11364

PRACTICAL FILE

OF

"PRINCIPLES OF OPERATING SYSTEM LAB"

DRONACHARYA COLLEGE OF ENGINEERING,
FARRUKH NAGAR (GURGAON)
AFFILIATED TO MAHARISHI DAYANAND UNIVERSITY, ROHTAK
APPROVED BY AICTE

Submitted by:
Name: MANISH GUPTA
Roll No.: 11364
Branch: I.T(V Sem.)

CERTIFICATE

This is to certify that the practical file on **Principles of Operating System Lab** prepared by **MANISH GUPTA, Roll No 11364** of **I.T(V SEM)** as an integral part of the practical curriculum and the information gathered has helped the cause of understanding the subject.

Mr.	**Ms.**	**Mr.**
(Head of Department) I.T.	(Assistant Professor)	(Lab Faculty)

INDEX

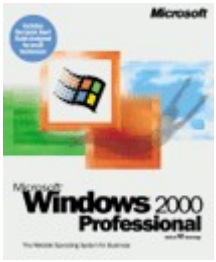
S No.	TOPIC	DATE	SIGNATURE
1.	STUDY OF WINDOW 2000 OPERATING SYSTEM.		
2.	ADMINISTRATION OF WINDOWS 2000 OPERATING SYSTEM(INCLUDING DNS,LDAP,DIRECTORY, SERVICES).		
3.	STUDY OF LINUX OPERATING SYSTEM (LINUX,KERNEL,SHELL, BASIC,COMMANDS,PIPE AND FILTER COMMANDS).		
4.	ADMINISTRATION OF LINUX OPERATING SYSTEM.		
5.	SHELL SCRIPTING(SHELL PROGRAMING).		
6.	TO STUDY LINUX BASIC COMMANDS.		
7.	WRITE A SHELL PROGRAM TO PERFORM OPERATIONS USING CASE STATEMENT SUCH AS:- (1).ADDITION (2).SUBTRACTION (3).MULTIPLICATION (4).DIVISION.		

8.	WRITE A SHELL SCRIPT TO SEE CURRENT DATE,TIME,USERNAME AND DIRECTORY.		
9.	WRITE A SHELL PROGRAM TO FIND LARGEST OF THREE NUMBERS.		
10.	STUDY OF DOS OPERATING SYSTEM(WITH INTERNAL AND EXTERNAL COMMAND).		

EXPERIMENT-1

STUDY OF WINDOWS 2000 OPERATING SYSTEM

Windows 2000 , also referred to as **Win2K**, is a **preemptive**, **interruptible**, **graphical** and **business-oriented** operating system designed to work with either **uniprocessor** or **symmetric multi-processor** computers. It is part of the **Microsoft Windows NT** line of operating systems. Windows 2000 is classified as a **hybrid kernel operating system**. It was released on **February 17**, **2000**.



CONSISTS OF FOUR PRODUCTS:-

- **Windows 2000 Professional:-** Windows 2000 is based of the **Windows NT Kernel** and is sometimes referred to as **Windows NT 5.0**, an operating system for **business desktop** and **laptop systems**. It includes security and mobile use enhancements. It is the most economical choice. It is used to run **software application**, connect to **Internet** and **intranet** sites, and access files, **printers**, and **network resources**.

  Features of Windows 2000 Professional are:-

  - Support for FAT16, FAT32 and NTFS.
  - Increased uptime of the system and significantly fewer OS reboot scenarios.
  - Windows Installer tracks applications and recognizes and replaces missing components.
  - Protects memory of individual apps and processes to avoid a single app bringing the system down.
  - Encrypted File Systems protects sensitive data.
  - Secure Virtual Private Networking (VPN) supports tunneling in to private LAN over public Internet.
  - Personalized menus adapt to the way you work.
  - Multilingual version allows for User Interface and help to switch, based on logon.
  - Includes broader support for high-speed networking devices, including Native ATM and cable modems.
  - Supports Universal Serial Bus (USB) and IEEE 1394 for greater bandwidth devices.

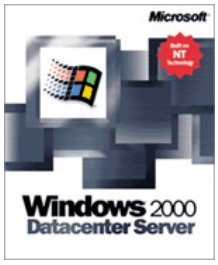
- **Windows 2000 Server**:- It aimed at small-to-medium size businesses. It can function as a **Web server** and/or a **workgroup** (or **branch office**) **server**. It can be part of a two-way symmetric multiprocessing system. **NT 4.0 servers** can be upgraded to this server. Windows 2000 Server lets users build **Web applications** and connect to the **Internet**.

  Features of Windows 2000 Server are:-

  - Active Directory improves manageability, enables security, and extends interoperability with other operating systems.
  - Provides high-level interfaces for database access and Active Directory services.
  - Lets you use COM+ to run component-based applications, integrated Web applications and message-queuing services.
  - Transaction Services feature makes it easier to develop and deploy server-centric applications.
  - Microsoft BackOffice is fully integrated into Windows 2000 Server.

- **Windows 2000 Advanced Server:-** It aimed at being a network operating **system server** and/or an **application server**, including those involving large **database**. It is an operating for **line-of-business applications** and **e-commerce.** This server facilitates **clustering** and **load-balancing,** plus additional features for applications that require higher levels of **scalability** and **availability**. NT 4.0 servers with up to eight-way SMP can upgrade to this product.

  Features of Windows 2000 Advanced Server are:-



  - The server operating system for e-commerce and line-of-business applications.
  - Includes all the features of Windows 2000 Server, with additional scalability and clustering support.
  - Increased reliability, ensure your business-critical applications are online when your customers need them.
  - Easier to use and manage clusters, applications, and updates.
  - Supports 8-way symmetric multiprocessing (SMP) and up to 8 GB of memory (RAM).

- **Windows 2000 Datacenter Server:-** It is designed for large **data warehouses**, **online transaction** processing (**OLTP**), **econometric analysis**, and other applications requiring high-speed computation and large databases. It developed to work in enterprises that need reliable high-end drivers and software. The Datacenter Server supports up to **16-way SMP** and up to **64 gigabytes** of physical memory.

  Features of Windows 2000 Datacenter Server are:-

- Designed for large businesses that move large quantities of confidential or sensitive data frequently via a central server.
- Support for FAT16, FAT32 and NTFS.
- Support up to 32 processors in a single system.
- Compatible with Windows 16-bit programs and Windows 32-bit programs.
- Windows 2000 DataCenter server provides a GUI interface.
- Support up to 16-way SMP and up to 64 gigabytes of physical memory.
- Support clustering, failover and load balancing.

