
UNIT 6 OPERATING SYSTEMS (DOS, WINDOWS, UNIX AND LINUX)

Structure

- 6.0 Objectives
- 6.1 Introduction
- 6.2 MS-DOS Environment
 - 6.2.1 MS-DOS Memory Types
 - 6.2.2 MS-DOS Version
 - 6.2.3 Loading MS-DOS
 - 6.2.4 MS-DOS Device Drivers
 - 6.2.5 Config.sys File and its Use
 - 6.2.6 Directory Structure of DOS
 - 6.2.7 MKDIR Command (Make Directory)
 - 6.2.8 Other MS-DOS Command
 - 6.2.9 MS-DOS Batch Files
 - 6.2.10 File and Hard Disk
- 6.3 MS-Windows
 - 6.3.1 Advantage of Using Windows vs DOS
 - 6.3.2 Versions of MS Windows
 - 6.3.3 Basic Features of Windows
 - 6.3.4 Support for MS-DOS
 - 6.3.5 Dial up Network and Internet Explorer
 - 6.3.6 The Internet Explorer Web Browser
 - 6.3.7 Customizing Windows Operating Systems
 - 6.3.8 Start Menu
 - 6.3.9 Windows Help
 - 6.3.10 Organisation of information
 - 6.3.11 Using Window Explorer
 - 6.3.12 Using Notepad
- 6.4 Unix
 - 6.4.1 Main Features
 - 6.4.2 Unix Operating System
 - 6.4.3 Utilities File Management
 - 6.4.4 Process Management
 - 6.4.5 Starting and Shutting Down UNIX/ LINUX
 - 6.4.6 UNIX Command Syntax
 - 6.4.7 UNIX File System
 - 6.4.8 Mountable File System
 - 6.4.9 Shell Scripts
 - 6.4.10 User Management
 - 6.4.11 Other System Activity
- 6.5 Linux
 - 6.5.1 Technical Features of Linux
 - 6.5.2 Linux File System
 - 6.5.3 Components of a Linux System
 - 6.5.4 Linux Kernel and its Modules
 - 6.5.5 Drawbacks of Using Linux
- 6.6 Summary
- 6.7 Answers to Self Check Exercises
- 6.8 Keywords
- 6.9 References and Further Reading

6.0 OBJECTIVES

After reading this Unit, you will be able to:

- 1 Understand the basic features of MS-DOS, Window, Unix and Linux operating systems; and
- 1 Work with various components of each types.

6.1 INTRODUCTION

Operating systems are software that controls the execution of a computer programme that may provide scheduling, debugging, input/output control, accounting, compilation, storage assignment, data management and related services. In the earlier unit you have been familiarised with the basic concepts of operating systems. In this unit you will get a detailed knowledge of the commonly used operating system viz. MS-DOS, MS-Windows, Unix and Linux. The commands and functions of each system has been dealt with in details to initiate you in using these operating systems.

MS-DOS ENVIRONMENT

DOS is an acronym for Disk Operating System. DOS is manufactured by Microsoft, and is used on PC-compatible computers to interact with the various parts of your computer's hardware and provide a somewhat user-friendly interface to them. Typing in commands are the means to do so. Tasks, which can be performed with these commands, include: files and directory management, disk maintenance, hardware configuration, memory optimization, and DOS customization. This system program must always be present when working with your computer.

Why You Need DOS

There are a variety of reasons why you need DOS. A few of them are listed below to satisfy your curiosity.

- 1) DOS controls the flow of information between you and the computer (translator).
- 2) DOS allows you to store information on your computer.
- 3) DOS allows you to retrieve information stored on your computer.
- 4) DOS interprets and translates the software you have on your computer.
- 5) DOS gives you access to all its function (i.e. saving, copying, and printing files).

6.2.1 MS-DOS Memory Types

There are two basic types of Memory RAM and ROM. RAM which, stands for random access memory, can be read from and written to. You can write to it again and again; new data replaces older data with no problems. However, RAM is volatile; it loses its data when you shut off the power.

ROM, which stands for read-only memory, has data stored in it by its manufacturer and can be read from repeatedly, but not written to. A ROM chip usually contains an essential program such as the Basic Input-Output Operating System (BIOS), which provides the elementary instructions to read and write data for your hardware. ROM is not volatile; it retains its memory when the power is off so that its information is available during the early parts of booting, before any programs have been loaded into RAM.

640K Restriction

The designers of the IBM PC anticipated that a significant amount of the software for the PC would reside in Read Only Memory, or ROM. For example the PC's BIOS, or Basic Input/Output System, and parts of the BASIC interpreter are both contained in the PC's ROM. As various manufacturers developed expansion hardware for the PC, IBM expected that the low-level software functions required to access this hardware would be supplied on ROM chips located on these boards. For example, the BIOS functions for EGA and VGA video boards, and for many hard disk adapters, reside in one or more ROM chips located on these adapter boards. To accommodate all the potential types of add-on boards that might put into a given PC, 384K of the PC's one-megabyte address space was reserved for ROM and for display RAM on the various types of video adapters. Fig. 1 shows a map of the one-megabyte address space in PC and PC/XT- class systems.

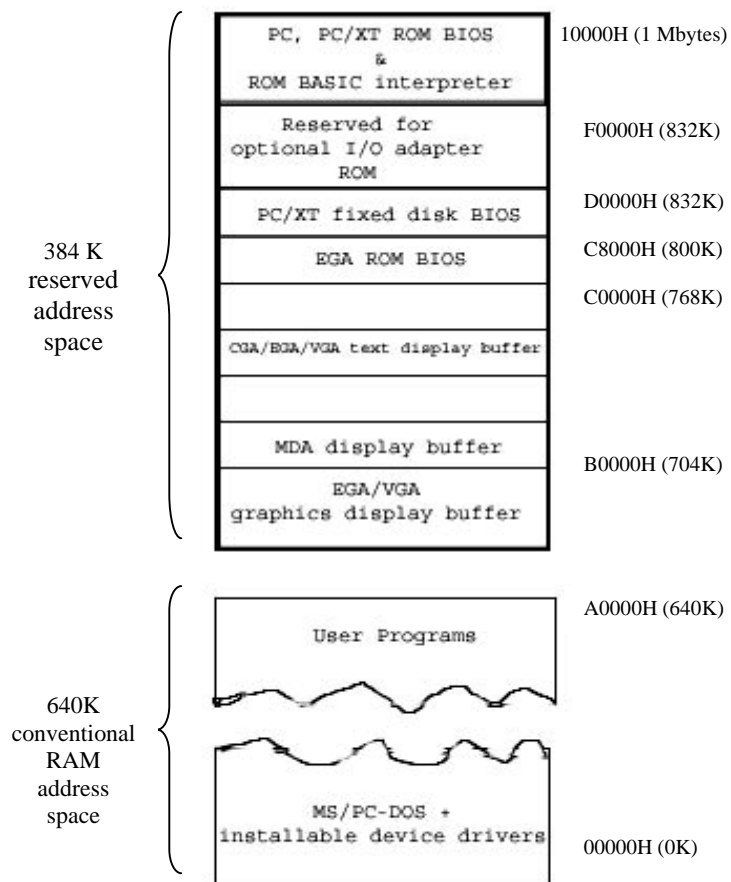


Fig.1: Address Mapping in DOS

Conventional Memory

Also known as base memory, conventional memory is the classic 640K of address space allocated to RAM for DOS and your applications.

Exactly where each element begins in memory depends on numerous factors, but this will give you a notion of the demands on memory

- 1 The lowest part of memory is a data area, a scratch pad for your PC's ROM BIOS, which includes such things as a record of your current machine configuration (about 2K).

- 1 DOS's own internal routines and data areas appear next (about 60K in DOS 5.0 when not loaded high).
- 1 The next block of memory contains elements such as file handles (created when DOS executes the FILES line in your CONFIG.SYS file) and disk buffers (the standard cache memory for faster disk I/O (input/output) that you size by using the BUFFERS line in CONFIG.SYS). You may choose to instruct DOS 5 to load the buffers in high memory instead, by using the DOS=HIGH line.
- 1 Next is the software device drivers that you designate in CONFIG.SYS, such as ANSI console drivers, RAM disks, and disk caches (demands for memory vary greatly).
- 1 The resident portion of COMMAND.COM, the standard DOS program that interprets what you type at the DOS command line, comes next (about 5K with DOS 5.0).
- 1 Next come any resident programs, or TSRs (usually activated when you press a hot key) that you have loaded using AUTOEXEC.BAT. There are all sorts of TSRs on the market; examples are Borland's SideKick program and various screen capture programs, like HotShot (demands for memory vary greatly).
- 1 The example supposes that you are using some sort of menu-driven shell program designed to simplify life with DOS, such as the Norton Commander. The shell is loaded above the TSRs, starts running, and probably remains in memory while you use it to summon one or another application.
- 1 When you "point and shoot" to start an application from the shell, the application has whatever is left of conventional memory to work in.

When you think of how demands on conventional memory are proliferating, you can see what a help it would be to move some of these memory hogs somewhere else than conventional memory', such as into reserved memory above the 640K limit

Upper Memory Area

The Upper Memory Area (UMA) is the part of the memory map extending from the top of conventional memory up to the 1MB DOS limit—384K all told. Just to make matters confusing, it is sometimes called reserved memory because it is normally reserved for BIOS for devices like video cards, disk controllers, and the PC's ROM itself.

Expanded Memory

An early solution to the memory crunch was called expanded memory. This memory is an extra set of RAM chips located on a separated board and treated just like any other hardware device. Expanded memory is accessed through upper memory. You can access only a portion of it at a time, but you can switch from portion to portion to take advantage of all the available space. A device driver called an expanded memory manager controls access to expanded memory and does the switching. Expanded memory is relatively slow compared to conventional or upper memory, but is still considered faster than a disk drive.

Even if your system has expanded memory, you may not be using it. A program

must be specially designed to take advantage of expanded memory. If you don't have any such programs, your expanded memory is lying fallow.

Extended Memory

Just as expanded memory was gaining a foothold, Intel developed another solution to the 640KB ceiling. The 286 chips, in addition to being a much better and faster processor, could directly access 16MB of memory. Memory above the 1MB mark, which is allocation unit RAM, is called extended memory. The 386 and 486 have pushed the limit even further, allowing several gigabytes of memory. Theoretically, you could load and run programs throughout this memory space. Sadly, DOS must adhere to its original design and has no way to directly access the extra space. This is why you can run out of memory even though you have many megabytes of unused RAM.

Still, your extended memory doesn't have to go waste. DOS can treat extended memory as separate device and access it through upper memory, using an extended memory manager to handle the details. As with expanded memory, DOS can't load and run programs in extended memory, with an exception - the high memory area.

High Memory Area

Because of a quirk in the way DOS addresses memory, it turns out that it can directly access not just 1MB, but nearly 64KB beyond that. When a PC has extended memory, a small portion (almost 64KB) is directly addressable by DOS, permitting you to run programs there. DOS's extended memory manager, named HIMEM.SYS, turns that part of extended memory into the high memory area (HMA), where it can load and run one program at a time. Normally, you load DOS into the HMA, freeing up 45KB or more of conventional memory for other programs.

6.2.2 MS-DOS Versions

The history of MS-DOS is surprisingly long. It started off as QDOS (Quick and Dirty Operating System), which was developed by Seattle Computer Products to run on IBM's new PC. This list is fairly comprehensive although a number of the more obscure versions of DOS have been omitted.

Version	Date	Comments
1.0	1981	The original version of MS-DOS. This was a renamed version of QDOS, which had been purchased by an upstart company called Microsoft.
1.25	1982	This added support for double-sided disks. Previously the disk had to be turned over to use the other side
2.0	1983	This added support for IBM's 10 MB hard disk, directories and double-density 5.25" floppy disks with capacities of 360 KB
2.11	1983	Support for foreign and extended characters was added.

3.0	1984	Support for high-density (1.2 MB) floppy disks and 32 MB hard disk was added.
3.1	1984	Network support was added.
3.3	1987	This release was written to take advantage of IBM's PS/2 computer range. It added support for high-density 3.5" floppy disks, more than one partition on hard disks (allowing use of disks bigger than 32 MB) and code pages.
4.0	1988	This version provided XMS support, support for partitions on hard disks up to 2 GB and a graphical shell. It also contained a large number of bugs and many programs refused to run on it.
4.01	1989	The bugs in version 4.0 were fixed.
5.0	1991	This was a major upgrade. It allowed parts of DOS to load itself in the high memory area and certain device drivers and TSRs to run in the unused parts of the upper memory area between 640K and 1024K. This version also added support for IBM's new 2.88 MB floppy disks. An improved BASIC interpreter and text editor were included, as was a disk cache, an undelete utility and a hard-disk partition-table backup program. After the problems with MS-DOS 4, it also provided a utility to make programs think they were running on a different version of MS-DOS.
5.0a	1992/3	This was a minor bug fix which dealt with possibly catastrophic problems with UNDELETE and CHKDSK.
6.0	1993	This was a catch-up with Novell's DR-DOS 6. It added a disk-compression utility called DoubleSpace, a basic anti-virus program and a disk defragmenter. It also finally included a MOVE command, an improved backup program, MSBACKUP and multiple boot configurations. Memory management was also improved by the addition of MEMMAKER. A number of older utilities, such as JOIN and RECOVER were removed. The DOS Shell was released separately as Microsoft felt that there were too many disks.
6.2	1993	Extra security was built into DoubleSpace following complaints of data loss. A new disk checker, SCANDISK, was also introduced, as well as improvements to DISKCOPY and SmartDrive.
6.21	1993	Following legal action by Stac Electronics, Microsoft released this version which had DoubleSpace removed. It came with a voucher for an alternative disk compression program.

6.22	1994	Microsoft licensed a disk-compression package called DoubleDisk from VertiSoft Systems and renamed it DriveSpace, which was included in this version.
7.0	1995	This version is part of the original version of Windows 95. It provides support for long filenames when Windows is running, but removes a large number of utilities, some of which are on the Windows 95 CD in the \other\oldmsdos directory.
7.1	1997	This version is part of OEM Service Release 2 and later of Windows 95. The main change is support for FAT 32 hard disks, a more efficient and robust way of storing data on large drives.

6.2.3 Loading MS-DOS

Use a brand new floppy disk or one that contains no essential data after formatting it by using the command:

FORMAT A: /S

Formatting can install the DOS core program files (IO.SYS, MSDOS.SYS and COMMAND.COM) so that you can boot from this floppy.

Copy the FDISK, FORMAT, and SYS programs to the disk using the COPY command.

Insert the diskette into the floppy drive of the machine in which the DOS has to be installed and then restart the machine and in the SETUP menu select BOOT SEQUENCE and then change the boot sequence from the A, C, CDROM; save and exit the setup. The DOS will be installed once the system boots up.

The Program (The MS-DOS Boot Process)

The term Booting comes from the old expression “**Lifting your self up from the boot straps.**” Hence the boot sector of a hard disk or floppy is known as the “**Boot Strap.**” The term formed as computer enthusiasts noticed that computers with internal hard disks seemed to turn themselves on just like one lifting them selves up by the bootstraps. Sort of developing “something from nothing.”

ROM, which stands for “**READ ONLY MEMORY**” is an IC chip on the main board of a computer, and is what begins the boot process. This ROM is also referred to as the **BIOS (BASIC INPUT OUTPUT SYSTEM.)**. One cannot change the contents of a ROM chip as it is read only. Its content is a software program to boot the system and cannot be changed. Below is the sequence of events that occur as a computer boots up.

1) **A POST test is initiated. (Power On Self-Test).**

The POSTtest will first attempt to:

- a) Initialize the keyboard and other system hardware such as the video card.
- b) Test the computer’s memory RAM (Random Access Memory.)

- c) If memory tested good, then check disk drives otherwise the system will halt.
 - d) Next, the POST will attempt to find a valid operating system. If a bootable disk is in the A: drive, the system will attempt to boot off such bootable disk. If not, the system will look for a valid operating system on the C: drive.
- 2) **If all goes well up to this point, the BIOS will find and recognize the File named**

IO.SYS in the C:\...(ROOT Directory) as a valid operating system, and load it into RAM. IO.SYS stands for INPUT OUTPUT SYSTEM. It provides services for peripherals such as printers, modems, disk drives, etc. Typically any device attached to the CPU is a peripheral. CPU stands for “Central Processing Unit” or Microprocessor. And is the main brain of the computer.

- 3) **Next IO.SYS will Load, if found, a file called MSDOS.SYS.** This program file is an extension of the system ROM or BIOS. It’s the gateway to which DOS can communicate to the computer’s Basic Input Output System.
- 4) **Next MSDOS.SYS will look for, and if found, load into memory, a file called CONFIG.SYS.** This file is optional and is used to tailor your system to your specific needs. Optional, meaning it is not necessary to exist for the computer to boot. At this point, config.sys is processed and executed.
- 5) **Next MSDOS.SYS will load a program file into memory, if found called COMMAND.COM.** Command.com contains all the internal DOS instructions such as DIR, CD, CLS and many more. It is a command Interpreter. Anything typed on the keyboard is first processed by command.com after the enter key is struck.
- 6) **Finally, Command.com will look for, and if found will load another optional file called AUTOEXEC.BAT.** Once autoexec.bat is loaded, it is processed and executed. (If neither config.sys nor autoexec.bat exist, the system’s boot process will be complete after COMMAND.COM is loaded.)

6.2.4 MS-DOS Device Drivers

A file that contains information used to control hardware devices such as memory, a monitor, a keyboard, a printer, and so on. Some device drivers create and control logical devices such as RAM drives.

Device Drivers	Description
ANSI.SYS	Provides function that control screen colors, cursor position and key assignments.
CHKSTATE.SYS	Used by MEMMAKER to control the optimization process.
DBLSPACE.SYS	Moves DBLSPACE.BIN to an appropriate memory location
DISPLAY.SYS	Supports code-page switching for your screen and keyboards.

DRIVER.SYS	Creates a logical drive assigned to a physical floppy drive.
DRVSPACE.SYS	Moves DRVSPACE.BIN to an appropriate memory location.
EGA.SYS	Saves and restores the display when the DOS shell's task swapper is used with EGA monitors
EMM386.EXE	Provides access to the upper memory area, and uses extended memory to simulate expanded memory.
HIMEM.SYS	Manages the extended memory, including the high memory area (HMA)
INTERLINK.EXE	Links a client computer to a server computer and redirects drive and printer port names.
POWER.EXE	Reduces power consumption in a laptop computer when applications and devices are ideal.
RAMDRIVE.SYS	Uses an area of your computer's random-access-memory (RAM) to simulate a hard disk drive
SETVER.SYS	Loads the DOS version table into memory
SIZER.SYS	Used by MEMAKER to determine the size in memory of device drivers and TSRs
SMARTDRV.EXE	Provides compatibility for hard-disk controllers that can't work with memory provided by EMM386.EXE or Windows running in 386 enhanced mode.

6.2.5 Config.sys File and its Use

A file that contains commands necessary to configure the system and load drivers, CONFIG.SYS is executed immediately after the operating system files are loading during booting.

CONFIG.SYS is an ASCII text file, somewhat like a batch file, that contains special configuration commands. Most of these commands are unique to CONFIG.SYS; you can't use them anywhere else. Nor can you use ordinary commands, such as DIR or FORMAT, in CONFIG.SYS.

Like a batch file, you can create or modify CONFIG.SYS using an ASCII editor. Unlike a batch file, you can't run CONFIG.SYS, it is used only during booting.

The following is a sample CONFIG.SYS file:

```
DEVICE=C:\DOS\SETVER.EXE
```

```
DEVICE=C:\DOS\HIMEM.SYS
```

DEVICE=C:\DOS\ANSI.SYS

DOS= HIGH

BUFFERS=50

FILES=30

SHELL=C:\DOS\COMMAND.COM C: \DOS /P

STACKS = 9,256

This example includes six different types of commands: DEVICE, DOS, BUFEERS, FILES, SHELL, and STACKS.

These are allocation unit commands that you can use only in CONFIG.SYS. Notice their general syntax. In each case, an equal sign (=) follows the command name. To the right of the equal sign are filespecs, parameters, and / or switches.

Autoexec.bat file and its uses

AUTOEXE.BAT is a batch file that is executed when the system boots, immediately after CONFIG.SYS is processed.

Almost every system has an AUTOEXEC. BAT file. When DOS boots or reboots, it looks for this file in the root directory of the boot drive. If AUTOEXEC.BAT is there, it turns immediately after CONFIG.SYS, before you get a chance to do anything from the command prompt.

Most people who use DOSKEY start it with a command in AUTOEXEC.BAT. A very basic AUTOEXEC.BAT file might look like this:

```
@ ECHO OFF
```

```
PATH C: \; C: \DOS;
```

```
DOSKEY
```

Most AUTOEXEC.BAT files are much more complex than this. You may want it to include commands to start antivirus and undelete programs from AUTOEXEC.

Take a look at your own AUTOEXEC.BAT:

- 1) Open C: \AUTOEXEC.BAT for editing.
- 2) You should now understand any REM, CLS, PAUSE, and ECHO commands contained in the file. There may be commands that you don't recognize yet. Be careful, do not change any commands that you don't understand. They might be essential to the running of your system.
- 3) Review your path command. Does it contain the directories you want to include in your search path, in the order of most used to least used? If not, edit or replace the command. But, don't remove any directories from the search path unless you're sure you don't need them any more.
- 4) If you want a different command prompt than the one you use now, insert or replace the PROMPT command. If you want to start up DOS shell or Windows automatically when you boot, insert your DOSSHELL or WIN command as the last line in the file. (Note that it's important to make this

the last command in the program.) Include whatever parameters and switches you use when starting up the program at the command prompt.

- 5) Save the file and exit the editor, then, reboot so that DOS will execute the new AUTOEXEC.BAT.

Self Check Exercises

- 1) What is the difference between RAM & ROM?
- 2) What does ROM-BIOS contain?
- 3) Discuss the different types of memory that MS-DOS Supports?

Note: i) Write your answer in the space given below.
 ii) Check your answer with the answers given at the end of this unit.

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

Configuring MS-DOS environment

MSDOS set aside a memory buffer called the environment that it uses for storage of crucial variables such as the PATH and the CONFIG. Other programs can also store and access variables in the DOS environment.

