b=-1:
LET y2b=INT (RND*3)-1
 318 IF x2s>18 THEN LET x2b=-1:
LET y2b=INT (RND*3)-1
 319 IF x2s<1 THEN LET x2b=1: LE
T y2b=INT (RND*3)-1
 320 IF x=x1 AND y=y1 OR x=x1+x1
b AND y=y1+y1b THEN GO SUB 2200
 330 PRINT AT x,y;" "
 335 PRINT AT x2s,y2s;"."
 336 LET x=x2s: LET y=y2s
 340 LET x2s=x2s+x2b
 350 LET y2s=y2s+y2b
 360 NEXT c
 370 LET cc=cc-1
 380 RETURN
 400 REM Shell1
 410 IF c1=3 THEN LET x1s=x1+x1b
: LET y1s=y1+y1b
 411 FOR d=0 TO 9
 412 IF d=0 THEN GO TO 435
 415 IF y1s<1 THEN LET y1b=1: LE
T x1b=INT (RND*3)-1
 416 IF y1s>29 THEN LET y1b=-1:
LET x1b=INT (RND*3)-1
```

```
  417 IF x1s>19 THEN LET x1b=-1:
LET y1b=INT (RND*3)-1
  418 IF x1s>18 THEN LET x1b=-1:
LET y1b=INT (RND*3)-1
  419 IF x1s<1 THEN LET x1b=1: LE
T y1b=INT (RND*3)-1
  420 IF x=x2 AND y=y2 OR x=x2+x2
b AND y=y2+y2b THEN GO SUB 2000
  430 PRINT AT x,y;" "
  435 PRINT AT x1s,y1s;"."
  436 LET x=x1s: LET y=y1s
  440 LET x1s=x1s+x1b
  450 LET y1s=y1s+y1b
  460 NEXT d
  470 LET c1=c1-1
  480 RETURN
  999 STOP
 1000 REM Define Graphics
 1010 FOR z=144 TO 146
 1020 FOR x=0 TO 7
 1030 READ a: POKE USR (CHR$ z)+x
,a
 1040 NEXT x
 1050 NEXT z
 1060 DATA 60,126,255,255,255,255
,126,60,24,24,24,24,24,24,24,24,
0,0,0,255,255,0,0,0
 1090 RETURN
 2000 PRINT AT x2,y2; FLASH 1;"S
t"
 2003 FOR z=5 TO 100: NEXT z
 2005 PRINT AT x2,y2; FLASH 0;"
 "
 2010 LET life2=life2-1
 2020 IF life2=0 THEN PRINT INVER
SE 1; FLASH 1;"G A M E  O V E R
                Player 1 Wins
               ": STOP
 2030 GO TO 100
 2200 PRINT AT x1,y1; FLASH 1;"S
t"
 2205 FOR x=5 TO 100: NEXT x
 2206 PRINT AT x1,y1; FLASH 0;"
 "
 2210 LET life1=life1-1
 2220 IF life1=0 THEN PRINT INVER
SE 1; FLASH 1;"G A M E  O V E R
                Player 2 Wins
               ": STOP
 2222 GO TO 100
```


MACHINE CODE UTILITIES

Memory Monitor

This program by David Perry allows you to examine the contents of ROM and RAM. It has good hex/dec and dec/hex routines, and allows you to move, save and load memory.

```
THIS PROGRAMME WILL REVEAL THE
COTENTS OF YOUR ROM & RAM IN
THE FORM OF ADDRESS, ADDRESS IN
HEXADECIMAL, DECIMAL VALUE AT
THAT ADDRESS,HEXADECIMAL VALUE
AT THAT ADDRESS & THE CHARACTER
CORRESPONDING TO THE VALUE AT
THE ADDRESS.
THE PROGRAMME WILL ALSO ALLOW
YOU TO FREELY MOVE MEMORY AROUND
,CONVERT HEXADECIMAL TO DECIMAL
AND VISA VERSA AND ALSO TO RUN
A ROUTINE FROM ANY ADDRESS
```

```
ADDR+HEX:  DEC:  HEXPEEK:  CHR$:
    0 0000   243   00F3    NEXT
    1 0001   175   00AF    CODE
    2 0002   17    0011    ?
    3 0003   255   00FF    COPY
    4 0004   255   00FF    COPY
    5 0005   195   00C3    NOT
    6 0006   203   00CB    THEN
    7 0007   17    0011    ?
    8 0008   42    002A    *
    9 0009   93    005D    ]
   10 000A   92    005C    \
   11 000B   34    0022    "
   12 000C   95    005F    _
   13 000D   92    005C    \
   14 000E   24    0018    ?
   15 000F   67    0043    C
   16 0010   195   00C3    NOT
   17 0011   242   00F2    PAUSE
  A:NEW ADDRESS   H:HEX-DEC
  E:EXECUTE CODE  D:DEC-HEX
   S-SAVE / L-LOAD BYTES.
  M-MOVE MEMORY  C-CONTINUE
```

```
  1 REM
  2 REM   MEMORY MONITOR
  3 REM   © DAVID PERRY!
  4 REM        1983
  5 REM
  6 REM
  7 REM
 20 BRIGHT 0: GO SUB 720: INPUT
"START ADDRESS ? ";ADD
 30 GO TO 330
 40 REM HEX-DEC
 50 PRINT AT 20,0;"
                   "
 60 PRINT AT 20,0;
 70 LET A=0: LET E=0
 80 INPUT "Hex. Characters ? ";
A$: LET 0$="0"
 90 LET 0=0
100 IF LEN A$>3 THEN GO TO 120
110 LET A$=0$+A$: GO TO 100
120 LET W=4096: LET A=E
130 FOR R=1 TO 4
140 IF FN P(A$(R))<E OR FN P(A$
(R))>15 THEN GO TO 100
150 LET A=A+W*FN P(A$(R)): LET
W=W/16
160 NEXT R
170 PRINT "      Hexadecimal:";A
$;" "
180 PRINT "        Decimal:";A;"
"
190 PAUSE 0: RETURN
200 REM DEC-HEX
210 PRINT AT 20,0;"
                   "
220 PRINT AT 20,0;
230 INPUT "Decimal Number ? ";A
240 IF A>65535 OR A<0 THEN GO T
O 230
250 PRINT AT 20,6;" Decimal:";A
;" ";: PRINT AT 21,5;" Hexadecim
al:";
260 LET U=4096: LET Y=A
270 LET T=INT (Y/U)
280 PRINT PAPER 7; BRIGHT 1;CHR
$ (48+T+7*(T>9));
290 LET Y=Y-U*T
295 LET U=U/16
300 IF U>=1 THEN GO TO 270
310 RETURN
```



```
320 REM MAIN LAYOUT
330 PRINT AT 0,0; PAPER 5; INK
0;" ADDR+HEX: DEC: HEXPEEK: CHR$
: "
340 LET GO=ADD: LET TO=ADD+17:
FOR L=GO TO TO
350 LET DEC=PEEK ADD: LET C$=CH
R$ (DEC AND DEC>32)
360 PRINT TAB (1 AND ADD<10)+(1
AND ADD<100)+(1 AND ADD<1000)+(
1 AND ADD<10000); INK 7; PAPER 4
;ADD; PAPER 7;" ";
370 LET A=ADD: GO SUB 260
380 PRINT "  "; PAPER 5;DEC; PA
PER 7;TAB 17;: LET A=DEC: GO SUB
260: PRINT PAPER 7;TAB 24; PAPE
R 6;C$
390 LET ADD=ADD+1: IF ADD>65535
THEN LET ADD=0
400 NEXT L
410 PRINT AT 19,2; PAPER 2; INK
7;" A:NEW ADDRESS  H:HEX-DEC "
420 PRINT AT 20,2; PAPER 2; INK
6;" E:EXECUTE CODE D:DEC-HEX ";
430 PRINT AT 21,0; INK 7;"  ";
PAPER 2;" S-SAVE / L-LOAD BYTES
 ";" PAPER 7;"  "
440 PRINT #1;"   M-MOVE MEMORY
C-CONTINUE "
450 LET I$=INKEY$: IF I$="" THE
N GO TO 450
460 IF I$="A" OR I$="a" THEN RU
N
470 IF I$="h" OR I$="H" THEN GO
SUB 40: GO TO 410
480 IF I$="S" OR I$="s" THEN GO
SUB 920
490 IF I$="d" OR I$="D" THEN GO
SUB 200: PAUSE 0 : GO TO 410
500 IF I$="e" OR I$="E" THEN GO
TO 570
510 IF I$="L" OR I$="l" THEN GO
SUB 880: GO TO 410
520 IF I$="C" OR I$="c" THEN CL
S : GO TO 330
530 IF I$="M" OR I$="M" THEN GO
TO 600
540 IF I$<>"S" OR I$<>"s" OR I$
<>"L" OR I$<>"l" OR i$<>"h" OR i
$<>"H" OR i$<>"d" OR i$<>"D" OR
i$<>"e" OR i$<>"E" OR i$="c" OR
i$<>"C" OR i$<>"M" OR i$<>"M" TH
EN GO TO 450
```

