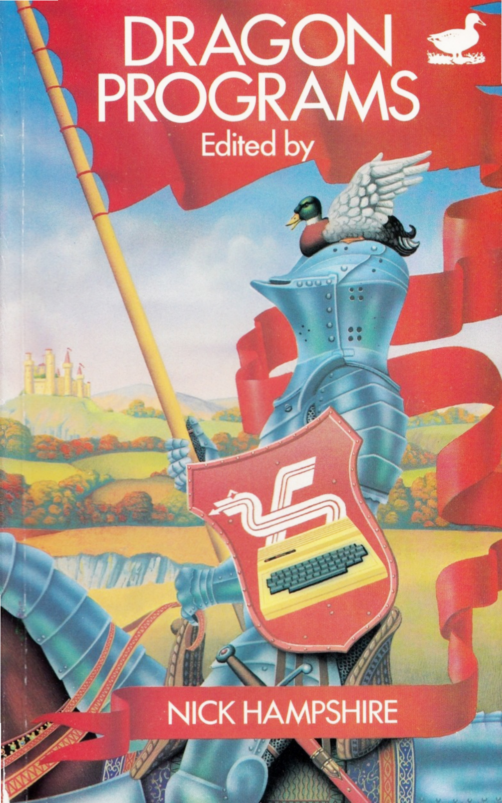
DRAGON PROGRAMS

Edited by



NICK HAMPSHIRE

DRAGON PROGRAMS

Edited By

NICK HAMPSHIRE

Duckworth

First published in 1984 by
Gerald Duckworth & Co. Ltd.
The Old Piano Factory
43 Gloucester Crescent, London NW1

ISBN 0 7156 1801 6

British Library Cataloguing in Publication Data
Hampshire, Nick
Dragon programs.—(Duckworth home computing)
1
1. Dragon 32 (Computer)—Programming
I. Title
001.64'25 QA76.8.D7
ISBN 0-7156-1801-6

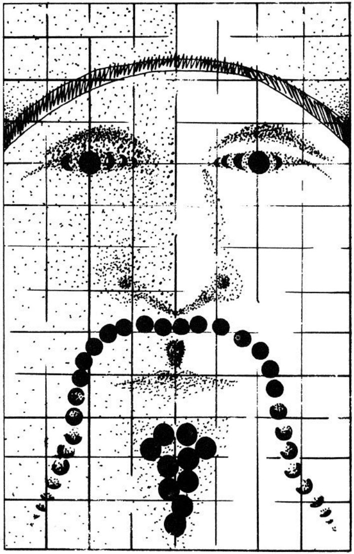
Printed and bound in Great Britain by
Redwood Burn Limited, Trowbridge, Wiltshire

CONTENTS

INTRODUCTION

This book is the first containing programs written principally by Mr John Alishaw, a dedicated Dragon User. All the programs makeexcellent use of the machine's capabilities. I have enjoyed editingthis book and am proud to bethefirsttointroducethis new programmer to the industry.

NICK HAMPSHIRE



GOMOKO

DESCRIPTION

In this game the computer takes on the guise of a wise oriental and offers us the ancient game of GO or Gomoko. The game involves two opponents attempting to create vertical, horizontal or diagonal patterns of five counters in a row. The board has 10 vertical and 10 horizontal intersecting lines to form a grid.

Each player takes it in turn to position a counter over an intersection on the grid, battling to be the first to create a line of five consecutive counters.

The player will find that the wise oriental (the computer) will take a long time – 40 seconds – to complete his move after calculating all the possible moves. It is possible to win, the player will find it very difficult to do so but will gain a tremendous amount of enjoyment when he does win.

RUNNING THE PROGRAM

After typing RUN the computer will display the playing board and will make its first move. It will then ask for the player's. The column number must first be entered and then the row letter required. The computer's counter is a '#' character and the player's is a '@'.

PROGRAM STRUCTURE

The lines of interest in this program are as follows:

70-190 Main program loop.
240-280 Produce computer's first move.
320-480 Input player's move and check for legality.
490-830 Main routine to determine and display computer's move which takes into account the calculations worked out in the subroutine 840 below.
840-1470 Stores player's move with respect to all the surrounding counters and spaces. Checks to see whether the player has won.
1850-1950 Display the board of play.
2010-2080 Input for another go.

```
10 REM GOMOKO
20 REM ************************
30 REM
40 DIM VMAX(2),NF(2),WD(2,20,4),BD(10,10),DB(20)
50 GOSUB1850
60 GOSUB240
70 GOSUB320
80 GOSUB1480
90 PRINT@448,"  PLEASE LET ME THINK A WHILE"
100 GOSUB840
110 PRINT@448
120 IF VMAX(2)=19 THEN WIN=2:GOTO160
130 GOSUB490
140 IF VMAX(1))>=14 THEN WIN=1:GOTO160
150 GOTO70
160 PLAY"O4V31L20F"
170 IF WIN=1 THENPRINT@493,"I WIN";:GOTO2010
180 PRINT@492,"YOU WIN";
190 GOTO2010
200 REM
210 REM PRODUCE COMPUTER'S
220 REM FIRST MOVE
230 REM
240 R1=RND(6)+2
250 C1=RND(6)+2
260 PF=1098+32*R1+C1
270 POKEPF,99
280 RETURN
290 REM
300 REM INPUT YOUR MOVE
310 REM
320 PRINT@448:PRINT@452,"YOUR MOVE ";
```

```
330 INPUTR$
340 PRINT@468,"AND ";
350 INPUTC$
360 IF C$="" OR R$="" THEN390
370 R=VAL(R$):C=ASC(C$)-64
380 IFABS(R-5.5)<=4.5 AND ABS(C-5.5)<=4.5 THEN400
390 PRINT@11,"TRY AGAIN":GOTO440
400 P=1098+32*R+C
410 D=PEEK(P)
420 IFD=110 THEN470
430 PRINT@9,"SPACE OCCUPIED"
440 GOSUB1990
450 PRINT@9
460 GOTO320
470 POKEP,64
480 RETURN
490 FORA=1TO2
500 IF VMAX(A)<14 THEN530
510 GOSUB1760
520 RETURN
530 NEXTA
540 FORA=1TO2
550 IF VMAX(A)<10 THEN590
560 IF A=2 AND VMAX(2)<12 AND VMAX(1)=9 THEN590
570 GOSUB1550
580 RETURN
590 NEXTA
600 FOR A=1TO2
610 IF VMAX(A)<9 THEN640
620 GOSUB1550
630 RETURN
640 NEXTA
650 PMAX=0
```

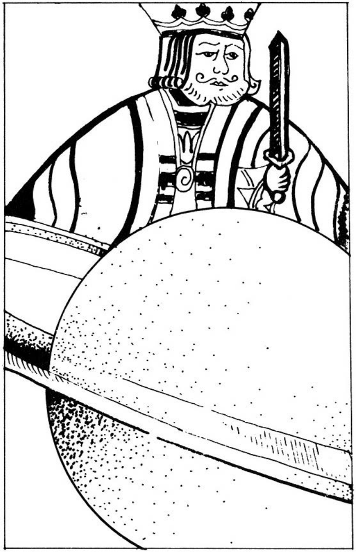
```
660 NT=0
670 FOR R1=1TO10
680 FOR C1=1TO10
690 IF BD(R1,C1)=0 THEN780
700 P=BD(R1,C1)
710 IF P<PMAX THEN780
720 IF P=PMAX THEN750
730 NT=0
740 PMAX=P
750 NT=NT+1
760 RM(NT)=R1
770 CM(NT)=C1
780 NEXTC1
790 NEXTR1
800 PN=RND(NT)
810 PF=1098+32*RM(PN)+CM(PN)
820 POKEPF,99
830 RETURN
840 V=0
850 FOR IL=1TO2
860 VMAX(IL)=0
870 NF(IL)=0
880 NEXT IL
890 RL=1
900 RU=6
910 CL=1
920 CU=10
930 F=32
940 GOSUB1140
950 RL=1
960 RU=10
970 CL=1
```

```
900 CU=6
990 F=1
1000 GOSUB1140
1010 RL=1
1020 RU=6
1030 CL=1
1040 CU=6
1050 F=33
1060 GOSUB1140
1070 RL=1
1080 RU=6
1090 CL=5
1100 CU=10
1110 F=31
1120 GOSUB1140
1130 RETURN
1140 FOR R=RL TO RU
1150 S=1098+32*R
1160 FOR C=CL TO CU
1170 SS=S+C
1180 FOR I=0 TO 4
1190 CD=SS+F*I
1200 D=PEEK(CD)
1210 IF D=110 THEN DW(I)=CD:GOTO1260
1220 IF TP=0 THEN TP=D:II=((TP-99)/(-35)+1)
1230 IF TP<>D THEN V=0:TP=0:GOTO1450
1240 V=V+5-ABS(I-2)
1250 DW(I)=0
1260 NEXT I
1270 IF V=0 THEN1450
1280 IF V<VMAX(II) THENV=0:GOTO1360
1290 IF V=VMAX(II) THEN1320
```

```
1300 NF(II)=0
1310 VMAX(II)=V
1320 NF(II)=NF(II)+1
1330 FOR J=0 TO 4
1340 WD(II,NF(II),J)=DW(J)
1350 NEXT J
1360 FOR J=0 TO 4
1370 IF DW(J)=0 THEN1420
1380 DD=DW(J)-1098
1390 R1=INT(DD/32)
1400 C1=DD-32*R1
1410 BD(R1,C1)=BD(R1,C1)+1
1420 NEXT J
1430 V=0
1440 TP=0
1450 NEXT C
1460 NEXT R
1470 RETURN
1480 FOR R=1 TO 10
1490 FOR C=1 TO 10
1500 BD(R,C)=0
1510 NF(II)=NF(II)+1
1520 NEXT C
1530 NEXT R
1540 RETURN
1550 PMAX=0
1560 NT=0
1570 FOR I=1 TO NF(A)
1580 FOR J=0 TO 4
1590 P=WD(A,I,J)
1600 IF P=0 THEN1690
1610 R1=INT((P-1098)/32)
```

```
1620 C1=(P-1093)-32*R1
1630 IF BD(R1,C1)<PMAX THEN1690
1640 IF BD(R1,C1)>PMAX THEN1670
1650 NT=0
1660 PMAX=BD(R1,C1)
1670 NT=NT+1
1680 DB(NT)=P
1690 NEXT J
1700 IF A=2 AND PMAX=1 THEN1740
1710 PN=RND(NT)
1720 POKEDB(PN),99
1730 RETURN
1740 GOSUB590
1750 RETURN
1760 FOR J=0 TO 4
1770 IF WD(A,1,J)=0 THEN1790
1780 P=WD(A,1,J)
1790 NEXT J
1800 POKEP,99
1810 RETURN
1820 REM
1830 REM DISPLAY BOARD
1840 REM
1850 CLS
1860 PRINT@43,"ABCDEFGHIJ"
1870 PRINT@74,STRING$(12,191)
1880 NN=8
1890 FORI=1TO10
1900 IF I>9 THENNN=7
1910 PRINTTAB(NN)STR$(I),CHR$(191);"..........";CHR$(191)
1920 NEXT
1930 WIN=0
```

```
1940 PRINTTAB(10)STRING$(12,191)
1950 RETURN
1960 REM
1970 REM TIME DELAY
1980 REM
1990 FOR TD=1 TO 1000:NEXTTD
2000 RETURN
2010 FOR TD=1 TO 5000:NEXT
2020 PRINT@487,"ANOTHER GO (Y/N) ?";
2030 A$=INKEY$
2040 IF A$="" THEN2030
2050 IF A$="Y" THEN RUN
2060 IF A$="N" THEN2080
2070 GOTO2030
2080 CLS
```



PLANET RULER

DESCRIPTION

The player controls the operation of a mining planet. Initially there are 100 resident workers, each man capable of controlling 10 mines simultaneously. Executive decisions concerning production and workforce welfare are made by the player.. Interplanetary sorties by Gajons and assassination attempts by starving miners combine to sabotage production. Power obviously corrupts, and winning involves fleeing to another planet when sufficient lucre has been gathered atthe expense of the workforce.

A simple programconsisting mainly of text.

RUNNING THE PROGRAM

After typing RUN, the program displays the annual report and questions.

PROGRAM STRUCTURE

The lines of interest in this program are as follows:

170-420	Annual report.
440-790	Number of mines to build and sell.
810-940	Amount of stock to feed the workers.
1020-1290	Number of mines to be worked.
1300-1390	Updates of stock and population.
1320-1350	Determines Gajon attack.
1400-1800	Determines whether player is warned of rebellion, assassinated or wins.

```
10 REM PLANET RULER
20 REM *************************
30 REM
40 CLS
50 L=L-X
60 P=1.00
70 I=5
80 D=0
90 S=3650
100 K=350
110 M=4000
120 L=1000
130 Y=1
140 C=4
150 N=2576
160 D=0
170 CLS
180 PRINT"ANNUAL CONTROLLER'S REPORT FOR"
190 PRINT"YEAR ";N
200 PRINT
210 PRINT"SIR, DURING THE LAST YEAR:"
220 PRINT
230 PRINTD;"WORKERS DIED FROM STARVATION";
240 PRINT" AND";I;"IMMIGRANTS";
250 PRINT" CAME TO RAITH.";
260 IF Y>0 THEN300
270 P=INT(P/2)
280 PRINT
290 PRINT"THE GAJONS ATTACKED AND HALF OF THE PEOPLE ARE DEAD"
300 PRINT
310 PRINT"THE POPULATION IS NOW";P
320 PRINT:PRINT:PRINT"PRESS C TO CONTINUE"
```

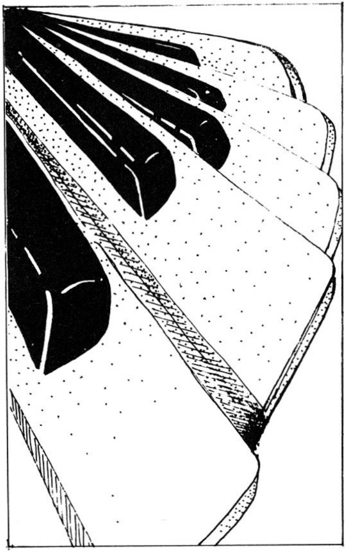
```
330 IF INKEY$<>"C" THEN330
340 CLS:PRINT:PRINT:PRINT
350 PRINT"THERE ARE";L;"MINE SHAFTS"
360 PRINT
370 PRINT"THE MINES PRODUCED";M;"TONNES"
380 PRINT"OF MINERALS AT";C;"TONNES/MINE BUT";
390 PRINT"THE GAJONS PIRATED";K;"TONNES"
400 PRINT"LEAVING";S;"TONNES IN STORAGE"
410 PRINT
420 PRINT:PRINT:PRINT"PRESS C TO CONTINUE"
430 IFINKEY$<>"C" THEN430
440 CLS
450 PRINT"THERE ARE";S;"TONNES OF"
460 PRINT"MINERALS IN STOCK"
470 C=RND(10)
480 V=C+17
490 PRINT
500 PRINT"IT CURRENTLY COSTS"
510 PRINTV;"TONNES"
520 PRINT"TO OPEN A MINE"
530 PRINT
540 PRINT"HOW MANY MINES DO YOU WANT TO"
550 INPUT"BUILD THIS YEAR ";X
560 X=INT(ABS(X))
570 IF X=0 THEN650
580 IF V*X<=S THEN630
590 PRINT"THERE IS ONLY ENOUGH STOCK TO"
600 PRINT"BUY MATERIALS FOR";
610 PRINT INT(S/V);"MINES"
620 GOTO530
630 L=L+X
640 S=S-V*X
```

```
650 FOR T=1TO1500:NEXT
660 CLS
670 PRINT"RAITH CURRENTLY HAS"
680 PRINTL;"MINE SHAFTS"
690 PRINT
700 PRINT"HOW MANY MINES ARE YOU WILLING"
710 INPUT"TO SELL ";X
720 X=INT(ABS(X))
730 IF X=0 THEN800
740 IF X<L THEN780
750 PRINT"CONTROLLER, YOU ONLY HAVE";L
760 PRINT"MINES. TRY SELLING A FEW LESS"
770 GOTO700
780 L=L-X
790 S=S+V*X
800 FOR T=1TO1500:NEXT
810 CLS
820 PRINT"THERE ARE";S;"TONNES OF"
830 PRINT"MINERALS IN STOCK"
840 PRINT
850 PRINT"HOW MUCH STOCK WILL YOU SELL TO"
860 INPUT"FEED THE WORK FORCE ";F
870 F=INT(ABS(F))
880 IF F<=S THEN950
890 PRINT
900 PRINT"SUCH GENEROSITY COMMANDER BUT"
910 PRINT"THERE ARE ONLY";S;"TONNES"
920 PRINT"OF MINERALS IN STOCK
930 PRINT
940 GOTO850
950 S=S-F
960 D=P-INT(F/20)
```

```
970 I=0
980 IF D>0 THEN1010
990 I=-D/2
1000 D=0
1010 FOR T=1TO1500:NEXT
1020 CLS
1030 PRINT"RAITH HAS";L;"MINE SHAFTS"
1040 PRINT
1050 PRINT"HOW MANY MINES WILL BE"
1060 INPUT"WORKED ";G
1070 G=INT(ABS(G))
1080 IF G=0 THEN1300
1090 IF G<=L THEN1140
1100 PRINT"IMAGINATIVE THINKING SIR BUT"
1110 PRINT"WE ONLY HAVE";L;"MINE SHAFTS"
1120 PRINT
1130 GOTO1050
1140 IF G<=S*2 THEN1220
1150 PRINT
1160 PRINT"0.5 TONNES OF STOCK HAS TO BE"
1170 PRINT"SOLD TO PURCHASE EQUIPMENT FOR"
1180 PRINT"EACH ACTIVE MINE. WE HAVE ENOUGH";
1190 PRINT"STOCK TO OPERATE";S*2;"MINES"
1200 PRINT
1210 GOTO1050
1220 IF G<=10*P THEN1300
1230 PRINT
1240 PRINT"COMMANDER, ONE WORKER CAN"
1250 PRINT"ONLY SUPERVISE 10 MINES. WE CAN"
1260 PRINT"ONLY HAVE A MAXIMUM OF";10*P
1270 PRINT"MINES OPERATING AT PRESENT."
1280 PRINT
```

```
1290 GOTO1050
1300 C=RND(6)
1310 M=G*C
1320 K=0
1330 Q=RND(5)
1340 IF INT(Q/2)<>Q/2 THEN1360
1350 K=INT(S/Q)
1360 S=S-K+M
1370 Q=5*(RND(2)-1)
1380 I=INT(C*(20*L+S)/P/100+1)
1390 P=P+I
1400 Q=INT(P/20)
1410 Y=INT(10*(3*(RND(2)-1)-.3))
1420 N=N+1
1430 IF P<Q THEN160
1440 REMIF Y>0 THEN3220
1450 REM P=INT(P/2)
1460 D=P-Q
1470 IF D>.5*P THEN1540
1480 IF D>.35*P THEN1610
1490 IF D>.2*P THEN1670
1500 P=Q
1510 FOR T=1TO2000 NEXT
1520 IF S>=100*P THEN1720
1530 GOTO170
1540 CLS
1550 PRINT@194,"YOU HAVE BEEN ASSASSINATED"
1560 FOR T=1TO3000 NEXT
1570 PRINT@352,"WOULD YOU LIKE ANOTHER GO ?(Y/N)"
1580 LINE INPUTA$
1590 IF LEFT$(A$,1)="Y" THEN RUN
1600 GOTO1810
```

```
1610 CLS
1620 PRINT@192,"THE WORKERS ARE HUNGRY AND"
1630 PRINT"THERE IS TALK OF REBELLION"
1640 PRINTD;"WORKERS HAVE STARVED TO"
1650 PRINT"DEATH."
1660 GOTO1500
1670 CLS
1680 PRINT@192,"THE WORKERS ARE HUNGRY."
1690 PRINTD;"WORKERS HAVE STARVED TO"
1700 PRINT"DEATH."
1710 GOTO1500
1720 CLS
1730 PRINT@160,"WELL DONE!!"
1740 PRINT"YOU HAVE GONE TO ANOTHER PLANET"
1750 PRINT"WITH THE CONTENTS OF THE"
1760 PRINT"TREASURY"
1770 PRINT"A TOTAL OF";S;"TONNES."
1780 PRINT
1790 FOR T=1TO4000:NEXT
1800 GOTO1570
1810 CLS
```



PIANO

DESCRIPTION

This program allows the DRAGON keyboard to be used as a piano keyboard. The program has a range of almost five octaves. Sharps are produced by hitting the space bar before pressing a key to sound the note.

The keys being used are:
Z-< First or bottom octave.
A-L Second octave.
Q-I Third octave.
1-8 Fourth octave and fifth or top octave.

To switch between the fourth and fifth octaves on the top row of keys hit the ENTER key.

RUNNING THE PROGRAM

After typing RUN use the DRAGON keyboard as a piano and play some music.

