Vantage Press, Inc.

BOOK PUBLISHERS • 516 WEST 34TH STREET • NEW YORK, N.Y. 10001 • TELEPHONE (212) 736-1767

Dear Author:

I am very sorry that you are not able to accept our proposal for the publication of your book.

As you requested, I am returning your manuscript to you.

Although our experience has shown that it is very difficult for a new writer to have a manuscript accepted by a publisher who does not require a fee, we do believe that it would be worthwhile for you to make the effort. I am therefore listing below ten of the leading publishers in the New York area. We do not recommend one over the other, nor do we know if they are all in the market for new material at the present time.

You might be one of the fortunate few new writers whose manuscripts are accepted each year by one of these publishing companies, and I do want to wish you success if you decide to make this effort.

Of course, if your circumstances should change in the future, and you find yourself in a position to authorize the publication of your book with Vantage Press, I sincerely hope that you will decide to reopen negotiations.

Cordially,

MARTIN LITTLEFIELD
Vice President

ML:vw

Doubleday & Co.
245 Park Avenue
New York, NY 10017
Tel: 953-4561

Harper & Row
10 East 53rd Street
New York, NY 10022
Tel: 593-7000

Prentice-Hall, Inc.
Englewood Cliffs,
New Jersey 07632
Tel: (201) 592-2000

E. P. Dutton & Co.
201 Park Ave. South
New York, NY 10003
Tel: 674-5900

Charles Scribner's Sons
597 Fifth Avenue
New York, NY 10017
Tel: 486-2700

Random House, Inc.
201 East 50th Street
New York, NY 10022
Tel: 751-2600

Vanguard Press
424 Madison Ave.
New York, NY 10017
Tel: PL5-4330

William Morrow & Co.
105 Madison Avenue
New York, NY 10016
Tel: 889-3050

Simon & Schuster, Inc
1230 Sixth Avenue
New York, NY 10020
Tel: 246-2121

Harcourt, Brace & Jovanovich, Inc.
757 Third Avenue
New York, NY 10017
Tel: 888-4444

BASIC POWER

BY MICHAEL BRADSHAW

DEDICATED TO:

MARY TOPP, MY VERY BEST FRIEND

SPECIAL THANKS TO:

MARILEE DEGNER, FOR HER SUPPORT

ELSIE KEPHART, FOR GETTING ME STARTED

BILLY PEARCE, FOR WHOM THIS BOOK IS WRITTEN

SANDY LOUDON, FOR HIS SUPPORT

Introduction

Now, you and I both know that there are hundreds of thousands of BASIC books floating around in bookstores today.

So many, that everytime you walk into a bookstore, you see a different one.

But this book IS different.

This book will teach you how to write BASIC programs, how to use BASIC commands, and teach you all you'll need to know to run a computer in no time.

Together with my commrade, Mr. Ray Heitkemper, I have written some BASIC programs.

This book is actually a workbook, because you do most of the work yourself. I give you the help and you write the programs.

Therefore, I hope you'll indulge yourself in.....BASIC.

Michael Bradshaw

Chapter One

First of all, I think you should understand what BASIC stands for. It's the acronym for: Beginner's All-purpose Symbolic Instructional Code.

In this chapter you'll learn the BASIC commands, and later on, I'll teach you how to use the commands to write BASIC programs.

So it is VERY important that you read this section before you read anything else.

The first BASIC command I'd like to introduce to you, is the command: PRINT.

PRINT:

The PRINT statement allows you to print anything on the screen you want.

This, very simply, is done like this:

Print "Hello"

Everything enclosed in the quotations will be printed on the screen. Therefore, "Hello" will be printed once.

When you use the PRINT statement in a program, the format is:

LINE NUMBER -QUOTATION MARKS (") -(WHATEVER YOU WANT PRINTED) -QUOTATION MARKS (")

The next BASIC command we will explore is: GOTO.

GOTO:

The GOTO statement might be easier to explain if we break it down into simple words.

The GOTO statement can be broken down into the two simple words: Go To. Therefore, the GOTO statement tells the computer to go somewhere in the program.

As in the program:

```
10 PRINT "HELLO"
20 GOTO 10
```

Line 20 tells the computer to go to line 10.

So this program is running in circles.

INPUT:

The INPUT command allows you as the operator, to feed information into the computer.

For example, let's say you've just written a quiz called, "20 QUESTIONS." As the computer program feeds question after question to the operator, the operator responds by answering the questions. The answers are the INPUT.

In Chapter Two, INPUT is used quite often in the BASIC programs.