WINDOWS 2000 FEATURES:-

- **Reliability:-**

  - Windows File Protection protects core system files from being overwritten by application installs.
  - Driver certification provides safeguards to assure you that device drivers have not been tampered with and reduces your risk of installing non-certified drivers.
  - Full 32-bit operating system minimizes the chance of application failures and unplanned reboots.

- **Mobility:-**

  - Hibernate turns off your computer and monitor after a pre-determined time, while retaining your desktop on disk.
  - Offline Viewing makes entire web pages with graphics available for viewing offline.
  - Synchronization Manager allows you to compare and update your offline files and folders with those on the network.
  - Smart Battery gives you a more accurate view of your battery's life, enabling you to reduce power to specific functions to extend your battery's power.
  - Hot Docking lets you dock or undock your notebook computer without changing hardware configuration or rebooting.
  - Universal Serial Bus (USB) lets you connect and disconnect a wide array of peripherals such as joysticks, scanners, and camcorders without configuring or rebooting your computer.
  - IrDA support provides secure, wireless communications between two Windows 2000 computers using the popular infrared protocol.
  - IEEE 1394 provides a higher bandwidth connection for devices that require faster data transfer.

- **Maintainability:-**

- System Preparation Tool (SysPrep) helps administrators clone computer configurations, systems, and applications.
- Setup Manager provides a graphical wizard that guides administrators in designing installation scripts.
- Multilingual support allows users to easily create, read, and edit documents in hundreds of languages.
- Windows 2000 offers 25% faster performance than Windows 95 or Windows 98 on systems with 64MB or more of memory.
- 32-bit architecture allows you to run more programs and perform more tasks at the same time than Windows 95 or 98.
- Windows 2000 can support up to 4GB of RAM and two symmetric multiprocessors.
- Encrypting File System (EFS) encrypts each file with a randomly generated key.
- IP Security (IPsec) support helps protect data transmitted across a network.
- Kerberos support provides industry standard high-strength authentication with a fast, single login to Windows 2000 enterprise resources.

- **Internet Capability:-**

  - Internet Information Services (IIS) 5.0 includes web and FTP server support, as well as support for FrontPage transactions, Active Server Pages (ASP), and database connections.
  - Windows 2000 has strong development platform support for Dynamic HTML behaviors and XML.
  - IntelliForms alleviates the tedium of filling out forms on the web by automatically entering your name, address, or other information that you've securely stored on your computer.
  - Automated Proxy automatically locates a proxy server and configures Internet Explorer 5.01 to connect to the Internet through that server.

CATEGORIZED INTO FOUR TYPES:-

- **Real-time operating system (RTOS):-**

  - Real-time operating systems are used to control machinery, scientific instruments and industrial systems.
  - An RTOS typically has very little user-interface capability, and no end-user utilities, since the system will be a "sealed box" when delivered for use.
  - A very important part of an RTOS is managing the resources of the computer so that a particular operation executes in precisely the same amount of time every time it occurs.
  - In a complex machine, having a part move more quickly just because system resources are available may be just as catastrophic as having it not move at all because the system is busy.

- **Single-user, single task**:-
    - This operating system is designed to manage the computer so that one user can effectively do one thing at a time.
    - The **Palm OS** for Palm handheld computers is a good example of a modern single-user, single-task operating system.

- **Single-user, multi-tasking:-**
    - This is the type of operating system most people use on their desktop and laptop computers today.
    - Microsoft's **Windows** and **Apple's MacOS** platforms are both examples of operating systems that will let a single user have several programs in operation at the same time.
    - For example, it's entirely possible for a Windows user to be writing a note in a word processor while downloading a file from the Internet while printing the text of an e-mail message.

- **Multi-user**:-
    - A multi-user operating system allows many different users to take advantage of the computer's resources simultaneously.
    - The operating system must make sure that the requirements of the various users are balanced, and that each of the programs they are using has sufficient and separate resources so that a problem with one user doesn't affect the entire community of users.
    - **Unix**, **VMS** and **mainframe operating systems**, such as **MVS**, are examples of multi-user operating systems.

ACCESSIBLITY SUPPORT:-

Microsoft included several utilities designed to make the system more accessible, they are:-

- **FilterKeys:-** These are a group of keyboard related features for people with typing issues, and include:
    - **SlowKeys:-** Ignore keystrokes that are not held down for a certain time period.
    - **BounceKeys:-** Multiple keystrokes to one key to be ignored within a certain timeframe.
    - **RepeatKeys:-** allows users to slow down the rate at which keys are repeated via the keyboard's keyrepeat feature.

- **ToggleKeys:-** When turned on, Windows will play a sound when either the **CAPS LOCK**, **NUM LOCK** or **SCROLL LOCK** keys are pressed.

- **MouseKeys:-** Allows the cursor to be moved around the screen via the numeric keypad instead of the mouse.
- **On-screen keyboard:-** Allows users to use a mouse to use the keyboard and enter on-screen keyboard characters.
- **SerialKeys:-** Gives Windows 2000 the ability to support speech augmentation devices.
- **StickyKeys:-** Makes modifier keys (ALT, CTRL and SHIFT) become "sticky" — in other words a user can press the modifier key, release that key and then press the combination key. Normally the modifier key must remain pressed down to activate the sequence
- **Microsoft Magnifier:-** A screen magnifier that assists users with visual impairments by magnifying the part of the screen they place their mouse over.
- **Narrator:-** Microsoft Narrator, introduced in **Windows 2000**, assists users with visual impairments with system messages, as when these appear the narrator will read this out via the sound system.
- **High contrast theme:-** To assist users with visual impairments.
- **SoundSentry:-** Designed to help users with auditory impairments, Windows 2000 will show a visual effect when a sound is played through the sound system.