PROGRAM STRUCTURE

The lines of interest in the program are as follows:

80	Checks for sharps.
90	Checks to see if top row of keys requires a change of octave.

100-420 Checks key pressed, gives its appropriate note and makes it a sharp if required.
460-470 Plays note in first octave.
510-520 Plays note in second octave.
560-570 Plays note in third octave.
610-630 If variable B, set by line 90, is even then play note in fourth octave.
670-680 Plays note in fifth octave.

```
10 REM PIANO
20 REM ****************************
30 REM
40 CLS
50 PRINT@202,"PRESS A KEY"
60 A$=INKEY$
70 IFA$=""THEN60
80 IFA$=" "THEN A=1:GOTO60
90 IFASC(A$)=13THENB=B+1:GOTO60
100 IFA$="Z" THEN B$="C":IFA=1 THENB$="C#" :A=0:GOTO460:ELSEGOTO460
110 IFA$="X" THEN B$="D":IFA=1 THEN B$="D#":A=0:GOTO460:ELSE460
120 IFA$="C" THENB$="E":A=0:GOTO460
130 IFA$="V" THEN B$="F":IFA=1 THENB$="F#":A=0:GOTO460:ELSE460
140 IFA$="B" THEN B$="G":IFA=1 THENB$="G#":A=0:GOTO460:ELSE460
150 IFA$="N" THEN B$="A":IFA=1 THENB$="A#":A=0:GOTO460:ELSE460
160 IFA$="M" THEN B$="B":A=0:GOTO460
170 IFA$="," THEN B$="C":IFA=1 THENB$="C#":A=0:GOTO510:ELSE510
180 IFA$="A" THEN B$="C":IFA=1 THENB$="C#":A=0:GOTO510:ELSE510
190 IFA$="S" THEN B$="D":IFA=1 THENB$="D#":A=0:GOTO510:ELSE510
200 IFA$="D" THEN B$="E":A=0:GOTO510
210 IFA$="F" THEN B$="F":IFA=1 THENB$="F#":A=0:GOTO510:ELSE510
220 IFA$="G" THEN B$="G":IFA=1 THENB$="G#":A=0:GOTO510:ELSE510
230 IFA$="H" THEN B$="A":IFA=1 THENB$="A#":A=0:GOTO510:ELSE510
240 IFA$="J" THEN B$="B":A=0:GOTO510
250 IFA$="K" THEN B$="C":IFA=1 THENB$="C#":A=0:GOTO560:ELSE560
260 IFA$="Q" THEN B$="C":IFA=1 THENB$="C#":A=0:GOTO560:ELSE560
270 IFA$="W" THEN B$="D":IFA=1 THENB$="D#":A=0:GOTO560:ELSE560
280 IFA$="E" THEN B$="E":A=0:GOTO560
290 IFA$="R" THEN B$="F":IFA=1 THENB$="F#":A=0:GOTO560:ELSE560
300 IFA$="T" THEN B$="G":IFA=1 THENB$="G#":A=0:GOTO560:ELSE560
310 IFA$="Y" THEN B$="A":IFA=1 THENB$="A#":A=0:GOTO560:ELSE560
320 IFA$="U" THEN B$="B":A=0:GOTO560
```

```
330 IFA$="I" THEN B$="C":IFA=1 THENB$="C#":A=0:GOTO610:ELSE610
340 IFA$="1" THEN B$="C":IFA=1 THENB$="C#":A=0:GOTO610:ELSE610
350 IFA$="2" THEN B$="D":IFA=1 THENB$="D#":A=0:GOTO610:ELSE610
360 IFA$="3" THEN B$="E":A=0:GOTO610
370 IFA$="4" THEN B$="F":IFA=1 THENB$="F#":A=0:GOTO610:ELSE610
380 IFA$="5" THEN B$="G":IFA=1 THENB$="G#":A=0:GOTO610:ELSE610
390 IFA$="6" THEN B$="A":IFA=1 THENB$="A#":A=0:GOTO610:ELSE610
400 IFA$="7" THEN B$="B":A=0:GOTO610
410 IFA$="8" THEN B$="C":IFA=1THENB$="C#":A=0:GOTO670:ELSE670
420 GOTO60
430 REM
440 REM 1ST OCTAVE
450 REM
460 PLAY"O1"+B$
470 GOTO60
480 REM
490 REM 2ND OCTAVE
500 REM
510 PLAY"O2"+B$
520 GOTO60
530 REM
540 REM 3RD OCTAVE
550 REM
560 PLAY"O3"+B$
570 GOTO60
580 REM
590 REM 4TH OCTAVE
600 REM
610 IFB/2<>INT(B/2)THEN670
620 B=0:PLAY"O4"+B$
630 GOTO60
640 REM
```

```
650 REM 5TH OCTAVE
660 REM
670 PLAY"O5"+B$
680 GOTO60
```

3D GRAPH

DESCRIPTION

Building on the routines for plotting two dimensional functions, it can be relatively easy to design a program for plotting in three dimensions. This program plots a 3D graph defined by the function in line 130. It can therefore be used to plot other three dimensional graphs by simply changing this function to the required one. For example change line 130 to read:
DEF FNA(Z)=90*EXP(−Z*Z/600)
and compare the resulting graph with the first.

RUNNING THE PROGRAM

After typing RUN the computer will begin to display the graph in its highest resolution. The screen will flash to indicate that the graph has been completely drawn. The graph will then remain on the screen until any key is pressed.

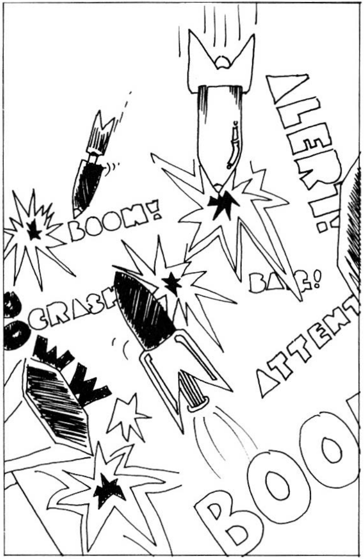
PROGRAM STRUCTURE

Below is a listing of the more interesting lines that make up the program:

130 Function of the graph.
170-400 Actual plotting routine of graph.
290 If point to be set is out of sight do not plot.
380 Check to see whether a point on the left hand side of the graph has been plotted. If this is so then plot a symmetrical point on the right hand side.
440-490 Flash screen once the graph has been plotted.

```
10 REM 3D GRAPH IN HIRES
20 REM ***********************
30 REM
40 REM
50 REM HIRES MODE & SCREEN
60 REM
70 PMODE4,1
80 COLOR0,5:PCLS
90 SCREEN1,1
100 REM
110 REM FUNCTION OF GRAPH
120 REM
130 DEF FNA(Z)=38*(SIN(Z/24)+.48*SIN(3*Z/24))+20
140 REM
150 REM LOOP FOR HALF OF X VALUES
160 REM
170 FORX=-100TO0
180 K=5
190 L=0
200 P=1
210 Z1=0
220 Y1=K*INT(SQR(10000-X*X)/K)
230 FOR Y=Y1 TO -Y1 STEP-K
240 Z=INT(80+FNA(SQR(X*X+Y*Y))-.707106*Y)
250 REM
260 REM IF OUT OF SIGHT
270 REM DO NOT SET
280 REM
290 IFZ<L THEN390
300 M=1
310 L=Z
320 PSET(128+M*X,183-Z,0)
```

```
330 REM
340 REM SYMETRICAL GRAPH
350 REM PLOT OTHER HALF OF
360 REM X COORDINATES
370 REM
380 IFM=1 THEN M=-1:GOTO320
390 NEXT
400 NEXT
410 REM
420 REM FLASH SCREENS
430 REM
440 FORT=1TO10
450 SCREEN1,0
460 FORTD=1TO500:NEXT
470 SCREEN1,1
480 FORTD=1TO500:NEXT
490 NEXT
500 REM
510 REM WAIT FOR KEY PRESS
520 REM
530 IFINKEY$=""THEN530
540 CLS
```



SOUND EFFECTS

DESCRIPTION

A simple program which uses the DRAGON SOUND and PLAY commands to produce nine interesting sound effects.

Should the user decide to write his own games program, he will find this program to be of great use when deciding which sound effects to add to spruce up the end result of his work.

RUNNING THE PROGRAM

After loading the program and typing RUN the display will invite the user to make his choice of sound by hitting the appropriate key, i.e. hitting 1 the user will hear a mystery noise. To retire from the program the user must hit the space bar.

PROGRAM STRUCTURE

The main lines of interest in this program are as follows:

40-200 Print the display on the screen and input choice of sound effect.
280-340 Subroutine to generate mystery noise.
380-420 Subroutine to generate computer mania.
470-550 Subroutine to generate explosion.
590-650 Subroutine to generate bombardment.
690-720 Subroutine to generate alarm.
760-820 Subroutine to generate laser fire.
860-890 Subroutine to generate siren.
930-950 Subroutine to generate silly noise.
990-1090 Subroutine to generate public telephone.

```
10 REM SOUND EFFECTS
20 REM *************************
30 REM
40 CLS
50 PRINT@5,"**  SOUND  EFFECTS  **"
60 PRINT
70 PRINT"     1 MYSTERY NOISE"
80 PRINT"     2 COMPUTER MANIA"
90 PRINT"     3 EXPLOSION"
100 PRINT"     4 BOMBARDMENT"
110 PRINT"     5 ALARM"
120 PRINT"     6 LASER FIRE"
130 PRINT"     7 SIREN"
140 PRINT"     8 SILLY NOISE"
150 PRINT"     9 PUBLIC TELEPHONE"
160 PRINT@426,"WHICH ONE ?"
170 A$=INKEY$
180 IFA$=""THEN170
190 IFA$=" " THEN1100
200 IFASC(A$)<49 OR ASC(A$)>57 THEN170
210 ON VAL(A$) GOSUB280,330,470,590,690,760,860,930,990
220 PRINT@426,"              "
230 FOR TD=1TO1000:NEXT
240 GOTO160
250 REM
260 REM MYSTERY NOISE
270 REM
280 FORL=250TO200 STEP-2
290 SOUNDL,1
300 NEXT
310 FORL=200TO250 STEP2
320 SOUNDL,1
```

```
330 NEXT
340 RETURN
350 REM
360 REM COMPUTER MANIA
370 REM
380 FORX=1TO100
390 Y=RND(150)+100
400 Z=RND(3)
410 SOUNDY,Z
420 NEXT
430 RETURN
440 REM
450 REM EXPLOSION
460 REM
470 FORX=1TO5
480 PLAY"O1;T155;L255;V10;G"
490 NEXT
500 PLAY"O1;L255;V13;F;L255;V16;E;L255;V22;D;L255;V27;C;L255;V30;L255;A"
510 FOR X=1TO5
520 PLAY"O1V31G"
530 NEXT
540 PLAY"O1;T120;L255;P255;V30;A;T90;L255;P255;V27;B;T50;L255;P255;V22;C;T30;L25
5;P255;V18;D"
550 RETURN
560 REM
570 REM BOMBARDMENT
580 REM
590 FORY=1TO4
600 FORTD=1TO400:NEXT
610 PLAY"O5T2L30V31BA#AG#CF#FED#DC#C"
620 FORTD=1TO400:NEXT
630 GOSUB470
```

```
640 NEXT
650 RETURN
660 REM
670 REM ALARM
680 REM
690 FORX=1TO30
700 PLAY"O5T2V31L10EP10"
710 NEXT
720 RETURN
730 REM
740 REM LASER FIRE
750 REM
760 FORX=1TO4
770 PLAY"O5T2V31L255BP255BP255BP255BP255BP255"
780 PLAY"L255BP255BP255BP255BP255BP255BP255BP255BP255BP255BP255BP255BP255BP255"
790 PLAY"L255BA#AG#GF#FED#DC#C"
800 FORTD=1TO1000:NEXT
810 NEXT
820 RETURN
830 REM
840 REM SIREN
850 REM
860 FORX=1TO10
870 PLAY"T2V31O4L4C#O3L4C#"
880 NEXT
890 RETURN
900 REM
910 REM SILLY NOISE
920 REM
930 PLAY"V31O4T155L255;1,2;3;4;5;6;7;8;9;10;11;12"
940 PLAY"O5T155L255;1;2;3;4;5;6;7;8;9,10;11;12"
950 RETURN
```

```
960 REM
970 REM PUBLIC TELEPHONE
980 REM
990 FORX=1TO6
1000 PLAY"O3T2V31L4AP10A"
1010 IFX<>6 THEN FORTD=1TO1500:NEXT
1020 NEXT
1030 SOUND235,1
1040 SOUND240,2
1050 FORZ=1TO12
1060 PLAY"O4L20C#P20C#P20"
1070 NEXT
1080 PLAY"O4L1T1C#"
1090 RETURN
1100 CLS
```

DRAGON KALEIDOSCOPE

DESCRIPTION

This simple program displays a constantly changing colourful pattern on the screen, reminiscent of a child's toy kaleidoscope.

RUNNING THE PROGRAM

After loading the program just type RUN and watch the pattern grow. To freeze the program hit any key. Hit any key again to end the program.

PROGRAM STRUCTURE

There are but few lines of interest in this program which are as follows:

200 Prints top left hand display of the kaleidoscope.
250 Prints top right hand display of the kaleidoscope.
300 Prints bottom left hand display of the kaleidoscope.
350 Prints bottom right hand display of the kaleidoscope.

```
10 REM KALEIDOSCOPE
20 REM *************************
30 REM
40 CLS
50 PRINT@227,"** DRAGON KALEIDOSCOPE **"
60 FORTD=1TO1500:NEXT
70 CLS0
80 REM
90 REM C=COLUMN
100 REM R=ROW
110 REM CL=COLOUR
120 REM
130 FORC=15TO1 STEP-1
140 FORR=7TO0 STEP-1
150 CL=RND(8)-1
160 REM
170 REM PRINT TOP LEFT HAND
180 REM CORNER
190 REM
200 PRINT@R*32+C,CHR$(143+16*CL);
210 REM
220 REM PRINT TOP RIGHT HAND
230 REM CORNER
240 REM
250 PRINT@R*32+(31-C),CHR$(143+16*CL);
260 REM
270 REM PRINT BOTTOOM LEFT HAND
280 REM CORNER
290 REM
300 PRINT@(15-R)*32+C,CHR$(143+16*CL);
310 REM
320 REM PRINT BOTTOM RIGHT HAND
```

```
330 REM CORNER
340 REM
350 PRINT@(15-R)*32+31-C,CHR$(143+16*CL);
360 IF INKEY$<>""THEN400
370 NEXT
380 NEXT
390 GOTO130
400 IFINKEY$<>""THEN420
410 GOTO400
420 CLS
```

GRAPH PLOTTING

DESCRIPTION

This program uses the highest resolution available on the DRAGON to plot a graph of SIN(x), COS(x), or SIN(x) with COS(x), depending on the user's choice.

The program uses the DRAGON command SET to plot the graphs. Since it is rathertime consuming to write in hi-res, the titles of each graph are 'flashed up' on the normal lo-res screen before each graph is actually plotted.

RUNNING THE PROGRAM

After typing RUN a display showing the choices of graph available will appear on the screen. Once the graph has been drawn it will stay on the screen until a key is pressed. If any key is then pressed again, the program will end.

PROGRAM STRUCTURE

The lines of interest in this program are as follows:
40-120 Print main display and input graph type.
210-580 Subroutine to draw the graph axis on hi-res screen.
620-700 Subroutine to set graph of SIN(x).
740-810 Subroutine to set graph of COS(x).
870-1010 Subroutine to set graph of SIN(x) and COS(x).

```
 10 REM SIN & COS CURVES
 20 REM ************************
 30 REM
 40 CLS
 50 PRINT@33,"** GRAPHS OF SIN(X) & COS(X) **"
 60 PRINT@135,"PRESS 1 FOR SIN(X)"
 70 PRINT@199,"PRESS 2 FOR COS(X)"
 80 PRINT@263,"PRESS 3 FOR BOTH"
 90 PRINT@419,"PRESS ANY OTHER KEY TO END"
 100 A$=INKEY$
 110 IFA$=""THEN100
 120 IFASC(A$)<49 OR ASC(A$)>51 THEN1020
 130 GOSUB210
 140 ON VAL(A$) GOSUB620,740,870
 150 B$=INKEY$
 160 IFB$=""THEN150
 170 RUN
 180 REM
 190 REM DRAW AXIS
 200 REM
 210 PMODE4,1:PCLS
 220 SCREEN1,1
 230 COLOR5,0
 240 DRAW"BM15,90R225"
 250 DRAW"BM235,95E5H5"
 260 DRAW"BM238,108E6"
 270 DRAW"BM238,102F6"
 280 DRAW"BM50,15D160
 290 DRAW"BM45,20E5F5"
 300 DRAW"BM50,40L10"
 310 DRAW"BM50,140L10"
 320 DRAW"BM36,136D6"
```

```
330 DRAW"BM32,140L4"
340 DRAW"BM16,10L6D4R4"
350 DRAW"BM10,14D6"
360 CIRCLE(26,16),6,5,1,.35,.65
370 DRAW"BM26,19E6"
380 DRAW"BM26,13F6"
390 CIRCLE(32,16),6,5,1,.85,.15
400 CIRCLE(44,96),4
410 DRAW"BM36,36D6"
420 DRAW"BM100,90D10"
430 DRAW"BM92,104R6"
440 DRAW"BM93,104D4"
450 DRAW"BM97,104D4"
460 DRAW"BM150,90D10"
470 CIRCLE(154,104),3,5,1,.6,.25
480 DRAW"BM154,108G2R6"
490 DRAW"BM160,104R6"
500 DRAW"BM161,104D4"
510 DRAW"BM165,104D4"
520 DRAW"BM200,90D10"
530 CIRCLE(190,104),3,5,1,.6,.25
540 CIRCLE(190,109),3,5,1,.8,.4
550 DRAW"BM195,104R6"
560 DRAW"BM196,104D4"
570 DRAW"BM200,104D4"
580 RETURN
590 REM
600 REM SET GRAPH OF SIN(X)
610 REM
620 FORTD=1TO1500:NEXT
630 CLS
640 PRINT@236,"SIN(X)"
```

```
650 FORTD=1TO1500:NEXT
660 SCREEN1,1
670 FORX=200TO916
680 PSET(X/4,90+50*SIN(X/64),5)
690 NEXT
700 RETURN
710 REM
720 REM SET GRAPH OF COS(X)
730 REM
740 FORTD=1TO1500:NEXT
750 CLS
760 PRINT@236,"COS(X)"
770 FORTD=1TO1500:NEXT
780 SCREEN1,1
790 FORX=200TO916
800 PSET(X/4,90+50*COS(X/64),5)
810 NEXT
820 RETURN
830 REM
840 REM SET GRAPHS OF SIN(X) &
850 REM COS(X)
860 REM
870 PMODE4,1
880 GOSUB220
890 CLS
900 PRINT@233,"FIRST SIN(X)"
910 FORTD=1TO1500:NEXT
920 SCREEN1,1
930 GOSUB670
940 FORTD=1TO3000:NEXT
950 CLS
960 PRINT@234,"NOW COS(X)"
```

```
970 FORTD=1TO1500:NEXT
980 PMODE4,1
990 GOSUB220
1000 GOSUB790
1010 RETURN
1020 CLS
```

RHINO

DESCRIPTION

The aim of the game is to make your way through the jungle and reach home, while avoiding wild and dangerous rhinos. A map of the jungle with the position of the trees, the house and the player's current position are displayed on the screen as 'H' and '&' characters respectively.

The opponents – the player and the rhinos – are hidden from each other by dense jungle. Once the rhino has seen you, its position is/are (depending on how far into the game the player has ventured) displayed as the letter 'R' and then the race is on. The rhino will charge, moving one step towards the player. It is up to the player to escape amongst the trees; seemingly an easy thing to do, but the rhinos are very smart critters. If a rhino catches up with the player, it's wham, and the game is lost. Each time the player outwits the rhinos one more comes to their aid, until he has to avoid a maximum of20 such beasts.

Without giving too much away a tip for the player is to try to trap the rhinos in a downward facing U-shaped group of trees. The player will be able to move diagonally out of the area, while the rhinos remain feeding on their frustration.

RUNNING THE PROGRAM

After typing RUN, the computer will indicate the number of rhinos hidden (1 initially). To move the man in the required direction use the number keys, thus:

7	8	9
4		6
1	2	3

PROGRAM STRUCTURE

The lines of interest in this program are as follows:

40	Limit top of BASIC memory to address 14000 so that address 14001 can be used to store the number of rhinos present.
50	Number of rhinos = 1 initially.
60	Dimension the arrays.
120-3280	Play the game.
640-960	Set up the display.
1000-1310	Move the man in the required direction.

```
          10 REM RHINO
20 REM ************************
30 REM
40 CLEAR200,14000
50 POKE14001,1:REM NO. OF RHINOS
60 DIM H(2),R(20,2),V(20)
70 CLS
80 GOSUB580
90 REM
100 REM PLAY GAME
110 REM
120 NF=0
130 X=H(1)
140 Y=H(2)
150 A$="&"
160 PRINT@0,"YOUR MOVE"
170 M$=INKEY$:IFM$=""THEN170
180 PRINT@0
190 M=VAL(M$)
200 IF M=0 THEN390
210 GOSUB1320
220 IF WIN>0 THEN540
230 GOSUB930
240 H(1)=X
250 H(2)=Y
260 RN=0
270 IF NF<PEEK(14001) THEN GOSUB1640
280 IF RN=0 THEN GOSUB2670
290 IF WIN>0 THEN460
300 IF RN=0 THEN130
310 FOR I=1 TO 15
320 IF V(I)=0 THEN130
330 AD=1024+32*R(I,1)+R(I,2)
```

```
340 POKEAD,82
350 NF=NF+1
360 NEXTI
370 GOTO130
380 GOTO3290
390 CLS
400 PRINT" YOU RESIGN"
410 INPUT"DO YOU WANT ANOTHER GO ";R$
420 IF RIGHT$(R$,1)<>"N" AND RIGHT$(R$,2)<>"NO" THEN450
430 CLS
440 GOTO3290
450 CLEAR200,14000:GOTO60
460 FORI=1 TO 4
470 POKE1024+32*H(1)+H(2),127+16*I
480 FOR TD=1 TO 1000:NEXT
490 NEXTI
500 FOR TD=1 TO 4000:NEXT
510 CLS
520 WIN=0
530 GOTO410
540 CLS:PRINT" YOU WIN"
550 IFPEEK(14001)<20.THEN POKE14001,PEEK(14001)+1
560 WIN=0
570 GOTO410
580 CLS
590 IF PEEK(14001)>1 THENPRINT"THERE ARE";PEEK(14001);"RHINOS":GOTO640
600 PRINT"THERE IS ONE RHINO"
610 REM
620 REM SET UP DISPLAY
630 REM
640 FORI=1 TO 14
650 A$=""
660 FORJ=1 TO 34
```

```
670 P=RND(100)
680 IF P>80 THEN A$=A$+"^" GOTO700
690 A$=A$+" "
700 NEXTJ
710 PRINTA$;
720 NEXTI
730 PRINT@44,"       ";
740 PRINT@76,"       ";
750 PRINT@108,"   H   ";
760 PRINT@140,"       ";
770 PRINT@172,"       ";
780 PRINT@434,"    ";
790 PRINT@466,"    ";
800 H(1)=14 H(2)=19
810 FOR I=1 TO PEEK(14001)
820 Y=RND(31)
830 X=RND(5)+4
840 R(I,1)=X
850 R(I,2)=Y
860 D=PEEK(1024+32*X+Y)
870 IF D=34 THEN820
880 NEXTI
890 FOR I=1 TO 20
900 V(I)=0
910 NEXTI
920 RETURN
930 IF ABS(M-5)>4 THEN M=5
940 C$=" "
950 GOSUB3220
960 C$=A$
970 REM
980 REM MOVE MAN
990 REM
```

```
1000 ON M GOTO 1020,1060,1090,1130,1160
1010 IF M>5 THEN ON M-5 GOTO 1180,1210,1250,1280
1020 X=X+1
1030 Y=Y-1
1040 GOSUB3220
1050 RETURN
1060 X=X+1
1070 GOSUB3220
1080 RETURN
1090 X=X+1
1100 Y=Y+1
1110 GOSUB3220
1120 RETURN
1130 Y=Y-1
1140 GOSUB3220
1150 RETURN
1160 GOSUB3220
1170 RETURN
1180 Y=Y+1
1190 GOSUB3220
1200 RETURN
1210 X=X-1
1220 Y=Y-1
1230 GOSUB3220
1240 RETURN
1250 X=X-1
1260 GOSUB3220
1270 RETURN
1280 X=X-1
1290 Y=Y+1
1300 GOSUB3220
1310 RETURN
1320 ON M GOTO 1340,1410,1440,1470,1500
```

```
1330 IF M>5 THEN ON M-5 GOTO 1510,1540,1570,1600
1340 IF X=15 OR Y=0 THEN M=0:RETURN
1350 AD=1024+32*(X+1)+Y-1
1360 T=PEEK(AD)
1370 IF T=94 THEN M=0
1380 IF T=72 THEN WIN=2:M=0
1390 IF T=82 THEN WIN=1:GOTO290
1400 RETURN
1410 IF X=15 THEN M=0:RETURN
1420 AD=1024+32*(X+1)+Y
1430 GOTO1360
1440 IF X=15 OR Y=31 THEN M=0:RETURN
1450 AD=1024+32*(X+1)+Y+1
1460 GOTO1360
1470 IF Y=0 THEN M=0:RETURN
1480 AD=1024+32*X+Y-1
1490 GOTO1360
1500 RETURN
1510 IF Y=31 THEN M=0:RETURN
1520 AD=1024+32*X+Y+1
1530 GOTO1360
1540 IF X=1 OR Y=0 THEN M=0:RETURN
1550 AD=1024+32*(X-1)+Y-1
1560 GOTO1360
1570 IF X=1 THEN M=0:RETURN
1580 AD=1024+32*(X-1)+Y
1590 GOTO1360
1600 IF X=1 OR Y=31 THEN M=0:RETURN
1610 AD=1024+32*(X-1)+Y+1
1620 GOTO1360
1630 RETURN
1640 FOR I=NF+1 TO PEEK(14001)
1650 IF V(I)=1 THEN2490
```

