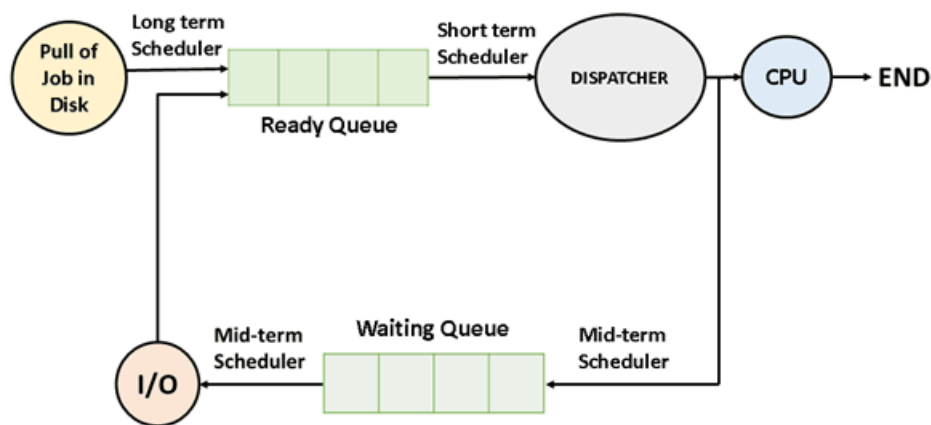
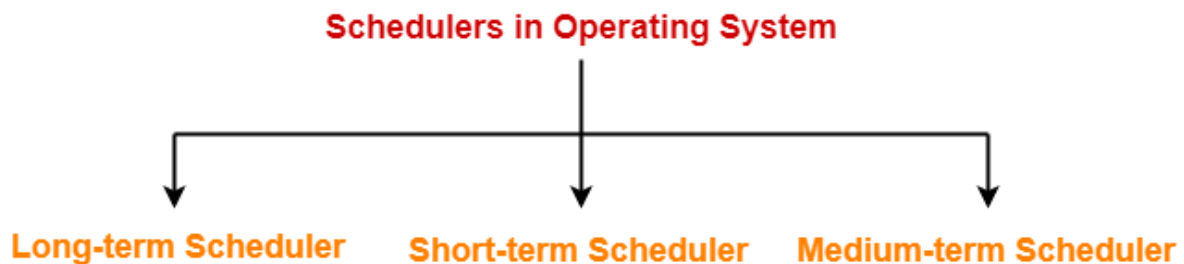


## Schedulers in OS-

- Schedulers in OS are special system software.
- They help in scheduling the processes in various ways.
- They are mainly responsible for selecting the jobs to be submitted into the system and deciding which process to run.

## Types of Schedulers-

There are 3 kinds of schedulers-



### 1. Long-term Scheduler-

- Long-term scheduler is also known as **Job Scheduler**.
- It selects a balanced mix of I/O bound and CPU bound processes from the secondary memory (new state).
- Then, it loads the selected processes into the main memory (ready state) for execution.

### 2. Short-term Scheduler-

- Short-term scheduler is also known as **CPU Scheduler**.
- It decides which process to execute next from the ready queue.

- After short-term scheduler decides the process, **Dispatcher** assigns the decided process to the CPU for execution.

### **3. Medium-term Scheduler-**

- Medium-term scheduler swaps-out the processes from main memory to secondary memory to free up the main memory when required.
- Thus, medium-term scheduler reduces the degree of multiprogramming.
- After some time when main memory becomes available, medium-term scheduler swaps-in the swapped-out process to the main memory and its execution is resumed from where it left off.
- Swapping may also be required to improve the process mix.

<b>Long-Term</b>	<b>Short-Term</b>	<b>Medium-Term</b>
Long term is also known as a job scheduler	Short term is also known as CPU scheduler	Medium-term is also called swapping scheduler.
It is either absent or minimal in a time-sharing system.	It is insignificant in the time-sharing order.	This scheduler is an element of Time-sharing systems.
Speed is less compared to the short term scheduler.	Speed is the fastest compared to the short-term and medium-term scheduler.	It offers medium speed.
Allow you to select processes from the loads and pool back into the memory	It only selects processes that is in a ready state of the execution.	It helps you to send process back to memory.