# PROGRAMMING
# IS FUN
# FOR THE APPLE

# BY
# MARK STEWART

THIS BOOK IS
DEDICATED TO
MY PARENTS
WHO BOUGHT
OUR COMPUTER
AND LET ME GO
TO COMPUTER
CAMP

## TABLE OF CONTENTS

| TOPIC | PAGE |
|---|---|

# Disk Commands

LOAD name    - loads a program into computer memory from the disk

SAVE name    - saves a program on the disk under a specified name

RUN name     - loads and runs basic program from disk

DELETE name - erases program from disk

CATALOG      - displays a list of programs on disk

LOCK name    - locks a program so that it can't be erased (in the
               catalog a locked file is marked with * )

UNLOCK name - unlocks locked file


Before you begin to write programs you are going to need an
formated disk to put your programs on. Here is how you format a
disk:

Put DOS 3.3 SYSTEM MASTER (comes with disk drive) into drive 1
and turn on the computer. After it has loaded type NEW then hit
the RETURN key. Put a blank disk into the disk drive, then
type this:

10 PRINT "HELLO" !RETURN!

20 PRINT CHR$(4); "CATALOG" !RETURN!

INIT HELLO !RETURN!


The disk drive should whirrr for a little while and then it will
stop. The disk has been formated. Now turn off the computer and
turn it back on again. It should show the catalog and stop. You
are ready to start typing your programs into the computer.
Remember that after you make a program you should save it.

Lesson 1

In this lesson, you will learn how to write a program. To do
this, you will have to know three commands REM, PRINT and END.

REM  -A remark that you put in a program to show the date or what
        the program does. When the computer sees a rem it will just
        skip over it and go to the next command.

PRINT-This command does just what it says, it prints. In order to
        "PRINT" you have to put whatever you want to print in
        quotes.

END  -Tell the computer to end a program.

Type NEW then press return

            Type this:

You must     10 REM  EXAMPLE PROGRAM        You must press the
put a line   20 REM  1985 MARK STEWART      "RETURN" key after
number       30 PRINT "HI THERE"            every line.
before       40 PRINT "YOU LOOK FUNNY"
every line   50 END

Now type: RUN

The Program should
do this when it is
run.

HI THERE
YOU LOOK FUNNY

Type: LIST

The computer will put the LIST of the program on the screen.

Type:

35 PRINT "JOHNY RAY"

Type LIST again

10 REM EXAMPLE PROGRAM
20 REM 1985 MARK STEWART
30 PRINT "HI THERE"
35 PRINT "JOHNY RAY"
40 PRINT "YOU LOOK FUNNY"
50 END
Note that line 35 is now in between 30 and 40

When program is RUN:

5

HI THERE
JOHNY RAY
YOU LOOK FUNNY

Keep inserting more lines and changing the ones that are there or
just start from scratch. When you are satisfied with your program
invite your friends over to see it or just save it (see disk
commands) and then go on to lesson 2.

Lesson 2

In this lesson we will be working on the GOTO statement and also
a bit of using a semi-colon in your PRINT statements.

GOTO - When the computer runs into this command it will jump to
       the line that you tell it to jump to.

Type NEW

Then type this program:

```
10 REM MARK STEWART
20 REM MARCH 21
30 PRINT "HELLO"
40 GOTO 30
```

The computer will skip over 10 & 20 because they are REM
statements and will go to 30. It will print "MARK" and will go to
40, it will see GOTO 30 and it will go back to 30 and will do it
again.

When program is run

```
MARK
MARK
MARK
MARK
MARK
MARK
MARK
etc.
```

This is called an endless loop because it will keep printing MARK
forever.

To stop the program press the CONTROL key, while holding it press
the letter C.

Type NEW

Now try this program.

```
10 REM 1985 MARCH 21
20 PRINT "F"
30 GOTO 80
40 PRINT "O"
50 GOTO 100
60 PRINT "S"
70 END
80 PRINT "R"
90 GOTO 40
100 PRINT "G"
110 GOTO 60
```

4

Type RUN

What does it say?

Now change these lines

```
20 PRINT "F";
40 PRINT "O";
60 PRINT "S";
80 PRINT "R";
100 PRINT "G";
```

Type RUN

Note that the word is running horizontally. This is because
Semi-colon (;) means that the next thing the computer prints out
will be right after the word or letter that the semi-colon is
after.

