

# Syllabus of Operating System

1. Basic. Introduction
  - Types
  - Process Diagram
  - System calls
  - Most Important
2. Process Scheduling
  - Algorithms.
    - FIFO
    - SJT
    - Round Robin
  - \*\*
3. Process Sync.
4. Deadlock & Threads.
  - Semaphore. / Complex. ↑↑
  - Def, Prevention
  - Banker's Algorithm.

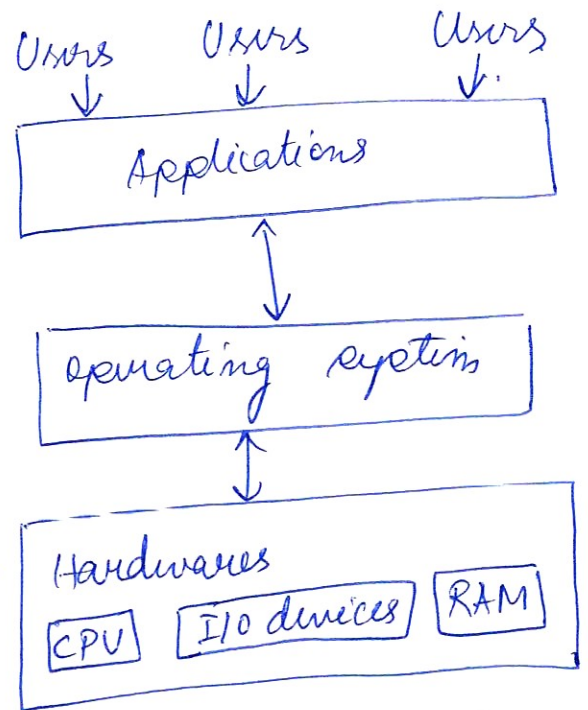
---

5. Memory Management
  - Paging.
  - Segmentation
  - Fragmentation.
  - Virtual Memory
  - Num \*\*
  - Page Replacement Algo. (Easy and Frequently asked)
6. Disk Scheduling
  - Scan
  - C-Scan
  - FCFS
  - Algorithms
7. UNIX commands
  - [BASIC LINUX CMDS].
  - open system call
  - file parameters.
8. File Management and security.
  - Sequential.
  - Random
  - Linked
  - Encryption.
  - cryptography.

## L-1.1. Introduction to Operating System and its Functions

It acts as an interface between the users and the hardware.

Primary goal is (95%) W to provide Instructions confinement to the as programs users  $\rightarrow$  throughput (Linux)



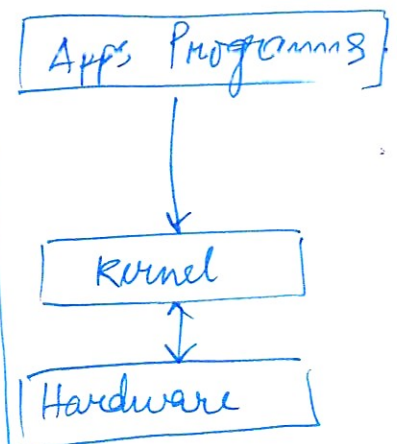
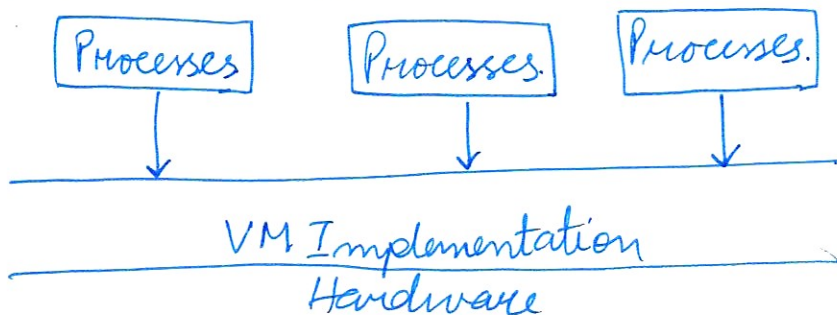
### User Mode Vs Kernel Mode

The mode bit is 1. user mode { Safe Mode  
mode bit is 0 system mode. Restricted Mode }  
kernel

### Kernel Mode

System Call  $\rightarrow$  a method to req. assistance from the kernel of the operating system's kernel.

### Virtual Machine



## Types of VMware Workstation.

XEN

Virtual Box.

Citrix

KVM

---

Open Source OS.