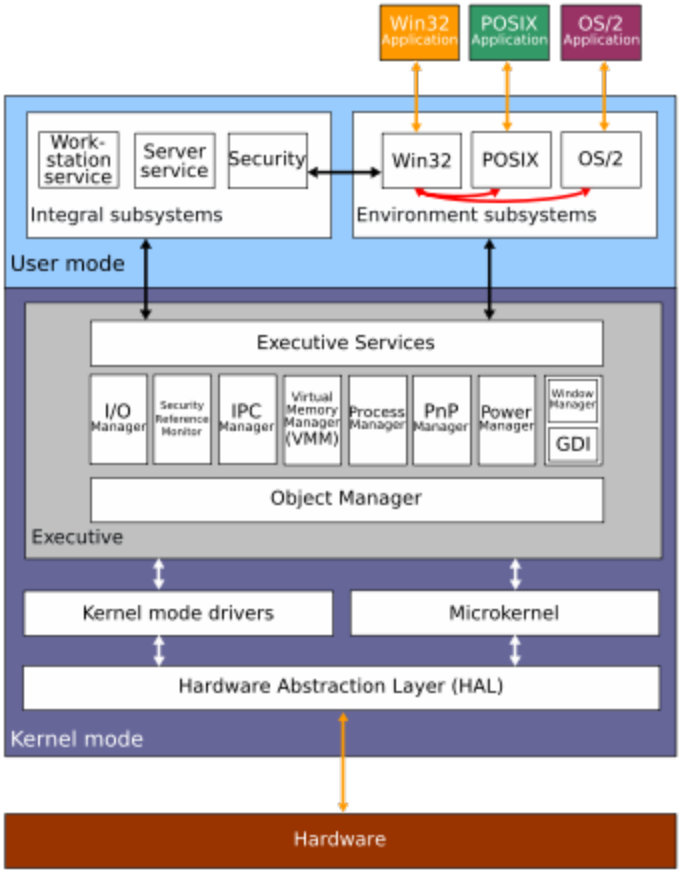
EXPERIMENT-2

ADMINISTRATION OF WINDOWS 2000 OPERATING SYSTEM

Windows 2000 is a **portable operating system** because of two design decisions:-

- First, the operating system was written in ANSI C, a language that enables programs to be ported easily to other hardware architectures.
- Second, all parts of Windows 2000 that must be written for a specific hardware are isolated in an area called the Hardware Abstraction Layer (HAL).

To move Windows 2000 to a new hardware platform, developers need to do little more than recompile the C code for the new hardware and create a new **HAL**. Designing an OS around a HAL means that a large portion of the code is exactly the same between hardware platforms. This also means that only the small slice of code that interfaces with the computer's hardware needs to be rewritten as Windows 2000 is ported between different processor architectures. Thus, it provides a high level of portability.



WINDOWS 2000 OPERATES IN TWO MODES:-

KERNEL MODE:-

In this mode, the software is able to access the hardware and system data, as well as access all other system resources.

The kernel mode has the following components:-

- **Executive**:- It contains components that implement memory management, process and thread management, security, I/O,

interprocess communication, and other base operating system services. For the most part, these components interact with one another in a modular, layered fashion. The executive interfaces with all the user mode subsystems. It deals with I/O, object management, security and process management.

It contains various components, including:-

- **Object manager:-** A special executive subsystem that all other executive subsystems must pass through to gain access to Windows 2000 resources. This is essentially a resource management infrastructure service that allows Windows 2000 to be an object oriented operating system.
- **I/O Manager:-** It allows devices to communicate with user-mode subsystems by translating user-mode read and write commands and passing them to device drivers.
- **Security Reference Monitor (SRM):-** The primary authority for enforcing the security rules of the security integral subsystem.
- **IPC Manager:-** Inter-Process Communication Manager, manages the communication between clients (the environment subsystem) and servers (components of the executive).
- **Virtual Memory Manager:-** It manages virtual memory, allowing Windows 2000 to use the hard disk as a primary storage device (although strictly speaking it is secondary storage).
- **Process Manager:-** It handles process and thread creation and termination
- **PnP Manager:-** It handles Plug and Play and supports device detection and installation at boot time.
- **Power Manager:-** The power manager coordinates power events and generates power **IRPs.**
- **Window Manager:-** The component of this driver is responsible for drawing windows and menus while the **GDI** (Graphics Device Interface) component is responsible for tasks such as drawing lines and curves, rendering fonts and handling palettes. Windows 2000 also introduced alpha blending into the Graphics Device Interface which reflects in the fade effect in menus. The display system is handled by a device driver contained in **Win32k.sys.**

- **Microkernel**:- The Microkernel's primary functions are to provide multiprocessor synchronization, thread and interrupt scheduling and dispatching, and trap handling and exception dispatching. During system startup, it extracts information from the Registry, such as which device drivers to load and in what order.
- **Hardware Abstraction Layer (HAL):-** The HAL is the code associated with Windows 2000 that changes with the hardware the operating system is being run on. Thus, it becomes compatible with multiple processor platforms. The HAL manipulates the hardware directly.
- **Device drivers**:- Device drivers send and receive load parameters and configuration data from the Registry.

- **Windowing and graphics system**:- This system implements the graphical user interface (GUI).

USER MODE:-

Software in the user mode cannot access hardware directly. **The user mode-protected subsystem has four primary responsibilities:-**

- Special system support processes, such as the logon process and the session manager.
- Windows 2000 services that are server processes, such as the Event Log and Schedule services.
- Environment subsystems that provide an operating system environment by exposing the native operating system services to user applications. They include Win32, POSIX, and OS/2 subsystems.
- User applications—either Win32, Windows 3.1, MS-DOS, POSIX, or OS/2.

User applications do not call the native Windows 2000 operating system services directly; instead, they go through subsystem dynamic link libraries (DLLs). The subsystem dynamic link libraries translate a documented function into the appropriate undocumented Windows 2000 system service calls.

Two subsystems make up the user mode layer of Windows 2000:-

- **Environment Subsystem:-** The environment subsystems are services that provide application programming interfaces (APIs) that are specific to an operating system.
    - The **three environment subsystems** are the **POSIX**, **OS/2**, and **Win32** subsystems. Applications and subsystems form a client/server relationship, in which the applications are the clients and the subsystems are the servers. One of the benefits of this type of architecture is that you can include support for other types of applications to Windows 2000 simply by adding subsystems.
    - Applications cannot interfere with each other because they run in separate address spaces. Operating system code and data in the subsystems are protected from applications because subsystems also reside in their own address spaces. The Executive shares address space with running processes, but it is protected by the wall between kernel mode and user mode. It is impossible for an application to corrupt code or store data in the Executive because the processor notifies the operating system of invalid memory access before these things occur.

- **Integral Subsystems:-** The integral subsystems are services that provide the APIs that Win32 applications call to perform important operating system functions, such as creating windows and opening files. It has **five main components**, which utilize four main support functions:-
    - **Process and thread manager:-** The process manager sees processes as objects. Its responsibility is to create and terminate processes and threads. It also suspends

and resumes the execution of threads, and stores and retrieves information about processes and threads.

