Akhilesh Anil Shingare
72018540J
COMPSFEB0836

Group - B

Experiment No. 1

Title: Write a java program (using OOP features) to implement following scheduling algorithms: FCFS, SJF (preemptive), Priority (Non preemptive) and Round Robin (Preemptive)

Objectives: 1) To understand OS and SCHEDULING concepts
2) To implement Schedulling FCFS, SJF, RR & Priority algorithms.
3) To study about Scheduling and scheduler.

Problem Statement:
Write a Java program (using OOP features) to implement following scheduling algorithms: FCFS, SJF, Priority and Round Robin

Outcomes: After completion of the assignments student will be able to
- Knowledge Scheduling policies
- Compare different scheduling algorithms

Software Requirements: JDK / Eclipse

Hardware Requirements:
- M/C Lenovo Think centre M700 Ci3, 6100 6th Gen. H81, 4GB RAM, 500GB HDD

Theory Concepts:-

CPU scheduling:- CPU scheduling refers to a set of policies and mechanisms built into the operating systems that govern the order in which the work to be done by a computer system is completed.

What is Scheduling :-

Scheduling is defined as the process that governs the order in which the work is to be done. Schedullings done in the areas where more no. of jobs of works are to be performed. Then it requires some plan i.e. schedulling that means how the jobs are to be performed i.e. order CPU scheduling is best example of schedulling.

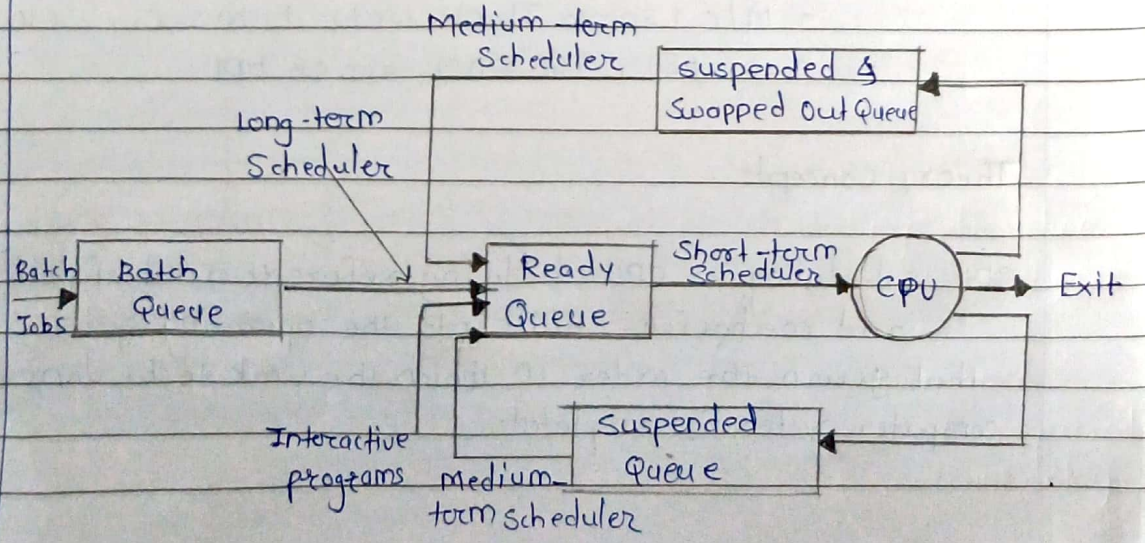
What is scheduler?

1. Scheduler in an Os module that selects the next job to be admitted into the system and the next process to run
2. Primary objective of the scheduler is to optimize system performance in accordance with the criteria deemed by the system designers. In short, scheduler is that module of os which schedules the programs in an efficient manner.

Types of scheduler:

In general, there are three different types of schedulers which may co-exist in a complex operating system.

- Long term scheduler.
- Medium term scheduler.
- Short term scheduler.



Scheduling Criteria:

- **CPU Utilization:**
  Keep the CPU as busy as possible. It range from 0 to 100%. In practice, it range from 40 to 90%.

- **Throughput:**
  Throughput is the rate at which processes are completed per unit of time.