```
 550 GO TO 410
 560 REM RUN ROUTINE
 570 INPUT "Execute Address ? ";
EX
 580 RANDOMIZE USR EX: PAUSE 0:
RUN
 590 REM MOVE MEMORY
 600 INPUT "Move From Address ?
";MF
 610 IF MF>65535 OR MF<0 THEN GO
 TO 600
 620 INPUT "Move To Address ? ";
MT
 630 IF MT>65535 OR MT<0 THEN GO
 TO 620
 640 INPUT "Number of bytes to m
ove ";N
 650 IF MT+N>65535 THEN PRINT #1
;"   TOO MANY BYTES TRY AGAIN ":
 PAUSE 0: GO TO 600
 660 LET ADD=MT
 670 FOR P=1 TO N
 680 POKE MT,PEEK MF
 690 LET MT=MT+1: LET MF=MF+1: N
EXT P
 700 CLS
 710 GO TO 330
 715 REM INSTRUCTIONS
 720 DEF FN P(A$)=(CODE A$-48-7*
(A$>"9")-32*(A$>"Z"))
 730 BORDER 7: PAPER 7: INK 0: C
LS : PRINT AT 1,6;"▄▄▄▄▄▄▄▄▄▄▄▄▄▄
▄▄▄▄▄"
 740 PRINT AT 2,6;"▌ MEMORY MONI
TOR ▐"
 750 PRINT AT 3,6;"▌ © DAVID PER
RY!  ▐"
 760 PRINT AT 4,6;"▌         1983
 ▐"
 770 PRINT AT 5,6;"▄▄▄▄▄▄▄▄▄▄▄▄▄▄
▄▄▄▄▄"
 780 PRINT AT 8,0;"THIS PROGRAMM
E WILL REVEAL THE": PRINT "COTEN
TS OF YOUR ROM & RAM IN"
 790 PRINT "THE FORM OF ADDRESS,
 ADDRESS IN": PRINT "HEXADECIMAL
, DECIMAL VALUE AT"
 800 PRINT "THAT ADDRESS,HEXADEC
IMAL VALUE": PRINT "AT THAT ADDR
ESS & THE CHARACTER": PRINT "COR
RESPONDING TO THE VALUE AT"
 810 PRINT "THE ADDRESS."
```



```
 820 PRINT "THE PROGRAMME WILL A
LSO ALLOW": PRINT "YOU TO FREELY
 MOVE MEMORY AROUND"
 830 PRINT ",CONVERT HEXADECIMAL
 TO DECIMAL": PRINT "AND VISA VE
RSA AND ALSO TO RUN": PRINT "A R
OUTINE FROM ANY ADDRESS"
 840 REM TAPE LOADING ROUTINE
 850 INPUT "DO YOU WANT TO LOAD
A PROGRAMME FROM TAPE (Y OR N) ?
 ";T$
 860 IF T$="Y" OR T$="y" THEN GO
 TO 880
 870 CLS : RETURN
 880 INPUT "Address to load byte
s to?";t
 890 IF t>65535 OR t<0 THEN GO T
O 880
 900 PRINT FLASH 1; PAPER 2; INK
 7;AT 21,0;" PRESS PLAY ON TAPE-
RECORDER! "
 910 LOAD ""CODE t: CLS : RETURN

 920 REM SAVE MEMORY
 930 INPUT "SAVE FROM ADDRESS ?
";SF
 940 IF SF<0 OR SF>65535 THEN GO
 TO 930
 950 INPUT "NUMBER OF BYTES ? ";
N
 960 IF N<1 OR N>65535 THEN GO T
O 950
 970 INPUT "SAVE NAME ? ";S$
 980 IF S$="" OR LEN S$>10 THEN
GO TO 970
 990 SAVE S$CODE SF,N
1000 RETURN
```

MC Screensave

This routine, from David Perry, is written for the 48K machine. It allows you to store and print up screens instantly. It can be used in any program.

```
  10 BORDER 0: PAPER 0: INK 7: C
LS
  15 RESTORE
  16 FOR n=40000 TO 40028
  20 READ a: POKE n,a
  30 NEXT n
  40 DATA 33,0,64,17,46,117,1,0,
27,237,176,201,33,46,117,17,0,64
,1,0,27,237,176,201,195,144,234,
0,0
  50 PRINT AT 0,0;"LOAD SCREEN F
ROM TAPE AND I WILL": PRINT "SAV
E IT INTO MEMORY FOR INSTANT": P
RINT "RECALL AT ANY TIME!!!"
  60 PRINT : PRINT "TO SAVE SCRE
EN TYPE"
  70 PRINT "RANDOMIZE USR 40000"
  80 PRINT : PRINT "TO PRINT UP
A SCREEN TYPE"
  90 PRINT "RANDOMIZE USR 40012"
 100 PRINT : PRINT : PRINT "HERE
 IS AN EXAMPLE."
 120 PRINT FLASH 1;"PRESS PLAY O
N TAPE"
 130 LOAD ""SCREEN$ : RANDOMIZE
USR 40000
 140 PAUSE 0: CLS : PRINT AT 0,0
;" PRESS A KEY TO PRINT UP SCREE
N": PAUSE 0: RANDOMIZE USR 40012
: GO TO 140
```


BASIC Screensave

This is a BASIC version of the preceding program, also intended for the 48K computer. It was also written by David Perry.

```
   1 CLS : PRINT "THIS PROGRAMME
 WILL STORE THE": PRINT "SCREEN
IN MEMORY AND THEN PRINT IT BACK
. AS AN EXAMPLE I WILL": PRINT "
LIST THIS PROGRAMME,SAVE IT AND
THEN PRINT IT BACK ONTO THE": PR
INT "SCREEN."
   2 PRINT : PRINT "THE PROGRAMM
E IS TOTALLY BASIC": PRINT "AND
CAN BE USED FOR MANY": PRINT "DI
FFERENT PURPOSES."
   3 PRINT : PRINT "PRESS ANY KE
Y TO START": PAUSE 0: CLS
  10 REM
  20 REM    BASIC SCREENSAVE
  30 REM    (c) DAVID PERRY
  40 REM
  45 CLS : LIST 50
  50 LET A=30000: REM ADDRESS TO
 SAVE TO
  60 FOR X=0 TO 21
  70 FOR Y=0 TO 31
  80 LET A$=SCREEN$ (X,Y)
  85 PRINT AT X,Y;" "
  90 POKE A,CODE A$
 100 LET A=A+1
 110 NEXT Y
 120 NEXT X
 130 REM
 140 REM ****PRINT UP****
 150 REM
 160 LET A=30000: REM ADDRESS TO
 LOAD FROM
 170 FOR X=0 TO 21
 180 FOR Y=0 TO 31
 190 LET B=PEEK A
 200 PRINT AT X,Y;CHR$ B
 210 LET A=A+1
 220 NEXT Y
 230 NEXT X
```

Ireland

To show your computer's graphics off at their best, run this program by David Perry. The awesome task of entering all that data will be rewarded by the quality of the final picture. You can incorporate this program with the SCREENSAVE one (either BASIC or machine code) so you can get Ireland up on your screen instantly.

