Unit-1

Operating System Overview

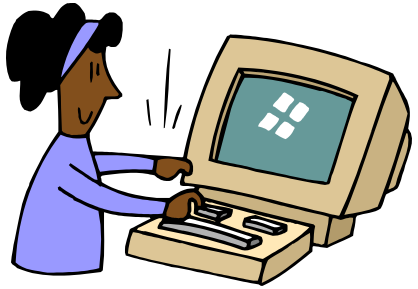
Outline

- What is Operating System (SO)
- Objectives / Goals of Operating System (OS)
- Types of Operating System
- Functions of Operating System
- Evolution/Generations of Operating Systems (OS)
- Operating Systems (OS) Structure:
  - Layered
  - Monolithic
  - Microkernel
- System Calls
- Linux Kernel and Shell

COMPUTER SYSTEM OVERVIEW

Basic Elements of Computer

- Processor
- I/O Modules
- Memory
- System Bus



Processor

Referred to as the ***Central Processing Unit*** **(CPU)**

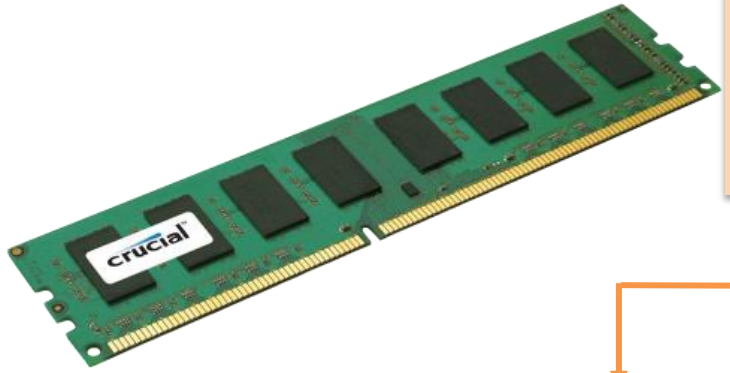
Arithmetic & Logic Unit

Performs the **data processing** functions

Control Unit

Controls the operation of the computer

Memory



Memory is device that is used to store data/information

Primary Memory

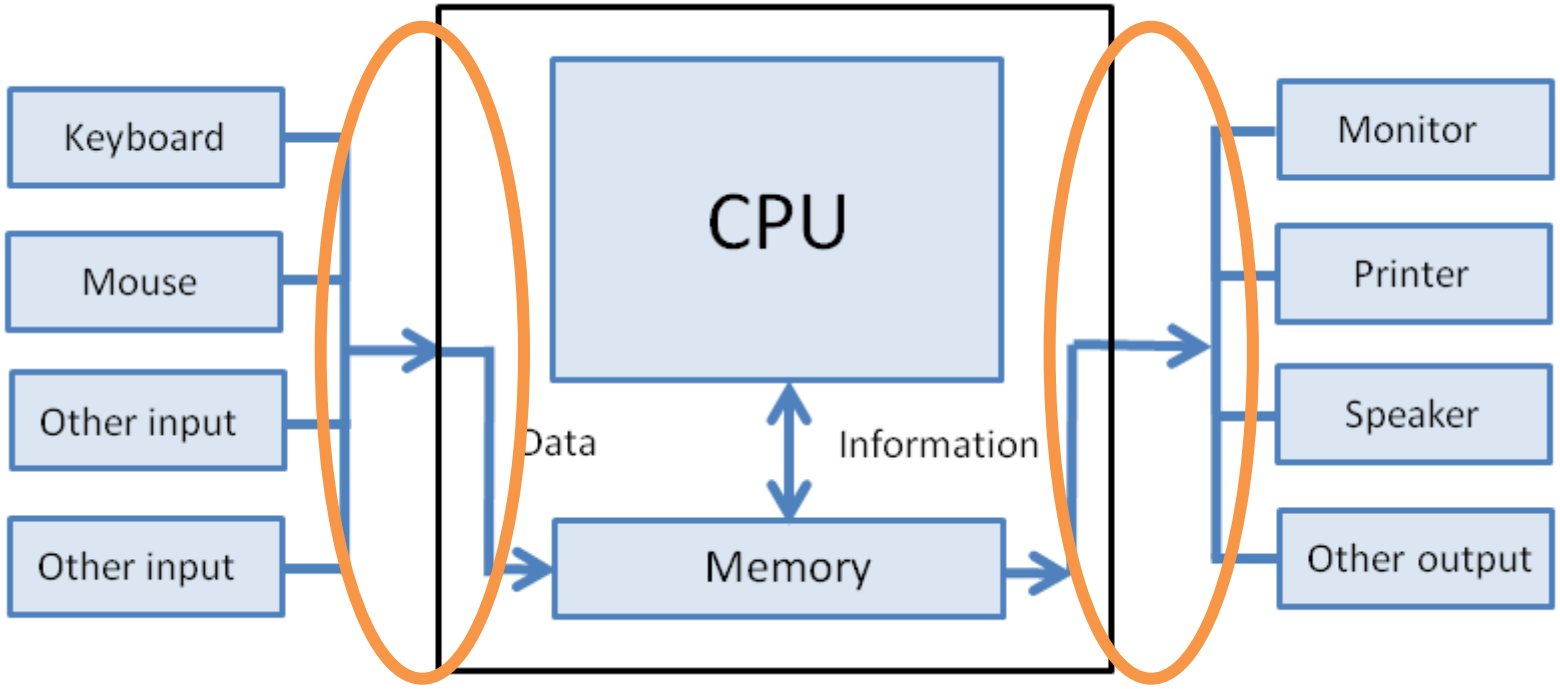
- Volatile
- RAM & ROM

Secondary Memory

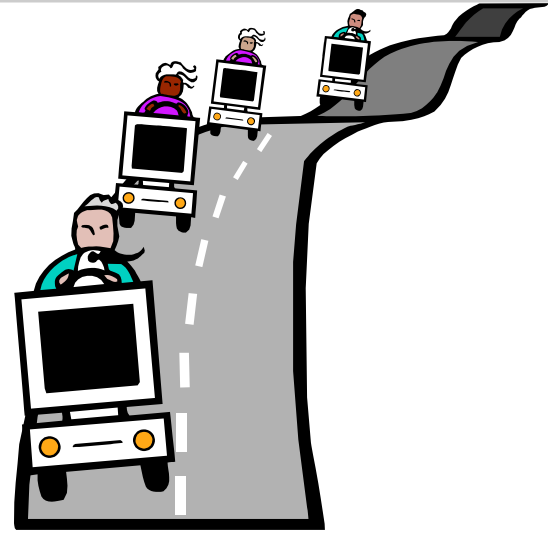
- Non-Volatile
- HDD, CD & DVD

I/O Module (Input/Output Module)

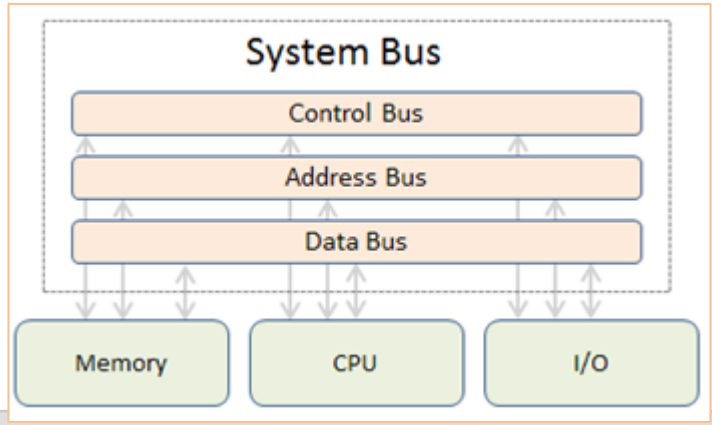
- Input/output module is a **device that acts as the connective bridge between a computer system** at one end **and an I/O or peripheral device** at the other, such as a printer, webcam or scanner.
- It **controls the data exchange** between the **external devices and main memory** or **external devices and CPU registers**.



