The Virgin Computer Games Series

GAMES FOR YOUR DRAGON

£££££'s of entertaining games for only £2.95

GAMES FOR YOUR DRAGON

By
Clive Gifford

Virgin Books

First published in Great Bntain in 1983 by Virgin Books Ltd,
61-63 Portobello Road,London W11 3DD.

ISBN 0 907080 89 8

Printed and bound in Great Britain by Richard Clay (The Chaucer Press) Ltd, Suffolk.

Production services by Book Productıon Consultants, Cambridge.

Designed by Ray Hyden

Illustrated by Sue Walliker.

Typeset by QV Typesetting.

Distributed by Hamlyn Paperbacks

TO PETER, SALLY, SUSAN AND CAROLINE

TIM HARTNELL — SERIES EDITOR

Tim Hartnell is a leading computer expert and journalist, who has contributed extensively to the Technical Consumer Press. He is also the author of several books including *Getting Acquainted With Your ZX81*, *Let Your BBC Micro Teach You to Program* and *Programming Your ZX Spectrum*.

CLIVE GIFFORD — THE AUTHOR

Clive Gifford is a 17-year-old student studying for his 'A' levels in Politics, Business Studies and Law. He has contributed to *ZX Computing*, *Interface* and *Dragon's Teeth* magazines, and has previously written *Making the Most of Your Dragon 32*. Any leisure time is spent playing golf and listening to music.

SUE WALLIKER — THE ILLUSTRATOR

Sue Walliker is a freelance illustrator.

ACKNOWLEDGEMENTS

The author wishes to thank his family for their help in preparing the programs, Tim Hartnell, Peter Shaw and Chris Palmer for their contributions, Liz North for her friendly advice and all his friends at Ashford College for their support.

CONTENTS

Editor's Introduction

Your computer is waiting to challenge you. Moving graphics games, brain stretchers, word games and puzzles are all here and ready to entertain you.

A wide variety of games are included in this book. The programs have been written by some of the most talented young programmers working in this country at the moment, and represent a variety of approaches to solving programming problems.

An examination of the listings should teach you many tricks and techniques to apply to your own programming And once you have mastered the programs in their present form, you might want to try your hand at improving them. There is no such thing as a 'perfect program', so these games are sure to benefit from your programming skill.

All that now remains is for you to turn the page and enter the programs. I can only hope that you enjoy playing the games as much as we did when preparing this volume.

Tim Hartnell, series editor
London
March 1983

Author's Introduction

The Dragon 32 is, in my opinion, one of the first family computers. At a reasonable price, the Dragon offers a robust design, a strong keyboard, an excellent BASIC and, furthermore, it's British!

This book has been written with the emphasis on fun. Among the arcade and strategy games are sprinkled several pattern programs which show the Dragon's graphics at their best. I have tried to achieve a fair balance between the strategy games, such as Checkers and Pandora's Box, and the action games, such as Lumberjack and Nightdriver.

The programs have been been kept to a reasonable length and once you've typed them in, feel free to alter them. My versions are by no means definitive ones, so go ahead and improve them. It's one of the best ways to improve your programming skills. I hope you enjoy the programs and learn a few programming techniques.

Clive Gifford
Ashford, Middlesex
April, 1983

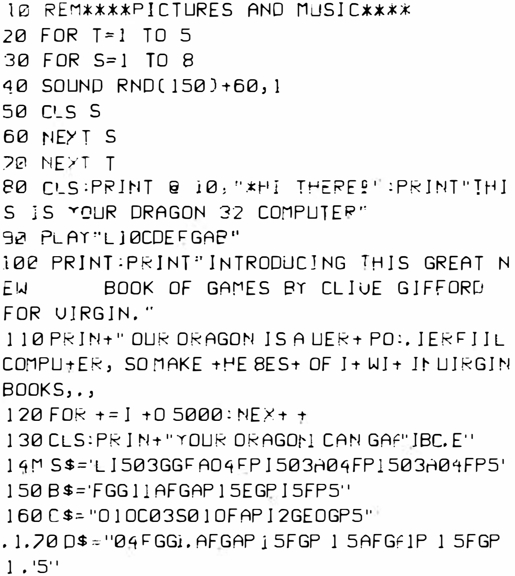
Program Notes

Many of the programs feature lower case letters which appear as inverse upper case onthe screen but as lower case letters on the printout of the program listing. To enter the lower case letters, you must press SHIFT O, then the letters required and then return to upper case

SHIFT The O up again arrow on the program listing appears as '^'.
Several programs feature a continuous loop at the end of the game: to stop this, press BREAK.

PICTURES AND MUSIC

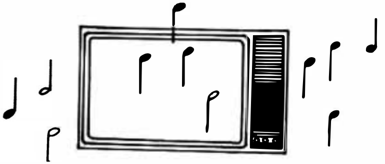
No instructions are needed for this demonstration program, just type it in and watch your television screen. By the way, turn up the volume on your television and have a listen.



```
10 REM****PICTURES AND MUSIC****
20 FOR T=1 TO 5
30 FOR S=1 TO 8
40 SOUND RND(150)+60,1
50 CLS S
60 NEYT S
70 NEYT T
80 CLS:PRINT @ 10;"*HI THERE!" :PRINT"THI
S IS YOUR DRAGON 32 COMPUTER"
90 PLAY"L10CDEFGAB"
100 PRINT:PRINT"INTRODUCING THIS GREAT N
EW        BOOK OF GAMES BY CLIVE GIFFORD
FOR VIRGIN."
110PRIN+" OUR ORAGON IS A UER+ PO:.IERFIIL
COMPU+ER, SOMAKE +HE 8ES+ OF I+ WI+ IN UIRGIN
BOOKS,.,
120 FOR +=I +0 5000:NEX+ +
130 CLS:PRIN+"YOUR ORAGON CAN GAF"IBC.E"
14M S$='LI503GGFA04FPI503A04FP1503A04FP5'
150 B$='FGG11AFGAP15EGPI5FP5''
160 C$="O1OC03S01OFAPI2GEOGP5"
.1.70 D$="04FGGi.AFGAP i5FGP 1 5AFGf1P 1 5FGP
1.'5''
```

```
180 E$="AFGAP15EGP15F"
190 PLAY"XS$;XB$;XS$;XC$;XS$;XB$;XD$;XE$;"
200 CLS
210 PRINT"TRAP YOU IN MAZES"
220 Z=64
230 FOR S=1 TO 479
240 R=RND(3):IF R=2 THEN PRINT CHR$(175);
ELSE PRINT CHR$(128);
250 NEXT S
260 PRINT @ 510,"&"; PRINT @ 240,"HELP!!";
270 FOR T=1 TO 1500:NEXT T
280 CLS
290 PRINT"AND PAINT BEAUTIFUL PATTERNS"
300 FOR T=1 TO 1200:NEXT T
310 PMODE 4,1:PCLS:SCREEN 1,1
320 FOR X=1 TO 255 STEP 2
330 FOR Y=1 TO 191 STEP 3
340 LINE(X,98)-(128,Y),PSET
350 LINE(128,X)-(Y,98),PSET
360 N=N+1:IF N>50 THEN GOTO 390
370 NEXT Y
380 NEXT X
390 FOR T=1 TO 1200:NEXT T
400 CLS:PRINT"WITH THE PROGRAMS IN THIS
BOOK!!"
410 FOR T=1 TO 2000:NEXT T
420 A$="U8R6BD4BL2L4D4BR8U8BR6D8L6BR8U8F
8U8"
430 PMODE 3,1:PCLS:SCREEN 1,1
440 DRAW "BM20,20;S12D4F4E4U4BR8D8BR8U8R
6D4L6BR2F4BR8"
450 DRAW "BU8BR6L6D8R6U4L2BR10BD4U8BR8BD"
```

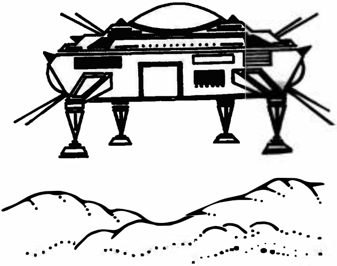
```
8U8F8U8"
460 DRAW "BM110,80;R10BL5U5D10"
470 DRAW"BM20,110;D8L2R8U8L8BR14D8U4REU4
L6D4R2F4ER6U8R6D4L6BR6D4"
480 DRAW "BR6U8R6BD4L2R2D4L6BR12REU8L6D8B
R12U8F8U8"
492 DRAW"BM110,160;R8D1L8U1BD6R8D1L8U1"
600 FOR T=1TO 1200:NEXT
610 PCLS: SCREEN 1,0
620 DRAW"BM30,30:C2;XA$;" 630
PLAY"02L100CDEFGA8"
6"0 DRAW"BM90,90;C3;XA$;" 662
PLAY"O1COEFGAB"
662 DRAW"BM16<', 160;C4;XA$;" 670PLAY"
04CDEFGABBB"
580 GOTO 610
```



MISSION APOLLO

Have you got the nerves of steel required to land a flimsy craft on the moon's surface? Whether your landing area is relatively simple or full of dangers, your on board com-puter will warn you first Enter the number of units you want to burn to slow your descent remembering that you do not have an unlimited supply you will be warned when your fuel reserve is very low.

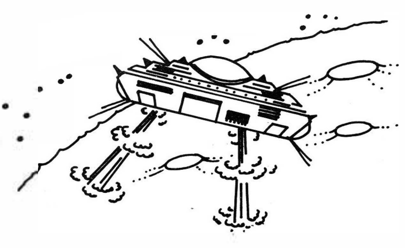
The program uses several mathematical formula to simulate realistically the correct velocity and height in relation to how much fuel is used. After you have played the game a couple of times you will start to discover the best strategy for landing your ship successfully.



```
10 REM****MISSION APOLLO****
20 CLS:PRINT"YOUR SHIP ..... APOLLO"
30 SOUND 100,8
40 FOR T=l TO 1000:NEXT T
50 PRINT:PRINT"YOUR MISSION ..... TO LAND
SAFELY"
60 SOUND 120, 10
70 FOR T=l TO 1000:NEXT T
80 PRINT:PRINT"REPORT OF LANDING SITE AR RIVED"
90 A=RND(3)
100 PRINT:PRINT"LANDING SITE IS,";
110 IF A=l THEN PRINT"EXCELLENT":L=5
120 IF A=2 THEN PRINT"FAIR":L=2
130 IF A=3 THEN PRINT"DANGEROUS":L=0
140 SOUND 140,12
150 FOR T=1 TO 1000:NEXT T
160 T=0:H=500:V=50:F=120
170 GOSUB 540
180 PRINT @ 352,"SEC FEET SPEED FUEL  ";
190 PRINT @ 384,T:PRINT @ 391,H:PRINT @
400,V:PRINT @ 408,F
200 PRINT @
416,"-------------------------------";
210 INPUT"ENTER FUEL UNITS(0-30) + ";B
220 SOUND RND(150)+50,RND(l2)
230 IF B>30 OR B<0 THEN B=0:GOTO 250
240 IF B>F THEN B=F
250 Y=V-B+5:F=F-B
260 H=H-.5*(V+Y):IF H<=0 THEN GOTO 310
270 T=T+l:V=Y:IF F<=0 THEN GOTO 300
280 IF F>0 AND F<10 THEN PRINT"FUEL LOW":SOUND
20,20
290 GOTO 170
```

```
300 PRINT"****OUT OF FUEL****" :SOUND 1,1
5:GOTO 41 0
310 PRINT
320 H=H+.5*(V+Y)
330 IF B=5 THEN 360
340 D=(-V+SQR(V*V+H*(10-2*B)))/(5-B)
350 GOTO 370
360 D=H/V
370 Y=V+(5-B)*D
380 IF Y>L THEN 410
390 GOTO 630
400 GOTO 90
410 PMDDE 0, 1:PCLS:SCREEN 1,1
420 FOR X=2 TD 300 STEP 10
430 PLAY"L255;C;E"
440 CIRCLE(128,194),X,1,.5
450 NEXT X
460 PLAY"O1V30L8GECGECGECC"
470 CLS0:PRINT @ 256,"----------YOU BLEW
IT!----------"
480 PRINT @ 288,"----YOU CRASHED AT ";FIX(Y)
+1;"M/S----";
490 FOR T=1 TO 1200:NEXT T:CLS 0
500 PRINT @ 320, "----- PLAY AGAIN? ENTER A
------";
510 INPUT Y$:IF Y$<>"A" THEN STOP ELSE RUN 520
STOP
530 REM****STAR DISPLAY****
540 CLS 0:FOR S=1 TO 10
550 R=RND(288)
560 POKE R+1024,200
570 NEXT S
580 FOR S=1 TO 5:R=RND(288)
590 POKE R+1024,152
```

```
600 NEXT S
610 PLAY"O3;L8;G;E"
620 RETURN
630 PMODE 4,1:PCLS:SCREEN 1,1
640 DRAW"BM0,60;R6D4R3D7R3D2R4D3R3D6R2D1
R4D7R6D3R4D2R3D1R4D3R5D2R4"
650 DRAW"BM213,137;U6R2U6R4U3R5U3R6U11R5
U4R3U12R3U4R1U6R3U5U3R3U12R3U10R2U1R1U2R3"
660 CIRCLE(150,100),100,1,.5,0.15,0.5
670 DRAW"BM145,114;R4F8D12G4L12H4U12E8BD
24BR2F6L12E6"
680 DRAW"BM158,134;F4D10L1R2BL32R2L1U10E4"
690 LINE(140,124)-(155,130),PSET,BF
700 DRAW"BM80,30;U4L8D8R8D8L8U4BR16BU12D
16R8U16BR16L8D16R8"
710 DRAW"BR16L8U16R8BR16L8D8R6BL6D8R8BR8
BU4D4R8U8L8U8R8D4"
720 DRAW"BR16U4L8D8R8D8L8U4BR16BD4U2BU4U
10BR8D10BD4D2"
730 REM*****MUSIC******
740 PLAY"P20V30O3L42CDEFGABAGFEDCCC"
750 PLAY"P24O4CDEFGABAGFEDCCC"
760 GOTO 740
```



BEAT THE BLUES

You, as doctor, must help a patient suffering from acute depression. You have decided against a psychiatrist – stronger action is necessary. You enter the dark void of your patient's brain in an attempt to destroy the Blue Meanies that have started to take over, armed with a syringe which is moved at the base of the screen by the left and right arrows. Fire a lethal dose of laughing gas by pressing the up arrow.