```
  10 BORDER 1: PAPER 1: INK 7: B
RIGHT 8: CLS
  11 CLS
  21 PLOT 123,135
  31 FOR f=1 TO 166
  41 READ a,b,c
  51 DRAW a,b,c
  61 NEXT f
  57 PLOT 109,129
  71 FOR f=1 TO 7
  81 READ a,b,c
  91 DRAW a,b,c
 101 NEXT f
 105 PLOT 55,60
 110 FOR f=1 TO 7
 120 READ a,b,c
 130 DRAW a,b,c
 140 NEXT f
 145 PLOT 36,52
 150 FOR f=1 TO 6
 160 READ a,b,c
 170 DRAW a,b,c
 180 NEXT f
 185 PLOT 68,87
 190 FOR f=1 TO 5
 200 READ a,b,c
 210 DRAW a,b,c
 220 NEXT f
 225 PLOT 32,90
 230 FOR f=1 TO 6
 240 READ a,b,c
 250 DRAW a,b,c
 260 NEXT f
```



```
 265 PLOT 36,93: DRAW 0,2,0: DRA
W -1,2,0: DRAW -2,-2,0: DRAW -2,
-3,0: DRAW 4,1,0
 270 PLOT 80,125: DRAW 0,3,-PI/2
: DRAW -2,2,0: DRAW -3,1,0: DRAW
 -3,-2,PI/3: DRAW 6,-3,-PI/3
 290 PLOT 85,115: DRAW 3,-1,0: D
RAW -3,4,PI/2: DRAW -1,3,PI/2: D
RAW -3,0,-PI: DRAW 0,-2,0: DRAW
2,-4,-PI/2
1000 DATA 3,5,0,-2,3,-PI/2,-4,6,
0,-1,5,0,-7,7,PI/2,-6,0,PI/2,-3,
-2,-PI
1010 DATA -5,0,0,-4,-5,0,-3,-1,0
,1,4,0,7,5,0,-5,2,PI/4,-3,5,0,-3
,-2,-PI,-2,-6,(PI/1.5)
1020 DATA -1,-4,0,-2,0,0,0,1,0,2
,3,0,-6,7,0,-3,-2,0,-4,-2,PI/1.5
1030 DATA -6,-2,0,-1,-5,PI/2,-2,
-7,PI/2,-3,-3,PI/1.5,-7,-3,PI/2,
3,-5,0,11,-1,0,-2,-3,0,-10,-5,0,
0,-2,0,3,-2,0,-1,-2,0,-7,1,PI/4
1040 DATA -3,2,0,-3,-6,PI/4,-1,5
,0,-5,2,PI/2,-6,2,0,-1,-4,PI,-3,
-10,PI/4

1050 DATA -2,-5,-PI
1060 DATA 3,-2,PI,4,-1,0,0,-2,PI
,-7,-1,0,-1,-6,0,-6,-2,0,1,-4,PI
,2,-4,PI,4,0,PI/1.5,5,-2,PI*1.2
1070 DATA 2,-1,PI/2,5,-5,PI/1.5,
9,1,0,-1,-3,0,-6,-1,0,-4,-7,PI/6
1080 DATA 0,-2,PI,-1,-5,0,-7,-5,
PI/6,-4,-3,0,1,-2,PI,7,2,0,6,0,P
I/4,1,-1,0,-9,0,0,-5,-5,-PI/3
1090 DATA -1,1,0,-1,-3,0,1,-2,0,
2,-4,PI/4,-4,-1,0,-5,2,PI,-4,0,P
I/4,-4,-2,0,0,-5,PI/2,7,0,-PI/4,
7,1,0,1,-1,0,-3,-2,PI/10,-9,-2,0
,-2,-6,PI/4,3,-2,PI
1100 DATA 2,0,0,3,-2,PI/1.5,4,1,
PI/4,6,2,0,-8,-4,0,-5,-5,0,1,-3,
PI,9,3,0,2,1,0,3,0,0,-7,-2,0,2,-
3,PI,5,2,0,2,-3,0,4,-2,-PI/6,4,0
,0,2,2,0,4,-1,0,3,2,0,5,1,PI/1.5
1110 DATA 4,2,PI/2,5,3,0,0,3,PI,
0,3,PI/1.5,4,0,0,1,-5,PI/4,2,0,0
,1,2,0,4,2,0,0,2,0,1,1,0,4,0,0,3
,2,0,-1,3,PI/2,4,2,0,5,1,PI/4,4,
1,PI/1.3,1,1,0,4,0,PI,4,-1,0,2,0
,0,7,0,PI/4
```

```
1120 DATA -1,3,PI/5,-2,3,PI/4,-3
,1,0,4,0,0,3,5,0,2,7,PI/4,2,6,PI
/6,3,6,0,-1,3,PI/2,0,11,PI/6
1130 DATA -3,3,PI/4,-2,0,0,3,1,0
,2,-1,0,0,3,PI,-1,3,0,0,5,PI/1.7
,-2,5,0,0,3,PI/3,0,3,PI/3,-1,0,0
,0,3,-PI,3,3,PI/4,4,2,PI/6,3,5,0
,0,1,0
1140 DATA 4,0,0,2,2,PI/3,-1,4,PI
,0,7,PI/6,2,1,PI,2,-9,0,2,0,PI,-
1,10,PI/6,-2,3,PI/2,-2,1,0
1150 DATA -5,-3,PI/4,0,2,0
1160 DATA 1,3,PI/6,2,5,PI/4,-3,0
,PI/1.5,-3,1,-PI/1.5,-3,-6,-PI/3
,1,-1,0,4,-1,PI/5
1170 DATA 3,4,0,5,6,-PI/4,-1,1,0
,-2,-1,0,-2,-3,-PI/3,-3,-4,-PI/3
,0,-3,-PI/2
1180 DATA 3,2,0,2,-1,0,5,0,0,-3,
1,PI/6,-4,3,-PI/1.75,-3,-4,PI/3
1190 DATA 2,3,0,-2,5,PI/2,0,-2,0
,0,-3,PI/2,0,-3,PI/2
1200 DATA 4,-3,PI/6,4,-4,PI/6,1,
0,0,0,3,0,-4,5,PI/6,-5,-1,0
```


Master Copier

This handy program from David Perry will copy machine code programs automatically.

If there is a BASIC loader, then you must copy it first, by typing MERGE "" then SAVE"name" LINE the line which loads the machine code part. This program reads the header, loads the program and then tells you how to save it. Just follow the prompts and you'll have no trouble using this program.

```
    5 GO SUB 31
    6 CLEAR 32747
    7 FOR i=32748 TO 32761
    8 READ x: POKE i,x
    9 DATA 55,62,0,221,33,0,125,1
7,20,0,205,86,5,201
   10 NEXT i
   11 PRINT AT 5,2; FLASH 1;"I AM
 NOW LOADING THE HEADER!"; FLASH
 0;AT 7,10;"PLEASE WAIT"
   12 RANDOMIZE USR 32748
   13 LET u=1: GO SUB 19: LET u=0
   14 CLS : PRINT AT 0,10;"Header
 Data"
   15 IF PEEK 32000=0 THEN GO TO
64
   16 IF PEEK 32000=1 THEN GO TO
69
   17 IF PEEK 32000=2 THEN GO TO
73
   18 IF PEEK 32000<>3 THEN RUN
   19 LET z$="": FOR b=32001 TO 3
2010
   20 LET z$=z$+CHR$ (PEEK b)
   21 NEXT b
   22 LET o$=z$
   23 LET A=PEEK 32011: LET B=PEE
K 32012
   24 LET N=A+(B*256)
   25 LET A=PEEK 32013: LET B=PEE
K 32014
```

```
  26 LET M=A+(B*256)
  27 LET A=PEEK 32015: LET B=PEE
K 32016
  28 LET O=A+(B*256)
  29 IF u=1 THEN RETURN
  30 GO TO 47
  31 BRIGHT 1: BORDER 0: PAPER 0
: INK 7: CLS : PRINT "          MA
STER COPIER"
  32 PRINT "          ==========
="
  33 PRINT
  34 PRINT "THIS PROGRAMME WILL
COPY MACHINE"
  35 PRINT "CODE PROGRAMMES AUTO
MATICALLY."
  36 PRINT
  37 PRINT "IF THERE IS A BASIC
LOADER THEN YOU MUST COPY IT FIR
ST"
  38 PRINT "BY TYPING MERGE""""
THEN"
  39 PRINT "SAVE""name"" LINE th
e line which  loads the machie c
ode part"

  40 PRINT : PRINT "THIS PROGRAM
ME READS THE HEADER"
  41 PRINT "LOADS THE PROGRAMME
AND THEN"
  42 PRINT "TELLS YOU HOW TO SAV
E IT."
  43 PRINT : PRINT "1....INSERT
CASSETTE AT START OFMACHINE CODE
 SECTION"
  44 PRINT "2....CHECK LEADS"
  45 PRINT "3....PRESS ANY KEY W
HEN READY"
  46 PAUSE 0: CLS : RETURN
  47 CLS : PRINT AT 1,6;"MASTER
COPIER"
  48 PRINT AT 2,6;"==========
"
  49 PRINT AT 3,0;"To save the p
rogramme now on": PRINT "cassett
e you must first of all"
  50 PRINT "have saved the basic
 loader and": PRINT "you must re
wind the cassette to": PRINT "th
e start of the MACHINE CODE"
```



