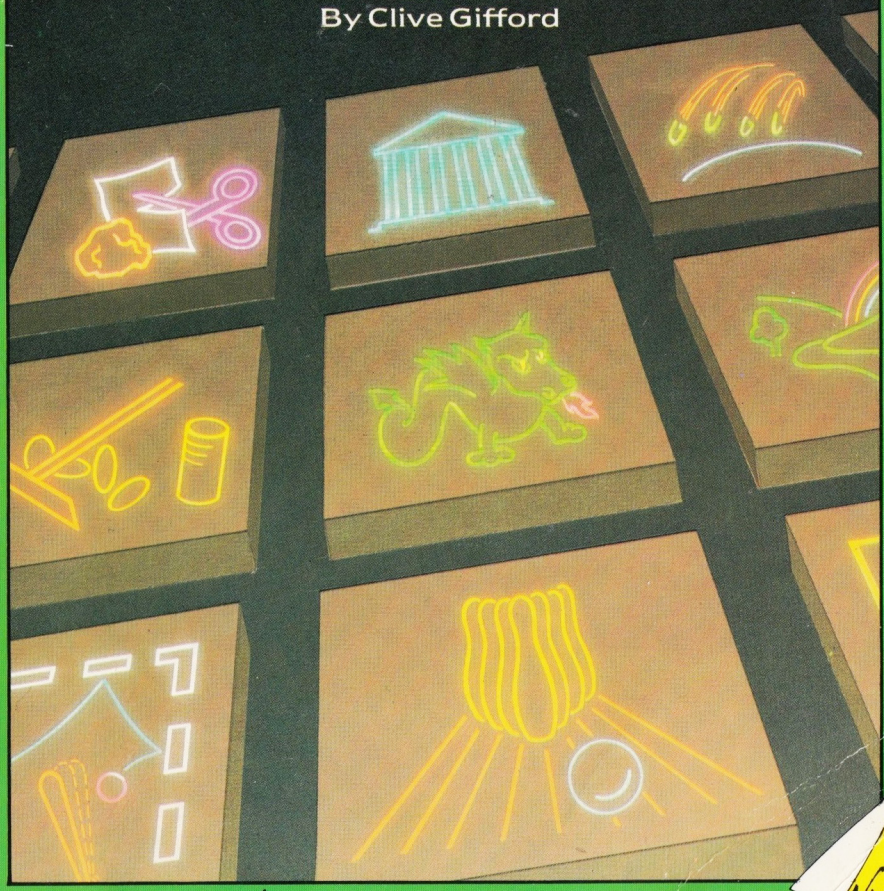
MORE GAMES FOR YOUR DRAGON 32

By Clive Gifford



The Virgin Computer Games Series

MORE GAMES FOR YOUR DRAGON 32

BY
CLIVE GIFFORD,
DAVID EDWARDS
AND
PHILLIP BROUGHTON

Virgin Books

First published in Great Britain in 1984 by Virgin Books Ltd, 61-63 Portobello Road, London W11 3DD.

ISBN 0 86369 033 5

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

Production services by Book Production Consultants, Cambridge.

Designed by Ray Hyden.

Illustrated by Sue Walliker.

Typeset by QV Typesetting.

Distributed by Arrow Books.

TO OUR PARENTS

TIM HARTNELL — THE SERIES EDITOR

Tim Hartnell is the most widely-published computer author in the world. Founder of the National ZX Users' Club, and founding editor of *ZX Computing* magazine, Tim has been involved over the years in a wide variety of computer activities. His published works include The Personal Computer Guide (Virgin Books) and The Giant Book of Computer Games (Fontana).

CLIVE GIFFORD — THE AUTHOR

Clive Gifford is a student planning to go to University this year to study Politics. He is the author of *Games for Your Dragon 32, Making the Most of Your Dragon 32, Creating Adventures for Your Dragon 32, Using Computers in Education, Dynamic Games for Your Oric* and co-author of *More Games for Your ZX81*. He also writes reviews and articles for a number of magazines, and in his spare time plays golf, hockey and listens to music

DAVID EDWARDS — THE AUTHOR

David Edwards, at just 13, is one of the youngest rising stars in the computer world. Despite his tender age, David has been in-volved with computers for two years. Among the machines he has worked with, apart from the Dragon, are the ZX81, the TRS 80 and the ZX Spectrum. His hobbies, when he is not at the keyboard, are drama and reading.

PHILLIP BROUGHTON — THE AUTHOR

Phillip Broughton is a 15-year-old studying for his 'O' levels. A relative newcomer to the computing field, Phillip has concentrated his efforts on the Dragon 32 from the very beginning. He has found that computers and computing take most of his spare time; any that is left is spent playing badminton.

SUE WALLIKER — THE ILLUSTRATOR

Sue Walliker is a freelance illustrator.

ACKNOWLEDGEMENTS

The authors wish to thank the following people for their friendly assistance in every way: Catherine, Richard, Craig, Dorothy, Paul, Susan and Helen

CONTENTS

Editor's Introduction

Typing in a computer program is like opening an unknown door. You do not know until you actually open the door — or, in our case, run the program — what experience is waiting for you. Of course, the sign on the door has given you some indication, but nothing can equal first-hand experience.

You do not know precisely what experiences are waiting for you in the great programs in this book. Of course, if the introduction says you're entering a space game, it's very likely the program won't play 'Guess My Number' when you get it up and running. But the listing rarely hints at the computer's game-playing strategy, or the screen display, or the fun that is waiting for you

This book has a number of unknown doors — doors leading into outer space and into the fiendish worlds of computer intelligence, wizards and Adventure.

We've provided the doors...and the keys. All you have to do to turn the lock is type in the program, and run it. Whatever you find behind each door, I guarantee you won't be disappointed.

Tim Hartnell
Series editor
London
March 1984

Authors' Introduction

The Dragon 32 has proven its worth as a reliable computer for all the family, as can be seen from the many months of occupying a high position in the top-selling computer charts.

We have tried to draw a balance between fast-action arcade games and more traditional games of thought and chance. There are a couple of displays using the Dragon's graphics commands in new ways.

We have unashamedly put much emphasis on the use of high resolution graphics for the games. We feel that the Dragon graphics should be more widely used in games, and in this book we have tried to do so.

We wish all Dragon owners enjoyment and success.

Clive Gifford, David Edwards, Phillip Broughton
Spelthorne
January 1984

METEOR

In this game, you pilot an orbitor ship suddenly driven into a terrible meteor storm. Your only hope is to go on as long as possible, dodging the meteors as they thunder past you. Your ship is moved to the right by the right arrow key; unless you hold this key down the ship will drift to the left. If you find yourself in a particularly difficult spot, press the space bar, and you can gain a split second's sanctuary. However, you only have ten of these movements and three lives, so watch your step. You will be given a score and a rating at the end of the game.

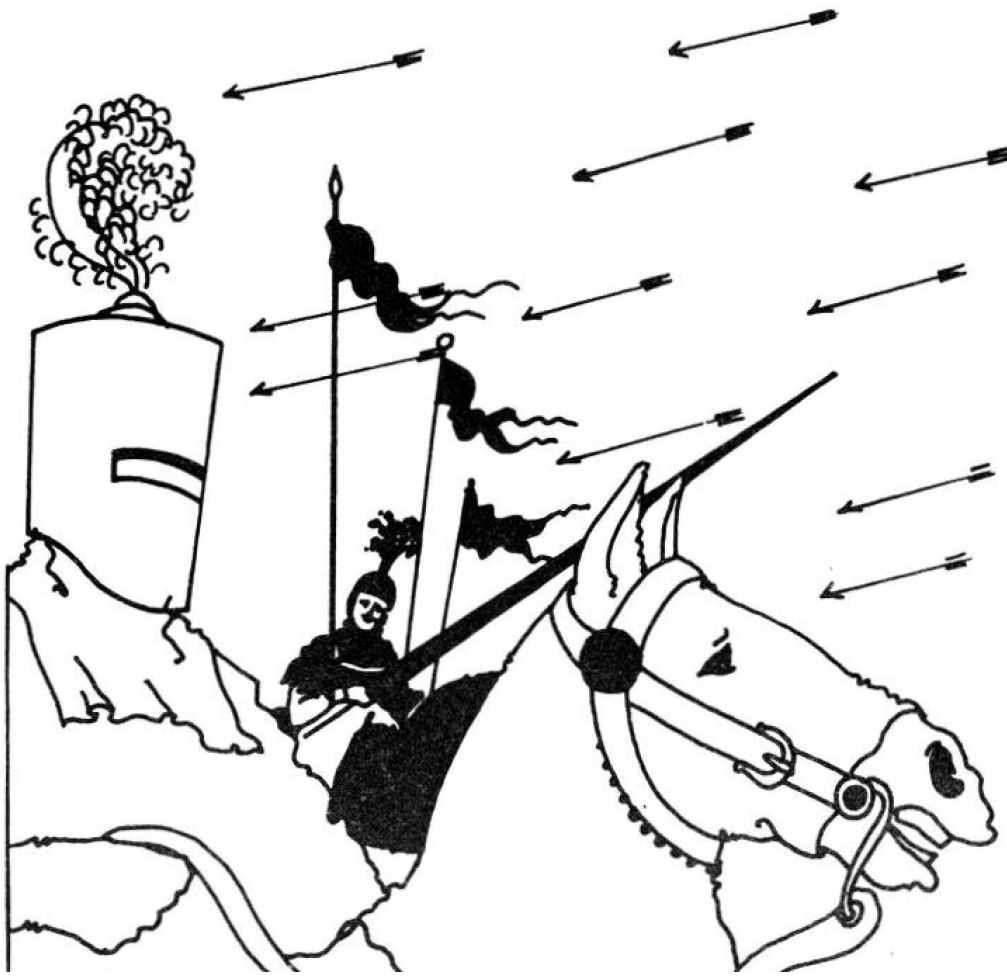
```
10 REM******METEOR******
20 GOTO 400
30 POKE 65495,0
40 X=100:L=3:P=0:F=10:Y=80:SC=0
50 A$="BM60,160;U30R30D15L30R30D15L30BR4
5U30R30D15L30R30D15BR20U30L5R30D30L30"
60 B$="BM45,160;U30R30L30D15R20BR10BF15U
30R30D15L30R30D15BR30U30BR30R30D15L30U15
D30U15R15F15"
70 C$="BM40,160;U30R30BD15L10R10D15L30BR
45U30R30D30L30BR45U30R30D30L30BR50U30L5R
30D30L30"
80 PMODE 4,1:PCLS:SCREEN 1,1
90 DRAW"BM40,40;U20F10E10D20BR10U20R20L2
0D10R15L15D10R20BR20U20L10R20BR10R20L20D
10R15L15D10R20BR10U20R20D20L20BR30U20R20
D10L20R10F10"
100 Y=100
```

```
110 REM***MAIN LOOP***
120 SC=SC+17
130 CIRCLE(X,Y),3,1,2:PAINT(X,Y),1,1
140 PLAY"O1L255GF"
150 P=P+1
160 IF PPOINT(X,Y+8)=5 OR PPOINT(X-5,Y)=
5 OR PPOINT(X+5,Y)=5 THEN 250
170 A=RND(210)+20
180 CIRCLE(A,170),RND(9)+4:PAINT(A,170),
1,1
190 CIRCLE(X,Y),3,0,2:PAINT(X,Y),0,0
200 FOR T=1 TO 5:EXEC 30101:NEXT T
210 IF PEEK(345)=223 AND P>5 AND F>0 AND
 Y>10 THEN Y=Y-10:F=F-1
220 IF PEEK(344)=223 THEN X=X+7:GOTO 240
230 IF X>20 THEN X=X-7
240 GOTO 110
250 FOR T=1 TO 10:SCREEN 1,0:PLAY"O1L255
DGF"
260 SCREEN 1,1:PLAY"O2BAC":NEXT
270 L=L-1:IF L=0 THEN 290
280 GOTO 80
290 REM***LOSE***
300 POKE 65494,0
310 PMODE 4,1:PCLS:SCREEN 1,1
320 PCLS:DRAW"BM2,170;C1U20R20BD10L5R5D1
0L20BR30U20R20D10L20R20D10BR10U20F10E10D
20BR10U20R20L20D10R15L15D10R20BR30U20R20
D20L20BR30BU20D10F10E10U10BR10R20L20D10R
15L15D10R20BR10U20R20D10L20R10F10"
```

```
330 FOR T=1 TO 100:EXEC 30101:NEXT T
340 PLAY"O2L20CDEFECDD"
350 IF SC<3500 THEN DRAW"XA$;":PLAY"O1L2
CDC"
360 IF SC>3500 AND SC<6000 THEN DRAW"XB$
;":PLAY"O2L4CDCDCDC"
370 IF SC>6000 THEN DRAW"XC$;":PLAY"O4L4
CDEFGABB"
380 FOR T=1 TO 500:NEXT T:CLS:PRINT @ 22
8,"SCORE IS ";SC
390 FOR T=1 TO 1000:NEXT T:PMODE 4,1:PCL
S:SCREEN 1,1:GOTO 340
400 REM****SCROLL****
410 CLEAR 300,30000
420 X=30000
430 DATA 8E,1D,DF,EC,83,ED,88,20,8C,06,1
0,24,F6,39
440 FOR D=1 TO 14:READ A$:POKE X+D,VAL("
&H"+A$):NEXT
450 DATA 8E,06,00,EC,88,20,ED,81,8C,1D,D
F,2F,F6,39
460 FOR D=1 TO 14:READ A$:POKE X+D+100,V
AL("&H"+A$):NEXT
470 GOTO 30
```

JOUST

The Black Knights of the Square Table in the medieval town of Dragonia are charging towards you on horseback. Once they are in range, fire your arrows at them. See how many you can kill before they charge at you again. Use the up and down cursor keys to manoeuvre yourself into position, then use the space bar to fire your arrows. You have a maximum of 40 arrows to fire, and this will determine your score at the end of the game. Good luck and try to stay on your horse.



```
10 AO=0:AP=0:AE=0:AY=0:SO=0:SP=0:SE=0:SY=
0:SC=0:J=0
20 CLS8:FORA=64TO415:PRINT@A,CHR$(128);:
NEXT
30 B2$=STRING$(32,240):PRINT@0,B2$;
40 PRINT@1,"joust";:PRINT@7,"joust";:PRI
NT@13,"joust";:PRINT@19,"joust";:PRINT@2
5,"joust";
50 M=254
60 B$=CHR$(128)+CHR$(128)
70 GOTO170
80 A$="<="
90 IF PEEK(342)=223 THEN M=M+32
100 IF PEEK(341)=223 THEN M=M-32
110 IF PEEK(345)=223 THEN GOSUB360
120 IF M>382 THEN M=382
130 IF M<126 THEN M=126
140 IF E=384 THEN PRINT@383,CHR$(128);:P
=128:O=192:Y=256:E=320:RETURN
150 PRINT@M+32,B$;:PRINT@M-32,B$;:PRINT@
M,A$;
160 RETURN
170 PRINT@128,"C";:PRINT@160,"H";:PRINT@
192,"A";:PRINT@224,"R";:PRINT@256,"G";:P
RINT@288,"E";:PRINT@320,"'";
180 PLAY""
190 PLAY"T20004C#CD#DE#EF#FG#GA#AB"
200 FORN=1TO3000:NEXT:C$=CHR$(128):PRINT
@128,C$;:PRINT@160,C$;:PRINT@192,C$;:PRI
NT@224,C$;:PRINT@256,C$;:PRINT@288,C$;:P
```

```
RINT@320,C$:210 P$="+":O$="+":Y$="+":E$="+"
220 P=128:O=192:Y=256:E=320
230 FORQ=1 TO2: IF AP=1 THEN GOTO 240 ELSE
 PRINT@P,P$;
240 IF AO=1 THEN GOTO 250 ELSEPRINT@O,O$;
250 NEXTQ
260 GOSUB80
270 PRINT@P,CHR$(128);:PRINT@O,CHR$(128);
:GOSUB290
280 P=P+1:O=O+1:E=E+1:Y=Y+1:GOTO230
290 GOSUB630
300 IF AY=1 THEN GOTO310 ELSE PRINT@Y,Y$;:
PRINT@Y-1,CHR$(128);
310 IF AE=1 THEN GOTO 320 ELSE PRINT@E,E$;:
PRINT@E-1,CHR$(128);
320 GOSUB330:RETURN
330 IF E=384 THEN PRINT@383,CHR$(128);:P
=128:O=192:Y=256:E=320:RETURN
340 IF AE=1 AND AY=1 AND AP=1 AND AO=1 THEN AE=0:
AY=0: AP=0: AO=0
350 RETURN
360 F=M-1
370 J=J+1: IF J=40 THEN GOTO550
380 F=F-1: GOSUB400: PRINT@F,"_";: PRINT@F+1,
CHR$(128);: IF F=106 OR F=138 OR F=170 OR F=202 OR
F=234 OR F=266 OR F=298 ORF=330 OR F=362 OR F=394
THEN PRINT@F,CHR$(128) ELSE GOTO380
```

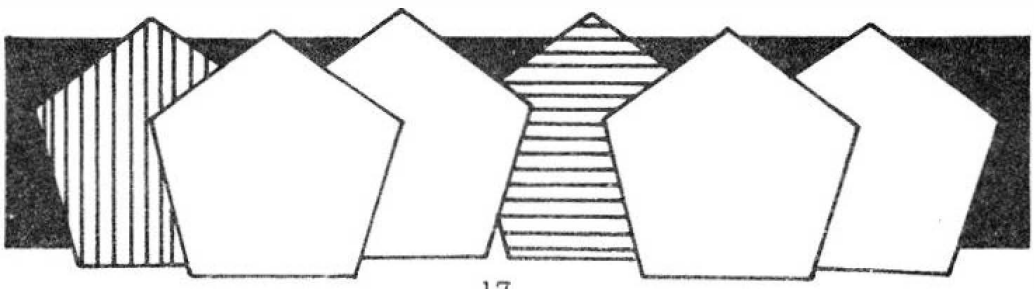
```
390 RETURN
400 IF F=P THEN AP=1:GOSUB590:IF SP=1 TH
EN RETURN ELSE GOSUB460:RETURN
410 IF F=O THEN AO=1:GOSUB600:IF SO=1 TH
EN RETURN ELSE GOSUB460:RETURN
420 IF F=Y THEN AY=1:GOSUB610:IF SY=1 TH
EN RETURN ELSE GOSUB460:RETURN
430 IF F=E THEN AE=1:GOSUB620:IF SE=1 TH
EN RETURN ELSE GOSUB460:RETURN
440 RETURN
450 GOTO230
460 PRINT@448,"YOUR SCORE";:PRINT@466,"H
IGH SCORE";
470 SC=SC+1:PRINT@458,SC;:IF SC>HI THEN
HI=SC:PRINT@476,HI;
480 PRINT@476,HI;
490 IF AE=1 THEN SE=1
500 IF AY=1 THEN SY=1
510 IF AO=1 THEN SO=1
520 IF AP=1 THEN SP=1
530 IF AE=1 AND AO=1 AND AP=1 AND AY=1 T
HEN SE=0:SO=0:SY=0:SP=0
540 RETURN
550 PRINT@230,"THY GAME HATH ENDED";
560 PRINT@198,"====================";
570 PRINT@262,"====================";
580 GOTO650
590 P$=CHR$(192):RETURN
600 O$=CHR$(192):RETURN
610 Y$=CHR$(192):RETURN
```

```
620 E$=CHR$(192):RETURN
630 B1$=CHR$(192):IF O$=B1$ AND P$=B1$ A
ND Y$=B1$ AND E$=B1$ THEN P=128:O=192:Y=
256:E=320:C$="+":P$=C$:O$=C$:Y$=C$:E$=C$
:C$=""
640 RETURN
650 FORI=1TO2500:NEXT:CLS
660 PRINT@100,"YOU SCORED";SC;"OUT OF";J
;
670 IF SC>35 THEN RA$="BRILLIANT" ELSE I
F SC>30 THEN RA$="GOODISH" ELSE IF SC>25
 THEN RA$="AVERAGE" ELSE IF SC>20 THEN R
A$="NEED  PRACTISE" ELSE IF SC>15 THEN R
A$="HOPELESS" ELSE IF SC<15 THEN RA$="TR
YING TO MISS THEM ARE WE?"
680 PRINT@164,"RATING=
                ";RA$
690 PRINT@388,"HIGHEST SCORE TODAY=";HI
700 PRINT@482,"PRESS ANY KEY TO START AG
AIN"
710 A$=INKEY$:IF A$="" THEN 710
720 FORI=1TO31:PLAY"V"+STR$(I)+"T200O3AB
O5AB04AB":NEXT
730 GOTO10
```

HEX-IT

This program draws a hexagon on the screen, fills it in and then blanks it out.

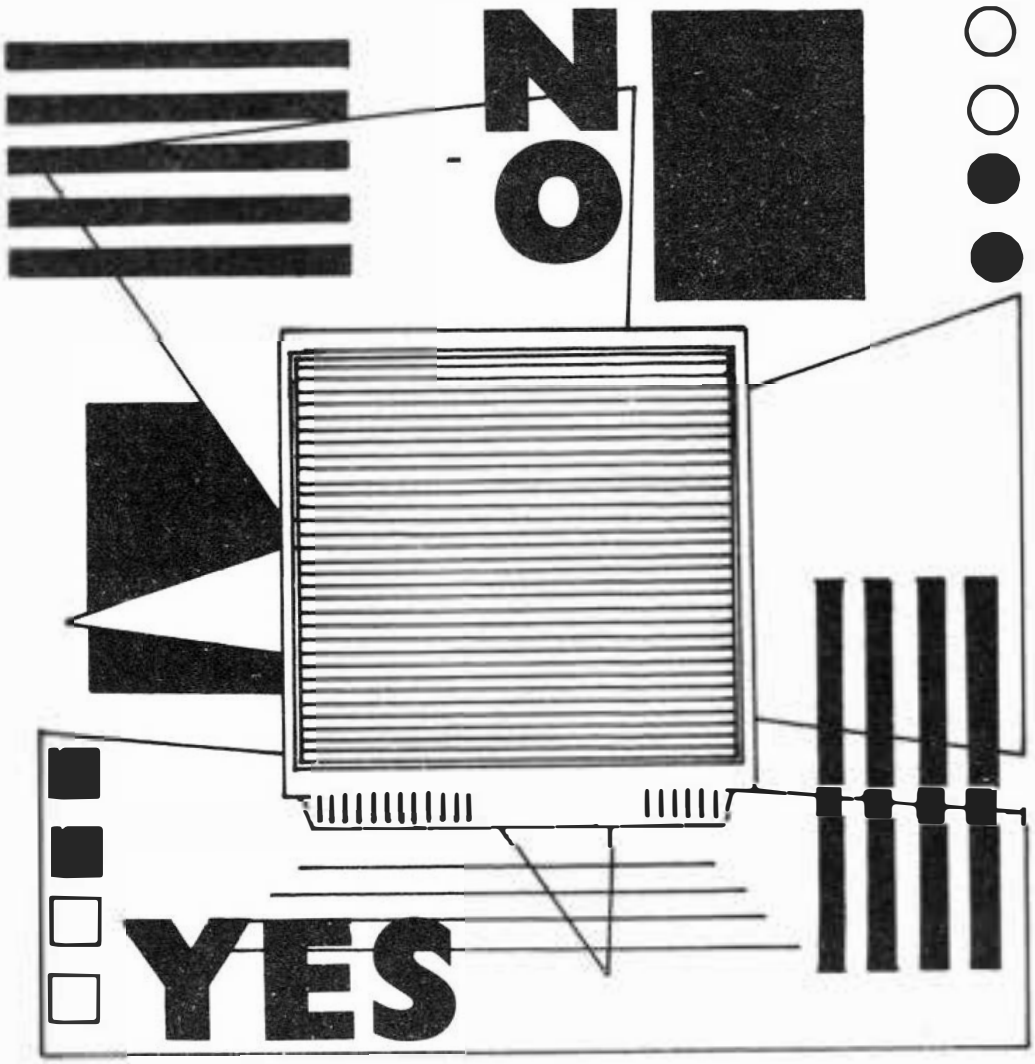
```
10 PMODE 1,1:SCREEN1,0:PCLS
11 A=0.5
12 COLOR RND(3)+1
20 FOR I=0 TO 1000
30 X=X+L*SIN(R):Y=Y+L*COS(R)
40 IF X<-128 OR X>128 THEN 90
50 IF Y<-96 OR Y>96 THEN 90
60 LINE -(X+128,Y+96),PSET
70 R1=R1+60:R=R1/57.29578:L=L+A
80 NEXT
90 R1=0:R=0:X=0:Y=0:L=0
91 PLAY"ABGCE"
95 FOR I=0 TO 1000
100 X=X+L*SIN(R):Y=Y+L*COS(R)
110 IF X<-128 OR X>128 THEN 160
120 IF Y<-96 OR Y>96 THEN 160
130 LINE-(X+128,Y+96),PRESET
140 R1=R1+60:R=R1/57.29578:L=L+A
150 NEXT
160 FOR N=0 TO 100:NEXT:RUN
```



PREDICT

This program measures recent changes in your life and the effects those changes have on you. Any change in your life, be it good or bad, creates measurable stress in your life.