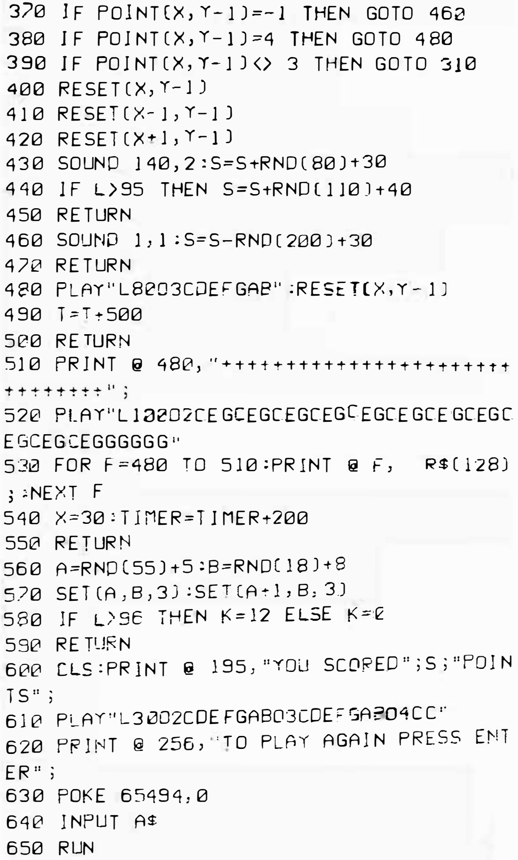
At first there is an orderly row of Meanies but then they turn mutant and appear all over the screen. If you narrowly miss a Meanie it is injured and turns orange, but you get no points for it. After some time, the patient's brain starts producing antibodies (inverse Os) which block your shots: if you hit one you lose points. Very occasionally a bonus red blood cell appears and if you hit it you gain extra time in which to destroy more Meanies.

Your time and score are displayed on the screen, and you continue until your time runs out.

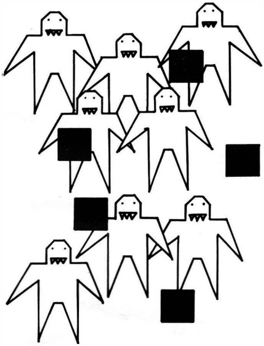
A small note about POKE 65495,0, which speeds up the game. This POKE affects the cassette output. When you finish the game type in POKE 65494,0 to reset the computer or, alternatively, switch the computer off. If you prefer not to use this POKE delete line 10, but the game will run a little slower.

```
10 POKE 65495,0
20 REM*****BEAT THE BLUES*****
30 REM******C.GIFFORD*******
40 CLS 3
50 PRINT @ 104,"BEAT THE BLUES!!!";
60 PLAY"O3L8DDEFGAGFEDL16EFEDL8CO2BO3L4C"
```

```
70 PRINT @ 228,"PRESS ENTER TO START";
80 INPUT A$
90 K=1:L=1:T=3500
100 CLS 0
110 PRINT "***SCORE: ;S;" ***TIME:      *
**"
120 FOR X=10 TO 50 STEP 4
 30 SET(X,4,3):SET(X+1,4,3)
140 NEXT X
150 TIMER = 0
160 SET(X,30,2)
170 U=RND(250)+200
180 PRINT @ 9,S;:PRINT @ 22, TIMER;
190 RESET(X,30)
200 K=K+1:IF K=20 THEN GOSUB 560
210 IF RND(200)>197 THEN SET(RND(58)+2,R
ND(20)+6,4)
220 L=L+1:IF L>100 THEN PRINT @ U,"o";:P
LAY"L2 005CG":L=97
230 IF RND(20)<>14 THEN PRINT @ U-11,CHR
$(128);
240 IF TIMER>T THEN  GOTO 600
250 IF PEEK(343)=223 THEN X=X-1
260 IF PEEK 344)=223 THEN X=X+1
270 IF PEEK(341) =223 THEN GOSUB 300
280 IF X<1 OR X>61 THEN GOSUB 510
290 GOTO 16
300 Y=28
310 SET(X, ,8)
320 IF Y>16 THEN PLAY"L23005G"
330 PRINT @ 22,TIMER;
340 RESET(X,Y)
350 SET(X,30,2)
360 Y=Y-1:IF Y<5 THEN RETURN
```



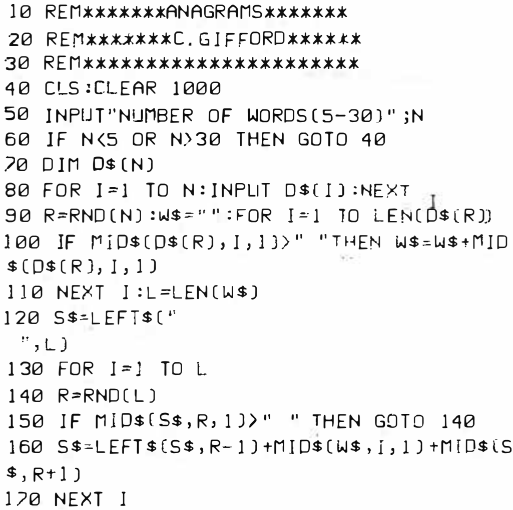
```
370 IF POINT(X,Y-1)=-1 THEN GOTO 460
380 IF POINT(X,Y-1)=4 THEN GOTO 480
390 IF POINT(X,Y-1)<> 3 THEN GOTO 310
400 RESET(X,Y-1)
410 RESET(X-1,Y-1)
420 RESET(X+1,Y-1)
430 SOUND 140,2:S=S+RND(80)+30
440 IF L>95 THEN S=S+RND(110)+40
450 RETURN
460 SOUND 1,1:S=S-RND(200)+30
470 RETURN
480 PLAY"L80O3CDEFGAB":RESET(X,Y-1)
490 T=T+500
500 RETURN
510 PRINT @ 480,"+++++++++++++++++++++++
+++++++++";
520 PLAY"L100O2CEGCEGCEGCEGCEGCEGCEGCEGC
EGCEGCEGGGGGG"
530 FOR F=480 TO 510:PRINT @ F,  R$(128)
;:NEXT F
540 X=30:TIMER=TIMER+200
550 RETURN
560 A=RND(55)+5:B=RND(18)+8
570 SET(A,B,3):SET(A+1,B,3)
580 IF L>96 THEN K=12 ELSE K=0
590 RETURN
600 CLS:PRINT @ 195,"YOU SCORED";S;"POIN
TS";
610 PLAY"L30O2CDEFGABO3CDEFGABO4CC"
620 PRINT @ 256,"TO PLAY AGAIN PRESS ENT
ER";
630 POKE 65494,0
640 INPUT A$
650 RUN
```



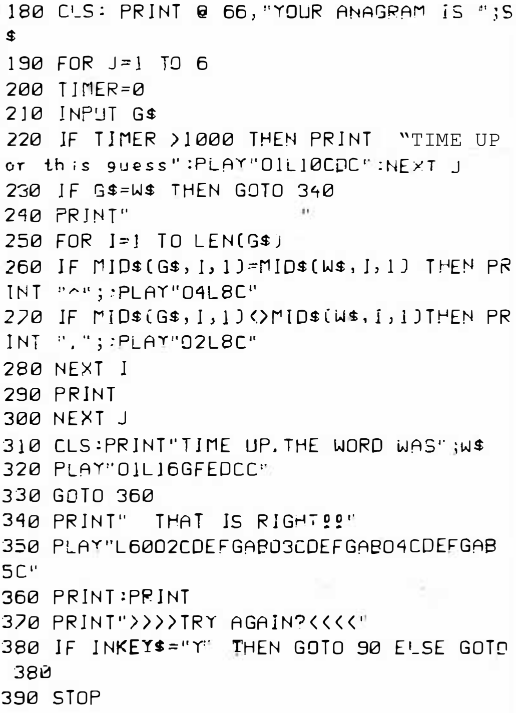
ANAGRAMS

If you think you're good at solving word jumbles, try this game (converted from Chris Palmer's TRS-80 version). Enter between 5 and 30 words of not more than 12 letters each. The computer randomly chooses one of the words and jumbles it up. You have six guesses to solve the word, but as each guess has a time limit you must be quick-thinking. The computer gives clues in the form of dots for incorrectly placed letters and up arrows for cor-rectly placed ones.

For another game using the same words in the computer's memory, just type Y.



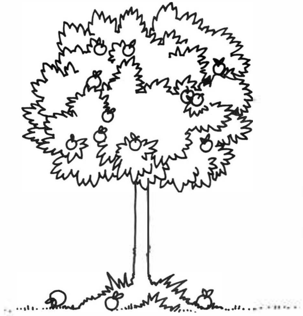
```
10 REM*******ANAGRAMS*******
20 REM*******C.GIFFORD******
30 REM**********************
40 CLS:CLEAR 1000
50 INPUT"NUMBER OF WORDS(5-30)";N
60 IF N<5 OR N>30 THEN GOTO 40
70 DIM D$(N)
80 FOR I=1 TO N:INPUT D$(I):NEXT
90 R=RND(N):W$="":FOR I=1 TO LEN(D$(R))
100 IF MID$(D$(R),I,1)>" "THEN W$=W$+MID
$(D$(R),I,1)
110 NEXT I:L=LEN(W$)
120 S$=LEFT$("
 ",L)
130 FOR I=1 TO L
140 R=RND(L)
150 IF MID$(S$,R,1)>" " THEN GOTO 140
160 S$=LEFT$(S$,R-1)+MID$(W$,I,1)+MID$(S
$,R+1)
170 NEXT I
```

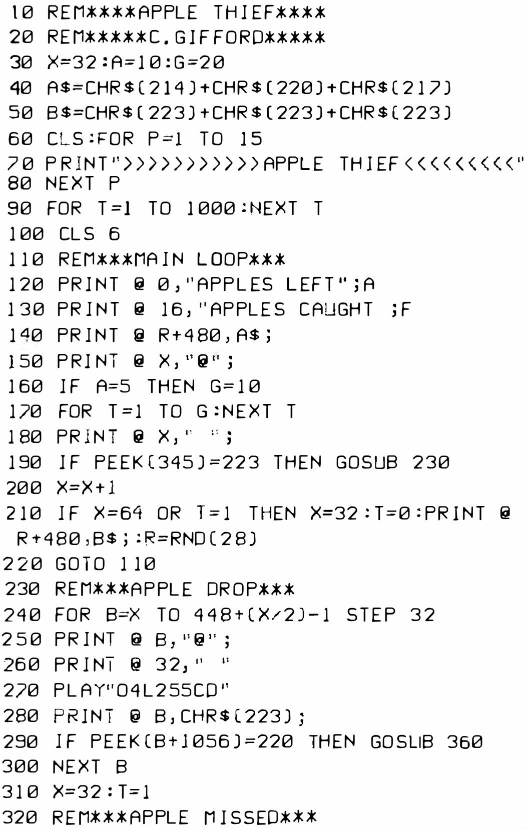


```
180 CLS: PRINT @ 66,"YOUR ANAGRAM IS ";S
$
190 FOR J=1 TO 6
200 TIMER=0
210 INPUT G$
220 IF TIMER >1000 THEN PRINT  "TIME UP
or this guess":PLAY"O1L10CDC":NEXT J
230 IF G$=W$ THEN GOTO 340
240 PRINT"                    "
250 FOR I=1 TO LEN(G$)
260 IF MID$(G$,I,1)=MID$(W$,I,1) THEN PR
INT "^";:PLAY"O4L8C"
270 IF MID$(G$,I,1)<>MID$(W$,I,1)THEN PR
INT ".";:PLAY"O2L8C"
280 NEXT I
290 PRINT
300 NEXT J
310 CLS:PRINT"TIME UP.THE WORD WAS";W$
320 PLAY"O1L16GFEDCC"
330 GOTO 360
340 PRINT"  THAT IS RIGHT!!"
350 PLAY"L60O2CDEFGABO3CDEFGABO4CDEFGAB
5C"
360 PRINT:PRINT
370 PRINT">>>>TRY AGAIN?<<<<"
380 IF INKEY$="Y" THEN GOTO 90 ELSE GOTO
 380
390 STOP
```

APPLE THIEF

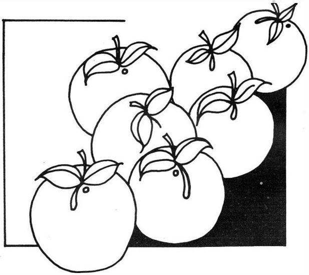
In this simple game you are in an orchard stealing apples. The apples move along the top of the screen and can be seen through the leaves of the tree. The basket into which the apples must be dropped is randomly positioned at the bottom of the screen. When you think the apple is directly above the basket, press the space bar and the apple falls from the tree If it does not land in the center of the basket it halts the ground; you are told how many apples you have caught at the end of the game. There are potentially 10 apples for you to catch.





```
10 REM****APPLE THIEF****
20 REM*****C.GIFFORD*****
30 X=32:A=10:G=20
40 A$=CHR$(214)+CHR$(220)+CHR$(217)
50 B$=CHR$(223)+CHR$(223)+CHR$(223)
60 CLS:FOR P=1 TO 15
70 PRINT">>>>>>>>>>>APPLE THIEF<<<<<<<<<"
80 NEXT P
90 FOR T=1 TO 1000:NEXT T
100 CLS 6
110 REM***MAIN LOOP***
120 PRINT @ 0,"APPLES LEFT";A
130 PRINT @ 16,"APPLES CAUGHT ;F
140 PRINT @ R+480,A$;
150 PRINT @ X,"@";
160 IF A=5 THEN G=10
170 FOR T=1 TO G:NEXT T
180 PRINT @ X," ";
190 IF PEEK(345)=223 THEN GOSUB 230
200 X=X+1
210 IF X=64 OR T=1 THEN X=32:T=0:PRINT @
 R+480,B$;:R=RND(28)
220 GOTO 110
230 REM***APPLE DROP***
240 FOR B=X TO 448+(X/2)-1 STEP 32
250 PRINT @ B,"@";
260 PRINT @ 32," "
270 PLAY"O4L255CD"
280 PRINT @ B,CHR$(223);
290 IF PEEK(B+1056)=220 THEN GOSLIB 360
300 NEXT B
310 X=32:T=1
320 REM***APPLE MISSED***
```

```
330 A=A-1:IF A=0 THEN GOTO 410
340 SOUND 1,6
350 RETURN
360 REM***APPLE CAUGHT***
370 T=1
380 F=F+1:A=A-1:IF A=0 THEN GOTO 410
390 SOUND 190,2:SOUND 200,1
400 GOTO 110
410 CLS:PRINT @ 197,"YOU COLLECTED";F;"A
PPLES";
420 PRINT @ 262,"TO PLAY AGAIN PRESS 'A'
";
430 PLAY"L10004CDEFGAB"
440 IF INKEY$="A" THEN RUN ELSE GOTO 430
```



ALPHAMANIA

All you have to do is repeat the sequence of letters — but it's not as easy as it sounds. There is only a short amount of time in which to enter the letters. You will be rewarded with either CORRECT, WRONG or, if you are too slow, TIME UP. Choose the number of letters you want to repeat and fire away. But remember always to press ENTER after you type in the combination of letters.

```
10 REM*****ALPHAMANIA*****
20 REM*****C.GIFFORD******
30 CLS
40 INPUT"HOW MANY LETTERS";N
50 IF N<2 OR N>12 THEN GOTO 40
60 DIM G$(12)
70 DIM W$(12)
80 J=0
90 GOSUB 270
100 PRINT:PRINT"STAND BY......":FOR T=1
TO 500:NEXT T
110 PRINT:PRINT:PRINT:PRINT"              ";
120 FOR S=1 TO N
130 PRINT W$(S);
140 NEXT S
150 SOUND 180,2
160 TIMER=0
170 INPUT G$
180 IF TIMER>N*40 THEN GOTO 230
190 FOR C=1 TO N
200 IF MID$(G$,C,1)=W$(C) THEN J=J+1 ELS
E GOTO 370
210 NEXT C
220 IF J=N THEN GOTO 400
```

```
230 CLS 4:PRINT @ 240,"timeup";
240 SOUND 1,8
250 GOTO 250
260 STOP
270 FOR B=1 TO N
280 FOR A=1 TO RND(26)
290 READ W$(B)
300 NEXT A
310 RESTORE
320 NEXT B
330 DATA "A","B","C","D","E","F","G","H"
,"I","J","K","L"
340 DATA "M","N","O","P","Q","R","S","T"
,"U","V","W","X","Y","Z"
350 RETURN
360 PRINT TIMER:STOP
370 CLS 3:PRINT @ 240,"wrong";
380 SOUND 1,8
390 GOTO 390
400 CLS 2
410 FOR X=1 TO 14
420 PRINT @ X*32+X,"CORRECT";
430 PLAY"O4L12DGAGAO1CCCO4AGADCDL6C"
440 NEXT X
450 END
```

ROULETTE

Your Dragon takes you to the gaming tables of Monte Carlo where you're given $100 to try your luck at roulette. You have a choice of three types of bet and are given details on the odds on each type of bet. The game contains all the instructions and prompts needed — so go ahead and gamble!