Date and Time

DOS will let you check or change the date and time once it has been set. The procedure to do this is as follows:

Directions:

- 1) Type: date and press return.
- 2) If the correct date is displayed, simply press return. If the date is incorrect, type the correct date and press return.
- 3) Type: time and press return.
- 4) If the correct time is displayed, simply press return. If the time is incorrect, type the correct time and press return.

Path variable

Defines a search path for external executable files (including batch files). The

search path created by PATH is assigned to an environment variable named PATH.

Each time the PATH command defines a search path; it completely replaces the former search path. If you want to add another directory to the current search path, be sure to include the complete current search path in the command.

SET command

Sets or changes values of environment variables or display current values. The command:

```
SET PATH=%PATH%; C:\PROGS
```

for example, adds the directory C:\PROGS to the end of the current search path.

SET - displays the current values of allocation unit environment variables.

```
SET DNT=C:\OLD_DOS
```

Creates (if it doesn't already exist) an environment variable named DNT and gives it the value C:\OLD_DOS. After this command, any time %DNT% is encountered in a batch file; it is replaced by C:\OLD_DOS.

SUBST command

Reassigns a drive name to refer to a path.

```
SUBST [drive: path]
```

Causes future reference to drive to access path instead; displays current substitutions.

Drive does not need to be existing drive, but it must be within the range of possible drive names for the system. Drive can't be current drive.

Path must refer to an existing directory on an existing drive. Path may include a drive name; if it does not, DOS assumes that path is on the current drive at the time SUBST is executed.

Once a substitution is made, any reference to drive is redirected to path. If drive actually exists, you will not be able to access it until you cancel the substitution.

Example:

```
SUBST F: C:\DATADIR
```

Causes any further reference to drive F to access the DATADIR directory on drive C.

COMPSEC command

COMPEC variable is set by DOS to show the location of the command interpreter. DOS accesses this variable whenever it needs to reload the memory resident portion of the command interpreter after some other program has overwritten it.

6.2.6 Directory Structure of DOS

DOS keeps track of each file by maintaining a list of files on the disk. This list is called directory. A directory can contain not only files but other directories that are called subdirectories. These subdirectories can contain other files or more subdirectories, creating a structure of directories within directories. The layered DOS file system of directories and Subdirectories is called Multilevel or hierarchical system, wherein the first level is the Root directory. It is represented by a backslash ”/”.

Files can be grouped into directories to help you keep track of them. In Windows 95, directories taken are as folders. Generally, it is a good idea to store groups of related files in a directory. You can name directories, as well. For example, if you wanted to store all of your history papers in a directory, you may want to name that directory History. Directories 99% of the time simply have a name up to 8 characters with no extension, but on rare occasions a directory will have an extension.

Directories can be placed inside each other. These inner directories are called subdirectories. If you had some essays and some homework questions and wanted to store those individually under the History directory, you can create an essay and/or homework subdirectory.

The name for the history directory with the essay subdirectory might look like this:

```
c:\history\essays>
```

6.2.7 MKDIR Command (Make Directory)

Now, let’s make a couple of directories so we can put our file away in a place that makes sense to us. The command for making a directory is “md” (for “make directory”) and is followed by the name of the directory you wish to make.

```
1 C:\> md dosclass
   (to make the directory DOSCLASS)
1 C:\> dir
   (to view the directory with the new DOSCLASS subdirectory)
1 C:\> cd dosclass
   (to change to the DOSCLASS directory)
1 C:\DOSCLASS> md samples
   (to make the directory “samples” inside the DOSCLASS directory)
1 C:\DOSCLASS> dir
   (to view the directory with the new SAMPLES subdirectory)
```

Note that as soon as we changed our directory, (cd dosclass) the prompt changed to represent the new directory.

CHDIR command

Changing from one directory to another.

You use the CHDIR command (CD for short) to change from the current directory to one in which you want to work. For example, to change to a

directory called DOSCLASS, type:

```
CHDIR \DOSCLASS {ENTER}
```

If SAMPLES were a subdirectory of the DOSCLASS directory you would type:

```
CHDIR \DOSCLASS\SAMPLES {ENTER}
```

To change to the ROOT directory from any directory or subdirectory type:

```
CHDIR \ {ENTER}
```

Note: that is one to remember.

To change to the parent directory of your current directory type:

```
CHDIR.. {ENTER}
```

RMDIR command

Removing a subdirectory

You use the RMDIR (RD for short) to delete a subdirectory. Before you run RMDIR bear in mind the following points:

- 1 You can only remove a subdirectory if it contains NO files or subdirectories.
- 1 You can only remove one subdirectory at a time
- 1 You cannot remove the ROOT directory
- 1 You cannot remove the current directory; you must first empty it, and then move out to its parent directory to remove it.

For example to remove the subdirectory called SAMPLES

- 1 Type DEL \SAMPLES {ENTER} this command deletes ALL the files in the reports directory
- 1 MS-DOS prompts you with the following message: Are you sure (Y/N) ? if you really want to delete all the files in the directory type Y for yes, if not type N for no.
- 1 Type RD \SAMPLES {ENTER} to remove the subdirectory.

DIR Command Directory

If you need more help in orienting yourself, it sometimes helps to take a look at the files and directories available where you are by using the DIR command (short for directory).

```
C:\>dir
```

This will give you a listing of all the files and directories contained in the current directory in addition to some information about the directory itself. You will see the word volume in this information. Volume is simply another word for a disk that the computer has access to. Your hard disk is a volume, your floppy disk is a volume, a server disk (hard disk served over a network) is a volume. Now you know fancy words for all the parts of the format DOS uses to represent a file.

Volume: C:

Pathname: \DEMO\DOS&WIN\SAMPLES\

Filename: SAMPLE

Here are some helpful extensions of the DIR command:

1 C:\>dir | more

(will display the directory one screen at a time with a < more> prompt—use control-C to escape)

1 C:\>dir /w

(wide: will display the directory in columns across the screen) i.e. width wise

1 C:\>dir /a

(all: will display the directory including hidden files and directories)

Now that we have a grasp on where we are in DOS and how to find out, let's take a look at how we manage the files.

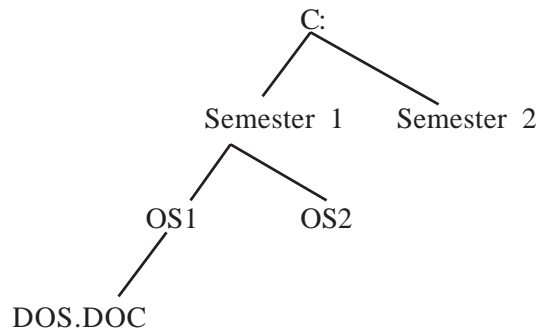


Fig. 2: File Structure in DOS

Self Check Exercises

- 4) Explain the directory structure of DOS?
- 5) Identify the filename and extension of file name CONFIG.SYS?
- 6) Give the path name of file DOS.DOC:

Note: i) Write your answer in the space given below.
 ii) Check your answer with the answers given at the end of this unit.

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

6.2.8 Other MS-DOS Commands

Attrib command

Attrib command changes or displays the attributes of a file. It can also be used to change the attributes of a directory.

A file attribute is a characteristic of the file. For example, a file can be read-only. If it is read-only, it has the r attribute, which prevents it from being changed or deleted.

The four DOS attributes are defined below:

- a** This indicates the archive attribute. It is used with the backup, xcopy and other commands and controls which files are backed up. If you do NOT want to make a file backup when using these commands, remove the archive attribute.
- r** This indicates the read-only attribute. As described previously, it prevents a file from being changed or deleted. You can look at files with this attribute; you just can't erase or change them.
- h** This indicates the hidden attribute. It prevents DOS from displaying a file in a directory list. The file is contained within the directory, but you cannot use the file unless you know its file name.
- s** This indicates the file as a system file. Files with this attribute are not displayed in directory listings. Do NOT mess around with system attributes unless you really know what you are doing.

+	Sets an attribute
-	Clears an attribute
r	Read-Only file attribute
a	Archive file attribute
s	System file attribute
h	Hidden file attribute

To view attributes of a file, use the ATTRIB command, followed by the file name. For example, to view the attributes of the autoexec.bat file on drive C, type:

attrib c:\autoexec.bat

To view the attributes of all the files on drive C using the wildcard command, you would type:

attrib c:*.*

To change the attributes of a file named first.txt:

In order to add the attributes type:

attrib +r first.txt

This will make the attributes of the file read only.

In order to remove the attributes of the file; say the file first.txt has a attribute a-archive then type:

attrib - a first.txt

This will remove the archive attribute of the file first.txt.

FASTOPEN

Starts FASTOPEN, which makes it faster to open frequently access files.

Every time you open a file, FASTOPEN records its name and location. If you reopen the file, FASTOPEN saves time by retrieving its location from the buffer instead of the disk's directory structure.

FASTOPEN works on hard disks only and does not work on network drives. The total number of files being tracked for allocation unit drives must not exceed 999. FASTOPEN requires approximately 48 bytes per file.

MEM (Memory)

A lot of problems with running programs in DOS has to do with memory. Many programs tell you what amount of Conventional memory and what amount of Expanded and/or Extended memory they need.

Knowing these values, you can use MEM to see if you indeed have enough memory available to run your application.

Use **MEM /c|more** to get more information, and have it scroll a page at a time. (See memmaker, below, to learn what to do if MEM turns up showing too little memory available.)

memmaker

When MEM, above, reports that you don't have enough Conventional Memory, Expanded Memory, or Extended Memory, you can use MEMMAKER to (hopefully) free some memory up.

MEMMAKER is a Microsoft program that tries to place programs, which are in Conventional memory somewhere else. Usually it's in a place called HIGH MEMORY, or UPPER MEMORY. DOS can actually directly access one megabyte plus 64K of memory, but unfortunately useless screen modes take about 30% of that area. MEMMAKER can make more Conventional memory available by filling in some nooks and crannies in the total addressable 1 meg+64K area.

deltree

DELTREE is brutal and can really mess you up. Its power is of course, its virtue. With DELTREE you essentially combine the DEL *.* command with the RD command, but it's even better because you can delete all the subdirectories as well, all in one swoop. Definitely this is a good command to know and respect. ALWAYS pause before hitting the ENTER key with this command!

rd (Remove Directory)

When you empty a directory and wish to remove it entirely, you can use RD. RD will not delete the entries (files) in a directory. They must already be deleted (the directory must be empty) for RD to work. However, these two steps can

be simplified by doing them in order. Use the DEL command (described below) and then bring the command line back (using your command stack manipulator). Change DEL to RD and a space character. That way, you can be sure that you will remove the exact directory you just emptied, because you don't retype the directory name. Actually, it's normally a three-step process, since you would normally start with DIR filename, then change DIR to DEL filename, and then change DEL to RD and a space.

For example, to remove the contents from the directory BARNEY and then remove BARNEY, use the following three commands. Bring them back each time with your command stack manipulator, and just change DIR to DEL and then DEL to RD and a space:

dir barney

del barney

rd barney

You could, in theory, also use the command DELTREE to remove both the contents and the directory. You could, but you should refrain from doing that, because DELTREE is a dangerous command and should be used slowly, carefully, and sparingly. It should be a special occasion to use DELTREE, but RD is an everyday thing.

XCOPY

Like COPY, XCOPY can use wildcards to copy a file or group of files to another directory. But XCOPY is more powerful because it has the ability to copy an entire directory structure in one command. Use the /s option to accomplish this. Use HELP XCOPY to learn more, of course!

COPY

The whole purpose of DOS is to let you get around the directory tree easily. You use DOS to start applications and to copy files, delete files, build places to put files, and compare files, and so on. Files, Files, Files. That's what DOS is really for.

The #1 tool for file handling is the COPY command. Specify the "from" file first, then the "to" file. COPY only copies from within one directory at a time, but the "to" directory can be different. For copying entire directory structures, use XCOPY instead.

What makes COPY so powerful is that you can use "wildcards" like * and ? within the filename.

You can use the **COPY** command to copy one file to another.

Directions:

- 1) Type: **copy quick.ltr quick2.ltr** and press **return**.
- 2) DOS will tell you one file has been copied.
- 3) Type: **dir q*** and press **return**.
- 4) You should have two files listed with different names.

Rename REN

The RENAME command lets you rename a previous file with a new name. If the new name which you want to give, already exists, DOS will give an error message.

Directions:

- 1) Type: **ren quick2.ltr sales.ltr** and press **return**.
- 2) Type: **dir *.ltr** and press **return**.
- 3) DOS lists only two files that have the .ltr extension: **quick.ltr and sales.ltr**.

Rename a Group of Files

With the wildcard character *, you can also use the RENAME command to change a group of files.

Directions:

- 1) Type: **ren *.ltr *.bob** and press **return**.
- 2) Type: **dir *.bob** and press **return**.
- 3) You should have two files listed with a .bob extension.
- 4) Type: **dir *.ltr** and press **.**
- 5) There should be no files found with .ltr extensions.

NOTE: You can also rename a group of files that have the same name but different extensions. Type the RENAME command then the filename followed by the dot and the wildcard, space and then type the new filename followed by a dot and the wildcard character. It will look like this: filename.* newfilename.*

ECHO

Turns command echoing features on or off or displays a message.

ECHO [message | ON |OFF]

When command echoing is on, DOS displays the command prompt when it is ready to receive and process a command. When processing a batch program, DOS displays a command prompt line followed by a blank line for each command in the program. Command echoing is on by default.

Turning off command echoing suppresses the command prompt, but you can still type and enter commands. When executing a batch program, it suppresses the command line and the blank line that normally follows it. Only command output is displayed.

ECHO with message displays message as ECHO's command output, so message displays even when command echoing is off.

TREE

TREE, an external command, graphically displays the path of each directory and subtraction-directory on the specified drive. Optionally, it can also list contents,

i.e. list of file in each directory.

Example:

```
C:\>TREE
```

In response to this command, your screen displays a directory structure.

UNDELETE

Enables deletion protection; reports protection status. UNDELETE offers two levels of deletion protection. The higher level, delete Sentry, and the lower deletion-tracking operate only if the UNDELETE TSR is loaded at the time you delete a file. If you don't use either method, you may still be able to recover a file using the information in the DOS directory.

UNFORMAT

Restores a disk reformatted by the FORMAT command.

If an unconditional format was done, UNFORMAT can't restore the disk. If you have added data to a disk since it was reformatted, you can unformat it but some of the old files will have been damaged by the new data.

A safe or quick format stores a copy of the old root directory and FAT in a mirror file, located on the disk where UNFORMAT can find it even after the disk has been reformatted. When used with no switches, UNFORMAT uses information from this file to re-establish the former FAT and root directory, thus reconnecting with the undamaged data in the clusters.

DEL

DEL is the electronic equivalent of a paper shredder. The only way to recover files you have accidentally DEL'ed to death is with some complicated software program that MIGHT or MIGHT NOT work, but then again it might give you new gibberish. And if you DEL'ed star-dot-star (*.*) you can pretty much forget most UNDELETE software programs from succeeding. Think of DEL as short for DISASTER.

That being said, of course, DEL is the standard way of deleting things. But because it's so final, you should almost always use DIR before you use DEL. Here's the plan: first, use DIR to view the filenames you wish to delete. Next, use your command-line buffer to get the same command back, and type DEL over DIR. That way, you will be absolutely sure that you will only DEL the things you just looked at with DIR. It's faster too, usually, than retyping the thing you want to delete. But even if it weren't, this is the best way to delete things.

By the way, whenever you type in DEL *.* , you should always take your hands away from the keyboard and review what you are doing. Are you on the right drive and in the right directory? Do you really want to do this? Same is true for any DEL command with wildcards, at least pause for a second or two, especially since the file recovery programs as mentioned work much more reliably if you just need to recover a single file, but if you need to recover multiple files, they are much less successful.

6.2.9 MS-DOS Batch Files

Use EDIT to create batch files you need to accomplish things you do more than once or twice, or to develop series of commands you want to execute repeatedly.

Batch files don't have to be big. Batch files can call other batch files or jump to other batch files, which increases their usefulness substantially. Batch files are time savers and they help you to do things quickly and efficiently and the same way every time.

Here are some brief descriptions of IF, GOTO, and SHIFT.

Note that these are specific to DOS batch files, and are not used at the command line. Also, note that both IF and SHIFT involve the variables that are available in DOS, namely %0 through %9. They contain the items between the spaces following the batch file name on the command line. For example, a DOS command line might read:

```
RUNPRGMS.BAT GENI MCGA
```

In this example there are two variables after the batch file name (RUNPRGMS.BAT).

%0 would be equal to GENI. %1 would be equal to MCGA. In a batch file, you could put the following line to use the first two variables:

```
AQUI.EXE %0 %1
```

The command line that the executable program (AQUI.EXE) gets would depend on the values of %0 and %1 at the time. Each parameter substitution would be separated by single spaces.

IF Command

IF used in a batch file compares something on the left of a double equal sign (==) with something on the right. So if the thing on the left is %0 and the thing on the right is %1, then IF %0==GENI would be TRUE in this example (the evaluator is case sensitive), and the operation that would follow it would be executed. That might be GOTO, which can follow an IF comparison, or it might be SHIFT, or many other things. You can also use IF NOT... In addition to comparing variables (strings), you can compare the ERRORLEVEL number, or do an EXIST of a file name.

It's often a good idea to check for a null variable separately. Doing something like this does this:

```
IF %0 . == . GOTO NOPARAMETERS
```

GOTO Command

```
GOTO
```

This will jump immediately to a label name — but note: ONLY the first 8 characters are significant, and the first character is a : (colon), and I don't know if it's counted as one of the eight significant characters. Labels can be ahead or behind the GOTO, that is, anywhere else in the batch file.

```
GOTO PLACENAME
```

This batch file command would cause the batch file interpreter to skip forward to the named location. Locations are named with a colon in front:

```
:PLACENAME
```

SHIFT Command

SHIFT

This batch file command will shift variable %1 into variable %0, variable %2 to variable %1, variable %3 to variable %2, variable %5 to variable %4, and so on through %9. There are no parameters to the SHIFT command.

REM command- Makes the following text a REMark and not a command

ECHO command - Puts or suppresses output to the screen

PAUSE command- Pauses the file and waits for a key to be pressed

CALL command- Allows you to CALL and run another batch file from within a batch file

FOR command- Used with IN and DO commands IF Used to test IF an input/parameter is equal or not equal to a given quantity.

6.2.10 File and Hard Disk

FAT (File Allocation Table)

You have seen that your disk drive divides tracks into sectors and reads and writes one sector at a time. But if Dos has to keep track of every sector in every file on a disk, its system, tables would take up more room than the files themselves. To keep its system tables down to a reasonable size, DOS groups sectors together into allocation units, and keeps track of the allocation units instead.

The size of an allocation unit depends on the overall size of the disk; DOS decides on the best size when it prepares the disk for use. Floppy disks have one or two sectors per allocation unit. A high capacity hard drive may have four or more sectors per allocation unit.

Because DOS tracks allocation units instead of sectors, it can't assign part of an allocation unit to a file. It always allocates file space in whole allocation units; that's why they call them allocation units. Even if file contains only one byte, DOS gives it a complete allocation unit; the rest of the space, which is often referred to as slack, is wasted. When the file expands, however, there's plenty of room for it. If the file expands beyond the first allocation unit, DOS assigns it a second complete allocation unit.

		EOF	0	0	6	10	EOF
EOF	0	11	EOF	13	14	15	EOF
0	0	BAD	20	21	EOF	23	24
37	0	0	0	0	0	31	32
33	34	35	36	54	38	39	EOF

A Simple File Allocation Table

Each allocation unit on the drive has a numeric address, starting with 2 and continuing through the end of the disk. The files directory entry shows the

address of the first allocation unit in the file. In the FAT, the entry for that allocation unit shows the address of the next allocation unit. That entry shows the address of the next allocation unit. This pattern continues until you reach the last allocation unit in the file, which contains a special code that means EOF (end of file). Because FAT entries link to each other in chain like fashion, the set of entries for one file is called the file chain.

Because FAT is so important to controlling both the file and the empty space on the disk, DOS maintains two identical FAT's- one as the copy/ backup of the other. Every time you create a new file, or expand or contract an existing one, DOS needs to update both FATs. If you interrupt it before it can complete its update, you throw the fats out of whack, and you may not be able to access your files correctly. To avoid this problem, don't reboot or turn the power off when a drive light is on (unless you need to interrupt the virus at work). The drive lights indicate that DOS is currently accessing the drive.

Volume and Volume Labels

You can create volume label for a disk when you format it. The label stored internally (in the root directory), and appears in such places as directory and tree listings. DOS does not use the label; it identifies disks by their serial numbers. The label is mostly for your use. Other programs, however, might pay attention to disk labels. Some Install or Setup programs, for example check disk labels to make sure that the correct disks have been inserted.

You can view the label of a disk with the VOL command. This command displays a volume label and serial number, if they exist.

Syntax: VOL [drive] (displays the label of the drive specified, else displays the label of the current drive)

To change a label, one can use the LABEL command at the command prompt.

Syntax: LABEL [drive] <label name>

NOTE:

One can most likely view the volume labels of the drives but probably can't change them.

Filename Attributes

Every file has five attributes: read-only, hidden, system, archive and directory. These attributes control how DOS and other programs access the files. Each attribute can be on or off. When an attribute is on, the programs are supposed to pay attention to it (but see the following warning). When an attribute is off, programs ignore it.

Warning: DO not trust attributes alone to protect your files from accidental or malicious damage. Many programs ignore or override the attributes. Furthermore, anyone who knows DOS basics knows how to change a file's attributes.

Read-Only:

When the read-only attribute is turned on, the file should not be modified or deleted. Most of the DOS utilities display an ACCESS DENIED message if you try to modify or delete the file with a positive read-only attribute.

Hidden:

A positive hidden attribute means that a program should ignore the file when looking for the files to process. The DIR command, for example normally will not list the hidden files, neither will DEL command delete the file because it cannot see it. Hide a file if you want to avoid any accidental moving, copying, or deleting of file.

System:

The system attribute combines features of the read-only and hidden attributes; a system file normally can't be modified or erased, and it usually can't be seen by programs when searching for files.