A number of these situations will be printed on the screen. If the question is true to your life, then answer 'Y' for yes, if not then press 'N' for no. At the end of the game the computer will give you an assessment.



```
20 REM***PREDICT***
30 CLS
40 PRINT " LET MYSTIC THE FORTUNETELLER/
   COMPUTER,MIND READER AND ALL-   ROUND
 BIG HEAD TELL YOUR          FORTUNE"
50 PRINT" PRESS (Y/N) TO ANSWER THE
  QUESTIONS"
60 L=0
65 INPUT" ENTER YOUR NAME HOPEFUL ONE";
T$
70 GOTO 150
80 PRINT:T$;"   ";L$"?
                "
90 :INPUT Q$:IF Q$<>"Y" AND Q$<>"N" THEN
 90
100 IF Q$="Y" THEN L=L+VAL (MID$(S$,I*2-
1,2))
110 I=I+1
120 FOR K=1 TO 10
130 NEXT K
140 RETURN
150 I=1
160 S$="997371656363535047454544404039393
938373635333130292929292826262524232120
202019191817161515131211"
170 L$="IS YOUR HUSBAND/WIFE DEAD"
180 GOSUB 80
190 L$="ARE YOU DIVORCED"
200 GOSUB 80
210 L$="ARE YOU GIVING UP DOPE/DRUGS"
```

```
220 GOSUB 80
230 L$="IS YOUR MARRIAGE ON THE ROCKS"
240 GOSUB 80
250 L$="ARE YOU BEHIND THE SLAMMER"
260 GOSUB 80
270 L$="HAS ONE OF YOUR RELATIONS
KICKED THE BUCKET LATELY"
280 GOSUB 80
290 L$=" HAVE YOU BEEN INJURED LATELY"
300 GOSUB 80
310 L$=" ARE YOU MARRIED"
320 GOSUB 80
330 L$=" ARE YOU UNEMPLOYED"
340 GOSUB 80
350 L$=" HAVE YOU ANY CHILDREN"
360 GOSUB 80
370 L$=" ARE YOU RETIRED"
380 GOSUB 80
390 L$="IS YOUR FAMILY IN GOOD HEALTH"
400 GOSUB 80
410 L$="ARE YOU PREGNANT"
420 GOSUB 80
430 L$="ARE YOU QUITTING SMOKING"
440 GOSUB 80
450 L$=" HAVE YOU ANY SEX PROBLEMS"
460 GOSUB 80
470 L$=" HAS THERE BEEN ANY BIRTHS IN
 YOUR FAMILY"
480 GOSUB 80
490 L$=" HAS THERE BEEN A BUSINESS RE-AJ
```

```
USTMENT AT WORK"
500 GOSUB 80
510 L$=" ARE YOU BROKE"
520 GOSUB 80
530 L$="HAS ANY CLOSE FRIENDS DIED     L
ATELY"
540 GOSUB 80
550 L$="HAS YOUR JOB CHANGED"
560 GOSUB 80
570 L$="DO YOU ARGUE WITH YOUR HUSBAND/
WIFE"
580 GOSUB 80
590 L$=" ARE YOU SUFFERING FROM PRE-
 MENSTRIAL TENSION"
600 GOSUB 80
610 L$=" HAVE YOU HAD A FORECLOSURE OF Y
OUR MORTAGE OR LOAN"
620 GOSUB 80
630 L$=" IS YOUR MORTAGE OVER $20,000"
640 GOSUB 80
650 L$=" HAS THERE BEEN A CHANGE IN YOUR
 WORK RESPONSIBILITY"
660 GOSUB 80
670 L$=" ARE YOU SUFFERING FROM JET LAG"
680 GOSUB 80
690 L$="ARE YOUR CHILDREN LEAVING HOME"
700 GOSUB 80
710 L$="ARE YOUR IN-LAWS A PAIN IN THE
NECK"
720 GOSUB 80
```

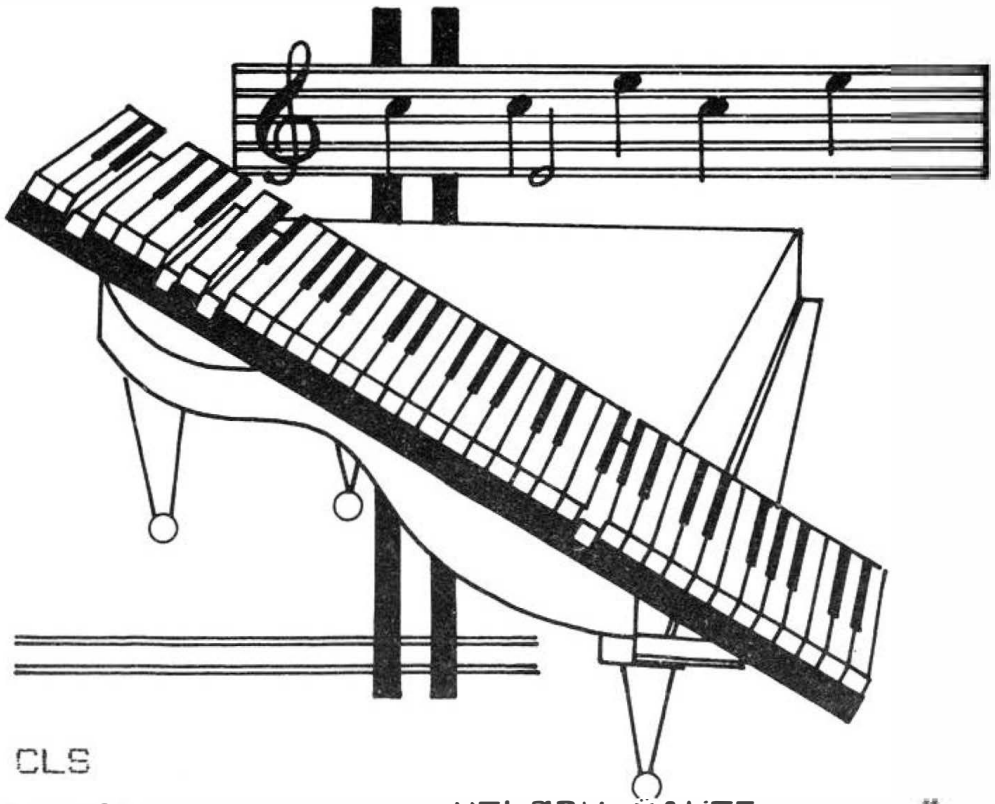
```
730 L$=" HAVE YOU REACHED ANY
 OUTSTANDING PERSONAL                ACHIEVE
MENTS"
740 GOSUB 80
750 L$=" IS YOUR WIFE/HUBBY STARTING OR
STOPPING WORK"
760 GOSUB 80
770 L$=" ARE YOUR CHILDREN START OR STOP
 SCHOOL"
780 GOSUB 80
790 L$=" HAVE YOU MOVED HOUSE LATELY"
800 GOSUB 80
810 L$=" HAVE YOU REVISED YOUR PERSONAL
 HABITS"
820 GOSUB 80
830 L$=" ARE YOU IN-TROUBLE WITH YOUR
 BOSS
840 GOSUB 80
850 L$=" ARE YOU GIVING UP SMOKING AGAIN
"
860 GOSUB 80
870 L$=" HAS THERE BEEN A CHANGE IN YOUR
 WORK HOURS AND/OR                   WORK CO
NDITIONS"
880 GOSUB 80
890 L$=" HAVE YOU CHANGED YOUR
 RESIDENCE"
900 GOSUB 80
910 L$=" HAVE YOU CHANGED YOUR  SCHOOL"
920 GOSUB 80
```

```
930 L$=" HAVE YOU CHANGED  YOUR  HOBBIES
"
940 GOSUB 80
950 L$=" ARE YOU CHANGING  YOUR  CHURCH
ACTIVITIES"
960 GOSUB 80
970 L$=" ARE YOU CHANGING YOUR  SOCIAL
 ACTIVITIES"
980 GOSUB 80
990 L$=" IS YOUR MORTGAGE OR LOAN UNDER
$20,000"
1000 GOSUB 80
1010 L$=" HAVE YOUR SLEEPING HABITS
  CHANGED"
1020 GOSUB 80
1030 L$=" IS THERE A CHANGE IN  THE NO.
  OF FAMILY GET TOGETHERS"
1040 GOSUB 80
1050 L$=" HAVE YOUR EATING HABITS CHANGE
D"
1060 GOSUB 80
1070 L$=" ARE YOU GOING ON HOLIDAY"
1080 GOSUB 80
1090 L$="IS IT CHRISTMAS"
1100 GOSUB 80
1110 L$=" ARE YOU VIOLATING THE LAW"
1120 GOSUB 80
1124 CLS
1125 PRINT :PRINT:PRINT:PRINT@140,  T$;
1130 PRINT:PRINT:PRINT:PRINT"   YOUR TOT
```

```
AL IS;";L
1140 PRINT"   the verdict"
1150 IF L<150 THEN PRINT"YOU LEAD A STAB
LE ,SAFE WAY OF  LIVING AND YOU ARE LESS
 LIKELY  THAN AVERAGE TO HAVE AN ACCIDEN
TOR BECOME ILL.":PRINT"UNLESS, YOU DIE O
F BOREDOM"
1160 IF L>149 AND L<200 THEN PRINT"THERE
 IS A 37% CHANCE OF YOU    HAVING HEALTH
 OR SAFETY PROBLEMS"
1170 IF L>199 AND L<300THEN PRINT"CALM D
OWN, AT THIS RATE YOU'LL NOT LIVE ANOTHE
R DAY"
1180 IF L>299  THEN PRINT" stop all work
 now***SIT DOWN   AND WAIT UNTIL YOUR SC
ORE GOES  DOWN (R.I.P)"
1190 PRINT"ANY MORE CLIENTS"
1200 INPUT A$
1210 IF A$="N" THEN STOP
1220 CLS
1230 RUN
```

MELODY MAKER

If you fancy yourself as a composer and want to beat Brahms or McCartney at their own game, then this will be the perfect program for you. Input your composition as the program asks and then listen to it until you are certain you have a Number One hit on your hands. If in any doubt about writing music on the Dragon refer to Chapter Nine of the Dragon manual.



```
10 CLS
20 PRINT"                MELODY MAKER                "
30 PRINT:PRINT"                    INSTRUCTIONS"
40 PRINT:PRINT" THE LETTERS A-G REPRESEN
T NOTESTHE LETTER 'O' MEANS CHANGE TO  A
 DIFFERENT OCTAVE FROM 1-5"
```

```
50 PRINT"THE LETTER 'P' MEANS PAUSE FOR
 A CERTAIN LENGTH OF TIME RANGINGFROM 1-
255"
60 PRINT"THE LETTER 'L' MEANS LENGTH OF
 NOTE RANGING FROM 1-255          THE LET
TER 'V' MEANS THE VOLUME OF THE NOTE RAN
GING FROM 1-31"
70 GOSUB400
80 CLS:PRINT"    START  COMPOSING  MOZAR
T"
90 PRINT@160," INPUT YOUR COMPOSITION":P
RINT:PRINT" ";:LINE INPUT B$
100 IF B$="" THEN 90
110 PRINT:PRINT" FANTASTIC TUNE YOU'VE G
OT THEREDO YOU WANT TO LISTEN TO IT(Y/N)
"
120 GOSUB410
130 IF A$="Y" THEN GOTO190
140 IF A$="N" THEN SOUND3,2:SOUND2,3:GOT
O160
150 GOTO 120
160 PRINT"  NOT THAT GOOD ARE WE. O.K TH
ENIF YOU WANT ANOTHER GO AT IT    PRESS
'Y' OR 'N' TO END"
170 GOSUB410:IF A$="Y" THEN RUN ELSE IF
A$="N" THEN CLS:END
180 GOTO170
190 CLS:PRINT"EXCUSE ME WHILE I PUT SOME
      COTTON WOOL IN MY EARS"
200 FORI=1TO2000:NEXT:PRINT@288," HOW MA
```

```
NY TIMES DO YOU WANT TO  HEAR YOUR COMPO
SITION? (1-5)":PRINT@384,"":PRINT@416,""
210 GOSUB410
220 IF A$="1" THEN P=1:GOTO280
230 IF A$="2" THEN P=2:GOTO280
240 IF A$="3" THEN P=3:GOTO280
250 IF A$="4" THEN P=4:GOTO280
260 IF A$="5" THEN P=5:GOTO280
270 GOTO210
280 ON P GOSUB 350,360,370,380,390
290 PRINT@384," THAT WAS NOT BAD I SUPPO
SE"
300 PRINT"DO YOU WANT TO HEAR IT AGAIN?Y
/N"
310 GOSUB410
320 IF A$="Y" THEN 200
330 IF A$="N" THEN RUN
340 GOTO310
350 PLAY B$:RETURN
360 PLAY B$:PLAY B$:RETURN
370 FORI=1TO3:PLAY B$:NEXT:RETURN
380 FORI=1TO4:PLAY B$:NEXT:RETURN
390 FORI=1TO5:PLAY B$:NEXT:RETURN
400 PRINT@489,"PRESS ANY KEY";
410 A$=INKEY$:IF A$="" THEN 410 ELSE RET
URN
```

TEN PIN

Ten pin bowling alleys are very crowded and expensive places. Now you have your own bowling alley in the comfort of your home. The game requires skill and concentration to knock down all ten pins, with only two attempts per frame. Hit the space bar to bowl and the man will bowl at his present position. After ten frames your score is shown as a percentage.

```
10 PCLEAR5
20 R=1
30 DIM P(15,24),B(25,20),U(25,20),Q(25,2
0),L(2),A(2),F(15,24)
40 L$=STRING$(32,134)
50 PCLS:PMODE4:IF R=1 THEN GOTO60 ELSE G
OTO 100
60 CLS:PRINT:PRINT:PRINT@109,"*ten*"
70 PRINT@205,"*pin*":PRINT@299,"*bowling
*"
80 PRINT@0,L$;:PRINTL$:PRINT@352,L$;:PRI
NTL$
90 FORI=9TO28STEP2:PLAY"T255V"+STR$(I)+"
O4C#CD#DE#EF#FG#GA#AB#AG#GF#FE#ED#DC#C"
:NEXT:FORI=31TO1STEP-1:PLAY"V"+STR$(I)+"
CDEFGAB":NEXT
100 R=0:CLS:CIRCLE(5,5),3,1,1,.3,.2:DRAW
"BM4,7;D2G1D3G1D3F1D3R4U3E1U3H1U3H1U2":P
AINT(5,5):PAINT(5,9)
110 LINE(4,22)-(6,22),PSET
```

```
120 GET(0,0)-(25,24),P
130 PCLS:DRAW"BM1,20;U2R5U1L5R5E8G8D1E3R
5D5L1U5L5E5F4R2U1L2H4G5E7R3U1L1R1U1R1H1U
1L1U1L3D1R3L4D3R1U4R1D2"
140 GET(0,0)-(25,20),B
150 PCLS:LINE(5,20)-(7,14),PSET:LINE-(10
,20),PSET:DRAW"BM7,14;U5F3R1L1H3U1R3U1L1
R1U1R1H1U1L1U1L3D1R3L4D3R1U4R1D2"
160 GET(0,0)-(25,20),U:PCLS:CIRCLE(5,5),
4:PAINT(5,5):GET(0,0)-(9,9),L,G:PCLS:SCR
EEN1,1
170 PUT(180,89)-(205,114),P
180 PUT(195,105)-(220,129),P:PUT(195,80)
-(220,105),P
190 PUT(210,89)-(235,114),P:PUT(210,65)-
(235,90),P
200 PUT(210,115)-(235,140),P
210 PUT(230,130)-(255,154),P
220 PUT(230,105)-(255,129),P
230 PUT(230,80)-(255,105),P:PUT(230,56)-
(255,80),P
240 LINE(0,55)-(255,55),PSET:LINE(0,155)
-(255,155),PSET
250 LINE(0,45)-(255,45),PSET:LINE(0,165)
-(255,165),PSET
260 FORY=59TO131STEP16:PUT(0,Y)-(25,Y+20
),U:GOSUB270:FORG=1TO100:NEXTG:PUT(0,Y)-
(25,Y+20),Q:NEXT:GOTO260
270 A$=INKEY$:IF A$=CHR$(32) THEN PUT(0,
Y)-(25,Y+20),B:GOSUB290:BO=BO+1:IF BO=2
```

```
THEN GOTO760:RETURN
280 RETURN
290 IF Y=59 THEN  GOSUB350
300 IF Y=75 THEN  GOSUB390
310 IF Y=91 THEN  GOSUB430
320 IF Y=107 THEN  GOSUB480
330 IF Y=123 THEN  GOSUB520
340 RETURN
350 Y=Y+5:FORX=25TO199:PUT(X,Y)-(X+9,Y+9
),L,PSET:NEXT:PUT(X,Y)-(X+9,Y+9),A,PSET:
N=RND(3):ON N GOTO 360,370,380
360 GOSUB640:GOSUB560:GOSUB580:PLAY"O3T2
00V25CABBAGE":Y=Y-5:RETURN
370 FORI=1TO2:ON I GOSUB 640,560:NEXT:PL
AY"O3V25T200CABBAGE":Y=Y-5:RETURN
380 GOSUB560:PLAY"O3T200V25CABBAGE":Y=Y-
5:RETURN
390 FORX=25TO183:PUT(X,Y)-(X+9,Y+9),L,PS
ET:NEXT:PUT(X,Y)-(X+9,Y+9),A,PSET:N=RND(
3):ON N GOTO 400,410,420
400 FORI=1TO5:ON I GOSUB 700,660,640,580
,560:NEXT:PLAY"O3V25T200CABBAGE":RETURN
410 FORI=1TO6:ON I GOSUB 700,660,640,600
,580,560:NEXT:PLAY"O3V25T200CABBAGE":RET
URN
420 FORI=1TO5:ON I GOSUB 700,660,640,600
,580:NEXT:PLAY"O3V25T200CABBAGE":RETURN
430 Y=Y+4:FORX=25TO168:PUT(X,Y)-(X+9,Y+9
),L,PSET:NEXT:PUT(X,Y)-(X+9,Y+9),A,PSET:
N=RND(4):ON N GOTO 440,450,460,470
```

```
440 FOR I=1TO10:ON I GOSUB 740,720,700,6
80,660,640,620,600,580,560:NEXT:PLAY"O3V
25T200CABBAGE":Y=Y-4:RETURN
450 FORI=1TO9:ON I GOSUB 740,720,700,680
,660,640,600,580,560:NEXT:PLAY"O3V25T200
CABBAGE":Y=Y-4:RETURN
460 FORI=1TO9:ON I GOSUB 740,720,700,690
,660,640,620,600,580:NEXT:PLAY"O3V25T200
CABBAGE":Y=Y-4:RETURN
470 FORI=1TO8:ON I GOSUB740,720,700,680,
660,640,600,580:NEXT:PLAY"O3T200CABBAGE"
:Y=Y-4:RETURN
480 Y=Y+5:FORX=25TO183:PUT(X,Y)-(X+9,Y+9
),L,PSET:NEXT:PUT(X,Y)-(X+9,Y+9),A,PSET:
N=RND(3):ON N GOTO 490,500,510
490 FORI=1TO6:ON I GOSUB 720,680,660,620
,600,580:NEXT:PLAY"O3V25T200CABBAGE":Y=Y
-5:RETURN
500 FORI=1TO5:ON I GOSUB 720,680,660,600
,580:NEXT:PLAY"O3V25T200CABBAGE":Y=Y-5:R
ETURN
510 FORI=1TO5:ON I GOSUB 720,680,660,620
,600:NEXT:PLAY"T200V25O3CABBAGE":Y=Y-5:R
ETURN
520 Y=Y+5:FORX=25TO198:PUT(X,Y)-(X+9,Y+9
),L,PSET:NEXT:PUT(X,Y)-(X+9,Y+9),A,PSET:
N=RND(3):ON N GOTO 530,540,550
530 FORI=1TO3:ON I GOSUB680,620,600:NEXT
:PLAY"O3V25T200CABBAGE":Y=Y-5:RETURN
540 FORI=1TO2:ON I GOSUB 680,620:NEXT:PL
```

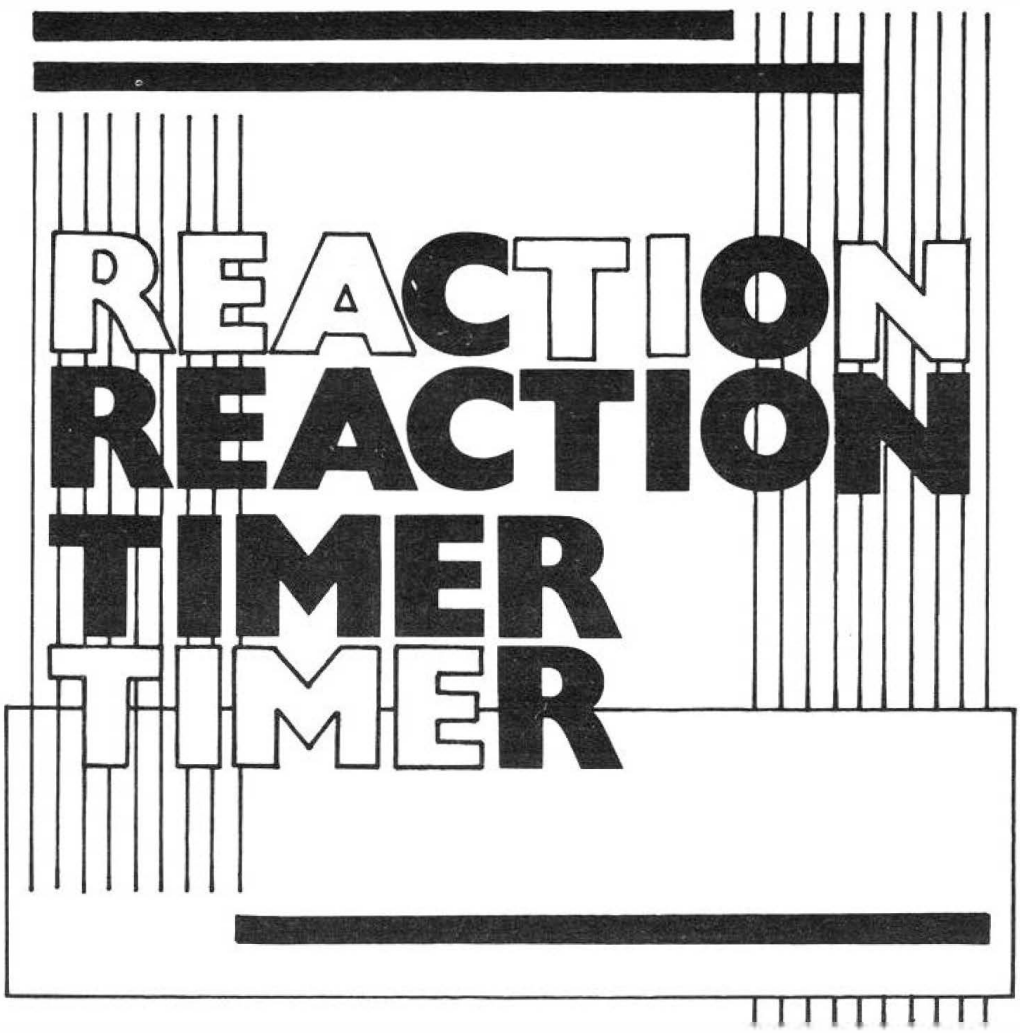
```
AY"T200V25O3CABBAGE":Y=Y-5:RETURN
550 GOSUB620:PLAY"T200V25O3CABBAGE":Y=Y-
5:RETURN
560 IF P1=0 THEN SC=SC+1:P1=1:PUT(230,56
)-(240,80),F:RETURN
570 RETURN
590 IF P2=0 THEN SC=SC+1:P2=1:PUT(230,80
)-(240,105),F:RETURN
590 RETURN
600 IF P3=0 THEN SC=SC+1:P3=1:PUT(230,10
5)-(255,129),F:RETURN
610 RETURN
620 IF P4=0 THEN SC=SC+1:P4=1:PUT(230,13
0)-(255,154),F:RETURN
630 RETURN
640 IF P5=0 THEN SC=SC+1:P5=1:PUT(210,65
)-(220,90),F:RETURN
650 RETURN
660 IF P6=0 THEN SC=SC+1:P6=1:PUT(210,89
)-(218,114),F:RETURN
670 RETURN
690 IF P7=0 THEN SC=SC+1:P7=1:PUT(210,11
5)-(223,140),F:RETURN
690 RETURN
700 IF P8=0 THEN SC=SC+1:P8=1:PUT(195,80
)-(220,105),F:RETURN
710 RETURN
720 IF P9=0 THEN SC=SC+1:P9=1:PUT(195,10
5)-(220,129),F:RETURN
730 RETURN
```

```
740 IF PL=0 THEN SC=SC+1:PL=1:PUT(190,89
)-(205,114),F:RETURN
750 RETURN
760 CLS:FR=FR+1:IF FR=10 THEN GOTO780
770 PRINTL$+L$+L$:PRINT"       FRAME=";FR
:PRINTL$:PRINT"       SCORE=";SC:PRINTL$:
FORI=1TO3000:NEXT
780 P1=0:P2=0:P3=0:P4=0:P5=0:P6=0:P7=0:P
8=0:P9=0:PL=0:B0=0:IF FR=10 THEN 790 ELS
E 50
790 CLS:PRINTL$::PRINTL$+L$+L$::PRINT:PR
INT"       OUT OF TEN FRAMES
  YOU ACHIEVED";SC;"%"
800 PRINTL$+L$+L$;:PRINTL$:
810 PRINTL$::PRINT"DO YOU WANT TO PLAY A
GAIN ?(Y/N)";
820 A$=INKEY$:IF A$="Y" THEN R=1 ELSE IF
 A$="N" THEN END ELSE GOTO820
830 GOTO50
```

REACTION TIMER

How quick are your reactions? First choose what level you want to play at. For example, Level 1 = 10 letters and Level 10 = 100 letters. Then simply press the key shown on the screen. At the end of the game the computer will give you your average and your rating.