```
  51 PRINT "section then 'NEW' t
his": PRINT "programme and type"
  52 PRINT : PRINT "LOAD """";z$;
""""CODE ";m;",";n
  53 PRINT "CHECK LEADS,PRESS A
KEY,PRESS": PRINT "PLAY ON THE T
APE RECORDER..."
  54 PRINT : PRINT "When it has
loaded type:"
  55 PRINT : PRINT "SAVE ";z$;"
CODE ";m;",";n
  56 PRINT "INSERT BLANK CASSETT
E,CHECK": PRINT "LEADS,PRESS REC
ORD & PRESS A KEY"
  57 INPUT "Copy to the printer
(y/n)";a$
  58 IF a$="y" THEN COPY
  59 IF a$="n" THEN PRINT AT 21,
0; BRIGHT 8;"YOU SHOULD WRITE TH
IS DOWN THEN!"
  60 INPUT "Are there any more m
achine code sections to copy (y/
n)";a$
  61 IF a$="y" THEN RUN
  62 PRINT AT 21,2; FLASH 1; PAP
ER 2;"PRESS ANY KEY TO 'NEW'!!!!
"
  63 PAUSE 0: PAUSE 0: NEW
  64 CLS : PRINT "You have just
loaded a BASIC": PRINT "header"
  65 PRINT "IT IS CALLED"
  66 PRINT z$;" LINE ";m
  67 PRINT "ITS BASIC LENGTH IS
";0
  68 PRINT : PRINT "Press any ke
y to run programme": PRINT "agai
n...": PAUSE 0: RUN
  69 CLS : PRINT "You have just
loaded a NUMERIC": PRINT " ARRAY
 header"
  70 PRINT "IT IS CALLED"
  71 PRINT z$
  72 PRINT : PRINT "Press any ke
y to run programme": PRINT "agai
n...": PAUSE 0: RUN
  73 CLS : PRINT "You have just
loaded a CRARACTER": PRINT " ARR
AY header"
  74 PRINT "IT IS CALLED"
  75 PRINT z$
```

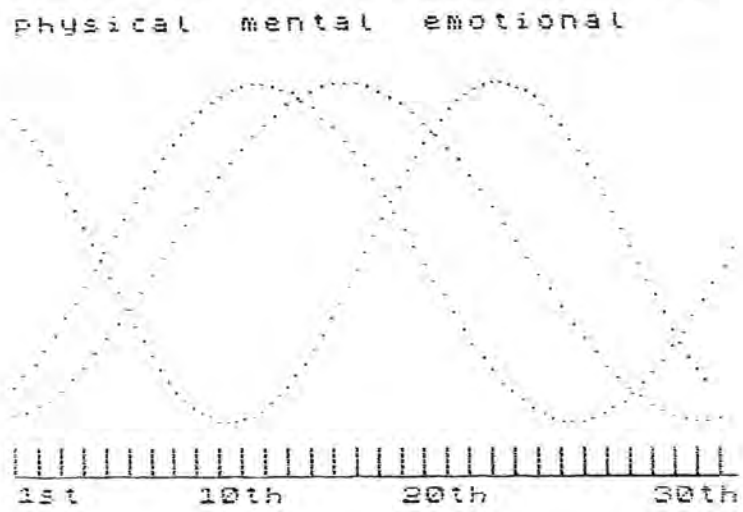
```
  76 PRINT : PRINT "Press any ke
y to run programme": PRINT "agai
n...": PAUSE 0: RUN
        MASTER COPIER
        ==============

THIS PROGRAMME WILL COPY MACHINE
CODE PROGRAMMES AUTOMATICALLY.

IF THERE IS A BASIC LOADER THEN
YOU MUST COPY IT FIRST
BY TYPING MERGE"" THEN
SAVE"name" LINE the line which
loads the machie code part

THIS PROGRAMME READS THE HEADER
LOADS THE PROGRAMME AND THEN
TELLS YOU HOW TO SAVE IT.

1....INSERT CASSETTE AT START OF
MACHINE CODE SECTION
2....CHECK LEADS
3....PRESS ANY KEY WHEN READY
```


UTILITIES
AND
DEMONSTRATIONS

Biorhythms

Now you can chart your ups and downs in this graphical BIORHYTHMS program.

There are three cycles, which begin at birth and continue right through our lives, which are believed by many to have



an effect on our lives. The three cycles are as follows:

PHYSICAL - this is a 23-day cycle which controls such things as endurance and strength

EMOTIONAL - the 28-day emotional cycle governs feelings of optimism and pessimism

MENTAL - Logic and reasoning come under the sway of this 33-day cycle

The program is self-prompting and produces a very effective display.

```
   1 RESTORE
   2 PRINT AT 0,0;"                BIO
-RHYTHM
  10 INPUT "Enter Date Of Birth"
;"Day  ";a;"  Month ";b;"   Year
  ";c
  20 INPUT "Enter Date Now   ",,"
Month  ";d;"    Year ";e
  25 CLS
  30 LET t=INT (((e-c)*365.25)+(
(d-b)*30.35)-a)
 600 FOR r=0 TO 255
 610 PLOT r,10
 615 IF r=INT (r/b)*b THEN FOR u
=10 TO 20: PLOT r,u: NEXT u
 620 NEXT r
 630 PRINT AT 21,0;"1st      10th
     20th        30th"
 640 PRINT AT 0,0; INK 1;"physic
al  "; INK 2;"mental  "; INK 4;"
emotional"
 900 FOR r=1 TO 3
 905 READ u
 910 LET l=2*PI*(t-(INT (t/u)*u)
)/u
 920 LET k=2*PI*(33-u)*.03
1000 FOR a=l TO k+l+(2*PI) STEP
.1
1010 PLOT INK ((1 AND u=23)+(2 A
ND u=28)+(4 AND u=33));(a-l)*(35
-28+u),90+SIN a*60
1020 NEXT a
1030 NEXT r
1040 DATA 23,28,33
1050 INPUT "Another Go ? ";a$: I
F a$(1)="y" THEN GO TO 1
```

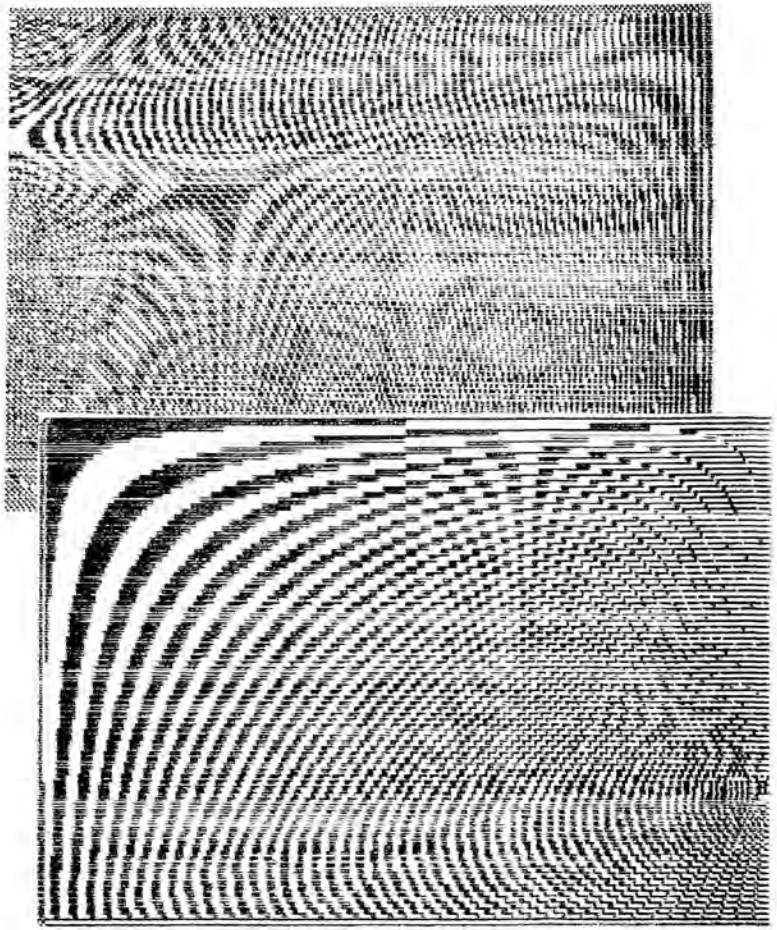

Paint Pot

This clever program from Graham Charlton fills a randomly generated shape with the current INK color. Lines 10 to 60 draw the shape, and the subroutine from 9000 onwards fills it. Make the strings (A$ and B$) as long as possible on your computer. Note that they take up a lot of memory.