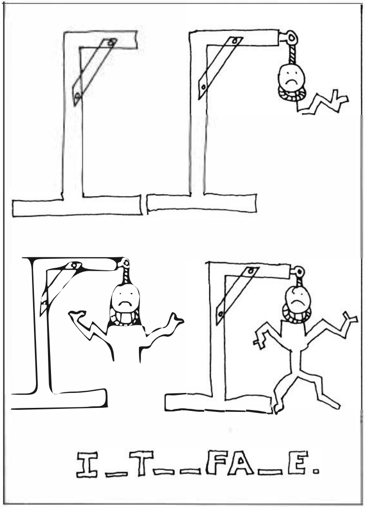
```
1660 XR=R(I,1)
1670 YR=R(I,2)
1680 XH=H(1)
1690 YH=H(2)
1700 IF (XR-XH)<>(YH-YR) THEN1900
1710 IF XH<XR THEN1810
1720 N=XH-XR-1
1730 FOR J=1 TO N
1740 AD=1024+32*(XR+J)+YR-J
1750 D=PEEK(AD)
1760 IF D=94 THEN2490
1770 NEXTJ
1780 RN=1
1790 V(I)=1
1800 GOTO2490
1810 N=XR-XH-1
1820 FOR J=1 TO N
1830 AD=1024+32*(XR-J)+YR+J
1840 D=PEEK(AD)
1850 IF D=94 THEN2490
1860 NEXTJ
1870 RN=1
1880 V(I)=1
1890 GOTO2490
1900 IF YR<>YH THEN2100
1910 IF XH<XR THEN2010
1920 N=XH-XR-1
1930 FOR J=1 TO N
1940 AD=1024+32*(XR+J)+YR
1950 D=PEEK(AD)
1960 IF D=94 THEN2490
1970 NEXTJ
1980 RN=1
```

```
1990 V(I)=1
2000 GOTO2490
2010 N=XR-XH-1
2020 FOR J=1 TO N
2030 AD=1024+32*(XR-J)+YR
2040 D=PEEK(AD)
2050 IF D=94 THEN2490
2060 NEXTJ
2070 RN=1
2080 V(I)=1
2090 GOTO2490
2100 IF (XH-XR)<>(YH-YR) THEN2300
2110 IF XH<XR THEN2210
2120 N=XH-XR-1
2130 FOR J=1 TO N
2140 AD=1024+32*(XR+J)+YR+J
2150 D=PEEK(AD)
2160 IF D=94 THEN2490
2170 NEXTJ
2180 RN=1
2190 V(I)=1
2200 GOTO2490
2210 N=XR-XH-1
2220 FOR J=1 TO N
2230 AD=1024+32*(XR-J)+YR-J
2240 D=PEEK(AD)
2250 IF D=94 THEN2490
2260 NEXTJ
2270 RN=1
2280 V(I)=1
2290 GOTO2490
2300 IF XR<>XH THEN2490
2310 IF YR<YH THEN2410
```

```
2320 N=YR-YH-1
2330 FOR J=1 TO N
2340 AD=1024+32*XR+YR-J
2350 D=PEEK(AD)
2360 IF D=94 THEN2490
2370 NEXTJ
2380 RN=1
2390 V(I)=1
2400 GOTO2490
2410 N=YH-YR-1
2420 FOR J=1 TO N
2430 AD=1024+32*XR+YR+J
2440 D=PEEK(AD)
2450 IF D=94 THEN2490
2460 NEXTJ
2470 RN=1
2480 V(I)=1
2490 NEXTI
2500 IF RN=1 THEN GOSUB2520
2510 RETURN
2520 I=1
2530 IF I=PEEK(14001)+1 THEN RETURN
2540 IF V(I)=0 THEN I=I+1:GOTO2530
2550 IF I=1 THEN I=I+1:GOTO2530
2560 IF V(I-1)=1 THEN I=I+1:GOTO2530
2570 V(I-1)=1
2580 V(I)=0
2590 SX=R(I-1,1)
2600 R(I-1,1)=R(I,1)
2610 R(I,1)=SX
2620 SY=R(I-1,2)
2630 R(I-1,2)=R(I,2)
2640 R(I,2)=SY
```

```
2650 I=I-1
2660 GOTO2540
2670 FOR I=1 TO PEEK(14001)
2680 XH=H(1)
2690 YH=H(2)
2700 IF V(I)=0 THEN RETURN
2710 X=R(I,1)
2720 Y=R(I,2)
2730 A$="R"
2740 XD=X-XH
2750 YD=Y-YH
2760 RM(1)=PEEK(1024+32*(X+1)+Y-1)
2770 RM(2)=PEEK(1024+32*(X+1)+Y)
2780 RM(3)=PEEK(1024+32*(X+1)+Y+1)
2790 RM(4)=PEEK(1024+32*X+Y-1)
2800 RM(6)=PEEK(1024+32*X+Y+1)
2810 RM(7)=PEEK(1024+32*(X-1)+Y-1)
2820 RM(8)=PEEK(1024+32*(X-1)+Y)
2830 RM(9)=PEEK(1024+32*(X-1)+Y+1)
2840 IF RM(2)=94 AND RM(4)=94 THEN RM(1)=94
2850 IF RM(4)=94 AND RM(8)=94 THEN RM(7)=94
2860 IF RM(8)=94 AND RM(6)=94 THEN RM(9)=94
2870 IF RM(6)=94 AND RM(2)=94 THEN RM(3)=94
2880 FOR J=1 TO 9
2890 IF RM(J)=94 THEN2940
2900 IF RM(J)=102 THEN WIN=1:RETURN
2910 IF RM(J)=72 THEN2940
2920 ON J GOSUB 3040,3060.3080,3100,3120
2930 IF J>5 THEN ON J-5 GOSUB 3140,3160,3180,3200
2940 NEXTJ
2950 MV=110
2960 FOR J=1 TO 9
```

```
2970 IF RM(J)<=MV THEN M=J:MV=RM(J)
2980 NEXTJ
2990 GOSUB930
3000 R(I,1)=X
3010 R(I,2)=Y
3020 NEXTI
3030 RETURN
3040 RM(1)=ABS(XD+1)+ABS(YD-1)
3050 RETURN
3060 RM(2)=ABS(XD+1)+ABS(YD)
3070 RETURN
3080 RM(3)=ABS(XD+1)+ABS(YD+1)
3090 RETURN
3100 RM(4)=ABS(XD)+ABS(YD-1)
3110 RETURN
3120 RM(5)=ABS(XD)+ABS(YD)
3130 RETURN
3140 RM(6)=ABS(XD)+ABS(YD+1)
3150 RETURN
3160 RM(7)=ABS(XD-1)+ABS(YD-1)
3170 RETURN
3180 RM(8)=ABS(XD-1)+ABS(YD)
3190 RETURN
3200 RM(9)=ABS(XD-1)+ABS(YD+1)
3210 RETURN
3220 AD=1024+32*X+Y
3230 IF C$=" " THEN POKEAD,96:RETURN
3240 IF C$<>"&" THEN3260
3250 POKEAD,102:RETURN
3260 IF C$<>"R" THEN3280
3270 POKE AD,82
3280 RETURN
3290 CLS
```



HANGMAN

DESCRIPTION

Hangman is the computerised version of the popular children's game, although playing against the computer enlivens the proceedings well enough for adults to find it challenging.

The player guessing the word has ten moves in which to work his miracle (the computer always plays as the questioner), no second chances are allowed as first the ground appears followed by the crossbar, the strut, the noose, the hanged man's head, his body, arms, legs and finally his feet. If the correct answer has not been found by this time, the player who is trying to find the word loses.

In this particular program the computer has a vocabulary comprising 200 of the most commonly mis-spelt words in the English language.

It is possible to cheat by looking at the vocabulary listing but most people would, one hopes, try to use this game in a more sporting way. By changing the words in the Data statements in lines 2480 to 3120 – always ensuring that there are 200 words therein – it is possible to use the game as an aid to learning a foreign language.

RUNNING THE PROGRAM

Once the program is loaded type RUN and follow the instructions.

PROGRAM STRUCTURE

The lines of interest in this program are as follows:

40-330	Instructions
400-520	Choose a word from the Data statements
560-590	Input a letter.
630-690	Not a letter.
730-870	Guessed the word.
910-940	Letter in word.
980-1050	Print letter in correct position on screen.
1090-1140	Letter is not in word.
1180-1240	Display the man's feet.
1280-1380	Failed to guess the word before the man was hung.
1420-1470	Display the base of the rig.
1510-1580	Display the upright.
1620-1680	Display the horizontal.
1720-1780	Display the diagonal support.
1820-1860	Display the rope.
1900-1980	Display the man's head.
2020-2140	Display the man's body.
2180-2220	Display the man's hands.
2260-2330	Display the man's legs.
2480-3120	Data containing about 200 words.

```
10 REM HANGMAN
20 REM *************************
30 REM
40 CLS:PRINT@108,"HANGMAN"
50 PRINT@298,"DO YOU NEED"
60 PRINTTAB(9)"INSTRUCTIONS ?"
70 PRINTTAB(7)"(REPLY Y FOR 'YES'"
80 PRINTTAB(9)"OR N FOR 'NO')"
90 X$=INKEY$
100 IFX$=""THEN90
110 IFX$="N"THEN350
120 CLS:PRINT"ALWAYS REPLY Y FOR 'YES' AND N   FOR 'NO'."
130 PRINT:PRINT"I WILL PRINT "
140 PRINTTAB(13)"'WORD------'"
150 PRINT" EACH '-' STANDS FOR A MISSING"
160 PRINT" LETTER.
170 PRINT:PRINT"WHEN I PRINT "
180 PRINTTAB(13)"'PRESS ANY LETTER',";
190 PRINT" FOLLOW THE INSTRUCTION."
200 PRINT@480," (PRESS SPACE BAR TO CONTINUE)";
210 IF INKEY$<>" "THEN210
220 CLS:PRINT"IF A LETTER IS IN THE UNKNOWN"
230 PRINT" WORD MORE THAN ONCE YOU WILL"
240 PRINT" NEED TO PRESS THE LETTER MORE"
250 PRINT" THAN ONCE."
260 PRINT"    E.G. 'APPLE' HAS TWO P'S,"
270 PRINT" PRESS P ONCE AND I REPLY "
280 PRINTTAB(9)"'WORD--P---'"
290 PRINT" PRESS P A SECOND TIME AND I     REPLY "
300 PRINTTAB(9)"'WORD--PP---'"
310 PRINT:PRINT"YOU HAVE TO COMPLETE THE WORD"
320 PRINT" BEFORE I HANG THE MAN."
```

```
330 PRINT@484,"PRESS ANY KEY TO START.";
340 IFINKEY$=""THEN340
350 CLS:PRINT@103,"HANGMAN"
360 REM
370 REM CHOSE A WORD FROM
380 REM THE DATA
390 REM
400 RA=RND(197)
410 FORI=1TORA
420 READA$:IFA$="END"THEN830
430 NEXT
440 RESTORE
450 LA=LEN(A$)
460 Z$=A$
470 D$=""
480 FORI=1TOLA
490 D$=D$+"-"
500 NEXT
510 PRINT@230,"WORD=";D$
520 N=0:L=0
530 REM
540 REM INPUT A LETTER
550 REM
560 PRINT@423," PRESS ANY LETTER "
570 L$=INKEY$
580 IFL$=""THEN570
590 IFASC(L$)>64 AND ASC(L$)<91THEN670
600 REM
610 REM NOT A LETTER
620 REM
630 PRINT@423,"'";L$;"' IS NOT A LETTER "
640 FORTD=1TO2000:NEXT
```

```
650 PRINT@423,STRING$(18," ")
660 GOTO560
670 GOSUB840
680 IF N<LA THEN560
690 IFN=LA+1THEN790
700 REM
710 REM FINISHED
720 REM
730 FORI=1TO8
740 PRINT@331,"YOU WIN!!!"
750 FORT1=1TO250:NEXTT1
760 PRINT@331,"            "
770 FORT=1TO250:NEXTT
780 NEXTI
790 CLS:PRINT@228,"DO YOU WANT ANOTHER GO ?"
800 B$=INKEY$
810 IFB$=""THEN800
820 IFB$="Y"THEN350
830 CLS:PRINT@229,"THANK YOU FOR PLAYING.":GOTO3130
840 LF=0
850 FORI=1TOLA
860 B$=MID$(A$,I,1)
870 IFB$<>L$THEN1030
880 REM
890 REM LETTER IN WORD
900 REM
910 PRINT@423,"  LETTER IN WORD  "
920 N=N+1
930 FORTD=1TO1500:NEXT
940 IA=4+I
950 REM
960 REM PRINT LETTER IN POSITION
```

```
970 REM
980 PRINT@230+IN,L$;
990 II=I
1000 GOSUB2340
1010 LF=LF+1
1020 RETURN
1030 NEXTI
1040 IFLF=0THENGOSUB1090
1050 RETURN
1060 REM
1070 REM LETTER NOT IN WORD
1080 REM
1090 L=L+1
1100 PRINT@423,"LETTER NOT IN WORD"
1110 FORTD=1TO1500:NEXT
1120 ON L GOSUB1420,1510,1620,1720,1820
1130 IFL>5THEN ON L-5 GOSUB1900,2020,2180,2260
1140 IFL<10 THENRETURN
1150 REM
1160 REM DRAW FEET OF MAN
1170 REM
1180 K=19
1190 SCREEN1,1:COLOR8,5
1200 FORII=1TO1000:NEXT
1210 DRAW"BM144,144D6L6D6R12U12L6"
1220 DRAW"BM168,144D6R6D6L12U12R6"
1230 PAINT(144,154)
1240 PAINT(168,154)
1250 REM
1260 REM FAILED TO GET WORD
1270 REM
1280 FOR TD=1TO3000:NEXT
```

```
1290 SCREEN0,0:FORTD=1TO2000:NEXT
1300 PRINT@423," SORRY YOU LOST!! "
1310 FORI=1TO15
1320 PRINT@235,Z$
1330 FORT=1TO250:NEXT
1340 PRINT@235,STRING$(LEN(Z$)," ")
1350 FORT=1TO250:NEXT
1360 NEXT
1370 N=LA+1
1380 RETURN
1390 REM
1400 REM DRAW BASE
1410 REM
1420 K=20
1430 PCLS:PMODE3,1:SCREEN1,1:COLOR6,5
1440 FORI=1TO1000:NEXT
1450 DRAW"BM0,180;R252"
1460 FORTD=1TO3000:NEXT
1470 RETURN
1480 REM
1490 REM DRAW UPRIGHT
1500 REM
1510 K=6
1520 SCREEN1,1
1530 FORI=1TO1000:NEXT
1540 DRAW"BM60,180;U168"
1550 DRAW"BM70,180;U168;NL10"
1560 PAINT(65,173)
1570 FORTD=1TO3000:NEXT
1580 RETURN
1590 REM
1600 REM DRAW HORIZONTAL
```

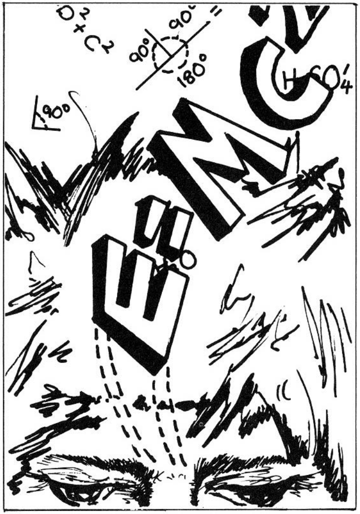
```
1610 REM
1620 K=6
1630 SCREEN1,1
1640 FORI=1TO1000:NEXT
1650 DRAW"BM70,12R90ND10;BM160,22L90"
1660 PAINT(130,17)
1670 FORTD=1TO3000:NEXT
1680 RETURN
1690 REM
1700 REM DRAW DIAGONAL SUPPORT
1710 REM
1720 K=10
1730 SCREEN1,1
1740 FORI=1TO1000:NEXT
1750 DRAW"BM70,52E33;BM70,60E42"
1760 PAINT(72,55)
1770 FORTD=1TO3000:NEXT
1780 RETURN
1790 REM
1800 REM DRAW ROPE
1810 REM
1820 K=6
1830 SCREEN1,1:FORI=1TO1000:NEXT
1840 DRAW"BM156,22D8;BM155,22D8"
1850 FORTD=1TO3000:NEXT
1860 RETURN
1870 REM
1880 REM DRAW HEAD
1890 REM
1900 K=7
1910 SCREEN1,1:COLOR8,5:FORI=1TO1000:NEXT
1920 CIRCLE(156,47),17
```

```
1930 CIRCLE(150,41),1
1940 CIRCLE(162,41),1
1950 DRAW"BM156,44D5"
 1960 CIRCLE(156,60),9,0,1,.35,.95
1970 FORTD=1TO3000:NEXT
1980 RETURN
1990 REM
2000 REM DRAW BODY
2010 REM
2020 K=11
2030 SCREEN1,1:COLOR7,5
2040 FORI=1TO1000:NEXT
2050 DRAW"BM140,64R32F8D30L7;BM140,64G8D30R7"
2060 DRAW"BM140,64D44R32U44
2070 DRAW"BM137,68R38;BM133,72R46;BM133,76R46;BM133,80R46;BM133,84R46"
2080 DRAW"BM133,88R46;BM133,92R46;BM133,96R46;BM133,100R46;BM140,104R32"
2090 PAINT(138,69):PAINT(156,69):PAINT(174,69):PAINT(134,78):PAINT(156,78):PAINT
(178,78)
2100 PAINT(134,86):PAINT(156,86):PAINT(178,86)
2110 PAINT(134,94):PAINT(156,94):PAINT(178,94)
2120 PAINT(134,101):PAINT(156,101):PAINT(178,101)
2130 FORTD=1TO3000:NEXT
2140 RETURN
2150 REM
2160 REM DRAW HANDS
2170 REM
2180 K=15
2190 SCREEN1,1:COLOR7,5:FORI=1TO1000:NEXT
2200 CIRCLE(136,106),4,0,1,.05,.95:CIRCLE(177,106),4,0,1,.55,.45
2210 FORTD=1TO3000:NEXT
2220 RETURN
2230 REM
```

```
2240 REM DRAW LEGS
2250 REM
2260 K=16
2270 SCREEN1,1:COLOR7,5
2280 FORI=1TO1000:NEXT
2290 DRAW"BM140,108D35R14U35"
2300 DRAW"BM158,108D35R14U35"
2310 PAINT(142,110):PAINT(160,110)
2320 FORTD=1TO3000:NEXT
2330 RETURN
2340 X=II-1:Y=LA-II
2350 IFX=0 AND Y=0 THEN2400
2360 IFX=0 THEN A$=" "+RIGHT$(A$,Y):RETURN
2370 IF Y=0 THEN A$=LEFT$(A$,X)+" ":RETURN
2380 A$=LEFT$(A$,X)+" "+RIGHT$(A$,Y)
2390 RETURN
2400 A$=""
2410 FORI=1TOLA
2420 A$=A$+" "
2430 NEXTI
2440 RETURN
2450 REM
2460 REM WORD DATA
2470 REM
2480 DATA ABSENCE,ACCEPT,ACCIDENTALLY
2490 DATA ACCOMMODATE,ACHIEVED,ACKNOWLEDGE
2500 DATA ACQUAINTED,ADDRESSES,AERIAL
2510 DATA AGGRAVATE,AGREEABLE,AMATEUR,AMONG
2520 DATA ANTARCTIC,ANXIETY,APPARENT,APPEARANCE
2530 DATA APPROPRIATE,ARCTIC,ARGUMENT,ARRANGEMENTS
2540 DATA ASCENT,ATHLETIC,AWFUL,BACHELOR
2550 DATA BEGINNING,BELIEVED,BENEFITED,BREATHE
```

```
2560 DATA BRITAIN,BUSINESS,CAPTAIN,CEILING
2570 DATA CEMETERY,CERTAIN,CHOICE,CLOTHES
2580 DATA COLLEGE,COMING,COMMITTEE,COMPARATIVE
2590 DATA COMPETENT,COMPLETELY,CONSCIENTIOUS
2600 DATA CONSCIOUS,CONSISTENT,CONVENIENCE
2610 DATA COPIES,COURSE,COURTEOUS,COURTESY
2620 DATA CRITICISM,DECEIVE,DECISION,DEFINITE
2630 DATA DESIRABLE,DESPERATE,DISAPPEARED
2640 DATA DISAPPOINTED,DISASTROUS,DISCIPLINE
2650 DATA DISSATISFIED,EFFICIENTLY,EIGHTH
2660 DATA ELIMINATED,EMBARRASSED,EMPHASIZE
2670 DATA ENTHUSIASM,EQUIPPED,ESPECIALLY
2680 DATA ESSENTIAL,EXAGGERATED,EXCELLENT
2690 DATA EXERCISE,EXHAUSTED,EXISTENCE
2700 DATA EXPENSE,EXPERIENCE,FAMILIAR
2710 DATA FEBRUARY,FINANCIAL,FOREIGN
2720 DATA FORMERLY,FORTY,FRIEND,GAUGE
2730 DATA GENIUS,GOVERNMENT,GRAMMAR
2740 DATA GRAMOPHONE,GRIEVANCE,GUARD
2750 DATA GUARDIAN,HANDKERCHIEF,HEIGHT
2760 DATA HEROES,HONORARY,HUMOROUS,HUNGRY
2770 DATA HURRIEDLY,HYPOCRISY,IMAGINATION
2780 DATA IMMEDIATELY,IMMIGRATE,INCIDENTALLY
2790 DATA INDIFFERENT,INDISPENSABLE
2800 DATA INFLUENTIAL,INTELLIGENCE
2810 DATA IRRESISTIBLE,KNOWLEDGE,LIGHTENING
2820 DATA LITERATURE,LIVELIHOOD,LOSE
2830 DATA LOSING,LYING,MAINTENANCE
2840 DATA MARRIAGE,MEANT,MEDICINE
2850 DATA MEDITERRANEAN,MINIATURE
2860 DATA MINUTES,MISCHIEVOUS,MURMUR
2870 DATA NECESSARY,NIECE,NOTICEABLE
```

```
2880 DATA OCCASIONAL,OCCURRED,OCCURRENCE
2890 DATA OMITTED,OPINION,OPPORTUNITY
2900 DATA OCCUPATION,PARALLEL,PARLIAMENT
2910 DATAPASTIME,PERMANENT,PERMISSIBLE
2920 DATA PERSEVERANCE,PHYSICAL,PLANNING
2930 DATA PLEASANT,POSSESSES,PRECEDING
2940 DATA PREFERENCE,PREJUDICE,PRIVILEGE
2950 DATA PROCEDURE,PROCEEDS,PROFESSIONAL
2960 DATA PROFESSOR,PRONUNCIATION
2970 DATA PROPRIETARY,PSYCHOLOGY,QUIET
2980 DATA REALLY,RECEIVED,RECOGNIZED
2990 DATA RECOMMENDED,PREFERRED,RELIEVED
3000 DATA REPETITION,RESTAURANT,RHYTHM
3010 DATA SCARCELY,SECRETARIES,SEIZE
3020 DATA SENTENCE,SEPARATE,SERGEANT
3030 DATA  SEVERELY,SHINING,SIEGE,SIMILAR
3040 DATA SINCERELY,SPEECH,STRENGTH
3050 DATA SUCCESSFUL,SUPERSEDE
3060 DATA SUPPRESSION,SURPRISING,SYNONYM
3070 DATA TENDENCY,TRAGEDY,TRANSFERRED
3080 DATA TWELFTH,UNCONSCIOUS
3090 DATA UNNECESSARY,UNTIL,USUALLY
3100 DATA VALUABLE,VIEW,WEDNESDAY
3110 DATA WOOLLEN
3120 DATA END
3130 FORTD=1TO2000:NEXT:CLS
```



SUPERMIND

DESCRIPTION

This game is modelled on the popular board game of a similar title. The program allows the user to guess at its random combination of colours and will reply with a white marker if the colour is in the combination but not in the right position and with a black marker if it is in the correct position. The keys 1 to 6 are used to choose the colour, the keys 7 to 0 to choose the row and the ENTER key to input the guess. It has a nice use of colours and is easy to follow.

