Introduction to Operating System

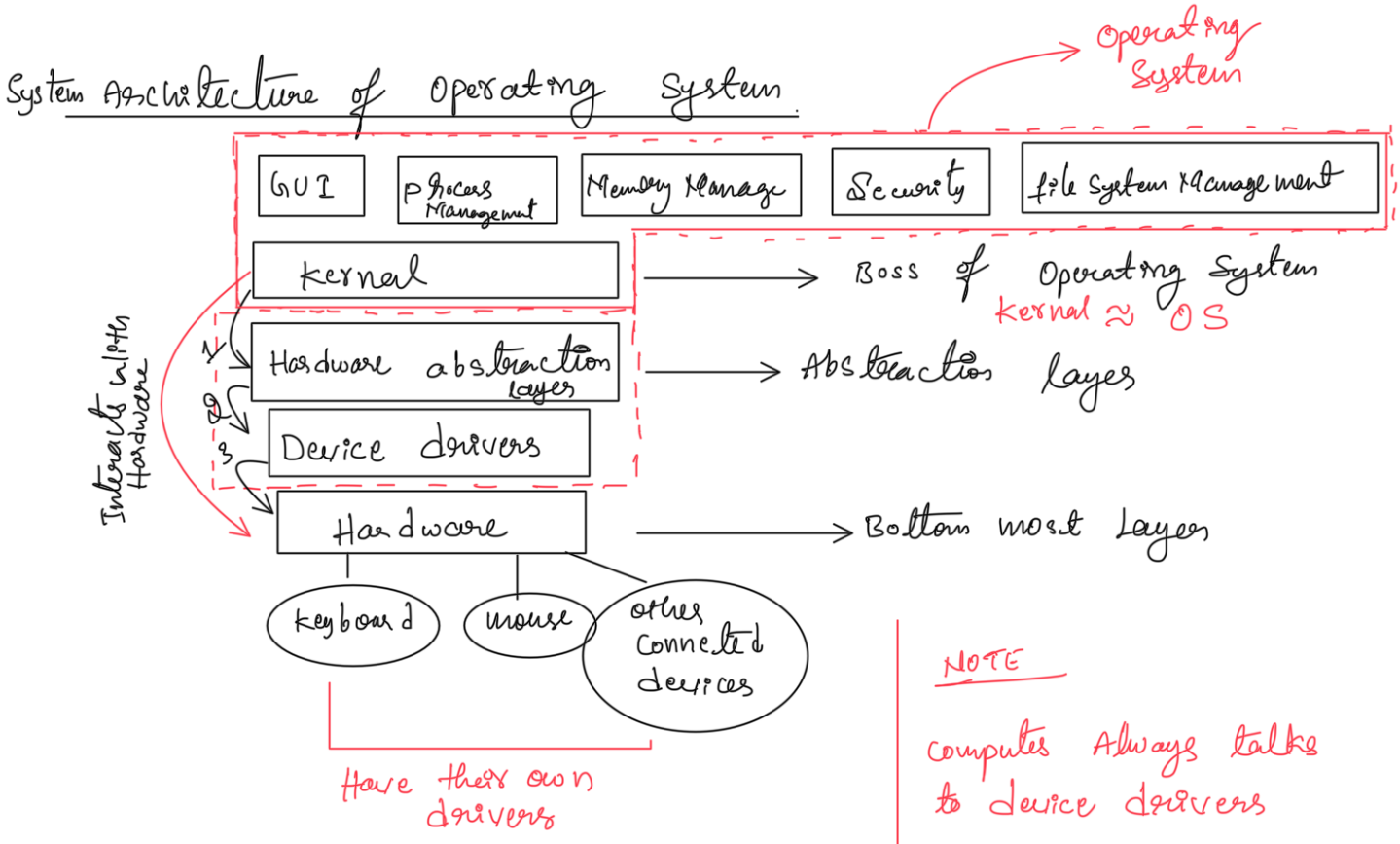
Day 3
13/07/25

Operating System

System software that helps to manage your Computer Resources & It also helps to Run programs on the Computer.

Operating System is the one of the important and Complex Software System.