It may sound easy, but just try it!



```
10 REM REACTION TIMER
20 TG=0
30 CLS
40 PRINT"       REACTION TIMER"
50 PLAY"V31L4003GCO2GCO1GCC"
60 FOR N=0 TO 1000:NEXT
70 SC=0
80 PRINT"  INPUT LEVEL (1-10)
90 INPUT LEVEL
95 IF LEVEL=0 OR LEVEL>10 THEN RUN
100 PMODE 3,1:SCREEN1,0:PCLS
110 R$="R18D26L6D2L12U28D28D24U24R12D2R0
6D20"
111 E$="R16L16D42R16L16U21R6U1L6"
112 A$="BM+25,-20;D42U21R20D21U42L20R20"
113 D$="BM+10,+0;R20D42L20U42R20"
114 Y$="BM+10,+0;D21R20U21D42L20"
115 S$="BM+10,+0R20L20D21R20D21L20R20U21
L20U21R20"
116 T$="BM+10,+0;R10D42U42R10"
117 G$="BM+10,+0;R20L20D42R20U21L10R10D2
1L20U42R20"
140 DRAW"BM10,50":DRAW R$:DRAW"BM+10,-42
":DRAW E$:DRAW A$:DRAWD$:DRAW Y$
145 PLAY"L4001CDCDCECE"
150 PCLS:DRAW"BM10 ,50;"+S$:DRAWT$:DRAW"
BM+10,+0"+ E$+A$+D$+Y$
155 PLAY"L4003CDCDCECE"
160 PCLS:DRAW"BM10 ,50;"+G$+D$
165 PLAY"L4005CDCDCECE"
```

```
170 FOR N=1 TO 4:FOR V=31 TO 0 STEP -4:P
LAY"V"+STR$(V)+"T255L25501GFEDCBA":NEXT
V,N
174 PLAY"T100"
175 PLAY"V31GFEDCBA"
1000 FOR N=0 TO 1000:NEXT
1010 B=LEVEL*10
1020 CLS
1030 A=RND(26)+64
1040 PRINT"CAN YOU FIND: ";CHR$(A);
1050 TIMER=0
1060 TI=TIMER/50
1070 PRINT @448,"TIME TAKEN :";TI
1080 A$=INKEY$
1090 IF A$=CHR$(A) THEN GOTO 1120
1100 GOTO 1060
1110 A=RND(26)+64
1120 CLS
1130 PRINT "     YOU TOOK ";TI;"
       SECOND(S)"
1140 B=B-1
1150 TG=TG+TI
1160 IF B=0 THEN GOTO 1190
1170 PRINT" YOU NOW HAVE ";B;" MORE
      LETTERS TO FIND"
1180 GOTO 1030
1190 CLS:PRINT"     YOUR AVERAGE WAS:";TG
/(LEVEL*10)
1200 PRINT"        SECONDS PER-KEY"
1210 PLAY"O1L1CDE"
```

```
1215 FOR N=0 TO 10:PLAY"GFEDCBA":NEXT
1220 CLS:PRINT "       YOUR RATING IS: "
;:GOSUB 1250
1230 GOTO 1300
1240 STOP
1250 IF TG/(LEVEL*10)<=.500 THEN PRINT"
YOUR EXECLLENT":PLAY"O3L12DGAGO1CCCO4AGA
DCDL6C":RETURN
1260 IF TG/(LEVEL*10)<=.800 THEN PRINT"
               ok! i guess":PLAY"CDECDECD
ECDECDE":RETURN
1270 IF TG/(LEVEL*10)<=.900 THEN PRINT"
            EVER HEARD OF A TYPING
      TUTOR":PLAY"AACACACA":RETURN
1280 IF TG/(LEVEL*10)<=1.00 THEN PRINT"
                  COME BACK ANOTHER DAY":
RETURN
1290 IF TG/(LEVEL*10)>1.00 THEN PRINT"
                      YOU BETTER GET SOME
PRACTICE":PLAY"O2T50DP99DP199L4DP199L2DP
99DP99FL4EP99L2EL4DL2P99DL4CL2DO3L4":RET
URN
1300 FOR N=0 TO 1000:NEXT:CLS:PRINT" ANO
THER GO ?":INPUT A$:PLAY"T1L1"
1310 PLAY"L20GABBABCD"
1320 IF A$="N" OR A$="NO"  THEN END:ELSE
 RUN
```

SPIRAL

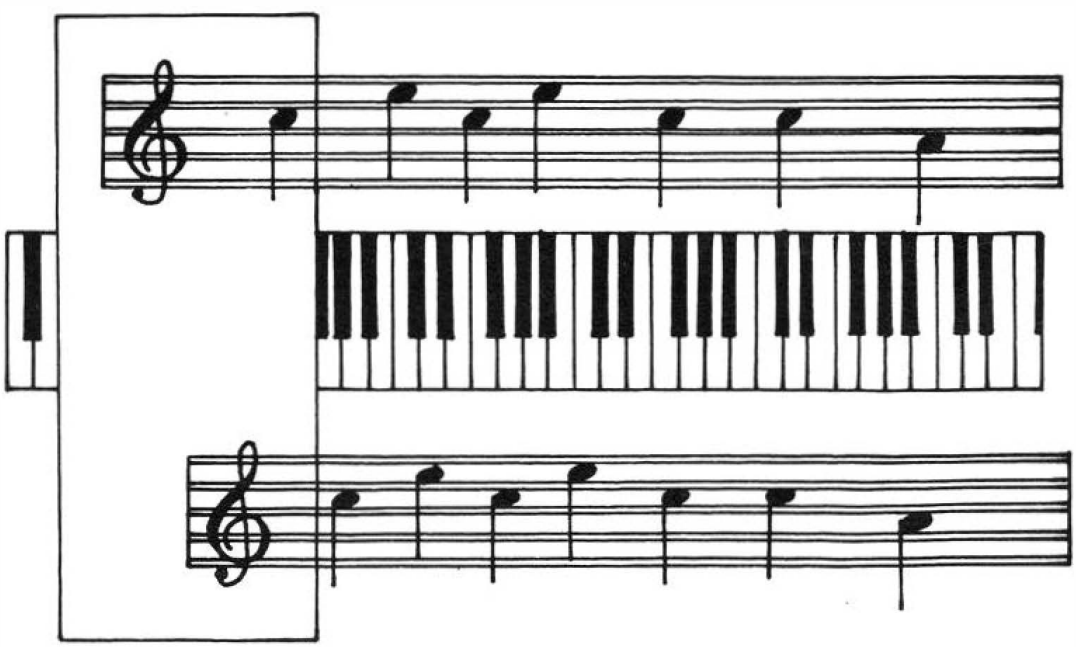
This is a pattern-drawing program that works all on its own. A spiral is drawn down the centre of the screen with lines and circles to give the picture more effect. Use poke & HFFD7, 0 if your Dragon can handle double speed. It is fascinating to watch high resolution graphics in black and white.

```
10 PCLS:PMODE4:SCREEN1,1
20 FORY=0TO191:S=.09:X=50*COS(S*Y):LINE(
X+168/2,Y)-(X+148,Y),PSET:NEXT:FORI=25TO
1STEP-1:CIRCLE(190,104),I:NEXTI
30 FORI=25TO1STEP-1:CIRCLE(45,70),I:NEXT
40 FORI=25TO1STEP-1:CIRCLE(190,35),I:NEX
T
50 FORI=25TO1STEP-1:CIRCLE(45,140),I:NEX
T
60 FORY=0TO191:X=15:LINE(X,96)-(0,Y),PSE
T:NEXT
70 FORY=0TO191:X=240:LINE(X,96)-(255,Y),
PSET:NEXT
80 FORY=0TO191:X=50*COS(S*Y):LINE(X+135,
Y)-(X+195/2,Y),PRESET:NEXT
90 FORY=191TO0STEP-1:X=50*COS(S*Y):LINE(
X+168/2,Y)-(X+148,Y),PRESET:NEXT:FORI=1T
025:CIRCLE(190,104),I,0:NEXT
100 FORI=1TO25:CIRCLE(45,70),I,0:NEXT
110 FORI=1TO25:CIRCLE(190,35),I,0:NEXT
120 FORI=1TO25:CIRCLE(45,140),I,0:NEXT
130 FORY=0TO191:X=15:LINE(X,96)-(0,Y),PR
ESET:NEXT:FORY=0TO191:X=240:LINE(X,96)-(
255,Y),PRESET:NEXT
140 PLAY"O3T30L2CC#BB-DD#AA-EE#GG-"
150 GOTO10
```

COPY THE TUNE

Can you keep up with the notes that the Dragon plays? First the Dragon plays a note and you must copy it. If the Dragon colours the top part of the screen then you must press(or play) the key marked 'A'. If the Dragon colours the second part of the screen then you must press 'B' and so on.

The keys, of course, represent the notes.



```
10 REM***COPY THE TUNE***
20 CLS
30 PRINT@06,"follow the tune"
40 A$="O1T4ABGCE"
50 PLAY A$
```

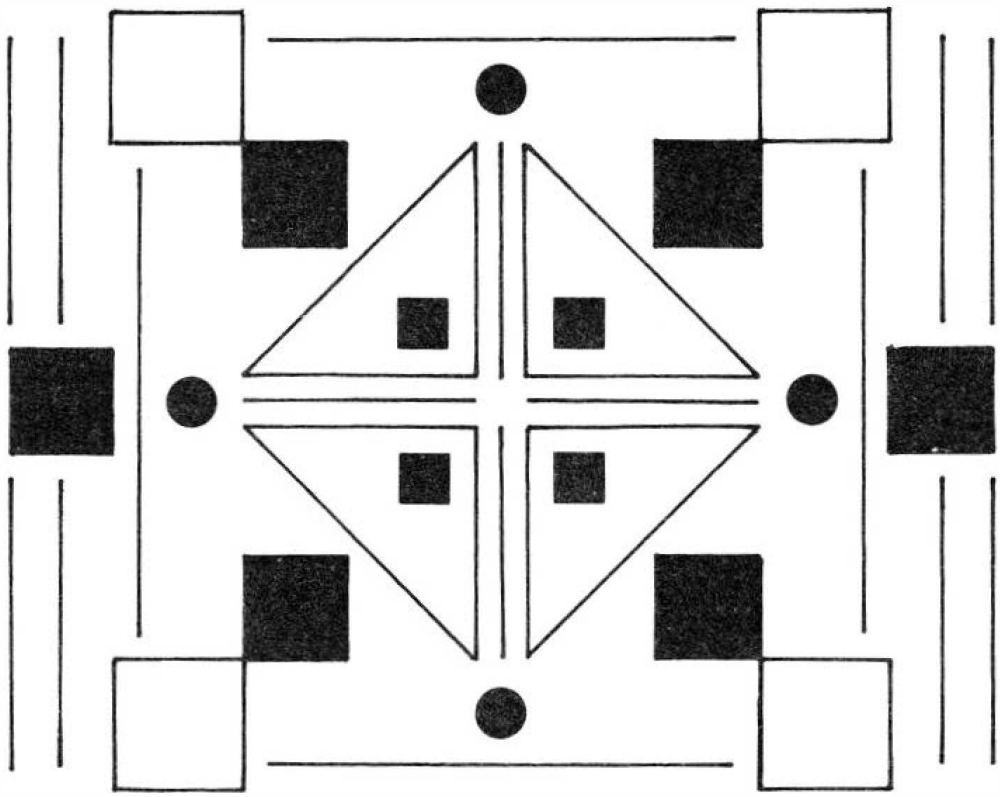
```
60 PRINT" THE OBJECT OF THE GAME
70 PRINT" IS TO COPY THE DRAGON"
80 PRINT" THE CONTROLS ARE A-D"
90 PRINT" A=THE TOP SECTION"
100 PRINT" B=THE SECOND SECTION"
110 PRINT" C=THE THIRD SECTION"
120 PRINT" D=THE FOURTH SECTION"
130 FOR PAUSE =0 TO 5000: NEXT PAUSE
140 PMODE 3,1:SCREEN1,0:PCLS
150 PCLEAR 8
160 POKE &HB3,250
170 PCLS
180 CIRCLE (128,96),80
190 LINE (64,48)-(189,48),PSET
200 LINE (47,96)-(209,96),PSET
210 LINE(63,143)-(189,143),PSET
220 PAINT (074,47),3,4
230 PAINT(120,95),1,4
240 PAINT (090,142),4,4
250 PAINT (130,174),2,4
260 PCOPY 1 TO 5:PCOPY 2 TO 6:PCOPY 3 TO
 7:PCOPY 4 TO 8
270 FOR I=0 TO 1000:NEXT I
280 PCLS
290 READ DA:DIM A$(8,90): DIM D$(DA):FOR
 N=0 TO DA:READ A$(1,N):NEXT:READ DA:FOR
N=0 TO DA :READ A$(2,N):NEXT:READ DA:FOR
 N=0 TO DA:READ  A$(3,N):NEXT:READ DA:FO
R N=0 TO DA:READ A$(4,N):NEXT
300 READ DA:FOR N=0 TO DA:READ A$(5,N):N
```

```
EXT:READ DA:FOR N=0 TO DA:READ A$(6,N):N
EXT:READ DA:FOR N=0 TO DA:READ A$(7,N):N
EXT:READ DA:FOR N=0 TO DA:READ A$(8,N):N
EXT
310 DATA20, A,B,A,B,C,D,B,A,A,B,B,C,C,D,
D,A,A,A,C,D,B,20,B,A,A,B,B,C,D,A,B,C,D,B
,A,D,B,C,C,C,A,D,D,20,C,D,B,C,D,B,B,D,D,
A,A,C,C,D,B,B,A,B,C,C,C,53,D,A,B,C,C,A,C
,B,C,D,A,B,B,B,C,A,F,C,A,D,C,A,D,B,C,D,A
,B,C,D,B,C,A,D,B,C,D,C,B,B,B,B,C,C,D,D,D
,A,A,A,A,A,A,
320 DATA 20,D,C,B,C,A,D,B,A,B,D,C,D,A,C,
B,D,A,B,C,D,A,20,C,C,D,D,C,D,A,C,B,C,A,D
,C,B,D,A,B,C,A,D,B,20,D,A,D,A,A,A,B,C,D,
A,D,A,D,D,C,C,C,C,A,B,C,20,B,A,A,C,B,D,C
,B,B,C,C,D,A,B,C,D,A,C,B,C,C
330 DA=RND(8)
340 Q=0
350 FOR N=0 TO Q
360 PMODE 3,1:SCREEN1,0:POKE &HB3,250
370 CIRCLE (128,96),80
380 IF A$(DA,N)="A" THEN PCOPY 5 TO1:ELS
E IF A$(DA,N)="B" THEN PCOPY 6 TO 2:ELSE
IF A$(DA,N)="C"  THEN PCOPY 7 TO 3:ELSEI
F A$(DA,N)="D" THEN PCOPY 8 TO 4
390 PMODE 3,1:SCREEN1,0
400 POKE &HB3,250
410 PLAY"O1"+ A$(DA,N):PCLS:PMODE 3,1:SC
REEN1,0
420 NEXT N
```

```
430  FOR M=0 TO Q
440 PMODE 3,1:SCREEN1,0:POKE &HB3,250
450 C$=INKEY$
460 IF C$="A" OR C$="B" OR C$="C"OR C$="
D" THEN GOTO470 ELSE GOTO440
470 IF C$="A" THEN PCOPY 5 TO 1:ELSE IF
C$="B" THEN PCOPY 6 TO 2:ELSE IF C$="C"
THEN PCOPY7 TO 3:ELSE IF C$="D" THEN PCO
PY 8 TO 4
480 CIRCLE(128,96),80
490 D$(M)=C$:PLAY C$
500 PCLS:PMODE 3,1:SCREEN1,0:POKE &HB3,2
50
510 IF C$=A$(DA,M)THEN POKE&HB3,250:: NE
XT M:ELSE GOTO 560
520 IF Q=20  THEN GOSUB 600
530 A$=""
540 Q=Q+1
550  FOR P=0 TO 255:NEXT P:GOTO 350
560 CLS
570 PRINT@100,"YOU FAILED AT SECTION NO.
";Q;"   ANOTHER GAME(Y/N).";
580 W$=INKEY$:IF W$="" THEN 580
590 IF W$="Y" THEN RUN:ELSE END
600 CLS:PRINT:PRINT:PRINT:PRINT"    CONG
RATULATIONS!!!":
610 PLAY"O4L3CDA"
620 RUN
```

PATTERNS

This program demonstrates more fully the drawing capabilities of the Dragon computer. It has three different pattern designs in it, and a random tune plays after each one has been drawn.



```
10 CLS:PMODE4:SCREEN1,1:PCLS
20 X=128:Y=96
30 PI=4*ATN(1):RA=20
40 T=T+1:ON T GOSUB 310,60,250
50 IF T=3 THEN T=0 ELSE GOTO40
60 Q=0
70 Q=Q+1:IF Q=5 THEN GOTO 220 ELSE ON Q
```

```
GOTO 80,90,100,110
80 FORD=0TO90:GOTO120
90 FORD=90TO180:GOTO120
100 FORD=180TO270:GOTO120
110 FORD=270TO360
120 CE=2*PI*D/360
130 X=100+RA*SIN(CE):Y=100+RA*COS(CE)
140 GOSUB170
150 NEXT
160 GOTO70
170 IF D<90 THEN LINE(255,191)-(X+20,Y-4
),PSET:RETURN
180 IF D<180 THEN LINE(255,0)-(X+20,Y-4)
,PSET:RETURN
190 IF D<270 THEN LINE(0,0)-(X+20,Y-4),P
SET:RETURN
200 IF D<360 THEN LINE(0,191)-(X+20,Y-4)
,PSET:RETURN
210 RETURN
220 CIRCLE(119,95),55
230 CIRCLE(119,95),85:PAINT(119,20):PAIN
T(50,95):PAINT(200,95):PAINT(119,170)
240 GOTO370
250 LINE(5,5)-(250,186),PSET,B
260 FOR X=5TO250:LINE(128,96)-(X,5),PSET
:NEXT
270 FOR X=5TO250:LINE(128,96)-(X,186),PS
ET:NEXT
280 FOR Y=5TO186 STEP5:LINE(5,96)-(128,Y
),PSET:NEXT
```

```
290 FOR Y=5TO186 STEP5:LINE(250,96)-(128
,Y),PSET:NEXT
300 T=0:GOTO370
310 LINE(5,5)-(250,186),PSET,B
320 FORY=5TO186:LINE(128,96)-(5,Y),PSET:
NEXT
330 FORY=5TO186:LINE(128,96)-(250,Y),PSE
T:NEXT
340 FORX=5TO250STEP5:LINE(128,5)-(X,96),
PSET:NEXT
350 FORX=5TO250STEP5:LINE(128,186)-(X,96
),PSET:NEXT
360 GOTO370
370 TU=RND(3):ON TU GOTO 380,400,410
380 FORI=1TO31:PLAY"T255V"+STR$(I)+"O3CD
EFGABO4CDEFGABO5CDEFGAB":NEXT
390 FORI=31TO1STEP-1:PLAY"V"+STR$(I)+"O5
BADCO4BADCO3BADC":NEXT:GOTO 420
400 FORI=1TO5:PLAY"V28T255O"+STR$(I)+"C#
CD#DE#EF#FG#GA#AB":NEXT:GOTO 420
410 FORI=1TO5:PLAY"T80O"+STR$(I)+"V"+STR
$(I+24)+"CABBAGED":NEXT
420 FOR X=0TO255:LINE(X,0)-(X,191),PSET:
NEXT
430 FOR X=255TO0STEP-1:LINE(X,0)-(X,191)
,PRESET:NEXT
440 RETURN
```

BAT 'N' BALL

You've heard of the ball game squash. Well here is a Dragonised version of the game where the object is to keep the ball in court for as long as possible.