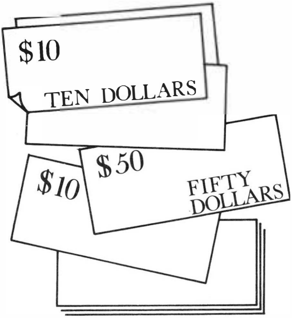
NB This program contains POKE 65495,0: read the comments on using this in the notes for Beat the Blues.

```
10 REM****ROULETTE****
20 REM****C.GIFFORD****
30 T=0:M=0:X=100
40 CLS
```

```
50 PRINT @ 0,"**************************
******"
60 PRINT @ 40,"BETTING TABLE"
70 PRINT @ 66,"SINGLE NO. ENTER N.ODDS 3
6/1"
80 PRINT @ 98,"DOUBLE NO. ENTER D.ODDS 1
8/1"
90 PRINT @ 130,"ODD/EVEN.  ENTER H.ODDS
1/2"
100 PRINT @ 160,"***********************
*********"
110 X=X+T
120 IF X<=0 THEN GOTO 630
130 PRINT @ 198,"*MONEY LEFT $";X;"*"
140 PRINT @ 224,"***********************
*********"
150 T=0
160 PRINT"ENTER SIZE OF BET"
170 INPUT M
180 IF M>X OR M<=0 THEN GOTO 170
190 PRINT"ENTER BETTING CODE"
200 INPUT C$
210 IF C$="N" THEN INPUT"WHAT NUMBER";A
220 IF C$="D" THEN INPUT"NUMBER ONE";A:I
NPUT"NUMBER TWO";B
230 IF C$="H" THEN INPUT"ODDS(O) OR EVEN
S(E)";A$
240 W=RND(37)-1
250 GOSUB 410
260 PRINT @200,">>NUMBER IS";W"<<";
270 PLAY"L2O3CDP4"
280 IF C$="N" AND A=W THEN T=M*36:GOTO 3
50
290 IF C$="D" AND A=W OR B=W THEN T=M*18
```

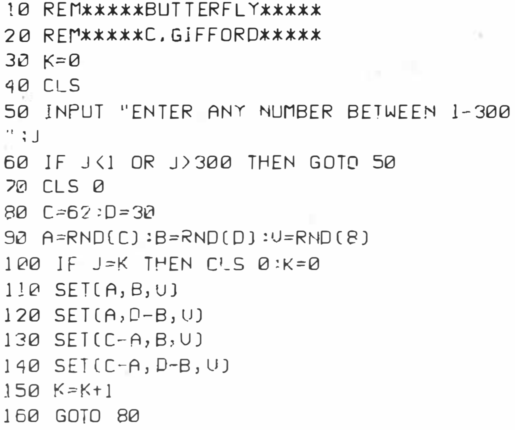
```
:GOTO 350
300 IF C$="H" AND A$="E" AND W/2=INT(W/2
) THEN T=M/2:GOTO 350
310 IF C$="H" AND A$="O" AND W/2<>INT(W/
2) THEN T=M/2:GOTO 350
320 PLAY"O1L100DECDECDECDECDECDEC"
330 X=X-M
340 GOTO 40
350 CLS 8
360 PRINT @ 198,"YOU HAVE WON $";T;
370 PLAY"L15O3GG#AO4FP15O3AO4FP15O3AO4FP
5"
380 PLAY"FGG#AFGAP15EGP15FP5"
390 GOTO 40
400 STOP
410 PMODE 4,1:PCLS:SCREEN 1,1
420 Z=75
430 POKE 65495,0
440 V=85
450 FOR A=150 TO 20 STEP -10
460 CIRCLE(128,96),70
470 CIRCLE(128,96),15
480 PAINT(128,96),1,1
490 LINE(76,96)-(180,96),PSET
500 LINE(128,44)-(128,148),PSET
510 LINE(93,61)-(163,131),PSET
520 LINE(163,61)-(93,131),PSET
530 CIRCLE(Z,V),5
540 SOUND A,1
550 CIRCLE(128,96),70,0
560 CIRCLE(Z,V),5,0
570 V=V-1
580 Z=Z+3
590 NEXT A
```

```
600 POKE 65494,0
610 CLS
620 RETURN
630 CLS 0
640 PRINT @ 262,"YOU'RE BROKE!!";
650 PLAY"L100CDEFGABP80AGFEDC"
660 GOTO 650
```



BUTTERFLY

This program produces a pleasing symmetrical pattern using the SET command. A small number gives a fast-moving display while a larger number gives a slower but see prettier possible pattern. Try altering this program many in any way you see possible. It's easy to achieve many attractive displays with just simple program changes.



```
10 REM*****BUTTERFLY*****
20 REM*****C.GIFFORD*****
30 K=0
40 CLS
50 INPUT "ENTER ANY NUMBER BETWEEN 1-300
";J
60 IF J<1 OR J>300 THEN GOTO 50
70 CLS 0
80 C=62:D=30
90 A=RND(C):B=RND(D):V=RND(8)
100 IF J=K THEN CLS 0:K=0
110 SET(A,B,V)
120 SET(A,D-B,V)
130 SET(C-A,B,V)
140 SET(C-A,D-B,V)
150 K=K+1
160 GOTO 80
```

MINE ESCAPE

You are on the bottom level of a deep mine and the air supply is fast running out. Can you escape in time or will you perish as many others have before you? This fast-moving game is great fun to play. Control your man with the four arrow keys and try to reach the exit in the top, left-hand corner (in magenta). Should you find yourself totally stuck, use one of your five sticks of dynamite to blow away the rocks in the immediate vicinity. Laying the dynamite (pressing the space bar) uses up time, but may prove the only way out. After the blast some rocks may go molten (orange), but you can still walk through them.

Your time is displayed at the top of the screen: how long you have to complete the maze depends on which level you choose (level five being hardest) If you fall, the computer will play some appropriate music.

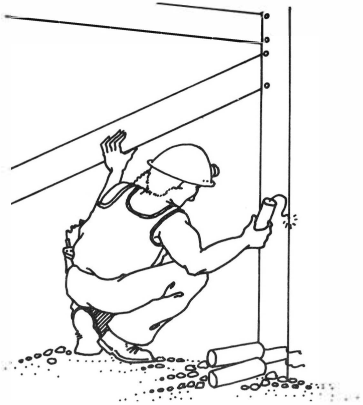
The short routine from line 100 to line 130 prints the maze. This little routine ensures that the maze is always different, thus making the game more interesting.

To restart the game press BREAK, retype RUN and press ENTER.

```
10 REM****MINE ESCAPE****
20 REM****C.GIFFORD****
30 CLS
40 PRINT @ 324,"INPUT SKILL LEVEL: 1-5"
50 INPUT S
60 IF S<1 OR S>5 THEN GOTO 50
70 X=62:Y=30
80 F=0
90 REM*****DRAW RANDOM MAZE*****
100 FOR M=1 TO 510
110 R=RND(3)
120 IF R=3 THEN PRINT CHR$(175);ELSE PRI
NT CHR$(128);
130 NEXT M
140 TIMER=0
150 REM***MAIN GAME***
160 PRINT @ 0,"+++MINE ESCAPE++++TIME:";
TIMER;"++"
170 PRINT @ 33,CHR$(227);
180 IF TIMER>2250-(250*S) THEN GOTO 620
190 SET(X,Y,2)
200 PLAY"O5L255DE"
210 RESET(X,Y)
220 IF PEEK(341)=223 THEN GOSUB 290
230 IF PEEK(342)=223 THEN GOSUB 330
240 IF PEEK(343)=223 THEN GOSUB 370
250 IF PEEK(344)=223 THEN GOSUB 410
260 IF PEEK(345)=223 THEN GOSUB 450
270 IF POINT(X,Y)=7 THEN GOTO 530
280 GOTO 160
290 Y=Y-1
300 IF POINT(X,Y)=3 THEN Y=Y+1
310 IF Y=0 THEN Y=1
320 RETURN
```

```
330 Y=Y+1
340 IF Y=31 THEN Y=30
350 IF POINT(X,Y)=3 THEN Y=Y-1
360 RETURN
370 X=X-1
380 IF X=0 THEN X=1
390 IF POINT(X,Y)=3 THEN X=X+1
400 RETURN
410 X=X+1
420 IF X=63 THEN X=62
430 IF POINT(X,Y)=3 THEN X=X-1
440 RETURN
450 REM***FIRE***
460 F=F+1:IF F>5 THEN PLAY"O1L30EEEGGGCC
C":RETURN
470 RESET(X-1,Y-1):RESET(X,Y-1):RESET(X+
1,Y-1)
480 RESET(X-1,Y):RESET(X+1,Y):RESET(X-1,
Y+1)
490 RESET(X,Y+1):RESET(X+1,Y+1)
500 PLAY"O2L30CGCGCGCGCG"
510 RETURN
520 REM****WIN ROUTINE****
530 D=TIMER
540 FOR S=1 TO 8
550 CLS S
560 PLAY"L100O4EEEEEEEEBBBBBBBEEEEEEEEB
BBBBBBB"
570 NEXT S
580 PRINT @ 168,"YOU’VE MADE IT!!";
590 PRINT @ 199,"IN JUST";D;"SECS!";
600 GOTO 600
610 REM****LOSE ROUTINE****
620 PMODE 4,1:PCLS:SCREEN 1,1
630 DRAW"BM68,160;R20U80R40U20L40U40L20D
```

```
40L40D20R40D80"
640 PAINT(78,150),1,1
650 PLAY"L4O1CL2AL10BL8GO2L2C#O1L10BL8GL
2FL4EL2F"
660 PLAY"L4O1AAGGL2BL4BO2L2C"
670 DRAW" BM160,120;S8U8R4D4L4F4BR4R2U1L
2BR6BD1R6BL3U8L3R6BD8BR4R2U1L2"
680 DRAW"BR6BD1U8R4D4L4BD4BR6R2U1L2"
690 PLAY"O1L2EDC"
700 GOTO 700
```

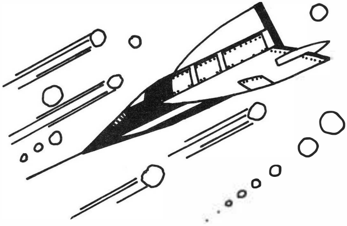


MINI TREK

In this game you must scour the galaxy to find your home base. Along the way you will encounter many hazards. Black holes, supernovae and numerous malevolent aliens are out to get you; but you do have some assistance, in the form of fuel pods, starbases and the magic capsule that gives details of your home base.

You are within an eight by eight grid, and any attempt to enter an incorrect move will be stopped. Every move you make uses 50 energy units; when you reach zero energy you are vaporised. If there is an alien in the same sector as you, you can either run or fight: enter R if you wish to run, and any other letter if you wish to fight. When choosing your weapon, enter the number to the left of the weapon name (eg for Dragon Beam enter three). If you are vaporised, press the BREAK key to stop the program.

Good luck and happy trekking!



