

BASIC FOR THE APPLE II®

A SELF-TEACHING GUIDE



GERALD R. BROWN
HERCULE FINKEL
BOB ALBRECHT

BASIC FOR THE APPLE II®

**JERALD R. BROWN
LEROY FINKEL
ROBERT L. ALBRECHT**

Dymax Corporation
Menlo Park, California



John Wiley & Sons, Inc.
New York • Chichester • Brisbane • Toronto • Singapore

Publisher: Judy V. Wilson
Editor: Dianne Littwin
Composition & Make-up: Cobb/Dunlop, Inc.

Revised and adapted by Jerald R. Brown for Applesoft BASIC from BASIC FOR HOME COMPUTERS by Albrecht, Finkel, and Brown, at Rainbow's End Farm, Sebastopol, California, and at Mr. F. Giovanni Smith's Nibble and Co., and Corol and George Bevier's Rum Point Inn, in Belize, Central America. Programs were checked by Paul Armitige.

Copyright © 1982, by John Wiley & Sons, Inc.

All rights reserved. Published simultaneously in Canada.

Reproduction or translation of any part of this work beyond that permitted by Section 107 or 108 of the 1976 United States Copyright Act without the permission of the copyright owner is unlawful. Requests for permission or further information should be addressed to the Permissions Department, John Wiley & Sons, Inc.

Library of Congress Cataloging in Publication Data

Brown, Jerald, 1940-

BASIC for the APPLE II.

(Wiley self-teaching guide)

Includes index.

1. Apple II (Computer)—Programming. 2. Basic
(Computer program language) I. Finkel, Leroy. II. Al-
brecht, Bob, 1930- III. Title.

QA76.8.A662B76 1982 001.64'2 82-10962

ISBN 0-471-86596-6

Printed in the United States of America

83 84 10 9 8 7 6 5 4 3

To the Reader

With the advent of integrated circuits and miniaturization in electronics, suddenly a complete computer, with all the peripherals or attachments needed to use it, can be purchased for as little as \$600. Now a computer for your business, home, club, or school is in the same price range as the less expensive stereo component systems! Many of these inexpensive computers, including the Apple®,* use the computer language BASIC.

BASIC was developed at Dartmouth College by John Kemeny and Thomas Kurtz, who recognized the need for an all-purpose computer language that would be suitable for beginning programmers whose educational backgrounds would be varied and diverse. Beginners All-purpose Symbolic Instruction Code (BASIC) was originally designed as a simple language which could be learned in a few short hours. With improvements over the years, the language now may take a few days to learn, but you will find that you can do nearly anything you want in BASIC.

The BASIC you learn in this book will apply to any computer that “understands” a similar version of BASIC. To show you how to converse with a computer using BASIC programs, we have used the particular version of BASIC called Applesoft®* BASIC, developed by the Microsoft Corporation for the Apple computer. (Interestingly, some of the most advanced and versatile versions of BASIC are those developed by various manufacturers of personal computers.) Most versions of BASIC are similar, the more so because Microsoft has been developing similar forms of BASIC for many recent computer models.

Apple computers have been supplied with two versions of BASIC: Integer BASIC, and Applesoft (also called Floating Point) BASIC. This book teaches Applesoft BASIC, the more versatile and advanced version of the two. In particular, the example programs and activities were developed, adapted, and tested on an Apple II Plus, with 48K of memory and one disk drive. However, the minimum system with Applesoft BASIC, 16K and no disk drive or cassette tape machine, is quite sufficient for the programs and activities in this Self-Teaching Guide.

If you have an Apple computer, be certain that Applesoft BASIC is available to use, so that the computer will understand the instructions you will be learning to give it in Applesoft BASIC.

*Apple® and Applesoft® are registered trademarks of Apple Computer, Inc.

How to Use This Book

Instructors will find this book to be excellent for courses in BASIC programming at the high school and college levels, as well as for computer center classes, university extension workshops, and in-house instructional settings.