- **Virtual Memory Manager**:- The Virtual Memory Manager (VMM) performs three essential functions: managing the virtual address space of each process, sharing memory between processes, and protecting each process's virtual memory. It is also the underlying support for the cache manager. Each processor that Windows 2000 supports implements virtual memory through hardware differently; therefore, the portion of Windows 2000 that directly interfaces with virtual memory hardware is not portable and must be recorded when moving to another platform to minimize headaches. This code is small and well-isolated in Windows 2000. Windows 2000 supports 4GB of virtual memory.
- **Security reference monitor:-** The security reference monitor is responsible for controlling which objects have permissions to which resources. Each object has an Access Control List (ACL) that is queried when the object makes a service request. Access to resources is allowed or disallowed according to the rights the module has in the ACL.
- **I/O system manager**:- The I/O manager is responsible for dispatching all system I/O requests. All I/O devices, network ports, printers, drives, and so on are mapped to virtual files. These virtual files are referred to as file objects and are managed by the object manager just like any other object.
- **Cache manager**:- The cache manager improves the performance of file-based I/O by causing recently referenced disk data to reside in main memory for quick access. It also defers disk writing by holding the updates in memory for a short time before sending them to disk.

The support functions are the following:-

- **Object manager:-** The object manager creates, manages, and deletes Executive objects. Executive objects are created in the Executive, and are accessible to the Executive and protected subsystems. They can be thought of as message packets that represent items such as processes, threads, semaphores, and other low-level objects.
- **LPC facility**:- Local Procedure Calls (LPCs) are used to pass messages between processes running on a single Windows 2000 system. Because LPC message-passing requires quite a bit of overhead, the LPC facility is utilized only when an API must change global data. Otherwise, API routines can be implemented directly in a private Dynamic Link Library (DLL).
- **Run-time library functions**:- Similar to string processing, arithmetic operations, data type conversion, and security structure processing.
- **Executive support routines**:- Similar to system memory allocation and interlocked memory access.

SYSTEM REQUIREMENTS FOR WINDOWS 2000:-

To run Windows 2000, Microsoft recommends:-

- 133 MHz CPU

- 256 MB of RAM recommended minimum
- 2 GB hard disk space

USER INTERFACE:-

Some selections using various icons and selections include:

- **Recycle Bin:-** Used to store deleted files and folders. When emptied, files or folders are gone for good.
- **My Network Places Icon:-**
    - **Add Network Place selection:-** Used to connect to a shared network folder or the world wide web.
    - **Computers Near Me selection:-** Used to connect to computers in your domain or workgroup.
    - **Entire Network selection:-**
        - Used to view all domains, workgroups, and computers on the organizational network.
        - Used to search for a specific computer.
        - Used to search for specific files or folders.
- **Windows Explorer:-** To run, select "Start", "Programs", "Accessories", and "Windows Explorer".

BASIC AND DYNAMIC DISK STORAGE:-

Windows 2000 introduced the Logical Disk Manager for **dynamic storage**. All versions of Windows 2000 support **three types** of dynamic disk volumes (along with basic disks):-

- **Simple volume**, a volume with disk space from one disk.
- **Spanned volumes**, where multiple disks (up to 32) show up as one, increasing it in size but not enhancing performance. When one disk fails, the array is destroyed. Some data may be recoverable. This corresponds to JBOD and not to RAID-1.
- **Striped volumes**, also known as RAID-0, store all their data across several disks in stripes. This allows better performance because disk read and writes are balanced across multiple disks.

EXPERIMENT-3

STUDY OF LINUX OPERATING SYSTEM

Linux is a version of the **UNIX** operating system, comparable to Windows or Mac OS X. Linux is one of the most prominent examples of **free software** and **open source development**, its underlying source code can be freely modified, used, and redistributed by anyone

Linux runs on a wide variety of hardware platforms, from huge **mainframes** to **desktop PCs** to **cell phones**. It is licensed under the **Free Software Foundation's GNU Project's GNU General Public License**, version 2, which lets users modify and redistribute the software.

The **Linux kernel** was first released to the public on **17 September 1991**, for the Intel **x86 PC architecture**. The kernel was augmented with system utilities and libraries from the GNU project to create a usable operating system, which later led to an alternate term, **GNU/Linux**.

Linux is supported by corporations such as **Dell**, **Hewlett-Packard**, **IBM**, **Novell**, **Oracle Corporation**, **Red Hat**, and **Sun Microsystems**.

Linux is used as an operating system for a wide variety of computer hardware, including **desktop computers**, **supercomputers**, **video game systems** such as **PlayStation 2**, **3**, several **arcade games** and embedded devices such as **mobile phones** and **routers**.

SERVICES INCLUDES:-

Typical services that an operating system provides include:-

- **A task scheduler:-** The task scheduler is able to allocate the execution of the CPU to a number of different tasks. Some of those tasks are the different applications that the user is running, and some of them are operating system tasks. The task scheduler is the part of the operating system that lets you print a document from your word processor in one window while you are downloading a file in another window and recalculating a spreadsheet in a third window.

- **A memory manager:-** The memory manager controls the system's RAM and normally creates a larger virtual memory space using a file on the hard disk. (See also this Question of the Day.)
- **A disk manager:-** The disk manager creates and maintains the directories and files on the disk. When you request a file, the disk manager brings it in from the disk.
- **A network manager:-** The network manager controls all data moving between the computer and the network.
- **Other I/O services manager:-** The OS manages the keyboard, mouse, video display, printers, etc.
- **Security manager:-** The OS maintains the security of the information in the computer's files and controls who can access the computer.

LINUX FEATURES:-

The Linux kernel can allow the operating system (OS) to be modified to include support for the features you want, every Linux kernel can offer the following features:-

- **Multiuser:-**
    - Not only can you have many user accounts available on a Linux system, you can also have multiple users logged in and working on the system at the same time.
    - Users can have their own environments arranged the way they want: their own home directory for storing files and their own desktop interface (with icons, menus, and applications arranged to suit them).
    - User accounts can be password-protected, so that users can control who has access to their applications and data.

- **Multitasking:-**
    - In Linux, it is possible to have many programs running at the same time, which means that not only can you have many programs going at once, but that the Linux operating system can itself have programs running in the background.
    - Many of these system processes make it possible for Linux to work as a server, with these background processes listening to the network for requests to log in to your system, view a Web page, print a document, or copy a file. These background processes are referred to as **daemons**.

- **Graphical user interface (X Window System):-**
    - The powerful framework for working with graphical applications in Linux is referred to as the X Window System (or simply **X**).
    - X handles the functions of opening X-based graphical user interface (GUI) applications and displaying them on an X server process (the process that manages your screen, mouse, and keyboard).

