DOS: A SHORT COURSE

FOR MS-DOS VERSIONS 5.0, 6.0/6.2
IBM PC-DOS VERSION 6.1

L. JOYCE ARNTSON
BARBARA F. STOCKLER

DOS: A SHORT COURSE

FOR MS-DOS VERSIONS 5.0, 6.0/6.2
IBM PC-DOS VERSION 6.1

L. JOYCE ARNTSON
Irvine Valley College

BARBARA F. STOCKLER
Coastline College

Wadsworth Publishing Company
Belmont, California
A Division of Wadsworth, Inc.

Publisher: Kathy Shields
Assistant Editor: Tamara Huggins
Production Editor: Karen Garrison
Print Buyer: Barbara Britton
Managing Designer: Andrew Ogus
Cover Design: Craig Hanson
Printer: Malloy Lithographing

This book is printed on acid-free recycled paper.

I(T)P
International Thompson Publishing
The trademark ITP is used under license.

© 1995 by Wadsworth, Inc. All rights reserved. No part of this book may be reproduced, stored in a retrieval system, or transcribed, in any form or by any means, without the prior written permission of the publisher, Wadsworth Publishing Company, Belmont, California 94002.

Printed in the United States of America

1 2 3 4 5 6 7 8 9 10—99 98 97 96 95

Library of Congress Cataloging in Publication Data

Arntson, L. Joyce.
DOS: a short course for MS-DOS versions 5.0, 6.0/6.2; and IBM PC-DOS version 6.1 / L. Joyce Arntson, Barbara F. Stockler.
p. cm.
Includes index.
ISBN 0-534-23406-2 (alk. paper)
1. MS-DOS (Computer file) I. Stockler, Barbara F. II. Title.
QA76.76.063A762 1994
005.4'469—dc20 94-6673

Preface

MS-DOS Versions 5.0, 6.0/6.2 and IBM PC-DOS Version 6.1: A Short Course has been written for the computer novice as well as the computer application wizard who has never taken the time to learn the Disk Operating System (DOS). Although this text assumes no prior computer experience or DOS knowledge and is directed to the novice, it can be used successfully by any individual who would like to brush up on his or her DOS skills. It is designed to help you learn the basics of DOS using versions 5.0, 6.0, 6.1, and 6.2 quickly and easily. Even though it was written specifically for versions 5.0, 6.0, 6.1, and 6.2, you will find the commands and operations to be largely compatible with all earlier versions. This book covers the important commands and operations of DOS to be covered in an introductory or short DOS course.

Learning DOS will prove to be invaluable as you work with software. Having DOS as your assistant can help you avoid traps and correct mistakes as you make them. Knowing DOS is a valuable tool as you work with the computer. It will provide the ability to perform numerous tasks from installing software and configuring the computer to disk and file management.

Because most computer users today have computers with hard disks, this text has been written with hard disk management in mind. Working with subdirectories and the DOS-based utilities will prove very useful in managing the hard disk.

Organization of the Text

This book is divided into ten chapters. Each chapter is written to facilitate learning with a read and do approach. The chapters begin with a list of objectives to prepare you for what is to be covered. For each new concept covered, you will find a reading assignment to explain the command or operation. Because nothing substitutes for hands-on practical application, each concept will be practiced after reading about it. All exercises are performance-based and have been written based on common usage by computer users in business and industry. To reinforce the lessons and so that you will be able to determine the abilities you will have mastered at their completion, each chapter ends with a Self-Check Quiz for reviewing the chapter and testing your understanding of the material. Having a perfect score is not as important as finding out what you do not understand about the concepts and operations. Always check your answers with those of your instructor and be sure to ask questions if you do not understand the answers.

You will find an assignment for each chapter which will challenge you to apply your new knowledge. These are designed to demonstrate how you can put DOS to work solving real-life practical problems.

We believe you will find DOS commands and operations worth the effort. With this knowledge you can take command of your computer and make it do your bidding. Be quick to ask questions so that you can gain a complete understanding of the concepts you are studying.

This book has been written for a short course in DOS. If you need more complete information, we suggest you consult the book MS/PC-DOS: Concepts, Exercises, and Applications written by the same authors.

Much apprciation goes to Trudy Gift of Hagerstown Jr. College and Floyd Winters, Manatee Community College. A great deal of the accuracy for this text may be attributed to the skills of the technical editor, Maria Tate, who tested all exercises and assignments many times.

Best wishes for success in learning and using DOS!

L. Joyce Arntson and Barbara F. Stockler

Contents

Chapter 1. An Introduction to the Microcomputer System

Objectives 1
Using this Textbook 1
The Disk Operating System 2
The Computer System 2
Software 11
Working with DOS 12
Self-Check Quiz 19
Assignment 20

Chapter 2. Basic DOS Commands

Objectives 21
Things to Remember When Using DOS 21
DOS Commands—Internal vs External 23
Clearing the Screen 24
Print Using Print Screen 24
Filenames and Extensions 25
Program Extensions and Device Names 26
The DIRECTORY 27
Wildcards 32
The Version (VER) Command 34
Asking for Help 35
Self-Check Quiz 36
Assignment 37

Chapter 3. Disk and File Preparation

Objectives 38
Floppy Diskettes 38
More on Internal versus External Commands 40
Setting the Search Path to the DOS Command Files 40
Aborting DOS Commands 41
Preparing Disks for Use 42
The Volume (VOL) Command 50
Specifying Disk Drives in Commands 50
Changing Default Drives 51
Using DOSKEY 52
Working with Text Files 53
Printing Files 57
Self-Check Quiz 59
Assignment 60

Chapter 4. File-Handling Commands

Objectives 62
Structure for File-Handling Commands 62
Copying Files 63
Comparing Files 74
Renaming Files with RENAME 76
DOS MOVE Command 78
DOS Editing Keys 79
Self-Check Quiz 82
Assignment 84

Chapter 5. Managing Files and Disks

Objectives 86
Deleting Files 86
The UNDELETE Command 87
Using MIRROR with DOS 5.0 to Undelete Files 89
UNDELETE using MS DOS 6.0/6.2 91
UNDELETE using IBM PC-DOS 6.1 93
UNFORMATTING Diskettes 95
Copying Disks 96
The DISKCOPY Command 96
Comparing Disks using the DISKCOMP Command 99
COPY *.* Compared to DISKCOPY 101
Checking a Disk with CHKDSK 103
SCANDISK (DOS 6.2) 105
The DEFRAG Command in PC-DOS 6.1 and MS-DOS 6.2 106
Self-Check Quiz 107
Assignment 109

Chapter 6. Creating and Using Subdirectories

Objectives 111
Introduction to Subdirectories 111
Creating a Subdirectory Structure 113
Specifying the PATH to a Directory 114
Changing the Current (Default) Directory 116
Printing Directory Listings 122
Searching for Files in a Tree Structure 122
Self-Check Quiz 124
Assignment 126

Chapter 7. Working with a Tree Structure

Objectives 127
Directory and File Management 127
Adding Subdirectories to Create a Third Level 129
Using the CD Command with Another Drive 130
Using Subdirectory Markers 131
Removing a Directory 132
Using the TREE Command to Display the Tree Structure 134
DOS 6.0 DELTREE Command 135
Renaming Directories 136
Using the XCOPY Command 137
Self-Check Quiz 140
Assignment 142

Chapter 8. An Introduction to EDLIN, the DOS Text Editor

Objectives 144
DOS Editing Techniques 144
Uses for EDLIN 145
Creating a Text File with EDLIN 145
Editing a Text File with EDLIN 150
The DOS Editing Keys for Editing in EDLIN 156
Self-Check Quiz 161
Assignment 163

Chapter 9. An Introduction to EDIT, The DOS Text Editor

Objectives 165
The DOS Editor 165
Installing the Mouse 166
Using EDIT 166
Using the Mouse 167
Selecting, Copying, Cutting, and Pasting Text 171
Printing a File 175
Using the Search Menu 175
Using HELP 177
Self-Check Quiz 178
Assignment 179

Chapter 10. An Introduction to BATCH Files

Objectives 181
Creating Simple Batch Files 181
Batch File Commands: REM, Pause, and ECHO 185
The AUTOEXEC.BAT File 188
CONFIG.SYS 190
Custom Menus 191
Self-Check Quiz 195
Assignment 196

Index 198

CHAPTER 1

AN INTRODUCTION TO THE MICROCOMPUTER SYSTEM

OBJECTIVES

When you have completed the activities and assignment in this chapter, you will be able to:

1. Explain basic concepts of microcomputer operation.
2. List and discuss the hardware components of a personal computer system.
3. Identify the parts of a floppy diskette and use the correct procedures for handling and storing floppy diskettes.
4. Describe features of the microcomputer system you will be using to learn DOS.
5. Load DOS starting with the computer off and on.
6. Recognize the DOS prompt.
7. Check and set the date and time in the computer's clock.
8. Set the prompt to display the current directory.

USING THIS TEXTBOOK

This textbook will be your guide and workbook for learning about DOS. To become thoroughly familiar with the operating system, we suggest you read the concepts in each chapter, complete the practice activities on the computer, and complete the quiz and assignment at the end of each chapter.

In this textbook, [Enter] denotes pressing the key marked Enter. (Do not key the [ ] or the word Enter.) Where commands to be keyed are shown, the letters in boldface indicate what should be keyed. You will see a bullet (■) preceding any required input. In the following example, you would key the letters DATE and then press the key marked Enter.

■ C>**DATE** [Enter]

DOS commands are shown in uppercase letters throughout this text. However, you can key them in upper- or lowercase on the computer.

THE DISK OPERATING SYSTEM

DOS is an acronym for Disk Operating System, a program designed to manage the flow of information, programs, and resources of IBM and IBM-compatible computers.

This system was developed by Microsoft Corporation as MS-DOS. It was later sold to IBM to market under their name of PC-DOS. There is very little difference between the two. Your capabilities in MS-DOS will translate to PC-DOS. There have been many previous versions of DOS with each version offering more capabilities than the previous ones. Although this book focuses on DOS versions 5.0, 6.0, 6.1, and 6.2, the main commands and operations are compatible with earlier versions. For example, a command, DIR (for directory to show a list of the files on the disk), will work the same with identical results beginning with DOS version 1.0 and going through 6.2. Some commands in DOS 5.0, 6.0, 6.1, and 6.2 are not available in earlier versions; however, it is important to know about the newer functions even if your DOS version does not include them.

DOS is the software tool that enables you to control the personal computer hardware, to manipulate data in the computer's memory and in disk files, and to utilize other software programs. The disk operating system software manages the flow of information within the microcomputer. With instructions from you, DOS directs disk drive operations, the use of memory, and the central processing unit (CPU), as well as the flow of information to and from devices such as the keyboard, monitor, and printer.

> **Note:** Technology with DOS-based computers continues to evolve. There are both MS-DOS and IBM PC-DOS, both of which have several recent versions. Most MS-DOS and IBM PC-DOS commands and operations are interchangeable. While this textbook focuses on MS-DOS versions 5.0, 6.0, and 6.2 and IBM PC-DOS version 6.1, most commands and operations are compatible with earlier versions.

THE COMPUTER SYSTEM

To begin working with DOS, you will need to know the basics about the components of the microcomputer. A microcomputer is a device that can process and manipulate numbers very quickly. By representing characters and graphic elements as numbers, a microcomputer can also process text and

graphics. As a user, you do not need to understand the technical details of how a computer operates; however, understanding some of the basics will make learning to use the computer easier. The components of the microcomputer are the system unit, the keyboard or mouse, the monitor, the disk drives, and the printer. As represented in Figure 1-1:

1. The **system unit** houses the computer's main electronic elements. These elements include a **system board** or **motherboard** on which are found the **microprocessor** (also called CPU or Central Processing Unit), **memory**, and **math co-processor chip**. Also, **adapter cards** which connect the other devices such as the monitor and printer are in the system unit. These adapter cards are inserted into the **expansion slots** located on the system board.

   The system unit also houses the power supply that powers the computer as well as a small speaker unit which is responsible for the beep you may hear when you make an error.

   On the back of the system unit are **ports** and **connectors** into which printers, monitors, the keyboard, and other peripheral devices may be plugged. It will be important to learn to recognize the connection points in order to connect and disconnect computer components.

2. Input devices such as the keyboard, mouse, and pen allow you to communicate with the computer. The **keyboard** is the most commonly used device. You might also use a **mouse** as an input device. Pens are now available with 386 or higher computers.

3. The **monitor** or screen allows the computer to communicate with you by displaying the program screens, keyboard or mouse input, and the results of your computer operations.

4. The **disk drives** located in the system unit house the hard or floppy diskettes that permanently store programs and data files on fixed or removable disks. The drives are connected to the power supply and, by ribbon cables, to the **disk drive controller card(s)**. These controller cards fit into expansion slots on the motherboard and manage the transfer of data to and from the disk drives.

5. A **printer** allows you to make paper copies of your work.

These five components of any basic microcomputer system are found in a variety of configurations. Other optional components or peripherals, such as scanners, tape drives, and modems for telephone communications, are often added to the basic system. It is important for you to examine the special features of the system that you are using.

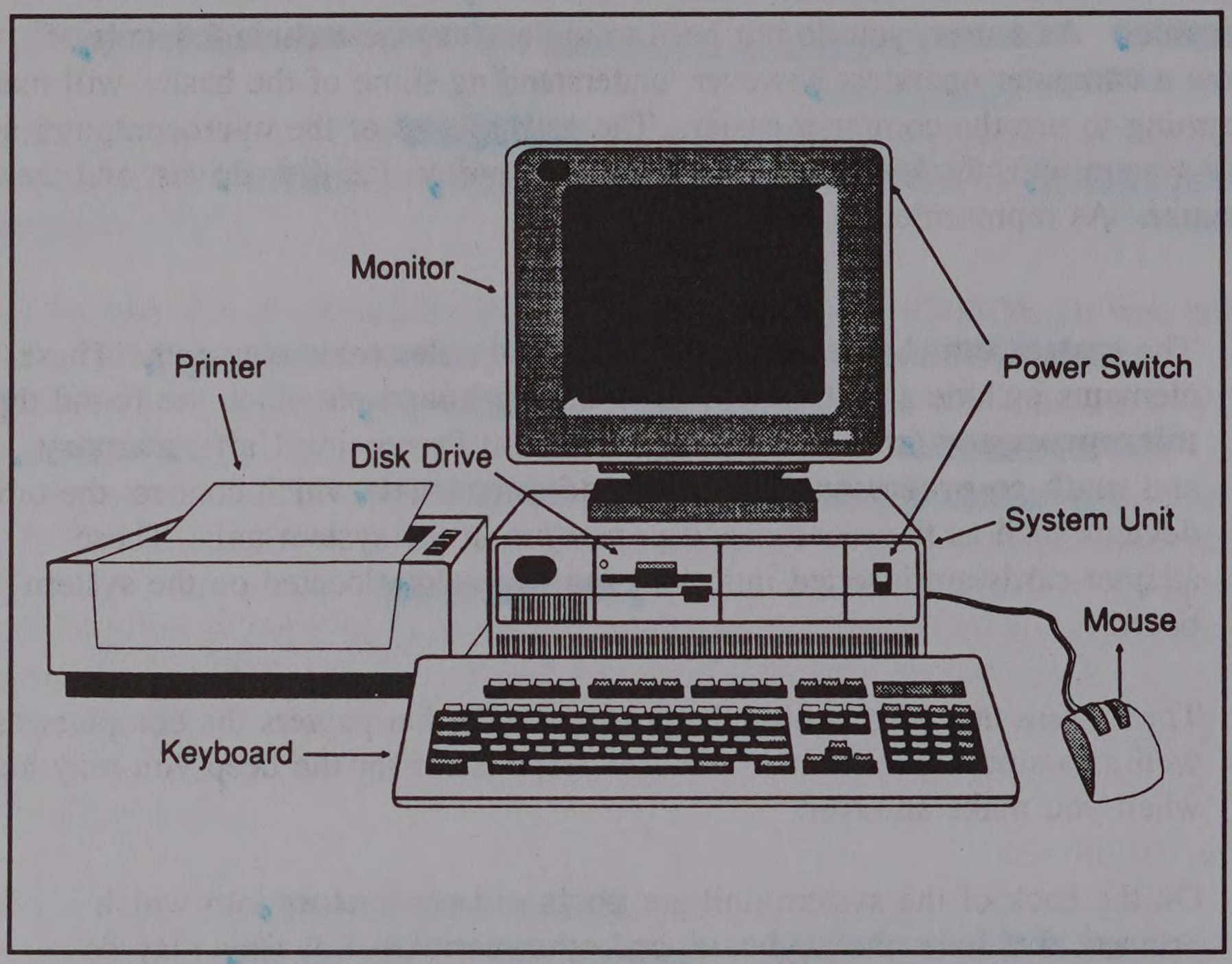

Figure 1-1. Components of a microcomputer system.

The System Unit

In each system unit, there is a **system board** or **motherboard**, which houses among other things the **microprocessor** chip, often thought of as the "brain" of the computer. It is the device that performs the calculations. It takes information in, manipulates it, and sends it out in a modified form. This in-and-out flow is called input and output. Intel Corporation manufactures the microprocessor chips. The first microprocessor chip used in the IBM personal computer was called the 8088 chip. It was replaced in the IBM PC-AT models by the 80286 chip. Today's computers use the 80386 and 80486. In a very short time the newest chip, the Pentium chip, will be commonly used.

On the system board or on special adapter cards are placed memory chips for processing and storing information. Part of the computer's memory is set aside as a keyboard "buffer" to remember the keys you have pressed. When you press a key, a numeric code defining that key is placed in memory in the keyboard buffer.

Memory Chips On the system board there are two kinds of memory chips. The system contains memory chips that have information permanently recorded on them. This information controls the computer's internal operations and its basic input/output system. This memory is called **read only memory**, or

ROM. ROM is used to store instructions that do not change and that the system requires for operation. You do not have access to read only memory.

The second kind of memory chips are called **random access memory** or **RAM.** You have access to RAM and will change its contents as you load application software programs and key data. Random access memory can be thought of as an electronic work space where you can place instructions and data for manipulations by the computer. For example, in this work space you can temporarily store the instructions of a word processing program and the document files you create with the program.

Pressing the Return or [Enter] key completes the input and sends the numeric character codes in the keyboard buffer to the microprocessor for action. This is **input** to the microprocessor. The microprocessor may then produce **output** that will be displayed on the screen and be stored in memory until further instructions tell the microprocessor what to do next.

The Keyboard

The keyboard is the primary means for communicating with the computer. Keyboards are manufactured by many companies, and although there are standard features, there are also differences. Most keyboards used in the United States have the alphanumeric keys (letters and numbers) arranged in a typewriter layout. To this basic keypad are added several special keys that you must learn to use to communicate effectively with your computer. See Figure 1-2.

Two of the special keys are the Control key and the Alternate key. The Control key is usually labeled CTRL or Ctrl. The Alternate key is labeled ALT or Alt. These two keys give different meanings to keys that are pressed simultaneously with either or both of them. In this way, the Ctrl and Alt keys give additional functions to the other keys.

When you press the Ctrl key, you are telling the computer that the next key or keys pressed will be sending control codes that should not be interpreted as characters. The Ctrl key functions this way in application software and in DOS. Some keyboards have several keys labeled with two names indicating that they have two functions, one activated when pressed alone and the other function activated when used with the Ctrl key.

The keyboard also has a numeric keypad, and it may have a separate group of cursor control keys and other data entry keys. The Number Lock (Num Lock) key switches the numeric keypad from numeric keys to "arrow" or cursor control keys. In the numeric mode, these keys are used for entering numbers. In the cursor control mode, the keys marked with arrows are used to move the

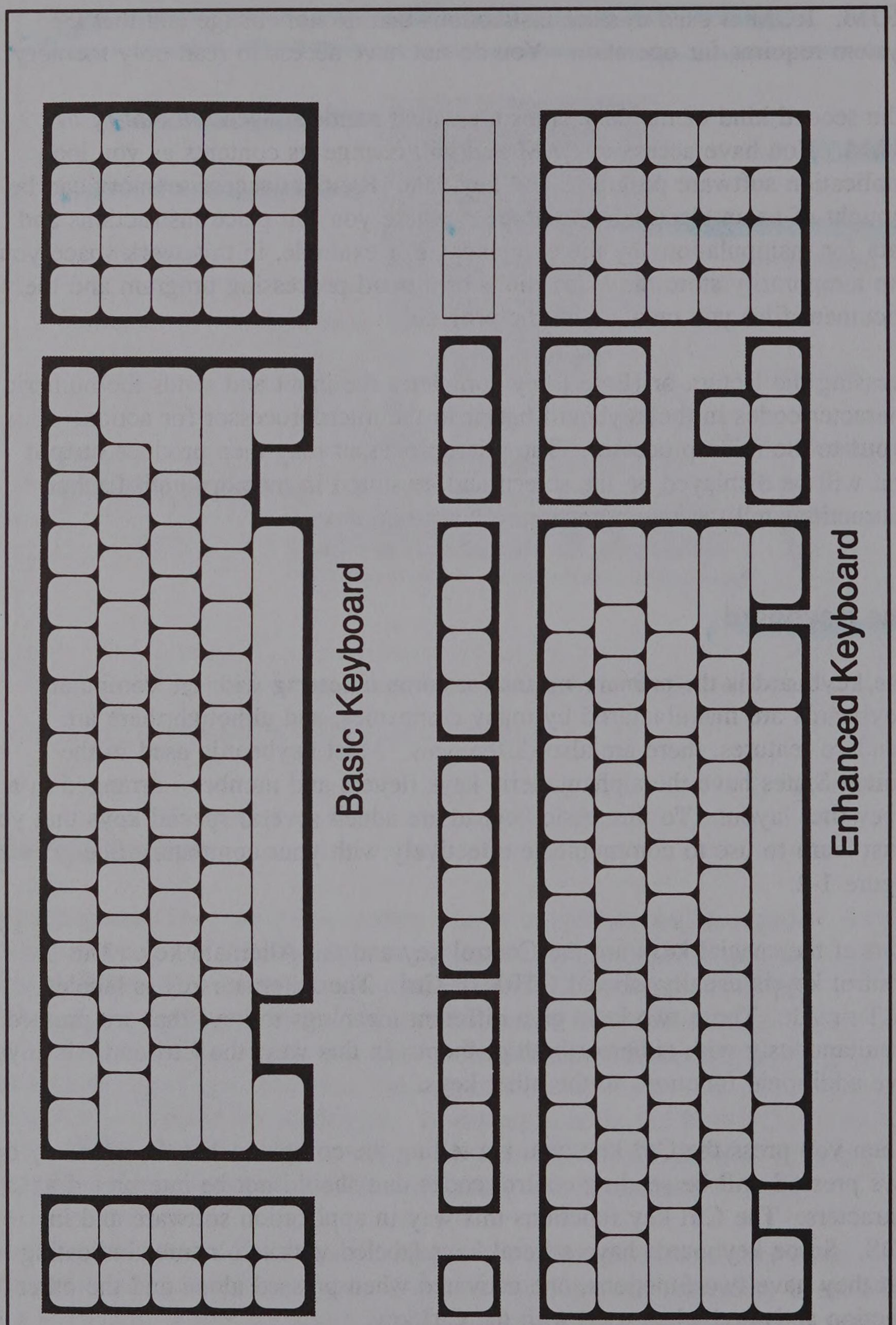

Figure 1-2. Keyboard layouts. Label the diagram that is most like your keyboard. Modify the drawing as necessary.

cursor, the blinking underscore or block on the screen that tells you where the next character you key will appear.

The keys on the computer keyboard are "typematic." That is, they will repeat when you hold them down. A quick tap is all that is necessary. The keys marked Home, Page Up (PgUp), Page Down (PgDn), and End are used by application software programs.

Two other special keys on the computer keyboard that are used in DOS are the Insert (Ins) and Delete (Del) keys. As you might suspect, these keys allow you to insert characters in an existing line or to delete characters from a line.

Perhaps the most unique keys on the computer keyboard are the function keys (F1, F2, etc.). This group of ten to twelve keys may be arranged in two vertical rows on the left side of the keyboard or in a single row across the top of the keyboard. The function keys work differently depending on the software being used. You should always check the software manuals to learn how the function keys are used by a particular program.

Take the time now to examine your keyboard setup and to identify the location of function keys and other important keys such as Ctrl and Alt.

The Monitor

The monitor is the primary device that the computer uses to communicate with you. It is also called the display, cathode ray tube (CRT) screen, or terminal. You will hear monitors described as CGA, MCGA, VGA, and Super VGA. This list represents an increasingly high quality of monitors based on the number of picture elements (pixels). Monitor technology continually brings bigger and better monitors to the marketplace. The sharpness and resolution of both monochrome and color monitors is constantly improving.

Disk Drives

Disk drives provide a means for storing and accessing information on magnetic disks. Understanding some technical details of how disk drives operate is essential for effectively using a microcomputer.

Disk drives are described first by the type of disks they house: floppy or hard. Floppy disks are made of Mylar coated with a metallic oxide. The diskette is flexible and rotates inside a protective jacket. The jacket for 5 1/4-inch disks is a strong, lightweight plastic. The 3 1/2-inch disk is enclosed in a thicker, more rigid plastic jacket. Hard drives contain rigid metal disks. Examples of common drive configurations are shown in Figure 1-3. On most computers, the upper or left floppy disk drive is called drive A; the lower or right disk

drive is called drive B. When floppy diskette drives are in use, depending on the type of drive, the drive door must be closed, the latch must be engaged, or the diskette must be inserted so that it is lodged.

If your system has a hard disk drive, it is called drive C. Most systems come equipped with hard disks which commonly range in size from 10 megabytes (MB) up to several gigabytes (GB). The front plate of the hard drive does not have an opening like a floppy drive, as the disk is permanently sealed in its compartment behind the plate. If a second hard disk drive exists, it is usually called drive D. A very large hard drive may be divided into several "logical" drives named drive C, D, E, F, and so on.

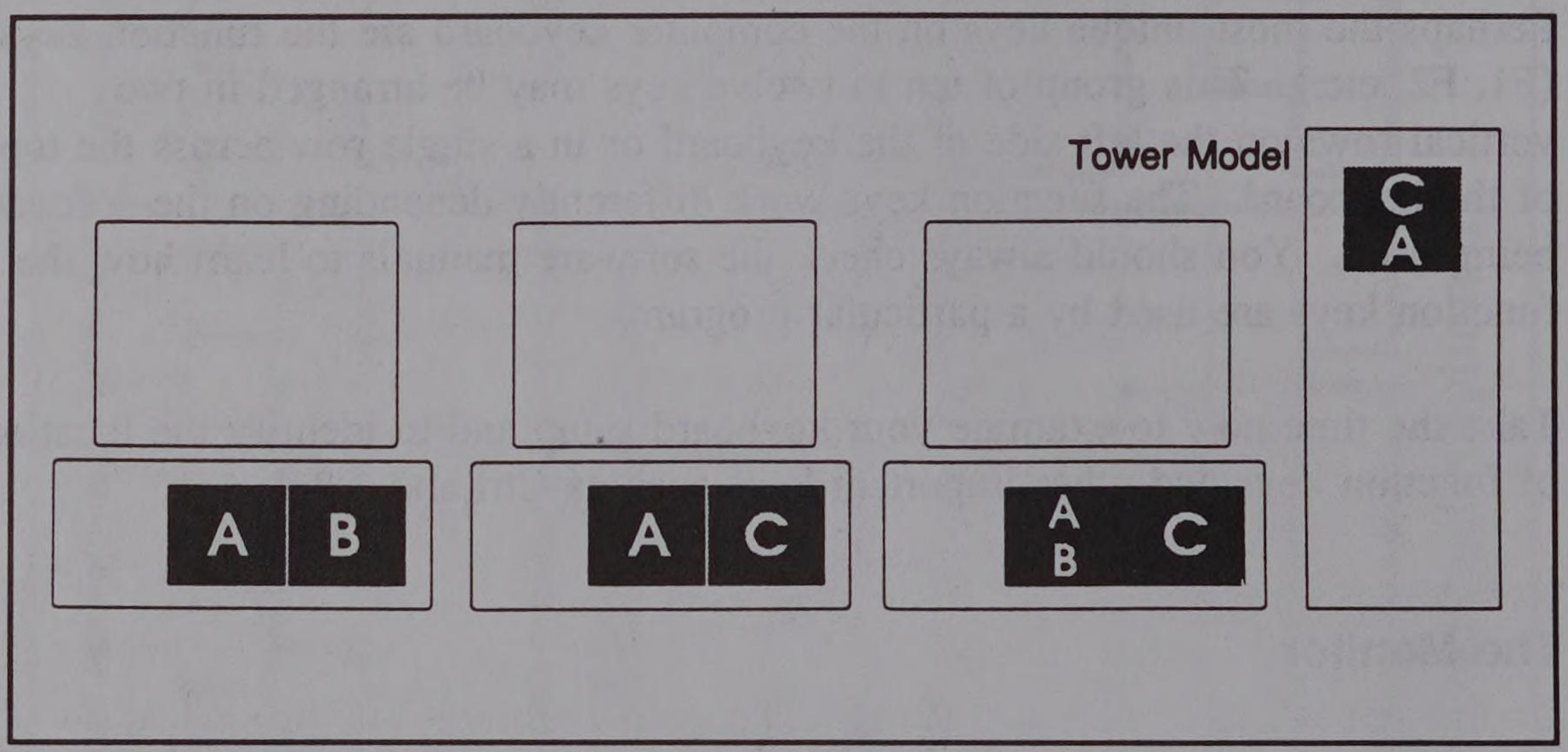

Figure 1-3. Common drive configurations. A and B are floppy drives; C is a hard drive. Tower models are available in many configurations.

The primary operational differences between floppy and hard disks are their storage capacity and speed. The 5 1/4-inch floppy disks in common use store 360 kilobytes (360K), or 1.2 megabytes (1.2MB), of data. The 3 1/2-inch floppy disks store 720K, or 1.44MB, of data. The 1.2MB and 1.44MB disks are called high-density (HD) disks and must be used in high-density drives designed to read and write information on them. The low-density 360K and 720K disks can also be used in high-density drives. Hard disks typically store 100MB or more, making it possible to store on hard disks large application programs and data that would not fit onto a floppy diskette.

An advantage of the hard disk is speed. Hard disk drives spin continuously when the computer is turned on. With internally installed fixed disk drives the disk is turning whenever the computer is on. Floppy drives rotate the diskette only when information is being written to it or read from it. Each time the computer accesses a floppy disk to read or write data, the drive's rotating mechanism must start and come up to operating speed. The hard drive is normally at operating speed.

Networked computers use other drives as well; in fact, drives named with all 26 letters of the alphabet may be used. The network drive names usually start with the letters immediately following the DOS drive names. For example, if DOS uses drives A–E, the network drives will begin with F. Your instructor will tell you about any special drive configurations in the system you are using.

Permanent Data Storage

Many people confuse computer memory and disk storage. It is important to differentiate between the two. Because you will use a computer for work that requires permanent records, the data stored temporarily in the RAM chips must be transferred to some form of permanent storage before the computer is turned off. Disks and tapes coated with a magnetic surface are used for permanent storage of the computer-generated programs and data.

Disks and tapes provide reliable, permanent storage for data, but not in a form that you can read. Printers are output devices that produce readable copies of computer program output. The microprocessor sends information to the printer in a form that can be translated into characters on paper.

Disk Maintenance

Care of your disks is very important. The most important practice to follow with both hard disks and floppy diskettes is to back up or copy data regularly in the event of a mishap. Then when the "impossible" happens, you will be able to restore your files. Examples of backup medium are other floppy diskettes, a tape drive, and another hard drive.

Hard drives require little maintenance. Some systems require manually parking the hard disk. Some disk drives are self-parking. Parking the hard disk ensures that the read/write heads do not cause data damage. Software is available for this task. Parking the hard drive heads is especially important before moving or shipping the computer. Your instructor will tell you whether or not parking is needed for the school computers.

One last type of hard disk maintenance may be necessary when the performance is slow when accessing files or displaying lists of files. This slow performance occurs when files become scattered over the disk surface as they are written, revised, and rewritten in new locations, creating "fragmented" files. You will learn later in this book some ways to remedy this situation with DOS; there are also other utilities which can work with file fragmentation.

There are some important maintenance tips to follow with floppy diskettes also. Observe the following precautions when handling your diskettes:

- DO NOT touch the exposed magnetic surfaces.
- DO NOT expose disks to excessive heat.
- DO NOT spill liquids or food on them.
- DO NOT expose disks to cigarette smoke, ashes, or dust.
- DO NOT bend disks or stack books or other items on them.
- DO NOT expose disks to magnetism.

Short-Term Data Storage: Memory

Computer memory is measured in **bytes.** A byte is eight **bits** of information. Each bit is actually an electronic switch that can be off or on. These two states are presented by the digits 0 and 1. Computers manipulate data using only these two digits in a "binary" numbering system. Using only 0's and 1's, eight bits can be arranged in 256 different orders to make up 256 different numbers, 0 to 255, representing each alpha character (upper- and lowercase), number, special character, and control code, such as a backspace or tab.

A standardized number coding system called **ASCII** (American Standard Code for Information Interchange) is used in most personal computers to provide a common computer language. The important thing to remember is that one byte represents one ASCII character; therefore, the computer's available memory and disk space measured in bytes can be an indication of how many characters are stored.

With the byte as the basic unit of measurement, computer memory and disk storage are expressed in **kilobytes** (K or KB), **megabytes** (MB), and also **gigabytes** (GB). In the binary numbering system, a kilobyte is 1,024 bytes, or 2^{10}. When we speak of a personal computer as having 640K of memory, we mean that its memory capacity is approximately 640,000 bytes. The actual memory size of a 640K computer is 640 x 1,024, or 655,360 bytes. A megabyte is 1,000 x 1,024, or over 1 million bytes. A gigabyte is 1,000,000 x 1,024, or over 1 billion bytes.

Memory capacity is the size of the computer's electronic work space. The memory size of most personal computers without memory expansion chips on the system board or special adapter boards is 640K or 1MB. However, many personal computers sold today are equipped with additional memory, perhaps 2, 4, 8, or 16MB. With special adapter boards to serve as memory expansion boards, personal computers can contain up to 64 or more megabytes of usable memory. Early versions of DOS are able to manage 640K of memory. In order to utilize additional memory, DOS requires the assistance of special

memory-managing utilities such as those provided with DOS 4.0 and later; memory expansion boards; and software such as Microsoft Windows 3.1. The 80386 and 80486 microprocessors could manage 4GB of memory if the computer could hold the chips.

You will hear memory described as conventional, traditional, low, high, upper, expanded, and extended. As computer software continues to grow in sophistication, the need for additional memory increases. You need to understand memory and storage limitations when investigating software packages.

Printers

The printer produces a "hard copy" of your work on paper. Computer printers are varied in design, function, and cost; however, they can be divided into two basic types: impact and non-impact. Impact printers produce an image by striking the paper through a ribbon. They include daisy wheel and dot matrix printers. Non-impact printers include inkjet printers, plotters, and laser printers.

Printers can also be divided into two groups depending on the way they communicate with the computer. Some printers are connected to a parallel port and use parallel data transmission. Others are connected to a serial port and use serial data transmission.

Examine the printer you will be using. Is it impact or non-impact? Is it serial or parallel? Does the printer serve one or more computers? Look for the following switches or buttons:

On-line: Press this to put the printer on-line for printing or to take it off-line.
Line-feed: Press this to advance the paper one line. (It may be necessary to take the printer off-line to operate this feature.)
Form-feed: Press this to advance the paper one page or to the next form. (Again, it may be necessary to take the printer off-line to operate this feature.)

SOFTWARE

Software is loadable into a computer because of DOS. DOS takes care of the disk and file management tasks. Learning about software is as important as learning about hardware. Software like DOS or a word processing application program can be thought of as instructions used to control the computer's work.

WORKING WITH DOS

Now that you are familiar with the system components, let's jump right into working with the disk operating system (DOS). The startup procedure for MS/PC-DOS computers involves a sequence of steps. The process of turning on the computer or resetting it is called "booting" or "powering up." There are instructions stored in a ROM chip for doing this. When the power is turned on, the system follows these instructions to perform a check of the computer memory chips and other hardware.

Loading DOS

The first step in using the computer is to load or **boot** DOS. We will presume that you have used the normal procedure to install DOS. Therefore, your DOS files are probably in a subdirectory named DOS on the hard disk, drive C. If you have any other configuration, change the commands to adjust for the difference.

You will use two different methods for loading DOS or booting the system: one when the computer is off, and the other when it is on. If the computer is off, you will be starting it from a "cold" condition; therefore, this startup procedure is called a **cold boot**. If the computer is already on, the procedure used for booting or powering up is called a **warm boot**. Try both of these procedures now. First, read all the instructions for the *hardware configuration you will be using*. Then perform each step as you read through the instructions a second time.

Practice: Booting the System

Cold Boot—from Hard Disk

- Make sure that drive A is empty or that the drive door is open. Turn on:

  a. *The printer*, if one is available.

  b. *The monitor screen.* Some monitors have an ON/OFF switch; some are automatically turned on when the computer is powered up.

  c. *The system unit.* The power switch is usually on the right side near the back, at the far right on the front panel, or near the top of the tower front panel.

Soon you will see a blinking dash (the cursor) near the top left corner of the screen. Wait while the system performs a check of memory and other internal components and loads the necessary files.

Warm Boot—from Hard or Floppy Disk

Now that you have loaded DOS starting with the computer off or cold, you can practice booting DOS with the computer already on. This is called a warm boot or "resetting" the computer. Warm booting is faster and less demanding on the hardware, so you should use this procedure whenever possible.

To boot from the hard disk system, the door of drive A must be open or the drive empty.

- Three keys are necessary to perform the warm boot. Hold down the Ctrl and Alt keys, then lightly tap the Del key.

> **Note:** Some computers have a **Reset button** on the front of the system unit that can be used like Ctrl-Alt-Del to warm boot the computer when it is on. The Ctrl-Alt-Del method is available on all computers.

The DOS Prompt

After you have booted the computer, there are a variety of possibilities which you may see on the screen. For example, you might see a menu of choices which someone has set up for you. One of these choices might be DOS. If there is a menu from which you can choose DOS, do so now according to the menu instructions.

Or you might see at the bottom of the menu the DOS prompt, which may look like one of the following:

```
C>          C:\>          or          C:\DOS>
```

There may be no menu, but you will see one of the above prompts. Or you may see a message that says something like the following:

```
Current date is Fri 07-02-1994
Enter new date: (mm-dd-yy)
```

This is asking you to accept or change the date. You would then press [Enter]. The system will then display something like the following:

```
Current time is 8:06:59.01a
Enter new time:
```

This is asking you to accept or change the time. You would then press [Enter]. You may then see displayed one of the following:

```
C>          C:\>      or      C:\DOS>
```

The main idea is to get to what is called the DOS prompt, which may look like one of the drive C prompts above. It may be that you are working from a floppy diskette, and you would instead see:

```
A>                    or      A:\>
```

If your computer has a different display than those described above, ask for directions from your instructor on how to access the DOS prompt so that you can begin work!

DOS Commands

The instructions you give DOS are **commands** that the software can interpret and carry out. There are two types of DOS commands called **internal** and **external** commands. The internal commands are instructions contained in one large command file named COMMAND.COM. These instructions are **loaded** or read into the computer's RAM when you boot, or power up the computer. The internal instructions or commands stay in the computer's memory and are available for use until you turn off the power to the computer. If you are using DOS files on a floppy disk, you can remove the DOS disk and still work with the internal commands.

External commands, on the other hand, are smaller DOS files stored in the DOS directory. These commands are loaded each time they are needed for a specific task.

Displaying or Changing the Date and Time

Both the DATE and TIME commands are internal commands. When the computer is booted and DOS is loaded, the screen displays the date from memory. If the current date does not display, you will probably see:

```
Current date is Tues 01-01-80
Enter new date (mm-dd-yy):
```

Each time you boot a computer that does not have an internal battery-powered clock, it displays an incorrect date, January 1, 1980. If the system has an internal clock, you may not need to reset the date and time.