With this book's self-instructional format, you'll be actively involved in learning BASIC. The material is presented in short numbered sections called frames, each of which teaches you something new about BASIC and gives you a question or asks you to write a program. Correct answers are given following the dashed line. For the most effective learning we urge you to use a thick paper to keep the answers out of sight until you have written your answer.

You will learn best if you take pen or pencil in hand and actually write out the answers or programs. The questions are carefully designed to call your attention to important points in the examples and explanations, and to help you learn to apply what is being explained or demonstrated.

Each chapter begins with a list of objectives—what you will be able to do after completing that chapter. If you have had some previous experience using BASIC and the objectives for that chapter look familiar, take the Self-Test at the end of that chapter first to see where you should start your close reading of the book. If you do well, study only the frames indicated for the questions you missed. If you miss many questions, start work at the beginning of that chapter.

The Self-Test can also be used as a review of the material covered in the chapter. You may test yourself immediately after reading the chapter. Or you may wish to read a chapter, take a break, and save the Self-Test as a review before you begin the next chapter. At the end of the book is a Final Self-Test which will allow you to test your overall understanding of BASIC.

This is a self-contained book for learning the computer language called BASIC. You do not need access to a computer to learn BASIC. However, what you learn will be theoretical until you actually sit down at a computer and apply your knowledge of the computer language and programming techniques. So we strongly recommend that you and this book get together with a computer. BASIC will be easier and clearer if you have even occasional access to a computer so that you can try the examples and exercises, make your own modifications, and invent your own programs for your own purposes. You are now ready to teach yourself how to use BASIC.

Contents

	To the Reader	iv
	How to Use This Book	v
Chapter 1	Introduction	1
Chapter 2	Getting Started	9
Chapter 3	Assignment Statements, Stored Programs, and Branching	41
Chapter 4	Decisions Using IF-THEN Statements	97
Chapter 5	READ and DATA Work Together	141
Chapter 6	FOR-NEXT Loops	183
Chapter 7	Low-Resolution Graphics	213
Chapter 8	String Variables and String Functions	239
Chapter 9	Subscripted Variables	265
Chapter 10	Double Subscripts	309
Chapter 11	Subroutines	357
	Final Self-Test	383
	BASIC Functions	397
Appendix	ASCII Character Codes	401
	Personal Computing Periodicals	403
	Program Index	405
	Index	407

CHAPTER ONE

Introduction

This chapter will introduce you to some of the vocabulary you will encounter in the world of small computers, whether they are used in business, home, school, or club. You will *not* need access to a computer to complete this chapter. When you complete this chapter, you will know more about such computers. You will also be able to use the following words and phrases from the language of the computer world:

- Computer language
- BASIC and, specifically Applesoft™ BASIC
- Keyboard
- Printer
- Disk drive
- Video monitor or display screen
- Program and programming
- BASIC statements
- Line numbers

1. We will start slowly and simply, easing you gently into the world of Apple computers and BASIC. Just keep in mind that this is intended to be a friendly book about personal computing and an easy-to-learn, easy-to-use computer language called BASIC. We will do our best to teach you how to read and understand computer programs written in BASIC and to help you get started writing original, never-before-seen-on-earth programs, *your* programs. The Apple computer uses a version of BASIC called Applesoft BASIC, which was produced by MICROSOFT, a company that has provided similar versions of BASIC for Radio Shack, IBM, recently for Atari, and versions of BASIC that can be implemented (used) on a wide variety of brands of computers. Apple is the brand name of a microcomputer manufactured by Apple Computer, Inc.

For this book to be useful to you, only one thing is essential: your computer must speak Applesoft, a version of the computer language called _____

BASIC

2. The personal computer we use speaks a version called Applesoft BASIC. Most BASICs used on personal computers are very similar to MICROSOFT BASIC. The differences that exist are minor; even though this book emphasizes Applesoft BASIC, you should find it useful no matter what particular BASIC you are using. (The reference manual for your version of BASIC will come in handy.)

Our Apple computer looks like the one shown below.



