Chapter 1

Introduction to personal computers

Why use the command line? 01
Big Picture 02
Terms 03
Definitions 04
Hardware 06
    Input 07
    Process (CPU + memory) 08
        Measuring memory 09
    Output 10
    *Storage 11
        Optional Advanced Topics 13
    Motherboard 15
Software 17
    System 17
        Operating Systems 17
        Utilities and Programming 18
    Applications 18

Why learn Command Line?

The command line is often the fastest way to accomplish a job, and the only way to apply a command to multiple objects, such as renaming hundreds of files at once. In some environments, the command line is the only method of taking control of the task at hand, such as troubleshooting, or advanced networking. Plus, learning paths from a command line is of great benefit when writing HTML, so these skills apply to many advanced classes which follow.

Why use DOS instead of the Windows Command Prompt?

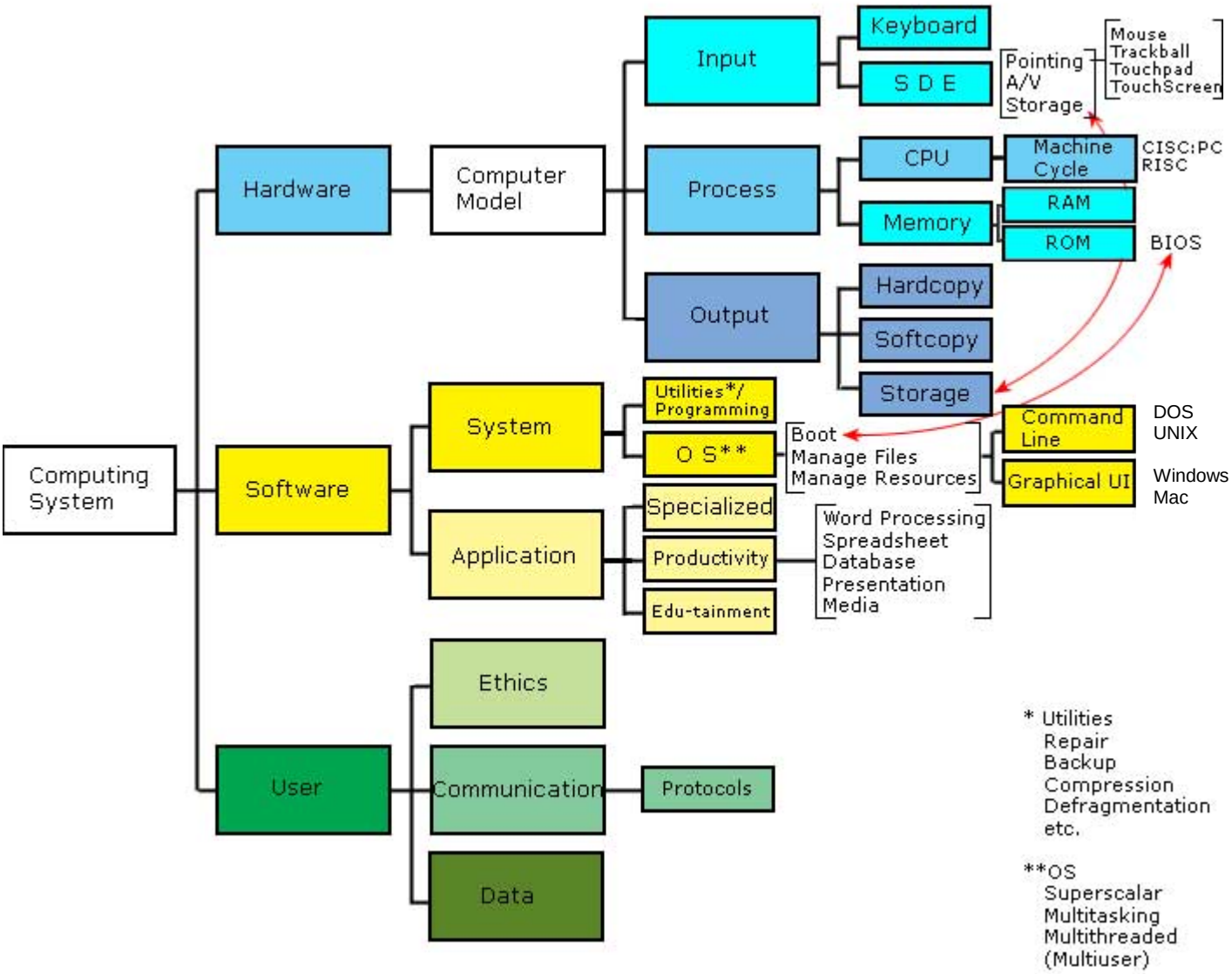
The command prompt is the full power of Windows, implemented with commands instead of menus and dialog boxes. When modern users open the command line, they are not faced with the limitations of DOS, just a different way to accomplish a task. So, why use DOS if it is available, instead of the command prompt?

Well, it is not because major differences in the commands or what they do. Most DOS and Windows commands are identical in purpose and usage; for our purposes there are only two commands in DOS mentioned that do not occur in Windows, simply because Windows provides a different solution. (Most of the commands that occur in Windows, and not in DOS, are advanced networking commands and are beyond the scope of this introduction.)

So why use real DOS if it is available? While learning the command line, it would be too simple for students to simply switch to a Window to rename files, copy documents, create folders, or use the Windows clipboard. Students learn the tool if that is the only tool available.

So, DOS is an instructional, pedagogical, choice. Once mastered, you may implement the exact same syntax in the Windows command prompt, and later, with minor tweaks, to the Linux Terminal. But the best way to learn command line is to use it.

The Big Picture



The following is a list of terms. The meanings follow, and will help as you begin reading the class textbooks...

Glossary

BIOS

Bit

Boot

Byte

CPU

Disk

Giga...

Hardware

Hertz

Input

Kilo...

Mega...

Monitor

Output

Peripheral

RAM

ROM

Software

Zip

Next are *my* definitions, which are explained a little differently than your books do. Hopefully the combination of my definitions, and the book definitions, will help you master the terms.

Definitions

Access	Random access, like disks or memory, is like a music, CD, you may skip to a favorite song; Sequential Access, like backup tapes, is like a music cassette, you have to go through songs to get to your favorite
Bandwidth	Transmission capacity, often incorrectly referred to as transmission speed
BIOS	Basic Input/Output System, a ROM chip used especially at boot up, when the OS loads
Bit	Binary Digit, a 1 or a 0, the only item a computer can understand
Byte	Binary Term, usually 8 bits, enough information to represent a typed character
CPU	Central Processing Unit, the brains of the computer a type of Microprocessor made up of the Control Unit and the Arithmetic Logic Unit
Computer Proficiency	In order to use a computer, you don't have to be an expert, just proficient... Being able to do basic functions with a computer
CRT	Cathode Ray Tube, a TV style monitor
Data	Facts that can be processed into useful information, input (the user communicating with the computer, such as entering items on the keyboard)
Desktop	The metaphor used in a Windows computer to show what files and programs you may use, and how they are organized
DOS	Disk Operating System, a command line OS that came before Windows
File	In databases, a collection of related records; in computers in general, the name given to a collection of stored data / instructions
Hard Disk	Long memory that keeps content even when the computer is turned off… Auxiliary or Secondary Storage
Hardware	The part of the computer you can touch, if it connects to the CPU, it is a 'peripheral'
Hertz	Repetitions, or cycles; often related as cycles per second, as in megahertz , about a million cycles
Icon	A graphical element on a desktop that represents an object, such as a file or program
I/O	Input/Output, typically a used as a hardware term.

Information	Data that has been processed info something useful; output (the computer communicating with the user, such as results displayed on a monitor)
Kilo...	about 1000, exactly 1024; can be used with bits, bytes, or hertz
Mega...	about a million, exactly 1024^2; can be used with bits, bytes, or hertz
Microprocessor	another name for CPU
O/S	Operating System; a subset of system software, the OS provides the interface between the hardware, application software, and the user; i.e. DOS or Windows
Program	A file that contains instructions, such as a word processor
RAM	Random Access Memory, (should've been Read Write Memory) Main Memory Short term Memory that can hold new information RAM is erased when the computer is turned off
ROM	Read Only Memory, memory that holds unchanging information
Software	Instructions or data in RAM; the part of the computer you can't touch; includes system software (coordinates activities between user, applications, and hardware) and application software (does something useful for the user, such as word processing)
User Interface	How the user interacts with the system software; Windows is a Graphical UI
www	World wide web; the area of the Internet that uses http protocol to transfer files written in html; also, the normal host name assigned to a web site, such as the *www* in http://**www**.templejc.edu
Zip	Not a DOS term, but important for class. ZIP: A file that can contain multiple files, or more importantly, a file that is compressed to take up less space; useful for emailing attachments, or for uploading. NOTE: the ZIP contents are not always usable as is. Thus, you should not try to read or edit zip files until you download them and extract them. If redoing a lab, delete the zip, and edit the original files.

Notes

A list of all the DOS commands and usage is in the DOS Appendix

Hardware

Computer System
Hardware → Software → You, the user

Computer Model
Input → Process → Output

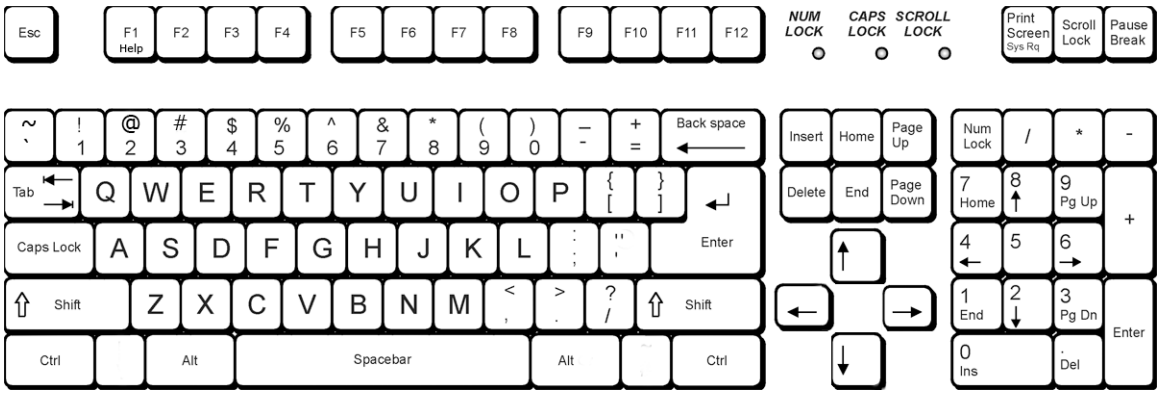
Fast Food Restaurant
Input → Process → Output
Raw Meat → Cook → packaged Hamburger

Computer
Input → Process → Output
Raw Data → Manipulated → Useful Information

Input
The user communicating with the computer
Input: Raw Data
Keyboards and/or Source Data Entry

→Keyboards

Typewriter differences
- Note: Function Keys may be used as shortcuts for commands
- Note: [Ctrl] modifies a keyboard selection into a command
- Note: [Alt] allows you to choose an alternative



Reserved Device Name
CON Console (keyboard)

Input
Keyboard, Continued

Enter	Execute
Ctrl + Brk	Cancel
Ctrl + S	Pause screen
F3	Re-type last DOS command
Print screen or shift print screen	Sends the screen to the default printer

→Source Data Entry

MAY NOT BE AVAILABLE IN DOS*

Source Data Entry basically refers to any input that is not a keyboard.

Source Date Entry devices include the Mouse, Track balls, and/or touch pads; modern devices include scanners, digital cameras, microphones, etc.)

Left mouse button: select

Right mouse button: execute

(*many source date entry devices require a driver.
Drivers are the instructions to the operating system on how a device works. Drivers are often loaded and configured during boot;
see Chapter 8, Batch files, for details on autoexec.bat and config.sys)

Reserved Device Name
AUX Auxiliary Serial Port
COM*x* Serial Port, x= 1, 2, 3 or 4

Serial ports move information a bit at a time, in sequence

Process
CPU + Memory

CPU
Control Unit + Arithmetic Logic Unit
Manager + Workers
Scheduling + Math and Comparisons
+ - * / < = >

Machine Cycle: Fetch Decode Execute Store
Fetch: retrieve and load information
Decode: convert 1s and 0s into computer usable instructions
Execute: carry out the instructions
Store: place results of execution where it can be used

→Measuring CPU speed. The number of times a computer can repeat the machine cycle process is roughly how fast the computer is, also known as the Clock rate, measured in Megahertz (MHz., millions of repetitions) or Gigahertz (GHz., billions of repetitions) See page 17 for CPU comparisons.

Memory
Just as a person may use a white board for a complicated math problem or keep a book of instructions, a computer needs two kinds of memory to keep up with things… one changeable, and on fixed.

→ROM
Read Only Memory
The contents normally don't change.
Example: the BIOS is a ROM chip, used to boot the computer.
(Transferring the OS from disk to working memory; these direction don't normally change, and must always be available)

ROM is similar to a book; if you close a book; all the words will still be there, in the same order, tomorrow, or in three years.

→RAM
Random Access Memory,
perhaps should have been called Read/Write Memory
(Working memory, like a white board… new material can be added or removed when no longer needed. RAM is typically volatile, or temporary)

RAM is like a white board; you may add a new math problem, or remove an old math problem when solved… but it is not permanent. In most computers, when you turn off the system, everything in RAM goes away.

→Measuring RAM

Consider a horizontal line, - .
By itself, it conveys no meaning. But if you string together /, then the -, then a \... you can form a typed character... / - \ together forms an A.

It is the same with an operating system... a single one or zero conveys no meaning, but if you put together a series of 1s and 0s... you can form a typed character.

A single 1 or 0 is called a bit, short for Binary Digit.
Enough 1s and 0s to form a typed character is called a byte... short for Binary Term.

Typically, a single word does not convey a story. You need sentences, paragraphs, and chapters. In RAM, we signify a number of bits or bytes using the prefixes of kilo, mega, and giga.
(There are more, but these will meet our needs for the moment.

Kilo in computer terms is 1024.
(Roughly a thousand, but that is not precise enough for this class)
So a kilobyte is 1024 typed characters.

Mega in computer terms is 1024 times 1024, or 1024^2.
(Roughly a million, but that is not precise enough for this class)
So a megabyte is 1024^2 typed characters.
A floppy disk is 1.44 MB or 1.44 Megabytes
A compact disk is 600-800 MB or 800 Megabytes

Giga in computer terms is 1024 times 1024 times, or 1024^3.
(Roughly a billion, but that is not precise enough for this class)
So a gigabyte is 1024^3 typed characters.
A DVD is 2-17 GB or 17 Gigabytes
A hard disk is 80+ GB or often in excess of 80 Gigabytes

→DOS and RAM

DOS natively can only use 640 KB of RAM
In the old days, 1 MB of RAM was installed; DOS used the first 640 KB, and the video system used the rest, called upper memory.

As more memory was needed for video, video memory was moved to the video card, leaving more RAM available for DOS. New utilities allowed DOS to use upper memory, and any additional memory, called extended.

Output
The computer communicating with the user.