Archive:

The archive attribute indicates that a file has been modified since last the last time you made backup copies. This attribute signifies that the file needs to be backed up (archived). DOS's backup programs, and most other backup programs, can select files to be backed up on the basis of their archive attributes. They turn the archive attribute off after making a backup copy, because at that point the file does not need to be archived

Partitioning Disks

Partitioning is one of the necessary steps to prepare a drive for use. It is the process of defining certain areas of the hard disk for the operating system to use. A volume is a section of the drive with a letter, like C: or D:. All hard drives must be partitioned, even if they will have only one partition.

A partition program writes a master partition boot sector to cylinder 0, head 0, sector 1. The data in this sector defines the start and end locations of each of the other partitions. It also indicates which of these partitions is active, or bootable, thus telling the computer where to look for the operating system.

All systems can handle 24 partitions, either spread out on the same drive or many drives. This means that one can have up to 24 different hard drives, according to DOS. DOS can't recognize more than 24 partitions, although some other operating systems may. The limiting factor is simply the availability of letters. All partitions must have a letter. There are 26 letters, A: and B: are reserved for floppy drives, leaving 24 letters available.

Although there are third party partitioning programs that boast added capabilities, DOS FDISK is the accepted program for partitioning. FDISK sets up the partition in an optimum way and allows more than one OS to operate on one system.

FDISK only shows two DOS partitions, the primary partition and the extended partition. The extended partition is divided into logical DOS volumes, each being a separate partition. The minimum partition size is one megabyte, due to the fact that FDISK in DOS 4.0 or later creates partitions based on numbers of MB. Partition size is usually limited to 2GB. DOS versions earlier than 4.0 allow maximum partitions of 32 MB.

Primary Dos Partition

The first partition is your primary DOS partition. This is your C: drive and can't be divided. This is also called the active partition. You can only have one active partition.

The second partition is optional. It is called an extended partition. This is the space left over after the primary partition. Each extended partition must be labeled with a letter D: through Z:.

To start this, type “fdisk” at the A: prompt. If this doesn’t work, it is because your drive is not installed correctly.

First, you have to setup a primary DOS partition. Choose Option 1 (Create DOS partition or Logical DOS drive). Choose Option 1 in the next menu. Now you can make your entire or only part of the C: drive the primary partition. Many people make the entire drive one partition just to stay simple. If you don’t want to do so, specify the amount of drive space you want to partition in either megabytes or percentage of total drive. If you are using a percentage, be sure to follow the number by a “%” or the computer will think you’re talking MBs.

Next, you’ll need to make this partition active. Return to the main FDISK menu and choose Option 2 (Set Active Partition). Follow the prompts.

Extended Dos Partition

If you’re going to create an extended partition, choose Option 1 again, but this time choose Option 2 in the next menu (Create Extended DOS partition). Plug in the percentage of drive to partition for this one. Do not make this partition active. Only one can be active.

After you create an extended partition, you will be given the Create Logical Drives option in the extended partition menu. Follow the on-screen instructions to assign drive letters to your partitions D: through Z:. Keep in mind that D: is often used for the CD-ROM.

Non-MS-DOS Partition

FDISK can be used to set up a non-DOS partition. For example, suppose you want to be able to use UNIX as well as DOS. After physically formatting the hard disk, you can use FDISK to create a DOS partition and a non-DOS partition; then you install UNIX in the non-DOS partition. When you boot from the DOS partition, DOS is installed. When you boot from the UNIX partition, UNIX is installed.

Partition SETUP

If you created a non-DOS partition for an alternative operating system, you need to designate which partition to boot from. On FDISK’s main menu, choose the second option, Set Active Partition. FDISK displays the partitions that are eligible as boot drives and lets you choose the one you want to boot from.

FDISK command

Starts the FDISK program, which partitions a hard disk for use with DOS; displays the status of hard disk(s).

FDISK [/STATUS] [/MBR]

/STATUS displays an overview of the partition information of your hard disk(s). Does not start FDISK.

/MBR rebuilds the Master Boot Record. Does not start FDISK.

Do not experiment with partitioning. If you accidentally delete or change a drive or partition, you could lose the data from it.

Use FDISK /MBR if you suspect that the Master Boot Record has a virus attached to it and you can't find another way to remove it. /MBR removes the virus; it does not repair any damage already done to the Master Boot Record. Reboot immediately after using FDISK /MBR.

EDIT command

Starts the DOS Editor, which creates and changes ASCII text files.

EDIT does not work if QBASIC.EXE is not in the current directory, in the search path, or in the same directory as EDIT.COM. If you delete QBASIC.EXE to save space, you cannot use the DOS Editor.

Self Check Exercises

- 7) What is the Difference between Internal and External commands?
- 8) What does the following command used for -
(i) MD (ii) CD (iii) Dir (iv) REN

Note: i) Write your answer in the space given below.
ii) Check your answer with the answers given at the end of this unit.
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....

6.3 MS-WINDOWS

Windows gives life to modern PCs; its what sets new PCs apart from their dull typewriter like forebears. Windows gives you a graphical environment, so you can see what you're doing. Windows gives a consistent point and shoot user interface, so that allocation unit of all software works alike. And Windows gives you the ability to run multiple programs at once, so your computer can keep up with the most valuable resource around you.

6.3.1 Advantage of Using WINDOWS vs DOS

Windows brings together features available in older style operating systems and adds to them user friendly features so that you can do more work quicker from your desktop or portable computer. Some of the main benefits include:

- 1 Improved interface
- 1 Easier file management, including support for networked connections and long filenames

- 1 New Plug and Play feature automatically detects and uses additional devices you attach to your computer
- 1 True 32-bit multitasking enables several programs to run simultaneously so you can get more work done
- 1 Improved search facility
- 1 Better multimedia support
- 1 Extended communications capability, including E-mail, faxes, bulletin boards, Internet.

The basic foundation underlying Windows is its 'windowing' capability. A window (spelt with a lower-case w) is a rectangular area used to display information or to run a program. Several windows can be opened at the same time to work with multiple applications, so you should be able to dramatically increase your productivity when using your computer.

6.3.2 Versions of MS WINDOWS

Windows 98

Windows 98 was officially released on June 25th 1998. Internet Explorer 4.0 is integrated as a component of Windows 98 just like Internet Explorer 3.0 to OSR2. There is also a built-in TV tuner, a better Dial-Up Network and a system shield called Dr.Watson which prevents crashes ensuring a better and smoother environment. Although Microsoft makes no guarantees that general system performance will increase after installing Windows 98, many users have seen an improvement after upgrading from Windows 95 + IE4.0. 486DX / 66 MHz, 16 MB of memory. Typical installation requires approximately 195 MB of free hard disk space, but may range between 120 MB to 295 MB, depending on your system configuration and the options you choose to install. CD-ROM or DVD-ROM drive (3.5" high-density disks available for additional charge). VGA or higher-resolution monitor. Microsoft Mouse or compatible pointing device. The Windows 98 interface is more like a Web browser. So you can use the same navigation and viewing techniques whether you're searching your hard disk or the Internet. Most of these options can be adjusted or switched off to resemble Windows 95.

Windows XP

Windows Whistler is the codename for the first joint business/consumer version of Windows built on the Windows 2000 OS kernel for stability and reliability. To bring the reliability of Windows 2000 to consumers as soon as possible and address the feedback from enterprise customers, Microsoft planned a phased release of Windows XP beginning with the desktop products and followed by the server versions. Windows XP is their first step in the Microsoft. Net vision. Windows XP Professional are available in 24 official localized versions in addition to English. In any version of Windows XP Professional, you can display, input, edit, and print documents in hundreds of languages. Users or administrators can set regional preferences, fonts, keyboard layouts, sorting orders, date formats, and so on.

The biggest innovation for multilingual computing is the Windows XP Multilingual User Interface Pack, which is an add-on to the English version of Windows XP Professional. It lets users or administrators switch the User Interface (menus, dialogs and help files) from one language to another. This feature greatly eases

desktop administration in multilingual computing environments, and lowers cost of desktop change and configuration management considerably. The Multilingual User Interface Pack will be available in 33 languages.

Windows Millennium Edition

Windows Millennium Edition is the latest release of the Windows 9* operating system. It is designed solely for the home user.

Not a great deal is new in Windows “Millennium Edition” despite the hype that is sure to come out around its release date. Here are the basics from the Press Release Microsoft sent out.

- 1 “Digital Media and Entertainment” Digital media is becoming increasingly popular, as illustrated by the exponential growth in areas like music on the Web and digital photography. The Consumer Windows Division will focus on enabling users to take advantage of all this new content, making it easy to access, play/view and store as well as providing an enhanced PC gaming experience.
- 1 “Online Experience” Providing consumers a premier home online experience is a primary goal for the Consumer Windows Division. This means consumers can easily connect to the Web, locate desired content and determine which content is right for their family.
- 1 “Home Networking” With more than 15 million households now owning two computers and the cost of new PCs continuing to fall, as well as the many intelligent hardware devices being created, networking at home is becoming a reality for more people. The Consumer Windows Division will work to simplify the process of connecting multiple computers in the home, enabling them to share information and an Internet connection, and provide the infrastructure for connecting different intelligent devices to the PC.
- 1 “It Just Works” The Consumer Windows Division is committed to providing consumers with a solution that ‘just works,’ from the moment a user starts their PC and throughout their daily computing experience. This promise will be delivered upon by the advancement of the PC’s self-healing functionality, in addition to providing a simpler set-up and a great out-of-the-box experience for new computer users.

There are a number of other new features such as the following:

- 1 **Universal Plug and Play:** Universal Plug and Play (UPnP) defines an architectural framework for creating self-configuring, self-describing devices and services. It enables seamless connectivity and communication between Windows and intelligent appliances in the home, office and everywhere in between.

Windows 2000

One of the main problems with Windows 2000 is compatibility. Windows 2000 isn’t compatible to all hardware and software. Though Microsoft is working on an added driver database most likely there is a chance you could experience problems with hardware, with no drivers installed or the wrong drivers installed. If you have only been using Windows for a year and a half or less, you are not an Advanced computer user, do not use a network or have a business, Windows 2000 is most likely not for you and you should explore the operating system before upgrading.

Windows CE .NET

Microsoft® Windows® CE .NET is an open, scalable, 32-bit operating system that is designed to meet the needs of a broad range of intelligent devices, from enterprise tools such as industrial controllers, communications hubs, and point-of-sale terminals to consumer products such as cameras, Internet appliances, and interactive televisions. A typical Windows CE-based embedded system is targeted for a specific use, often runs disconnected from other computers, and requires an operating system that has a small footprint and a built-in deterministic response to interrupts.

New and enhanced features include:

- 1 New built-in base configurations expedite development for common platforms
- 1 Export SDK, New Platform, and BSP wizards streamline platform development
- 1 Kernel Independent Transport Layer (KITL) supports customized communication services
- 1 Feature-based Catalog enables easy platform feature selection

Automatic dependency checking seamlessly adds supported features. Windows CE. Net has Wireless following technologies :

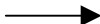
- 1 **Bluetooth:** Bluetooth is an emerging wireless communication technology that allows devices within a 10-meter proximity to communicate with each other.
- 1 **802.1x/Zero Configuration:** Windows CE .NET introduces 802.11 Zero Configuration, 802.1x and Extensible Authentication Protocol (EAP).
- 1 **Object Exchange (OBEX) Protocol:** Object Exchange (OBEX) is an efficient, compact binary protocol that enables a wide range of devices to exchange data spontaneously in a simple, efficient manner.
- 1 **MediaSense:** The MediaSense feature improves the roaming experience for mobile device users by notifying applications of network status. For example, MediaSense will notify the application whether the embedded device is in or out of range.

6.3.3 Basic Features of WINDOWS

The following are the different features of Windows:

Desktop

When you start your computer, the first thing you see is the desktop. The desktop is your work area.

Start Button 

Taskbar	By default, the taskbar is located on the bottom edge of the desktop. You can click on the taskbar and drag it to other locations. The Start button, active program buttons, icons for quick access to programs, and the current time are located on the taskbar.
My Computer	The My Computer icon provides access to the resources on your computer. You can access your drives and other peripherals by clicking on the My Computer icon. You can also access the Control Panel through My Computer.
Internet Explorer	The Internet Explorer icon launches the Internet Explorer browser.
Network Neighborhood	If you are working on a network, Network Neighbourhood displays all the computers on the network.
The Recycle Bin	When you delete an object, Windows 98 sends it to the Recycle Bin. You can restore objects that are located in the Recycle Bin or you can permanently delete them.
My Briefcase	My Briefcase is used to coordinate files between two computers.
Shortcut icon	Icons with an arrow in the lower left corner are shortcut icons. Click on the icon for quick access to the object they represent (program, document, printer, and so on).
Program, Folder, and Document icons	Program, folder, and document icons do not have an arrow in the lower left corner. They represent the actual object and provide direct access to the object.

Start Button

The Start button on the task bar is designed with the beginner in mind. Because it is not intuitive for a beginner to double click the mouse button to launch a program, the Start button allows you to select and start a program quickly just with a single mouse click. Other common tasks that you need to do using your computer are also available directly from the Start button.

The Start button option include:

Programs	Quickly allows you to start a program.
Documents	Enables you to start work on one of the 15 most recent documents you've been working on.
Settings	Allows changes to settings for your computer
Find	A sophisticated search facility
Help	A complete on-line help system
Run	Used to start a program
Shutdown	A safe way to switch off, restart or log off your computer.

My Computer

The My Computer icon provides access to the resources on your computer. You can access your drives and other peripherals by clicking on the My Computer icon. You can also access the Control Panel through My Computer.

Recycle Bin

When you delete an object, Windows 98 sends it to the Recycle Bin. You can restore objects that are located in the Recycle Bin or you can permanently delete them.

Network Neighborhood

If you are working on a network, Network Neighborhood displays all of the computers on the network.

6.3.4 Support for MS-DOS

The three ways of running dos programs in win95

- 1) **In a Windows 95 "DOS Box":** This is the easiest way to run DOS programs in Windows 95. You can create shortcuts, set properties, and run them from "Start/Run" etc. Note that under "properties" you can customize each DOS program shortcut as needed with a custom config.sys and

autoexec.bat file, run it via “MSDOS Mode” etc. This is a huge waste of time. One can load everything DOS applications need in the real config.sys and autoexec.bat files just like older DOS versions!

- 2) **In Windows 95 “MSDOS Mode”:** This special mode, new in Windows 95, allows you to run DOS programs that don’t like being run in a “DOS Box.” With this mode you get a little better emulation of real DOS and usually a little more free conventional memory for DOS applications. But-this mode has full blown Win95 lurking in the background. To exit “MSDOS Mode” you do not reboot your computer, you type “EXIT” at a command prompt.

IMPORTANT: There is a very special batch file called **DOSSTART.BAT** in your \WINDOWS directory is executed ONLY when you run a program in “MSDOS Mode” or “Shutdown/Restart in MSDOS Mode.” Usually, the Windows 95 setup will REM out your real mode mouse, CDROM (MSCDEX.EXE) and soundcard drivers in your original CONFIG.SYS and AUTOEXEC.BAT files and copy some of these lines to your DOSSTART.BAT.

Check your DOSTART.BAT and see what is in it before you try playing with MSDOS mode

- 3) **From a boot up Win95/DOS7.0 “DOS prompt”:** There are some DOS programs, especially games, that will not run from a “DOS Box” but may either run OK but slower or not at all in “MSDOS Mode” because of the overhead of Win95 lurking in the background.

With few exceptions, you can edit your Win95 MSDOS.SYS file to have it give you a “Boot Menu” at startup so you can either go into the GUI (“Normal”) or boot to “Command Prompt Only.” This prompt is just plain “DOS7.0” without Windows 95 “lurking” in the background. See my Windows 95/DOS Tips for details.

6.3.5 Dial up Network and Internet Explorer

Connecting to the Internet using Dial-Up Networking

The steps in this procedure will guide you through the process of connecting to the Internet with Dial-Up Networking.

Here’s a quick look at the steps required for connecting to the Internet with Dial-Up Networking:

- 1) Get an Internet account with an Internet service provider.
- 2) Set up your communications hardware.
- 3) Make sure Dial-Up Networking is installed.
- 4) Install the TCP/IP protocol and bind it to the Dial-Up Networking adapter.
- 5) Enter TCP/IP information.
- 6) Create a connection to your access provider in Dial-Up Networking.
- 7) Dial your Internet service provider

Step 1 Getting an Internet Account

When you call an ISP to sign up for an Internet account, make sure you get the following information:

- 1 User name
- 1 Password
- 1 Access phone number
- 1 Your host name and domain name
- 1 Domain Name System (DNS) server address

The service provider may also supply the following information:

- 1 IP address and subnet mask
- 1 DNS search order (if required)
- 1 Gateway address
- 1 Authentication procedure (whether or not the service provider requires a terminal window to log on)

Step 2 Setting up your Communications Hardware

You need a modem-9600 bits per second (bps) or faster is recommended, to connect to the Internet. Or, if your organization has a direct connection to the Internet, you can use a network adapter card instead of a modem.

Step 3 Make Sure You Have Dial-Up Networking Installed

- 1) Double-click **My Computer** on your desktop. If you do not see a **Dial-Up Networking** icon, then follow steps 2 through 4.
- 2) Click **Add/Remove Programs Properties** dialog box at the Windows Setup tab.
- 3) Click **Communications**, and then click **Details**.
- 4) Select the **Dial-Up Networking** check box, and then click **OK**.

Step 4 Install the Windows 98 TCP/IP Protocol

- 1) Display the **Network** dialog box.
- 2) Click **Add**. (If the Windows 98 TCP/IP protocol is already installed, you can go to step 6.)
- 3) Click **Protocol**, and then click **Add**.
- 4) Click **Microsoft**, and then click **TCP/IP**.
- 5) Click **OK**.
- 6) Click **Dial-Up Adapter**, and then click **Properties**. Or, if your organization has a direct Internet connection, click your network adapter, and then click **Properties**.
- 7) Click the **Bindings** tab, and then make sure the **TCP/IP** check box is selected.

If prompted, restart your computer. Then, start Help and return to this procedure.

Step 5 Entering TCP/IP Information

- 1) Display the **Network** dialog box.
- 2) Click **TCP/IP** (if there is more than one **TCP/IP** entry, click **TCP/IP -> Dial-Up Adapter**), and then click **Properties**.
- 3) If your service provider gave you an **IP Address**, on the **IP Address** tab, click **Specify an IP address**, and then type your IP address and subnet mask.

If your service provider did not give you an IP address, click **Obtain an IP address automatically**.

If you are using a direct network connection to the Internet, type your IP address and subnet mask. If your network uses a DHCP server to assign IP addresses, click Obtain an IP address automatically.

- 4) On the **Bindings** tab, it is recommended that you click to clear the **File and printer sharing** check box.

Step 6 Setting up a Connection to your Access Provider

- 1) Click **Start**, point to **Programs**, point to **Accessories**, point to **Communications**, and then click **Dial-Up Networking**.
- 2) Click **Make New Connection**. If it is your first Dial-Up Networking connection, you'll be prompted automatically. Follow the instructions on your screen.
- 3) Right-click the connection icon, and then click **Properties**.
- 4) If your provider requires a terminal window to log on, click **Configure**. Click the **Options** tab, make sure the **Bring up terminal window after dialing** check box is selected, and then click **OK**.
- 5) Click the **Server Type** tab, and then make sure **Type of Dial-Up Server** is set to PPP.

You can decrease the time required to connect by making sure that the following options are not selected:

- 1 **Log On To Network**
- 1 **NetBEUI**
- 1 **IPX/SPX Compatible**

Step 7 Dialing your Internet Service Provider

In the Dial-Up Networking window, click the connection for your Internet service provider, enter your name and password, and then click **Connect**.

Follow the instructions provided by your Internet service provider for browsing the Internet, reading e-mail, and accessing information.

6.3.6 The Internet Explorer Web Browser

Internet Explorer makes it easier to get the most from the World Wide Web, whether you are searching for new information or browsing your favorite Web sites. And built-in IntelliSense technology can save your time completing routine Web tasks, such as automatically completing Web addresses and forms for you, and automatically detecting your network and connection status.

The easiest way to get around the Web

- 1 When you start typing a frequently used Web address in the Address bar, a list of similar addresses appears that you can choose from. And if a Web-page address is wrong, Internet Explorer can search for similar addresses to try to find a match.
- 1 Search for Web sites by clicking the Search button on the toolbar. Then in the Search bar, type a word or phrase that describes what you're looking for. When your search results appear, you can view the individual Web pages without losing your list of search results.
- 1 You can also search directly from the Address bar. Just type common names or words, and Internet Explorer can automatically take you to the site that most likely matches what you are searching for, and list other likely sites as well.
- 1 Go to other Web pages similar to the one you are viewing, without even doing a search. Just use the Show Related Links feature.
- 1 Once you're on a Web page, Internet Explorer can help you complete entries in any kind of Web-based form. Start typing, and a list of similar entries appears that you can choose from.
- 1 Browse through a list of Web pages you recently visited by clicking the History button on the toolbar. In addition you can rearrange or search the History list.
- 1 Listen to broadcast and Internet-only radio stations while you browse. Choose from a wide variety of music and talk radio stations, and add them to your Favorites list. The Radio bar is available in Internet Explorer when you install Windows Media Player.

A customizable browser

- 1 Put shortcuts to your most-used Web pages on your Links bar for quick access.
- 1 Add other frequently visited Web pages to your Favorites list for easy access. Organize your favorite items by using folders, and sort them in the order you want them.
- 1 Choose from your favorite search providers and keep them handy for all your searches.
- 1 Take your favorite Web pages with you: to another computer or browser, or to share with your friends. You can even import bookmarks from Netscape Navigator.

Safety in Browsing the Web

- 1 Using Content Advisor, you can screen out objectionable content by using industry-standard ratings that have been defined independently by the Platform for Internet Content Selection (PICS) committee.
- 1 Using security zones, you can set different levels of security for different areas of the Web to help protect your computer.

Displays Web pages in other languages

- 1 If your Web browsing takes you to sites written in numerous languages, Internet Explorer can update your computer with the character sets you need to view these sites correctly.

Accessibility Features

You can open the **Accessibility Properties** dialog box by clicking Start, pointing to Settings, clicking **Control Panel**, and then double-clicking **Accessibility Options**.

Microsoft is committed to making its products and services easier for everyone to use. This section provides information on the following features, products, and services that make Microsoft Windows operating systems more accessible for people with disabilities:

- 1 Customizing Windows Operating Systems
- 1 Microsoft Services for People Who Are Deaf or Hard-of-Hearing
- 1 Microsoft Documentation in Alternative Formats
- 1 Utilities to Enhance Accessibility

6.3.7 Customizing Windows Operating Systems

There are many ways you can customize Windows operating systems to make your computer more accessible.