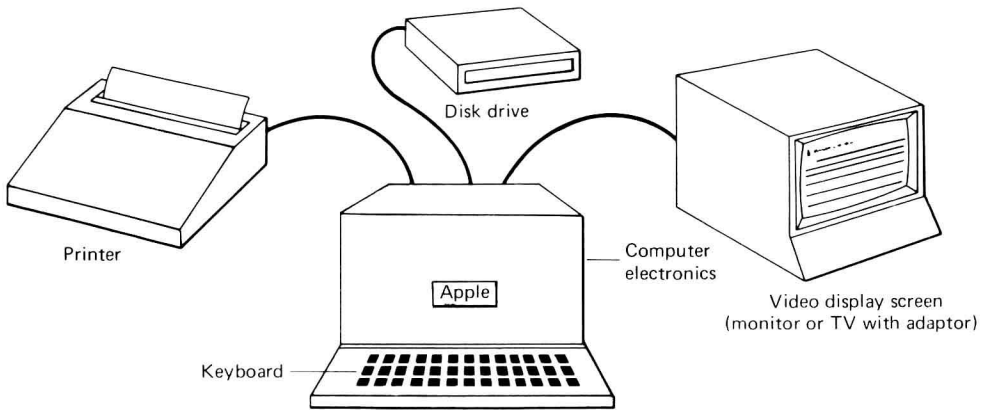
Photo: David Burns

Our Apple computer system consists of an Apple II PLUS computer connected to a television set. The Apple includes a built-in keyboard similar to an electric typewriter keyboard. We use the keyboard to type information into the computer. The “computer” itself is the electronics inside the Apple’s case, in back of the keyboard. The computer, with its keyboard, video display screen, and any other attachments, including printer, disk drive, or cassette recorder, is referred to as a computer system. The attachments are often called peripherals, since the computer itself is the core or center of the system.

As we type characters, the information we type is sent to the computer and also is displayed on the screen, for our reference. Information originated by the computer is also displayed on the screen, whether the display is a video monitor or a standard television equipped with an adaptor for connection to the computer. Thus, the combination of keyboard and display screen provides (one-way or two-way—choose the correct answer) _____ communication with the computer.

two-way

3. Below is a sketch of a computer system based around the Apple computer.



(a) We use the keyboard to type information into the computer. What happens to the information that we type?_____

(b) How does the computer communicate with us, the users?_____

(a) It is sent to the computer and is also displayed on the screen.

(b) It displays information on the video display screen (TV or monitor).

4. The cassette recorder and the more popular (and expensive) disk drive provide memory storage which is separate from the computer's own electronic memory. Both information storage systems can save or record programs or play them back into the computer's memory. Disks, in addition, can store information (or data) such as mailing lists, appointments, scientific data, and budget or accounting figures. Once you have begun to master Applesoft BASIC, you will find these storage devices quite handy and easy to use. Two reasons for such external storage devices are (1) the computer's electronic memory goes blank when the power is turned off; and (2) the computer's memory can only hold one BASIC program at a time. Therefore, such recorded programs are very handy for changing one program for another inside the computer itself. All such external storage methods are convenient ways to feed or "load" programs and information you often use into the computer without taking all the time needed to type them in from a keyboard. Punched paper tape, magnetic disks, and even phonograph recordings are external storage methods. However, in this book you need only the three main parts of a computer system we have discussed earlier, which are _____

a computer, a keyboard, and a display device (printer or video screen)

5. In this book, we will show you many programs in Applesoft BASIC, and will help you learn to read, understand, and use these programs for your own enjoyment. We will concentrate on applications that we think will be of interest to users of home/school/personal computers. And so . . . for our appetizer, a computer game.

This program may be "Greek" to you now, but by the time you finish Chapter 4, you will not only be able to read this program, but be able to write a program at least as complex as this one. For now, however, we just want to familiarize you with the form of a computer program and the names of its parts. An understanding of the actual instructions to the computer contained in the program will come later.