System Bus

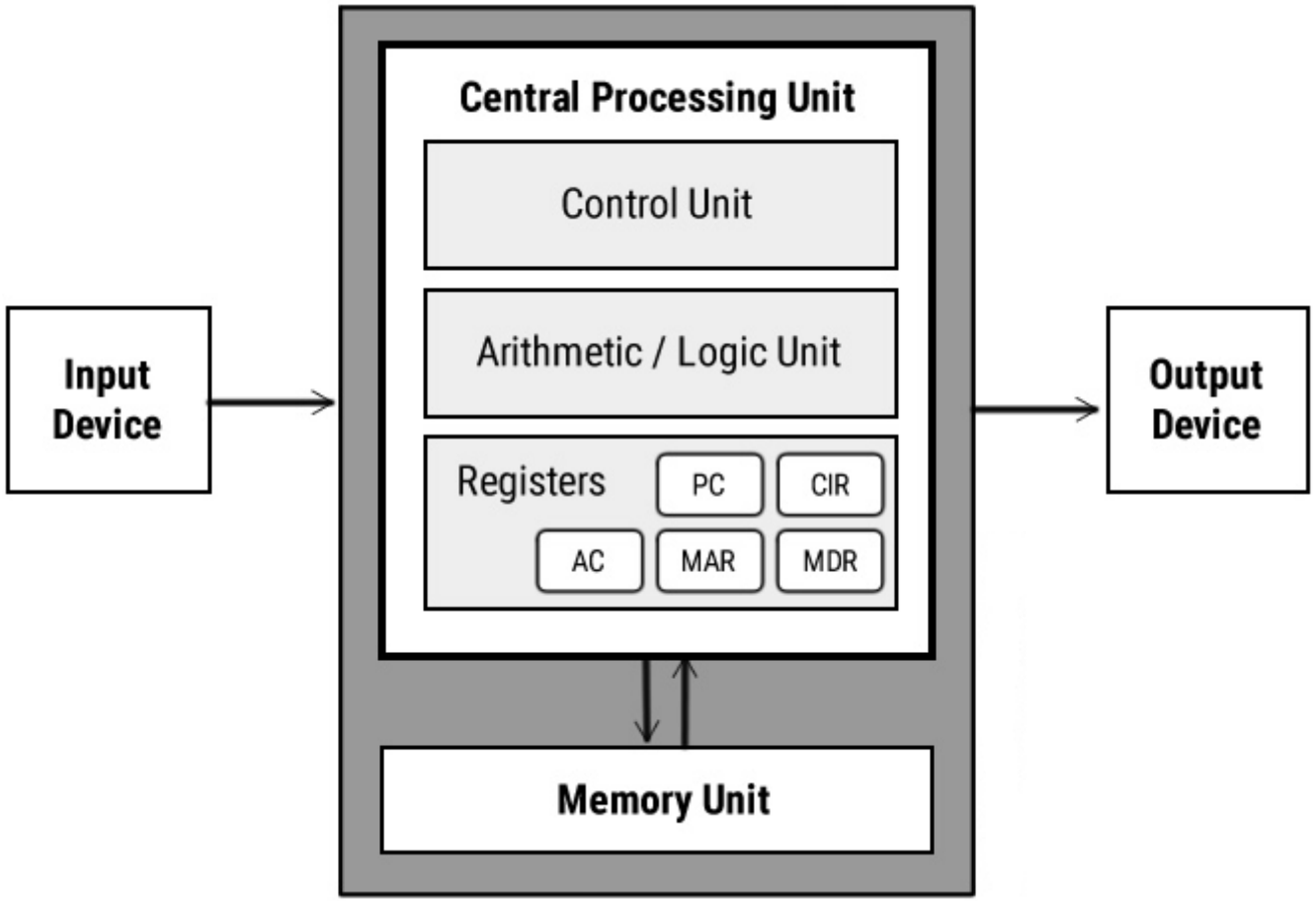


- **Provides communication among processors, main memory, and I/O devices**.
- The system bus is a pathway **composed of cables and connectors used to carry data** between a computer microprocessor and the main memory.
- Types of buses
  - Address bus - **carries memory addresses** from the processor to other components such as primary storage and input/output devices.
  - Data bus - **carries the data** between the processor and other components.
  - Control bus - **carries control signals** from the processor to other components.



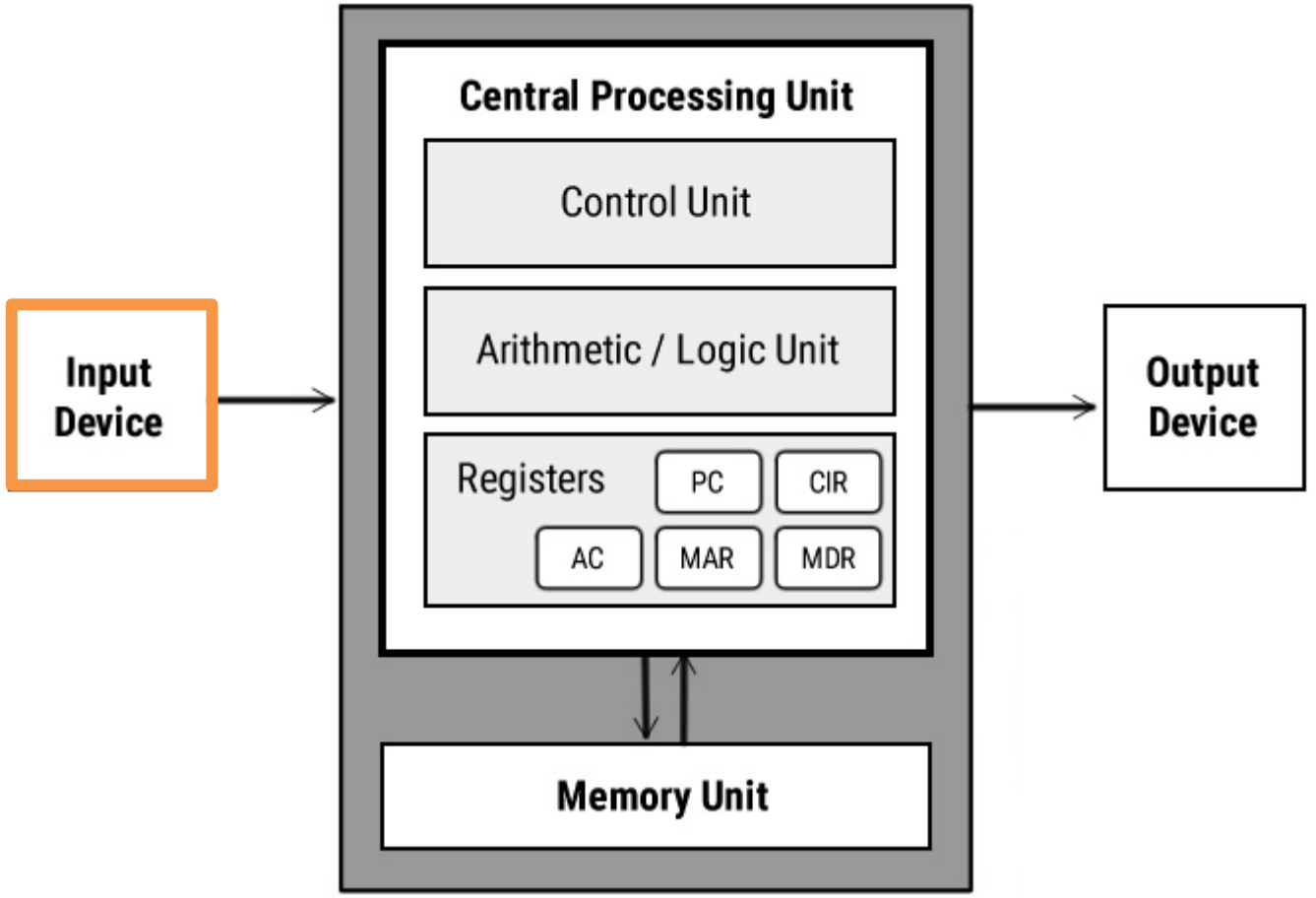
COMPUTER SYSTEM ARCHITECTURE

Computer system architecture



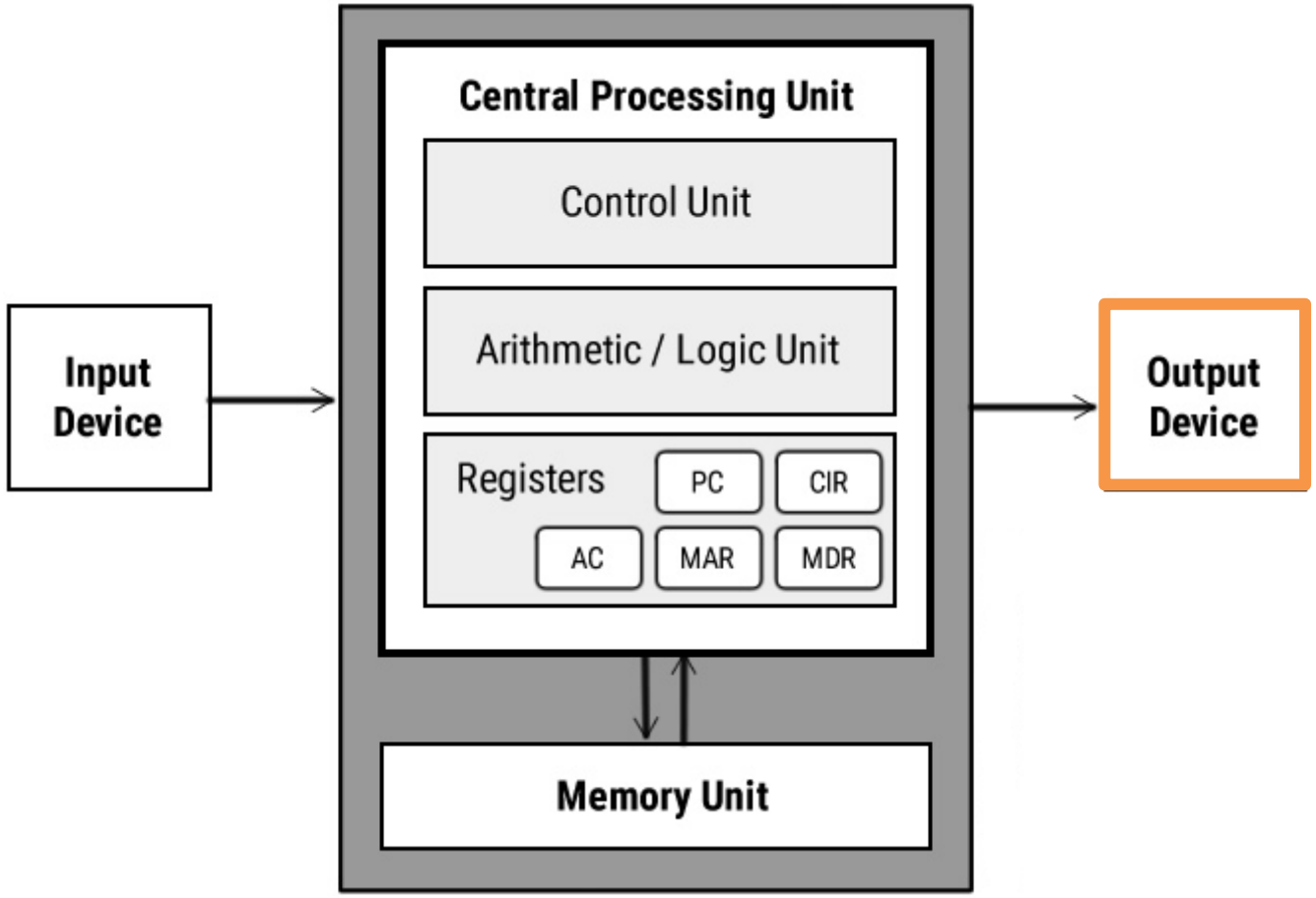
Input unit

- It **provides data** and **instructions** to the computer system.
- Commonly used input devices are keyboard, mouse, magnetic tape etc.
- Input unit performs following tasks:
  - Accept the data and instructions from the outside environment.
  - Convert it into machine language.
  - Supply the converted data to computer system.