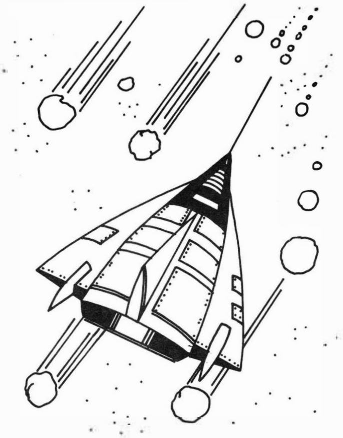
```
10 REM*********************
20 REM*****MINI-TREK******
30 REM*****C.GIFFORD******
40 REM*********************
60 K=0
70 E=1000
80 PMODE 4,1:PCLS:SCREEN 1,1
90 DRAW"BM40,130;U20F12E12D20BR10U20BR10
D20BU20F20U20BR10D20"
100 DRAW"BR30U20L10R20BR10D20U10F10BU10U
10L10D10R10BR10U10D20R12L12U10R10L10U10R
12"
110 DRAW"BR10D20U10E10BD20H10"
120 FOR P=1 TO 100
130 G=RND(255):H=RND(191)
140 PSET(G,H,1)
150 NEXT P
160 CIRCLE(210,50),15
170 CIRCLE(210,50),25,1,0.4
180 CIRCLE(210,50),28,1,0.4
190 A=RND(8):B=RND(8)
200 FOR T=1 TO 1000:NEXT T
210 PLAY STR$(RND(12))
220 CLS
230 REM
240 PRINT CHR$(159)+CHR$(159);" ENERGY U
NITS";E;CHR$(159);" MOVE";K;CHR$(159)+CH
R$(159)
250 PRINT:PRINT"WARP TO WHERE?"
260 INPUT X,Y
270 IF X<1 OR X>8 OR Y<1 OR Y>8 THEN PRI
NT"'DUFF'MOVE GUV!'TRY AGAIN":GOTO 260
280 K=K+1
290 E=E-50:IF E<1 THEN GOTO 950
```

```
300 IF X=A AND Y=B THEN GOTO 470
310 R=RND(9):IF R>6 THEN GOSUB 630
320 REM
330 FOR Z=1 TO RND(22)
340 READ ZZ$
350 NEXT Z
360 DATA"A BLACK HOLE","A CAPSULE","A SU
PERNOVA","AN ASTEROID BELT","A FUEL POD"
,"A STARBASE"
370 DATA"A SUPERNOVA","NOTHING AT ALL","
A TINY PLANET"," SPACE DEBRIS","A DISUSE
D CRAFT","NOTHING","LITTLE OF VALUE"
380 DATA"SOME SMALL STARS","A STARBASE",
"A STARBASE","A SPINNING ORB","A PASSING
 COMET","A CRUISING SHIP","NIL","A FUEL
POD","------"390 PRINT:PRINT"SECTOR CONTA
INS ";ZZ$
400 IF ZZ$="A FUEL POD" THEN PRINT"*****
extra energy units*****":E=E+500:SOUND 1
80,6
410 IF ZZ$="A SUPERNOVA" OR ZZ$="A BLACK
 HOLE" THEN GOTO 950
420 IF ZZ$="A CAPSULE" THEN PRINT"WHICH
GIVES YOU YOUR HOMEBASE COORDINATES":PRI
NT:PRINT A;",";B
430 IF ZZ$="A STARBASE" THEN PRINT"ENERG
Y SUPPLIES ARE REPLENISHED ALL DAMAGES A
RE REPAIRED.LEAVE  NOW AND SEARCH FOR HO
MEBASE":E=1000:FOR T=1 TO 800:NEXT T
440 IF ZZ$="A SPINNING ORB" THEN PLAY"L8
002C04C02C04C02C04C02C04C"
450 RESTORE
460 GOTO 200
470 PMODE 4,1:PCLS:SCREEN 1,1
480 CIRCLE(128,96),70
```

```
490 CIRCLE(128,96),45,1,1.5
500  CIRCLE(128,96),14,1,5
510 PAINT(78,96),1,1
520 PAINT(180,96),1,1
530 PAINT(128,96),1,1
540 FOR T=1 TO 50:G=RND(255):H=RND(191)
550 PSET(G,H,1)
560 NEXT T
570 FOR E=1 TO 10
580 DRAW"BM20,20U10D5R8U5D10BR5U10R10D10
L10BR15U10F5E5D10BR5R10BL10U5R8BL8U5R10"
590 PLAY"L100O4CDECDECDECDE"
600 LINE(20,10)-(80,40),PRESET,BF
610 NEXT E
620 END
630 PRINT"SECTOR CONTAINS ";
640 PLAY"V31L20O3BCBCBCBC"
650 PRINT"aliens"
660 PRINT:PRINT"DO YOU WANT TO FIGHT OR
RUN?"
670 INPUT A$:IF A$="R" THEN GOTO 840
680 PRINT"BRAVE MAN...CHOOSE YOUR WEAPON
;":PRINT
690 PRINT"1*PHASERS------NO UNITS
  2*PLASMA BOLT--NO UNITS           3*DRAG
ON BEAM--400 UNITS"
700 INPUT W:IF W>3 OR W<1 THEN GOTO 700
710 IF W=3 THEN E=E-400:GOTO 810
720 PRINT"THE FIGHT BEGINS.TO FIRE,PRESS
  ENTER"
730 INPUT N$:PLAY"L200EDEFEDEFEDEFEDEFED
EFEDEFEDEF"
740 M=RND(11):IF M>5 THEN GOTO 810
750 E=E-800:IF E<1 THEN GOTO 950
760 PRINT"YOU LOST AND ARE BADLY DAMAGED
```

```
."
770 SOUND 1,4
780 PRINT"YOU MUST GET TO A STARBASE FAS
T OR IT'S CURTAINS FOR YOU."
790 FOR T=1 TO 1000:NEXT T
800 GOTO 200
810 PLAY"L4V25O3ABGO2GO3L2D"
820 PRINT"           ";CHR$(255)+CHR$(255);"S
UCCESS";CHR$(255)+CHR$(255)
830 GOTO 200
840 PRINT"++++++CHICKEN++++++"
850 PLAY"L50O1CGC"
860 PRINT"PRESS HYPERSPACE BUTTON.(H)"
870 IF INKEY$="H" THEN GOSUB 890 ELSE GO
TO 870
880 END
890 X=RND(8):B=RND(8)
900 PRINT"YOU HAVE BEEN RANDOMLY PLACED
  SOMEWHERE IN THE UNIVERSE AND   HAVE L
OST 200 ENERGY UNITS."
910 E=E-200:IF E<1 THEN GOTO 950
920 PLAY"L80CGCGCGCGCGCG"
930 FOR T=1 TO 1000:NEXT T
940 GOTO 200
950 FOR T=1 TO 1000:NEXT T
960 PMODE 3,1:PCLS:SCREEN 1,0
970 FOR R=1 TO 80 STEP 4
980 CIRCLE(128,96),R,3,0.75
990 NEXT R
1000 PLAY"L20O1CDE"
1010 FOR R=0 TO 79 STEP 5
1020 CIRCLE(128,96),R,4,0.85
1030 NEXT R
1040 PLAY"L20EFG"
```

```
1050 FOR R=2 TO 81 STEP 5
1060 CIRCLE(128,96),R,2,0.5
1070 NEXT R
1080 PLAY"L20GABBAB"
1090 CLS0:PRINT @ 196,"YOUVE BEEN VAPOUR
IZED";
1100 SOUND 1,12
1110 GOTO 960
```



CREATION

With this program you have the power to create colonies of cells which you can watch thriving or dying, genera-tion by generation. It is an adaptation of Chris Palmer's version of Life, a form of game originally invented by John Conway in 1970.

Enter the coordinates of where you want a cell to start. By entering the coordinates of between 5 and 20 cells you can form a small colony. Once you have the start cells, enter O for the X coordinate and the program will start evolving the cell pattern. Each generation takes some time to appear as there is much calculation involved. After a new generation appears you can either stop the game or continue to a new generation.

Start with a symmetrical cell pattern, then try a few odd shapes. Always have more than four or five cells at the start or your colony will only last a few generations Try to create a cell pattern that lasts for many generations.

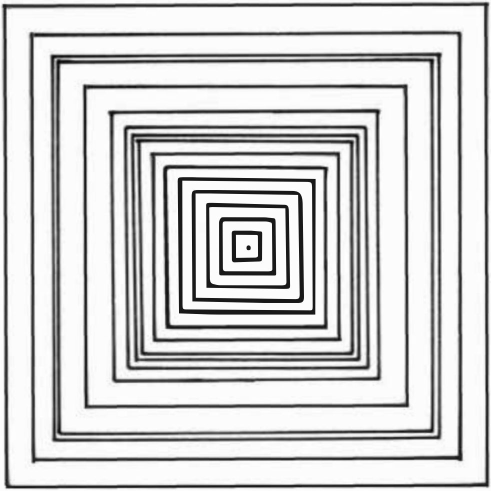
As a programming exercise alter this program so it continues without you pressing C. You could then go off and have a cup of tea while a whole new life form generates on your TV screen!

```
10 REM*****CREATION******
20 REM*****C.GIFFORD*****
30 REM*******************
40 CLS
50 DIM M(1,10,10)
60 G=0
70 GOSUB 190
80 INPUT"ACROSS";X:IF X=0 THEN 120
90 IF X<1 OR X>10 THEN 80
100 INPUT"DOWN";Y:IF Y<1 OR Y>10
THEN 100
```

```
110 M(1,X,Y)=ABS(M(1,X,Y)-1):GOTO 70
120 FOR Y=1 TO 9: FOR X=1 TO 9:C=M(0,X-1
,Y-1)+M(0,X-1,Y)+M(0,X-1,Y+1)+M(0,X,Y-1)
+M(0,X,Y+1)+M(0,X+1,Y-1)+M(0,X+1,Y)+M(0,
X+1,Y+1)
130 IF C=3 THEN M(1,X,Y)=1
140 IF C<2 OR C>3 THEN M(1,X,Y)=0
150 NEXT:NEXT
160 G=G+1: GOSUB 190
170 GOTO 120
180 END
190 IF G<2 THEN CLS
200 PRINT @ 12, "**CREATION**"
210 IF G>0 THEN PRINT @ 44,"*GENERATION*
";G
220 PRINT"123456789    ************"
230 FOR Y=1 TO 9: X$=RIGHT$(STR$(Y),1)
240 FOR X=1 TO 9:IF M(1,X,Y)=1 THEN PRIN
T "O";:GOTO 260
250 PRINT".";
260 M(0,X,Y)=M(1,X,Y):NEXT:PRINT X$:NEXT
270 PRINT"123456789"
280 PLAY"L14002CEFCEFCEFCEF"
290 IF G<1 THEN RETURN
300 PRINT @ 448,"PRESS: S-STOP, C-CONTIN
UE."
310 A$=INKEY$
320 IF A$="S" THEN END
330 IF A$="C" THEN SOUND 60,1: RETURN
340 GOTO 310
```

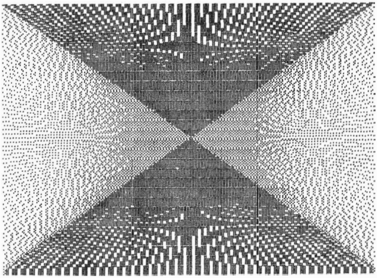
MOIRE VARIATIONS

The pattern, Moire, is widely-known and available on many computers. Here is a slightly different version, based on an original idea by Sally Gifford.



MOIRE

```
10 REM***MOIRE VARATIONS***
20 REM**ORIGINAL IDEA:SALLY GIFFORD**
30 N=0
40 PMODE 3,1:PCLS:SCREEN 1,N
50 FOR A=0 TO 255 STEP 3
60 LINE(128,96)-(0,A),PSET
70 COLOR 3,1
80 LINE(128,96)-(A,0),PSET
90 LINE(128,96)-(A,255),PSET
100 COLOR 4,1
110 LINE(128,96)-(255,A),PSET
120 NEXT A
130 FOR T=1 TO 2500:NEXT T
140 N=N+1:IF N>1 THEN N=0
150 GOTO 40
```



LUMBERJACK

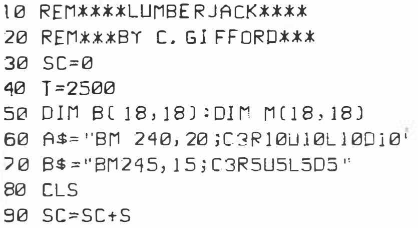
This game takes you to the forests of British Columbia and your task is to chop down as many trees as possible. You control your lumberjack with the arrow keys and his axe must hit the tree first. Chopping down the trees is not always straightforward: there are also several dead trees worth no points, even if you cut them down successfully. You will hear a tone for every living tree that you cut down, and score 100. If you attain 500 points in one go then two things happen:

1. You get a chance to fell the Giant Redwood (which must be hit three times but is worth 500 points if successful)

2. When your time runs out you receive another screenful of forest - but the time allowed decreases slightly for every new forest area.

3. The blue square that appears in the top, right-hand cor-ner tells you that approximately a quarter of your time is up, while the smaller, red square, that appears within the blue one, tells you there is little time left

This program takes some time to master, particularly in the technique needed for chopping down trees. But after some time you should be able to achieve a high score - try to beat my high score of 16,700.



```
10 REM****LUMBERJACK****
20 REM***BY C. GIFFORD***
30 SC=0
40 T=2500
50 DIM B(18,18):DIM M(18,18)
60 A$="BM 240,20;C3R10U10L10D10'
70 B$="BM245,15;C3R5U5L5D5"
80 CLS
90 SC=SC+S
```

```
100 PRINT @ 197,"SCORE SO FAR";SC
110 FOR L=1 TO 10
120 PLAY"L150O4CGO3CGO2CGO1CGO4CGO3CGO2C
GO1CG"
130 NEXT L
140 S=0
150 N=0
160 PMODE 3,1:PCLS 2:SCREEN 1,0
170 GET(30,30)-(48,48),B
180 FOR TREE=1 TO 14
190 X=RND(220)+20:Y=RND(160)+20
200 IF PPOINT(X,Y)=1 THEN GOTO 190
210 IF A>110 AND A<140 AND B>80 AND B<11
0 THEN GOTO190
220 COLOR 1,2
230 LINE(X,Y)-(X,Y+12),PSET
240 LINE(X,Y+1)-(X-6,Y+6),PSET
250 LINE(X,Y+1)-(X+6,Y+6),PSET
260 NEXT TREE
270 DRAW"BM120,90;S6C3R3D3L3U3BR2BD3D2BR
4L8BR4D2G4BR8H4BR4U8D1R4D2L4"
280 GET(118,88)-(136,106),M
290 A=118:B=88
300 TIMER=0
310 REM
320 IF TIMER>750 THEN DRAW A$
330 IF TIMER>1500 THEN DRAW B$
340 IF TIMER>T AND S<500 THEN GOTO 580
350 IF TIMER>T AND S>500 THEN T=T-250:GO
TO 80
360 IF PPOINT(A+18,B+4)=1 OR PPOINT(A+18
,B)=1 THEN GOSUB 490
370 IF A<5 THEN A=5
380 IF A>232 THEN A=232
```

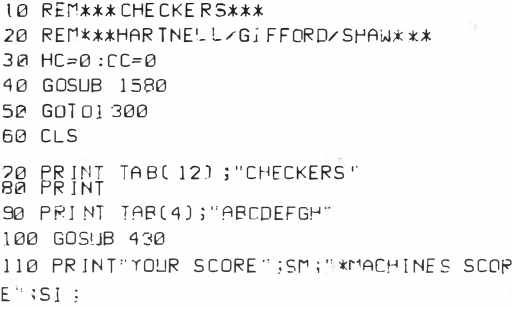
```
390 IF B<5 THEN B=5
400 IF B>172 THEN B=172
410 IF PPOINT(A+18,B+2)=4 THEN GOSUB 550
420 PUT(A,B)-(A+18,B+18),M
430 PUT(A,B)-(A+18,B+18),B
440 IF PEEK(344)=223 THEN A=A+5
450 IF PEEK(343)=223 THEN A=A-5
460 IF PEEK(342)=223 THEN B=B+5
470 IF PEEK(341)=223 THEN B=B-5
480 GOTO 310
490 REM
500 PUT(A+18,B-10)-(A+26,B+10),B
510 PLAY"L250EFEFEFEFEFEFEFEFFFF"
520 S=S+100
530 IF S=500 THEN DRAW"BM220,70;C4U20D1G
4BR8H4D4G6BR12H6D4G8BR16H8"
540 RETURN
550 N=N+1:SOUND N*60,4
560 IF N=3 THEN PLAY"V31L40O3GCO2GCO1GCC
":S=S+500:GOTO 80
570 RETURN
580 CLS 0
590 PRINT @ 198,"YOU LOST!!";
600 PRINT @ 228,"YOU SCORED";SC+S;
610 GOTO 610
```

CHECKERS

In this computerized version of the classic board game (also known as Draughts) the computer plays an attacking game.

