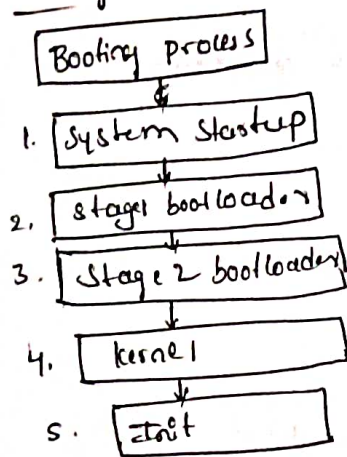


## 01) Bootling Process :



Bios / Boot monitor

MBR (master boot record)

LILO, GRUB etc

tr

Linux

User space

1. system startup → when a clm is first booted (reset the processor execute code at a well known location. In PC-BIOS stored on a flash memory on motherboard.

→ Bios is a small piece of code (512 bytes) - 1st prog to run.

→ Bios identifies HWC (boot device) which is bootable machine

2) stage1 Boot loader → job is to find and load secondary boot loader.

→ MBR loads this stage1 boot loader

3) Stage2 boot loader → A pthy called as kernel loader its job is to load the Linux kernel.

→ 1<sup>st</sup> & 2<sup>nd</sup> stage boot loader combined are called LILO (Linux loader) or GRUB (Grand Unified Boot loader)

4) kernel → At the head of the kernel image, a routine does some HW setup & decompress the kernel image & place it in high memory.

→ kernel is then called and kernel boot begins.

5) Init → kernel starts the first user space application

→ compiled into c library.

→ Post prog /sbin/init

## 02y Functions of OS.

### i) Security

→ protects user data

→ It prevents unauthorized access to programs & user data

### ii) Job accounting.

→ OS keeps track of time and resources used by various tasks.

### iii) Error detecting aids

→ constantly monitors the system to detect errors & avoid malfunctioning of system

### iv) Coordination between software & users

→ OS co-ordinates & assigns interpreters, compilers, assembly

### v) Memory management.

→ The OS manages the primary memory

→ It keeps track of memory, which bytes are used by which prog.

→ Allocate memory for the process, deallocate when the process has terminated

### vi) Processor management!

→ OS decides the order in which the processes have access to the processor

→ keeps track of the states of the processes.

### vii) File management!

→ A file system is organized into directories for efficient navigation & usage

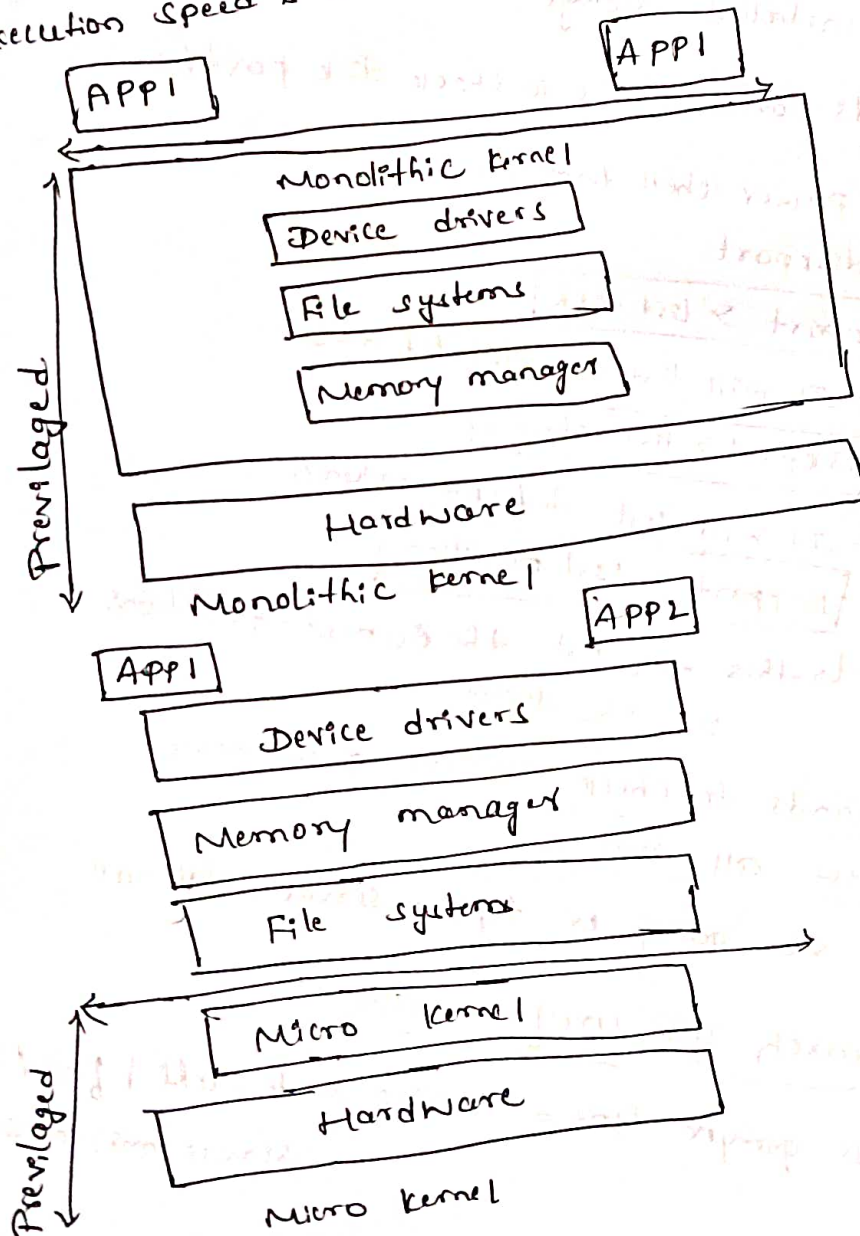
### 037 Monolithic and Microkernel difference with diagram

#### Micro kernel

- user services and kernel are kept in separate address space.
- OS is complex to design.
- Microkernel is small in size.
- Easier to add new functionalities.
- Failure of one component does not effect the working of micro kernel.
- Execution speed is low.

#### Monolithic kernel

- Both user services and kernel services are kept in same address space.
- OS is easy to design and implement.
- Monolithic kernel are larger in size.
- Difficult to add new functionalities.
- Failure of one component leads to failure of entire system.
- Execution speed is high.





## ④. UEFI and Legacy boot

→ UEFI stands for unified extensible firmware interface. Most new motherboards consist of this type it has more advantage than using BIOS.

Most importantly, it provides user friendly graphical user interface (GUI).

BIOS provides blue screen. BIOS cannot recognize large storage drives. UEFI provides a good alternate.

→ In regular BIOS model that uses the keyboard to select the option. UEFI allows controls via mouse.

UEFI contains Secure boot

→ legacy BIOS used by BIOS firmware. It stores a list of installed storage device that are bootable.

## ⑤ Commands on Windows to check disk partition.

→ open power shell terminal

→ type diskpart

→ `Diskpart > list disk`

↳ It will list any detected disks.

→ `Diskpart > list volumes`

↳ It will list detected volumes

→ `Diskpart > list partition`

↳ This will list the current partitions on the device.

## ⑥ Commands to check services in Windows.

→ List all services

> `sc query ex type = service state = all`

→ search for specific service

> `sc queryex type = service state = all | find /i "service name" my service`

## 077 Steps to start or stop services in Windows

### Stop service

- open start
- search services
- click on the service you want to start
- click start button
- Apply button

### stop service

- Open start
- search services.
- click on the service you want to stop.
- click stop
- click apply.