- 1 Accessibility features have been built into Windows operating systems since the introduction of Windows 95. These features are useful for individuals who have difficulty typing or using a mouse, have moderately impaired vision, or who are deaf or hard-of-hearing. The features can be installed during setup, or you can add them later from your Windows installation disks. Look up “accessibility” in the Help Index for information about installing and using these features.
- 1 Some of the accessibility features built into Windows operating systems can be added to earlier versions of those products, and to MS-DOS, through Access Pack files. You can download these files or you can order them on disks from Microsoft. (See details in “Accessibility Notes and Utilities to Download or Order” later in this topic.)
- 1 You can also use Control Panel and other built-in features to adjust the appearance and behavior of Windows operating systems to suit varying vision and motor abilities. These include adjusting colors and sizes, sound volume, and the behavior of the mouse and keyboard.

Microsoft Services for People Who Are Deaf or Hard-of-Hearing

For those who are deaf or hard-of-hearing, complete access to Microsoft product and customer services is available through a text telephone (TTY/TDD) service.

Microsoft Documentation in Alternative Formats

- 1 In addition to the standard forms of documentation, many Microsoft products are also available in other formats to make them more accessible.
- 1 Many of the Internet Explorer documents are also available as online Help, online user’s guides, or on a CD that comes with the package.
- 1 If you have difficulty reading or handling printed documentation, you can obtain many Microsoft publications from Recording for the Blind & Dyslexic, Inc. RFB&D distributes these documents to registered, eligible members of their distribution service, either on audio cassettes or on floppy disks. The RFB&D collection contains more than 80,000 titles, including Microsoft product documentation and books from Microsoft Press. You can also download many of these books from the Microsoft Web site at <http://microsoft.com/enable/>.

Utilities to Enhance Accessibility

A wide variety of hardware and software products are available to make personal computers easier to use for people with disabilities. Among the different types of products available for the MS-DOS and Windows operating systems are:

- 1 Programs that enlarge or alter the color of information on the screen for people with visual impairments.
- 1 Programs that describe information on the screen in Braille or synthesized speech for people who are blind or have difficulty reading.
- 1 Hardware and software utilities that modify the behavior of the mouse and

keyboard.

- 1 Programs that enable people to “type” by using a mouse or their voice.
- 1 Word or phrase prediction software that allows users to type more quickly and with fewer keystrokes.

Alternative input devices, such as single switch or puff-and-sip devices, for people who cannot use a mouse or a keyboard.

6.3.8 Start Menu

To restart the computer

To shutdown and immediately restart your computer:

- 1) Click on the Start button. The Start menu will appear.
- 2) Click on Shut Down. The Shut Down Windows dialog box will appear.
- 3) Click on the Restart radio button.
- 4) Click on OK.

To start a program

To start a program:

- 1) Click on the Start button, located in the lower left corner of your screen.
- 2) Highlight Programs. The Program menu will appear.
- 3) Move to the Program menu and highlight the program you want to start. If you see a right pointer next to your selection, a submenu will appear. Refine your choice by highlighting the appropriate selection on the submenu. Continue until you get to the final menu.
- 4) Click on the program name to start the program.

To add an item located on the desktop to the Start or to a Program menu

To add an item on the desktop to the Start or to a Program menu:

- 1) Left-click on the item, and drag the item on top of the Start button.
- 2) Release the left mouse button when the Start menu appears.
- 3) The item will appear on the Start menu.
- 4) If you would prefer to have the item on a Program or submenu of the Start menu, drag the item from the Start menu to the Program or submenu.

To add an item located in Windows Explorer to the Start or to a Program menu

To add an item located in Windows Explorer to the Start or to a Program menu:

- 1) Click on the Start button. The Start menu will appear.

- 2) Click on Settings. A submenu will appear.
- 3) Click on Taskbar & Start Menu. A dialog box will appear.
- 4) Click on the Start Menu Programs tab.
- 5) Click on the Add button.
- 6) Type the path to the item you want to add, or use Browse to navigate to the item.
- 7) Click on Next.
- 8) Double-click on the appropriate folder.
- 9) Click on Finish.
- 10) Click on OK.

To remove an item from the Start or from a Program menu

To remove an item from the Start or from a Program menu:

- 1) Click on the Start button. The Start menu will appear.
- 2) Click on Settings. A submenu will appear.
- 3) Click on Taskbar & Start Menu. A dialog box will appear.
- 4) Click on the Start Menu Programs tab.
- 5) Click on the Remove button.
- 6) Find and click on the item you want to remove.
- 7) Click on the Remove button.
- 8) Click on Close.
- 9) Click on OK.

To copy an item that is located on the Start or on a Program menu

To copy an item located on the Start or on a Program menu:

- 1) Highlight the item.
- 2) Right-click.
- 3) Left-click on Copy.

To rename an item on the Start or on a Program menu

To rename an item on the Start or on a Program menu:

- 1) Highlight the item.
- 2) Right-click on the item.
- 3) Click on Rename. The Rename dialog box appears.

- 4) Type the new name in the field New Name

To delete a file from the Start or from a Program menu

To delete a file from the Start or from a Program menu:

- 1) Highlight the item.
- 2) Right-click.
- 3) Left-click on Delete.

To re-sort the Start or a Program menu

To re-sort the Program menu:

- 1) Go to the menu.
- 2) Right-click.
- 3) Click on Sort By Name.

Note: Not all menus can be re-sorted.

The Favorites option on the Start menu

You can use the Favorites option on the Start menu to quickly move to files, programs, or Web sites. When you bookmark a Web page in Internet Explorer, the bookmark is listed on the Favorites menu. To add a file or program to the Favorites menu, create a shortcut and then move the shortcut to the Favorites menu.

The most recently used document list

As you work, Windows 98 tracks the last 15 files you used. It lists these files on the most recently used document list. To view the most recently used document list:

- 1) Click on Start.
- 2) Highlight Documents. The most recently used documents will display.

To open a file listed on the most recently used document list, click on the file name.

To clear the most recently used document list

To clear the most recently used document list:

- 1) Click on the Start button. The Start menu will appear.
- 2) Highlight Settings.
- 3) Click on Taskbar and Start menu. A dialog box will appear.
- 4) Click on the Start Menu Program tab.

Click on Clear. Click on OK.

To shut down your computer:

- 1) Click on the Start button. The Start menu will appear.
- 2) Click on Shutdown. The Shut Down Windows dialog box will appear.
- 3) Click on the Shut Down radio button.
- 4) Click on OK.

6.3.9 Windows Help

Windows 98 includes a new Help system that makes your computer easier to use. With Help, you can find answers to your questions quickly and easily. You can also get up-to-date technical support from the World Wide Web. To start Help, click Start, and then click Help. To get Help from the Web, click Web Help at the top of the Help page

Using Applications

executable = what you should type in the "START>RUN" dialog box

Windows95 Utilities

winfile	Windows file manager
command	C: prompt
explorer	navigate all types of Windows95 stuff with this one interface

Network Utilities

winipcfg	Print out network card specifications (including hardware address, ip address, gateway, etc...)
ping	see if the network interface is configured properly by pinging the gateway (ip address with ".1" instead of your network number at the end)
telnet	Assuming that you have telnet installed and the network interface is configured properly on your Windows95 system, telnet allows you to log in.
ftp	upload and download files to and from your Windows95 computer with this program

Opening Applications

To start a program, click the **Start** button. Position the mouse over **Programs**. A submenu will appear. Position the mouse over the name of the program you want to start and click one time. If you wanted to open the Windows95 “Sound Recorder” program, for example, you would:

- 1) Click the **Start** button.
- 2) Position the mouse over **Programs**. A submenu will appear.
- 3) Position the mouse over **Accessories**. A second submenu will appear.
- 4) Position the mouse over **Multimedia**. A third submenu will appear.
- 5) Select **Sound Recorder** by positioning the mouse over it and clicking one time.

The **Sound Recorder** program will open.

Using Paint

Paintbrush is the graphic drawing program provided with windows. Using Paintbrush you can create images such as the one shown in the Figure below.

After you create an image using Paintbrush, you can print the image, save it to a file, or copy the image to the clipboard for use by another Windows application such as Write. To start Paintbrush, perform these steps:

- 1 Click the Start button; from this Start menu select Programs -> Accessories -> Paint.

Menus, Buttons, and Icons

What is an Icon?

An icon is a graphic image. Icons help you execute commands quickly. Commands tell the computer what you want the computer to do. To execute a command by using an icon, click on the icon.

Icons are the most basic element in your Windows System. They represent:

- 1 Individual files ranging from Microsoft Word Documents to Adobe Photoshop graphics files to Excel Spreadsheets,
- 1 Folders which are simply groups of files,
- 1 Application Programs which you can run by double-clicking on the icon, or
- 1 Shortcuts, which provide an “alias” as a more convenient way to access any of the aforementioned three items.

Single-clicking on an icon with the left mouse button (i.e. “left-clicking” on the icon) selects that icon. Selected icons are displayed in reverse-video. You can add to the list of selected icons by holding down the Control key while clicking on other icons. Alternatively, if you want to select a group of adjacent icons, you can click and drag a box around those icons with the mouse.

Right-clicking on an icon brings up a menu for that icon. The items contained

in this menu change depending on what icon you right-click on, and they usually are a list of actions which pertain to the icon in question. This “shortcut menu” provides convenient access to the various functions you can perform on the item that the icon refers to.

Right-clicking on and dragging an icon to another place (let’s call that the destination), Windows 95 gives you the option to either move the icon to the destination, make the icon to the destination, or create a shortcut to that icon and leave the shortcut in the destination.

Menu

Menus provide a way for you to send commands to the computer (tell the computer what you want the computer to do). When you open a window, menu options are listed from left to right just below the title bar on the menu bar. When you click on a menu item, a drop-down menu appears. Select the command you want to execute from the drop-down menu. An ellipse after a drop-down menu item signifies that there are additional options; if you select that option, a dialog box will appear.

What is a shortcut key?

You can use shortcut keys to execute a command quickly by pressing key combinations instead of selecting the commands directly from the menu or clicking on an icon. When you look at a menu, you will notice that most of the options have one letter underlined. You can select a menu option by holding down the Alt key and pressing the underlined letter. You can also make Alt-key selections from drop-down menus and dialog boxes.

A drop-down or pull-down menu

Fields with a drop-down menu have a small downward pointing arrow next to them. You click on the arrow and a list of options appears. You select the option you want from the list. You can also open the drop-down menu by holding down the Alt key and pressing the down arrow.

You can use the arrow keys to move up and down in a drop-down menu. You can also move to an item by typing the first few letters of the option.

Radio buttons

Windows 98 and programs that run under Windows 98 use radio buttons to present a list of mutually exclusive options. You can select only one of the options presented. Radio buttons are usually round. A dot in the middle indicates that the option is selected.

Saving and exiting

To save a file a WordPad document:

- 1) Click on File, which is located on the menu bar. A drop-down menu will appear.
- 2) Click on Save. A dialog box similar to the one shown here will appear.

Field/Icon	Entry
Save In field	Click to open the menu-box and select the drive and folder to which you want to save the file.
Up One Level icon	Click on this icon to move up one level in the folder hierarchy.
View Desktop icon	Click on this icon to move to the Desktop folder.
Create a New Folder icon	Use the Create a New Folder icon to create a new folder: 1. Click on the Create New Folder icon.

	<ol style="list-style-type: none"> 2. Type the folder name and press Enter. 3. Click on the folder you just created to open the folder.
List icon	Your program displays files and folders in the center of the dialog box. To have the files display without the size, type, and date modified, click on the List icon.
Detail icon	Your program displays files and folders in the center of the dialog box. To have the files display with the size, type, and date modified, click on the Detail icon.
File/Folder box	Your program displays files and folders in File/Folder box. Click on a folder to open the folder. Click on a file if you want the current file to write over (replace) that file.
File Name field	Enter the name you want your file to have in this field.
Save As Type field	Click to open the drop-down box and select a file type.
Save button	Click on the Save button to save your file.
Cancel button	Click on the Cancel button if you have changed your mind and do not wish to save your file.

Long Filenames:

Windows 95 the ability to use “Long File Names” greater than 8.3 (filename.txt) we were stuck with in older versions of DOS

6.3.10 Organisation of Information

Files and folders

Folders

Folders are used to organize the data stored on your drives. The files that make up a program are stored together in their own set of folders. You will want to organize the files you create in folders. You will want to store files of a like kind in a single folder.

Finding Files

To quickly find a file or folder:

- 1) Click on the Start button.
- 2) Highlight Find.
- 3) Highlight Files or Folders.
- 4) Press Enter. The dialog box shown here will appear. You do not need to complete every field. Only enter the criteria you want to use for your search.

Field	Entry Type
Name & Location Tab	
Named	Enter the filename, the first few letters of the filename, or any letters found in the file name. Use the * as a wild card. For example, to find all of the files that begin with r and end in the extension .doc, enter r*.doc. To find files that begin with resume and have any extension, enter resume.*.
Containing Text	If you are looking for a file that has a specific word or phrase in the file, enter the word or phrase in this field.
Look In	Select the drive or folder you want to search.
Include Subfolders	Check this box if you want subfolders to be searched
Browse	Click on the Browse button to manually select the drive and folder to be searched.
Date Tab	
All Files	Select this option if you do not want to restrict your search to a specific date or period.
Find Allocation unit Files	Select from Modified, Created, or Last Accessed. Select Modified to find all files modified since the date criteria you enter, select Created to find all files created since the date criteria you enter, or select Last Accessed to find all files accessed since the date criteria you enter.
Between/During	Specify the date search criteria you
the Previous	want to use. Between allows you to search for files modified, created, or accessed between two dates. During allows you to search for files modified, created, or accessed during the previous number of days or months you specify.

Advanced Tab	
Of Type	Select a file type from the drop-down menu.
Size Is	Select from At Least and At Most and specify the size criteria you want to use in your search.
Name & Location Tab	
Find Now	Click on Find Now to start your search.
Stop	Click on Stop to abort your search.
New Search	Click on New Search to clear your present criteria and start a new search.

6.3.11 Using Windows Explorer

To open Windows Explorer, locate Windows Explorer on your Start menu and then click on it. Alternatively, hold down the Windows key and type e (Windows-e).

Organisation of Windows 98 files and folders on drives

Windows 98 organizes folders and files in a hierarchical system. The drive is the highest level of the hierarchy. You can put all of your files on the drive without creating any folders. But, that is like putting all of your papers in a file cabinet without organizing them into folders. It works fine if you have only a few files, but as the number of files increases, there comes a point at which things become very difficult to find. So you create folders and put related material together in folders.

A diagram of a typical drive and how it is organized is shown here.

At the highest level, you have some folders and perhaps some files. You can open any of the folders and put additional files and folders into them. A hierarchy develops.

Windows Explorer window

When you open Windows Explorer, the screen shown here will appear.

Windows 98 separates the window into two panes. If you click on an object in the left pane, the contents of the object displays in the right pane. Click on Desktop and the contents of the Desktop folder display on the right. Click on My Computer and your computer resources display on the right. To see the contents of a drive, click on the drive. To see the contents of a folder, click on the icon for the folder in the left pane.

To create a new folder:

- 1) In the left pane, click on the drive or folder in which you want to create the new folder.
- 2) Click on any free area in the right pane. A context menu will appear.
- 3) Highlight New.
- 4) Click on Folder.
- 5) Type in a name for the folder.

Windows Explorer views

Views control how Windows Explorer displays information in the right pane. Windows Explorer provides you with the following choices: Large Icons, Small Icons, List, and Details. Large Icons and Small Icons, as their names imply, determine the size of the icon. List displays all of the files and folders without supplying the size, type, or date modified. Details displays the size, type, and date modified. To change the view:

- 1) Right-click on any free area in the right pane. A context menu will appear.
- 2) Highlight View.

- 3) Select the view you want from the drop-down menu.

To delete a file or folder:

- 1) Right-click on the file or folder you want to delete. A context menu will appear.
- 2) Click on Delete. Window Explorer will ask, “Are sure you want to send this object to the recycle bin?”
- 3) Click on “Yes.”

To copy a file or folder:

- 1) Right-click on the file or folder you want to copy. A context menu will appear.
- 2) Click on Copy. The file or folder should now be on the Clipboard.

To cut a file or folder:

- 1) Right-click on the file or folder you want to cut. A context menu will appear.
- 2) Click on Cut. The file or folder should now be on the Clipboard.

Note: Cutting differs from deleting. When you cut a file, the file is placed on the Clipboard. When you delete a file, the file is sent to the Recycle Bin.

To paste a file or folder:

- 1) After cutting or copying the file, right-click on the object or in the right pane of the folder to which you want to paste. A context menu will appear.
- 2) Click on Paste.

To rename a file or folder:

- 1) Right-click on the file or folder. A context menu will appear.
- 2) Highlight Rename.

Type the new name.

Sending files to recycle bin

You can drag file or folder icons into the **Recycle Bin**. If you press **SHIFT** while dragging, the item will be deleted from your computer without being stored in the **Recycle Bin**.

Recovering files from recycle bin

To retrieve deleted files or shortcuts

- 1) On the desktop, double-click the.
- 2) Click the file or shortcut you want to retrieve.
- 3) On the **File** menu, click **Restore**.

Notes

- 1 If you restore a file that was originally located in a deleted folder, the folder is recreated, and then the file is restored in the folder.
- 1 Files deleted at the command prompt, files deleted from network locations, and files deleted from removable media (such as disks) are not moved to the **Recycle Bin**. They are permanently removed when you delete them.
- 1 To open a file that is in the **Recycle Bin**, drag the icon onto the desktop, and then click it.
- 1 To retrieve several files at once, hold down the CTRL key, click each file you want to retrieve, and then click **Restore** on the **File** menu.

6.3.12 Using Notepad

To start the Notepad Application click the Start Button-> choose Programs -> Accessories->Notepad.

To find specific characters or words

- 1) On the **Search** menu, click **Find**.
- 2) In **Find what**, type the characters or words you want to find.
- 3) Click **Find Next**.

To cut, copy, paste, or delete text

- 1 To cut text so you can move it to another location, select the text, and then on the Edit menu, click Cut.
- 1 To copy text so you can paste it in another location, select the text, and then on the Edit menu, click Copy.
- 1 To paste text you have cut or copied, place the insertion point where you want to paste the text, and then on the Edit menu, click Paste.
- 1 To delete text, select it, and then on the Edit menu, click Delete.


To wrap text to the window size

On the Edit menu, click Word Wrap.

Note

- 1 Wrapping text makes all the text visible on the screen, but it doesn't affect the way text appears when it is printed.

Minimize and restoring Windows**Minimize a window**

- 1 Click the Minimize button  on the menu bar or title bar of the active document to minimize the document window to a button on the Windows taskbar.

OR to Minimize all program windows on the Windows desktop


- 1 Right-click the background of the Windows taskbar, and then click Minimize All Windows on the shortcut menu.

To Restore the Minimized Window:

Do one of the following:


- 1 If the Word window or another Word document is showing, click the name of the document you want to restore on the Window menu.
- 1 If all Word documents are minimized, click the name of the document you want to restore on the Windows taskbar, or hold down ALT and press TAB until you see the document you want.


Change Size of the Window


- 1) If the windows are taking up the entire screen, click on the Restore button  to shrink the size of your window
- 2) You can also manually resize the window by bringing the mouse to the edges of the window until the mouse turns into a double-headed arrow
- 3) Then hold down the left mouse button, and drag the edge of the window to the desired size.

TIP: resizing from the lower right corner of the window keeps the edges proportional.


- 4) Once you've resized your windows, you can arrange them on the screen by left-mouse clicking on the title bar, holding down the mouse, and dragging the window over to the desired location the screen.

 **Minimize Button** : Push this button to minimize the Netscape window. This closes the window, but does not stop the application. The window may be re-opened on the desktop by pressing the Netscape button on the taskbar (see below).

 **Full Screen Button** : Push this toggle button to expand the Netscape window to full screen.

 **Terminate Program Button** : Push this button to terminate the Netscape application.

Many application windows contain a re-size button in the bottom right corner; below is the bottom window bar of the Netscape application.

 **Re-size Button** : Drag this button to make the window larger or smaller.

Adding Documents to the Desktop

A desktop shortcut

A desktop shortcut, usually represented by an icon, is a small file that points to a program, folder, document, or Internet location. Clicking on a shortcut icon takes you directly to the object to which the shortcut points. Shortcut icons contain a small arrow in their lower left corner. Shortcuts are merely pointers-deleting a shortcut will not delete the item to which the shortcut points.

Create a desktop shortcut

To create a shortcut to an item located on the Start menu:

- 1) Click on Start. The Start menu will appear.
- 2) Locate the item to which you want to create a shortcut. If the item is located on a submenu, go to the submenu.
- 3) Right-click and drag the item to your desktop.

To create a shortcut to items visible to Windows Explorer:

- 1) Open Windows Explorer.
- 2) Minimize the Windows Explorer window.
- 3) Locate in Windows Explorer the item to which you want to create a shortcut.
- 4) Hold down the right mouse button and drag the item onto the desktop.
- 5) Release the right mouse button. A context menu will appear.
- 6) Click on Create Shortcut(s) Here.

Turn a Web link into a desktop shortcut

To turn a Web link into a desktop shortcut, click on the link in your browser window (usually underlined text) and drag it to the desktop. An icon will appear on your desktop. When you click on the icon, your browser will open and you will go directly to the Web page.

The desktop shortcut wizard

Right-click on the desktop. The context menu will appear.

- 1) Click on New. A submenu will appear.
- 2) Click on Shortcut. The Create Shortcut menu appears.
- 3) Type in the location and name of the item to which you want to create a shortcut. Alternatively, browse to find the item.
- 4) Click on Next. A dialog box will appear.
- 5) Accept the default name or type in a new name.
- 6) Click on Finish.

Rename a desktop shortcut

To rename a shortcut:

- 1) Right-click on the shortcut.
- 2) Click on Rename.
- 3) Type the new name.

Delete a desktop shortcut

To delete a shortcut:

- 1) Click on the shortcut.
- 2) Press the delete key.