There is also a high score feature.

```
10 REM****BAT 'N BALL****
20 H=0
30 C=0:S=0:CLS
40 PRINT @ 138,"BAT 'N BALL"
50 PRINT @ 200,"BOUNCE THE BALL"
60 PLAY"L40CDCDCD"
70 PRINT @ 264,"OFF THE WALL!!!"
80 PLAY"EFEFEF"
90 PRINT @ 322,"USE THE '<' AND THE '>'
KEYS"
100 PLAY"GAGAGABB"
110 FOR T=1 TO 1000:NEXT T:CLS 3
120 J=RND(10)+4:K=RND(15)+5
130 Q=-1:P=-1
140 FOR A=32 TO 63
150 PRINT @ A,CHR$(198)
160 NEXT
170 FOR A=2 TO 12
180 PRINT @ (32*A),CHR$(198)
190 PRINT @ (32*A)+31,CHR$(198)
200 NEXT
210 X=RND(6)+7
```

```
220 PRINT @ 416+X,CHR$(140)+CHR$(140);
230 A$=INKEY$
240 FOR T=1 TO 50:NEXT T
250 IF A$="," THEN GOSUB 440
260 IF A$="." THEN GOSUB 450
270 IF X<0 THEN X=0
280 IF X>30 THEN X=30
290 PRINT @ (32*J)+K," ";
300 K=K+(1*Q)
310 J=J+(1*P)
320 PRINT @ (32*J)+K,"*";
330 IF K>29 OR K<2 THEN GOSUB 670
340 IF J<3 THEN GOSUB 700
350 IF J=13 THEN 370
360 GOTO 220
370 IF(417+X)=(32*J)+K OR (416+X)=(32*J)
+K THEN 410
380 PLAY"O1;L4;E;G;C"
390 C=C+1:IF C=3 THEN 460
400 GOTO 120
410 S=S+1674:P=P*-1
420 PLAY "L90;O4;C;E;C"
430 GOTO 220
440 PRINT @ (416)+X,"  ";:X=X-2:RETURN
450 PRINT @ (416)+X,"  ";:X=X+2:RETURN
460 CLS
470 PRINT "YOU SCORED ";S;" POINTS"
480 IF S=0 AND S>=H THEN 730
490 IF S<H THEN GOTO 630
500 PRINT "THAT IS THE HIGHEST SCORE!!!"
```

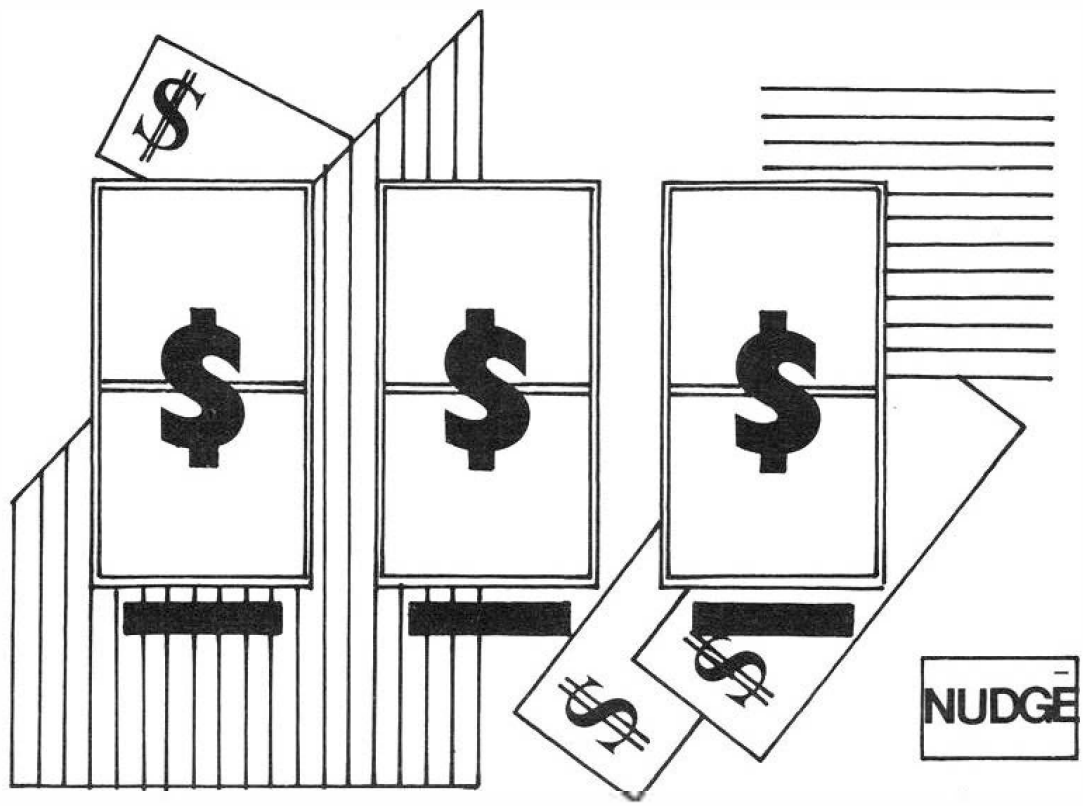
```
510 PLAY"O2;L8;C;D;E;F;G;A;B"
520 PLAY"O3;C;D;E;F;G;A;B"
530 PLAY"O4;C;D;E;F;G;A;B;B;B"
540 H=S
550 PRINT:PRINT
560 FOR P=1 TO 8
570 PRINT"********* ";H;" *********"
580 NEXT P
590 PLAY"O4;F;A;G"
600 GOTO 730
610 IF J$="Y" THEN GOTO 30
620 PLAY"O2;G;B;A;A"
630 PRINT "NOT BAD, BUT THE HIGHEST SCOR
E"
640 PRINT "IS*** ";H;"***"
650 PLAY"O2;A;F;G"
660 GOTO 730
670 Q=Q*-1
680 PLAY"L40;O4;A"
690 RETURN
700 P=P*-1
710 PLAY"L40;O5;C"
720 RETURN
730 PRINT"HAVE ANOTHER GO?"
740 IF INKEY$="Y" THEN GOTO 30 ELSE GOTO
 740
```

BANDIT

Can you beat the system? Just place your bets into the machine and hey presto!

You start off with a credit of $100 in the bank; if you lose this credit then your game is over and you must pay a fine to the judge. If you get a 'BAR' then you have the choice to nudge one of the three reels. To nudge the first reel you should press 'A', to nudge the second reel press 'B' and to nudge the third press 'C'. The Dragon will tell you whether you have won or lost. If you win then the Dragon will say whether it is a Double or a Jackpot, and so on.

You cannot bet more than $600 dollars at one time. When you have more than $10,000 the game ends. NB: when you enter your name keep it less than six letters.



```
 0 REM***BANDIT***
20 REM***V 2.10******************
40 PCLEAR 8
50 PMODE 3,4:SCREEN1,0:PCLS4
60 COLOR 2
70 LINE(0,0)-(0,255),PSET
80 LINE(230,0)-(230,255),PSET
90 PMODE 3,1:SCREEN1,0:PCLS
100 PAINT(96,128),4
110 COLOR 2
120 LINE(230,30)-(230,255),PSET:LINE(0,3
0)-(0,255),PSET
130 LINE(0,140)-(230,140),PSET
140 LINE(0,110)-(230,110),PSET
150 LINE(0,30)-(230,30),PSET
160 LINE(0,165)-(230,165),PSET
170 PAINT(20,130),1,2
180 DRAW"BM20,120;C3D10R5U5L5R5U5L5R5BR5
D10U5R5D5U10L5R5BR5D10U10R5D10BR5U10R5D1
0L5R5BR5U10BR5R2D10U10R2BR40U5R40D1L40D1
R40D1L40D1R40D1L40D1R40"
190 CLS
200 PLAY"T20GFEDCBAGFEDCBAGFEDCBA"
210 PRINT"          GAMBIT ":PRINT:PRINT:P
RINT:
220 INPUT " WELCOME TO THE THRILL OF A L
IFE TIME 'GAMBIT'. PLEASE ENTER      YOUR
  NAME GAMBLER";NA$:NA$=NA$+"  "
230 IF LEN(NA$)>10 THEN RUN
240 FOR PAUSE=0  TO 500:NEXT PAUSE
```

```
250 CLS
260 A=40
270 P=100
280 PRINT@0,"
      ";:PRINT@3,"   ";NA$;"PLACE YOUR BET
S"
290 PRINT
300 INPUT M
310 IF M<=-1 OR M>600 THEN GOTO 300
320 IF P<=00 OR P>10000 THEN 2040
330 IF M=0 THEN 710
340 PRINT@135,"YOU HAVE BET    :$":M;"
350 FOR B=0 TO 2000:NEXT B
360 M=INT(M)
370 PMODE 3,1:SCREEN1,0:GOSUB 2190
380 X=INT (RND (5))+1
390 Y=INT (RND (5))+1
400 Z=INT (RND (5))+1
410 A=30
420 B=40
430 ON X GOSUB 1260,1260,1300,1360,1390,
1430
440 SOUND 200,1
450 A=110
460 ON Y GOSUB 1260,1260,1300,1360,1390,
1430
470 SOUND 200,1
480 A=182
490 ON Z GOSUB 1260,1260,1300,1360,1390,
1430
```

```
500 SOUND 200,1
510 FOR N=0 TO 500:NEXT N
520 IF X=2 AND Y=2 AND Z=2 THEN GOTO 100
0
530 IF X=2 OR Y=2 OR Z=2 THEN GOSUB 1470
540 IF X=2 AND Y=2 AND Z=2 THEN 1000
550 IF X=Y THEN 750
560 IF X=Z THEN 780
570 IF Y=Z THEN 800
580 REM***SCORING SEC.**********
590 DRAW"C2BM30,150;D5R5U5D10BR3U10R5D10
L5R5BR3U10D10R5U10D10BR10U10D10R5BR5U10R
5D10L5R5BR5R5U5L5U5R5BR5R5L5D5R5L5D5R5L5
R5
600 FOR N=0 TO 12
610 SOUND N*10+10,1
620 NEXT N
630 FOR N=0 TO 200:NEXT
640 PCOPY 7 TO 4
650 LINE(0,165)-(230,165),PSET
660 P=P-M
670 CLS:PRINT@448," YOU NOW HAVE ";P:"DO
LLARS  TO     SPEND"
680 FOR PAUSE =0 TO 500:NEXT PAUSE
690 GOTO 280
700 IF P>=0 OR P<10000THEN 2040
710 IF P<0 THEN 820
720 IF P=0 THEN 840
730 IF P>0 THEN 860
740 GOTO 1150
```

```
750 IF Y=Z THEN 880
760 IF Y=1 THEN 1090
770 GOTO 1170
780 IF Z=1 THEN 1090
790 GOTO 570
800 IF Z=1 THEN 1090
810 GOTO 1170
820 CLS:PRINT@0,"  PAY UP MATE AND MAKE
IT QUICK!";
830 GOTO 2040
840 CLS:PRINT@0,"  YOU BROKE EVEN MATE!"
850 GOTO 1150
860 CLS:PRINT@0,"  COLLECT YOUR WINNINGS
!!!"
870 GOTO 2040
880 IF Z=1 THEN 1000
890 DRAW"C2BM30,150;R2D10U10R2BR5D10R5U1
0L5R5BR5D10U5R5U5L5R5BR10D10R5U5D5R5U10B
R5D10U10BR5D10U10R5D10BR5U10R5D10BR5R5L5
U5R5L5U5R5BR5D10U5R2D3R3D2U2L3U3L2U5R5D5
U5B
900 FOR N=1 TO 12
910 SOUND N*10+1,1
920 NEXT N
930 FOR N=1 TO 12
940 PLAY "O4T200ABCDE"
950 NEXT N
960 PCOPY 7 TO 4:FOR N=0 TO 200:NEXT
970 LINE(0,165)-(230,165),PSET
980 P=(((10*M)+M)+P)
```

```
990 GOTO 670
1000 DRAW"C2BM30,150;R2D10L3U5D5R3U10R2B
R5D10U5R5D5U10L5R5BR5R5L5D10R5BR5U5R5D5U
5L1U5D5L4U5D5R4U5BR10D10U5R5U5L5D5R5U5BR
5D10R5U10L5R5BR5R2D10U10R2"
1010 FOR N=0 TO 12
1020 SOUND N*10+1,1
1030 NEXT N
1040 PLAY"O4T255L255GFEDCBAGFEDCBAGFEDCB
AGFEDCBAGFEDCBA"
1050 P=(((100*M)+M)+P)
1060 PCOPY 7 TO 4:FOR N=0 TO 200:NEXT
1070 LINE(0,165)-(230,165),PSET
1090 GOTO 670
1090 DRAW"C2BM30,150D10R2U5D5R2U10BR5D10
U10BR5D10U10R5D10BR5U10R5D10BR5U10BR5D10
U10R5D10BR5U10R5D2U2L5D10R5U5L2BR20D5U10
D5R5U5D10BR5U10R5D10U5L5R5D5BR5U10R5D10B
R5U10R5D10L5R5"
1100 P=(((5*M)+M)+P)
1110 FOR N=0 TO 12:SOUND N*10+1,1:NEXT:F
OR N=0 TO 200:NEXT
1120 PCOPY 7 TO 4
1130 LINE(0,165)-(230,165),PSET
1140 GOTO 670
1150 FOR N=0 TO 1000:NEXT N
1160 GOTO190
1170 DRAW"C2BM30,150;D10R5U10L5R5BR5D10R
5U10L5R5BR5D10R5U10BR5R5L5D5R5L5D5R5U4BU
2U4BR5D10R5BR5R5L5U5R5L5U5R5L5"
```

```
1180 FOR N=0 TO 12
1190 SOUND N*10+1,1
1200 NEXT
1210 P=(((2*M)+M)+P)
1220 FOR N=0 TO 200:NEXT:PCOPY 7 TO 4
1230 LINE(0,165)-(230,165),PSET
1240 GOTO 670
1250 STOP
1260 REM*****FRUIT SECTION******
1270 A$="C2D14R1U14R2D1L2R2F2D1H2F2D4L1U
4D4G2D1F2D4R1U4D4G2L2U4D4R2BM+10,+0D1G4D
8R4L4D4U4R4R4D4U4U4U4H4BM+10,+18D14U8R4F
4D4U4H4L4U6R4F4D2
1280 DRAW"BM"+STR$(A)+","+STR$(B)+":"+A$
1290 RETURN
1300 A$="C1R6D1L6D1R6D1L12R16D1L16D1L2R1
6D1L16R16U4R2F4D1H4F4R10U1L10U1R16U1L16D
5D34E1U30R1D29E1U28F1D26E1U25F1D24E1U22F
1D20E1U19F1D18F1U15E1D10L13D18L1U18L1D24
L1U24L1D30L1UJ30L1D24L1U24L1D18L1U18L1D13
L1U13L1D11L1U11L1D9L1U9L1D7L1U7L1D5L1U6R
19U10L4U7"
1310 B$="C1L8D1R8D1L8D1R8D1L8D1R8D1L8D1R
8D1L8D1R8D1L8D1R8D1L8D1R8D1L8D1R8D1L8R9U
9R1D9L17U8R2D8U13R2D13U16
1320 DRAW "BM"+STR$(A)+","+STR$(B)
1330 DRAW A$
1340 DRAW B$
1350 RETURN
1360 A$="C1R12D1L12D1R12L4D1L12D1R12D1L1
```

```
2D1R12F8D6G6L14H6U6E8D20H1U18G1D16H1U14G
1D12H1U10G1D8H1U6R10U12D12D13R1U25R1D25R
1U25R1D25R1U25R1D25R1U25F1D24E1U18F1D16E
1U14E1D12F1U10
1370 DRAW"BM"+STR$(A)+","+STR$(B)+ A$:DR
AW"BM+10,+5"+A$:DRAW "BM-50,+5"+A$
1380 RETURN
1390 A$="C2R8D10L1U9L7D09L1U10D10R8U1L8D
2R1U2R8D2L1U2D2R8D6R2D14R2D18L1U18L1U14L
2U6L20D1R10L10D6L2D14L2D18R27L1U18L2U14L
2U6L16D6L2D14L2D17R21U18D18L1U38L1D38L1U
38L1D38L1U38L1D38L1U38L1D38L1U38L1D38L1U
38L1D38L1U38L1D38L1U38L1D38L1U32D14L1D18
R18F5R9D9L9U9
1400 DRAW"BM"+STR$(A)+","+STR$(B)
1410 DRAW A$
1420 RETURN
1430 A$="C2R12D1L12D1R12D1L12R2D1R12D1L1
2D1R12D1L12R2D1R12D1L12D1R12D1L12D1R12":
B$="L8D1R8D1L8D1R8D1L12D1R12D1L12D1R12D1
L12D1R12D1L12D1R12D1L12R4D1R8D1L8D1R8D1L
8U3R9U10R1D10R1U10R1D10"
1440 DRAW"BM"+STR$(A)+","+STR$(B)+A$:DRA
W "BM-14,+4"+B$:DRAW "BM+9,+4"+B$:DRAW "
BM+4,-32"+B$
1450 RETURN
1460 REM**************************
1470 PLAY"T1003CP10DF10E"
1480 DRAW"C2BM0,30;U20R231D20"
1490 DRAW"BM80,30;U20L1D20;BM156,30;U20L
```

```
1D20"
1500 DRAW"BM020,15;D10U10R5D10BR3U10D10R
5U10BR3D10R5U10L5R5BR3D10U10R5L5D10R5U5L
2R2D5BR3U10R5L5D5R5L5D5R5L5"
1510 DRAW"BM100,15;D10U10R5D10BR3U10D10R
5U10BR3D10R5U10L5R5BR3D10U10R5L5D10R5U5L
2R2D5BR3U10R5L5D5R5L5D5R5L5"
1520 DRAW"BM170,15;D10U10R5D10BR3U10D10R
5U10BR3D10R5U10L5R5BR3D10U10R5L5D10R5U5L
2R2D5BR3U10R5L5D5R5L5D5R5L5"
1530 C$=INKEY$:IF C$="" THEN 1530
1540 IF C$="A" OR C$ ="B" OR C$="C" THEN
 GOTO 1550:ELSE GOTO 1500
1550 PMODE 3,1:SCREEN1,0
1560 IF C$="A" THEN GOSUB 1620
1570 IF C$="B" THEN GOSUB 1760
1580 IF C$="C" THEN GOSUB 1900
1590 FOR N=0 TO 500
1600 NEXT N
1610 RETURN
1620 A=30
1630 FOR N=0 TO 6
1640 X=INT (RND(5))+1
1650 A=30
1660 COLOR 2
1670 LINE(00,30)-(80,110),PSET,B
1680 COLOR 4
1690 LINE(05,35)-(75,105),PSET,BF
1700 COLOR 3
1710 LINE(05,35)-(75,105),PSET,BF
```

```
1720 SOUND 200,1
1730 NEXT N
1740 ON X GOSUB 1260,1260,1300,1360,1390
,1430
1750 RETURN
1760 A=110
1770 FOR N=0 TO 6
1780 Y=INT (RND(5))+1
1790 A=110
1800 COLOR 1
1810 LINE(85,35)-(150,105),PSET,BF
1820 COLOR 2
1830 LINE(79,30)-(155,110),PSET,B
1840 COLOR 3
1850 LINE(85,35)-(150,105),PSET,BF
1860 SOUND 200,1
1870 NEXT N
1880 ON Y GOSUB 1260,1260,1300,1360,1390
,1430
1890 RETURN
1900 A=30
1910 FOR N=0 TO 6
1920 Z=INT (RND(5))+1
1930 A=182
1940 COLOR 2
1950 LINE(155,30)-(230,110),PSET,B
1960 COLOR 2
1970 LINE(160,35)-(227,105),PSET,BF
1980 COLOR 3
1990 LINE(160,35)-(227,105),PSET,BF
```

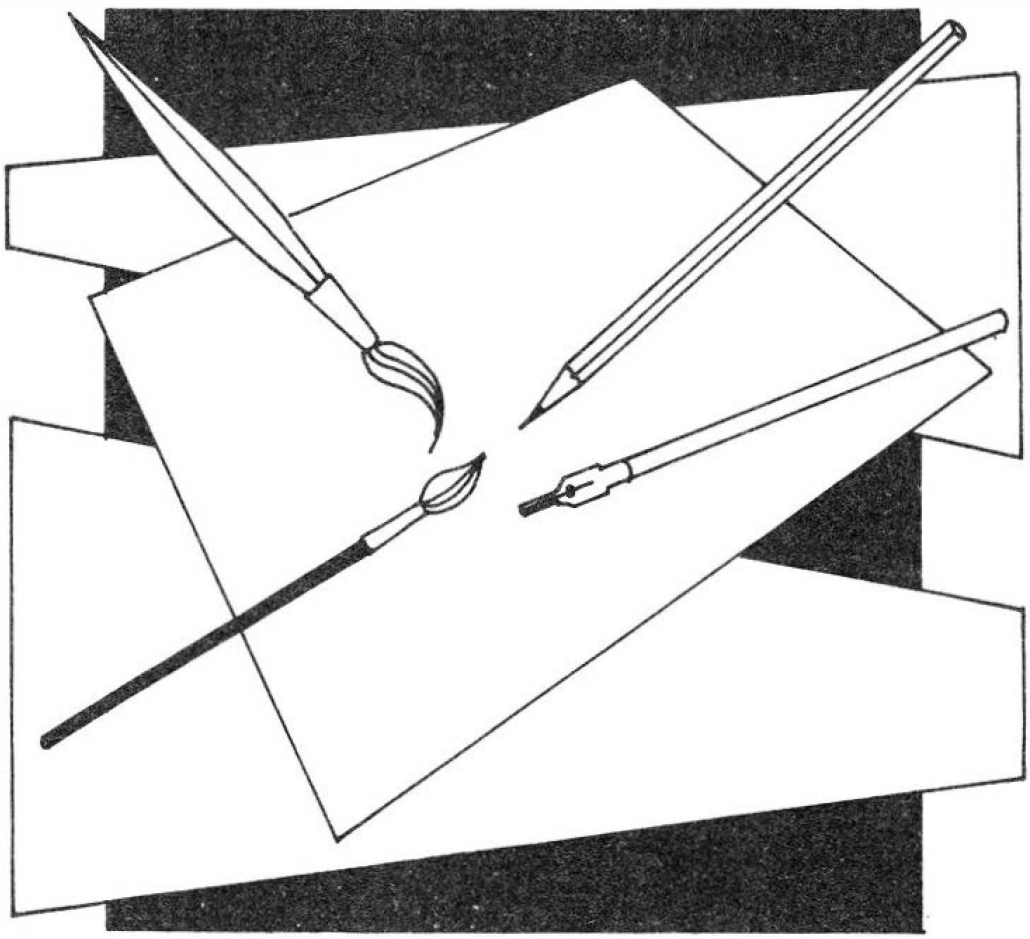
```
2000 SOUND 200,1
2010 NEXT N
2020 ON Z GOSUB 1260,1260,1300,1360,1390
,1430
2030 RETURN
2040 IF P<=00 THEN 2050:ELSE 2080
2050 CLS: PRINT" YOU ARE ORDERED TO PAY
THE FINE OF ";INT(P); "   DOLLARS....
                  OR GOTO PRISON
2060 PLAY"O2T5DP99DP199L4DP199L2DP99FL4E
P99L2EL4DL2P99DL4CL2DO3L4"
2070 FOR N=0 TO10000:NEXT:GOTO220
2080 CLS
2090 PRINT"////////////////////////////
///"
2100 PRINT"  PAY ";NA$;" THE SUM OF....
        ";P;" DOLLARS"
2110 PRINT
2120 PRINT
2130 PRINT" ACCY.NO.1234598754A"
2140 PRINT" WIN-A-LOT GAMBLE. COMPANY"
2150 PRINT
2160 PRINT
2170 PRINT"////////////////////////////
///"
2180 FOR N=0  TO 9999:NEXT:GOTO 220
2190 FOR N=0 TO 4
2200 COLOR 2
2210 LINE(00,30)-(80,110),PSET,B
2220 LINE(05,35)-(75,105),PSET ,BF
```

```
2230 COLOR 3
2240 LINE(05,35)-(75,105),PSET,BF
2250 SOUND 200,1
2260 COLOR 2
2270 LINE(85,35)-(150,105),PSET,BF
2280 LINE(79,30)-(155,110),PSET,B
2290 COLOR 3
2300 LINE(85,35)-(150,105),PSET,BF
2310 SOUND 200,1
2320 COLOR 2
2330 LINE(157,030)-(230,110),PSET,B
2340 COLOR 2
2350 LINE(160,35)-(227,105),PSET,BF
2360 COLOR 3
2370 LINE(160,35)-(227,105),PSET,BF
2380 SOUND 200,1
2390 NEXT
2400 RETURN
```

DRAGON DRAW

This is a simple program that will draw anything you want to design. The instructions are in the program and are very straightforward. 'P' simply means 'pen-down' to draw a line and 'U' means 'pen-up' not to draw a line or to delete an existing line. Using the cursor keys you can move in any direction.