- On top of X, you use an X-based desktop environment to provide a desktop metaphor and window manager to provide the look-and-feel of your GUI (icons, window frames, menus, and colors, or a combination of those items called themes).
- There are several desktop environments and several desktop managers to choose from. (Fedora and RHEL focus on the GNOME and KDE desktop environments, but make several other desktop environments and window managers available in Fedora Extras.)

- **Hardware support:-**

  - You can configure support for almost every type of hardware that can be connected to a computer.
  - There is support for floppy disk drives, CD-ROMs, removable disks (such as DVDs and pen drives), sound cards, tape devices, video cards.
  - As device interfaces, such as USB and FireWire, have been added to computers, support for those devices has been added to Linux as well.

- **Networking connectivity:-**

  - To connect your Linux system to a network, Linux offers support for a variety of local area network (LAN) cards, modems, and serial devices.
  - In addition to LAN protocols, such as Ethernet (both wired and wireless), all the most popular upper-level networking protocols can be built-in.
  - The most popular of these protocols is **TCP/IP** (used to connect to the Internet). Other protocols, such as IPX (for Novell networks) and X.25 (a packet-switching network type that is popular in Europe), are also available.

- **Network servers:-**

  - Providing networking services to the client computers on the **LAN** or to the entire Internet is what Linux does best.
  - A variety of software packages are available that enable you to use Linux as a print server, file server, FTP server, mail server, Web server, news server, or workgroup (DHCP or NIS) server.

Application support:-

- Because of compatibility with POSIX and several different application programming interfaces (APIs), a wide range of freeware and shareware software is available for Linux.
- Most GNU software from the Free Software Foundation will run in Linux.

LINUX DESKTOP:-

Desktop environments are more than a graphical interface that comes with tools, utilities, games, and other applications to make the user's computing experience a richer one.

Popular desktop environments that work with Linux are:-

- **KDE (K Desktop Environment):-**
    - KDE is most popular, that runs on any Unix operating system, including Linux.
    - The entire KDE project is supported by the software development community and is provided to Linux users at no cost.
    - All of the source code for KDE is licensed under the terms of the GNU General Public License, which means that anyone can access and change KDE to suit specific purposes. KDE comes packaged with most Linux distributions and includes standardized menus, toolbars, and color schemes, as well as a complete help system, networking tools, graphics and multimedia applications, and a complete office productivity solution, and dozens of other software tools.
- **GNOME (GNU Network Object Model Environment):-**
    - GNOME is another ubiquitous GUI or desktop environment for Linux.
    - It is also licensed under the terms of the GNU General Public License, which means it is freely available, along with the source code, for use on any Unix-based operating system.
    - GNOME comes packaged with just about every Linux distribution.
    - It is a part of the GNU project, which created the GNU operating system, parts of which are included with all standard Linux distributions.
    - The GNOME desktop environment also includes more than just toolbars, icons and menus. Help files, networking tools, games, and productivity applications like GNOME Office round out the free software offering.
- **XPDE desktop environment:-** This tries to make it easier for Windows XP users to use a Linux box.
- **Xfce:-** It is a lightweight desktop environment for various *NIX systems. Designed for productivity, it loads and executes applications fast, while conserving system resources.
- **Enlightenment:-** It is advanced graphical libraries, tools, and environments.
- **IceWM:-** The goal of IceWM is speed, simplicity, and not getting in the user's way.
- **Blackbox:-** Blackbox is the fast, lightweight window manager for the X Window System you have been looking for, without all those annoying library dependencies.
- **Window Maker:-** Window Maker is an X11 window manager originally designed to provide integration support for the GNUstep Desktop Environment.
- **FluxBox:-** A fast compact window manager based on the Blackbox, but offering more features.

ADVANTAGES:-

- Linux source code is freely distributed
- Linux has the best technical support available
- Linux has no vendor lock-in
- Linux runs on a wide range of hardware
- Linux is exceptionally stable
- Linux has the tools and applications you need
- Linux interoperates with many other types of computer systems
- Linux has a low total cost of ownership

EXPERIMENT-4

ADMINISTRATION OF LINUX OPERATING SYSTEM

Linux is a modular **Unix-like** operating system. Linux uses a **monolithic kernel**, the **Linux kernel**, which handles process control, networking, and **peripheral** and **file system access**. **Device drivers** are integrated directly with the kernel.

Much of Linux's higher-level functionality is provided by separate projects which interface with the kernel. The **GNU userland** is an important part of most Linux systems, providing the **shell** and **Unix tools** which carry out many basic operating system tasks. Atop these tools **graphical user interfaces** can be placed, usually running via the **X Window System**.

LINUX KERNAL:-

The **Linux kernel** is the core of a large and complex operating system, and while it's huge, it is well organized in terms of **subsystems** and **layers**.

PROPERTIES OF LINUX KERNEL:-

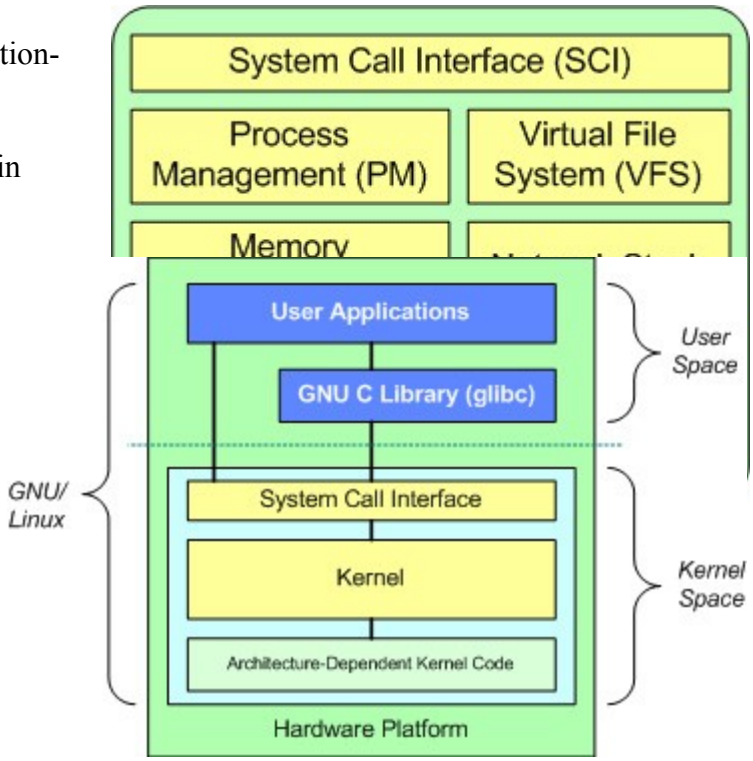
- The Linux kernel implements a number of important architectural attributes.
- At a high level, and at lower levels, the kernel is layered into a number of distinct subsystems.
- Linux can also be considered monolithic because it lumps all of the basic services into the kernel.
- Linux kernel is efficient in terms of both memory and CPU usage as well as extremely stable.
- The most interesting aspect of Linux, given its size and complexity, is its portability.
- Linux can be compiled to run on a huge number of processors and platforms with different architectural constraints and needs.

