SNS COLLEGE OF ENGINEERING

Kurumbapalayam(Po), Coimbatore – 641 107
Accredited by NAAC-UGC with 'A' Grade
Approved by AICTE, Recognized by UGC & Affiliated to Anna University, Chennai

Department Of Artificial Intelligence and Data Science

Course Name – Operating Systems

II Year / IV Semester

Unit 1 - OPERATING SYSTEMS OVERVIEW(PROCESS SCHEDULING)

Process Scheduling

- The process scheduling is the activity of the process manager that handles the removal of the running process from the CPU and the selection of another process on the basis of a particular strategy

- The module of operating system that makes this decision is called the scheduler

Scheduling Queues

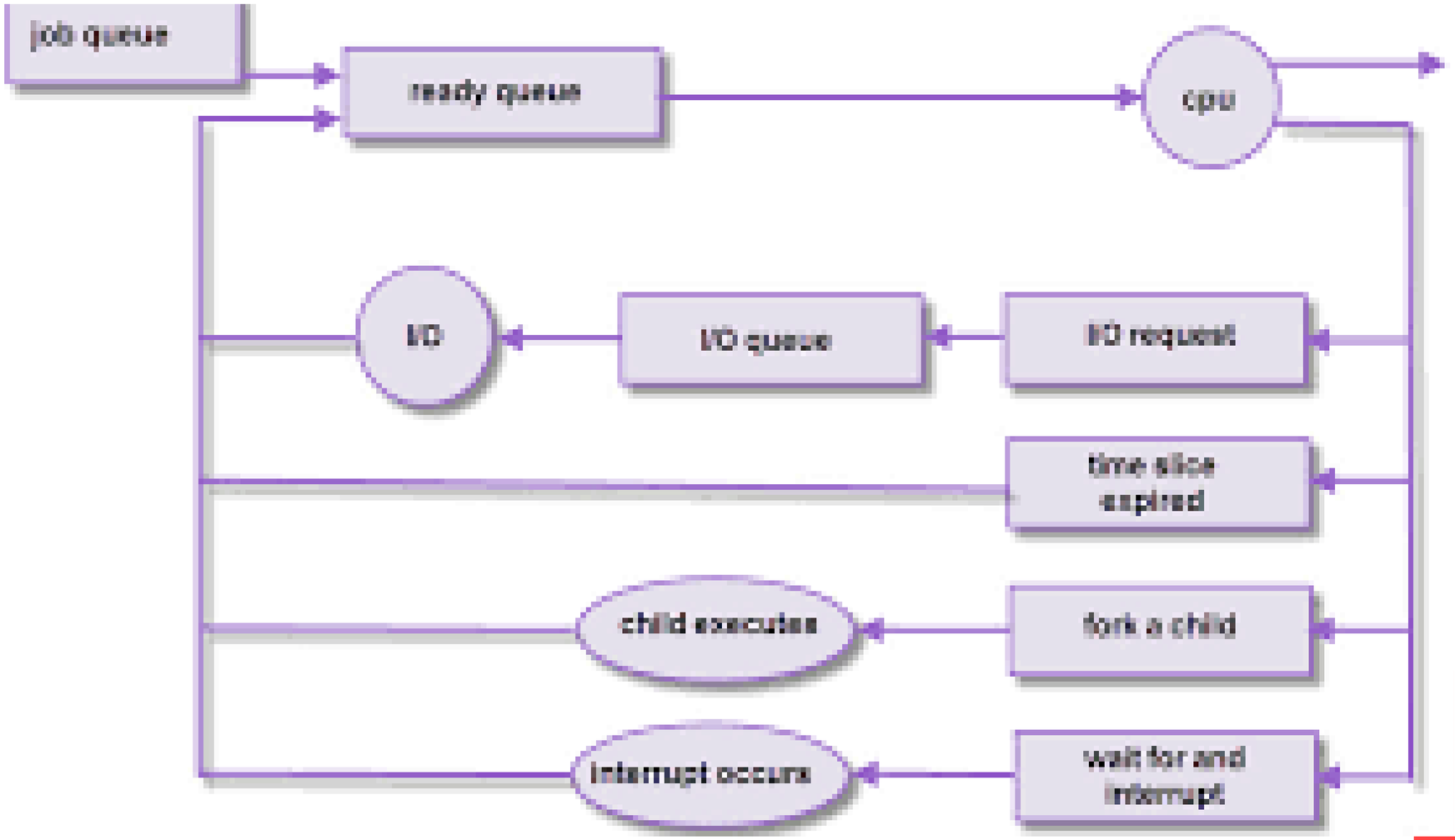
Scheduling queues refer to queues of process or devices.

