

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD

Protected by PDF Anti-Copy Free

(Upgrade to Pro Version to Remove the Watermark)

Lecture-18 By Vivek Dubey

PDF

COMPUTER SCIENCE AND ENGINEERING

CS403PC

COMPUTER SCIENCE AND ENGINEERING (ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING)

CS403PC

BHARAT INSTITUTE OF ENGINEERING AND TECHNOLOGY

Scheme of Operating System

Protected by PDF Anti-Copy Free
(Upgrade to Pro Version to Remove the Watermark)



S. No.	Course Code	Course Title	L	T	P	Credits
3	CS403PC	Operating Systems	3	0	0	3
6	CS406PC	Operating Systems Lab	0	0	3	1.5

UNIT-2:

Protected by PDF Anti-Copy Free
(Upgrade to Pro Version to Remove the Watermark)



- **Process and CPU Scheduling –**
 - Process concepts and scheduling, Operations on Processes, Cooperating Processes, Threads, and Interposes Communication, Scheduling Criteria, **Scheduling Algorithms**, Multiple-Processor Scheduling.
- **System call interface for process management-**
 - fork, exit, wait, waitpid, exec

Scheduling Algorithms

Protected by PDF Anti-Copy Free
(Upgrade to Pro Version to Remove the Watermark)



Different Scheduling Algorithms

- *First Come First Serve (FCFS)*
- *Shortest Job First(SJF)*
- *Longest Job First(LJF)*
- *Shortest Remaining Time First(SRTF)*
- *Longest Remaining Time First(LRTF)*
- *Round Robin Scheduling*
- ***Priority Based scheduling (Non-Preemptive)***
- *Highest Response Ratio Next (HRRN)*
- *Multilevel Queue Scheduling*
- *Multi level Feedback Queue Scheduling*

Priority Based Scheduling Algorithm

Protected by PDF Anti-Copy Free
(Upgrade to Pro Version to Remove the Watermark)



- **Priority Scheduling** *is a method of scheduling processes that is based on priority.*
- In this algorithm, the scheduler selects the tasks to **work** *as per the priority.*
- The processes with higher priority **should be carried out first**, (whereas jobs with equal priorities are carried out on a round-robin or FCFS basis).
- Priority depends upon memory requirements, time requirements, etc.

Types of Priority Scheduling

Protected by PDF Anti-Copy Free
(Upgrade to Pro Version to Remove the Watermark)

PDF

Preemptive Scheduling

- In Preemptive Scheduling, the tasks are mostly assigned with their priorities. Sometimes it is important to run a task with a higher priority before another lower priority task, even if the lower priority task is still running. The lower priority task holds **for some time and resumes** when the higher priority task finishes its execution.

Non-Preemptive Scheduling

- In this type of scheduling method, the CPU has been allocated to a specific process. The process that keeps the CPU busy, will release the CPU either by switching context or terminating. It is the only method **that can be used for various hardware platforms**. That's because it doesn't need special hardware (for example, a timer) like preemptive scheduling.

Characteristics of Priority Scheduling



- A CPU algorithm that schedules processes based on priority.
- It used in Operating systems for performing batch processes.
- If two jobs having the same priority are READY, it works on a FIRST COME, FIRST SERVED basis.
- In priority scheduling, a number is assigned to each process **that indicates its priority level.**
- **Lower the number,** higher is the priority.
- In this type of scheduling algorithm, if **a newer process arrives,** that is having a higher priority than the currently running process, then the **currently running process is preempted.**

Example of Priority Scheduling

Protected by PDF Anti-Copy Free
(Upgrade to Pro Version to Remove the Watermark)

Consider following five processes P1 to P4. Each process has its unique priority and burst time.

Step 1: Start

Step 2: Accept the number of processes in the ready Queue

Step 3: For each process in the ready Q, assign the process id and accept the CPU burst time

Process	Priority	Burst time
P0	2	12
P1	1	2
P2	4	3
P3	1	5

Step 4: Sort the ready queue according to the priority number

Protected by PDF Anti-Copy Free
(Upgrade to Pro Version to Remove the Watermark)



Process	Priority	Burst time
---------	----------	------------

P0	2	12
----	---	----

P1	1	2
----	---	---

P2	4	3
----	---	---

P3	1	5
----	---	---

Sort

Process	Priority	Burst time
---------	----------	------------

P1	1	2
----	---	---

P3	1	5
----	---	---

P0	2	12
----	---	----

P2	4	3
----	---	---

Step 5: Set the waiting of the first process as “0” and its burst time as its turn around time