Try this program:

```
10 REM MARCH 22
20 PRINT "HELLO";
30 PRINT "THERE";
40 PRINT "GUY"
50 END
```

Now RUN it

It should look like this:

HELLOTHEREGUY

The words are all bunched together. To fix this make these
changes:

```
20 PRINT "HELLO ";
30 PRINT "THERE ";
```

Now it should work

Note that now there are spaces after the the "O" and the "E".
Experiment with the GOTO and use the semi-colon in your programs
until you get used to using them. When you feel you are ready
skip over to lesson 3

Lesson 3

In this lesson we will be doing Math on the computer and we will
also be using variables in the program.

Variables - A variable is a letter or word that takes the place
              of a number or a string of letters.

Math is very easy to do on the computer. All you have to do if
you want to know what 5 - 3 is you just type:

10 PRINT 5 - 3

When it is run:

2

It is very simple as you can see. There is only one thing that is
different than you are usually used to and that is the signs.
Here is a little chart that you can follow to remember your
signs.

| MATH | COMPUTER | |
|------|----------|-|
| ÷ | / | Division |
| X | * | Multiplication |
| + | + | Addition |
| - | - | Subtraction |

Try a program like this:

```
10 REM MARCH 26
20 PRINT "5 * 6 = ";
30 PRINT 5 * 6
40 PRINT "12 - 5 = ";
50 PRINT 12 - 5
60 PRINT "8 - 6 * 5 =";
70 PRINT 8 - 6 * 5
80 END
```

When it is RUN

```
5 * 6 = 30
12 - 5 = 7
8 - 6 * 5 = 10
```

Try this program:

```
10 REM VARIABLES
20 A=12
30 B=5
```

```
40 PRINT A+B
50 PRINT A*B
60 PRINT A/2
70 PRINT A+B-7
80 END
```

When program is run:

```
17
60
6
10
```

Variables are very helpful if you are making a program that uses
the same number a lot or if you have one that finds the perimeter
of a circle that would look something like this:

```
10 REM CIRCLE
20 D=6
30 PI=3.14
40 P=D*PI
50 PRINT "THE PERIMETER IS ";P
```

In this program D stands for diameter P stands for perimeter and
PI stands for PI.

Correct the errors in this program:

```
10 REM FIX VARIABLES
20 10 = A
30 B = 5
40 PRINT "B * A"
50 END
```

When the program is corrected it should look like this when RUN:

```
50
```

Lesson 4

In this lesson we will be using the INPUT statement. The input
statement allows you to enter numbers or letters when the program
is running. This will make it a lot easier to do math on the
computer or you can ask someone their name and then say "HI THERE
MARK, YOU LOOK FUNNY".


INPUT - Is a command than is used to get information from the
        user. The information that the user put in will then be
        inserted into the variable that the programmer specifies.

Try this program:

```
5  REM MULTIPLY
10 INPUT "A NUMBER";N1
20 PRINT "ANOTHER"
30 INPUT N2
40 PRINT N1;" * ";N2;" = ";N1*N2
50 END
```

This program asks you for a number then asks you for another one
and it multiplies the two numbers. The first time it asks you for
a number it asks "a number" in quotes. With any message that you
put in INPUT statement you have to use quotes or else the
computer will think that your message is a variable and it won't
work. It is just like the PRINT statement when you put a message
in. When you put a message in an INPUT statement you have to put
a semi-colon after the second quote or the blinking cursor will
be on the next line.

Make a calculator program that asks you if you want to add,
subtract, multiply or divide. Then the user gives two numbers.
The program will give him or her the answer. It should go back to
the beginning and start all over again.

When it is RUN it should look something like this:

1. ADD
2. SUBTRACT
3. MULTIPLY
4. DIVIDE

WHICH NUMBER? 1

NUMBER? 3
NUMBER? 5

3 + 5 = 8

etc.

You are probably wondering how you INPUT word. Well that is very simple. It is the same as numbers except instead of using variables called A,B,C,D,E. etc. you would use A$,B$,C$,D$,E$ etc. The "$" at the end of the variable is called a string and that means that the variable has letters in it or a "string" of letters.

Try this program:

```
10 REM STRINGS
20 A$="MARK STEWART"
30 PRINT "HELLO, ";A$
40 PRINT "YOU LOOK FUNNY"
50 END
```

When this program is RUN it should say "HELLO, MARK STEWART YOU LOOK FUNNY". When you are putting a word into a variable you have to put quotes around the word. It is like the PRINT statement.