For now, try this BASIC program:

```
10 PRINT "WHAT IS YOUR NAME?"
20 INPUT N$
30 PRINT N$
40 GOTO 10
```

Notice in both lines, 20 and 30, there is a N with a dollar sign after it. This is called a STRING VARIABLE.

A STRING VARIABLE is a symbol that stores a set of characters, such as a word, in the program.

So, after the operator has fed in the information, it is stored in the STRING VARIABLE.

Here are some STRING VARIABLES I use in writing my BASIC programs:

N$- a number

W$- a word

A$- an answer

L$- a letter

P$- a password

A STRING VARIABLE doesn't really have to stand for anything. It can be any letter with a dollar sign.

IF.....THEN

The IF.....THEN statement is a conditional statement.

It excludes a STRING VARIABLE from the generalization of the rest of the program.

For example, let's add an IF.....THEN statement to our last program:

```
10 PRINT "WHAT IS YOUR NAME?"
20 INPUT N$
30 IF N$="JOE" THEN 60
40 IF N$="SALLY" THEN 80
50 GOTO 100
60 PRINT "HELLO JOE"
70 END
80 PRINT "HELLO SALLY"
90 END
100 PRINT "HELLO ";N$
110 END
```

The IF.....THEN statement in lines: 30 and 40, tell the computer that under the condition that the names Joe and Sally are entered, to go to a specific line.

Lines 50 and 100 are the lines the computer "reads" if neither Joe or Sally are entered, and therefore prints the name with a "Hello".

Notice the space after Hello in line 100. This is so the INPUT, N$, can be printed. You MUST leave a space or the words will be run together.

Also notice the semicolon after the quotations. This must be there in order to seperate the two units, Hello and the name.

END:

The END command simply tells the computer to terminate the program.

For instance, notice the three END statements in our last program. This is so that after either Joe or Sally is "greeted" by the computer, the computer will stop the program.

RUN:

The RUN command puts your BASIC, or any other program for that matter, into operation. You simply type in RUN and press the Return Key. (Or Enter Key, depending on the computer you are using.)

REM:

The REM statement, or REMark statement, has no effect on the operator. The REM statement allows the author of the program to put little notes in the program. These notes have no effect on the program or the operator.

Take this program for example:

```
10 REM GEORGE HENRY
20 REM WAS HERE
30 PRINT "TREE"
40 GOTO 10
```

Lines 10 and 20 have no effect on the rest of the program. They don't print anything, and they don't tell the computer to go anywhere. Also notice that REM statements don't need quotation marks.

INT:

INT stands for integer. This command tells the computer to turn any decimal number into a whole number.

For instance, let's say the computer comes up with the number: 6.3893. The INT statement will turn this number into the whole number: 6.

The INT command is used with the following command:

RND:

RND stands for random or randomize. The RND statement generates numbers from 0 to any given number.

Example:

```
10  S INT(10*RND)
20  PRINT S
30  END
```

Line 10 will generate random numbers from 0 to 10. We use the INT statement so that if the computer comes up with a number like: 1.3, the computer will change the number to: 1.

You MUST place an asterik (*) between the number and the RND statement.

LET:

The LET command lets you pick a space in the program's memory and label it.

You generally use the LET command when you will be using a word or a number alot.

For example:

```
10 LET M$="PIE"
20 LET A$="APPLE"
30 PRINT A$;M$
40 END
```

The value of M$ is "PIE" and the value of A$ is "APPLE". Therefore the computer will print: "APPLE PIE".

When using the LET command, you MUST ALWAYS put quotation marks around the word that the STRING VARIABLE stands for.

However, you do not have to use quotation marks when the character the STRING VARIABLE stands for is a number.

TAB:

The TAB statement allows you to position the output anywhere on the screen.

Example:

```
10 PRINT TAB(15);"BASIC"
20 END
```

Line 10 will print BASIC 15 spaces from the right. The parenthesis are used to tell the computer how many spaces over to the right, to position the output.

FOR/NEXT:

The FOR/NEXT command is used in a loop. The FOR starts the loop, and the NEXT ends the loop.

Example:

```
10 PRINT "BASIC POWER"
20 FOR PAUSE=1 TO 500
30 NEXT PAUSE
40 GOTO 10
```

After BASIC POWER is printed, the FOR starts the 500 count pause, and the NEXT ends the pause.

GOSUB:

The GOSUB command is semi-related to the GOTO command. the GOSUB statement tells the computer to go to a certain line in the program, like the GOTO statement. But unlike the GOTO statement, the GOSUB statement allows you to return to the place where you left off.

For instance if you used the GOSUB command to go to line 60 from line 30, you could return to line 40.