Keying the correct date and time to set the system clock is a good practice, as DOS records the date and time when you create or revise your data files. If the correct information is recorded, it helps you determine when the files were created and which version is the most recent.

To enter a new date on the line that reads "Enter new date (mm-dd-yy)," key:

A number from 1 to 12 representing the month, followed by a hyphen (-), a slash (/), or a period (.).

A number from 1 to 31 representing the day of the month, (-), (/), or (.).

The two digits representing the year. Press [Enter].

For example, 9-17-94 represents September 17, 1994.

When you have correctly entered the date, the current time in the computer's memory is displayed on your screen. The time display will be similar to the following:

```
Current time is 0:02:54.97
Enter new time:
```

The time is shown in hours, minutes, seconds, and hundredths of a second. The time is given using a 24-hour clock, commonly used by the military. With this clock, 0 is midnight, 12 is noon, 18 is 6 p.m., and so on. In DOS version 4.0 and later, you can indicate a 12-hour clock by keying the letter a or p for a.m. or p.m. after the time.

To set the time, key the following:

A number from 0 to 23 for the hour, followed by a hyphen (-), a slash (/), or a period (.).

A number from 0 to 59 for the minutes. It is unnecessary to key the seconds and hundredths of a second. Use the a or p if you wish to use a 12-hour clock. Press [Enter].

If the date and time displayed are correct, or if you wish to bypass entering the correct date and time, just press [Enter] at the date and time prompts.

If you do not receive prompts requesting you to enter the date and time, you may want to initiate them to be sure your files are correctly date and time stamped.

Practice: Setting the Date and Time

Again, your DOS prompt may look like any of the following:

C> C:\> *or* C:\DOS>

To set the date, key the following at the DOS prompt:

- **C:\>DATE 9-17-94** (or any date you wish) [Enter]

To set the time, key the following:

- **C:\>TIME 8:00a** (or any time you wish) [Enter]

To confirm the date and time, key the following:

- **C:\>DATE** [Enter]

The date you entered is displayed. Press [Enter] to accept the date and return to the DOS prompt.

- **C:\>TIME** [Enter]

The time you entered is displayed. Press [Enter] to accept the time and return to the DOS prompt.

- Reset the date and time to the current date and time.

Error Messages

If you make a mistake while performing tasks on the system, an error message will be displayed on the screen. DOS uses helpful messages to inform you about errors. You will learn what they mean and will use them as instant clues to what has gone wrong.

One common type of message is a device error message, such as "Not ready error reading drive A." When a device error occurs and this message is displayed, the system requires you to select an option from a list similar to the following:

```
Not ready reading drive A
Abort, Retry, Ignore?        or   Fail
```

You must key the initial letter for your choice: A to abort, R to retry, I to ignore, or F to fail. Abort means to abandon or end the attempted command. If you no longer wish to use that command, key **A** for abort. Keying **R** for

retry instructs the computer to try again to execute the command. Keying **I** for ignore tells the computer to ignore the error and, if possible, continue execution of the command. If you are using DOS version 3.3 or later, you will see the option **F** for Fail in some of these error messages. Fail is equivalent to Ignore.

The usual reason for the "Not ready" error message is that you have forgotten to shut the door on the drive, do not have the disk fully inserted into the disk drive, or do not have a disk in the drive. If this is the case, correct the situation and key **R** to retry the command.

Another common error message generated from DOS is "Bad command or file name." This message tells you that you have entered a command or file name that DOS cannot recognize or find. When you see this message, carefully check the keyed command and reenter the command correctly.

DOS's cryptic error messages can be very helpful clues in working with the operating system.

Changing the DOS Prompt with PG

The A> or C> at which you keyed the DATE and TIME commands is called the **DOS prompt**. This all-important signal can tell you several things. The letter tells you what disk drive is the **current** or **default** drive. The default drive is the drive DOS will use unless you give other instructions. A much more efficient prompt is a prompt that not only tells you the disk drive that is current but also tells you the current **subdirectory**. A subdirectory is a subdivision of the hard disk or floppy diskette; you will learn in a later chapter how to create and work with subdirectories.

The number of files a hard disk or floppy disk will store can be increased by creating subdirectories to hold file information. Subdirectories can also make your disk and file management more effective by listing files in categories you create. This is especially helpful on hard disks, which often contain thousands of files. For example, you could place all DOS files in one subdirectory. When you install MS/PC-DOS 4.0 or a later version, a special subdirectory is created on drive C for the DOS files. When your computer is ready for your input, you want the DOS prompt to show the current drive as drive C and also indicate the current subdirectory.

The following prompts are more informative prompts showing not only the current drive but also the current subdirectory.

```
C:\>          or          C:\DOS>
```

The C represents the current drive; the \ represents the root or main directory. The word DOS following the backslash (\) in the second example is the name of the subdirectory in which the DOS files are stored.

To change the prompt from C> or A> to also include the current subdirectory, key:

- **PROMPT PG** [Enter]

Your prompt should now look like this:

```
C:\>    or    C:\DOS>
```

When you are instructed to key something at the DOS prompt, these are the prompts to which the instructions refer. The blinking line or **cursor** to the right of the prompt indicates the point where anything you key will be displayed. In DOS you will usually key commands here. The line is called a "command" line.

Practice: Setting the Prompt

Regardless of what your prompt is, key the following:

- **PROMPT** [Enter]

This should make your prompt look like C>.

Change the prompt to display the current directory by keying the following:

- **PROMPT PG** [Enter]

This should display your prompt as one of the following: C:\> or C:\DOS.

Nice work on Chapter 1! Test your knowledge now with the Self-Check Quiz and Assignment.

CHAPTER 1 SELF-CHECK QUIZ

True/False (Write T if the statement is true, F if the statement is false.)

_____ 1. ROM (read only memory) is temporary memory that the user can change.

_____ 2. A hard disk can store more information than a floppy disk, but floppy drives access data faster than hard drives.

_____ 3. The computer's internal clock tracks the date and time.

_____ 4. To warm boot the computer, turn it off and back on.

_____ 5. To show the current directory in the DOS prompt, key PROMPT PG.

Multiple Choice (Circle the correct answer.)

6. The microprocessor is a component of

   a. the disk operating system.
   b. the monitor.
   c. the system unit.
   d. optional adapter cards.

7. To retain a permanent record of the computer's output, it can be saved in

   a. random access memory.
   b. RAM disk files.
   c. read only memory.
   d. floppy or hard disk files.

8. The Ctrl and Alt keys are important in operating the computer because

   a. they are used in combination with other keys to give them special functions.
   b. they are toggle keys that turn the numeric keypad on and off.
   c. they control scrolling of the display on the monitor screen.
   d. they are used to perform the same special functions with all software.

9. Which of the following is NOT a hazard to floppy diskettes?

   a. magnetism
   b. light
   c. heat
   d. dust

10. A device that is commonly used for input and output is the

   a. monitor.
   b. disk drive.
   c. printer.
   d. keyboard.

ASSIGNMENT

To practice using the commands in this chapter, complete the following steps. Check off each step when you understand its operation, or mark the step with a question mark if it is not clear. Write the commands or keystrokes in the space provided.

1. Cold-boot the computer.

_____

2. If necessary, change the prompt to reflect the current directory.

_____

3. Reset the date to your next birthday.

_____

4. Reset the date back to today's date.

_____

5. Reset the time to one hour earlier.

_____

6. Reset the time back to the current time.

_____

7. Perform a warm boot.

_____

8. Reset the date if it is not correct.

_____

9. Reset the time if it is not correct.

_____

10. If necessary, change the prompt to show the current directory.

_____

CHAPTER 2

BASIC DOS COMMANDS

OBJECTIVES

When you have completed the activities and assignment in this chapter, you will be able to:

1. Clear the screen.

2. Print the screen.

3. Display a directory of a disk in several formats.

4. List the conventions for naming files.

5. Work with wildcards.

6. Use the DOS command VER (Version).

7. Use the HELP command and switch.

THINGS TO REMEMBER WHEN USING THIS TEXTBOOK

There are several important considerations that are unique to DOS and important to remember while using this textbook.

1. When using a floppy diskette, never open the drive door when the light on the front of the disk drive is on, indicating that the drive is in use.

2. DOS commands or instructions and file names may be keyed in upper- or lowercase. As a general rule, DOS is not case-sensitive. We will point out the exceptions when they occur.

3. A DOS operation can be stopped by holding down the Ctrl key and tapping the C key or the Break key.

4. DOS treats letters and numbers differently: the lowercase letter "l" cannot be substituted for the number "1" key. The uppercase letter "O" cannot be substituted for the number "0."

5. You will see a bullet (■) preceding any required input. You must press the Enter key when you have finished keying a command to signal DOS to execute the command. In this textbook, [Enter] denotes pressing the key marked Enter. (Do not key the [ ] or the word Enter.) Where commands to be keyed are shown, the letters in boldface indicate what should be keyed. In the following example, you would key the letters DATE and then press the key marked "Enter."

   ■ C>**DATE** [Enter]

   DOS commands are shown in uppercase letters throughout this text. However, you can key them in upper- or lowercase.

6. If your prompt reads C> instead of C:\>, key:

   C>PROMPT PG

   If your prompt reads C:\>, and you would like it to read C>, key:

   C:\>PROMPT

7. If your prompt reads C:\DOS> instead of C:\>, key the following:

   C:\DOS>CD \

   This command is one you will learn in Chapter 6. The CD means Change Directory; the backslash represents the root directory. Therefore, using this command, you are requesting to change directories to the root directory.

8. It is important to know how to correct errors or delete characters when keying DOS commands. There are several ways to correct errors:

   a. Use the Backspace key and backspace once for each character to be deleted.

   b. When you have keyed a command at the DOS prompt and wish to cancel the entire line, press the Esc and [Enter] keys and the DOS prompt will again be displayed.

Practice: Canceling a Line

Again, your DOS prompt may look like any of the following:

C> C:\> *or* C:\DOS>

To cancel a line, key the following at the DOS prompt:

- **C:\>DOS is easy to learn.**

To cancel this line, you could backspace each letter off the screen, but a faster way is to press the Esc key. Do that now. The line will then display as:

```
C:\>DOS is easy to learn.\
```

The line is still there and is followed by a backslash (\) symbol. The backslash symbol is a result of your having pressed Esc. Press [Enter] to display the DOS prompt.

DOS COMMANDS—INTERNAL vs EXTERNAL

As you read in Chapter 1, the commands in DOS are either internal or external. It is important to know the difference between internal and external commands and to remember which commands are in each group. The internal commands are instructions contained in one large command file named COMMAND.COM. These instructions are **loaded** or read into the computer's RAM when you boot, or power up the computer. The internal instructions or commands stay in the computer's memory and are available for use until you turn off the power supply to the computer. If you are using DOS files on a floppy disk, you can remove the DOS disk and still work with the internal commands. Therefore, once you have loaded DOS, you can use any of the internal commands, such as **DIR** (directory of disk), **VER** (version of DOS), and **VOL** (volume label or name of disk).

External commands are smaller DOS files stored in the DOS directory. These commands are loaded each time they are needed for a specific task.

Study the following list to learn which commands are internal and which are external:

Internal		External	
CHDIR or CD	PROMPT	CHKDSK	HELP
CLS	REN or RENAME	COMP	LABEL
COPY	RMDIR or RD	DEFRAG *, **	MIRROR
DATE	TIME	DISKCOMP	PRINT
DEL	TYPE	DISKCOPY	SCANDISK *
DIR	VER	DOSKEY	SYS
ERASE	VOL	E ***	TREE
MKDIR or MD		EDIT	UNDELETE
MOVE *, **		EDLIN	UNFORMAT
PATH		FC	XCOPY
		FORMAT	

Commands: * MS-DOS 6.0 ** MS-DOS 6.2 *** IBM PC-DOS 6.1

CLEARING THE SCREEN

You may wish to work with a blank screen from time to time. To do so, key at the C:\> prompt:

CLS

The screen will be cleared and only the DOS prompt will be displayed. You can then key another command.

Practice: Clearing the Screen

Of course, you remember that you can change from C> to C:\>, for example, by changing the prompt with the PROMPT PG command as you learned in Chapter 1. Regardless of what your DOS prompt is, try this new command by keying the following:

- C:\>**CLS** [Enter]

PRINTING USING PRINT SCREEN

It is important that you begin to print some of your work. There are several ways to print using DOS. The first method you will use is to print the contents of the screen display; this is like taking a picture of what is currently on the screen. This operation and the resulting print is called a **print screen**, screen print, or screen dump.

Before you begin to print, make certain there is a printer connected to your computer, that the printer is turned on, and that it is on-line. You may need to ask for help if this is your first experience with printing or if you are unsure about the printer setup.

When you are ready to print, locate the key marked Print Screen, PrtSc, or a similarly marked key. On many systems, to print the contents of the screen, you must hold down one of the Shift keys and tap the PrtSc key. On some of the newer systems you need only to press the PrtSc key. You will see the cursor move across and down the screen as its contents are being printed. If the cursor does not move, but remains at the top left corner of the screen, the information to be printed is not going to the printer, or the printer is unable to print the information. Hold down the Ctrl key and press the Break key, then look for the problem. Many laser printers will not screen-print a partial page, and you will have to research the printer documentation for the method to screen-print. Some computers and printers do not work well together to print the screen. Read the printer documentation to research alternate methods.

Practice: Printing with PrtSc

Experiment with PrtSc in the following exercise. First, clear the screen:

- C:\>**CLS** [Enter]

Next, check the date and time:

- C:\>**DATE** [Enter] [Enter]
- C:\>**TIME** [Enter] [Enter]

Now print the screen by pressing the PrtSc key (or hold down the Shift key and press the PrtSc key) on your keyboard. Your printed copy should match your screen.

FILENAMES AND EXTENSIONS

Programs and data are stored as files on a disk. Each file has a unique **file name** so that the user *and* the system can identify and locate the file. Below are the rules and conventions for naming files.

Filenames and extensions are shown by DOS in uppercase letters. If you key them in lowercase, DOS will convert them to uppercase.

A few words should not be used as file names. These words include the DOS commands and the **device names** listed in the following section–**Program**

Extensions and Device Names. Unpredictable results might occur if you were to use a DOS command or device name as a file name.

1. The first part of the file name is called the **filename** (one word). The filename can be from one to eight characters long. It can include any letter, number, or special character except:

   Blank spaces
   The following invalid characters:
   . " / \ [ ] : | < > + = ; , * ?

   Some examples of valid filenames are:

Irvine	Mary	07_02_94
1	ABC	ABC_1234

2. The second part of the file name is called the **extension.** The extension is optional— file names are not required to include an extension. The filename and extension are separated by a period or "dot." The same characters and symbols can be used in the extension as in the filename; however, the extension is limited in length to three characters, numbers, or symbols. Extensions are often used to provide additional information about a file. For example, the extension TXT is conventionally used to tell you that the file is a text file. The examples of file names with extensions are:

IRVINE.CA	NOV-28.94
JANKE.LTR	1993-94.BFS

PROGRAM EXTENSIONS AND DEVICE NAMES

1. Some software programs automatically add their own default extensions. By convention, certain extensions have special meanings. They can help you recognize the source of files and tell you what the files might contain. Here are a few examples:

BAK	Backup file
BAS	Program written in the BASIC programming language
BAT	Batch file
BMP	Bit-mapped graphics file from Windows Paintbrush
C	Program written in the C programming language
COM	Executable program command file
CO_	Compressed DOS COM file
DBF	dBASE data file
DOC	Text data file created in Microsoft Word
EXE	Executable program file
INI	Initialization file storing information for setting up software

PCX	Graphic format for Paintbrush files
PM5	PageMaker version 5.0 file
SYS	System file or file that enables use of a device such as a mouse
TIF	Graphics file storing a scanned image
TXT	Data file containing text
WK1	Lotus 1-2-3 spreadsheet file
ZIP	Compressed file

2. Device names that should not be used as file names include:

AUX	A communications port
COM1	The first communications port (interchangeable with AUX, PRN)
COM2	The second communications port (also COM3, COM4, etc.)
CON	The console (keyboard and screen)
LPT1	A parallel (line) printer port (interchangeable with PRN)
LPT2	A second parallel printer port (interchangeable with PRN)
LPT3	A third parallel printer port (interchangeable with PRN)
NUL	A dummy (nonexistent) device
PRN	The on-line printer

THE DIRECTORY

One of the most frequently-used internal commands is the DIR (directory) command, which displays a list of files and subdirectories on a disk. The DIR command, used without any parameters, displays the following information:

- The name (Volume label) of the disk.
- The Serial Number of the disk.
- The name of the directory.

and five columns of information for each file as follows:

- filename
- file extension
- size of file in bytes
- date the file or subdirectory was created or last revised
- time the file or subdirectory was created or last revised

At the end of the directory listing, you will find information about the total number of bytes in the files listed and the amount of free unoccupied space on the disk.

Practice: Displaying the Directory

Let's try displaying a directory from the DOS prompt of drive C. If you wish the prompt to display the current directory, key the following as you learned in Chapter 1:

- **C>PROMPT PG** [Enter]

Now display the directory with the following command:

- **C:\>DIR** [Enter]

You should see a directory such as the partial one that follows. Depending on your DOS prompt, your directory contents may be significantly different.

```
Volume in drive C has no label
Volume Serial Number is 1B63-B812
Directory of C:\

COMMAND    COM        54,619    06-19-92   12:00p
CONFIG     SYS           437    06-19-92   12:00p
DASDDRVR   SYS         1,170    06-19-92   12:00p
NUMOFF     COM            11    06-19-92   12:00p
AUTOEXEC   BAT           111    06-19-92   12:00p
WINA20     386         9,349    06-19-92   12:00p
TEXT1      TXT         5,349    06-20-94   01:00p
CHAPTER1   TXT        12,894    07-23-94   11:00p
BOOK4      DOC         4,098    08-19-94   12:00p
BANKLOT    XLS        22,450    06-20-94   02:30p
BKBAL      WK1        33,712    09-21-94   02:03p
BKDEP      XLS        45,067    08-21-94   10:00p
TAX93      TXT         3,128    05-22-94   11:05p
TAXCAL     WK1        12,986    04-19-94   10:00a
TAXUSCO    TXT           500    01-07-94   07:00p
ANAH       TXT         1,763    10-23-94   02:10p
NORD       TXT           879    07-18-94   05:07p
SEAR       DIV            50    01-12-94   10:10a
KIMBERY    BAB         4,590    09-30-94   09:06p
NOVELL     TXT        98,745    04-19-94   06:30a
PM         EXL         2,417    08-09-94   12:09p
HOME       TXT        32,123    10-30-94   07:45p
MARIA      TTE         7,000    02-15-94   12:00p
SCREEN     WK1           920    06-19-92   02:25p
etc.
```

The file names will not be the same but the format will show the name and serial number of the disk, and each file with its filename and extension, size, and date and time of creation or last revision.

Switches Used with Directory Commands

You can modify many DOS commands by adding optional "switches" that signal DOS to perform the command in a specific way. A **switch** is a special parameter that follows the command and always begins with a slash (/). The DIR command has several switches. Two that are frequently used to modify its operation are /P and /W.

Using the Pause Directory Switch (/P) with DIR

If the directory has more than 23 files in it, the entire directory will not fit on the screen at one time. The /P switch can be added to the DIR command to tell DOS to display the directory in screen-length portions. The slash tells the system to modify the command; the P tells DOS to pause the directory display after each 23 lines. When you are ready to see more of the directory, you can press any key and the next 23 lines will be displayed. Therefore, the command to pause the DIR command is as follows:

C:\>DIR/P

Practice: Using the Pause Switch (/P) with DIR

Key the following to display the current directory of drive C with the /P switch.

■ C:\>**DIR /P** [Enter]

The display will pause after the first 23 lines. Depending on your DOS prompt, the directory contents may be significantly different.

```
Volume in drive C has no label
Volume Serial Number is 1B63-B812
Directory of C:\

COMMAND    COM        54,619    06-19-92   12:00p
CONFIG     SYS           437    06-19-92   12:00p
DASDDRVR   SYS         1,170    06-19-92   12:00p
NUMOFF     COM            11    06-19-92   12:00p
AUTOEXEC   BAT           111    06-19-92   12:00p
WINA20     386         9,349    06-19-92   12:00p
TEXT1      TXT         5,349    06-20-94   01:00p
```

```
CHAPTER1   TXT        12,894     07-23-94   11:00p
BOOK4      DOC         4,098     08-19-94   12:00p
BANKLOT    XLS        22,450     06-20-94   02:30p
BKBAL      WK1        33,712     09-21-94   02:03p
BKDEP      XLS        45,067     08-21-94   10:00p
TAX93      TXT         3,128     05-22-94   11:05p
TAXCAL     WK1        12,986     04-19-94   10:00a
TAXUSCO    TXT           500     01-07-94   07:00p
ANAH       TXT         1,763     10-23-94   02:10p
NORD       TXT           879     07-18-94   05:07p
SEAR       DIV            50     01-12-94   10:10a
KIMBERY    BAB         4,590     09-30-94   09:06p
NOVELL     TXT        98,745     04-19-94   06:30a
PM         EXL         2,417     08-09-94   12:09p
HOME       TXT        32,123     10-30-94   07:45p
MARIA      TTE         7,000     02-15-94   12:00p

Press any key to continue . . .
```

To continue the display of information, follow the screen prompt instructing you to press any key.

Using the Wide Directory (/W) Switch with DIR

Notice that with the directory commands you have used so far, the directory listing has included five columns of information about each file: the name of the file, the file extension, the size of the file (in bytes), and the date and time the file was created or last revised. You won't always need all of this information about the contents of the disk. There are occasions when just the file names are enough information. There are also times when you will want to see on one screen the names of all files on the disk. A wide directory obtained with the /W switch will usually display this. Key:

C:\>DIR /W

The same 23 filenames shown above are displayed in columnar form:

```
Volume in drive C has no label
Volume Serial Number is 1B63-B812
Directory of C:\

COMMAND.COM    CONFIG.SYS     DASDDRVR.SYS    NUMOFF.COM    AUTOEXEC.BAT
WINA20.386     TEXT1.TXT      CHAPTER1.TXT    BOOK4.DOC     BANKLOT.XLS
BKBAL.WK1      BKDEP.XLS      TAX93.TXT       TAXCAL.WK1    TAXUSCO.TXT
ANAH.TXT       NORD.TXT       SEAR.DIV        KIMBERY.BAB   NOVELL.TXT
PM.EXL         HOME.TXT       MARIA.TTE       etc.
```

Does your screen display look like the above? The files will not be exactly those listed here, but the format should be the same.

Practice: Using the Wide Switch (/W) with DIR

Key the following:

- **C:\>DIR /W** [Enter]

Obtaining Directory Listings for Selected Files

Occasionally you will want directory information about only one file. You can ask DOS for a directory of just one file by specifying the file name as a parameter after the command DIR. For example, if you wanted information about the file COMMAND.COM in the default directory, you would key:

C:\>DIR COMMAND.COM

Five columns of information including the filename, extension, date and time the file was created or last revised, and the size of the file is displayed.

```
Volume in drive C has no label
Volume Serial Number is 1B63-B812
Directory of C:\

COMMAND      COM  54,619    06-19-92   12:00p
```

Practice: Obtaining Directory Listings for Selected Files

Display information about the following files by keying:

- **C:\>DIR AUTOEXEC.BAT** [Enter]

and

- **C:\>DIR CONFIG.SYS** [Enter]

Five columns of information about each of the files are displayed.

Changing the Directory Order

Additional directory switches were introduced with DOS 5.0 to sort the directory in various orders. These commands display the files as follows:

DIR /ON	Arranges alphabetically in ascending order by name (A–Z).

DIR /OE — Arranges alphabetically in ascending order by extension (A–Z).

DIR /OS — Arranges by size (smallest to largest).

DIR /OD — Arranges by date (earliest to latest).

DIR /OG — Arranges subdirectories before files.

DIR /B — Displays each filename and extension on a line by itself with no other information.

Most of these commands begin with **O** which represents the word **order**. Most of these new switches also may be combined with the /P and /W switches. The combination DIR /ON /P will display a directory sorted by name in screenfuls of 23 lines.

To reverse the order of the above commands, insert a minus sign (–) after the /O. For example:

DIR /O-N — Sorts the directory files in descending order by name beginning with the letter Z.

Practice: Sorting the Directory

Display the directory of drive C:

- C:\>**DIR** [Enter]

Now display the directory of drive C in alphabetic ascending order by filename as follows:

- C:\>**DIR /ON** [Enter]

Do you see a difference in the order of the directory listing? Now sort by date with most current date first:

- C:\>**DIR /O-D** [Enter]

In both cases you should see the files listed in the appropriate order.

WILDCARDS

A wildcard can be a substitute for a filename or an extension. Wildcards are one of the most useful tools in DOS. There are two wildcards: The wildcard

that represents a single character is the question mark (?); the wildcard that represents a group of characters is the asterisk (*). An example of a way to use a ? as a wildcard to display the directory listing for COMMAND.COM is to key:

DIR ?OMMAND.COM

Here the ? represents the C in COMMAND. You could also substitute question marks for other letters in the filename or extension such as the following:

DIR ??MMAND.?OM

Notice that one question mark is substituted for each unknown character in the name. COMMAND.COM fits this description.

The asterisk, or "star" as it is more often called, represents a number of characters. Used in the filename, one asterisk stands for any characters in the filename that come after it. For example, to display a list of all files in the directory that have any filename starting with the letter C and with the extension COM, you would key the following at the DOS prompt:

DIR C*.COM

One asterisk used in the extension represents all characters that come after it. To see a list of all files that have filenames beginning with F and that have any extension, you would key:

DIR F*.*

To see a list of all files with any filename and with the extension COM, the command would be

DIR *.COM

To see a list of all files with any filename that begins with the letter C with any extension, the command would be

DIR C*.*

Practice: Working with Wildcards

Key the following commands using wildcards. Depending on your configuration, you may or may not have files for some of the following.

- C:\>**DIR *.SYS** [Enter]

- **C:\>DIR *.S??** [Enter]
- **C:\>DIR COMM*.?OM** [Enter]
- **C:\>DIR C*.C*** [Enter]
- **C:\>DIR *.COM** [Enter]

Print the screen.

THE VERSION (VER) COMMAND

As you learned in Chapter 1, there have been several versions of DOS. As new capabilities in computer hardware and application software have required new commands and operations, new versions of DOS have been released. The first version of DOS was numbered 1.0 as is the convention for numbering software releases. Minor revisions to a version are numbered with a decimal number (1.1, 1.2, 1.3, etc.). When a major revision is made, the first number is changed. For DOS this occurred when version 2.0 was introduced to handle the hard drive of the IBM XT computer. The designers of the operating system have been careful to keep the commands and operations of the new versions compatible with the earlier ones. Therefore, you can use the older versions when necessary. Sometimes it is important to know which version of DOS you are using. Some software application packages require more recent versions of DOS than others or might require upgrading to run with the newer versions of DOS.

Another important reason for knowing the version of DOS you are using is that DOS will not allow you to mix COMMAND.COM and most external commands of various versions of DOS. The internal command VER (version) provides an easy way to quickly determine the version of DOS used to boot the computer. By keying VER at the DOS prompt and pressing [Enter], you can obtain a report on the version of DOS currently residing in memory. VER is an internal command.

Practice: Using the VER Command

At any DOS prompt, key

- **VER** [Enter]

DOS will display a report similar to the following:

```
MS-DOS Version 6.2 (or whatever version you are using),
or IBM Version 6.1
```

ASKING FOR HELP

One of the important features of the latest versions of DOS is the addition of the HELP external command. You can obtain information on the DOS commands you specify in two ways. One way is to key the command HELP followed by the command you would like explained. For example, if you would like help on the DATE command, key:

C:\>HELP DATE

You can display on-line HELP information by keying:

C:\>HELP

You also receive on-line information using the /? switch, by keying:

C:\>DATE /?

The screen display shows a brief description of the command, the command syntax, and other information pertinent to the command.

To display a list of all the commands listed under the Help function, key the word HELP without any parameter. The DOS commands are listed alphabetically.

Practice: The HELP Command

Request information regarding the DIR command with:

- C:\>**HELP DIR** [Enter]

or

- C:\>**DIR /?** [Enter]

Good work! Now practice your newly found HELP skills as needed for your work on the Self-Check Quiz and Assignment.

CHAPTER 2 SELF-CHECK QUIZ

True/False (Write T if the statement is true, F if the statement is false.)

_____ 1. The wildcard that represents a single character is the question mark (?).

_____ 2. To display the list of HELP topics, key HELP.

_____ 3. To cancel a line, depress the Alt and Del keys.

_____ 4. Internal DOS commands are contained in the COMMAND.COM file.

_____ 5. To determine the version of DOS on the computer you are using, key the following: VERSION

Multiple Choice (Circle the correct answer.)

6. Which of the following switches is used with the DIR command to sort the directory alphabetically by filename?

a. /ON
b. /OS
c. /OG
d. /OA

7. Which one of the following is a valid file name?

a. @WASH-DC.DOC
b. INVOICES.1993
c. SMITH,R.LTR
d. CON

8. Which command would you key to list all files with filenames beginning with COM and with any extension?

a. COM ?.*
b. DIR COM*.*
c. DIR /COM.*
d. DIR COM?.???

9. Which switch would you use to display the directory 23 lines at a time?

a. DIR /23
b. DIR /W
c. DIR /P
d. DIR /?

10. To clear the screen, key which of the following?

a. CLEAR
b. CLR
c. CLS
d. CLRSC

ASSIGNMENT

To practice using DOS commands in this assignment, complete the following steps. Check off each step when you understand its operation or mark the step with a question mark if it is not clear. Write the commands or keystrokes in the space provided.

1. Warm-boot the computer.

2. Change the prompt to reflect the current directory.

3. Display a directory sorted by file size from smallest to largest. Print the screen. Print the screen again of a directory sorted by file size from largest to smallest.

4. Clear the screen.

5. Determine the version of DOS the computer is using.

6. Print, using PrtSc, a list of the files in the root directory of drive C that begin with a C and have any extension.

7. Print, using PrtSc, a list of the files in the root directory of drive C that have any filename but have an extension which begins with an E.

8. Request HELP for the PROMPT command using a switch. Print the screen.

9. Clear the screen and display a directory showing each filename and extension with no other information. Print the screen.

10. Clear the screen and display a wide directory. Print the screen.

CHAPTER 3

DISK AND FILE PREPARATION

OBJECTIVES

When you have completed the activities and assignment in this chapter, you will be able to:

1. Set a search path.
2. Abort DOS commands.
3. Format disks with the system and volume label.
4. Use the FORMAT and Unconditional FORMAT command.
5. Use the LABEL command to name or rename a disk.
6. Use the VOL command.
7. Change the default drive.
8. Specify disk drives in commands.
9. Use the DOSKEY command.
10. Create text files.
11. Display and print text files.
12. Use echo printing.

FLOPPY DISKETTES

Unlike the fixed hard disks that are sealed in a metal casing and never seen by the user, floppy diskettes are routinely handled. Forming good habits for handling and storing floppy diskettes can be one of the most important outcomes of your study of DOS.

Important features of floppy diskettes are shown in Fig. 3-1. Both 5 1/4- and 3 1/2-inch floppy diskettes can be write-protected. **Write-protected** means that the disk can only be read, not written to, and that the files on it cannot be changed or erased. On 5 1/4-inch floppy diskettes, the write-protect notch is a square notch cut in one side of the diskette. The notch can be covered with a small adhesive tab to make the disk "read only" and prevent erasing, formatting, or overwriting files on the disk.

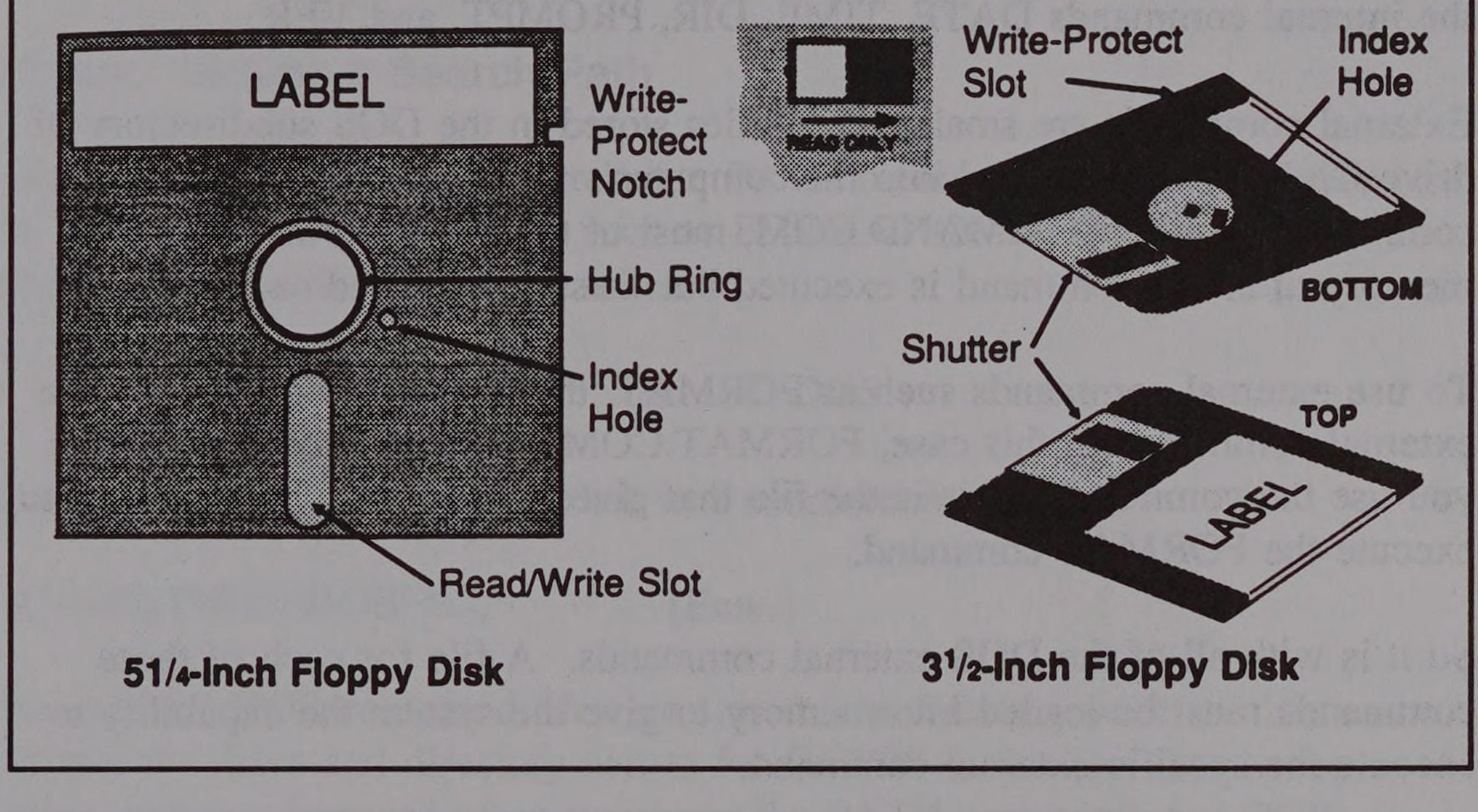

Figure 3-1. Features of floppy diskettes.

On the 3 1/2-inch diskette, the write-protect tab slides in a slot near one corner. To make this disk "read only" and prevent erasing, formatting, or overwriting files on the diskette, the sliding tab must be pushed toward the corner of the diskette so that the hole in the slot is open.

In many DOS operations accidental overwriting of files can easily occur. Because you seldom want to change application software program files, it is a good idea to always write-protect application software diskettes. Write-protecting your data diskettes helps safeguard your data but prevents you from adding to or updating the data. In this case, having a second copy as a backup may be a better security measure.

Both sizes of floppy diskettes have an indexing hole and a read/write slot in their protective sheath. The indexing hole is used by DOS to find locations on the diskette where files can be read and written. The read/write slot on both sides of the diskette jacket exposes the diskette surface for access by the drive read/write heads. On the 5 1/4-inch diskette, the Mylar surface is visible through these openings. On the 3 1/2-inch diskette, the read/write slot is protected by a metal shutter that slides to one side when the disk is inserted in the drive.

MORE ABOUT INTERNAL AND EXTERNAL DOS COMMANDS

As you know, the instructions you give DOS are **commands** that the software can interpret and carry out. You have learned that there are two types of DOS commands, **internal** and **external**. The internal commands are instructions contained in one large command file named COMMAND.COM. These instructions are **loaded**, or read into the computer's RAM, when you **boot** or power up the computer. They stay in the computer's memory and are available for use until you turn off the power supply to the computer. You have used the internal commands DATE, TIME, DIR, PROMPT, and VER.

External commands are smaller DOS files stored in the DOS subdirectory of drive C. They are not read into the computer's memory until you use the command. Unlike COMMAND.COM, most of these files do not stay in memory after the command is executed but must be reloaded each time.

To use external commands such as FORMAT, the DOS program file for the external command (in this case, FORMAT.COM) must be loaded each time you use the command. This is the file that gives the system the capability to execute the FORMAT command.

So it is with all of the DOS external commands. A file for each of these commands must be loaded into memory to give the system the capability to execute the specific external command.

SETTING THE SEARCH PATH TO THE DOS COMMAND FILES

Before DOS can load an external command file such as FORMAT.COM, the system must know where it is stored on the hard drive. If you want DOS to execute an external command, you must tell DOS the drive and directory where the DOS external command files are stored. Usually this directory is called DOS. If you are using PC/MS-DOS versions 4.0 or higher, the DOS external files were stored in a subdirectory called DOS on drive C during the installation process.

There are several ways to help DOS find the external command files. One way is to use the internal command PATH to tell DOS the drive or directory name where the files can be found. This is called "**setting the search path.**" For example, if the files are on the hard disk in a directory named DOS, you can enter this command at any prompt to set the search path to the place where the files are stored:

```
PATH C:\DOS
```

If the external command files are in a directory with a different name, you must use that name instead of "DOS." This PATH command will ensure that

the system can find any DOS files in that directory and execute them from any DOS prompt.

The system stores the search path in memory until the computer is turned off or rebooted. You need to enter the PATH command only once during a computing session, unless you cold-boot or warm-boot the computer. After a warm or cold boot, you must reset the path by entering the PATH command line again.

Practice: Setting a Search Path

At the DOS prompt, key the appropriate PATH command to tell DOS where to look for external command files, such as FORMAT.COM. Again, your DOS prompt may look like any of the following:

C> C:\> *or* C:\DOS>

If the DOS files are on the hard disk in a subdirectory, key

- C:\>**PATH C:\DOS** [Enter]

If the external DOS command files are stored in a directory with another name, substitute the drive and directory names for C:\DOS in the line shown above. Nothing apparent happens when you enter the PATH command, but DOS stores the entry in memory.

To check the current search path, key the following:

- C:\>**PATH** [Enter]

DOS displays the search path stored in memory.

```
C:\>PATH=C:\DOS
```

ABORTING DOS COMMANDS

If you discover that you are executing a command that you really don't want to complete, you can abort a command by holding down the Ctrl (Control) key and tapping the letter C. Using this key combination that stops any ongoing operation is sometimes referred to as "Pressing the Panic Button." You can also hold down the Ctrl key and tap the Break key. This tells DOS to abort the command process. However, if the process has already begun, any data on the disk may no longer be readable.

Practice: Aborting DOS Commands

For this exercise, you will key the DIR command and then decide to abort it as follows:

- C:\>**DIR** [Enter]

Quickly, press Ctrl-C or Ctrl-Break to stop the execution of the command. You may need to try this several times before you move quickly enough to be successful.