→Soft copy
Soft copy is computer dependant, as in video on the monitor

A Monitor works with a video card to display images;
The image is a series of glowing dots of light called pixels

Monitor terms
- Refresh rate: how often pixels are lit, often measure in Megahertz
  (that is, each pixel is re-lit millions of times per second)
- Resolution: the number of pixels across or up and down
  (the more dots, the smoother the image,
  more pixels requires more video memory;
  DOS typically works at 640 wide by 480 high, called VGA)
- Color depth: the number of colors that can be displayed
  (the more colors, the more video memory is needed)

Other softcopy output
Speakers, or computer controlled devices such as lathes or auto welders

→Hard copy
Hard copy is tangible (touchable) without a computer, as print on paper

A Printer often places dots of ink on paper to form typed characters, so the term resolution, measure in dots per inch, or dpi, applies to printers.

Reserved Device Name
LPTx parallel Port, x= 1, 2, or 3
PRN Printer (if attached to LPT1)

Parallel ports move information one byte at a time.

Storage
Four Functions of a Modern Computer

Input → Process → Output → Store

Since material cannot be added to ROM, and since RAM is volatile and may go away, storage is needed to *save* **data** or **instructions** long term.

Further, storage devices may be used as an input, or an output.

RAM is primary, so storage devices are often called Auxiliary or Secondary Storage. Most often, these devices are Disks.

→Magnetic Disks, such as floppy disks or hard drives
Positive magnetic charge can be a 1, a negative charge can be a 0.
Formatting prepares a disk for use; adds organizational storage areas

A hard disk is a metal or glass disk coated with a magnetic material. As an area of the disk could have a positive or a negative charge, this could be interpreted as a 1 or a 0, suitable for computers to record data on.

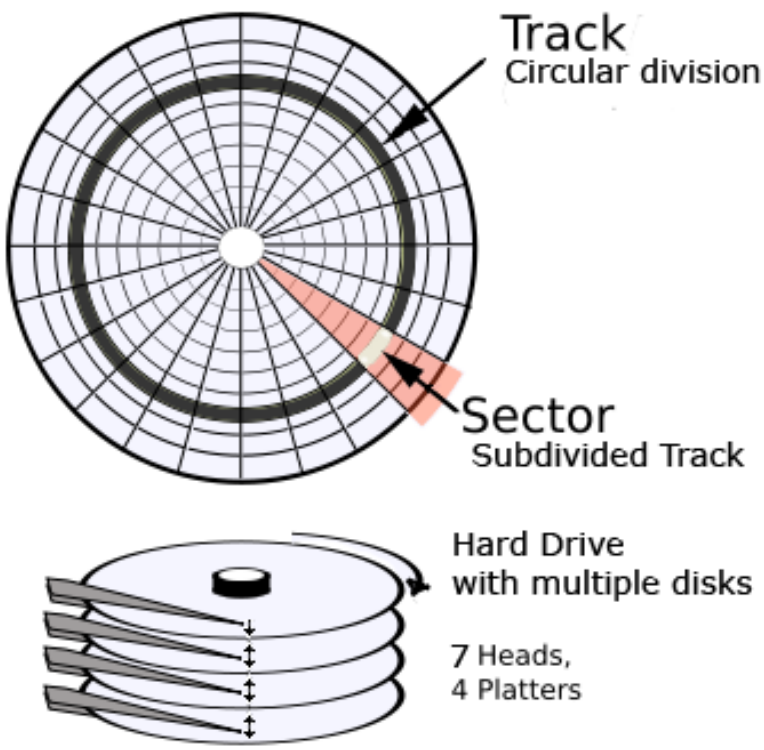


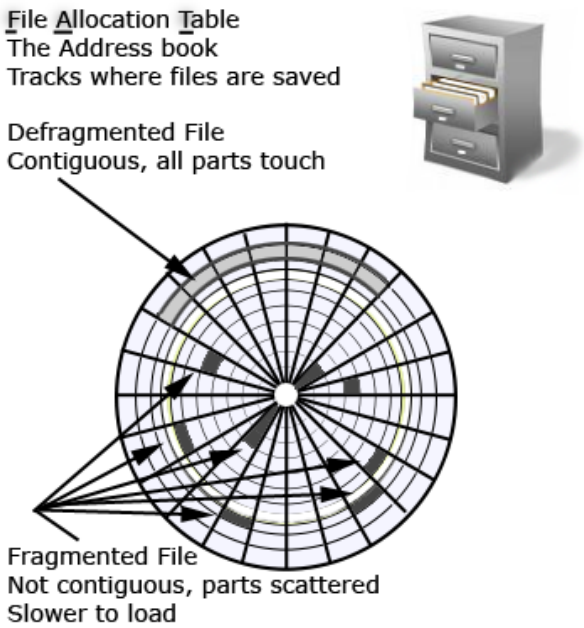
Diagram of a disc showing the invisible tracks and sectors created during formatting.

(Actually there are many more tracks and sectors than shown).

Tracks are the circular lines, similar to a bulls eye. The tracks are divided into sectors, which act as 'buckets' that can hold information.

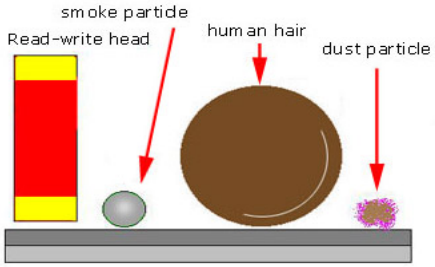
Sectors typically are too small to hold an entire file.

The very first sector on the very first track is often used for record keeping, an address book of sorts; in DOS, the system is called FAT.



Fragmented files take longer to locate and load into memory, so many DOS and Windows disks need a defragmentation utility to optimize disk access.

Sectors from time to time lose the ability to reliably store data. This is called a bad sector; bad sectors on a disc volume may be repaired with DOS's chkdsk or scandisk, or the Windows Error Checking tool.



Note: the gap between the read-write head is smaller than smoke, hair, and even dust. If something gets lodged in the gap, magnetic material is removed, destroying the data

Floppy disks, if present, are typically called the A: and the B: drive.
The first hard drive, if present, is typically called the C: drive.

→Optical Disks, such as CDs or DVDs (May not be read by DOS)
Light reflected one way can be a 1, light reflected the other can be a 0.
The first optical drive, if present, is typically called the D: drive.

NOTE: Many computers that run both DOS and Windows place DOS on the C: drive, and Windows on the D: drive. Therefore, any optical disk assignment may start with E:

→Disk Capacity
Recall, from the discussion on bits and bytes and kilo, mega, and giga:
Floppy Disk 1.44 MB
CD 800-600 MB and DVD 1.2-17 GB
Hard disk 10 MB-80 GB (old) or 80-120 GB and up.
1024^4 (Terabyte, trillions of bytes) drives exist, but are unusable by DOS)

ADVANCED TOPICS
Note: the following items may apply to Non-DOS command prompt operating systems.

→Non-disk storage (May not be read by DOS)
Flash drives, thumb drives, jump drives, etc. are all versions of a solid state (no moving parts) storage concept that doesn't require electricity (non-volatile). Typically these devices are treated as a disk drive, and are assigned drive letters following any hard disk and optical disks.

→Other storage devices (prior to DOS)
Recall, computer storage simply requires one of two possible states that can be read as a 1 or a 0… such as positive charge or not positive charge; light bounced here or light did not bounce here.
An early storage method was punch card, as in 1s and 0s being assigned as hole punched here or no hole punched here.
Magnetic tape preceded magnetic disks.

→Remote Disk access: Networks
Computers read storage devices, and it doesn't matter if the disk is in the computer or not.

A series of computers that share information are called a Network.

LAN: If the computers are close together, they may form a Local Area Network. Typically these computers connect via a Network Interface Card (NIC) and cable.

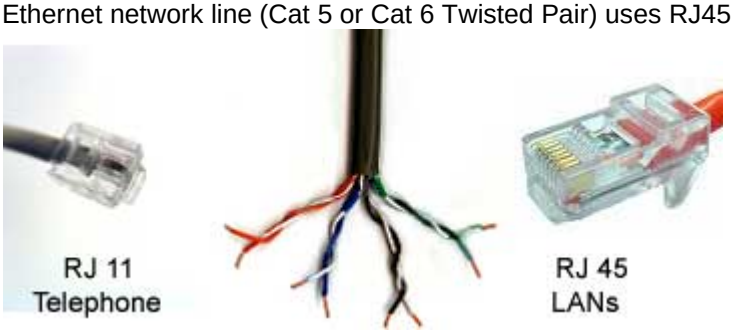
WAN: If the computers are not close together, they may form a Wide Area Network. Typically these computers connect via telephone or wireless access.

Older systems used analog phone lines, which required a modem to modulate or demodulate the binary signals to analog.

Note: a 4 lane freeway can move more cars that a single lane road, even if the speed limit is the same. So often fast networks are called broadband networks, which actually measures capacity… or how much information is moved per second, rather than how fast it is being moved.

Note: Hardware continues with the Motherboard, beginning on page 16.

Dial up Phone line (Cat 3 Twisted Pair) uses RJ11 connectors
Ethernet network line (Cat 5 or Cat 6 Twisted Pair) uses RJ45 connectors



Cable TV networking implement Coax (coaxial) cable and BNC connectors

Fiber Optic cable (normally many cables bundled) and FDDI connector

An analog phone line's 'speed' is measured from about 28 kilobits to 56 kilobits per second. A digital phone line can be about 128 kilobits per second. But most **broadband** connections all can move at about 1.544 megabits per second. What separates DSL from T1 and T3 broadband connections is the additional capacity... a T1 has perhaps the capacity of about 24 ISDN 'hoses' while a T3 has the capacity of perhaps 28 T1 'hoses.'

Connection	Capacity	Time to download 3 MB	Time to download 30 MB
Dial Up	14.4 to 56 Kb	7 minutes to 31 minutes	69 minutes to 5 hours
ISDN	~128 Kb	3 minutes	30 minutes
DSL/Cable	128 Kb~1.544 Mb	37 seconds at 640Kb	6 minutes at 640Kb
T1	1.544 Kb	15 seconds	2 minutes 30 seconds
Wireless LAN	to 10 Mb	1 second	24 seconds
T3	44.736 Mb	less than one second	5 seconds
Wired LAN	100 Mb	less than one second	2 seconds
Fiber Optic	to 2.488 Gb	less than one second	less than one second

Notes:

Motherboard

In its most simple terms, the motherboard connects the CPU and memory to all the other input, output, and storage devices.

The motherboard typically is stored inside a system case.

Other items typically found on a motherboard:

ROM BIOS	The ROM BIOS is the chip responsible for booting the computer. That is, loading the operating system from disk to RAM. THE ROM BIOS is a Read Only Memory chip that contains the Basic Input Output System that allows the computer to interface with the keyboard, monitor, etc.
CPU	The Central Processing Unit, the brains of the computer.
RAM	Working memory, RAM is changeable read/write memory for short term use. RAM is volatile, and the contents are lost when power is removed.
CMOS	The Complimentary Metal Oxide Semiconductor Think of the CMOS as a little chunk of RAM that is kept current by a battery, and keeps track of certain settings when the computer is off. The battery also powers the clock. Reserved Device Name CLOCK$ the system clock

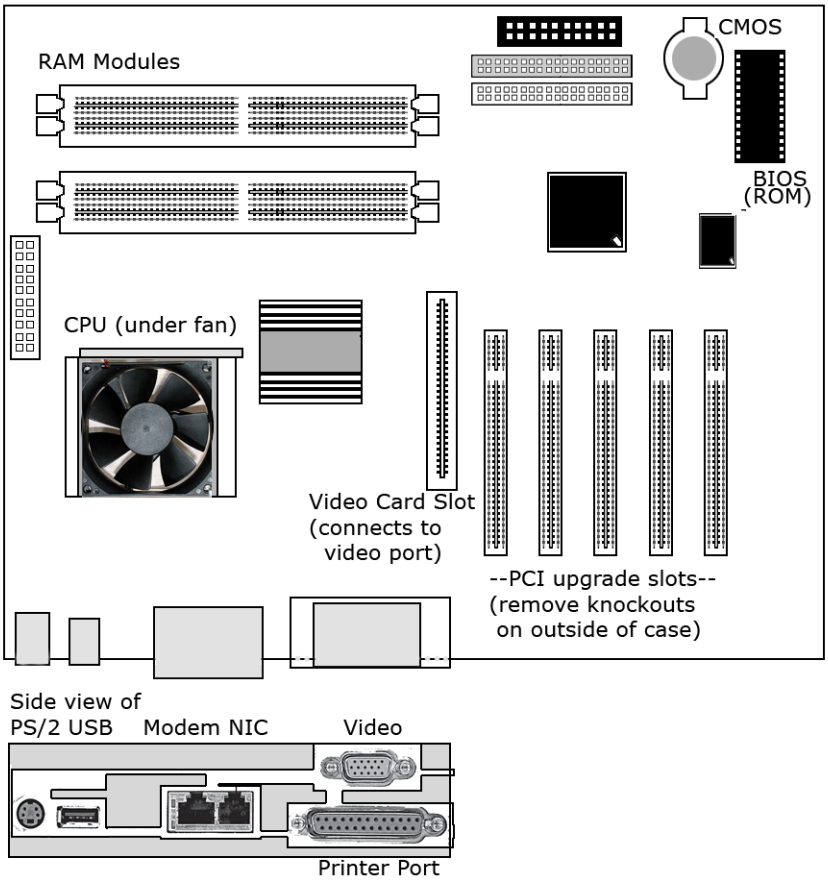
Devices that reside outside the system case (often called peripherals) connect to the motherboard through slots or ports.

Slots hold adapter cards for non-standard or specialized devices.
Ports connect standard devices to the motherboard.

Typical Ports:

Serial Ports	(obsolete) for 1st generation keyboard, mouse or modem
Parallel Ports	For 1st generation printer
PS/2 Ports	For 2nd generation keyboard or mouse
USB or IEEE 1394 Firewire Ports	For 3rd generation keyboard and mouse Can also provide high speed access to printers and (Non-DOS) devices such as cameras, networks, and removable storage devices

Typical Motherboard Layout



Note: The egg crate looking items are where cables are connected.

Aside from the video port connected to the video adapter, many modern computers don't have any of the above ports. They instead use multiple USB ports for the mouse, keyboard, printer, and network card.

Many laptop computers use a combination of PCMCIA PC-cards and USB.

The term often used for non-USB peripheral devices is legacy. Legacy slots used to include ISA, EISA, and VESA… each slot was a different shape and color, and accepted different kinds of adapter cards. Currently, if a computer has any slots, they typically are PCI.

BONUS: Brief Intel CPU History		Clock Speed
4004-8088 class	1971-1979	400 KHz (.4 MHZ) -5 MHz
80286 class	1982	8-16 MHz
80386 class	1985	16-40 MHz
80486 class	1989	20-100 MHz
Pentium class (1-4)	1993-2000	60MHz – 1.5 GHz (~1500 MHz)
Multiple core Pentium	2007-	2.5 GHz and up per core

Software

Software is the part of the computer you can't touch. True, you can touch a disk on which an instruction or data is saved; this is called a file. But touching a saved file is NOT touching software in the true meaning.

Software is the *intangible* data and instructions a computer *is processing*. People alone can't manipulate software. A user directs hardware to process software.

Software is typically classified as system software or application software.

System Software
System software doesn't really do much for users; it runs the computer.

System software includes Operating Systems, Utilities and Programming.

→Operating Systems Software (OS)
Texas is part of the US, not the other way around. The OS, the topic of this class, is a member of the larger System software family.

OS role in the computer system: **User** - Operating System < **Hardware** / Application **Software**

Aside from Operating systems provide the user access to hardware and applications, they also provide housekeeping tools, such file location, and the ability to create, edit, rename, copy, move, or delete files.