This is done with the next command:

RETURN:

The RETURN command does just that. It RETURNS the GOSUB command back to the line it was on. You will understand fully the two commands: GOSUB and RETURN in a later chapter.

Chapter Two

Building BASIC programs is a very useful learning experience in developing your computer skills.

The commands that have been taught to you from this book will be very helpful in our first project: SIMULATIONS.

Our programs will simulate something that is real.

For instance, simulations have been written for a dice roll, a number-guessing game, and even a flight simulator.

So, with my help, you'll write your own simulations.

First, let's start with an easy warm-up. Let's simulate a decision maker.

A decision maker lets you ask a yes or no question, and then answers the question.

To start off with, we need the computer to let you enter a question. We'll do that like this:

```
10 PRINT "WHAT IS YOUR QUESTION?"
```

Since it asks you a question, you must answer. So, we need an INPUT:

```
20 INPUT Q$
```

NOTE: Q$ was chosen because Q is the first letter in Question.

Now we need the computer to randomly pick the answer (Yes or No).

So we must think in mathematical terms. Let's have the computer generate random numbers from 0 to 100, and all of the numbers greater than 49 will be "Yes". And all of the numbers less than 49 will be "No".

30 A INT(100*RND)

After each answer is given, let's pause for a while before we enter another question. We'll use our FOR/NEXT command for this task.

So, all together, the decision maker will look like this:

```
10 PRINT "WHAT IS YOUR QUESTION?"
20 INPUT Q$
30 A=INT(100*RND)
40 IF A)49 THEN 60
50 IF A(49 THEN 100
60 PRINT "YES"
70 FOR PAUSE=1 TO 500
80 NEXT PAUSE
90 GOTO 10
100 PRINT "NO"
110 FOR PAUSE=1 TO 500
120 NEXT PAUSE
130 GOTO 10
RUN
```

Now that you have done your first SIMULATION, let's write a SIMULATION for each of the things listed below:

Password System

Coin Toss

Filing System

Decision Maker II

Turn to the section headed SIMULATIONS when you are ready to write these SIMULATIONS. It'll tell you all you'll need to know to write the SIMULATIONS. It also provides work space. But FIRST read this:

WRITING A BASIC PROGRAM:

When you are writing a BASIC program, just check for these things:

1. PURPOSE
2. INPUT
3. OUTPUT

If you check for these things, you've got the skeleton for a BASIC program.

Let's take the Decision Maker simulation, for instance.

(1) The PURPOSE was for the computer to answer your question.

(2) The INPUT was the question you asked.

(3) The OUTPUT was the answer, yes or no.

So, if you remember these three things, you should be able to write any BASIC program you want.

Chapter Three

This portion of the book is for you to experiment by building different programs from the instructions given. The answers are at the back of this book.

The programs will start out easy, but will get harder.

You may write your program in the space provided.

****

A.

1. Write a program that will print your name once:

10

20

2. Write a program that will print your name in a column up and down the screen:

10

20

3. Using STRING VARIABLES, write a program that will print your name in a column six times and then terminate:

10

20

30

40

50

60

70

80

4. Using the semicolon, write a program that prints a screen full of your name:

10

20

B.

1. Using STRING VARIABLES and INPUT, write a program that will ask you your name and print it once:

10

20

30

40

2. Using STRING VARIABLES, INPUT, and the SEMICOLON, create a program that will ask you your name and then greet you by printing: HELLO, (YOUR NAME) :

10

20

30

EXPERIMENT:

Try to write the program described below. It involves something we haven't covered yet: BASIC MATH.

If you cannot figure it out, wait until you've read the chapter: BASIC MATH.

Give it your best shot:

Using STRING VARIABLES, INPUT, the LET command, and BASIC MATH, write a program that will let you INPUT two numbers and will multiply them. Then have the computer print the OUTPUT:

10

20

30

40

50

60

70

Chapter Four

SHORT PROGRAMS:

Now it's my turn to show you some programs that I've written. There is nothing for you to do here, unless you want to run these programs on your computer.

PROGRAM #1:

Here's one to use when you want to display the name of your program. TITLE DISPLAY.....

```
10 PRINT "####################"
20 PRINT "#                  #"
30 PRINT "#    (TITLE)       #"
40 PRINT "#                  #"
50 PRINT "####################"
60 END
RUN
```

Line 30 is where you print the title of your program.

PROGRAM #2:

Here's one that prints random numbers in a strip.