When the process enters the system, then this process is put into a **job queue**. This queue consists of all processes in the system.

The processes that are residing in main memory and are ready and waiting to execute are kept on a list called the **ready queue**. This queue is generally stored as a linked list.

The operating system also maintains other queues such as device queue. **Device queue** is a queue for which multiple processes are waiting for a particular I/O device.

Representation of process scheduling



A common representation of process scheduling is a queueing diagram. This queueing diagram was shown in the Figure. IN the figure .

* Queue is represented by rectangular box.
* The circles represent the resources that serve the queues.
* The arrows indicate the process flow in the system.

Queues are of two types. They are:

* Ready Queue
*Device Queue

A newly arrived process is put in the ready queue. Processes wait in a ready queue for allocating the CPU. Once the CPU is assigned to a process, then that process will execute. While executing the process, any one of the following events can occur.

*The process could issue an I/O request and then it would be placed in an IO queue.
*The process could create new sub process and will wait for its termination.
*The process could be removed forcibly from the CPU, as a result of interrupt and put back in the ready queue.

Two state process model

Two state process model refers to the running and non running states which are described below:

Running state:
When new process is created by operating system that process enters into the system as in the running state.

Non Running state:
Process that are not running are kept in the queue waiting for their turn to execute.

Schedulers

Schedulers are special system softwares which handle process scheduling in various ways. Their main task is to select the jobs to be submitted into the system and to decide which process to run. There are three types of schedulers.

They are:
* Long Term Scheduler
* Short Term Scheduler
* Medium Term Scheduler

Long Term Scheduler

*It is also called as job scheduler.
*determines which programs are admitted to the system for processing.
*selects processes from the queue and loads them into memory for execution
*provide a balanced mix of jobs
*It also controls the degree of multiprogramming.
If the degree of multiprogramming is stable, then the average rate of process creation must be equal to the average departure rate of process leaving the system.
*On some systems, the long term scheduler may not be available or minimal.
*Time-sharing operating systems have no long term scheduler.
When process changes the state from new to ready then there is a use of long term scheduler

Short Term Scheduler

*Short-term scheduler or CPU scheduler selects from among the processes that are ready to execute and allocates the CPU to one of them. (i.e. a process that resides in main memory will be taken by CPU for execution).

*The short-term scheduler must select a new process for the CPU frequently.

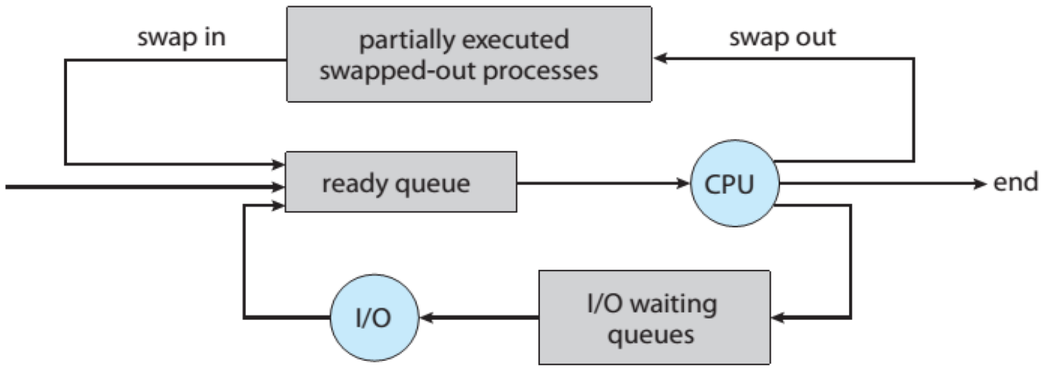
*The short term scheduler must be very fast because of the short time between executions of processes.

Medium Term Scheduler

Medium Term Scheduler does two tasks:

. Swapping: Medium-term scheduler removes a process from main memory and stores it into the secondary storage. After some time, the process can be reintroduced into main memory and its execution can be continued where it left off. This procedure is called Swapping.

2. Medium Term Scheduler moves a process from CPU to I/O waiting queue and I/O queue to ready queue.



The processes can be described as two types:

•1. I/O bound process is one that spends more of its time doing I/O than it spends doing computations.

•2. CPU Bound process using more of its time doing computations and generates I/O requests infrequently.

The long-term scheduler selects a good process mix of I/O-bound and CPU-bound processes.

• If all processes are I/O bound, the ready queue will almost always be empty and the CPU will remain idle for long time because I/O device processing takes a lot of time.