A user can interact with the Operating System through a command line which, while exacting, offers the most control.

A user can also interact with the Operating System via a Graphical User Interface (GUI). A GUI typically is easier to use, but is limited in what it can do.

OPERATING SYTEM	INTERFACE
DOS	Command line only*
Windows	Graphical user interface with Command line access
Mac X	Graphical user interface with Command line access
UNIX\Linux	Command Line or Graphical user interface with Command line access

*While there were tools that gave DOS a graphical look, DOS Shell and Windows 3.1 had all the limits of DOS. These limits are covered in Chapter 2 and 3.

Reminder: While DOS is a very simple, and limited command line OS, many of the commands are identical to the Windows command prompt.

And both DOS and the Windows command prompt can do things that CANNOT be done through Windows Graphical User Interface.

Thus this workbook: in order to extend Windows to its full capabilities you need to understand the command line. In order to get the most out of UNIX, Linux, or the Mac, you need to understand the command line.

You can learn the command line using DOS, or the Windows command line, but using DOS sort of forces you to learn the command line… Windows lets you bail out to the GUI. Use DOS if it is available.

→Utility Software
Tools that provide *computer support* beyond the interface and file management are often called utilities. Again these programs don't really do anything for the user, at least not directly… they keep the computer running so the user can use it. Some utilities are purchased separately, such as anti-virus software. However some utilities come with the operating system; a few of these are covered in Chapter 7.

→Programming support
The tools used to create operating systems, or other programs, falls in this area. Again, these programs don't really do anything for the user. Programming is well beyond the scope of this book, but at the lowest level, programming is giving the computer instructions… a simple way to deliver a series of instructions is a Batch files, covered in Chapter 8.

Application software
Application software does some real world task *for* the user, such as typing a letter, adding numbers, or listening to music. Many times applications are called programs.

Note: a program file contains the instructions to make your word processor work, for example. The letter you create *with* the word processor can be saved in a data file.

This workbook will discuss applications only in passing, to illustrate operating system concepts: operating systems start applications, operating systems send what you are typing on the keyboard to the application, and the operating system take information from the application and delivers it to monitor or printer.

Chapter 2

Introduction to DOS/Windows Command Prompt Concepts

Booting 22
IO 22
Command Execution 23
File Management 23
    Common file type extensions 24
Directories 24
    Directory trees 25

The big three jobs of the OS are input/output (I/O), starting programs or applications (also known as command execution), and file management.

Booting

While the ROM BIOS provides a POST (Power On Self Test) to verify the minimum required I/O devices are present, and the BIOS has enough control of basic I/O devices to use a disk as an input and direct the OS files to RAM, at some point the BIOS is no longer up to the job; however, by then there is enough of the OS in memory to control the rest of the job, and load specific files designed to handle configuration and setup. This is often called Real mode. Even Windows starts life just like a DOS computer going from BIOS to Real mode… as more of the OS gets loaded into RAM, more features become available, and Windows shifts into Protected mode.

If the computer is completely off, the start up is called a cold boot, and RAM is blank.

If for some reason you want or need to restart the computer (Control+Alt+Delete can force a reboot if the computer is hung), this is called a warm boot. There is still information in RAM, which may or may not be a good thing.

Warm boots are easier on the computer mechanically; cold boots ensure nothing troublesome in RAM is there to interfere with the computer.

I/O

Again, while the ROM BIOS provides basic access to the keyboard, monitor, etc., the BIOS cannot provide access to individual applications… that is the job of the OS. Further, there are many devices that the OS can recognize and use that the BIOS does not understand. So, some of the files DOS loads during the end of the boot process are specifically to deal with input and output and allows the OS to act as middleman between hardware and applications; these files include MSDOS.SYS and IO.SYS.

Further setup and configuration will be dealt with in Chapter 8, Batch files. Batch files don’t change the operating system, they simply automate any tweaking a user may wish to make to the system.

Advanced topic: You may view what the BIOS can control by using the BIOS setup. Newer BIOS chips support power saving, etc., that may not be supported by your OS; the BIOS setup can let you turn some features on and off… and your choices are saved in the motherboard CMOS.

Command Execution

Starting programs is another domain of the OS. As a matter of fact, the whole boot process is about starting a program… the OS itself. In DOS, the main program file is named COMMAND.COM. But, there are other small programs that are part of the OS, too, and they are called commands.

A command is just a small program that carries out a specific task of the operating system. And, of course, there are the larger application program files that the OS can start as well.

The model typically used to describe starting a program is
'Select, then do.'

In DOS, this means type out the command that starts the program or application, and press the Enter key to let DOS know you are ready to execute.

Commands are typed at a flashing cursor on the monitor called the command prompt or the system prompt; the flashing cursor indicates DOS is ready. More about the command prompt in Chapter 3.

In Windows, 'select, then do' may mean to click on an icon to select it, then press the Enter key… or double click… basically the first click selects and the second click replaces the enter key function.

File Management

When you save a file, the OS usually wants to know three things:
1) where you will save the file 2) what you want to name the file, and 3) what type of file it is. Then the File Allocation Table, or something similar, for that disk will record where the parts of the file are stored.

These pieces, the location, file name, and file type are all assembled using a combination of back slashes '\' and periods '.', called a path.

Location\File name.File type
such as
`C:\test.txt`

C: is the *location*, a '\' connects the drive to the file name.
Test is the *file name*, and it is connect to the *file type* txt using a '.'

There is NEVER a space within a file path. (More on paths later.)

Full file names include the file name and the file type. File types are also called the extension, and are for DOS typically three characters, such as the .SYS extensions for MSDOS.SYS and IO.SYS mentioned in the IO section.

Common file type extensions

Executable files (commands and programs) end with:
bat
com or
exe

Text files that may be read by DOS, and displayed on the screen
txt
ltr
bak (used by backup copies of a text file)
doc (was used as documentation… now used by Microsoft Word)
ini (initialization files)
hlp (help files)

Proprietary extensions used by applications or operating systems
sys for DOS system files
pdf (Adobe acrobat file)
gif (graphic file)
jpg (image file)
htm (web page file)
rtf (word processing file)
zip (Zipped, compressed file)
doc (Microsoft Word word processing file)
xls (Microsoft Excel spreadsheet file)
ppt (Microsoft PowerPoint presentation file)

Directories

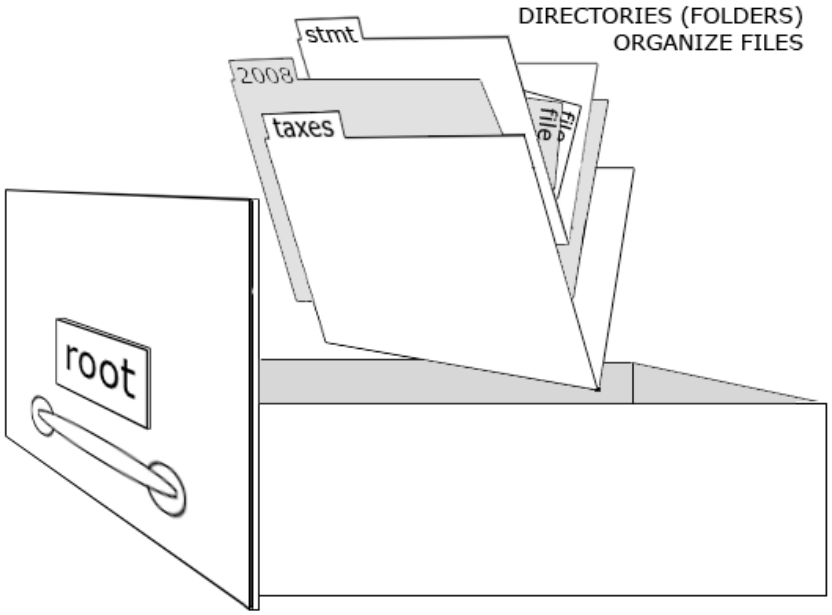
Just as you wouldn't dump all of your tax receipts into one shoe box, you shouldn't throw all of your program and data files randomly onto a disk.

DOS uses directories to organize files. Windows calls these folders, but they are the same thing.

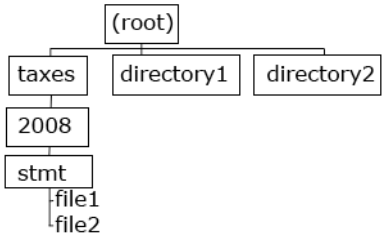
When you format a disk, part of the process is to create the 'root directory.' The root directory become your filing cabinet, and will then hold any directories or files you wish to create and store there.

Think how paper files are organized. You open a filing cabinet drawer, and flip though folders until you locate the logical place to drop your file. Just as you can place a folder inside of another folder, you can place a directory inside another directory. This is called a subdirectory, but it is the same thing.

Perhaps you are doing your taxes. After looking through your file cabinet, you have no good place for all the receipts. You make a directory called Taxes, and in that directory create a subdirectory called 2008 for the year. In that directory, perhaps you create subdirectories to hold tax forms, another for bank statements, and perhaps another for receipts.



A graphical representation of above, using a tree structured chart



The tree, as described in DOS

```
(root of C) C:\┐
               ├taxes ┐
               │      ├2008 ┐
               │      │     ├stmt ┐
               │      │     │     ├ file1.txt
               │      │     │     ├ file2.txt
```

Path to file1.txt `c:\taxes\2008\stmt\file1.txt`

Chapter 3

Introduction to DOS/Windows Command Prompt Commands

HELP	28
command /?	28
FORMAT	29
Optional parameters	30
Syntax	30
Changing Drives	31
Command Prompt change	31
Exploring and using Directories	31
DIR	31
MD	31
CD	32
CD .. and CD \	33
RD	34
Intro to COPY	34
TYPE	35
DEL	36
Paths	36
Limits of DOS	37

NOTE: IF YOU ARE NOT USING A FLOPPY DISK, PAY ATTENTION TO EXTRA NOTES IN THE LABS

HELP

The only command you need to remember is HELP.
Type in HELP at the command prompt; then press the Enter key to start a small program that lists all the commands that are available.

DOS Help

Choose the command and press the Enter key; a description of the command is displayed.

Note: This workbook focuses on only a few of the available commands

DOS 6.22 Help Window

```
      File  Search                                                Help
                        MS-DOS Help: Command Reference

Use the scroll bars to see more commands. Or, press the PAGE DOWN key. For
more information about using MS-DOS Help, choose How to Use MS-DOS Help
from the Help menu, or press F1. To exit MS-DOS Help, press Alt, F, X.

<What's New in MS-DOS 6.22?>

<ANSI.SYS>                    <EMM386.EXE>                  <Multi-config>
<Append>                      <Erase>                       <Nlsfunc>
<Attrib>                      <Exit>                        <Numlock>
<Batch commands>              <Expand>                      <Path>
<Break>                       <Fasthelp>                    <Pause>
<Buffers>                     <Fastopen>                    <Power>
<Call>                        <Fc>                          <POWER.EXE>
<Cd>                          <Fcbs>                        <Print>
<Chcp>                        <Fdisk>                       <Prompt>
<Chdir>                       <Files>                       <Qbasic>
<Chkdsk>                      <Find>                        <RAMDRIVE.SYS>
<CHKSTATE.SYS>                <For>                         <Rd>
<Choice>                      <Format>                      <Rem>
<Cls>                         <Goto>                        <Ren>
<Command>                     <Graphics>                    <Rename>
<Alt+C=Contents> <Alt+N=Nest> <Alt+B=Back>
```

There are many more commands than will fit on the screen. If you are using DOS with no mouse, you use the arrow keys to move into the command you wish to find out about, and then press the enter key.

DOS and Windows HELP alternative

If you know the command you wish to find out about, instead of starting HELP, you may type out the command followed by a space, then /?

Example (Note: do not type the C:\>… that is the system prompt)

```
C:\>FORMAT /?
```

This will display HELP for just this command, including optional parameters and syntax, then return to the command prompt.

More on optional parameters and syntax after the Windows HELP screen, in FORMAT.

Windows Help

If using Windows command prompt, the HELP screen will look similar to:

```
C:\>help
For more information on a specific command, type HELP command-name
ASSOC    Displays or modifies file extension associations.
AT       Schedules commands and programs to run on a computer.
ATTRIB   Displays or changes file attributes.
BREAK    Sets or clears extended CTRL+C checking.
CACLS    Displays or modifies access control lists (ACLs) of files.
CALL     Calls one batch program from another.
CD       Displays the name of or changes the current directory.
CHCP     Displays or sets the active code page number.
CHDIR    Displays the name of or changes the current directory.
CHKDSK   Checks a disk and displays a status report.
CHKNTFS  Displays or modifies the checking of disk at boot time.
CLS      Clears the screen.
CMD      Starts a new instance of the Windows command interpreter.
COLOR    Sets the default console foreground and background colors.
COMP     Compares the contents of two files or sets of files.
COMPACT  Displays or alters the compression of files on NTFS partitions.
CONVERT  Converts FAT volumes to NTFS.  You cannot convert the
         current drive.
COPY     Copies one or more files to another location.
DATE     Displays or sets the date.
DEL      Deletes one or more files.
DIR      Displays a list of files and subdirectories in a directory.
DISKCOMP Compares the contents of two floppy disks.
DISKCOPY Copies the contents of one floppy disk to another.
DOSKEY   Edits command lines, recalls Windows commands, and creates
macros.
ECHO     Displays messages, or turns command echoing on or off.
ENDLOCAL Ends localization of environment changes in a batch file.
ERASE    Deletes one or more files.
EXIT     Quits the CMD.EXE program (command interpreter).
FC       Compares two files or sets of files, and displays the differences
         between them.
FIND     Searches for a text string in a file or files.
FINDSTR  Searches for strings in files.
FOR      Runs a specified command for each file in a set of files.
FORMAT   Formats a disk for use with Windows.
```

Again, this is not the full list, just what may fit on the screen.
To find usage, use `HELP command-name` or `command-name /?`

Note: You may use lower case or upper case when typing commands.
A list of all the DOS commands & usage is in the DOS Appendix, right after Chapter 9

FORMAT (Not to be used on hard disk or thumb drive)

Since it is not a good idea to mess around with files on the hard drive, it would be a good idea to use a floppy disk, if available. To prepare a floppy disk for use with DOS or Windows, it needs to be formatted.

The way this workbook will relay information on commands will following this model:
Command Name:
Purpose:
Syntax, including common parameters
Example

Command Name	FORMAT
Purpose	To prepare a disk for use
Syntax	FORMAT *drive*:
Common parameters	/q (quick format on a previously formatted disk)
Example	`FORMAT A:` or `FORMAT A: /q` Note: you have the option to label the disk

→*What is syntax*?
Syntax is **how** a command can be used, but shouldn't be a specific example...
it should be more of a guide, or a fill in the blank starting point.

In the format command above, the syntax includes the required elements, but uses the word *drive* as a place holder for an actual drive letter.

In this workbook, the syntax is similar to what HELP would provide; future syntax examples will include optional items that *may* be included.

→*What are common parameters?*
Common parameters *may* be added at the end of the command to tweak the command.

In the example above, you *may* add /q IF the disk has previously been formatted, and IF you prefer to do the quick format instead of the regular format.

Other brief examples of syntax and common parameters
md
md makes directories... but you can't just type in md by itself;
the syntax is **md** ***foldername***, indicating you need to use the command, *and* provide the name of the directory you are making
an example would be **md lab1**

dir
dir generates a directory listing
you can use **dir** by itself, but you can *also* do
dir /w for a wide directory listing
or
dir /p for a directory listing that pauses after each full screen;
so you might put the syntax as
dir [/w] [/p]
with the bracket items being optional

Changing Drives

Once your floppy disk is formatted, you may tell the computer to begin paying attention to the drive. To change drives, type in the drive letter and a ':', then press the enter key.