System Architecture of Operating System



NOTE

Computer Always talks to device drivers

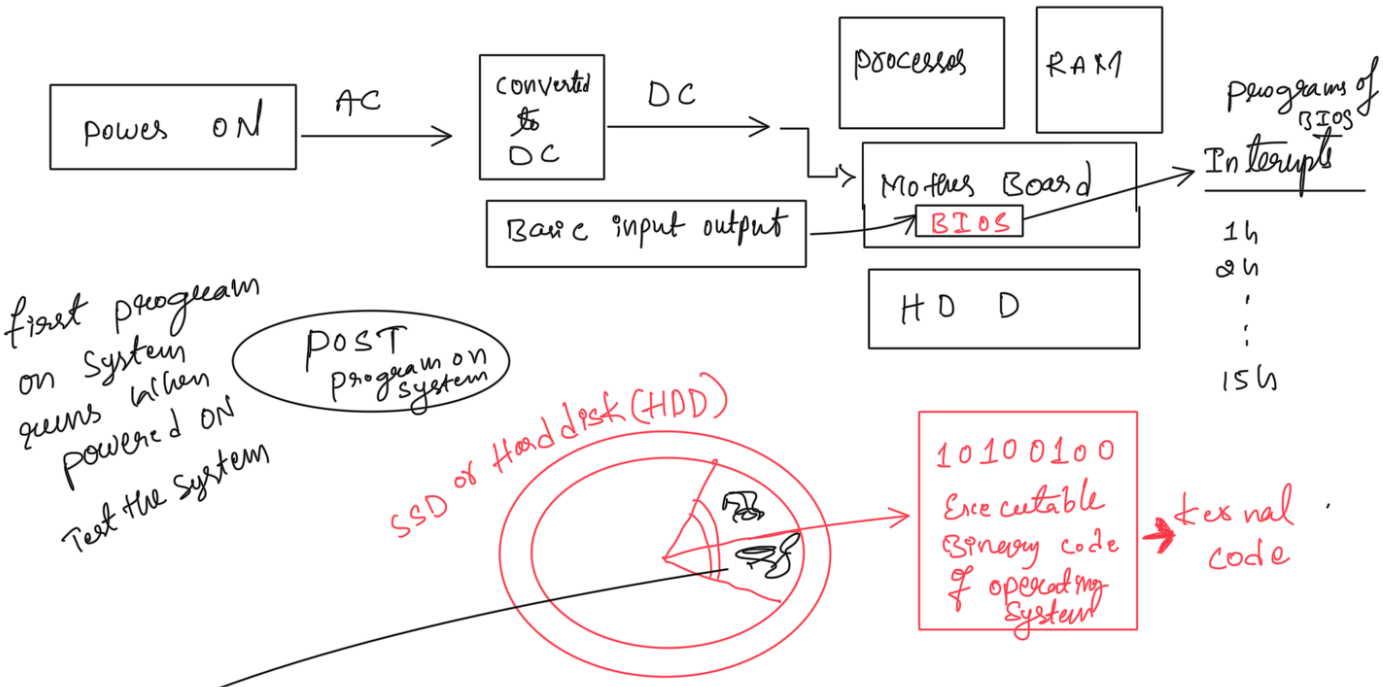
The Device drivers internally talks to Hardware

In this way System will work

What Happens when we Turn on the Computer ?

① power on — power supply starts and signals Mother Board
② BIOS/UEFI Basic Input output system performs POST
③ Boot loader is loaded from bootable storage

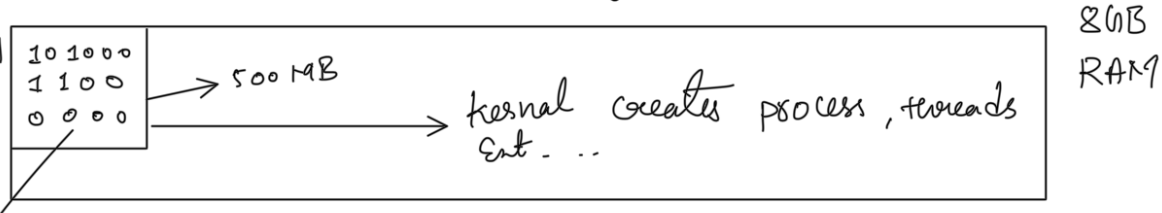
③ Boot loader is loaded from secondary storage
④ OS kernal is loaded into memory
⑤ kernal initializes system memory process device drivers
⑥ User level programs like login screen desk top environment



To load the kernal or executable operating System code there is some special program called Boot loader.

Boot Loader (Master Boot Record)