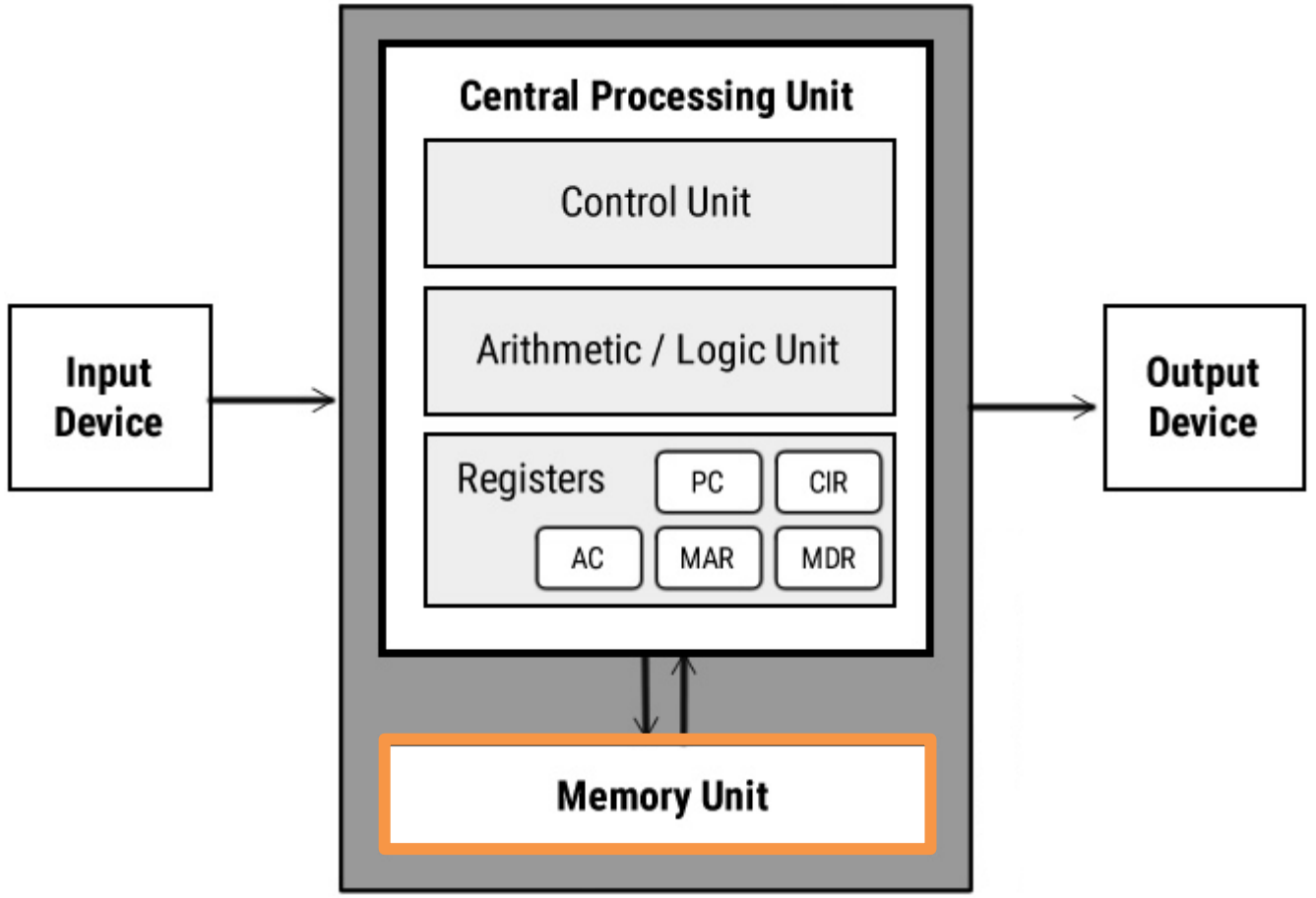
Output unit

- It **connects the internal system** of a computer **to the external environment**.
- It **provides the results** of any computation, or instructions to the outside world.
- Some output devices are printers, monitor etc.



Storage unit

- This unit **holds the data** and **instructions**.
- It also **stores the intermediate results** before these are sent to the output devices.
- It also **stores the data for later use**.
- The storage unit of a computer system can be divided into two categories:
  - Primary Storage
  - Secondary Storage

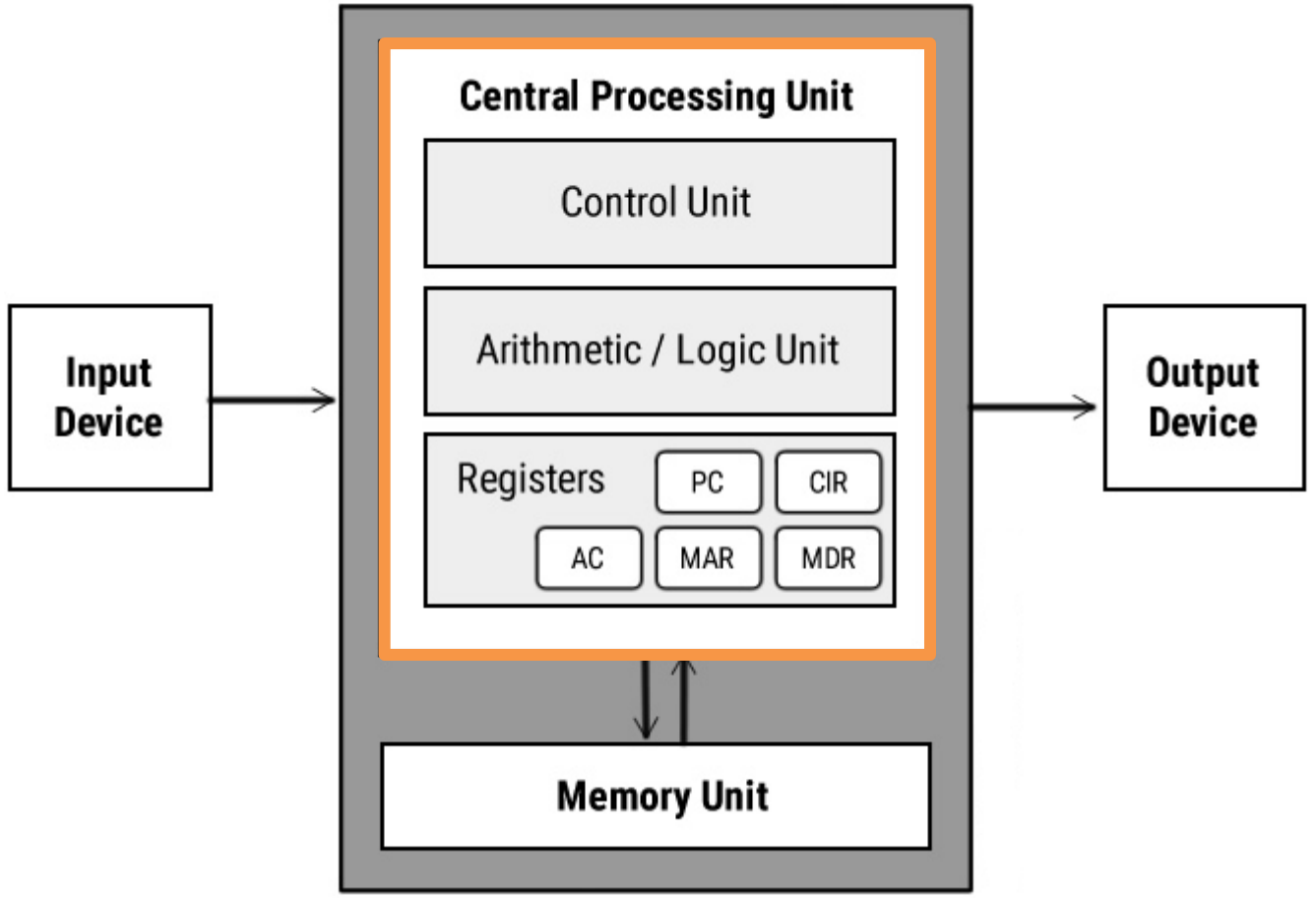


Primary storage (memory) vs Secondary storage (memory)

Primary storage	Secondary storage
Examples: RAM, ROM, Cache memory, PROM, EPROM, Registers, etc.	Examples: Hard Disk, Floppy Disk, Magnetic Tapes, etc.
It is temporary and volatile.	It is permanent and Non-volatile.
Primary memory is directly accessible by Processor/CPU.	Secondary memory is not directly accessible by the CPU.
Primary memory devices are expensive.	Secondary memory devices are cheaper.
The memory devices used for primary memory are semiconductor memories.	The secondary memory devices are magnetic and optical memories.
Primary memory is also known as Main memory or Internal memory.	Secondary memory is also known as External memory or Auxiliary memory.