```
  10 CIRCLE 120,80,15
  20 PLOT 100,60
  30 DRAW 40,0
  40 DRAW 0,40
  50 DRAW -40,0
  60 DRAW 0,-40
  70 GO SUB 9000
 100 STOP
9000 DIM a$(9999)
9010 DIM b$(9999)
9020 LET z=1
9030 INPUT x
9040 INPUT y
9050 PLOT x,y
9060 IF POINT (x+1,y)+POINT (x-1
,y)+POINT (x,y+1)+POINT (x,y-1)<
>4 THEN GO TO 9120
9070 LET z=z-1
9080 IF z=0 THEN RETURN
9090 LET x=CODE a$(z)
9100 LET y=CODE b$(z)
9110 GO TO 9050
9120 LET a$(z)=CHR$ x
9130 LET b$(z)=CHR$ y
9140 LET z=z+1
9150 FOR c=-1 TO 1
9160 FOR d=-(c=0) TO (c=0) STEP
1
9170 IF POINT (x+c,y+d)=0 THEN G
O TO 9200
9180 NEXT d
9190 NEXT c
9200 LET x=x+c
9210 LET y=y+d
9220 GO TO 9050
```

Timothy Leary

This program by Damian Steele creates a number of dramatic graphic displays to demonstrate convincingly (if you still needed convincing) how effective your computer's graphics can be.




```
  10 PAPER 6: BORDER 1: INK 1: L
ET D=1: LET E=1: LET F=0: LET X=
0: LET Y=175: LET S=D: LET B=0:
LET C=0: OVER 1
  20 CLS : PRINT #0;AT 0,0;"
          ZX TRIP                P
RESS ANY KEY IF BORED..."
 100 FOR A=X TO Y STEP S
 110 IF A>=128 THEN GO TO 400
 120 IF A>=87 THEN GO SUB 200
 130 PLOT 0,A: DRAW (Y*1.457)-B,
A-C
 135 IF INKEY$<>"" THEN GO TO 10
00
 140 NEXT A
 200 LET B=B+5.1: LET C=C+2
 300 RETURN
 400 LET S=2
 450 GO SUB 700
 500 OVER 1
 510 FOR A=1 TO 255 STEP S
 515 IF INKEY$<>"" THEN GO TO 10
00
 520 PLOT A,0: DRAW A,175
 530 NEXT A
 535 IF B=1 THEN GO TO 600
 537 GO SUB 700
 540 FOR A=1 TO 175 STEP S
 545 IF INKEY$<>"" THEN GO TO 10
00
 550 PLOT 0,A: DRAW 255,0
 560 NEXT A
 565 GO SUB 700
 570 IF B<>1 THEN LET B=1: GO TO
 450
 600 IF E=1 THEN LET E=0: GO TO
537
 620 LET F=F+1: IF F<=1 THEN LET
 S=1.4: LET B=0: GO TO 410
 650 GO SUB 700: GO TO 1000
 700 FOR A=1 TO 10: BEEP .007,(A
ND*30)+35: NEXT A
 800 RETURN
1000 PAUSE 0: CLS
1001 LOAD ""
9500 STOP
9990 PRINT FLASH 1;AT 10,10;"STO
P THE TAPE": PRINT AT 21,2;"PRES
S ANY KEY TO CONTINUE...": PAUSE
 0: GO TO 1
```

Color Test

This simple program by David Perry puts all of the computer's colors up on the screen so you can tune in your television for the best possible picture.

```
  10 BORDER 2: PAPER 0: BRIGHT 1
: INVERSE 0: OVER 0: FLASH 0: CL
S
  20 LET i=-1: FOR n=0 TO 31 STE
P 4
  30 LET i=i+1
  40 FOR y=0 TO 3
  50 FOR x=0 TO 21
  60 PRINT AT x,n+y; INK i;"■"
  70 NEXT x: NEXT y: NEXT n
  80 INK 7
```


Logic Gate Emulator

This program, by Neville Predebon, shows you the result of having AND, NAND, OR or NOR gates in a circuit. Just follow the prompts.

```
   1 REM "gates"
   5 REM © Predebon 1983
   7 BORDER 6
  10 PRINT INK 1;AT 2,13;"GATES"
; INK 0;AT 10,6;"A logic gate ut
ility"
  12 INK 0
  15 PRINT AT 18,3;"press any ke
y to continue": PAUSE 0: CLS
  20 PRINT AT 5,1;"Key the gate
-": PRINT
  25 PRINT TAB 5;"AND = 1"
  30 PRINT TAB 5;"NAND = 2"
  35 PRINT TAB 5;"OR = 3"
  40 PRINT TAB 5;"NOR = 4"
  45 PAUSE 0: CLS
  47 GO SUB 115
  50 PRINT AT 8,8;"1=ON, 0=OFF":
 PRINT : PRINT
  55 PRINT TAB 2;"Switch A - 001
1"
  60 PRINT TAB 2;"Switch B - 010
1": PRINT
  65 IF INKEY$="2" THEN GO TO 85
  70 IF INKEY$="3" THEN GO TO 90
  75 IF INKEY$="4" THEN GO TO 95
  80 PRINT TAB 4;"Output - 0001"
: GO TO 100
  85 PRINT TAB 4;"Output - 1110"
: GO TO 100
  90 PRINT TAB 4;"Output - 0111"
: GO TO 100
  95 PRINT TAB 4;"Output - 1000"
 100 INPUT INK 1;"Do you need an
other run? ";a$
 105 IF a$="yes" OR a$="YES" THE
N GO TO 15
 110 PRINT : PRINT INK 2;TAB 9;"
Very well": STOP
```

```
115 IF INKEY$="1" THEN PRINT AT
3,4;"AND";
120 IF INKEY$="2" THEN PRINT AT
3,4;"NAND";
125 IF INKEY$="3" THEN PRINT AT
3,4;"OR";
130 IF INKEY$="4" THEN PRINT AT
3,4;"NOR";
135 PRINT " GATE": RETURN
```


Aural Assault

Each RANDOMIZE call in this brief program by Graham White produces a different sound. There are five such sounds which you can incorporate into your own programs.

```
  10 CLEAR 32399
  20 FOR a=32400 TO 32549
  30 READ n: POKE a,n
  40 NEXT a
  50 DATA 6,3,197,33,15,0,17,40,
0,205,161,3,225,17,4,0,167,2
37,193,16,
250,201,0
  60 DATA 6,20,197,33,0,3,17,1,0
,205,161,3,225,17,16,0,167,20
7,82,32,240,193,16,233,201,0,0,
0,0
  70 DATA 6,5,197,33,15,0,17,40,
0,205,161,3,225,17,16,0,167,
237,82,32,240,193,16
,233,201,0
  80 DATA 6,2,197,33,0,6,17,5,0,
205,161,3,225,17,8,0,167,237
,82,32,240,193,16,233,201,0,0,0,
0
  90 DATA 6,50,197,33,0,1,17,1,0
,205,161,3,225,17,16,0,167,20
7,82,32,240,193,16,233,201,0,0,
0,0
 100 RANDOMIZE USR 32400
 110 RANDOMIZE USR 32430
 120 RANDOMIZE USR 32460
 130 RANDOMIZE USR 32490
 140 RANDOMIZE USR 32520
 150 GO TO 100
```

Rainbird

Now you can see how your computer can mix its standard colors to produce up to 128 different shades.

Some programs which produce this effect use a graphics character with a checker-board pattern in order to produce a pattern of dots to mix the colors. This program allows for this option, as well as two others. You can see that horizontal lines produce a color which is more steady on the screen than the checker-board pattern produces.

