CSCE 5640

Operating System Design

Project Proposal

Scheduling Algorithms

Group-8

Overview

In this project we like to implement different types of algorithms like FCFS,Round Robin,SJF, SRTF,Priority Scheduling, Priority with round robin, Multilevel queue and Multi level priority queue which has different types of pros and cons. We like to implement algorithms which perfectly align with requirements of the client.

FCFS(First come First serve):

FCFS is an operating system scheduling algorithm which automatically executes queued requests and processes in order of their arrival. It is one of the easiest and simplest CPU scheduling algorithms, processes which request the CPU first get the CPU allocation first. It is managed by the First In First Out queue.

A real life example is like reserving a table in the restaurant or booking a movie ticket/bus ticket.

SJF(Shortest Job First):

SJF(Shortest Job First) depends on average running time of the processes. The algorithm SJF selects the process with the shortest execution time as the one to run next. This scheduling can be both preemptive or non preemptive. It significantly reduces the average waiting time for other processes lined for execution. Shortest job first is also known as shortest job next or shortest process next.

One of the applications of SJF in real life is food delivery. Online delivery apps always choose to deliver the nearest order first as it searches for the first nearest delivery location.

Priority scheduling:

Priority is used as the foundation for scheduling decisions made during the scheduling process.

With this method, the scheduler decides which jobs to execute based on their relative importance. Processes with higher priorities should be completed first, while those with equal priorities are completed in a round-robin or first-come, first-served (FCFS) fashion. Priorities are set according to resource constraints (time, space, etc.).

Priority Scheduling is of two types: Priority Preemptive Scheduling, and Priority Non-Preemptive Scheduling.

In Priority Preemptive Scheduling, the tasks are mostly assigned with their priorities.

In terms of Priority Through a non-preemptive scheduling technique, the central processing unit is committed to a single task.

High-priority processes do not have to wait as long since they are run first. If high-priority tasks use a disproportionate amount of CPU time, then lower-priority tasks may starve and be delayed indefinitely.

Round Robin (RR) Scheduling:

An example of CPU scheduling, Round Robin assigns tasks to running periods of time in a recursive fashion. This technique is a preemptive version of the First Come, First Served CPU Scheduling model. It is the earliest and most basic scheduling technique and is mostly employed for multitasking.

The fundamental advantage of utilizing a round-robin scheduling method is that, given the total number of processes in the run queue, you can forecast the worst-case response time for a specific process.

Priority Round Robin:

The priority-based Round-Robin CPU Scheduling method is a hybrid of the round-robin and priority scheduling techniques. The advantage of not having anybody go hungry is preserved, and the advantage of prioritization in scheduling is included as well.

Because of its high context transition rates, long waiting time, long reaction time, long turnaround time, and low throughput, the current round robin CPU scheduling method is not suitable for use in real time operating systems. The suggested technique fixes every problem with the traditional round-robin approach to scheduling central processing units.

Multilevel queue:

In a general system, from processes in the ready queue, some processes require scheduling using a priority algorithm, while some processes want to stay in the system **interactive processes,** others are **background processes** whose execution can be delayed based on its own scheduling needs.

In such cases an algorithm multi-level queue scheduling divides the ready queue into numerous distinct queues according to a characteristic of the processes, such as memory size, priority, or process type; they are often permanently assigned to one queue. The scheduling algorithm for every queue can be different.

Multilevel Priority queue:

By providing a separate priority level for each of the processes, the Multi-Level Priority Queues (MPQ) feature enables you to build multiple priority queues for numerous processes.

Objective

The CPU's total count and waiting for I/O of some form alternate in almost all programs. (Even a straightforward memory fetch requires a significant amount of time compared to CPU speeds).
The full utilization of otherwise wasted CPU cycles is achieved by allowing one process to utilize the CPU while another waits for I/O.
As long as the OS has at least one task in the ready queue that is ready for execution, the major goal or objective of CPU scheduling algorithms is to prevent the CPU from ever being in an idle state by fair allocation of CPU time to every process. The other main objectives are to maximize throughput and to minimize turnaround time and response time.

Team size and Team Members:

Our team is of 4 people

- Satish babu nalajala
- Dunnala Naveen
- Ishitha Konda
- Hari prasad Reddy sheelam

Project Plan:

Task divisions for the team members:

Student ID	Student Name	Email	Task
11551660	Dunnala Naveen	naveendunnala@my.unt.edu	Round Robin, Priority Round Robin,multilevel queue
11524982	Satish babu nalajala	satishbabunalajala@my.unt.edu	FCFS,SJF, Priority scheduling
11592291	Ishitha Konda	ishithakonda@my.unt.edu	Priority round robin, multilevel priority queue,SJF,round robin
11601562	Hari prasad Reddy sheelam	hariprasadreddysheelam@my.unt.edu	FSCS,priority scheduling,multi-level queue

Due date for subtasks:

- Implementation of First come First serve algorithm.
- Implementation of Shortest Job First algorithm.
- Implementation of Priority scheduling algorithm.
- Implementation of Round Robin (RR) Scheduling algorithm.
- Implementation of Priority Round Robin algorithm.
- Implementation of Multilevel queue algorithm.
- Implementation of Multilevel Priority queue algorithm.

Experimental environment:

Programming language for implementation:

- Programming languages used are C++

Operating system to test the project:

- Windows/Ubuntu

Test cases:

Manual testing by providing different scenarios whether the code provides high accuracy or not.