Change line 20 to:

```
20 INPUT "YOUR NAME";A$
```

Now when you run the program it will ask you your name. Then it will say that you look funny. This is done by using the INPUT statement with a string variable. It is just like numbers except there is a "$" at the end of the variable.

Correct the errors in this program:

```
10 REM STRING ERRORS
20 INPUT "A WORD - ";A
30 INVERSE
40 PRINT A$
50 NORMAL
60 PRINT A
70 END
```

Now we are going to learn how to compare words or numbers. You will have to learn the IF/THEN command. It is fairly easy it just works like this:

```
20 IF A=0 THEN GOTO 50
30 END
```

When the computer sees this comand it will check to see if A=0. if A=0 it will jump to 50 and do whatever is on line 50. If A does not equal 0 then if will go to the next line which is 30 and do whatever it says.

Try this program:

```
10 REM IF/THEN
20 PRINT "WHO WAS THE FIRST"
30 PRINT "VOICE OF MICKEY MOUSE"
40 INPUT A$
50 IF A$="WALT DISNEY" THEN GOTO 70
60 PRINT "--WRONG--"
65 END
70 PRINT "RIGHT, YOU ARE A BRAIN!!"
80 END
```

In this program it asks you a question. Then it checks to see if the answer right. If not it says "--WRONG--". If it is it says "RIGHT, YOU ARE A BRAIN!!".

Make a quiz program that asks you if you want History, Geography, Math, or Science. Then it asks you questions in each. Make it keep score to see how you are doing.

What if you are doing a program that could have two answers. You could put an IF/THEN satement on two different lines or you could put IF/OR/THEN in a program like this.

```
10 INPUT "NAME ONE OF THE WRITE BROTHERS";A$
20 IF A$="ORVILLE" OR A$="WILBUR" THEN GOTO 50
30 PRINT "WRONG"
40 END
50 PRINT "RIGHT"
60 END
```

You can also use AND instead of OR. It works the same way. Here is a program with it.

```
10 INPUT "NAME";A$
20 INPUT "PASSWORD";B$
30 IF A$="MARK STEWART" AND B$="TEN" THEN GOTO 60
40 PRINT "NOT RIGHT"
50 GOTO 10
60 PRINT "YOU ARE ALLOWED IN"
70 END
```

In this program the user must enter the name correctly and the password correctly or they will not be allowed in.

Here is then general form of OR and AND within an IF/THEN statement.

IF condition OR condition THEN action

IF condition AND condition THEN action

Correct this program:

```
10 REM CORRECT AND/OR
20 INPUT "FIRST NAME";F$
30 INPUT "SECOND NAME";S$
40 IF F$="MARK" AND "STEWART" THEN GOTO 70
50 PRINT "YOU ARE NOT SMART"
60 END
70 PRINT "YOU ARE SMART"
80 END
```

In This lesson you will be learning the FOR/NEXT/STEP commands.
We will also be doing a bit of animation with them.

FOR/NEXT - A Command used to count how many times it does
something. It counts how many times it will do
something between FOR and next. Here is an example:

```
10 FOR L = 1 TO 10
20 PRINT L
30 NEXT L
40 END
```

When RUN:

```
1
2
3
4
5
6
7
8
9
10
```

STEP - Is what number the computer is going to count by when
using the FOR/NEXT commands. Here is an example program.

```
10 FOR D=9 TO 30 STEP 3
20 PRINT D
30 NEXT D
40 END
```

When RUN:

```
9
12
15
18
21
24
27
30
```

You can also step backwards like this:

```
10 REM STEP BACKWARDS
20 FOR C=10 TO 1 STEP -1
30 PRINT C
40 NEXT C
```

50 END

WHEN RUN:

```
10
9
8
7
6
5
4
3
2
1
```

Try This program:

```
10 REM ROCKET
20 SPEED = 100
30 FOR C=10 TO 1 STEP -1
40 PRINT C
50 NEXT C
60 PRINT
70 SPEED=200
80 PRINT "BLAST OFF!!!!"
90 PRINT
100 PRINT
110 PRINT "   *"
120 PRINT " *C*"
130 PRINT " *A*"
140 PRINT "* N *"
150 PRINT "*****"
160 PRINT "  +"
170 FOR P=1 TO 25
180 PRINT
190 NEXT P
200 SPEED=255
210 END
```

In this program it uses FOR/NEXT/STEP -1 to draw the countdown.
Then it draws a rocket. It then uses FOR/NEXT to print 25 blank
lines under the rocket which forces it to move up and makes it
blast off.