RUNNING THE PROGRAM

After typing RUN, three pages of instructions will appear separated by 'press any key'. When the instructions have been read, pressing any key starts the game. Using the keys 7 to 0, the row on which the colour is to be placed may be set. The computer will not accept any colours until the ENTER key has been pressed, and only then, provided that there are colours in all four positions and that no colour is used more than once in that column. If the correct combination has been guessed or all nine guesses are used up, the computer will display its secret. If any key is then pressed the computer will ask if another go is required. Press 'X' if you give in before all guesses are spent.

PROGRAM STRUCTURE

The lines of interest in this program are as follows:

40-100 Input for instructions.
140-340 Display the board on the screen.
390-570 Choose four random colours all different.
620-690 Input and check if key pressed is acceptable.
730-840 Display pointers so that all of them are to the right, except the one on the row chosen.
880-930 Put the chosen colour in the chosen position.
980-990 Failed to guess the combination.
1000-1060 Check to see if four colours are used and that all are different.
1100-1140 Check for colours in the correct position.
1180-1270 Check for correct colours in the wrong position.
1310-1420 Display the results of the last two routines above the corresponding column.
1460-1490 Display the correct colour combination.
1550-1600 Input for another go.
1640-2060 Print instructions.

```
 10 REM SUPERMIND
 20 REM **************************
 30 REM
 40 CLS
 50 PRINT@230,"INSTRUCTIONS(Y/N)?"
 60 I$=INKEY$
 70 IFI$=""THEN60
 80 IFI$="Y"THEN1610
 90 IFI$="N"THEN140
 100 GOTO60
 110 REM
 120 REM SET UP DISPLAY
 130 REM
 140 ZZ=0
 150 L=0
 160 CLS
 170 PRINT@230,"+-+-+-+-+-+-+-+-+-+""
 180 FORM=1TO3
 190 PRINTTAB(6)"!";
 200 FORW=1TO9
 210 PRINT"@!";
 220 NEXT
 230 PRINT"  "
 240 PRINTTAB(6)"+-+-+-+-+-+-+-+-+-+"
 250 NEXT
 260 PRINT@454,"!";
 270 FORW=1TO9
 280 PRINT"@!";
 290 NEXT
 300 PRINT@486,"+-+-+-+-+-+-+-+-+-+";
 310 FORA=4TO6
 320 PRINT@26+64*A,CHR$(95);"  ";A+3
```

```
330 NEXT
340 PRINT@474,CHR$(95);" ";0
350 REM
360 REM CHOSE RANDOM COLOURS
370 REM ALL DIFFERENT
380 REM
390 IFRND(10)<6THEN420
400 X(1)=RND(3)
410 GOTO430
420 X(1)=RND(3)+4
430 IFRND(10)<6THEN460
440 X(2)=RND(3)
450 GOTO470
460 X(2)=RND(3)+4
470 IFX(2)=X(1) THEN430
480 IFRND(10)<6THEN510
490 X(3)=RND(3)
500 GOTO520
510 X(3)=RND(3)+4
520 IFX(3)=X(1) OR X(3)=X(2) THEN480
530 IFRND(10)<6THEN560
540 X(4)=RND(3)
550 GOTO570
560 X(4)=RND(3)+4
570 IFX(4)=X(1) OR X(4)=X(2) OR X(4)=X(3) THEN530
580 FOR Z=4TO7
590 PRINT@3+64*Z,CHR$(63);
600 NEXT
610 I=2:GG=10-22
620 Q$=INKEY$
630 IFQ$=""THEN620
640 IFASC(Q$)=13THEN940
```

```
650 IFASC(Q$)=88THEN980
660 IFASC(Q$)>54 AND ASC(Q$)<58 THEN730
670 IF Q$="0" THEN730
680 IFASC(Q$)>48 AND ASC(Q$)<55 THEN880
690 GOTO620
700 REM
710 REM PUSH POINTERS OFF ROW
720 REM
730 FOR I=4TO7
740 PRINT@25+64*I," ";
750 PRINT@26+64*I,CHR$(95);
760 NEXT
770 REM
780 REM PUSH POINTER TO ROW
790 REM
800 L=(VAL(Q$)-3)*64+25
810 IFL=-167 THENL=473
820 PRINT@L+1," ";
830 PRINT@L,CHR$(95);
840 GOTO610
850 REM
860 REM PUT COLOUR IN POSITION
870 REM
880 F=VAL(Q$)
890 IFF>3THENF=F+1
900 IF L=0 THENL=281
910 PRINT@L-2-ZZ,CHR$(143+16*F);
920 D((L-281)/64+1)=F
930 GOTO610
940 IFZZ<>16THEN1000
950 REM
960 REM FILLED ALL 9 COLUMNS
```

```
970 REM
980 PRINT@485,"YOU HAVE NOT SUCCEEDED";
990 GOTO1460
1000 ZZ=ZZ+2
1010 FORW=1TO4
1020 IFD(W)=0 THEN ZZ=ZZ-2:GOTO610
1030 NEXT
1040 IF D(1)=D(2) OR D(1)=D(3) OR D(1)=D(4) THEN ZZ=ZZ-2:GOTO610
1050 IF D(2)=D(3) OR D(2)=D(4) THEN ZZ=ZZ-2:GOTO610
1060 IFD(3)=D(4) THENZZ=ZZ-2:GOTO610
1070 REM
1080 REM CHECK FOR BLACK REPLY
1090 REM
1100 Z=0
1110 V=0
1120 FORT=1TO4
1130 IFX(T)=D(T) THEN Z=Z+1
1140 NEXT
1150 REM
1160 REM CHECK FOR WHITE REPLY
1170 REM
1180 FORE=1TO4
1190 FORH=1TO4
1200 IFX(E)=D(H) THENV=V+1
1210 NEXT
1220 NEXT
1230 V=V-Z
1240 FORW=1TO4
1250 D(W)=0
1260 NEXT
1270 IF Z=0THEN1390
1280 REM
```

```
1290 REM DISPLAY BLACK REPLY
1300 REM
1310 FORW=1TOZ
1320 PRINT@(8-W*2)*32+13+GG,CHR$(128);
1330 NEXT
1340 IFZ=4THEN1460
1350 IFV=0THEN610
1360 REM
1370 REM DISPLAY WHITE REPLY
1380 REM
1390 FORW=1TOV
1400 PRINT@(8-(W+Z)*2)*32+13+GG,CHR$(207);
1410 NEXT
1420 GOTO610
1430 REM
1440 REM DISPLAY STORED COLOURS
1450 REM
1460 G=0
1470 FORZ=1TO4
1480 PRINT@195+64*Z,CHR$(143+16*X(Z));
1490 NEXT
1500 IF INKEY$="" THEN1500
1510 CLS
1520 REM
1530 REM INPUT FOR ANOTHER GO
1540 REM
1550 PRINT@231,"ANOTHER GO ?(Y/N)"
1560 A$=INKEY$
1570 IFA$=""THEN1560
1580 IFA$="Y"THEN140
1590 IFA$="N"THENGOTO2070
1600 GOTO1560
```

```
1610 REM
1620 REM INSTRUCTONS
1630 REM
1640 CLS:PRINT@38,"THIS IS THE GAME"
1650 PRINT@106,"supermind"
1660 PRINT@192,"IN THIS GAME YOU HAVE ";
1670 PRINT"TO GUESS  THE COLOUR-";
1680 PRINT"COMBINATION DEVISED"
1690 PRINT"BY THE COMPUTER. THE";
1700 PRINT" 4 COLOURS  ARE HIDDEN ";
1710 PRINT"BY THE COMPUTER AND ";
1720 PRINT" IT WILL TELL YOU ",
1730 PRINT"WHETHER YOU     HAVE ";
1740 PRINT"GUESSED CORRECTLY OR";
1750 PRINT" NOT."
1760 GOSUB2040
1770 CLS:PRINT@32,"A WHITE MARKER TELLS ";
1780 PRINT"YOU THAT A COLOUR IS ";
1790 PRINT"IN THE COMBINATION BUT";
1800 PRINT"NOT IN THE RIGHT PLACE ",
1810 PRINT"AND A    BLACK MARKER ";
1820 PRINT"TELLS YOU THAT A"
1830 PRINT"COLOUR THAT YOU HAVE ";
1840 PRINT"GUESSED IS IN THE ",
1850 PRINT"COMBINATION AND IN THE"
1860 PRINT"RIGHT PLACE."
1870 PRINT:PRINT"CHOOSE THE ROWS BY PRESSING ";
1880 PRINT"KEYS7,8,9 OR 0 ",
1890 PRINT"RESPECTIVELY."
1900 GOSUB2040
1910 CLS:PRINT@32,"THE COLOURS THAT YOU ";
1920 PRINT"MAY CHOOSE ARE 1-6. ";
```

```
1930 PRINT"THE COLOURS ARE AS ";
1940 PRINT"    FOLLOWS: 1- YELLOW"
1950 PRINTTAB(9)"2- BLUE"
1960 PRINTTAB(9)"3- RED"
1970 PRINTTAB(9)"4- CYAN"
1980 PRINTTAB(9)"5- MAGENTA"
1990 PRINTTAB(9)"6- ORANGE"
2000 PRINT:PRINT"HIT 'X' IF YOU";
2010 PRINT" GIVE IN BEFORE   TIME."
2020 GOSUB2040
2030 GOTO140
2040 PRINT@458,"HIT ANY KEY"
2050 IFINKEY$=""THEN2050
2060 RETURN
2070 CLS
```

MOONLANDER

DESCRIPTION

The aim of the game is to land a space module on the surface of the moon without crashing. It requires a great deal of skill to place the ship on the surface, and caution is advisable when judging the speed of your approach.

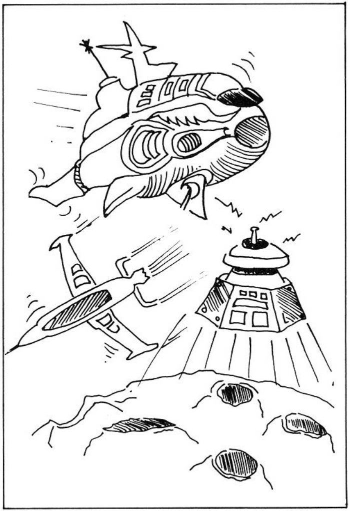
The display on the screen for this game is very simple, consisting of a lunar module on the right hand side of the screen with an information panel on the left, indicating the height, speed, time and fuel usage.

Two hi-res screens come into use once the module has reached the surface. The first one displays a drawing of the moon with the ship approaching and the other shows a moonscape with the module descending to its fate.

One final note for those interested, the game could be modified to use a joystick instead of the keyboard.

RUNNING THE PROGRAM

After typing RUN the program displays a set of instructions for playing the game. Pressing the space bar will start the game. There is an initial countdown of five seconds before the game starts. The speed of the module's descent can be controlled by burning fuel in the retro-rocket, the amount of fuel is controlled by the keys 1-9. Key 1 uses very little fuel and therefore slows the descent only slightly.



Note that the speed shown on the gauge will increase for all fuel usages below 5 since the ship has to counteract the effects of gravity. If the player uses all the fuel he will crash, but if he keeps his speed less than 15 m.p.h. a successful landing will be achieved.

PROGRAM STRUCTURE

The lines of most interest in this game are as follows:

130-180 Sets up landing parameters for initial fuel, speed and height information.
220-240 Inputs fuel usage from keyboard.
280-380 Calculates new values of speed and height.
430-480 Displays above values.
520-660 Sets up space module on the screen.
710-730 Prints display above landing parameters.
770-810 Displays burn out combustion lines.
850-930 Sets random dots to represent stars.
940-970 Prints panel for landing parameters.
1010-1110 Draws moon on first hi-res screen.
1140-1210 Draws and moves approaching ship on hi-res screen.
1260-1370 Draws moonscape on second hi-res screen.
1410-1510 Draws and lands module.
1550-1750 Displays crashed ship.
1790-1810 Inputs new go.
1850-1900 Displays five second countdown at the beginning of the game.
1940-2070 Prints instructions for the game.
2110-2190 Sounds explosion.

```
10 REM MOONLANDER
20 REM ***************************
30 REM
40 PCLEAR8
50 CLS:GOSUB1010:GOSUB1260:GOSUB1930
60 TIMER=0
70 CLS0
80 I4=I5
90 GOSUB850
100 REM
110 REM SET UP PARAMETERS
120 REM
130 TH=0:T=0
140 X0=52800
150 V0=-176:F=1:S=1
160 GOSUB520
170 GOSUB1840
180 AT=TIMER:AG=TIMER
190 REM
200 REM INPUT FUEL USAGE
210 REM
220 RR$=INKEY$:IFRR$<>"" THEN R=VAL(RR$)
230 IF (TIMER-AT)>25 THENAT=TIMER:GOTO280
240 GOTO230
250 REM
260 REM CALCULATE NEW VALUES
270 REM
280 T=T+R
290 TH=TH+1
300 A=5.4*(1-0.2*R)
310 X=V0+0.5*A:X0=X0+X
320 IFX0>0 THEN340
```

```
330 X=X-X0:F=0:X0=0
340 D=V0*V0-2*X*A
350 IFD>0 THEN370
360 D=-D:S=-1
370 V0=SQR(D)*S*SGN(X)
380 S=1
390 REM
400 REM DISPLAY LANDING
410 REM PARAMETERS
420 REM
430 PRINT@65,MID$(STR$(V0),2,5);
440 PRINTLEFT$(STR$(X0)+A$(0),7);
450 PRINT@161,2500-T;TAB(8)TH;
460 GOSUB770:IF F=0 THEN PRINT@71,"  0      ";:FORTD=1TO1500:NEXT:GOSUB1140:GOTO14
10
470 IFT>2500 THEN R=0:GOTO290
480 GOTO220
490 REM
500 REM SET SHIP
510 REM
520 SET(48,0,6):SET(49,0,6)
530 FORX=47TO50:SET(X,1,6):NEXT
540 FORX=46TO51:SET(X,2,6):NEXT
550 FORY=3TO5:FORX=45TO52
560 SET(X,Y,6)
570 NEXTX:NEXTY
580 FORX=44TO53:SET(X,6,6):NEXT
590 FORX=43TO54:SET(X,7,6):NEXT
600 FORX=42TO55:SET(X,8,6):NEXT
610 FORX=47TO50:SET(X,9,6):NEXT
620 FORX=46TO51:SET(X,10,6):NEXT
630 FORX=45TO52:SET(X,11,6):NEXT
```

```
640 RESET(48,10):RESET(48,11):RESET(49,10):RESET(49,11):RESET(47,11):RESET(50,11
)
650 RESET(48,4)
660 PRINT@119,"USA";
670 REM
680 REM PRINT DISPLAY ABOVE
690 REM PARAMETERS
700 REM
710 PRINT@33,"speed height";
720 PRINT@129," fuel  time";
730 RETURN
740 REM
750 REM IF KEY GREATER THAN 0
760 REM PRESSED, DISPLAY 'BURN'
770 IF R=0 THEN RETURN
780 FOR I=1 TO R*2 STEP2:SET(48,I+11,4):SET(49,I+11,4)
790 NEXT
800 FORI=1TO18 STEP2
810 RESET(48,I+11):RESET(49,I+11):NEXT:RETURN
820 REM
830 REM SET 14 RANDOM '.' ON
840 REM SCREEN
850 CLS0
860 FORP=1TO13 STEP2
870 IF RND(10)>4 THEN SET(RND(13)+27,P,5):GOTO890
880 SET(RND(7)+56,P,5)
890 NEXT
900 FORP=15TO28 STEP2
910 IFRND(10)>4 THEN SET(RND(47)-1,P,5):GOTO930
920 SET(RND(13)+56,P,5)
930 NEXT
940 FORP=0TO6
```

```
950 PRINT@32*P,STRING$(14," ");
960 NEXT
970 RETURN
980 REM
990 REM DRAW MOON IN HI-RES
1000 REM
1010 PMODE4,1:PCLS
1020 FORY=0TO199
1030 PSET(RND(255)-1,Y,5)
1040 NEXTY
1050 CIRCLE(127,96),70,5
1060 PAINT(127,96),5
1070 FORY=26TO166 STEP20
1080 FORX=57TO197 STEP20
1090 IFRND(10)>5THEN CIRCLE(X,Y),RND(15),0
1100 NEXTX:NEXTY:RETURN
1110 REM
1120 REM DRAW AND MOVE SHIP
1130 REM
1140 DIMR(8,10):PMODE4,1:SCREEN1,1:FORTD=1TO2000:NEXT
1150 DRAW"BM253,1D2G2D2G1H1G1H2E1H1E1R2E2R2"
1160 PAINT(251,3)
1170 GET(246,0)-(254,9),R,G
1180 FORX=1TO56
1190 PUT(246-X,X)-(254-X,9+X),R,PSET:NEXT
1200 LINE(190,56)-(198,65),PRESET,BF
1210 RETURN
1220 REM
1230 REM DRAW MOONSCAPE
1240 REM IN HI-RES
1250 REM
1260 PMODE3,5:PCLS:COLOR8,5
```

```
1270 DRAW"BM0,100R2F3R2F2R2F6D5F10D3R3D3R3D3R3D3R3D4F5R2D2R2D2F10E3F3E3F3E3F3E3F
32"
1280 DRAW"BM0,190R254"
1290 FORY=131TO189
1300 PSET(RND(255)-1,Y,8):NEXT
1310 DRAW"BM30,130R224"
1320 PAINT(2,101)
1330 CIRCLE(180,170),20,8,.33
1340 CIRCLE(150,140),10,8,.33
1350 CIRCLE(80,160),30,8,.33
1360 CIRCLE(230,150),25,8,.33
1370 RETURN
1380 REM
1390 REM DRAW AND LAND SHIP
1400 REM
1410 DIMS(30,38):PMODE3,5:SCREEN1,1:IFV0<-15THEN1550
1420 COLOR6,5:DRAW"BM126,1R6F5D11F5D5L10F8L8U4L8D4L8E8L9U5E5U10E5"
1430 DRAW"BM122,8R5D5L5U5"
1440 PAINT(127,2)
1450 GET(115,0)-(145,37),S,G
1460 FORY=1TO100
1470 PUT(115,Y)-(145,37+Y),S,PSET:NEXT
1480 COLOR8,5:DRAW"BM119,130L4,BM142,130L4"
1490 FORTD=1TO2000:NEXT
1500 -1760
1510 FORTD=1TO1500:NEXT:GOTO1790
1520 REM
1530 REM DISPLAY CRASHED SHIP
1540 REM
1550 COLOR6,5
1560 LINE(115,0)-(115,135),PSET
1570 LINE(145,0)-(145,135),PSET
1580 LINE(126,0)-(126,135),PSET
```

```
1590 LINE(132,0)-(132,135),PSET
1600 FORTD=1TO100:NEXT
1610 COLOR8,5
1620 DRAW"BM132,135E40;BM136,135U30;BM126,135H40;BM122,135U30;BM145,135E50;BM149
,135U40;BM115,135H50;BM111,135U40"
1630 PSET(70,100,8):PSET(80,105,6):PSET(56,80,8):PSET(90,90,6):PSET(65,75,6):PSE
T(75,80,8)
1640 PSET(190,95,6):PSET(200,105,8):PSET(170,85,8):PSET(200,75,6):PSET(180,90,8)
:PSET(172,100,6)
1650 CIRCLE(130,130),30,8,.33,0,.5
1660 GOSUB2110
1670 FORTD=1TO500:NEXT
1680 PMODE4,1:SCREEN1,1
1690 FORD=1TO1000:NEXT
1700 DRAW"BM189,63U2R2U2R2U2R2U2R2"
1710 DRAW"BM189,63E2H2E2H2E2H2E2H2"
1720 DRAW"BM189,63F2E2F2E2F2E2F2E2"
1730 FORTD=1TO500:NEXT
1740 LINE(189,48)-(204,65),PRESET,BF
1750 FORTD=1TO500:NEXT
1760 REM
1770 REM INPUT NEW GO
1780 REM
1790 CLS:PRINT@230,"";:INPUT"ANOTHER GO ";G$
1800 IFLEFT$(G$,1)="Y" THENCLS:GOSUB1260:RUN90
1810 GOTO2200
1820 REM
1830 REM COUNTDOWN TO START
1840 REM
1850 A=TIMER
1860 PRINT@480,"START OVER";5-INT((TIMER-A)/50);
1870 PRINT"SEC.";
```

```
1880 IF(TIMER-A)<250THEN1880
1890 PRINT@480,STRING$(17,128):R=0
1900 RETURN
1910 REM
1920 REM INSTRUCTIONS
1930 REM
1940 PRINT@4,"***** MOONLANDER *****"
1950 PRINT"YOU MUST LAND YOUR SPACE SHIP ON";
1960 PRINT"THE MOON."
1970 PRINT:PRINT"YOU HAVE CONTROL OVER THE SHIP"
1980 PRINT"BY CHANGING THE FUEL USAGE."
1990 PRINT:PRINT"YOU CHANGE THE FUEL USAGE WITH "
2000 PRINT"KEYS 0...9  -5 KEEPS CONSTANT"
2010 PRINT"SPEED."
2020 PRINT:PRINT"YOU MUST LAND WITH A SPEED OF"
2030 PRINT"LESS THAN 15 M.P.H."
2040 PRINT:PRINT"GOOD LUCK!"
2050 PRINTTAB(4)"HIT SPACE BAR TO START.";
2060 IF INKEY$<>" "THEN2060
2070 RETURN
2080 REM
2090 REM EXPLOSION
2100 REM
2110 FORX=1TO5
2120 PLAY"O1;T155;L255;V10;G"
2130 NEXT
2140 PLAY"L255;V13;F;L255;V16;E;L255;V22;D;L255;V27;C;V30;B;L255;A"
2150 FORX=1TO5
2160 PLAY"T155;L255;P255;V31;G"
```

```
2170 NEXT
2180 PLAY"T120;L255;P255;V30;A;T90;L255;P255;V27;B;T50;L255;P255;V22;C;T30;L255;
P255;V18;D"
2190 RETURN
2200 CLS
```

BOSS PUZZLE

DESCRIPTION

This game is a test of mental ability. The aim of the game is to rearrange numbers from 0 to 15 and one space into the following order:

1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	

The numbers are randomly placed in the grid to start with, and by moving into and out of spaces they can be rearranged. While the game is being played, a clock ticks away on the top right of the screen and a counter to count the number of moves taken is displayed below the clock. A colourful display and quite enjoyable to play .