PREPARING DISKS FOR USE

Preparation of a floppy diskette and a hard disk is considerably different. To prepare a floppy diskette for use, simply format it. A hard disk often requires low-level formatting, partitioning, and formatting. The preparation of a hard disk is beyond the scope of this textbook. Prior to using a disk, there are several procedures which should be followed for maximum productivity.

Formatting a Disk

To make a disk ready for use on a DOS machine, the disk surface must be set up in a format that DOS recognizes for storing data. This setup process is called "formatting." Formatting is accomplished with the FORMAT command, an external DOS command. To use the FORMAT command, you must have the file FORMAT.COM in the DOS directory on the hard disk. Formatting does the following:

- Sets up a structure for the directory of the hard or floppy disk.
- Checks for defects in the disk.
- Makes previously recorded data or programs on the disk being formatted inaccessible.
- Prepares the disk so that the system can work with it by reading from and writing to its tracks and sectors with the disk drive read/write heads.

Data is stored on floppy and hard disks in the format shown in Fig. 3-2. On both types of disks, the tracks for recording data are arranged in concentric circles. The tracks are divided into sectors. A sector is a wedge-shaped section of the disk that is the basic unit of storage on a disk. When a disk is formatted, DOS maps the disk so that the operating system can find specific sectors and tracks. Each sector commonly stores 512 bytes. A 5 1/4-inch

360K double-density diskette is formatted with nine sectors numbered 0 through 8 and 40 tracks numbered 0 through 39 on each side of the disk. A 5 1/4-inch 1.2MB high-density disk has 80 tracks with 15 sectors per track on each side. A 3 1/2-inch 1.44MB high-density disk has 80 tracks with 18 sectors per track on each side. A 3 1/2-inch 2.88MB disk has 80 tracks with 36 sectors per track. DOS allocates space to files in groups of one to four (usually two) sectors called allocation units or clusters.

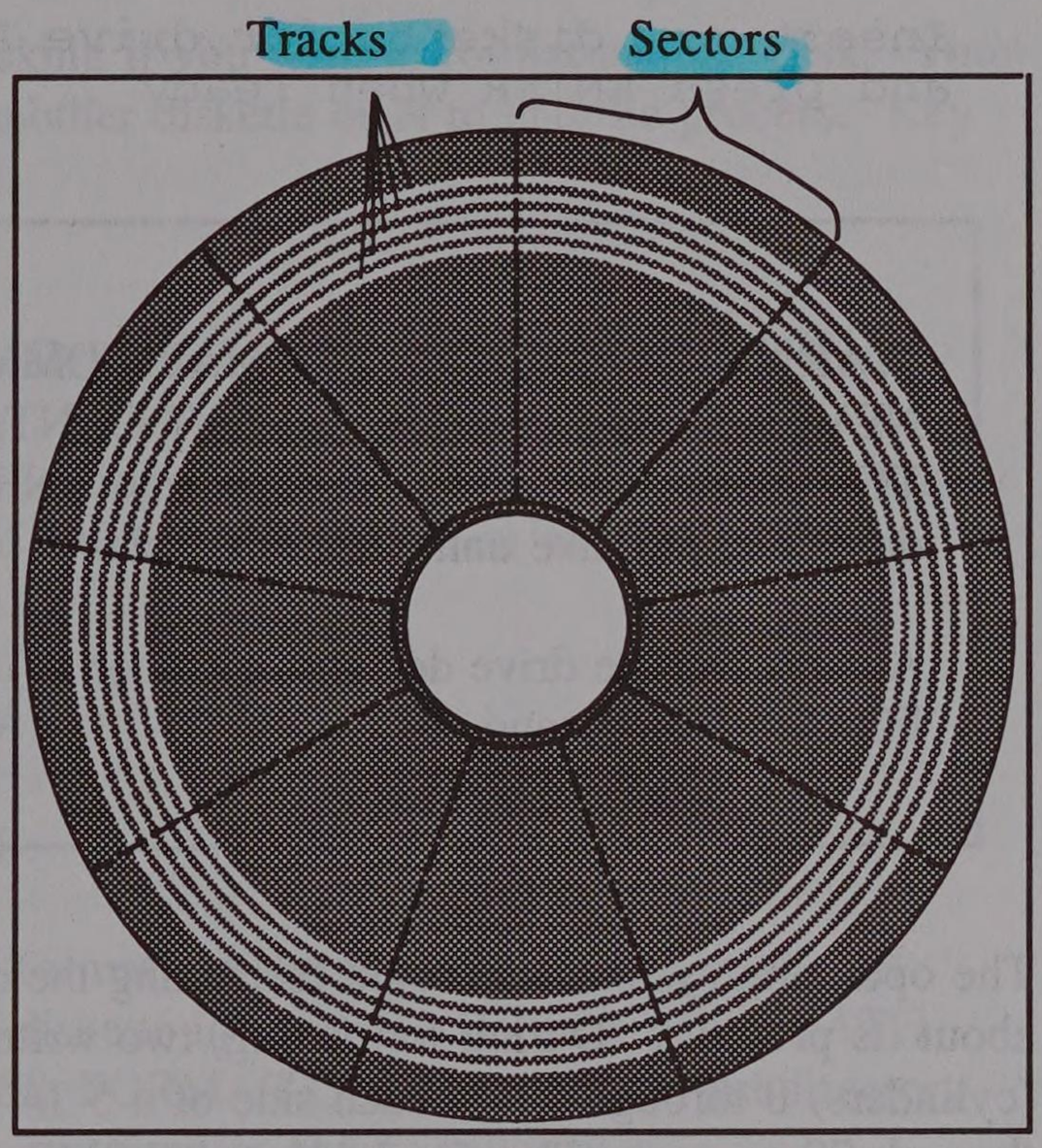

Figure 3-2. Information is stored on disks in sectors and tracks.

You can use switches to modify the FORMAT command, just as you used switches to modify the DIR command. You will use several switches available for the FORMAT command in the formatting activities later in this chapter.

The command syntax for formatting a disk from the DOS prompt is:

C:\>FORMAT A: (or other drive letter)

Practice: Formatting a Disk

Be sure the path is set to the DOS subdirectory on drive C, and write the following on an adhesive label to affix to a diskette:

(Your name)
(Your class)

- C:\>**FORMAT A:** [Enter]

This command tells DOS to format a disk in drive A. DOS will prompt you to insert a new disk. Insert the labeled blank diskette (or one with data of no value) in drive A. First the system will find the file FORMAT.COM on the hard disk in the DOS subdirectory. Next, DOS will load FORMAT.COM into memory. When the command is loaded, DOS will prompt you as follows:

```
Insert new diskette for drive A:
and press ENTER when ready . . .
```

WARNING: USE THE FORMAT COMMAND WITH CAUTION. NEVER FORMAT A DISK THAT CONTAINS DATA YOU WANT TO KEEP. Even though DOS has an UNFORMAT command, you don't want to take unnecessary risks!

Always use the drive designator (A: or B:) as a parameter after the FORMAT command.

The operating system will begin formatting the disk and will keep you updated about its progress. During formatting, two write heads (0 and 1) prepare tracks (cylinders) 0 through 39 on each side of a 5 1/4-inch 360K diskette or tracks 0 through 79 on each side of a 5 1/4-inch 1.2MB diskette or a 3 1/2-inch 720K, 1.44MB, or 2.88MB diskette. The number of the active head and the cylinder being formatted are displayed by many versions of DOS. Other versions display the percentage of the disk that has been formatted.

When the formatting process is complete, DOS will report:

```
Format complete.
```

You will be prompted to enter a "volume label" or disk name as follows:

```
Volume label (11 characters, ENTER for none)?
```

Key the name you want the disk to have, or press [Enter] to leave the disk with no label. For now, press [Enter]. (If you are using an earlier version of DOS and you want to name the disk, read the section "Formatting with the System" that follows.) With any version, DOS will report the total disk space, the amount of unusable space, if any, and the amount of space available. For a 5 1/4-inch 360K disk, the following information will be displayed:

```
362496 bytes total disk space
362496 bytes available on disk
```

If there are any "bad" or unusable areas on the disk, DOS will report the number of bytes in any bad sectors and will prevent their being used. The size and number of **allocation units** is also displayed.

There will be a screen prompt asking if you wish to format another disk. Your options are to key Y to format another diskette or N to end the process. Key N for No.

Formatting with Two Floppy Drives

Some computer systems have two floppy diskette drives. In this case you may wish to format a disk in either of these two drives. You would key the command as shown:

C:\> or C:\DOS>FORMAT A:
or
C:\> or C:\DOS>FORMAT B:

The first command tells DOS to format a diskette in drive A. The second command tells DOS to format a diskette in drive B. Because of the PATH command, the system finds the file FORMAT.COM in the DOS subdirectory and loads it into memory. DOS will prompt you as follows:

```
Insert a new diskette for drive A:     (or drive B:)
and press ENTER when ready . . .
```

When the formatting process is complete, DOS will report:

```
Format complete.
```

You will be prompted to enter a "volume label" or disk name as follows:

```
Volume label (11 characters, ENTER for none)?
```

Key the name you want the disk to have, or press [Enter] to leave the disk with no label. (If you are using an earlier version of DOS and you want to name the disk, read the section "Formatting with the System" that follows.) With any version, DOS will report the total disk space, the amount of unusable space, if any, and the amount of space available. For a 5 1/4-inch 360K disk, information similar to the following will be displayed:

```
362496 bytes total disk space
362496 bytes available on disk
```

There will be a screen prompt asking if you wish to format another disk. Your options are to key Y to format another diskette or N to end the process.

Formatting with the System

There are times when it is helpful to have on your data disk the three DOS system files that make it possible to boot the computer. If properly prepared with these files, a data disk can be inserted into drive A and used to start the computer. Such a disk is sometimes called a **self-booting** disk or a system disk, because it can be used in drive A instead of DOS files on a hard disk to boot the computer.

The three required system files are IBMBIO.COM and IBMDOS.COM in IBM PC-DOS, or their equivalent files in MS-DOS: IO.SYS and MSDOS.SYS, and COMMAND.COM. The first two are called **hidden files** because they do not appear in a directory listing. Nonetheless, they are present on the DOS disk or any self-booting disk. The third file COMMAND.COM appears in the directory listing. This file contains the internal commands that are loaded into memory when the computer boots and which stay there until the power is turned off.

The three system files can be added to a data disk when it is formatted. This is accomplished by modifying the FORMAT command with the system switch /S. To request DOS to format a data disk and install the system files on it from the hard disk, you would key the following at the DOS prompt:

C:\> or C:\DOS>FORMAT A: /S

Note: If you are using a version earlier than 4.0, you can also add the /V switch to tell DOS you want to label the disk:

C:\>FORMAT A: /S/V

Key the name you want the disk to have when prompted by the screen display.

Formatting with Different Floppy Disk Drive Configurations

In the years since DOS was introduced, many features and options have been added to DOS personal computers so that there is no longer a "standard" PC. Floppy disk drives are an example of this diversity. There are four configurations in common use: 360K and 1.2MB 5 1/4-inch drives, and 720K and 1.44MB 3 1/2-inch drives. Your computer may have drives in one or more of these configurations. If you work with more than one computer, you

might be using different floppy drive configurations on each machine. Knowing how to format and manage disks in several configurations can be essential to DOS users particularly if you wish to work with the same disk in multiple computers.

The 1.2 and 1.44MB diskettes are called "high-density" or "high-capacity" disks. They can be used only in high-density drives: the 1.2MB drives introduced with the AT machines and the 1.44MB drives introduced with the PS/2 machines. The 5 1/4-inch 1.2MB diskettes and the 3 1/2-inch 1.44MB diskettes are certified for high-density storage and are usually labeled as "HD" disks. You may also be able to recognize the 5 1/4-inch HD disk by its lack of a hub ring and the 3 1/2-inch HD disk by the square hole cut in the corner opposite the write-protection opening.

Low-density disks can be formatted and used in low- or high-density drives. However, high-density disks cannot be formatted for high density and used in low-density drives. If a diskette is already formatted, a high-density drive detects whether the disk is a high- or low-density disk and reads and writes data in the disk's format. By default, a drive attempts to format a disk in the drive's native format—if you wish to format a low-density disk in a high-density drive, you must add switches to the FORMAT command to specify this. The two most common modifications to the command are:

To format a 5 1/4-inch 360K disk in a 1.2MB drive:

```
C:\>FORMAT A: /4
      or
C:\>FORMAT A: /F:360          (DOS 4.0 and later)
```

To format a 3 1/2-inch 720K disk in a 1.44MB drive:

```
C:\>FORMAT A: /N:9  /T:80
      or
C:\>FORMAT A: /F:720          (DOS 4.0 and later)
```

The /4 switch tells DOS to format a 5 1/4-inch disk as a low-density 360K disk. Many 1.2MB drives will not format a disk labeled high density with this switch. As a general rule, it is more reliable to format a 360K (low density) 5 1/4-inch disk in a low-density drive. In fact, the DOS manual warns that a disk formatted with the /4 switch or /F:360 (DOS 4.0 and later) may not be written to or read reliably in a low-density drive. Before storing data on a disk formatted in this way, make sure the low-density drive can write data to the disk and read data from it. Whenever possible, format and use 5 1/4-inch disks in the type of drive for which they were designed.

The /N:9 switch tells DOS to format a 3 1/2-inch disk as a low-density disk with 9 sectors per track; the /T:80 switch specifies 80 tracks. You can use the

same switches in DOS 4.0 and later, or you can use the /F:720 switch to specify the size in kilobytes. Disks formatted in this way in a high-density drive are very reliable when used in a high- or low-density drive.

FLOPPY DISK CAPACITIES				
Size of Disk	Disk Capacity	Density	No. of Sectors	No. of Tracks
5 1/4-inch Disks				
	360KB	Double Density	9	40
	1.2MB	High Density	15	80
3 1/2-inch Disks				
	720KB	Double Density	18	80
	1.44MB	High Density	18	80
	2.88MB	High Density	36	80

Practice: Formatting a Disk with the System

Before formatting a floppy disk, check to see the size of the drive you have and make sure the diskette is compatible. Set the search path to the DOS files and then format with the following command:

This time, format the diskette with the system as follows:

- C:\>**FORMAT A: /S** [Enter]

Name the diskette CLASS when asked for a volume label.

Unconditional Format

By default, the DOS 5.0 and later versions of the FORMAT command perform a "safe" format, that is, one that can be unformatted. The directory and FAT (**file allocation table**) information of the existing format is stored as UNFORMAT information, and for the new format, a new directory and FAT are created to replace the old information. (The file allocation table is a special file created when the disk is formatted.) DOS uses the FAT to record the numbers of the clusters or allocation units where each file is located. When DOS erases a file, it changes the first character of the file name in the directory so that the file can no longer be accessed and revises the FAT to

show that the allocation units are available for other files. The data in the allocation unit(s) where the file is stored is not erased, however.

You will see how this is used to good advantage when you learn to undelete a file or unformat a disk in a later section. In a safe format, the remainder of the data on the disk is not changed, but the files are no longer accessible without the directory or FAT. If other files have not been written to the disk after a safe format, the DOS command UNFORMAT (covered in a later section) can be used to restore the original directory and FAT and make the files accessible again.

However, for security purposes if you want to completely erase a disk so that the files cannot be restored, you can format the disk "unconditionally" by using the switch /U, as in FORMAT A: /U. In this case, all data is physically erased from the disk and the disk cannot be unformatted. The command, FORMAT /U, cannot be used on a hard drive.

Labeling a Disk

An adhesive disk label can be peeled off accidentally, so DOS permits you to record a disk name or label electronically in a hidden file on the disk itself. At one time, disks were referred to as "volumes," so the electronically recorded disk name is called the **volume label**. The volume label you give the disk can be used to quickly identify its contents.

How can you give a disk an electronic label? The volume label can be recorded during the format process. The FORMAT command switch for giving a disk a volume label is /V. Therefore, the command to format and name a disk is FORMAT A: /V. DOS versions 4.0 and later automatically ask for a volume label even when the switch is not used.

In version 3.0, the external command LABEL was introduced. With it, you can add or change a disk label whenever you want. This is useful when the contents of a disk have changed or when you did not elect to give a disk a name when you formatted it.

Practice: Labeling a Diskette

Insert the diskette you formatted and labeled CLASS in drive A. Key the following at the DOS prompt (remember that LABEL is an external command, and you must have access to the command file):

- C:\>**LABEL A:** [Enter]

When prompted for a volume label, name the disk PRACTICE. You can also label the diskette by using the command:

- C:\>**LABEL A: PRACTICE** [Enter]

Repeat the LABEL command once more and change the name of the disk back to CLASS.

THE VOLUME (VOL) COMMAND

Have you wondered how you might later check the volume label of a disk? One way to do this is to display the directory; here you will see the name of the disk as well as the names of the files on it. Another way is to use the **VOL** (VOLUME) command. The command displays the name of the disk when used as follows:

C:\>VOL or C:\>VOL A:

Practice: Using the VOL Command

Check the volume label of the disk you just formatted with the following:

- C:\>**VOL A:** [Enter]

DOS will report the volume label as CLASS.

SPECIFYING DISK DRIVES IN COMMANDS

To instruct DOS to carry out an action, key a DOS command at the DOS prompt. As an example, use the command DIR. The DIR command instructs DOS to display a directory listing of the information stored on the disk in the drive shown in the prompt. DOS will display the directory for the disk in the default drive unless you specify otherwise. To show how this would work, key the command DIR at the prompt on your screen:

C:\>DIR or A:\>DIR

If you key DIR at the C:\> prompt, you know that the information displayed is the directory listing for the disk in drive C. If you key DIR at the A:\> prompt, the information displayed is for the disk in drive A. So, whenever you key a DOS command, the system assumes you are referring to the drive named in the DOS prompt or the default drive, *unless you specify otherwise* in the command line.

You can designate or specify a different drive by adding the **drive designator** after the command. The drive designator is the drive name followed by a colon (A: or C:). The drive designator is a **parameter**, additional information keyed after the command to specify exactly what you want DOS to do.

Whenever you want DOS to operate on a disk in a drive that is not shown in the prompt, you can instruct the system to do this by including a parameter, the drive designator, after the command. If you have a floppy diskette, you may wish to take time now to check the DIR in both the disk drives of your computer.

Practice: Specifying Disk Drives in Commands

Try the following to display the contents of drive C with either of the following commands:

- C:\>**DIR** [Enter] A:\>**DIR C:** [Enter]

Now try either of the following commands to display the contents of the disk in drive A.

- C:\>**DIR A:** [Enter] A:\>**DIR** [Enter]

CHANGING DEFAULT DRIVES

It is important to be able to navigate among disk drives to execute commands. For example, you may be at the A prompt and wish to go to the C prompt, or conversely, be at the C prompt and wish to go to the A prompt. If you are at the A prompt and wish to go to the C prompt, key the drive designator C:

```
A:\>C:
```

You will see the C prompt display as:

```
C:\>
```

If you have a formatted diskette in drive A, you can access the drive by keying the following:

```
A:
```

Practice: Changing Drive Prompts

With the formatted diskette, CLASS, in drive A, let's change the drive prompt as follows:

- C:\>**A:** [Enter]

You should see the A prompt display such as:

```
A>   or   A:\>
```

Now change back to drive C with:

- A:\>**C:** [Enter]

USING DOSKEY

DOSKEY is a memory-resident program that records the DOS commands you have keyed in a **command history** buffer. From the buffer, you can recall, revise, and reissue commands. With DOSKEY you can access a long series of earlier commands.

Keys for Use with DOSKEY	
Up and Down Arrows	Recall commands
Escape	Clears command line
Page Up	Recalls first command in memory
Page Down	Recalls last command in memory
F6	Perform DOS editing key functions as do Ins, Del, and Backspace
F7	Displays a numbered list of commands in memory
Alt-F7	Clears the command history
F8	Searches command history for most recent command starting with character(s) you press next
F9	Selects command by number entered next

Practice: Using DOSKEY

Set a path to the DOS files and install DOSKEY in memory by keying the following at the DOS prompt:

- **C:\>DOSKEY** [Enter]

DOS should display the message: `DOSKEY installed.`

Key the following:

- **C:\>CLS** [Enter]
- **C:\>VER** [Enter]
- **C:\>DIR/W** [Enter]

Press the **up arrow key**; do you see DIR/W? Press **F7**. You should see the following numbered list of commands:

```
1.  CLS
2.  VER
3.  DIR/W
```

Key **F9** and **1** and press [Enter] to recall line 1, the CLS command. Press Enter to repeat the command.

DOSKEY uses the DOS editing keys which you will learn in a later chapter and other cursor control and function keys. See the following list.

You may wish to clear any former command history. To do so, key:

- **Alt-F7** to clear any former command history.

Use **Ctrl + T** (displays the symbol ¶) to string multiple commands on a line as in DIR A: ¶ DIR C:.

String multiple commands using DOSKEY with the following:

- **C:\>VER ¶ VOL A:** [Enter]

WORKING WITH TEXT FILES

A text file is a file that, as the name implies, stores a document or other text—perhaps a short message, letter, or memo. An easy way to create text files for all of these purposes is to use the DOS internal command COPY with

CON, which is short for console. CON is the device name given to the keyboard and display screen. The command syntax is:

COPY CON [filename.ext]

This instructs the system to copy input from the keyboard (CON) and store the input in the file you named in the command line. After you key the command line and press [Enter], DOS will begin storing in memory the text you key. You can key lines of text, pressing [Enter] at the end of each line. Word wrap is not available in this command.

If you make a keying error and have not pressed [Enter] on that line, you can backspace until you have deleted the error and rekey that portion of the line. If, however, you have already ended a line by pressing [Enter], you won't be able to correct the error at this time but must edit the file with an editing program, such as EDLIN, EDIT (version 5.0 or later), E, or a word processor. Later chapters in this book will explain how to use EDLIN or EDIT, but for now, just ignore your typos.

When you have finished keying the text, hold down the Ctrl key and press the letter Z (Ctrl-Z) or the F6 function key to signal the system that you want to end the file. When you press [Enter], DOS will copy the text stored in memory to a disk file and give it the name you specified in the COPY CON command line.

> **Reminder:** When you do not specify a disk drive, your command will apply to the drive shown in the prompt.

As you will see in this and later chapters, you can utilize a text file in a variety of ways. You can view it on the screen, print it, revise it, combine it with other files, and make duplicate copies of it.

Practice: Creating Text Files

Insert the CLASS diskette in drive A and create a text file. Key the following command at the A prompt:

- A:\>**COPY CON DOC1.TXT** [Enter]

After you key the COPY CON DOC1.TXT command line and press [Enter], the cursor returns to the left margin and is ready to receive your input. Key

the text shown in bold type that follows. Substitute your name and current date. To create a blank line, simply press [Enter].

If you make a error while keying the document and have not pressed [Enter] at the end of the line, backspace until you have deleted the error and rekey that portion of the line. If, however, you have already ended a line by pressing [Enter], you won't be able to correct the error. If you want to start over, use Ctrl-C keys to cancel the COPY CON command and return to the DOS prompt. Then rekey the command and the file from the beginning. Later you will learn how to edit text files such as this:

A:\>COPY CON DOC1.TXT
- **(your name)** [Enter]
- **(the current date)** [Enter]
- [Enter]
- **This is my first text file.** [Enter]
- **It will be stored on my disk** [Enter]
- **in drive A.** [Enter]
- [Enter]

To end and store your file, hold down the Ctrl key and then press the letter Z key (Ctrl-Z) and press [Enter]. Or you can press the F6 function key and [Enter] to accomplish the same purpose. Either way, these lines will display:

- ^Z [Enter]
  ```
  1 file(s) copied
  ```

(When you press Ctrl-Z or F6 and [Enter] to end the file, watch for the drive A light to flash signaling that the file is being written to disk.) You should be back at the DOS prompt. Now look at the directory of the disk on which you stored the DOC1.TXT file.

- A:\>**DIR** [Enter]

You will see the name of your disk and a list of the files on the disk. In this list you should see the new DOC1.TXT file. Notice that in the directory list of files, the filename and extension are in separate columns with spaces between the filename and extension. However, when you key the filename and extension in a DOS command line, you must separate them only with a period (called a "dot" in computer lingo), as in the following:

DIR COMMAND.COM *or* DIR DOC1.TXT

Now try creating a text file and storing it on a disk that is *NOT* in the default drive. Change to drive C and use the COPY CON technique to create a file called DOC2.TXT.

- **C:\>COPY CON A:DOC2.TXT** [Enter]
- **This is my second text file.** [Enter]
- **I keyed A: in front of the** [Enter]
- **file name so the file will be** [Enter]
- **stored on my disk in drive A.** [Enter]
- **(your name)** [Enter]
  [Enter]
- **[F6]** *or* **[Ctrl-Z]** [Enter]

Use the DIR command to see if DOC2.TXT is on the disk in drive A. To tell DOS to look at the disk in drive A, you must key the drive designator A: as well as the file name DOC2.TXT. This combination of the drive designator and the file name is called a **filespec**. There are no spaces in a filespec.

- **C:\>DIR A:DOC2.TXT** [Enter]

DOS will display the directory information for DOC2.TXT.

Displaying the Contents of Text Files with the TYPE Command

You have seen DOC1.TXT and DOC2.TXT listed in the directory of the disk in drive A, but to see if the system actually recorded the contents of what you keyed, you can use the DOS internal command TYPE. DOS provides this easy way to display the contents of files, both on the screen and printed on paper. To display the contents of the file DOC2.TXT on the screen, key the command as follows:

A:\>TYPE DOC2.TXT *or* C:\>TYPE A:DOC2.TXT

Remember, if the file is not in the default drive shown by the prompt, you must include the drive designator as part of the filespec.

It is also possible to display DOS command files, such as FORMAT.COM, with the TYPE command, but these program files are written in machine language and stored in code that cannot be read by the layperson.

Practice: Displaying Text Files

Display the contents of DOC1.TXT with the following:

- **A:\>TYPE DOC1.TXT** [Enter]
  or
- **C:\>TYPE A:DOC1.TXT** [Enter]

PRINTING FILES

It is important to be able to record on paper what is on the screen or stored on disk. There are several ways to print files.

Print Screen

The first method you learned was to print the contents of the screen; this is like taking a picture of what is currently on the screen.

Echo Print

Echo printing is a method of continuous printing that "echoes" everything that appears on the screen. This works best with a dot matrix printer. While output to the printer is on, anything that you key or that DOS displays on the screen also goes to the printer and is reproduced as hard copy. To turn on the echo-print option, press the Ctrl key and the PrtSc or the P key (Ctrl-PrtSc or Ctrl-P).

These key combinations act as toggle switches. When you press the keys once, the echo or continuous print function is turned on; press the keys again, and the function is toggled off. If you turn off the printer, you turn off the echo-print function.

Not all printers are capable of echo printing. In some networked systems where many computers share a printer, echo printing cannot be used. If your system, for whatever reason, does not have the capability for echo printing, read this section but do not enter the commands.

Using the TYPE Command to Print

Previously you displayed the contents of a file using the DOS internal command TYPE. You can also use that command to print the file. First, you would be sure that a printer is connected to your computer, is turned on, and is on-line. Next, key the following command to print the file DOC2.TXT:

```
A:\>TYPE DOC2.TXT > PRN
```
or
```
C:\>TYPE A:DOC2.TXT > PRN
```

In this command line, the greater-than sign (>) redirects the output of the TYPE command from the screen, where it would normally go, to the printer. PRN is the system's device name for the printer, like CON is the device name

for the console. So here you are telling the system to TYPE or display the file DOC2.TXT on the printer, rather than on the screen.

Using the COPY Command to Print

You can also use the COPY command to display and print the file. To print the file using the COPY command, key the following:

A:\>COPY DOC2.TXT PRN

This instructs the system to copy the file DOC2.TXT from the drive A disk to the printer. The message `"1 file(s) copied"` appears on the screen but not on the printed copy.

Practice: Printing Text Files

Try the various ways of printing you have learned so far with the following:

1. Clear the screen and key the command to display the directory of the disk in drive A. Print the screen.

   ■ **C:\>DIR A:** [Enter]

   Press the Shift and PrtSc keys or just PrtSc depending on your system.

2. Using the TYPE command, print the DOC1.TXT file as follows:

   ■ **C:\>TYPE A:DOC1.TXT > PRN** [Enter]

3. Using the COPY command, print the DOC1.TXT file as follows:

   ■ **C:\>COPY A:DOC1.TXT PRN** [Enter]

4. To echo print, while you are holding down the Ctrl key, tap the PrtSc or key the letter P. (With some systems, only one of these key combinations will work.) Nothing will happen until you key the next letter. The next letter you key will print. To print the directory, key

   ■ **C:\>DIR**

To turn off the echo-print function, key the Ctrl + PrtSc or Ctrl + P key combinations again.

Great work on Chapter 3! Test your knowledge by answering the Self-Check Quiz and working through the Assignment.

CHAPTER 3 SELF-CHECK QUIZ

True/False (Write T if the statement is true, F if the statement is false.)

_____ 1. The search path provides access to the internal DOS commands.

_____ 2. To rename a disk without reformatting it, use the LABEL command.

_____ 3. By default the FORMAT command performs an unconditional format.

_____ 4. TYPE, PATH, and VOL are internal commands.

_____ 5. DOSKEY will store a history of the commands used during the current computer session.

Multiple Choice (Circle the correct answer.)

6. The TYPE command can be used to:

   a. key text files.
   b. display any file on the screen.
   c. display only text files on the screen.
   d. determine what type of drive you are using.

7. If drive A is the default drive, what command would you use to create a file named MYDOC in drive B?

   a. B:COPY CON MYDOC
   b. COPY CON B:MYDOC
   c. TYPE B:MYD
   d. COPY B:MYDOC CON

8. Which of the following do you use to indicate the end of a text file?

   a. Ctrl-Z
   b. Ctrl-E
   c. Ctrl-T
   d. Ctrl-C

9. Which two commands will print a copy of the file DOC4.TXT?

   a. TYPE DOC4.TXT PRN
   b. TYPE DOC4.TXT > PRN
   c. COPY DOC4.TXT PRN
   d. COPY DOC4.TXT > PRN

10. Which command would you use to create a self-booting disk in drive A?

   a. FORMAT \S A:
   b. COPY /S A:
   c. FORMAT A: /S
   d. FORMAT A: SYS

ASSIGNMENT

To practice using the commands in this assignment, complete the following steps. Check off each step when you understand its operation or mark the step with a question mark if it is not clear. Write the commands or keystrokes in the space provided.

1. Change the prompt to show the current directory and set the search path to the DOS files.

2. Activate the DOSKEY command, or if it is already activated, clear the command history.

3. Format a second floppy diskette with the system; name it TEST.

4. Clear the screen. Display a directory of the disk in drive A. Print the screen.

5. Rename the diskette to LAB FILES. Display the volume label of the disk. Print the screen.

6. Create a text file named A-DOC.TXT on the LAB FILES diskette. Compose your own wording, but include your name and date in the file. Print the file using the COPY [filename.ext] PRN technique.

7. Display the contents of A-DOC.TXT using the TYPE command. Print the screen.

8. Display a directory of the LAB FILES data disk. Then check to see what version of DOS you are using. Print the screen.

9. Using the DOSKEY command, display a list of the commands used during this assignment. Print the screen.

---

10. Complete the following chart on printing techniques:

COMPARISON OF PRINTING METHODS		
METHOD	ADVANTAGES	DISADVANTAGES
Print Screen		
Echo Print		
Copy File to Printer (Using COPY Command)		
Redirect File to Printer (Using TYPE Command)		

CHAPTER 4

FILE-HANDLING COMMANDS

OBJECTIVES

When you have completed the activities and assignment in this chapter, you will be able to:

1. Make copies of files on the same disk.
2. Copy files onto a different disk.
3. Use wildcards with the COPY command.
4. Combine files with the COPY command.
5. Compare files with the COMP or FC command.
6. Rename files with the REN command.
7. Move files with the MOVE command.
8. Use the DOS Editing keys.

STRUCTURE FOR FILE-HANDLING COMMANDS

The file-handling commands described in this chapter (COPY, FC, COMP, RENAME, and MOVE) all use the same structure or syntax in the command line. For all of them, the command is followed by the **source** filespec and then the **destination** or target filespec. You can mentally translate the command line as including:

COMMAND		*SOURCE*		*DESTINATION*
COPY	(from)	FILE1.DOC	(to)	FILE2.DOC

The words "from" and "to" are represented by spaces separating the command and its two parameters, source and destination. With few exceptions, both the source and destination must be specified in the command line.

COPY FILE1 FILE2 COMP FILE1 FILE2 RENAME FILE1 FILE2

COPYING FILES

The COPY command is one of the most useful internal commands of DOS. You have already used a form of COPY to create and print text files; COPY will also duplicate files, make copies of files on other disks, and combine files. Duplicate files are useful when you want to back up or create variations of the same document. You can make several copies and modify each one with a word processor or with the DOS editors, EDLIN, EDIT, and E. For example, you could use this technique to create notes to clients or personalized thank-you messages. You will work with EDLIN in Chapter 8 and EDIT in Chapter 9. The text editor, E, is not covered in the text.

Or you might want to make a copy of a file you have created to give to someone else. You will often use COPY to make backup files for protection in case a floppy disk is lost or damaged. You can use COPY to move files from one disk to another or from one directory of a disk to another directory. And, you can use COPY to combine files into a new file and to add or append information to an existing file.

There are three options for using COPY to copy files:

- You can make a copy with the same name.
- You can make a copy with a different name.
- You can combine files into one file.

Here are some important file naming conventions you must remember when you use any of these options:

- Each file in a directory listing must have a unique name. If you are copying a file into the same directory of a floppy or hard disk, you must give the copy a different name from the name of the file being copied. Changing at least one character in the filename or extension will satisfy this requirement.

- If you are copying a file onto another floppy disk, you can use the same name or a different name. Copies of a file with the same name can reside on different disks, as they will appear in different directory listings.

> **Note:** One important caution about this command is that if there are any files on the target disk with names identical to names on the source disk, the files on the target disk will be overwritten. However, DOS 6.2 has added a prompt that advises that a file of the same name exists and gives you the opportunity to overwrite or not.

- More than one file with the same name can be stored on a disk if the files are in different directory listings. You will learn about creating and working with subdirectories in Chapters 6 and 7.

Copying Files to the Same Disk

When copying a file onto the same disk, you need to ensure that there is room for the copy, that the file to be copied is on the source disk, that you give a unique name for the copy, and that no file with the new name already exists on the disk. Except for MS-DOS v6.2, files with the same name will be overwritten without warning. Display the directory to be sure the file is listed and that there is no file with the new name you plan to give to the copy. Since the copy will be listed in the same directory as the source file, you must give the copy a *new and unique name*. A name is different if you change one character in the file name or the extension. In this case, you will change the entire extension to change the file name, DOC1.TXT to DOC1.DOC with the following:

```
COPY DOC1.TXT DOC1.DOC
```

Remember that COPY is an internal command; therefore, you do not have to access the DOS external command files.

Practice: Copying a File to the Same Disk

For this practice, insert the CLASS diskette, copy the file, and give the new file a different extension, NEW:

- A:\>**COPY A:DOC1.TXT A:DOC1.NEW** [Enter]

  or

- A:\>**COPY DOC1.TXT DOC1.NEW** [Enter]

You have asked the system to copy the existing (source) file DOC1.TXT *from* the disk in drive A *to* the same disk and to give the destination file a different name, DOC1.NEW.

Note that if your disk is in the default drive A and you want DOS to find the source file there and place the copy on the same disk, the drive designators (A:) before the file names are not necessary. With either entry above, the system will respond with the message

```
1 file(s) copied.
```

If, instead, you see the error message "`Bad command or file name`" or "`File not found,`" carefully examine the command line you keyed. Did you spell the filename exactly as it is spelled in your directory? Did you key a period (not a comma) between the filename and the extension? If you used the drive designator, did you key a colon (not a semicolon) after the letter A? Did you space the command and filenames exactly as shown above, with spaces only after COPY and the first file name? DOS *insists* that command lines such as this be keyed perfectly! If you find an error, key the line again.

Clear the screen and use the TYPE command to verify that DOC1.NEW is indeed an exact copy of DOC1.TXT.

- A:\>**TYPE DOC1.TXT** [Enter]

- A:\>**TYPE DOC1.NEW** [Enter]

Do the two files look the same? Later in this chapter you will learn how DOS can verify that two files are identical.

Copying Files onto Another Disk

There are many times when you will want to copy a file onto a different diskette. Follow the instructions that fit your system's configuration of one or two floppy drives.

One Floppy Drive System

If your system has only one floppy drive, DOS will use that drive as both drive A and drive B. For example, if you would like to copy the file TEXT1.DOC to TEXT1.DOC on another diskette from the same floppy drive, you would key the following:

```
A:\>COPY A:TEXT1.DOC B:
```

Here you have asked the system to copy the source file, TEXT1.DOC, from the disk in drive A. The destination for the copy is the disk in drive B. Because you did not specify a new file name after the B:, the system knows you want the new file to have the same file name as the source. In this example, you

must include the destination drive designator B:, but since the source file is in the default drive, keying A: before the source file name is optional.

The system will load the contents of the source file TEXT1.DOC into RAM and then will instruct you to exchange the floppy disks so that it can write the copy onto the destination disk. The system will prompt you with the following message:

```
Insert diskette for drive B: and press any key when
ready.
```

Remove the source diskette and insert a different disk (the destination or target disk) into drive A. After you press a key, DOS will write the file from memory onto the disk and report:

```
1 file(s) copied
Insert diskette for drive A: and press any key when
ready.
```

Do as prompted and insert the source diskette and press any key.

Practice: Copying a File Using a One Floppy Drive System

Let's practice! Insert the CLASS diskette and key the following at the A:\> prompt. Follow the prompts to insert the LAB FILES diskette as the target diskette.

- A:\>**COPY A:DOC1.TXT B:** [Enter]

  or

- A:\>**COPY DOC1.TXT B:** [Enter]

Display the directory of both the CLASS and LAB FILES diskettes to see that DOC1.TXT is listed.

Two Floppy Drive System

Copying files is faster and easier if you have two floppy drives. It is *not* essential that these two drives be the same size to be able to copy files from one to the other. Insert the CLASS diskette into drive A. The destination drive will be drive B. To copy the file COMPUTER.FIL, you would key the following:

```
A:\>COPY A:COMPUTER.FIL B:
```

Here you have asked the system to copy the source file COMPUTER.FIL from the disk in drive A, to the destination disk in drive B. Because you did not specify a new file name after the drive B the system knows you want the new file to have the same file name as the source. In this example, you must include the destination drive designator B:, but if the source file is in the default drive, the A: before the source file name is optional. DOS will load the contents of COMPUTER.FIL into RAM and then write the file onto the drive B disk and report:

```
1 file(s) copied
```

Practice: Copying a File Using a Two Floppy Drive System

Try this operation after inserting the CLASS disk in drive A and the LAB FILES disk in drive B:

- C:\>**COPY A:DOC2.TXT B:** [Enter]

or

- A:\>**COPY DOC2.TXT B:** [Enter]

Display the directory of the disk in drive B to see if DOC2.TXT is listed:

- A:\>**DIR B:** [Enter]

Copying Files to the Hard Drive

The same principles apply when copying files to the hard drive. To copy the file, COMPUTER.CGM from drive A to the hard drive, key the following:

COPY A:COMPUTER.CGM C:

Here you have asked the system to copy the source file COMPUTER.CGM from the disk in drive A. The destination for the copy is drive C. Because you did not specify a new file name after the C:, the system knows you want the new file to have the same file name as the source. In this example, if you do not include the destination drive designator C:, the file will be copied to the default drive resulting in the error message, "A file cannot be copied onto itself.".

DOS will load the contents of COMPUTER.CGM into RAM and then write the file onto drive C. Then DOS will report:

```
1 file(s) copied
```

Practice: Copying a File to the Hard Drive

Be sure that the source disk, CLASS, is in drive A and the destination is drive C. Key the following:

- C:\>**COPY A:DOC2.TXT C:** [Enter]

or

- A:\>**COPY DOC2.TXT C:** [Enter]

Display the directory of drive C to see if DOC2.TXT is listed.