Example (Note: do not type the C:\>… that is the system prompt)

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

You will know your computer is now paying attention to the floppy disk as the system prompt will change to reflect the current drive and directory

```
A:\>_
```

Note: in the above example, the current drive is A: and the current directory is the root… recall, you connect drives to directories using the '\' If no other directory is showing, you must be viewing the root… so read A:\ as the root of the A drive.

Exploring and using Directories

Now that the computer is paying attention to the A: drive, we need to see what is on the A: drive… this is done using the DIR command.

Command Name	DIR
Purpose	Generates a directory listing
Syntax	DIR [*drive*:\][*directory-name*]
Common parameters	/w (wide directory listing) /p (directory listing pauses after each full screen)
Example	DIR or DIR /w or DIR /p

Go ahead and try DIR.
Example (Note: do not type the A:\>… that is the system prompt)

```
A:\>DIR
```

If you do not specify a drive or directory, DOS used the path in the prompt, A:\ You should get a message that no files were found… this shouldn't be a surprise as the disk was just formatted. Let's make a directory named test.

Command Name	MD (or MKDIR)
Purpose	Makes a directory
Syntax	MD [*drive*:\][*directory-name*\] directory-name
Common parameters	n/a
Example	MD TEST

Example (Note: do not type the A:\>… that is the system prompt)

```
A:\>MD test
```

No news at the system prompt is usually good news… as we often do not get confirmation that anything happened.

To verify the directory was created run DIR again… you should see something like the following:

```
A:\>DIR

03/19/2008  10:26 AM    <DIR>          test
               0 File(s)              0 bytes
               1 Dir(s)   1.400 bytes free
```

You should see buried in all the information <DIR> test.

This verifies a directory named test exists. Let's see what is inside

Command Name	CD
Purpose	To change directories
Syntax	CD [*drive*:\][*directory-name*\] *directory name*
Common parameters	n/a
Example	CD TEST

```
A:\>CD test
```

Should yield the new system prompt

```
A:\TEST>_
```

Indicating the computer is now paying attention to the test directory on the root of A:

Fire off a DIR. You may expect to see an empty directory, but instead you see something like:

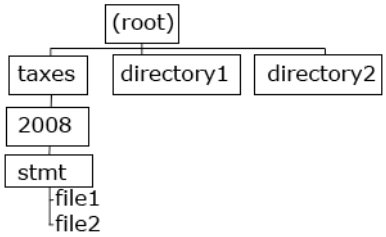
```
A:\TEST>DIR

03/19/2008  10:26 AM    <DIR>          .
03/19/2008  10:26 AM    <DIR>          ..

               0 File(s)              0 bytes
               2 Dir(s)   1.400 bytes free
```

For now, ignore the . Try the following

```
A:\TEST>CD ..
```

Your system prompt should now be back at the root of A:
So, **CD directory-name** changes us **into** a directory, and **CD ..** changes us **back out** of the directory.

If we had something like the following:



And our prompt indicated we are in the stmt directory

```
A:\taxes\2008\STMT>_
```

Then CD .. would back us out of STMT to 2008

```
A:\taxes\2008\STMT>CD ..
```

Changing the system prompt to

```
A:\taxes\2008\>_
```

What if we quickly wanted to change to the root directory ?
Recall, the root directory of A: is indicates as A:\
So to change directories to the root, use CD \

```
A:\taxes\2008\STMT>CD \
```

Don't do this if working on a hard drive or thumb drive, use cd ..
And our prompt will then indicates we are now in the root directory

```
A:\>_
```

Let's modify our CD syntax to include the following

Command Name	CD
Purpose	To change directories
Syntax	CD [*drive*:\][*directory-name*\] *directory name*
Common parameters	n/a
Example	`CD TEST` `CD ..` to change out of the current directory `CD \` to change to the root directory (for floppy)

Now, lets get rid of that empty directory; but first, an analogy.
If a farmer has an old barn that may fall over, he may instead decide to demolish the barn. So, he loads up the barn with TNT.
Do you suppose he wants to be in that barn when he blows it up? No.

The same logic is applied to formatting drives and removing directories… your system prompt MUST indicate you are NOT in the directory… so use CD .. first

Command Name	RD (or RMDIR)
Purpose	To remove an empty directory
Syntax	RD [drive:\][directory-name\] *directory name*
Common parameters	n/a
Example	RD TEST

Up until now, all we have dealt with are directories, and directories are only used to organize files. We need some files to look at. The problem is of course, a newly formatted disk has no files. Our solution is to use copy.

Note: a more detailed discussion of copy will follow.

Intro to COPY

Command Name	COPY
Purpose	Duplicates files
Syntax	COPY [*drive*:][*directory-name*\]*source* [*drive*:][*directory-name*\]*destination* COPY *source destination*
Common parameters	n/a
Example	COPY file1.txt file2.txt

Any place you see ***Source*** in this book, it means
"where is the file now, and what is it called?".
Destination means
"where is the file going, and what will it be called?" You may specify exactly where the file is, but you may omit anything that is part of your prompt… so from here on out I'll use COPY *source destination* instead of

COPY [*drive*:][*directory-name*\]*source* [*drive*:][*directory-name*\]*destination*
with the understanding that I could include specifics, if needed.

At first glance, the syntax seems lacking. Let's say for the time being *source* simply tells the computer where to find the data to be copied and *destination* tells the computer where you want the file to be copied, and what you want to call it.

But this still seems to leave us short, as we have no data to copy.

The magic solution? The keyboard can be a source; recall the reserved device name for the keyboard, (in the Hardware section of Chapter 2) is CON.

Now we can plug in CON for source, and let's call the file we are copying from the keyboard to the disk test.txt.
Change the syntax of
COPY *source destination* to
COPY CON test.txt

When the command is entered, you get a flashing cursor without a prompt… this is where you type the source data.
A special keyboard sequence will be used to signal the end of typing.

Example

```
A:\>COPY CON TEST.TXT
```

At the flashing cursor, type in something like your name

```
A:\>COPY CON TEST.TXT
_
```

```
A:\>COPY CON TEST.TXT
Craig
```

When finished, press and hold the Control key, press the Z key, and let go.

Recall, the Control key turns other keys into commands.
One way to remember this sequence is to say you are finished typing in your data from *A to* **Z**… Control+Z
Note: The computer displays ^Z, but you can't type ^Z… it is Control+Z

When finished you should see something similar to

```
A:\>COPY CON TEST.TXT
Craig ^Z

One file copied
A:\>_
```

To verify the file was created, at the flashing system prompt type in DIR

```
A:\>DIR

03/19/2008  11:29 AM                 5 test.txt
               1 File(s)              5 bytes
```

Recall, a byte is a typed character.
Craig has 5 typed characters, hence 5 bytes for my example.

But how can we read the contents of the file, to verify it contains what we typed?

Command Name	TYPE
Purpose	To display the contents of a text file
Syntax	TYPE [drive:\][directory-name\] *file-name*
Common parameters	n/a
Example	TYPE TEST.TXT

To verify the file, at the flashing system prompt type TYPE TEST.TXT
The contents of the text file will display, then return to the prompt

```
A:\>TYPE TEST.TXT
Craig
A:\>_
```

Finally to wrap up our introduction to commands, let's get rid of the test.txt file

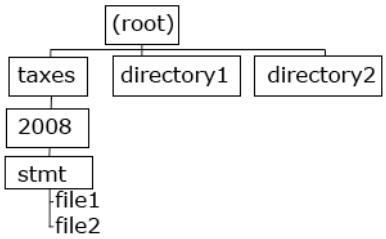
Command Name	DEL
Purpose	To delete a file
Syntax	DEL [drive:\][directory-name\] *file-name*
Common parameters	n/a
Example	DEL TEST.TXT

```
A:\>DEL TEST.TXT
```

A quick DIR should verify the file is gone… and your disk should be back the way it was before you began.

Paths

Any place you encounter
[drive:\][directory-name\] *file-name*
in syntax, this means you ***may*** supply a drive letter, or directory names.



If our prompt is
A:\directory1\>
But we wish to type out file1.txt
We could
cd ..
cd taxes
cd 2008
cd stmt
type file1.txt

Or we could simply use the path, and supply drive and directories info
type a:\taxes\2008\stmt\file1.txt

Anything that is part of the system prompt may be left out of the command.

Visualizing a path in Windows using 'My Computer'

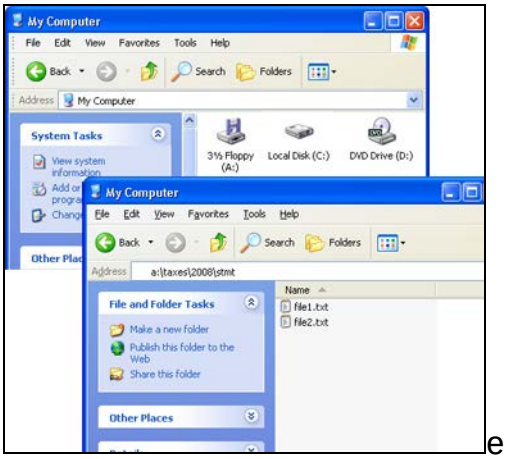
If you were to do the above example in Windows

- Open My Computer
- Select (double click) the 3 1/2 Floppy (A:) to see the lab1,.
- Select (double click) the lab1 folder to see the taxes folder
- Select (double click) the taxes folder to see file1.txt, and file2.txt

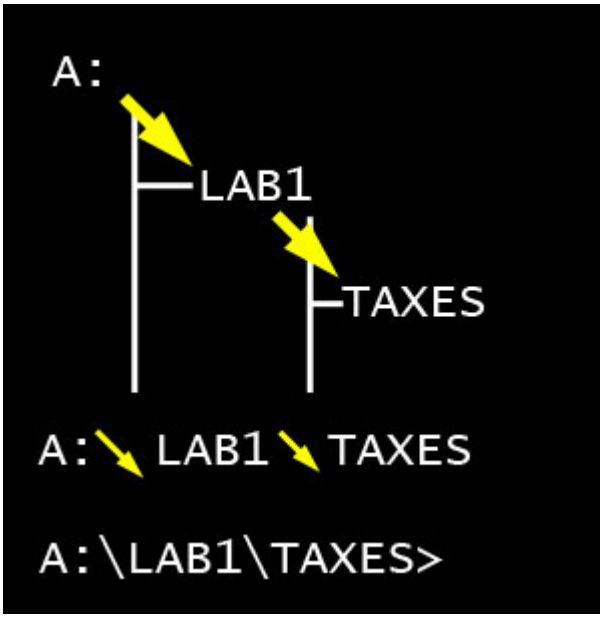
As you do double click to open drives and folders, the address bar keeps track of where the computer is looking...

The \ (backslash) is used to connect the drive to a folder, a folder to another folder, or a folder to a file. The full path for file1.txt would be: A:\LAB1\TAXES\file1.txt

start



end



Limits of DOS

While for pedagogical reasons, DOS is a great way to start command lines, DOS went away for a reason… it was limited in what it could do.

DOS can only do one thing at a time.
DOS file names are limited to 8 characters, with 3 characters for the extension… no spaces allowed.
DOS cannot efficiently use more than 640KB RAM
DOS cannot efficiently use large hard drives
DOS cannot use many modern input and output devices
No modern software is designed to run on DOS.

None of these limits prevents one from using DOS as a learning tool.

The Windows command prompt does not have the above limits, but really doesn't add anything from a learning point of view.

Chapter 4

If no DOS access
Start\All Programs\Accessories\Command Prompt

Lab	
Lab 1	40
Zipping	44
Renaming	44
Dropbox	44

At the end of this process, you will be turning in answers using the Quizzes area of Desire2Learn, based on the following questions. So, to get the best score, fill in all the answers in this document first.

1. FORMAT
What does the command do?
What is the syntax for the command?
(Syntax does not include specific drives or directories)
Write out one example of how the command is entered into DOS

2. MKDIR or MD
What does the command do?
What is the syntax for the command?
(Syntax does not include specific drives or directories)
Write out one example of how the command is entered into DOS

3. CD
What does the command do?
What is the syntax for the command?
(Syntax does not include specific drives or directories)
What does cd .. mean? What does cd \ mean?
Write out one example of how the command is entered into DOS

4. DIR
What does the command do?
What is the syntax for the command?
(Syntax does not include specific drives or directories)
Are there any common parameters mentioned in class? List them
Write out one example of how the command is entered into DOS

From Appendix A
5. TREE
What does the command do?
What is the syntax for the command?
(Syntax does not include specific drives or directories)
Are there any common parameters mentioned in class? List them
Write out one example of how the command is entered into DOS

See Chapter 3 for other useful commands, such as copy and type.

Check the Class website for a short video similar to Lab1

Activity

Recreate the following in DOS or the Windows command prompt, using the steps below.

```
A:\>tree /f
Folder PATH listing for volume
Volume serial number is
A:.
└───lab1
    └───taxes
        ├───2007
        │   ├───qtr1
        │   ├───qtr2
        │   ├───qtr3
        │   └───qtr4
        │           bankstmt.txt
        └───2008
```

Don't type the part of the instruction that represents the prompt, such as **C:\>** or **A:\>**

Optional: Format your floppy disk, if needed; DO NOT FORMAT C: or thumbdrives
C:\>**format a:**

☐ 1. Change to your drive (any available drive; examples use A:, If working with no floppy you may use O:, or USB drive , ie **f:**, then **md floppy**, then **cd floppy**)
C:\>**a:**

☐ 2. Make a Directory (folder) named lab1
A:\>**md lab1** (Again A:\> is not part of the command, your prompt may differ.)

☐ 3. Verify the Directory was created by requested a Directory Listing
A:\>**dir**

☐ 4. Change into the Directory named lab1
A:\>**cd lab1**

☐ 5. Make a Directory (folder) named taxes
A:\lab1>**md taxes**

☐ 6. Verify the Directory was created by requested a Directory Listing
A:\lab1>**dir**

☐ 7. Change into the Directory named taxes
A:\lab1>**cd taxes**

☐ 8. Make a Directory named 2007
A:\lab1\taxes>**md 2007**

☐ 9. Make a Directory named 2008
A:\lab1\taxes>**md 2008**

☐ 10. Change into the Directory named 2007
A:\lab1\taxes>**cd 2007**

☐ 11. Make a directory for Quarter 1
A:\lab1\taxes\2007>**md qtr1**

(continued)

☐ 12. Make a directory for Quarter 2

A:\lab1\taxes\2007>**md qtr2**

☐ 13. Make a directory for Quarter 3

A:\lab1\taxes\2007>**md qtr3**

☐ 14. Make a directory for Quarter 4

A:\lab1\taxes\2007>**md qtr4**

☐ 15. Change into the Directory named qtr4

A:\lab1\taxes\2007>**cd qtr4**

☐ 16. Copy data from the Console to a file named bankstmt.txt

A:\lab1\taxes\2007\qtr4>**copy con bankstmt.txt** (then Enter)

Type your name

Press **[Ctl]+Z** to end the file, and return to DOS

1 file(s) copied.