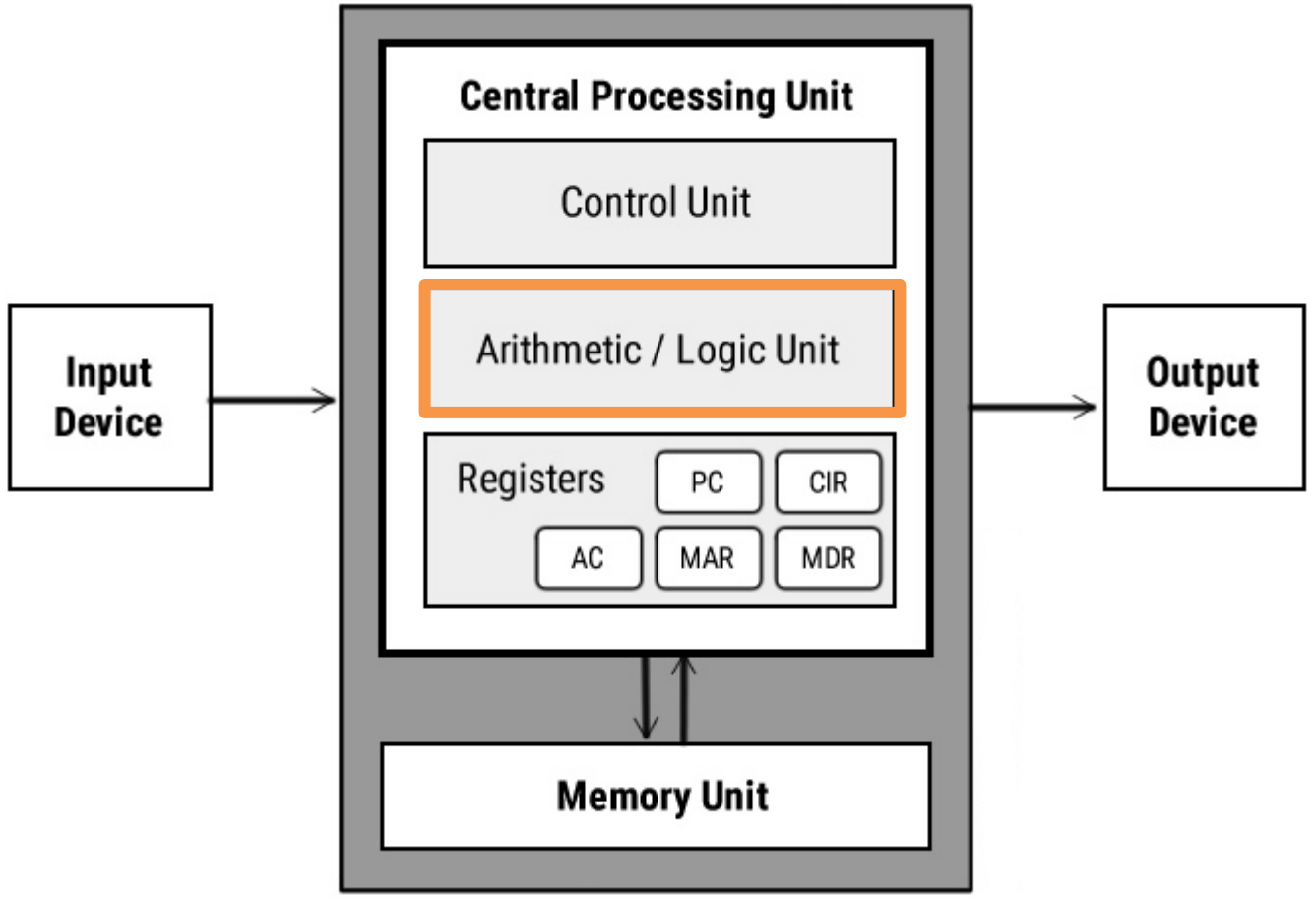
CPU (Central Processing Unit)

- The **Arithmetic Logical Unit and Control Unit are together known as CPU**.
- CPU is the **brain** of computer system.
- It performs following tasks:
  - performs all operations.
  - takes all decisions.
  - controls all the units of computer.



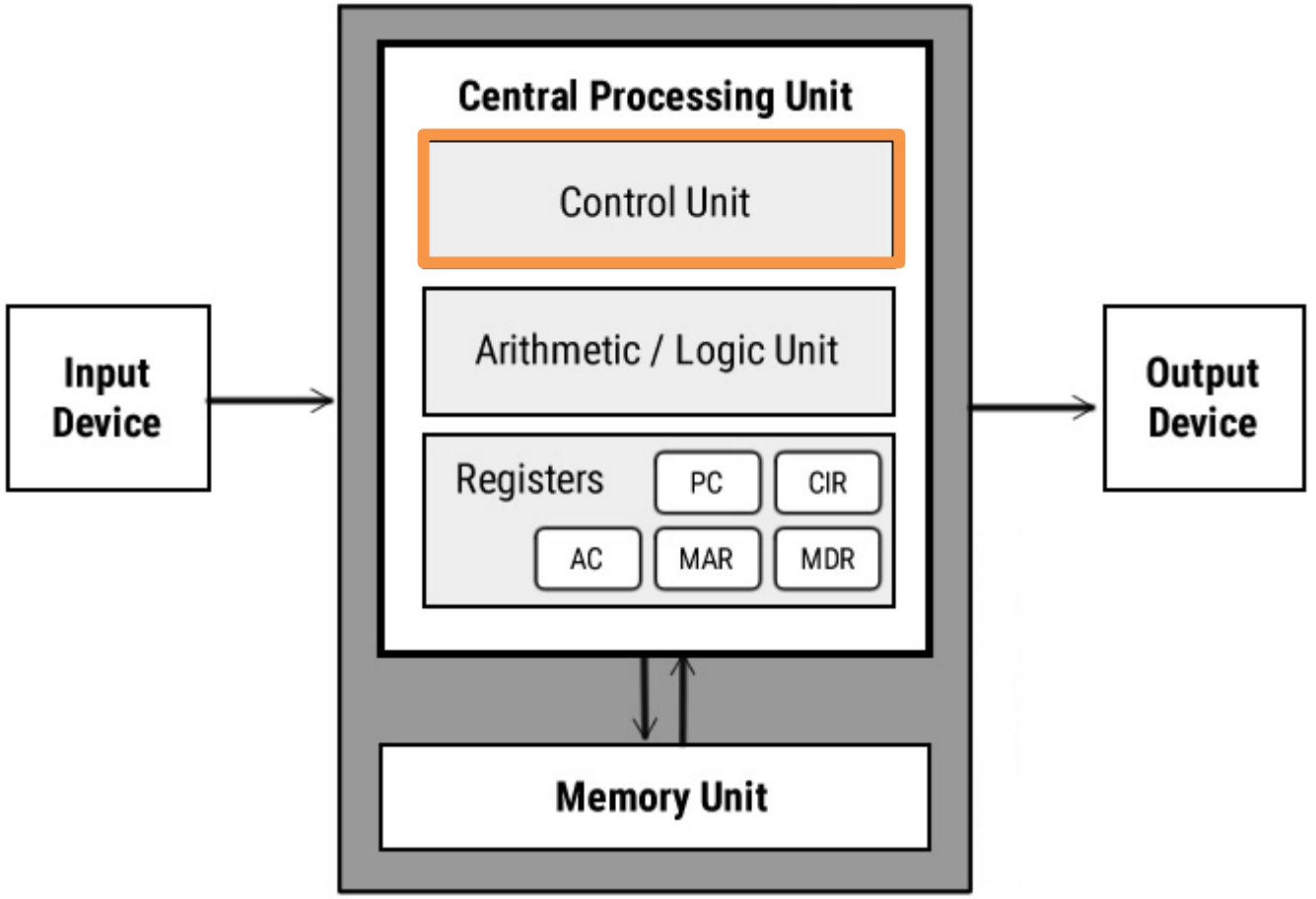
ALU (Arithmetic Logical Unit)

- All the **calculations** are performed in ALU of the computer system.
- The ALU can **perform basic operations** such as addition, subtraction, division, multiplication etc.
- Whenever calculations are required, the **control unit transfers the data from storage unit to ALU**.
- When the operations are done, the result is transferred back to the storage unit.



CU (Control Unit)

- It **controls all other units** of the computer.
- It **controls the flow of data and instructions** to and from the storage unit to ALU.
- Thus it is also known as central nervous system of the computer.



WHAT IS OPERATING SYSTEM (OS)

What is Operating System (OS)?