In this lesson we will be learning the HTAB and VTAB commands.
These commands are used for placing the cursor at a particular
spot on the screen VTAB is for vertically and HTAB horizontally.

Try this program:

```
10 REM HTAB/VTAB
20 HOME
30 VTAB 5
40 PRINT "THERE"
50 VTAB 3
60 PRINT "HELLO"
70 VTAB 7
80 HTAB 5
90 PRINT "GOODBYE"
100 END
```

When run:

HELLO

THERE

    GOODBYE

VTAB and HTAB can also be used in animation. Try this program:

```
10 REM ANIMATION WITH VTAB/HTAB
20 FOR H=1 TO 40
30 VTAB 12
40 HTAB H
50 HOME
60 PRINT "HI"
70 NEXT H
80 END
```

This program should make the word "hi" move across the screen. it
it does this by drawing the word in one spot, erasing it and the
drawing it in another spot.

In this lesson we will be working on the LEFT$, RIGHT$, MID$ and LEN commands.

RIGHT$ - This command will take a chunk of letters from the right side of a word. To use it you have to do something like this: A$ = RIGHT$ (B$,3) This command will make A$ the three letters from the right of B$ so if B$="PEOPLE" then A$ would ="PLE"

LEFT$  - Is the same as RIGHT$ except the letters are taken from the letters are taken from the left.

MID$   - This command will take a chunk of letters from the center of a word. It works like this: A$=MID$ (B$,3,5). In this command if B$ = "MARK STEWART" THEN A$ = "RK ST".

LEN    - Then command finds the length of a word. It works like this: L=LEN (A$). In this command if A$ = "HELLO" then L=5.

Try this program:

```
10 INPUT "ENTER A WORD - ";A$
20 A =LEN (A$)
30 FOR L=1 TO A
40 PRINT MID$ (A$,L,1);" ";
50 NEXT L
60 END
```

This program will ask you a word and then it will PRINT out the word spaced out.

Try this one:

```
10 INPUT A$
20 FOR L=1 TO LEN (A$)
30 PRINT LEFT$(A$,L)
40 NEXT L
50 END
```

When program is run,

GREEN BEAN

```
G
GR
GRE
GREE
GREEN
GREEN
```

```
GREEN B
GREEN BE
GREEN BEA
GREEN BEAN
```

Correct this program:

```
10 REM CORRECT LEFT/RIGHT$
20 A="HELLO THERE"
30 PRINT LEFT$ (A$,5)
40 PRINT RIGHT (A$,5)
50 END
```

Lesson 8

In this lesson we will be working on the ON/GOTO statements.

ON/GOTO - Instead of putting IF A = 1 THEN GOTO 40, IF A -2 THEN
          GOTO 50, IF A=3 THEN GOTO 60 etc. In a program ON/GOTO
          makes it a lot easier by putting ON A GOTO 40,50,60
          etc.

Example program:

```
10 PRINT "1. HISTORY"
20 PRINT "2. GEOGRAPHY"
30 PRINT "3. SCIENCE"
40 PRINT "4. MATH"
50 INPUT "WHICH ONE - ";W
60 ON W GOTO 100,150,200,250
```

This is just one part of a quiz program that uses the ON/GOTO
command. You can add it to your other quizz program if you want.
This program asks which category you want and then it goes to the
right spot. If W=1 it will go to 100, if W=2 it will go to 150
etc.

Correct this:

```
10 REM FIX ON/GOTO
20 INPUT"ENTER MONTH# YOU WERE BORN";M
30 ON M$ GOTO 60,70,80,90,100,110,120,130,140,150,160,170
```