Change the icon associated with an object

To change the icon associated with an object:

- 1) Right-click on the icon. The context menu will appear.
- 2) Click on Properties.
- 3) Click on the Change Icon button.
- 4) Click on the icon of your choice.
- 5) Click on OK.

Note: Not all icons can be changed. If you do not see the Change Icon button, the icon can not be changed

Arranging Desktop

To arrange the icons on the Desktop Right-Click and choose the option Arrange the Icons and then the way you want it to because i.e. either by name, type, size etc.

Changing Settings

To change the settings click on the Start->Settings -> Control Panel

Mouse setting

Open the Mouse Properties by clicking the Start button->Setting->Control panel ->Mouse icon.

To adjust the double-click speed for your mouse

- 1) Under **Double-click speed**, drag the slider.

To reverse your mouse buttons

- 1) Click **Right-handed** or **Left-handed**.

Using a mouse with two buttons

Use the left button-the primary button-for most tasks, unless the right button is specified in a Help procedure.

The right button enables you to quickly accomplish common tasks. If you click an item by using the right button, a menu is displayed containing commands specific to the item. People who are more comfortable with Windows will find this useful. If you are less comfortable with Windows, you can still use the left button and menus to complete these same tasks.

To turn on and adjust the mouse pointer trail

- 1) Click the **Motion** tab.
- 2) Under **Pointer trail**, click Show pointer trails.
- 3) To adjust the length of the pointer trail, drag the slider.

To adjust the speed of your mouse pointer

- 1) Click the **Motion** tab.
- 2) Under **Pointer speed**, drag the slider.

To change the appearance of your mouse pointer

- 1) Click the **Pointers** tab.
- 2) To change all your pointers at one time, click the box under **Scheme**.

- 3) To change only one pointer, click it, click **Browse**, and then double-click the file name of the pointer you want to use.

Screen setting

You can open the **Display Properties** dialog box by clicking **Start**, pointing to **Settings**, clicking **Control Panel**, and then double-clicking **Display**.

To change the background of the desktop

- 1) In **Wallpaper**, click the background you want to use or click **Pattern** to choose or modify the background.
- 2) Click **Apply** to see the changes before you close the dialog box, or click **OK** to accept the changes and close the dialog box.

Note

- 1 You can use most graphic files, such as bitmaps (.bmp), GIFs (.gif), and JPEG (.jpeg) images, as background.
- 1 To cover your entire screen with a small wallpaper image, click **Tile**. To center a wallpaper image, click **Center**.
- 1 You can use patterns and wallpaper simultaneously. However, if **Tile** is selected, you cannot see the pattern.
- 1 By default, only a limited set of wallpapers is installed during **Windows Setup**. For information about installing other wallpaper, click **Related Topics**, and then click **To add or remove a Windows component**.

To change the number of colors your monitor displays

- 1) In **Colors**, click the number of colors that you want your monitor to display.

Note: Your monitor and display adapter determine the maximum number of colors that can appear on your monitor.

To copy the window or screen contents

- 1 To copy an image of the window that is currently active, press **ALT+PRINT SCREEN**.
- 1 To copy an image of the entire screen, press **PRINT SCREEN**.

Note

- 1 To paste the image into a document, on the **Edit** menu in the document window, click **Paste**.

To change the size of the screen area

- 1) In **Screen area**, click the desktop size.

Notes

- 1 Your monitor and display adapter determine whether you can change your screen resolution.

To calibrate your display to actual dimensions

- 1) In **Screen area**, make sure the setting is higher than 640 by 480 pixels. If 640 by 480 pixels is the only setting available to you, you cannot calibrate your display to actual dimensions.
- 2) Click **Advanced** to open the properties page for your computer.
- 3) On the **General** tab, in **Font Size**, click **Other**.
- 4) Hold a ruler up to the ruler on your screen, and then drag the screen ruler until it matches the ruler you are holding.

To use larger or smaller display fonts

- 1) Click **Advanced** to open the properties page for your monitor.
- 2) On the **General** tab, in **Font Size**, click the size you want your displayed fonts to be.

To customize the size of displayed fonts, click **Other**.

Notes

- 1 If **Font Size** is unavailable, make sure your setting in **Screen area** is higher than 640 by 480 pixels. If 640 by 480 pixels is the only setting available to you, you cannot change your display font.

To turn on High Contrast

- 1) Open the **Accessibility Properties** dialog box.
- 2) Click the Display tab.
- 3) Under **High Contrast**, select the **Use High Contrast** check box.

Notes

- 1 You can open the **Accessibility Properties** dialog box by double-clicking **Start**, pointing to **Settings**, clicking **Control Panel**, and then clicking **Accessibility Options**.
- 1 To change settings for click **Settings**.
- 1 If the **Use shortcut** check box in the **Settings for High Contrast** dialog box is selected, you can turn on and turn off the high-contrast scheme by pressing LEFT ALT+LEFT SHIFT+PRINT SCREEN.

To change the way items on the desktop look

- 1) Open the **Display Properties** dialog box at the **Appearance** tab.
- 2) If you want to change the appearance of only one screen element, click that element in **Item**, and then change the settings in **Size** and **Color** for the item and its font.

If you want to change the appearance of all screen elements simultaneously, click an appearance scheme in **Scheme**.

Notes

- 1 You can open the **Display Properties** dialog box at the **Appearance** tab by

clicking **Start**, pointing to **Settings**, clicking **Control Panel**, double-clicking **Display**, and then clicking the **Appearance** tab.

- 1 If you select the **Active Title Bar** or the **Inactive Title Bar** in **Item**, you can change the Gradient. It is not available if your computer is set to 256 colors.

If you change individual settings, you can save them by clicking **Save As** and then typing a name for the scheme. That name will appear in Scheme, so you can easily restore the settings later.

Printer setting

To install a new printer:

- 1) Make sure your printer is plugged in, connected to your computer, turned on, and has paper in it.
- 2) Click on the Start button. The Start menu will appear.
- 3) Highlight Settings. A submenu will appear.
- 4) Click on Printers. The Printer control panel will appear.
- 5) Double-click on Add Printer. The Add Printer Wizard will open.
- 6) Click on Next. A dialog box will open.
- 7) Select the appropriate option, either Local Printer or Network Printer.
- 8) Click on Next. A dialog box will open.
- 9) Select the manufacturer of your printer from the Manufacturers list box.
- 10) Select the model of your printer from the Printers list box. If your printer is not on the list, contact the manufacturer to obtain the proper drivers.
- 11) Click on Next. You might be prompted to insert the Windows CD so the necessary files can be copied from it.
- 12) Select the correct port from the available ports list box. The most commonly used ports are LPT1 and COM1 (or COM with some other number). Use LPT1 if your printer uses a parallel cable. If you need to use the COM, click on the Configure Port button. Use the documentation for your printer to determine the correct settings.
- 13) Click on Next. A dialog box will appear.
- 14) Accept the printer that defaults, or type in a new printer name. The name you choose will appear under the Printer icon and in program dialog boxes.
- 15) Select "Yes" in the "Do you want Windows based programs to use this as the default printer?" field, if you want this printer to be your default printer.
- 16) Click on Next.
- 17) You will be prompted: "Would you like to print a test page?" Select "Yes."
- 18) Click on Finish. After a minute or two, your test page should print

To cancel a print job:

- 1) Click on the Start button. The Start menu will appear.
- 2) Highlight Settings. A submenu will appear.
- 3) Click on Printers. The Printer control panel will appear.
- 4) Double-click on the printer to which you sent the print jobs. The Printer window will open.
- 5) Click on the job you want to stop. If you want to stop more than one job, hold down the Control key while you click on the additional jobs.
- 6) Click on Document, which is located on the menu bar.
- 7) Click on Cancel.

To cancel every print job :

- 1) Click on the Start button. The Start menu will appear.
- 2) Highlight Settings. A submenu will appear.
- 3) Click on Printers. The Printer control panel will appear.
- 4) Double-click on the printer to which you sent the print jobs. The Printer window will open.
- 5) Click on Printer, which is located on the menu bar.
- 6) Click on Purge Print Documents.

To temporarily stop selected jobs from printing:

- 1) Click on the Start button. The Start menu will appear.
- 2) Highlight Settings. A submenu will appear.
- 3) Click on Printers. The Printer control panel will appear.
- 4) Double-click on the printer to which you sent the print job. The Printer window will open.
- 5) Click on the document you want to pause. If you want to pause more than one document, hold down the Control key as you select the additional documents.
- 6) Click on Document, which is located on the menu bar.
- 7) Click on Pause Printing. A checkmark should appear next to Pause Printing.

To restart a print job you temporarily stopped:

- 1) Click on the Start button. The Start menu will appear.
- 2) Highlight Settings. A submenu will appear.

- 3) Click on Printers. The Printer control panel will appear.
- 4) Double-click on the printer to which you sent the print job. The Printer window will open.
- 5) Click on the documents you paused. If more than one document has been paused, hold down the Ctrl key as you click on the additional documents.
- 6) Click on Document, which is located on the menu bar.
- 7) Click on Pause Printing. The checkmark next to Pause Printing should disappear.

To temporarily stop all jobs from printing:

- 1) Click on the Start button. The Start menu will appear.
- 2) Highlight Settings. A submenu will appear.
- 3) Click on Printers. The Printer control panel will appear.
- 4) Double-click on the Printer to which you sent the print jobs. The Printer window will open.
- 5) Click on Printer, which is located on the menu bar. A drop-down menu will appear.
- 6) Click on Pause Printing. A checkmark should appear next to Pause Printing.

To restart a print queue that has been stopped:

- 1) Click on the Start button. The Start menu will appear.
- 2) Highlight Settings. A submenu will appear.
- 3) Click on Printers. The Printer control panel will appear.
- 4) Double-click on the printer to which you sent the print job. The Printer window will open.
- 5) Click on Printer, which is located on the menu bar. A drop-down menu will appear.
- 6) Click on Pause Printing. The checkmark next to Pause Printing should disappear.

Date and time

To change the date and/or time:

- 1) Click on the Start button, which is located in the lower left corner of the screen. The Start menu will appear.
- 2) Highlight Settings. A submenu will appear.
- 3) Click on Control Panel. The Control Panel will open.
- 4) Click on Date/Time. The Date/Time Properties dialog box will appear.

Software

- 5) In the Date frame, select the month and year.
- 6) In the Month field, click to open the drop-down menu and select the current month.
- 7) Type the year in the Year field or use the arrows next to the field to move forward or backward until you get to the current year.
- 8) The Time field is divided into four segments: hour, minutes, seconds, and AM and PM. To make an adjustment:
- 9) Click in the segment and either type in the correct information or use the arrow keys on the right side to select the correct hour, minute, second or AM or PM.
- 10) Select the correct time zone from the drop-down menu.
- 11) If you want the clock to automatically adjust to daylight saving time, check the box on the screen.
- 12) Click on the Apply button.
- 13) Click on OK.

Keyboard setting

You can open the **Keyboard Properties** dialog box at the **Language** tab by clicking **Start**, pointing to **Settings**, clicking **Control Panel**, double-clicking **Keyboard**.

To change the way the keyboard responds

- 1) On the **Speed** tab, make changes as necessary:
 - 1 To adjust how much time elapses before characters repeat when you hold down a key, drag the **Repeat delay** slider.
 - 1 To adjust how quickly characters repeat when you hold down a key, drag the **Repeat rate** slider.

To add a keyboard language or layout

- 1) Click **Add**.
- 2) In the **Add Language** dialog box, click the you want to add, and then click **OK**.

Notes

- 1 When the **Enable indicator on taskbar** check box on the **Language** tab is selected and you have two or more keyboard languages installed, an indicator representing the default keyboard language appears on the taskbar. To quickly switch between keyboard languages, click the indicator, and then click the keyboard language you want to use.

To change the keyboard layout for an installed keyboard language

- 1) Click the **Language** tab.
- 2) Click the keyboard language of the you want to change, and then click **Properties**.
- 3) Click the keyboard layout you want to use, and then click **OK**.

To click by using MouseKeys

- 1 To click (for example, on a menu command), press 5 on your numeric keypad.
- 1 To double-click, press the PLUS SIGN (+) on your numeric keypad.
- 1 To right-click, press the MINUS SIGN (-) on your numeric keypad, and then press 5 to click, or press the PLUS SIGN (+) to double-click.
- 1 To click as if you were using both mouse buttons at once, press the ASTERISK (*) on your numeric keypad, and then press 5 to click or the PLUS SIGN (+) to double-click.
- 1 To switch back to standard clicking, press SLASH (/) on your numeric keypad.

To delete a keyboard language or layout

- 1) At the **Language** tab.
- 2) Click the keyboard language and keyboard layout you want to delete, and then click **Remove**.

To drag by using MouseKeys

- 1 To move the pointer over the object you want to move, press the arrow keys on your numeric keypad.
- 1 To hold down the mouse button, press the INS key on your numeric keypad.
- 1 To move the pointer over the area you want to move the object to, press the arrow keys on your numeric keypad.

To release the mouse button, press the DEL key on your numeric keypad.

Password

You can open the Password Properties dialog box by clicking Start, pointing to Settings, clicking Control Panel, and then double-clicking Passwords.

To change your Windows password

- 1) Click **Change Windows Password**, and then click **OK**. If you have other network passwords, these will be listed. If you want, you can change them to match your new Windows password.
- 2) Type your old password.
- 3) Type your new password, and then type it again in **Confirm New Password**.

Notes

If you are using Windows with a network, you need to be logged on to the network to change your Windows password.

To change your network password

- 1) Click **Change Other Passwords**.
- 2) Click the password you want to change, and then click **Change**.
- 3) Type your old password.
- 4) Type your new password, and then type it again in **Confirm New Password**.

To change your Windows password

- 1) to open the **Passwords Properties** dialog box.
- 2) Click **Change Windows Password**, and then click **OK**. If you have other network passwords, these will be listed. If you want, you can change them to match your new Windows password.
- 3) Type your old password.
- 4) Type your new password, and then type it again in **Confirm New Password**.

Notes

If you are using Windows with a network, you need to be logged on to the network to change your Windows password.

To change your network password

- 1) Click **Change Other Passwords**.
- 2) Click the password you want to change, and then click **Change**.
- 3) Type your old password.
- 4) Type your new password, and then type it again in **Confirm New Password**.

Adding and removing programs

You can open the **Add/Remove Programs Properties** dialog box by clicking **Start**, pointing to **Settings**, clicking **Control Panel**, and then double-clicking

To install a program

- 1) Follow the instructions on your screen.

To install a program from a network

- 1) If the **Install/Uninstall** tab is selected, and your computer is already connected to a network, click **Install**, and then click **Next**. Follow the instructions on your screen to browse for the setup program on your network.

If the **Network Install** tab is selected, follow the instructions on your screen.

Notes

- 1 If you do not have a **Network Install** tab in the **Add/Remove Programs Properties** dialog box, this feature has not been enabled on your computer, your network, or both. See your system administrator.

To remove a program from your computer

- 1) Follow the instructions on your screen.

Notes

Only programs that were designed for Windows can be removed by using the **Add/Remove Programs Properties** dialog box.

Windows Registry. The Windows Registry is a database where all the information about your computer is stored. Everything from colors, installed applications, or changes you make in Control Panel, are included in the Registry database.

If you want to extend the customization of Windows, you'll need to understand the Registry, and how it works.

It is possible to break Windows by changing the Registry. Any changes you make should be done with caution. If you take the simple step of making a backup copy of the Registry, you virtually eliminate the possibility of disaster. This is explained in detail in another chapter. Users of Windows will find this a very simple operation to perform.

The Registry is edited with a tool found in your Windows folder. That tool is the Registry Editor. It's called Regedit.exe and is included with Windows for the purpose of viewing and editing the Registry.

To start the Registry Editor, all you do is type "regedit" into the Run line of the Start Menu. You'll see the following window.

This is where you'll do your editing. The left pane is the tree view with folders called **keys**. The right pane shows the contents of the currently selected folder. The contents are called **values**. The items in the left pane aren't really folders, they're just called folders in order to give them a convenient method of naming and displaying the information.

Here's the Registry Editor with a key selected, displaying its values in the right pane, shown below

Each key can contain other keys, as well as values. Each value contains the actual information stored in the Registry itself. Keys are shown in the left pane, and values are shown in the right pane.

There are three types of values: String, Binary, and DWORD. Their use is determined by the program that created them. To display a value, click on the key (folder) of choice and look at its value in the right pane. To expand a key, click the plus sign next to it, or double click on the folder.

The Registry Editor allows you to perform the following operations:

- 1 New keys or values can be added by choosing the location for the new object, and selecting New from the Edit menu.
- 1 Rename any existing value and almost any key with the same method used to rename files: right-click and select Rename, click twice on the name slowly and type the new name, or select the name press F2.
- 1 Delete a key or value by clicking on it and pressing the Del key, or right-click and select Delete.
- 1 Search for text in a key or value by selecting Find from the Edit menu.
- 1 Refresh the displayed portion of the Registry by selecting Refresh from the View menu.

Self Check Exercises

- 9) What are the basic components of the windows environment?
- 10) What is the function of the Taskbar?
- 11) Explain Window Registry

Note: i) Write your answer in the space given below.
 ii) Check your answer with the answer given at the end of this unit.

.....

6.4 UNIX

Overview

After three decades of use, the UNIX computer operating system from Bell Labs is still regarded as one of the most powerful, versatile, and flexible operating

systems (OS) in the computer world. Its popularity is due to many factors, including its ability to run a wide variety of machines, from micros to supercomputers, and its portability – all of which led to its adoption by many manufacturers.

History of UNIX

Like another legendary creature whose name also ends in ‘x,’ UNIX rose from the ashes of a multi-organizational effort in the early 1960s to develop a dependable timesharing operating system.

The joint effort was not successful, but a few survivors from Bell Labs tried again, and what followed was a system that offers its users a work environment that has been described as “of unusual simplicity, power, and elegance...”

The system also fostered a distinctive approach to software design – solving a problem by interconnecting simpler tools, rather than creating large monolithic application programs.

Its development and evolution led to a new philosophy of computing, and it has been a never-ending source of both challenges and joy to programmers around the world.

6.4.1 Main Features

Multi-user Capabilities

A multi-user system permits several users to use the same computer simultaneously. More than one terminal can be connected to one computer, and the users of the terminal can run allocation unit programs, access files, and print the documents at once. The O/S manages the requests made to the computer by managing users, sorts them from interfering with each other, and assigns priorities when two or more users want to access the same file at the same time. A computer system that can support multiple users is generally less expensive than the equivalent number of single user machines.

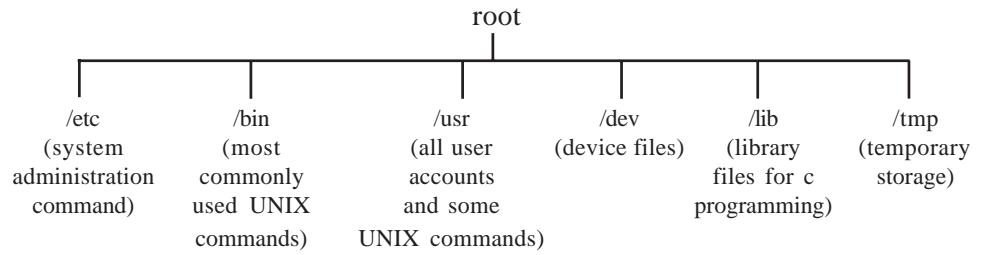
Multitasking Capabilities

Multitasking means that a given user can do more than one task at the same time. Multitasking with computer lets you simultaneously perform tasks formerly performed sequentially. This helps the set of tasks to be completed quickly. This is achieved by placing some tasks in the “background” and some of the tasks in the “foreground”.

Portability

Portability is the ability of the software that operates on one machine to operate on another, different machine. Hence, it is easier to modify the UNIX system code for installation on a new computer. The ability to transport the UNIX system from one brand of computer to another has been the major reason of acceptance of this system. The Unix System runs on more brands and types of computers.

The Hierarchical File System



At the top is the root or the first directory from which the second level of the directories descends. The initial level of the directories is standard to most UNIX systems and is described in the above figure.

6.4.2 Unix Operating System

The UNIX operating system is made up of three parts; the kernel, the shell and the programs.

The kernel

The kernel of UNIX is the hub of the operating system. It allocates time and memory to programs and handles the filestore and communications in response to system calls.

As an illustration of the way that the shell and the kernel work together, suppose a user types **rm myfile** (which has the effect of removing the file “**myfile**”). The shell searches the file store for the file containing the program **rm**, and then requests the kernel, through system calls, to execute the program **rm** on **myfile**. When the process **rm myfile** has finished running, the shell then returns the UNIX prompt % to the user, indicating that it is waiting for further commands.

The shell

The shell acts as an interface between the user and the kernel. When a user logs in, the login program checks the username and password, and then starts another program called the shell. The shell is a command line interpreter (CLI). It interprets the commands the user types in and arranges for them to be carried out. The commands are themselves programs. When they terminate, the shell gives the user another prompt (% on our systems).

The adept user can customize his/her own shell, and users can use different shells on the same machine. Staff and students in the school have the **tcsh** shell by default.

The tcsh shell has certain features to help the user inputting commands.

Self Check Exercises

12) Discuss the main features of UNIX OS?

13) Describe the ports of UNIX OS?

Note: i) Write your answer in the space given below.

ii) Check your answer with the answer given at the end of this unit.

.....
.....

.....
.....
.....
.....

6.4.3 Utilities File Management

ls command- Listing files and directories

When you first login, your current working directory is your home directory. Your home directory has the same name as your user-name, for example, ee9lab, and it is where your personal files and subdirectories are saved.

To find out what is in your home directory, type

```
% ls (short for list)
```

The ls command lists the contents of your current working directory.

There may be no files visible in your home directory, in which case, the UNIX prompt will be returned. Alternatively, there may already be some files inserted by the System Administrator when your account was created.

ls does not, in fact, cause all the files in your home directory to be listed, but only those ones whose name does not begin with a dot (.) Files beginning with a dot (.) are known as hidden files and usually contain important program configuration information. They are hidden because you should not change them unless you are very familiar with UNIX.

To list all files in your home directory including those whose names begin with a dot, type

```
% ls -a
```

ls is an example of a command which can take options: -a is an example of an option. The options change the behaviour of the command. There are online manual pages that tell you which options a particular command can take, and how each option modifies the behaviour of the command.

rm (Remove)

To delete (remove) a file, use the rm command. As an example, we are going to create a copy of the science.txt file then delete it.