- A:\>**DIR C:** [Enter]

Copying and Giving the Destination File a Different Name

You can also choose to give the destination file (the copy) a *different name*. For example, you might wish to copy OLDNAME.TXT and give the file a new name, NEWNAME.TXT, with the following:

```
COPY A:OLDNAME.TXT C:NEWNAME.TXT
```

Here you told the system to copy the OLDNAME.TXT file from the disk in drive A, to place the copy on drive C, and to give the duplicate file a different name, NEWNAME.TXT.

Practice: Copying and Giving the Destination File a Different Name

Try this new concept by keying the following command:

- C:\>**COPY A:DOC2.TXT C:NEWNAME.DOC** [Enter]

or

- A:\>**COPY DOC2.TXT C:NEWNAME.DOC** [Enter]

Another variation is to copy a file from another drive to the default drive. From the A prompt, copy NEWNAME.DOC from drive C (source) to the disk in drive A (destination) with this instruction:

- C:\>**COPY NEWNAME.DOC A:** [Enter]

or

- A:\>**COPY C:NEWNAME.DOC** [Enter]

Both of the preceding command lines tell the system to make a copy of the file NEWNAME.DOC in drive C, to place the copy in the default drive A, and to give the copy the same name.

Copying Files Using Wildcards

A useful technique for copying files is that of using wildcards when specifying files for copying. Used with the COPY command, the wildcards ? and * function in the same way as when used to specify the file names with the DIR command as explained in Chapter 2.

Remember, the wildcard ? stands for a single character in a filename or extension; the wildcard * stands for all characters to the right of the * in the filename or extension.

By using wildcards, you can make copies of a series of files that have similar names by entering only one command. Understanding how wildcards are used can save time in copying files and will also help you make decisions about naming files to facilitate the use of wildcards. For example, if you have files for Mr. Johnson and have begun all his file names with J such as J-TXT1.DOC, J-TXT2.DOC, and so on, you would copy all the files with the following commands:

COPY A:J-TXT?.DOC B: *or* COPY A:J*.DOC B:

These would select all the files beginning with J-TXT or J and ending with the extension DOC regardless of the letters following J-TXT or J.

To copy all of the files with the DOC extension, from drive A to drive B, key the following:

COPY A:*.DOC B:

Practice: Copying with Wildcards

To begin this activity, display the directory of your CLASS disk to see if you have files named DOC1.TXT and DOC2.TXT files. Use the following commands to copy both files:

- **A:\>COPY DOC1.TXT A-DOC.TXT** [Enter]
- **A:\>COPY DOC2.TXT B-DOC.TXT** [Enter]

Check your directory to see that you have the following files. (You may have more, but be sure you have these listed.)

```
DOC1.TXT                    B-DOC.TXT
DOC2.TXT                    NEWNAME.DOC
A-DOC.TXT                   DOC1.NEW
```

Now you will see how wildcards can be used in making copies of files. First experiment with the ?, which stands for a single character in a file name. You can use it to make copies of the files A-DOC.TXT and B-DOC.TXT. You will name the new copies A-DOC.NEW and B-DOC.NEW. The ? will stand for the A and B at the beginning of each file name. Key the following command line:

- A:\>**COPY ?-DOC.TXT ?-DOC.NEW** [Enter]

You have told DOS to copy all the files that have any first letter followed by DOC as a filename and the extension TXT. DOS is to give the files the same filename but change the extension to NEW. A list of the files is displayed as they are copied:

```
A-DOC.TXT
B-DOC.TXT

2 file(s) copied
```

Note that you could not use the * wildcard in place of the ? in this command because the * would substitute for all characters to its right in the filename. This would have been the same as entering the command

```
A:\>COPY *.TXT *.NEW
```

and would have copied every file with the extension TXT.

Next, copy all the files that begin with DOC and have TXT as the extension. In addition to copying them, change their extension to NEW.

At the A:\> prompt, key:

- A:\>**COPY A:DOC*.TXT A:DOC*.NEW** [Enter]

or

- A:\>**COPY DOC*.TXT DOC*.NEW** [Enter]

The system will identify all the files on the disk that begin with DOC and have the extension TXT, make copies of them on the disk, and give the copies the same filename but a different extension, NEW. A list of the files is displayed as they are copied. Display the directory to see the file copies listed.

Try another example. Copy all the files that have the extension NEW onto the same disk, give them the same filename, but give them your initials as the extension. Mary Todd Lincoln would use this command:

```
COPY *.NEW *.MTL
```

Practice using your own initials with

- A:\>**COPY *.NEW *.[your initials]** [Enter]

Display a directory of the disk to see a listing of your files.

Combining Files into a New File

Another application of the COPY command is to **combine** the contents of two or more small files into one larger file. For example, if you had several text files that contained memos written during the week, you might want to archive them as a group into a single file. Wildcards can be used in this process.

Given two files, NEWNAME.DOC and OLDNAME.DOC, you would combine the files by keying the command below. Key a plus (+) between the file names, and leave a space between these file names (the source files) and the name of the new, combined file (the destination file):

```
COPY NEWNAME.DOC + OLDNAME.DOC TOGETHER.DOC
```

The system will respond with the following message:

```
NEWNAME.DOC
OLDNAME.DOC

  1 file(s) copied
```

The two files listed were combined into one file; therefore, the system reports that one file was copied.

Use the TYPE command to display TOGETHER.DOC. You will see that both files have been combined into a single file.

Practice: Combining Files into a New File

Insert the CLASS diskette in drive A. Use the TYPE command to see what the two files, DOC1.TXT and DOC2.TXT, contain. Then combine DOC1.TXT and DOC2.TXT and put the results in DOC3.TXT.

- A:\>**TYPE DOC1.TXT** [Enter]
- A:\>**TYPE DOC2.TXT** [Enter]
- A:\>**COPY DOC1.TXT + DOC2.TXT DOC3.TXT** [Enter]

The system will respond with the following message:

```
DOC1.TXT
DOC2.TXT

1 file(s) copied
```

Use the TYPE command to display DOC3.TXT. You will see that both files have been combined into a single file.

C:\>TYPE A:DOC3.TXT [Enter]

Now use the copy command to combine DOC3.TXT and A-DOC.TXT and put the results in DOC4.TXT:

- A:\>**COPY DOC3.TXT + A-DOC.TXT DOC4.TXT** [Enter]

Use the TYPE command to display DOC4.TXT. Does it contain the contents of both DOC3.TXT and A-DOC.TXT?

Using Wildcards When Combining Files

To combine groups of files, you can also use wildcards in the command. You would key this command line to combine all files with the DOC extension:

```
A:\>COPY *.DOC LINKED.DOC
```

As DOS combines all the source DOC files into the destination file, LINKED.DOC, the system displays the list of files being merged. Use the DIR command to verify that LINKED.DOC is listed. This process does not change or remove the original files; you will see all of the DOC files also.

Use the TYPE command to be sure that LINKED.DOC is a combination of all the files with the DOC extension.

Practice: Combining Files with Wildcards

Try combining files with wildcards by using the following command with the CLASS diskette in drive A:

- A:\>**COPY *.TXT COMBINE.DOC** [Enter]

Use the DIR and TYPE commands to check to see that all files exist and contain what they should.

Appending Files

You can also combine files adding information to an existing file. This is called **appending**. For example, you could append or add a file containing a list of references to a résumé file. In this case, the *first* file listed in the command line is the file to which the contents of the other file(s) should be added.

To see how this works, use the TYPE command to view the contents of NEWDOC.TXT and then OLDDOC.TXT. After you see what each file contains, add the file OLDDOC.TXT to NEWDOC.TXT by keying the following command:

```
COPY NEWDOC.TXT + OLDDOC.TXT
```

DOS will add or append the *second* file listed to the first file. After this operation, the file OLDDOC.TXT will be unchanged, but its contents will have been added to the end of NEWDOC.TXT.

> **Note:** The sequence of files is very important when combining files into an existing file. The contents of the second file is added to that of the first.

Practice: Combining Files without Creating a New File

Now, with the CLASS diskette in drive A, add A-DOC.TXT to DOC3.TXT as follows:

- A:\>**COPY DOC3.TXT + A-DOC.TXT** [Enter]

Use the TYPE command to confirm that DOC3.TXT does now indeed contain both files. Note that if one of the files had been on another disk, you could have combined them simply by adding the drive designator in front of the file name. The contents of the appended file will be placed in the file in the location shown in the prompt.

If you want the file to be placed on the disk in another drive, add the drive designator at the end of the command line. For example, if you would like the above example to be placed in a disk on drive B, key the following:

- A:\>**COPY DOC3.TXT + A-DOC.TXT B:** [Enter]

If you have a one-floppy diskette system, you will need to exchange the diskettes. Use the DIRectory command to see that the new file is on the disk in drive B.

You can use this append technique to print two or more files as if they were one file. Display DOC1.TXT and DOC2.TXT again with the TYPE command. Then make sure your printer is on-line and enter:

■ A:\>**COPY DOC1.TXT + DOC2.TXT PRN** [Enter]

Both files will be printed as one file. Neither file will be changed, as the output went to the printer (PRN) and was appended in the printed copy only.

DOS places an invisible marker at the end of a combined or appended file. If you combine this file again with a command that uses wildcards, the marker is not always removed. If this happens, there will be an erroneous end-of-file marker in the middle of the combined file! DOS stops any operation on a file when it encounters an end-of-file marker. Because portions of such a file would be ignored, using wildcards to combine *previously combined files* is not a reliable method. *Always* check the file contents when you use this technique.

COMPARING FILES

There will be occasions when you will want to compare one file to another. If there are differences in the two files, the COMP command will tell you.

COMP is not available in MS-DOS prior to version 3.3. Use the command FC instead of COMP. In IBM PC-DOS, only COMP is available prior to version 5.0, which has both commands. In IBM PC-DOS, version 6.1, both FC and COMP can be used. In the case of IBM PC-DOS, version 6.1, when you use FC the contents of each file are actually shown for your review. The external commands COMP and FC compare two files, byte for byte.

If a difference is found, you will be told where the difference occurs and the contents of the bytes that differ. By default, COMP compares files until it finds ten mismatches, then it concludes that further comparison is useless and reports:

```
10 mismatches - ending compare
```

By default, COMP and FC will not compare two files of different sizes. If you attempt to compare files of different sizes, the system may beep and will display a message like:

```
Files are different sizes.
```

A comparison is made until the end-of-file (EOF) marker Ctrl-Z is found. If there is no mark, the system reports this. Sometimes discrepancies in compared files occur after the unmarked "end" of the file; therefore, COMP reports when there is no end-of-file marker.

Since COMP and FC are external commands, you must set a search path for DOS to find the external command files. ***If you are using DOS files on the hard drive***, enter the following PATH command:

A:\>PATH C:\DOS

If the DOS files are in another directory on the hard disk, substitute its name in the command line.

Compare OLDDOC.TXT with NEWDOC.TXT with the appropriate command depending on your version of DOS:

COMP OLDDOC.TXT NEWDOC.TXT

or

FC OLDDOC.TXT NEWDOC.TXT

Practice: Comparing Files

Display a directory of your CLASS diskette and copy the largest file you have listed. This will probably be the file called COMBINE.DOC. Name the copy COMBINE2.DOC. To compare the two files, key the appropriate command for your version of DOS:

- A:\>**COMP A:COMBINE.DOC A:COMBINE2.DOC** [Enter]

  or

- A:\>**FC A:COMBINE.DOC A:COMBINE2.DOC** [Enter]

The system compares the files. Did it report that an EOF (end of file) mark was not found? Did the files compare OK? Key N and press [Enter] at the prompt `Compare more files (Y/N)?`

Using COMP with Wildcards

Versions of FC prior to DOS 5.0 will not compare files selected with wildcards. If you are using MS-DOS 4.0 or earlier, read this activity for possible use in the future.

Groups of files selected with wildcards can be compared by using the COMP or FC commands with the various versions of MS/PC-DOS.

Practice: Comparing Files with Wildcards

You previously copied all files with the extension NEW and gave the copies your initials as the extension. If you were Mary Todd Lincoln, you would have used the command:

```
COPY *.NEW *.MTL
```

Now, with the CLASS diskette in drive A, compare the original files and the copies. Simply substitute COMP for COPY in the command line. (Use your own initials.)

- **A:\>COMP *.NEW *.[your initials]** [Enter]

Key N and press [Enter] to the prompt `Compare more files (Y/N)?`

RENAMING FILES WITH RENAME

In addition to renaming files when they are copied, DOS will rename an existing file without duplicating it if you use the internal command RENAME or REN. The command RENAME does exactly what it says: It changes the file name in the directory listing; it does not make a copy of the file with a new name or change the contents of the file in any way.

Since RENAME is one of the internal commands loaded into memory with COMMAND.COM, it is always available. You do not need access to the DOS external command files on a disk to use this command.

To rename a file, you can use RENAME or the shortened version, REN. Here is a typical rename command:

```
REN WHYDOS.NOW STUDYDOS.DOC
```

The original name of the file is given first, followed by a space and the new name. When you have renamed a file, the original name of the file no longer appears in the directory and can be used for another file. After the preceding command is entered, the name WHYDOS.NOW will no longer appear in the directory.

Apart from changing your mind about what a file should be called, you will find this command useful when you want to reserve and reuse a file name for a special purpose. For example, you might use COPY CON to write a daily list of things to do and name the file TO-DO. On rare occasions you may not complete all the tasks listed and need to save the list to refer to later. You could rename the file TO-DO.OLD and continue to use the name TO-DO for your current list.

Practice: Renaming Files

To rename the file called DOC1.TXT to FILE1.TXT on the CLASS diskette, key the following at the DOS prompt:

- C:\>**RENAME A:DOC1.TXT FILE1.TXT** [Enter]

or

- C:\>**REN A:DOC1.TXT FILE1.TXT** [Enter]

Note that the command can be shortened to REN or spelled out. Also note that you did *not* use a disk drive designator with the new file name. DOS will not allow such a disk drive designator with a new file. This is logical because the system is only changing the name of the file in the directory of the disk. REN does not rename a file and move it to a different disk (that would be copying the file).

When you press [Enter], the drive light comes on briefly. DOS is changing the directory. Then DOS displays the DOS prompt on the screen. To verify the absence of DOC1.TXT and the existence of FILE1.TXT, try to display the DOC1.TXT file using the TYPE command:

- C:\>**TYPE A:DOC1.TXT** [Enter]

DOS informs you that the file was not found. You can also display a directory of the disk in drive A to see that FILE1.TXT is there and that DOC1.TXT is not. Then display the contents of FILE1.TXT using the TYPE command. Do you recognize the contents of the old DOC1.TXT file?

Change the name back to DOC1.TXT by keying

- C:\>**REN A:FILE1.TXT DOC1.TXT** [Enter]

To rename a file on a disk that is not in the default drive, specify the drive designator before the original file name. If necessary, change to drive C and key the following command line to rename a file in drive A:

- C:\>**REN A:B-DOC.TXT RECORD.ABC** [Enter]

Examine the directory of the CLASS diskette in drive A to verify that the name change for the file was made.

Using Wildcards to Rename Groups of Files

Using wildcards, * and ?, it is possible to rename a group of files with one command entry. Check the directory of the disk in drive A for files with an extension TXT.

```
C:\>DIR A:*.TXT
```

The following command will give all files with the extension TXT the extension ABC:

```
C:\>REN A:*.TXT *.ABC
```

Practice: Renaming Files with Wildcards

To practice renaming all files with the extension TXT on the CLASS diskette, key:

- C:\>**REN A:*.TXT *.XYZ** [Enter]

  or

- A:\>**REN *.TXT *.XYZ** [Enter]

In response to this command, the system gives all files with the extension TXT the new extension XYZ. The filenames are not changed.

To verify the change, key:

- C:\>**DIR *.TXT** [Enter]

The system reports that no files are found. Now request a directory listing files with the XYZ extension. The system displays a list of the files that formerly had the TXT extension and that now have the XYZ extension.

DOS MOVE COMMAND

Introduced with DOS 6.0 is the capability to move a file from one location to another and delete the file from its original place. This command is similar to the COPY command in syntax. With the COPY command, a duplicate file is made; with the MOVE command, the file location is changed. The MOVE command will move files only to existing drives and directories. If you move a file and neglect to give a destination, the system will prompt with the following:

```
Required parameter missing
```

To move the file COMPUTER.TXT from drive A to drive C, you would key the following command:

```
MOVE C:COMPUTER.TXT A:
```

The screen displays the new location of the file with this message:

```
a:\computer.txt => c:\computer.txt [OK]
```

Note: This command requires special attention to the files in the target subdirectory. If there is a file with the same name as the file you are moving, the file in the target subdirectory will be overwritten without any warning message. However, MS-DOS 6.2 has added a protective warning which asks for confirmation before overwriting the file.

Practice: Moving Files

To move a file, DOC3.XYZ, from the CLASS diskette in drive A to drive C, key the following command:

- C:\>**MOVE A:DOC3.XYZ C:** [Enter]

The screen displays the new location of the file with this message:

```
a:\doc3.xyz => c:\doc3.xyz [OK]
```

Practice your newly-found skills by moving the file DOC3.XYZ back from the hard disk to the floppy disk in drive A. Also, move the files DOC2.TXT and NEWNAME.DOC copied to drive C earlier back to the CLASS diskette.

DOS EDITING KEYS

Have you found that when you key a command and DOS reports "Bad command or file name," it is tedious to rekey the whole command to correct one small mistake? The DOS editing keys can be used to make corrections in the most recent command line after you have entered the command. You can also use them to repeat commands.

When you key a command and press [Enter], the system retains the line you just keyed in an **input buffer**, a special segment of memory. The line in the buffer can be used as a **template** or pattern for a new line. You can copy the template line and edit it with the DOS editing keys. You can also repeat the line with the editing keys.

The DOS editing keys are the function keys F1, F2, F3, F4, F5, F6, and the Del, Esc, Ins, Backspace, and right and left arrow keys. They operate as follows on the last command line entered and stored in the input buffer.

F1	Copies one character from the last command or template line and displays the character on the screen.
F2	Copies and displays the command line up to the character keyed after pressing F2.
F3	Copies and displays the entire previous command line or the remainder of the line.
F4	Skips over (deletes) characters in the line to the character keyed after pressing F4.
F5	Copies the current command line to the template, but does not carry out the command.
F6	Places a Ctrl+Z (^Z) in the current command line.
Del	Removes one character from the command line template. The display does not change.
Esc	Cancels the line displayed; the template remains unchanged.
Ins	Inserts characters in the template and the display at the cursor. Press any editing key to stop inserting.
Backspace	Erases the last character from the display, but not from the template.
Right Arrow	Same as F1.
Left Arrow	Same as Backspace.

Practice: Using the DOS Editing Keys

Try using some of these timesaving keys. Key the following:

- **DIR /W** [Enter]

Suppose you had meant to use the Pause switch instead of the wide switch. It's easy to correct as follows:

Press the **F3** key to recall this most recent command.

Do you see the DIR /W command you keyed earlier? Backspace to erase the "W" and key "P." Of course, you would need to press [Enter] to execute the command.

A way to use the Ins key is to insert A: between the "R" and "/W" as follows:

> Press the **F1** key three times to display the DIR portion of the previous command. Next, press the **Ins** key to tell DOS to allow you to insert characters without erasing any of the previous command.
>
> Press the **[spacebar]** once, then key **A:**
>
> Finally, press the **F3** key to display the remainder of the previous command. The command line should be: DIR A:/P. Of course, you need to press [Enter] to execute the command.
>
> Another way you might have corrected this would have been to use the F1 key. Try the same scenario in reverse.
>
> Press the **F1** key.

Do you see the "D" on the command line? This is the "D" in the DIR A: /P you corrected in the last activity.

> Press **F1** enough times to display through DIR A:/. Then key the "W" which you would like to change back to. Of course, you would have to press [Enter] in order to execute the command.

Now try the F2 and Del keys using editing techniques as follows:

> Press the **F2** key and then **A** to display the characters up to, but not including, the **A**. Next, press the **Del** key three times to delete from the buffer the A:. Finally, press **F3** and [Enter].

These are good examples of the use of the DOS editing keys. You will work with more examples later.

Very nice work! Test your knowledge with the Self-Check Quiz and Assignment.

CHAPTER 4 SELF-CHECK QUIZ

True/False (Write T if the statement is true, F if the statement is false.)

_____ 1. When you rename a file with REN, the original file name is still in the directory.

_____ 2. When copying files with wildcards, the more common wildcard is the asterisk or "star."

_____ 3. It is possible to copy files by using wildcards in the command parameters.

_____ 4. Files can have the same name if copied to a different disk.

_____ 5. The command COMP compares two files byte by byte. After the first mismatch, the system will halt the COMP program.

Multiple Choice (Circle the correct answer.)

6. Which one of the following function keys copies up to the character keyed?

a. F1
b. F2
c. F4
d. F3

7. To combine the contents of FILE1.TXT with FILE2.TXT into a new file named FILE3.TXT, which command would you key?

a. COPY FILE1.TXT FILE2.TXT + FILE3.TXT
b. COPY FILE1 + FILE2.TXT FILE3.TXT
c. COPY FILE2.TXT + FILE1.TXT=FILE3.TXT
d. COPY FILE1.TXT + FILE2.TXT

8. To append the contents of FILE2.DOC to FILE1.TXT, which command would you key?

a. COPY FILE1.TXT FILE2.DOC
b. COPY FILE1.TXT + FILE2.DOC FILE3.DOC
c. COPY FILE1.TXT + FILE2.DOC
d. COPY FILE2.DOC + FILE1.TXT FILE3.DOC

9. To compare the file FILE1.TXT to FILE2.TXT, which command would you key?

   a. COMP FILE1.TXT FILE2.TXT
   b. COMP FILE1.TXT + FILE2.TXT
   c. COPY FILE2 COMP FILE1.TXT
   d. COMP FILE*.*

10. Which of the following is true about the DOS 6.2 MOVE command?

   a. The file will not overwrite an existing file of the same name in the destination directory without a warning.
   b. The file will remain in both locations.
   c. The file destination must not be a different disk drive.
   d. If you try to move a file and neglect to name the target location, the system will prompt with "Required parameter missing."

ASSIGNMENT

To practice using the commands in this chapter, complete the following steps. Check off each step when you understand its operation, or mark the step with a question mark if it is not clear. Write the commands or keystrokes in the space provided.

1. Print a directory of your CLASS diskette.

   _____

2. Create a text file describing the advantages of DOS. Name the file TEXT1.TXT.

   _____

3. Copy the TEXT1.TXT file to TEXT2.TXT on the same disk.

   _____

4. Compare TEXT1.TXT and TEXT2.TXT.

   _____

5. Copy all the files with names that begin with TEXT and that have a TXT extension and give them an ABC extension.

   _____

6. Copy all the files that begin with TEXT and that have an ABC extension and give them the same filename but an XYZ extension. Print a directory of the files.

   _____

7. Compare all the files with an ABC extension with those having an XYZ extension.

   _____

8. Rename the file TEXT1.XYZ to a file with an extension of your initials. Print a directory of the disk.

   _____

9. Move the file TEXT1.[your initials] to drive C. Print the screen or a directory of the CLASS disk.

   _____

10. Move the file TEXT1.[your initials] back to the CLASS disk in drive A. Print the screen or a directory of the disk.

CHAPTER 5

MANAGING FILES AND DISKS

OBJECTIVES

When you have completed the activities and assignment in this chapter, you will be able to:

1. Erase files with the DEL and ERASE commands.
2. Undelete files.
3. Unformat disks.
4. Use the DISKCOPY command to copy disks.
5. Use the COPY *.* and DISKCOPY commands at appropriate times.
6. Use the DISKCOMP command to compare disks.
7. Use the CHKDSK command to check disks.
8. Use the SCANDISK command to repair disk problems.
9. Use the DEFRAG command.

DELETING FILES

You have learned how to create, copy, rename, and move text files; now you will learn how to delete or erase them. The commands ERASE and DEL can be used to erase any file (text or program) that is not protected with read-only status or that is not hidden.

ERASE and DEL are internal DOS commands. You do not need access to the DOS external command files to use them. Simply enter the command followed by the name of the file you want to remove. ERASE and DEL work identically for erasing files.

Practice: Deleting Files

Use the DEL or ERASE command to delete DOC1.XYZ on your CLASS disk in drive A. At the DOS prompt, key

> CAUTION: Use ERASE and DEL carefully. Even though DOS 5.0 and later versions contain UNDELETE capabilities, you do not want to take unnecessary risks!

- **DEL A:DOC1.XYZ** [Enter]

  or

- **ERASE A:DOC1.XYZ** [Enter]

Note that the drive designator is optional if the file is on the disk in the default drive. However, if there is ever a time when you want to be as specific as possible, it is when you are erasing files.

The drive light will come on briefly when you press [Enter]. DOS will display a new prompt on the screen.

Request a directory listing for the deleted file:

- A:\>**DIR A:DOC1.XYZ** [Enter]

The display will show that the file is not found.

You can use wildcards with ERASE and DEL to delete or erase a group of files; however, you should make it a practice to always check wildcard entries by using them with the DIR command first. In this way, you can review the list of the files that will be erased when you enter the ERASE or DEL command line. First use:

- A:\>**DIR *.DOC** [Enter]

When you have checked all the files on your disk with any filename and the extension DOC and have decided that you want to erase all the files shown in the directory listing, key:

- A:\>**DEL *.DOC** [Enter]

You can use the UNDELETE command to retrieve these files, if necessary.

THE UNDELETE COMMAND

Prior to DOS 5.0, DOS could not reverse the changes made by DEL, ERASE, or FORMAT. Special utility programs could do this, but DOS did not have the capability. In version 5.0, the new command UNDELETE was introduced to

restore deleted directory entries. In the later versions, some improvements were made on these capabilities.

As you read in an earlier chapter, on every disk there is a special file called the file allocation table (FAT) that was created when the disk was formatted. When DOS deletes a file, it does two things: (1) it changes the FAT to mark the disk space used by the file as available, and (2) it changes the first character of the filename in the FAT to σ (the Greek letter Sigma). Therefore, the FAT is revised to show that the allocation units are available for storage of other files, and the file can no longer be accessed using its former name. However, by replacing σ with a first letter for the filename, the file can be undeleted until any part of the disk space occupied by the file is overwritten.

Practice: Undeleting Files with DOS 5.0

Since UNDELETE is an external command, you will need to set the search path to the external DOS command files. To have access to these on the hard disk, key:

- A:\>**PATH C:\DOS** [Enter]

First undelete a file you deleted earlier, DOC1.XYZ, as follows:

- A:\>**UNDELETE A:DOC1.XYZ** [Enter]

DOS responds with messages similar to the following:

```
Directory:  A:\
FILE SPECIFICATIONS:  DOC1.XYZ
Deletion-tracking file not found.
MS-DOS directory contains   1 deleted files.
Of those,    1 files may be recovered.

Using the MS-DOS directory method.
 ?OC1.XYZ    16   7-03-94   12:50p  ...A  Undelete
(Y/N) ?
```

At this point, key **Y** for yes to confirm that you want to undelete the file. DOS will then ask you to key the first character for DOC1.XYZ. Key the first character, **D**. When you do so and if you have not written any new files on the disk, DOS can restore the file.

To undelete all the files you have deleted, key

- A:\>**UNDELETE /ALL** [Enter]

DOS undeletes all files. However, the first character of the filenames which are undeleted has a # or a ? representing the first letter in the directory listing.

If you wish to supply the first character of each file as it is undeleted, key:

- A:\>**UNDELETE *.*** [Enter]

In this case, you have already undeleted all files. However, DOS will be able to undelete the files after you supply the first character of each filename.

USING MIRROR WITH DOS 5.0 TO UNDELETE FILES

The UNDELETE command is very useful. By using the DOS 5.0 external command MIRROR, you can set up a deletion-tracking file for the disk. Thereafter, when you delete a file, its name and related information are stored in the deletion-tracking file. UNDELETE, the command for restoring a deleted file, uses this information to restore deleted files.

When using UNDELETE, if you have installed the MIRROR.FIL on your disk drive, you will not be prompted to key the first character of the filename. If you have installed the MIRROR command before using the UNDELETE command, just follow the screen displays to successfully undelete the file.

Each disk must have its own deletion-tracking file. To tell DOS you want to track file deletions on your disks in drives A, B, and C, use the MIRROR command with the tracking switch /T and the drive name for each drive as in the following:

```
MIRROR /TA /TB /TC
```

This command creates the file MIRROR.FIL and a small system file named MIRRORSAV.FIL in the root directory of the disks located in the drives you have named. MIRROR.FIL contains a copy of the disk's FAT and directory. When you delete files from the disk, another system file named PCTRACKR.DEL is created. Thereafter, when you delete files from the disk, DOS adds the file information to the PCTRACKR.DEL file. The PCTRACKR.DEL file keeps track of only a limited number of files. When the tracking file is full, the names of the files deleted first are removed to make room for new files. The size of the PCTRACKR.DEL file is relative to the size of the disk: 25 files for a 360K drive and up to 303 files for a drive with more than 32MB.

Practice: Using Mirror to Undelete Files (DOS 5.0)

If you are using PC-DOS 6.1 or MS-DOS 6.0/6.2, skip to the undelete section using those versions.

To use the MIRROR command to create MIRROR.FIL and to instruct DOS to perform deletion tracking on drives A, B, and C, key the following. (Note that there is no colon after the drive name.)

- C:\>**MIRROR /TA /TB /TC** [Enter]

MIRROR informs you that it is creating an image of the system area for drives A, B, and C and that it is installing the deletion-tracking software.

Display the directory and look for MIRROR.FIL. Do you find it in your default directory?

- C:\>**DIR MIRROR.FIL** [Enter]

Insert your CLASS disk in drive A and display the directory. Key the following command line to delete the files from the disk:

- C:\>**DEL A:*.*** [Enter]

Did it take longer for the DOS prompt to reappear this time? DOS was creating PCTRACKR.DEL.

Now use the UNDELETE command to restore COMBINE.DOC.

- C:\>**UNDELETE A:COMBINE.DOC** [Enter]

DOS responds with a message similar to:

```
Directory:  A:\

File Specifications:  COMBINE.DOC

   Deletion-tracking file contains       1 deleted files
   Of those            1 files have all clusters
   available,
                  0 files have some clusters available,
                  0 files have no clusters available.

   MS-DOS directory contains        1 deleted files.
   Of those,    1 files may be recovered.

Using the deletion-tracking file.

   COMBINE.DOC  715 02-07-94  10.29p...A  Deleted:
```

```
   5-5-94 11:25a
All of the clusters for this file are available.
Undelete (Y/N)?
```

When you key **Y** for Yes to restore the file, DOS reports:

```
   File successfully undeleted.
```

When UNDELETE can use the deletion-tracking file, there is no need to supply the first letter of the file name. Just key **Y** for each file.

Use the UNDELETE switch **/LIST** to display the files available for deletion.

- C:\>**UNDELETE A: /LIST** [Enter]

Now retrieve all of the files on your CLASS diskette. Can you remember how?

- C:\>**UNDELETE A:*.*** [Enter]

To retrieve all the files without confirmation of each file, key:

- C:\>**UNDELETE A:/ALL** [Enter]

Run a directory to assure yourself that the files on your diskette have been completely restored. Use the TYPE command to confirm that their contents are unchanged.

UNDELETE USING MS DOS 6.0/6.2

DOS 6.0 and 6.2 provide three levels of protection for retrieving files previously deleted. The most efficient level is called Sentry; the second level is deletion tracking like that available in DOS 5.0; and the third level is DOS protection, which is also offered in DOS 5.0.

Since Sentry is the highest level of protection, that is the technique we will use. The Sentry level, introduced in MS-DOS 6.0, creates a special directory where all deleted files are placed. This directory is a hidden directory and is created as soon as any files are deleted from the drive. The deleted files are available until they are replaced by newer deleted files, or because the time frame for saving the files has expired. These are options contained in a file named UNDELETE.INI that can be changed if the defaults do not suit your needs.

To provide the Sentry protection to the disk in drive A, key

C:\>UNDELETE /SA

Note that this command also protects drive C. You may include protection for other drives on the same command line. For example, to load the Sentry protection for both drive A and drive C at the same time, key:

C:\>UNDELETE /SA /SC

To view the status of the protection of the drives, key:

C:\>UNDELETE /STATUS

The status of Sentry displays. To see the number of files available for UNDELETE, key:

C:\>UNDELETE /LIST

Practice: Undeleting Files Using MS-DOS 6.0/6.2

Display a directory of the DOS subdirectory on drive C to see the UNDELETE.INI file.

- C:\DOS>**DIR UNDELETE.INI** [Enter]

Unless you have installed UNDELETE or deleted a file using the UNDELETE /SA /SB /SC command, you should not find the file.

Next install the first level of protection, Sentry, for drives A, B, and C by keying:

- C:\>**UNDELETE /SA /SB /SC** [Enter]

The UNDELETE.INI file has been created and should now be listed in the directory. To check, key:

- C:\DOS>**DIR UNDELETE.INI** [Enter]

Insert your CLASS disk in drive A and key the following line to delete the DOC1.NEW file from the disk:

- C:\>**DEL A:DOC1.NEW** [Enter]

To see if the file has been deleted, check the directory of drive A.

- C:\>**DIR A:DOC1.NEW** [Enter]

You should receive a message stating that the file is not found.

To retrieve the file DOC1.NEW, key

- C:\>**UNDELETE A:DOC1.NEW** [Enter]

Again, if you wish to check the directory for the file, key:

- C:\>**DIR A:DOC1.NEW** [Enter]

This time the file should be listed.

View the status of the protection of each drive by keying:

- C:\>**UNDELETE /STATUS** [Enter]

To delete all files from the CLASS disk in drive A, key the following:

- C:\>**DEL A:*.*** [Enter]

To see the number of files that are available now to be undeleted, key:

- C:\>**UNDELETE A:/LIST** [Enter]

To finish this practice session, undelete all the files and run a directory of the disk in drive A. Use the following commands:

- C:\>**UNDELETE A:*.*** [Enter]

- C:\>**DIR A:** [Enter]

UNDELETE USING IBM PC-DOS 6.1

IBM PC-DOS 6.1 provides three levels of protection for retrieving deleted files. However, the Sentry level, which creates a special directory where all deleted files are placed, is the one we will use as the preferred level. When deleted, the files are available until they are replaced by newer deleted files.

To provide the Sentry protection to the disk in drive A, key

C:\>UNDELETE A: /S

Note that this command can also be used to protect drive C as follows:

C:\>UNDELETE C: /S

To view the status of the protection of the drives, key:

To see the name, date, time, size, and the condition of the file for UNDELETE, key:

C:\>UNDELETE /LIST

Practice: Undeleting Files Using IBM PC-DOS 6.1

Insert your CLASS disk in drive A and install the Sentry level of protection for drive A by keying:

- **C:\>UNDELETE A: /S** [Enter]

Key the following command to delete the DOC1.NEW file from the disk:

- **C:\>DEL A:DOC1.NEW** [Enter]

To see if the file has been deleted, check the directory of drive A.

- **C:\>DIR A:DOC1.NEW** [Enter]

You should receive a message stating that the file is not found.

To retrieve the file DOC1.NEW, key

- **C:\>UNDELETE A:DOC1.NEW** [Enter]

You will need to provide the first character of the filename, and the file will be undeleted. Again, if you wish to check the directory for the file, key:

- **C:\>DIR A:DOC1.NEW** [Enter]

This time the file should be listed.

To delete all files from the CLASS disk in drive A, key the following:

- **C:\>DEL A:*.*** [Enter]

To see the number of files that are available now to be undeleted, key:

- **C:\>UNDELETE /LIST** [Enter]

To finish this practice session, undelete all the files and run a directory of the disk in drive A. Use the following commands:

- **C:\>UNDELETE A:*.*** [Enter]
- **C:\>DIR A:** [Enter]

UNFORMATTING DISKETTES

If you reformat a disk accidentally and are using DOS 5.0 or later, you may be able to use the UNFORMAT command to restore the disk.

By default, the DOS 5.0 and later FORMAT command performs a "safe" format. As explained in Chapter 3, if you haven't written files to a disk since accidentally formatting it, then you will likely be able to unformat the disk. The command to unformat a disk in drive A is:

UNFORMAT A:

Practice: Unformatting a Disk

Let's restore a disk. First, format the CLASS disk, which you should have in the A drive. Key:

- **FORMAT A:** [Enter]

Label the disk again with the name, CLASS.

Display a directory to assure yourself that no files are listed and that the disk has, indeed, been formatted. Then use the UNFORMAT command as follows:

- **UNFORMAT A:** [Enter]

After asking for the disk to be inserted, DOS will deliver warning messages about the disk's being restored. The MIRROR image is located and the date and time it was last used is noted. You are then asked to confirm that you wish the disk to be updated and the unformat process begins.

When the unformat process is complete, display a directory to see that all files have been recovered and that the disk is just as it was before you formatted it.

COPYING DISKS

Often you will need to copy disks. They may be copied sector by sector using the DISKCOPY command, or they may be copied file by file using the COPY command.

THE DISKCOPY COMMAND

As you use the COPY command to duplicate files, you use the command DISKCOPY to duplicate floppy diskettes so that if the original disk is lost or damaged, you can use the copy in its place. Another application is to make several copies of a disk containing data or program files you have created and want to distribute to others. DISKCOPY will copy the contents of floppy diskettes onto diskettes *of the same size and capacity*. You cannot use DISKCOPY to copy a 360K floppy onto a 1.2MB floppy or to copy a 3 1/2-inch floppy onto a 5 1/4-inch floppy or a hard disk. If you specify an incompatible disk with the DISKCOPY command, DOS will display an error message.

The system refers to the disk from which you are copying as the **source**, master, or original. The disk that will hold the copy may be called the **target**, backup, or destination disk. Make sure that the source and target disks are distinctly labeled. Write-protect the source disk. Then, if the source is accidentally used for the target, DOS will alert you that the wrong disk is inserted because it cannot write to the disk.

> **CAUTION:** DISKCOPY formats (erases) as it copies files onto a disk. Use this command only when you do not want to save any existing files on the target disk that will hold the disk copy.
>
> Do not use Ctrl-C or Ctrl-Break to interrupt the DISKCOPY process when the target drive light is on.

DISKCOPY is one of the few DOS commands other than FORMAT that have formatting capabilities. When you attempt to make a disk copy on an unformatted disk, the system detects this and DISKCOPY formats the target disk during the copy operation. As a result, disk copying takes longer with unformatted disks. DOS will display a message similar to "Formatting while copying."

When formatting a disk with the FORMAT command, the system tells you if there are any defective areas (bad sectors) on the disk and locks out the bad sectors so that they will not be used. When a disk is formatted during the DISKCOPY process, if there are bad sectors the copy may or may not be usable, depending on what data were affected. For this reason, it is a good practice to format disks with the FORMAT command before using them for backup copies and to use only disks with no bad sectors when you make copies with DISKCOPY.

The syntax for copying diskettes is as follows:

C:\>DISKCOPY A: A: *or* C:\>DISKCOPY A: B:

Use the first command if you are copying with only one drive; use the second command for copying diskettes from drive A drive to drive B.

Because most computers today have different size and/or capacity disk drives, and the DISKCOPY command requires disk drives of the same size and/or capacity, you will most likely copy diskettes using one floppy drive. This is, however, untrue with MS-DOS 6.2. With this new version and the appropriate switch, you can work with the hard drive as interim storage using the DISKCOPY command.

Practice: Copying a Disk Using DISKCOPY

Using One Floppy Drive with DOS Files in Drive C

Because DISKCOPY is an external command, you must have a search path set to the drive or directory where your DOS files are located. Check this with the PATH command below, and reenter the search path if it is not set correctly.