☐ 17. To read the contents of the file

A:\lab1\taxes\2007\qtr4>**type bankstmt.txt**

☐ 18. Change into the directory one step higher than qtr 4

A:\lab1\taxes\2007\qtr4>**cd ..**

☐ 19. If using a floppy, change to the root directory (\) in one step

A:\lab1\taxes\2007>**cd **

(if not using floppy, cd .. and repeat until at the starting point, such as **e:\floppy>**)

☐ 20. Display the entire disk structure with files

A:\>**tree /f**

Your screen should look like the image.

```
A:\>tree /f
Folder PATH listing for volume
Volume serial number is
A:.
└───lab1
    └───taxes
        ├───2007
        │   ├───qtr1
        │   ├───qtr2
        │   ├───qtr3
        │   └───qtr4
        │           bankstmt.txt
        └───2008
```

Hands On (the Quiz in D2L is 60% of your grade)

Answer the following questions based on this lab to prepare you for the D2L quiz

Question 1 (2 points)

The format command

a) displays the contents of a directory, including sub-directories
b) prepares a disk for use
c) changes into to a directory
d) generates a listing of the specified directory
e) creates a directory

Question 2 (2 points)
Syntax: format
a) format source destination
b) format [drive]:
c) format directoryname
d) format source

Question 3 (2 points)
What is typed after the C:\> prompt to format the floppy?
a) a: format b) a:\ format
c) format a: d) format a:\

Question 4 (2 points)
The MD (or MKDIR) command:
a) changes into to a directory
b) prepares a disk for use
d) generates a listing of the specified directory
c) displays the contents of a directory, including sub-directories
e) creates a directory

Question 5 (2 points)
Syntax: MD
a) md /w b) md directoryname
c) md source destination d) md [drive]:

Question 6 (2 points)
What would you enter after the prompt to make or create a directory named junk?
a) md a: b) rd junk
d) cd junk d) md junk

Question 7 (2 points)
The command CD
a) creates a directory b) changes into to a directory
c) displays the contents of a directory, including sub-directories
d) prepares a disk for use e) generates a listing of the specified directory

Question 8 (2 points)
Syntax: CD
a) cd /w b) cd source destination
d) cd directoryname d) cd [drive]:

Question 9 (2 points)
The following is the correct use of change directory **cd a:**
a) True b) False

Question 10 (2 points)
The DIR command
a) creates a directory
b) displays the contents of a directory, including contents of sub-directories
c) changes into to a directory
d) generates a listing of the specified directory
e) prepares a disk for use

Question 11 (2 points)
Syntax: DIR
a) dir [directoryname] [/w] [/p] b) dir source destination
c) dir d) dir [drive]:

Question 12 (2 points)
To get your directory listin in columns across the whole screen is
a) dir /p b) dir /w
c) dir d) p /dir
e) w /dir

Question 13 (2 points)
The tree command
a) displays the contents of a directory, including sub-directories
b) creates a directory
c) generates a listing of the specified directory
d) changes into to a directory e) prepares a disk for use

Question 14 (2 points)
Syntax: TREE
a) tree [directoryname] [/f] b) tree /w
c) tree [drive]: d) tree source destination

Question 15 (2 points)
Which DOS command shows the contents of a directory, as well as the file contents of any sub-directory?
a) tree /f b) dir c) tree d) dir /f

Question 16 (3 points)
What would be typed after the prompt to get help with all the DOS commands?
a) man b) DOS /? c) help d) F1

Question 17 (3 points)
What is the command you would enter in DOS to format the floppy, if needed?
Not just the command but what needs to be typed after the prompt to actually format the floppy c:\>_____

Question 18 (3 points)
Enter what would be typed after the prompt change from the C drive to the A drive
c:\>___

Question 19 (3 points)
Enter what would be typed after the prompt to generate a directory listing
c:\>____

The remaining questions assume you are using the floppy disk to complete the activity DO NOT INCLUDE THE A:\> prompt when choosing your responses.

Question 20 (1.5 points)
What would be typed after the A:\> prompt to change from the root to the lab 1 directory?
a) md lab 1 b) cd lab 1 c) rd lab 1 d) lab1

Question 21 (1.5 points)
What would be typed after the A:\lab1> prompt to change from the lab1 directory to the taxes directory?
a) md taxes b) cd taxes c) rd taxes d) taxes

Question 22 (1.5 points)
What would be typed after the A:\lab1\taxes> prompt to change from the taxes directory to the 2007 directory?
a) cd taxes\2007 b) cd 2007 c) 2007 d) md taxes\2007

Question 23 (1.5 points)
What would be typed after the A:\lab1\taxes\2007> prompt to change from the 2007 directory to the qtr4 directory?
a) cd qtr4 b) md taxes\2007\qtr4 c) cd taxes\2007\qtr4 d) qtr4

Question 24 (3 points)
What would be typed after the prompt to display the contents of the bankstmt.txt file?
a:\lab1\2007\qtr4>____ bankstmt.txt *Only supply the missing command.*

Question 25 (3 points)
Enter what would be typed after the prompt to change from the qtr4 directory to the 2007 directory *NOTE: the cd command is already there... what goes after cd?*
a:\lab1\2007\qtr4>cd __

Question 26 (3 points)
Enter what would be typed after the prompt to change from the 2007 directory to the root directory *Note: the cd is already there, what goes after the cd?*
a:\lab1\2007\>cd __

In Windows, zip the entire lab1 directory, and rename to yourname-lab1.zip.

Zipping files

When you wish to email or upload several files and folders, the best way to do this is to 'zip' them.

Using My Computer in Windows, select the file(s) you wish to zip
1. Right click the selected item
   (if you have selected several items, right click one of them)
2. Choose Send To
3. Choose Compressed (zipped) Folder
4. The new file will be created in the same location
5. Rename the file to the form *yourname*-lab*x*.zip

Renaming files

When you wish rename a file:
1. Select the file you wish to rename
2. Right click the file
3. Choose Rename

(Note the three character extension if visible, before proceeding)
4. Once the name is selected you type in the new filename to replace the old
5. Make sure the extension, if visible, remains the same
6. When finished, press the [Enter] key.

Submitting labs in D2L

1. Choose the class for which you want to drop off an assignment.
2. Find the Dropbox tool, click on it.
3. Choose the appropriate Dropbox folder, click on it.
4. Choose [Add a File], then choose [Browse]
   In Vista or Windows 7, you may need to choose ▼*Browse Folders*
   You then may need to choose *Computer* to browse for your file
5. Find the file that you would like to upload and click open, then choose [Upload]
6. Type in a short description or comment about the file, if you like.
7. Select [Add] for any additional files that may be pertinent, and repeat
8. After your files have been selected, click [Submit].
   (You will always click upload AND submit)
9. Some files may take a long time to upload
   (especially if there are graphics in the file).

Take the quiz for lab 1 in D2L

Choose the class for which you want to take the quiz.
Choose the Quizzes menu, and locate Lab 1.
Answer the questions based on this documentation.
SAVE EACH ANSWER. WHEN FINISHED, PRESS SUBMIT

Chapter 5

Internal DOS Commands/External DOS Commands

Internal DOS Commands	46
External DOS Commands	48
Tree	50
Wildcards	51
XCOPY	53

As DOS must fit in a very small chunk of RAM, it probably is not surprising that not all DOS commands get loaded into memory during the boot process. Only the commands that are a part of command.com are in RAM at all times. The remaining DOS commands typically reside in a directory on the C: drive called DOS (C:\DOS), and are loaded into RAM when the command is actually used.

For all practical purposes, the most common commands are internal, and the newer or less frequently used commands are external.

As far as usage goes, however, there is no difference.

INTERNAL DOS COMMANDS

A short list of the most frequently used internal commands, many of which were covered in Chapter 3.

CLS, used to clear the screen
    Syntax CLS
COPY, used to duplicate files
    Syntax, see Chapter 3
DATE, used to display or change the system calendar
    Syntax DATE [mm-dd-yy]
DEL, used to remove a file
    Syntax, see Chapter 3
MD, used to make an empty directory (may use MKDIR)
    Syntax, see Chapter 3
RD, used to remove an empty directory (may use RMDIR)
    Syntax, see Chapter 3
DIR, used to generate a directory listing
    Syntax, see Chapter 3
REN (RENAME), used to rename a file
    Syntax REN *source destination*
TIME, used to display or change the system clock
    Syntax TIME [hh:[mm:[ss]]]
TYPE, used to display the contents of a text file
    Syntax, see Chapter 3
VER, used to display the DOS or Windows version
    Syntax VER
VOL, used to display the volume label
    (labels are normally set with FORMAT, and changed with LABEL)
    Syntax VOL [drive:]

Really useful 'new' Internal commands.

After entering a lot of DOS commands, your screen is junked up with a record of previous commands given; to wipe the screen clean and get back to a pristine system prompt, use CLS

Command Name	CLS
Purpose	Clears the screen
Syntax	CLS
Common parameters	n/a
Example	CLS

While COPY technically was introduced in lab 1, we used a very limited subset of COPY… copying from the keyboard (CON).

Command Name	COPY
Purpose	Duplicates files
Syntax	COPY *[path\]source [path]destination*
Common parameters	n/a
Example	`COPY file1.txt file2.txt`

A few rules.

1. Memorize ***copy source destination***, you will use it often.
2. If no path is specified, it is assumed you mean the path shown in the system prompt.
3. No files can have the same full filename in the same directory, but files in different directories may have the same full file name.

Some examples of COPY

```
A:\> COPY file1.txt file2.txt
```

copies **file1.txt** in the current directory (A:\), to the same directory (A:\) and calls the duplicated file **file2.txt** (the full file name is different)

```
A:\> COPY file1.txt file1.ltr
```

copies **file1.txt** in the current directory (A:\), to the same directory (A:\) and calls the duplicated file **file1.ltr** (the full file name is different)

```
A:\> COPY file1.txt C:\file1.txt
```

copies **file1.txt** in the current directory, (A:\), to C:'s root (C:\) and calls the duplicated file **file1.txt**

```
A:\> COPY C:\file1.txt file1.txt
```

copies **file1.txt** in C:'s root (C:\) to the current directory (A:\), and calls the duplicated file **file1.txt**

Rename behaviors in a similar fashion, REN source destination, but there is one caveat… you identify where the file is with the source, so **no path is allowed for the destination.**

Command Name	REN or RENAME
Purpose	Renames files
Syntax	REN *[path\]source destination*
Common parameters	n/a
Example	`REN file1.txt file2.txt`

Some examples of REN

```
A:\> REN file1.txt file2.ltr
```

takes **file1.txt** in the current directory (A:\),
and renames the file to **file2.ltr**

```
A:\> REN C:\file1.txt file2.ltr
```

takes **file1.txt** in the C:'s root (C:\),
and renames the file to **file2.ltr**

EXTERNAL DOS COMMANDS

Available commands not loaded into RAM at boot include the following:

ATTRIB, used to display or change attributes
    Syntax ATTRIB [±R or A or S or H] [filename]
CHKDSK, used to display a status report of a disk, or
    Fix errors in the File Allocation table
    Syntax CHKDSK [drive:] [/f]
    Replaced by SCANDISK
*DELTREE, *not available in all versions of Windows*, used to remove
    non-empty directories DANGEROUS, use with caution
    Syntax DELTREE [path]
DISKCOPY, used to copy a floppy
    Syntax DISKCOPY source destination
LABEL, used to change a disk's volume label
    Syntax LABEL [drive:] [*new volume label*]
TREE, used to display directories, but unlike DIR,
    Tree can also display the contents of subdirectories (files)
    Syntax TREE [path] [/f]
*UNDELETE, *not available in Windows* (replaced by the recycle bin)
    May be able to recover recently deleted files
    Syntax UNDELETE [path\] [filename]
XCOPY, used to copy files and directories, including empty directories
    Syntax XCOPY source destination [/e]

Attrib is for attributes. The attributes a file can have are
Read Only: the file cannot be deleted nor existing material changed
Archive: the file has never been backed up and is in need of archiving
System: a DOS file
Hidden: the file is not visible to any DOS command aside from ATTRIB

Neither read only or hidden files can be deleted, so if you are getting an error, check attrib.

Command Name	ATTRIB
Purpose	used to display or change attributes
Syntax	ATTRIB [±R or A or S or H] [path\][filename]
Common parameters	n/a
Example	`ATTRIB +R file1.txt`

Some examples of ATTRIB

```
A:\> ATTRIB

RA      file1.txt
```

Displays all the attributes of files in the current directory;
In this case file1.txt is Read Only, and has never been backed up.

```
A:\> ATTRIB -R file1.txt
```

Removes the Read Only attribute from **file1.txt**

```
A:\> ATTRIB +H file1.txt
```

Adds the Hidden attribute to **file1.txt**
Not only is the file protected from deletion, it will not list using DIR, etc.

DELTREE is worth mentioning, even if it is no longer used in Windows.
As RD will not delete a directory with contents, DELTREE was how you could quickly delete a bunch of stuff… and perhaps including items you didn't want to get rid of.

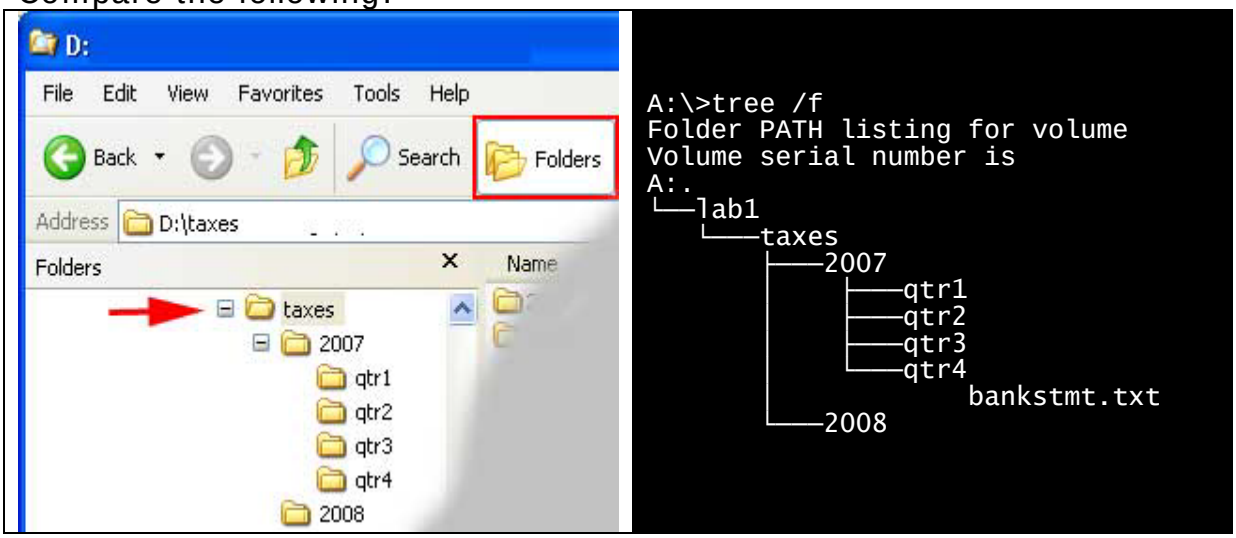
DISKCOPY is handy if you want to make copies of floppies. It is what is called a bit copy, that is, instead of copying files, it copies the pattern of 1s and 0s on the disk. The disks must match, as in 3.5 inch 1.44MB floppy and another 3.5 inch 1.44MB floppy. As many computers only have one drive, half way through you are prompted to replace the source with the destination… meaning the copy temporarily resides in RAM. If a disk has bad sectors and CHKDSK has marked those sectors to be avoided, the copy will also avoid the same sectors.

