os works as an interface between user and hardware or software and hardware . it reduces the effort of writing , individual program for every command , primary goal of operating system is to provide convenience for the users so they can access the hardware easily without any extra effort . it was expected that whatever the operating system is gonna be use must have a high convenience throughput

**Function of os**

* Management of resources
* Process Management
* Storage management
* Memory management
* Secure management

**What is process**

A process can be thought as a program which is in execution

**THREAD**

A THREAD is the basic unit of execution within a process which has data dublicacy or data copies

Process trees

A process state depend on current activity of process and it changes constantly

**PROCESS CONTORL BLOCK**

GENERALLY each process is represented by a process control block or task control block

Process id / pid

A process control block is data structure used by an os to store info about the process the structure of pcb may vary

Depending on the os

A unique identifier assigned to each process by the O.S

Program Counter/P.C:-

The address to of the next instruction to be Executed by the

Process State:-

Same from above

C.P.U registers:-

**1)**.The values of the processor , registors that are used by the process.

**2)**.Memory management Information(mmi)

Information about the processes memory allocation and such as base and limit registers.

**3**).Input Output Status Information ➖Information about the processes input/output operations such as open files and network connections.

**4**).Accounting Info:-Info about the resources used by the process such as cpu time and memory.

**5**). C.P.U Scheduling Information:-Information about the processes priority and scheduling environment.

Process Scheduling Queues:-

All pcb’s are kept in process scheduling queues by the Operating Systems. Each processing state has its own processing queue in the OS.

1. Job Queue: It contains all the system’s processes.
2. Ready Queue: This queue maintains a list of all process in the main memory that are ready to run. This is always filled with new processes.
3. Device Queue: This queue consists of processes which are waiting to have an access to input and output devices.

**JOB QUEUE** ---> **READY QUEUE** —---> CPU-->OUTPUT

| |

| |

DEVICE INPUT —----------> **DEVICE**

-OUTPUT  **QUEUE**

**SCHEDULERS -** are special codes and programmes that manage process scheduling in so many ways. Their primary responsibility is to choose which processes will go to the system and which processes will go to the running state.

**TWO SCHEDULAR TYPES**:

1. **PREEMPTIVE**
2. **NON-PREEMPTIVE**

**NON-PREEMPTIVE SCHEDULING: In this way of scheduling, a process’s resource cannot be taken before the process has finished running. When a running process finishes and transits to a waiting state, then the resources are switched.**

**PREEMPTIVE:** In this way of scheduling, The OS assigns resources to a process for a predetermined period of time. The process switches from running state to ready state or from waiting state to ready state during resource allocation.

**LONG-TERM JOB SCHEDULER**:

1. This scheduler allows the new processes to the ready state.
2. It controls the degree of multiprogramming.

**//processes ko instructions deke ready state main laane ka kaam long-term scheduler ka hai.**

**Degree of multiprogramming :** the no. of processes present in ready state at any point of time or n th point of time.

The efficiency of job scheduler is increased by maintaining a balance between input/output bound and CPU bound.

**SHORT TERM SCHEDULER / CPU SCHEDULER:** It is majorly used to improve system performance. It helps in transition of processes from ready to running state.

**MEDIUM-TERM SCHEDULER:** it is responsible for suspending and resuming the processes(swapping)**.** To improve the process mix or because a change in memory requirements over committed available memory.

**CONTEXT SWITCHING:** For a process to be executed and then taking a pass, then to be continued again from the same point. It is a mechanism to store and restore the data of a cpu

In the **pcb**(process control block) a context switcher makes it possible for multiple processes to share a single cpu