- C:\>**PATH** [Enter]

If you do not have a PATH, enter:

- C:\>**PATH C:\DOS** [Enter]

Now make a copy of the CLASS disk. It will be the *source* disk. You will need to use a compatible disk with no useful files on it (LAB FILES) for the *target* disk. Place the *source* disk in drive A and key the following:

- C:\>**DISKCOPY A: A:** [Enter]

After DOS has read the DISKCOPY.COM file into memory from the DOS directory of the hard disk in drive C, the system will prompt:

```
Insert SOURCE diskette in drive A:
Press any key to continue . . .
```

The source disk (CLASS) is in drive A. Strike any key. The in-use light on the front of the disk drive will come on as the system reads the contents of the source diskette into the computer's random access memory. DOS will display a message such as:

```
Copying 80 tracks
9 sectors per track, 2 side(s)
```

When the disk contents have been read into memory, the system will prompt:

```
Insert TARGET diskette in drive A:
Press any key to continue . . .
```

Remove the source disk and insert the target disk (LAB FILES). Press any key to begin the writing process. Again, the in-use light on the disk drive will come on. If the target disk has not been formatted, DOS will inform you that it is being formatted during the copy process. After the information has been written from memory, DOS will prompt you for the next step.

> **Note:** If the source diskette has more information that the computer's memory can hold at one time, the system will instruct you to exchange the source and target diskettes more than once to copy the entire disk.

When copying is complete, DOS will ask:

```
Copy another diskette  (Y/N)?
```

If you wish to copy the same disk or another disk, key **Y** for yes; if not, key **N** for no.

Using Two Compatible Floppy Drives with DOS Files in Drive C

With the CLASS disk in drive A, key at the prompt:

- C:\>**DISKCOPY A: B:** [Enter]

After DOS loads the DISKCOPY command file into memory from the DOS files on drive C, the system prompts:

```
Insert SOURCE diskette in drive A:
Insert TARGET diskette in drive B:
Press any key to continue . . .
```

The source disk (CLASS) is in drive A. Place the compatible target disk (LAB FILES) in drive B. Remember that the information on the disk in drive B will be erased. When you are ready, press any key.

The in-use light on drive A comes on as the system reads the contents of the source diskette into the computer's memory. The drive B light comes on as the system writes the contents of memory onto the target diskette.

While the disk is being copied, you will see a message similar to:

```
Copying 80 tracks
9 sectors per track, 2 side(s)
Formatting while copying
```
(unless the disk was previously formatted)

When the copy is complete, you will see a message similar to:

```
Copy another diskette (Y/N)?
```

If you wish to copy the same disk or another disk, key **Y** for yes; if not, key **N** for no.

> **Note:** Backup copies made with DISKCOPY provide reliable protection against loss *providing* you store the originals and backups in separate locations. If both copies are subject to the same calamity, having a duplicate will be of no help. Plan your disk storage with this in mind.

COMPARING DISKS USING THE DISKCOMP COMMAND

The DISKCOMP command enables you to compare entire disks, as well as to determine if a disk was correctly copied. Use this command when you want to be sure that two disks match exactly—track by track and sector by sector. Because DISKCOMP compares two disks exactly, it is not possible to compare

a floppy disk to a hard disk or to compare floppy disks of different capacities or sizes.

DISKCOMP is useful if you need to make several copies of the same disk and want to verify that they are exact duplicates. Use each copy as the source disk for the next copy, then compare the last copy with the first disk copied. If these two disks compare ok, all the copies must be identical.

DISKCOMP is an external command. You must have a search path set to the drive or directory where your DOS files are located.

The syntax for comparing diskettes is as follows:

C:\>DISKCOMP A: A: *or* C:\>DISKCOMP A: B:

Use the first command if you are comparing using only one drive; use the second command for comparing diskettes in drive A and drive B.

Practice: Comparing Disks

Using One Floppy Drive with DOS Files on Drive C

Check to see if the search path is set and, if not, set a path to the external command files with a command such as PATH C:\DOS. To begin the disk comparison process, key:

- C:\>**DISKCOMP A: A:** [Enter]

After DISKCOMP.COM is loaded into memory, the system prompts you to insert the first diskette and then to exchange it for the second diskette. Follow the prompts to insert the source (CLASS) and target (LAB FILES) disks. If the comparison is successful, the system will report:

```
Compare ok
Compare another diskette (Y/N)?
```

If you do not have another disk to compare, key **N**.

Depending on the amount of memory available in the computer, you may need to insert the diskettes more than once. DOS will prompt you.

If the disks do not compare, the screen display will report the locations where errors are found in a message similar to the following:

```
Compare error(s) on side 0, track 1
```

Press Ctrl-C or Ctrl-Break to interrupt the compare process. You will need to use DISKCOPY again to obtain an exact copy. Then use DISKCOMP to compare again.

CAUTION: Do not use Ctrl-C or Ctrl-Break to interrupt the disk comparing process when the target drive light is on.

Using Two Floppy Drives with DOS Files on Drive C

Key at the A:\> or C:\> prompt:

- A:\>**DISKCOMP A: B:** [Enter]

  or
- C:\>**DISKCOMP A: B:** [Enter]

After DOS loads the DISKCOMP command into memory, the system prompts you to insert the first diskette and the second diskette. Follow the prompts, using the CLASS disk in drive A and the LAB FILES disk, the copy you just made of the CLASS disk, in drive B. Wait until the system reports the following:

```
Compare ok
Compare another diskette  (Y/N)?
```

If you do not have another disk to compare, key **N**.

DISKCOPY also copies the volume name of the source diskette to the target diskette. It is necessary to change the name of the target diskette to LAB FILES using the LABEL command as follows:

- **LABEL B: LAB FILES** [Enter]

COPY *.* COMPARED TO DISKCOPY

Because you may want to copy from one disk to another without erasing the contents of the target disk, it is important to know an alternative to the DISKCOPY method. One is COPY *.* *or* COPY "Star-dot-star," as this command is usually stated. This process is also called making a **file-by-file** copy of a disk. If you were to key the following command, you would copy

all the files from the source disk in drive A to drive C without erasing the existing files on drive C, the target disk.

```
A:\>COPY *.* C:
```

One important caution about this command is that if there are any files on drive C with names identical to files on the source disk in drive A, the files on drive C will be overwritten. However, DOS 6.2 has added a prompt that advises of a file of the same name and gives you the opportunity to overwrite or not.

Another consideration for using COPY *.* is that there must be enough space remaining on drive C to hold all the files on the drive A disk.

With DISKCOPY, the source and target disks must be identical in size and capacity. With COPY *.*, the source and target disks can be different sizes and capacities. For example, COPY *.* can be used to copy from a 5 1/4-inch 360K floppy diskette to a 3 1/2-inch 144MB floppy diskette.

Another circumstance where COPY *.* may be preferable to DISKCOPY is when you are making backup copies of data files. As data files are created, erased, revised, and added to a disk, the system may write portions of the same file in widely scattered locations on the disk. DOS must keep track of these locations and find the file segments each time you work with the file. Working with **fragmented files,** or "non-contiguous" files, slows down the system. One way to correct this condition is to back up the files onto a blank, formatted disk with COPY *.*. When you use the COPY command to back up files onto a fresh diskette, the system can copy the fragments of each file into a contiguous, unfragmented file. On the other hand, DISKCOPY makes an *exact* copy of the source disk and will reproduce any fragmentation.

Certain files cannot be copied with the COPY command. For example, hidden files, such as the hidden system files (IBMBIO.COM and IBMDOS.COM or MSDOS.SYS and IO.SYS) and the files MIRRORSAV.FIL and PCTRACKR.DEL cannot be duplicated with COPY. Software manufacturers may include hidden files on their disks. In cases like this, DISKCOPY must be used to obtain a complete backup or working copy.

Practice: Using COPY *.* to Copy Disks

Use the CLASS diskette as the source and the LAB FILES diskette as the target with the following command:

- A:\>**COPY A:*.* B:** [Enter]

Remember that DOS will use the A drive as both A and B if you have only one drive. However, you will have to exchange diskettes for each file. If you have two floppy disk drives even if they are different size and/or capacity, files can be copied between them. Each file will be listed as it is copied. If you would like to be sure that the files have copied correctly, use the COMP *.* command to compare files.

CHECKING A DISK WITH CHKDSK

To receive a status report on the amount of space left on your disk and in your computer's memory, use the CHKDSK command. This versatile command reports the total disk space; the amount of space used for hidden files, directories, and user files; the space occupied by defective sectors on the disk, if any; and the space available for use. CHKDSK also reports how much internal memory is available in the computer and how much memory is not in use.

CHKDSK has two switches that add to its usefulness. The /F switch tells the system to fix any errors found in the checking process. The /V switch causes all files and their paths on the disk to be listed, including hidden files.

You can also use CHKDSK to check for fragmented, or non-contiguous, files on a data disk. Earlier we described how portions of data files can become scattered over the disk as the files are revised and enlarged. The system must collect the fragments of these files from the disk and reassemble them in memory when you use the files. This increases the file access time and slows system performance. As you will see in the next activity, you can determine how many files are fragmented by using the CHKDSK command and correct this condition by copying the files to a newly formatted disk.

Practice: Checking the Diskette

CHKDSK is an external command; be sure your search path is set correctly. To check the disk in drive A, key

- **C:\>CHKDSK A:** [Enter]

Your screen will display information similar to the following:

```
Volume CLASS created (date and time)
Volume Serial Number is (number)

362494 bytes total disk space
71680 in 2 hidden files
54272 bytes in (26) user files
236544 bytes available on disk
```

```
1024 bytes in each allocation unit
354 total allocation units on disk
231 available allocation units on disk

655360 bytes total memory
586848 bytes free
```

If you are using DOS 6.2, you will also see the following:

```
Instead of using CHKDSK, try using SCANDISK. SCANDISK
can reliably detect and fix a much wider range of disk
problems.  For more information, type HELP SCANDISK
from the command prompt.
```

The first lines identify the disk being checked. The serial number line appears in version 4.0 and later.

The next block of lines refers to the disk space. The number of bytes of total disk space indicates whether the disk is a single-sided (183,296 bytes), double-sided (362,494 bytes), or high-capacity 5 1/4-inch (1,213,952 bytes) or a 3 1/2-inch double-sided (730,112) or high capacity (1,457,664 bytes) floppy disk.

The next block of lines in the example is reported only by DOS 4.0 and later versions. It tells how many allocation units are available on the disk and how large they are. An allocation unit is the smallest amount of space a single file can occupy. Files will use space in multiples of this size.

The last block of lines in the example refers to the computer's RAM. The amount of total memory and the amount of memory not in use are shown.

Now use the /V switch with the CHKDSK command to check for a listing of all files including the hidden files. Use the Pause key to pause the display.

- A:\>**CHKDSK /V** [Enter]

Does this command display the three hidden files on the disk? Look for the two system files IBMBIO.COM and IBMDOS.COM or their MS-DOS equivalents, MSDOS.SYS and IO.SYS in the list.

The /F switch is used with CHKDSK to find and fix "lost" file sectors on the disk. It is more likely to be helpful if you are checking a hard disk, which may contain thousands of files and may have errors in the file location information that can be corrected with this switch. If lost sectors are reported by CHKDSK, use the /F switch to combine the sectors into a file. Try the following:

- C:\>**CHKDSK /F A:** [Enter]

If there are lost sectors, you should answer yes to combine them into a file.

To check for any fragmented, or non-contiguous, files on the disk in drive A, key:

- C:\>**CHKDSK A:*.*** [Enter]

You will see a listing of the fragmented files containing non-contiguous blocks. If there are some, the system will display a report like this:

```
A:\><Filename>
Contains 2 non-contiguous blocks.
```

You can also use CHKDSK to check individual files or groups of files for fragmentation by keying the file name after the CHKDSK command at the DOS prompt.

- C:\>**CHKDSK A:DOC1.NEW** [Enter]

  or

- C:\>**CHKDSK A:*.DOC** [Enter]

to check for a group of files with the extension DOC.

If a data disk contains a large number of fragmented files with non-contiguous blocks, it is good practice to copy its contents onto an empty, formatted disk with COPY *.*. Remember, if you want the disk copy to be self-booting, format it with the /S switch to copy the hidden system files to the disk. COPY *.* will not copy hidden files.

SCANDISK (DOS 6.2)

SCANDISK is a disk analysis and repair utility that was introduced with MS-DOS 6.2. It checks for errors on the disk and provides an opportunity to correct problems that it finds.

SCANDISK will test the file structure on the drive. If there are lost clusters or sectors, SCANDISK will put them into files with a CHK extension similar to CHKDSK/F. You then can check the file using the TYPE command to see if the file is an important one. If it is not, you can delete the file.

The syntax for SCANDISK is:

C:\>SCANDISK

Practice: SCANDISK Using MS-DOS 6.2

If you have DOS 6.2 and would like to check and fix the current drive, key the following:

- C:\>**SCANDISK** [Enter]

THE DEFRAG COMMAND IN PC-DOS 6.1 and MS-DOS 6.2

You will be surprised how quickly you fill your disk with tens, if not hundreds, of files. As these files are written to your disk over time, you will have files that are fragmented. Fragmented files are broken into sections that are stored in different locations on the disk. Fragmentation doesn't affect the data, but it can slow your system down noticeably. If you have checked your disk and know that you have fragmented files, you can use the PC-DOS 6.1 and MS-DOS 6.2 command, DEFRAG.

The syntax is DEFRAG A:

Practice: Defragmenting a Disk

It is important to be sure you are at the DOS prompt and not in any programs, including Windows or the Shell, a graphic representation of DOS. Key the following:

- C:\>**DEFRAG A:** [Enter]

When you issue the DEFRAG command, the system displays a screen showing all the drive options. Select the drive that holds the disk you want to defrag and press [Enter]. After the drive selection, the system tells you whether or not the disk needs to be optimized. If it has no fragmentation, the system tells you that defragmentation is not necessary. If there is fragmentation, the system will mark a recommended optimization method. You will have the option to use the Optimize or Configure setting. The system will show its preference. While the files are being defragmented, you can see the progress on the display screen.

Nice work on Chapter 5! Test your knowledge and skills with the Self-Check Quiz and Assignment.

CHAPTER 5 SELF-CHECK QUIZ

True/False (Write T if the statement is true, F if the statement is false.)

_____ 1. The PC-DOS 6.1 and MS-DOS 6.2 DEFRAG command will make non-contiguous files contiguous.

_____ 2. To see how much space is still available on your disk use the CHKDSK command.

_____ 3. DEL and ERASE both erase files and can be used with wildcards.

_____ 4. The COMP command compares two disks.

_____ 5. It was possible to use the UNDELETE command prior to DOS version 5.0.

Multiple Choice (Circle the correct answer.)

6. Which of the following is an advantage of using COPY *.* instead of DISKCOPY?

   a. Files can be copied without erasing the target disk.
   b. All files, including hidden files, can be copied with COPY.
   c. The target disk does not need to be formatted.
   d. All existing files on the target disk are erased.

7. Which two of the following can be true concerning the DISKCOPY command?

   a. Disks of different sizes can be copied using the DISKCOPY command.
   b. DISKCOPY formats as it copies.
   c. DISKCOPY will not copy the label of a disk as it copies the files.
   d. DISKCOPY does not erase the target disk.

8. Which command would you use to back up all files on the disk in drive A and to add the backup files to a new disk in drive B?

   a. COPY A: A:
   b. DISKCOPY A:*.* B:
   c. COPY A:*.* B:
   d. DISKCOPY A: A:

9. Which command will erase only the files with the extension MOS?

   a. ERASE *.MOS
   b. DEL *.M?
   c. DELETE *.MOS
   d. DEL *.*

10. SCANDISK, like DEFRAG, looks similar to a(n) _____ on the screen.

a. directory
b. utility
c. subdirectory
d. attribute

ASSIGNMENT

To practice using the commands in this chapter, complete the following steps. Check off each step when you understand its operation, or mark the step with a question mark if it is not clear. Write the commands or keystrokes in the space provided.

1. Copy all files with the XYZ extension from your CLASS diskette to a new diskette named LAB FILES. Print a directory of the LAB FILES diskette. Format the LAB FILES diskette. Print a directory of the LAB FILES diskette.

---

2. Unformat the LAB FILES diskette. Print a directory of the LAB FILES diskette.

---

3. Set up Sentry or deletion tracking protection (depending on your version of DOS) for the LAB FILES diskette. Delete all the files with the XYZ extension. Print the directory of the LAB FILES diskette.

---

4. Undelete the files with the XYZ extension. Print the directory of the LAB FILES diskette.

---

5. Diskcopy the CLASS diskette onto the LAB FILES diskette. Print a wide directory of both diskettes.

---

6. Compare the CLASS diskette with the LAB FILES diskette. Print a wide directory of both diskettes.

---

7. Check the CLASS diskette for fragmentation using the CHKDSK command.

---

8. Use the DEL command to erase all files from the CLASS diskette in one operation. Print a directory of the diskette.

---

9. Use the COPY command that will copy all the files from the LAB FILES diskette to the CLASS diskette using wildcards.

---

10. If you have PC-DOS 6.1 or MS-DOS 6.2, check the CLASS diskette to look for fragmentation using the DEFRAG command. Print a copy of the results.

CHAPTER 6

CREATING AND USING SUBDIRECTORIES

OBJECTIVES

When you have completed the activities and assignment in this chapter, you will be able to:

1. Describe the purpose of subdirectories.
2. Create subdirectories.
3. Specify paths to directories.
4. Change the current directory and display its path and name.
5. Create and store files in directories.
6. Print subdirectory listings.
7. Locate a file in any directory of a disk.

INTRODUCTION TO SUBDIRECTORIES

When a floppy or hard disk is formatted, a file directory structure is established that allows for the description of each file on the disk. The description includes the file name, size, and time and date of creation or last revision, as well as information about the file's location on the disk and other attributes. The directory created during formatting is called the main or **root directory**. The size of the root directory file is fixed. The root directory can hold a limited number of file entries because of its fixed size: 64 entries on a single-sided 180KB diskette, 112 on a 360KB or 720KB double-sided diskette, 224 on a 1.2MB or 1.44MB high-density disk, and 512 or more on a hard disk. For this reason, you can have bytes free on a disk but cannot add files to the disk because adding a file would exceed the maximum number allowed in the root directory.

You can increase the number of files a floppy diskette or hard disk will store by creating **subdirectories** to hold file information. Subdirectories can also make your disk and file management more effective, by listing files in categories you create. This is especially helpful on hard disk systems, which

often contain thousands of files. Since subdirectories can be used to organize the files into different groups, you may want to use subdirectories to segregate different application software files and data files. For example, you could place all DOS files in one subdirectory, as was done in the installation of DOS 5.0 and 6.x versions. You might create a subdirectory for word processing program files and one for database program files. In addition, you might want to create directories by subject area for your data files. For instance, documents created for different projects or classes could be separated into different subdirectories for easy access.

A main or root directory exists on every floppy and hard disk. Subdirectories and then subdirectories of each subdirectory can be added to the directory structure. This file structure is often called a **tree-structured file system,** since the structure resembles an upside-down tree, with the main or root directory branching to several levels of subdirectories as shown in Figure 6-1. A more appropriate analogy might be a family tree. You can also think of subdirectories as filing cabinet drawers each containing the files you place there.

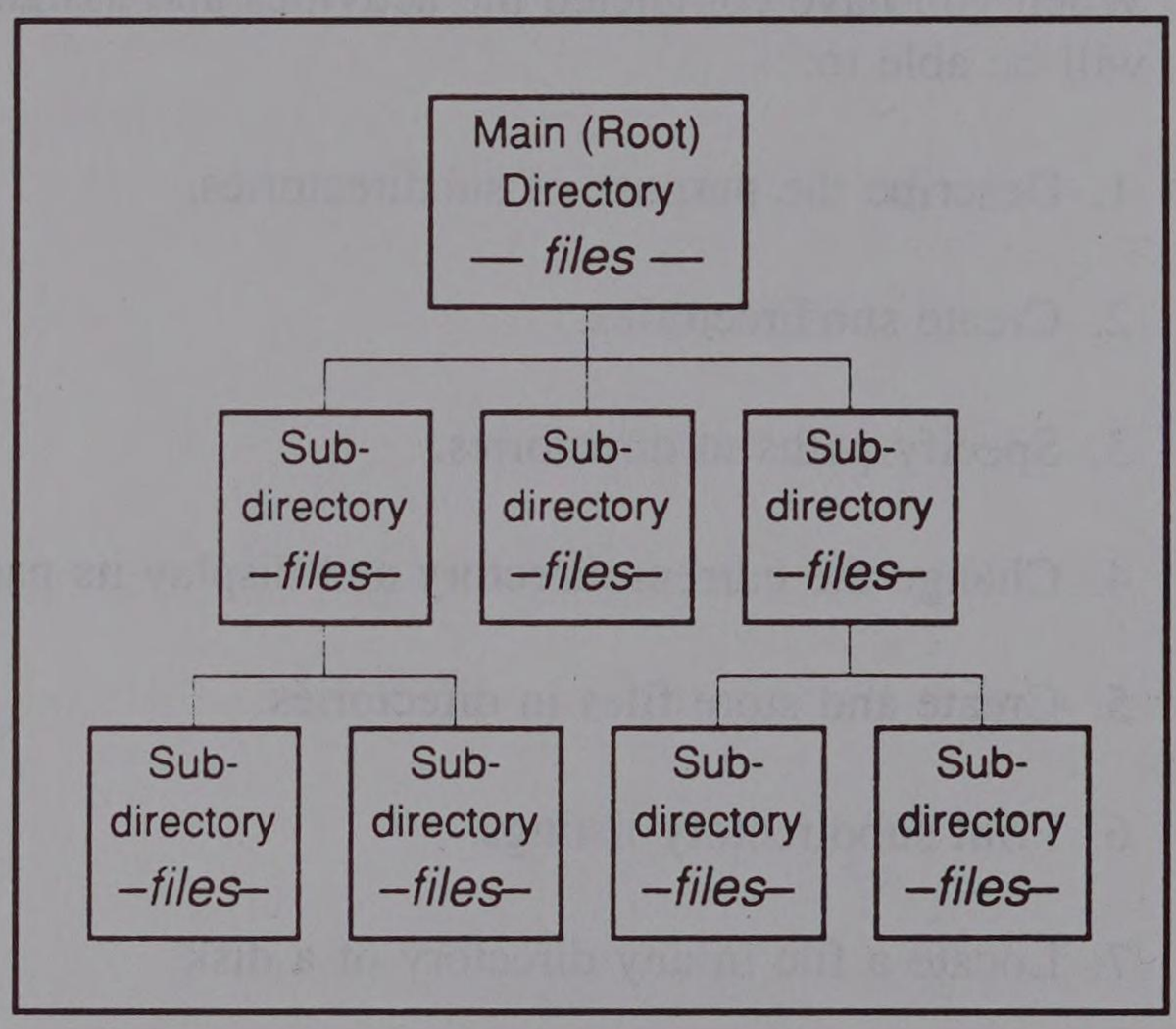

Figure 6-1. A tree-structured file system branches from the root directory.

Of course, it is important that you set up a directory structure and name your directories in a logical way in order to work with them easily. Keep the branching as simple as possible so that you won't have to search through a complex tree structure to access the files you need.

Subdirectory names follow the same naming conventions as the names of other files—a maximum of eight characters with no spaces. Although you can add an extension to a subdirectory name, this is seldom done. Instead, it is a good practice to keep subdirectory names as short as possible. As you will see, short directory names can save many keystrokes.

When you use DIR to display a directory listing of a subdirectory, you will see the same information about the files in that subdirectory as you see displayed for the files in the root directory, and in the same format: the filename and extension, size, and date and time of creation or last revision. A subdirectory

is actually a file, like the root directory, that contains information for each file listed in the subdirectory. However, the files storing subdirectory information are not limited in size like the file storing the root directory information. Therefore, there is no limit to the number of entries that can be placed in a subdirectory. In actual practice, you may want to limit the number to what the command DIR /W will display on a single screen.

You cannot use the command COPY or DEL to create or remove a subdirectory. In this chapter and the next, you will learn DOS commands for creating and removing subdirectories.

To access files in subdirectories, you will sometimes need to change the default directory from the root to one of the subdirectories. You can think of this as "moving around" in the tree structure.

CREATING A SUBDIRECTORY STRUCTURE

A common reason for setting up subdirectories on a hard disk or floppy diskettes is to segregate software or data files. For example, you may set up one subdirectory for word processing, one for spreadsheets, and one for desktop publishing.

The syntax for creating these subdirectories is MD (directory name). To create a subdirectory named WP, key the following from the DOS prompt:

```
A:\>MD WP
```

Practice: Setting Up a Subdirectory Structure

Even though you will do this exercise on a floppy diskette, the principles apply identically to a hard disk. To prepare for creating a tree-structured diskette, reformat the LAB FILES diskette with the system. Name it TREE DISK.

Create three subdirectories in which you could install software. One directory is for word processing software; name this directory WPRO. The second directory is for spreadsheet software used for accounting work; name this directory ACCT. The third directory is for desktop publishing software; name this directory DTP. Thus your tree structure will have three subdirectories named WPRO, ACCT, and DTP. These directories will all be subdirectories of the root directory. For now, the data files or documents relating to each application can be stored in the application's directory. The tree can be visualized as shown in Figure 6-2.

To create the three directories shown in Figure 6-2, key:

- A:\>**MD WPRO** [Enter]
- A:\>**MD ACCT** [Enter]
- A:\>**MD DTP** [Enter]

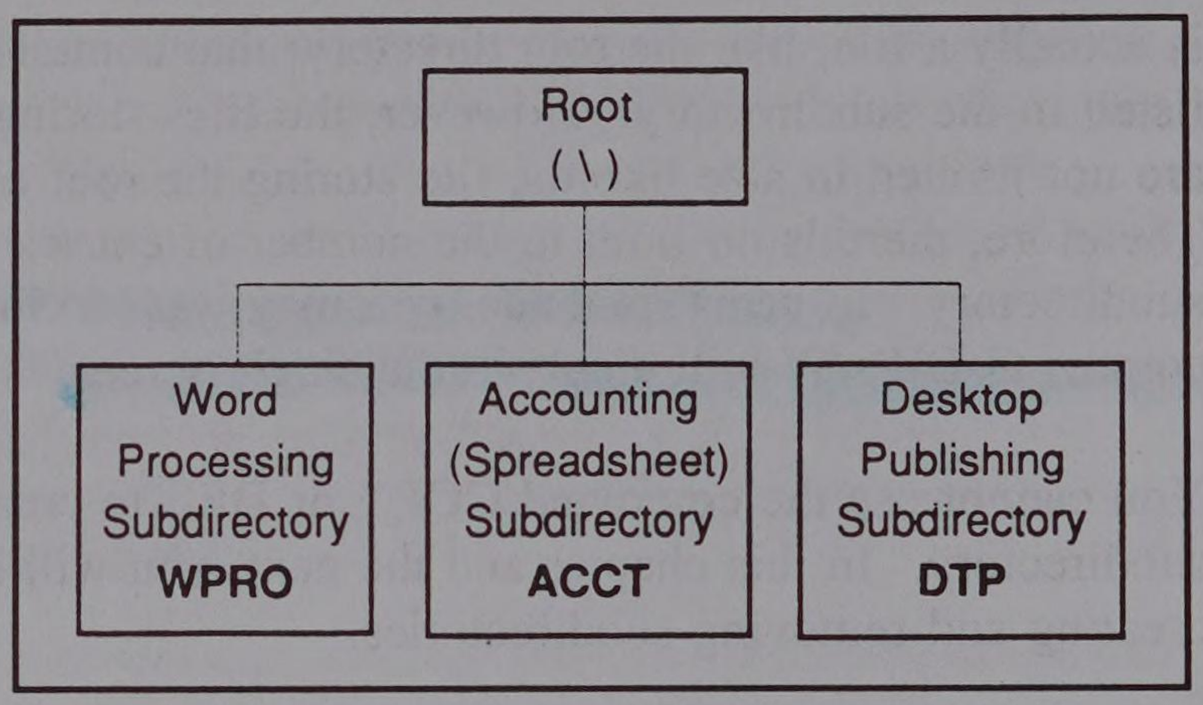

Figure 6-2. The structure for TREE DISK.

To see the directories you created on the disk in drive A, key

- A:\>**DIR** [Enter]

The system displays:

```
 Volume in drive A is TREE DISK
 Volume Serial Number is _______
 Directory of A:\

COMMAND      COM  52589  06-29-93      12:00p
WPRO         <DIR>       09-17-94      10:15a
ACCT         <DIR>       09-17-94      10:16a
DTP          <DIR>       09-17-94      10:17a
             4 file(s)   52589 bytes
                       727040 bytes free
```

Note that the directories are identified by <DIR> and that no size is shown for them. The backslash in the line that says "Directory of A:\" refers to the root directory of the disk. The line below the list of directories shows that there are four files—DOS counts the directories as files.

If you make a mistake and want to remove a directory, key the command RD (remove directory) and the directory name.

- A:\>**RD <directory name>** [Enter]

You will learn more about removing directories in the next chapter.

SPECIFYING THE PATH TO A DIRECTORY

Because WPRO is a directory, the contents can be displayed with the directory command DIR, just as you display the contents of the root directory. However,

it is necessary to tell DOS the route to follow to find the WPRO directory; this route is referred to as the **pathname** or **path**. The pathname specifies the way to a directory, usually starting from the root directory. Notice in the command line in the next Practice section that a backslash (\) is used before the WPRO directory name. The backslash means that the path starts at the root directory, represented by the backslash. The command line instructs DOS to start at the root directory, to go from there to the WPRO directory, and to display a directory listing of the WPRO directory.

Practice: Specifying Paths to Directories

Key the following command line:

- A:\>**DIR \WPRO** [Enter]

The contents of the WPRO directory are displayed as:

```
 Volume in drive A is TREE DISK
 Volume Serial Number is ____
 Directory of A:\WPRO

.              <DIR>  09-17-94      10:15a
..             <DIR>  09-17-94      10:15a
        2 file(s)          0 bytes
                      727040 bytes free
```

It looks as though WPRO contains two directories named . (one dot) and .. (two dots). However, these are not true directories; they are **subdirectory markers** used to access directories. Remember that we said that directories are actually files containing information about files? The subdirectory marker file .. contains the name of the **parent** or next higher directory (in this case, the root directory). The marker file . contains the name of the **current** directory, in this case, WPRO. More about the markers later.

The display tells you that you are looking at the directory listing of A:\WPRO. The path is shown as going from the drive A root directory to WPRO.

The pathname will continue with another backslash if you add a file name. For example, if the directory WPRO contained a file named LETTER.JRB, the pathname with the drive and file name attached would be A:\WPRO\LETTER.JRB. Here the second backslash is a separator—it does not signify the root directory. Notice there are no spaces in this entry, which is called a **filespec**. This filespec includes the drive designator, pathname, and file name.

When the default directory is the root directory, it is *not* mandatory that you begin the path with the backslash (\). For example, if you are in the root

directory and want to display a listing of the DTP directory, which is in the next level below the root, the backslash is not necessary. You can key

- A:\>**DIR DTP** [Enter]

You can also key

- A:\>**DIR \DTP** [Enter]

Try both of these options.

CHANGING THE CURRENT (DEFAULT) DIRECTORY

When you want to work with files in a directory other than the default shown by the prompt, it is often more efficient to change to that directory—that is, to make it the current or default directory. To do this, use the change directory command, CHDIR or CD. CD is shown in this textbook. It must be followed by the path to the directory you want to make current.

Practice: Changing the Current Directory

For example, if you want to change the current directory from the root directory to the subdirectory named WPRO, you can do so by keying

- A:\>**CD \WPRO** [Enter]

This is another example of a time when the backslash is optional. Since you are in the root directory, it is not mandatory.

After you enter the CD command to change to the WPRO directory, notice that the prompt has changed to show that the current directory is now WPRO.

```
A:\WPRO>
```

Now key the directory command to display the directory and verify that you are in the WPRO directory.

- A:\WPRO>**DIR** [Enter]

You will see the following:

```
Volume in drive A is TREE DISK
Volume Serial Number is ___
Directory of A:\WPRO

.              <DIR>  09-17-94      10:15a
..             <DIR>  09-17-94      10:15a
               2 file(s)      0 bytes
                         727040 bytes free
```

You have seen how the DOS prompt and top lines of the directory display tell you where you are in the tree structure. There is a third way to determine the name of the current directory: If you enter the change directory command CD with no parameters, rather than change the directory, the system will check the current directory and display its path and name.

To verify this, key the following check directory command at the DOS prompt:

- **A:\WPRO>CD** [Enter]

The system should respond with:

```
A:\WPRO
```

To check the current directory on another drive, add the drive designator as a parameter to the CD command. Try

- **A:\WPRO>CD C:** [Enter]

To review, the CD command can be used to *change* or *check* directories. If you use CD with a path or directory name as the parameter, the system will change to the directory specified. If you use CD without a parameter, the system will identify the current directory on the default drive by drive and pathname. If you use CD with a drive designator as the parameter, the system will identify the current directory on the drive specified.

When you are in a subdirectory and want to work with another subdirectory that is part of another "branch" of the tree, you must always specify the full path including the backslash. This tells DOS that the path begins at the root directory.

- A:\WPRO>**DIR \DTP** [Enter]

Without the \ , DOS would assume that DTP was at the level below WPRO in the tree structure. Remember, the backslash is always needed unless you are in the root directory and/or want to specify a path to a directory that is listed in the current directory. In other words, if the directory you want to list is a child

of (one level below) the current directory shown by the prompt, you do not need to use the backslash.

Since the backslash is used to signify the root directory, you can change back to the root directory using the CD command and the backslash. Change to the root directory with:

- A:\WPRO>**CD ** [Enter]

The prompt will change to:

```
A:\>
```

Now the prompt tells you that you are working in the root directory and that commands you enter will apply to the root directory of the diskette in drive A.

Change to the ACCT subdirectory:

- A:\>**CD ACCT** [Enter]

The prompt will change to:

```
A:\ACCT>
```

The entries in the ACCT directory are displayed when you key:

- A:\ACCT>**DIR** [Enter]

Practice: Creating Files in the Directories

Now you can work with files in some of the subdirectories. Change to the WPRO directory. To create a file for this directory using the COPY CON technique, key the following text while in the WPRO subdirectory:

- A:\WPRO>**COPY CON SUB.TXT** [Enter]
- **I am creating a file for the WPRO subdirectory.** [Enter]
- **It is easy!** [Enter]

Press the F6 key or Ctrl-Z and [Enter] to end the file and save it.

Since you instructed the system to create the file SUB.TXT and placed it in the current directory, you should see the new file listed when you list the WPRO directory. Try it.

- A:\WPRO>**DIR** [Enter]

The contents of the WPRO directory are displayed.

```
 Volume in drive A is TREE DISK
 Volume Serial Number is _____
 Directory of A:\WPRO

  .              <DIR>  09-17-94      10:15a
 ..              <DIR>  09-17-94      10:15a
 SUB     TXT        62  09-17-94      10:55a
        3 file(s)     62 bytes
                  724992 bytes free
```

Now, from the WPRO subdirectory, create a file for the root directory and place it there *without* changing to the root directory. Key the following:

- A:\WPRO>**COPY CON \REVIEW.TXT** [Enter]
- **I am creating a file for the root directory.** [Enter]
- **I keyed a backslash before its name so it** [Enter]
- **will be stored there.** [Enter]

Press the F6 key or Ctrl-Z and [Enter] to end the file and save it.

Notice that you instructed the system to put the file REVIEW.TXT in the root directory by placing the backslash before the file name.

Check to see that REVIEW.TXT is in the root directory by using the backslash as a parameter for the DIR command.

- A:\WPRO>**DIR ** [Enter]

The system lists the entries in the root directory, including the new file REVIEW.TXT. (Your listing may have additional entries.)

```
 Volume in drive A is TREE DISK
 Volume Serial Number is ________
 Directory of A:\

 COMMAND     COM    52589   06-29-93     12:00p
 WPRO            <DIR>       09-17-94     10:15a
 ACCT            <DIR>       09-17-94     10:20a
 DTP             <DIR>       09-17-94     10:25a
 REVIEW  TXT          112    09-17-94     11:00a
                5 file(s)  52701 bytes
                         723968 bytes free
```

The root directory of the disk in drive A contains at least three subdirectories and the file you just created. Note that you are still in the WPRO subdirectory.

Create another file for the WPRO directory. Key the following:

- A:\WPRO>**COPY CON ADD.TXT** [Enter]
- **This file belongs in the WPRO directory.** [Enter]
- **^Z** [Enter]

Now display the directory listing of the current directory:

- A:\WPRO>**DIR** [Enter]

Move to the root directory as follows:

- A:\WPRO>**CD **

You may want to see a list of only the subdirectories with no files included. To display a directory listing of the subdirectories, key

- A:\>**DIR *.** [Enter]

With this command you request a listing of files with no extension in their file name. Since the directory names have no extension, they will be selected. If the directory contained other files named without an extension, such as TODOFORM, these would be included in the listing.

Create a file of your own called FILE1.ABC for the ACCT directory. You can do this from the WPRO directory. Change to the WPRO subdirectory and key the following command:

- A:\WPRO>**COPY CON \ACCT\FILE1.ABC** [Enter]

The pathname \ACCT tells DOS to place the new file in the ACCT directory below the root directory. The second backslash is a separator between the directory name and file name.

You can also change to the ACCT directory and create a file as follows:

- A:\WPRO>**CD \ACCT** [Enter]
- A:\ACCT>**COPY CON FILE2.ABC** [Enter]

Note that you do not have to specify the path for the file when you are in the ACCT directory. Use this technique to create a short file and store it in the ACCT directory. Display the ACCT directory listing to see that your two new files are listed.

Next, create two files for the DTP directory called JOB1.DOC and JOB2.DOC using the same techniques. First create JOB1.DOC for the DTP directory while

you are in the ACCT directory. Then change to the DTP directory and create the file JOB2.DOC. List the directory of the DTP directory to see that your two new files are there.

Practice: Copying Files from Floppy Diskettes to the Hard Disk

You may need to copy from one floppy diskette to another using the same floppy drive. For example, you may have only one floppy drive. You could use the following command to do this copying (remember, if you are using a system with only one floppy, DOS will use that drive as both A and B).

- A:\>**COPY *.* B:** [Enter]

If you are working with a system with only one floppy drive, the command will require that you swap the source disk and target disk for each file copied. To avoid this tedious process, you can set up a directory on the hard disk and copy the files from the source floppy to this directory with COPY *.*. Then, after exchanging floppy disks, copy the files from the hard disk directory to the target floppy disk. Finally, erase the files from the hard disk and remove the directory.

To make a temporary directory on the hard disk named TEMP, key the following at the C:\> prompt:

- C:\>**MD TEMP** [Enter]

Insert the CLASS disk in drive A from which you would like all the files copied. Copy all the files on the disk in drive A into the TEMP directory with this command:

- C:\>**COPY A:*.* TEMP** [Enter]

Next, insert the TREE DISK diskette in the A drive and use the following command to copy the files to this floppy diskette:

- **C:\>COPY TEMP*.* A:** [Enter]

If you would like to be sure that the files have copied correctly, use the COMP *.* command to compare files.

Erase the files in the TEMP directory and remove the directory with these commands:

- **C:\>DEL TEMP** [Enter]
- **C:\>RD TEMP** [Enter]

PRINTING DIRECTORY LISTINGS

You can print directory listings by using the redirect process. The **right redirect**, the greater-than sign (>), redirects output from the default device to another device. The default device for the display of the directory is the screen. You can redirect the display from the screen to the printer by using the right redirect symbol between the command DIR and the device name PRN.

Practice: Printing Directory Listings

Make sure the printer is on-line. Change to the root directory and key these command lines:

- **A:\>DIR > PRN** [Enter]
- **A:\>DIR WPRO > PRN** [Enter]
- **A:\>DIR ACCT > PRN** [Enter]
- **A:\>DIR DTP > PRN** [Enter]

SEARCHING FOR FILES IN A TREE STRUCTURE

Now that you are working with subdirectories, you may find that it is sometimes difficult to locate files. For example, you might not remember in which directory you placed a file or be uncertain about the exact name of a file. DOS 5.0 introduces a new directory command that will help you locate files in these circumstances. The switch /S tells DOS to list the files in all subdirectories where the file is found.