Command Name	DISKCOPY
Purpose	Duplicates floppy disks
Syntax	DISKCOPY *source destination*
Common parameters	n/a
Example	`DISKCOPY A: A:`

TREE

A great command, snuck in to Lab 1, is TREE. By itself, it is little more than a glorified DIR… but jump to the root and add the /f, and this command can show you every directory on your disk, and every file in those directories… DIR can only show you the contents of the current directory. This command has no equal in Windows.

Compare the following:



In the Windows My Computer view, the folders button has been turned on to let you see all of the subdirectories stored in the 'taxes' folder… the problem is, that is all you can see… folders. You still must open each folder to see if there are any files.

Now look at the tree /f command results to the right. Not only can you see the directories in taxes, but you can also see the single file that is there.

Command Name	TREE
Purpose	To display a tree structure
Syntax	TREE [drive:\][directory-name\]
Common parameters	/f (will display directories and directory contents)
Example	`Tree /F  or tree a:\ /f`

WILDCARDS

So while your GUI is handy, that TREE command is unequaled in Windows. And there is another trick in the command line that makes it a required skill for power users.

Before we begin discussing the next generation copy command, what if you wish to copy more than one file at a time, or for that matter, delete more than one file at a time, or perhaps rename more than one file at a time.

Windows finds it impossible to rename multiple files.

A wildcard can be used when a DOS command should be applied to multiple files.

The two wildcard are * and ?

*, the asterisk (computer people call it star) can mean anything, or nothing, and range over multiple characters in a filename.

?, the question mark can mean something, for a specific character position in a filename or file type.

Some quick examples.

If you wish to delete all files in a directory
DEL *.*
Read this as DEL any filename with any file type extension.

If you wish to delete all the files that end in .txt, use
DEL *.txt
Read this as DEL any filename as long as it ends in .txt

IMPORTANT RULE: If using a wildcard for the source, duplicate the same wildcard pattern for the destination.

If you wish to copy file1.txt and file2.txt, but **not** file10.txt or file.txt use
COPY a:\file?.txt c:\file?.txt
Note: the ? is in the 5th character spot. ? wont get file10.txt as it doesn't cover the 6th spot, and wont get file.txt because there is no 5th spot in the filename.
Read **COPY a:\file?.txt c:\file?.txt** as copy from the root of A: anything that starts with **file** followed by **one character**, that ends with **.txt**, and copy it using the same names to C:

Wildcard scenarios

Scenario: A:\ file.txt file1.txt file2.txt file3.txt file4.txt file10.txt fred.txt file.doc file1.doc file2.doc file3.doc file4.doc file10.doc fred.doc	If you wished to select any file, as long as it ends with .txt you would use: *.txt (could be read *anything*.txt) This would select file.txt file1.txt file2.txt file3.txt file4.txt file10.txt fred.txt but not any file that ends with .doc If you wished to select any txt file, as long as it starts with file you would use: file*.txt (could be read file*anything.txt*) This would select file.txt file1.txt file2.txt file3.txt file4.txt and file10.txt but not fred.txt or any file that ends with .doc
Scenario: A:\ file.txt file1.txt file2.txt file3.txt file4.txt file10.txt fred.txt file.doc file1.doc file2.doc file3.doc file4.doc file10.doc fred.doc	If you wished to select any file, as long as it starts with the letters file you would use: file*.* (could be read file*anything* with *any* extension) This would select file.txt file1.txt file2.txt file3.txt file4.txt file10.txt file.doc file1.doc file2.doc file3.doc file4.doc file10.doc but not fred.txt or fred.doc
Scenario: A:\ file.txt file1.txt file2.txt file3.txt file4.txt file10.txt fred.txt file.doc file1.doc file2.doc file3.doc file4.doc file10.doc fred.doc	If you wished to select any text file, as long as it includes a **single** digit number you would use: file?.txt (could be read file *something* ***if only in space number 5***.txt) This would select file1.txt file2.txt file3.txt file4.txt but not file.txt (no 5th character in the filename) or file10.txt (? works only for space 5, not space 6) or any file that ends with .doc If you wished to select **any** file, with **any** extension, (all) you would use: *.* (could be read *any* file with *any* extension)

Before we got sidetracked discussing wildcards, a newer version of the COPY command was mentioned. COPY is great at copying files, but not so good at copying directories, especially empty directories.

You never mess with an existing command, you come up with a new command that does what you want. That command is XCOPY.

This doesn't mean you abandon COPY. You don't drive a 2 penny nail with a 12 lb. sledge hammer… you use the appropriate tool for the job. When copying files, use COPY, when copying directory structures, use XCOPY.

And just as tree works best with /f, XCOPY works best with /e (think Everything, including Empty subdirectories).

XCOPY

Command Name	XCOPY
Purpose	Duplicates files and directories
Syntax	XCOPY *[path\]source [path]destination*
Common parameters	/e
Example	`XCOPY dir1\*.* dir2\*.* /e`

Note that the example above is copying dir1 AND all the files in dir1, using wildcards for the source; so the same wildcard pattern is used in the destination, followed by the /e.

Chapter 6

Lab

Lab 2	56
Lab 2 Cheat sheet	57
Lab 2 Questions	58
Zipping	44
Renaming	44
Dropbox	44

Make a folder named Lab2 on your floppy or O: drive.

At the end of this process, you will be turning in answers using the Quizzes area of Desire2Learn, based on the following questions. So, to get the best score, fill in all the answers in this document first.

Documentation 30%

COPY
What does the command do?
What is the syntax for the command?

(Syntax does not include specific drives or directories)
Write out one example of how the command was entered in this lab

XCOPY
What does the command do?
What is the syntax for the command?

(Syntax does not include specific drives or directories)
Are there any common parameters mentioned in class? List them
Write out one example of how the command was entered in this lab

ATTRIB
What does the command do?
What is the syntax for the command?

(Syntax does not include specific drives or directories)
Are there any common parameters mentioned in class? List them
Write out one example of how the command was entered in this lab

RENAME or REN
What does the command do?
What is the syntax for the command?

(Syntax does not include specific drives or directories)
Write out one example of how the command was entered in this lab

UNDELETE (DOS ONLY)
What does the command do?
What is the syntax for the command?

(Syntax does not include specific drives or directories)
Write out one example of how the command was entered in this lab

DOS Activity (There is a cheatsheet on the next page…)

☐ 1. Create a file called test.txt (put your name for the file contents)
Refer to Lab 1, for a reminder on creating a file.

☐ 2. Create a directory named LAB2

☐ 3. Create a directory named TEST inside of LAB2

☐ 4. Copy test.txt from the root of A: to inside LAB2

☐ 5. Copy test.txt from the root of A: to inside LAB2, but call the new copy test.ltr

☐ 6. XCOPY LAB2, and its files, to a new folder named LAB2BKUP
Verify the empty TEST folder was copied as well
At this point, if you run tree /f it should look like

```
A:\>tree /f
Folder PATH listing for volume
Volume serial number is
A:.
│   test.txt
│
├───lab2
│   │   test.txt
│   │   test.ltr
│   │
│   └───test
│
└───lab2bkup
    │   test.txt
    │   test.ltr
    │
    └───test
```

☐ 7. Set a:\test.txt to be a hidden file

☐ 8. Generate a directory listing to verify the file is hidden

☐ 9. Set a:\test.txt to no longer be a hidden file

☐ 10. Type out the contents of test.txt

☐ 11. Rename all of the files in LAB2 from test.*something* to homework.*something*, but leave the extensions the same (ONE command, using a wildcard)

☐ 12. Rename a:\test.txt to test.doc
* DOS only, skip if using command prompt: Delete test.doc then undelete test.doc
Clear your screen, then run a tree /f to see if your screen is similar to mine

Select ALL items: test.doc, lab2, and lab2bkup. Zip them together & place in the D2L Lab 2 dropbox.

Lab 2 Cheat sheet

☐ 1. In lab 1, we created a FILE by misusing the copy command. Instead of copying from one file, we can also copy from the keyboard, called `con`.

Copy syntax `copy source destination`
In lab one, we did `copy con bankstmt.txt`, so for lab two it would be `copy con ____.txt`

☐ 2. After `M`aking a `D`irectory `md ___`

☐ 3. if we wish to make another directory ***inside*** of it, we should first open, or `C`hange `D`irectors, or specify where it is going, and what it would be called `md lab2\ ___`

☐ 4. To copy your file from the root directory to a spot inside the lab2 directory, again consider the Copy syntax:
`copy source destination` which could be expanded to
`copy [path\]source [path\]destination`

The path is the where we look, perhaps starting with the drive, and directories, in order to point to a file.
Now look at what we are trying to make

```
A:\>tree /f
Folder PATH listing for volume
Volume serial number is
A:.
│   test.txt
│
├───lab2
│   │   test.txt
│   │   test.ltr
│   │
│   └───test
│
└───lab2bkup
    │   test.txt
    │   test.ltr
    │
    └───test
```

To locate the source, take your finger, and start at the upper left. That gives us the beginning of the path: `A:`
Remember, the directory that holds everything is the root directory, and it is represented with `\`
so, the root of the A drive is `A:\`

Every time you want to connect a drive to a directory, a directory to a directory, or a directory to a file, you are moving down the tree, sort of like ↘, so use `\`

So, the path to our source, where `test.txt` is right now, is
`A:\test.txt`
So, the path to where we want to copy that file would be `A:` + `\` + `lab2` + `\` or `A:\lab2\`
and since I didn't specify a different name, we would just reuse `test.txt` for a destination of
`A:\lab2\test.txt`

Plug this in to our syntax,
`copy [path\]source [path\]destination`
and you get
`copy a:\test.txt A:\lab2\test.txt`

☐ 5. If we want to rename the file as we copy it, plug the new name in to our syntax,
`copy [path\]source [path\]destination`
and you get
`copy a:\test.txt A:\lab2\test.ltr`

☐ 6. If we want to copy files and directories, including empty directories, the syntax is
`xcopy [path\]source [path\]destination /e,` or with wildcards an example would be

Cheat Sheet, continued

`xcopy lab2\*.* lab2bkup\*.* /e`

☐ 7. `ATTRIB [±R or A or S or H] [path\][filename]` leads to `ATTRIB +h test.txt`

☐ 11/12. If we want to rename a single file, the syntax is
`ren [path\]source destination`
So to rename a single file in the lab2 folder from test to homework, you would use
`ren A:\lab2\test.txt homework.txt`
But, to rename **multiple** files, we need to use a wildcard.
The logic is "I am trying to rename *test* files, regardless of the extension", so the wild card pattern would be
`test.*`
The logic continues "I am trying to rename *test* to *homework*, regardless of the extension", `homework.*`
So, the command should be
`ren A:\lab2\test.* homework.*`

```
A:\>tree /f
Folder PATH listing for volume
Volume serial number is
A:.
│   test.doc
│
├───lab2
│   │   homework.txt
│   │   homework.ltr
│   │
│   └───test
│
└───lab2bkup
    │   test.txt
    │   test.ltr
    │
    └───test
```

After completing the activity, answer the following to prepare for D2L quiz:

Question 1 (2 points)
The copy command duplicates files and directories, even empty directories
❍ renames files
❍ displays or changes attributes
❍ perhaps can retrieve an erased file
❍ duplicates files

Question 2 (2 points)
Syntax: copy
[remember, items in brackets are optional]
❍ copy [+r][-r][+a][-a][+s][-s][+h][-h] filename
❍ copy /w
❍ copy [filename]
❍ copy [path\]source [path\]destination

Question 3 (2 points)
An example of the proper use of the copy command is
❍ copy a:\test.txt a:\lab2\test.txt
❍ copy a:\test.txt a:\lab2\test.ltr
❍ All of the above ❍ none of the above

Question 4 (2 points)
The XCOPY command:
❍ duplicates files and directories, even empty directories
❍ renames files
❍ perhaps can retrieve an erased file
❍ displays or changes attributes
❍ duplicates files

Question 5 (2 points)
Syntax: XCOPY
❍ xcopy /w
❍ xcopy [+r][-r][+a][-a][+s][-s][+h][-h] filename
❍ xcopy [filename]
❍ xcopy [path\]source [path\]destination [/e]
Question 6 (2 points)
The command copy everything including empty subdirectories is
❍ xcopy lab2*.* lab2bkup*.* /f
❍ xcopy lab2*.* lab2bkup*.* /e
❍ xcopy lab2*.* lab2bkup*.*
❍ copy lab2*.* lab2bkup*.* /e
Question 7 (2 points)
The command ATTRIB
❍ duplicates files
❍ perhaps can retrieve an erased file
❍ renames files
❍ displays or changes attributes
❍ duplicates files and directories, even empty directories
Question 8 (2 points)
Syntax: ATTRIB
❍ attrib [path\]source destination
❍ attrib [path\]source [path\]destination
❍ attrib [+r][-r][+a][-a][+s][-s][+h][-h] [filename]
❍ attrib [filename]
Question 9 (2 points)
The following is the correct use of attribute command **attrib +ra test.txt**
❍ True
❍ False
Question 10 (2 points)
The REN command
❍ renames files
❍ duplicates files
❍ displays or changes attributes
❍ perhaps can retrieve an erased file
❍ duplicates files and directories, even empty directories
Question 11 (2 points)
Syntax: REN
❍ ren [path\]source [path\]destination
❍ ren [filename]
❍ ren [path\]source destination
❍ ren [+r][-r][+a][-a][+s][-s][+h][-h] filename
Question 12 (2 points)
To rename a:\red.txt to green.txt, you would type
❍ rename a:\red.txt a:\green.txt
❍ ren a:\red.txt a:\green.txt
❍ ren a:\red.txt green.txt
❍ rename a:\red.txt green.txt
Question 13 (2 points)
The undelete command
❍ displays or changes attributes
❍ perhaps can retrieve an erased file
❍ can retrieve any erased file
❍ duplicates files
❍ renames files
❍ duplicates files and directories, even empty directories

Question 14 (2 points)
Syntax: undelete
❍ undelete [+r][-r][+a][-a][+s][-s][+h][-h] filename
❍ undelete [path\][filename]
❍ undelete [path\]source [path\]destination
❍ undelete [path\]source destination
Question 15 (2 points)
The command to undelete a:\test.doc is
❍ undel ❍ undelete
Question 16 (6 points)

```
Current situation                    Desired result

C:\                                  C:\
   │                                    │   homework.txt
   │                                    │
   ├───DOS                              ├───DOS
   │       homework.txt                 │       homework.txt
   │                                    │
```

Scenario, your prompt is C:\>, there is a file called homework.txt in C:\DOS. Give the command to copy the file to the root of C without changing directories… that is, write out a copy command with a full path statement for the source and a full path statement for the destination.
❍ copy c:\dos\homework.txt c:\homework.txt
❍ copy c:\dos\homework.txt ..\homework.txt
❍ copy c:\homework.txt c:\dos\homework.txt
❍ This cannot be done with one command

Question 17 (6 points)
What is the command you would enter in DOS to copy all the files that are named homework in the lab2 directory to a lab3 directory, if that directory existed?
❍ copy lab2\homework.txt>ltr lab3\homework.txt>ltr
❍ copy lab2\homework.? lab3\homework.?
❍ copy lab2\homework.* lab\homework.*
❍ copy lab2*.* lab3*.*

Question 18 (6 points)
What is the command you would enter in DOS to copy a directory and all its contents, as in from lab2 to lab2bkup?
Only fill in the missing command
A:\>_____lab2*.* lab3*.* /e

Question 19 (6 points)
What is the command you would enter in DOS to change the attributes of a file named fred.txt to hidden?
A:\>_____ _____ _____

Question 20 (6 points)
What is the command you would enter in DOS to rename a file from test.txt to test.doc?
A:\>ren test.txt _____

Can you use a full path statement for the destination?
❍Yes
❍No?