ARCHITECTURE OF LINUX KERNEL:-

- At the top is the **user**, or **application**, **space**. This is where the user applications are executed. Below the user space is the **kernel space**. Here, the Linux kernel exists.
- There is also **the GNU C Library** (**glibc**). This provides the system call interface that connects to the kernel and provides the mechanism to transition between the user-space application and the kernel. This is important because the kernel and user application occupy different protected address spaces. And while each user-space process occupies its own virtual address space, the kernel occupies a single address space.
- The Linux kernel can be further divided into **three gross levels**:-
    - At the top is the **system call interface**, which implements the basic functions such as read and write.
    - Below the system call interface is the **kernel code**, which can be more accurately defined as the architecture-independent kernel code. This code is common to all of the processor architectures supported by Linux.
    - Below this is the **architecture-dependent code**, which forms what is more commonly called a **BSP** (**Board Support Package**). This code serves as the processor and platform-specific code for the given architecture.

MAJOR SUBSYSTEMS OF THE LINUX KERNEL:-

- **System call interface:-**
    - The SCI is a thin layer that provides the means to perform function calls from user space into the kernel.
    - This interface can be architecture dependent, even within the same processor family.
    - The SCI is actually an interesting function-call multiplexing and demultiplexing service.
    - You can find the SCI implementation in ./linux/kernel, as well as architecture-dependent portions in ./linux/arch.



- **Process management:-**
    - Process management is focused on the execution of processes.
    - In the kernel, these are called **threads** and represent an individual virtualization of the processor (thread code, data, stack, and CPU registers).

- In user space, the term **process** is typically used, though the Linux implementation does not separate the two concepts (processes and threads).
- The kernel provides an application program interface (API) through the SCI to create a new process (fork, exec, or Portable Operating System Interface [POSIX] functions), stop a process (kill, exit), and communicate and synchronize between them (signal, or POSIX mechanisms).
- Also in process management is the need to share the CPU between the active threads. The kernel implements a novel scheduling algorithm that operates in constant time, regardless of the number of threads vying for the CPU. This is called the O(1) scheduler, denoting that the same amount of time is taken to schedule one thread as it is to schedule many.

- **Memory management:-**

  - Another important resource that's managed by the kernel is memory. For efficiency, given the way that the hardware manages virtual memory, memory is managed in what are called **pages** (4KB in size for most architecture).
  - Linux includes the means to manage the available memory, as well as the hardware mechanisms for physical and virtual mappings.
  - Linux provides abstractions over 4KB buffers, such as the slab allocator.
  - This memory management scheme uses 4KB buffers as its base, but then allocates structures from within, keeping track of which pages are full, partially used, and empty. This allows the scheme to dynamically grow and shrink based on the needs of the greater system.
  - Supporting multiple users of memory, there are times when the available memory can be exhausted. For this reason, pages can be moved out of memory and onto the disk. This process is called **swapping** because the pages are swapped from memory onto the hard disk.
  - You can find the memory management sources in ./linux/mm.

- **Virtual file system:-**

  - The virtual file system (VFS) is an interesting aspect of the Linux kernel because it provides a common interface abstraction for file systems.
  - The VFS provides a switching layer between the SCI and the file systems supported by the kernel.
  - At the top of the VFS is a common API abstraction of functions such as open, close, read, and writes.
  - At the bottom of the VFS are the file system abstractions that define how the upper-layer functions are implemented. These are plug-ins for the given file system You can find the file system sources in ./linux/fs.

- Below the file system layer is the buffer cache, which provides a common set of functions to the file system layer (independent of any particular file system). This caching layer optimizes access to the physical devices by keeping data around for a short time (or speculatively read ahead so that the data is available when needed).
- Below the buffer cache are the device drivers, which implement the interface for the particular physical device.

- **Network stack:-**

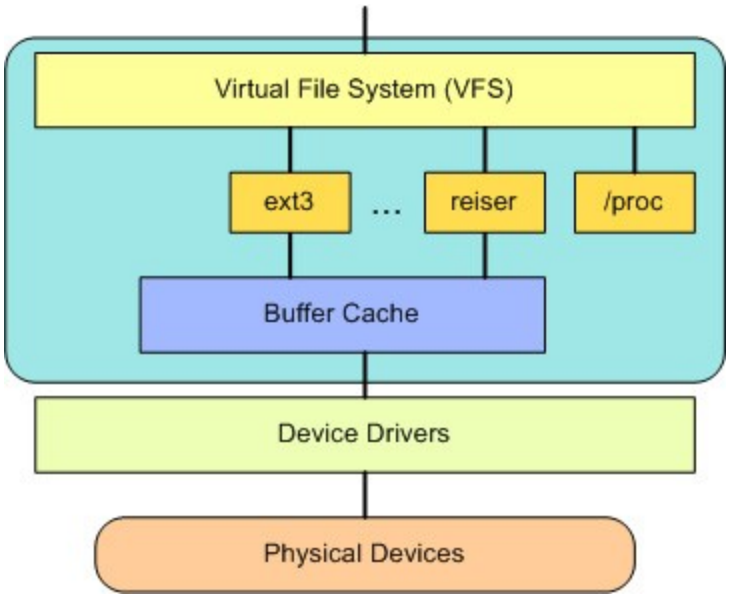
  - The network stack, by design, follows a layered architecture modeled after the protocols themselves.
  - The sockets layer is the standard API to the networking subsystem and provides a user interface to a variety of networking protocols.
  - From raw frame access to IP protocol data units (PDUs) and up to TCP and the User Datagram Protocol (UDP), the sockets layer provides a standardized way to manage connections and move data between endpoints.
  - You can find the networking sources in the kernel at ./linux/net.

- **Device drivers:-**

  - The vast majority of the source code in the Linux kernel exists in device drivers that make a particular hardware device usable.
  - The Linux source tree provides a drivers subdirectory that is further divided by the various devices that are supported, such as Bluetooth, I2C, serial, and so on.
  - You can find the device driver sources in ./linux/drivers.