Practice: Using /S to Search for Files in a Tree Structure

To find the file ADD.TXT, change to the root directory and key the following command line:

- **A:\>DIR ADD.TXT /S** [Enter]

DOS will begin looking in the root directory and search in all subdirectories for the file. If a copy is in more than one directory, all the locations of the file will be listed.

Suppose you remember creating a file named JOB1, but have forgotten what its extension is and where you placed it. You can search for it with a wildcard.

- **A:\>DIR JOB1.* /S** [Enter]

The /S switch can be used to display all the directories and files on your disk in one step.

- A:\>**DIR /S** [Enter]

Also, you might try

- A:\>**DIR /B** [Enter]

Notice that you have the directory listing with only a portion of the information, specifically the filename and extension.

You can redirect the directory list to the printer in the same way that you did previously:

- A:\>**DIR /S > PRN** [Enter]

Nice work on your introduction to subdirectories! Test your knowledge of the chapter with the Self-Check Quiz and Assignment.

CHAPTER 6 SELF-CHECK QUIZ

True/False (Write T if the statement is true, F if the statement is false.)

_____1. Subdirectories have a fixed number of entries.

_____2. Use the DIR /S command to locate a file anywhere on drive C.

_____3. The root directory is created during the formatting process.

_____4. You must always include the backslash when you specify a path to a subdirectory.

_____5. The contents of a subdirectory can be printed using the right redirect (>) technique.

Multiple Choice (Circle the correct answer.)

6. Which represents the maximum number of files that can be placed on a 3 1/2-inch double density diskette?

   a. 512
   b. 224
   c. 112
   d. No limit

7. What command can you use to change from the TEXT directory to the root directory?

   a. A:\TEXT> CD ROOT
   b. A:\TEXT> CD \
   c. A:\TEXT> CD
   d. A:\TEXT> CH DIR

8. Which command would you enter to create the subdirectory named GRAPHS branching from the root directory?

   a. A:\>MD GRAPHS
   b. A:\>MAKE GRAPHS
   c. A:\TEXT>MKDIR GRAPHS
   d. A:\GRAPHS>MAKE GRAPHS

9. What are the naming conventions for subdirectories?

   a. The same naming conventions as for a file
   b. A maximum of eight letters including spaces
   c. There are no naming conventions for subdirectories
   d. A maximum of eight letters, no spaces allowed

10. What command would you enter to create a file named DOC7.TXT in the WPRO subdirectory from the root directory?

    a. A:\>COPY CON DOC7.TXT\WPRO
    b. A:\>COPY CON WPRO DOC7.TXT
    c. A:\>WPRO\COPY CON \DOC7.TXT
    d. A:\>COPY CON WPRO\DOC7.TXT

ASSIGNMENT

To practice using the commands in this chapter, complete the following steps. Check off each step when you understand its operation, or mark the step with a question mark if it is not clear. Write the commands or keystrokes in the space provided.

1. Change to the A drive and change the prompt to reflect the root directory.

   _____

2. Clear the screen. To store communication software and data, create a subdirectory named COMM below the root directory of the TREE DISK.

   _____

3. Change to the COMM subdirectory and create a text file named COMMDATA.TXT. Compose the wording of the text file and save it.

   _____

4. Print the root directory of the disk. Print the COMM directory listing.

   _____

5. Locate the file COMMDATA.TXT from the root directory using the directory search command. Print the screen.

   _____

6. Create a new subdirectory named TEMP on the root directory of drive C. Print a directory of the TEMP subdirectory.

   _____

7. Create a file named REPORTS.TXT in the COMM directory of drive A. Use your own wording to compose the file. Print the file.

   _____

8. Copy the file REPORTS.TXT to the TEMP subdirectory on drive C. Print the TEMP directory.

   _____

9. Move and overwrite the file, REPORTS.TXT, from the TEMP directory of drive C back to the COMM directory on the TREE DISK diskette in drive A. Print the directory of the TREE DISK listing the REPORTS.TXT file.

   _____

10. Remove the TEMP directory on drive C. Print the screen.

    _____

CHAPTER 7

WORKING WITH A TREE STRUCTURE

OBJECTIVES

When you have completed the activities and assignment in this chapter, you will be able to:

1. Copy files from one directory to another.
2. Make subdirectories within existing subdirectories.
3. Specify the path to a file.
4. Use subdirectory markers to change directories and to delete all files in a directory.
5. Remove a subdirectory with the remove directory (RMDIR or RD) command.
6. Display and print the directories on a disk with the TREE command.
7. Use the DOS 6.0 DELTREE command to delete a directory and all subordinate directories and files.
8. Use the DOS 6.0 MOVE command to rename a subdirectory.
9. Use the XCOPY command to copy files within a tree structure.

DIRECTORY AND FILE MANAGEMENT

Managing files on a hard disk often requires that you move from one directory to another and specify file locations within a multi-level tree structure. To expand your file management skills and practice using them in a more complex tree structure, you will learn to copy files from one directory to another. You will create a third level of directories and copy files into these directories.

Practice: Copying Files from One Directory to Another

A file can be copied from one directory to another. Use the COPY command with the necessary parts of the filespecs (drive, pathname, and file name) for the source and destination files.

First, insert the TREE DISK diskette and change to the WPRO directory with the following command:

- A:\>**CD WPRO** [Enter]

Display a directory of the root to see if REVIEW.TXT is there.

- A:\WPRO>**DIR \REVIEW.TXT** [Enter]

To copy REVIEW.TXT from the root directory to a file named LESSON.TXT in the WPRO directory, key:

- A:\WPRO>**COPY \REVIEW.TXT LESSON.TXT** [Enter]

The system responds with:

```
1 file(s) copied
```

You included the pathname (the backslash meaning the root directory), the source file name (REVIEW.TXT), and the new file name for the copy (LESSON.TXT); you did not include a pathname for LESSON.TXT because it is the same as the path shown in the prompt. You did not include the drive because you wanted the copy to be on the disk in the default drive shown in the prompt.

Now display the default directory WPRO. Does the list have three TXT files in it?

A:\WPRO>**DIR** [Enter]

```
 Volume in drive A is TREE DISK
 Volume Serial Number is _________
 Directory of A:\WPRO

 .            <DIR>  09-17-94      10:15a
 ..           <DIR>  09-17-94      10:15a
SUB      TXT     63  09-17-94      10:55a
ADD      TXT     45  09-17-94      11:05a
LESSON   TXT    112  09-17-94      11:00a
            5 file(s)      220 bytes
                       713728 bytes free
```

The subdirectory markers also count as a file since they represent directory files.

You learned earlier that a file copied into the same directory must have a different name. This is another advantage of using subdirectories. Two (or more) files on the same *disk* can have the same name providing *each is stored in a different directory*. Since their filespecs are different, the system can tell them apart.

You can verify this by copying the file named LESSON.TXT from the WPRO directory into the ACCT directory, using the same file name. Be sure you are in the WPRO directory. Key the following:

- A:\WPRO>**COPY LESSON.TXT \ACCT** [Enter]

The file name is not repeated for the destination directory because you want the file in the ACCT directory to have the same name as the original file. You must, however, give the new file's complete pathname starting at the root directory, since the file is to be placed in a directory that is another branch of the tree structure.

Check to see that the file was copied. Display the directory of ACCT by keying at the DOS prompt:

- A:\WPRO>**DIR \ACCT** [Enter]

Now there are two files named LESSON.TXT on the same disk, but they are in different directories. The different pathnames give them different filespecs and make them as different to the system as if they had different file names.

ADDING SUBDIRECTORIES TO CREATE A THIRD LEVEL

Subdirectories can contain any type of file, including other subdirectories. Like putting dividers between dividers in a file drawer, this further structuring narrows the subject of a storage compartment and makes file and disk management easier.

A common practice is to store data files in directories a level below the application software directories. For example, you could divide the work you do with each software application by type or by project. You could add the subdirectories shown in capital letters in the following lists and then place data files pertaining to each type of work or project in those directories. Complete your tree structure by creating the third level of subdirectories as indicated:

Word Processing	Accounting	Desktop Publishing
WPRO	ACCT	DTP
REPORTS (reports)	CORP (corporate)	TEXT (text)
CORRES (letters)	PROJ1 (project 1)	ART (graphics)
MISC (miscellaneous)	PROJ2 (project 2)	PUBS (publications)

Practice: Creating Subdirectories on a Third Level

If necessary, insert the TREE DISK diskette. To create the third level of subdirectories under WPRO, be sure you are in the WPRO subdirectory and key these commands to create the three subdirectories:

- A:\WPRO>**MD REPORTS** [Enter]
- A:\WPRO>**MD CORRES** [Enter]
- A:\WPRO>**MD MISC** [Enter]

Display the WPRO directory. You should see the three new directories just made plus the files copied previously.

Change to the ACCT subdirectory and create the third level of subdirectories (CORP, PROJ1, PROJ2). Then change to the DTP directory and create the third level of subdirectories for DTP (TEXT, ART, PUBS). When you have finished, compare your structure to that shown in Figure 7-1.

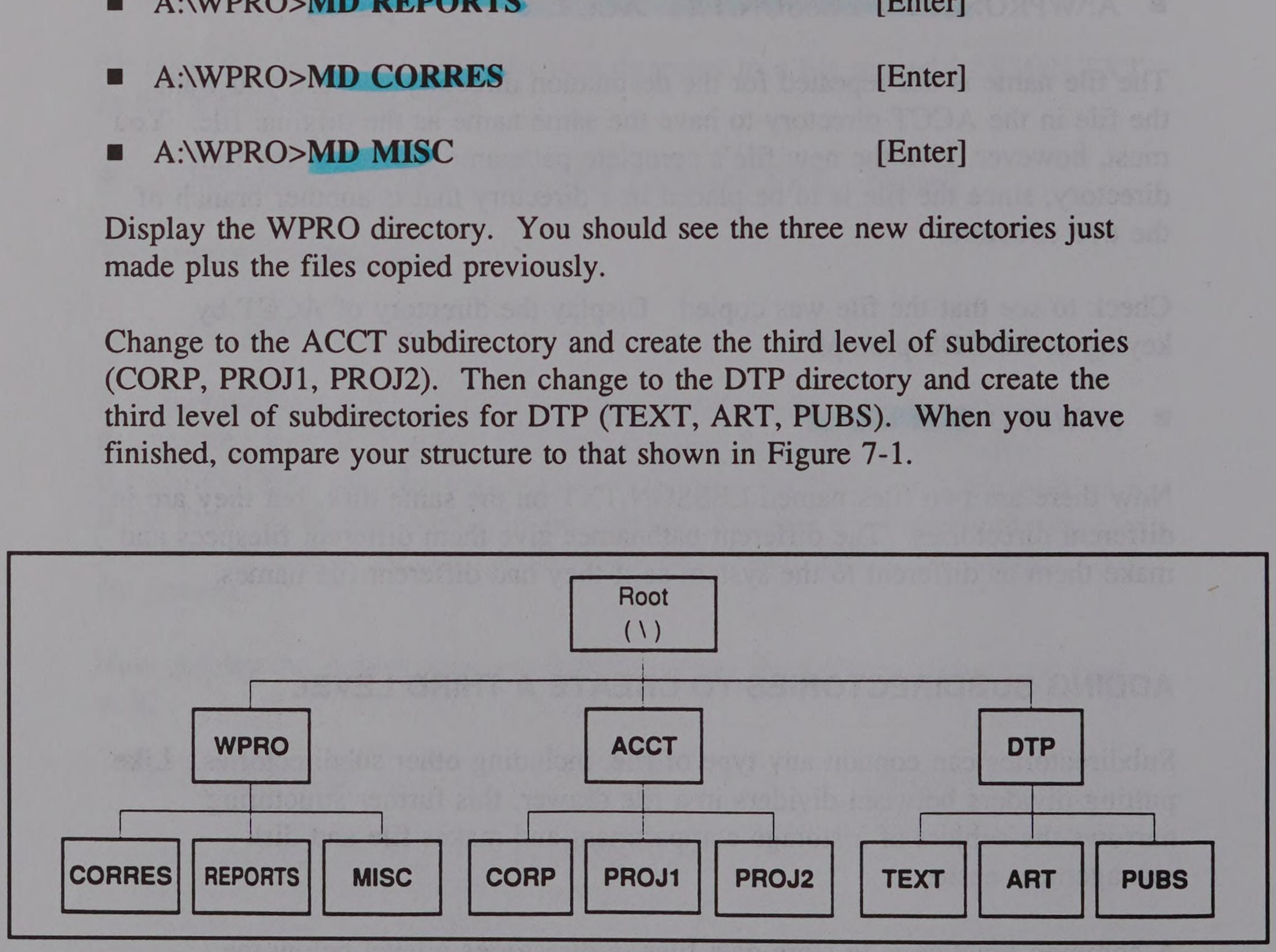

Figure 7-1. Three-level TREE DISK tree structure.

USING THE CD COMMAND WITH ANOTHER DRIVE

The change directory command (CD) has two parameters, drive and pathname or path. Drive is the disk drive designator (A:). Path includes the names of

any directories through which DOS will be passing, ending with the directory you want to be current. If you omit the drive, the default drive is used. If you specify a drive, DOS looks for the path you specify on the disk in the drive you specify.

Practice: Using the CD Command

To verify that DOS will look for the path you specify on the disk in the drive you designate, change to drive C and key the following:

- C:\>**CD A:\WPRO\REPORTS** [Enter]

The system will look at the disk in drive A, since you specified that drive. DOS will begin looking there in the root directory (\) for a directory named WPRO. Next, DOS will look in the WPRO directory for a directory named REPORTS. DOS will change the current directory for drive A to the REPORTS directory. It is important to note that DOS will check the REPORTS subdirectory in the A drive, but that your prompt will still reflect the C drive.

Note that *there are no spaces* in the path. If you key a space following WPRO, as C:\>CD A:\WPRO \REPORTS, the system will change to the WPRO directory and will ignore the portion of the pathname keyed after the space.

If you omit the drive and pathname and just enter CD, DOS will display the current directory name. If you enter the drive and omit the path, DOS will report the current directory of the other drive. To see the current directory of drive A, key:

- C:\>**CD A:** [Enter]

DOS will report the status of the A drive; however, the prompt will still reflect the C drive. This trick can be a timesaver when you are copying files between tree-structured disks and want to know the current directories on both the source disk and the target disk.

USING SUBDIRECTORY MARKERS

Remember the markers . (one dot) and .. (two dots) listed in each subdirectory? The marker . represents a file containing the name of the current directory. The marker .. represents a file containing the name of the parent directory. You can use this marker with the CD command to move to that directory—that is, to move up one level in the tree structure. You can use the . marker to perform operations on the current directory, such as copying, displaying, or deleting its contents.

Practice: Using Subdirectory Markers

If necessary, change to drive A; then change to the REPORTS directory by keying

■ A:\>**CD WPRO\REPORTS** [Enter]

To move from this directory up one level toward the root directory, key:

■ A:\WPRO\REPORTS>**CD ..** [Enter]

```
A:\WPRO>
```
(This will be the new prompt.)

As you can see, the current directory has changed to the parent directory WPRO, which is one level above REPORTS. By pressing F3, you can repeat the command CD .. to return to the root directory.

REMOVING A DIRECTORY

When you want to delete directories from your tree structure, you will use the remove directory (**RMDIR** or **RD**) command. Most users use RD because it requires fewer keystrokes. In this textbook RD is used as the command to remove subdirectories.

There are two requirements for removing a directory:

- A directory cannot be deleted if it lists any subdirectories or files other than the subdirectory marker files. This is to ensure that no files or directories are left without a directory structure.

- The current directory cannot be removed. You must change to another directory before issuing the remove directory command. DOS will not allow you to saw off the limb you are sitting on!

The remove directory command has two parameters: drive and pathname or path. These are used in the same way with RD as with the CD and MD commands.

If you want to remove a directory but need some of the files it contains, copy the files you need to another directory or disk, then erase all the files in the obsolete directory and remove the directory.

There are two ways to remove all files from a directory. One method is to use the DEL or ERASE command with the wildcards that select all files:

A:\>DEL WPRO\REPORTS*.* *or* A:\>DEL WPRO\REPORTS\ .

or

A:\>WPRO\REPORTS>DEL *.*

Another method is to use the DEL or ERASE command with the directory name alone to remove all files in the specified subdirectory:

A:\WPRO>DEL REPORTS

or

A:\WPRO\REPORTS>DEL .

The last example shows how you can use the one-dot directory marker (.). Remember that this marker is for a file that stores the name of the directory where the marker appears. DOS interprets the dot in the command DEL . as the current directory name, in this case, REPORTS.

ERASE can be used in place of DEL in the preceding examples. Either command tells DOS to erase the files in the specified directory.

Practice: Removing Directories

Change to the appropriate directory and use either the DEL or ERASE command to remove the files from the REPORTS directory. Both of the following commands will provide the same results: All files are deleted from the REPORTS directory.

- A:\WPRO\REPORTS>**DEL *.*** *or* A:\WPRO\REPORTS>**DEL .** [Enter]

  or

- A:\WPRO>**DEL REPORTS** [Enter]

When the system asks if you are sure you want to erase the files, key **Y** for yes. This will not remove the directory; it simply erases the files within the directory. Display the REPORTS directory to confirm that it has no files.

When the REPORTS directory is empty, make sure that it is *not* the current directory; then delete the REPORTS directory by keying

- A:\WPRO>**RD REPORTS** [Enter]

  or

- A:\>**RD WPRO\REPORTS** [Enter]

Display the WPRO directory to ensure that REPORTS no longer exists. Remember: You use ERASE or DEL commands to remove files and the RD command to remove directories.

USING THE TREE COMMAND TO DISPLAY THE TREE STRUCTURE

When you create a tree structure with several levels, you may not remember exactly what subdirectories you created or exactly where they are. You might display the contents of each directory on the disk to find the files marked <DIR>, but there is a quicker way to display the complete structure. Use the TREE command to view a graphical representation of the contents of your disk.

Practice: Displaying the Tree Structure

Since TREE is an external command, you must be sure your search path is set to the DOS external command files. Check to be sure you are in the root directory and then key:

- **A:\>TREE** [Enter]

This instructs DOS to find the TREE command file, load the command into memory, and display a tree of the disk in drive A.

You can also request DOS to display the tree structure of a disk in another drive. To view the tree structure of the C drive, key:

- A:\>**TREE C:** [Enter]

Notice that there are no files included in the display. The TREE command has a switch that modifies the command to list all files in all directories for a complete listing of every file on the disk, except hidden files. To display the tree of the disk in drive A and include the files in the root directory and each subdirectory, key:

- A:\>**TREE /F** [Enter]

To display the tree and file listing for a different disk that is not in the default drive, key:

- A:\>**TREE C:/F** [Enter]

When you use the /F switch, the displayed tree includes the name of every file in every directory. This can be quite a long list if you have many small files or are displaying the tree of a hard disk. A printed copy of such a report can be helpful, especially if your filing system has several levels. You can use the right-redirect symbol (>) to send the output of the TREE command to the printer.

- **A:\>TREE /F > PRN** [Enter]

Still another option is to redirect the output of the TREE command to a file, to obtain a permanent record on disk. Try this:

- A:\>**TREE /F > TREEFILE** [Enter]

Similar to the technique used to print the directory with the command DIR > PRN, you have redirected a tree listing to a file.

To send the file TREEFILE to the printer, key:

- A:\>**COPY TREEFILE PRN** [Enter]

You could also use

- A:\>**TYPE TREEFILE > PRN** [Enter]

DOS 6.0 DELTREE COMMAND

DOS 6.0 has a new command that eliminates the need to delete all files and subdirectories below a given subdirectory before removing a directory. With the command DELTREE it is possible to delete all levels of subdirectories below the directory including the files in one command. For example, if you want to delete all branches of the DTP directory, key:

A:\>DELTREE DTP

Answer the screen prompt `Delete directory DTP and all its subdirectories? (Y/N).` When you respond **Y** for yes, all files and subdirectories of the DTP directory will be removed.

It is also possible to remove only one subdirectory without first removing the files by keying the name of the subdirectory. To remove all files and the subdirectory CORP from the ACCT directory, key:

A:\ACCT>DELTREE CORP

As you can see, this is a command to be used with extreme caution. You cannot use UNDELETE to restore the files deleted with DELTREE.

Practice: Using the DELTREE Command

Look at the directory structure of the TREE disk again with the following command:

- A:\>**TREE /F** [Enter]

With the TREE DISK in drive A, remove a branch of the directory structure with the following command:

- A:\>**DELTREE DTP** [Enter]

Respond to the screen prompt and all the subdirectories and files below DTP will be deleted.

Now erase one subdirectory and its files with the following command:

- A:\>**DELTREE A:\ACCT\CORP** [Enter]

  or

- A:\>**CD ACCT** [Enter]
- A:\ACCT>**DELTREE CORP** [Enter]

Run a tree structure to ensure that the CORP subdirectory has been deleted.

Use the DELTREE command to delete the TEMP subdirectory from drive C.

RENAMING DIRECTORIES

In DOS 5.0 you were able to rename a subdirectory only by using the SHELL, a graphic interface to DOS, but in DOS 6.0 the MOVE command will rename a directory. It is not possible to rename the current directory just as it is not possible to remove the current directory. To rename the directory SMYTH to SMITH, key:

A:\>**MOVE SMYTH SMITH**

The use of this command will rename the SMYTH directory to SMITH.

Practice: Using the RENAME Directory Command

Access the ACCT directory with the following:

- A:\>**CD \ACCT** [Enter]

Rename the PROJ2 directory to TRANSPROJ with the following command:

- A:\ACCT>**MOVE PROJ2 TRANSPROJ** [Enter]

USING THE XCOPY COMMAND

XCOPY is a very powerful command for copying files that was added to DOS version 3.2. XCOPY is more versatile than COPY, as it can copy files that exist in different directories and will create those directories while copying the files. XCOPY is an external command that has the same basic syntax as COPY:

XCOPY source file(s) destination file(s)

The files to be copied are specified just as you would with the COPY command. You can use wildcards with XCOPY. A number of useful switches are available for controlling file transfer with XCOPY. Suppose that scattered throughout your hard disk directories were files with the extension LTR containing correspondence. You want to copy these files to a backup diskette in directories that correspond to the source directories. With the COPY command, you would have to create each directory on the target disk and then copy the source files from the source directory to the corresponding destination directory on the target disk, working with one directory at a time.

With XCOPY, you would simply enter the following command with the switch /S:

```
A:\>XCOPY C:\*.LTR A: /S
```

This command line issues the following instructions to DOS:

- Search through the root directory and all subdirectories of the source disk in drive C for files with the LTR extension.
- Create the necessary subdirectories on the target disk in drive A.
- Copy the LTR files to the appropriate destination directories on the target disk.

XCOPY does not limit the copy process to one size or capacity disk. Therefore, you can XCOPY to and from hard disks and floppy diskettes of any size or capacity provided you do not exceed the capacity of the target disk.

XCOPY can be used with the following switches:

/A Copies files that have the archive attribute on; does not reset the attribute to off.

/D Copies only the files created on or after a date you specify.

/E Duplicates subdirectories on the target disk that are on the source disk even if they are empty.

/M Copies files that have the archive attribute on; resets the attribute to off.

/P Prompts you before each file is copied for selective copying of files.

/S Copies the current directory and its subdirectories.

/V Verifies the correct copying of files.

/W Waits for you to insert a source disk if you are copying to the same drive.

Another important advantage of XCOPY for systems with one floppy drive is that XCOPY reads as many files as possible into memory and then writes them to the target disk. Thus, XCOPY can be used to copy a group of files from one floppy disk to another on a one floppy drive system without swapping disks after each file, as you must do with the COPY command.

> **Note:** One important caution about this command is that if there are any files on the target disk with names identical to the files on the source disk, the files on the target disk will be overwritten. However, DOS 6.2 has added a prompt that advises if a file of the same name exists and gives you the choice to overwrite it or not.

Practice: Using the XCOPY Command

Put the TREE DISK diskette in drive A and have a blank formatted diskette named FILES BACKUP disk handy to serve as the target diskette. Use XCOPY to make a backup of the TREE DISK diskette as follows:

- A:\>**XCOPY A:*.* B: /S /E** *or* A:\>**XCOPY A: B: /S /E** [Enter]

You have instructed DOS to copy files in the default (root) directory and all subdirectories of the drive A source disk to the target disk in drive B. In the process, XCOPY will create the subdirectories if they do not exist on the target disk. The /E switch tells DOS to create the directories even if the source disk directories are empty.

Read the screen messages and prompts. XCOPY will read a series of files into memory and then write them to disk. If you are using one floppy drive, watch

for instructions to exchange disks. You may need to swap disks several times, but not nearly as often as you would with the COPY *.* command.

The result of this process will be a backup copy of your work disk with the same tree structure. Note that the source and target disks do *not* have to be the same size and capacity. If you want to verify the contents of the disks, insert the source disk in drive A, and key:

- **A:\>TREE /F > PRN** [Enter]

Now insert the target disk and key the same command. Compare the printouts.

Nice work with subdirectories! Test your knowledge with the Self-Check Quiz and Assignment.

CHAPTER 7 SELF-CHECK QUIZ

True/False (Write T if the statement is true, F if the statement is false.)

_____ 1. To redirect the screen display of the TREE command to the printer, use the right redirect (>).

_____ 2. The root directory, subdirectories, and their files make up the tree structure.

_____ 3. The complete tree structure, including file names, can be displayed but cannot be printed.

_____ 4. The same file can exist in two subdirectories but must have different names.

_____ 5. You cannot delete a subdirectory unless named files are still in it.

Multiple Choice (Circle the correct answer.)

6. Which *two* of the following commands are valid commands for copying ADD.TXT to the root directory of drive A?

   a. A:\>COPY ACCT\CORP\ADD.TXT
   b. A:\>COPY CORP\ADD.TXT
   c. A:\ACCT\CORP>COPY ADD.TXT \ ..
   d. A:\ACCT>COPY CORP\ADD.TXT \

7. Which *two* of the following are valid commands for moving up one level in the tree structure?

   a. A:\WPRO\TEXT> CD..
   b. A:\WPRO\TEXT>CD \
   c. A:\WPRO\TEXT>CD \WPRO
   d. A:\WPRO\TEXT>MD \

8. Which command would you use to rename the subdirectory WPRO to WP?

   a. REN WPRO WP
   b. COPY WPRO WP
   c. MOVE WPRO WP
   d. RD WPRO WP

9. Which of the following is true of the DELTREE command?

   a. DELTREE is available only in DOS 5.0 and later.
   b. All files must be removed from the subdirectories before the DELTREE command can be used.
   c. The DELTREE command will not delete files.
   d. The DELTREE command will delete the directory and all levels of the directory below it, including all files, in one command.

10. Which command would you use to copy all files on the disk in drive A, including all its subdirectories to the disk in drive B?

    a. XCOPY A:\ A /S
    b. XCOPY A:S B:
    c. XCOPY A:*.* B:/S
    d. XCOPY A:\ A:/D/D

ASSIGNMENT

To practice using the commands in this chapter, complete the following steps. Check off each step when you understand its operation, or mark the step with a question mark if it is not clear. Write the commands or keystrokes in the space provided.

1. Clear the screen. Using the TREE DISK, make the root directory of drive A the current directory. Use the redirect function to print the directory listing for the root directory.

---

2. Without changing the directory, use the redirect function to print a directory listing of the COMM directory on the disk.

---

3. Copy REVIEW.TXT from the root directory to the COMM directory. Copy SUB.TXT from the WPRO directory to the COMM and the ACCT directories. Do you see any different messages displayed when copying this file to the two different subdirectories?

---

4. Using the redirect function, print the disk's tree structure with files.

---

5. Remove the WPRO directory and print a copy of the tree structure with files.

---

6. Using XCOPY, back up the TREE DISK diskette onto the FILES BACKUP diskette.

---

7. Use the MOVE command to rename the ACCT subdirectory to NEWACCT. Print a copy of the tree with files.

---

8. Create a subdirectory of the NEWACCT directory. Name the directory STOCK. Compose a short file for the STOCK subdirectory called JAN.94. Use your own wording.

---

9. Print a tree showing all directories and files.

---

10. Remove the NEWACCT subdirectory using the DELTREE command.

CHAPTER 8

An Introduction to EDLIN, the DOS Line Editor

OBJECTIVES

When you have completed the activities and assignment in this chapter, you will be able to:

1. Describe uses for EDLIN.
2. Create and edit text files in EDLIN.
3. Use EDLIN commands to list a text file.
4. Use EDLIN commands to append, insert, and delete lines.
5. Use the DOS editing keys for editing in EDLIN.
6. End editing and save the file or quit without saving.

DOS EDITING TECHNIQUES

MS-DOS has two **text editors** to help you create and revise files. One program, EDLIN, is available in all versions of MS-DOS and PC-DOS. The other text editor, EDIT, is available in MS-DOS Versions 5.0 and later and PC-DOS 5.0.

In **EDLIN**, files are created and edited line by line by keying EDLIN commands, specifying lines of text by line number, and keying or modifying individual lines of text. **EDIT**, described in Chapter 9, gives you more flexibility in working with blocks of text, has drop-down menus, and can be used with a mouse.

IBM PC-DOS 6.1 introduces another **text editor** named **E** which is similar to the EDIT program in MS-DOS. If you would like to use the E program, research your PC-DOS documentation for further information.

By using any of the DOS text editors, you produce files that are stored in ASCII format. As you read in Chapter 1, ASCII is a standardized code used for translating text characters into numbers that all computers can process. ASCII files can be used by almost every computer and software program that processes text.

In previous chapters, you learned to create short text files using the COPY CON method which also produces ASCII text. A disadvantage of this method is that there is no way to correct or edit a line after you have ended the line by pressing [Enter]. In EDIT, EDLIN, and E, you can edit any ASCII text file, including those created with COPY CON or with other text editing software.

This chapter introduces you to some features of EDLIN; others will not be covered. Check your documentation if you want additional information. Chapter 9 will help you get started with EDIT. If you complete both this chapter and Chapter 9, you will be able to use both.

USES FOR EDLIN

EDLIN is useful for writing or correcting short lists, messages, and other text files without loading a word processing software package. Using EDLIN is usually preferable to using the COPY CON technique for creating text files longer than a few lines because of the editing capabilities of EDLIN. You can use EDLIN to revise files such as batch files created with COPY CON.

While EDLIN is not a substitute for a word processing software package, you will find it useful when you want to create or revise a short text file without loading a word processor. And, because it is one of the DOS command files, it will always be available when you are using a DOS machine. The EDLIN command file is small compared to the combined size of the command files for that are required for using EDIT. This might be a significant advantage if you need to have a text editor available on a floppy disk.

CREATING A TEXT FILE WITH EDLIN

To access EDLIN, you would key the following:

```
C:\>EDLIN <filename>
```

If the file is new, you will be given the message that you are creating a new file. At that point, you will use EDLIN's initial letter commands.

EDLIN uses initial letter commands. They are easy to remember because they all stand for the action they cause, as in D to delete. In this introduction, you'll learn the most commonly used commands: I (insert), L (list), D (delete), C (copy), M (move), P (page), E (end edit and save), and Q (quit without saving). The DOS editing keys are also useful in editing with EDLIN.

You key text one line at a time and press [Enter] to end each line. While a line can be up to 253 characters in length, you will want to limit line length to a screen width. Form the habit of pressing [Enter] before the text reaches the right edge of the screen and the line wraps to the next line. Revisions are made to one line at a time. Lines can be copied, deleted, or moved singly and in blocks of adjacent lines.

Practice: Creating a Text File with EDLIN

Begin by keying a new text file using EDLIN. EDLIN is an external command and therefore must be accessed in the hard disk directory where DOS files are stored. If necessary, set a search path to the DOS files.

- C:\>**PATH C:\DOS** [Enter]

Place your TREE DISK diskette in drive A. If necessary, change the prompt to display the current drive and directory.

- C:\>**PROMPT PG** [Enter]

To practice using your tree-structured filing system, change to the COMM directory and store the text files you will be creating there.

To load EDLIN and begin a new file, key the command EDLIN, a space, and the file name.

- A:\COMM> **EDLIN TEXTMSG.DOC** [Enter]

The system finds the EDLIN command file in the DOS directory in your search path and loads EDLIN into memory. EDLIN responds with the message `New file` because there is no existing file named TEXTMSG.DOC. The "New file" message is often helpful because it tells you there is no file by that name.

```
New file
*
```

Notice the asterisk below the "New file" message. The * is the EDLIN prompt. Just as DOS uses a prompt to show you where you can key DOS commands, EDLIN has its own command prompt, the asterisk.

The EDLIN screen is divided into two areas (Figure 8-1). At the far left is the command area. When the asterisk prompt and blinking cursor are in this area, you can enter EDLIN commands. Actual text editing takes place in the center and far right of the screen, where text is displayed. When the blinking cursor is in this area, you can revise or add lines.

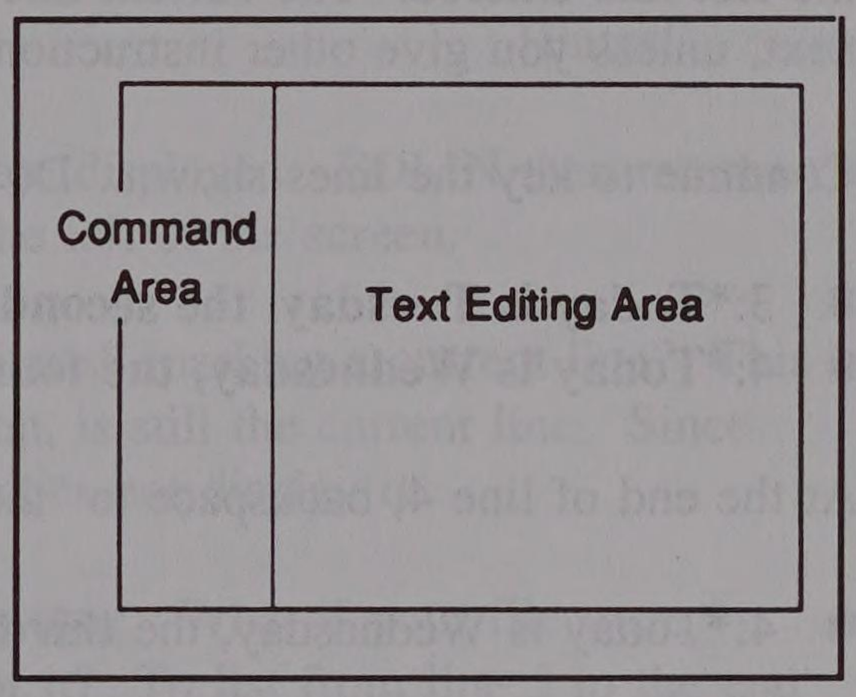

Figure 8-1. The two areas of the EDLIN screen.

To start keying text, it is necessary to be in the insert mode. To get into the insert mode, key the insert command initial letter **I (or i)** at the * prompt and press [Enter].

```
New file
*I                                                    [Enter]
```

Now the line number 1: followed by an asterisk appears in the text editing area. You are ready to key lines of text. Each time you finish a line, press [Enter] and the next numbered line will appear for keying additional text. Backspace to erase errors that you catch before you press [Enter]. Each line is placed in memory when you press [Enter]. The completed lines of text in memory will comprise your TEXTMSG.DOC file.

Key the lines shown in bold type, ignoring any errors you discover after you have pressed [Enter]. You will learn to correct them soon.

```
1:*Today is Monday, the day after Sunday.             [Enter]
2:*Today is Monday, the first day of the week.        [Enter]
3:*
```

The text lines are automatically numbered by EDLIN. Later you will see that when you use EDLIN's delete or insert functions, the lines have been automatically renumbered. The line numbers that appear on the EDLIN screen are not saved with the file and will not appear when it is printed or displayed on the screen. The numbers are displayed only in EDLIN.

Notice that each new line number is followed by an asterisk signifying that this is the line of the file with which you are currently working. In the text editing area, the last asterisk always indicates which line is the **current line** of entry or the last line entered. The current line (3:) is the line EDLIN will work on next, unless you give other instructions.

Continue to key the lines shown. Do *not* press [Enter] at the end of line 4.

- 3:***Today is Tuesday, the second day of the week.** [Enter]
- 4:***Today is Wednesday, the fourth day of this month.**

At the end of line 4, backspace to "the" and rewrite the sentence as follows:

- 4:*Today is Wednesday, the **third day of the week.** [Enter]

Then continue:

- 5:***Today is Thursday, the fourth day of the week.** [Enter]
- 6:***Today is Friday, TGIF.** [Enter]

  7:*

You are now finished keying text and are ready to exit the insert mode because you have no more new lines to key. You must exit the insert mode on a new line so that you retain all keyed lines. Be sure the cursor is at the *beginning* of line 7. Then hold down the Ctrl key, tap the letter Z (Ctrl-Z), and press [Enter].

- 7:*^Z [Enter]

You can press the F6 function key and [Enter] instead of Ctrl-Z and [Enter] to exit the insert mode. F6 also places ^Z on the line. This code is the DOS end-of-file mark. If you use Ctrl-Z or F6 anywhere except at the beginning of a line, you will still be in the insert mode after you press [Enter]. The characters ^Z will appear on the line where you pressed [Enter], and the cursor will move to the next line. If you save a file with ^Z in it and then reload the file into EDLIN, only the portion of the file before the end-of-file mark will be loaded. The exception to this is in DOS 5.0 and 6.1, where you can use the switch /B when you load the file to instruct EDLIN to ignore end-of-file marks.

Other options for exiting from insert mode are to press Ctrl-C or Ctrl-Break. Be sure the cursor is on a blank line—using these options will cancel the current line, the line the cursor is on, erasing it from the file buffer.

When you exit the insert text mode, EDLIN returns to the command mode with an asterisk prompt in the command area. Now list the lines of text you keyed. At the * prompt, key the **list** command initial letter **L** (or **l**) and press [Enter].

■ ***L** [Enter]

EDLIN lists the file. The six lines of text are displayed. EDLIN then returns to the command mode with a * prompt at the left of the screen.

Did you notice that in the list there is no asterisk marking a current line? This is because line 7, the last line you worked on, is still the current line. Since you canceled this line with Ctrl-Z, EDLIN does not display it.

EDLIN can list 23 lines (one screenful) at a time. What if your file were longer or if you wanted to list only a part of it? To list from line 5 to the end of the file, key:

■ ***5L** [Enter]

If the file were longer, the command above would list through line 27.

You can also list from a starting point through a specific line. For example, if you want to list lines 2 through 5, key at the * prompt in the command area: the beginning line number, a comma, the ending line number, and the letter L.

■ ***2,5L** [Enter]

Now end editing and save the completed file. At the command * prompt, key the end edit command initial letter **E** (or **e**) to end editing and exit. When you press [Enter], the file is saved and you are returned to the DOS prompt.

■ ***E** [Enter]

If, for some reason, you decide *not* to save text you have written or edited in EDLIN, you can exit from the EDLIN program without saving the file by keying the quit command, initial letter **Q** (or **q**). Do this only when you do not want to save the text you have entered or edited.

■ ***Q** [Enter]

When you use Q, you are required to confirm that you want to "Abort edit (Y/N)?" Key **Y** to return to DOS or **N** to continue working on the file.

Use the TYPE command to display the TEXTMSG.DOC file. Notice that the line numbers do not appear.

EDITING A TEXT FILE WITH EDLIN

Now edit the file you saved. First you must reload the file into memory. When you load an existing file, EDLIN displays the message, "End of input file." Be sure to key the correct file name or EDLIN will tell you that you are creating a new file. If you try to retrieve an existing file for editing and receive a "New file" message, you will know the system was not able to find your file. Recheck the file name, pathname, and drive designator; then check to see if you have the correct disk in the drive.