Inside your unixstuff directory, type

```
% cp science.txt tempfile.txt
```

```
% ls (to check if it has created the file)
```

```
% rm tempfile.txt
```

```
% ls (to check if it has deleted the file)
```

cat (concatenate)

The command **cat** can be used to display the contents of a file on the screen. Type:

```
% cat science.txt
```

As you can see, the file is longer than the size of the window, so it scrolls past making it unreadable.

mkdir (make directory)

We will now make a subdirectory in your home directory to hold the files you will be creating and using in the course of this tutorial. To take a subdirectory called **unixstuff** in your current working directory type

```
% mkdir unixstuff
```

To see the directory you have just created, type

```
% ls
```

cd (change directory)

The command **cd <directory>** means “change the current working directory. The current working directory may be thought of as the directory you are in, i.e. your current position in the file-system tree.

To change to the directory you have just made, type

```
% cd unixstuff
```

Type **ls** to see the contents (which should be empty)

Make another directory inside the **unixstuff** directory called **backups**

Still in the **unixstuff** directory, type

```
% ls -a
```

As you can see, in the **unixstuff** directory (and in all other directories), there are two special directories called “.” and “..”

In UNIX, “.” means the current directory, so typing

```
% cd . (NOTE: there is a space between cd and the dot)
```

means stay where you are (the **unixstuff** directory). This may not seem very useful at first, but using “.” as the name of the current directory will save a lot of typing.

“..” means the parent of the current directory, so typing

```
% cd ..
```

will take you one directory up the hierarchy (back to your home directory). Try it now.

Note: typing **cd** with no argument always returns you to your home directory. This is very useful if you are lost in the file system.

rmdir (Remove Directory)

You can use the **rmdir** command to remove a directory (make sure it is empty

first). Try to remove the backups directory. You will not be able to since UNIX will not let you remove a non-empty directory.

cp (copy)

cp file1 file2 is the command which makes a copy of **file1** in the current working directory and calls it **file2**

What we are going to do now, is to take a file stored in an open access area of the file system, and use the **cp** command to copy it to your **unixstuff** directory.

First, **cd** to your **unixstuff** directory.

```
% cd ~/unixstuff
```

Then at the UNIX prompt, type,

```
% cp /vol/examples/tutorial/science.txt .
```

(**Note:** Don't forget the dot "." at the end. Remember, in UNIX, the dot means the current directory.)

The above command means copy the file **science.txt** to the current directory, keeping the name the same.

mv (move)

mv file1 file2 moves (or renames) **file1** to **file2**

To move a file from one place to another, use the **mv** command. This has the effect of moving rather than copying the file, so you end up with only one file rather than two.

It can also be used to rename a file, by "moving" the file to the same directory, but giving it a different name.

We are now going to move the file **science.bak** to your backup directory.

First, change directories to your **unixstuff** directory. Then, inside the **unixstuff** directory, type

```
% mv science.bak backups/.
```

Type **ls** and **ls backups** to see if it has worked.

User Management

passwd

To change your password:

passwd

Old password: enter your current password

New password: enter your new password

Retype new password: re-enter your new password

Enter the command `passwd` and then respond to the prompts by entering your old password followed by your new one. You are then asked to confirm your new password.

The passwords will not appear on the screen as you type, to prevent other people from seeing them. If you make a mistake, the message

Mismatch - password unchanged.

is displayed and your password remains unchanged. Try again.

chmod (changing a file mode)

Only the owner of a file can use `chmod` to change the permissions of a file. The options of `chmod` are as follows

Symbol	Meaning
u	user
g	group
o	other
a	all
r	read
w	write (and delete)
x	execute (and access directory)
+	add permission
-	take away permission

For example, to remove read write and execute permissions on the file **biglist** for the group and others, type

% `chmod go-rwx biglist`

This will leave the other permissions unaffected.

To give read and write permissions on the file **biglist** to all,

% `chmod a+rw biglist`

chgrp

Changing group ownership of files and directories

Every user is a member of one or more groups. To find out which groups you belong to use the command:

groups

To find out which groups another user belongs to use the command:

groups username

Your files and directories are owned by the group (or one of the groups) that you belong to. This is known as their “group ownership”.

To list the group ownership of your files:

```
ls -gl
```

You can change the group ownership of a file or directory with the command:

```
chgrp group_name file/directory_name
```

You must be a member of the group to which you are changing ownership to.

6.4.4 Process Management

When you enter a command it invokes a program. While this program is running it is called a process. It is important to grasp that although there is only one copy of a program held in the file system, any number of processes can be invoked which run this program.

When the operating system is started after a boot, a single process is started. This process is the parent of all subsequent processes. Each process created on the system has a unique number, known as its PID, associated with it.

When you login to the system a process is started to run your shell program. Any processes that are started from within your shell - such as entering a command - are the children of this process. A process can have many children, but only one parent.

ps Monitoring processes

To monitor the state of your processes use the ps command.

```
ps [-option]
```

Used without options this produces a list of all the processes owned by you and associated with your terminal.

Options for this command vary with the type of UNIX system that you are using. Consult the **man** page for the **ps** command to check which command options are available on your UNIX.

Understanding process information

The information displayed by the ps command varies according to which command option(s) you use and the type of UNIX that you are using.

These are some of the column headings displayed by the different versions of this command.

```
UID\USER  PID  PPID  SZ  TT(Y)  TIME  COMMAND
```

A process is an executing program identified by a unique PID (process identifier). To see information about your processes, with their associated PID and status, type

```
% ps
```

A process may be in the foreground, in the background, or be suspended. In general the shell does not return the UNIX prompt until the current process has finished executing.

Some processes take a long time to run and hold up the terminal. Backgrounding a long process has the effect that the UNIX prompt is returned immediately, and other tasks can be carried out while the original process continues executing.

kill Killing processes

Each process you start is usually completed within a few seconds. Sometimes a background process or a process without a controlling terminal hangs up and you will need to destroy this process by killing it.

To kill a process use the **kill** command.

kill [-signal] process_identifier (PID)

Use the `ps` command if you need to find the PID of a process.

Always try to kill a process with a simple `kill` command.

kill process_identifier

This is the cleanest way to kill a process and has the same effect as cancelling a process.

If this does not work use the `-1` (one) signal.

kill -1 process_identifier

This tells the process to hangup just as though you were logging out. The system will attempt to kill any child processes.

As a last resort use the `-9` signal:

kill -9 process_identifier

This will stop the process dead in its tracks but it may leave any child processes still running.

Printing

To print a file use the command:

lpr filename

The job is placed in the print queue and you can carry on working.

Some versions of UNIX provide both `lp` and `lpr` command for printing files; other versions support only one or the other. If the `lpr` command is not available on your system use the `lp` command instead.

Using a different printer

To send a file to a different printer use the command:

lpr -Pname_of_printer filename

To find out the names of the printers that you can use enter the command:

```
lpstat -p
```

There may be a local command that gives you this information.

Printing multiple copies

To print multiple copies of a file use the # (hash) option. For example:

```
lpr -#3 memo.txt
```

This prints three copies of the file memo.txt on your default printer.

You can also print multiple copies of several files. For example:

```
lpr -Pps1 -#3 survey.doc letter.txt
```

This prints three copies of the files survey.doc and letter.txt on the printer ps1.

Cancelling a print request

To cancel a print request use the command:

```
lprm job_number
```

This removes the print job from the print queue.

You cannot remove another user's print job from the print queue; only the ones that belong to you.

Foreground and Background Process

Running processes in the foreground

When you enter a command at the shell prompt, your shell forks a child process in which to execute the command.

The parent process (your shell) waits for the command to complete and for its child process to die. Until then you are unable to enter another command. Commands entered in this way are known as foreground processes.

Once the command completes and the child process dies, the parent process is reactivated, the shell prompt returns, and you can enter another command.

Cancelling a foreground process

To cancel the process that is currently running enter the key combination:

```
Ctrl-c
```

Hold down the key marked Ctrl and press the C key. This cancels the current foreground process completely: it no longer exists.

Placing a foreground process in the background

To run a foreground process as a background job:

- 1) Suspend the foreground process.
- 2) Enter the **bg** (background) command to move the process into the background.

For example:

```
lpr -Pps23 part[1-8].ps
```

```
^Z
```

```
Stopped
```

```
bg
```

```
[1] lpr -Pps23 interface.ps &
```

```
jobs
```

```
[1] Running lpr -Pps7 interface.ps
```

This starts a foreground process to print several files. The process is then stopped and the job moved into the background. The state of this job is then checked.

Running processes as background jobs

To run a process in the background as a job and carry on working add an & (ampersand) at the end of the command line. For example:

```
find /usr/man -name *.1 -print > sect1 & [1] 2548
```

The shell forks a child process to run the command and displays the job number ([n]) and the PID (Process IDentifier) number. The shell prompt returns and you can enter further commands.

Redirect the standard output for a command that is being run in the background to a file. This prevents the output from the command appearing on your screen and interrupting your current work.

If the command is likely to produce error messages you will need to redirect standard error. Otherwise all error messages are sent to your screen.

Do not run an interactive command that requires you to type something at the terminal as a background job. If you do, the job will stop and wait for user input. You will then have to kill its process.

Bring a job into the foreground

To bring a specific background job into the foreground:

- 1) Enter the **fg** (foreground) command followed by the job number.

```
fg job_number
```

If there is only one job running in the background just enter the fg command.

For example:

lpr -Pps23 interface.ps &

jobs

```
[1] + Running          xterm -g 90x55
[2] - Running          xterm -g 90x55
[3]  Running          lpr -Ppsa interface.ps
```

fg 3

```
lpr -Pps23 interface.ps
```

This starts a process to print a file as a background job. The user then checks the current active jobs and brings the print job - job number three (3) - into the foreground.

nice command

The nice command enables you to reduce the priority of the process. The command line format is as follows:

```
%nice command line
```

here command line is the command line that you wish to place at a lower priority.

6.4.5 Starting and Shutting down UNIX/LINUX

UNIX BOOT Process till the User Logs on

A computer without a program running is just an inert hunk of electronics. The first thing a computer has to do when it is turned on is start up a special program called an Operating System. The operating system's job is to help other computer programs to work by handling the messy details of controlling the computers hardware.

The process of bringing up the operating system is called booting. Your computer knows how to boot because instructions for booting are built into one of its chips, the BIOS (or Basic Input Output System) chip.

The BIOS chip tells it to look in a fixed place on the lowest-numbered hard disk (the boot disk) for a special program called a boot loader (under Linux the boot loader is LILO). The boot loader is pulled into core and started. The boot loader's job is to start the real operating system.

The loader does this by looking for a kernel, loading it into core, and starting it. When you boot Linux and see LILO on the screen followed by a bunch of dots, it is loading the kernel.

Once the kernel starts, it has to look around, find the rest of the hardware and get ready to run programs. It does this by poking not an ordinary memory locations but rather at I/O ports -special bus addresses that are likely to have device controller cards listening at them for commands. The kernel doesn't poke at random; it has a lot of built in knowledge about what's likely to find where and how controllers will respond if they're present. This process is called autoprobing.

Most of the messages you see at the boot time are the kernel autoprobing your

hardware through the I/O ports, figuring out what it has available to it and adapting itself to your machine. The Linux kernel is extremely good at this, better than most Unices and much better than DOS and Windows. In fact, many Linux old-timers think the cleverness of Linux's boot-time probes was a reason it broke out of the pack of free-Unix experiments to attract a critical mass of users.

But getting the kernel fully loaded and running isn't the end of the boot process it's just the first stage (sometimes called run level 1)

The kernel's next step is to check to make sure your disks are OK. Disk file systems are fragile things. If they've been damaged by a hardware failure or a sudden power outage, there are good reasons to take recovery steps before your Unix is all the way up.

The kernel's next step is to start several daemons. A daemon is a program like a spooler, a mail listener, or a WWW server that lurks in the background, waiting for things to do. These special programs often have to coordinate several requests that could conflict. They are daemons because it's often easier to write one program that runs constantly and knows about all requests than it would be to try to make sure that a flock of copies (each processing one request and all running at the time) don't step on each other. The particular collection of daemons your system starts may vary, but will almost always include a print spooler (a gatekeeper daemon for your printer).

Once all daemons are started, we're at run level 2. The next step is to prepare for users. The kernel starts a copy of a program called `getty` to watch your console (and maybe more copies to watch dial-in serial port). This program is that which issues the login prompt to your console. We're now at run level 3 and ready for you to log in and run programs.

When you log in (give a name and password) you identify yourself to `getty` and the computer. It then runs a program called (naturally enough) `login`, which does some housekeeping things and then starts up a command interpreter, the shell. (Yes, `getty` and `login` could be one program. They're separate for historical reasons not worth going into here).

Shutting down.

Shutting down a Linux system can be tricky. You should never simply turn off the power or press the reset switch. The kernel keeps track of the disk read/write data in memory buffers. If you reboot the system without giving the kernel a chance to write its buffers to disk, you can corrupt the file systems.

Other precautions are taken during shutdown as well. All processes are sent a signal that allows them to die gracefully (by first writing and closing all files, for example). File systems are unmounted for safety. If you wish, the system can also alert users that the system is going down and give them a chance to log off.

The easiest way to shut down is with the `shutdown` command. The format of the command is

```
shut down time warning-message
```

The `time` argument is the time to shut down the system (in the format `hh:mm:ss`), and `warning-message` is a message displayed on all user's terminals before shutdown. Alternately, you can specify the time as "now", to shut down immediately. The `-r` option may be given to `shutdown` to reboot the system after shutting down.

For example, to shut down and reboot the system at 8:00 pm, use the command

```
# shut down -r 20:00
```

The command halt may be used to force an immediate shutdown without any warning messages or grace period. Halt is useful if you're the only one using the system and want to shut down and turn off the machine.

Don't turn off the power or reboot the system until you see the message:

```
The system is halted
```

It is very important that you shut down the system, "cleanly," using the shutdown or halt command. On some systems, pressing Ctrl-Alt-Del will be trapped and cause a shutdown.

6.4.6 UNIX Command Syntax

man command

The command **man** gives you access to an on-line manual containing a complete description of every command available on this system.

man can also provide you with one line descriptions of commands specified by name; or for all commands whose description contains any of a set of keywords.

Self Check Exercises

- 14) Discuss the following commands - (i) rm, (ii) mkdir (iii) cd (iv) cp
- 15) How could you place a foreground process in the background?

Note: i) Write your answer in the space given below
ii) Check your answer with the answers given at the end of this unit.

.....
.....
.....
.....
.....
.....

6.4.7 UNIX File System

What is a File?

A file system is a logical method for organising and storing large amounts of information in a way which makes it easy manage. The file is the smallest unit in which information is stored. The UNIX file system has several important features.

Different types of file

To the user, it appears as though there is only one type of file in UNIX - the file which is used to hold information. In fact, the UNIX filesystem contains several types of file.

- 1 Ordinary files
- 1 Directories
- 1 Special files
- 1 Pipes

Ordinary files

This type of file is used to store your information, such as some text you have written or an image you have drawn. This is the type of file that you usually work with.

Files which you create belong to you - you are said to “own” them - and you can set access permissions to control which other users can have access to them. Any file is always contained within a directory.

Directories

A directory is a file that holds other files and other directories. You can create directories in your home directory to hold files and other sub-directories.

Having your own directory structure gives you a definable place to work from and allows you to structure your information in a way that makes best sense to you.

Directories which you create belong to you - you are said to “own” them - and you can set access permissions to control which other users can have access to the information they contain.

Special files

This type of file is used to represent a real physical device such as a printer, tape drive or terminal.

It may seem unusual to think of a physical device as a file, but it allows you to send the output of a command to a device in the same way that you send it to a file. For example:

```
cat scream.au > /dev/audio
```

This sends the contents of the sound file `scream.au` to the file `/dev/audio` which represents the audio device attached to the system. Guess what sound this makes?

The directory `/dev` contains the special files which are used to represent devices on a UNIX system.

Pipes

UNIX allows you to link commands together using a pipe.

The pipe acts a temporary file which only exists to hold data from one command until it is read by another

Security

In your **unixstuff** directory, type

% **ls -l** (l for long listing!)

You will see that you now get lots of details about the contents of your directory, similar to the example below.

Each file (and directory) has associated access rights, which may be found by typing **ls -l**. Also, **ls -lg** gives additional information as to which group owns the file (beng95 in the following example):

```
-rwxrw-r-- 1 ee51lab beng95 2450 Sept29 11:52 file1
```

In the left-hand column is a 10 symbol 'string' consisting of the symbols **d, r, w, x, -,** and, occasionally, **s** or **S**. If **d** is present, it will be at the left hand end of the string, and indicates a directory: otherwise '-' will be the starting symbol of the string.

The 9 remaining symbols indicate the permissions, or access rights, and are taken as three groups of 3.

- 1) The left group of 3 gives the file permissions for the **user** that owns the file (or directory) (ee51lab in the above example);
- 2) The middle group gives the permissions for the **group** of people to whom the file (or directory) belongs (eebeng95 in the above example);
- 3) The rightmost group gives the permissions for all **others**.

The symbols r, w, etc., have slightly different meanings depending on whether they refer to a simple file or to a directory.

Access rights on files.

- 1) **r** (or -), indicates read permission (or otherwise), that is, the presence or absence of permission to read and copy the file
- 2) **w** (or -), indicates write permission (or otherwise), that is, the permission (or otherwise) to change a file
- 3) **x** (or -), indicates execution permission (or otherwise), that is, the permission to execute a file, where appropriate

Access rights on directories.

- 1) **r** allows users to list files in the directory;
- 2) **w** means that users may delete files from the directory or move files into it;
- 3) **x** means the right to access files in the directory. This implies that you may read files in the directory provided you have read permission on the individual files.

So, in order to read a file, you must have execute permission on the directory containing that file, and hence on any directory containing that directory as a subdirectory, and so on, up the tree.

Some examples

-rwxrwxrwx a file that everyone can read, write and execute (and delete).

-rw—— a file that only the owner can read and write - no-one else can read or write and no-one has execution rights (e.g. your mailbox file).

Changing access rights**chmod (changing a file mode)**

Only the owner of a file can use chmod to change the permissions of a file. The options of chmod are as follows

Symbol	Meaning
u	user
g	group
o	other
a	all
r	read
w	write (and delete)
x	execute (and access directory)
+	add permission
-	take away permission

For example, to remove read write and execute permissions on the file **biglist** for the group and others, type

```
% chmod go-rwx biglist
```

This will leave the user permissions unaffected.

To give read and write permissions on the file **biglist** to all,

```
% chmod a+rw biglist
```

chown command

The chown command is used to change individual ownership of a file. The command syntax is

```
%chown owner file
```

for changing the ownership of file to owner.

Wild Card character

The characters * and ?

The character * is called a wildcard, and will match against none or more character(s) in a file (or directory) name. For example, in your **unixstuff** directory, type

```
% ls list*
```

This will list all files in the current directory starting with “list...”

Try typing

```
% ls *list
```

This will list all files in the current directory ending with “...list”

The character ? will match exactly one character.

So **ls ?ouse** will match files like **house** and **mouse**, but not **grouse**.

Try typing

```
% ls ?list
```

6.4.8 Mountable File System

mount and unmount command

Before a file system is accessible to the system, it must be **mounted** on a directory. For example, if you have a file system on a floppy, you must mount it under a directory like /mnt in order to access the files on the floppy. After mounting the file system, all of the files in the file system appear in that directory. After unmounting the file system, the directory (in this case, /mnt) will be empty.

The same is true of file systems on the hard drive. The system automatically mounts file systems on your hard drive at bootup time. The so-called “root file system” is mounted on the directory /. If you have a separate file system for /usr, it is mounted on /usr. If you only have a root file system, all files (including those in /usr) exist on that file system mount and unmount (not unmount) are used to mount and unmount file systems. The command

```
Mount -av
```

is executed automatically by the file /etc/rc at boot time, or by the file /etc/rc.d/boot on some Linux systems. The file /etc/fstab provides information on file systems and mount points. An example /etc/fstab file is

```
# device      directory      type      options
/dev/had2     /              ext2      defaults
/dev/had3     /usr           ext2      defaults
/dev/had4     None           swap      sw
/proc         /proc          proc      none
```

The first field, device, is the name of the partition to mount. The second field is the mount point. The third field is the file system type, like ext2 (for ext2fs) or minix (for Minix file systems). The Table below lists the various file system types that are mountable under Linux. Not all of these file system types may be available on your system, because the kernel must have support for them compiled in.

Table : Linux File System Types

File system	Type name	Comment
Second Extended File system	Ext2	Most common Linux file system.
Extended File system	Ext	Superseded
Minix File system	minix	Original Minix file system; rarely used.
Xia File system	Xia	Like ext2, but rarely used.
UMSDOS File system	umsdos	Used to install Linux on an MS:DOS partition.
MS-DOS File system	msdos	Used to access MS-DOS files.
/proc File system	proc	Provides process information for ps etc.
ISO 9660 File system	Iso9660	Format used by most CD-ROMs.
Xenix File system	xenix	Used to access file from Xenix.
System V File system	sysv	Used to access file from System V variants for the x86
Coherent File system	coherent	Used to access files from Coherent.
HPFS File system	hpfs	Read-only access for HPFS partitions (DoubleSpace)

The last field of the fstab file are the mount options. This is normally set to defaults. Swap partitions are included in the /etc/fstab file. They have a mount directory of none, and type swap. The swapon -a command, which is executed from /etc/rc or /etc/init.d/boot, is used to enable swapping on all of the swap devices that are listed in /etc/fstab.

The /etc/fstab file contains one special entry for the /proc file system. The /proc file system is used to store information about system processes, available memory, and so on. If /proc is not mounted, commands like ps will not work.

The mount command may be used only by root. This ensures security on the system. You wouldn't want regular users mounting and unmounting file systems on a whim. Several software packages are available which allow non-root users to mount and unmount file systems, especially floppies, without compromising system security.

The mount -av command actually mounts all of the file systems other than the root file system (in the table above, /dev/hda2). The root file system is automatically

mounted at boot time by the kernel.

Instead of using `mount -av`, you can mount a file system by hand. The command

```
# mount -t ext2 /dev/hda3 /usr
```

is equivalent to mounting the file system with the entry for `/dev/hda3` in the example `/etc/fstab` file, above.