These lines, down to the word RUN, are a computer program.

```
100 REM *** THIS IS A SIMPLE COMPUTER GAME
110 LET X = INT(100*RND(1))+1
120 PRINT
130 PRINT "I'M THINKING OF A NUMBER FROM 1 TO 100."
140 PRINT "GUESS MY NUMBER!!!"
150 PRINT : INPUT "YOUR GUESS?"; G
160 IF G<X THEN PRINT "TRY A BIGGER NUMBER." : GOTO 150
170 IF G>X THEN PRINT "TRY A SMALLER NUMBER." : GOTO 150
180 IF G=X THEN PRINT "THAT'S IT!!! YOU GUESSED MY NUMBER." : GOTO 110
```

RUN

I'M THINKING OF A NUMBER FROM 1 TO 100.
GUESS MY NUMBER!!!

And this is what the computer
does that appears on the display
screen.

YOUR GUESS? 50
TRY A BIGGER NUMBER.

YOUR GUESS? 90
TRY A SMALLER NUMBER.

YOUR GUESS? 75
TRY A SMALLER NUMBER.

The guesses are typed by the computer
user.

YOUR GUESS? 65
TRY A SMALLER NUMBER.

YOUR GUESS? 58
TRY A BIGGER NUMBER.

YOUR GUESS? 62
TRY A SMALLER NUMBER.

YOUR GUESS? 60
TRY A BIGGER NUMBER.

This part, from the word RUN on
down, is called the RUN of the pro-
gram. Following the instructions in
the program, the computer generates a
random number from 1 to 100. The
player types in guesses. After each
guess, the computer types a clue to
help the player make a better guess.

YOUR GUESS? 61
THAT'S IT!!! YOU GUESSED MY NUMBER.

I'M THINKING OF A NUMBER FROM 1 TO 100.
GUESS MY NUMBER!!!

YOUR GUESS? And so on. The game continues. Chances are that the computer
will have a different number this time.

If this sounds confusing, read on! All will be revealed. And it won't be long
before you can read and write programs like this in BASIC.

Look again at the program shown in our computer game. The program consists of nine (9) lines, each containing one or more BASIC statements. Each line begins with a line number.

This is a line number.

130 PRINT "I'M THINKING OF A NUMBER FROM 1 TO 100."

This is a statement.

In our program, each numbered line contains one or more BASIC _____
_____. The numbers 100 through 180 are called _____

statements; line numbers

6. The program in frame 5 was typed one line at a time on the keyboard. As we typed it, the program was stored in the computer and also displayed on the screen.

On a video display, as we type the program, it will be stored in the computer and also will appear on the _____ .

screen

Note: The Apple video screen displays 24 lines. So, if the screen is filled (all 24 lines used), new information typed in will cause old information to be "pushed off" the top of the screen, but not out of the computer's memory.

7. First, we typed in the entire program (lines 100 through 180). This process is called "entering the program." This stored the program in the computer's memory. Then we typed RUN. This tells the computer to RUN, or carry out, the program. Computer people also say "to execute the program." In other words, after storing the program, we then told the computer to follow the instructions (statements) of the program, or execute the program.

If there is a program in the computer's memory, then typing RUN tells the computer to _____ .

carry out or execute the instructions in the program

8. During the RUN, the computer obeyed the instructions (statements) in the program, as follows: First, the computer generated a random number from 1 to 100, inclusive (line 110). This number is an integer—a “whole” number with no fractional part.

```
110 LET X = INT(100*RND(1))+1
```

In the chapters to come, you will learn how this curious combination of letters, numbers, and symbols tells the computer to “think up” (generate) a random number.

Next, the computer typed instructions to the player (lines 120, 130, and 140).

```
120 PRINT
130 PRINT "I'M THINKING OF A NUMBER FROM 1 TO 100."
140 PRINT "GUESS MY NUMBER!!!"
```

Compare these statements with what the computer causes to be displayed immediately under the word RUN.

