ESPINA, Allain Daniel S.

3ITF

Asst. Prof. Bernard G. Sanidad, DIT, DBA

November 23, 2023

# HOMEWORK 3: PROCESS SCHEDULING ALGORITHMS

Directions:

Research on Six (6) Different Process Scheduling Algorithms. Provide examples for each.

1. First Come First Serve (FCFS)

The First Come First Serve Scheduling Algorithm is one of the simplest and easiest to understand and implement among different types of Operating System Scheduling Algorithms. As the name implies, the process who requests for the CPU first will be the one to acquire the CPU resources first. This process scheduling algorithm follows the concept of FIFO, or "First In, First Out," typically implemented in Queues. The First Come First Serve Algorithm is an example of Preemptive Scheduling, where the process will not have to wait in queue before the ongoing process terminates.

2. Shortest Job Next (SJN)

The Shortest Job Next Scheduling Algorithm, also known as Shortest Job First, selects the process with shortest execution time to be executed next. This process scheduling algorithm reduces the average waiting time for all process but in the case that a lot of process with short execution time keeps on requesting resources, starvation may occur for processes with longer waiting times. This disadvantage, however, can be solved through the implementation of Aging, or the process of increasing the priority of processes that are waiting in the system for a long time.

3. Round Robin

Round Robin Scheduling is an algorithm wherein each process is provided a fixed amount of time, called quantum, for it to execute. This algorithm prevents starvation as each process holds their required CPU resources for only a limited amount of time, thus, making it another example of Preemptive Scheduling. This type of scheduling provides balance in the sense that every process gets an equal share of CPU resources and execution time.

4. Shortest Remaining Time (SRT)

Shortest Remaining Time Scheduling is an algorithm that provides CPU resources to the process closest to its completion, but it can be preempted by a newer process with a shorter completion time. This algorithm is the preemptive version of the Shortest Job First Algorithm. However, unlike its non-preemptive counterpart, this algorithm's processing time is much faster in comparison. However, despite SRT Scheduling's advantages over SJF Scheduling, the two algorithms share a common problem in the form of Starvations.

5. Highest Response Ratio Next

The Highest Response Ratio Next Scheduling is an optimized version of the Shortest Job First Algorithm to solve the latter's issues with the occurrence of Starvation. As the name implies, the process with the highest response ration will be given the highest priority and, therefore, will be executed next. This non-preemptive algorithm allows for the execution of shorter processes while simultaneously reducing the wait time of longer processes. It must be noted, however, that this algorithm is susceptible to the occurrence of CPU overload.

6. Priority Based

Priority Based Scheduling involves the assignment of priorities among the different processes available. The process given with the highest priority will be executed first, and in the case that two process which possess the same priority value is encountered, the system will execute the processes on a first come first serve basis. This scheduling algorithm is simpler and offers a lower average waiting time in comparison to the FCFS Scheduling Algorithm. However, similar to other scheduling schemes, Priority Based Scheduling is also prone to Starvation as the system will preempt the process undergoing execution once it encounters a process with higher priority.

References:

*Difference between preemptive and non-preemptive scheduling - javatpoint*. www.javatpoint.com. (n.d.-a). https://www.javatpoint.com/preemptive-vs-non-preemptive-scheduling

GeeksforGeeks. (2023a, March 23). *Starvation and aging in operating systems*. GeeksforGeeks. https://www.geeksforgeeks.org/starvation-and-aging-in-operating-systems/

GeeksforGeeks. (2023b, May 1). *CPU scheduling in operating systems*. GeeksforGeeks. https://www.geeksforgeeks.org/cpu-scheduling-in-operating-systems/#fcfs

*Operating system scheduling algorithms*. tutorialspoint. (n.d.). https://www.tutorialspoint.com/operating_system/os_process_scheduling_algorithms.htm

*Scheduling algorithms in OS (operating system) - javatpoint*. www.javatpoint.com. (n.d.-b). https://www.javatpoint.com/os-scheduling-algorithms