CPU SCHEDULING

PRESENTED BY

Ms. VIJI PARTHASARATHY
ASSISTANT PROFESSOR
DEPARTMENT OF COMPUTER SCIENCE
SHRIMATI INDIRA GANDHI COLLEGE
TIRUCHIRAPPALLI-2

CPU SCHEDULING IN OPERATING SYSTEM

Support:

III BCA	16SCCCA6
III B.Sc. Information Technology	16SCCIT7
III B.Sc. Computer Science	16SCCCS8
and PGDCA	P16DCA12

CPU SCHEDULING

CPU scheduling is a process of determining which process will own CPU to run while the execution of another process is on hold. (Means in waiting state due to unavailability of any resource like I/O etc., thereby maximizing utilization of CPU.)

CPU SCHEDULER

Whenever the CPU becomes idle, it is the job of the CPU Scheduler (the short-term scheduler) to select another process from the ready queue to run next.

SCHEDULING OCCURS IN FOUR CASES

When a process switches from

- The running state to the waiting state
- The running state to the ready state
- The waiting state to ready state
- Terminates

Scheduling Criteria

- CPU utilization
- Throughput
- Turnaround time
- Waiting time
- Response time

CPU utilization

- **CPU utilization** is a measure of how busy the **CPU** is right now.

Throughput

- The amount of work that a computer can do in a given period of time.

Turnaround Time

- The **time** interval from the **time** of submission of a process to the **time** of the completion of the process

Waiting time

- The time processes spend in the Ready Queue Waiting their turn to get on the *CPU*

Response time

- *Response time* is the total amount of *time* it takes to respond to a request for service

(Transmission time is also to be added)

TWO TYPES OF SCHEDULING

- **PREEMPTIVE**:
- The scheduling in which a running process can be interrupted if a high priority process enters the queue and is allocated to the CPU is called **preemptive scheduling**. In this case, the current process switches from the running queue to ready queue, and the high priority process utilizes the CPU cycle.
- .

NON-PREEMPTIVE:

❖The scheduling in which a running process can be interrupted if a high priority process enters the queue and is allocated to the CPU is called **preemptive scheduling**. In this case, the current process switches from the running queue to ready queue, and the high priority process utilizes the CPU cycle

Scheduling Algorithms

- ✓First Come First Serve
- ✓Shortest Job First
- ✓Optimal Page Replacement Algorithm
- ✓Round Robin Algorithm

First Come First Serve

- First come, first served (**FCFS**) is automatically execute queued requests and processes by the order of their arrival.
- This algorithm is non-preemptive.

Advantage

- Suitable for batch system.
- It is simple to understand and code.

Disadvantage

- Waiting time can be large if short request wait behind the long process.
- It is not suitable for time sharing system where it is important that each user should get the CPU for equal amount of time interval.

Example

Consider the following set of processes having their CPU-burst time. CPU burst time indicates that for how much time, the process needs the CPU.

process	Time for which cpu is required
	CPU burst time
p1	10
P2	5
p3	5

If the processes have arrived in order P1, P2, P3 then the average waiting time for process will be obtained from Gantt chart.

First Come First Out

come first out

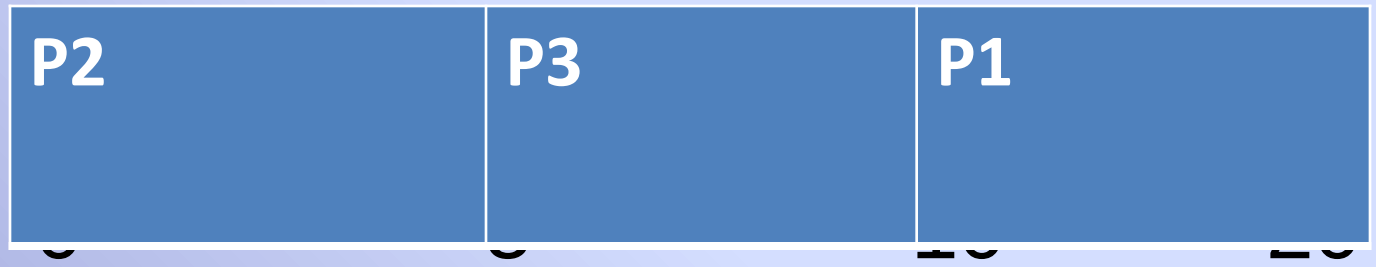
Waiting Time for P1 = 0 Millisecond

Waiting Time for P2 = 10 Millisecond

Waiting Time for P2 = 15 Millisecond

Average Waiting Time = (0+10+15)/2 = 8.33 Millisecond

If the process arrive in order P2, P3, P1 then



First Come First Out

Average Waiting Time = (0+5+10) / 3 = 5 Millisecond

Thus average waiting time will always depends upon the order in which the processes arrive. And vary depending upon whether the processes having less CPU-burst arrive first in the ready queue. Therefore this algorithm is never recommended whenever performance is major issue.