RUNNING THE PROGRAM

After typing RUN, the program displays a screen of instructions. After hitting the ENTER key, the grid, with numbers in order, will appear and start to shuffle. When the shuffling has been completed, the clock starts and the game begins. Use the ARROW keys to move. The time and amount of moves taken will be displayed when all the rectangles are in their correct positions, including the blank space.

PROGRAM STRUCTURE

The lines of interest in this game are as follows:

130-230 Subroutine to display rectangles at co-ordinates (P, Q).
280-400 Print instructions.
440-520 Set up required variables.
560-570 Print title and clear screen.
610-690 Fill in the rectangles.
730-890 Shuffle the rectangles with random moves.
930-980 Start the game.
1020-1100 Check to see if the rectangles are in order.
1140-1530 Accept and perform move input using arrow keys.
1570-1630 Give nasty display.
1670-1820 Give result and input for another go

```
10 REM BOSS PUZZLE
20 REM ***************************
30 REM
40 GOTO270
50 REM
60 REM PAUSE
70 REM
80 FOR TD=1TO3000:NEXT
90 RETURN
100 REM
110 REM DRAW SQUARE @ COORDINATES
120 REM P,Q
130 Y=10+3*P
140 Z=2+3*Q
150 N$=S$(P,Q)
160 N=VAL(N$)
170 IF N=0 THENN=16
180 X=VAL(MID$(C$,N,1))
190 PRINT@Z*32+Y,STRING$(3,143+16*X);
200 PRINT@(Z+1)*32+Y,STRING$(3,143+16*X);
210 PRINT@(Z+1)*32+Y+1,N$;
220 PRINT@(Z+2)*32+Y,STRING$(3,143+16*X);
230 RETURN
240 REM
250 REM INSTRUCTIONS
260 REM
270 CLS
280 PRINT"BOSS PUZZLE"
290 PRINT@75,"YOU WILL BE SHOWN A 4";
300 PRINT"BY 4 MATRIX OF SQUARES";
310 PRINT" NUMBERED 1 TO 15,WITH";
320 PRINT" ONE SQUARE MISSING."
```

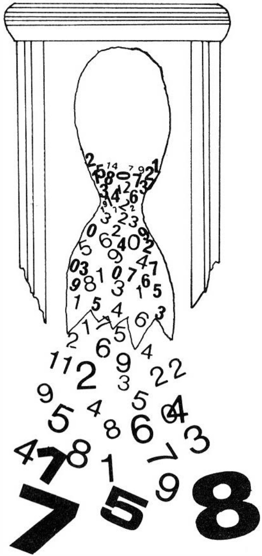
```
330 PRINT:PRINT"THE COMPUTER WILL THEN ";
340 PRINT"SHUFFLE   THE SQUARES RANDOMLY."
350 PRINT:PRINT"THE OBJECT IS TO SLIDE";
360 PRINT" THE        SQUARES BACK INTO ";
370 PRINT"THEIR ORIGI-  NAL ORDER";
380 PRINT" USING ARROW KEYS."
390 PRINT:PRINT:PRINTTAB(5)"PRESS enter TO START";
400 IFINKEY$<>CHR$(13) THEN400
410 REM
420 REM SET UP VARIABLES ETC
430 REM
440 DIMS$(4,4)
450 C$="7216537216537214"
460 L=0
470 F$(1)=CHR$(94)
480 F$(2)=CHR$(9)
490 F$(3)=CHR$(8)
500 F$(4)=CHR$(10)
510 D$="1234567891012345678920123"
520 R$="123456789101224567892012"
530 REM
540 REM PRINT TITLE & SCREEN
550 REM
560 CLS5
570 PRINT@14,"BOSS";
580 REM
590 REM FILL IN SQUARES
600 REM
610 N=0
620 FORQ=0TO3:FORP=0TO3
630 N=N+1
640 N$=STR$(N):N$=MID$(N$,2,LEN(N$)-1)
```

```
650 IFVAL(N$)=16 THENN$=""
660 S$(P,Q)=N$
670 GOSUB120
680 NEXT:NEXT
690 GOSUB80
700 REM
710 REM SHUFFLE
720 REM
730 FORI=1TO40
740 P1=RND(4)-1
750 P2=RND(4)-1
760 Q1=RND(4)-1
770 Q2=RND(4)-1
780 IF P1=P2 AND Q1=Q2 OR S$(P1,Q1)="" OR S$(P2,Q2)="" THEN740
790 T$=S$(P1,Q1)
800 S$(P1,Q1)=S$(P2,Q2)
810 S$(P2,Q2)=T$
820 P=P1
830 Q=Q1
840 GOSUB120
850 P=P2
860 Q=Q2
870 GOSUB120
880 NEXT
890 GOSUB80
900 REM
910 REM START PLAY
920 REM
930 P=3
940 Q=3
950 TIMER=0
960 H=0
```

```
970 M=0
980 PRINT@193,"USE";:PRINT@225,"ARROW";:PRINT@257,"KEYS TO";:PRINT@289,"MOVE";
990 REM
1000 REM CHECK IF IN ORDER
1010 REM
1020 N=0
1030 FL=-1
1040 FORX=0TO3
1050 FORY=0TO3
1060 N=N+1
1070 IFN=16 AND VAL(S$(Y,X))=0 THEN1090
1080 IFVAL(S$(Y,X))<>N THENFL=0
1090 NEXT:NEXT
1100 IF FL THEN1570
1110 REM
1120 REM ACCEPT & PERFORM MOVE
1130 REM
1140 A$=INKEY$
1150 IFA$<>""THEN1370
1160 PRINT@121,"TIME";
1170 PRINT@281,"MOVE";
1180 IFTIMER>=60000 THENTIMER=0:M=M+20
1190 IFM=60THENM=0:H=H+1
1200 T=TIMER
1210 HH=INT(T/180000)+H
1220 H$=STR$(HH):H$=MID$(H$,2,LEN(H$)-1)
1230 IFVAL(H$)<10THENH$="0"+H$
1240 PRINT@183,H$;" ";
1250 T=T-INT(T/180000)*180000
1260 MM=INT(T/3000)+M
1270 M$=STR$(MM):M$=MID$(M$,2,LEN(M$)-1)
1280 IFVAL(M$)<10THENM$="0"+M$
```

```
1290 PRINTM$;"  ";
1300 T=T-INT(T/3000)*3000
1310 SS=INT(T/50)
1320 S$=STR$(SS):S$=MID$(S$,2,LEN(S$)-1)
1330 IFVAL(S$)<10THENS$="0"+S$
1340 PRINTS$;
1350 L$=STR$(L):L$=MID$(L$,2,LEN(L$)-1)
1360 PRINT@346,L$;:GOTO1140
1370 IF NOT(A$=F$(1) AND Q<>3) THEN1400
1380 S$(P,Q)=S$(P,Q+1):GOSUB120
1390 Q=Q+1:GOTO1500
1400 IF NOT (A$=F$(4) AND Q<>0) THEN1430
1410 S$(P,Q)=S$(P,Q-1):GOSUB120
1420 Q=Q-1:GOTO1500
1430 IF NOT(A$=F$(3) AND P<>3) THEN1460
1440 S$(P,Q)=S$(P+1,Q):GOSUB120
1450 P=P+1:GOTO1500
1460 IF NOT(A$=F$(2) AND P<>0) THEN1490
1470 S$(P,Q)=S$(P-1,Q):GOSUB120
1480 P=P-1:GOTO1500
1490 GOTO1140
1500 S$(P,Q)=""
1510 GOSUB120
1520 L=L+1
1530 GOTO1020
1540 REM
1550 REM GIVE NASTY DISPLAY
1560 REM
1570 T$=H$+M$+S$:GOSUB80
1580 FORX=1TO100
1590 IFRND(10)<6 THENPRINT@RND(31)+32*(RND(14)),"DONE IT!";:GOTO1610
1600 PRINT@RND(31)+32*(RND(14)),"done it!";
```

```
1610 SOUND RND(150)+100,RND(3)
1620 NEXT
1630 GOSUB80
1640 REM
1650 REM GIVE RESULT
1660 REM
1670 CLS:PRINT@67,"IT TOOK YOU ";L$;" MOVES"
1680 PRINT:PRINTTAB(2)"AND ";
1690 IFVAL(H$)=0THEN1720
1700 IFLEFT$(H$,1)="0"THENH$=RIGHT$(H$,1)
1710 PRINTH$;" HOUR(S)";
1720 IFLEFT$(M$,1)="0" THEN M$=RIGHT$(M$,1)
1730 PRINT@162,M$;" MINUTES";
1740 IFLEFT$(S$,1)="0" THEN S$=RIGHT$(S$,1)
1750 PRINT@194,S$;" SECONDS";
1760 PRINT:PRINT:PRINTTAB(10)"ANOTHER GAME?"
1770 A$=INKEY$
1780 IFA$=""THEN1770
1790 IFA$="N"THEN1820
1800 IFA$<>"Y"THEN1770
1810 RUN440
1820 CLS
```



EGG TIMER

DESCRIPTION

A digital egg timer showing minutes and seconds in large characters displayed on the screen. The timer sounds an alarm once the preset time has been reached (a maximum of ten minutes). This program can be modified to act as a clock or a stopwatch.

RUNNING THE PROGRAM

The program starts with a prompt to set the required timing. This is input in the format MMSS, M is the minutes and S is the seconds. To stop the alarm hit either the space bar which ends the program, or the ENTER key which resets the timer.

PROGRAM STRUCTURE

The lines of interest in this program are as follows:

130-490 String arrays holding large number characters.
530-640 Inputs time and checks to see that the maximum of ten minutes is not exceeded.
690-760 Converts time in seconds into separate two figure minutes and seconds, and stores them in the string variable T$.
800-970 Prints timer display.
1010-1090 Sounds the alarm.

```
10 REM **************************
20 REM EGG TIMER
30 REM
40 CLS
50 DIMA$(9,4)
60 G=0
70 REM
80 REM CHARACTER ARRAYS
90 REM
100 REM
110 REM ZERO
120 REM
130 A$(0,0)=CHR$(128)+CHR$(128)+CHR$(128):A$(0,1)=CHR$(120)+CHR$(143)+CHR$(128):
A$(0,2)=A$(0,1):A$(0,3)=A$(0,2):A$(0,4)=A$(0,0)
140 REM
150 REM ONE
160 REM
170 A$(1,0)=CHR$(143)+CHR$(128)+CHR$(143):A$(1,1)=A$(1,0):A$(1,2)=A$(1,1):A$(1,3
)=A$(1,2):A$(1,4)=A$(1,3)
180 REM
190 REM TWO
200 REM
210 A$(2,0)=A$(0,0):A$(2,1)=CHR$(143)+CHR$(143)+CHR$(128):A$(2,2)=A$(2,0):A$(2,3
)=CHR$(128)+CHR$(143)+CHR$(143):A$(2,4)=A$(0,0)
220 REM
230 REM THREE
240 REM
250 A$(3,0)=A$(0,0):A$(3,1)=A$(2,1):A$(3,2)=A$(0,0):A$(3,3)=A$(2,1):A$(3,4)=A$(0
,0)
260 REM
270 REM FOUR
280 REM
```

```
290 A$(4,0)=A$(0,1):A$(4,1)=A$(0,1):A$(4,2)=A$(0,0):A$(4,3)=A$(2,1):A$(4,4)=A$(2
,1)
300 REM
310 REM FIVE
320 REM
330 A$(5,0)=A$(0,0):A$(5,1)=A$(2,3):A$(5,2)=A$(0,0):A$(5,3)=A$(2,1):A$(5,4)=A$(0
,0)
340 REM
350 REM SIX
360 REM
370 A$(6,0)=A$(0,0):A$(6,1)=A$(2,3):A$(6,2)=A$(0,0):A$(6,3)=A$(0,1):A$(6,4)=A$(0
,0)
380 REM
390 REM SEVEN
400 REM
410 A$(7,0)=A$(0,0):A$(7,1)=A$(2,1):A$(7,2)=A$(7,1):A$(7,3)=A$(7,2):A$(7,4)=A$(7
,3)
420 REM
430 REM EIGHT
440 REM
450 A$(8,0)=A$(0,0):A$(8,1)=A$(0,1):A$(8,2)=A$(8,0):A$(8,3)=A$(8,1):A$(8,4)=A$(8
,2)
460 REM
470 REM NINE
480 REM
490 A$(9,0)=A$(0,0):A$(9,1)=A$(0,1):A$(9,2)=A$(9,0):A$(9,3)=A$(2,1):A$(9,4)=A$(0
,0)
500 REM
510 REM INPUT TIME
520 REM
530 PRINT@187,"MMSS"
540 PRINT@205,"TIME"
```

```
550 PRINT@213,"(E.G. 0711)";
560 PRINT@283,"MMSS"
570 INPUT"(MAXIMUM TIME=10 MINUTES)";B$
580 CLS
590 C$=LEFT$(B$,2)
600 M=VAL(C$)*3000
610 D$=RIGHT$(B$,2)
620 S=VAL(D$)*50
630 T=M+S
640 IF T>30000 OR T=0 THEN540
650 REM
660 REM SET MINUTES & SECONDS
670 REM
680 TIMER=0
690 IFTIMER>=T THEN T$=B$:G=1:GOTO820
700 M$=STR$(INT(TIMER/3000))
710 M$=RIGHT$(M$,2)
720 IFVAL(M$)<10 THEN M$=RIGHT$(M$,1):M$="0"+M$
730 S$=STR$(INT((TIMER-VAL(M$)*3000)/50))
740 S$=RIGHT$(S$,2)
750 IF VAL(S$)<10 THEN S$=RIGHT$(S$,1):S$="0"+S$
760 T$=M$+S$
770 REM
780 REM PRINT EGG TIMER
790 REM
800 PRINT@207," ";
810 PRINT@271," ";
820 IF RIGHT$(T$,3)="000" THENZ=1:GOTO860
830 IFRIGHT$(T$,2)="00" THENZ=2:GOTO860
840 IFRIGHT$(T$,1)="0" THENZ=3:GOTO860
850 Z=4
860 FORI=4TOZ STEP-1
```

```
870 C(I)=VAL(MID$(T$,I,1))
880 IFI<3THENE=0:GOTO900
890 E=2
900 FORJ=0TO4
910 PRINT@167+4*(I-1)+32*J+E,A$(C(I),J);
920 NEXT
930 NEXT
940 IFG=1THEN1010
950 PRINT@207,CHR$(128);
960 PRINT@271,CHR$(128);
970 GOTO690
980 REM
990 REM ALARM
1000 REM
1010 FORX=1TO1000
1020 FORY=1TO20
1030 PLAY"O5V31L255EP255"
1040 NEXT
1050 FORZ=1TO200:NEXT
1060 E$=INKEY$
1070 IF E$=CHR$(13) THEN RUN
1080 IF E$=CHR$(32) THEN1100
1090 NEXT
1100 CLS
```

LANDMINE

DESCRIPTION

This is a game of increasing difficulty, the object being to get your man home safely through the minefield. The computer offers help by warning you where the mines are, up to two moves away from the man in any direction. The warning given consists of a high pitched alarm should there be a mine one move away from the man and a lower pitched alarm if the mine is two moves away.

Each time the player elects to play again one more mine is added to the minefield to increase the difficulty of play.

RUNNINGTHEPROGRAM

After typing RUN the computer will print the number of mines present, which are 8 initially. It will then display the minefield, the man and his home represented by '&' and 'H' characters respectively.

To move the man in the required direction use the number keys, thus:

```
7    8    9
4         6
1    2    3
```

Remember to listen for the landmine warnings!

PROGRAM STRUCTURE

The lines of interest in this program are as follows:

50 Sets up 8 mines initially.
60 Array for storing grass and mines.
80-120 Sets up grass in all positions in the array.
160-230 Chooses random positions for mines.
240 Sets home, 'H', in bottom right position.
300-550 Displays minefield.
590-610 Inputs moves.
650-880 Increases or decreases X and Y positions according to move input.
890-920 Check for valid move.
930-990 Displays grass path taken by man.
1040-1190 Check for mines near man.
1240-1320 Displays minefield successfully crossed.
1360-1460 Displays mine stepped on.
1470-1530 Input for new go or end game.
1520 Increases the number of mines by one.
1640-1750 Displays routine.
1820-2050 Authorises warning to man that mines are in the vicinity.
2150-2330 Displays explosion.
2370-2470 Input move and sound alarm if mines are close.

```
10 REM LANDMINE
20 REM ************************
30 REM
40 MM=8
50 DIMA$(28,11)
60 V=0
70 CLS
80 FORI=1TO28
90 FORJ=1TO11
100 A$(I,J)=":"
110 NEXT
120 NEXT
130 REM
140 REM SET UP MINES AT RANDOM
150 REM
160 FORI=1TOMM
170 X=RND(28)
180 Y=RND(11)
190 IFA$(X,Y)="@" THEN170
200 IFX=1 AND Y=1 THEN170
210 IFX=26 AND Y=11 THEN170
220 A$(X,Y)="@"
230 NEXT
240 A$(28,11)="H"
250 PRINTTAB(8)"NO. OF MINES =";MM;
260 REM
270 REM PRINT MINEFIELD ON
280 REM SCREEN
290 REM
```

```
300 CL=29
310 Z$=":"
320 T$="H"
330 ST=2
340 FOR RW=2TO12
350 GOSUB1640
360 NEXT
370 CL=29
380 RW=13
390 Z$=CHR$(239)
400 T$="H"
410 ST=2
420 GOSUB1640
430 RW=1
440 GOSUB1640
450 CL=1
460 RW=13
470 Z$=CHR$(239)
480 T$="V"
490 GOSUB1640
500 CL=30
510 GOSUB1640
520 X=1
530 Y=1
540 Z$="H":PRINT@413,Z$;
550 GOTO1040
560 GOSUB2100
570 IF V=0 THEN PRINT@66,"S";:V=V+1:GOTO590
580 PRINT"&";
```

```
590 GOSUB2370
600 GG=VAL(I$)
610 IFGG=0THEN560
620 AA=X
630 BB=Y
640 ON GG GOTO650,680,710,740,770,780,810,840,870
650 XX=X-1
660 YY=Y+1
670 GOTO890
680 XX=X
690 YY=Y+1
700 GOTO890
710 XX=X+1
720 YY=Y+1
730 GOTO890
740 XX=X-1
750 YY=Y
760 GOTO890
770 GOTO560
780 XX=X+1
790 YY=Y
800 GOTO890
810 XX=X-1
820 YY=Y-1
830 GOTO890
840 XX=X
850 YY=Y-1
860 GOTO890
870 XX=X+1
```

```
880 YY=Y-1
890 IFXX<1 OR XX>28 THEN560
900 IFYY<1 OR YY>11 THEN560
910 X=XX
920 Y=YY
930 CL=AA+1
940 RW=BB+1
950 T$="H"
960 Z$=":"
970 ST=2
980 GOSUB1640
990 A$(AA,BB)="&"
1000 REM
1010 REM CHECK FOR MINES
1020 REM IN VICINITY
1030 REM
1040 IFA$(X,Y)="@"THEN1360
1050 A$(X,Y)="&"
1060 IFX=28 AND Y=11 THEN1240
1070 REM
1080 REM CHECK FOR 1 MOVE AWAY
1090 REM
1100 MS=1
1110 GOSUB1820
1120 IFMN<>0 THEN1180
1130 REM
1140 REM CHECK FOR 2 MOVES AWAY
1150 REM
1160 MS=2
```

```
1170 GOSUB1820
1180 GOSUB2100
1190 GOTO560
1200 REM
1210 REM MINEFIELD CROSSED
1220 REM SUCCESSFULLY
1230 REM
1240 CL=1
1250 RW=15
1260 T$="H"
1270 PRINT@448,""
1280 Z$="MINEFIELD SUCCESSFULLY CROSSED"
1290 ST=1
1300 GOSUB1640
1310 A$(28,11)="H"
1320 GOTO1380
1330 REM
1340 REM LANDMINE EXPLODES
1350 REM
1360 GOSUB2150
1370 A$(X,Y)=CHR$(191)
1380 FORY=1TO11
1390 X=1
1400 GOSUB2100
1410 FORX=1TO28
1420 PRINT@(Y+1)*32+X+1,A$(X,Y);:GOTO1430
1430 NEXT
1440 NEXT
1450 PRINT@66,"S";
```

```
1460 PRINT@448,""
1470 PRINT@480,"       ANOTHER GO ? (Y/N)      ";
1480 C$=INKEY$
1490 IFC$=""THEN1480
1500 IFC$<>"N"THEN 1520
1510 GOTO2480
1520 MM=MM+1
1530 GOTO60
1540 REM
1550 REM SUBROUTINE TO DRAW
1560 REM ON SCREEN
1570 REM CL=START COLUMN
1580 REM RW=START ROW
1590 REM T$="V" - VERTICAL
1600 REM T$="H" - HORIZONTAL
1610 REM ST=START
1620 REM Z$=TEXT TO PRINT
1630 REM
1640 IF T$="V" THEN1720
1650 GOTO1680
1660 PRINT@RW*32+CL,Z$;
1670 RETURN
1680 FOR I=ST TO CL
1690 PRINT@RW*32+I,Z$;
1700 NEXT
1710 RETURN
1720 FORI=1TORW
1730 PRINT@I*32+CL,Z$;
1740 NEXT
```

```
1750 RETURN
1760 REM
1770 REM CHECK FOR MINES IN
1780 REM VICINITY
1790 REM MS=1 CLOSE SEARCH
1800 REM MS=2 WIDE SEARCH
1810 REM
1820 MN=0
1830 FORI=X-MS TO X+MS
1840 FORJ=Y-MS TO Y+MS
1850 IFI<1 OR J<1 THEN1880
1860 IFI>28 OR J>11THEN1880
1870 IFA$(I,J)="@" THENMN=MN+1
1880 NEXT
1890 NEXT
1900 Z$=STRING$(29," ")
1910 IFMN<=0 THEN1930
1920 Z$=STRING$(9," ")+STR$(MN)+" MINE(S)"
1930 CL=1
1940 RW=14
1950 T$="H"
1960 ST=0
1970 GOSUB1640
1980 IFMN<=0THEN2000
1990 Z$=STRING$(7," ")+STR$(MS)+" MOVE(S) AWAY"
2000 CL=1
2010 RW=15
2020 T$="H"
2030 ST=0
```

```
2040 GOSUB1640
2050 RETURN
2060 REM
2070 REM POSITION WITHIN
2080 REM MINEFIELD
2090 REM
2100 PRINT@(Y+1)*32+X+1,"";
2110 RETURN
2120 REM
2130 REM EXPLOSION
2140 REM
2150 CL=X+1
2160 RW=Y+1
2170 Z$=" "
2180 GOSUB1660
2190 FORP=1TO5
2200 PLAY"O1;T155;L255;V10;G"
2210 NEXT
2220 Z$="*"
2230 GOSUB1660
2240 PLAY"01;L255;V13;F;V16;L255;E;V22;L255;D;V27;L255;C;V30;L255;B;L255;A"
2250 Z$="#"
2260 GOSUB1660
2270 FORP=1TO5
2280 PLAY"O1;T155;L255;P255;V31;G"
2290 NEXT
2300 Z$=CHR$(191)
2310 GOSUB1660
2320 PLAY"O1;T120;L255;P255;V30;A;T90;V27;B;T50;V22;C;T30;V18;D"
```

```
2330 RETURN
2340 REM
2350 REM GET MOVE AND ALARM?
2360 REM
2370 IFMN<>0 THEN2410
2380 I$=INKEY$
2390 IFI$="" THEN2370
2400 RETURN
2410 IFMS=2 THEN2440
2420 PLAY"O4V31T2L30P30F"
2430 GOTO2450
2440 PLAY"O4V31T2L10P10C"
2450 I$=INKEY$
2460 IFI$="" THEN2410
2470 RETURN
2480 CLS
```

MINDREADER

DESCRIPTION

This program was developed as an example of pseudo-artificial intelligence. The program is limited and will only make sense if the user keeps within the limitations of the program and specifically the database.