UNIX and Standard Devices

Many Linux commands get input from what is called **standard input** and send their output to **standard output** (often abbreviated as `stdin` and `stdout`). Your shell sets things up so that standard input is your keyboard, and standard output is the screen.

Here's an example using the `cat` command. Normally, `cat` reads data from all of the files specified by the command line, and sends this data directly to `stdout`. Therefore, using the command

```
//home/larry/papers# cat history-final masters-thesis
```

displays the contents of the file `history-final` followed by `masters-thesis`.

However, if you don't specify a filename, `cat` reads data from `stdin` and sends it back to `stdout`. Here's an example:

```
/home/larry/papers# cat
```

```
Hello there.
```

```
Hello there.
```

```
Bye.
```

```
Bye.
```

```
Ctrl-D
```

```
/home/larry/Papers#
```

Each line that you type is immediately echoed back by `cat`. When reading from standard input, you indicate the input is "finished" by sending an EOT (end-of-text) signal, in general, generated by pressing `Ctrl-D`.

Here's another example. The `sort` command reads lines of text (again, from `stdin`, unless you specify one or more filenames) and sends the sorted output to `stdout`. Try the following.

```
/home/larry/Papers# sort
```

```
bananas
```

```
carrots
```

```
apples
```

```
Ctrl-D
```

```

apples

bananas

carrots

/home/larry/Papers#

```

Now we can alphabetize our shopping list... isn't Linux useful?

Redirecting input and output.

Now, let's say that you want to send the output of sort to a file, to save our shopping list on disk. The shell lets you redirect standard output to a filename, by using the ">" symbol. Here's how it works:

```

/home/ larry/ papers* sort > shopping-list

bananas

carrots

apples

```

Ctrl-D

```

/ home/ larry / papers*

```

As you can see, the result of the sort command isn't displayed, but is saved to the file named shopping-list. Let's look at this file:

```

/home / larry / papers* cat shopping-list

apples

bananas

carrots

/home / larry / papers*

```

Now you can sort your shopping list, and save it, too! But let's suppose that you are storing the unsorted, original shopping list in the file items. One way of sorting the information and saving it to a file would be to give sort the name of the file to read, in lieu of standard input, and redirect standard output as we did above, as follows:

```

/home / larry / papers* sort items > shopping-list

/home / larry / papers* cat shopping-list

apples

bananas

carrots

/ home / larry / papers*

```

However, there's another way to do this. Not only can you redirect standard output, you can redirect standard input as well, using the "<" symbol.

```
/ home / larry / papers* sort < items
```

```
apples
```

```
bananas
```

```
carrots
```

```
/ home / larry / papers*
```

Technically, `sort < items` is equivalent to `sort items`, but lets you demonstrate the following point: `sort < items` behaves as if the data in the file `items` was typed to standard input. The shell handles the redirection. `sort` wasn't given the name of the file (`items`) to

read; as far as `sort` is concerned, it still reads from standard input as if you had typed the data from your keyboard.

This introduces the concept of a **filter**. A filter is a program that reads data from standard input, processes it in some way, and sends the processed data to standard output. Using redirection, standard input and standard output can be referenced from files. As mentioned above, `stdin` and `stdout` default to the keyboard and screen respectively. `sort` is a simple filter. It sorts the incoming data and sends the result to standard output. `cat` is even simpler. It doesn't do anything with the incoming data, it simply outputs whatever is given to it.

Using pipes.

We already demonstrated how to use `sort` as a filter. However, these examples assume that you have data stored in a file somewhere or are willing to type the data from the standard input yourself. What if the data that you wanted to sort came from the output of another command, like `ls`?

The `-r` option to `sort`, sorts the data in reverse-alphabetical order. If you want to list the files in your current directory in reverse order, one way to do it is as follows:

```
/ home / larry / papers* ls
```

```
english-list
```

```
history-final
```

```
masters-thesis
```

```
notes
```

Now redirect the output of the `ls` command into a file called `file-list`:

```
/ home / larry / papers* ls > file-list
```

```
/ home / larry / papers* sort -r file-list
```

```
notes
```

```
masters-thesis
```



```
history-final
```

```
english-list
```

```
/ home / larry / papers*
```

Here, you save the output of `ls` in a file, and then run `sort -r` on that file. But this is unwieldy and uses a temporary file to save the data from `ls`.

The solution is **pipelining**. This is a shell feature that connects a string of commands via a “pipe.” The stdout of the first command is sent to the stdin of the second command. In this case, we want to send the stdout of `ls` to the stdin of `sort`. Use the “|” symbol to create a pipe, as follows:

```
/ home / larry / papers* ls | sort -r
```

```
notes
```

```
masters-thesis
```

```
history-fional
```

```
english-list
```

```
/home / larry / papers*
```

This command is shorter and easier to type.

Here’s another useful example, the command

```
/home / larry / papers* ls / usr / bin
```

displays a long list of files, most of which fly past the screen too quickly for you to read. So, let’s use `more` to display the list of files in `/usr/bin`.

```
/home/larry/papers# ls /usr/bin | more
```

Now you can page down the list of files at your leisure.

But the fun doesn’t stop here! You can pipe more than two commands together. The command `head` is a filter that displays the first lines from an input stream (in this case, input from a pipe). If you want to display the last filename in alphabetical order in the current directory, use commands like the following:

```
/home/larry/papers# ls | sort -r | head -1
```

```
notes
```

```
/home/larry/papers#
```

where `head -1` displays the first line of input that it receives (in this case, the stream of reverse-sorted data from `ls`).

Non-destructive redirection of output.

Using “>” to redirect output to a file is destructive: in other words, the command

```
/ home / larry / papers* ls > file-list
```

overwrites the contents of the file file-list. If instead, you redirect with the symbol “>>”, the output is appended to (added to the end of) the named file instead of overwriting it. For example,

```
/ home / larry / papers* ls >> file-list
```

appends the output of the ls command to file-list.

Keep in mind that redirection and pipes are features of the shell—which supports the use of “>”, “>>” and “|”. It has nothing to do with the commands themselves.

Self Check Exercises

16) Discuss the various types of files in a UNIX file system?

17) Describe the various access rights on files and directories?

18) What is a filter?

Note: i) Write your answer in the space given below.

ii) Check your answer with the answers given at the end of this unit.

.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....

6.4.9 Shell Scripts

Let’s say that you use a series of commands often and would like to save time by grouping all of them together into a single “command”. For example, the three commands

```
/home/larry# cat chapter1 chapter2 chapter3 > book  
  
/home/larry# wc -l book  
  
/home/larry# lp book
```

concatenates the files chapter1, chapter2, and chapter3 and places the result in the file book. The second command displays a count of the number of lines in book, and the third command lp book prints book.

Rather than type all these commands, you can group them into a shell script. The shell script used to run all these commands might look like this:

```
#!/bin/sh  
  
# A shell script to create and print the book
```

```
cat chapter1 chapter2 chapter3 > book

wc -l book

lp book
```

Shell scripts are just plain text files; you can create them with an editor such as `emacs` or `vi`.

Let's look at this shell script. The first line, `#!/bin/sh`, identifies the file as a shell script and tells the shell how to execute the script. It instructs the shell to pass the script to `/bin/sh` for execution, where `/bin/sh` is the shell program itself. Why is this important? On most Linux systems, `/bin/sh` is a Bourne-type shell, like `bash`. By forcing the shell script to run using `/bin/sh`, you ensure that the script will run under a Bourne-syntax shell (rather than a C shell). This will cause your script to run using the Bourne syntax even if you use `tcsh` (or another C shell) as your login shell.

The second line is a comment. Comments begin with the character `#` and continue to the end of the line. Comments are ignored by the shell—they are commonly used to identify the shell script to the programmer and make the script easier to understand.

The rest of the lines in the script are just commands, as you would type them to the shell directly. In effect, the shell reads each line of the script and runs that line as if you had typed it at the shell prompt.

Permissions are important for shell scripts. If you create a shell script, make sure that you have execute permission on the script in order to run it. When you create text files, the default permissions usually don't include execute permission, and you must set them explicitly. See the discussion of file permissions on page for details. Briefly, if this script were saved in the file called `makebook`, you could use the command

```
/home/larry# chmod u+x makebook
```

to give yourself execute permission for the shell script `makebook`.

You can use the command

```
/home/larry# makebook
```

to run all the commands in the script.

Shell variables and the environment

A shell lets you define **variables**, as do most programming languages. A variable is just a piece of data that is given a name.

`tcsh`, as well as other C-type shells, use a different mechanism for setting variables than is described here. This discussion assumes the use of a Bourne shell like `bash`. See the `tcsh` manual page for details.

When you assign a value to a variable (using the `=` operator), you can access the variable by prepending a `$` to the variable name, as demonstrated below.

```
larry# foo="hello there"
```

The variable `foo` is given the value `hello there`. You can then refer to this value by the variable name prefixed with a “\$” character. For example, the command

```
/home/larry# echo $foo
```

```
hello there
```

```
/home/larry#
```

produces the same results as

```
/home/larry# echo "hello there"
```

```
hello there
```

```
/home/larry#
```

These variables are internal to the shell, which means that only the shell can access them. This can be useful in shell scripts; if you need to keep track of a filename, for example, you can store it in a variable, as above. Using the `set` command displays a list of all defined shell variables.

However, the shell lets you **export** variables to the **environment**. The environment is the set of variables that are accessible by all commands that you execute. Once you define a variable inside the shell, exporting it makes the variable part of the environment as well. Use the `export` command to export a variable to the environment.

Again, here we differ between `bash` and `tcsh`. If you use `tcsh`, another syntax is used for setting environment variables (the `setenv` command is used). See the `tcsh` manual page for more information.

The environment is very important to the UNIX system. It lets you configure certain commands just by setting variables which the commands know about.

Here’s a quick example. The environment variable `PAGER` is used by the `man` command and it specifies the command to use to display manual pages one screenful at a time. If you set `PAGER` to the name of a command, it uses that command to display the man pages, instead of `more` (which is the default).

Set `PAGER` to “`cat`”. This causes output from `man` to be displayed to the screen all at once, without pausing between pages.

```
/home/larry# PAGER=cat
```

Now, export `PAGER` to the environment.

```
/home/larry# export PAGER
```

Try the command `man ls`. The man page should fly past your screen without pausing for you.

Now, if we set `PAGER` to “`more`”, the `more` command is used to display the man page.

```
/home/larry# PAGER=more
```

Note that we don’t have to use the `export` command after we change the value of `PAGER`. We only need to export a variable once; any changes made to it thereafter will automatically be propagated to the environment.

It is often necessary to quote strings in order to prevent the shell from treating various characters as special. For example, you need to quote a string in order to prevent the shell from interpreting the special meaning of characters such as “*”, “?” or a space. There are many other characters that may need to be protected from interpretation.

The manual pages for a particular command tell you if the command uses any environment variables. For example, the man page explains that PAGER is used to specify the pager command.

Some commands share environment variables. For example, many commands use the EDITOR environment variable to specify the default editor to use when one is needed.

The environment is also used to keep track of important information about your login session. An example is the HOME environment variable, which contains the name of your home directory.

```
/home/larry/papers# echo $HOME
/home/larry
```

Another interesting environment variable is PS1, which defines the main shell prompt. For example,

```
/home/larry PS1="Your command, please: "
```

To set the prompt back (which contains the current working directory followed by a “#” symbol),

```
Your command, please: PS1="\w# _ "
/home/larry#
```

The bash manual page describes the syntax used for setting the prompt.

The PATH environment variable

When you use the ls command, how does the shell find the ls executable itself? In fact, ls is in /bin on most systems. The shell uses the environment variable PATH to locate executable files for commands you type.

For example, your PATH variable may be set to

```
/bin:/usr/bin:/usr/local/bin:.
```

This is a list of directories for the shell to search, each directory separated by a “:”. When you use the command ls, the shell first looks for /bin/ls, then /usr/bin/ls, and so on.

Note that the PATH has nothing to do with finding regular files. For example, if you use the command

```
/home/larry# cp foo bar
```

the shell does not use PATH to locate the files foo and bar—those filenames are assumed to be complete. The shell only uses PATH to locate the cp executable.

This saves you time, and means that you don’t have to remember where all the

command executables are stored. On many systems, executables are scattered about in many places, such as /usr/bin, /bin, or /usr/local/bin. Rather than give the command's full pathname (such as /usr/bin/cp), you can set PATH to the list of directories that you want the shell to automatically search.

Notice that PATH contains ".", which is the current working directory. This lets you create a shell script or program and run it as a command from your current directory without having to specify it directly (as in ./makebook). If a directory isn't in your PATH, then the shell will not search it for commands to run; this also includes the current directory.

6.4.10 USER Management

Super User -The root account.

Linux differentiates between different users. What they can do to each other and the system is regulated. File permissions are arranged so that normal users can't delete or modify files in directories like /bin and /usr/bin. Most users protect their own files with the appropriate permissions so that other users can't access or modify them. (One wouldn't want anybody to be able to read one's love letters.) Each user is given an account that includes a user name and home directory. In addition, there are special, system defined accounts which have special privileges. The most important of these is the root account, which is used by the system administrator. By convention, the system administrator is the user, root.

There are no restrictions on root. He or she can read, modify, or delete any file on the system, change permissions and ownerships on any file, and run special programs like those which partition a hard drive or create file systems. The basic idea is that a person who cares for the system logs in as root to perform tasks that cannot be executed as a normal user. Because root can do anything, it is easy to make mistakes that have catastrophic consequences.

If a normal user tries inadvertently to delete all of the files in /etc, the system will not permit him or her to do so. However, if root tries to do the same thing, the system doesn't complain at all. It is very easy to trash a Linux system when using root. The best way to prevent accidents is:

- 1 Sit on your hands before you press Enter for any command that is non-reversible. If you're about to clean out a directory, re-read the entire command to make sure that it is correct.
- 1 Use a different prompt for the root account. root's .bashrc or .login file should set the shell prompt to something different than the standard user prompt. Many people reserve the character "#" in prompts for root and use the prompt character "\$" for everyone else.
- 1 Log in as root only when absolutely necessary. When you have finished your work as root, log out. The less you use the root account, the less likely you are to damage the system. You are less likely to confuse the privileges of root with those of a normal user.

Picture the root account as a special, magic hat that gives you lots of power, with which you can, by waving your hands, destroy entire cities. It is a good idea to be a bit careful about what you do with your hands. Because it is easy to wave your hands in a destructive manner, it is not a good idea to wear the magic hat when it is not needed, despite the wonderful feeling.

Adding and Removing User

The system keeps track of the following information about each user:

user name	This identifier is unique for every user. Example user name are larry, karl, and mdw. Letters and digits may be used, as well as "-" and "." (period). User names are usually limited to 8 characters in length.
user ID	This number, abbreviated UID, is unique for every user. The system generally keeps track of users by UID, not user name.
group ID	This number, abbreviated GID, is the user's default group. Each user belongs to one or more groups as defined by the system administrator.
password	This is the user's encrypted password. The passwd command is used to set and change user passwords.
full name	The user's "real name," or "full name," is stored along with the user name. For example, the user schmoj may be "Joe Schrno" in real life.
home directory	This is the directory the user is initially placed in at login, and where his or her personal files are stored. Every user is given a home directory, which is commonly located under /home.
login shell	The shell that is started for the user at login. Examples are /bin/bash and /bin/tcsh.

This information is stored in the file /etc/passwd. Each line in the file has the format

```
User name : encrypted password:UID:GID:full name:home
Directory:login shell
```

An example might be

```
Kiwi:xvSQ981g71okk:102:10:Laura
```

```
Poole:/home/kiwi:/bin/bash
```

In this example, the first field, "kiwi," is the user name.

The next field, "Xv8Q981g71oKK", is the encrypted password. Passwords are not stored on the system in human-readable format. The password is encrypted using itself as the secret key. In other words, one must know the password in order to decrypt it. This form of encryption is reasonably secure.

Some systems use “shadow passwords,” in which password information is stored in the file `/etc/shadow`. Because `/etc/passwd` is world-readable, `/etc/shadow` provides some degree of extra security because its access permissions are much more restricted. Shadow passwords also provide other features, like password expiration.

The third field, “102”, is the UID. This must be unique for each user. The fourth field, “100”, is the GID. This user belongs to the group numbered 100. Group information is stored in the file `/etc/group`.

The fifth field is the user’s full name, “Laura Poole”. The last two fields are the user’s home directory (`/home/kiwi`), and login shell (`/bin/bash`), respectively. It is not required that the user’s home directory be given the same name as the user name. It simply helps identify the directory.

Adding users.

When adding users, several steps must be taken. First, the user is given an entry in `/etc/passwd`, with a unique user name and UID. The GID, full name, and other information must be specified. The user’s home directory must be created, and the permissions on the directory set so that the user owns the directory. Shell initialization files must be installed in the home directory, and other files must be configured system-wide (for example, a spool for the user’s incoming e-mail).

It is not difficult to add users by hand, but when you are running a system with many users, it is easy to forget something. The easiest way to add users is to use an interactive program which updates all of the system files automatically. The name of this program is `useradd` or `adduser`, depending on what software is installed.

The `adduser` command takes its information from the file `/etc/adduser.conf`, which defines a standard, default configuration for all new users.

A typical `/etc/adduser.conf` file is shown below.

```
# /etc/adduser.conf:  'adduse'  configuration.

# /See  adduser  (8)  and  adduser,  conf(5)  for  full
documentation.

#  The  DSHELL  variable  specifies  the  default  login
shell  on  your

#  system.

DSHELL=/bin/bash

#  The  DHOME  variable  specifies  the  directory
containing  users'  home

#  directories.

DHOME=/home

#  If  GROUPS  HOMES  is  "yes",  then  the  home  directories
will  be  created  as
```


Software

```
# /home/groupname/user.

GROUPHOMES=no

# If LETTERHOMES is "yes", then the created home
directories will have

# an extra directory - the first letter of the user
name. For example:

# /home/u/user.

LETTERHOMES=no

# The SKEL variable specifies the directory containing
"sheletal" user

# files: in other workds, files such as a sample.
Profile that will be

#copied to the new user's home directory when it
is created.

SKEL=/etc/skel

# FIRST_SYSTEM_UID to LAST_SYSTEM_UID inclusive is
the range for UIDs

# for dynamically allocated administrative and system
accounts.

FIRST_SYSTEM_UID=100

LAST_SYSTEM_UID=999

# FIRST_UID to LAST_UID inclusive is the range of
UIDs of dynamically

# allocated user accounts.

FIRST_UID=100

LAST_UID=29999

# The USERGROUPS variable can be either "yes" or
"no". If "yes" each

# created user will be given their own group to use
as a default, and

# their home directories will be gts. If "no", each
created user will

# be placed in the group whose gid is USERS_GID (see
below).

USERGROUPS=yes

# If USERGROUPS is "no", then USERS_GID should be
the GID of the group
```

```
# 'users' (for the equivalent group) on your system.
USERS_GID=100
```

In addition to defining preset variables that the `adduser` command uses, `/etc/adduser.conf` also specifies where default system configuration files for each user are located. In this example, they are located in the directory `/etc/skel`, as defined by the `SKEL=` line, above. Files which are placed in this directory, like a system-wide, default `.profile`, `.tcshrc`, or `.bashrc` file, will be automatically installed in a new user's home directory by the `adduser` command.

Deleting users.

Deleting users can be accomplished with the commands `userdel` or `deluser`, depending on the software installed on the system.

If you'd like to temporarily "disable" a user from logging in to the system without deleting his or her account, simply prepend an asterisk ("`*`") to the password field in `/etc/passwd`. For example, changing kiwi's `/etc/passwd` entry to

```
kiwi:*Xv8Q98lg71okk:102:100:Laura
```

```
Poole:/home/kiwi:/bin/bash
```

prevents kiwi from logging in.

Setting user attributes.

After you have created a user, you may need to change attributes for that user, like the home directory or password. The easiest way to do this is to change the values directly in `/etc/passwd`. To set a user's password, use `passwd`. The command

```
# passwd larry
```

will change larry's password. Only root may change other users' passwords in this manner. Users can change their own passwords, however.

On some systems, the commands `chfn` and `chsh` allow users to set their own full name and login shell attributes. If not, the system administrator must change these attributes for them.

Groups.

As mentioned above, each user belongs to one or more groups. The only real importance of group relationships pertains to file permissions. As you'll recall from Section 3.10, each file has a "group ownership" and a set of group permissions which defines how users in that group may access the file.

There are several system-defined groups, like `bin`, `mail`, and `sys`. Users should not belong to any of these groups; they are used for system file permissions. Instead, users should belong to an individual group like `users`. You can also maintain several groups for users, like `student`, `staff`, and `faculty`.

The file `/etc/group` contains information about groups. The format of each line is

```
group name:Password:GID: other member
```

Some example groups might be:

```
root:*;0;

users;*;100;mdw;larry

guest;*;200;

other;*;250;kiwi
```

The first group, root, is a special system group reserved for the root account. The next group, users, is for regular users. It has a GID of 100. The users mdw and larry are given access to this group. Remember that in `/etc/passwd` each user was given a default GID. However, users may belong to more than one group, by adding their user names to other group lines in `/etc/group`. The `groups` command lists what groups you are given access to.

The third group, guest, is for guest users, and other is for “other” users. The user kiwi is given access to this group as well.

The “password” field of `/etc/group` is sometimes used to set a password on group access. This is seldom necessary. To protect users from changing into privileged groups (with the `newgroup` command), set the password field to “*”.

The commands `addgroup` or `groupadd` may be used to add groups to your system. Usually, it’s easier just to add entries in `/etc/group` yourself, as no other configuration needs to be done to add a group. To delete a group, simply delete its entry in `/etc/group`

`/etc/passwd`

The password file is the database that describes the accounts on your UNIX system. This file has the pathname `/etc/passwd` and is readable by all the system users, but usually only the superuser can change it. The entry for each account consists of a single line that has several (usually seven) colon-separated fields.

Generally, the fields contain the following information:

- 1 The user name, or login name, consisting of up to 8 characters
- 1 The password is present in encrypted form, which consists of 13 characters from the character set (`.`), (`/`), 0 to 9, A to Z and a to z. If this field is blank no password is requested when signing on to the account. Only the `passwd` command should be used to install a password, although you can remove a password completely with a editor.
- 1 The user ID number (UID), a unique number that is assigned to the account. This number ranges from 0 (reserved for root) to 65,535.
- 1 The group ID number (GID), a number shared by accounts belonging to the same group. This number ranges from 1 to 65,535.
- 1 A comment field. The usage of this field is implementation-dependent, but usually is used for a comment or identifying information. Frequently, the full name of the account user is placed here.
- 1 The home directory, which is the initial working directory location after signing on.