FIX VARIABLES

CHANGES:

```
20 A=10
40 PRINT B*A
```


STRING ERRORS

CHANGES:

```
20 INPUT "A WORD - ";A$
60 PRINT A$
```


CORRECT AND/OR

CHANGES:

```
40 IF F$ = "MARK" AND S$ = "STEWART" THEN GOTO 70
```


CORRECT LEFT$/RIGHT$

CHANGES:

```
20 A$ = "HELLO THERE"
40 PRINT RIGHT$ (A$,5)
```


FIX ON/GOTO

CHANGES:

```
30 ON M GOTO 60, 70, 80, 90, 100, 110, 120, 130, 140, 150,
   160,170
```

```
JLIST

1   HOME
2   REM   WORM ATTACK
3   REM   BY MARK STEWART
100  REM   MAIN LOOP
101  :
102  COLOR= 8
105 DR =  PEEK (AR)
107 S =  PEEK (ST)
110  IF DR = LL THEN D = D - 1: IF
     D = 0 THEN D = 4
111  IF DR = R THEN D = D + 1: IF
     D = 5 THEN D = 1
112  VTAB 22: PRINT "SCORE :";SC
115  ON D GOTO 120,122,124,126
120 Y = Y - 1: GOTO 130
122 X = X - 1: GOTO 130
124 Y = Y + 1: GOTO 130
126 X = X + 1
130  IF  SCRN( X,Y) = 7 THEN  GOTO
     3001
131  IF  SCRN( X,Y) = 1 THEN  GOSUB
     3010
132  IF  SCRN( X,Y) = 8 THEN  GOTO
     3001
133  IF  SCRN( X,Y) = 10 THEN  GOTO
     3001
134  IF  SCRN( X,Y) = 13 THEN  GOTO
     3001
135  IF  SCRN( X,Y) = 15 THEN  GOSUB
     3010
136  IF  SCRN( X,Y) = 12 THEN  GOSUB
     3010
139  PLOT X,Y
140 A = B:B = C:C = E:E = F:F = G
     :G = A1:A1 = C1:C1 = D1:D1 =
     E1:E1 = F1:F1 = H1:H1 = B1
141 L = M:M = N:N = O:O = P:P = Q
     :Q = A2:A2 = C2:C2 = D2:D2 =
     E2:E2 = F2:F2 = H2:H2 = B2
145  COLOR= 0: PLOT A,L
148 B2 = Y:B1 = X
150  COLOR= 0: PLOT A,L
199  GOTO 100
999  END
1000 :
1001  REM    ** WORM ATTACK **
1002 :
1010  REM    BY:  MARK STEWART
1011 :
2000  REM  BORDER
2010  GR : COLOR= 7
```

1a

```
2020    HLIN 1,38 AT 0
2022    VLIN 1,38 AT 39
2024    HLIN 1,38 AT 39
2046    VLIN 1,38 AT 0
2050    GOSUB 3100
2055    COLOR= 7
2060    GOSUB 3120
2070    GOSUB 3156
2100 LL = 149
2101 WO = 3
2102 R = 136: VTAB 22: HTAB 20: PRINT
     "WORMS- ";WO
2105 AR = 49152
2107 STROBE = 49168
2110 X = 38:Y = 20
2115 A = X:B = X:C = X:E = X:F =
     X:G = X:L = Y:M = Y:N = Y:O =
     Y:P = Y:Q = Y
2116 D = 2
3000    COLOR= 1: GOTO 100
3001    FOR LOOP = 1 TO 1000: NEXT
     LOOP:WO = WO - 1: IF WO = 0 THEN
     GOTO 10000
3002    VTAB 22: HTAB 20: PRINT "WO
     RMS- ";WO:X = 38:Y = 20
3003    COLOR= 0: PLOT A,L: PLOT B,
     M: PLOT C,N: PLOT E,O: PLOT
     F,P: PLOT G,Q: PLOT A1,A2: PLOT
     B1,B2: PLOT C1,C2: PLOT D1,D
     2: PLOT E1,E2: PLOT F1,F2: PLOT
     H1,H2
3006 A = X:B = X:C = X:E = X:F =
     X:G = X:A1 = X:C1 = X:D1 = X
     :E1 = X:F1 = X:H1 = X
3007 L = Y:M = Y:N = Y:O = Y:P =
     Y:Q = Y:A2 = Y:C2 = YD2 = Y:
     E2 = Y:F2 = Y:H2 = Y
3008 D = 2
3009    GOTO 100
3010 SC = SC + 10:SOUND = PEEK (
     - 16336): IF SC = 960 THEN
     GOTO 11000
3011    RETURN
3020    FOR S9 = 1 TO 10
3021 H9 =  INT ( RND (1) * 37) +
     1
3022 V9 =  INT ( RND (1) * 37) +
     1
3023    PLOT H9,V9: NEXT S9: RETURN

3050    COLOR= 13: PLOT X + 1,Y + 1
     : PLOT X - 1,Y + 1: PLOT X +
     1,Y - 1: PLOT X - 1,Y - 1: FOR
     LOOP = 1 TO 1000: NEXT LOOP:
     GOTO 3001
3100    COLOR= 1: VLIN 19,23 AT 17:
     VLIN 18,24 AT 18: VLIN 18,2
     4 AT 19: VLIN 18,24 AT 21: VLIN
     18,24 AT 22: VLIN 19,23 AT 2
     3
3105    COLOR= 7
3106    VLIN 1,19 AT 13: VLIN 21,38
     AT 13: VLIN 1,19 AT 27: VLIN
     21,38 AT 27
3110    COLOR= 10: VLIN 17,23 AT 20
     : HLIN 19,21 AT 19: HLIN 19
```