- **Architecture-dependent code:-**

  - While much of Linux is independent of the architecture on which it runs, there are elements that must consider the architecture for normal operation and for efficiency.
  - The ./linux/arch subdirectory defines the architecture-dependent portion of the kernel source contained in a number of subdirectories that are specific to the architecture (collectively forming the BSP).
  - For a typical desktop, the i386 directory is used.



- Each architecture subdirectory contains a number of other subdirectories that focus on a particular aspect of the kernel, such as boot, kernel, memory management, and others.
- You can find the architecture-dependent code in ./linux/arch.

EXPERIMENT-6

LINUX BASIC COMMANDS

- **cal:-** Produces a calender of the current month as standard output. The month (1-12) and year (1-9999) must be specified in full numeric format.

  Syntax:- **cal** [[month] year]

- **cat**:- Read (con**cat**enate) one or more files and print them on standard output. Read standard input if no files are specified or if **-** is specified as one of the files; input ends with EOF. You can use the **>** operator to combine several files into a new file, or **>>** to append files to an existing file. When appending to an existing file, use Ctrl-D, the end-of-file symbol, to end the session.

  Syntax:- **cat** [filename(s)]: To display the contents of file
  cat > [filename]: To input data in file
  Data can be appended to a file using >>

- **date**:- Print the current date and time. You may specify a display format. format can consist of literal text strings as well as field descriptors, whose values will appear. A privileged user can change the system's date and time.

  Syntax:- **date** [+format]
  date + "%d": Day of month (01-31)
  date + "%h": Abbreviated month name
  date + "%m": Month of year (01-12)
  date + "%D": Date in **%m/%d/%y** format

- **mkdir:-** Used to create directories. Used to create subdirectories alongwith directories.

  Syntax:- **mkdir** directory/subdirectory.

- **cp:-** The cp (copy) command is used to copy a file. Copy file1 to file2, or copy one or more files to the same names under directory. If the destination is an existing file, the file is overwritten; if the destination is an existing directory, the file is copied into the directory (the directory is not overwritten).

  Syntax:- **cp** [filename1] [filename2] files directory

- **ls**:- List contents of directories. If no names are given, list the files in the current directory. With one or more names, list files contained in a directory name or that match a file name.

  Syntax:- ls [names]

  ls-x: Used to produce a multicolumnar output.

  ls -f: Used to identify the executable file and directories. * and / are type indicators. The '*'indicates the file contains executable code and the '/' refers to a directory.

  ls -a: Used to list all the hidden files.

- **mv:-** The command takes filename and pathnames as source names and a filename or exiting directory as target names.

  Syntax:- mv [source-file] [target-file]

- **rm:-** The rm (remove) command is used to delete files from a directory. A number of files may be deleted simultaneously. A file(s) once deleted cannot be retrieved. To remove a file, you must have write permission in the directory that contains the file, but you need not have permission on the file itself. If you do not have write permission on the file, you will be prompted (**y** or **n**) to override.

  Syntax:- rm [filename 1] [filename 2]…

  rmdir directory

- **head:-** Print the first few lines (default is 10) of one or more files. If files is missing or -, read from standard input. With more than one file, print a header for each file.

  Syntax:- head [options] [filename]

  head -c [filename]: To display the contents of the first c lines in the file

- **tail:-** The tail command may be used to view the end of a file. Print the last 10 lines of each named file (or standard input if **-** is specified) on standard output. If more than one file is specified, the output includes a header at the beginning of each file.

  Syntax:- tail [options] [filename]

  tail -c [filename]: To display the contents of the last c lines in the file

- **pwd:-** (**print working directory**). Print the full pathname of the current working directory.

  Syntax:- pwd

- **Uname:-** (**User Name**). Print information about the machine and operating system. Without options, print the name of the kernel (Linux).

  Syntax:- Uname

- **bc:-** To generate a calculator.

  Syntax:- bc

- **tput clear:-** Clears the screen.

  Syntax:- tput clear

- **cd:-** Used to change directories.

  Syntax:- cd [directory]

- **who:-** Displays information about all the users currently logged onto the system. The user name, terminal number and the date and time that each user logged onto the system.

  Syntax:- who am i

- **echo:-** Send the input string to standard output.

  Syntax:- echo [arguments]

- **expr:-** expr (command) command is used for numeric computation. The operators + (add), -(subtract), *(multiplye), /(divide), (remainder) are allowed. Calculations are performed in order of normal numeric precedence.

  Syntax:- expr `[expression]`

- **break:-** It is used to break out of a loop, **while** or **for** statement. It does not exit from the program.

  Syntax:- break

- **vi:-** A screen-oriented text editor based on **ex**. **vi** is bi-modal, with a command mode and an insert mode.

  Syntax:- vi [filename]

- **sh:-** The standard Unix shell, a command interpreter into which all other commands are entered. Used to display ouput written in vi editor.

Syntax:- sh [filename]

- **chmod:-** Allows file permissions to be changed for each user. File permissions can be changed only by the owner (s).

  Syntax:- chmod [u + x][filename]: u is user and x is permission

- **ps:-** Gives information about all the active processes.

  Syntax:- ps

- **cut:-** Used to extract the specific column with a list of column numbers, delimited by a comma. Range can be used using hyphen. Cut out selected columns or fields from one or more files. In the following options, list is a sequence of integers. Use a comma between separate values, and a hyphen to specify a range.

  Syntax:- cut -c [field1] [- field2], [field3] [- field4] [filename]

- **grep:-** The grep (**global regular expression and print**) command can be used as a filter to search for strings in files. The pattern may be either a fixed character string or a regular expression.

  Syntax:- grep [pattern] [filename(s)]

- **tee:-** It handles a character stream by duplicating its input.it saves one copy in a file and writes the other to standard output.

  Syntax:- tee [filesname]

- **wc:-** The wc command can be used to count the number of lines, words and characters in a file.

  Syntax:- wc –[options] [filename]
  wc –l [filename]: For number of lines
  wc –w [filename]: For number of words
  wc –c [filename]: For number of charcters

- **sort:-** Used to order the files. It identifies the field and sort on specified fields by default list s sorted in alphabetical order.

  Syntax:- sort [filename(s)]

EXPERIMENT-5

SHELL PROGRAMMING

The simplest shell script is a sequence of Linux commands, but when you add the power of variables and flow control, you can do a lot more with it. Shell scripts are similar to DOS batch files (those files that end in **.bat**), but shell scripts are more powerful and actually easier to create.

Shell scripts are interpreted, which means that the shell reads each line and acts on it immediately. This process differs from that of a formal programming language like C or C++.