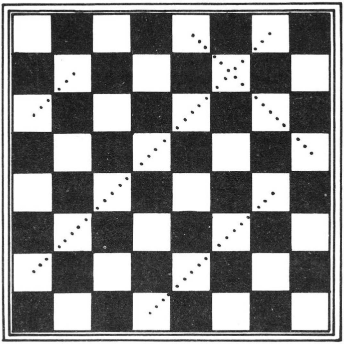
For those unfamiliar with this game, it is played on an eight by eight board, each player having 12 pieces. Play is confined to the black squares and all moves are diagonal. You move 'up' the board and the computer moves 'down' — until you get a king, which can move any way, providing it is diagonal and lands on a black square. You take any pieces that you jump over, and you cannot jump over your own pieces.

When you run the program the computer asks if you want to go first. After you enter either Y or N, the board is displayed. Your pieces are the Xs and the computer's are Os. To make a move, enter the letter and number of the square you want to move from, then enter the number and letter of the square you want to move to. If you make a conquest your score is updated and the computer's piece disappears. There are no multiple jumps, but there are kings (inverse Xs and Os respectively).



```
10 REM***CHECKERS***
20 REM***HARTNELL/GIFFORD/SHAW***
30 HC=0:CC=0
40 GOSUB 1580
50 GOTO1300
60 CLS
70 PRINT TAB(12);"CHECKERS"
80 PRINT
90 PRINT TAB(4);"ABCDEFGH"
100 GOSUB 430
110 PRINT"YOUR SCORE";SM;"*MACHINES SCOR
E";SI;
```

```
120 PLAY"O3L24CDEFEDC"
130 PRINT
140 IF U$="N" THEN GOTO 670
150 IF SI=12 THEN PRINT"I WIN":STOP
160 IF SM=12 THEN PRINT"YOU WIN":STOP
170 IF Q=2 THEN GOTO 390
180 PRINT"LAST TO ";F$;
190 INPUT "      FROM..";C$
200 INPUT"                 TO..";B$
210 F$=B$
220 D$=C$
230 GOSUB 1510
240 C=M(I)
250 IF I=0 THEN GOTO 190
260 D$=B$
270 GOSUB 1510
```



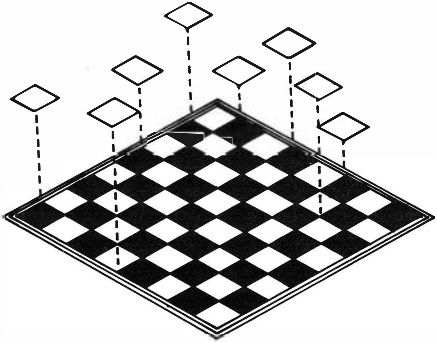
```
280 B=M(I)
290 IF I=0 THEN GOTO 190
300 U$="N"
310 IF B-C=10 THEN A(B-5)=0
320 IF B-C=8 THEN A(B-4)=0
330 IF C-B=10 THEN  A(C-5)=0
340 IF C-B=8 THEN A(C-4)=0
350 A(B)=A(C)
360 A(C)=0
370 Q=2
380 GOTO 60
390 U$="":Q=0
400 IF ABS(C-B)=10 OR ABS(C-B)=8 THEN PR
INT:PRINT:INPUT U$
410 IF U$<>"Y" THEN GOTO 670
420 GOTO 60
430 REM
440 M1=0:K=1
450 J=-1
460 HC=0:CC=0
470 FOR I=40 TO 6 STEP-1
480 IF A(I)=1 AND I>37 THEN A(I)=2
490 IF A(I)=-1 AND I<10 THEN A(I)=-2
500 IF I=14 OR I=32 OR I=23 THEN GOTO 56
0
510 IF M1=0 THEN PRINT K;"  ";:M1=0:K=K+1
:J=-1*J:IF J=1 THEN PRINTCHR$(128);
520 A=A(I)
530 GOSUB 600
540 IF M1<>3 OR J=-1 THEN PRINT CHR$(128
);
550 M1=M1+1:IF M1>3 THEN M1=0:PRINT
560 NEXT I
570 SI=12-HC:SM=12-CC
```

```
580 PRINT
590 RETURN
600 REM**PRINT PIECES**
610 IF A=0 THEN PRINT" ";
620 IF A=1 THEN PRINT"X";:HC=HC+1
630 IF A=-1 THEN PRINT"O";:CC=CC+1
640 IF A=-2 THEN PRINT"o";:CC=CC+1
650 IF A=2 THEN PRINT"x";:HC=HC+1
660 RETURN
670 U$="  ":Q=0
680 Z=6
690 IF Z<9 THEN GOTO 730
700 IF A(Z)<0 AND (A(Z-4)=1 OR A(Z-4)=2)
 AND A(Z-8)=0 THEN GOTO 920
710 IF Z<11 THEN GOTO 730
720 IF A(Z)<0 AND (A(Z-5)=1 OR A(Z-5)=2)
 AND A(Z-10)=0 THEN GOTO 1010
730 IF Z>25 THEN GOTO 760
740 IF A(Z)=-2 AND (A(Z+4)=1 OR A(Z+4)=2
) AND A(Z+8)=0 THEN GOTO 1110
750 IF A(Z)=-2 AND (A(Z+5)=1 OR A(Z+5)=2
) AND A(Z+10)=0 THEN GOTO 1210
760 Z=Z+1:IF Z<=40 THEN GOTO 690
770 REM**RANDOM**
780 U=0
790 Z=RND(34)+6
800 K=0
810 U=U+1
820 IF A(Z)<0 AND A(Z-4)=0 THEN K=1
830 IF A(Z)<0 AND A(Z-5)=0 AND K=0 THEN
K=2
840 IF K=0 AND Z<26 AND A(Z)=-2 AND A(Z+
4)=0 THEN K=-7
850 IF Z<10 THEN GOTO 870
```

```
860 IF (K=1 OR K=2) AND U<200 AND (A(Z-(
10 AND Z>10))=1 OR A(Z-8)=1) THEN GOTO 7
90
870 IF K=0 AND U<400 THEN GOTO 790
880 IF K=0 THEN SM=12:GOTO 60
890 A(Z-(3+K))=A(Z)
900 A(Z)=0
910 GOTO 60
920 A(Z-8)=A(Z)
930 A(Z)=0
940 A(Z-4)=0
950 IF Z<24 THEN GOTO 60
960 IF (A(Z-13)-1 OR A(Z-13)=2) AND A(Z-
18)=0 THEN P=2
970 IF P=1 THEN A(Z-18)=A(Z-8):A(Z-13)=0
0
980 IF P=2 THEN A(Z-12)=0:A(Z-12)=0
990 IF P>0 THEN A(Z-8)=0
1000 GOTO 60
1010 A(Z-10)=A(Z)
1020 A(Z)=0
1030 A(Z-5)=0
1040 IF Z<25 THEN GOTO 60
1050 IF (A(Z-15)=1 OR A(Z-15)=2) AND A(Z
-20)=0 THEN P=1
1060 IF (A(Z-14)=1 OR A(Z-14)=2) AND A(Z
-18)=0 THEN P=2
1070 IF P=1 THEN A(Z-15)=0:A(Z-20)=A(Z-1
0)
1080 IF P=2 THEN A(Z-14)=0:A(Z-18)=A(Z-1
0)
1090 IF P>0 THEN A(Z-10)=0
1100 GOTO 60
1110 A(Z+8)=-2
```

```
1120 A(Z+4)=0
1130 A(Z)=0
1140 IF Z<32 AND (A(Z+3)=1 OR A(Z+3)=2)
AND A(Z-2)=0 THEN P=1
1150 IF Z<23 AND (A(Z+14)=1 OR A(Z+14)=2
) AND A(Z+16)=2 THEN P=2
1160 IF Z<23 AND (A(Z+13)=1 OR A(Z+13)=2
) AND A(Z+18)=0 THEN P=3
1170 IF P=1 THEN A(Z+3)=0:A(Z-2)=-2
1180 IF P=2 THEN A(Z+14)=0:A(Z+16)=0
1190 IF P=3 THEN A(Z+13)=0:A(Z+18)=-2
1200 IF P>0 THEN A(Z+8)=0
1210 LET A(Z+10)=-2
1220 A(Z+5)=0
1230 A(Z)=0
1240 SI=SI+1
1250 GOTO 60
1260 PRINT:PRINT
1270 PRINT:PRINT
1280 RETURN
1290 REM**INITIALISE**
1300 DIM A(45)
1310 PRINT
1320 FOR Z=1 TO 45
1330 IF Z<6 THEN A(Z)=9
1340 IF Z>5 AND Z<19 THEN A(Z)=1
1350 IF Z>18 AND Z<28 THEN A(Z)=0
1360 IF Z>27 AND Z<41 THEN A(Z)=-1
1370 IF Z>40 THEN A(Z)=9
1380 NEXT Z
1390 A(14)=9:A(23)=9:A(32)=9
1400 F$=".-- "
1410 P=0:Q=0
1420 SI=0:SM=0
```

```
1430 PRINT:PRINT
1440 INPUT"FIRST MOVE?(Y/N)";Q$
1450 U$=""
1460 PRINT
1470 IF Q$<>"Y" THEN GOTO 670
1480 U$=""
1490 PRINT
1500 GOTO 60
1510 REM**DECODE MOVE**
1520 I=1
1530 IF M$(I)=D$ THEN GOTO 1570
1540 I=I+1
1550 IF I=33 THEN I=0 :GOTO 1570
1560 GOTO 1530
1570 RETURN
1580 DIM M$(32):DIM M(32)
1590 FOR I=1 TO 32
1600 READ M$(I)
1610 NEXT I
1620 DATA "B1","D1","F1","H1","A2","C2",
"E2","G2","B3","D3","F3","H3","A4","C4",
"E4","G4","B5","D5","F5","H5","A6","C6",
"E6"
1630 DATA "G6","B7","D7","F7","H7","A8",
"C8","E8","G8"
1640 FOR I=1 TO 32
1650 READ M(I)
1660 NEXT I
1670 DATA 40,39,38,37,36,35,34,33,31,30
1680 DATA 29,28,27,26,25,24,22,21,20,19
1690 DATA 18,17,16,15,13,12,11,10,9,8,7,6
1700 RETURN
```

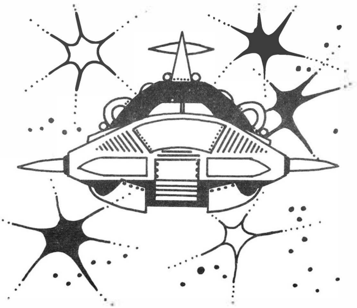


ZT GO HOME

You are the Commander of the Zeon Trekker which unfortunately is fast running out of energy. You must maneuverer your ship through the asteroid belt to reach the planet on the other side. There you can land and refuel safely.

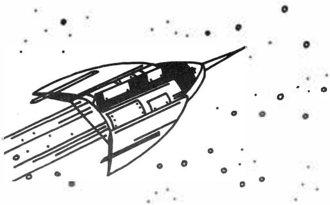
To steer your craftuse Z to go left and M to move right. You have a choice of warp speeds, O is the fastest and gets you home quicker, while 3 is the slowest and most maneuverable speed. You must avoid the asteroids (the yellow blobs), but if you pass very close to them you will unbalance them and they will turn supernova (red).

If you find the game too difficult to complete — though I assure you it can be done — then change line 150 to M = M + 2 instead of M = M + 1.



```
10 REM***Z.T.GO HOME**
20 REM****C.GIFFORD****
30 X=30:SC=0:M=0
40 CLS
50 INPUT"WARP SPEED.(0-3)";W
60 CLS 0
70 B=8:Y=0
80 FOR G=1 TO 28
90 A=RND(30)+15
100 B=RND(26)+5
110 SET(A+1,B,2):SET(A+2,B,2)
120 NEXT G
130 SET(X,Y,4)
140 FOR T=1 TO W*7:NEXT T
150 M=M+1:IF M=1000+(W*150) THEN GOTO 300
160 RESET(X,Y)
170 T=POINT(X,Y+1)
180 IF T<>0 THEN PLAY"L80O4CDEFGGFEDC":S
C=SC+1
190 IF SC=3 THEN GOTO 260
200 IF PEEK(340)=223 THEN X=X-1
210 IF PEEK(343)=247 THEN X=X+1
220 IF X<15 OR X>47 THEN SOUND 1,1:X=30
230 Y=Y+1
240 IF Y=31 THEN GOTO 60
250 GOTO 130
260 CLS:PRINT @ 194,"YOU TRAVELLED";M;"W
ARPS"
270 PRINT:PRINT"  BUT YOU FAILED TO GET
HOME"
280 PLAY"L10O1CDEFP10DEFGP10GFEEDCC"
290 END
300 PMODE 4,1:PCLS:SCREEN 1,1
```

```
310 CIRCLE(128,190),180,1,0.5
320 CIRCLE(158,140),50,1,0.3
330 CIRCLE(158,150),30,1,0.3,0.5,0
340 CIRCLE(30,170),50,1,0.3
350 CIRCLE(30,180),20,1,0.3
360 PAINT(125,145),1,1
370 PAINT(1,175),1,1
380 FOR S=1 TO 50:A=RND(250):B=RND(100)
390 PSET(A,B,1)
400 NEXT
410 LINE(59,130)-(65,140),PSET,BF
420 DRAW"BM59,130;E3F3BD10F6BL18E6"
430 PRESET(62,135)
440 PLAY"O3L18CDEFP10DEFGP10EFGAP10ABO4C
DDD"
450 GOTO 440
```



GUILLOTINE

This game is based on Hangman, with an added twist which you will see if you don't guess the word in time.

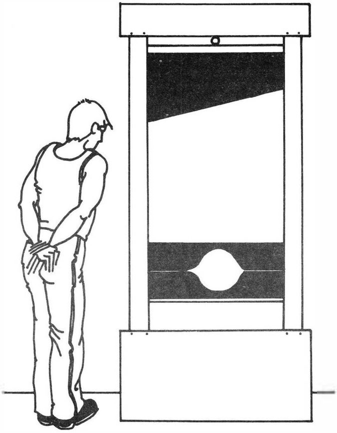
The game starts with a number of dashes on the screen. These dashes are the spaces for the letters which make up the mystery word. When you enter the letter you think is in the word, the computer will tell you if you are correct or not. Once you think you know what the word is, then type the whole word in. If you are correct, you have won; if you are wrong you lose 3 of your 10 guesses.