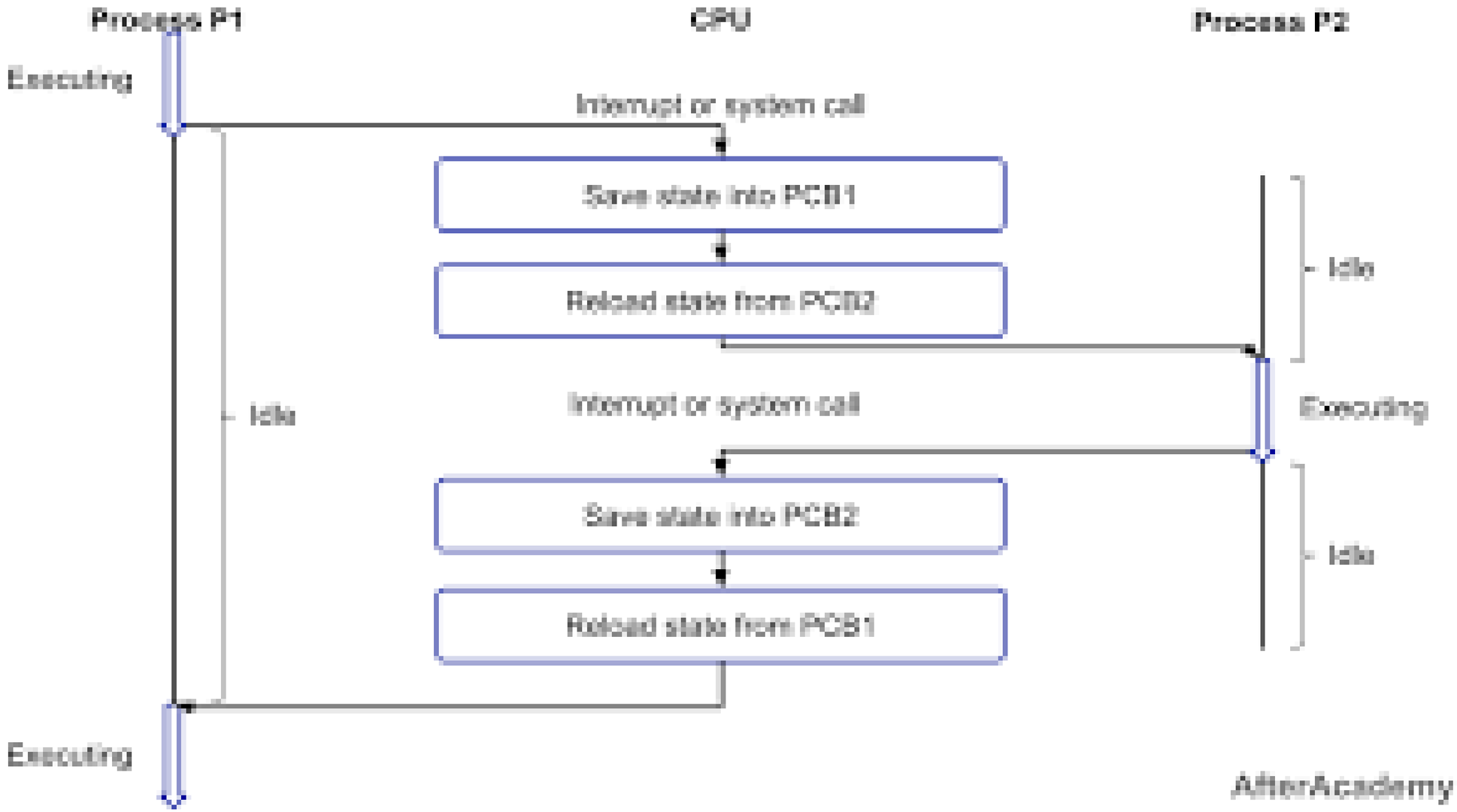
• If all processes are CPU bound, the I/O waiting queue will almost always be empty. I/O devices will be idle and CPU is busy for most of the time.

• Thus if the system maintains the combination of CPU bound and I/O bound processes then the system performance will be increased.

Note: Time-sharing systems such as UNIX and Microsoft Windows systems often have no long-term scheduler but simply put every new process in memory for the short-term scheduler.

Context Switching

- Switching the CPU from one process to another process requires performing a state save of the current process and a state restore of a different process. This task is known as a Context Switch.
- The context is represented in the PCB of the process. It includes the value of the CPU registers, the process state and memory-management information.
- When a context switch occurs, the kernel saves the context of the old process in its PCB and loads the saved context of the new process scheduled to run.
- Context-switch time is pure overhead, because the system does no useful work while switching. Context switch time may be in few milliseconds.


Process P1
CPU
Process P2
Executing
Interrupt or system call
Save state into PCB1
Reload state from PCB2
Idle
Idle
Interrupt or system call
Executing
Save state into PCB2
Reload state from PCB1
Idle
Executing