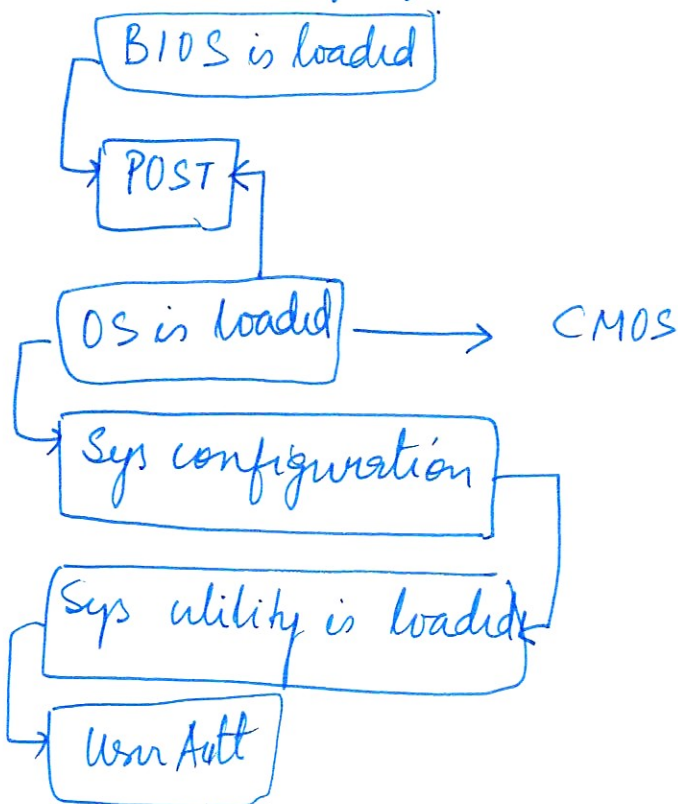
Linux, Open Solaris, FreeBSD, Minix etc.

---

Booting → process of starting a computer.

Bootloader → software that is ~~not~~ responsible

to load the operating system into the memory of the computer.



BIOS search for MBR → if found.

utilize



## Linker and Loader

↳ .exe file ↳ loads the program.

Source code → Translator.

Assembler → { Binary to Machine code.

Interpreter → line by line conversion.

6<sup>th</sup> March, 2025 | Thread = lightweight process.

A basic unit of CPU utilization.

- ↳ Thread ID
- ↳ Program counter
- ↳ Register stack / and set

What is shared?

- ↳ other thread belonging to the same process

Single thread → heavyweight process.

of execution | If a process has multiple threads of  
of control | control, it can perform more than  
one task at a time.

Block Diagram.

## Pros of Multithreadings

- |                       |   |
|-----------------------|---|
| 1. Responsiveness ;   | 3. Economic.                            |
| 2. Resource sharing : | 4. Utilizes Microprocessor architecture |

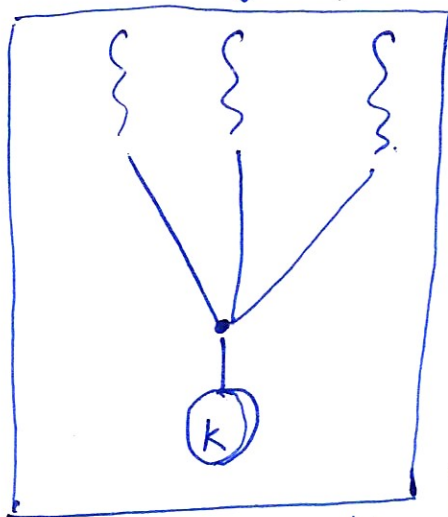
## Types of Threads

1. User Threads (no kernel support)
2. Kernel Threads (managed by OS itself).

Relationships (a) Many to one model

(b) One to one

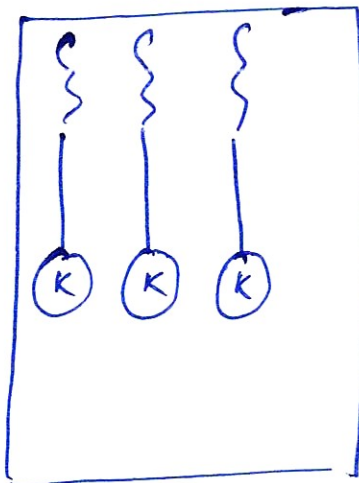
(c) Many to Many



Cons: The entire process will be blocked if a thread makes a blocking system.

a) Because one thread can access the kernel at a time.