Process	Priority	Burst time	Waiting Time	Turnaround Time
P1	1	2	0	2
P3	1	5	2	7
P0	2	12	7	19
P2	4	3	19	22

Step-6 and 7

Protected by PDF Anti-Copy Free
(Upgrade to Pro Version to Remove the Watermark)



- Step-6
 - Calculate
 - Waiting time for process(n)
 - = waitingtime of process (n-1) + Burst time of process(n-1)
 - Turn around time for Process(n)
 - = waiting time of Process(n)+ Burst time for process(n)
 - Calculate
 - Average waiting time
 - = Total waiting Time / Number of process
 - Average Turnaround time
 - = Total Turnaround Time / Number of process
- Step 7: Stop



Advantages of priority scheduling

- Easy to use scheduling method.
- Processes are executed on the basis of priority *so high priority does not need to wait for long which saves time.*
- This method provides a good mechanism *where the relative important of each process may be precisely defined.*
- Suitable for applications with fluctuating time and resource requirements.

Disadvantages of priority scheduling



- If the system eventually crashes, *all low priority processes get lost.*
- If high priority processes take lots of CPU time, then the *lower priority processes may starve and will be postponed for an indefinite time.*
- This scheduling algorithm may leave some low priority processes *waiting indefinitely.*
- A process will be blocked *when it is ready to run* but has to wait for the CPU *because some other process is running currently.*
- If a new higher priority process keeps on coming in the ready queue, *then the process which is in the waiting state may need to wait for a long duration of time.*

Coding- Priority Scheduling

Protected by PDF Anti-Copy Free
(Upgrade to Pro Version to Remove the Watermark)



Code-1

```
#include<stdio.h>
int main()
{
int p[20],bt[20],pri[20], wt[20],tat[20],i, k, n, temp;
float wtavg, tatavg;
printf("Enter the number of processes : ");
scanf("%d",&n);
```

Coding- Priority Scheduling

Protected by PDF Anti-Copy Free
(Upgrade to Pro Version to Remove the Watermark)



Code-2

```
for(i=0;i<n;i++)  
{  
    p[i] = i;  
    printf("Enter the Burst Time & Priority of Process %d :",i);  
    scanf("%d %d",&bt[i], &pri[i]);  
}
```

Coding- Priority Scheduling

Protected by PDF Anti-Copy Free
(Upgrade to Pro Version to Remove the Watermark)



Code-3

```
for(i=0;i<n;i++)  
    for(k=i+1;k<n;k++)  
        if(pri[i] > pri[k])  
        {  
            temp=p[i]; p[i]=p[k]; p[k]=temp;  
            temp=bt[i]; bt[i]=bt[k]; bt[k]=temp;  
            temp=pri[i]; pri[i]=pri[k]; pri[k]=temp;  
        }
```

Protected by PDF Anti-Copy Free
(Upgrade to Pro Version to Remove the Watermark)

Coding- Priority Scheduling



Code-4

```
wtavg = wt[0] = 0;
tatavg = tat[0] = bt[0];
for (i=1;i<n;i++)
{
    wt[i] = wt[i-1] + bt[i-1];
    tat[i] = tat[i-1] + bt[i];
    wtavg = wtavg + wt[i];
    tatavg = tatavg + tat[i];
}
```

Code-5

Coding- Priority Scheduling

Protected by PDF Anti-Copy Free
(Upgrade to Pro Version to Remove the Watermark)



```
printf("\nPROCESS\t\tPRIORITY\tBURST TIME\tWAITING TIME\tTURNAROUND TIME");  
for(i=0;i<n;i++)  
printf("\n%d \t\t %d \t\t %d \t\t %d \t\t %d ",p[i],pri[i],bt[i],wt[i],tat[i]);  
printf("\nAverage Waiting Time is --- %f",wtavg/n);  
printf("\nAverage Turnaround Time is --- %f",tatavg/n);  
return 0;  
}
```

Output

Protected by PDF Anti-Copy Free
(Upgrade to Pro Version to Remove the Watermark)

Lecture-18 By Vivek Dubey



- Enter the number of processes : 4
- Enter the Burst Time & Priority of Process 0 :12 2
- Enter the Burst Time & Priority of Process 1 :2 1
- Enter the Burst Time & Priority of Process 2 :3 4
- Enter the Burst Time & Priority of Process 3 :5 1

PROCESS	PRIORITY	BURST TIME	WAITING TIME	TURNAROUND TIME
• 1	1	2	0	2
• 3	1	5	2	7
• 0	2	12	7	19
• 2	4	3	19	22

- Average Waiting Time is --- 7.000000
- Average Turnaround Time is --- 12.500000