Then, the computer asked for a guess (line 150).

```
150 PRINT : INPUT "YOUR GUESS?"; G
```

After the player typed a guess, the computer compared the guess with its secret number and gave the player the appropriate response (lines 160, 170, and 180).

```
160 IF G<X THEN PRINT "TRY A BIGGER NUMBER." : GOTO 150
170 IF G>X THEN PRINT "TRY A SMALLER NUMBER." : GOTO 150
180 IF G=X THEN PRINT "THAT'S IT!!! YOU GUESSED MY NUMBER." : GOTO 110
```

If the player did not guess the computer’s number, the computer went back to line 150 and asked for another guess. But if the lucky player *did* guess the secret number, the computer acknowledged the correct guess and went back to line 110 to “think” of another number.

```
100 REM *** THIS IS A SIMPLE COMPUTER GAME
```

Oh, yes, line 100 is a REM (REMark) statement. It doesn’t tell the computer to *do* anything. It is simply included to tell something *about* the program to us humans who may read the program itself.

What do we type to instruct the computer to execute or carry out a program?

RUN

Now, to see how much you've learned from this first chapter, try the Self-Test. Then on to Chapter 2, where you'll start to learn how to actually use the computer.

Self-Test

Try this Self-Test, so you can evaluate how much you have learned so far.

1. If your computer does not have a printer in addition to the keyboard for communicating with the computer, what other kind of display device would you expect it to have?_____
2. If another computer "speaks" a different BASIC than your computer does, would you expect your computer to "understand" the terminology taught in this book?_____
3. Since a computer doesn't have legs, what do we want a computer to do when we tell it to RUN?_____
4. What do we call the number that begins a statement in a program?

Answers to Self-Test

The frame numbers in parentheses refer to the frames in the chapter where the topic is discussed. You may wish to refer to these for quick review.

1. A video display (TV or monitor). (frames 2-4, 6)
 2. Yes, because most versions of BASIC for personal computers are like the one we use, with only minor differences which will quickly become obvious. (frame 2)
 3. Execute the program (follow the instructions we give it). (frames 5-9)
 4. Line number. (frame 5)
-

CHAPTER TWO

Getting Started

To get you started in computer programming in BASIC, we will now introduce you to some of the statements used to instruct the computer, that is, to tell it what you want accomplished. In this chapter, you will use the *direct*, or immediate, mode of operation. Using direct mode, you tell the computer something to do, and it does it immediately. When you complete this chapter, you will be able to:

- Use direct statements to instruct the computer.
- Recognize error messages from the computer.
- Use the PRINT statement with quotation marks to print strings (messages).
- Correct typing errors or delete a statement with errors.
- Use direct statements to do arithmetic.
- Compute values of simple mathematical expressions using the symbols and rules of BASIC for arithmetic.
- Use the short form for PRINT statements—a question mark (? = PRINT).
- Recognize and convert floating-point or E notation to ordinary numbers.

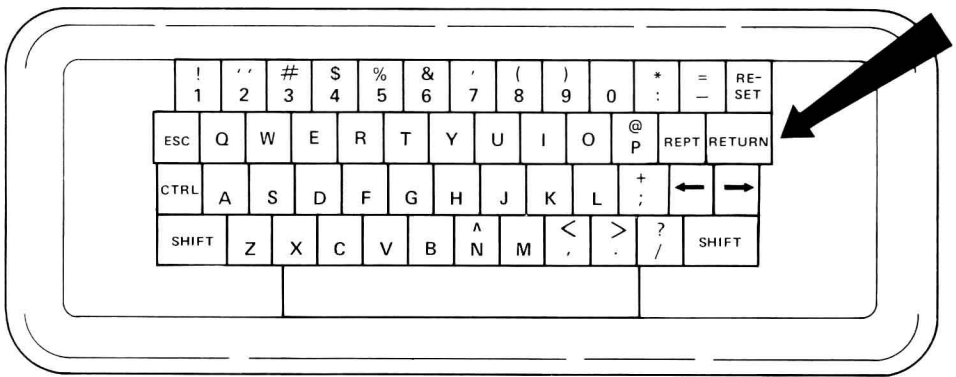
1. Now we can begin “talking” to the computer. The computer is plugged in, the video display is connected to the computer and turned on, and the computer itself is turned on.