But remember, it was on such a program as this that designs like Concorde were created. So, let's put pen to paper...



```
1 'REPLACE THIS LINE WITH          POKE&H
FFD7,0 IF YOUR COMPUTER   CAN HANDLE IT
10 CLS:PRINT"          DRAGON DRAW
                ==========="
20 PRINT@128,"WHAT COLOUR SCREEN DO YOU
      REQUIRE? (BLACK/WHITE)"
30 GOSUB270:IF A$="B" THEN PRINT@170,"BL
ACK)":PCLS0:B=1
40 IF A$="W" THEN PRINT@170,"WHITE)":PCL
S1:B=0
50 IF A$="W" OR A$="B" THEN 60 ELSE GOTO
20
60 SOUND200,1:PRINT:PRINT"          INSTR
UCTIONS                "
70 PRINT" USE THE CURSER KEYS TO DRAW TH
EPICTURE"
80 PRINT"P=DRAW A LINE  U=DELETE/NO LINE
"
90 X=128:Y=96
100 PRINT:PRINT" PRESS   ANY   KEY   TO
  START"
110 GOSUB270
120 GOSUB280
130 PMODE4:SCREEN1,1:GOSUB140
140 GOSUB270:MO=ASC(A$)
150 IF MO=94 THEN Y=Y-1
160 IF MO=10 THEN Y=Y+1
170 IF MO=8 THEN X=X-1
180 IF MO=9 THEN X=X+1
190 IF MO=80 THEN GOSUB250
```

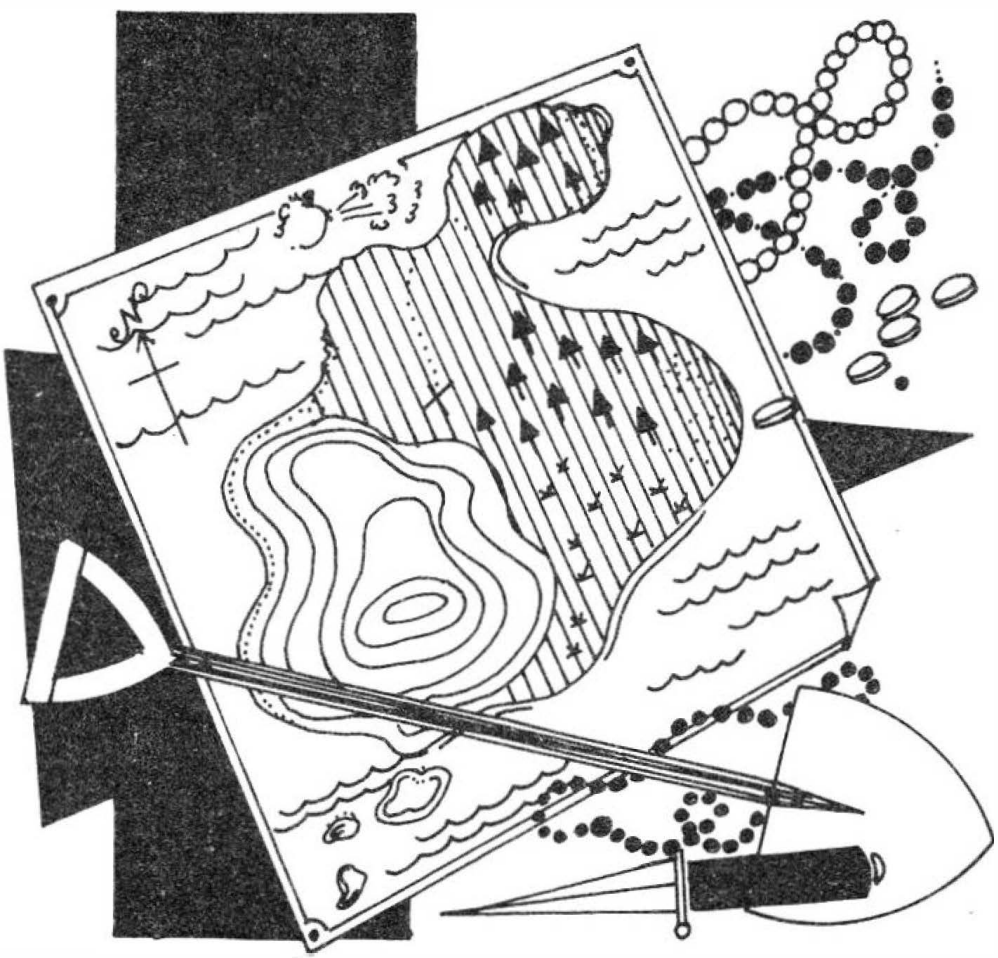
```
200 IF MO=85 THEN GOSUB230
210 IF B=1 THEN PSET(X,Y) ELSE IF B=0 TH
EN PSET(X,Y,0) ELSE IF B=2 THEN PSET(X,Y
,0) ELSE IF B=3 THEN PSET(X,Y)
220 GOTO140
230 IF B=1 THEN B=2:RETURN
240 IF B=0 THEN B=3:RETURN
250 IF B=3 THEN B=0:RETURN
260 IF B=2 THEN B=1:RETURN
270 A$=INKEY$:IF A$="" THEN 270 ELSE RET
URN
280 IF B=1 THEN 290 ELSE 320
290 LINE(0,0)-(25,25),PSET:LINE-(230,166
),PSET,B:LINE-(255,191),PSET
300 LINE(230,25)-(255,0),PSET:LINE(25,16
6)-(0,191),PSET
310 RETURN
320 LINE(0,0)-(25,25),PRESET:LINE-(230,1
66),PRESET,B:LINE-(255,191),PRESET
330 LINE(230,25)-(255,0),PRESET:LINE(25,
166)-(0,191),PRESET
340 RETURN
```

TREASURE TRAIL

Trail blaze your way through this game of skill. How many treasures can you get your hands on before your time runs out? You have two minutes to collect all five treasures.

Each time the game restarts, the treasures are in a different location; so the same trail is not worth blazing twice.

Full instructions are given in the program and a rating system is also included. My rating is 'average': see if you can do any better.



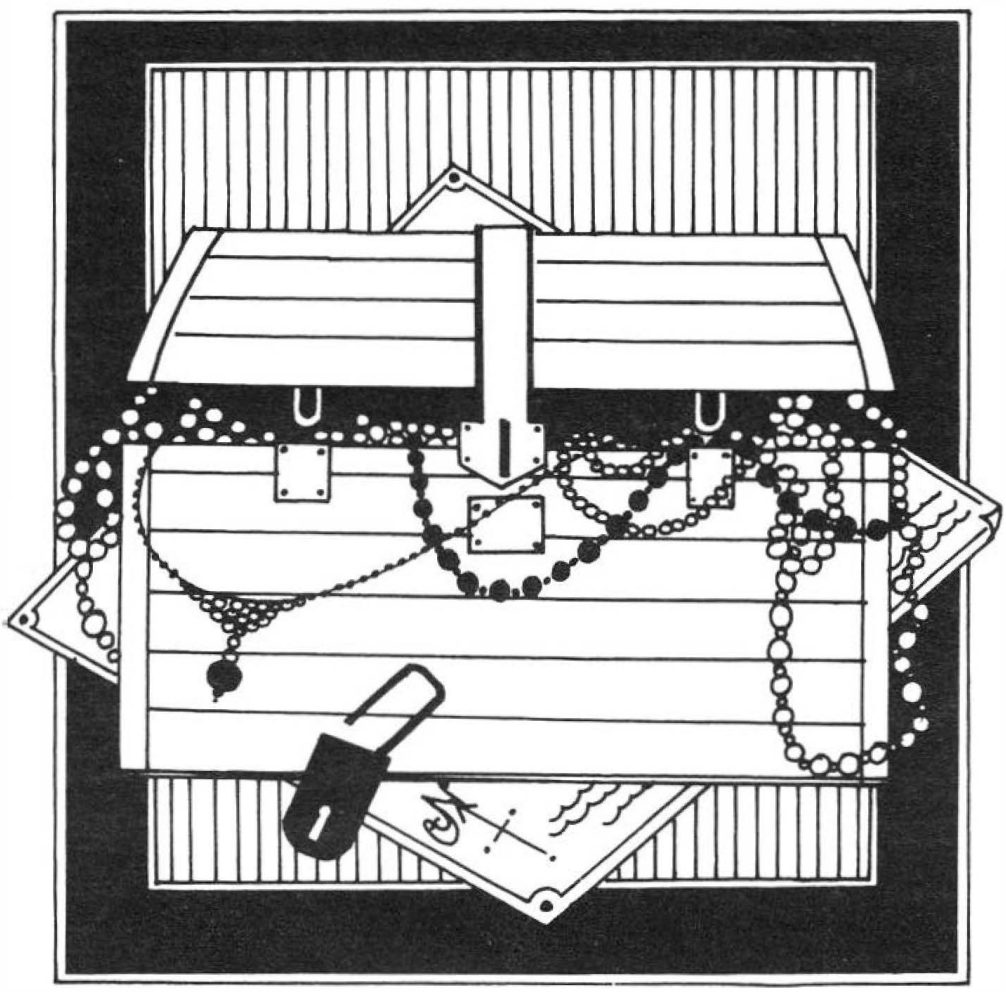
```
10 PCLS
20 PCLS:PMODE4:SCREEN1,0:DRAW"S8:BM10,10
;R10L5D10U10R5;BR5R10D5L10R3F5H5L3D5U10R
10;BR5R10L10D5R5L5D5R10;BR5U5R10D5U5L10E
5F5D5:BR5R10U5L10U5R10:BR5D10R10U10;BR5R
10D5L10R3F5H5L3D5U10R10:BR5R10L10D5R5L5D
5R10;"
30 LINE(10,32)-(245,32),PSET
40 LINE(11,33)-(244,33),PSET
50 LINE(12,34)-(243,34),PSET
60 DRAW"BM45,150;S8;R10L5D10U10R5;BR5R10
D5L10R3F5H5L3D5U10D5R10;BR5D5U5E5F5L10R1
0D5;BR5R10L5U10L5R10;BR5D10R10;"
70 LINE(40,173)-(190,173),PSET
80 LINE(41,174)-(189,174),PSET
90 LINE(42,175)-(188,175),PSET
100 FORP=0TO3
110 DRAW"S4;BM50,50;R12D3L12U3D3E2F2E2F2
E2F2H2G2H2G2H2G2R1D1G5F5R10E5H5U1
120 CIRCLE(100,100),15:DRAW"BM90,100;S2B
R3R3D2L6U6R6;BR6R6D6L6U6R6;BR6D6R6;BR6U6
R2D1R2D2R2D2L2D2L2DC1L2
130 DRAW"S7BM160,75;E5F5D5G5H5U5R9D5L9U5
140 PAINT(161,76),1
150 FORI=1TO31:PLAY"V"+STR$(I)+"T200CBDA
EGF":NEXT:FORI=31TO1STEP-1:PLAY"V"+STR$(
I)+"CDECDE":NEXT
160 CLS:PRINT"       TREASURE   TRAIL
                 ========   =====":"
170 PRINT"  THE IDEA OF THIS GAME IS TO
```

```
  TAKE ALL THE TREASURES ON THE  SCREEN
 BEFORE YOUR TIME RUNS OUT":PRINT"  TO D
O THIS USE THE CURSER KEYSTO MOVE IN THE
 APPROPRIATE      DIRECTION. MOVING OFF
THE SCREENCAUSES YOU TO COME ON THE OTHE
R SIDE"
180 PRINT:PRINT"           good  luck
                  happy  hunting";
190 PRINT@484,"(PRESS ANY KEY TO START)"
;
200 A$=INKEY$:IF A$="" THEN 200
210 PCLS
220 PMODE3,1:SCREEN1,1
230 TS=6000
240 HI=0
250 PCLS
260 FORI=1TO5:X=RND(18):Y=RND(18):PSET(X
*13+4,Y*10+4,2):NEXT
270 FORI=1TO31:PLAY"T200V"+STR$(I)+"BGBG
":NEXT:FORI=31TO1STEP2:PLAY"V"+STR$(I)+"
CAB":NEXT
280 PMODE3,1:SCREEN1,1
290 TIMER=0
300 X=127:Y=96
310 A$=INKEY$
320  IF A$=CHR$(8) THEN M=1
330  IF A$=CHR$(9) THEN M=2
340  IF A$=CHR$(10) THEN M=3
350  IF A$=CHR$(94) THEN M=4
360  IF M=1 THEN X=X-1
```

```
370 IF M=2 THEN X=X+1
380 IF M=3 THEN Y=Y+1
390 IF M=4 THEN Y=Y-1
400 IF X<0 THEN X=255
410 IF X>255 THEN X=0
420 IF Y>191 THEN Y=0
430 IF Y<0 THEN Y=191
440 IF TIMER>6000 THEN GOTO500
450 IF PPOINT(X,Y)=6 THEN GOSUB 470
460 PSET(X,Y,3):GOTO310
470 HI=HI+1
480 PLAY"T20004BBAAGGFFEEDDCC"
490 IF HI=5 THEN PLAY"T130ABFGABFGABFGAB
FGABFGFG":GOTO500 ELSE RETURN
500 CL=TIMER:IF CL<=TS THEN TS=CL
510 CLS0::PRINT@1,"SHORTEST TIME";CHR$(1
92);TS;
520 PRINT@129,"YOUR TIME";CHR$(192);CL;
530 IF CL<500 THEN C$="YOU CHEAT" ELSE I
F CL>500 AND CL<1000 THEN C$="EXCELLENT"
 ELSE IF CL>1000 AND CL<1500 THEN C$="VE
RY GOOD"
540 IF CL>1500 AND CL<2000 THEN C$="AVER
AGE" ELSE IF CL>2000 AND CL<2500 THEN C$
="NEED PRACTISE" ELSE IF CL>2500 AND CL<
3500 THEN C$="NO GOOD"
550 IF CL>3500 AND CL<4500 THEN C$="USEL
ESS" ELSE IF CL>4500 THEN C$="DO YOU WHE
RE GLASSES?"
560 PRINT@225,"RATING=":C$;
```

```
570 PRINT@489,"ANOTHER GAME(Y/N)":
580 Z$=INKEY$:IF Z$="Y"THEN GOTO240 ELSE
 IF Z$="N" THEN GOTO600
590 GOTO580
600 CLS4:S$=CHR$(128):FORI=0TO510STEP32:
PRINT@I,CHR$(134)::NEXT
610 X$=CHR$(134):PRINT@31,X$;:PRINT@63,X
$;:PRINT@95,X$;:PRINT@127,X$;:PRINT@159,
X$::PRINT@191,X$;:PRINT@223,X$;:
620 PRINT@255,X$;:PRINT@287,X$;:PRINT@31
9,X$;:PRINT@351,X$;:PRINT@383,X$;:PRINT@
415,X$;:PRINT@447,X$;:PRINT@479,X$;:
630 FORI=1TO30:PRINT@I,X$;:NEXT:FORI=481
TO510:PRINT@I,X$;:NEXT
640 S$=CHR$(128):FORI=101TO104:PRINT@I,S
$;:NEXT
650 PRINT@136,S$;:PRINT@137,S$;:PRINT@13
3,S$;
660 PRINT@165,S$;:PRINT@169,S$;:PRINT@19
7,S$;:PRINT@200,S$;:PRINT@201,S$;:FORI=2
29TO231:PRINT@I,S$;:NEXT
670 PRINT@261,S$;:PRINT@264,S$;:PRINT@26
5,S$;:PRINT@293,S$;:PRINT@297,S$;:PRINT@
325,S$;:PRINT@328,S$;:PRINT@329,S$;
680 PRINT@357,S$;:PRINT@358,S$;:PRINT@35
9,S$;
690 PRINT@108,S$;:PRINT@116,S$;:FORI=113
TO124:PRINT@I,S$;:NEXT
700 PRINT@141,S$;:PRINT@147,S$;:PRINT@15
0,S$;
```

```
710 PRINT@174,S$;:PRINT@178,S$;:PRINT@19
2,S$;
720 PRINT@207,S$;:PRINT@209,S$;:PRINT@21
4,S$;
730 PRINT@240,S$;:FORI=246TO249:PRINT@I,
S$;:NEXT
740 PRINT@271,S$;:PRINT@278,S$;:PRINT@30
2,S$;:PRINT@310,S$;:PRINT@333,S$;:PRINT@
342,S$;
750 PRINT@364,S$;:FORI=374TO380:PRINT@I,
S$;:NEXT
760 FORI=1TO6000:NEXT:CLS:END
```



THE BIANCO MANSION

For many years you have lived the life of a soul tormented with the need to find a particular map. You have at last found the building where it lies, the picturesque Bianco Mansion, home of Count Phillipe and Contessa Catherine de Bianco. They are away at present and you have entered the building. The race is on; you must find the map and leave via the courtyard. There are security guards patrolling the mansion in the owners' absence, so be careful.

This game is a standard-style text adventure. You must type in commands to the computer in the format of VERB NOUN such as GET TORCH. A few examples to aid you are given below, but the vast majority you must find out for yourself — for this is the fun of the game.

The computer allows you to enter quite a wide range, but if you cannot get a command to work, try altering the syntax as opposed to trying another command. There are a couple of objects, with double-barrelled names, to pick up, and you should enter the second word of the object. But I'm not going to tell you any more than that.

Some example commands:

LOOK
MOVE (N, S, E, W)
TAKE (Object such as rope, ring, torch, and so on)
HELP
PUT (object, object)

```
10 REM****THE BIANCO MANSION****
20 REM*****A TEXT ADVENTURE*****
30 CLS:PRINT @ 230,"THE BIANCO MANSION":
GOSUB 1470
40 REM***MAIN LOOP***
50 ST=ST-1:IF ST=0 THEN 1340
60 CLS:PRINT:PRINT"YOU ARE IN ";L$(L)
70 IF L=7 AND L(17)=0 THEN 1200
80 IF L=7 AND L(17)<>0 THEN 1320
90 FOR M=1 TO 4:IF P(L,M)=-1 THEN PRINT"
LOCKED DOOR TO THE ";MID$(D$,(5*M)-4,5)
100 S=0:PRINT:PRINT"YOU HAVE:";
110 FOR D=1 TO 18:IF D=6 OR D=9 OR D=10
OR D>14 THEN A$=" " ELSE A$=" A "
120 IF S>0 AND L(D)=0 THEN PRINT TAB(9);
130 IF L(D)=0 THEN PRINT A$;O$(D):S=S+1
140 NEXT D
150 PRINT
160 Z=0:PRINT"YOU CAN SEE:";
170 FOR D=1 TO 18:IF D=6 OR D=9 OR D=10
OR D>14 THEN A$=" " ELSE A$=" A "
180 IF L(D)=L AND Z>0 THEN PRINT TAB(11)
;
190 IF L(D)=L THEN PRINT A$;O$(D):Z=Z+1
200 NEXT D
210 IF Z=0 THEN PRINT" NOTHING MUCH"
220 IF G1=L THEN PRINT:PRINT:PRINT"  gas
p! a security guard":DC=DC-1:IF DC=0 THE
N 1340
230 PRINT:INPUT"WHAT NOW";N$
```

```
240 GOSUB 1400
250 IF G1=L AND T<>13 AND T<>14 AND T<27
 THEN 290
260 IF T>96 OR T<1 THEN PRINT:PRINT TAB(
12):MID$(P$,((RND(4)-1)*8)+1,8):GOTO 280
270 ON T GOSUB 330,330,410,410,410,500,5
60,620,640,670,680,680,810,810,900,910,9
40,940,1000,1040,1050,1050,1080,1110,111
0,1110,1110
280 FOR TT=1 TO 500:NEXT:IF T<3 THEN 310
290 IF G1=L THEN PRINT"THE GUARD FIGHTS
BACK..."
300 FOR TT=1 TO 1000:IF INKEY$="" THEN N
EXT ELSE TT=1000
310 GOTO 40
320 REM*****COMMANDS*****
330 REM
340 IF LEFT$(CT$,1)="N" THEN K=1
350 IF LEFT$(CT$,1)="S" THEN K=2
360 IF LEFT$(CT$,1)="E" THEN K=3
370 IF LEFT$(CT$,1)="W" THEN K=4
380 N=L:L=P(L,K)
390 IF L<1 THEN PRINT"YOU CAN'T GO THAT
WAY!":L=N
400 RETURN
410 DV=0:FOR CT=1 TO 54 STEP 3:IF MID$(O
$,CT,3)=LEFT$(CT$,3) THEN DV=CT
420 NEXT:CT=(DV+2)/3:IF DV=0 THEN T=100:
RETURN
430 IF S>4 THEN PRINT"YOU MUST DROP SOME
```

```
THING FIRST":RETURN
440 IF (CT=8 OR CT=13) AND S>2 THEN PRIN
T"TO CARRY THE ";O$(CT);", YOU CAN   ON
LY CARRY 2 OTHER ITEMS.":RETURN
450 IF L(CT)<>L THEN PRINT"IT IS NOT HER
E":RETURN
460 IF CT=3 OR CT=11 THEN PRINT"DON'T BE
 SILLY,IT'S TOO HEAVY.":RETURN
470 IF L(17)=3 THEN ST=25:PLAY"L1503C01C
03C01C03C01C03C01C":PRINT"THE ALARMS HAV
E SOUNDED, YOU DO NOT HAVE MUCH TIME.":L
(17)=0:RETURN
480 IF L(CT)=L THEN L(CT)=0:PRINT"YOU AR
E CARRYING THE":PRINT TAB(6);O$(CT):RETU
RN
490 IF L=3 AND CT=17 AND L(8)<>0 AND L(1
3)<>0 THEN PRINT"THEY ARE OUT OF REACH":
RETURN
500 REM*****DROP*****
510 DV=0:FOR CT=1 TO 54 STEP 3:IF LEFT$(
CT$,3)=MID$(O$,CT,3) THEN DV=CT
520 NEXT:CT=(DV+2)/3:IF DV=0 THEN T=100:
RETURN
530 IF S=0 THEN PRINT"YOU HAVE NOTHING T
O DROP":RETURN
540 IF L(CT)<>0 THEN PRINT"YOU DON'T HAV
E IT TO DROP.":RETURN
550 L(CT)=L:PRINT"YOU HAVE DROPPED THE":
PRINT TAB(6);O$(CT):RETURN
560 IF L(2)<>0 THEN PRINT"IT'S TOO DARK
```

```
TO READ!":RETURN
570 IF L(4)<>0 AND L(17)<>0 AND L(18)<>0
 THEN PRINT"THERE IS NOTHING HERE TO REA
D!"
580 IF L(4)=L THEN PRINT"READS: WHAT YOU
 NEED IS IN THE  STUDY"
590 IF L(18)=0 THEN PRINT"I'M SURE YOU H
AVE SEEN AN ATLAS BEFORE"
600 IF L(17)=0 THEN PRINT"your task is t
o steal them not  read them."
610 PRINT:PRINT"PRESS ENTER TO CONTINUE"
:INPUT L$:RETURN
620 IF (L(8)=0 OR L(13)=0) AND (CT$="LAD
" OR CT$="STO") AND L=3 THEN PRINT"YOU C
AN REACH THE MAP NOW":L(17)=3:RETURN
630 IF L=6 THEN PRINT"YOU CANNOT CLIMB O
VER THE FENCE":RETURN
640 IF L=6 AND L(6)=0 AND L(9)=0 AND CT$
="FEN" THEN PRINT"THE FENCE HAS BEEN CUT
, YOU CAN GET OUT BY GOING NORTH.":P(6,1
)=7:RETURN
650 IF L=6 AND L(6)<>0 AND L(9)=0 AND CT
$="FEN" THEN PRINT"YOU CUT THE FENCE BUT
 FORGOT IT WAS electrocuted!!!":SOUND 24
5,20:GOTO 1340
660 RETURN
670 PRINT"O.K. I CHANGED YOU INTO A FROG
..":GOTO 1340
680 DDV=0:FOR M=1 TO 4:IF P(L,M)=-1 THEN
 DDV=1
```

```
690 NEXT M:IF DDV=1 AND L(14)=0 THEN KY=
KY+1:IF KY=3 THEN KY=0:L(14)=37:PRINT"TH
E KEY WAS OLD AND YOU USED IT TOO MUCH,
IT HAS CRUMBLED INTO  DUST":RETURN
700 IF DDV<>1 OR L(14)<>0 OR CT$<>"DOO"
THEN 770
710 PRINT"WITH A CREAK,THE DOOR OPENS...
":PLAY"V31L120CDC"
720 IF L=19 THEN P(19,2)=18:P(18,1)=19
730 IF L=2 OR L=27 THEN P(2,1)=27:P(27,2
)=2
740 IF L=16 OR L=28 THEN P(28,2)=16:P(16
,1)=28
750 IF L=21 OR 22 THEN P(22,3)=21:P(21,4
)=22
760 RETURN
770 IF L=19 THEN PRINT"YOU ARE TRAPPED I
NSIDE THE VAULT":IF L(14)<>0 THEN 1330
780 IF DDV<>1 THEN PRINT"THERE IS NO DOO
R TO UNLOCK FOOL!":RETURN
790 IF L(14)<>0 THEN PRINT"YOU DON'T HAV
E THE KEY":RETURN
800 T=100:RETURN
810 IF GS=1 AND G1=L AND L(1)=0 AND LEFT
$(CT$,2)="GU" THEN 870
820 IF G1<>L THEN PRINT"THERE IS NO GUAR
D HERE!"
830 IF L(1)<>0 THEN PRINT"YOU HAVE NOTHI
NG TO FIRE WITH"
840 IF LEFT$(CT$,2)<>"GU" THEN PRINT"SHO
```

```
OT WHAT?"
850 IF L(1)=0 AND G1=L AND LEFT$(CT$,2)=
"GU" AND GS<>1 THEN PRINT"YOU KILLED HIM
 BUT THE GUNSHOTS HAVE ACTIVATED THE SON
IC ALARM. YOU HAVE LITTLE TIME TO ESCAPE
":ST=15:PLAY"O3BCBCBCBCBC":G1=RND(28)+2:
IF G1=L OR G1=28 THEN G1=G1-1
860 RETURN
870 IF RND(3)<2 THEN PRINT"YOU MISSED HI
M":RETURN
880 PRINT"YOU KILLED HIM":G1=RND(28)+2:I
F G1=L OR G1=28 OR G1=7 THEN 880
890 RETURN
900 PRINT"TIME PASSES...":FOR TT=1 TO 20
00:NEXT:RETURN
910 IF LEFT$(CT$,3)<>"WIN" OR L<>4 THEN
PRINT"DRINK WHAT?":RETURN
920 IF L=4 AND L(7)<>0 THEN PRINT"YOU HA
VE NOTHING TO DRINK FROM":RETURN
930 PRINT"YUK! IT TASTES AWFUL,BUT IT WA
S ONLY COOKING WINE!!!":RETURN
940 IF L(2)<>0 THEN PRINT"IT'S TOO DARK
TO LOOK AROUND OR SEARCH":RETURN
950 IF L=3 THEN PRINT"THE MAP IS ON THE
TOP SHELF":RETURN
960 IF L=21 THEN PRINT"YES, THAT IS A MO
NET ON THE FAR WALL, HOW OBSERVENT OF YO
U...":RETURN
970 IF L=2 THEN PRINT"THERE IS A LOCKED
DOOR WESTWARDSIT CAN ONLY BE OPENED BY T
```

```
HE    LETTER KNIFE":RETURN
980 IF L=30 THEN PRINT"THERE IS A NOTE O
N THE TABLE    ADDRESSED TO CATHERINE DE
 BIANCO, THE CONTESSA. IT SAYS THAT THEE
LECTRICIAN LEFT HIS GLOVES IN  THE GALLE
RY AND COULD HE HAVE   THEM BACK PLEASE.
":RETURN
990 PRINT"I'M LOOKING, BUT THERE'S LITTL
E TO SEE.":RETURN
1000 IF L(6)=0 THEN PRINT"THEY FIT YOU P
ERFECTLY":G$="YOU ARE WEARING GLOVES":L(
6)=35:RETURN
1010 IF L(15)=0 THEN PRINT"VERY NICE AND
 VERY VALUABLE, IT IS YOUR SIZE...":RETU
RN
1020 PRINT"YOU ARE WEARING ALL THAT YOU
OWN":RETURN
1030 RETURN
1040 PRINT"HELP YOURSELF, NO CLUES FROM
ME":RETURN
1050 PRINT:INPUT"HAVE YOU REALLY HAD ENO
UGH";A$
1060 IF LEFT$(A$,1)="Y" THEN PRINT"OK YO
U HAVE GIVEN UP...coward":SOUND 1,16:END
1070 RETURN
1080 IF LEFT$(CT$,3)="GUN" AND MID$(N$,5
,8)="SILENCER" AND L(12)=0 AND L(1)=0 TH
EN GS=1:PRINT"SILENCER ATTACHED, YOU CAN
 NOW  FIRE QUIETLY.":RETURN
1090 IF LEFT$(CT$,3)="LOC" AND MID$(N$,5
```

