**Coding-Challenge**

* **Process life cycle:**

Process life cycle in OS is one of the **five states** in which a process can be starting from the time it has been submitted for execution, till the time when it has been executed by the system.

A process can be in any of the following states:

* New state
* Ready state
* Running state
* Waiting state
* Terminated.
* **Role of an operating system:**

The operating system acts as an **interface between the hardware and the programs requesting I/O**. It is the most fundamental of all system software programs.

Responsibilities of the OS include:

* Hiding the complexities of hardware from the user.
* **Process:**

In Operating system we define any process as **a program in execution or instance of a program of the system**, program is a static term however process means a running or active entity. On batch systems, it is called as a “job” while on time sharing systems, it is called as a “task”.

* **Job of a long-term scheduler:**

Long-Term Scheduler is also called Job Scheduler and is responsible for **controlling the Degree of Multiprogramming** i.e., the total number of processes that are present in the ready state. So, the long-term scheduler decides which process is to be created to put into the ready state.

* **Response time:**

Response time is the **amount of time after which a process gets the CPU for the first time after entering the ready queue**.

Response Time = Time at which process first gets the CPU – Arrival time 4.

* **Pre-emptive scheduling:**

Pre-emptive Scheduling is a **CPU scheduling technique that works by dividing time slots of CPU to a given process**. The time slot given might be able to complete the whole process or might not be able to it. When the burst time of the process is greater than CPU cycle, it is placed back into the ready queue and will execute in the next chance.

* **Non -Pre-emptive scheduling:**

Non-pre-emptive scheduling is called rigid as even if a critical process enters the ready queue the process running CPU is not disturbed. The pre-emptive Scheduling has to maintain the integrity of shared data that’s why it is cost associative as it which is not the case with Non-pre-emptive Scheduling.