If a bracket like this] appears on the display screen, with the cursor (a little square of light) flashing next to the bracket, then your Apple is ready to serve you with the facilities of Applesoft BASIC.

The reference materials from Apple Computer, Inc. include the *BASIC Programming Reference Manual for Applesoft BASIC*, *The Applesoft Tutorial*, and the *DOS (Disk Operating System) Manual*. The *Tutorial* provides an introduction to using the Apple keyboard to good effect and may be helpful if you haven’t been shown how to get your particular Apple system started. This is especially true if Applesoft BASIC doesn’t appear automatically when you turn on the power switch. *The Applesoft Tutorial* tends to emphasize many features unique to the Apple at the expense of providing a thorough background in the fundamentals of BASIC programming. The *Self-Teaching Guide* you are reading provides a systematic approach to acquiring these fundamentals, oriented toward the beginner. Instead of presenting a lot of definitions and then expecting you to

use a batch of these concepts together, we present one thing at a time and carefully build up your knowledge of programming by adding to previous concepts. Ideas are introduced one at a time, followed by examples of useful ways to combine them with other BASIC instructions. Both the *Tutorial* and *Reference Manual* provide supplementary and advanced information of value to those using and completing this *Self-Teaching Guide*.

We will start by assuming that we are using a keyboard and video display to communicate with the computer. Here is a diagram of the Apple keyboard.



In the diagram, there is a large black arrow pointing to the key labeled

_____ .

RETURN

2. If you press RETURN (without typing anything else), you will see the cursor (that little block of light) jump down one line on the display screen, and sit there flashing next to the bracket ([]). On a printer, the paper will space up one line, performing what is called a *line feed*. If you have an Apple computer at hand, you could try this: Type your name and press RETURN. The computer will probably print an error message on the printer.

You type: BOB, LEROY, AND JERRY
 It types: ?SYNTAX ERROR

By "it" we mean the computer.
 Saves space in writing this book!

Hmmmm . . . we know what ERROR means, but what about SYNTAX? You see, the computer is of quite limited intelligence. It simply did not understand us.

BASIC can be thought of as a very simple-minded foreign language, similar to English but with a very small, limited vocabulary. However, there are very exact and strict rules as to the arrangement, or format, or syntax of its "sentences," that

is, the instructions we call program *statements*. So a SYNTAX ERROR may be an incorrectly spelled word that BASIC would normally understand, or the placing of a word or symbol in the wrong place, or the omission of some word or symbol necessary for BASIC to “understand” or interpret the statement or instruction. The SYNTAX ERROR message will alert you to typing errors, for one thing.

(a) If you type something, then press RETURN, and the computer types ?SYNTAX ERROR, what is the computer trying to tell you?_____

(b) After typing ?SYNTAX ERROR, what did the computer type?_____

(a) It does not understand you.

(b)] (the bracket indicating Applesoft BASIC is standing by for your next keyboard entry)

3. Despite the error message, no damage has been done. The computer is very patient and forgiving. It will let you make as many mistakes as you wish. Whenever you see] you know it is your turn to type.

To avoid misunderstandings with a computer, we must learn its language. We will start with some simple, one-line statements that the computer *does* understand.

In this chapter we will use direct statements. Direct statements do not have line numbers. When you type a direct statement and press RETURN, the computer executes the statement immediately, then forgets the statement. We call this BASIC's *direct mode* of operation. Here is an example of a statement that is “executed in direct mode.”

You type: PRINT “MY HUMAN UNDERSTANDS ME” Then press RETURN.

It types: MY HUMAN UNDERSTANDS ME
]

Now you complete the following. If you make a typing error, press RETURN and start again. Other ways of correcting errors will be shown shortly.

You type: PRINT “WAKE UP! BURGLARS ARE IN THE HOUSE.” Then press RETURN.

It types: _____

WAKE UP! BURGLARS ARE IN THE HOUSE.
]

Along with the wake-up message, we could also arrange (as we'll see later) to have the computer sound a beep several times or to sound some other audible alarm (with appropriate electronic connections).