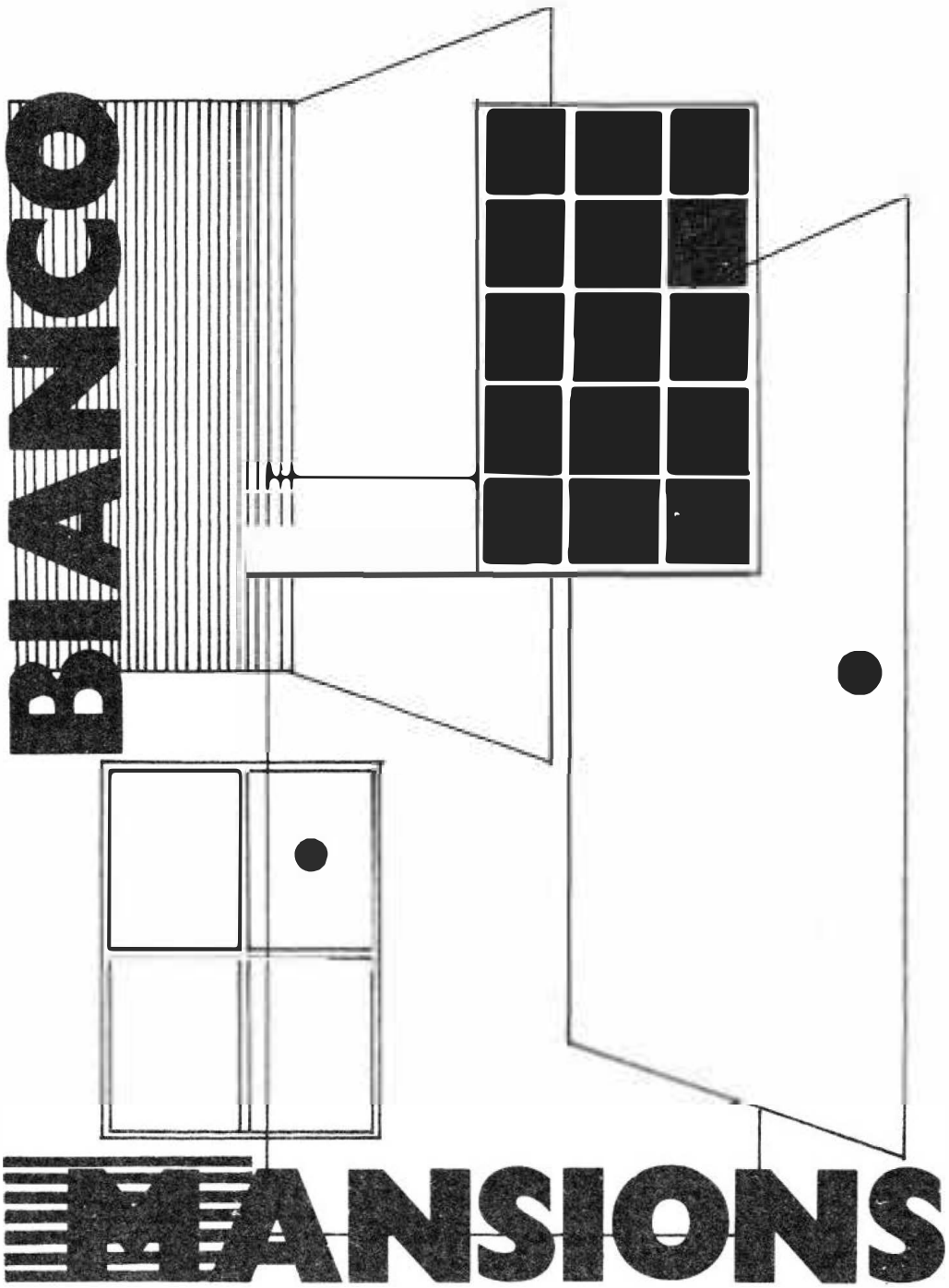
```
,3)="KNI" AND L=2 AND O(5)=0 THEN PRINT"
THE DOOR QUIETLY OPENS. JUST ENTER GO WE
ST":P(2,4)=3:P(3,3)=2:RETURN
1100 PRINT"YOU CAN'T PUT THAT THERE!":RE
TURN
1110 IF G1=L THEN 1130
1120 PRINT"THERE'S NO ONE NEAR YOU":RETU
RN
1130 INPUT"WITH WHAT";W$:IF LEFT$(W$,4)=
"WIRE" AND L(9)=0 AND RND(2)=1 THEN PRIN
T"THAT GOT HIM!":G1=0
1140 IF W$="ROPE" AND L(16)=0 AND RND(2)
=1 THEN PRINT"YOU FINISHED HIM OFF!":G1=
0
1150 IF LEFT$(W$,4)<>"WIRE" AND W$<>"ROP
E" THEN 1180
1160 IF G1=0 THEN G1=RND(28)+2:IF G1=28
OR G1=L OR G1=7 THEN 1160
1170 RETURN
1180 PRINT"YOU WILL HAVE TO FIGHT WITH Y
OURHANDS...":IF RND(5)=1 THEN PRINT"YOU'
VE KILLED HIM":G1=0:GOTO 1160
1190 RETURN
1200 REM*****SUCCESS*****
1210 FOR T=1 TO 2000:NEXT:CLS:PRINT:PRIN
T:PRINT"WELL DONE YOU ESCAPED WITH THE
MAP. YOUR TASK IS COMPLETED.        YOU C
AN NOW GO A FREE MAN."
1220 PLAY"L804CEDFGAL20BAB":FOR T=1 TO 2
000:NEXT T
```

```
1230 PMODE 3,1:PCLS:SCREEN 1,0
1240 FOR T=1 TO 85:CIRCLE(3*T,80),T
1250 NEXT T
1260 FOR T=1 TO 85:CIRCLE(255-(3*T),110)
,T,3
1270 NEXT
1280 FOR T=1 TO 47:CIRCLE(128,(4*T)),T,2
,0.8
1290 NEXT T
1300 GOTO 1300
1310 END
1320 REM****ESCAPED BUT WITHOUT MAP****
*
1330 FOR T=1 TO 2000:NEXT T:CLS:PRINT @
224,"YOU ESCAPED BUT WITHOUT THE MAP THE
REFORE, YOU MUST ENTER AGAIN":FOR T=1 TO
 2000:NEXT:RUN
1340 PRINT"  OOOOO   OOOOO   OOOOO   O
  ":PLAY"L1602G"
1350 PRINT"  O       O   O     O     O":
PLAY"D"
1360 PRINT"  OOOO    OOOOO     O     O":
PLAY"O1A"
1370 PRINT"  O       O   O     O     O":
PLAY"E"
1380 PRINT"  O       O   O   OOOOO   OOO
OO":PLAY"C"
1390 PRINT:PRINT:GOTO 1340
1400 REM****COMMAND HANDLING****
1410 T=1:IF LEN(N$)<3 THEN T=100:RETURN
```

```
1420 B$=MID$(C$,T,3):IF B$=LEFT$(N$,3) T
HEN T=(T+2)/3:GOTO 1450
1430 IF T>97 THEN 1460
1440 T=T+3:GOTO 1420
1450 FOR CT=LEN(N$) TO 1 STEP-1:IF MID$(
N$,CT,1)=" " THEN CT$=MID$(N$,CT+1,3) EL
SE NEXT CT
1460 RETURN
1470 REM***INITIALISATION***
1480 DIM O$(18),L$(30),P(30,4),L(18),A(1
8)
1490 FOR T=1 TO 18:READ O$(T):NEXT
1500 FOR T=1 TO 30:READ L$(T):NEXT
1510 FOR T=1 TO 18:READ L(T):NEXT
1520 DATA GUN,TORCH,WINE-CASK,PERFUMED-L
ETTER,LETTER KNIFE,GLOVES,CUP,STOOL,WIRE
CUTTERS
1530 DATA MONEY,STATUE,SILENCER,LADDER,K
EY,RING,ROPE,MAP,ATLAS
1540 DATA "THE MAIN ENTRANCE,THE COUNT A
ND CONTESSA ARE AWAY,   YOU CAN ENTER",T
HE RECEPTION AREA,THE STUDY,THE KITCHENS
,THE COURTYARD,THE SHADOW OF THE ELECTRO
CUTED FENCE,"SAFETY.FREE AT LAST!",A GRE
AT HALLWAY,THE BOILER ROOM
1550 DATA THE WINE CELLAR,AN OAKEN CORRI
DOR,AN UNKEMPT GARDEN,THE BILLIARD ROOM,
THE LIBRARY,A DIMLY-LIT STOREROOM,THE MA
GNIFICENT DINING ROOM,A SPACIOUS STUDIO
1560 DATA REACH OF THE COUNT'S VAULT, TH
```

```
E COUNT'S VAULT,THE SMOKING ROOM,A ROOM
FILLED WITH PAINTINGS,A DARKENED BEDROOM
,THE SERVANTS QUARTERS,THE CARD ROOM,THE
 MASTER BEDROOM,A LARGE CUPBOARD
1570 DATA AN AUSTERE LIVING    ROOM,THE
NURSERY,"THE DOORWAY LEADING TO THE CONT
ESSA'S ROOM",THE CONTESSA'S ROOM
1580 DATA 0,2,4,27,8,21,25,13,9,19,20,30
,5,10,13,11,34,11
1590 D$="NORTHSOUTHEAST WEST "
1600 C$="GO MOVTAKGETSTEDROREACLICUTCHAD
PEUNLFIRSHOWAIDRISEALOOWEAHELCLUQUIFUTHI
TFIGATTKIL"
1610 O$="GUNTORWINLETKNIGLOCUPSTOWIRMONS
TASILLADKEYRINROPMAPATL"
1620 P$="PARDON? WHAT?   RUBBISH!REPHRAS
E"
1630 L=1:L(1)=0:K=0:ST=10000:KY=0:GS=0
1640 G1=RND(28)+2:IF G1=28 OR G=7 THEN 1
640
1650 DC=RND(6)+4
1660 FOR X=1 TO 30:FOR Y=1 TO 4:READ P(X
,Y):NEXT:NEXT
1670 DATA 2,0,0,0,-1,1,4,0,0,0,2,0,0,10,
9,2,0,15,6,0,0,0,5,0,0,0,0,0,15,24,0,21,
20,8,13,9
1680 DATA 4,0,9,1,0,27,14,12,17,0,11,16,
16,0,0,9,26,0,18,11,5,8,0,21,-1,23,12,26
,22,12,15,0
1690 DATA 19,20,18,14,0,-1,0,0,18,9,0,0,
```

```
2,17,8,-1,0,17,-1,28,16,23,0,17,8,0,26,1
2,0,0,0,27
1700 DATA 6,14,0,24,11,-1,25,13,29,16,0,
0,0,28,0,30,0,0,29,0
1710 RETURN
```



PLAY YOUR CARDS RIGHT

Fancy a gamble? Why not place your bets on this fascinating game of cards. The rules are quite easy to follow; all you have to do is decide after the first, 'base' card has been dealt, whether the next card will be higher or lower than the last. If you guess correctly you will gain a point; and you can continue playing until you make an incorrect guess.

Use the up and down cursor keys to make your choice — and when you have finally decided, press the space bar. Good luck!

```
10 DIM T$(3),C$(12):GOSUB570
20 CLS:PRINT@3,CHR$(138);:PRINT"PLAY  YO
UR  CARDS  RIGHT"::PRINTCHR$(133)
30 LI$=STRING$(24,131)
40 CA$=CHR$(142)+STRING$(3,128)+CHR$(141
)
50 CR$=CHR$(138)+STRING$(3,128)+CHR$(133
):CD$=CHR$(139)+STRING$(3,128)+CHR$(135)
60 PRINT@35,CHR$(139)+LI$+CHR$(135)
70 PRINT@64,"PLEASE INPUT YOUR NAME";:IN
PUT NA$
80 PRINT@64,"":PRINT@96,"                ";
90 FORI=321TO345STEP6:PRINT@I,CA$:NEXT
100 GOSUB110:GOTO150
110 FORI=353TO380STEP6:PRINT@I,CR$:NEXT
120 FORI=385TO411STEP6:PRINT@I,CR$:NEXT
130 FORI=417 TO445STEP6:PRINT@I,CR$:NEXT
140 RETURN
150 FORI=449TO475STEP6:PRINT@I,CD$:NEXT
160 R=RND(13)-1:P=R
170 L=RND(4)-1:Z=L
180 GOSUB190:GOTO240
190 PRINT@196,"YOUR BASE CARD IS":PRINT@
164,NA$
200 PRINT@153,CA$
210 PRINT@185,C$(R):PRINT@213,"OF":PRINT
@249,T$(L)
220 PRINT@281,CD$
230 RETURN
240 PRINT@488,"lower";
```

```
250 PRINT@494,"<  >";
260 PRINT@498,"higher";
270 GOSUB280:GOTO330
280 GOSUB560:IF A$="↑" THEN 290 ELSE IF
A$=CHR$(10) THEN 310 ELSE IF CH=0 AND A$
=CHR$(32) THEN 280 ELSE IF CH>0 AND A$=C
HR$(32) THEN RETURN ELSE 280
290 CH=1:FORI=1TO5:PRINT@499,"HIGHER";:F
ORS=1TO500:NEXTS:PRINT@498,"higher";:FOR
S=1TO500:NEXTS:NEXTI
300 GOTO280
310 CH=2:FORI=1TO5:PRINT@488,"LOWER";:FO
RS=1TO500:NEXTS:PRINT@488,"lower";:FORS=
1TO500:NEXTS:NEXTI
320 GOTO280
330 PRINT@353,"      ";:R=RND(13)-1:L=RND
(4)-1:IF R=P AND L=Z THEN 330 ELSE PRINT
@353,C$(R);
340 PRINT@385," OF  ";:PRINT@417,T$(L);
350 GOSUB360:GOTO440
360 IF CH=1 AND R<P THEN PRINT@69,"bad l
uck":FORI=1TO2000:NEXT:PLAY"T50O1L30BAGF
EDCCC":GOTO 620
370 IF CH=1 AND R>P THEN PRINT@69,"good
guess":SC=SC+1
380 IF CH=1 AND R=P THEN PRINT@69,"lucky
"
390 IF CH=2 AND R>P THEN PRINT@69,"bad l
uck":FORI=1TO2000:NEXT:PLAY"T50L30O1BAGF
EDCCC":GOTO620
```

```
400 IF CH=2 AND R<P THEN PRINT@69,"good
guess":SC=SC+1
410 IF CH=2 AND R=P THEN PRINT@69,"lucky
"
420 PRINT@265,"SCORE=";SC;
430 RETURN
440 P=R:Z=L:CH=0:R=RND(13)-1:L=RND(4)-1:
IF R=P AND L=Z THEN 440
450 GOSUB280:PRINT@359,"      ";:PRINT@35
9,C$(R);:PRINT@391," OF   ";:PRINT@423,T$
(L);:GOSUB360
460 CH=0:P=R:Z=L:R=RND(13)-1:L=RND(4)-1:
IF R=P AND L=Z THEN 460
470 GOSUB280:PRINT@365,"      ";:PRINT@36
5,C$(R);:PRINT@397," OF   ";:PRINT@429,T$
(L);:GOSUB360
480 CH=0:P=R:Z=L:R=RND(13)-1:L=RND(4)-1:
IF R=P AND L=Z THEN 480
490 GOSUB280:PRINT@371,"      ";:PRINT@37
1,C$(R);:PRINT@403," OF   ";:PRINT@435,T$
(L);:GOSUB360
500 CH=0:P=R:Z=L:R=RND(13)-1:L=RND(4)-1:
IF R=P AND L=Z THEN 500
510 GOSUB280:PRINT@377,"      ";:PRINT@37
7,C$(R);:PRINT@409," OF   ";:PRINT@441,T$
(L);:GOSUB360
520 PRINT@66," WELL DONE ";NA$;:PRINT@99
,"YOU HAVE PAST THE";:PRINT@131,"LAST ST
AGE":FORI=1TO5000:NEXT
530 FORI=1TO5:PLAY"T2550"+STR$(I)+"V28CD
```

```
EFGABAB":NEXT:PRINT@66,"":PRINT@99,"
                ";:PRINT@131,"                ";
540 GOSUB110:GOSUB190
550 GOSUB280:GOTO330
560 A$=INKEY$:IF A$="" THEN 560 ELSE RET
URN
570 FORI=0TO3:READ T$(I):NEXT
580 DATA CLUBS,DIAMD,HEART,SPADE
590 FORI=0TO12:READ C$(I):NEXT
600 DATA ACE,TWO,THREE,FOUR,FIVE,SIX,SEV
EN,EIGHT,NINE,TEN,JACK,QUEEN,KING
610 RETURN
620 FORI=64TO448STEP32:PRINT@I,"":NEXT
630 L$=STRING$(30," "):PRINT@480,L$;
640 PRINT@164,"YOU SCORED A TOTAL OF";SC
;
650 PRINT@230," HARD LUCK ";NA$;
660 PRINT@294," YOU GUESSED WRONG!"
670 PRINT:PRINT:PRINT" DO YOU WANT ANOTH
ER GO (Y/N)?"
680 GOSUB560
690 IF A$="Y" THEN SC=0:CH=0:P=0:Z=0:R=0
:L=0:GOTO20
700 IF A$="N" THEN END ELSE 680
```

THE ANAGRAM'S REVENGE

Once, long ago, a young person was playing anagrams with his Dragon. He went out for a cup of tea and when he came back the anagram program had been changed.