- **Turnaround time:**
  This is the how long a process take to execute a process. It is calculated as the time gap between the submission of a process and its completion.

- **Waiting time:-**
  Waiting time is the sum of the time periods spent in waiting in the ready queue.

- **Response time:-**
  Response time is the time it takes to start responding from submission time - It is calculated as the amount of time it takes from when a request was submitted until the first response is produced.

- **Non-preemptive Scheduling:-**
  In non-preemptive mode, once if a process enters into running state, it continues to execute until it terminates or blocks itself to wait for input/output or by requesting some operating system service.

Preemptive Scheduling:

In preemptive mode, currently running process maybe interrupted and moved to the ready state by the operating system.
When a new process arrives or when an interrupt occurs, preemptive policies may incur greater overhead than non-preemptive version but preemptive version may provide better service.

It is desirable to maximum CPU utilization and throughput, and to minimize turnaround time, waiting and response time.

Types of scheduling algorithms:

- In general, scheduling disciplines may be pre-emptive or non-pre-emptive
- In batch, non-pre-emptive implies that once schedule a selected job turns to completion. There are different types of scheduling algorithms such as:
    - FCFS (First come Firstserve)
    - SJF (Short Job First)
    - Priority scheduling
    - Round Robin Scheduling algorithm.

First come first serve Algorithm:-

- FCFS is working on the simplest scheduling discipline
- The workload is simply processed in an order of their arrival, with no pre-emption.
- FCFS scheduling may result into poor performance
- Since there is no discrimination on the basis of required services, short jobs may considerable in turnaround delay and waiting time.

First Come, First served.

Process	Burst Time
P1	24
P2	3
P3	3

- Suppose that the processes arrive in the order: P1, P2, P3
- The Gantt chart for the schedule is:

- Waiting time for P1 = 0; P2 = 24; P3 = 24
- Avg waiting time = (0+24+27)/3 = 17

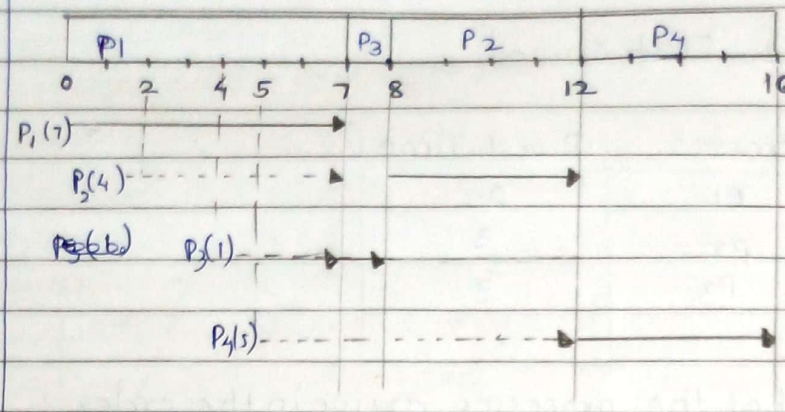
Shortest Job First Algorithm:

- This is known as shortest job first or SJF
- This is non-preemptive, pre-emptive scheduling algorithm.
- Best approach to minimize waiting time.

Ex.

Process	Arrival Time	Burst Time
P1	0	7
P2	2	4
P3	4	1
P4	5	4

Non preemptive SJF



P1's waiting time = 0
P2's waiting time = 6
P3's waiting time = 3
P3's waiting time = 7

Avg waiting time = (0+6+3+7)/4 = 4

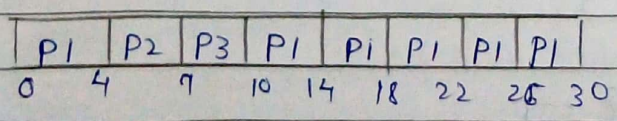
Round Robin Scheduling:

- Round Robin is the preemptive process scheduling algorithm
- Each process is provided a fix time to execute, it is called a quantum
- Once a process is executed for a given time period, it is premted and other process executes for a giventime period
- Context switching is used to save states of preempted procese.

Round Robin

Process	Burst Time
P1	24
P2	3
P3	3

Quantum time = 4 milliseconds
The Gantt Chart is :



Avg waiting time = {[0+(10-4)]+4+7}/3 = 5.6.