```
  2 FOR a=0 TO 7 STEP 2: POKE U
SR "a"+a,BIN 11111111: POKE USR
"a"+a+1,BIN 00000000: NEXT a
  3 FOR a=0 TO 7 STEP 2: POKE U
SR "b"+a,BIN 10101010: POKE USR
"b"+a+1,BIN 01010101: NEXT a
  4 FOR a=0 TO 7 STEP 2: POKE U
SR "c"+a,BIN 11110000: POKE USR
"c"+a+1,BIN 00001111: NEXT a
  6 PAPER 7: INK 1: BORDER 7: C
LS : GO TO 100
  10 PAPER 0: INK 7: BORDER 0: C
LS : PRINT AT 16,0;" 0   1   2
 3   4   5   6   7   paper"
  15 FOR c=0 TO 1
  20 PRINT AT 19,12;"bright ";c;
: PRINT #0;"Press any key or 'sp
ace' for the          Menu"
  25 FOR a=0 TO 7: FOR b=0 TO 7
  30 BEEP .003,RND*40-10
  35 IF b=0 THEN PRINT AT a*2+1,
b*4;a;AT 0,0;"ink"
  40 IF c=1 THEN PRINT BRIGHT 1
; INK a; PAPER b;AT a*2,b*4;"▒▒▒▒
▒▒";AT a*2+1,b*4;"▒▒▒▒▒▒"
```



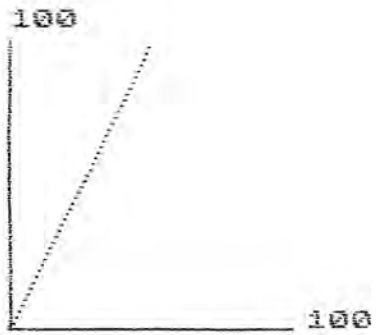
```
  41 IF c=2 THEN PRINT BRIGHT 1
; INK a; PAPER b;AT a*2,b*4;"▒▒▒▒
▒▒";AT a*2+1,b*4;"▒▒▒▒▒▒"
  42 IF c=3 THEN PRINT BRIGHT 1
; INK a; PAPER b;AT a*2,b*4;"▒▒▒▒
▒▒";AT a*2+1,b*4;"▒▒▒▒▒▒"
  45 IF b=0 THEN PRINT AT a*2+1,
b*4;a;AT 0,0;"ink"
  50 NEXT b: NEXT a
  60 IF INKEY$="" THEN GO TO 60
  70 IF INKEY$<>" " THEN NEXT c:
 GO TO 15
  80 GO TO 150
 100 PRINT AT 0,1;"128 COLOUR SP
ECTRUM "; OVER 1;AT 0,1;"
                         "
 110 PRINT AT 2,1;"This program
demonstrates how  the standard e
ight colours of   the ZX Spectru
m can be mixed to produce up to
128 different     shades."
 120 PRINT '" Some programs whic
h use this   effect use a graphi
c character  with a chessboard t
ype pattern  of dots to mix the
colours, this option is provided
but generally a series of horizon
tal lines    will work better as
 this way the slight dot crawl ef
fect is      reduced rather than
 exaggerated!"
 130 PRINT AT 19,9;"HIT  ANY KEY
"
 140 PAUSE 0
 145 CLS
 150 PAPER 7: INK 1: BORDER 7: C
LS : PRINT AT 1,13; INVERSE 1;"M
ENU"
 160 PRINT AT 5,0;"1:Horizontal
lines ▒▒▒▒  HIT A";AT 7,2;"2:Ch
essboard pattern ▒▒  KEY";AT 9
,0;"3:Dashed lines ▒▒▒▒  1
-3"
 170 PRINT AT 13,1;"The colour c
ombination for eachshade can be
read from the axes of the chart.
 It must be noted  that the ink
and paper colours  are not inter
changeable, that is for example,
blue ink on red   paper is a di
```

```
fferent colour to red ink on bl
ue paper."
 180 PRINT #0;AT 0,1; INK 1;"The
 routines for generating theuser
 graphics are in lines 2-4."
 190 IF INKEY$<>"" THEN GO TO 19
0
 200 LET a$=INKEY$
 205 IF a$>"3" OR a$<"1" THEN GO
 TO 200
 210 LET cl=VAL a$: GO TO 10
```


Line of Best Fit

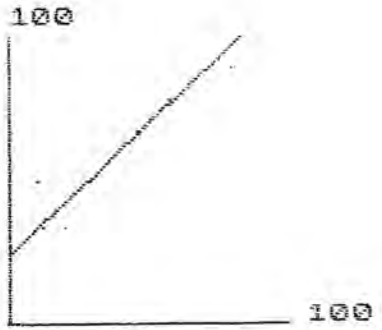
This useful utility program, written by Neal Cavalier-Smith, allows you to enter a number of points. The computer will then plot a line of best fit for these points, and give the equation of this line. You can see it in action in this sample run:

```
Equation of best fitting line is
Y=2X+0
```




```
Input points were... (1,2)
( 4, 8) (40,80) (23,46)
```

```
Equation of best fitting line is
Y=0.93X+24.5
```



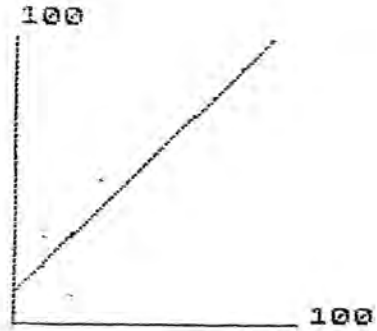
This is the listing for your own 'line of best fit':

```
  10 LET a=0: LET b=0: LET m=0
  20 LET k=0: INK 7  : PAPER 0
  30 CLS : BORDER 0
  40 INPUT "How many points?";n
  50 DIM x(n)
  60 DIM y(n)
  70 FOR c=1 TO n
  75 INPUT x(c),y(c)
  76 PLOT x(c),y(c)
  80 LET a=a+x(c)
  90 LET b=b+y(c)
 100 NEXT c
 110 LET a=a/n: LET b=b/n
 120 FOR c=1 TO n
 130 LET m=m+(((x(c)-a)*(y(c)-b)
)/((ABS (x(c)-a))↑2))
 140 NEXT c
 145 LET m=m/n
 150 LET k=b-m*a
 160 LET c=0
 195 PLOT 0,0: DRAW 100,0: PRINT
AT 21,13;"100"
 196 PLOT 0,0: DRAW 0,100: PRINT
AT 8,0;"100"
```



```
 200 PRINT AT 0,0;"Equation of b
est fitting line is"
 210 PRINT AT 2,0;"Y=";INT (m*10
0+.5)/100;"X+";INT (k*10+.5)/10
 220 FOR c=1 TO 100
 225 LET d=m*c+k
 230 IF d<100 THEN PLOT c,d
 240 NEXT c
 250 STOP
```

```
Equation of best fitting line is
Y=0.98X+11
```



Hall of Fame

This can be used in any arcade type game. As an example, type in a number above 100, then your name, and in a second or so the computer will calculate and print up the new Hall of Fame. This program was created by David Perry.

```
  1 REM  HALL OF FAME
  2 REM (C) DAVID PERRY 1983
  3 REM
  4 REM NUM=NUMBER OF SCORES
  5 REM
  6 REM
  9 BORDER 0: PAPER 0: INK 7: B
RIGHT 1: CLS
 10 LET NUM=10: DIM N(NUM+1): D
IM N$(NUM+1,8): FOR N=1 TO NUM:
LET N(N)=1100-(N*100): LET N$(N)
="SPECTRUM": NEXT N
 15 CLS : INPUT "FOR AN EXAMPLE
 ENTER YOUR SCORE              SCO
RE:";SC
 20 IF SC<=N(10) THEN  GO TO 20
0
 30 LET NUM=11: IF SC>=N(NUM) T
HEN  INPUT "ENTER 8 INITALS! ";P
$: IF LEN P$>8 THEN  GO TO 20
 35 PRINT AT 7,0; FLASH 1; BRIG
HT 0; INK 7; PAPER 2;"THIS WILL
ONLY TAKE A FEW SECS!"
 40 IF SC>=N(NUM) THEN  LET N(N
UM)=SC: LET N$(NUM)=P$
```



```
 50 FOR A=1 TO (NUM-1): LET B$=
N$(A): LET C$=N$(A+1): LET B=N(A
): LET C=N(A+1): IF B<C THEN  LE
T N(A)=C: LET N(A+1)=B: LET N$(A
)=C$: LET N$(A+1)=B$
 60 NEXT A: FOR N=1 TO NUM-1: I
F N(N)<N(N+1) THEN  GO TO 50
 70 NEXT N
 80 CLS
 90 PRINT AT 2,4;"H A L L  O F
 F A M E !"
 100 PRINT AT 3,4;"==========
==========="
 110 FOR N=1  TO NUM-1: PRINT AT
 N+5,7; INK 6;"("; INK 2;N; INK
6;")";AT N+5,12; INK 7;N(N): PRI
NT AT N+5,17; INK 5;N$(N): NEXT
N
 120 PLOT 0,0: DRAW 255,0: DRAW
0,175: DRAW -255,0: DRAW 0,-175:
 PRINT AT 17,8; PAPER 2; INK 7;"
 PRESS ANY KEY! ": LET I=0
 130 LET I=I+1: IF I>7 THEN  LET
 I=0
 140 PRINT AT 2,4; INK I;"H A L
L  O F  F A M E !"
 150 BEEP .01,I*7: PAUSE 2: IF I
NKEY$="" THEN  GO TO 130
 160 GO TO 15
 200 PRINT AT 10,4;"SORRY SCORE
TO LOW!!!"
 210 PRINT AT 13,7;" PRESS A KEY
! "
 220 PAUSE 0: RUN
```

Spec File

This outstanding database program comes from Michael Briggs. It stores data in a very compact way, ensuring maximum use is made of the memory.

When the program is run, you'll be asked to enter your name, the file title, field titles and the like. Note that an entry cannot be longer than 256 characters, and a field must be 28 characters or less.

A menu will be displayed, with the following options:

1/ BEGIN NEW FILE: This runs the program, clearing all data

2/ ENTER RECORD(S): This will ask you to enter a record, field by field. It will then sort the record into alphabetical order with the rest of the file. If you make a mistake when entering any field, simply enter "Q" to return to the menu

3/ SEARCH: This searches through the file for either a string of letters or a numbered record

4/ PRINT ALL RECORDS: This prints the first field of all files and allocates a specific entry number to them

5/ SAVE FILE: This saves the file under the title of the file. If the title is an invalid file name, the word "index" will be used

6/ STATUS: This will tell you various facts about the state of the program, such as much space is left in the file, how many entries there are, and how long the computer has been running the file

You'll find that even though this sounds complex when you're reading about it, you'll find it is really quite simple to run the program.