This is the program that the boy found. The idea is to guess the anagram; however, the computer will change one of the letters in the anagram so that it is harder. If you get it wrong the computer will change another letter until all but the last letter has been changed. Then the computer will tell you the anagram. (The changed letters are the ones that are inverse.)

```
10 REM*****THE ANAGRAMS REVENGE***
30 REM*****************************
40 CLS:PRINT:PRINT:PRINT"        THE ANAGR
AMS REVENGE"
50 PRINT@234,"PLEASE WAIT"
60 GOSUB 510
70 PLAY"O1GAC"
80 GOSUB 430
90 PLAY"O1CAG"
100 FOR A=1 TO LEN(W$)-1:TIMER=0
110 GOSUB 340'INPUT GUESS
120 GOSUB 320'CHECK GUESS
130 PLAY"ABGCE"
140 NEXT A:A=0
```

```
150 REM**************************
160 REM FAILURE
170 CLS:PRINT@256,"   THE WORD WAS "+W$
180 PRINT @256+35," MUFFED IT !"
190 A=A+1:IF A=8 THEN A=1
200 FOR N=0 TO 5:NEXT N:SOUND A*25,1
210 IF INKEY$<>"" THEN 220:ELSE GOTO 150
220 RUN
230 REM *** GOT IT********
240 CLS
250 A=1
260 REM
270 PRINT @256+5,CHR$(A)+" GOT IT !"
280 A=A+1:IF A=9 THEN A=1
290 FOR N=0 TO 5:NEXT N:SOUND A*25,1
300 IF INKEY$<>"" THEN 310 ELSE GOTO 260
310 RUN
320 IF G$=W$ THEN 230
330 RETURN
340 CLS
350 PRINT"GUESS NUMBER ";A
360 PRINT
370 B=RND(LEN(S$)):MID$(S$,B,1)=CHR$(RND
(26)+96)
380 PRINT "THE ANAGRAM IS ";S$
390 PRINT
400 INPUT "WHAT IS YOUR GUESS":G$
410 IF TIMER/50>90 THEN PRINT"OUT OF TIM
E, NEXT GUESS...":A=A+1
420 RETURN
```

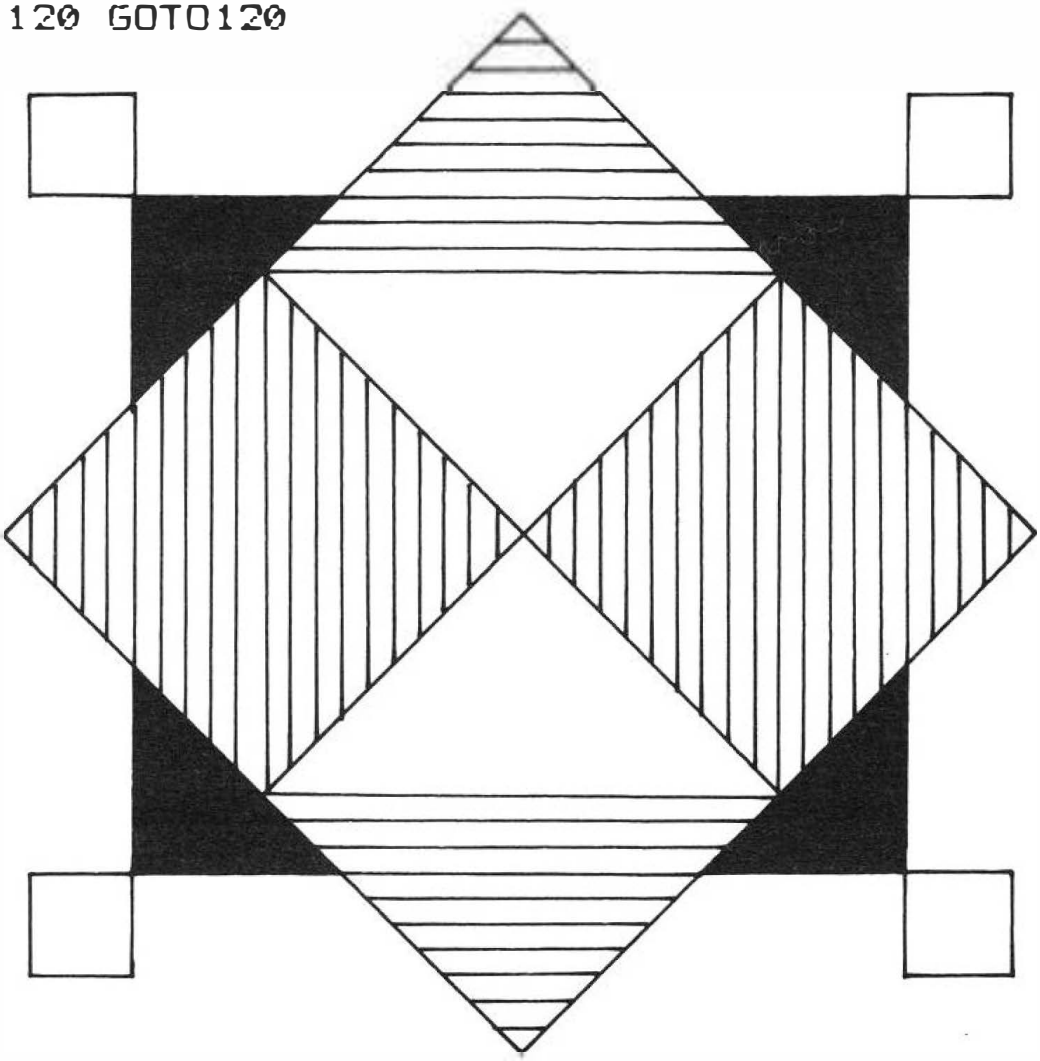
```
430 L=LEN(W$)
440 S$=LEFT$("                    ",L-1)
450 FOR I=1 TO L
460 R=INT(RND(L))
470 IF MID$(S$,R,1)>" " THEN 460
480 S$=LEFT$(S$,R-1)+MID$(W$,I,1)+MID$(S
$,R+1)
490 NEXT I
500 RETURN
510 READ C
520 FOR N=1 TO RND(INT(C))
530 READ W$
540 NEXT N
550 RETURN
560 DATA 40
570 REM*******DATA FOR ANAGRAMS*
590 DATA PROGRAM,SOFTWARE,BASIC,HARDWARE
,PRINTOUT
590 DATA COMPUTER,GRAPHICS,KEYBOARD,PRIN
TER
600 DATA HELP,SILICON,ARCADE,MEMORY,TOOL
KIT
610 DATA DISASSEMBLER,INTERFACE,VIDEO
620 DATA DIGITAL,PERIPHALS,CENTRONICS,PA
RALLEL
630 DATA SERIAL,HIRES,CHRACTER,MACHINE,N
IBBLE,BYTE,BIT,PEEK,POKE,ASCII,PLAY,SOUN
D,NEW,AT
640 DATA COMPUTER,CASSETTE-RECORDER,AUDI
O,MOTOR,DISK
```

MOIRE, CURVES AND A MOSAIC

The next two programs do exactly what their titles suggest. Moire is a famous pattern generated by nearly all high-res microcomputers nowadays.

```
10 REM*******MOIRE/CURVES********
20 PMODE 3,1:SCREEN1,0:PCLS
30 FOR N=0 TO 190
40 POKE &HB2,RND(255)
50 LINE (0,N)-(N,250-N),PSET
60 LINE (255,N)-(N,250-N),PSET
70 NEXT N
80 PLAY "O1T1CACACA"
90 FOR N=0 TO 500:NEXT N
100 PMODE 3,1:SCREEN1,0
110 PCLS
120 FOR N=0 TO 190 STEP2
130 COLOR 4
140 LINE(0,190-N)-(250,N),PSET
150 COLOR 3
160 LINE (190-N,0)-(N,250),PSET
170 NEXT N
180 PLAY "O1T1ABCABCABC"
190 FOR N=0 TO 500:NEXT N
200 RUN
```

```
10 REM*** MOSAIC ***
20 PMODE 3,1:SCREEN 1,0:PCLS
30 FOR N=30 TO 220 STEP 2
40 POKE &HB2,N
50 PLAY "T25501AB"
60 LINE(30,N)-(250-N,00),PSET
70 LINE (N,0)-(220,N),PSET
80 LINE (220,N)-(250-N,220),PSET
90 LINE (30,N-30)-(N,220),PSET
100 NEXT N
110 SOUND 180,4
120 GOTO120
```



ROCK SCISSORS PAPER

This is a high-res version of the famous game. Most of you probably know the rules already — if not, then read on.

Originally this game was played with your hands, and the idea was to beat your opponent by trumping his choice — the choice being between rock, scissors and paper. Both players simultaneously announce what they have chosen, miming the object at the same time — a clenched fist for rock, two fingers for scissors and a flat palm for paper.

The winner of each game is determined by how the objects interact. Thus, if one player has paper and the other scissors, the second player wins because scissors cut paper; similarly rock beats scissors and paper beats rock.

Once you have decided your choice push the key which holds the first letter of the object (S for scissors, P for paper, and R for rock). The computer will tell you what it chose, and what the result of that game was.

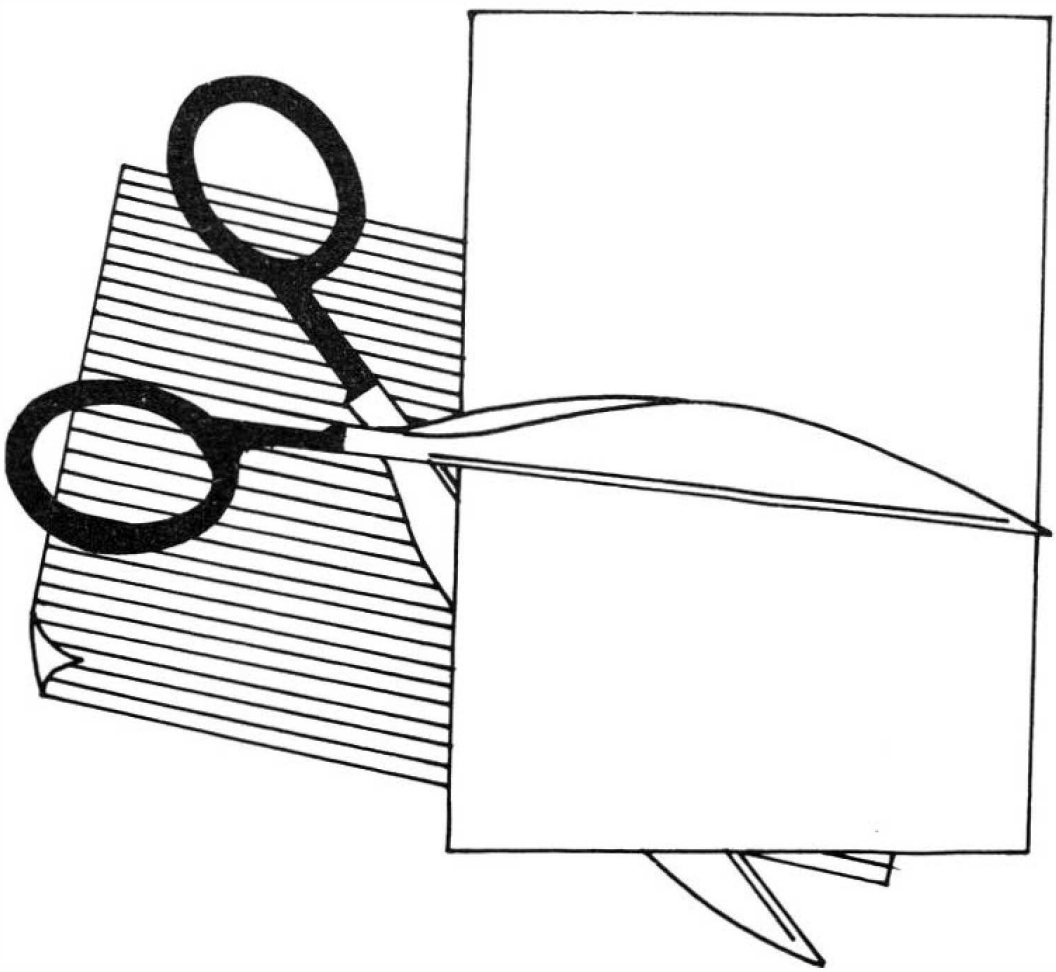
```
10 REM****ROCK SCISSORS PAPER
30 REM**********GRAPHICS**********
40 ME=0:YO=0
50 A$="E2R10E2R10E2R10F6R20F3R4G3L4G2L4G
3L40H10L3BR20D6U6R3D3L3F3BR3R3U3L3D3R3BR
3R3L3U3R3L3D3R3BR3U6D4E2G2F2
60 B$="R40D60L40U60D20BR5D6U6R3D3L3R3U3B
R3D6U3R3D3U6L3R3BR3D6U6R3D3L3R3U3BR3D6R3
```

```
L3U3R3L3U3R3BR3D6U6R3D3L3F3
70 C$="R46E1R1E1R1E1R1L48G1L1G1L1G1L1F1R
1F1R1F1R1R48H1L1H1L1H1L1BU4BD1BR5U10R22D
10L22BU4BR4U6R12D6L12BL4D10R22D10L22U10B
D4BR4D6R12U6L12BD15BL50L3D3R3D3L3R3BR3U6
R3L3D6R3BR3U6D6BR3R3U3L3U3R3BR3R3L3D3R3D
3L3R3BR3R3U6L3D6R3BR3U6R3D3L3F3BR3R3U3L3
U3R3
80 REM******************************
90 CLS:PRINT"        rock scissors paper"
100 PLAY"T05O2EP100EP100EP10EP100EP100EP
100EP100GCP100DP100EP100FP100FP100FP10EP
100EP100EP10EP100DP100DP100EP100DP10EP10
0EP100EP10EP100EP100EP100P10EP100GP100CP
100DP100EP10FP100FP100FP10EP100EP100EP10
0GP100GP100FP100DP100CP1
110 A=RND(3)
120 PRINT:PRINT:PRINT:PRINT" I'VE GUESSE
D MY THING NOW YOU   GUESS YOURS"
130 PRINT" PRESS (R/S/P) ?
140 Q$=INKEY$:IF Q$="" THEN 140:ELSE IF
Q$<>"R" AND  Q$ <>"S" AND Q$ <>"P" THEN
140
150 PMODE 4,1:SCREEN1,0:PCLS
160 DRAW"BM90,5D12U12F6E6D12U12BR3BD6D6R
6U6D12L6BR12BU6U6R6D6L6R6BR3U6R6D2G1L4R4
F1D2L6R6BR6U6L3R6L3D12L6R6U6BR6U6R6L6D3R
6L6D3R6BR3U6R6L6D6R6BR6U6L3R6
170 LINE(0,090)-(255,090),PSET
180 DRAW"BM90,100D6R6U6D12L6R6BR3U6R6D6L
```

```
6R6BR3U6D6R6U6D6BR3U6R6D3L6F3BR9U6R6D6L6
R6BR3U6R6D2G1L4R4F1D2L6R6BR3R4U6L4R6L2D6
BR5U6R6L6D3R6L6D3R6BR3U6R6L6D6R6BR6U6L3R
6
190 ON A GOSUB 380,400,420
200 IF Q$="R" THEN  GOSUB 450
210 IF Q$="P" THEN GOSUB 470
220 IF Q$="S" THEN GOSUB 490
225 FOR N=0 TO 100:NEXT N
230 IF Q$="R" AND A=1 THEN CO=1
240 IF Q$="S" AND A=3 THEN CO=2
250 IF Q$="P" AND A=2 THEN CO=3
260 IF Q$="S" AND A=1 THEN CO=4
270 IF Q$="P" AND A=1 THEN CO=5
280 IF Q$="P" AND A=3 THEN CO=6
290 IF Q$="R" AND A=3 THEN CO=7
300 IF Q$="S" AND A=2 THEN CO=8
310 IF Q$="R" AND A=2 THEN CO=9
320 IF CO=1 OR CO=2 OR CO=3 THEN D=1
330 IF CO=4 OR CO=6 OR CO=9 THEN D=2
340 IF CO=5 OR CO=7 OR CO=8 THEN D=3
350 FOR N=0 TO 1000:NEXT
360 GOTO 530
370 IF Q$="S" AND A= 2 THEN CO=2
380 DRAW"BM90,50"+A$
390 RETURN
400 DRAW"BM90,20"+B$
410 RETURN
420 DRAW"BM90,50"
430 DRAW C$
```

```
440 RETURN
450 DRAW"BM90,140"+A$
460 RETURN
470 DRAW"BM90,120"+B$
480 RETURN
490 DRAW"BM90,140"
500 DRAW C$
510 RETURN
520 PMODE 4,1:SCREEN1,0:PCLS
530 X$="D10E5H5BR8D10U10R5D5L5R3D2R2D3BR
3U10R5D5L5R5D5BR3U10D10E5F5U10"
540 Y$="R4L2D10L2R4BR10U10D10E5F5U10BR3R
5D10L5U10R5BR3D10U10R5D10"
550 Z$="D5R5U5D10BR3U10R5D10L5R5BR3U10D1
0R5U10D10BR10U10D10E5F5U10BR3D10R5U10L5R
5BR3D10U10R5D10
560 IF D=1 THEN PCLS:DRAW"BM100,100"+X$
570 IF D=2 THEN PCLS:DRAW"BM100,100"+Y$
580 IF D=3 THEN PCLS:DRAW"BM100,100"+Z$
590 PLAY"V30L8GECGECGECC"
600 Q$=INKEY$:IF Q$="" THEN GOTO 600
610 IF D=2 THEN ME=ME+1
620 IF D=3 THEN YO=YO+1
630 CLS
640 PRINT:PRINT:PRINT
650 IF YO=10 THEN CLS:FOR N=0 TO 1000 :C
LS:PRINT @100," YOU won!":NEXT:END
651 IF ME=10 THEN CLS:FOR N=0 TO 1000 :C
LS:PRINT @100," I win";:NEXT:END
660 PRINT"   THE score IS ";:IF ME>YO TH
```

```
EN PRINT ME;"----";YO;"          TO ME":ELS
E PRINT YO;"----";ME;"           TO YOU"
670 PRINT @385," press any key to contin
ue":Q$=INKEY$:IF Q$="" THEN 670:ELSE  FO
R N=0 TO 1000:NEXT:CLS:PRINT"  rock scis
sors paper ":GOTO 110
```

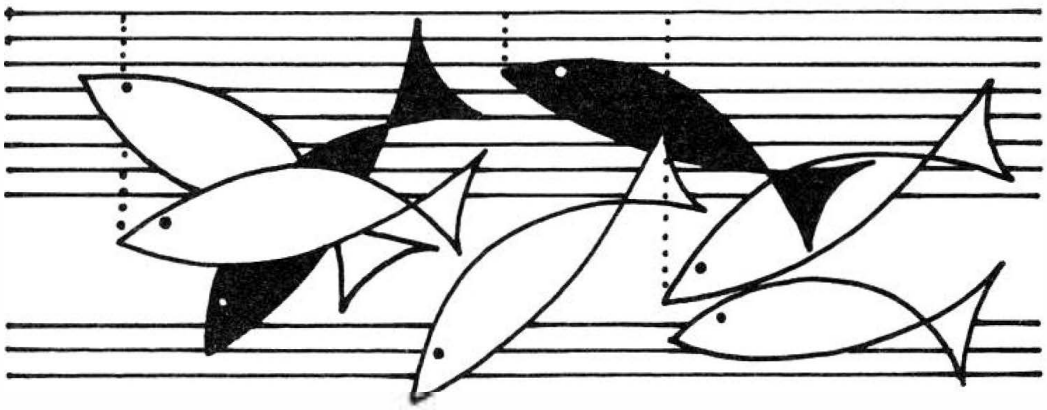


FISH FUN

Angling is one of the most popular sports, but why go to the riverside when you can type in this game.

Featuring some excellent high-resolution graphics, this game relies on a certain amount of skill as well as luck in trying to catch the fish that move along the bottom of the screen.

When the game starts, you will be asked how long you wish to fish for. The computer scales down your answer to approximately four casts per hour. The screen shows a fisherman with his rod and line. Pressing the up and down cursor keys moves the line up and down, so that the hook is at the depth you think will yield the greatest catch. When you are satisfied with your positioning, press the space bar and see if your calculations were correct.



```
10 PCLEAR6:DIM F(7),FI(7),FD(7),H(2),HI(
2),FA(8),FB(8)
20 CLS:PRINT"                FISH  FUN":PRINT
"              ****==****"
30 PRINT:PRINT:PRINT:PRINT"  HOW LONG DO
 YOU WANT":PRINT" TO STAY FISHING (1-12)
HRS";
```

```
40 INPUT A$
50 SD=VAL(A$):IF SD>12 OR SD<1 THEN 40
60 SS=SD*4
70 PCLS:DRAW"BM5,5;G1D1G1D3G1D2F1D4F1D1G
2R6H2U1E1U4E1U2H1U3H1U1H1":PAINT(5,7):GE
T(0,0)-(10,25),FA,G
80 PCLS
90 DRAW"BM3,3;D4E2R1F1R4F1R2E1R3E1R1E1H1
L1H1L3H1L2G1L4G1L1F2H4D2":PAINT(10,5):PA
INT(4,5):PSET(18,4,0)
100 GET(0,0)-(20,8),F,G
110 PCLS:DRAW"BM2,3;E1R1E1R3E1R2F1R4F1R1
G2E4D6H2L1G1L4G1L2H1L3H1L1H1E1":PAINT(10
,4):PAINT(19,3):PSET(4,3,0)
120 GET(0,0)-(20,8),FD,G
130 PCLS:DRAW"BM4,2;D6G1L2H1U1":GET(0,0)
-(6,10),H,G
140 PCLS:PMODE4:CLS:PRINT@266,"PLEASE WA
IT"
150 FORX=5TO181STEP12:CIRCLE(X,120),6,1,
1,0,.5:NEXT
160 DRAW"BM255,98;L90D1L2D1L1G2D2R1L1G1F
2D2F1L1R1F1D2R1D1F1D2F2R1D1F4L1D2R1D1R1F
2R3D2R2D3G2D2F2D2L1F1D2G1D3F2D2G2D1L2F1D
4L1D1G1D1R1G3D2R1F2D1G4D2L1D2L1G1L6H3L1D
1G1L3D1L3H1L3G2L1H1L4U1"
170 DRAW"L2G1L1G1D1H2L5H1L7D1H2L1G1L5G1L
2D1H4U1L2G2D2L3G1L2H1L4G4L3U1L2H1L3G2L5D
2L3U1L2D2L4H1L2H3L1G4L3D1L2G2L3U1L1D2L3U
1L2D1L1R3"
```

```
180 PAINT(250,100)
190 DRAW"BM210,98;E10H10R3F10E10R3G10F10
L3H10G10;BM210,78;R7F1R3F1R2E1R3E1R7"
200 DRAW"BM210,79;L2G1L3G1L2H1L4G1L3R3H3
F3G2D2G1D4G1D2L1DG1L1D2R1D2L5U2L3G2L6H1U
2E1R3U1R3E1U2R1D1R1D1R3F1R2F2"
210 DRAW"BM180,90;H2U2E1U2E1U2E1U2E1U2E1
U1E1U1R1U1E2R1U1R3E1R4D1U1R5F1R4E2R1E1U2
H1U3R1U2H1U3H1U4E1U2E1U3E2U1R1E1R1U1R3E1
U3H1R3E1U3D3G1L8"
220 DRAW"H1U1E1H1U1R2E1G1H1L1E1U1L2U1E3R
1L2H1U2H1L1H1L1U1E1R1E1R1F1R1F1R1F1R2F1R
2F1R1F1R1F1R1F1R1D1F1D1F1L2H1L2H1L2H1L2H
1L2H2L2H2
230 PAINT(220,15)
240 DRAW"U4E2R1E2R1E1R1F1R2F2R2F1R2D2F1D
2L1D2G1D6"
250 DRAW"F1D1G1D2G1D5R1F1D1F1D1F1D3F1D3F
1D2F1D1F1D2G1D1G1D1G1L1D1G1L1D1G1L1D1G1L
1D1G2L1D1G2D1F1U1H1D1H3U1H3F1E3U1R1E2U1R
1E1U1R1E2R1L1E1G1U2"
260 DRAW"H1U2H1U3H1U1H1U1H1U1H2U1E1U1R1E
1R1E1R2:BM234,55;D3F1D7G1D6G1D1G1L1G1L1
270 LINE(235,65)-(250,70),PSET:LINE-(248
,72),PSET:LINE-(236,69),PSET
280 LINE(210,56)-(80,12),PSET:LINE(81,13
)-(210,58),PSET
290 LINE(210,57)-(80,13),PSET:LINE(235,6
6)-(250,71),PSET:LINE(235,67)-(250,72),P
SET
```

```
300 CLS:SCREEN1,1
310 GOTO350
320 FORI=0TO130:PUT(I,Y1)-(I+20,Y1+8),F,
OR:PUT(I,Y1)-(I+20,Y1+8),F,PSET:NEXT
330 FORI=130TO0STEP-1:PUT(I,Y1)-(I+20,Y1
+8),FD,OR:PUT(I,Y1)-(I+20,Y1+8),FI,AND:N
EXT:CA=0:RETURN
340 PUT(X-4,Y-8)-(X+2,Y+2),H,OR:CA=1:RET
URN
350 X=80:Y=160
360 IF CA=1 THEN 400 ELSEGOSUB550:IF A$=
"A" THEN 370 ELSE IF A$=CHR$(10) THEN 38
0 ELSE IF A$=CHR$(32) THEN 390 ELSE 360
370 IF Y=136 THEN GOTO 360 ELSE LINE(X,Y
)-(X-2,Y),PRESET:Y=Y-1:LINE(X,Y)-(X-2,Y)
,PSET:GOTO360
380 IF Y=180 THEN GOTO 360 ELSELINE(X,Y)
-(X-2,Y),PRESET:Y=Y+1:LINE(X,Y)-(X-2,Y),
PSET:GOTO360
390 LINE(X,Y)-(X-2,Y),PRESET:PUT(X-4,13)
-(X+2,23),HI,AND:FORI=12TOY:PSET(X,I):FO
RA=1TO50:NEXTA:NEXT:GOSUB340:GOTO360
400 R=RND(34):Y1=134+R:FORI=-8TO2:IF Y1-
Y=I THEN 430 ELSE NEXT
410 GOSUB320
420 PUT(X-4,Y-8)-(X+2,Y+2),HI,AND:FORI=Y
 TO20STEP-1:PRESET(X,I):FORA=1TO10:NEXTA
:NEXT:PUT(X-4,I-8)-(X+2,I+2),H,OR:DD=DD+
1:IF DD=SS THEN 500 ELSE PLAY"T25504BAED
CO2BAEDCO3BAEDC":GOTO360
```

```
430 GOSUB490:PUT(X-4,Y-8)-(X+2,Y+2),HI,A
ND:FORI=Y TO95STEP-1:PRESET(X,I):FORA=1T
050:NEXTA:NEXT:FORI=95TO20STEP-1:PUT(X-5
,I)-(X+5,I+25),FA,OR:PUT(X-5,I)-(X+5,I+2
5),FB,AND:NEXT:PUT(X-4,I-8)-(X+2,I+2),H,
OR:CA=0
440 SC=SC+1:C=C+5
450 DRAW"BM"+STR$(C)+",2;D8"
460 PLAY"T255O3CDEAB02CDEAB04CDEAB"
470 DD=DD+1:IF DD=SS THEN 500
480 GOTO360
490 FORI=0TO X-20:PUT(I,Y1)-(I+20,Y1+8),
F,OR:PUT(I,Y1)-(I+20,Y1+8),FB,AND:NEXT:F
ORI=1TO10000:NEXT:RETURN
500 FORJ=31TO20STEP-1:PLAY"V"+STR$(I)+"O
3CDEAB02CDEAB04CDEAB":NEXT:CLS:PRINT"
         FISH  FUN
****==***":PRINT:PRINT:PRINT"  YOU CAUGH
T";SC;"FISH":PRINT"  IN";SS/4;"HOUR/S"
510 PRINT:PRINT:PRINT"DO YOU WANT ANOTHE
R GO? (Y/N)"
520 GOSUB550:IF A$="Y" THEN SC=0:SS=0:CA
=0:DD=0:C=0:PLAY"V30":GOTO20
530 IF A$="N" THEN CLS:END
540 GOTO520
550 A$=INKEY$:IF A$="" THEN 550 ELSE RET
URN
```

How To Write Better Programs

INVENTING AND DEVELOPING YOUR OWN GAMES PROGRAMS

By Series Editor, Tim Hartnell

It's all very well spending your time typing in programs like those in this book, but there is sure to come a time when you decide you'd like to develop some games programs of your own. In this section of the book, I'd like to discuss a few ideas which may help you write games which you'll both enjoy developing and — more importantly — you and your friends will enjoy playing.

