Day 1

OS Fundamentals

- The interface between a computer user and computer hardware.

important functions of an operating System

- Memory Management
- Processor Management

Keeps tracks of processor and status of process. The program responsible for this task is known as **traffic controller**.

- Device Management

Keeps tracks of all devices. Program responsible for this task is known as the **I/O controller**.

- File Management
- Security
- Control over system performance
- Job accounting
- Error detecting aids
- Coordination between other software and users

Types of OS

1. Batch OS : processes are submitted by each user on an offline device and then processor does the job, to speed up similar jobs are run as batches
2. Time sharing OS: Multiple terminal on same computer. To manage time better, Each CPU runs multiple jobs by switching. this reduces response time. uses CPU scheduling and multiprogramming
3. Distributed OS: Multi processor for multiple users. hese are referred as **loosely coupled systems** or distributed systems.
4. Network OS: runs on server
5. Real time OS:

1. hard real time
2. soft real time

Spooling

Spooling is an acronym for simultaneous peripheral operations on line. Spooling refers to putting data of various I/O jobs in a buffer. This buffer is a special area in memory or hard disk which is accessible to I/O devices.

Processes

Scheduling

It is also called a **job scheduler**. A long-term scheduler determines which programs are admitted to the system for processing. It selects processes from the queue and loads them into memory for execution. Process loads into the memory for CPU scheduling.

It is also called as **CPU scheduler**. Its main objective is to increase system performance in accordance with the chosen set of criteria. It is the change of ready state to running state of the process. CPU scheduler selects a process among the processes that are ready to execute and allocates CPU to one of them.

A running process may become suspended if it makes an I/O request. A suspended processes cannot make any progress towards completion. In this condition, to remove the process from memory and make space for other processes, the suspended process is moved to the secondary storage. This process is called **swapping**, and the process is said to be swapped out or rolled out. Swapping may be necessary to improve the process mix.

Scheduling Algorithms

- First-Come, First-Served (FCFS) Scheduling
- Shortest-Job-Next (SJN) Scheduling
- Priority Scheduling
- Shortest Remaining Time : The processor is allocated to the job closest to completion but it can be preempted by a newer ready job with shorter time to completion.

- Round Robin(RR) Scheduling : Each process is provided a fix time to execute, it is called a **quantum**.
- Multiple-Level Queues Scheduling

Processes and Threads

A process is the execution of a program that allows you to perform the appropriate actions specified in a program.
Thread is an execution unit that is part of a process. A process can have multiple threads, all executing at the same time
Process means a program is in execution, whereas a thread means a segment of a process.

MULTITHREADING

Difference between User-Level & Kernel-Level Thread

S.N.	User-Level Threads	Kernel-Level Thread
1	User-level threads are faster to create and manage.	Kernel-level threads are slower to create and manage.
2	Implementation is by a thread library at the user level.	The operating system supports the creation of Kernel threads.
3	User-level thread is generic and can run on any operating system.	Kernel-level thread is specific to the operating system.
4	Multi-threaded applications cannot take advantage of multiprocessing.	Kernel routines themselves can be multithreaded.