The program asks the user a question and then asks the user his feelings on the subject. The program waits forthe user to ENTER an answer which includes personal references i.e. 'I', 'ME', or 'MY'. If, after three answers, the user still hasn't used one of the three personal references, Mindreader then asks him to suggest another subject.

If the user has used a personal reference in a response, Mindreader changes it from the first person to the second person and in this way uses the response to direct a question atthe user.

RUNNING THE PROGRAM

When loaded type RUN. Mindreader will start by asking the player's name and then the subject he wants to talk about. If you wish to end the conversation, type 'BYE'.

PROGRAM STRUCTURE

The lines of most interest in this program are as follows:

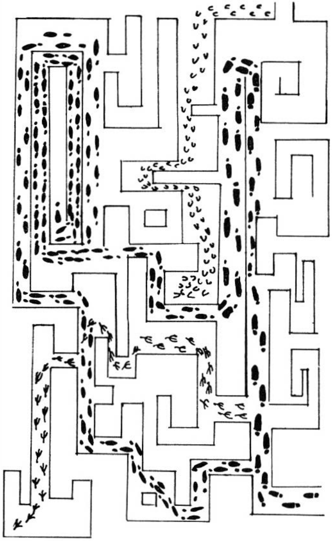
80-170 Asks the player's name, a topic and what he wants to say about the subject.

180-440 Asks for the player's further views about the subject under discussion. Asks for another subject if the player makes no personal references in three answers or tells the player to talk about his views if he were to ask Mindreader a question.

480-1080 Analyses the player's statement and directs an appropriate response at him.

```
650 GOTO1050
660 IF MID$(A$,C,4)<>" ME "THEN730
670 L=3
680 N=1
690 D=C
700 IF (D+L)<LE THENB$=B$+" YOU" M=2
710 IF (D+L)>=LE THEN M=1
720 GOTO1050
730 IF MID$(A$,C,8)<>"YOU ARE " THEN780
740 B$=B$+"I AM "
750 L=8
760 N=2
770 GOTO1050
780 IF MID$(A$,C,5)<>"YOUR " THEN830
790 B$=B$+"MY "
800 L=5
810 N=2
820 GOTO1050
830 IF MID$(A$,C,3)<>"MY " THEN880
840 B$=B$+"YOUR "
850 L=3
860 N=1
870 GOTO1050
880 IF MID$(A$,C,4)<>"YOU " OR NV=1 THEN930
890 B$=B$+"I "
900 L=4
910 N=2
920 GOTO1050
930 IF MID$(A$,C,4)<>"YOU " OR NV=0 THEN980
940 B$=B$+"ME "
950 L=4
960 N=2
```

```
970 GOTO1050
980 IF MID$(A$,C,2)<>"I " THEN1030
990 B$=B$+"YOU "
1000 L=2
1010 N=1
1020 GOTO1050
1030 B$=B$+MID$(A$,C,1)
1040 L=1
1050 IFN>NV THENNV=N
1060 C=C+L
1070 IF C<=LE THEN610
1080 RETURN
1090 IF M=1 THENPRINT"WHY DO YOU THINK ":PRINTB$;" YOU"
1100 IF M=2 THENPRINT"WHY DO YOU THINK":PRINTB$
1110 GOTO170
1120 FORTD=1TO3000:NEXT
1130 CLS
```



LABYRINTH

DESCRIPTION

Provided with five arrows, the player's aim is to search out and slay the slumbering dragon hidden in one of the rooms of the maze. Dragons have an incredibly bad odour about their bodies, and so the dragon can be detected within one tunnel distance, but the chances are that it will move to the next room once it has been woken up by the passing of an arrow or the presence of the player entering the room.

Arrows can be guided down a total of 5 tunnels but may ricochet and hit the player. Other dangers are bottomless pits and gigantic bats which do nasty things to players of this game.

The program consists mainly of text and the maze is designed around a dodecahedron.

RUNNING THE PROGRAM

Load and type RUN.

PROGRAM STRUCTURE

The lines of interest in this game are as follows:

160-190 Sets up random positions for players, hazards and the dragon. Checks to ensure their positions are different.

230-570 Main routine calling the subroutines as required.

580-750 Informs of hazards, gives the player his position and tunnels radiating from that position.

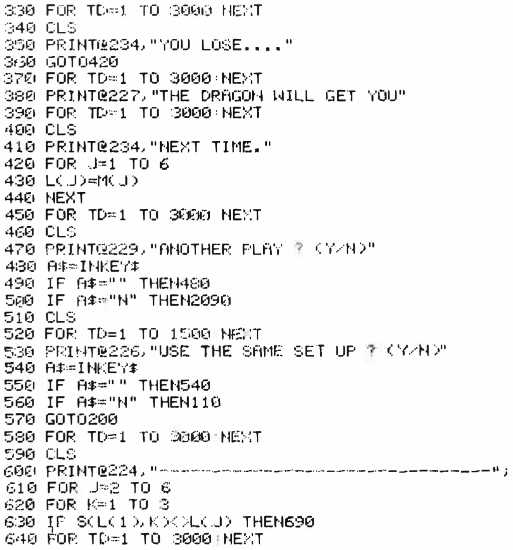
760-890 Asks whether player wishes to fire arrows or make a move.

900-1400 Fires arrows and stores the result.

1470-1570 Moves dragon and burns up player if in the same room as the dragon.

1580-1970 Moves player and immediately actuates any hazards encountered.

```
10 REM LABYRINTH
20 REM *************************
30 REM
40 CLS
50 DIMS(20,3),L(6),M(6),P(5)
60 FOR J=1 TO 20
70 FOR K=1 TO 3
80 READA
90 S(J,K)=A
100 NEXTK,J
110 FOR J=1 TO 6
120 L(J)=RND(20)
130 M(J)=L(J)
140 NEXT
150 FOR J=1 TO 6
160 FOR K=1 TO 6
170 IF J=K THEN190
180 IF L(J)=L(K) THEN110
190 NEXTK,J
200 A=5
210 L=L(1)
220 CLS
230 PRINT@235,"LABYRINTH"
240 GOSUB580
250 GOSUB760
260 IF G=2 THEN300
270 GOSUB900
280 IF F=0 THEN240
290 GOTO310
300 GOSUB1580
310 IF F=0 THEN240
320 IF F>0 THEN370
```



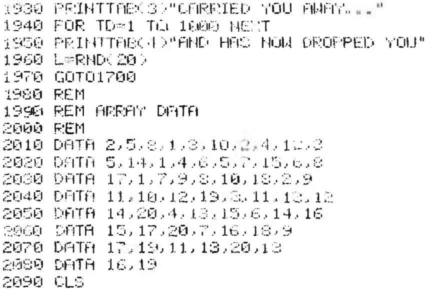
```
330 FOR TD=1 TO 3000 NEXT
340 CLS
350 PRINT@234,"YOU LOSE...."
360 GOTO420
370 FOR TD=1 TO 3000:NEXT
380 PRINT@227,"THE DRAGON WILL GET YOU"
390 FOR TD=1 TO 3000:NEXT
400 CLS
410 PRINT@234,"NEXT TIME."
420 FOR J=1 TO 6
430 L(J)=M(J)
440 NEXT
450 FOR TD=1 TO 3000 NEXT
460 CLS
470 PRINT@229,"ANOTHER PLAY ? (Y/N)"
480 A$=INKEY$
490 IF A$="" THEN480
500 IF A$="N" THEN2090
510 CLS
520 FOR TD=1 TO 1500 NEXT
530 PRINT@226,"USE THE SAME SET UP ? (Y/N)"
540 A$=INKEY$
550 IF A$="" THEN540
560 IF A$="N" THEN110
570 GOTO200
580 FOR TD=1 TO 3000:NEXT
590 CLS
600 PRINT@224,"--------------------------------------";
610 FOR J=2 TO 6
620 FOR K=1 TO 3
630 IF S(L(1),K)<>L(J) THEN690
640 FOR TD=1 TO 3000:NEXT
```

```
650 CLS
660 IF J=2 THEN PRINT@228,"I CAN SMELL A DRAGON..."
670 IF J=3 OR J=4 THEN PRINT@231,"I FEEL A DRAFT"
680 IF J=5 OR J=6 THEN PRINT@231,"I HEAR WING-FLAPS"
690 NEXTK,J
700 FOR TD=1 TO 3000:NEXT
710 CLS
720 PRINT@456,"YOU ARE IN ROOM";L(1)
730 FOR TD=1 TO 1000:NEXT
740 PRINT TAB(4)"TUNNELS LEAD TO"S(L,1);S(L,2);S(L,3)
750 RETURN
760 FOR TD=1 TO 1000:NEXT
770 PRINTTAB(5)"SHOOT OR MOVE ? (S/M)"
780 B$=INKEY$
790 IF B$="" THEN780
800 IF B$<>"S" THEN850
810 G=1
820 PRINT:PRINT TAB(12)"SHOOT"
830 FOR TD=1 TO 1000:NEXT
840 RETURN
850 IF B$<>"M" THEN780
860 PRINT:PRINT TAB(12)"MOVE"
870 FOR TD=1 TO 1000:NEXT
880 G=2
890 RETURN
900 F=0
910 FOR TD=1 TO 1000:NEXT
920 INPUT "      HOW FAR TO GO (1-5) ";D
930 PRINT:PRINTTAB(15)D
940 IF D<1 OR D>5 THEN920
950 FOR K=1 TO D
960 FOR TD=1 TO 1000:NEXT
```

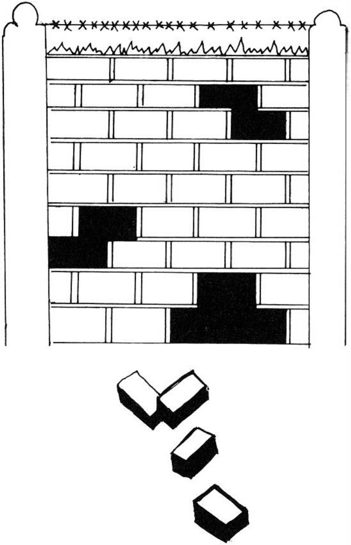
```
970 INPUT"      ROOM TO GO THROUGH ";L
980 PRINTTAB(14)L
990 P(K)=L
1000 IF K<3 THEN1050
1010 IF P(K)<>P(K-2) THEN1050
1020 FOR TD=1 TO 1000:NEXT
1030 PRINT TAB(9)"NO U-TURNS FOR ARROWS."
1040 GOTO960
1050 NEXT
1060 L=L(1)
1070 FOR K=1 TO D
1080 FOR M=1 TO 3
1090 IF S(L,M)=P(K) THEN1340
1100 NEXTM
1110 L=S(L,RND(3))
1120 FORTD=1 TO 1000 NEXT
1130 CLS
1140 PRINT@228,"....BOINK.."
1150 GOTO1350
1160 FOR TD=1 TO 1000:NEXT
1170 CLS
1180 PRINT@231,"ARROW HIT A WALL."
1190 GOTO1410
1200 NEXTK
1210 FOR TD=1 TO.1000:NEXT
1220 CLS
1230 PRINT@234,"MISSED...."
1240 L=L(1)
1250 GOSUB1470
1260 A=A-1
1270 IF A>0 THEN RETURN
1280 F=-1
```

```
1290 FOR TD=1 TO 1000:NEXT
1300 PRINT@228,"OH DEAR. YOU HAVE SPENT"
1310 FOR TD=1 TO 1000:NEXT
1320 PRINT TAB(4)"ALL YOUR ARROWS."
1330 RETURN
1340 L=P(K)
1350 IF L<>L(2) THEN1160
1360 FOR TD=1 TO 1000:NEXT
1370 CLS
1380 PRINT@228,"AHA....YOU SHOT THE DRAGON"
1390 F=1
1400 RETURN
1410 IF L<>L(1) THEN1200
1420 FOR TD=1 TO 1000:NEXT
1430 CLS
1440 PRINT@230,"ARGHH....ARROW GOT YOU"
1450 F=-1
1460 GOTO1330
1470 K=RND(4)
1480 IF K=4 THEN1500
1490 L(2)=S(L(2),K)
1500 IF L(2)<>L THEN RETURN
1510 FOR TD=1 TO 1000:NEXT
1520 CLS
1530 PRINT@232,"WHAT A SHAME...."
1540 FOR TD=1 TO 1000:NEXT
1550 PRINT"THE DRAGON BURNT YOU TO A CRISP"
1560 F=-1
1570 RETURN
1580 F=0
1590 FOR TD=1 TO 1000:NEXT
1600 INPUT"            WHERE TO ";L
```

```
1610 PRINTTAB(14)L
1620 IF L<0 OR L>20 THEN1590
1630 FOR K=1 TO 3
1640 IF S(L(1),K)=L THEN1700
1650 NEXTK
1660 IF L=L(1) THEN1700
1670 FOR TD=1 TO 1000:NEXT
1680 PRINTTAB(2)"NO TUNNEL IN THAT DIRECTION"
1690 GOTO1590
1700 L(1)=L
1710 IF L<>L(2) THEN1800
1720 FOR TD=1 TO 1000:NEXT
1730 CLS
1740 PRINT@229,"OOPS YOU'VE JUST BUMPED"
1750 FOR TD=1 TO 1000:NEXT
1760 PRINTTAB(5)"INTO A SLUMBERING DRAGON"
1770 GOSUB1470
1780 IF F=0 THEN1880
1790 RETURN
1800 IF L<>L(3) AND L<>L(4) THEN1880
1810 FOR TD=1 TO 1000:NEXT
1820 CLS
1830 PRINT@224,"YOU JUST FELL INTO A VERY DEEP"
1840 FOR TD=1 TO 1000:NEXT
1850 PRINTTAB(5)"PIT .... YOU'RE DEAD!"
1860 F=-1
1870 RETURN
1880 IF L<>L(5) AND L<>L(6) THEN RETURN
1890 FOR TD=1 TO 1000:NEXT
1900 CLS
1910 PRINT@192,"A GIGANTIC BAT GRABBED YOU,"
1920 FORTD=1 TO 1000:NEXT
```



```
1930 PRINTTAB(3)"CARRIED YOU AWAY..."
1940 FOR TD=1 TO 1000 NEXT
1950 PRINTTAB(4)"AND HAS NOW DROPPED YOU"
1960 L=RND(20)
1970 GOTO1700
1980 REM
1990 REM ARRAY DATA
2000 REM
2010 DATA 2,5,8,1,3,10,2,4,12,3
2020 DATA 5,14,1,4,6,5,7,15,6,8
2030 DATA 17,1,7,9,8,10,18,2,9
2040 DATA 11,10,12,19,3,11,13,12
2050 DATA 14,20,4,13,15,6,14,16
2060 DATA 15,17,20,7,16,18,9
2070 DATA 17,19,11,18,20,13
2080 DATA 16,19
2090 CLS
```



BREAKOUT

DESCRIPTION

This game is the computer version of the popular arcade game Breakout. It makes good use of the DRAGON graphics character set, resulting in a very colourful display.

RUNNING THE PROGRAM

After typing RUN, the program will display a few instructions mainly explaining that the bat is moved using the left and right arrow keys. Hitting any key will start the game running. There are five balls to start with and the object of the game is to knock out as many of the bricks as possible. When all the bricks have been cleared from the screen, a new set will appear and the game is played over again.

PROGRAM STRUCTURE

The lines of interest in this game are as follows:

100-400 Set up the display.

540-570 Choose random ball start position and direction.

580-600 Display the score at the bottom of the screen.

610-680 Input for another game.

720-830 Check to see which arrow key has been pressed, if any, and move the bat. If either key has been hit again or the bat has reached the end of the screen, stop the bat.

870-940 Check if the ball has hit a brick, the bat, one of the sides or the top of the screen.

950-990 The bat has missed the ball. Decrease the number of balls left by one.

1000 POKE 'ball' graphic character onto the screen, if it encounters a blank space.

1010-1690 Routines to POKE the ball onto the screen.

1700-1950 Blank out the bricks hit by the ball and score the appropriate points.

1990-2080 Print the instructions.

```
10 REM BREAKOUT
20 REM *************************
30 REM
40 GOSUB1990
50 P=3
60 BA=5
70 REM
80 REM DISPLAY
90 REM
100 CLS0
110 PRINTCHR$(238);
120 PRINTSTRING$(30,236);
130 PRINTCHR$(237),
140 FORI=1 TO 14
150 PRINTCHR$(234);
160 PRINTSTRING$(30,128);
170 PRINTCHR$(229);
180 NEXTI
190 PRINT@449,STRING$(30,128);
200 PRINT@463,STRING$(2,195);
210 PRINT@33,"";
220 FORX=1TO15
230 PRINTCHR$(131);
240 PRINTCHR$(130);
250 NEXTX
260 PRINT@65,"";
270 FOR X=1 TO 15
280 PRINTCHR$(179);
290 PRINTCHR$(178);
300 NEXT
310 PRINT@97,"";
320 FOR X=1 TO 15
```

```
330 PRINTCHR$(147);
340 PRINTCHR$(146);
350 NEXTX
360 PRINT@129,"";
370 FOR X=1 TO 15
380 PRINTCHR$(163);
390 PRINTCHR$(162);
400 NEXTX
410 R=0
420 O=0
430 A(1)=200
440 A(2)=193
450 A(3)=194
460 A(4)=196
470 B=463
480 V=33
490 Y=1
500 REM
510 REM RANDOM BALL START
520 REM POSITION AND DIRECTION
530 REM
540 Z=1184+RND(30)
550 G=RND(2)-1
560 IF G=1 THEN R=1
570 IF G=0 THEN R=3
580 PRINT@485,"SCORE";SC;
590 PRINT@498,"BALL NO.";BA;
600 IF BA<>0 THEN720
610 FOR TD=1 TO 4000:NEXT
620 CLS
630 PRINT@233,"ANOTHER GAME ?";
640 A$=INKEY$
```

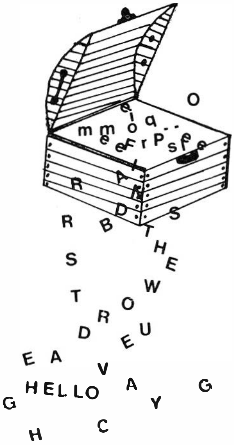
```
650 IF A$="" THEN640
660 IF A$="Y" THEN RUN
670 IF A$<>"N" THEN640
680 GOTO2090
690 REM
700 REM PLAY GAME
710 REM
720 B$=INKEY$
730 IF B$=CHR$(8) OR B$=CHR$(9) THEN FF=FF+1
740 IF FF/2<>INT(FF/2) THEN PRINT@B,STRING$(2,128);
750 IF B$<>"" THEN C$=B$
760 IF FF/2=INT(FF/2) THENFF=0:GOTO820
770 I=ASC(C$)
780 IF I=9 THEN B=B+1:GOTO800
790 IF I=8 THEN B=B-1
800 IF B>477 THENB=477:FF=0
810 IF B<449 THEN B=449:FF=0
820 PRINT@B,STRING$(2,195);
830 R=R+1
840 IF O=1 THEN IF R>4 THEN R=3:Z=Z+V:POKEZ-V,128:GOTO870
850 IF O=1 THEN870
860 IF R>2 THEN R=1:Z=Z+V:POKEZ-V,128
870 X=PEEK(Z)
880 IF X=195 THEN 1520
890 IF X=130 OR X=178 OR X=146 OR X=162 THEN K=-1:GOTO1700
900 IF X=131 OR X=179 OR X=147 OR X=163 THEN K=1:GOTO1700
910 IF X=234 THEN1180
920 IF X=229 THEN1350
930 IF Z<=1055 THEN1010
940 IF Z<=1502 THEN1000
950 BA=BA-1
960 PLAY"V31L2O2C"
```

```
970 PRINT@B,STRING$(2,128);
980 FF=0
990 GOTO410
1000 POKEZ,A(R):GOTO720
1010 Y=1:PLAY"V31L3005A#"
1020 IF NOT(R=1 OR R=2) THEN1100
1030 Z=Z+32
1040 R=3
1050 V=31
1060 O=1
1070 A(3)=196
1080 A(4)=194
1090 GOTO720
1100 IF NOT(R=3 OR R=4) THEN1180
1110 Z=Z+32
1120 R=1
1130 V=33
1140 O=0
1150 A(1)=200
1160 A(2)=193
1170 GOTO720
1180 PLAY"V31L3005A#"
1190 IF NOT(R=1 OR R=2) THEN1270
1200 Z=Z+1
1210 R=3
1220 V=-31
1230 O=1
1240 A(3)=194
1250 A(4)=196
1260 GOTO720
1270 IF NOT(R=3 OR R=4) THEN1350
1280 Z=Z+1
```

```
1290 R=1
1300 V=33
1310 O=0
1320 A(1)=200
1330 A(2)=193
1340 GOTO720
1350 PLAY"V31L3005A#"
1360 IF NOT(R=1 OR R=2) THEN1440
1370 Z=Z-1
1380 R=3
1390 V=31
1400 O=1
1410 A(3)=196
1420 A(4)=194
1430 GOTO720
1440 IF NOT(R=3 OR R=4) THEN1520
1450 Z=Z-1
1460 R=1
1470 V=-33
1480 O=0
1490 A(1)=193
1500 A(2)=200
1510 GOTO720
1520 Y=0
1530 PLAY"V31L3005A#"
1540 IF NOT(R=1 OR R=2) THEN1620
1550 Z=Z-32
1560 R=3
1570 V=-31
1580 O=1
1590 A(3)=194
1600 A(4)=196
```

```
1610 GOTO580
1620 IF NOT(R=3 OR R=4) THEN1700
1630 Z=Z-32
1640 R=1
1650 V=-33
1660 O=0
1670 A(1)=193
1680 A(2)=200
1690 GOTO580
1700 PLAY"V31L30O3D"
1710 POKEZ,128
1720 POKEZ+K,128
1730 IF Z<1088 THEN SC=SC+7:GOTO1770
1740 IF Z<1120 THEN SC=SC+5:GOTO1770
1750 IF Z<1152 THEN SC=SC+3:GOTO1770
1760 IF Z<1184 THEN SC=SC+1
1770 IF SC/240=INT(SC/240) THENPRINT@B,STRING$(2,128);:FF=0:GOTO210
1780 IF Y=1 THEN1520
1790 Y=1
1800 IF NOT(R=1 OR R=2) THEN1880
1810 Z=Z+32
1820 R=3
1830 V=31
1840 O=1
1850 A(3)=196
1860 A(4)=194
1870 GOTO580
1880 IF NOT(R=3 OR R=4) THEN1990     REM IF SO THEN MISTAKE
1890 Z=Z+32
1900 R=1
1910 V=33
1920 O=0
```

```
1930 A(1)=200
1940 A(2)=193
1950 GOTO580
1960 REM
1970 REM INSTRUCTIONS
1980 REM
1990 CLS
2000 PRINT:PRINT"COMMANDS:-"
2010 PRINT@4,"*** DRAGON BREAKOUT ***"
2020 PRINT:PRINT"COMMANDS:-"
2030 PRINT;PRINT"ARROW LEFT - MOVE LEFT"
2040 PRINT:PRINT"ARROW RIGHT - MOVE RIGHT"
2050 PRINT:PRINT"EITHER - TO STOP
2060 PRINT@425,"PRESS ANY KEY"
2070 IF INKEY$="" THEN2070
2080 RETURN
2090 CLS
```



FIND THE WORD

DESCRIPTION

In this game the player inputs a word length, and the computer will choose a word from the data statements of that length and put it in a matrix with a mix of random letters. The player then has to find that word in the matrix. The word may be horizontal, vertical or diagonal and may be backwards. If the word has not been found after three guesses, the computer will give clues by telling the player the first letter of the word, then the second, etc, until it can give no more clues without giving the whole word.