- A Computer System consists of various hardware's such as

Processor

RAM

Keyboard & Mouse

Hard Disk

Monitor

Printer

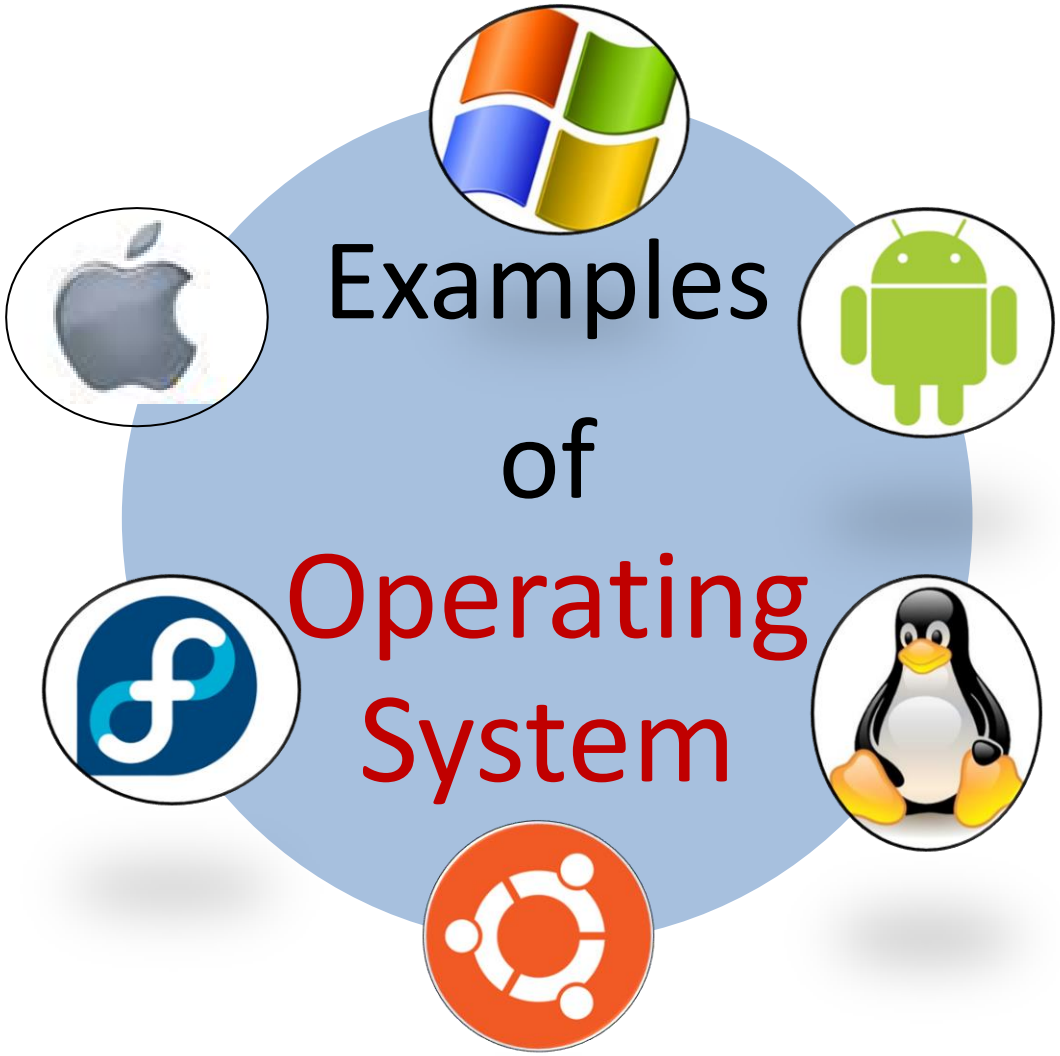
Who manages (controls) these hardwares???

Operating System

Definition of Operating System (OS)

- An Operating System (OS) is a **collection of software** that
  - **manages hardware resources**
  - **provides various service** to the users

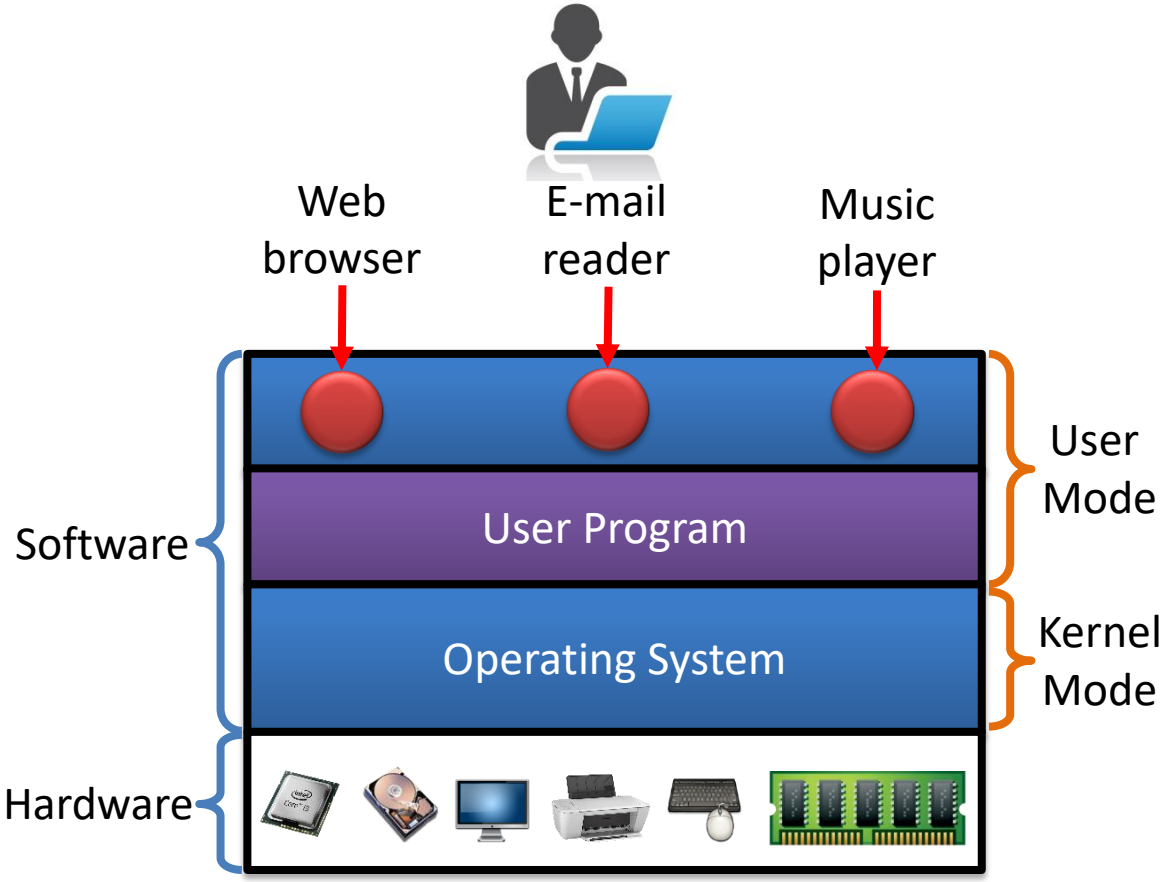
Examples of Operating System (OS)



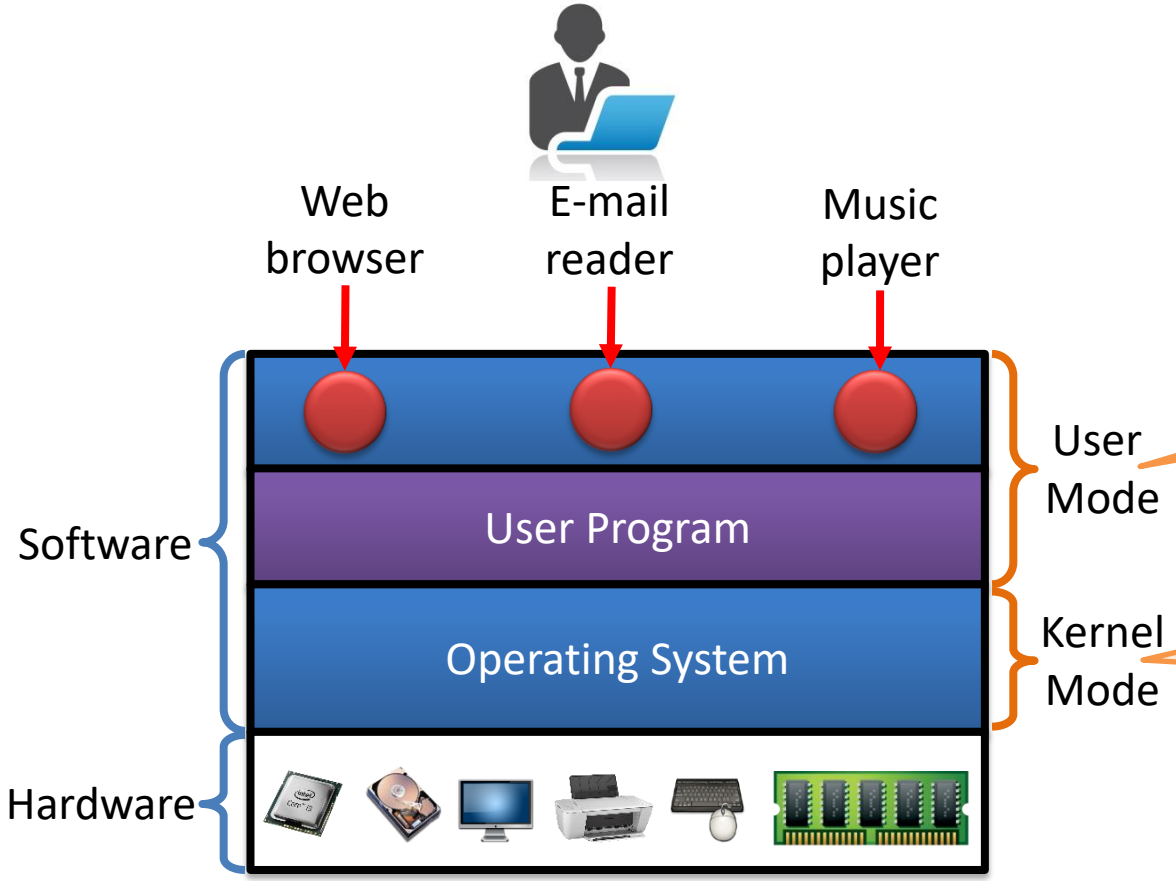
Where OS lies? (Interaction of OS & Hardware)

- OS **lies between hardware and user program**.
- It **acts as an intermediary** between the user and the hardware.
- Modes of operation of computer

1. Kernel Mode
   - has **complete access** to all the hardware
   - can **execute any instruction** that a machine is capable of executing
   - has **high privileged** (rights)
2. User Mode
   - has **limited access** to limited hardware
   - can **execute only subset (few)** of the machine instructions
   - has **less privileged** (rights)



Why and How switch occur?