The program has quite a large vocabulary so you will find it difficult to remember the words even if you have just typed them in. To restart the game press BREAK, retype RUN and press ENTER.

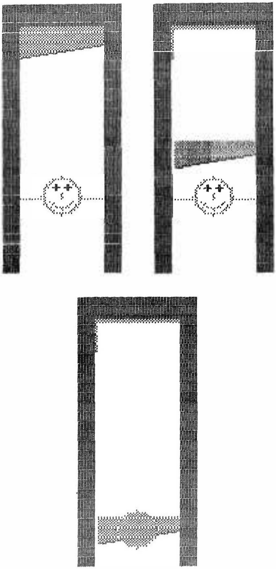
```
10 REM****GUILLOTINE****
20 CLS:PRINT @ 40,"GUILLOTINE"
30 GOSUB 460
40 L=LEN(W$)
50 FOR X=1 TO L:PRINT @ 232+X,"-";:SOUND
 X*15,4:NEXT X
60 PRINT @ 98,"WORD IS ";L;"LETTERS LONG";
70 FOR P=1 TO 10
80 PRINT @ 387,"ENTER YOUR GUESS.NUMBER";P
90 INPUT V$:SOUND 180,2:IF LEN(V$)=L THEN
GOSUB 360
100 G=INSTR(W$,V$):IF G<>0 THEN PRINT @
200+G,V$;
110 H=INSTR(G+1,W$,V$):IF H<>0 THEN PRIN
T @ 200+H,V$;
120 NEXT P
130 REM**LOSE ROUTINE**
```

```
140 DIM K(60,20):DIM 8(60,20)
150 PMODE 3,1:PCLS:SCREEN 1,0
160 CIRCLE(130,140),15,4
170 CIRCLE(135,134),2,3
180 CIRCLE(126,134),2,3
190 DRAW"BM130,138;D4BR13R25BL50L30"
200 CIRCLE(130,140),10,4,1,0.1,0.42
210 DRAW"BM160,195;C3U175L60D175L10U190R
80D190L10"
220 LINE(100,40)-(160,30),PSET
230 PAINT (120,10),3,3:PAINT(105,30),2,3
240 GET(110,20)-(160,40),K
250 FOR X=20 TO 140 STEP 20
260 PUT(108,X)-(160,X+20),K
270 SOUND 180-X,2
280 PUT(108,X)-(160,X+20),B
290 NEXT X
300 PUT (108,X)-(160,X+20),K
310 CIRCLE(130,169),15,4
320 PAINT(130,169),4,4
330 PLAY"O1L3EDCEDCEDCC"
340 CLS0:PRINT @ 204,"YOU LOST BUDDY!";
350 PRINT @ 264,"THE WORD WAS ";W$;:GOTO
 350
360 IF U$=W$ THEN 390
370 P=P+2:PLAY"O2V30L8C"
380 RETURN
390 REM**WIN ROUTINE**
400 FOR T=1 TO 8
410 CLS T
420 SOUND T*30,1
430 NEXT T
440 CLS:PRINT @ 330,"WELL DONE!!"
450 FOR T=1 TO 500:NEXT T:GOTO400
```

```
460 REM**WORDBASE**
470 FOR C=1 TO RND(100)
480 READ W$
490 NEXT C
```



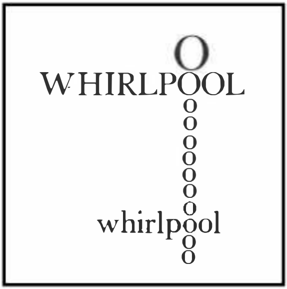
```
500 DATA "ALPHABET","FUTURE","CAPABLE","
INTELLIGENCE","MYSTIFY","KEYBOARD","DICT
IONARY","ALCOHOLIC","CASSETTE","EMBROIDE
RY"
510 DATA "XYLOPHONE","OPTICIAN","QUEEN",
"UTOPIA","SEAGULL","MOVEMENT","NOTATION"
,"RANDOM","DUMB","SAUSAGES"
520 DATA "ANAGRAM","BRUSQUE","AQUADUCT",
"JEOPARDY","GENESIS","HORTICULTURE","PRE
JUDICE","SUBTLE","CIVILISATION","ARCHITE
CTURE"
530 DATA "GENETICS","HANDICAP","INDIGO",
"JUDICIAL","PRECEDENT","KALEIDOSCOPE","B
ONDAGE","OVERDRAFT","EXODUS","BULBOUS"
540 DATA "FARCE","ZOOLOGIST","VACATION",
"LUDICROUS","JOCULAR","CHORTLE","MIRROR"
,"OSTRICH","SATIRE","DIAMONDS"
550 DATA "MEDUSA","WHISKEY","BANANA","UN
IVERSE","TOTALITARIAN","WRINKLED",
"ORTHODOX","WIZENED","TEMPESTUOUS","I
MPRESSIVE"
560 DATA "SYNCHRONISE","FRACTURE","MISSI
LE","VIDEO","YEARN","MEDIAN","CHESTNUT",
"RHODODENDRON","PEASANT","TUNIC"
570 DATA "DERISE","GYRATORY","ZIRCONIUM"
,"YELLOW","XENOPHOBIA","VENERABLE","TERR
ESTRIAL","STARLING","FREESIA","NAUTICAL"
580 DATA "INTERVENTION","AMPLIFIER","DIR
IGABLE","ENZYME","TROPHY","DRAGON","ORIG
INAL","FAULTY","MOCCASIN","LUBRICATE"
590 DATA "KANGAROO","HARPSICORD","GLADIO
LI","AUSTRALIA","GARGOYLE","FLAMINGO","D
ACHSHUND","ANTIBIOTIC","ALBATROSS","UNIV
ERSITY"
700 RETURN
```



WHIRLPOOL

The title of this program describes the effect that the pattern gives. Be patient, however, since it does take nearly two minutes to generate the complete pattern. A few 'dashes' of sound enhance the display.

```
10 REM****WHIRLPOOL****
20 REM****C.GIFFORD****
30 PMODE 4,1:PCLS:SCREEN 1,1
40 FOR J=1 TO 8 STEP.15
50 FOR A=1 TO 12 STEP .5
60 PSET(A*J*COS(J)+110,A*J*SIN(J)+80,1)
70 NEXT A
80 SOUND INT(J*30-25),1
90 NEXT J
100 FOR M=240 TO 1 STEP-3
110 SOUND M,1
120 NEXT M
130 SOUND1,20 1<10 GOTO 140
```



SONIC FORCE

Can you guess the value of the tone the computer plays? Are your ears sensitive enough to distinguish between two similar sounds? Simply enter your guess, which will be a number between 1 and 210, and the computer plays your guess alongside its own tone. You have six guesses. The winning pattern at the end of the program is by Alan Robertson.

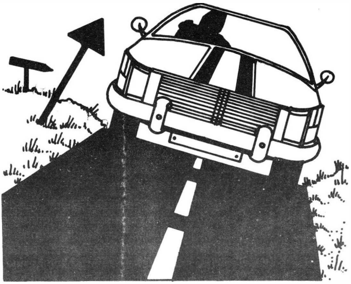
```
10 REM****SONIC FORCE****
20 REM*****C.GIFFORD*****
30 CLS
40 PRINT"WAIT FOR THE TONE......"
50 R=RND(210)
60 FOR C=1 TO 6
70 FOR T=1 TO 500:NEXT T
80 SOUND R,5
90 INPUT"WHAT IS YOUR GUESS";G
100 IF G<1 OR G>210 THEN GOTO 90
110 SOUND G,4
120 IF G=R THEN GOTO 250
130 NEXT C
140 FOR T=1 TO 1000: NEXT T
150 CLS
160 PRINT @ 227,"HARD LUCK..."
170 PLAY"O1V31DCDCL12EFGL4CP99L16CP99L2C
P4"
180 PRINT @ 264,"YOU MUST BE tone deaf"
190 PLAY"O1L100FEDCFEDCFEDC"
200 PRINT @ 328,"TONE WAS";R
210 PRINT @ 357,"PLAY IT AGAIN SAM? (Y/N
)"
220 INPUT A$
```

```
230 IF A$="Y" THEN RUN ELSE STOP
240 REM***PATTERN.A.ROBERTSON***
250 PMODE 4,1:PCLS:SCREEN 1,1
260 FOR A=196 TO 0 STEP -2
270 LINE(128,96)-(0,A), PSET
280 NEXT A
290 FOR B=0 TO 255 STEP 3
300 LINE(128,96)-(B,0),PSET
310 NEXT B
320 FOR C=0 TO 191 STEP 2
330 LINE(128,96)-(255,C),PSET
340 NEXT C
350 FOR D=255 TO 0 STEP -2
360 LINE(128,96)-(D,191),PSET
370 NEXT D
380 PLAY"L20O1CDEFGABO3CDEFGABO4CC"
390 FOR T=1 TO 1250:NEXT T
400 RUN
```

SONIC FORCE

NIGHTDRIVER

This game puts you in the driving seat of a racing car during an all-night rally. You are tired but if you come off the edge of the road you will crash. To keep your car on the road use the left and right arrow keys. The longer you manage to remain on the road the higher your score will be at the end of the game.

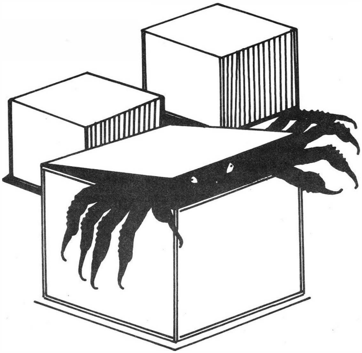


```
10 REM****NIGHTDRIVER****
20 REM*****C.GIFFORD*****
30 REM**IDEA:PETER SHAW**
40 P=125:N=0:CLS
50 PMODE 4,1:PCLS:SCREEN 1,1
60 CIRCLE(20,20),8
70 PAINT(20,20),1,1
80 FOR S=1 TO 30
90 X=RND(255):Y=RND(38)
100 PSET(X,Y,1)
110 NEXT S
120 LINE(0,40)-(255,40),PSET
130 DRAW"BM125,160;R10U3D6BL10U6BD3BR5U1
0R4U2D4BL8U4BD2R5D10L2U10"
140 IF P<100 OR P>160 THEN GOTO 230
150 LINE (P,42)-(P-40,170),PSET
160 LINE (P+5,42)-(P+45,170),PSET
170 LINE(P,42)-(P-40,170),PRESET
180 LINE(P+5,42)-(P+45,170),PRESET
190 IF PEEK(343)=223 THEN P=P+10
200 IF PEEK(344)=223 THEN P=P-10
210 P=P+RND(20)-10:N=N+RND(8)
220 GOTO 130
230 FOR C=1 TO 10
240 SCREEN 1,1
250 PLAY"T100L80BAGFEDC"
260 SCREEN 1,0
270 PLAY"T100L80O3BAGFEDC"
280 NEXT C
290 CLS 5:PRINT @ 134,"CCCRRRAAASSSHHH!!
!";
300 PRINT @ 199,"YOU SCORED ";N;
310 PRINT @ 262,"AGAIN? PRESS 1 KEY";
320 B$=INKEY$:IF B$="1" THEN RUN
330 GOTO 320
```

PANDORA'S BOX

Pit your wits and mental agility against the might of your Dragon in this updated version of Nim. There are a random number of boxes and you and the Dragon take it in turns to remove them from the line.

By skilful play you must leave your opponent with the last box which, when opened, releases the most evil creature into the world. The computer is hard to beat though it is possible.



```
10 REM*****PANDORA'S BOX*****
20 REM*******C.GIFFORD*******
30 A$=CHR$(148)+CHR$(152)
40 E=0:F=0:M=0:L=0
50 Z=RND(20)+10
60 H=RND(3)+3
70 CLS
80 PRINT"          pandora's box":PRINT
90 IF M=0 THEN PRINT"BOXES LEFT:";Z:PRIN
T"THE MOST YOU CAN TAKE IS ";H
100 PRINT
110 IF E>0 AND M=0 THEN PRINT"YOU TOOK";
E;"     I TOOK";Q
120 PRINT
130 FOR K=1 TO Z
140 PRINT A$;"   ";
150 SOUND K*8,2
160 NEXT K
170 IF Z=1 THEN PRINT @ 224,CHR$(187)+CH
R$(183);:PLAY"P8O1V31L20CDECC"
180 PRINT:PRINT
190 IF M=1 THEN PRINT" you win";:PLAY "L
2004DGAFGAFEDCC":GOTO 190
200 IF M=2 THEN GOTO 370
210 PRINT
220 INPUT"HOW MANY WILL YOU TAKE";E
230 IF E<1 OR E>H THEN GOTO 220
240 Z=Z-E
250 IF Z>0 THEN GOTO 280
260 M=2
270 GOTO 350
280 Q=Z-1-INT((Z-1)/(H+1))*(H+1)
290 IF Q=0 THEN Q=RND(H)
300 IF Q>Z THEN GOTO 280
```

GLOSSARY

A

Accumulator — the place within the computer in which arithmetic computations are performed and where the results of these computations are stored.

Algorithm — the series of steps the computer follows to solve a particular problem.

Alphanumeric — this term is usually used in relation to a keyboard, as in 'it is an alphanumeric keyboard', which means that the keyboard has letters as well as numbers. It is also used to refer to the 'character set' of the computer. The character set comprises the numbers and letters the computer can print on the screen.

ALU (Arithmetic/Logic Unit) — the part of the computer which does arithmetic (such as addition, subtraction) and where decisions are made.

AND — a Boolean logic operation that the computer uses in its decision-making process. It is based on Boolean algebra, a system developed by mathematician George Boole (1815-64). In Boolean algebra the variables of an expression represent a logical operation such as OR and NOR.

ASCII — stands for American Standard Code for Information Exchange, the most widely used encoding system for English language alphanumerics. There are 128 upper and lower case letters, digits and some special characters. ASCII converts the symbols and control instructions into seven-bit binary combinations.

Assembler — a program which converts other programs written in assembly language into machine code (which the computer can understand directly).

Assembly language is a low level programming language which uses easily memorised combinations of two or three letters to represent a particular instruction which the assembler then converts so the machine can understand it. Examples of these are ADD (add), and SUB (subtract). A computer programmed in assembly language tends to work more quickly than one programmed in a higher level language such as BASIC.

B

BASIC — an acronym for Beginners All-Purpose Symbolic Instruction Code. It is the most widely used computer language in the microcomputer field. Although it has been criticised by many people, it has the virtue of being very easy to learn. A great number of BASIC statements resemble ordinary English.

Baud — named after Baudot, a pioneer of telegraphic communications. Baud measures the rate oftransfer of information and is approximately equal to one bit per second.

BCD — an abbreviation for Binary Coded Decimal.

