Introduction

In various circumstances, not every algorithm works better on the significant job. Sometimes, FCFS algorithm is better than the other in short burst time while Priority Scheduling is better for multiple processes in every single time. However, it cannot be expected what process will come after. Therefore, we want to compare two algorithms between FCFS and Priority Scheduling. The target is to know which is the best scheduling algorithm to implement in finding the best Class Scheduling solution.

Priority Scheduling Algorithm 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. In Priority Preemptive Scheduling, the tasks are mostly assigned with their priorities. Meanwhile, in the Priority Non-preemptive scheduling method, the CPU has been allocated to a specific process. Processes are executed on the basis of priority so high priority does not need to wait for long which saves time. If high priority processes take lots of CPU time, then the lower priority processes may starve and will be postponed for an indefinite time.

FCFS is the simplest operating system scheduling algorithm that automatically executes queued requests and processes by order of their arrival. It supports non-preemptive and preemptive scheduling algorithms. A real-life example of the FCFS method is buying a movie ticket on the ticket counter. It is a Non-Preemptive CPU scheduling algorithm, so after the process has been allocated to the CPU, it will never release the CPU until it finishes executing.

Consideration

FCFS could not stimulate priority scheduling for all possible parameters of priority scheduling because suppose job one(1) arrives at time one(1) with priority ten(10) and length 1,000,000 and job two(2) arrives at time two(2) with priority one(1) and length 1,000,000. FCFS will run job one(1) to completion, then job two(2) to completion. Priority will run job one(1) for one(1) unit, then job two(2) to completion, then job one(1) to completion. Meanwhile, priority scheduling can stimulate FCFS for all possible parameters of FCFS as we assume that the priority queue breaks ties with FCFS and infinite time quantum meaning that all jobs equal priority.

Analysis

1. **Priority Scheduling**

2. FCFS

Three processes arriving at different times. Each process has a different waiting time. Table 1 shows the data in the process.

Table 1 Data Sampling.

Process	Arrival Time	Waiting Time
P1=2201	1	0
P2=3401	2	3
P3=1103	3	5

Let's consider that Process is P, Arrival Time is AT, Waiting Time is WT.

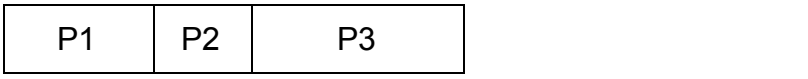
$$TWT = \sum WT$$

Calculate the Total Waiting Average Waiting Time of the FCFS algorithm.
It produces the total waiting time and finally the average waiting time is obtained

TWT =WT1+WT2+WT3
=0+3+5
=8

AWT =TWT/TOTAL PRECESS
=8/3
=2.66667

The average waiting time will be 2.66667 ms.



This is the GANTT chart for the above processes

For the above given processes, first P1 will be provided with the CPU resources hence; waiting time for P1 will be 0. P1 requires 3 ms for completion, hence

waiting time for P2 will be 3ms. Similarly, waiting time for princess P3 will be execution time of P1 + execution time for P2, which will be (3+2)ms = 5ms.

CONCLUSION

The calculation of two algorithms shows the different average waiting time. The FCFS is better for small burst time. The priority schedule?
All algorithms are good, but the speed of the process depends on the processor load.