Go to the Quizzes area of D2L and answer the Lab 2 questions, using this as a guide.
Select test.doc AND lab2 AND lab2bkup by holding down the Ctrl key, then zip
Don't forget you can look up how to zip files, rename files, etc., at the end of Lab 1

Chapter 7

Misc.: Hard Disk Management, Utilities, Redirection

Hard Disk Management 62
Utilities 64
Redirection 65

Hard Disk Management

Hard disks are used to store files, but directories actually organize all of these files. COPY and DEL can actually manage all the files on your disk, including rearranging files in various directories.

If a file is in directory A, but the file should be in directory B,
1) COPY the file in directory A to directory B, and then
2) DEL the original file

The problem was, if you didn't pay attention the copy didn't get created and the original would be deleted, leaving nothing… so most careful users actually made this a three part process… checking to see if the copy was made before the deletion was executed; making the actual command sequence COPY-DIR or TREE-DEL.

In the early days of DOS, this was sufficient. But as the OS matured, a one step solution that took the guesswork out of it was made: MOVE

Command Name	MOVE
Purpose	To move a file from
Syntax	MOVE source destination
Common parameters	n/a
Example	`MOVE A:\DIRA\file1.txt A:\DIRB\file1.txt`

Full paths may be used for both the source and destination.

As this was a new command, it was not added to command.com, instead, the executable command MOVE was added to the DOS directory as an external command.

But we did not type in
`A:\>C:\MOVE A:\DIRA\file1.txt A:\DIRB\file1.txt`
we simply typed in
`A:\>MOVE A:\DIRA\file1.txt A:\DIRB\file1.txt`

This raises an interesting question. How did the OS know where MOVE was, in order to execute it? The answer lies in a command DOS normally executes when it first starts up (Windows too, for that matter). PATH

Command Name	PATH
Purpose	Directs the OS to external commands.
Syntax	PATH [external file storage location] [; other]
Common parameters	n/a
Example	`PATH C:\DOS`

While we are at it… originally the system prompt would only show the drive letter and a flashing cursor.

```
A_
```

Now the command prompts includes current drive, directory path, and the > to separate the system prompt from the cursor

```
A:\> _
```

This again is another command normally executed when the system first starts up: PROMPT

Command Name	PROMPT
Purpose	Displays info in the system prompt
Syntax	PROMPT [option 1] … [option n]
Common options	$P Current Drive AND path $G > (greater than sign)
Example	PROMPT pg includes the directory Path and a >

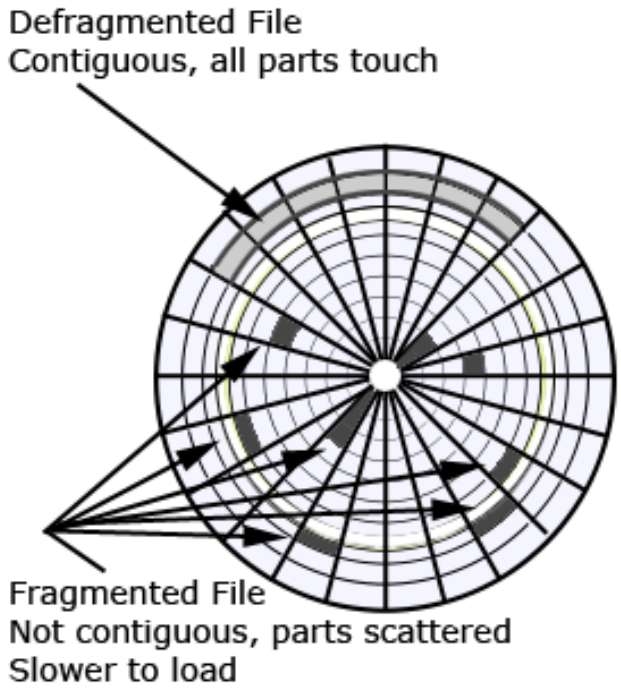
Other PROMPT characters:
$Q =
$$ $
$T time $D date
$N just the current drive
$L <
$_ start a new line

More on MOVE, PATH and PROMPT in Chapter 8, Batch files.

Utilities

While utilities are a different family of System Software than Operating Systems, a few utilities should be covered in this discussion of the Operating System simply because they're included when you install the Operating System. The following utilities address specific situations described earlier in the book.

Earlier in this book, defragmentation was described as parts of files scattered across the hard drive in non-adjacent sectors.



The solution for this situation is the DOS-included defrag command.

DEFRAG used to defragment files into contiguous (adjacent) sectors
    Syntax DEFRAG [drive:]

Another previously mention item is CHKDSK, which technically is a utility. However, a better solution is available now, SCANDISK.

SCANDISK used to check and **repair** a drive
    Syntax SCANDISK [drive:] [autofix]

MSBACKUP © Symantec 1993, shipped with later versions of DOS to back up or restore files, but this was the era of 10 MB hard drives and floppies; Windows or 3rd party versions are much better suited to modern media.

Both of DEFRAG and SCANDISK tools are also available in Windows, along with the Windows backup utility. (Recall ATTRIB? One of the ways a backup utility locates files to archive is by looking for the A attribute).

Redirection

→Redirection with >
If you enter the command DIR, the information is directed to the standard output device… in this case, the monitor.

Redirection simply over-rides the default devices by using the > (not to be confused with the > at the end of the system prompt… think of it as a pointer, as in ***goes to***).

Example: DIR > PRN (think DIR ***goes to*** PRN) would redirect the output to the printer instead of the monitor.

You may also redirect to a file, using DIR > c:\dirA\dir.txt

→Redirection with the pipe: | (may also look like ¦)
A different way of redirecting output is to grab the output and use it as the input of a different command. This is called piping, as you use the pipe [|] or [¦] key… (often the shifted [\]).

About the best example of this is using the ¦MORE.

DIR comes with a /p to pause the screen, but TREE does not.

But if you enter

```
A:\> TREE /f ¦more
```

The output is redirected to a holding spot, and is fed to the monitor a page at a time.

So you could think of this as
TREE /f ***goes to*** 'MORE'
'MORE' ***goes to*** MONITOR, a page at a time.

Chapter 8

Batch Files

Executable files	68
Batch files	68
Autoexec.bat	69
EDIT	69
ECHO	71
PAUSE	71
REM	72
Mov.bat	72
(replaceable parameters)	73

Executable files

If you have access to DOS, type in

```
C:\> DIR C:\DOS\help.*
```

to locate the external command HELP that we used earlier.
If using the Windows command line, type in

```
C:\> DIR C:\windows\system32\help.*
```

instead.

Either way, DIR will locate a program that starts with help, regardless of how it ends. But what you will see is not just HELP, but HELP.EXE

Recall, earlier in the book, there was a discussion of file type extensions, and the list started with .exe, .com, and .bat.

These are the so called executable file types… that is, DOS or the command prompt ***expects*** when you type something in at the system prompt that it will be either an internal or external command, or an application. But for all practical purposes, an external command ***IS*** an application… albeit a small one.

Therefore, the OS is SO expecting a program that it assumes what you type in will end in .exe, .com, or .bat! So, if you type in HELP, the OS will append .exe and give it a try; if no luck, it will append .com and give it a try; if still no luck, it will append .bat and give it a try before it give up and flashes an error message.

EXE and COM files are not normally readable by people… they have been converted into the 1's and 0's that computers like. But batch files are different.

Batch files

A batch file is simply

a series of DOS commands,
listed in a plain text file,
whose extension is bat.

When a batch file is executed, DOS or the COMMAND prompt reads one line of the file at a time, and does what ever it says.

So, if you wanted to tackle the pre-MOVE issue of the last chapter, you could create a batch file named a:\MOV.bat with the following lines:

```
copy a:\dirA\file1.txt a:\dirB\file1.txt
del a:\dirA\file1.txt
```

At the command prompt A:\>, if you typed in MOV, file1 would be copied to dirB, and then deleted… just like the MOVE command.

But it would be ridiculous to create a batch file unless you planned to reuse it. After all, it would take more time to type the batch file than it would to simply type in the two commands. So, batch files should be created when you plan to do the same thing over and over.

Recall in Chapter 7 how it was mentioned that PATH and PROMPT are normally executed when you start your system? Did you wonder how the C:\DOS argument for PATH was delivered, or how the pG argument was delivered for the PROMPT command? If you suspect it was done using a batch file, you are on the right track.

Autoexec.bat

As its name implies, autoexec.bat is automatically executed when a DOS computer is booted. The most common lines in autoexec.bat are:

```
@echo off
PATH C:\DOS
PROMPT $P$G
```

There are often others, but this is the near universal standard.

If you are running a DOS system, run a DIR on the root of C:, you'll probably see it. Run TYPE autoexec.bat to compare your system to the universal.

The next trick, how are batch files created? Well, as they are plain text files, you could use our lab 1 standby COPY with the CON as the source… but this solution gets tricky, as you can't go back for making changes… you'd have to retype the whole thing if you didn't like one item. The key is a small application that is available from the system prompt: EDIT.

EDIT

EDIT is like a small, plain text word processor… the DOS version of Notepad. (If your computer doesn't have edit, you'll use notepad.)

As you may not have a mouse, press the [Alt] key to highlight menu items... just as in DOS help, when you are finished you choose [Alt]+[F]+[X] to open the File menu, and select the eXit command.

```
File  Edit  Search  View  Options  Help
UNTITLED1
F1=Help
```

The advantage of using EDIT is you can change items anywhere by using the arrow keys or your mouse and save your change, rather than having to use COPY CON to retype the entire file.

You can also edit an existing text or batch file. If EDIT is running, choose [Alt]+[F]+[O] to get to the open dialog box.

If EDIT is not running, type in EDIT [path\] filename, such as in

```
C:\> EDIT a:\test.bat     or    C:\> NOTEPAD a:\test.bat
```

to open test.bat

Even better, if test.bat does not exist, EDIT will create it, and then allow you to edit it.

Example

```
C:\> EDIT a:\test.bat     or    C:\> NOTEPAD a:\test.bat
```

to open test.bat Type in PROMPT PT$G to add the time to the prompt.

```
File  Edit  Search  View  Options  Help
TEST.BAT
PROMPT $P$T$G
F1=Help
```

Type [Alt]+[F]+[S] to save the file, then [Alt]+[F]+[X] to close EDIT.

Test the batch file with

```
A:\> TEST
```

to execute test.bat

Your prompt should change to include the time. Manually reset it with

```
A:\ 12:00:00:00> PROMPT $P$G
```

Building better batch files

The reason you probably never saw the PATH and PROMPT being set is in the first line of the most standard autoexec.bat.

```
@echo off
PATH C:\DOS
PROMPT $P$G
```

Command Name	ECHO
Purpose	To control what displays on the monitor
Syntax	ECHO [off] [message] [on]
Common parameters	n/a
Example	@ECHO OFF immediately suppresses display, not even @ECHO OFF will be seen ECHO OFF will suppress the batch file display immediately after the command ECHO off is run. ECHO HELLO will display HELLO, even if the ECHO is off ECHO ON reverses ECHO OFF

Another batch file command is PAUSE.
PAUSE, temporarily halts the batch file execution,
and displays the message Press any key to continue.
Once a key is pressed, execution continues.

I often use PAUSE immediately after a copy command, to give the system a chance to finish before attempting the next line in the batch file, or if I wish to inspect something on the screen during testing of a batch file.

The last batch file command is REM, for remark. This has two purposes... to temporarily skip a line in a batch file, or to leave yourself a note.

Let's say you want the screen to be wiped clear, but not during testing of the batch file. REM CLS leave the line in place, but the batch will skip that command. Take out the REM when finished testing.

Another use is to leave yourself notes… documenting part of the batch file.

```
REM $T is to add time to the prompt
```

is not executed by the computer, but you can read it a year later and understand what the $T in the PROMPT command does.

Basically, any line that begins with REM is not even read by the computer, but is only there for your use.

Command Name	REM
Purpose	To prevent a line in a batch file from executing
Syntax	REM [DOS command]
Common parameters	n/a
Example	REM CLS *tells the batch file to skip* CLS Or REM $T is to add time to the prompt *Can be a note to you to explain PROMPT $T*

Mov.bat

Let's get back to that early attempt to create something that works like the MOVE command.

```
copy a:\dirA\file1.txt a:\dirB\file1.txt
del a:\dirA\file1.txt
```

The problem was, you specified the file names, making it a one time only batch file.

So, how does COPY or REN work? These commands don't know what you are going to use for source or for destination. There must be something built into these commands that basically is a place holder for source, and something that is a placeholder for destination. These place holders, or parameters, must be replaced at the time of execution… hence the name: replaceable parameters.

Replaceable parameters
"The user will provide a filename when the command is used".

Actually, this makes sense. You can't just type MOVE, hit the enter key, and expect it to work.

If you just type MOVE and hit the enter key, DOS tells you it was expecting a filename (source).

If you just type MOVE filename and hit the enter key, DOS tells you it was expecting a filename (destination).

What is used in DOS is:
%1 to act as a place holder for the first filename to be entered (source) and
%2 to act as a place holder for the first filename to be entered (destination).

So, we change our batch file from this

```
copy a:\dirA\file1.txt a:\dirB\file1.txt
del a:\dirA\file1.txt
```

to the following:

```
COPY %1 %2
DEL %1
```

This batch file will now work for any file names.

Here's how it works
If you have a file named a:\folder1\junk.txt, and you want to move it to a:\folder2, you could use the MOV.BAT command

Syntax for MOV
(recall, executable files don't require you to include the extension)
MOV source destination

Example of MOV
(recall, executable files don't require you to include the extension)
MOV a:\folder1\junk.txt a:\folder2\junk.txt

What happens
1) DOS grabs the first filename given (the source),
a:\folder1\junk.txt,
and replaces every %1 with
a:\folder1\junk.txt

2) DOS grabs the second filename given (the destination),
a:\folder2\junk.txt,
and replaces every %2 with
a:\folder2\junk.txt

3) DOS executes each line in the batch file, using the replaced filenames

COPY **%1** **%2** turns into **COPY** **a:\folder1\junk.txt** **a:\folder2\junk.txt**

DEL **%1** turns into **DEL** **a:\folder1\junk.txt**

The result is the original file is copied, then that original file is deleted, leaving only the copy... net effect, moving the file.

Note: if you improperly state the destination, the first line of the batch file will not be able to make the copy, but the second line will still delete the original... leaving you with no file at all.

Chapter 9

Lab

Lab 3	76
Lab 3 questions	77

Make a folder named Lab3 on your floppy or O: drive.

At the end of this process, you will be turning in answers using the Quizzes area of Desire2Learn, based on the following questions. So, to get the best score, fill in all the answers in this document first.

RD
What does the command do?
What is the syntax for the command?
(Syntax does not include specific drives or directories)
Are there any common parameters mentioned in class? List them
Write out one example of how the command was entered in this lab

DELTREE
What does the command do?
What is the syntax for the command?
(Syntax does not include specific drives or directories)
Are there any common parameters mentioned in class? List them
Write out one example of how the command was entered in this lab

PROMPT
What does the command do?
What is the syntax for the command?
(Syntax does not include specific drives or directories)
Are there any common parameters mentioned in class? List them
Write out one example of how the command was entered in this lab

PATH (Usually in autoexec.bat)
What does the command do?
What is the syntax for the command?
(Syntax does not include specific drives or directories)
Are there any common parameters mentioned in class? List them
Write out one example of how the command was entered in autoexec.bat

ECHO
What does this BATCH file command do?
@ECHO OFF What is displayed?
ECHO OFF What is displayed after the line is processed?
ECHO ON What is displayed after the line is processed?
ECHO Message What is displayed?