```
  1 REM MICHAEL BRIGGS
  2 POKE 23609,40: BORDER 6: IN
K 1: PAPER 7: CLS : DIM Z$(35000
): LET Z=1: LET CO=0
  8 INPUT "FORENAME...":C$
  9 INPUT "SURNAME...":S$
 10 INPUT "TITLE OF FILE ":T$
 20 INPUT "NUMBER OF FIELDS ":F
 30 DIM A$(F,20): FOR A=1 TO F:
 PRINT "ENTER TITLE FOR FIELD "
A: INPUT B$: LET A$(A,2 TO )=B$:
 LET A$(A,1)=CHR$ LEN B$: PRINT
":-": INK 2:B$: PRINT : NEXT A
 70 PAUSE 50
 100 POKE 23609,40: BORDER 6: CL
S : INK 0: PLOT 0,0: DRAW 255,0:
 DRAW 0,175: DRAW -255,0: DRAW 0
,-175: INK 3
```

```
 110 PRINT  INK 0: BRIGHT 1:AT 1
,13-LEN T$/2:T$:" MENU"
 120 PRINT AT 3,3:"  NO 1.BEGIN
NEW FILE."
 130 PRINT AT 5,3:"  NO 2.ENTER
NEW RECORD(S)."
 140 PRINT AT 7,3:"  NO 3.SEARCH
 RECORDS."
 150 PRINT AT 9,3:"  NO 4.PRINT
ALL RECORDS."
 160 PRINT AT 11,3:"  NO 5.SAVE
FILE."
 165 PRINT AT 13,3:"  NO 6.FILE
STATUS."
 170 PRINT  INK 2:AT 15,8:"@ MIC
HAEL BRIGGS"
 180 PRINT  INK 1:AT 17,11-(LEN
C$+LEN S$)/2:"FILE FOR ":C$:" ":
S$
 190 INPUT "CHOICE...":C
 195 CLS
 196 INK 1
 200 IF 0<C AND C<8 THEN  GO TO
C*1000
 210 GO TO 100
1000 PRINT AT 10,2:"ARE YOU SURE
 ":C$:" ?"
1010 INPUT Y$
1020 IF Y$(1)="Y" OR y$(1)="y" T
HEN  RUN
1030 GO TO 100
2000 INK 2: FOR A=1 TO F
2010 PRINT AT A*2-1,13-(CODE A$(
A,1)/2):"ENTER ":A$(A,2 TO )
2020 NEXT A
```



```
2021 PRINT AT 21,0: INK 1:"ENTER
 ^Q^ TO RETURN TO MENU."
2030 LET E$=""
2040 FOR A=1 TO F
2050 PRINT AT A*2-1,13-(CODE A$(
A,1)/2):"ENTER ": FLASH 1:A$(A,2
 TO 1+CODE A$(A))
2060 INPUT F$
2061 IF F$="Q" THEN  GO TO 100
2070 LET E$=E$+CHR$ LEN F$+F$
2080 PRINT AT A*2-1,13-(CODE A$(
A,1)/2):"ENTER ":A$(A,2 TO 1+COD
E A$(A))
2090 PRINT  INK 0:AT A*2,16-LEN
F$/2:F$
2100 NEXT A
2110 LET X1=1: FOR A=1 TO CO: LE
T X$=Z$(X1+2 TO X1+CODE Z$(X1+1)
+1)
2115 IF X$>E$(2 TO ) THEN  GO TO
 2400
2120 LET X1=X1+CODE Z$(X1)
2125 NEXT A
2126 LET E$=CHR$ (LEN E$+1)+E$:
LET Z$(Z TO Z+LEN E$)=E$
2131 LET Z=Z+LEN E$
2140 LET CO=CO+1: CLS : GO TO 20
00
2400 LET E$=CHR$ (LEN E$+1)+E$
2410 LET Z$(X1+LEN E$ TO Z+LEN E
$)=Z$(X1 TO Z)
2420 LET Z$(X1 TO X1+LEN E$-1)=E
$
2430 GO TO 2131
2999 REM SEARCH ROUTINE
```

```
3000 PRINT TAB 11:"SEARCH MODE":
AT 2,0:"ENTER...":AT 3,0:" ^N^ I
F ENTERING RECORD NUMBER.    ^^ I
F ENTERING CHARS ONLY.": INPUT "
KEYWORD":K$
3004 LET M$=" ": IF K$(1)="N" TH
EN  LET M$=K$(1): LET K$=K$(2 TO
 )
3005 PRINT TAB 8:"KEYWORD=":K$
3006 LET RET=3040
3010 LET X=1: FOR A=1 TO CO
3015 LET X1=X: LET V$=Z$(X TO (C
ODE Z$(X)+X-1))+" ": IF M$="N" T
HEN  GO TO 3300
3016 FOR J=1 TO LEN V$-LEN K$
3020 IF V$(J TO J-1+LEN K$)=K$ T
HEN  GO TO 3200
3030 NEXT J
3040 LET X=X1+LEN V$-1: NEXT A
3050 PRINT ''' INK 0:"PRESS A KE
Y": PAUSE 0: GO TO 100
3200 PRINT '' INK 4:"ENTRY NO...
":A: LET X2=X1: LET X1=X1+1: FOR
 K=1 TO F
3210 PRINT  INK 2:A$(K,2 TO CODE
 A$(K)+1):"...": PRINT  INK 1:TA
B 32-CODE Z$(X1):Z$(X1+1 TO X1+C
ODE (Z$(X1)))
3220 LET X1=X1+1+CODE Z$(X1)
3225 NEXT K
3240 LET X1=X2: GO TO RET
3300 IF A<>VAL K$ THEN  GO TO 30
40
3310 LET RET=3050
3320 GO TO 3200
```



```
4000 PRINT "ANY KEY WILL STOP TH
E SCROLLING DISPLAY UNTIL KEY IS
 RELEASED."
4005 PAUSE 0
4010 PRINT AT 18,16-LEN T$/2: IN
K 2;T$
4020 INK 1: PRINT
4030 PRINT
4040 LET X=1: FOR A=1 TO CO
4049 PRINT AT 21,0: POKE 23692,-
1
4050 PRINT "0000":AT 21,4-LEN (S
TR$ A):A;":";Z$(X+2 TO X+1+CODE
(Z$(X+1)))
4060 LET X=X+(CODE Z$(X))
4065 IF INKEY$<>"" THEN  GO TO 4
065
4070 NEXT A
4075 PRINT '"PRESS ANY KEY TO R
ETURN TO MENU."
4080 PAUSE 0: GO TO 100
5000 PRINT "I SHALL SAVE THE FIL
E UNDER THE                NAME OF"
: PRINT AT 10,16-LEN T$/2: FLASH
 1;T$
5005 IF T$="" OR LEN T$>10 THEN
 PRINT AT 10,1:"                IN
DEX                ": SAVE "INDEX"
LINE 100: GO TO 100
5010 SAVE T$ LINE 100
5020 GO TO 100
6000 PLOT 0,0: DRAW 255,0: DRAW
0,175: DRAW -255,0: DRAW 0,-175
6010 PRINT AT 1,12-(LEN T$)/2; B
RIGHT 1; INK 2;T$;" STATUS"
```