Benchmark — a test against which certain functions of the computer can be measured. There are a number of so-called 'standard Benchmark tests', but generally these only test speed. This is rarely the aspect of a microcomputer that is most of interest to the potential buyer.

Binary — a numbering system that uses only zeros and ones.

Bit — an abbreviation for Binary Digit. This is the smallest unit of information a computer circuit can recognise.

Boolean Algebra — the system of algebra developed by mathematician George Boole which uses algebraic notation to express logical relationships (see AND).

Bootstrap — a short program or routine which is read into the computer when it is first turned on. It orients the computer to accept the longer, following program.

Bug — an error in a computer program which stops the program from running properly. Although it is generally used to mean only a fault or an error in a program, the term bug can also be used for a fault in the computer hardware.

Bus — a number of conductors used for transmitting signals such as datainstructions, or power in and out of a computer.

Byte — a group of binary digits which make up a computer word. Eight is the most usual number of bits in a byte.

C

CAI — Computer Assisted Instruction.

CAL — Computer Assisted Learning. The term is generally used to describe programs which involve the learner with the learning process.

Chip — the general term for the entire circuit which is etched onto a small piece of silicon. The chip is, of course, at the heart of the microcomputer

Clock — the timing device within the computer that synchronises its operations.

COBOL — a high level language derived from the words Common Business Orientated Language COBOL is designed primarily for filing and record-keeping

Comparator — a device which compares two things and produces a signal related to the difference between the two

Compiler — a computer program that converts high level programming language into binary machine code so the computer can handle it.

Complement — a number which is derived fromanother according to specified rules.

Computer — a device with three main abilities or functions:
1) to accept data
2) to solve problems
3) to supply results

CPU — stands for Central Processing Unit. This is the heart of the computer's intelligence, where data is handled and instructions are carried out.

Cursor — a character which appears on the TV screen when the computer is operating. It shows where the next character will be printed. On a computer there are usually 'cursor control keys' to allow the user to move the cursor around the screen.

D

Data — information in a form which the computer can process.

Debug — the general term for going through a program and correcting any errors in it, that is, chasing down and removing bugs (see Bug).

Digital Computer —a computer which operates on information which is in a discrete form.

Disk/Disc — this is a magnetically sensitised plastic disk, a little smaller than a single play record. This is used for storing programs and for obtaining data. Disks are considerably faster to load than a cassette of the same length program. The disk can be searched very quickly while a program is running for additional data.

Display — the visual output of the computer, generally on a TV or monitor screen.

Dot Matrix Printer — a printer which prints either the listing of a program or that which is displayed on the TV screen. Each letter and character is made up of a number of dots. The higher the number of dots per character the finer the resolution of the printer.

Dynamic Memory — a memory unit within the computer which 'forgets' its contents when the power is turned off.

E

Editor— this term is generally used for the routine within the computer which allows you to change lines of a program while you are writing it.

EPROM — stands for Erasable Programmable Read-Only Memory. This is like the ROM in the computer, except that it is fairly easy to load material into an EPROM and it doesn't disappear when you turn the power off. EPROMs must be placed in a strong ultra violet light to erase them.

Error Messages — the information given by a computer where there is a fault in the coding during a part of a program, usually shown by the computer stopping, and printing a word, or a word and numbers, or a combination of numbers only, at the bottom of the screen. This tells you what mistake has been made. Common mistakes include using the letter O instead of zero in a line, or leaving out a pair of brackets, or one of the brackets, in an expression, or failing to define a variable.

F

File — a collection of related items of information organised in a systematic way.

Floppy Disk — a relatively cheap form of magnetic disk used for storing computer information, and so named because it is quite flexible (see Disk/Disc).

Flow Chart — a diagram drawn up before writing a program, in which the main operations are enclosed within rectangles or other shapes and connected by lines, with arrows to represent loops, and decisions written at the branches. It makes writing a program much easier because traps such as infinite loops, or non-defined variables can be caught at an early stage It may not be worth writing a flow chart for very short programs, but generally a flow chart aids in creating programs.

Firmware — there are three kinds of 'ware' in computers: software 'temporary' programs; hardware like the ROM which contains permanent information; and firmware in which the information is relatively permanent, as in an EPROM (see EPROM).

Flip-Flop — a circuit which maintains one electrical condition until changed to the opposite condition by an input signal.

FORTRAN — an acronym for FORmula TRANslation, this is a high level, problem orientated computer language for scientific and mathematical use.

G

Gate — an electrical circuit which, although it may accept one or more incoming signals, only sends out a single signal.

Graphics — pictorial information as opposed to letters and numbers.

H

Hard Copy — computer output which is in permanent form.

Hardware — the physical parts of the computer (also see software and firmware).

Hexadecimal (Hex) — a numbering system to the base sixteen. The digits zero to nine are used, as well as the letters A, B, C, D, E and F to represent numbers. A equals 10, B equals 11, C equals 12, and so on. Hex is often used by microprocessor users.

Hex Pad — a keyboard designed specifically for entering hexadecimal notation.

High Level Language — a programming language which allows the user to talk to the computer more or less in English In general, the higher the level of the language (that is, the closer it is to English), the longer it takes for the computer to translate it into a language it can use. Lower level languages are far more difficult for human operators but are generally executed far more quickly.

I

Input — the information fed into the computer via a keyboard, a microphone, a cassette or a disk.

Input/Output (I/O Device) — a device which accepts information or instructions from the outside world, relays it to the computer, and then, after processing, sends the information out in a form suitable for storing, or in a form which could be understood by a human being

Instruction — data which directs a single step in the processing of information by the computer (also known as a command)

Integrated Circuit — a complete electronic circuit imprinted on a semiconductor surface.

Interface — the boundary between the computer and a peripheral such as a printer.

Interpreter — a program which translates the high level language fed in by the human operator, into a language which the machine can understand.

Inverter — a logic gate that changes the signal being fed in, to the opposite one.

Interactive Routine — part of a program which is repeated over and over again until a specified condition is reached.

J

Jump Instruction — an instruction which tells the computer to go to another part of the program, when the destination of this move depends on the result of a calculation just performed.

K

K — this relates to the size of the memory. Memory is usually measured in 4K blocks. 1K contains 1,024 bytes.

Keyword — the trigger word in a line of programming, usually the first word after the line number. Keywords include STOP, PRINT and GOTO.

L

Language — computer languages are divided into three sections: high level languages, such as BASIC, which are reasonably close to English and fairly easy for humans to use; low level languages, such as Assembler, that use short phrases which have some connection with English (ADD for add and RET for return, for instance); and machine code which communicates more or less directly with the machine.

LCD — this stands for Liquid Crystal Diode. Some computers such as the TRS-80 Pocket Computer use an LCD display.

LED — this stands for Light Emitting Diode. The bright red numbers which are often used on watch or clock displays are made up of LEDs.

Logic — the mathematical form of a study of relationships between events.

Loop — a sequence of instructions within a program which is performed over and over again until a particular condition is satisfied.

M

Machine Language or Machine Code — an operation code which can be understood and acted upon directly by the computer.

Magnetic Disk — see Disk and Floppy Disk.

Mainframe — computers are generally divided into three groups, and the group a computer falls into depends more or less on its size. The computer you are thinking of buying is a microcomputer; medium sized computers are known as minicomputers; and the giant computers that you sometimes see in science fiction movies are mainframe computers. Until 15 years ago mainframe computers were, in practical terms, the only ones available.

Memory — there are two types of memory within a computer. The first is called ROM (read-only memory); this is the memory that comes already programmed on the computer, which tells the computer how to make decisions and how to carry out arithmetic operations. This memory is unaffected when you turn the computer off. The second type is RAM (random access memory) This memory holds the program you type in at the keyboard or send in via a cassette or disk. In most computers the computer 'forgets' what is in RAM when you turn the power off.

Microprocessor — the heart of any computer. It requires peripheral unit interfaces, such as a power supply and input and output devices, to act as a microcomputer.

MODEM — stands for Modulator Demodulator. This is a device which allows two computers to talk to each other over the telephone. The computers usually use a cradle in which a telephone receiver is placed.

Monitor — this has two meanings in computer terms. One meaning is a television-like display. A monitor has no facility for tuning television programs, and usually the picture produced on a monitor is superior to that produced by an ordinary television. The second meaning of a monitor relates to ROM. The monitor of a computer is described as the information it has built in when you buy it. This information allows it to make decisions and carry out arithmetic computations.

Motherboard — a framework to which extra circuits can be added. These extra circuits often give the computer facilities which are not built-in, such as that of producing sound or of controlling a light pen.

MPU — an abbreviation for Microprocessor Unit.

N

Nano-second — a nano-second is one thousand billionth of a second, the unit of speed in which a computer or a memory chip is often rated.

Non-Volatile Memory — memory which is not lost when the computer is turned off. Some of the smaller computers such as the TRS-80 Pocket Computer have non-volatile memory. The batteries hold the program you enter for several hundred hours.

Not — a Boolean logic operation that changes a binary digit into its opposite.

Null String — a string which contains no characters. It is shown in the program as two double quote marks, without anything between them.

Numeric — pertaining to numbers as opposed to letters (that is, alphabetic). Many keyboards are described as being alphanumeric which means both numbers and letters are provided.

O

Octal — a numbering system which uses eight as the base, and the digits 0, 1, 2, 3, 4, 5, 6 and 7. The Octal system is not used very much nowadays in microcomputer fields. The Hexadecimal system is more common (see Hexadecimal).

Operating System — the software or firmware generally provided with the machine that allows you to run other programs.

OR — an arithmetic operation that returns a 1, if one or more inputs are 1.

Oracle — a method o f sending text messages with a broadcast television signal. A teletext set is required to decode the messages. Oracle is run by Independent Television Service in the UK, and a similar service — Ceefax — is provided by the BBC.

Output — information or data fed out by the computer to such devices as a TV-like screen, a printer or a cassette tape. The output usually consists of the information which the computer has produced as a result of running a program.

Overflow — a number too large or too small for the computer to handle.

P

Pad — see Keypad.

Page — often used to refer to the amount of information needed to fill one TV screen, so you can talk about seeing a page of a program, the amount of the listing that will appear on the screen at one time.

PASCAL — a high level language.

Peripheral — anything which is hooked onto a computer, for control by the computer, such as a disk unit, a printer or a voice synthesiser.

Port — a socket through which information can be fed out of or in to a computer.

Prestel — the British telecom name for a system of calling up pages of information from a central computer via the telephone and displaying them on a television screen. A similar commercial version in the United States is known as The Source.

Program — in computer terms program has two meanings One is the list of instructions that you feed into a computer, and the second is used as a verb, as in 'to program a computer'

PROM — stands for Programmable Read Only Memory. This is a device which can be programmed, and once it is then the program is permanent (also see EPROM and ROM).

R

Random Access Memory(RAM) — the memory within a computer which can be changed at will by the person using the computer. The contents of RAM are usually lost when a computer is turned off. RAM is the memory device that stores the program that you type in and also stores the results of calculations in progress.

Read-Only Memory(ROM) — in contrast to RAM, information in ROM cannot be changed by the user of the computer, and the information is not lost when the computer is turned off. The data in ROM is put there by the manufacturers and tells the computer how to make decisions and how to carry out arithmetic computations. The size of ROM and RAM is given in the unit K (see K)

Recursion — the continuous repetition of a part of the program.

Register — a specific place in the memory where one or more computer words are stored during operations

Reserved Word — a word that you cannot use for a variable in a program because the computer will read it as something else. An example is the word TO. Because TO has a specific computer meaning, most computers will reject it as a name for a variable. The same goes for words like FOR, GOTO and STOP.

Routine — this word can be used as a synonym for program, or can refer to a specific section within a program (also see Subroutine).

S

Second Generation — this has two meanings. The first applies to computers using transistors, as opposed to first generation computers which used valves. Second generation can also mean the second copy of a particular program; subsequent generations are degraded by more and more noise.

Semiconductor — a material that is usually an electrical insulator but under specific conditions can become a conductor.

Serial — information which is stored or sent in a sequence, one bit at a time.

Signal — an electrical pulse which is a conveyor of data.

Silicon Valley — the popular name given to an area in California where many semiconductor manufacturers are located.

SNOBOL — a high level language.

Software — the program which is entered into the computer by a user which tells the computer what to do.

Software Compatible — this refers to two different computers which can accept programs written for the other.

Static Memory — a non-volatile memory device which retains information so long as the power is turned on, but does not require additional boosts of power to keep the memory in place.

Subroutine — part of a program which is often accessed many times duringthe execution of the main program. A subroutine ends with an instruction to go back to the line after the one which sent it to the subroutine.

T

Teletext — information transmitted in the top section of a broadcast television picture. It requires a special set to decode it to fill the screen with text information. The BBC service is known as Ceefax, the ITV service as Oracle. Teletext messages can also be transmitted by cable, for example the Prestel service in Britain or The Source in the United States.

Teletype — a device like a typewriter which can send information and also receive and print it.

Terminal — a unit independent of the central processing unit. It generally consists of a keyboard and a cathode ray display.

Time Sharing — a process by which a number of users may have access to a large computer which switches rapidly from one user to another in sequence, so each user is under the impression that he or she is the sole user of the computer at that time.

Truth Table — a mathematical table which lists all the possible results of a Boolean logic operation, showing the results you get from various combinations of inputs.

U

UHF — Ultra High Frequency (300-3000 megaHertz)

Ultra Violet Erasing — Ultra violet light must be used to erase EPROMs (see EPROM).

V

Variable — a letter or combination of letters and symbols which the computer can assign to a value or a word during the run of a program.

VDU — an abbreviation for Visual Display Unit.

Volatile — refers to memory which 'forgets' its contents when the power is turned off.

W

Word — a group of characters, or a series of binary digits, which represent a unit of information and occupy a single storage location. The computer processes a word as a single instruction.

Word-Processor — a highly intelligent typewriter which allows the typist to manipulate text, to move it around, to justify margins and to shift whole paragraphs if necessary on a screen before outputting the information onto a printer. Word-processors usually have memories, so that standard letters and the text of letters, written earlier, can be stored

BIBLIOGRAPHY

Compiled by Tim Hartnell

The A to Z Book of Computer Games (McIntire, Thomas C, Tab Books, Blue Ridge Summit, Pa.).

This is a fine Tab book to give you program ideas and ready-to-run programs, although some of the games are a disappointment, such as the overly long Othello program which does not even play, but simply records the moves made by two human players. Others, however, such as Fivecard and Hotshot, are well written, and well worth entering into your microcomputer.

BASIC Computer Games (ed. Ahl, David, Creative Computing Press, Morristown, New Jersey).

This is a classic work, the source of more programming ideas than any other computer games book ever published. I had a meal with David Ahl one night in London after a PCW show and discussed the book. He said that he'd been in the personal computer field almost before there were personal computers, and while many of the games in this book do not seem startling now, the fact that people could write and play games for computer interaction at all seemed quite incredible in the late seventies. The Checkers program, and Life for Two are just a couple of the treasures you will find in this splendid program and idea source book.