- User is writing program in C.
- Once finish writing will execute it.

user-mode program performs trap instruction

- To execute this program
  1. Load program into RAM
  2. Program is executed by processor

TYPES OF OPERATING SYSTEM (OS)

Types of Operating System

Type of operating system	Features/benefits	Example	Applicable to which type of application
Batch systems	• More than one job can be stored in main memory • Batches of same type of jobs can be executed quickly	FMS (FORTRAN monitor system), IBM's operating system for 7094	Background jobs in which the user interaction is not necessary
Multiuser systems	• Jobs of different users who are connected to a main computer are executed through the multiprogramming • Interaction of jobs with the user is possible • Debugging is easy	CTSS by MIT, TSS by IBM, MULTICS, UNIX	When multiple users need to share a single system

Types of Operating System

Type of operating system	Features/benefits	Example	Applicable to which type of application
Multitasking systems	• Multiple tasks of a single user can be opened on the system through multiprogramming	Windows	When a user wants to open and work simultaneously on many windows on the system
Network systems	• The user is able to connect to another machine and perform many operations • The user is aware of the location of the network node where he/she wants to connect	Novell Netware, Windows NT, Windows 2000, Windows XP, Sun Solaris	When a user wants to remote login on a system, wants to transfer a file, etc. on a network system

Types of Operating System

Type of operating system	Features/benefits	Example	Applicable to which type of application
Distributed systems	• When multiple nodes of a wide network realized as a powerful machine sharing the resources on the network. • The users are not aware where their processes are being sent and executed.	Amoeba, V system, Chorus	When computational speed and resource sharing is required and implemented through various full computer systems in a network
Real-time systems	Used to handle time-bound responses to the applications	pSOS, VxWorks, RTLinux, etc.	Applicable to systems which require time-bound response, i.e., for the real-time processing systems

Types of Operating System

Type of operating system	Features/benefits	Example	Applicable to which type of application
Embedded systems	Specialized systems with size, memory and power restrictions	Palm Pilot, Toshiba Pocket PC, Palm OS, Symbian OS, iPhone OS, RIM's BlackBerry, Windows Phone, Linux, Palm WebOS, Android and Maemo.	Used in consumer electronics items, mobile phones, smart cards, etc.

OBJECTIVES / GOALS OF OPERATING SYSTEM (OS)

Objectives / Goals of Operating System (OS)

- Make the computer system **convenient to use in an efficient manner**.
- **Hide the details of the hardware** resources from the users.
- **Provide** users a **convenient interface** to use the computer system.
- **Act as an intermediary** between the **hardware and its users**, making it easier for the users to access and use other resources.
- **Manage the resources** of a computer system.
- **Keep track** of who is using which resource, granting resource requests, and mediating conflicting requests from different programs and users.
- **Provide efficient** and **fair sharing of resources** among users and programs.

Section - 6

FUNCTIONS OF OPERATING SYSTEMS (OS)

Services / Functions / Tasks of Operating System (OS)

1. User Interface
   - This interface can take several forms.
     1. DTrace command line interface - which uses text commands and a method for entering them (say, a program to allow entering and editing of commands).
     2. Batch interface - in which commands and directives to control those commands are entered into files, and those files are executed.
     3. Graphical user interface - the interface with a pointing device to direct I/0, choose from menus, and make selections and a keyboard to enter text.
   - Some systems provide two or all three of these variations.

Services / Functions / Tasks of Operating System (OS)

2. Program Execution
   - The system must be able to load a program into memory and to run that program.
   - The program must be able to end its execution, either normally or abnormally (indicating error).

3. I/O Operations
   - A running program may require I/O, which may involve a file or an I/O device.
   - For efficiency and protection, users usually cannot control I/O devices directly.
   - Therefore, the operating system must provide a means to do I/O

Services / Functions / Tasks of Operating System (OS)

4. File Management
   - The operating system manages resource allocation and de-allocation.
   - It specifies which process receives the file and for how long.
   - It also keeps track of information, location, uses, status, and so on.
   - These groupings of resources are referred to as file systems.
   - The files on a system are stored in different directories.
   - The OS:
     - Keeps records of the status and locations of files.
     - Allocates and deallocates resources.
     - Decides who gets the resources.

Services / Functions / Tasks of Operating System (OS)

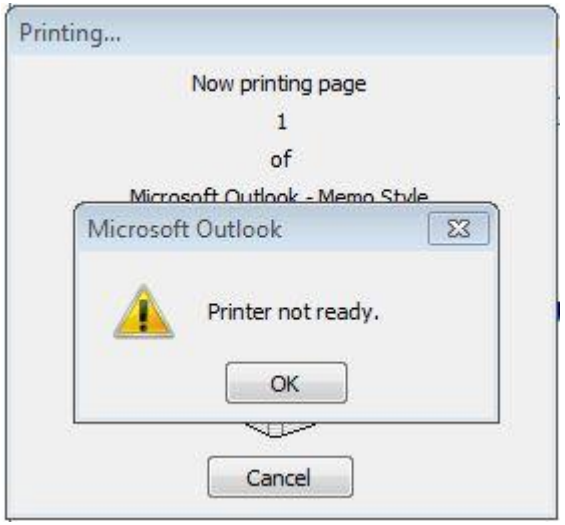
5. Resource Allocation
   - When there are multiple users or multiple jobs running at the same time, resources must be allocated to each of them.
   - The operating system manages many different types of resources, such as CPU cycles, main memory, file storage, I/O devices.

Services / Functions / Tasks of Operating System (OS)

6. Communication
   - In multitasking environment, the processes need to communicate with each other and to exchange their information.
   - Operating system performs the communication among various types of processes in the form of shared memory or message passing, in which packets of information in predefined formats are moved between processes by the operating system.

Services / Functions / Tasks of Operating System (OS)

7. Error detection and response
   - An error may occur in CPU, in I/O devices or in the memory hardware.
   - Following are the major activities of an operating system with respect to error handling –
     - The OS constantly checks for possible errors.
     - The OS takes an appropriate action to ensure correct and consistent computing.



Services / Functions / Tasks of Operating System (OS)

8. Accounting
   - We want to keep track of which users use how much and what kinds of computer resources.
   - This record keeping may be used for accounting (so that users can be billed) or simply for accumulating usage statistics.

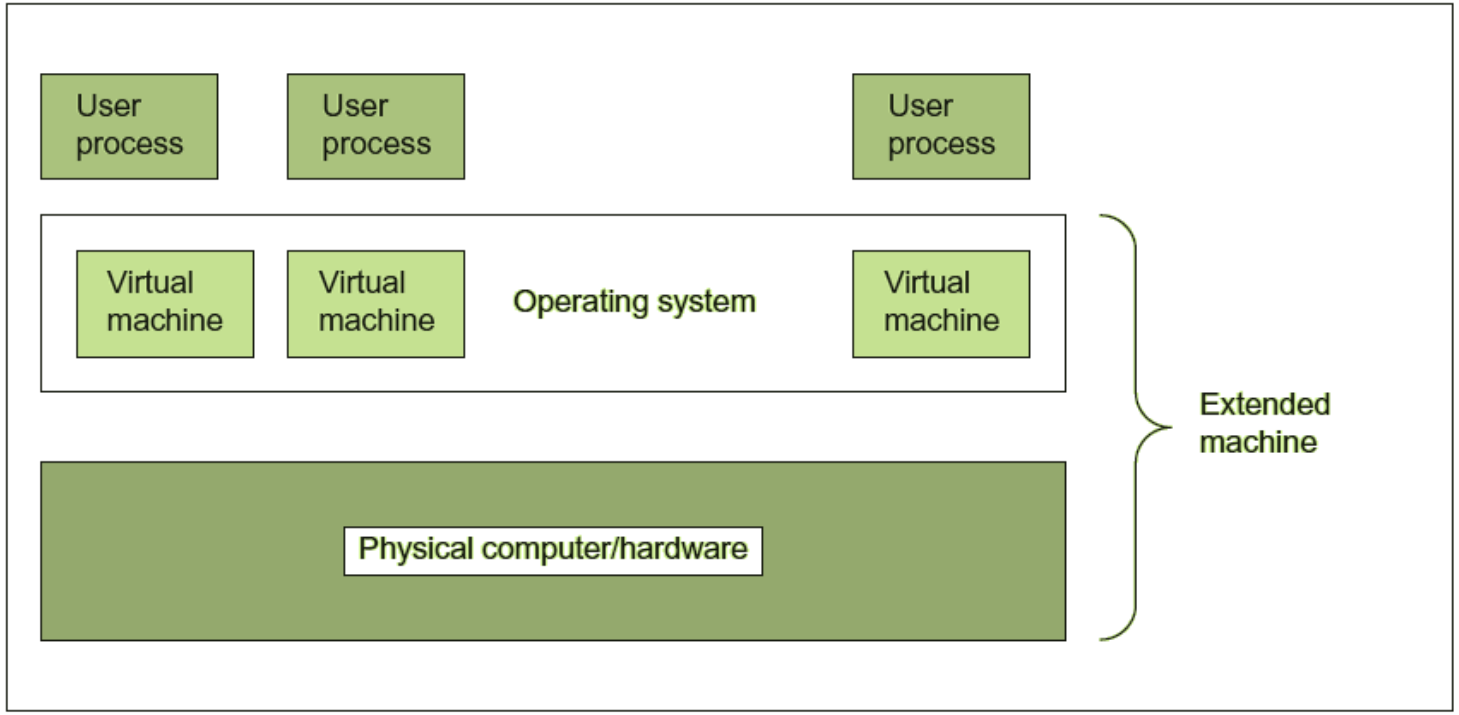
Services / Functions / Tasks of Operating System (OS)