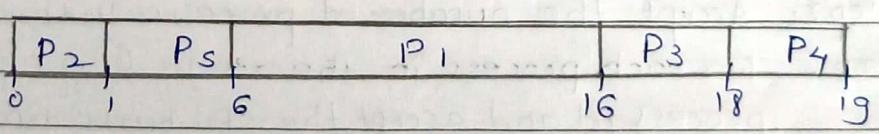
Priority Scheduling:-

- Priority scheduling is a non-preemptive algorithm and one of them most common scheduling algorithms in batch system
- Each process is assigned a priority. Process with highest priority is to be executed first and so on.
- Processes with same priority are executed on first come first served basis.
- Priority can be decided based on memory requirements time requirements or any other resource requirement.

Priority:

Process	Burst Time	Priority
P1	10	3
P2	1	1
P3	2	4
P4	1	5
P5	5	2

Gantt Chart:



Avg waiting time = (6+0+16+18+1) / 5 = 8.2

8 Algorithms (procedure):

FCFS:

step1: start the process
step2: Accept the number of processes in a ready Queue
step3: For each process in the ready Queue, assign the process id and accept the cpu Burst time
step4: set the waiting of the first process as '0' and its burst time as its turn around time
step5: For each process in the Ready Q calculate
(a) Waiting time for process (n) = waiting time of process (n-1) + Burst time of process (n-1)
(b) Turn around time for process (n) = waiting time of process (n) + Burst time process (n)
step6: calculate
(a) Avg waiting time = Total waiting Time / no. of processe
(b) Avg Turnaround time = Total Turnaround time / no. of proces
step7: Stop the process.

SJF:

step1: start the process
step2: Accept the number of processes in the ready Queue
step3: For each process in the ready Queue, assign the process id and accept the CPU burst time.
step4: Start the ready Q according the shortest Burst time by sorting according to lowest to Highest to time
step5: set the waiting time for the first process as '0' and its turnaround time as its burst time.
step6: For each process in the nearby ready queue, calculate
(1) Waiting time for process (n) = waiting time of process (n-1) + Burst time of process (n-1)

(d) Turnaround time for process (n) = waiting time of process (n) + Burst time for process (n)

step 7: Calculate

(c) Avg waiting time = Total waiting time / no of process

(d) Avg Turnaround time = Total turnaround time / no. of process.

step 8: Stop the process.

RR

step 1: Start the process

step 2: Accept the number of processes in the ready Queue and time quantum (or) time slice

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

step 4: Calculate the no. of time slices for each process where.

No. of time slice for process (n) = burst time process(n) // time slice

step 5: If the burst time is less than the time slice then the no. of time slices = 1.

step 6: Consider the ready queue is a circular Q. calculate

(a) Waiting time for process(n) = waiting time of process (n-1) + burst time of process (n-1) + the time difference in getting the CPU from process (n-1)

(b) Turnaround time for process (n) = waiting time of process (n) + burst of process (n) + the time difference in getting CPU from process(n)

step 7: Calculate

(c) Avg waiting time = Total waiting Time / No. of process

(d) Avg Turnaround time = Total turnaround time / No. of process

step 8: Stop the process.

