

Lab Assignment 3

Title: CPU scheduling Algorithms (FCFS and Round Robin)

Problem statement:

To implement the following CPU scheduling algorithms in C/C++

- 1) FCFS(Non-Preemptive)
- 2) Round Robin(Preemptive)

Objectives:

- To understand the concept of CPU scheduling
- To understand need of CPU scheduling
- To use C programming to implement the algorithm
- To understand FCFS and Round robin approach

Theory:

Types of scheduling policies:

- FCFS (First Come First Serve)
- SJF (Shortest Job First)
- Priority scheduling
- Round Robin scheduling

1. FCFS (First Come First Serve):

2. **Selection Function:** $\max(w)$, selects the process which is waiting in the ready queue for maximum time.
3. **Decision Mode:** Non_preemptive
4. **Throughput:** Not emphasized
5. **Response Time:** May be high, especially if there is a large variance in the process execution times.
6. **Overhead:** Minimum
7. **Effect on Processes:** Penalizes short processes
8. **Starvation:** No

- **Completion Time**

Time at which process completes its execution.

- **Turn Around Time**

Time Difference between completion time and arrival time.

Turn Around Time = Completion Time – Arrival Time

- **Waiting Time(W.T)**

Time Difference between turn around time and burst time.

Waiting Time = Turn Around Time – Burst Time

2. Round Robin Algorithm

- Each process gets a small unit of CPU time (*time quantum*)
- After time has elapsed, the process is preempted and added to the end of the ready queue.
- If there are n processes in the ready queue and the time quantum is q , then each process gets $1/n$ of the CPU time in chunks of at most q time units at once.
- No process waits more than $(n-1)q$ time units

- **Selection Function:** constant
- **Decision Mode:** Preemptive (At time quantum)
- **Throughput:** May be low if time quantum is too small
- **Response Time:** Provides good response time for short processes
- **Overhead:** Minimum
- **Effect on Processes:** Fair treatment

- **Starvation:** No

Input:

- 1) Total number of processes
- 2) Time-quantum (for RR scheduling)

For each process:

- 1) CPU burst time (ms)
- 2) Arrival time (ms)

Output:

For each algorithm:

- 1) Gantt Chart
- 2) Waiting time (ms)
- 3) Average Waiting time (ms)
- 4) Turnaround time (ms)
- 5) Average Turnaround time (ms)

Conclusion: Thus, we have studied and implemented the concept of CPU scheduling algorithms.

FAQs

1. Explain the need of CPU scheduling.
2. Explain non-preemptive and preemptive decision mode.
3. Explain FCFS algorithm and Round Robin algorithm with examples.