BASIC Computer Programs for the Home (Sternberg, Charles D, Hayden Book Company, Inc., Rochelle Park, New Jersey).

Traditionally, home computers (when first purchased) have been used for playing games. One reason why

they have not been used for more serious applications stems from the lack of a readily available, comprehensive set of home applications programs that were easy to use and understand and that satisfied the practical requirements of the home. This book provides a set of programs to make your computer start earning its keep. The programs provide a good cross-section of practical applications; these have been designed so as not to rely upon the availability of tape or disk-storage devices. The programs cover a wide field, and are divided into a number of sections: home financial programs (including household expenses and income tax recording); car related programs (including fuel use and trip planning); 'Kitchen Helpmates' (including diet and meal planning programs); scheduling programs for home use (including a reminder calendar and a couple of programs which I imagine are designed to short circuit arguments about which television programs will be watched); and 'List programs for every purpose' (including Christmas cards, music collections and three versions of an addresses program).

The BASIC Handbook (Lien, David A, Compusoft Publishing, San Diego, California).

This is an encyclopedia of the BASIC language. Now that BASIC is so firmly established throughout the microcomputer world, it is necessary to make its many dialects understandable so that programs can be transported between different computers. When you have found exactly the program you've been looking for, it is very frustrating to be unable to run it on your computer. This book addresses that problem by discussing in detail just about every commonly used BASIC statement, function, operator and command. For the most part, BASIC words mean the same thing to every computer which recognises them. If a computer does not possess the capabilities of a needed or specified word, there are often ways to accomplish the same function by using another word, or combination of words. Although the handbook requires some application to transform the information into usable form, it is a very valuable reference work indeed. Every BASIC word you have ever heard of (and many you may not have heard of, such as LE, NE, GOTO-OF, RES and TIME) is probably in the book. It may be of limited use to you in your early days of computing, but it should become an indispensable handbook once you get more involved in the subject.

Beat the Odds, Microcomputer Simulations of Casino Games (Sagan, Hans, Hayden Book Company, Inc., Rochelle Park, New Jersey).

The book explains how to play certain casino games (trente-et-quarante, roulette, chemin-de-fer, craps and blackjack) and gives complete program listings in BASIC with commentaries on systems and optimal strategies. Professor Sagan (Professor of Mathematics at North Carolina State University) says he wrote the book in an attempt to convince people that, in the long run, they could not win — except possibly at blackjack — and to explain some popular systems and their pitfalls, and above all to provide very realistic computer simulations of the games themselves. He has succeeded in his attempt. The listings are possibly longer than other computer versions of the same games, but this is because the Sagan versions strictly duplicate the odds involved in playing the game 'in real life', and cover all the eventualities that a real game can produce. The programs are well-structured, and an examination of the listings should give you ideas for improving your own programming.

Beginner's Guide to Chess (Keene, Raymond, Pelham Books Ltd, London).

An ideal guide to simple chess-playing techniques which you can turn into algorithms if you intend to write a chess program of your own.

The Calculator Game Book for Kids of All Ages (Hartman, Arlene, Signet Books, New York).

The book's title says it all, and the names of the games (which include Fibonacci Follies, Stretch to Sixty and Casting Out 9s) suggest the book's contents. There are some worthwhile brain-stretching puzzles, and 15 or so ideas definitely calling for conversion to computer games.

33 Challenging Computer Games for TRS-80/Apple/PET (Chance, David, Tab Books, Blue Ridge Summit, Pa.).

Even if you don't have any of the three computers named in the title, you will still find the book a goldmine of ideas for your own development, and many of them will run, with minimal alteration, on any BASIC-using computer. Particularly commendable programs are Life Support, Scrambled Eggs and Tank Assault.

Communicating with Microcomputers (Witten, Ian H., Academic Press, London).

This is an introduction to the technology of man/computer communications for the non-specialist. By placing particular emphasis on low-cost techniques associated with small systems and personal computers, the reader's attention is focused on the positive nature of the 'microprocessor revolution' — how machines can help people — rather than the negative aspects which are often highlighted in the non-technical press. The level of the book is suitable for the layman with some acquaintance with electronics. The final section, on speech communication, provides the most fascinating reading.

Computer Appreciation (Fry, T.F., Newnes, Butterworths, 1975).

A fairly 'straight' but useful overview of computer operation, and business applications. Designed to be used as a text for a course of business studies, the book covers a wide range of topics from a short account of the historical development of calculating devices, through computer hardware and programming, to the organisation of a modern data-processing department. It concludes with a brief consideration of the applications of computers and a discussion on the effects of computers upon management matters. It is surprisingly undated, despite the extraordinary increase in hardware availability and capability since the book was written.

The Computer Book: An Introduction to Computers and Computing (Bradbeer, Robin; De Bono, Peter; Laurie, Peter; BBC Publications).

This book was published in conjunction with the BBC television series 'The Computer Programme', first transmitted on BBC2 from January 1982, and produced by Paul Kriwaczek. I discussed this book with Robin Bradbeer while it was being written, and he told me that the BBC editors were ruthless in pointing out any use of jargon They insisted, said Robin, that nothing could be taken for granted. This insistence has resulted in a book which anyone can understand. It assumes nothing, not even the knowledge of how to use a shift key — or the effect of using it — on a typewriter. The many illustrations and photographs break up the text, which gives a detailed introduction to computers, especially micros, and their possible applications.

Computer Games for Businesses, Schools and Homes (Nahigian, J Victor and Hodges, William S. Winthrop Publishers Inc., Cambridge, Mass.).

Some of the programs are a little thin for the size and price of the book, but the best ones are well worth adapting to run on your computer. The inclusion of long, clear sample run printouts ensures that you know exactly what the programs will do before you run them. The Tennis and Star Trek programs are especially good.

Dice Games Old and New (Tredd, William E., Oleander Press, Cambridge).

This will give you enough clearly written games explanations to keep you creating games programs on your microcomputer for a long time to come.

The Electronic Calculator in Business, Home and School (Birtwistle, Claude, Elliot Right Way Books, Kingswood, Surrey).

To get the best out of a calculator, you need to understand the mathematics which lies behind the operations. That is the purpose of the book, and in general it succeeds in this aim. The maths involved is, however, fairly simple and basic, since the book was written with a wide range of people in mind — the pupil at school, the student at college, the business person and the householder. It is a practical book which should be read and worked through with a calculator to hand.

Everyman's Indoor Games (Brandreth, Gyles, J M Dent and Sons Ltd, London)

If you're looking for games to convert into computer programs, ignore the chapters entitled Parlour Games and Children's Party Games and stick to the rest of the book, a treasure trove of games concepts which are certainly worth using as a starting point. Fox and Geese, Poker Dice and Billiards, as described in the book, are only a few of the programs you might write after reading it.

Games and Puzzles for Addicts (Millington, Roger, M and J Hobbs, Walton-on-Thames).

These games and puzzles first appeared in the weekly computer news-magazine 'Datalink', so they are especially likely to appeal to computer buffs. There are many ideas here that can be converted into games to be played with the computer.

Games for Home, Travel and Parties (Jensen, Helen, Western Publishing Company Inc., Racine, Wisconsin).

Aimed squarely at children, this book gives some games which are simple to program (these include Snakes, Lift-Off and Fish), and contains a complete chapter on how to play chess.

Home Computers, Questions and Answers, Hardware (Didday, Rich, dilithium Press).

The book has two main purposes. Firstly, it is intended to give readers a real feeling for what is involved in home computing, so that they can make rational decisions before buying equipment. Secondly, it is intended to give people who have no specialised knowledge of computing a general background to the subject, and specifically to microcomputers. The book succeeds in imparting enough information to ensure you will have little trouble understanding articles about advanced projects in computer hobbyist magazines, advertisements for home computing equipment, or other people who do have advanced computer knowledge

Inside BASIC Games (Mateosian, Richard, Sybex).
This book is a guide, albeit a slightly overwritten one, for anyone who wishes to understand computer games. You will learn how to write interactive programs in BASIC and how the principles of systems development are applied to small computers. The book also looks at how the features of specific small computer systems have been supported in BASIC. The sections of the book include: Arithmetic Games, Guessing Games, Time Games, Date Games, Taxman, and programming in 'Free BASIC', a structured BASIC that is translated manually into the actual BASIC instructions to be entered into the computer. Free BASIC is not a language; it is a program description medium (like flowcharts) that has no line numbers, and uses symbolic names for subroutines. Additional chapters look at The Match-Up Game, Craps and Alien Life. If you can contend with the verbiage, you will find this book well worthwhile.

An Introduction to Personal and Business Computing (Zaks, Rodnay, Sybex).
I had lunch with Rodnay in London during a PCW show and he told me that he thought current American predictions on the growth of the personal computer field were grossly pessimistic. He pointed out that the predictions current in 1978, when he wrote this book,

have been proved so inaccurate that would-be prophets should take warning and assume that whatever they say will be wrong by a factor of 10 or 100. Despite its age — and computer books do age uncommonly quickly — this book is a good introduction to the field, explaining in clear, snappy English the fundamentals of computer operation. Dr Zaks also gives suggestions on what to look for when buying a computer.

Microsoft BASIC (Knecht, Ken, dilithium Press, Forest Grove, Oregon).

This book presents a complete introduction to programming in Microsoft BASIC. The concepts presented are illustrated with short, working programs. By starting with the simplest and most commonly used commands, and then progressing on to the more complex BASIC commands, Mr Knecht shows how the more powerful versions of the language can save valuable programming time and effort.

The Personal Computer Book (Bradbeer, Robin, Input Two-Nine).

The title says it all. Robin is deeply involved in the microfield in Great Britain. He started the North London Polytechnic Computer Fairs, assisted the BBC with their microcomputer television show (and co-authored *The Computer Book*, published by the BBC), and edited the monthly publication *Educational Computing*. This gave him a strong background from which to write the book. It explains what a computer is and how it works; it elucidates the mysteries inside the 'black boxes' which make up a computer; and it gives a number of very useful appendices, including bus standards, manufacturers and distributors, magazines, a selected bibliography (compiled by Richard Ross-Langley, of *Mine of Information*) and a glossary. But perhaps the most interesting and useful section of the book is the part which describes, in some detail, the majority of computer systems available on

the British market, their price and their capabilities. Overall, this is a very impressive source book.

Personal Computers: What They Are and How to Use Them (Wels, Byron G., Trafalgar House Publishing).

A great deal has happened in the computer world since this book was written in 1978, but there is still a great deal of value and interest in it. The book details some of the personal computers available, and the improvements that are likely to be made in the future. It explains, in layman terms, how a computer works, and how to make it work for you. There is also material on the construction and maintenance of small computer systems.

Play the Game (Love, Brian, Michael Joseph and Ebury Press, London).

This is a splendid book, containing 40 or so full-size reproductions of Victorian (and pre-Victorian) board games, many suitable for playing against a computer. It is even possible to use the boards in the book (the computer then tells you where it is moving on this external board) rather than write a routine within the program to display a board.

The Pocket Calculator Games Book (Schlossberg, Edwin and Brockman, John, Wilton House Publications Ltd., London).

There are many ideas here suitable for conversion into computer games.

57 Practical Programs and Games in BASIC (Tracton, Ken, Tab Books, Blue Ridge Summit, Pa.).

There are more serious programs than games(of the Chi-Square Evaluation and Fibonacci Numbers variety) in this book. They are well-programmed, and supported by adequate (if brief) documentation, and by flow-charts. The Space Wars programs (versions one and two) at the end of the book are particularly good.

Problems for Computer Solution (Rogowski, Stephen J, Creative Computing Press, Morristown, New Jersey).

This outlines over 50 simple (and a few not-so-simple) problems which can be solved by writing a program. There are both teacher and student editions of this book; the teacher edition has a suggested program and sample run printout to solve the difficulty. It is an excellent source for educational ideas.

Stimulating Simulations (Engel, C.W , Hayden Book Company, Inc , New Jersey).

Here, according to the cover, are '12 unique programs in BASIC for the computer hobbyist' Inside you will find some fascinating programs: Forest Fire, Rare Birds and The Devil's Dungeon are three you are sure to enjoy playing, while Diamond Thief (the computer decides who has committed the crime, then challenges you to discover which of the suspects is guilty) is both well written and tightly programmed.

TAKE TWO! 32 Board Games for 2 Players (Tapson, Frank, A & C Black, London)

This book is aimed at children, but it does give many fascinating ideas that could be transformed into computer games (even if some of them are duplicated elsewhere in the book)

24 Tested, Ready-to-Run Game Programs in BASIC (Tracton, Ken, Tab Books, Blue Ridge Summit, Pa.).

Tab Books are prolific publishers in the microcomputer program field, and their books are deservedly successful. If nothing else, reading a book such as this one will give you ideas for structuring programs neatly, and for writing them to ensure the maximum compatibility between different versions of BASIC. Many of the games, such as Auto Rally and Capture the Alien, are (despite their weak titles) well thought out, carefully constructed programs.

1001 Things to Do with Your Personal Computer (Sawush, Mark, Tab Books, Blue Ridge Summit, Pa.).

I bought this book at a computer fair in Atlanta, and read it (making notes, and turning down page corners) on the flight to London. And I still hadn't finished it on arrival. If you feel you have come to the end of possible applications for your computer, buy this book and

discover that you have barely scratched the surface. It tells you about writing music and stories, aiding a mechanic or a carpenter, solving simultaneous equations, astrology, and much, much more.

The World Computer Chess Championship (Hayes, Jean E., and Levy, David N.L., Edinburgh University Press, Edinburgh).

This is a fascinating account of the world 's first machine versus machine chess championship, held in 1974 , when the dozen or so computer programs taking part were the only chess programs in existence. The games are analyzed in detail, and the final section of the book outlines a board-numbering system which you could use if you're considering writing your own chess program. The book makes you realize how far the computer world has come in only a few years.

The Virgin Computer Games Series

GAMES FOR YOUR DRAGON

More than 20 challenging programs, each one especially written for the series and guaranteed to provide hours of entertainment.

The games include MINE ESCAPE (trapped in a mine with your air running out – dynamite yourself out in time); CHECKERS (outmanoeuvre the computer in this classic board game); MISSION APOLLO (you, your delicate spacecraft and the treacherous lunar landscape); BEAT THE BLUES (cure depression with laughing gas!); and LUMBERJACK (chop down enough Canadian trees and you get a crack at the Giant Redwood Pine).

GAMES FOR YOUR DRAGON will improve your programming skills as you follow the instructions to put each of the programs into your machine, and comes complete with a brief dictionary of computer terms, a selective bibliography and some hints on how to improve and extend the programs in the book.

Programs of originality and quality for all the family.

0 907080 89 8
£2.95