Advanced Operating Systems
COEN 383-01 Project-2

Group 2
(Tanmay Singla, Stephen Tambussi, Sushmita Sathyanarayanan, Srishti Kumari, Prakarsha Dahat)

Objective:- The Objective of this project is to get the experience of various Process Scheduling Algorithms. We have written a C program that runs on the following algorithms:-

- First come first-served (FCFS) [non-preemptive]
- Shortest job first (SJF) [non-preemptive]
- Shortest remaining time (SRT) [preemptive]
- Round robin (RR) [preemptive]
- Highest priority first (HPF) [non-preemptive]
- Highest priority first (HPF) [preemptive]

Constraints:-

- Arrival Time of process should be less than 100 quanta.
- Arrival Time, Expected Run Time & Priority are generated randomly.
- Only One process Queue exists**.**
- There is no I/O time.
- Time Slice for Round Robin is 1 quantum.
- Use 4 queues for HPF.

The following is the final statistical output obtained from the 6 algorithms implemented on 52 processes generated by considering all the constraints.
The average of the 5 runs of all algorithms are as follows:

First Come First Serve (Non-Preemptive):

```
Average Response Time: 24.0
Average Wait Time: 24.4
Average Turn Around Time: 29.3
Average Throughput: 20.0
```

Shortest Job First (Non-Preemptive):

```
Average Response Time: 3.3
```

```
Average Wait Time: 3.7
Average Turn Around Time: 7.0
Average Throughput: 27.0
```

Shortest Remaining Time First (Preemptive):

```
Average Response Time: 1.8
Average Wait Time: 2.9
Average Turn Around Time: 5.9
Average Throughput: 29.0
```

Round Robin (Preemptive):

```
Average Response Time: 13.2
Average Wait Time: 48.1
Average Turn Around Time: 53.0
Average Throughput: 27.0
```

Highest Priority First (Non-Preemptive):

```
Average Response Time: 10.0
Average Wait Time: 10.2
Average Turn Around Time: 12.7
Average Throughput: 18.0
```

Highest Priority First (Preemptive):

```
Average Response Time: 4.6
Average Wait Time: 6.3
Average Turn Around Time: 8.0
Average Throughput: 54.0
```

The following observations were made from the results by running the 6 algorithms:

1. **First Come First Serve (Non-Preemptive)**: The response time, Wait time and turnaround time are high and as a result, the throughput is lower than the others for this algorithm. Since the new processes have to wait until the earlier process completes execution, there are chances of starvation. However FCFS is easy to implement.

2. **Shortest Job First (Non-Preemptive)**: The response time, Wait time and turnaround time are lower compared to other algorithms, but for processes which have high burst time, there may be cases of starvation.

3. **Shortest Remaining Time First (Preemptive):** The response time, wait time and turnaround time are low and similar to the Shortest job first algorithm. The jobs which have least remaining time to complete execute first in this algorithm.

4. **Round Robin (Preemptive)**: For this algorithm, all the processes get an equal time slice for execution. Due to this all processes in the queue get CPU for a limited time and completion of long processes does not take place in a single turn after CPU allocation. This majorly increases the turn around time of all the processes and the response time and wait time also increases in all the processes resulting in less throughput than the above algorithms. However a key challenge is deciding the duration of the time slice. If the time slice is large, the results are similar to FCFS and if we select a small time slice, there is high overhead of context switching.

5. **Highest Priority First (Non-Preemptive):** This algorithm has lowest throughput as compared to other algorithms. The Highest priority First Preemptive has better results compared to this algorithm. This algorithm reduces the amount of starvation that occurs among long processes.

6. **Highest Priority First (Preemptive):** Among all the observed algorithms, the HPF preemptive algorithm has the best throughput, though the processes with lesser priority can run into starvation. Due to its preemptive nature, the newer processes with high priority are allowed to run quickly. This scheduling algorithm, as per the observations, has very less response time, wait time and turn around time as preemptiveness allows newer processes with high priority to run quickly.

Conclusion:-

We see that the Highest priority first (preemptive) algorithm has the highest throughput among the algorithms considered. The response time, wait times and turnaround time are lowest for SRTF Preemptive and HPF Preemptive. Highest priority first (Non-Preemptive) has the lowest throughput and Round Robin has the highest wait time for running the processes.