Practice: Editing a Text File with EDLIN

In order to edit the file you created, recall the document to the display screen by keying EDLIN followed by the name of the file you want to edit.

- A:\COMM>**EDLIN TEXTMSG.DOC** [Enter]

The screen displays "End of input file." This tells you that such a file exists and that the entire file is loaded into memory. You have seen how EDLIN informs you if the file is a new file or if it already exists by displaying either "New file" or "End of input file." COPY CON does not give you this information. With COPY CON, you could accidentally overwrite a file by giving a new file the same name as an existing file.

DOS 5.0 and 6.1 users can display an EDLIN help screen by keying a **?** at the EDLIN prompt. The displayed information is valid for all versions of EDLIN:

```
*?
Edit line                       line#
Append                          [#lines]A
Copy                            [startline],[endline],toline[,times]C
Delete                          [startline][,endline]D
End (Save file)                 E
Insert                          [line]I
List                            [startline][,endline],L
Move                            [startline],[endline],tolineM
Page                            [startline][,endline]P
Quit (throw away changes)       Q
Replace                         [startline][,endline][?]R[oldtext]
                                [Ctrl+Znewtext]
Search                          [startline][,endline][?]Stext
Transfer                        [toline]T[drive:][path]Filename
Write                           [#lines]W
```

Study the EDLIN help screen displayed above to review the commands you have used so far. (Remember, parameters shown in brackets are optional.) Then key the list command (L or l) to list the file and display the numbered lines you will be editing.

■ ***L** [Enter]

To change to a specific line, first make it the current line by keying its line number at the * prompt in the command area. To change to line 1, key:

■ ***1** [Enter]

When you select a line for editing, two versions of the line are displayed. EDLIN lists the line as it currently exists and, under that line, displays a blank line where you can key the line as you want it to appear.

```
1:*Today is Monday, the day after Sunday.
1:*
```

At the blinking cursor on the bottom line, key the bold type below:

```
1:*Today is Monday, the day after Sunday.
```

■ 1 : ***Today is Monday, the day following Sunday.** [Enter]

Now list the file to see that line 1 reflects your change.

Inserting Lines

Insert a blank line before line 1. To do this, key at the * prompt in the command area the line number 1 followed by the letter i (for insert).

■ **1i** [Enter]

At the 1:* in the text area, enter the following text:

■ 1:***Days of the week move by very quickly.** [Enter]

Exit the insert mode by pressing Ctrl-Z and [Enter] at the beginning of line 2. List your file to see that the new line 1 is inserted and the remaining lines are renumbered.

To insert a blank line just above line 2 of your text, at the command prompt key the line number 2 followed by the letter i (for insert).

■ ***2i** [Enter]

At the 2:* in the text area, press the [Enter] key.

■ 2:* [Enter]

Press Ctrl-Z and [Enter] at the beginning of line 3 to return to the * prompt in the command area. When you list the file, there is a new blank line 2 between lines 1 and 3, and the remainder of the lines are renumbered.

Appending Lines

When you want to append (add) lines to the end of a file, at the command prompt key the pound sign (#) and i (for insert).

■ ***#i** [Enter]

This takes you to the next blank line at the end of the file. Key the following in the text area:

■ 9:***Saturday--the weekend is here!** [Enter]
■ 10:***Sunday--are you ready for Monday?** [Enter]

Press Ctrl-Z and [Enter] at the beginning of line 11 to return to the command area.

> **Remember:** You can also use Ctrl-C or Ctrl-Break to exit from the Insert mode. Whenever you press Ctrl-C while on a line, you cancel any changes made to that line. If you press Ctrl-C on line 10, for example, you would cancel the entry you just made on that line. For this reason, you should go to a blank line to exit the Insert mode with Ctrl-C as you must with Ctrl-Z.

Deleting Lines

Before deleting lines, always list them to make certain you are entering the correct line numbers. To delete a line, at the * command prompt key the line number followed by the delete command initial letter **D** (or **d**).

To delete line 4, key the following at the * prompt:

■ ***4D** [Enter]

To see the revised file with the line deleted, list the file. The former line 4 has been deleted and the remaining lines have been automatically renumbered. Line 4 is marked with the * indicating that it is the current line.

It is a good practice to always list a file after you have deleted lines and before you go on to the next step. In this way, you can be certain that you are using the correct line numbers when you issue the next command.

To delete several adjacent lines, key this sequence: the number of the first line to be deleted, a comma, the number of the last line to be deleted, and the command initial D or d. Press [Enter] to carry out the command.

Delete lines 1 and 2 with this command:

■ ***1,2D** [Enter]

List the file. The first two lines have disappeared and the remaining lines are renumbered.

Now exit the file and save it with the end edit command (E or e). Display a listing of the COMM directory to see the file listed. Notice that EDLIN also saved the earlier version of the file as TEXTMSG.BAK. Each time you edit a file with EDLIN, the earlier version is saved as a backup and is renamed with a BAK extension.

Copy the TEXTMSG.DOC file to the printer so that you can see the results of all your hard work!

Practice: Creating and Revising Text

1. Create this file named TEXTLTR.DOC in the COMM directory. Key:

■ A:\COMM>**EDLIN TEXTLTR.DOC** [Enter]

You will see:

```
*
New file
```

Key **I** for Insert. Continue with the document as follows:

- 1:*(Your Name) [Enter]
- 2:*(Your Address) [Enter]
- 3:*(Your City, State, and ZIP) [Enter]
- 4:* [Enter]
- 5:* [Enter]
- 6:*Trabuco Valley College [Enter]
- 7:*1311 Alicia Allyson Avenue [Enter]
- 8:*Orland Park, IL 60462 [Enter]
- 9:* [Enter]
- 10:* [Enter]
- 11:*Dear Friends: [Enter]
- 12:* [Enter]
- 13:*I would like to attend your college this [Enter]
- 14:*fall. My brother attends now and he is very [Enter]
- 15:*pleased with the classes, particularly those [Enter]
- 16:*in the computer department. [Enter]
- 17:* [Enter]
- 18:*Please send me the schedule of classes for [Enter]
- 19:*the coming fall semester. [Enter]
- 20:*Thank you, [Enter]
- 21:* [Enter]
- 22:*(your name) [Enter]

2. Save and print the file.

   a. End editing and save the file.

   b. Copy TEXTLTR.DOC to the printer.

   c. Redirect a listing of the COMM directory to the printer.

3. Edit TEXTLTR.DOC as follows. Remember to list the file after each change.

   a. Delete one blank line after the college address.

   b. Add one blank line before the complimentary close, "Thank you,"

   c. Exit the edit mode and save the file.

   d. Make a copy of the file for use in the next activity. Name the copy TEXTLTR1.DOC.

4. Compose three of your own revisions to TEXTLTR1.DOC:

a. Append a line at the end of the letter giving your title as "Student."

b. Delete a line.

c. Insert several lines of text in the body of the letter. Press [Enter] to end each line before it wraps.

d. Save and print the revised file. Mark the revisions.

5. Compose a short text file named MYFILE using EDLIN. Remember to end each line before it wraps. Print the file.

6. Revise the file you created in step 5 and print it. Mark the revisions. Redirect a listing of the COMM directory to the printer.

EDLIN COMMAND SUMMARY CHART

EDLIN Command	Examples	Action
I or **i** (insert)	**2i**	Inserts a line at line 2, renumbers lines below
#	**#i**	Appends a line at end of file
L or **l** (list)	**1L** **2,8L**	Lists file beginning at line 1 Lists lines 2 through 8
D or **d** (delete)	**5d** **4,7d**	Deletes line 5, renumbers lines below Deletes lines 4 through 7, renumbers lines below
E or **e** (end edit)	**e**	Exits from EDLIN and saves file
Q (quit)	**q**	Quits (aborts) edit, does not save file

Revising Files with a BAK Extension

As you learned, EDLIN stores a backup file when you revise an existing file. For example, when you recalled TEXTMSG.DOC, revised it, and stored the revised version by keying the end edit command (E or e) and pressing [Enter], the original version was not replaced, but was left unchanged on the disk. When you saved the new version, the original version's file name extension

was changed to BAK (for backup). The new version was stored as revised, but with the original's filename and extension, TEXTMSG.DOC.

There is one important thing to remember about the backup file. EDLIN will not access any file with a BAK extension. If you wish to edit the backup file, *you must rename the file*. For example, to revise TEXTLTR.BAK, you could rename it TEXTLTR.OLD. The exception to this is PC-DOS version 3.3, which will load BAK files into EDLIN.

THE DOS EDITING KEYS FOR EDITING IN EDLIN

The DOS editing keys work exactly the same way for editing in EDLIN as they do for editing DOS commands. As you have seen, the EDLIN commands operate on *complete lines* within a file or document. The function keys and other editing keys are used to edit *within a line*.

There is a special section in the computer's memory called an input buffer. The last line you enter from the keyboard is retained in this buffer when you press [Enter]. When you are working in EDLIN, the current line is stored in the buffer. EDLIN identifies the current line for you by displaying an asterisk (*) after its line number. Because the current line is in the input buffer, you can use it as a template or pattern for editing.

Use the DOS editing keys to repeat or modify lines in EDLIN just as you use them to repeat or modify DOS command lines. At the EDLIN command prompt, key the number of the line you wish to revise and press [Enter]. EDLIN will display the requested line as a template line with a blank line below it. The DOS editing keys operate on the template line in the following ways:

F1 The F1 key copies one character at a time from the template line in the buffer and displays it on the screen. For example, to copy the first part of the line, press **F1** until you reach the point of change.

F2 The F2 key copies the line up to the character you specify. For example, in the line "The quick brown fox jumps over the lazy dog," to change the word fox to cat, press **F2** and the letter "**f**" in the word "fox." This will copy "The quick brown ." Key "**cat**"; then press **F3** to complete the line.

F3 The F3 key copies the entire current line or the rest of the line from the buffer to the display screen. For example, to add something to an existing line, press **F3** to copy the entire line. Then simply key what you want to add to the line. Or, if you have made a revision in the beginning of a line

and want to use the rest of the line unchanged, press **F3** to quickly copy the remainder of the line.

F4 The F4 key deletes letters up to the character you specify after pressing **F4**. In the line "The quick brown fox jumps over the lazy dog," if you press **F4** and the letter "**b**," the characters up to b in "brown" will be erased from the buffer. When you press **F3**, the line will begin "brown fox jumped . . ."

Right and Left Arrow Keys The right and left arrow keys can also be used. The right arrow key acts like the F1 key, copying one character at a time from the buffer onto the display screen. The left arrow key acts like the Backspace key, erasing one character at a time and moving to the left. The character erased remains in the buffer, however.

Delete A pointer accesses the same place on the template line in the input buffer as the cursor's position in the current line on the screen. Each time you press the Del key, one character showing on the display screen is deleted from the buffer. No change appears on the screen until you press **F1** or **F3** to display the next character(s) in the line. The deleted characters do not appear.

Insert When you press the Ins key, the next character(s) you key will be placed on the screen and in the buffer at the cursor location, and all following characters in the line will move to the right. The insert mode is turned off by pressing any function key, any other DOS editing key, or [Enter].

Practice: Using the DOS Editing Keys to Edit a File

Retrieve the file TEXTLTR.DOC. It should be stored in the COMM subdirectory. This time, rather than changing to that directory, use the file's pathname to tell the system where the file is:

- **C:\>EDLIN A:\COMM\TEXTLTR.DOC** [Enter]

If all goes well, you will see this message:

```
End of input file
*
```

If not, check to see if you keyed the path name and file name correctly. If your entry is correct, next check the COMM directory to make sure the TEXTLTR.DOC file is there.

List the file by keying **L** at the EDLIN prompt. Then with EDLIN commands and the DOS editing keys, make the changes described in the following steps.

Use **F2** and **2** to change the ZIP in the inside address for the college to 60464. To do this, first make line 8 the current line by keying **8** and pressing [Enter] at the * command prompt. EDLIN displays line 8 with a blank line below it.

```
8: Orland Park, IL 60462
8:*
```

Press **F2** and **2** to copy line 8 up to line 2. Then key **4** to change the ZIP to 60464. Press [Enter] to return to the command prompt.

Now make line 10 the current line by keying **10** at the * prompt and pressing [Enter]. When line 10 displays, press **F1** five times to copy "Dear" and the space. Key a new salutation, "**Ladies and Gentlemen:**" and press [Enter].

Make line 12 the current line. Press **F1** twice to copy the "I" and the space; then change the rest of the first line of the letter to:

- I **hope to enroll in and attend your college** [Enter]

Make line 12 the current line by pressing [Enter] again. Press **Ins** and key the word "this" and a space at the beginning of the line. Press **F3** to copy the remainder of the line. Press [Enter].

List the file. Key the text in bold type below after "semester." on line 18. At the * command prompt, key **18** and press [Enter] to make line 18 current. Press **F3** to copy the line and then add the text shown in bold to the line. Press [Enter] to return to the * prompt. Key **19i** to insert new lines at line 19. Press [Enter] to start inserting the six new lines.

- the coming fall semester. **I would also** [Enter]
- **appreciate a list of graduation requirements.** [Enter]
- **A college education has long been one of my** [Enter]
- **goals. I will, no doubt, need counseling** [Enter]
- **since I do not know what I would like to do** [Enter]
- **when I graduate. Please recommend a career** [Enter]
- **counselor for me to see.** [Enter]
- **[Ctrl-Z]** [Enter]

List the file. Insert two more blank lines for your signature by keying **28i** at the * prompt and pressing [Enter] three times (once to go to the text area, twice to add two lines). Press Ctrl-Z and [Enter] to exit from the insert mode.

List the file again. Divide the second paragraph into two paragraphs. First insert a line above the line that begins "goals." Key "**goals.**" on this line and press [Enter] twice. Exit the insert mode on the second blank line.

List the file. Next edit the first line of the new paragraph. Press **F4** and the uppercase letter **I** to delete the word "goals." Press **F3** to complete the revised line. Press [Enter] to return to the * prompt.

List the file to proofread it. The revised file should look like the following:.

```
1:   Your Name
2:   Your Street Address
3:   City, State  Zip
4:
5:
6:   Trabuco Valley College
7:   1311 Alicia Allyson Avenue
8:   Orland Park, IL  60464
9:
10:  Dear Ladies and Gentlemen:
11:
12:  I hope to enroll in and attend your college
13:  this fall.  My brother attends now and he is very
14:  pleased with the classes, particularly those
15:  in the computer department.
16:
17:  Please send me the schedule of classes for
18:  the coming fall semester.  I would also
19:  appreciate a list of graduation requirements.
20:  A college education has long been one of my
21:  goals.
22:
23:  I will, no doubt, need counseling
24:  since I do not know what I would like to do
25:  when I graduate.  Please recommend a career
26:  counselor for me to see.
27:
28:  Thank you,
29:
30:
31:  Your Name
```

Notice that the file is now longer than 23 lines (a screenful). When you use the L command alone, EDLIN displays the current line with the 11 lines before it and the 11 lines after it. To display the entire file in two passes, list the file with the following two commands:

- ***1,10L** [Enter]
- ***11L** [Enter]

This enables you to see the entire file. You can use other line number combinations to accomplish this, of course.

When you are satisfied with your letter, press **E** to exit and save the file. Then use the COPY or TYPE command to print the letter.

Your introduction to the first of the two text editors is completed. Test your comprehension of the material by completing the Self-Check Quiz and Assignment before going on to the next chapter on EDIT.

CHAPTER 8 SELF-CHECK QUIZ

True/False (Write T if the statement is true, F is the statement is false.)

_____ 1. EDLIN can be used to edit files created with COPY CON.

_____ 2. EDLIN recognizes commands keyed in uppercase or lowercase letters.

_____ 3. The EDLIN command can be keyed either before or after the line number(s).

_____ 4. EDLIN line numbers can be displayed with the TYPE command.

_____ 5. EDLIN, EDIT, and COPY CON create files in ASCII format.

Multiple Choice (Circle the correct answer.)

6. Which EDLIN command would you use to insert a line at line 6?

a. i6
b. 5i
c. 6
d. 6i

7. Which EDLIN command would you use to remove line 10?

a. r10
b. 10r
c. 10d
d. d10

8. Which EDLIN command would you use to revise line 4?

a. 4
b. 4c
c. 4*
d. 4r

9. Which function key would you use to copy all characters up to the letter keyed?

a. F1
b. F3
c. F2
d. F4

10. If you were editing the line
1:*Employer: Ace Computers
and pressed F4, A, and F3, what would the result be?

a. Employer
b. Ace Computers
c. Employer: A
d. A

ASSIGNMENT

To practice using the commands in this chapter, complete the following steps. Check off each step when you understand its operation, or mark the step with a question mark if it is not clear. Write the commands or keystrokes in the space provided.

1. Create a new EDLIN file named WINNERS.TXT on your TREE DISK in the COMM subdirectory. Input the following text:

Mr. and Mrs. John Brown
111 Main Street
San Diego, CA 92706

Dear Mr. and Mrs. Brown:

Today you have been declared the winners of the lottery sponsored by the Main Street Camera Store.

Your winnings will be announced later in the week but I can assure you that you will be very happy with the amount of your check.

Please bring the lottery ticket to the counter when you come and we will be happy to help you carry your winnings home or to the bank.

Sincerely,

Your friendly sales clerk,

2. Print a copy of the file.

---

3. Revise the file as follows:

   Add your name under the "friendly sales clerk," line.

---

4. Change the name and address in the letter to your name and address.

---

5. Change the salutation to CONGRATULATIONS! Save and exit the file.

6. Print the revised letter.

7. List only the first 15 lines of the letter. Print the screen showing the command line.

8. Delete the last paragraph. Print the screen showing the command line.

9. Revise the letter again and add the date at the top of the letter and the following paragraph at the end of the second paragraph.

   As you know, as the sellers of your lottery ticket we are entitled to a percentage of your winnings. Thank you for purchasing the winning ticket from our store.

10. Save and print the file.

CHAPTER 9

AN INTRODUCTION TO EDIT, THE DOS TEXT EDITOR

OBJECTIVES

When you have completed the activities and assignment in this chapter, you will be able to:

1. Use EDIT to create, revise, and save text files.

2. Open and use EDIT menus and dialog boxes.

3. Select, insert, delete, copy, cut, and paste text.

4. Find and change words and phrases.

5. Print text files.

6. Look up information in EDIT HELP screens.

THE DOS EDITOR

The DOS Editor is a full-screen text editor. It is a simple text editing program that allows users to easily create, revise, and print short documents. The Editor can be used in the Shell or started at the DOS prompt with the command EDIT. The Editor can be used to revise all text files created with COPY CON and with other text editors such as EDLIN, E, and Notepad.

The EDIT command requires access to the QBASIC program that is bundled with DOS as well as EDIT.COM. EDIT utilizes the part of the QBASIC code that is used for editing and writing BASIC programs. You will not be aware that you are using QBASIC and you do not need to know any programming to work with EDIT. EDIT also requires access to EDIT.HLP, a file containing help information for the Editor.

Although the Editor is much more flexible than the older methods of creating and revising text files, COPY CON and EDLIN, it still lacks the power of a true word processor. For example, you cannot set margins or custom tabs, control line spacing, or spell check. The Editor does not have the word-wrap capability found in a true word processing program. Nevertheless, EDIT is

extremely useful when you need to create a short document in ASCII or to write or revise a batch file. Batch files are the subject of Chapter 10.

INSTALLING THE MOUSE

As you learned in Chapter 1, the mouse is an excellent input device. When working with EDIT, you may choose to work with a mouse or the keyboard.

Depending on the way the computer has been set up, the mouse may or may not be automatically installed when you boot your computer. If the mouse is not active, install it by keying at the DOS prompt:

MOUSE	(Assuming MOUSE is the appropriate driver name for the mouse device.)

DOS will display a message verifying that the mouse has been successfully installed.

Practice: Installing the Mouse

If necessary, install the mouse as follows:

- **MOUSE** [Enter]

Other than the verification of successful installation of the mouse, you will see no indication of the mouse on the screen.

USING EDIT

Since EDIT is an external command, you will either need to change the directory to the DOS subdirectory or have the path set to the DOS files in order to use EDIT. The command to load EDIT is:

```
C:\>EDIT
```

> **Note:** EDIT is not available in PC-DOS 6.1.

Practice: Loading EDIT

Use the TREE DISK diskette in this exercise. Access the COMM directory on the disk in drive A to work from with the EDIT program. To start the Editor, key the following from the DOS prompt:

■ **EDIT** [Enter]

USING THE MOUSE

When the mouse is installed on your computer and you are in the EDIT program, move the mouse to see if it is active. You should see a small rectangle move on the screen when you move the mouse. This is the mouse pointer.

The **scroll bars**, scroll arrows, and scroll boxes are used to scroll the screen with the mouse (see Figure 9-1). There are several actions you will use with the mouse.

They are:

- Point — Move the mouse to move the small rectangle or arrow that moves on the screen when you move the mouse. The mouse is used to point at objects such as scroll bars, menu names, icons, or other objects on the screen.
- Click — Press and release a mouse button, usually the left button.
- Double-click — Click the mouse button twice and quickly release.
- Drag — Hold down the mouse button and move the mouse pointer to a different place.

You will work with only the left button on the mouse in EDIT.

Features of the EDIT Screen

If you do not specify a file name after EDIT, the first screen you see is empty except for a dialog box asking if you want to see the "Survival Guide" or go directly into the EDIT program. The Survival Guide is the Help menu for EDIT (see Figure 9-1). At this point, press Esc to clear the dialog box and go to the EDIT screen.

You will see a blank editing area with a **menu bar** above it listing five options: File, Edit, Search, Options, and Help. The document title appears at

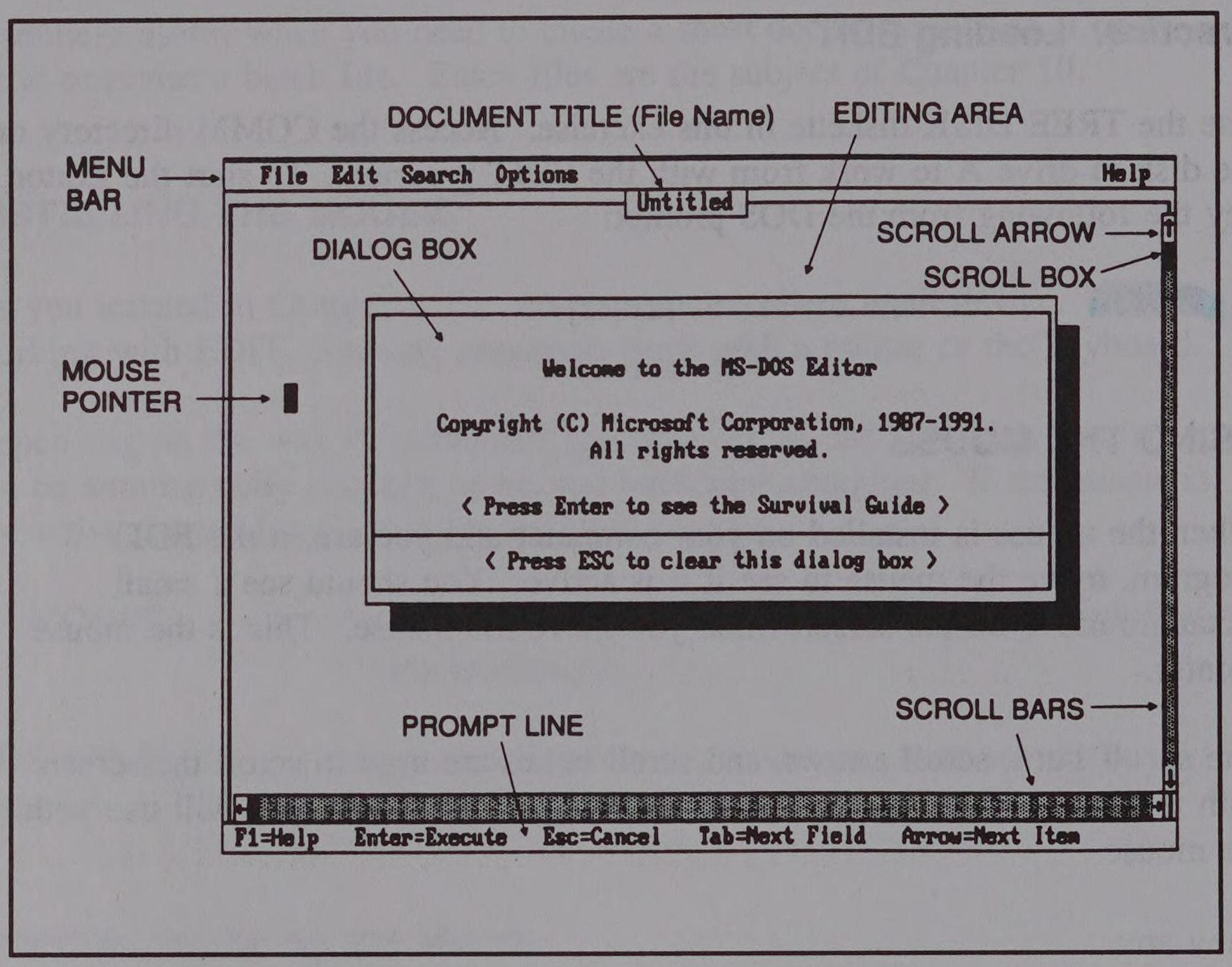

Figure 9-1. EDIT screen.

the top center of the editing screen. At this time the document is "Untitled." At the bottom of the screen there is a **prompt line**. This line provides help by reminding you of the keystrokes you may want to use next. Its contents change as you perform different operations.

Practice: Creating a File Using EDIT

To provide practice in creating files in specific directories, create the following file in the COMM directory. You should be at the EDIT screen. Press Esc to clear the full-screen display and start your document by keying the following text on one line:

- **I am creating this file to test the features of the new DOS editor.**

Revise the line as follows:

- Insert the word **text** in front of the words "file" and "editor" as follows:

  Press the left arrow key enough times to position the cursor under the "e" in "`editor,`" key the word **text** and press the spacebar once to create a space.

You can also position the cursor with the mouse by moving the mouse pointer to the place where you want the cursor to be and clicking (pressing) the left button.

- Next, hold down the Ctrl key and tap the left arrow key (Ctrl-Left) to move the cursor one word to the left. Continue to press Ctrl-Left until the cursor is on the "f" in "file." (If you go too far, use the Ctrl and the right arrow keys (Ctrl-Right) to move one word to the right.) Key the word **text** and press the spacebar once to create a space. The sentence should now read:

```
I am creating this text file to test the features of
the new DOS text editor.
```

- Delete the word "new" in the sentence. Use Ctrl-Right or the mouse to position the cursor under the "n" in "new." Press the Del key four times (once for each character in the word and the following space).

The sentence now reads:

```
I am creating this text file to test the features of
the DOS text editor.
```

Practice: Saving and Exiting from EDIT

You have keyed text and made minor revisions. Now save the file. Notice the message in the prompt line at the bottom of the screen:

```
Press ALT to activate menus
```

Five menu names are listed in the menu bar across the top of the screen: File, Edit, Search, Options, and Help. Pressing the Alt key and the first character of the menu name (F, E, S, O, or H) will open that menu. Try this by pressing

- **Alt**

The File menu name is highlighted or selected. (See Figure 9-2.) Use the right arrow key to move the highlight and select the other options in the menu bar. The initial letters of the menu names become bright white as you move from one option to another.

To save the file, open the File menu by pressing **F**. (Press **Alt** first if the menu initial letters are no

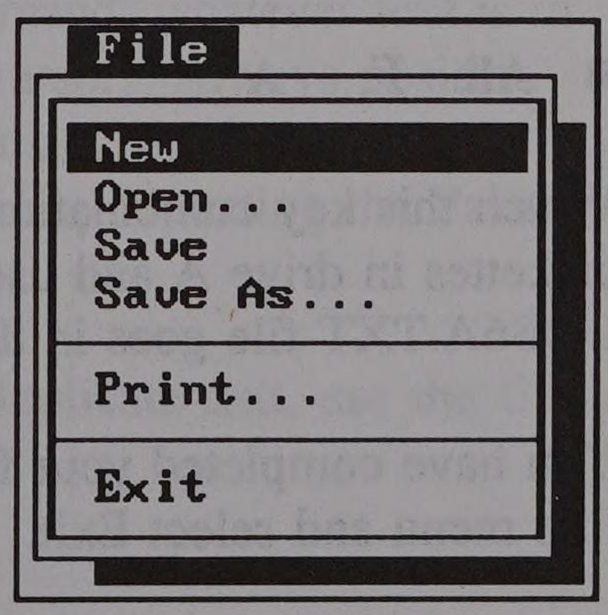

Figure 9-2. File menu.

longer bright white.) You can also open the menu with the mouse by pointing at the File menu name and clicking the left button.

A **menu** contains a list of command options. The options available in the File menu are shown in Figure 9-2. "New" is highlighted. Use the down arrow key to move the highlight down through the options. Watch the prompt line at the bottom of the screen for a brief explanation of what each option in the menu does.

Notice that several of the options have an ellipse (...) following them, such as "Save As..." If a menu option has an ellipse, a **dialog box** will display when the command is chosen. A dialog box allows you to key additional information or make other choices required by the command.

Two options exist for saving a file, Save and Save As. When the file is Untitled, you can use either option. The different color (or bold) letter indicates the keyboard letter that activates the option. To activate the Save option, key **S**; to activate the Save As option, key

- **A**

With the mouse, click the left button anywhere on the Save As option. The Save As dialog box is displayed. Press **F1** to see a Help screen that identifies the parts of the dialog box. Press **Esc** to remove the Help screen.

Below "File Name:" the current directory A:\COMM is shown. Since this is where you want the file to be stored, you can simply key the file name at the blinking cursor. Key

- **DOS6A.TXT** [Enter]

The file is written to disk and the dialog box is removed. Notice that the file name DOS6A.TXT has replaced Untitled at the top of the screen. It is wise to form the habit of backing up all of your work; start by backing up this document on the CLASS diskette. Press the following keys

- **Alt F A**

Watch this key combination, choose the File menu and Save As. Exchange the diskettes in drive A and use the same name you used previously. Note that the DOS6A.TXT file goes in the root directory of the destination diskette.

You have completed your first EDIT document. To exit from EDIT, open the File menu and select Exit. Use the mouse or these keystrokes:

- **Alt F X**

If you had not saved the file, you would be prompted to do so before exiting EDIT.

SELECTING, COPYING, CUTTING, AND PASTING TEXT

If you are not familiar with word processing, selecting, copying, cutting, and pasting text into new locations may be new to you. The following descriptions explain each of these operations briefly. See Figure 9-3.

Selecting Text: Before text can be copied, cut, or deleted, you must **select** it. In EDIT and most other editing software, text is selected by using keystroke combinations or a mouse. You will learn the keystrokes for selecting text in EDIT. When text is selected, it appears in reverse video or the opposite color from the surrounding text.

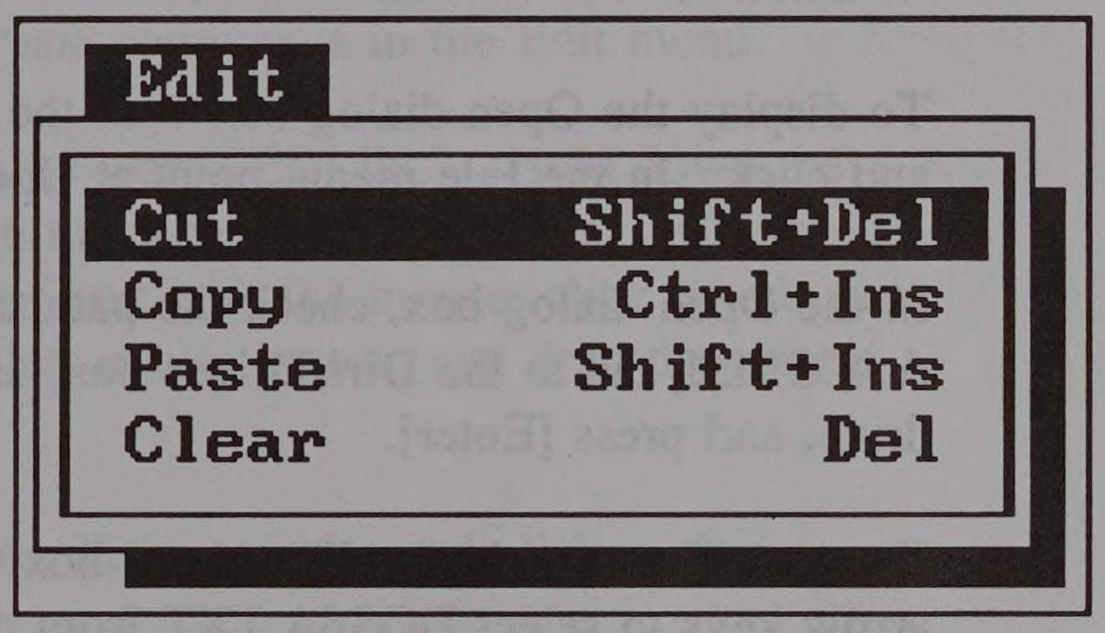

Figure 9-3. Edit menu.

Copying Text: When you use the EDIT **Copy** command to copy text, a duplicate of the selected text is placed in a memory buffer. The copied text replaces any other text that might have been placed in the buffer by an earlier Copy or Cut command. The selected text on the screen does not change.

Cutting Text: When you use the EDIT **Cut** command, the selected text is placed in the memory buffer, replacing any other text that might be there. The selected text is removed from the screen.

When you remove text from the screen with the Delete key or the EDIT Clear command, it does not go into the memory buffer.

Pasting Text: When you use the EDIT **Paste** command, whatever text is in the memory buffer is inserted at the location of the cursor. Always position the cursor where you want the text to appear before using the Paste command. Multiple copies of the text in the buffer can be pasted by repeating the Paste command.

To move text, use the Cut and Paste commands; to duplicate text, use the Copy and Paste commands.

Practice: Revising a Text File

Practice using the Cut, Copy, and Paste commands by revising the text file DOS6A.TXT now. Be sure you are in the COMM directory. Reload the file by keying EDIT DOS6A.TXT at the prompt, or you can select the file from a list displayed in EDIT. Try that method now. At the DOS prompt, key

- A:\COMM>**EDIT** [Enter]

At the EDIT opening screen press Esc. To open an existing file, open the File menu and select Open. Use the mouse or these keystrokes:

- **Alt F O**

To display the Open dialog box with the mouse, point at File in the menu bar and click. In the File menu, point at Open and click.

In the Open dialog box, check the pathname under File Name:. If it is not A:\COMM, tab to the Dirs/Drives box, use the arrow keys to select the correct drive, and press [Enter].

Key the file name in the File Name box, or tab to the file list and use the arrow keys to select DOS6A.TXT from the list. When the file name is highlighted, press [Enter] to open the file.

You can open the file with the mouse by double-clicking on its name. (A double click is two rapid clicks of the left button. The second click is the equivalent of pressing [Enter].)

A file in another drive or directory can be opened by keying its filespec (drive, pathname, and file name) in the File Name box and pressing [Enter].

You should recognize the file you created earlier. Move the cursor to the second line. Key two more lines of text; at the end of each line, you must press [Enter]. Remember that EDIT is only an editing program, not a word processing program. It does not word-wrap at the right margin (that is, automatically move down to the next line). Key the following text on two lines:

- **I will now learn other methods for editing and manipulating text.**[Enter]
- **Before DOS 5.0 I used a different method: EDLIN.** [Enter]

Your file should now resemble the following:

```
I am creating this text file to test the features of
the DOS text editor.
```

```
I will now learn other methods for editing and
manipulating text.
Before DOS 5.0 I used a different method:  EDLIN.
```

Save the file as DOS6B.TXT in the COMM directory with these keystrokes:

- **Alt F A DOS6B.TXT** [Enter]

Practice: Cutting and Pasting Text

Let's improve on the text in the third sentence by moving "EDLIN" to the right of "used." To do this use the Cut and Paste commands in the Edit menu.

Move the cursor under the space before the word "EDLIN." Hold down the Shift key and press the right arrow key 6 times to select or "block" EDLIN (without the period). To select a block of text with the mouse, click where you want the block to begin, hold down the left mouse button, and drag the mouse to where you want the block to end.

Next open the Edit menu. Press **Alt E** for Edit, or point and click with the mouse.

The options available in the Edit menu are shown in Figure 9-3. Use the down arrow key to select each option and watch the prompt line for a brief description of what it does.

Key **T** or click on the Cut option. Notice that the highlighted text on the screen disappeared. It was moved to the memory buffer, waiting for your next action.

Move the cursor to the space between "used" and "a." Then open the Edit menu and select Paste with these keystrokes:

- Alt **E P**

or open the menu and click on the Paste option with the mouse.

The block of text you cut will be inserted or pasted in the line which now reads:

```
Before DOS 5.0 I used EDLIN a different method:
```

At this point you will need to place the cursor on the space following EDLIN and insert a "," and then move the cursor under the ":" and press **Del** enough

times to delete the ":" and the spaces in front of the period. The sentence should now read:

```
Before DOS 5.0 I used EDLIN, a different method.
```

Practice: Copying Text

To complete the last sentence, copy the phrase "for editing and manipulating text" from the second line and place it after "a different method" and before the period (.) in the third line.

Select the block of text by placing your cursor under the "f" in the word "for." Hold down the Shift key and use the right arrow key to highlight the phrase "for editing and manipulating text" up to the period. (Or use the mouse and highlight the phrase by dragging the mouse.)

Open the Edit menu and select Copy with these keystrokes:

- **Alt E C**

Position the cursor on the period at the end of the sentence in the third line. Paste the copied phrase from the memory buffer with these keystrokes:

- **Alt E P**

or click on the Paste option with the mouse.

You will need to place the cursor on the "f" in "for" and press the spacebar once for appropriate spacing. If any line extends beyond the right edge of the screen, press the End key to view the right end of the line. To scroll back to the left, press the Home key.

If any lines are too long, shorten them by moving the cursor to the beginning of the word of the text you would like to move to the next line and press [Enter].

The text should now read:

```
I am creating this text file to test the features of
the DOS text editor.
I will now learn other methods for editing and
manipulating text.
Before DOS 5.0 I used EDLIN, a different method for
editing and manipulating text.
```

Save the file as DOS6C.TXT with these keystrokes:

- **Alt F A DOS6C.TXT** [Enter]

PRINTING A FILE

With the EDIT **Print** command you are able to print the current file directly from EDIT. Open the file menu and select Print with these keystrokes:

Alt F P

The Print dialog box is displayed. Note that you can choose to print "Selected Text Only" or the "Complete Document." If you had selected a block of text in the file, you could print that portion of the file with the first option. At this time you want to print the complete document. The dot within the parentheses shows that Complete Document is selected. Press [Enter] to begin printing.

Since the document is saved, you can elect to exit the EDIT program or continue with the next activity.

Practice: Printing a File

With the file DOS6C.TXT on the screen, print the file using the mouse or the following keystrokes:

- **Alt F P**

USING THE SEARCH MENU

In this activity, you will use the Search menu (see Figure 9-4) to find words in a text file. EDIT's search function provides several alternative methods of finding or finding and changing text. You can find one or more characters or a whole word. The Search menu allows you to find, repeat the previous find, and change.