RANDOM NUMBER FILMSTRIP.....

```
10 A=INT(9*RND)
20 IF A<0 THEN 10
30 IF A>9 THEN 10
40 PRINT "**********"
50 PRINT "*        *"
60 PRINT "*  ";A;"  *"
```

```
70  PRINT "*          *"
80  PRINT "**********"
90  FOR  PAUSE=1  TO  500
100  NEXT  PAUSE
110  GOTO  10
```

PROGRAM #3:

Here's one that will tell you how old you'll be in 23 years:

```
10  PRINT "HOW  OLD  ARE  YOU?"
20  INPUT  A$
30  LET  T=A$+23
40  PRINT "YOU  WILL  BE  ";T;"  YEARS  OLD"
50  PRINT "IN  TWENTY-THREE  YEARS!
60  END
```

PROGRAM #4:

Here's one that will figure out how much money you will make in a given amount of hours:

```
10  PRINT "WAGES  PER  HOUR?"
20  INPUT  W$
30  PRINT "HOW  MANY  HOURS  DO  YOU  WORK?"
40  INPUT  H$
50  LET  T=W$*H$
60  PRINT "YOU'VE  MADE  ";T;"  DOLLARS!
70  END
```

PROGRAM #5:

Here's one that will figure out how much money you'll get for a given number of ounces of gold.

```
10  PRINT  "DOLLARS  PER  OUNCE?"
20  INPUT  D$
30  PRINT  "OUNCES?"
40  INPUT  O$
50  LET  T D$*O$
60  PRINT  "YOU'LL  GET  ";T;" DOLLARS"
70  PRINT  "FOR  YOUR  ";O$;" OUNCES!
80  END
```

Chapter Five

SIMULATIONS:

If you have not read, BUILDING BASIC PROGRAMS, do not read this section yet. Go back and read: BUILDING BASIC PROGRAMS.

You have learned how to create a simulation from reading: BUILDING BASIC PROGRAMS. Now is your chance to create your own simulations.

Just use the information given and you'll be able to create a simulation in no time.

The answers are in the back of the book.

SIMULATION #1:

Create a simulation that will ask you for a file name, access code, and file contents. Then have the computer relay the information back to you. End the simulation by having the operator enter the file date.

SIMULATION:

SIMULATION::

SIMULATION #2:

Now create a password system simulation that will let four people access a file.

Your simulation must include the following: four different account numbers, and four different passwords.

NOTE: The account numbers and passwords do not be the same as in the back of the book.

Your simulation must first ask for an account number, then the password.

SIMULATION:

SIMULATION:

SIMULATION #3:

This simulation will generate random numbers from 0 to 100. All of the numbers smaller than 50, will be "HEADS". And all the numbers larger than 50 will be "TAILS".

Have the operator press The RETURN KEY to flip the coin.

SIMULATION:

SIMULATION:

SIMULATION #4:

This simulation is called: Decision Maker II. It is like the Decision Maker we made earlier, but this time the computer answers with five different answers: Maybe, Sometimes, Yes, No, or Ask Again. Create this simulation:

SIMULATION:

Chapter Six

Using your computer, you can run many mathematical operations. The signs are:

\+ Addition

\- Subtraction

* Multiplication

/ Division

Some computers operate like this:

```
PRINT 3*4
12
```

But on some other computers all you have to do id:

```
3*4
12
```

Back on the program I wrote, the one that tells you how much money you make, I used BASIC MATH.

After you entered how much money you made and how many hours you worked, I had the computer multiply them together.

It's the same with the program that tells you how old you'll be in 23 years. After you entered your age, I had the computer add 23 to it.

Now if you haven't figured out the last program on the Program Workbook, do it now.

Conclusion

I hope you have gained something from this experience. I hope I've gotten you more familiar with your computer.

If you were already familiar with your computer, I hope I enriched your computer skills.

Michael Bradshaw

ANSWERS

Program Workbook:

A.

1.
```
10 PRINT "(YOUR NAME)"
20 END
RUN
```

2.
```
10 PRINT "(YOUR NAME)"
20 GOTO 10
RUN
```

3.
```
10 LET A$="(YOUR NAME)"
20 PRINT A$
30 PRINT A$
40 PRINT A$
50 PRINT A$
60 PRINT A$
70 PRINT A$
80 END
RUN
```

4.
```
10 PRINT "(YOUR NAME)";
20 GOTO 10
RUN
```

Program Workbook:

B.

1.
```
10 PRINT "WHAT'S YOUR NAME?"
20 INPUT N$
30 PRINT N$
40 END
RUN
```

NOTE: The STRING VARIABLES in lines 20 and 30 do not have to be N$. They can be A$, B$, C$, D$, etc. Just as long as the STRING VARIABLES in both lines 20 and 30 are the same.