Creating a Shell Script

To create a shell script, use a text editor and enter your Linux commands as if you were typing them at the command prompt. For example, try this:

```
cd /tmp
echo "Removing temp files..."
ls -al
rm junk*
```

If you save those four lines in a file named **deltemp**, you will have a simple shell script that automates the process of switching to the **/tmp** directory, listing all the files there, and deleting the ones that start with the word *junk.*

Writing of shell script

By comparison to most programming language,shellscripts are relatively "highlevel",since each command is infact a program itself.Thus, shell script can be extremely powerful, but may not be efficient for performing complicated calculations because of the overheads involved.

UNIX SHELLS

The bourne shell lacks job control, but it has good support for shell programming, its standard on all unix systems, and it is the basis for the shell in the POSIX standard. On some linux systems, SH does not actually exist.
Csh introduced job control.it is good for interactive use, but it is bad for writing shell scripts.(eg nested if don't always work)
The korn shell(ksh) is an extension of sh;it is suitable for both interactive use and shell programming.bash is aclone of korn shell.

The bourn shell and the korn shells are named after their authors, steve bourne and david korn.korn shell is proprietary to AT&T.

Unix allows the user to choose a shell.Most operating system have command interpreter built into them, and users have no choice but to use that command interpreter on Unix, on the otherhand, the command interpreter is not a part of o.s, but a wrapper around it , which is where the name "shell" comes from.

PROGRAM 1
WRITE A PROGRAM TO ADD TWO NUMBERS

```
echo enter 1
read a
echo enter 2
read b
c=`expr $a + $b`
echo addition = $c
```

OUTPUT :

```
enter 1
7
enter 2
3
addition =10
```

EXPERIMENT-9

WRITE A PROGRAM TO FIND LARGEST OF THREE NUMBERS

```
echo enter 1
read n1
echo enter 2
read n2
echo enter 3
read n3

if [ $n1 -gt $n2 ]&&[ $n1 -gt $n3 ]
 then     echo $a is big
elif [ $n2 -gt $n3 ]&&[ $n2 -gt $n3 ]
 then    echo $b is big
 else    echo $c is big
fi
```

OUTPUT

```
enter 1
4
enter 2
3
enter 3
2
4 is big
```

EXPERIMENT-8

WRITE A PROGRAM TO FIND CURRENT DATE AND DIRECTORY

```
echo current date=`date`
echo user =`who am i`
echo current dir =`pwd`
```

OUTPUT

current date =Fri Nov 16 13:12:25 IST 2007
user =localhost.localdomain!user5 pts/1 Nov 16 12:48 (192.168.1.46)
current dir =/home/user5

EXPERIMENT NO. 7

WRITE A SHELL PROGRAM TO PERFORM OPERATIONS USING CASE STATEMENT AS
A)ADDITION
B)SUBSTRACTION
C)MULTIPLICATION
D)DIVISION

```
echo  a b
read a b

echo a= add
echo b= sub
echo c= mul
echo d= div

echo ch
 read ch

case $ch  in
a)
let z= $a + $b
echo add= $z
 ;;

b)
 let z= $a - $b
echo sub=$z
 ;;

c)
let z= $a * $b
echo mul= $z
 ;;
```

```
d)
let z= $a / $b
echo div= $z
;;

*)
echo invalid option
;;
Esac
```

OUTPUT

```
a b
3 4

a= add
 b= sub
 c= mul
 d= div


ch
a
add = 7
```

EXPERIMENT NO. 10

STUDY OF DOS WITH INTERNAL & EXTERNAL COMMANDS

In DOS there are two types of commands. An Internal command, which is a command embedded into the command.com file, and an external command, which is not embedded into command.com and therefore requires a separate file to be used.

Internal Commands:

- **CHDIR , CD :-**

  These are used to change current working directory.

  Syntax:

```
C:\PROG>cd qbasic
C:\PROG\QBASIC>
```

- **MKDIR ,MD :-**

  These are used to make directory.

  Syntax:

  md directory

- REN:-

  It renames a file.

  Syntax:

ren filename newname

- **CLS:-**

  Clears the screen.

  Equivalent to the Unix clear.

- **COPY:-**

  Copies files from one location to another. The destination defaults to the current directory. If multiple source files are indicated, the destination must be a directory, or an error will result.
  Syntax:

  copy from [source\filename] to [destination\folder]

  Files may be copied to devices. For example, copy *file* lpt1 sends the *file* to the printer on LPT1. copy *file* con outputs *file* to the screen ("console"), which can also be done using type *file*

- **DEVICE**: The terminal device to be used

  Example of usage:
  Ctty com1

- **DEL:-**

  This command is used to delete a particular or more files.

  Syntax:

  del filename

- **DIR:-**

  It is used to list the contents of a directory.

  Syntax:

  dir [drive:][path][filename] [parameters]

- **ECHO:-**

  Prints its own arguments back to DOS.

```
echo this is text        Outputs 'this is text'
```

- **EXIT:-**

  Exits the current command processor. If the exit is used at the primary command, it has no effect unless in a DOS window under Microsoft Windows, in which case the window is closed and the user returns to the desktop.

```
exit
```

External Commands:-

ASSIGN:-

The command redirects requests for disk operations on one drive to a different drive.

```
assign [x[:]=y[:][...]]
assign /STATUS
```

Options: **x**

The drive letter to reassign.

The drive letter that x: will be assigned to.

ATTRIB:-

Change or view the attributes of one or more files. It defaults to displaying the attributes of all files in the current directory.

ATTRIB [+R|-R] [+A|-A] [+S|-S] [+H|-H][drive:][path][filename] [/S [/D]]

Options:

- To add an attribute attach a '+' in front of it.
- To remove an attribute attach a '-' in front of it

Attributes include

- R - Read-only
- A - Archive
- S - System
- H - Hidden
- /D - Process folders as well.
- /S - Process matching files in the current folder and all subfolders.

- **HELP:-**

It helps about DOS.

Help ‘command’ will help about that particular command.

LABEL:-

Changes the label on a logical drive, such as a hard disk partition or a floppy disk. In Unix and Unix-like systems, this differs from filesystem to filesystem. e2label can be used for ext2 partitions.

FORMAT:-

Delete all the files on the disk and reformat it for MS-DOS
In most cases, this should only be used on floppy drives or other removable media.

Syntax:-

format [options] drive

MEM:-

Displays memory usage.

Syntax:-

Mem

- **MODE:-**

It configures system devices, changes graphics mode and adjusts keyboard settings.

- **MOVE:-**

It noves files .

Syntax:

Move filename newname