9. Protection & Security
   - Protection involves ensuring that all access to system resources is controlled.
   - Security of the system from outsiders is also important.
   - Such security starts with requiring each user to authenticate himself or herself to the system, usually by means of a password, to gain access to system resources.
   - If a system is to be protected and secure, precautions must be instituted throughout it.

Services / Functions / Tasks of Operating System (OS)

10. Virtual Machine Manager

- The fundamental idea behind a virtual machine is to abstract the hardware of a single computer (the CPU, memory, disk drives, network interface cards, and so forth) into several different execution environments, thereby creating the illusion that each separate execution environment is running its own private computer.



GENERATIONS/EVOLUTION OF OPERATING SYSTEMS (OS)

Evolution / Generations of Operating System (OS)

Generation	Period	Computer architecture	Problems and development of Operating systems
First	1940s–1950s	• Vacuum tubes based technology. • They used punch cards for input. The outputs were always given as printouts on paper tapes.	No operating system

Vacuum tubes

Vacuum tubes are electron tubes that control the motion of electrons and hence electrical current in a high vacuum space. Vacuum tubes also help in amplifying the power of signals using vacuum tubes. These tubes used a lot of electricity and produced a lot of heat.

Evolution / Generations of Operating System (OS)

Generation	Period	Computer architecture	Problems and development of Operating systems
Second	1950s–1960s	• Transistors based technology. • Magnetic tapes were used as storage devices. They use punch cards for inputs and printouts for outputs. • Assembly language and high-level programming languages like FORTRAN, COBOL were used.	• Set up delay problem due to loading and unloading of tapes. • CPU was idle. • Jobs of users prepared with same programming language were batched together. • Batch systems • Mismatch between the speed of CPU and I/O devices, since in • I/O devices were mostly electromechanical. On the other hand, the CPU was an electronic device.

Evolution / Generations of Operating System (OS)

Generation	Period	Computer architecture	Problems and development of Operating systems
Third	1960s–1980s	• IC based technology. • Magnetic disk for storage was used.	• Hard disks came into existence • Multiprogramming • Multiprogrammed batch systems • Lack of user/programmer interaction with their jobs in multiprogrammed batch systems • Timesharing multiuser systems

Evolution / Generations of Operating System (OS)

Generation	Period	Computer architecture	Problems and development of Operating systems
Fourth	1980s–Present	• LSI(Large Scale Integration) and VLSI(Very Large Scale Integration) based technology. • Personal Computers • Processing time of the CPU increased	• MS-DOS • Multiuser facilities were not there in DOS • XENIX • OS/2 • No user friendliness and convenience due to command driven and complex file systems • Apple Macintosh (GUI) • Windows (GUI) • Multitasking • Network operating systems • Distributed operating systems

Section - 9

SYSTEM CALLS

What is System Call?

- A system call is a **way for programs to interact with the operating system**.
- A system call is a **mechanism that provides the interface between a process and the operating system**.
- A computer program makes a system call **when it makes a request to the operating system's kernel**.
- It is a **programmatic method** in which a **computer program requests a service from the kernel** of the OS.
- System call **provides the services of the operating system** to the user programs via **Application Program Interface(API)**.
- System calls are the **only entry points for the kernel system**.

Types of system calls

- **Process Control**: This system calls perform the task of **process creation, process termination**, etc.
  - Functions:
    - End and abort
    - Load and execute
    - Create process and terminate process
    - Wait and signed event
    - Allocate and free memory
- **File Management**: File management system calls handle **file manipulation jobs like creating a file, reading, and writing**, etc.
  - Functions:
    - Create a file
    - Delete file
    - Open and close file
    - Read, write and reposition
    - Get and set file attributes

Types of system calls

- **Device Management**: Device management does the job of **device manipulation like reading from device buffers, writing into device buffers**, etc.
  - Functions
    - Request and release device
    - Logically attach/ detach devices
    - Get and Set device attributes
- **Information Maintenance**: It **handles information** and its **transfer between the OS and user program**.
  - Functions:
    - Get or set time and date
    - Get process and device attributes

Types of system calls

- **Communication**: These types of system calls are specially used for **interprocess communications (IPC)**.
  - Functions:
    - Create, delete communications connections
    - Send, receive message
    - Help OS to transfer status information
    - Attach or detach remote devices

Section - 10

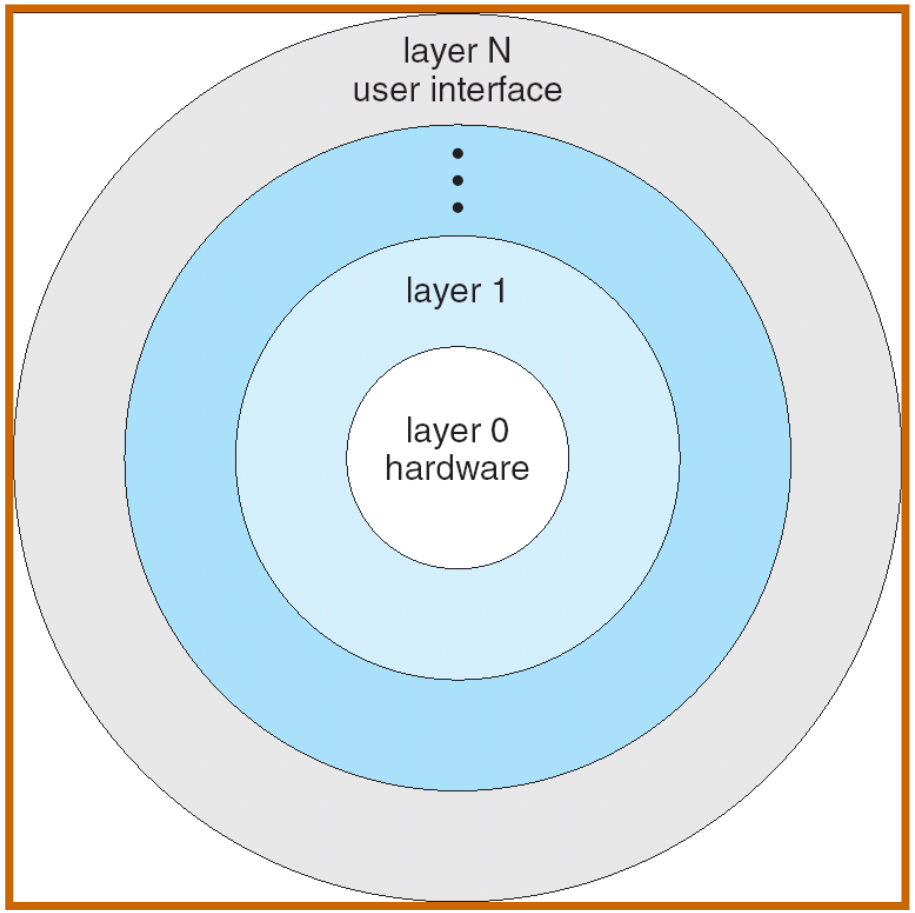
OPERATING SYSTEMS (OS) STRUCTURE

Operating Systems (OS) structure

1. Layered System
2. Monolithic System
3. Microkernel

Layered System

- In this system, the OS is organized as a **hierarchy of layers**.
- In a layered approach, the operating system is divided into a number of layers (levels), each built on top of lower layers.
- The bottom layer (layer 0), is the hardware; the highest (layer N) is the user interface.
- **Advantage:**
  - Each layer is independent of each other.
- **Disadvantage:**
  - There might be a negative impact on the performance as we have the extra overhead of passing through layers instead of calling a component directly.