PAUSE
What does this BATCH file command do?
What is displayed?

REM
What does this BATCH file command do?
What is displayed?

DOS Activity
(Examples use A:\, modify accordingly if not using A:\)

Using EDIT, create a batch file setpromt.bat that does the following
(edit a:\setpromt.bat or notepad a:\setprompt.bat)
Turns the echo off immediately
Sets the prompt to include the time,
  it should still start with the current path and end with the greater than sign
Displays the text message, "Do you like my prompt?"
waits for you to continue
Clears the screen
Makes a directory called lab3
Copy your batch file to setpromt.bak Optional have batch create a backup directory, & copy there
Optional, have your batch file run tree a:\ /f to let you see what happened
IF YOU HAVE CODED THIS PROPERLY, YOU WILL BE ABLE TO SELECT THE BEST BATCH FILE CHOICE IN THE QUIZ

Using EDIT, create another batch file named mymove.bat in a:\lab3
that does the following
(edit a:\lab3\mymove.bat)

- Makes a copy of any file you specify (%1)
- Copies that to any filename or location you specify (%2)
- Deletes the original file (%1)

see page 73
IF YOU HAVE CODED THIS PROPERLY, YOU WILL BE ABLE TO SELECT THE BEST BATCH FILE CHOICE IN THE QUIZ

I suggest you test your batch file
Example of a test: setpromt.bat is moved from lab3 to the root using

```
a:\lab3\>mymove a:\lab3\setpromt.bat a:\setpromt.bat
or a:\lab3\>mymove a:\lab3\setpromt.bat a:\lab3bksetpromt.bat
```

You will then transfer your answers to the D2L Quiz for lab 3.
No dropbox for this lab… ALL the points come from the quiz.

(You will match your batch file to one of several versions in the quiz)

Choose the Quizzes menu, and locate Lab 3. Video like lab 3 on class web site

Question 1 (2 points)
The RD command
❍ is used only in BATCH files
❍ tells the computer where to locate the external DOS command directory
❍ deletes a directory, and its contents
❍ changes the tools displayed in the DOS prompt
❍ deletes a single, empty directory

Question 2 (2 points)
The DELTREE command:
❍ deletes a single, empty directory
❍ deletes a directory, and its contents
❍ is used only in BATCH files
❍ changes the tools displayed in the DOS prompt
❍ tells the computer where to locate the external DOS command directory

Question 3 (2 points)
Syntax: RD or DELTREE
❍ rd [path\]DOS or deltree [path\]DOS ❍ rd $[tool] or deltree $[tool]
❍ rd [source] [destination] or deltree [source] [destination]
❍ rd [path\]directoryname or deltree [path\]directoryname

Question 4 (1 point)
The command to delete an empty directory named junk is
❍ format junk ❍ rd junk
❍ deltree junk ❍ del junk

Question 5 (1 point)
The command to delete a directory named junk AND its contents is
❍ deltree junk ❍ del junk
❍ format junk ❍ rd junk

Question 6 (2 points)
The command PROMPT
❍ deletes a directory, and its contents
❍ tells the computer where to locate the external DOS command directory
❍ changes the tools displayed in the DOS prompt
❍ is used only in BATCH files
❍ deletes a single, empty directory

Question 7 (2 points)
Syntax: PROMPT
❍ prompt [path\]directoryname ❍ prompt [path]\DOS
❍ prompt [source] [destination] ❍ prompt $[tool]

Question 8 (2 points)
The following is the correct use of attribute command: attrib pg
❍ True ❍ False

Question 9 (2 points)
The PATH command
❍ is used only in BATCH files ❍ deletes a directory, and its contents
❍ deletes a single, empty directory ❍ changes the tools displayed in the DOS prompt
❍ tells the computer where to locate the external DOS command directory

Question 10 (2 points)
Syntax: PATH

❍ PATH [path\]random-directory ❍ PATH [source] [destination]
❍ PATH $[tool] ❍ PATH [path\]DOS-directory

Question 11 (2 points)
To set the path to DOS on most computers
❍ path c:\command.com ❍ path c:\DOS
❍ prompt c:\DOS ❍ path %1

Question 12 (10 points) Match the batch file command with what it does

	suppresses display after displaying echo off	1. @echo off
	lets you leave a remark that is not executed	2. echo off
	immediately suppresses display	3. echo [message]
	displays your message	4. pause
	interrupts the execution of the batch file	5. rem

Question 13 (20 points) Choose the correct code that makes mymove.bat

copy %1 %2 del %1	del %1 copy %1 %2	move %1 %2 del %1	copy $1 $2 del $1

Question 14 (20 points) Which accomplishes the goals of setpromt.bat?

@echo off prompt pt$g echo Like prompt? cls md lab3 copy setpromt.bat setpromt.bak	@echo off prompt pt$g echo Like prompt? cls rd lab3 copy setpromt.bat setpromt.bak
@echo off prompt %p%t%g echo Like prompt? cls md lab3 copy setpromt.bat setpromt.bak	@echo off prompt pt$g echo Like prompt? cls md lab3 move setpromt.bat setpromt.bak

Question 15 (6 points)
The actual command is tree.com, not tree. So, if presented an instruction DOS automatically tests to see if the instruction ends with what three extensions? (leave out the period) _____ _____ _____

Question 16 (6 points)
What is a batch file? A series of ____ commands executed in order, from a plain _____ document whose extension is _____

Question 17 (6 points)
Where is the prompt and path typically set?
❍command.com ❍ A dumpster in Brazil
❍ config.sys ❍ autoexec.bat

Question 18 (6 points)
What is the prompt statement to show the standard prompt that includes path and greater than sign? Example C:\>
❍ prompt %p%g ❍ path pg
❍ prompt pt$g ❍ prompt pg

Question 19 (6 points)
What is the prompt statement to show the prompt that includes path, the time, and greater than sign?
❍ path pg ❍ prompt %p%g ❍ prompt pg ❍ prompt pt$g

Transfer these answers to the D2L quiz for lab 3

Chapter 10

Review

DOS Review

For tests I am trying to see how much of all the material you are retaining, so the test by design is timed, about 35 minutes, and covers a very broad area with fairly deep coverage; I consider it both an assessment and a learning opportunity… trying to let us both see what you have mastered, and what you may need a little more work on…

So, as with the SAT, you are not expected to make a perfect grade on my tests;
here is how I curve the test:
○ questions missed by the majority don't count against you,
○ the highest score set the curve,
Example: if the highest grade is a 93, everyone gets 7 points

Notes:
The computer doesn't grade the test, I do; I will give credit where appropriate when I grade

Your lowest test grade can be improved by doing well on that section of the final.
Example, if your lowest test grade is a 60, but you make an 80 on that part of the Final exam, I will add 60% of the improvement to that test… meaning it is not a 60 anymore, but 72.

Directories and Files	
Path	[drive:\][directory\]filename.ext (8.3)
Directory (Folder) Commands	
List contents	`dir`
Show structure	`tree /f`
Change directory focus (open another folder)	`cd foldername`
Make a new directory	`md foldername or mkdir foldername`
Remove an empty directory	`rd foldername or rmdir foldername`
Delete directories, subdirectories, and files	`deltree foldername`
File Commands	
Copy files	`copy source destination`
»Example	`copy a:\test.txt a:\class\test.txt`
»Example, destination renamed	`copy a:\test.txt a:\class\test.ltr`
»Example, copy all .txt files, regardless of file name	`copy a:\*.txt a:\class\*.txt`
»Example, copy all files named *homework*, regardless of extension	`copy a:\homework.* a:\class\homework.*`
»Example, copy all files	`copy a:\*.* a:\class\*.*`
»Example, create a text file	`copy con filename.ext`
Copy files and folders	`xcopy source destination /e`
»Example, copy all files, folders, and subdirectories	`xcopy a:\*.* c:\backup\*.* /e`
Copy a diskette	`diskcopy source destination`
»Example	`diskcopy a: a:`
Move a file to a different folder	`move source destination`
Delete a file	`del filename.ext`
Undelete a file	`undelete filename.ext`
Set an attribute of a file	`attrib action (±) target file`
»Example	`attrib +h a:\class\test.txt`
Rename a file	`ren oldname (no path)newname`
View or change attributes	`attrib [+][-]|[r][a][s][h]target`
Batch Files	
set the Prompt	`prompt $p$g`
set the Path to DOS	`path c:\DOS`
insert a comment	`rem note to me`
wait to continue	pause
display	`@echo off nothing displays`
display	`echo off nothing displays after this`
display	`echo on next actions will display`
display	`echo this displays "this displays"`

Utilities	
arrange files to a faster loading, contagious state	`defrag`
check for and repair, errors	`chkdsk or scandisk`
copies information for safekeeping	`backup`

Glossary

BIOS

Bit

Boot

Byte

CPU

Disk

Giga...

Hardware

Hertz

Input

Kilo...

Mega...

Monitor

Output

Peripheral

RAM

ROM

Software

A list of all the DOS commands and usage is in the DOS Appendix

Appendix A

DOS COMMAND REFERENCE

ATTRIB	-2-
CD	-2-
CHKDSK	-2-
CLS	-2-
COPY	-2-
DATE	-2-
DEFRAG	-3-
DEL	-3-
*DELTREE	-3-
DISKCOPY	-3-
DIR	-3-
ECHO	-3-
FORMAT	-3-
LABEL	-4-
MD	-4-
MOVE	-4-
PATH	-4-
PAUSE	-4-
PROMPT	-4-
RD	-4-
REM	-5-
REN or RENAME	-5-
TIME	-5-
TREE	-5-
TYPE	-5-
*UNDELETE	-5-
VER	-5-
VOL	-6-
XCOPY	-6-

ATTRIB

Command Name	ATTRIB
Purpose	used to display or change attributes
Syntax	ATTRIB [±R or A or S or H] [path\][filename]
Common parameters	n/a
Example	`ATTRIB +R file1.txt`

CD

Command Name	CD
Purpose	To change directories
Syntax	CD [drive:\][directory-name\] *directory name*
Common parameters	n/a
Example	`CD TEST` `CD ..` to change out of the current directory `CD \` to change to the root directory

CHKDSK, used to display a status report of a disk, or
Fix errors in the File Allocation table
Syntax CHKDSK [drive:] [/f]
Replaced by SCANDISK

CLS

Command Name	CLS
Purpose	Clears the screen
Syntax	CLS
Common parameters	n/a
Example	`CLS`

COPY

Command Name	COPY
Purpose	Duplicates files
Syntax	COPY *[path\]source [path]destination*
Common parameters	n/a
Example	`COPY file1.txt file2.txt`

DATE, used to display or change the system calendar
Syntax DATE [mm-dd-yy]

DEL

Command Name	DEL
Purpose	To delete a text file
Syntax	DEL [drive:\][directory-name\] *file-name*
Common parameters	n/a
Example	`DEL TEST.TXT`

DEFRAG used to defragment files into contiguous (adjacent) sectors
Syntax DEFRAG [drive:]

*DELTREE, *not available in Windows*, used to remove non-empty directories
DANGEROUS, use with caution
Syntax DELTREE [path]

DISKCOPY

Command Name	DISKCOPY
Purpose	Duplicates floppy disks
Syntax	DISKCOPY *source destination*
Common parameters	n/a
Example	`DISKCOPY A: A:`

DIR

Command Name	DIR
Purpose	Generates a directory listing
Syntax	DIR [drive:\][directory-name\]
Common parameters	/w (wide directory listing) /p (directory listing pauses after each full screen)
Example	`DIR` or `DIR /w` or `DIR /p`

ECHO

Command Name	ECHO
Purpose	To control what displays on the monitor
Syntax	ECHO [off] [message] [on]
Common parameters	n/a

Example	`@ECHO OFF immediately suppresses display, not even @ECHO OFF will be seen`
	
	`ECHO OFF will suppress the batch file display immediately after the command ECHO off is run.`
	
	`ECHO HELLO will display HELLO, even if the ECHO is off`
	
	`ECHO ON reverses ECHO OFF`

FORMAT

Command Name	FORMAT
Purpose	To prepare a disk for use
Syntax	FORMAT *drive*:
Common parameters	/q (quick format on a previously formatted disk)
Example	`FORMAT A:` or `FORMAT A: /q` Note: You have the option to label the disk

LABEL, used to change a disk's volume label

Syntax LABEL [drive:] [*new volume label*]

MD

Command Name	MD or MKDIR
Purpose	Makes a directory
Syntax	MD [*drive*:\][*directory-name*] directory-name
Common parameters	n/a
Example	`MD TEST`

MOVE

Command Name	MOVE
Purpose	Move a file to a different directory
Syntax	MOVE [*path*\]*directory-name* [*path*\]directory-name
Common parameters	n/a
Example	`MOVE dira\file1.txt dirb\file1.txt`

PATH

Command Name	PATH
Purpose	Directs the OS to external commands.
Syntax	PATH [external file storage location]
Common parameters	n/a
Example	`PATH C:\DOS`

PAUSE, temporarily halts a batch file execution,

and displays the message Press any key to continue

Once a key is pressed, execution resumes

PROMPT

Command Name	PROMPT
Purpose	Displays info in the system prompt
Syntax	PROMPT [option 1] … [option n]
Common options	$P Current Drive AND path $G > (greater than sign)
Example	`PROMPT $p$g` `includes the directory Path and a >`

RD

Command Name	RD or RMDIR
Purpose	To remove an empty directory
Syntax	RD [drive:\][directory-name\] *directory name*
Common parameters	n/a
Example	`RD TEST`

REM

Command Name	REM
Purpose	To prevent a line in a batch file from executing
Syntax	REM [DOS command]
Common parameters	n/a
Example	REM CLS *tells the batch file to skip* CLS Or REM $T is to add time to the prompt *Can be a note to you to explain PROMPT $T*

REN or RENAME

Command Name	REN or RENAME
Purpose	Renames files
Syntax	REN *[path\]source destination*
Common parameters	n/a
Example	REN file1.txt file2.txt

TIME, used to display or change the system clock
Syntax TIME [hh:[mm:[ss]]]

SCANDISK, used to check and repair a drive
Syntax SCANDISK [drive:] [autofix]

TREE

Command Name	TREE
Purpose	To display a tree structure
Syntax	TREE [drive:\][directory-name\]
Common parameters	/f (will display directories and directory contents)
Example	Tree /F or tree a:\ /f

TYPE

Command Name	TYPE
Purpose	To display the contents of a text file
Syntax	TYPE [drive:\][directory-name\] *file-name*
Common parameters	n/a
Example	TYPE TEST.TXT

* UNDELETE, *not available in Windows* (replaced by the recycle bin)
May be able to recover recently deleted files

VER, used to display the DOS or Windows version
Syntax VER

VOL, used to display the volume label
(labels are normally set with FORMAT, and changed with LABEL)
Syntax VOL [drive:]

XCOPY

Command Name	XCOPY
Purpose	Duplicates files and directories
Syntax	XCOPY *[path\]source [path]destination*
Common parameters	/e
Example	`XCOPY dir1\*.* dir2\*.* /e`