20

```
         21 AT 23: RETURN
3120   COLOR= 1
3140   COLOR= 1
3150   COLOR= 1: PLOT 30,6: PLOT 3
       0,8: PLOT 29,7: PLOT 31,7: PLOT
       34,6: PLOT 35,5: PLOT 35,7: PLOT
       36,6
3152   COLOR= 10: PLOT 30,7: PLOT
       35,6: PLOT 31,5: PLOT 32,4: PLOT
       33,3: PLOT 34,2: PLOT 33,5
3155   RETURN
3156   COLOR= 15: VLIN 8,9 AT 10: VLIN
       8,11 AT 9: VLIN 9,13 AT 8: VLIN
       10,16 AT 7: VLIN 12,16 AT 6:
       VLIN 14,16 AT 5
3157   COLOR= 13: PLOT 1,16: PLOT
       2,15: PLOT 3,15: PLOT 4,16: PLOT
       8,16: PLOT 9,15: PLOT 10,15:
       PLOT 11,16: VLIN 17,20 AT 5
       : VLIN 17,22 AT 6: VLIN 17,2
       4 AT 7: VLIN 21,25 AT 8: VLIN
       23,24 AT 9
3158   COLOR= 12: VLIN 36,37 AT 30
       : VLIN 33,38 AT 31: VLIN 32,
       38 AT 32: VLIN 33,37 AT 33: VLIN
       33,38 AT 34: VLIN 36,37 AT 3
       5
3159   COLOR= 0: PLOT 31,31: PLOT
       35,31
3200   RETURN
10000  VTAB 23: INPUT "WOULD YOU
       LIKE TO PLAY AGAIN";A$: IF  LEFT$
       (A$,1) = "Y" THEN  RUN
11000  VTAB 23: PRINT "      YOU
       WIN ! ! !":WO = 0: GOTO 300
       1

]

]

]PR#0
]LIST

1   REM  ANIMATION
2   REM  BY MARK STEWART
5   HOME
6   INPUT "SPEED (1-255)";S
7   INPUT "TIMES";T
10  INPUT "STRING ";W$
11  SPEED= S
12  HOME
13  FOR TI = 1 TO T
20  FOR I = 1 TO  LEN (W$)
30  HTAB 20: VTAB 12: PRINT  MID$
    (W$,I,1)
40  NEXT I
45  NEXT TI
50  SPEED= 255

]PR#0
]

]LIST

1  REM  TURN BY MARK STEWART
```

21

```
5    HOME
6    INPUT "SPEED";S
10   W$ = "/-\!"
11   SPEED= S
12   HOME
15   FOR L = 1 TO 10
20   FOR I = 1 TO  LEN (W$)
30   HTAB 20: VTAB 12: PRINT  MID$
     (W$,I,1)
40   NEXT I
45   NEXT L
50   SPEED= 255

]PR#0
]BOMBS AWAY

?SYNTAX ERROR
]LIST

0    HOME : HTAB (15): VTAB (24): PRINT
     "BOMBS AWAY!!": PRINT : PRINT
     : PRINT : PRINT : PRINT : PRINT
     : PRINT : PRINT : PRINT : PRINT
     : PRINT : PRINT
10   FOR I = 1 TO 200 STEP 2
20   POKE 768,1: POKE 769,25: CALL
     770
30   NEXT I
40   FOR M = 1 TO 6
50   FOR C = 240 TO 250
60   POKE 768,C: POKE 769,05: CALL
     770
70   NEXT C
80   FOR B = 250 TO 240 STEP  - 1
90   POKE 768,B: POKE 769,05: CALL
     770
100  NEXT B
110  NEXT M
120  PRINT : PRINT : PRINT : PRINT
     : PRINT : PRINT : PRINT : PRINT
     : PRINT : PRINT : PRINT : PRINT
     : PRINT
```