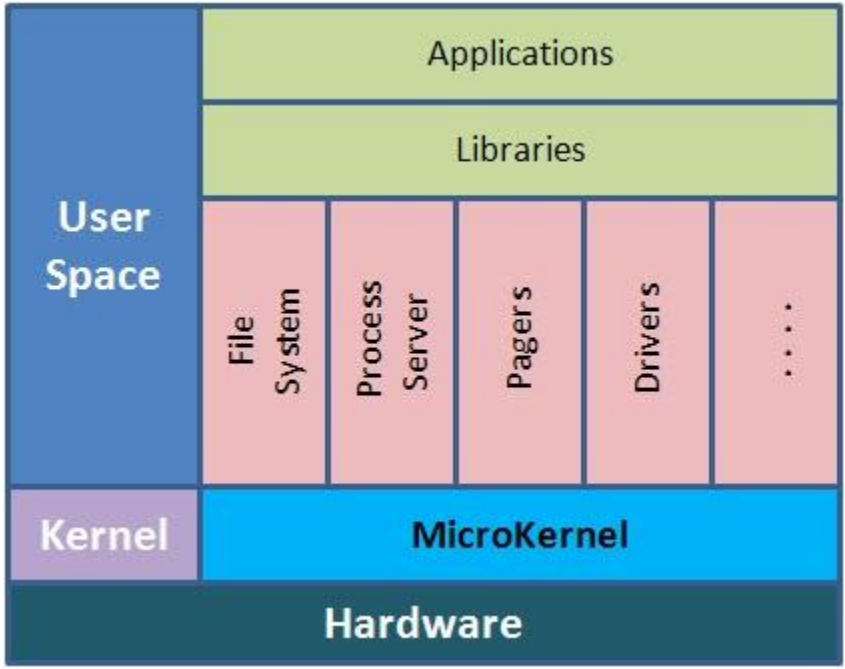


Monolithic Systems

- *A large kernel containing the complete OS.*
- A **monolithic kernel** is an operating system architecture where the entire operating system is working in kernel space. This increases the size of the kernel as well as the operating system.
- **Advantages of Monolithic Kernel**
  - The execution of the monolithic kernel is quite fast as the services such as memory management, file management, process scheduling etc., are implemented under the same address space.
  - A process runs completely in a single address space in the monolithic kernel.
- **Disadvantages of Monolithic Kernel**
  - If any service fails in the monolithic kernel, it leads to the failure of the entire system.
  - To add any new service, the entire operating system needs to be modified by the user.

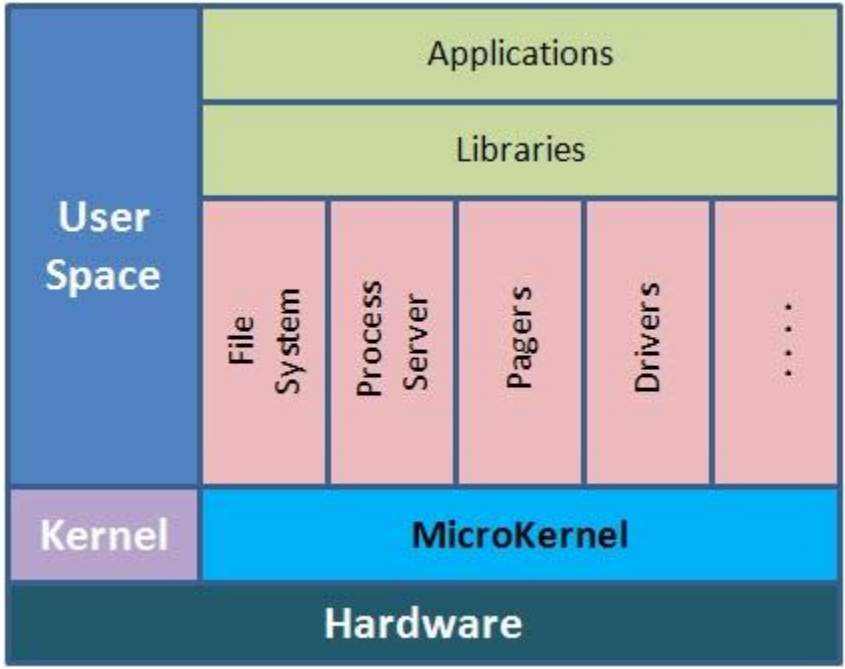
Microkernel

- The microkernel design **splits OS up into small well defined modules, only one module run in kernel and rest of all run in user mode**.
- This results in smaller kernel space.
- The main function of the microkernel is to provide communication facility between the client program and various services.
- Communication is provided by message passing.



Microkernel

- Advantages
  - Provides ease of extending the OS as all new services are added to user space and consequently do not require modification of the kernel.
  - Microkernel also provides more security and reliability, since most services are running as user rather than kernel processes.
  - If the service fails rest of the OS remains untouched.
- Disadvantages
  - Poor Performance due to increased overhead from message passing.



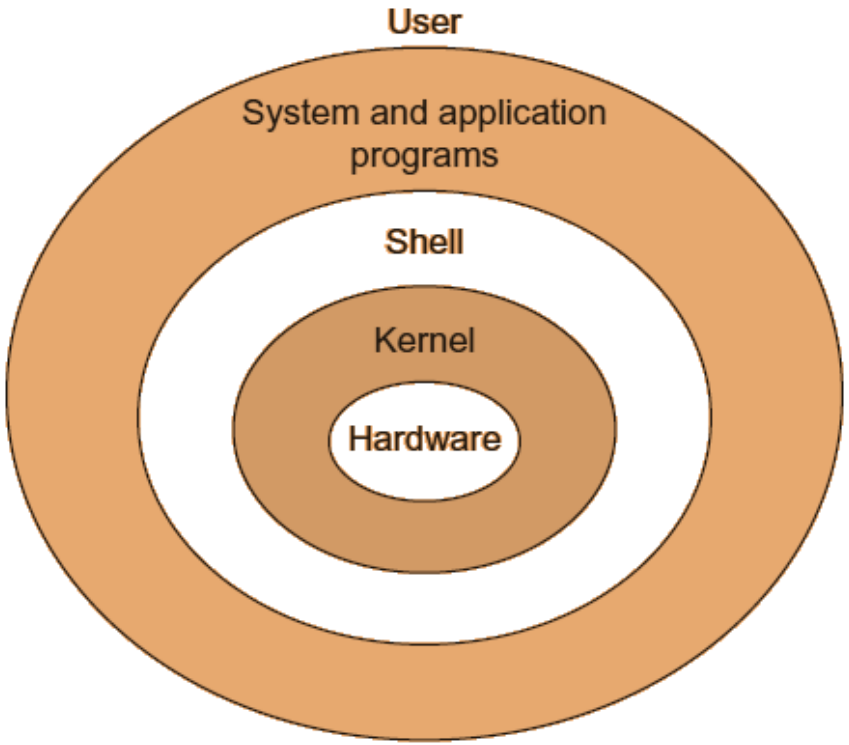
Differences Between Microkernel and Monolithic Kernel

Parameters	Monolithic kernel	MicroKernel
Basic	It is a large process running in a single address space	It can be broken down into separate processes called servers.
Code	In order to write a monolithic kernel, less code is required.	In order to write a microkernel, more code is required
Security	If a service crashes, the whole system collapses in a monolithic kernel.	If a service crashes, it never affects the working of a microkernel.
Communication	It is a single static binary file	Servers communicate through IPC.
Example	Linux, BSDs, Microsoft Windows (95,98, Me), Solaris, OS-9, AIX, DOS, XTS-400, etc.	L4Linux, QNX, SymbianK42, Mac OS X, Integrity, etc.

Linux Kernel and Shell

Kernel

- The kernel is one of the core section of an operating system.
- It is responsible for each of the major actions of the Linux OS.
- The kernel facilitates required abstraction for hiding details of low-level hardware or application programs to the system.



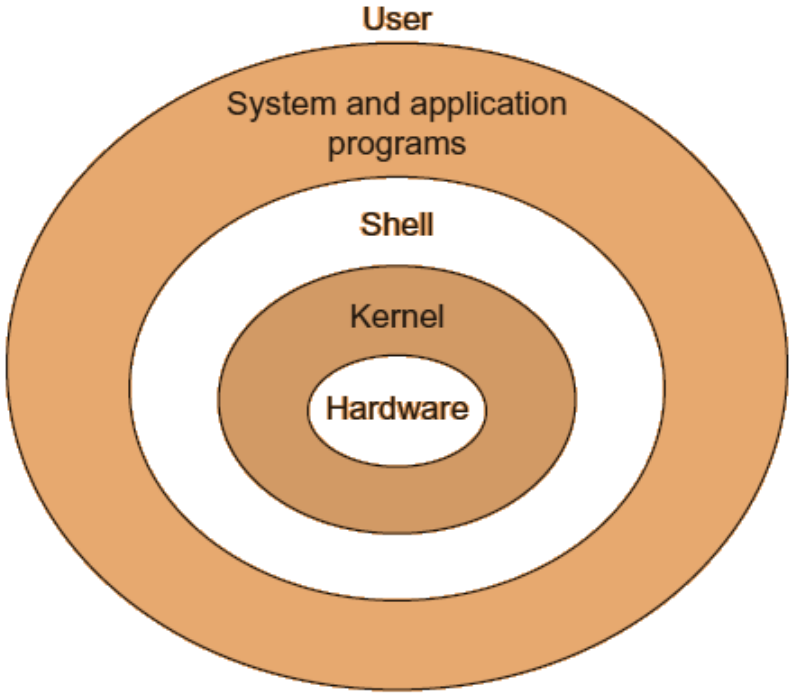
Hardware layer

- Linux operating system contains a hardware layer that consists of several peripheral devices like CPU, HDD, and RAM.

Linux Kernel and Shell

Shell

- It is an interface among the kernel and user.
- It can take commands through the user and runs the functions of the kernel.
- The shell is available in distinct types of OSes. These operating systems are categorized into two different types, which are the **graphical shells** and **command-line shells**.
- The graphical line shells facilitate the graphical user interface, while the command line shells facilitate the command line interface.
- Thus, both of these shells implement operations.
- However, the graphical user interface shells work slower as compared to the command-line interface shells.



Thank You