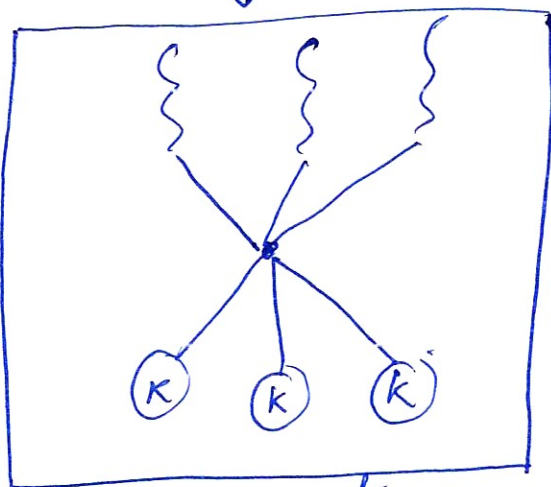
Eg:- Unix.



Provide more concurrency than the many to one model  
Allow multiple threads.

Eg:- Windows 95.

Cons:- creating a user thread reqd.  
creating kernels.



User can create as many as threads they want.  
Cons: Thread fiber  
Windows NT

## Hyperthreading

SMT.

Hyperthreading allows their processor core resource to become multiple logical processor for performance.

It enables the processor to execute

[cmd - unic. and cpu get number of core, number of logical processor.

2 and 4 means hyperthreading is enabled.

System call.

fork() → create a separate, duplicate process.

exec() → when an exec() system call is invoked, the program specific parameters to ex

getpid().

getppid()

2 no. of fork. 1 → fork();

2 → fork();

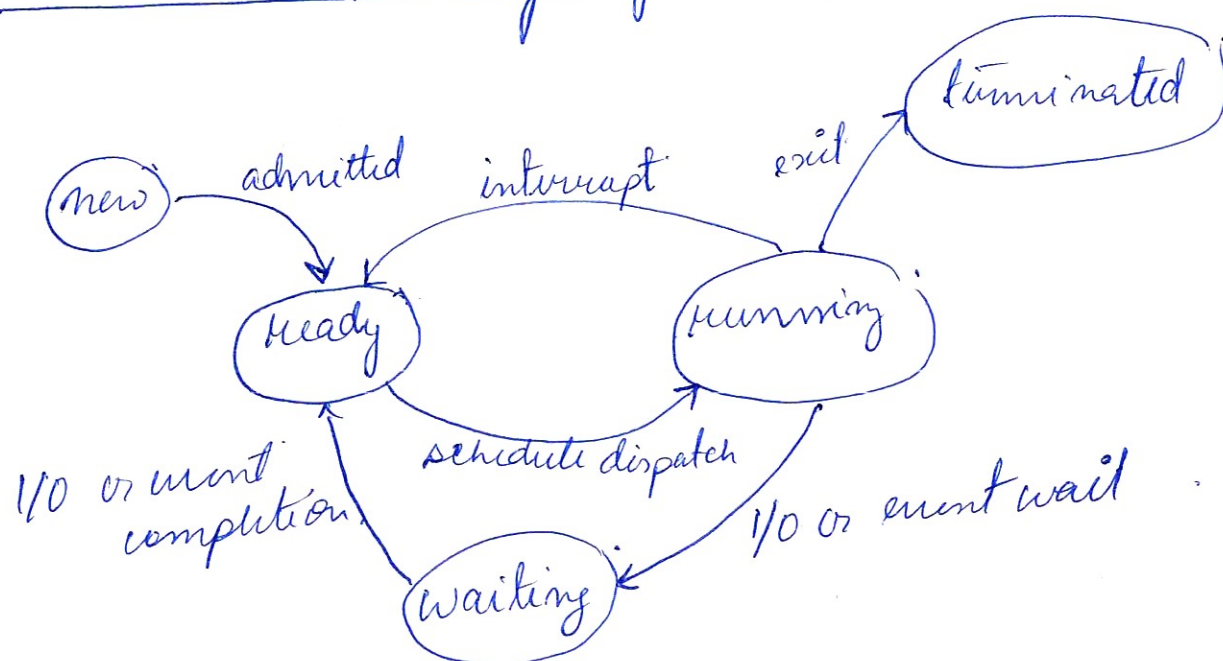
count <= n 00 ...;

$$2^2 = 4$$

(P<sub>1</sub>) (1259) (P<sub>id</sub>)



## Process in Operating System



**Process:-** A program in execution.

**Program:-** A passive entity, such as a file containing a list of instructions stored on the disk (exe file).

A program becomes a process when an exe file is loaded into memory.



FIFO