RUNNING THE PROGRAM

After you have typed RUN, the computer will ask if instructions are required, and if the answer is yes then a page of instructions will be printed on the screen. The game will than start by asking how many letters in the word. The variable M chooses whether the word is diagonal or not. Then the guessing of the word begins. When the word has been guessed or the computer has run out of clues, the word will be printed along with the running total.

PROGRAM STRUCTURE

The lines of interest in this program are as follows:

260-340	Input word length.
390-460	Choose a word of input length.
510-910	Create grid and put the word somewhere in the grid.
950-1050	Print the grid on the screen.

1090-1420 Guess at word and check to see if the guess is correct.
1470-1580 Guessed the word or no more clues available.
1620-1630 Data for three letter words.
1670-1690 Data for four letter words.
1730-1750 Data for five letter words.
1790-1820 Data for six letter words
1860-1930 Print instructions.

```
10 REM FIND THE WORD
20 REM ************************
30 REM
40 S=1
50 T=0
60 T$="ABCDEFGHIJKLMNOPQRSTUVWXYZ"
80 CLS:PRINT@230,"*** FIND THE WORD ***"
85 FORTD=1TO2000:NEXT
90 CLS:PRINT@226,"DO YOU WANT INSTRUCTIONS ?"
100 A$=INKEY$
105 IFA$=""THEN100
110 IF A$="Y" THEN GOSUB1870
120 P=0:GOSUB260
130 GOSUB400
140 GOSUB510
150 GOSUB940
160 GOSUB1070
170 IFO=0THEN150
180 GOSUB1470
190 IFO=0THEN1590
200 GOSUB1380
210 GOTO120
220 REM
230 REM INPUT WORD LENGTH
240 REM
260 M=RND(2)
300 CLS:PRINT@229,"HOW LONG IS THE WORD ?"
310 PRINTTAB(8)"(LENGTH 3 TO 6)"
320 M$=INKEY$
325 IFM$=""THEN320
330 IFASC(M$)<51 OR ASC(M$)>54 THEN320
340 L=VAL(M$)
```

```
350 N=L+1
360 RETURN
370 REM
380 REM CHOSE A WORD OF THAT
385 REM LENGTH
390 REM
400 RESTORE
410 READC$
420 IFLEN(C$)<>L THEN410
430 Q=RND(15)
440 FORX=1TOQ
450 READW$
460 NEXT
470 RETURN
480 REM
490 REM CREATE A GRID OF RANDOM
495 REM LETTERS
500 REM
510 X=0
520 R=RND(N)
530 K=RND(N)
540 IFK<=2 OR K>=N-1 THEN580
550 IFR<=2 OR R>=N-1 THEN580
560 X=X+1
570 GOTO520
580 FORX=1TON
590 FORY=1TON
600 Q=RND(26)
610 Q$=MID$(T$,Q,1)
620 R$(X,Y)=Q$
630 NEXTY
640 NEXTX
```

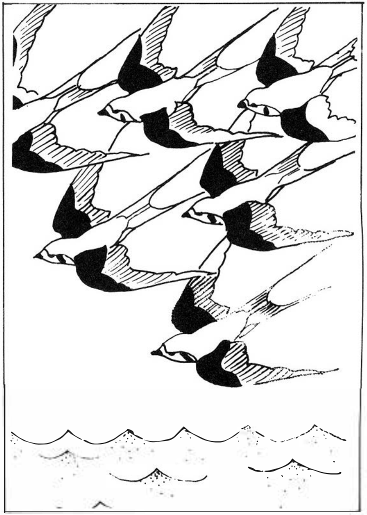
```
650 IFM=2THEN810
660 IFR<=2 OR R>=N-1 THEN740
670 IFK<=2THEN KR=1
680 IFK>=N-1 THEN KR=-1
690 FORX=1TOLEN(W$)
700 R$(K,R)=MID$(W$,X,1)
710 K=K+KR
720 NEXT
730 RETURN
740 IFR<=2 THENRR=1
750 IFR>=N-1 THENRR=-1
760 FORX=1TOLEN(W$)
770 R$(K,R)=MID$(W$,X,1)
780 R=R+RR
790 NEXT
800 RETURN
810 KR=0:RR=0
815 IFR<=2 THENRR=1
820 IFR>=N-1 THENRR=-1
830 IFK<=2 THENKR=1
840 IFK>=N-1 THENKR=-1
850 FORX=1TOLEN(W$)
860 R$(K,R)=MID$(W$,X,1)
870 K=K+KR
880 R=R+RR
890 NEXT
900 RETURN
910 REM
920 REM PRINT GRID ON SCREEN
930 REM
940 CLS:PRINTTAB(5)STRING$(20,"=")
950 FORY=1TON:PRINTTAB(16-N)"";
```

```
960 FORX=1TON
970 PRINTR$(X,Y);" ";
980 NEXTX
990 PRINT
1000 NEXTY
1010 PRINTTAB(5)STRING$(20,"=")
1015 PRINT
1020 PRINTTAB(5)"LENGTH";L
1030 RETURN
1040 REM
1050 REM GUESS AT WORD
1060 REM
1070 INPUT"WHAT IS THE WORD";B$
1090 P=P+1
1100 IFB$=W$THEN1310
1110 PRINTTAB(5)"SORRY, wrong"
1120 IF P-2=L THEN1240
1130 IFP>2THEN1170
1135 IFP=1THEN1150
1140 PRINTTAB(5)"NOT VERY GOOD"
1150 O=0:FORJ=1TO3000:NEXT
1160 RETURN
1170 H$=LEFT$(W$,P-2)
1180 PRINT"YOU NEED HELP"
1190 IFP>3THEN1220
1200 PRINTTAB(5)"THE FIRST LETTER IS ";H$
1210 GOTO1150
1220 PRINT"THE FIRST";P-2;"LETTERS ARE ";H$
1230 GOTO1150
1240 PRINT"YOU DID NOT GET THE WORD RIGHT."
1260 PRINT:PRINT"IT WAS ";W$;
1270 FORTD=1TO3000:NEXT
```

```
1280 O=-1
1290 RETURN
1310 PRINTTAB(5)"THAT'S IT"
1320 PRINT"YOU GOT IT IN ";P;
1330 IFP>1THENPRINTTAB(5)"TRIES":GOTO1350
1340 PRINTTAB(5)"TRY"
1350 FORTD=1TO3000:NEXT
1360 O=1
1370 RETURN
1380 S=S+1
1390 RR=0
1400 KR=0
1410 P=0
1420 RETURN
1430 REM
1440 REM GUESSED THE WORD OR NO
1450 REM CLUES AVAILABLE
1460 REM
1470 T=T+(L/P)*(L/P)*M
1490 CLS:PRINT"THE SCORE FOR THIS WORD";INT((L/P)*(L/P)*M)
1500 PRINT:PRINT"(MAXIMUM SCORE WAS";INT(L*L*M);")"
1510 PRINT:PRINT"TOTAL SCORE ";INT(T/S)
1520 PRINT:PRINT:PRINTTAB(5)"ANOTHER GO ?"
1530 A$=INKEY$
1540 IFA$=""THEN1530
1550 IF A$="Y"THEN120
1560 IF A$="N"THEN1590
1570 PRINT:PRINTTAB(5)"Y OR N PLEASE."
1580 GOTO1530
1590 CLS:GOTO9999
1600 REM
1610 REM THREE LETTER WORDS
```

```
1620 REM
1630 DATA 333,TIN,GET,THE,SIT,CAT,DOG,ONE,TWO
1640 DATA SIX,TEN,FIT,BET,SET,DIG,BAD
1650 REM
1660 REM FOUR LETTER WORDS
1670 REM
1680 DATA 4444,FOUR,FIVE,COOL,MAKE,HOLE,POSH
1690 DATA NINE,CLAP,COLD,GOOD,YOUR,WHAT,READ
1700 DATA VAST,LAMP,DUEL
1710 REM
1720 REM FIVE LETTER WORDS
1730 REM
1740 DATA 55555,CHIPS,GREAT,CLEAR,GLASS,MAKER
1750 DATA SPOOL,SLEEP,BEAST,SEVEN,STOOL
1760 DATA EIGHT,PLANT,RADAR,RADIO,WATER
1770 REM
1780 REM SIX LETTER WORDS
1790 REM
1800 DATA 666666,ARCTIC,GUITAR,MINUTE,MIDDLE
1810 DATA PISTOL,ANIMAL,PERMIT,TOWERS,FIRING
1820 DATA DRAGON,ROBOTS,ROCKET,MOSAIC,MEMORY
1830 DATA BANDIT
1840 REM
1850 REM INSTRUCTIONS
1860 REM
1870 CLS:PRINTTAB(10)"INSTRUCTIONS"
1880 PRINT:PRINT" FIND THE WORD IS A GAME WHERE"
1890 PRINT:PRINT"YOU HAVE TO FIND A HIDDEN WORD"
1900 PRINT:PRINT"IN A GRID OF LETTERS."
1910 PRINT:PRINTTAB(12)"CLEAR ?"
1920 IF INKEY$=""THEN1920
```

```
1930 CLS
1940 RETURN
9999 REM END
```



BIRDS DEMO

DESCRIPTION

This program is a demonstration of high resolution graphics and their animation. Displayed on the screen is a sea view. Attractively coloured birds are soaring over the sea.

As the display stops moving, an arrow is fired from the sea and shoots down a bird. A tune is then played before the whole process repeats itself.

RUNNING THE PROGRAM

When the program has loaded, type RUN and watch the display appear on the screen.

PROGRAM STRUCTURE

The program uses a resolution which allows it to use four pairs of pages to animate the birds and waves. Each pair of pages contains the waves and wings of the birds in a slightly different position to each other. Hence when the computer displays the contents of each pair one after another, continuously, the birds appear to be flying and the waves to be moving.

The lines of most interest in this program are:

60-220	Flock 1. Draws the birds in hi-resand stores them in pages 1 and 2.
260-280	Waves 1. Draws the waves in hi-res and stores them in pages 1 and 2.
320-470	Flock 2. Draws the birds in hi-res and stores them in pages 3 and4.

510-530 Waves 2. Draws the waves in hi-resand stores them in pages 3 and 4.

570-730 Flock 3. Draws the birds in hi-resand stores them in pages 5 and 6.

770-790 Waves 3. Draws the waves in hi-res and stores them in pages 5 and 6.

830-990 Flock 4. Draws the birds in hi-res and stores them in pages 7 and 8.

1030-1050 Waves 4. Draws the waves in hi-res and stores them in pages 7 and 8.

1090-1180 'Flicks' through the pairs first forward then backwards one hundred times to give the effect of animation.

1220-1270 Draws and then moves the arrow using GET/PUT functions.

1310-1360 Moves the dead bird to the sea.

```
10 REM BIRDS DEMO
20 REM**************************
30 REM
40 REM FLOCK 1
50 REM
60 PCLEAR8:PMODE1,1:PCLS
70 DIMA(46,12)
80 DRAW"BM20,20R4E2R2E2R4F2R2E2R4F2R2F2R4"
90 DRAW"BM31,16R4U4E2R2F2D4"
100 PAINT(38,12)
110 COLOR7,5:DRAW"BM120,30R4E2R2E2R4F2R2E2R4F2R2F2R4"
120 DRAW"BM131,26R4U4E2R2F2D4"
130 PAINT(138,22)
140 COLOR6,5:DRAW"BM220,35R4E2R2E2R4F2R2E2R4F2R2F2R4"
150 DRAW"BM231,31R4U4E2R2F2D4"
160 PAINT(238,27)
170 DRAW"BM50,50R4E2R2E2R4F2R2E2R4F2R2F2R4"
180 DRAW"BM61,46R4U4E2R2F2D4"
190 PAINT(68,42)
200 GET(17,9)-(63,21),A,G
210 PUT(180,49)-(226,61),A,PSET
220 PUT(90,70)-(136,82),A,PSET
230 REM
240 REM WAVES 1
250 REM
260 DRAW"BM0,150F4R4E4F4R4E4F4R4E4F4R4E4F4R4E4F4R4E4F4R4E4F4R4E4F4R4E4"
270 DRAW"BM108,150F4R4E4F4R4E4F4R4E4F4R4E4F4R4E4F4R4E4F4R4E4F4R4E4F4R4E4"
280 DRAW"BM216,150F4R4E4F4R4E4F4R4E4F3"
290 REM
300 REM FLOCK 2
310 REM
320 PMODE1,3:PCLS:DIMB(46,12)
```

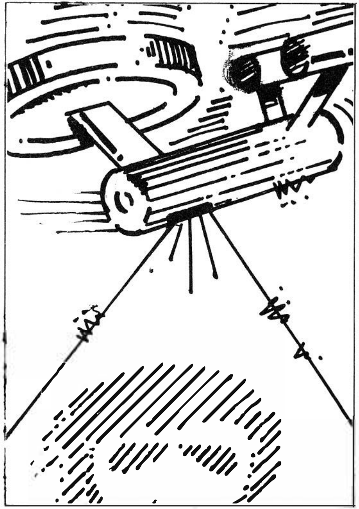
```
330 DRAW"BM18,18R10E2R4F2R2E2R4F2R10"
340 DRAW"BM31,16R4U4E2R2F2D4"
350 PAINT(38,12)
360 COLOR7,5:DRAW"BM118,28R10E2R4F2R2E2R4F2R10"
370 DRAW"BM131,26R4U4E2R2F2D4"
380 PAINT(138,22)
390 COLOR6,5:DRAW"BM218,33R10E2R4F2R2E2R4F2R10"
400 DRAW"BM231,31R4U4E2R2F2D4"
410 PAINT(238,27)
420 DRAW"BM48,48R10E2R4F2R2E2R4F2R10"
430 DRAW"BM61,46R4U4E2R2F2D4"
440 PAINT(68,42)
450 GET(17,9)-(63,21),B,G
460 PUT(180,49)-(226,61),B,PSET
470 PUT(90,70)-(136,82),B,PSET
480 REM
490 REM WAVES 2
500 REM
510 DRAW"BM0,154R2E4F4R4E4F4R4E4F4R4E4F4R4E4F4R4E4F4R4E4F4R4E4F4R4E4F4R2"
520 DRAW"BM108,154R2E4F4R4E4F4R4E4F4R4E4F4R4E4F4R4E4F4R4E4F4R4E4F4R4E4F4R2"
530 DRAW"BM216,154R2E4F4R4E4F4R4E4F4R4"
540 REM
550 REM FLOCK 3
560 REM
570 PMODE1,5:PCLS
580 DIMC(46,12)
590 DRAW"BM16,16R18F2R2E2R18"
600 DRAW"BM31,16R4U4E2R2F2D4"
610 PAINT(38,12)
620 COLOR7,5:DRAW"BM116,26R18F2R2E2R18"
630 DRAW"BM131,26R4U4E2R2F2D4"
640 PAINT(138,22)
```

```
650 COLOR6,5:DRAW"BM216,31R18F2R2E2R18"
660 DRAW"BM231,31R4U4E2R2F2D4"
670 PAINT(238,27)
680 DRAW"BM46,46R18F2R2E2R18"
690 DRAW"BM61,46R4U4E2R2F2D4"
700 PAINT(68,42)
710 GET(17,9)-(63,21),C,G
720 PUT(180,49)-(226,61),C,PSET
730 PUT(90,70)-(136,82),C,PSET
740 REM
750 REM WAVES 3
760 REM
770 DRAW"BM0,150F4R4E4F4R4E4F4R4E4F4R4E4F4R4E4F4R4E4F4R4E4F4R4E4F4R4E4"
780 DRAW"BM108,150F4R4E4F4R4E4F4R4E4F4R4E4F4R4E4F4R4E4F4R4E4F4R4E4F4R4E4"
790 DRAW"BM216,150F4R4E4F4R4E4F4R4E4"
800 REM
810 REM FLOCK 4
820 REM
830 PMODE1,7:PCLS
840 DIMD(46,12)
850 DRAW"BM18,12R2F2R6F2R4F2R2E2R4E2R6E2R2"
860 DRAW"BM31,16R4U4E2R2F2D4"
870 PAINT(38,12)
880 COLOR7,5:DRAW"BM118,22R2F2R6F2R4F2R2E2R4E2R6E2R2"
890 DRAW"BM131,26R4U4E2R2F2D4"
900 PAINT(138,22)
910 COLOR6,5:DRAW"BM218,27R2F2R6F2R4F2R2E2R4E2R6E2R2"
920 DRAW"BM231,31R4U4E2R2F2D4"
930 PAINT(238,27)
940 DRAW"BM48,42R2F2R6F2R4F2R2E2R4E2R6E2R2"
950 DRAW"BM61,46R4U4E2R2F2D4"
960 PAINT(68,42)
```

```
970 GET(17,9)-(63,21),D,G
980 PUT(180,49)-(226,61),D,PSET
990 PUT(90,70)-(136,82),D,PSET
1000 REM
1010 REM WAVES 4
1020 REM
1030 DRAW"BM0,154R2E4F4R4E4F4R4E4F4R4E4F4R4E4F4R4E4F4R4E4F4R4E4F4R4E4F4R2"
1040 DRAW"BM108,154R2E4F4R4E4F4R4E4F4R4E4F4R4E4F4R4E4F4R4E4F4R4E4F4R2"
1050 DRAW"BM216,154R2E4F4R4E4F4R4E4F4R2"
1060 REM
1070 REM ANIMATE
1080 REM
1090 FORZ=40TO50
1100 FORX=1TO7 STEP2
1110 PMODE1,X:SCREEN1,1
1120 FORTD=1TO100:NEXT
1130 NEXT
1140 FORX=7TO1 STEP-2
1150 PMODE1,X:SCREEN1,1
1160 FORTD=1TO100:NEXT
1170 NEXT
1180 NEXTZ
1190 REM
1200 REM DRAW AND MOVE ARROW
1210 REM
1220 COLOR7,5:DRAW"BM111,143U6"
1230 DIME(2,8)
1240 GET(110,136)-(112,144),E,G
1250 FORY=-1TO-53 STEP-1
1260 PUT(110,136+Y)-(112,144+Y),E,PSET
1270 NEXT
1280 REM
```

```
1290 REM KILL BIRD
1300 REM
1310 GET(90,70)-(136,82),A,G
1320 LINE(110,83)-(112,91),PRESET,BF
1330 FORX=1TO60
1340 PUT(90,70+X)-(136,82+X),A,PSET
1350 NEXT
1360 LINE(90,130)-(136,142),PRESET,BF
1370 REM
1380 REM MUSIC
1390 REM
1400 PLAY"GEDEFECDP20CDEFG"
1410 RUN
```

STAR TREKKER



DESCRIPTION

The captain is on holiday and war with the Klingons has just broken out in sector 3. So, off you go with the rest of the crew of the Star Ship Enterprise to fight the demonic Klingons. Your reactions must be fast, because the Klingons don't wait for anyone and have a disconcerting habit of exterminating sluggish star ships. In this war you will be struggling for survival.

RUNNING THE PROGRAM

When the game is run, it asks for the difficulty level: 1 is easy and 9 is virtually impossible for a beginner. After the message "ALL SYSTEMS GO" the screen will display a real time short range scan of the quadrant of the galaxy you are in. To the right of this is your information panel, showing fuel levels, current position, shield levels, etc. Below this, the message "SHIELD FAILURE" is displayed if the shield has been drained by continuous Klingon bombardment. The bottom part of the screen is reserved for your commands.

The following characters are used to represent the various objects displayed by the short range scanner:

'E' –The Enterprise.
'K' –A Klingon ship.
'#' –A starbase.
'*' –A star.
'.' –Enterprise torpedo fire.

The battle field is a galaxy 10 by 10 quadrants in size, each quadrant also being 10 by 10. The number of Klingons you have to destroy and the number of star bases for refuelling are set by the difficulty level at the start of the game. You have the following commands available:

F: Fire photon torpedoes. You will be told the number of torpedo banks that remain unfired and then asked for a direction (see below). This will be repeated until all banks have been fired or you type "S".

M: Move Enterprise. You are asked for a heading (see below).

L: The long range scanner is activated until another key is pressed. It shows the amount of stars, star bases and Klingons in the surrounding quadrants, under the heading "S", "B" and "K" respectively.

S: Asks for new value of shield power. If the number of Klingons is high, more shield power is required and this is deducted from the fuel level each second.

R: Repair damage to the Enterprise. Once started this cannot be stopped until the repairs have been completed. While you are undergoing repairs you are vulnerable to attack.

While commands that ask for more information are being typed, the display will freeze. When you press return on a line of data, for example a heading, the program will update the display to show what was happening while you were typing. When it has finished your command will be completed. So you may have been killed while you were typing and did not realise until you pressed return.

DIRECTIONS AND DISTANCES

A distance of0.1 will move you to the next position horizontally and 0.125 vertically. Also a distance of 1 will move you a quadrant width horizontally and 1.25 vertically. Headings are given in degrees, for instance 0 is down, 90 is right, -90 is left and 180 is going up. Any angle between these can be used.

```
      180
       |
-90 — * — 90
       |
       0
```

Helpful hint: Ifyou getswamped by Klingonstry shooting a star!

PROGRAM STRUCTURE

The lines of interest in this program are as follows:
130-200 Printsthe mission.
210-250 Inputs mission level.
660-910 Prints main scanner display.
950-1080 Displays the indicators.
1120-1250 Routine to input the command.
1290-1500 Indicator update.
1940-2390 Fire photon torpedoes.
2490-3080 Move the Enterprise.
3120-3230 Countthe number of Klingons left.