Shortest Job First

- SJF is a scheduling policy that selects the *waiting* process with the smallest execution time to execute next.
- SJF is a *non*-preemptive algorithm

Advantage

- Having minimum average waiting time among all scheduling algorithms.
- It is a Greedy Algorithm.

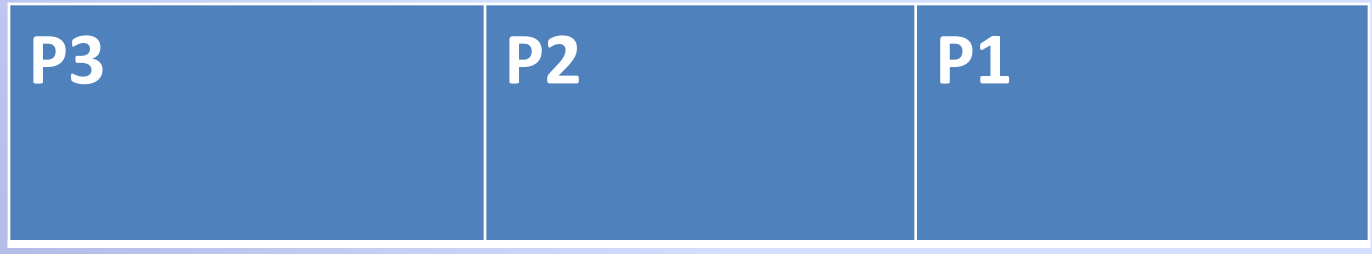
Disadvantage

- Impossible to predict the amount of CPU time a job has left.
- Long running CPU bound jobs can starve.

The following set of processes having their CPU burst time mentioned in millisecond and having arrived almost at the same time.

process	CPU Burst Time
p1	10
P2	5
p3	2

The Gantt chart:



Shortest Job First Scheduling

0 2 7 17

Waiting time for P3 = 0 millisecond
Waiting time for P2= 2 millisecond
Waiting time for P1 = 7 millisecond
Average waiting time= (0+2+7) /3 = 3 millisecond

Optimal Page Replacement Algorithm

- Whenever a new **page** is referred and not present in memory, **page** fault occurs and **Operating System** replaces one of the existing **pages** with newly needed **page**.

If referred page is already present, increment hit count.
If not present, find if a page that is never referenced in future. If such a page exists, replace this page with new page. If no such page exists, find a page that is referenced farthest in future. Replace this page with new page.

7	0	1	2	0	3	0	4	2	3	0	3	2	1	2	0	1	7	0	1
7	7	7	2		2	2				2				2			7		
	0	0	0		0	4				0				0			0		
		1	1		3	3				3				1			1		

Round Robin Scheduling

- A fixed time is allotted to each process, called **quantum**, for execution.
- Once a process is executed for given time period that process is preemptied and other process executes for given time period.

Advantage

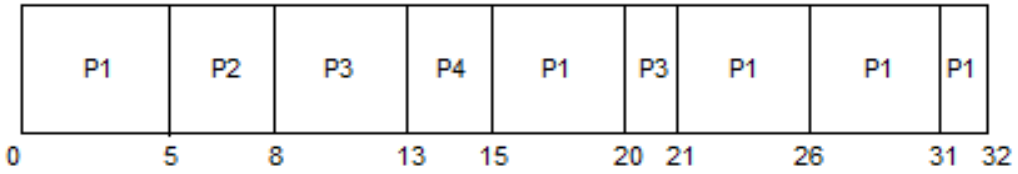
- Every Thread / Process gets a chance to run.
- CPU is shared between all processes.
- Threads with the same priority are handled perfectly with Round Robin.

Context switching is used to save states of preemptied processes.

P1	P2	P3	P4	P5	P6	P7	P8	P9

PROCESS	BURST TIME
P1	21
P2	3
P3	6
P4	2

The GANTT chart for round robin scheduling will be,



The average waiting time will be, 11 ms.

Process id	Arrival time	Burst time
P1	5	5
P2	4	6
P3	3	7
P4	1	9
P5	2	2
P6	6	3

Queue: P4,P5,P3,P2,P4,P1,P6,P3,P2,P4,P1,P3

Process id	Arrival time	Burst time	Completion time	Turnaround time	Waitig time	Response time
P1	5	5	32	27	22	10
P2	4	6	27	23	17	5
P3	3	7	33	30	23	3
P4	1	9	30	29	20	0
P5	2	2	6	4	2	2
P6	6	3	21	15	12	12

Probable Questions:

What do you know by CPU utilization?

What is Throughput in OS?

What do you mean by following terms:

- **Turnaround time**
- **Waiting time**
- **Response time**

Write the task of CPU scheduler.

Write briefly about the following algorithms.

- **First in First out**
- **SJF**
- **Optimal Replacement algorithm**
- **Roud Robin**

THANK YOU