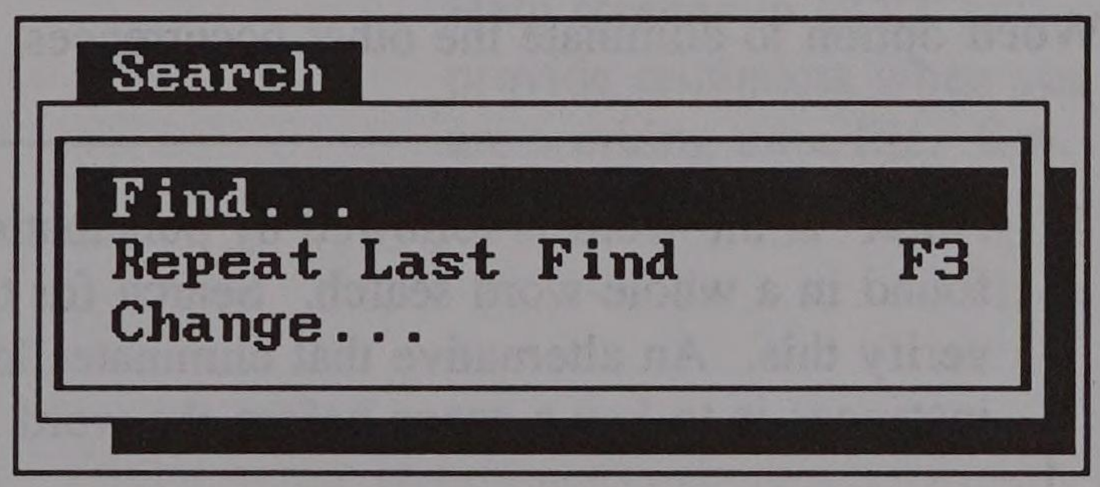

Figure 9-4. Search menu.

Practice: Using the Search Menu

If necessary, load the last file, DOS6C.TXT, into EDIT by keying this command line:

- A:\COMM>**EDIT DOS6C.TXT** [Enter]

When you enter the file name as a parameter to the EDIT command, EDIT loads the file and it appears in the opening screen.

Open the Search menu by pressing **Alt S**. "Find..." is highlighted. Use the down arrow to select each option and read the description in the prompt line.

Press [Enter] to select Find. In the dialog box, key the word **text**, and press [Enter]. The cursor moves to an occurrence of the word "text."

Open the Search menu and select Repeat Last Find (or press **F3**); the cursor moves to the next occurrence of "text." Press **F3** repeatedly and you will see that the cursor moves in a circle from occurrence to occurrence of the word "text."

Now try something different. Open the Search menu and press [Enter] to select Find. In the Find dialog box, key "**to**" and press [Enter]. Press **F3** repeatedly until the cursor stops at "to" in the word "editor." Press the Home key to move the cursor back to the beginning of the line.

Access the Find dialog box again and key "to" as the word to find. This time, press Tab twice to move the cursor to [ ] Whole Word.

Press the spacebar to activate this option. An X will appear within the brackets.

Press [Enter], then **F3**. Notice that now the cursor ignores the "to" in editor and stops only at the word "to" where it appears in the text. When you want to search for a word and not for a combination of letters, select the **Whole Word** option to eliminate the other occurrences.

> **Note:** If the word is followed by punctuation, as in "text". it will not be found in a whole-word search. Search for the whole word "text" to verify this. An alternative that eliminates letter combinations in many instances is to key a space before the word in the Find What: entry.

Your final task is to find words and change them to other words. This is similar to the "search and replace" function of many word processors. To find and change words, access the Change option in the Search menu by keying

- **Alt S**

Press down arrow to select Change.

In the Find What: box, key

- **editing and manipulating**

Press Tab. In the Change To: box, key

- **finding and changing** [Enter]

You will next see the Change dialog box with options to Change, Skip, Cancel, or Help. With the cursor on Change, press [Enter] to confirm the change. The next occurrence of "editing and manipulating" was changed. If, however, the phrase extended over two lines, EDIT would not replace the phrase.

You can specify whether the search should be case-sensitive—that is, match upper- and lowercase letters in a word or phrase. Since you did not select Match Upper/Lowercase in the Search or Change dialog box, you can key all lowercase or a mixture of upper- and lowercase and the results will be the same. Save the file as DOS6D.TXT.

USING HELP

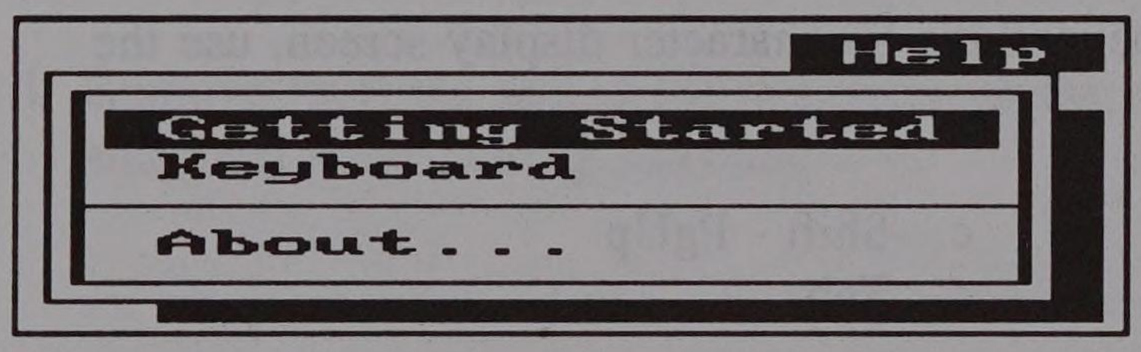

Figure 9-5. HELP menu.

You have used many keystroke combinations for moving the cursor and selecting text. Can you remember them all? The Help screens in EDIT will provide reminders when you are working on a file. See Figure 9-5.

Practice: The Help Menu

To access the Help menu, press

- **Alt H**

Use the down arrow key to select each command option, and read the prompt line at the bottom of the screen for a brief description of what each option does.

Nice work on EDIT! Test your knowledge of the information by completing the Self-Check Quiz and Assignment.

CHAPTER 9 SELF-CHECK QUIZ

True/False (Write T if the statement is true, F if the statement is false.)

_____ 1. EDIT is an external command and must have access to the EDIT.COM and QBASIC.EXE files.

_____ 2. It is possible to open a menu by pressing the Ctrl key and then the highlighted initial letter for that menu.

_____ 3. To copy selected text in EDIT, press Alt E C or Ctrl-Ins.

_____ 4. HELP is available only when you first load EDIT.

_____ 5. The function keys operate in EDIT just as they do at the DOS prompt.

Multiple Choice (Circle the correct answer.)

6. Which combination of operations is used to move selected text?

a. copy and paste
b. delete and paste
c. cut and paste
d. clear and paste

7. To view text that extends beyond the 80-character display screen, use the following key(s):

a. Alt - Right
b. End
c. Shift - PgUp
d. Tab

8. If Whole Word and Match Upper/Lowercase are selected, which of the following will be found when the Find What: entry is "Ton"?

a. Ton and ton.
b. Ton and Tone
c. Ton and ton
d. Ton

9. If you want to copy selected text in order to paste it, use these keystrokes:

a. Alt E or Shift - Del
b. Alt C T or Alt - Ins
c. Alt E C or Ctrl - Ins
d. Alt E T or [Shift - Del

10. To open the Find dialog box, use these keystrokes:

a. Alt E C
b. Alt S F
c. Alt S C
d. Alt C F

ASSIGNMENT

To practice using the commands in this chapter, complete the following steps. Check off each step when you understand its operation, or mark the step with a question mark if it is not clear. Write the commands or keystrokes in the space provided.

1. Install the mouse, if necessary.

2. Access the COMM directory and EDIT.

3. Key and print the following text:

   Selecting Text: Before text can be copied, cut, or deleted, you must select it. In EDIT and most other editing software, text is selected by using keystroke combinations or a mouse. You will learn the keystrokes for selecting text in EDIT. When text is selected, it appears in reverse video or the opposite colors from the surrounding text.

4. Remove (cut) the text at the beginning of the second sentence, "In EDIT and most other editing software,".

5. Capitalize the "t" in text which will now begin the second sentence.

6. Add the following to the beginning of the paragraph after "Selecting Text" and before the colon ":" **in a file created with EDIT.** Adjust the lines appropriately for even printing.

7. Save the file as SELECT.TXT and print the file.

8. Create a letter that you might send to a friend who would like to attend college with you. Tell her or him about your interesting classes in Computer Science. Save the file as LEARN.TXT. Print the file.

9. Retrieve the file LEARN.TXT and make the following change:

   Using the Change feature of the Search menu, change the words "Computer Science" to "Technical Computer Sciences."

10. Print the file.

CHAPTER 10

AN INTRODUCTION TO BATCH FILES

OBJECTIVES

When you have completed the activities and assignment in this chapter, you will be able to:

1. Describe the contents and functions of batch files.
2. Create a batch file.
3. Run a batch file.
4. Use the batch file commands REM, PAUSE, and ECHO.
5. Create a simple AUTOEXEC.BAT file and describe its function.
6. Create a simple CONFIG.SYS file and describe its function.
7. Create a custom menu system.

CREATING SIMPLE BATCH FILES

Batch files are special text files containing a sequence of DOS commands that are performed together to accomplish a task. These user-developed command files must have the extension BAT in order for the system to recognize them as executable files. When you key the filename minus the extension, DOS executes the sequence of commands contained in the file.

Batch files are useful because they enable you to store all the steps required for a complex DOS procedure. Also, you can repeat the execution of the steps innumerable times without rekeying the commands each time. This saves time and avoids errors that might occur when rekeying.

If you display the DOS directory, you will notice that most of the files in the directory have the extension COM or EXE. From your experience with DOS external commands, you know that you can "execute" files like LABEL or FORMAT without keying the extension. In fact, any file from any DOS-compatible software that has either the COM or EXE extension is an executable file and can be loaded into memory and "run" by keying the

filename. For example, to load and run WordPerfect, you key WP, which executes the file WP.EXE to start the WordPerfect program. Batch files work similarly in that you do not have to key the extension.

Batch files are useful because they enable you to store all the steps required for a complex DOS procedure. Also, you can repeat the execution of the steps innumerable times without rekeying the commands each time. This saves time and avoids errors that might occur when rekeying.

An example of a batch file is when you set the search path to the DOS external command files and change the default prompt to show the current drive and directory. To continue with this example, rather than keying the PATH and PROMPT commands each time you boot the computer, you can put the two command lines in a batch file named P.BAT. Here P stands for "path" and "prompt."

```
PATH C:\DOS
PROMPT $P$G
```

Then, by entering the filename P, you can execute the P.BAT file. If the disk storing the batch file is not in the default drive, you must also enter the drive designator so that DOS will know where to look for the batch file. When you key

```
A:\>P
```

the DOS commands are executed to set the new search path and to change the default prompt to display the current drive and directory. Note that it was not necessary to key the extension for P.BAT to be executed.

Note that the filename P could be any legal filename. However, the *extension must be BAT*. This extension tells the system that the file is a batch file and to load it and perform the DOS commands contained within the file in the sequence listed.

DOS command files are executed in the following order: COM, EXE, and then BAT files. When you key a command such as COPYIT at the DOS prompt, DOS first looks for an internal command with that name and, if one is found, executes the command. If it does not find such an internal command, DOS next looks in the current directory for a file named COPYIT.COM, then for COPYIT.EXE, and finally for COPYIT.BAT. If you gave your batch file the same name as a file with the extension COM or EXE, the batch file would never be executed inasmuch as DOS would always execute the DOS COM or EXE command first *if* there is a COM or EXE file with that filename in the directory.

Like other text files, batch files can be created with EDLIN, EDIT, E or the COPY CON technique. Use the COPY CON technique for short files. If you make an error, you can rekey the file or correct it with a text editor. For longer batch files, you may prefer to start with EDLIN, EDIT, or E. If you use a word processor, you must save the file as a DOS text file.

Practice: Creating and Running a Batch File

First insert the TREE DISK diskette and create a BAT subdirectory. Change to that directory by keying:

- A:\>**MD BAT** [Enter]
- A:\>**CD BAT** [Enter]

Now create the batch file described above that will set the search path and change the prompt. Use COPY CON or any text editor. Include the BAT directory you created in the search path so that DOS can find the executable batch files you will place there.

- A:\BAT>**COPY CON P.BAT** [Enter]
- **PATH C:\DOS;A:\BAT** [Enter]
- **PROMPT PG** [Enter]
- **[F6]** [Enter]

To run P.BAT, ensure you are in the BAT directory. Once you have run the P.BAT file, since you set the path to the BAT directory, DOS should have no problem finding the location where P.BAT is stored. Simply key the filename **P** and press [Enter]. If DOS displays a "Bad command or file name" message, make certain that you stored P.BAT in the BAT directory and that you have set the search path.

After you run the file, check the current search path by entering the command

- A:\>**PATH** [Enter]

The path to the DOS and BAT directories is displayed:

```
A:\>PATH C:\DOS;A:\BAT
```

When there are multiple subdirectories in a path statement, each is separated with a semicolon. The semicolon serves only to separate the subdirectories. It is practical to list the subdirectories in the order that you most commonly used their files. Some software, in the installation process, will insert the subdirectory in which its files are located first in the path statement.

Create a second batch file in the BAT directory. Since you just used the batch file P.BAT to set your path, DOS should have no trouble finding the external command files necessary to locate the text editing program files.

The next batch file is one that you might use with a data file that you revise often. It checks to see if the file is in the COMM directory on the disk in the default drive, tells you when the file was last revised, and displays the contents of the latest version. Create the following batch file in the BAT directory:

- A:\BAT>**COPY CON** (or other text editor) **AUTOCHK.BAT** [Enter]

When the display screen appears, key the following:

- **CLS** [Enter]
- **DIR \COMM\DOS6D.TXT** [Enter]
- **TYPE \COMM\DOS6D.TXT** [Enter]

Save the file as **AUTOCHK.BAT** and exit the text editor.

Because the file has a BAT extension, you are able to key the filename AUTOCHK without the extension BAT at the DOS prompt and the system will load and run the file and execute the commands it contains. To run the file, key:

- A:BAT\>**AUTOCHK** [Enter]

You see the screen cleared as DOS executes the CLS command. Next the directory listing for the file DOS6D.TXT in the COMM subdirectory is displayed, verifying that the file is there and showing the date and time when it was last revised. Finally the contents of the file are displayed.

You can rerun AUTOCHK as many times as you wish. If DOS6D.TXT were a file that you revised frequently, this batch file would enable you to call it up for review quickly and easily.

Now you will create a batch file that can be used to obtain a printed copy of the root directory, to see how much free space remains on the disk, and to determine if the disk has any bad or lost sectors.

Create AUTODIR.BAT in the BAT subdirectory. Key the following at the DOS prompt:

- **A:\BAT>COPY CON AUTODIR.BAT** [Enter]
- **CLS** [Enter]
- **CD ** [Enter]
- **DIR /W** [Enter]
- **DIR /W > PRN** [Enter]

■ **CHKDSK** [Enter]

Save the file and exit the text editor. The AUTODIR.BAT file directs the system to clear the screen, to change to the root directory, to display and print a wide version of the directory, and to check the disk. In this batch file you have used one external command, CHKDSK. It is essential to have access to the DOS commands. The path command in the P.BAT file should take care of this.

To run the file, key:

■ A:\BAT>**AUTODIR** [Enter]

Again, keying the BAT extension is unnecessary. Following the sequence of commands listed in the file, DOS clears the screen, changes to the root directory, displays and prints the root directory in wide form, checks the disk, and displays the results. This information does seem to fly by on the screen. Read on for help in correcting this.

BATCH FILE COMMANDS: REM, PAUSE, and ECHO

Special DOS internal commands are used in batch files. Several of these are used to give information or instructions about the batch file. Others give the user control over the execution of commands or base the execution on specific conditions. Often batch files are used by several people. Documentation within the file can help everyone know what the file does and how to use it. The REM (remark) command is used to include useful information in the batch file without affecting the execution of other commands.

Another batch file command that can be used to provide information to the user performs a second function that is even more important. The PAUSE command will halt execution of the batch commands until a key is pressed. When a batch file is run, the screen display of commands and other data can scroll by so quickly that it is difficult, if not impossible, to read what is shown.

The PAUSE command will temporarily halt the display of information, freezing the screen until the user is ready to proceed. The prompt `"Press a key to continue. . ."` appears at the cursor location. You can take any amount of time you wish to read the screen before pressing a key. The PAUSE command can include a prompt to give the user instructions, as you will see in the following activities.

Have you noticed that DOS commands in the batch files you have created so far appear on the screen when the file is run? Batch files often perform tasks where you want the DOS command and its results displayed; at other times the display would be less confusing if the command itself did not appear on the

screen. When this is the case, the command display can be turned off so that only the results of the command appear on the screen.

The ECHO ON and ECHO OFF commands are used to turn the display of the commands on and off. With ECHO ON, the commands appear when the system processes them as the batch file is running. With ECHO OFF, commands that are processed do not appear on the screen; only their results are displayed.

If the ECHO command is followed by any wording other than ON or OFF, the wording is displayed as a comment similar to a REM command remark. One important thing to remember is that the default for ECHO is the ON condition. When a batch file starts running, ECHO is always ON, and commands in the file will be echoed or displayed on the screen. When a batch file finishes running, the system automatically returns to the ECHO ON condition, even though ECHO may have been turned off with an ECHO OFF command within the batch file.

Practice: Using REM, PAUSE, and ECHO

REM

Retrieve the AUTODIR.BAT file to add a REM line in the file to clarify its use. From the BAT directory key:

- A:\BAT>**EDIT** (or **EDLIN**) **AUTODIR.BAT** [Enter]

  Insert a blank line at the end of the first line (CLS) [Enter]
  On the blank line you have created, key:

- **REM This file displays and prints the** [Enter]
- **REM root directory and checks the disk.** [Enter]

Save the file under the same name and exit the text editor.

Run the AUTODIR batch file to be sure that it runs as before but now has two lines of explanation added.

PAUSE

As with the REM command, the PAUSE command can be used to add documentation to the file with no effect on the execution of the program, except for halting execution until you are ready to proceed. After keying the PAUSE command, space once and then key the information you wish to add on the same line.

To demonstrate how PAUSE works, add a PAUSE command to the AUTODIR.BAT file. Retrieve the file again by keying

- A:\BAT>**EDIT** (or **EDLIN**) **AUTODIR.BAT** [Enter]

After the line DIR /W, insert a line by pressing [Enter] and add the following statement into the blank line you have created:

- **PAUSE If the printer is not on, turn it on now.** [Enter]

Save the file under the same name and exit the text editor.

Run the AUTODIR file. When DOS encounters the PAUSE command, the system halts execution of the file to give you a chance to read the directory and turn on the printer. The system will not continue until a key is pressed.

ECHO

To see how ECHO works, create a new batch file named EX.BAT to experiment with the ECHO ON and ECHO OFF commands. Key the following:

- A:\BAT>**COPY CON EX.BAT** (or **EDIT** or **EDLIN EX.BAT**) [Enter]
- **ECHO OFF** [Enter]
- **ECHO Echo is OFF now.** [Enter]
- **TIME** [Enter]
- **ECHO ON** [Enter]
- **ECHO Will this ECHO line display?** [Enter]
- **TIME** [Enter]

Save the file as **EX.BAT** and exit the editor.

Run EX.BAT by keying the filename **EX** and pressing [Enter]. (It will be necessary to press [Enter] for each time prompt.)

Since the default for ECHO is on, you will see the first command ECHO OFF. However, after that point you will not see the command itself, only the results of the command. Notice the difference in how the ECHO and TIME commands are displayed the first time, when ECHO is off, and the second time, when ECHO is on.

To prevent the ECHO OFF command from displaying on the first line as it did with EX.BAT, key an @ symbol at the beginning of the line as in the following:

```
@ECHO OFF
```

To further experiment with the ECHO ON and ECHO OFF commands as well as the @ symbol, create the TRIAL.BAT file. Key the following:

- A:\BAT>**EDIT TRIAL.BAT** [Enter]
- **@ECHO OFF** [Enter]
- **ECHO Echo is OFF now.** [Enter]
- **TIME** [Enter]
- **REM Will this REM line display?** [Enter]
- **PAUSE Will this PAUSE line display?** [Enter]
- **ECHO Will this ECHO line display?** [Enter]

Save the file as **TRIAL.BAT** and exit the text editor.

Run the file. Again, you will have to press [Enter] each time the system stops for the time prompt. After ECHO is turned off at the beginning of the file with the @ECHO OFF command, the commands themselves are not displayed—only the results appear as each command is executed in sequence. The information displayed by DOS for the TRIAL.BAT file looks similar to the following:

```
ECHO is OFF now.
Current time is (current time)
Enter new time:
Press any key to continue ...

Will this ECHO line display?
```

Note that the results for the TIME and PAUSE commands show, but not the commands TIME and PAUSE. The REM command and its text do not show at all with ECHO OFF. With ECHO OFF the ECHO command does not display. However, the text Will this ECHO line display? definitely does show. Only the statements following an ECHO command are displayed when preceded by ECHO OFF or @ECHO OFF.

With ECHO OFF, you can see that remark (REM) statements are useful for including reminders about the file itself—information that need not be displayed to the user. ECHO is preferable for displaying messages to the user, since the command ECHO is not displayed.

THE AUTOEXEC.BAT FILE

A special batch file with the reserved name of AUTOEXEC.BAT will load specific files and set special defaults when you boot the computer. For example, AUTOEXEC.BAT can cause DOS to display a custom prompt of your design, or change a color monitor screen to your favorite hue, or load a software program for you each time you turn on or warm-boot the computer.

These and other customized setups are made possible by storing commands in the AUTOEXEC.BAT file.

Whenever the computer is booted, the system looks for and loads COMMAND.COM; then the system checks to see if there is a file named AUTOEXEC.BAT in the root directory of the disk used for booting—the floppy disk in drive A or the hard disk. If it finds an AUTOEXEC.BAT file in the root directory, the system automatically executes the commands in the file.

The AUTOEXEC.BAT file is similar to other batch files in that it contains a series of DOS commands. It is keyed using a text editor and must be stored in ASCII format as you have done with all batch files.

Practice: Creating an AUTOEXEC.BAT File

Write an AUTOEXEC.BAT file on the CLASS diskette to execute some of the commands you have learned so far. This AUTOEXEC.BAT file will set a search path for DOS, modify the DOS prompt to show the current drive and directory, install the DOSKEY program, install a mouse driver, list the root directory and pause to let you read the directory, and finally clear the screen. Key the following:

■ A:\>**EDIT (or EDLIN) A:AUTOEXEC.BAT**	[Enter]
■ **@ECHO OFF**	[Enter]
■ **PATH C:\DOS** (Or to your DOS files)	[Enter]
■ **PROMPT PG**	[Enter]
■ **MOUSE**	[Enter]
■ **DOSKEY**	[Enter]
■ **DIR **	[Enter]
■ **PAUSE**	[Enter]
■ **CLS**	[Enter]

Save the file as **AUTOEXEC.BAT** and exit the text editor.

Now run the file by either keying the name, **AUTOEXEC**, and pressing [Enter], or by rebooting the computer. AUTOEXEC.BAT will execute the following:

Turn the display of commands off.
Set the path to the DOS files.
Set the prompt to show the current directory.
Install the mouse.
Install DOSKEY.
Display a wide directory of the root directory of the disk that booted the computer.
Pause the display so you will have time to read the directory.
Clear the screen.

Note that in order for the system to run the AUTOEXEC.BAT file as part of the boot process, DOS must find it and the system files required for booting in the root directory. In other words, the disk on which you store the AUTOEXEC.BAT file must also have the system files IBMBIO.COM and IBMDOS.COM (or their MS-DOS equivalents) and COMMAND.COM in the root directory. You placed the system files there when you formatted with the /S switch.

To the AUTOEXEC.BAT file you have created you could add many other commands including one to start an application program, such as WordPerfect, after the CLS command.

CONFIG.SYS

Like AUTOEXEC.BAT, the CONFIG.SYS file is a user-created and modified text file that customizes system operation. Like AUTOEXEC.BAT, CONFIG.SYS also has a reserved name and must be in the root directory of the boot disk. When you boot the computer, the CONFIG.SYS file gives DOS any special instructions you need for configuring the system. For example, some application software has specific requirements that must be put in the CONFIG.SYS file in order for them to run successfully. When the computer is booted, the CONFIG.SYS file is run before COMMAND.COM or AUTOEXEC.BAT and, therefore, before the application software is loaded. Consequently, CONFIG.SYS ensures that the requirements of the software program are met.

The following is a typical CONFIG.SYS file:

```
FILES=20
BUFFERS=20
DEVICE=C:\DOS\ANSI.SYS
LASTDRIVE=J
SHELL=COMMAND.COM /P /E:512
```

Note that none of these commands are DOS internal or external commands. When CONFIG.SYS is run, DOS has not been fully loaded and only CONFIG.SYS commands can be used. The following is a brief description of each of the commands used in the above-listed CONFIG.SYS file:

The **FILES=** command tells DOS how many files can be open at one time, in this example 20. Otherwise, the default number of eight, with five dedicated to the standard input and output devices, such as the keyboard. This leaves only three for program files. Some software requires that additional files be open. The number can be as high as 255. For each file over eight, the memory requirement for the system increases by 64 bytes.

The **BUFFERS=** command tells DOS how much buffer space to reserve in memory for temporary storage of data it reads from the disk. Each buffer takes 528 bytes of memory. The default is 15 for a system with 640K of memory. Generally, there should be as many buffers as files.

The **DEVICE=** command tells DOS where to find and load drivers, which are system (SYS) files that control special devices, such as a mouse.

The **LASTDRIVE=** instruction gives the drive name of the last local drive you want DOS to recognize and is used for standalone and networked computers.

The **SHELL=** instruction tells DOS to use COMMAND.COM as the shell or umbrella program, to make it the permanent shell (/P), and to set the size of the environment (/E) to 512 bytes. The environment is a segment of memory set aside to store special instructions, such as the search path and any special prompt you specify.

Practice: Creating a CONFIG.SYS File

Use EDIT or other text editor to create the following CONFIG.SYS file in the root directory of your CLASS disk.

- **A:\>EDIT CONFIG.SYS** [Enter]
- **FILES=20** [Enter]
- **BUFFERS=20** [Enter]
- **DEVICE=C:\DOS\ANSI.SYS** [Enter]
- **SHELL=COMMAND.COM /P /E:512** [Enter]

Save the file as **CONFIG.SYS** and exit the text editor.

After you have saved the CONFIG.SYS file in the root directory, you must reboot the computer so that DOS will reconfigure the computer according to the CONFIG.SYS file instructions. There will be no apparent effect on the computer. However, the system will have these specified capabilities.

CUSTOM MENUS

Many users find it helpful to boot the computer and see a menu on the screen. A menu usually displays a number or letter-coded list of the various software options available on the computer. The user keys a number or letter to load and use the desired software package.

A custom menu system can be set up by creating a text file that is displayed by the AUTOEXEC.BAT file and a set of batch files to start each software program. Here is one way a custom menu system can work:

1. When you boot the computer, the system reads and executes the AUTOEXEC.BAT file.

2. Within the AUTOEXEC.BAT file there is a TYPE command to display a text file that shows the list of menu options.

3. You select from these options and key the letter or number representing your selection. Each option you choose has a corresponding batch file with a matching filename (A.BAT, B.BAT, 1.BAT, 2.BAT, etc.). When you key the letter or number for your choice, which is the filename of the related batch file, the batch file executes.

4. The batch file contains commands for starting the software application you have chosen.

In this activity you will create a custom menu of the type just described. Rather than setting up an application software menu on a hard disk, you will create a more modest, but equally useful, project on your floppy disk in drive A. You will set up a menu and batch files for using DOS commands such as FORMAT, DISKCOPY, DIR, and CHKDSK. The same principles apply when you set up an application software menu system on your hard disk.

Practice: Creating a Custom Menu

On your CLASS diskette, create a custom menu system of the type just described. The first step in creating the custom menu is to modify the AUTOEXEC.BAT file by adding the line in bold at the end of the AUTOEXEC.BAT file. It is also important to add the BAT directory on the CLASS disk to the path statement so that the 1.BAT, 2.BAT, 3.BAT, and 4.BAT files can be found.

```
@ECHO OFF
PATH C:\DOS (Or to your DOS files);A:\BAT
PROMPT $P$G
MOUSE
DOSKEY
DIR \
PAUSE
CLS
■ TYPE TEXT\MENU.TXT                                    [Enter]
```

Now use a text editor to create a text file named MENU.TXT on the disk in drive A. You could place the MENU.TXT file in the root directory of drive A; however, for practice in working with paths and subdirectories, store it with other text files in the TEXT directory or in the CLASS disk in drive A. If you have no TEXT directory, it will be necessary for you to create one in which to

store MENU.TXT. MENU.TXT is a *text* file that will be displayed with the TYPE command.

Key the following:

- A:\TEXT>**EDIT** (or other text editor) **MENU.TXT** [Enter]
- [Enter]
- [Enter]
- **DISK OPERATING SYSTEM UTILITY MENU** [Enter]
- **CREATED BY (YOUR NAME)** [Enter]
- [Enter]
- **1. Check the volume label on the disk.** [Enter]
- **2. Check the version of DOS.** [Enter]
- **3. List files on a data disk.** [Enter]
- **4. Check a data disk.** [Enter]

- **Enter your choice at the DOS prompt** [Enter]

Save the file as **MENU.TXT**.

In order for the system to execute these commands when you select 1, 2, 3, or 4 from the menu, you must have a batch file for each option. For example, when you want the VOL command to be executed, you will need a file named 1.BAT. VER requires 2.BAT, DIR requires 3.BAT, and CHKDSK requires 4.BAT. When you key 1, 2, 3, or 4, the system will run the appropriate batch file, since the number is the filename. Of course, the search path must be set to the BAT and DOS directories.

Create batch files that will carry out each of the numbered functions above. Store the files in the BAT directory. Adapt the command parameters if necessary to accommodate the drive configuration on the computer where the menu will be used. Key the batch file for option 1 on the menu as follows:

- A:\BAT>**EDIT 1.BAT** [Enter]
- **@ECHO OFF** [Enter]
- **CLS** [Enter]
- **VOL A:** [Enter]
- **PAUSE** [Enter]
- **CLS** [Enter]
- **TYPE \TEXT\MENU.TXT** [Enter]

Save the file as **1.BAT** with the SAVE AS command and do **not** exit the text editor.

With the file still on display, cursor to the command line that reads VOL A:. Change that line to read as follows:

- **VER** [Enter]

Save this file as **2.BAT** with the SAVE AS command and do **not** exit the text editor.

With the file still on display, now the title reading 2.BAT, cursor to the command line for VER and change the line to read as follows:

- **DIR A:\ /P** [Enter]

Again, use the SAVE AS command to save the file as **3.BAT**. Do **not** exit the text editor. The file will still be displayed, but the title will now read 3.BAT. Cursor to the command line DIR A:\ /P and change it to read as follows:

- **CHKDSK A:**

Save the file as **4.BAT** and exit the text editor. Reboot the computer with this disk in drive A or key AUTOEXEC. The AUTOEXEC.BAT file executes and displays the menu. Does it work for you? If not, go back and troubleshoot the files to find your errors. Try each menu option.

Very nice work on batch files! Check your comprehension of the concepts presented in this chapter by completing the Self-Check Quiz and Assignment.

CHAPTER 10 SELF-CHECK QUIZ

True/False (Write T if the statement is true, F if the statement is false.)

_____ 1. It is not necessary to key the file extension to run a batch file.

_____ 2. The REM command is used to remove lines from a batch file.

_____ 3. After the PAUSE command halts the system operation, it will automatically restart in a few seconds.

_____ 4. The ECHO command gives you a choice of displaying or not displaying the commands when a batch file is run.

_____ 5. The CONFIG.SYS file must be in the root directory.

Multiple Choice (Circle the correct answer.)

6. Following an ECHO OFF statement, which of these lines in a batch file will display the instruction "Select one" when the file is run?

   a. PAUSE Select one
   b. ECHO Select one
   c. Select one
   d. REM Select one

7. Which of these commands should appear before the others in an AUTOEXEC.BAT file?

   a. PATH C:\DOS
   b. CHKDSK
   c. DIR /P
   d. ECHO OFF

8. Which command would you use in a batch file to display the directory of all backup (BAK) files on the disk in drive A?

   a. A:DIR*.BAK
   b. DIR A:*.BAK /S
   c. REM DIR*.BAK FILES
   d. PAUSE DIR A:*.BAK

9. Which of the following is not a legal batch file name?

   a. ERASEA.BAT
   b. REPEAT.BAT
   c. COPY.BAT
   d. &25_$220.BAT

10. Which of the following lines might appear in a CONFIG.SYS file?

    a. PROMPT PG
    b. PATH C:\DOS
    c. FILES=20
    d. CLS

ASSIGNMENT

To practice using the commands in this chapter, complete the following steps. Check off each step when you understand its operation, or mark the step with a question mark if it is not clear. Write the commands or keystrokes in the space provided.

1. Insert the TREE DISK diskette, create and print a batch file using a text editor, name it LOOKAT.BAT, and store it in the BAT subdirectory. Use DOS commands to accomplish the following in the batch file:

   a. Leave ECHO ON for trouble shooting purposes.
   b. Clear the screen.
   c. Explain the purpose of the batch file with a REM statement.
   d. Display a directory of the BAT subdirectory.
   e. Display a directory listing for AUTODIR.BAT.
   f. Display the AUTODIR.BAT file using the TYPE command.

   a. _____

   b. _____

   c. _____

   d. _____

   e. _____

   f. _____

2. Clear the screen and run the LOOKAT.BAT file. Print the screen or run the file with echo printing toggled on.

   _____

3. The batch file may have scrolled past you on the screen so that you could not read certain data. Place a PAUSE command where appropriate.

   _____

4. Add a REM statement explaining the purpose of the file.

   _____

5. Clear the screen and run the LOOKAT.BAT file. Print the screen or run the file with echo printing toggled on.

6. If the commands in LOOKAT.BAT run properly, turn ECHO OFF.

7. Print a directory of the BAT directory.

8. Use the TYPE command to print a copy of the LOOKAT.BAT file.

9. Create a simple AUTOEXEC.BAT file in the root directory on TREE DISK that will do the following:

   a. Turn ECHO OFF.
   b. Clear the screen.
   c. Display the root directory of drive C.
   d. Print the directory of drive A sorted by date, latest date first.
   e. Access a text editor such as EDLIN or EDIT.

   a. _____

   b. _____

   c. _____

   d. _____

   e. _____

10. Run the AUTOEXEC.BAT file with echo printing on. If you do not have echo print capability, print the screen. Print a copy of the AUTOEXEC.BAT file.

INDEX

A

Aborting DOS Commands, 41-42
Adapter Card, 3
Allocation Unit, 43, 88
Alternate (Alt) Key, 5
ANSI.SYS, 190
APPEND, 73-74
ASCII Codes, 10
AUTOEXEC.BAT, 188-190

B

Backslash, 115
BAK Extension, 155-156
Batch Files, 181-189
Booting the System, 12
    Cold, 12
    Warm, 13
Break Key, 41-42
Bytes, 10

C

Canceling a line, 22
CD or CHDIR, 116-118
CHKDSK, 103-105
CLS (Clearing the screen), 24
Cluster, 43, 88,
    See also Allocation Unit
Cold Boot, 12
Combing Files, 73-74
COMMAND.COM, 14, 23, 46
COMP (Comparing Disks) 99-101
Comparing Files, 74-76
CONFIG.SYS, 190-191
Control (Ctrl) key
    Ctrl-C or Ctrl-Break, 41-42
    Ctrl-PrtSc or Ctrl-P, 24-25, 57
    Ctrl-Z or F6, 54, 152
COPY, 63-70
    combining files, 71-73
    to a different disk, 65
    to a different subdirectory, 128-129
    to the Hard Drive, 67-68, 121
    to the same disk, 64
    used with wildcards, 69-70
CPU, 3
Current Drive, 16
Custom Menu, 192-194

D

DATE, 14
Default Directory, 16, 51, 116
DEFRAG, 106
DEL, 86-87
Deletion Tracking (Deletion-Tracking File), 89
DELTREE, 135-136
Device names, 26, 30
DIR, 27-34
    searching directories for
    selected file(s), 28
    sorted, 32
    switches
        /P, 29
        /W, 30
Disk drives, 3, 7
Disk Operating System (DOS), 1-2
DISKCOMP, 99-101
DISKCOPY, 96-98
DOS Editing Keys, 79-81
    See also Function Keys
DOS Prompt, 13
DOSKEY, 53
Drive Prompts, 23

E

Echo, 187-188
Echo Printing, 57
    See Printing
EDIT, 166-177
EDLIN, 145-160
End-of-File marker, 75
ERASE, 86-87
    See DEL
Error Messages, 16
Extensions, (File Name), 25
External commands, 14, 23, 0

F

FAT (See File Allocation Tab), 43, 88
FC, 99-101
    See COMP
File Allocation Table, 43, 88
File Names, 25
    extension, 26
Floppy Disks, 38, (Illus.), 39
FORMAT, 42-48,
Fragmented Files, 103
Function Keys, 7

G

Gigabytes (GB) 10

H

HELP, 35
Hidden files, 46, 103

I

IBM PC-DOS, 2
Internal Commands, 1, 23, 40

J

K

Keyboard, 5-6
Kilobytes (K or KB), 10

L

LABEL, 49
Labeling a disk, 49

M

MD MKDIR, 113
Megabytes (MB) 10
Memory
    Random-access memory (RAM), 4-5
    Read-only memory (ROM), 4-5
Menu (Custom), 192-194
Microsoft MS DOS, 2
MIRROR, 89
Monitor, 4, 7, 12
Mother Board, 3-4
Mouse, 3
MOVE, 78-79
MS/PC DOS, 2

N

Non-Contiguous Files, 105
    See Fragmented Files

O

Operating System, 2

P

PATH, 41, 114-115
PAUSE, 186-187
    See Batch File Commands
PC-DOS, 2
Permanent Data Storage, 9
Print, 23, 57-58
    Copy command to print, 58
    Directory listings, 22
    Echo Print, 57
    Print Screen, 24-25, 57
    Type command to print, 58
Printer, 3, 11-12
Program Extensions, 26
Prompt, 18
    PROMPT PG, 17-18

Q

QBASIC, 165

R

RAM, 5
RD or RMDIR, 132-133
REM, 185-186
Removing Directories, 132-133
REN or Renaming files, 76-77
    Using wildcards, 77-78
Reset button, 13
ROM, 104-105
Root Directory, 112

S

SCANDISK, 105-106
Sector, 42
Search PATH, 40-41
    See also PATH
Self-booting, 46
Sentry, 91
    See UNDELETE
Software, 11
Subdirectory, 111-135
Subdirectory Marker, 131-132
Subdirectory
    Move (Rename), 136
System unit, 3, 4, 12

T

Text files
    Creating, 53-56
    Displaying contents, 56
Things to Remember, 21
TIME, 15
TREE, 134-135
TYPE, 56

U

Unconditional FORMAT, 48
UNDELETE, 87-94
UNFORMAT, 95

V

Version (VER), 34
Versions of DOS, 2
VOL (Volume label), 49

W

Warm Boot, 12
Wildcards, 32-34
Write-Protect, 39

X

XCOPY, 137-138

ITP

QUICK AND EASY DOS—

FOR NOVICES AND THOSE WHO WANT A SIMPLE BRUSH UP

In ten easy chapters you'll master DOS 5.0, 6.0, 6.1, and 6.2 basics, including essential commands and operations. With this knowledge DOS becomes your reliable assistant, allowing you to perform numerous tasks including installing software, configuring your computer, and managing files more effectively.

HIGHLIGHTS:

- Written for first-time DOS users or anyone who would like to brush up on DOS skills
- Covers the latest DOS upgrades (5.0, 6.0, 6.1, and 6.2), yet is compatible with earlier versions
- Chapter format facilitates learning with a list of objectives, a reading assignment to explain each new command or operation, practice exercises reflecting usage in business and industry, and a self-check quiz to review progress
- A more in-depth assignment in each chapter is designed to apply DOS skills to solve real, practical problems

Also available for a more in-depth explanation of DOS 5.0: ***MS/PC DOS: Concepts, Exercises and Applications*** *by Mary Auvil, L. Joyce Arntson, and Barbara F. Stockler*

Wadsworth

ISBN 0-534-23406-2

W7-AJR-754