MLFQ Scheduler Implementation

Overview

This project implements a **Multi-Level Feedback Queue (MLFQ) scheduling algorithm** using multithreading in C. The scheduling consists of four queues with different scheduling algorithms:

1. **Q0 (Round-Robin - RR)**
2. **Q1 (First-Come, First-Served - FCFS)**
3. **Q2 (Priority Scheduling - Inverse of CPU time needed)**
4. **Q3 (Shortest Job First - SJF)**

Each process is assigned a maximum of **5ms** per queue. If a process is not completed after reaching Q3, it is moved back to Q0. The scheduling repeats until all processes finish execution. The program calculates and prints the **turnaround time for each process** and the **average turnaround time**.

Prerequisites

Ensure you have a **C compiler (GCC recommended)** and POSIX thread libraries installed. If using Debian-based systems:

```
sudo apt update && sudo apt install build-essential
```

Compilation

To compile the program, run:

```
gcc mlfq.c
```

Execution

Run the compiled program:

```
./a.out
```

Input Format

The program takes user input for the number of processes and their CPU burst times:

```
Enter number of processes: 5
Enter CPU burst time for each process:
300 150 250 350 450
```

Output Format

The program prints scheduling information and turnaround times for each process. Example output:

```
In RRB, process 0, time remaining 295, current time 5
In RRB, process 1, time remaining 145, current time 10
...
Process Turnaround Times:
Process 0: 1500 ms
Process 1: 1395 ms
...
Average Turnaround Time: 1210.00 ms
```