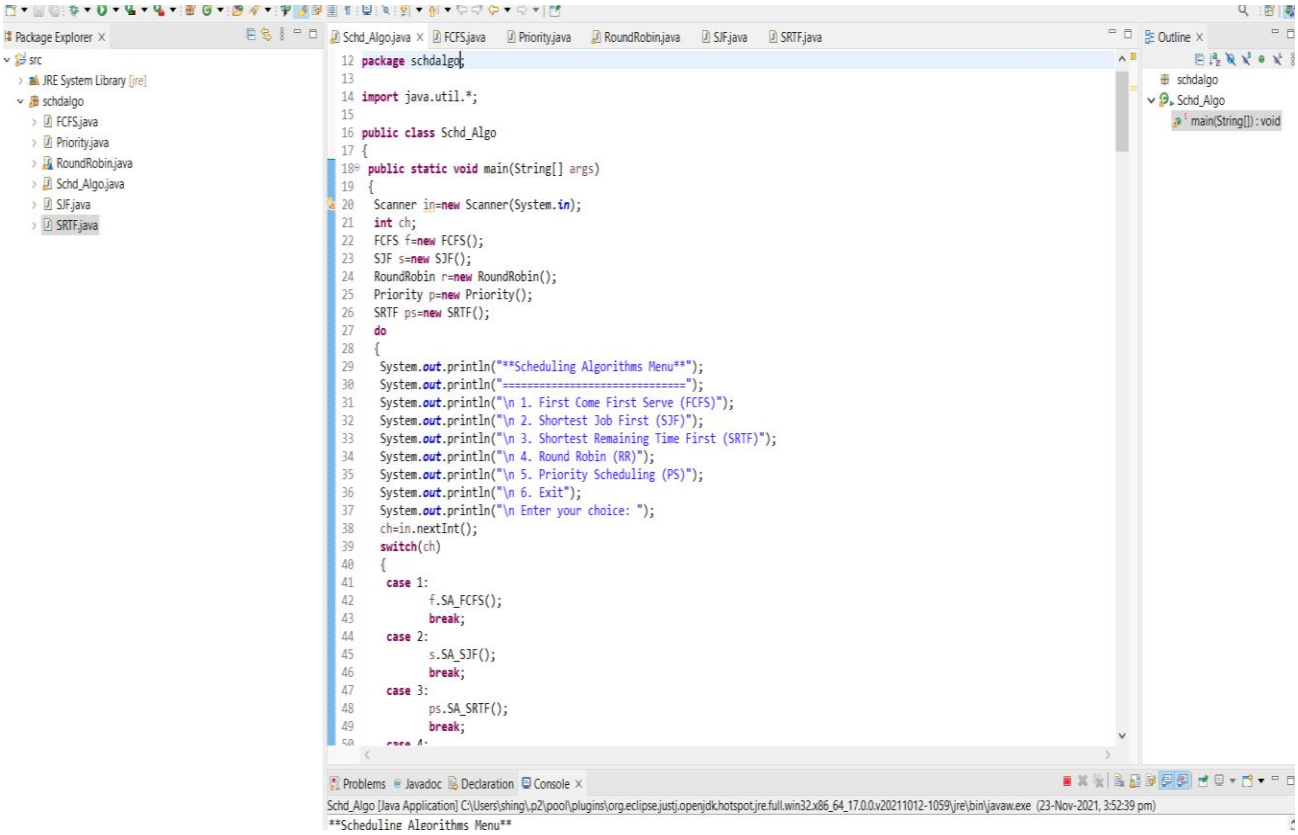
Priority scheduling:

Algorithms:

Step 1: Start the process
Step 2: Accept the number of processess in the ready Queue.
Step 3: For each process in the ready Q, assign the process id and accept the CPU burst time, priority
step 4: Start the Ready Q according the priority by sorting according to lowest to highest burst time and process.
Step 5: set the waiting time of the first processer as '0' and its turnaround time as its burst time
step 6: For each process in the ready queue, calculate
(a) Waiting time for process(n) = waiting time of process (n-1) + Burst time of process(n-1)
(b) Turn around time for process(n) = waiting time of process(n) + Burst time process(n)
step 6: calculate.
(c) Avg waiting time = Total waiting time / No. of process
(d) Avg Turnaround time = Total turnaround time / No. of process.
Step 7: Stop of process

9 Conclusion:-
Hence we have studied that -
- CPU scheduling concepts like context switching, types of schedulers, different timing parameters like waiting time, turnaround time, burst time, etc.
- Different CPU scheduling algorithm like FIFO, SJF, etc.

- FIFO is the simplest for implementation but produces large waiting times and reduces system performance.
- SJF allows the process having shortest burst time to execute first.