3270-3370	Mission failure.
3420-3650	Display the long range scanner.
3660-3760	Input new shield power.
3800-3970	Explode star.
4010-4350	Dock with star base.
4390-4470	Enterprise collides with Klingon or star.
4510-4640	Repair the Enterprise.
4680-4760	Play explosion.
4800-4870	Input for another go.

```
10 REM STAR TREK
20 REM *************************
30 REM
40 CLS
50 DIMG(9,9),Q(9,9),E(3)
60 PI=22/7
70 T5=255
80 GOSUB1510
90 E$="E"
100 AK$="K"
110 AS$="*"
120 AB$="#"
130 PRINT@43,"STAR TREK"
140 PRINTSTRING$(32,"=")
150 PRINT"YOU ARE THE CAPTAIN OF THE"
160 PRINT"STAR-SHIP ENTERPRISE."
170 PRINT"YOUR MISSION IS TO DESTROY THE"
180 PRINT"KLINGONS",
190 PRINT", BEFORE THEY DESTROY   'YOU'."
200 PRINT
210 INPUT"HOW DIFFICULT DO YOU WANT THE   MISSION (1-9) ";ID
220 IFID>9THEN PRINT:PRINT"THAT'S IMPOSSIBLE":GOTO210
230 IFID<1 THEN PRINT:PRINT"YOU ARE NOT PLAYING SNAP!":GOTO210
240 NB=INT(15-RND(ID))
250 NS=INT(600+RND(100))
260 CLS
270 PRINT
280 PRINT
290 PRINT
```

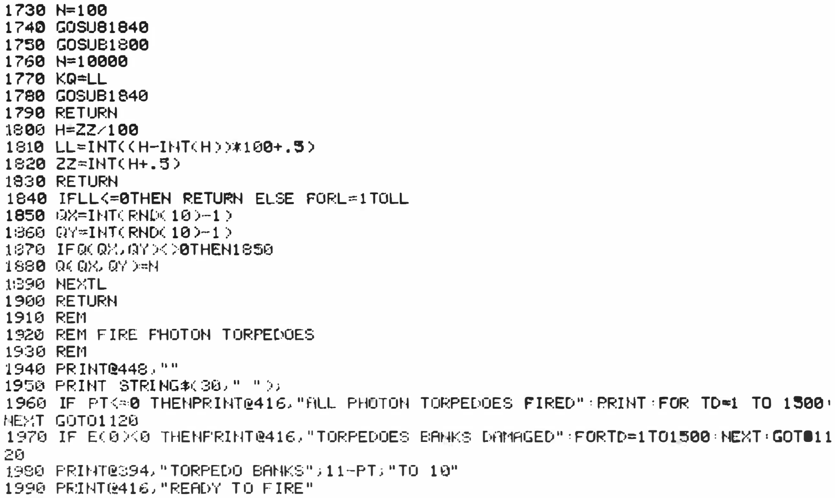
```
300 NK=INT(ID^3-RND(ID^2)+10)
310 PRINT"  ENTERPRISE ENTERING GALAXY"
320 FORI=1 TO NS
330 LX=INT(RND(10)-1)
340 LY=INT(RND(10)--1)
350 G(LX,LY)=G(LX,LY)+1
360 NEXTI
370 PRINT"WARNING: ";NK;"KLINGONS INVADING"
380 PRINT"GALAXY"
390 FORI=1 TO NK
400 LX=INT(RND(10)-1)
410 LY=INT(RND(10)--1)
420 G(LX,LY)=G(LX,LY)+10000
430 NEXT I
440 PRINTNB;"STARBASES TAKING UP"
450 PRINT"POSITIONS"
460 FORI=1 TO NB
470 LX=INT(RND(10)-1)
480 LY=INT(RND(10)-1)
490 G(LX,LY)=G(LX,LY)+100
500 NEXT I
510 PRINT
520 PRINT"        ALL SYSTEMS GO"
530 TIMER=0
540 DEFFNT(X)=INT(TIMER/50)
550 FL=10000
560 PT=10
570 SR=20
580 XE=RND(9)-1+RND(0)
```

```
590 YE=RND(9)-1+RND(0)
600 NB=844
610 GOSUB1580
620 CLS
630 REM
640 REM PRINT MAIN DISPLAY
650 REM
660 L$=STRING$(10," ")
670 PRINT@0,CHR$(142)
680 FOR K=1 TO 11
690 PRINT@K,CHR$(140)
700 NEXT K
710 FORK=32 TO 320 STEP32
720 PRINT@K,CHR$(138);L$;CHR$(138)
730 NEXT K
740 PRINT@352,CHR$(139)
750 FOR K=1 TO 11
760 PRINT@K+352,CHR$(131)
770 NEXTK
780 FORI=0TO9
790 FOR J=0TO9
800 NP=INT(33+I+32*J)
810 Q=INT(Q(I,J))
820 IFQ=0THEN880
830 IFQ=1THENPRINT@NP,AS$;:GOTO890
840 IFQ=100THENPRINT@NP,AB$;:GOTO890
850 IFQ=10000THENPRINT@NP,AK$;:GOTO890
860 IFQ=-1THENPRINT@NP,BH$;:GOTO890
870 IFQ=-10THENPRINT@NP,E$;:GOTO890
```

```
880 PRINT@NP," ";
890 NEXTJ,I
900 GOSUB950
910 GOTO1120
920 REM
930 REM DISPLAY INDICATORS
940 REM
950 PRINT@16,"CURRENT";
960 PRINT@48,"POSITION";
970 PRINT@80,USING"#.#    #.# ";XE,YE
980 JJ=80
990 PRINT@JJ+32,"ENERGY LEFT ";
1000 PRINT@JJ+64,USING"**####";FL;
1010 PRINT@JJ+96,"SHIELD POWER";
1020 PRINT@JJ+128,USING"**###";SR;
1030 PRINT@JJ+160,"PHOTON TORP.S";
1040 PRINT@JJ+192,USING"###";PT;
1050 T=FNT(X)
1060 PRINT@JJ+224,"MISSION TIME";
1070 PRINT@JJ+256,T;
1080 IF D>0 THENPRINT@JJ+288,"SHIELD FAILURE";:RETURN ELSE PRINT@JJ+288,"
        ";:RETURN
1090 REM
1100 REM INPUT COMMAND
1110 REM
1120 PRINT@384,"COMMAND ";CHR$(128)
1130 C$=INKEY$
1140 PRINT@416,""
1150 PRINT
```

```
1160 PRINTSTRING$(30," ");
1170 PRINT@384,"COMMAND >"
1180 IF C$<>"" THEN SOUND240,1 ELSE 1240
1190 IFC$="F"THEN1940
1200 IFC$="M"THEN2430
1210 IFC$="L"THEN3420
1220 IFC$="S"THEN3660
1230 IFC$="R"THEN4510
1240 GOSUB1290
1250 GOTO1120
1260 REM
1270 REM INDICATOR UPDATE
1280 REM
1290 IFFL>=0THENFL=FL-SR ELSE 3270
1300 HT=KQ*5*RND(0)
1310 D=INT(HT-SR)
1320 IF D>0 THEN I=INT(RND(4)-1):E(I)=E(I)-D:DT=DT+D: IF DT>100 THEN3320
1330 MX=INT(RND(10)-1)
1340 MY=INT(RND(10)-1)
1350 IF MX=IX AND MY=IY THEN1420
1360 IFG(MX,MY)<9900THEN1420
1370 G(MX,MY)=G(MX,MY)-10000
1380 M1=INT(SGN(IX-MX)*RND(ABS(IX-MX)))
1390 M2=INT(SGN(IY-MY)*RND(ABS(IY-MY)))
1400 G(MX+M1,MY+M2)=G(MX+M1,MY+M2)+10000
1410 IFMX+M1=IX AND MY+M2=IY THENLL=1:N=10000:GOSUB1840:DD$=AK$:GOSUB3350:KQ=KQ+
1
1420 IFQ(MX,MY)<>10000THEN1480 ELSE Q(MX,MY)=0:DD$=" ":QX=MX:QY=MY:GOSUB3350
1430 LL=1
```

```
1440 N=10000
1450 GOSUB1840
1460 GD$=AK$
1470 GOSUB3350
1480 GOSUB950
1490 IF E(2)<0 THEN SR=0
1500 RETURN
1510 E(0)=10
1520 E(1)=15
1530 E(2)=20
1540 E(3)=10
1550 DT=0
1560 RETURN
1570 REM
1580 NX=INT((XE-INT(XE))*10)
1590 NY=INT((YE-INT(YE))*10)
1600 FOR LA=0 TO 9
1610 FOR LB=0 TO 9
1620 O(LA,LB)=0
1630 NEXTLB,LA
1640 G=INT(G(INT(XE),INT(YE)))
1650 O(NX,NY)=-10
1660 IX=INT(XE)
1670 IY=INT(YE)
1680 ZZ=G
1690 GOSUB1800
1700 N=1
1710 GOSUB1840
1720 GOSUB1800
```



```
1730 N=100
1740 GOSUB1840
1750 GOSUB1800
1760 N=10000
1770 KQ=LL
1780 GOSUB1840
1790 RETURN
1800 H=ZZ/100
1810 LL=INT((H-INT(H))*100+.5)
1820 ZZ=INT(H+.5)
1830 RETURN
1840 IFLL<=0THEN RETURN ELSE FORL=1TOLL
1850 QX=INT(RND(10)-1)
1860 QY=INT(RND(10)-1)
1870 IFQ(QX,QY)<>0THEN1850
1880 Q(QX,QY)=N
1890 NEXTL
1900 RETURN
1910 REM
1920 REM FIRE PHOTON TORPEDOES
1930 REM
1940 PRINT@448,""
1950 PRINT STRING$(30," ");
1960 IF PT<=0 THENPRINT@416,"ALL PHOTON TORPEDOES FIRED":PRINT:FOR TD=1 TO 1500:
NEXT GOTO1120
1970 IF E(0)<0 THENPRINT@416,"TORPEDOES BANKS DAMAGED":FORTD=1TO1500:NEXT:GOTO11
20
1980 PRINT@394,"TORPEDO BANKS";11-PT;"TO 10"
1990 PRINT@416,"READY TO FIRE"
```

```
2000 PRINT@448,"DIRECTION (S=TO STOP)"
2010 TS=FNT(X)
2020 PRINT@470,"===>>";
2030 LINE INPUTC$
2040 IF C$="S" THEN 1120
2050 PT=PT-1
2060 FOR JZ=1 TO FNT(X)-TS
2070 GOSUB1290
2080 NEXT
2090 V=VAL(C$)
2100 XF=SIN(V/180*PI)
2110 YF=COS(V/180*PI)*.8
2120 TV=.14
2130 TV=TV+.95
2140 X5=NX+TV*XF
2150 Y5=NY+TV*YF
2160 X6=X5+.1
2170 Y6=Y5+.1
2180 IFX6<0 OR X6>9.999 OR Y6<0 OR Y6>9.999 THEN PRINT@448,"BANK";10-PT;"FIRED":
FOR ZS=1TO3000:NEXT:GOTO1940
2190 IF INT(X6)=NX AND INT(Y6)=NY THEN2130
2200 IFQ(X6,Y6)=0 THENDD$=".":QX=INT(X6):QY=INT(Y6):GOSUB3350:DD$=" ":GOSUB3350
2210 IFQ(X6,Y6)<=0THEN2130
2220 IFQ(X6,Y6)=100 THEN PRINT@448,"YOU HIT YOU SHOT A STARBASE":FORTD=1TO1500:N
EXT:GOTO2250
2230 IFQ(X6,Y6)=1 THEN PRINT@448,"TORPEDO HIT A STAR":FOR I=0TO1500:NEXT:IFRND(9
)<7 THENGOSUB4320:GOTO3300 ELSE 1940
2240 IFQ(X6,Y6)=10000THENK0=K0-1
2250 Q(X6,Y6)=0
```

```
2260 QX=INT(X6)
2270 QY=INT(Y6)
2280 DD$=CHR$(128)
2290 GOSUB4680
2300 GOSUB3350
2310 DD$="*"
2320 GOSUB4680
2330 GOSUB3350
2340 DD$=" "
2350 GOSUB4680
2360 GOSUB3350
2370 GOSUB3120
2380 IFG<10000THENCLS:PRINT@192,"WELL DONE! MISSION ACCOMPLISHED.":PRINT:PRINTTA
B(5)"ALL KLINGONS DESTROYED":GOTO4800:ELSEPRINT@416,"":PRINTINT(G/10000);"KLINGO
N(S) LEFT":FORTD=1TO1500:NEXT
2390 GOTO1940
2400 REM
2410 REM MOVE ENTERPRISE
2420 REM
2430 PRINT@416,""
2440 PRINT
2450 PRINT STRING$(30," ");
2460 IF E(1)<0 THEN PRINT@384,"ENGINES DAMAGED":FOR TD=1 TO 700:NEXT:GOSUB1290:I
F INKEY$<>""THEN1120 ELSE 2430
2470 TS=FNT(X)
2480 PRINT@384,;
2490 LINE INPUT"GIVE HEADING ++--->";C1$
2500 FORJZ=0TOFNT(X)-TS
2510 GOSUB1290
```

```
2520 NEXT
2530 TS=FNT(X)
2540 PRINT@416,;
2550 LINE INPUT"GIVE DISTANCE ++---->";C2$
2560 FORJZ=0TOFNT(X)-TS
2570 GOSUB1290
2580 NEXT
2590 IF INSTR(C1$+C2$,"S")<>0THEN1120
2600 Q(NX,NY)=0
2610 G=0
2620 FOR I=0 TO 9
2630 FOR J=0 TO 9
2640 G=G+Q(I,J)
2650 NEXTJ,I
2660 Q(NX,NY)=-10
2670 G(IX,IY)=G
2680 HA=VAL(C1$)
2690 HD=VAL(C2$)
2700 FL=FL-ABS(HD*50)
2710 XE=XE+SIN(HA*PI/180)*HD
2720 YE=YE+COS(HA*PI/180)*HD*.8
2730 KX=INT(XE)
2740 KY=INT(YE)
2750 Q(NX,NY)=0
2760 IF NOT (KX=IX AND KY=IY)THEN3040
2770 QX=NX
2780 QY=NY
2790 DD$=" "
2800 GOSUB3350
```

```
2810 NX=INT((XE-IX)*10)
2820 NY=INT((YE-IY)*10)
2830 IFQ(NX,NY)=100THENGOSUB1510:DD$=E$:QX=NX:QY=NY:GOSUB3350:Q(NX,NY)=-10:PT=10
:FL=FL+10000:GOTO4010
2840 IFQ(NX,NY)<>0THEN4390
2850 Q(NX,NY)=-10
2860 DD$=E$
2870 QX=NX
2880 QY=NY
2890 GOSUB3350
2900 GOTO1120
2910 PRINT@384,"YOU LEFT THE GALAXY."
2920 PRINT
2930 PRINT"RANDOM RE-ENTRY ACTIVATED"
2940 FORJK=1TO100STEP4
2950 SOUNDJK,1
2960 NEXT
2970 SOUND100,15
2980 FOR JK=100 TO 1 STEP-1
2990 SOUNDJK,1
3000 NEXT
3010 SOUND1,15
3020 XE=RND(9)-RND(0)
3030 YE=RND(9)-RND(0)
3040 IX=INT(XE)
3050 IY=INT(YE)
3060 IF IX>=0 AND IY>=0 AND IX<=9 AND IY<=9 THEN3070 ELSE 2910
3070 GOSUB1580
3080 GOTO780
```

```
3090 REM
3100 REM COUNT KLINGONS LEFT
3110 REM
3120 G=0
3130 Z=0
3140 Q(NX,NY)=0
3150 G(IX,IY)=0
3160 FOR I=0 TO 9
3170 FOR J=0 TO 9
3180 G=G+G(I,J)
3190 Z=Z+Q(I,J)
3200 NEXTJ,I
3210 G=G+Z
3220 Q(NX,NY)=-10
3230 RETURN
3240 REM
3250 REM MISSION FAILURE
3260 REM
3270 FL=0
3280 GOSUB8950
3290 PRINT@128,"OUT OF FUEL LIFE SUPPORT FAILED"
3300 PRINT@160,"YOU'RE DEAD !!!!!!!!....."
3310 GOTO4800
3320 GOSUB8950
3330 PRINT@128,"ENTERPRISE DESTROYED !!"
3340 GOTO4800
3350 NP=INT(33+QX+32*QY)
3360 PRINT@NP,DD$;
3370 RETURN
```

```
3380 REM
3390 REM DISPLAY LONG RANGE
3400 REM SCANNER
3410 REM
3420 IF E(3)<0THENPRINT@416,"SCANNER DAMAGED . . . . . . .":GOSUB1290:GOSUB1290:
GOSUB1290:GOTO1120 ELSE PRINT@384,"S   B   K     S   B   K     S   B   K"
3430 Q(NX,NY)=0
3440 G=0
3450 FOR IU=0 TO 9
3460 FOR IC=0 TO 9
3470 G=G+Q(IU,IC)
3480 NEXTIC,IU
3490 G(IX,IY)=G
3500 Q(NX,NY)=-10
3510 GOSUB1290
3520 C$=INKEY$
3530 IF C$<>"" THEN1170
3540 FORIU=-1TO1
3550 FOR IC=-1 TO 1
3560 IF IX+IU<0 OR IY+IC<0 OR IX+IU>9 OR IY+IC>9 THEN 3570 ELSE G$=STR$(G(IX+IU,
IY+IC)):LZ=LEN(G$):G$=RIGHT$(G$,LZ-1):G$=STRING$(7-LZ,"0")+G$:GOTO3590
3570 Z1=0
3580 G$="        "
3590 PRINT@416+11*(IU+1)+32*(IC+1),USING"%% %% %%";MID$(G$,5,2),MID$(G$,3,2),MID
$(G$,1,2);
3600 NEXT
3610 GOSUB1290
3620 NEXT
3630 GOSUB1290
```

```
3640 C$=INKEY$
3650 IF C$<>"" THEN1140 ELSE 3400
3660 PRINT@416,""
3670 PRINT
3680 PRINTSTRING$(30," ");
3690 PRINT@384,"NEW SHIELD POWER"
3700 TS=FNT(X)
3710 LINE INPUT C$
3720 FOR I=1 TO FNT(X)-TS
3730 GOSUB1290
3740 NEXTI
3750 SR=VAL(C$)
3760 GOTO1130
3770 REM
3780 REM STAR EXPLOSION
3790 REM
3800 CLS0
3810 PRINT@206,CHR$(137);
3820 PRINT@207,CHR$(134);
3830 PRINT@238,CHR$(134);
3840 PRINT@239,CHR$(137);
3850 FORI=1TO400
3860 NEXT
3870 CLS5
3880 FOR I=1 TO 100 NEXT
3890 CLS0
3900 GOSUB4080
3910 G(IX,IY)=0
3920 GOSUB1530
3930 GOSUB3120
```

```
3940 IFG<10000THENCLS:PRINT@192,"WELL DONE! MISSION ACCOMPLISHED.":PRINT:PRINTTA
B(5)"ALL KLINGONS DESTROYED":GOTO4800 ELSECLS:PRINT@196,INT(G/10000);"KLINGON(S)
 LEFT"
3950 FL=FL-5000
3960 SR=RND(SR)
3970 GOTO580
3980 REM
3990 REM DOCK WITH STARBASE
4000 REM
4010 CLS
4020 PRINT
4030 PRINT
4040 PRINT
4050 PRINT"ENTERPRISE DOCKED AT STARBASE"
4060 PRINT
4070 PRINT"FUEL BEING LOADED"
4080 PRINT
4090 PRINT"TORPEDO BANKS BEING REARMED"
4100 PRINT
4110 PRINT"REPAIRS COMPLETE"
4120 FOR I=0 TO 3000:NEXT
4130 MX=INT(RND(10)-1)
4140 MY=INT(RND(10)-1)
4150 G(MX,MY)=G(MX,MY)+100
4160 CLS
4170 PRINT@5,"STARBASE LEAVING"
4180 N9=360
4190 FOR I=1 TO 9
4200 PRINT@N9,"#==#";
```

```
4210 FOR J=0 TO 50-I
4220 NEXTJ
4230 PRINT@N9,"     ";
4240 N9=N9-31
4250 NEXTI
4260 FORI=1TO20
4270 PRINT@N9,"##";
4280 PRINT@N9,"   ";
4290 N9=N9+1
4300 NEXTI
4310 GOTO660
4320 QX=INT(X6)
4330 QY=INT(Y6)
4340 DD$=CHR$(143)
4350 GOTO3350
4360 REM
4370 REM HIT KLINGON OR STAR
4380 REM
4390 CLS
4400 PRINT
4410 PRINT
4420 PRINTTAB(3)"BANG ! ! ! ! ! ! !......"
4430 PRINT
4440 PRINT"I THINK YOU HIT SOMETHING BIG"
4450 PRINT"PS.  YOU'RE VERY DEAD'.'.'...."
4460 GOSUB4680
4470 GOTO3310
4480 REM
4490 REM REPAIR SHIP
```

```
4500 REM
4510 PRINT@416,""
4520 PRINT
4530 PRINTSTRING$(30," ");
4540 PRINT@384,"REPAIRS STARTED"
4550 FOR I=1 TO DT/5
4560 GOSUB1290
4570 NEXTI
4580 FL=FL-5*DT
4590 GOSUB1510
4600 PRINT@448,"THEY ARE NOW COMPLETED"
4610 GOSUB1290
4620 GOSUB1290
4630 GOSUB1290
4640 GOTO1120
4650 REM
4660 REM EXPLOSION
4670 REM
4680 FORX=1TO5
4690 PLAY"O1;T155;L255;V10;G"
4700 NEXT
4710 PLAY"O1;L255;V13;F;L255;V16;E;L255;V22;D;L255;V27;C;L255;V30;B;L255;A"
4720 FORX=1TO5
4730 PLAY"O1;T155;L255;P255;V31;G"
4740 NEXT
4750 PLAY"O1;T120;L255;P255;V30;A;T90;L255;P255;V27;B;T50;L255;P255;V22;C;T30;L2
55;P255;V18;D"
4760 RETURN
4770 REM
```

```
4780 REM INPUT ANOTHER GO
4790 REM
4800 FOR TD=1 TO 3000
4810 NEXT
4820 CLS
4830 PRINT@231,"ANOTHER GO ? (Y/N)"
4840 Q$=INKEY$:IF Q$=""THEN4840
4850 IF Q$<>"Y" AND Q$<>"N" THEN4840
4860 IF Q$="Y" THEN RUN
4870 CLS
```

Duckworth Home Computing

DRAGON PROGRAMS
Edited by Nick Hampshire

This book provides a collection of 20 exciting and interesting programs for the Dragon computer. There is a fascinating version of Star Trek as well as other arcade favourites such as Breakout, Rhino and Landmine. Games of logic like Gomoko, Supermind and Planet Ruler are also included, while Mindreader simulates artificial intelligence. Other programs demonstrate the best features of the Dragon: Piano – 3D Graph – Sound Effects – Kaleidoscope – Graph Plotting – Egg Timer – Birds Demo. A must for all Dragon users.

ISBN 0-7156-1801-6

9 780715 618011

Duckworth
The Old Piano Factory
43 Gloucester Crescent, London NW1

ISBN 0 1756 1801 6

IN UK ONLY £6.95 NET

DRAGON PROGRAMS

NICK HAMPSHIRE