HAVE A CLEAR GOAL IN MIND

Although in many (perhaps most) cases, your computer program will take on a life of its own as you write it, developing away from the concept you had in mind when you started programming, it is important at the outset to have a pretty good idea of what your game will involve.

This is not as obvious a suggestion as you might think. Of course, you'll know if you're developing a 'chase the ghosts around the maze as you eat the power pills' program that you are going to need a different sort of program layout to one which places you inside a Haunted Oak, peopled with gremlins and halflings. But you have to go beyond the basic "I'm going to write me an Adventure" stage to work out such things as (a) what the object of the game will be; (b) what the screen display will look like; (c) what variables, and variable names, you'll need;

(d) the nature of the player input; (e) how 'winning' or 'losing' will be determined; and so on.

Let's look at these one by one.

THE OBJECT OF THE GAME

This can usually be stated very succinctly: "To find the lost treasure of the Aztecs"; "To destroy as many asteroids as possible before running out of ships"; or "To play a game of chess". But even though this stage of the game production can be accomplished very quickly, it should not be overlooked. Get this statement — which might be just a sentence, or may run to a paragraph length or more, if there is more than one 'screen' to be worked through, with a different scenario for each screen — down in writing.

You may well discard the original aim as the program develops, and it looks like the direction it is taking is better than the one you first thought of. Despite this, it is important to have something concrete to aim at, to stop you wasting hour after hour doodling aimlessly.

THE SCREEN DISPLAY

I've found that making a sketch, or sketches, of what the display will look like once the program is up and running, is of tremendous benefit. Once you have your drawing, and it doesn't matter how rough it is so long as it shows all the important things you want on the screen, and their relative positions and size, you'll discover the program concept is likely to crystalize.

As well as seeing immediately how you will write parts of the code to achieve the game's aim, you'll get an idea of whether or not the game is even worth writing in the form you had considered. Perhaps the game will be too complex if you leave everything on the screen you were intending to; or maybe most of the screen will be wasted space, so that a more complicated game scenario should be devised.

I've discovered that sketching the proposed screen display before starting to program is particularly useful, especially when creating arcade and simulation games. You get an indication of the variables you'll need, the user-defined graphics, the kind of player inputs which will be most conducive to good player interaction, and so on.

Simulation games, as you probably know, are those in which the computer models an external reality — such as running a cake shop, a war, or an airport — and allows you to experience (after a fashion) what it would be like to take part in such an activity in real life. Simulation games are not particularly difficult to write — in terms of needing clever coding — but instead demand a methodical, painstaking approach to the program.

In my book *The ZX Spectrum Explored* (Sinclair Browne, 1982), there is a program with the unlikely name of 'Workin' for the Man', in which you are running a factory, staffed with a highly-erratic workforce, involved in the manufacture of some mythical product called 'The Zibby'. The player gets a factory report two or three times a week, and from this report has to decide how many staff he or she will hire or (attempt to) fire, how many Zibbies will be the production target for the week, and so on.

This report is the key to the program, and when I wrote the game, I started by making a sketch of how the screen would look. It was a bit like this:

FACTORY REPORT: WEEK 5

Capital in hand is $2,657.92

Your stores hold 12 Zibbies worth $169.68
They sell for $14.14 each and cost $7.41
each to make

Workforce is 7 people
Their wages are $41 each and the wage bill
this week is $287

Each person can make 10 Zibbies a week,
a total output of 70

Once I had this sketch drawn up, I was ready to go. As you can see, it gives a very good indication of the variables which will be needed. For a start, I know I'll have to control the number of the week, the capital, the contents of the stores (and their value) and so on.

I found that once I'd completed the screen display sketch, the rest of the program was relatively easy to write. Doing a sketch in this way gives you an instant guide to the main variables you'll need.

USE HELPFUL VARIABLE NAMES

I also tend to use variable names which relate in some way to that which they are representing, as it saves having to keep a list of the variables which have been assigned, and what they've been assigned to. For example, I could use WK for week, CH for capital in hand, MZ for the cost of making each Zibby and SZ for the selling price. If Z was the number of Zibbies, I would know that the total value of Zibbies I had was Z (the number of them) multiplied by SZ (their selling price) and it cost me Z multiplied by MZ (their price of manufacture) to make them. My profit, if I sold them all, would then be Z*SZ minus Z*MZ.

If you follow a similar idea, you'll find it is much easier to keep track of what is happening in your program than might otherwise be the case.

THE NATURE OF THE PLAYER INPUT

It's important to make games *easy* and fun to play. It's not good having the best Asteroids-derivative program in the world if players have trouble hitting the fire button because you've placed it right next door to the 'rotate' control.

Many programs which provide 'up', 'down', 'right' and

'left' controls, automatically use arrow or cursor keys, even though these might be most inconvenient for the player to use. Have a look at your keyboard, and see if you can find better ones. I often use "Z" and "M" for programs which need just left and right movement, with the space bar for fire. These keys seem logical to me, and no player time is wasted in learning them, or trying to remember them when the game is underway. In a similar way, I tend to use "A" (for up) and "Z" (for down) for the left hand, and the "greater than" and "less than" keys for left and right (pointing out to the player that the < and > symbols point in the relevant directions).

Use INKEY$ or GET$ whenever you can, to prevent the player from having to use the RETURN or ENTER keys to get the program underway.

HOW THE GAME WILL END

The way the game will be won and lost needs to be defined, and clear to the player. Do you need to blast all the aliens to win, and will you lose automatically if one alien lands, and you've still got ships left, or only if you have no ships left. In a two-player game, is the loser the first player to lose three lives, or seven pieces, or does the game only end when the *difference* between the two scores is three or seven or whatever.

Work this out, and make it very clear to the player. Whether the goal of the game is to clear the left-hand side of the screen of the Screaming Widgies, or to clock up a fortune of $7.3 billion, it must be both clear to the player, and *possible to achieve*. A 'win condition' which can never be achieved on the higher levels of play is most unsatisfactory. No matter how difficult it is to do, you are only defrauding players if you set goals whose achievement is not possible within the constrictions you've put into the game.

I hope these five points may give you a few ideas on how you can go ahead and write programs which will be relatively easy to write, and which will be satisfying for you and your friends to play.

GLOSSARY

A

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

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

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

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

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

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

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

B

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

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

BCD — an abbreviation for Binary Coded Decimal.

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

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

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

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

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

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

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

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

C

CAI — Computer Assisted Instruction.

CAL — Computer Assisted Learning. The term is

generally used to describe programs which involve the learner with the learning process.

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

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

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

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

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

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

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

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

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

D

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

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

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

Disk/Disc — this is a magnetically sensitised plastic disk, a little smaller than a single play record. This is used for

storing programs and for obtaining data. Disks are considerably faster to load than a cassette of the same length program. The disk can be searched very quickly while a program is running for additional data.

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

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

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

E

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

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

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

F

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

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

Flow Chart — a diagram drawn up before writing a program, in which the main operations are enclosed within

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

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

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

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

G

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

Graphics — pictorial information as opposed to letters and numbers.

H

Hard Copy — computer output which is in permanent form.

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

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

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

High Level Language — a programming language which allows the user to talk to the computer more or less in English. In general, the higher the level of the language (that is, the

closer it is to English), the longer it takes for the computer to translate it into a language it can use. Lower level languages are far more difficult for human operators but are generally executed far more quickly.

I

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

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

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

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

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

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

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

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

J

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

K

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

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

L

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

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

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

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

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

M

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

Magnetic Disk — see Disk and Floppy Disk.

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

Memory — there are two types of memory within a computer. The first is called ROM (read-only memory); this is the memory that comes already programmed on the

computer, which tells the computer how to make decisions and how to carry out arithmetic operations. This memory is unaffected when you turn the computer off. The second type is RAM (random access memory). This memory holds the program you type in at the keyboard or send in via a cassette or disk. In most computers the computer 'forgets' what is in RAM when you turn the power off.

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

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

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

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

MPU — an abbreviation for Microprocessor Unit.

N

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

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

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

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

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

O

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

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

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

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

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

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

P

Pad — see Keypad.

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

PASCAL — a high level language.

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

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

Prestel — the British telecom name for a system of calling up pages of information from a central computer via the

telephone and displaying them on a television screen. A similar commercial version in the United States is known as The Source.

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

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

R

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

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

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

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

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

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

S

Second Generation — this has two meanings. The first applies to computers using transistors, as opposed to first

generation computers which used valves. Second generation can also mean the second copy of a particular program; subsequent generations are degraded by more and more noise.

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

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

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

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

SNOBOL — a high level language.

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

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

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

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

T

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

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

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

Time Sharing — a process by which a number of users may have access to a large computer which switches rapidly

from one user to another in sequence, so each user is under the impression that he or she is the sole user of the computer at that time.

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

U

UHF — Ultra High Frequency (300-3000 megaHertz).

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

V

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

VDU — an abbreviation for Visual Display Unit.

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

W

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

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

BIBLIOGRAPHY

Compiled by Tim Hartnell

Usborne have released a number of very attractive books in their Usborne Computer Books series. Drawing on their vast experience in the field of producing low-priced, highly-coloured, attractive books for young readers, they've produced some books which will enlighten both young and not-so-young readers.

I'll look at three of their titles, three which cover just about the whole field of computer interests.

Information Revolution
(Lynn Myring and Ian Graham, Rigby).
Presenting an eminently readable introduction to the 'revolution' which covers such fields as computers (of course), text information services via the television screen, word processing, 'future phones' and satellite communications, *Information Revolution* is an ideal guide for the person who wants an easy-to-read introduction to the field.

Computer Jargon
(Corinne Stockley and Lisa Watts).
The tone of this book is set by the frontispiece, which has a number of odd little coloured robots sitting around a table laden with computer junk, pointing at each piece saying "This is a disk drive", "This is a digital tracer" (!) and "This is a printer".

Robotics — What Robots Can Do and How They Work
(Tony Potter and Ivor Guild).
This is definitely a candidate for the award of 'the longest title of the year'. But it is very accurate. Don't be put off by the pretty pictures, as you'll soon discover this book has a lot of solid information. Topics covered include "What robots can and cannot do", "How arm robots work", "How to teach a robot" and "Build your own micro-robot"; this last section actually includes nine pages of circuit diagrams and all to build a little two-motor robot which, following a program typed into your micro, will run about the floor. Robotics is a field of the near future (with personal robots certain to be a bigger craze — when 'real robots' finally arrive — than computers will ever be).

Practise Your BASIC
(Gaby Waters and Nick Cutler).
You'll find this book — which predictably contains a number of exercises, puzzles and problems to solve by writing programs — should be useful in giving you a number of 'core problems' which will run on your computer and which can then be modified to take advantage of your system's special features. Program listings include 'Pattern Puzzles', 'Jumping Man', 'Horse Race', 'Word Editor' and 'Treasure Hunt', a mini-Adventure.

Help With Computer Literacy
(June St Clair Atkinson, Houghton Mifflin).
This is a large format book with an attractive cover, fairly priced for its 122 pages. It appears to be aimed at the early to middle years of secondary education, but contains a lot of material which those teaching younger children could easily adapt. Although it avoids the 'Gee Whiz' approach of the Usborne texts, it uses cartoons and diagrams to get its message across in an inviting manner.

The Interface Computer Encyclopedia
(Ken Ozanne, Interface Publications).
Compiled by a lecturer in mathematics at the NSW Institute of Technology, this work could perhaps be more accurately called 'The Computer Book of Lists', rather than an encyclopedia. It contains annotated references to 'all' microprocessors, 'all' microcomputers, and 'most' microcomputing magazines. The inverted commas are there because — as the author admits candidly in his introduction — any such work is likely to be out of date even before it is published. Fat (445 pages) with minimalist presentation (the whole book is dumped directly from a word processor onto a dot-matrix printer) you'll find this a useful work if you want a ready reference to chips, computers and the ever-growing field of specialist magazines.

Computer Resource Book — Algebra
(Thomas Dwyer and Margot Critchfield, Houghton Mifflin).
Dwyer and Critchfield have clocked up an enviable string of successful computer books, and this one, part of a series, shows why. With simple, but valuable programs, the authors lead the reader (who can be a secondary student, or an instructor) through most of the phrases of the BASIC programming language which are common to all low-priced computers, and most educational time sharing systems.

Apple II BASIC
(David Goodfellow, Tab Books Inc.).
Attractively packaged, this book is clearly laid out, with an abundance of example programs; it takes a commendable approach to the business of teaching programming, with the qualities of 'programming style' introduced without fanfare. In the crowded field of 'how to program your Apple' books, this one stands out. Much of the material presented is applicable to any microcomputer.

Pre-Computer Activities
(Dorothy Diamond, Hulton Educational).
This practical guide for teachers and parents can help make children familiar with essential computer processes and language before they have hands-on experience. The book contains a number of interesting activities, including investigating binary numbers using little lights, and working with cardboard 'calculators' before getting to the real thing. The discussion on computer graphics is enlivened by reference to the solid blocks which make up a 'Pacman' figure.

Word Processing Experience
(Janet Pigott and Roger Atkins-Green, Stanley Thornes Publishers Ltd.).
Designed for schools, but ideal for adapting if you'd like to increase your skill with a word processor (or simply because you'd like to see what word processors can do so you can write one for your own microcomputer), this book looks at the mechanics of word-processing, while passing on a great deal of useful information about word-processing techniques.

An Introduction to Micro-electronics and Microprocessor Systems
(G H Curtis and P G Wilks, Stanley Thornes, Publishers Ltd.).
This work was written for junior college students and older school pupils, as well as for non-specialists who wanted a comprehensive — if dry — technical introduction to the subject. The going is not easy, but it's worth the effort. Topics covered include 'Logic', 'Programming the Microcomputer' and 'Analogue, Binary and Digital Systems'.

Computer Images — State of the Art
(Joseph Deken, Thames and Hudson).
This is a beautiful book, large and glossy, and packed with quality full-colour computer-generated (or, in some cases, computer-modified) images. The whole fascinating field of modern computer graphics is discussed — from television pro-

gramme introductions using photographs which are colour-modified, twisted and tweeked, to the use of incredible high-resolution images in simulators for flight training and tank manoeuvring. You'll read (and see) how computers are used to produce images, how these are used for education and communication, why 'art for art's sake' is a goal worth pursuing, and how computer images can evolve using processes uncannily akin to the processes by which groups of cells multiply and divide. If you want to see what can be done with high resolution graphics and when time, money and skill abound, you should get this book.

Computer Bluff
(Stephen Castell, Quartermaine House Ltd.).
A much more valuable book than its title indicates, it contains a lot of information on the what and how of computers, along with a generous dollop of computer jargon (or 'How to Cheat in Computer-Speak'). The style is gentle and amusing, with no appalling puns or excessive asides (such as 'didja get that joke, buster?'). A pleasant, painless book which you can digest, then give to a parent.

Penguin Books has moved into the computer field with enthusiasm. As well as a 'Getting the Most Out of Your...' series, they have a number of games books. Two which stand out are **The Penguin Book of VIC 20 Games** (Paul Copeland) and **The Penguin Book of Commodore 64 Games** (Robert Young and Paul Copeland). Priced at £4.95 each, these large format books include such programs as 'Space Venture', 'Oil Rig' and 'Red Alert'. Worth buying, even if you do not have a VIC or a Commodore 64, simply as a source of ideas for new programs to create on your own microcomputer.

Arcade Games for Your VIC 20 and **Arcade Games for Your Commodore 64** (Brett Hale, Corgi/Addison-Wesley) by contrast, are definitely only for those who have the machine specified. The programs are locked irrevocably to the computer named. Taking advantage of a number of machine-specific features (such as sprite graphics on the 64), Brett has produced a selection of around 20 programs for each machine. Each one is listed twice, the first time for the joystick and the second time for the keyboard. Titles include 'Galaxy Robbers', 'Bullet Heads' and 'Yackman'.

CREATING ADVENTURE PROGRAMS

There are a number of books, some of which are aimed at com-

puter owners, which will help you if you are one of the many, many computer games players who are interested in developing 'Adventure' and 'Dungeons' type programs. The place to start is with TRS Hobbies' **Dungeons and Dragons** (TM) Basic Set, which comes with the introductory rule book, Dungeon Dice (tm) and an instruction module, along with a sample scenario 'The Keep on the Borderlands'. If you're new to the field, you should start with this set to give you an idea how 'real life' Adventure programs are built up.

Additional information is provided by **Fantasy Role-Playing Games** (J. Eric Holmes, Hippocrene Books Inc.) which looks at the whole field and, despite some disparaging things to say on computer versions of such games, is worth looking for. Another overview of the field — with more sympathetic comments on the use of computers — is provided by **Dicing With Dragons — An Introduction to Role-Playing Games** (Ian Livingstone, Routledge and Kegan Paul), which includes a full 'solo Adventure', a review of the major games on the market, and a fascinating chapter on the pleasures and perils of being Dungeon Master in 'Playing God'.

Fantasy Wargaming (compiled Bruce Galloway, published Patrick Stephens) provides a complete unified system for 'historically accurate' (or at least in tune with the beliefs and circumstances of individuals in the peasant, feudal-economy times in which many Adventures are set) games. The fight, weapon and monster tables alone are worth the book, as many of their ideas can easily be incorporated into your Adventures.

There are two computer Adventure books which you could get to help you in the fascinating area of producing Adventure games on your machine.

Creating Adventure Programs on Your Computer
(Andrew Nelson, Interface Publications).
Written by the author of *More Games for Your VIC 20* and *Games for Your TI 99/4A*, in the Virgin Books games series, this book takes you through the task of developing an Adventure program of your own, concentrating more on the 'Loot and Pillage' school of gaming than the Scott Adams' 'solve this puzzle to advance' field. Three complete Adventure programs are included.

Write Your Own Adventure Programs for Your Microcomputer (Jenny Tyler and Les Howarth, Usborne) is a much quicker introduction to the field than Nelson's, but nevertheless packs a lot of valuable information into its 48 pages. Step-by-step instructions are provided for creating an Adventure from

scratch. A complete program — 'Haunted House' — is included in the book.

The Age of Computers is the general title of four fine books produced by Wayland Publisher Limited. Each priced at £4.95, the books present a careful, but inviting, view of four aspects of the computer field, one on the history of computers and the others looking at specific areas of modern computer application. Each book is by Ian Litterick and Chris Smithers. The four titles are **The Story of Computers,** with Charles Babbage and Uncle Sir Clive Sinclair just inside the cover (and these two pictures accurately sum up the historical period covered by the book); **How Computers Work** (with chapter headings including 'Bits, Bytes and Binary', 'Decision-making by Transistor', and 'Talking With Computers'); **Computers in Everyday Life** (such things as 'Robots in the Home', 'Magnetic Money' and 'Medicine and the Disabled'); and **Computers and You** ('Computopia', 'Big Brother', 'War and Peace' and — a fascinating final chapter — 'Will Computers Need Us?').

Inside BASIC Games
(Richard Mateosian, Sybex).
This book is a slightly overwritten guide to understanding computer games. You'll learn how to write interactive programs in BASIC and how the principles of system development are applied to small computers. The book also looks at how the features of specific small computer systems have been supported in BASIC. If you can contend with the verbiage, you'll find this book well worthwhile.

1001 Things to Do With Your Personal Computer
(Mark Sawush, Tab Books).
Big and fat, and full of ideas, you'll find much here of interest to enlarge your computer horizons. The book tells you about writing music and stories with your computer, aiding a mechanic or a carpenter, solving simultaneous equations, astrology and much, much more.

Stimulating Simulations
(C. W. Engel, Hayden Book Company).
Here are 12 unique programs written in a good, general version of BASIC. The fascinating programs include 'Forest Fire', 'Rare Birds' and 'The Devil's Dungeon'. You're sure to enjoy playing those three, along with 'Diamond Thief', in which the computer decides who has committed the crime, then challenges you to discover which of the suspects is guilty. The material in this book is generally tightly programmed, and can be a helpful source of ideas to improve your own computer work.

The BASIC Handbook
(David A. Lien, Compusoft Publishing).
This is an encyclopedia of the BASIC language. It comes into its own when you find a program in a magazine or book which you'd love to try, but are frustrated because it is written for another version of BASIC. Every BASIC word you've ever heard of (and many you may not have, such as NE, GOTO-OF and LE) is in here, along with a number of variations, one of which will almost certainly be on your machine.

BASIC Computer Games
(David Ahl, Creative Computing Press).
This is a classic work, still selling well despite the fact it was one of the first such books — if not *the* first — on the market. David Ahl has been in personal computers even before there were such things. Although several of the games are overly-dependent on the random number generator, you'll find there are many, many games you'll want to adapt and improve for your own computer.

How to Buy (and Survive) Your First Computer
(Carolee Nance Kolve, McGraw-Hill Book Company).
When is a business ready for a computer? How do you make an intelligent, informed choice among the hundreds of computers available? Will a computer improve a company's operations? Answers to these and a score of similar questions are in this book, which explains in detail what to consider before buying, how to select the right computer, and what to do after ordering the computer to ensure a successful installation. Ms Kolve has over 15 years computer experience (including a stint with IBM) and brings her experience to bear in a relatively easily-digestible guide.

Your First BASIC Program
(Rodnay Zaks, Sybex).
This book, liberally illustrated with large red dinosaurs in a variety of situations vaguely related to the text (one, for instance, as a cowboy getting tangled up in his ropes with the caption 'Be careful when looping'), is a gentle and worthwhile introduction to the not-so-secret secrets of programming in BASIC. When you want to move beyond just typing in other people's programs from books and magazines, this may be a good place to start.

This bibliography was compiled by the series editor, Tim Hartnell, who has felt constrained not to recommend any of his own books. However, he asked us to mention two which could be of use and interest to you.

The first is **The Personal Computer Guide** (Virgin Books) which explains what a personal computer is, and answers questions like "Will it help my kids?", "What sort of games can we play on it?" and "What can I use it for in the home?". The book describes many of the most popular computers available today, with illustrations, technical specifications and other information to help you to choose the equipment best suited to your requirements. Also included is an introduction to BASIC programming, with details of programs suitable for use in the home, a list of suppliers and user clubs, and a guide to further reading. There are also chapters covering the personal computer's history and its future. When you're ready to upgrade, you'll find this book a good, unbiased, reference work which looks at the choices facing you.

Tim Hartnell's Giant Book of Computer Games.
Described by *Personal Computer News* as 'a good source of ideas', this 386-page book, published by Fontana, for £3.95, contains over 40 programs which will run with minimum modifications on most popular microcomputers. The games include chess (of a sort!), a 17K Adventure and 'Hyperwar'.

MORE GAMES FOR YOUR DRAGON 32

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

THE BIANCO MANSIONS (solve the riddle of the Bianco family in this complex and dangerous adventure); JOUST (can your Knight beat the Dragon?); ROCK, SCISSORS, PAPER (a computerised, graphic version of an old favourite); METEOR (brave the storm ahead in this high resolution, action-packed game!); TREASURE TRAIL (can you find the hidden treasure in the time you have left?); BAT 'N' BALL (keep a fast-moving ball in play); and REACTION (this vivid program tests your split-second reactions).

MORE GAMES FOR YOUR DRAGON 32 will improve your programming skills as you follow the instructions to put each of the programs into your machine, and comes complete with a brief dictionary of computer terms, a selective bibliography and some hints on how to extend the programs in the book.

Programs of originality and quality for all the family.

Virgin

ISBN 0 86369 033 5

United Kingdom £2.95
Australia $9.95 (recommended)