```
6020 PRINT AT 3,2;"COMPUTER ON T
IME  :";INT (((PEEK 23672+256*PE
EK 23673)+65536*PEEK 23674)/50);
" SECS"
6030 PRINT AT 5,10-(LEN S$+LEN C
$)/2;"FILE FOR ";C$;" ";S$
6040 PRINT AT 7,8;"NO.OF RECORDS
 :";CO
6050 PRINT AT 9,2;"CHARACTER SPA
CE LEFT :";35000-Z
6060 PRINT AT 11,4;"CHARACTERS I
N FILE :";Z
6070 PRINT AT 13,3;"PROGRAM:-@ M
ICHAEL BRIGGS"
6075 PRINT AT 20,5;"PRESS ANY KE
Y FOR MENU"
6080 PRINT AT 3,2;"COMPUTER ON T
IME  :";INT (((PEEK 23672+256*PE
EK 23673)+65536*PEEK 23674)/50);
" SECS"
6085 IF INKEY$<>"" THEN  GO TO 6
100
6090 GO TO 6080
6100 GO TO 100
```


STRUCTURED PROGRAMMING

Sketching an outline

Many times I've written articles and chapters in books which are supposedly going to improve the programming skill of those who read them. But every time I produce such a list of 'things you really should be doing when you program', or explain the material to someone, I am reminded of an old story about a farmer.

He was approached by a young man selling correspondence courses in 'Effective Farming'. "Don't you see," the young salesman said as part of his pitch, "that if you take this course you'll know so much more about farming?". The farmer replied "I don't even farm now as good as I know how."

So it goes, I suggest, for advice on programming. I know full well that I do not "program as good as I know how". Many times I break all the rules, wading straight into coding a complex program without even a thought for the ideas of 'structured programming'.

So I suggest you keep in mind, when reading this appendix, that I do not really belive that you – or anyone else – is really going to take the ideas here as rules which must be obeyed, come hell or high water. The best way to approach this,

and any other material in a similar vein which you come across, is to read it carefully, and make your own assessment on each suggestion given. Then, just apply the ideas which seem most sound to you.

The basic idea I'm outlining in this appendix is that of approaching structured programming by 'sketching an outline'.

The idea is simple, but very valuable in order to help you write complicated and involved programs, such as many of those in this book. Of course, you may well be already creating very complex programs without using anything like the idea I'm about to outline. Even if you are, I suggest you think carefully about these ideas, so that you can see that they may make your job easier.

The fundamental idea of structured programming lies hidden in the phrase 'top down programming'. This suggests that you start the process of programming by first stating inwords the broad aim of the program you are about to write. You follow this up by writing a series of notes - each of which will eventually be a subroutine - which cover each of the main tasks which the computer will carry out.

To clarify this last paragraph, I'll give you a concrete example. Here are the steps you could follow when trying to write a Noughts and Crosses program. (Note that

'structured programming' produces listings which are much longer than might otherwise be the case. However, this greater length is more than offset by the fact that programs built up in this way are very simple to debug and improve, and their structure can be readily understood by other programmers. You may not be able to make such claims with confidence about some of your current programs.)

```
SET UP INITIAL BOARD
  COMPUTER MOVE
     (A) CHECK IF MIDDLE SQUARE EMPTY,
     IF NOT MOVE THERE
     (B) CHECK IF THERE IS A COMPUTER
     WINNING MOVE, IF SO MAKE IT
     (C) CHECK IF HUMAN WILL WIN ON NEXT
     MOVE, IF SO BLOCK
     (D) IF NO MOVE MADE, CHECK TO SEE
     IF A RANDOM MOVE CAN BE MADE, IF SO
     MAKE IT, IF NOT DECLARE A DRAW
  PRINT BOARD
ACCEPT PLAYER MOVE
  PRINT BOARD
CHECK IF PLAYER HAS WON, IF SO STOP
  GOTO 'COMPUTER MOVE'
```

As you'll see if you take a few moments to study this sketched outline, you can tell exactly which steps the computer will follow; the decisions it will be called to make; and the alternative results of those decisions.

You have already completed the first step towards writing a Noughts and Crosses program. You'll see, by the way, that it doesn't matter at all if you do not yet know how you are going to get your computer to carry out the necessary steps. All that matters is that you know that the steps must, in some fashion, be executed in due course.

The second step in the process consists of turning the sketched outline into a series of subroutine calls. In languages which are more structured than the BASIC you have on your computer (such as the BASIC provided on the BBC Micro, or Pascal) it is relatively easy to call a number of subroutines within an endless loop, with a structure like DO/WHILE or REPEAT/UNTIL. These allow you to repeat a series of program steps indefinitely until a certain condition is satisfied, or the state of a pointer or flag changes.

In the BASIC we have here, however, we have to be content with the humble, and much-maligned GO TO. (Much of the reaction against GO TO, which many 'serious programmers' see as the greatest insult to the art of true programming ever developed, came from early and relatively primitive versions of BASIC, in which you can only follow an IF/THEN with a GO TO. This meant the program leapt all over the place, leading to 'spaghetti code' which was horrendously difficult to interpret.)

The second stage is to turn our sketched outline into a series of subroutine calls, endlessly cycled by a GO TO as follows:

```
10 REM NOUGHTS AND CROSSES
20 GOSUB 9000:REM INITIALISE
30 GOSUB 1000:REM COMPUTER MOVE
40 GOSUB 8000:REM PRINT BOARD
50 GOSUB 2000:REM ACCEPT PLAYER MOVE
60 GOSUB 8000:REM PRINT BOARD
70 IF human has not won AND computer
   has not won THEN GO TO 30
80 PRINT "CONGRATULATIONS"
```

Now, as you can see, we have the 'framework' for a workable Noughts and Crosses program, even though we do not yet have a clue as to how the program will actually work. We can now set about writing the program from first principles.

There are two further advantages of this 'outline' approach. If there is something we cannot, at this stage, actually program (like the basis upon which the computer finds out who has won) we can simply put in a PRINT statement within the subroutine like PRINT "CHECKING FOR WIN" and continue to use the program, as we work on it. Then, each time the computer should be checking for a win, it will print up CHECKING FOR WIN. This means you can continue working on the program, without being needlessly held up on a minor

subroutine which you cannot, for the moment, solve.

The second advantage comes towards the end of the program development stage, the debugging stage. I always find this the most frustrating, and in many ways, the least rewarding aspect of programming. Although I can often get a program working reasonably well fairly quickly (although my first chess in BASIC too nearly six months), to get the program from 'working reasonably well' to 'performing without faults under all conditions' can take as long as it took to get the first version even working at all. However, when you set up the program in the way described in this section, you'll see that the debugging stage can be greatly simplified.

For example, you may find in your Noughts and Crosses program that the computer tends to ignore the bottom right hand corner, when a move into this position would enable it to win, or to block a win from the human. From the 'subroutine loop' we set up before, we know the computer's move must be made somewhere between lines 1000 and 1999. This immediately narrows down the search. If you've been clever, and have made each subroutine a series of further subroutine, each constructed in a similar way to our major subroutine, it would be even easier to track down the bug.

Let's look at this idea - making each subroutine a series of further subroutines - a little more closely. We'll look at the subroutine starting at line 1000, the one in which the computer actually makes its move, as this is the most important (and difficult to program) of all those in this game.

The subroutine could begin as follows:

```
2000 REM COMPUTER MOVE
2010 LET MOVE=0:REM IF THIS BECOMES 1 A
  VALID MOVE HAS BEEN FOUND
2020 GOSUB 2200:REM CHECK IF MIDDLE
  SQUARE IS EMPTY
2030 IF MOVE=1 THEN RETURN
2040 GOSUB 2400:REM CHECK IF A POSSIBLE
  WINNING MOVE EXISTS
2050 IF MOVE=1 THEN RETURN
2060 GOSUB 2600:REM CHECK IF A POSSIBLE
  HUMAN WIN CAN BE BLOCKED
2070 IF MOVE=1 THEN RETURN
2080 GOSUB 2800:REM CHECK IF ANY MOVE AT
  ALL CAN BE MADE
2090 IF MOVE=1 THEN RETURN
2100 REM A RETURN WITH MOVE=0 MEANS NO
  FURTHER MOVES ARE POSSIBLE
2110 RETURN
```

As I said earlier, programs constructed using the 'endless loop of subroutines' coupled with 'subroutines within subroutines' make listings which may be far longer than usual. However, there is usually little need to worry about running

out of memory (as was the case on the ZX81 and the T/S 1000 when counting bytes became the dominant feature of our programming lives). The extra typing involved in producing the longer listings will be amply repaid by the extra clarity your programs attain, and the much shorter time it will take to produce a debugged masterpiece of which you can be proud.

You'll find if you take the trouble to start with a 'sketched outline', the whole process of constructing a major program is much, much simpler than might otherwise be the case. You'll also find that the time involved will be much more constructively spent that it would have been if you had just waded into the programming without taking the time to do your sketching first.

Ignore the old farmer, and try now to program 'as good as you know how'.