2.
```
10 PRINT "WHAT'S YOUR NAME?"
20 INPUT N$
30 PRINT "HELLO ";N$
```

3.
```
10 PRINT "ENTER THE TWO NUMBERS"
20 PRINT "YOU WANT MULTIPLIED."
30 INPUT A$
40 INPUT B$
50 LET C=A$*B$
60 PRINT C
70 END
RUN
```

Simulations:

NOTE: Your answers might differ from these given here, but I'm sure they'll be the same concept.

Here's the answer to the Filing System simulation:

```
10  PRINT  "FILE  NAME:"
20  INPUT  F$
30  FOR  PAUSE=1  TO  500
40  NEXT  PAUSE
50  PRINT  "ACCESS  CODE:"
60  INPUT  A$
70  FOR  PAUSE=1  TO  500
80  NEXT  PAUSE
90  PRINT  "FILE  CONTENTS:"
100  INPUT  C$
110  FOR  PAUSE=1  TO  500
120  NEXT  PAUSE
130  PRINT
140  PRINT  "FILE  NAME:  ";F$
150  PRINT
160  PRINT  "ACCESS  CODE:  ";A$
170  PRINT
180  PRINT  "FILE  CONTENTS:  ";c$
190  FOR  PAUSE=1  TO  500
```

```
200  NEXT  PAUSE
210  PRINT
220  PRINT  "ENTER  FILE  DATE:"
230  INPUT  D$
240  FOR  PAUSE=1  TO  500
250  NEXT  PAUSE
260  PRINT
270  PRINT  "THE  FILE  ";F$
280  PRINT  "HAS  BEEN  CLOSED!
290  END
RUN
```

Here's the answer to the Password System simulation:

```
10  PRINT  "WHAT'S  YOUR  ACCOUNT  NUMBER?"
20  INPUT  N$
30  IF  N$="10101"  THEN  70
40  IF  N$="20202"  THEN  70
50  IF  N$="30303"  THEN  70
60  IF  N$="40404"  THEN  70
65  GOTO  66
66  PRINT  "ACCOUNT  NUMBER  IS  ILLEGAL"
67  FOR  PAUSE=1  TO  500
68  NEXT  PAUSE
69  GOTO  10
70  FOR  PAUSE=1  TO  500
80  NEXT  PAUSE
90  PRINT
100  PRINT  "WHAT'S  THE  PASSWORD?"
```

```
110  INPUT  P$
120  IF  P$="GALAXY"  THEN  210
130  IF  P$="STARS"  THEN  210
140  IF  P$="PLANETS"  THEN  210
150  IF  P$="SPACE"  THEN  210
160  GOTO  170
170  FOR  PAUSE=1  TO  500
180  NEXT  PAUSE
190  PRINT  "YOUR  PASSWORD  IS  ILLEGAL."
200  GOTO  100
210  FOR  PAUSE=1  TO  500
220  NEXT  PAUSE
230  PRINT  "YOUR  ACCOUNT  NUMBERS"
240  PRINT  "AND  PASSWORD  HAVE"
250  PRINT  "BEEN  VERIFIED....."
260  PRINT
270  PRINT  "YOU  HAVE  TOTAL  ACCESS..."
280  END
RUN
```

Here's the Coin Toss simulation:

```
10  A=INT(100*RND)
20  IF  A<0  THEN  10
30  IF  A<50  THEN  50
40  IF  A>50  THEN  70
50  PRINT "HEADS"
55  PRINT  "PRESS  RETURN...."
56  INPUT  K$
57  PRINT
60  GOTO  10
70  PRINT  "TAILS"
75  PRINT  "PRESS  RETURN...."
76  INPUT  K$
77  PRINT
80  GOTO  10
```

Here's the answer to the Decision Maker II....

```
10  PRINT "WHAT IS YOUR QUESTION?"
20  INPUT Q$
30  S=INT(5*RND)
40  IF S=1 THEN 90
50  IF S=2 THEN 130
60  IF S=3 THEN 170
70  IF S=4 THEN 210
80  IF S=5 THEN 250
90  PRINT "MAYBE"
100  FOR PAUSE=1 TO 500
110  NEXT PAUSE
120  GOTO 10
130  PRINT "SOMETIMES"
140  FOR PAUSE=1 TO 500
150  NEXT PAUSE
160  GOTO 10
170  PRINT "YES"
180  FOR PAUSE=1 TO 500
190  NEXT PAUSE
200  GOTO 10
210  PRINT "NO"
220  FOR PAUSE=1 TO 500
230  NEXT PAUSE
240  GOTO 10
```

```
250  PRINT  "ASK  AGAIN"
260  FOR  PAUSE=1  TO  500
270  NEXT  PAUSE
280  GOTO  10
```