Operating Systems - Assignment 2
(CPU Scheduling Algorithms)

Motivation:

In class, you have learned many CPU scheduling algorithms. This assignment asks you to implement some of these algorithms and propose your own one.

Assignment Content:

In this assignment, you are asked to schedule ten processes. The data structure for a process is as follows:

```
struct process{
    int identification;   //process ID
    int arrival_time;     //arrival time of the process (in ms)
    int burst_time;       //required CPU time for the process (in ms)
    int priority;         //process's priority
}
```

The values of some variables can be set as follows:

- arrival_time: Randomly picked from three values: 0, 5, and 10.
- burst_time: Randomly picked from the range of [6, 28].
- priority: Randomly picked from the range of [1, 3]. A smaller number means a higher priority.

Please implement the following three scheduling algorithm:

- Preemptive SJF scheduling
- Round-robin (RR) scheduling + nonpreemptive priority scheduling: The size of time quantum for RR scheduling is set to 5ms.
- Multilevel feedback queue scheduling: There are three queues:
  - Queue Q_0: RR scheduling with time quantum of 5ms
  - Queue Q_1: RR scheduling with time quantum of 10ms
  - Queue Q_2: FCFS scheduling

Moreover, please propose your own scheduling algorithm. This algorithm cannot be identical to anyone taught in class (or just modify the parameters), but it can be improved from an algorithm in the textbook.

For each algorithm, you need to show its Gantt chart, the waiting time, the turnaround time, and the number of context switch. For waiting time and turnaround time, you need to list their calculation formulas. You must also demonstrate the advantages of your proposed algorithm. For example, your algorithm can outperform RR + priority scheduling in terms of the turnaround time.

Requirements:

- There is no restriction on the programming language you use in this assignment, but you should prepare the execution environment when performing the demonstration.
- You must submit a README file along with your program. The README file should briefly describe how you write your codes (for example, the idea of your program).
- You must demonstrate your program. TA will announce the demonstration time.

Grading Policy:

You need to submit your codes and demonstrate your program to TA. The due date of this homework is **5/28**. You will get **no point** if you do NOT demonstrate your program (even if you submit codes). Discussion among your classmates is encouraged. However, plagiarists (including copying code from ChatGPT) will get **ZERO point**. Below are the points you can get in this homework:

- Preemptive SJF scheduling: 20%
- RR + priority scheduling: 20%
- Multilevel feedback queue scheduling: 20%
- Your scheduling algorithm: 30%
- Code comments & README: 10%