- 1 The login program pathname. This is the program executed by login after signing on. Usually a general-purpose shell, such as the Bourne or C Shell, is specified here. When no login program is specified, a program, usually the Bourne Shell, is used. If a UNIX utility is specified after signing on, the utility is executed and when the program is finished, the user is logged out. A few common examples for such utilities would be who, sync, ps or an editor.

A typical password file entry is depicted here:

Encrypted Password	GID	Home Directory
┌──────────┐	┌┐	┌──────────┐
Becca:k8xOEltQsk6jc:103:10:Becca Thomas:/usr/becca:/bin/csh		
└──┘	└┘	└──────────┘
User Name	UID	Log in Program
	┌──────────┐	
	Comment	

Shadow password

Shadow passwords are means of keeping your encrypted password information secret from normal users. Normally, these encrypted passwords are stored in etc/passwd file for all to read. Anyone can then run password guesser programs on them and attempt to determine what they are. Shadow passwords, by contrast, are saved in etc/shadow, which only privileged users can read. In order to use shadow passwords you need to make sure all your utilities that need access to password information are recompiled to support them.

User Information

who - Determine Who Is On the System:

%who [option...] [who-file]

am I display the name under which you are logged in

In a multuser environment, the more users the system, the slower the system's response to your commands. You can use the **who** command to get an idea of how heavily the system is loaded. It's also a good idea to use this command before employing the write program in order to verify that the persons you intend to write to are currently logged in to the system.

If you do not specify the who-file, the file named /etc/utmp will be displayed by default. This file lists all the users who are currently logged on, the time they logged on, and the terminal or terminals they are using.

who am I

The system will respond with your user name, terminal name and time of login

```
root          console          Nov 7 08:34
Username     tty01             Nov 7 13:57
```

Backup files

Running programs and Scheduling

at -Execute Commands at a Later Time:

```
%at time [month monthday][day of week][week][file]
```

The `at` command is used for scheduling one or more commands to be run at a specified time and date. This command is not supplied with some UNIX systems because of security problems.

You may use `at` to schedule execution of one or more UNIX commands anytime from the next minute to the next year. You may either type a desired command as you invoke `at`, or specify a file that contains the commands. The `at` utility places your commands together with information necessary for establishing the appropriate environment in a memo file in the `/usr/spool/at` directory. Periodically another program, `atrun`, is invoked by your UNIX system to examine `/usr/spool/at`; `atrun` will execute any memo file that is scheduled to run. Generally the memo files are examined several times each hour. If you schedule a file to be executed at some intermediate time, the system waits until the regularly scheduled execution of the `atrun` program before executing your commands. You may determine how often your system invokes `atrun` to execute memo files by examining the system file `/usr/lib/crontab`.

crontab command

Maintain crontab files for individual users. A crontab file contains instructions to the cron daemon of the general form: “run this command at this time on this date”. Each user has own crontab.

6.4.11 Other System Activity**fdisk**

To modify the partitions we use the program `fdisk`. For example, let’s say that you have an 80 mb hard drive dedicated to MS-DOS. You’d like to split it in half - 40 mb for MS-DOS and 40 mb for Linux. In order to do this, you run `FDISK` under MS-DOS, delete the 80 mb MS-DOS partition, and recreate a 40 mb MS-DOS partition. You can then format the new partition and reinstall your MS-DOS software from backups. 40 mb of the drive is left empty. Later, you create Linux partitions on the unused portion of the drive.

fsck

The file system check utility is an interactive program that will prompt you for yes or no decisions on whether or not to correct certain error conditions. It is a good idea to run `fsck` one time through, answering no to all queries, in order to assess the scope of any problems before `fsck` repairs any errors. One command line for invoking `fsck` is

```
% /etc/fsck [-y] [-n] [filesystem]
```

Here `filesystem` is the name of the file system to be checked. In general you don’t have to supply this argument since `fsck` will obtain the name(s) of the file system(s) to be checked from the ordinary file `/etc/checklist` or `/etc/fstab`

The program will automatically answer “no” to all queries if you invoke `fsck` with the `-n` option. One reason for this conservative approach would be that if there was a problem with the hardware, and not with the file system, you wouldn’t want `fsck` to repair the file system.

mkfss

df

The df command reports on the space left on the file system. For example, to find out how much space is left on the fileserver, type

% df .

du

The du command outputs the number of kilobytes used by each subdirectory. Useful if you have gone over quota and you want to find out which directory has the most files. In your home-directory, type

% du

Run level

Self Check Exercises

- 19) What is a shell script?
- 20) What information does system keep about each user?
- 21) Describe file system check utility?

Note: i) Write your answer in the space given below.
ii) Check your answer with the answers given at the end of this unit.

.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....

6.5 LINUX

6.5.1 Technical Features of Linux

The benefits derived from using the UNIX operating system, and hence Linux, stem from its power and flexibility. These are the result of the many features built into the system, ready for use as soon as you turn the system on. The various features available are described below:

Linux is Network-friendly

A team of programmers developed Linux over the Internet, therefore its networking features are given high priority. Linux is capable of acting as client

and/or server to any of the popular operating systems in use today, and is quite capable of being used to run Internet Service Providers. It is also well suited to serving as a dial-in network station. Information exchanged by users on the same systems is called internal communication and is accomplished by terminal-to-terminal communication, e-mail and an automatic calendar that serves as a scheduler or Information manager. Linux provides all this networking capabilities. By using Linux operating system you can talk to another user through an e-mail utility or download large files from another system across the nation. You can also accomplish internal messaging or file transfer through a number of Linux commands, such as write, call, mail and mailx. Linux supports most of the major protocols. Support for Internet, Novell, Windows, and Appletalk networking have been part of the Linux kernel. Linux support Simple Network Management Protocol and other services (such as Domain Name Service) and is also well suited to serving large networks. All these networking options will run acceptably on minimal hardware configurations.

Linux is Multitasking

Multitasking describes the capability to seemingly execute multiple programs at the same time without hindering each application from running. Most UNIX variants provide a type of multitasking called Preemptive multitasking. The name is based on the fact that each program is guaranteed a chance to run, each program executes until the operating system preempts it to allow another program to run. This type of multitasking is exactly what Linux does.

Linux is Multi-user

Linux is an implementation of the UNIX design concept, which means that it is a multi-user system from the word “go.” UNIX and Windows NT helped in developing the concept of many user-accessing applications or processing power from a single PC. The important feature of Linux is that more than one person can work in the same version of the same application at the same time, from the same or separate terminals, Linux is capable of managing the traffic.

Linux is Open

Linux is open, that means that for the entire base system, which includes the kernel, the GNU tools, and all the basic utilities, all the programmers and users have access to the source code as well as the right to modify it. An important aspect of open software is the ability to write kernel extensions and drivers as needed. Open system provides a large number of software authors and beta testers. This makes the software testing and refinement process faster and better. Linux provide the high quality software, which means less prone to crash, more efficient.

Linux is “Free”

Linux is “Free” in two ways. That is, the Linux consumer is free to modify the system and do anything he or she wishes with it and also, acquiring Linux does not necessarily require any cash. One of the greatest advantages of Linux is that it is the freedom to modify the existing code makes it a very easily extensible platform.

Linux is Reliable

Linux is one of the more stable operating systems because programmers who were writing for other programmers and not for the corporate system wrote the large part of the Linux. The only people who made the decisions on what went

structure. The large branches contain more branches, and the branches at the end contain the tree's leaves or normal files.

Types of files

Most files are just files, called regular files, they contain normal data, for example text files, executable files or programs, input for or output from a program. We suppose that everything we encounter on a Linux system is a file, there are some exceptions:

Directories: files that are lists of other files.

Special files: the mechanism used for input and output. Most special files are in /dev.

Links: a system to make a file or directory visible in multiple parts of the system's file tree.

(Domain) sockets: a special file type, similar to TCP/IP sockets, providing inter-process networking protected by the file system's access control.

Named pipes: act more or less like sockets and form a way for processes to communicate with each other, without using network socket semantics.

Table: File types

Symbol	Meaning
-	Regular File
d	Directory
l	Link
c	Special File
s	Socket
p	Named Pipe

The tree of the file system starts at the trunk or slash, indicated by a forward slash (/). This directory, contains all underlying directories and files, is also called the root directory or "the root" of the file system. Directories that are only one level below the root directory are often preceded by a slash, to indicate their position and prevent confusion with other directories that could have the same name. When starting with a new system, it is always a good idea to take a look in the root directory.

Table: Subdirectories of the root directory

Directory	Content
/bin	Common program, shared by the system, the system administrator and the users.
/boot	The startup files and the kernel, vmlinuz. In recent distributions also grub data. Grub is the Grand Unified Boot loader and is an attempt to get rid of the many different boot-loaders we know today.
/dev	Contains references to all the CPU peripheral hardware, which are represented as files with special properties.

/etc	Most important system configuration files are in /etc, this directory contains data similar to those in the Control Panel in Windows.
/home	Home directories of the common users.
/initrd	(on some distributions) Information for booting. Do not remove!
/lib	Library files, includes files for all kind of programs needed by the system and the users.
/lost+found	Every partition has a lost+found in its upper directory. Files that were saved during failures are here.
/misc	For miscellaneous purposes.
/mnt	Standard mount point for external file system, e.g. a CD-ROM or a digital camera.
/net	Standard mount point for entire remote file systems.
/opt	Typically contains extra and third party software.
/proc	A virtual file system containing information about system resources. More information about the meaning of the files in proc is obtained by entering the command man proc in a terminal window. The file proc.txt discusses the virtual file system in detail.
/root	The administrative user's home directory. Mind the difference between /, the root directory and /root, the home directory of the root user.
/sbin	Programs for use by the system and the system administrator.
/tmp	Temporary space for use by the system.
/usr	Programs, libraries, documentation etc. for all user-related programs.
/var	Storage for all variable files and temporary files created by users, such as long files, the mail queue, the print spooler area, space for temporary storage of files downloaded from the Internet, or to keep an image of a CD before burning it.

6.5.3 Components of a Linux System

The Linux system is composed of three main bodies of code, in line with most traditional UNIX implementations:

- 1) **Kernel:** the kernel is responsible for maintaining all the important abstractions of the operating system, including such things as virtual memory and processes.
- 2) **System Libraries:** the system libraries define a standard set of functions through which applications can interact with the kernel and that implementation much of the operating system functionally that does not need the full privileges of the kernel code.
- 3) **System utilities:** the system utilities are programs that perform individual, specialized management tasks. Some system utilities may be invoked just once to initialize and configure some aspect of the system, others known as daemons in UNIX terminology may run permanently, handling tasks such as responding to incoming network connections, accepting logon requests from terminals or updating log files.

System Management Programs	User Processes	User utility programs	Compilers
System Shared Libraries			
Linux Kernel			
Loadable Kernel Modules			

Fig. 3: Components of the Linux System

The above figure illustrates the various components of the Linux system. The important distinction here is between the Kernel and everything else. All the kernel code executes in the processor’s privileged mode with full access to all the physical resources of the computer. Linux refers to this privileged mode as **Kernel Mode**. Under Linux, no user-mode code is built into the kernel. Any operating-system-support code that does not need to run in kernel mode is placed into the system libraries.

Self Check Exercises

- 24) What are the basic components of Linux operating system?
- 25) What are the different types of files provided by Linux?

Note: i) Write your answer in the space given below.
ii) Check your answer with the answer given at the end of this unit.
.....
.....
.....
.....
.....
.....
.....
.....

6.5.4 Linux Kernel and its Modules

The kernel

The kernel is the heart of the system. It manages the communication between the underlying hardware and the peripherals. The kernel also makes sure that processes and server processes are started and stopped the kernel is the most important file on the system. The Linux kernel has the ability to load and unload arbitrary sections of kernel code on demand. These loadable kernel modules run in privileged kernel mode and have full access to all the hardware capabilities of the machine on which they run. Kernel modules are convenient for several reasons. Linux’s source code is free, so anybody wanting to write kernel code is able to compile a modified kernel and to reboot to load that new functionality, however recompiling, relinking and reloading the entire kernel is cumbersome procedure, when you are developing a new driver. If you use kernel modules, you do not have to make a new kernel to test a new driver, the driver may be compiled on its own and loaded into the already-running kernel.

Kernel Modules allows a Linux system to be set up with a standard, minimal kernel without any extra device drivers built in. Any device drivers that the user

needs can be either loaded explicitly by the system at startup or loaded automatically by the system on demand and unloaded when not in use. For Example, a CD-ROM driver might be loaded when a CD is mounted and unloaded from memory when the CD is dismounted from the file system.

The Module support under Linux has three components:

- 1) The **Module management** allows module to be loaded into memory and to talk to the rest of the kernel.
- 2) The **driver registration** allows modules to tell the rest of the kernel that a new driver has become available.
- 3) A **conflict-resolution mechanism** allows different device drivers to reserve hardware resources and to protect those resources from accidental use by another driver.

The shell

The shell is an advanced way of communicating with the system, because it allows for conversation and taking initiative. Both partners in the communication are equal, so new ideas can be tested. The shell allows the user to handle a system in a flexible way, and is also a means of stress reduction.

Shell types

There are different types of shells, which are:

- 1 **sh or Bourne Shell:** The original shell still used on UNIX systems and in UNIX related environments. This is the basic shell, a small program with few features.
- 1 **bash or Bourne Again shell:** the standard GNU shell, intuitive and flexible. Probably most advisable for beginning users while being at the same time a powerful tool for the advanced and professional user. On Linux, **bash** is the standard shell for common users. This shell is a so-called superset of the Bourne shell, a set of add-ons and plug-ins. The Bourne Again shell is compatible with the Bourne shell.
- 1 **csh or C shell:** the syntax of this shell resembles that of the C programming language. Sometimes used by programmers.
- 1 **tcsh or Turbo C shell:** a superset of the common C shell, enhancing user-friendliness and speed.
- 1 **ksh or the Korn shell:** it is appreciated by people with a UNIX background. This shell is the superset of the Bourne shell with standard configuration .

6.5.5 Drawbacks of Using Linux

The biggest disadvantage of using Linux is the fact that no single corporate entity is in charge of its development. There are several disadvantages discussed below:

- 1) **lack of technical Support:**
- 2) **Hardware problems:** one disadvantage of Linux is that it can be hard to install and does not work on all hardware platforms. Unlike a commercial program development operation, where a cohesive group spends months building and testing a program against a variety of conditions and hardware,

Linux developers are scattered around the globe. There are no formal quality-assurance programs. The hardware supported by Linux depends on the hardware each developer owns while writing that portion of the code.

- 3) **Inability to use current software:** Another disadvantage is that your current applications for such operating systems as DOS are more likely won't work under Linux. You cannot use both operating systems at the same time, you can leave Linux and boot the other operating system to use your applications there.
- 4) **Lack of Experience:** finally, unless you are already a UNIX user, you must learn how to manage a Linux system. Linux and UNIX need to be managed. The system administrator is responsible for maintaining the system and for performing such duties as adding and deleting user accounts, backing up the system on a regular basis, installing new software, configuring the system and fixing things when they go wrong.
- 5) **Inconsistent User Interface:** the development of GNU/Linux reflects different interfaces, design goals, etc. It is not easy learning to use the vi editor, for example, or learning the command line syntax of find. Nor is there any consistency enforced among the various programs and utilities included in a standard Linux distribution. This leads to user confusion and frustration

Self Check Exercises

- 26) What are the drawbacks of using Linux?
- 27) What are the different types of Shells Available?

Note: i) Write your answer in the space given below.

ii) Check your answer with the answer given at the end of this unit.

.....

.....

.....

.....

.....

.....

.....

6.6 SUMMARY

In this unit you have learnt the basic commands for different operating systems : DOS, Windows, Unix and Linux. DOS commands are used on PC-Compatible computers to interact with the various parts of the computer hardware. DOS commands are used for file and directory management, disk maintenance hardware configuration, memory optimization and DOS customization. Windows operating system provides life to modern PCs. It provides graphical interface which is easy to use and one can see what is being done. Unix operating system from Bell Labs is one of the most powerful, versatile and flexible operating system. It can be run on a wide variety of machines, from micros to supercomputers. Its portability feature has made it very popular. The main features of Unix are multiuser and multitasking capabilities and hierarchical file system. Unix OS is made up of three parts the kernel the shell and the programs. The Linux operating

system is the only one that is freely available that provides multitasking and multiprocessing capabilities for multiple users of IBM PC-compatible hardware platforms. The source code can be modified to fit individual needs. The basic components of Linux are Kernel, System libraries and system utilities.

6.7 ANSWERS TO SELF CHECK EXERCISES

1) **RAM** - Random Access Memory. Its volatile and can be read from as written to.

ROM - Read Only Memory, can be read from repeatedly but can't be written to - Its not volatile

2) A ROM-BIOS is the instructions written on ROM chip for performing the basic services like elementary instructions to read and write data for your hardware.

3) MS-DOS supports - conventional memory, extended memory and expanded memory, high memory area & upper memory area.

4) DOS supports the hierarchical file system, in which directory contains subdirectories and these subdirectories in turn contain other file or more subdirectories.

5) In the given file name CONFIG is the file name and SYS is the extension which specify that its a System file.

6) The pathname of DOS.DOC is -

C:\Semester1\OS1\DOS.DOC\

7) **External commands** are stored in the disk is the form of a file with .com and .exe extensions. **Internal commands** are provide with OS and written in the COMMAND.COM files.

8) Following commands used are:

i) **MD** - Creates the directory with specified name under the current directory.
Syntax - md<directory_name>.

ii) **CD** - Changes the directory to the subdirectory with the specified name.
Syntax - CD<directory_name>.

iii) **DIR** - Displays list of all files/directories under the root directory. Syntax - dir.

iv) **REN** - Renames old name of the file with the new name. **Syntax** - REN<oldname> <newname>

9) In Windows, the things that are present on the computer, like drives, programs, documents, and data files, are all accessible from one place called My Computer. The icon of my computer is present on the desktop of the computer.

10) The taskbar contains the **Start** button, which is used to quickly start a program or to find a file. It is also the fastest way to get Help. By default, the taskbar and the Start menu are always visible when Windows is running.

Also when you open a program, a document, or window, a button appears on the taskbar. You can use this button to quickly switch between the windows you have opened.

11) Small pictures that represent a program or a file are called on icon. In Windows 98, the term object is used instead of icons. Objects represent data files, programs, devices and folders.

12) The main features of UNIX OS are:

- 1 Multiuser Capabilities
- 1 Multitasking Capabilities
- 1 Hierarchical file system

13) Commands

- i) **rm** - Remove - Used to delete a file
- ii) **mkdir** - Make directory - Used to create directories .
- iii) **cd** - Change directory - Used to change the current working directory to another directory
- iv) **cp** - Copy - Used to copy a file to another directory.

14) The UNIX OS is made up of three parts - The Kernel, The Shell and the programs.

15) To run a foreground process as a background job.

- i) Suspend the foreground process
- ii) Enter the by (background) command to move the process into the background.

16) Unix file system supports following types of files-

- 1 Ordinary files
- 1 Special files
- 1 Directories
- 1 Pipes

17) Access right on file.

r/(-): indicates read permission

w/(-): Indicates write permission

x/(-): Indicates execute permission

Access rights on directories

- 1) r: allows the user to list files in the directory
- 2) w: users may delete or move files from the directory.
- 3) x: means right to access files in the directory

- 18) A filter is a program that reads data from standard input, process it and sends the processed data to standard output
- 19) Shell scripts are plaintext files. These files can be created using editor such as emacs or vi
- 20) The system keep track of following information about each user:
- 1 User name
 - 1 UserID
 - 1 GroupID
 - 1 Password
 - 1 Filename
 - 1 Home directory
 - 1 Login shell
- 21) The file system check utility (fsck) is an interactive program that prompt for yes or no decisions on whether or into to correct certain error conditions.
- 22) The various features of Linux are:
- 1 Multitasking
 - 1 Multiuser
 - 1 Reliability
 - 1 Networking capabilities
- 23) The reasons why we use Linux is that Linux is the only operating system today that is freely available to provide multitasking and multiprocessing capabilities for multiple users on IBM PC-compatible hardware platforms. There are no royalty or licensing fees for using Linux, and the source code can be modified to fit your needs.
- 24) The basic components of Linux are:
- 1 Kernel
 - 1 System libraries
 - 1 System utilities
- 25) The different types of files provided by Linux are:
- 1 Directories
 - 1 Special files
 - 1 Links
 - 1 (Domain) sockets
 - 1 Named pipes
- 26) The various drawbacks of using Linux are:
- 1 Lack of technical support
 - 1 Hardware problems

- 1 Inability to use current software
- 1 Lack of experience

27) The different types of Shells are:

- 1 sh or Bourne Shell
- 1 bash or Bourne Again shell
- 1 csh or C shell
- 1 tcsh or Turbo C shell
- 1 ksh or the Korn shell

6.8 KEY WORDS

- BIOS** : Acronym for Basic Input/Output System. It is the part of the operating system that controls communication with a monitor screen, keyboard, printer and other peripheral devices.
- Echo** : A program command to display information lines on a monitor screen, as of a command that is being executed or data that is being transmitted.
- Multitasking** : The ability of a CPU to execute two or more programs or routines at the same time either by independent processing or by interleaving.
- Partitioning** : The division of a large hard disk into smaller segments for more efficient management of programs and data.

6.9 REFERENCES AND FURTHER READING

- Duncan, Ray (1994). Advanced MSDOS programming. - New Delhi: BPB Publication.
- Harris, J. Archer (2002). Schann's Outlines Operating Systems. - New Delhi : Tata Mcgraw-Hill.
- Kerningham, Brian and Pike, Rob (1984). The Unix Programming Environment. - New Jersey : Prentice-Hall.
- Levine, John and Young, Margaret Levine (1988). The Complete Reference Windows 98. - New Delhi: Tata Mc-graw Hill.
- Stroel, Stefan and Ulil, Thomas (1996). Linux Universe : Installation and Configuration. - 2nd ed. - Heil bronn : Springer.