Boot Loader loads the operating system to RAM

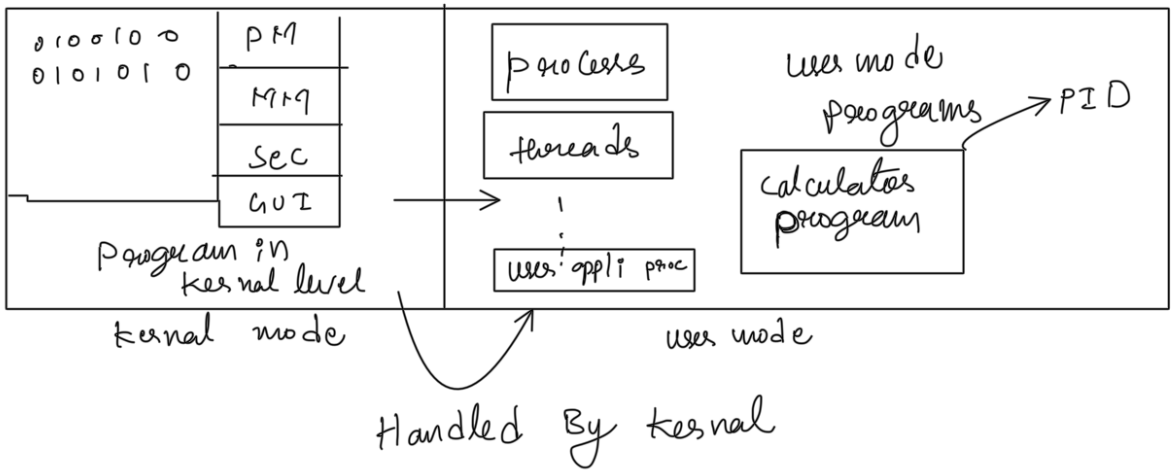


NOTE The kernal code which is executed is not present in the process, The process comes after the kernal program Execution.

kernal dont have process Id.

Contains code of process Managers, Memory Manager Security Ext.

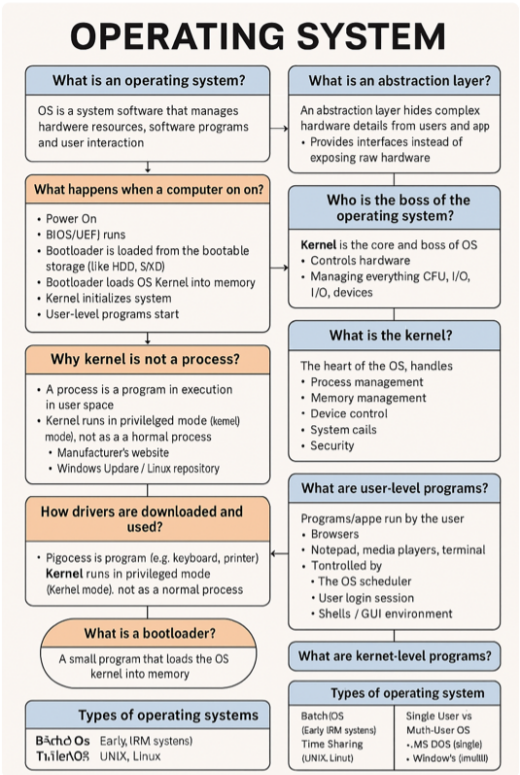
kernals are written in C language
Most of the System software are written in C/C++



user level programs are Handled By Operating System

Types of operating System

1. Distributed
2. Real Time operating System



Operating Systems - Detailed Notes

What is an Operating System?

An Operating System (OS) is system software that manages hardware and software resources. It acts as a bridge between users/applications and computer hardware, ensuring efficient operation of the system.

What Happens When a Computer is Turned On?

1. Power On: Power supply initiates.
2. BIOS/UEFI performs POST.
3. Bootloader is loaded from disk.
4. Bootloader loads OS Kernel.
5. Kernel initializes system and hardware.
6. User-level programs (login, GUI) start.

Architecture of an Operating System

Typically layered as:

User Programs -> System Calls -> Kernel -> Hardware.

Kernel includes process, memory, file, and device management.

Abstraction Layer

Abstraction layer hides hardware complexity from applications by providing clean interfaces, like file systems instead of disk sectors.

Bottom-most Layer

The hardware (CPU, memory, I/O devices) is the bottom-most layer, directly managed by the OS kernel.

Who is the Boss of the Operating System?

The Kernel is the boss of the OS. It manages resources, security, and provides services to programs.

Why Kernel is Not a Process?

The kernel runs in privileged mode, not as a normal user process. It is the underlying control system that manages all other processes.

What is Kernel?

Operating Systems - Detailed Notes

Kernel is the core of OS that handles process scheduling, memory, device communication, and system calls.

How Drivers are Downloaded and Used?

1. OS detects hardware.
2. Drivers are loaded (from OS or online).
3. Kernel uses drivers to control hardware.

What is Bootloader?

Bootloader is a small program that loads the OS kernel into memory. E.g., GRUB for Linux, Windows Boot Manager.

What are User-Level Programs?

Programs run by users (e.g., browsers, editors). Controlled by OS scheduler and login session.

What are Kernel-Level Programs?

Internal services or modules of the kernel: scheduler, memory manager, device drivers, etc.

Types of Operating Systems

- Batch OS: Executes jobs without user input.
- Time-Sharing OS: Allows multiple users.
- Distributed OS: Manages multiple systems as one.
- Embedded OS: For specialized devices.
- Real-Time OS: Immediate response systems.
- Network OS: Manages networked systems.
- Mobile OS: Android, iOS.
- Single/Multi-user and Single/Multi-tasking OS.