/* OUTPUT

Scheduling Algorithms Menu

==========

1. First Come First Serve (FCFS)

2. Shortest Job First (SJF)

3. Shortest Remaining Time First (SRTF)

4. Round Robin (RR)

5. Priority Scheduling (PS)

6. Exit

Enter your choice:

1

Enter the total no. of processes:

3

Enter the burst time for each process, starting with the first one

Burst time for Process 0:

12

Burst time for Process 1:

6

Burst time for Process 2:

9

Enter the arrival time for each process, starting with the first one

Arrival time for Process 0:

0

Arrival time for Process 1:

1

Arrival time for Process 2:

4

The data entered is:

Process 0:

Burst Time: 12

Arrival Time: 0

Process 1:

Burst Time: 6

Arrival Time: 1

Process 2:

Burst Time: 9

Arrival Time: 4

The Gantt Chart generated is:

0 12 18 27 Average Waiting Time: 8.333333333333334

Average Turnaround Time: 17.333333333333332

Scheduling Algorithms Menu

==========

1. First Come First Serve (FCFS)

2. Shortest Job First (SJF)

3. Shortest Remaining Time First (SRTF)

4. Round Robin (RR)

5. Priority Scheduling (PS)

6. Exit

Enter your choice:

2

Enter the total number of processes:

4

Enter the arrival time for process 0:

0

Enter the burst time for process 0:

7

Enter the arrival time for process 1:

2

Enter the burst time for process 1:

4

Enter the arrival time for process 2:

4

Enter the burst time for process 2:

1

Enter the arrival time for process 3:

5

Enter the burst time for process 3:

4

Gantt Chart

0

7

11

12

16

The Average Waiting Time is: 4

The Average Turnaround Time is: 8

Scheduling Algorithms Menu

==========

1. First Come First Serve (FCFS)

2. Shortest Job First (SJF)

3. Shortest Remaining Time First (SRTF)

4. Round Robin (RR)

5. Priority Scheduling (PS)

6. Exit

Enter your choice:

3

Enter the Total Number of Processes:

4

Enter the arrival time for process 0:

0

Enter the burst time for process 0:

8

Enter the arrival time for process 1:

1

Enter the burst time for process 1:

4

Enter the arrival time for process 2:

2

Enter the burst time for process 2:

9

Enter the arrival time for process 3:

3

Enter the burst time for process 3:

5

Average Waiting Time: 6.5

Average Turnaround Time: 13.0

Scheduling Algorithms Menu

==========

1. First Come First Serve (FCFS)

2. Shortest Job First (SJF)

3. Shortest Remaining Time First (SRTF)

4. Round Robin (RR)

5. Priority Scheduling (PS)

6. Exit

Enter your choice:

4

Enter the total number of processes:

3

Enter the Burst Time for 0:

24

Enter the Burst Time for 1:

3

Enter the Burst Time for 2:

3

Enter the Time Quantum:

4

Process0:

Burst Time: 24

Waiting Time: 6

Turnaround Time: 30

Process1:

Burst Time: 3

Waiting Time: 4

Turnaround Time: 7

Process2:

Burst Time: 3

Waiting Time: 7

Turnaround Time: 10

Average Waiting Time: 5.6666665

Average Turnaround Time: 15.666667

Scheduling Algorithms Menu

==========

1. First Come First Serve (FCFS)

2. Shortest Job First (SJF)

3. Shortest Remaining Time First (SRTF)

4. Round Robin (RR)

5. Priority Scheduling (PS)

6. Exit

Enter your choice:

5

Enter the total no. of processes:

5

Enter the burst time for Process 0:

10

Enter the priority of Process 0:

3

Enter the burst time for Process 1:

1

Enter the priority of Process 1:

1

Enter the burst time for Process 2:

2

Enter the priority of Process 2:

4

Enter the burst time for Process 3:

1

Enter the priority of Process 3:

5

Enter the burst time for Process 4:

5

Enter the priority of Process 4:

2

The data entered is:

Process: 0

Burst Time: 10

Priority: 3

Process: 1

Burst Time: 1

Priority: 1

Process: 2

Burst Time: 2

Priority: 4

Process: 3

Burst Time: 1

Priority: 5

Process: 4

Burst Time: 5

Priority: 2

The Gantt Chart generated is:

0 1 6 16 18 19 Average Waiting Time: 8.2

Average Turnaround Time: 12.0

Scheduling Algorithms Menu

==========

1. First Come First Serve (FCFS)

2. Shortest Job First (SJF)

3. Shortest Remaining Time First (SRTF)

4. Round Robin (RR)

5. Priority Scheduling (PS)

6. Exit

Enter your choice:

6