Operating Systems, Assignment 2

CS 4345 — Dr. Mihail
Department of Mathematics Computer Science
Valdosta State University
Due: June 27th before midnight

June 19, 2021

1 Problem statement

Write a program simulating a simple CPU scheduler to calculate average waiting time.

1.1 Detailed Description

Initial ready queue has 6 processes where each process has the following attributes:

- an identifier (value between 0 and 10, randomly assigned – however, **no two processes can have the same id)**
- a burst length between 20 – 100, randomly assigned when it is created
- a priority between 1 and 10 (low value indicates higher priority), randomly assigned when it is created (two or more processes may have same priority rank).

1.2 Requirements

- Requirement 1: Display initial snapshot of the system with the information of above 6 processes as follows (order the entries in increasing ID):
  ProcessID | Priority | Burst-length

- Requirement 2: Now, allow user of the program to enter attributes (id, priority, and burst length) of another process (the 7th one). Make sure the following:
  - user may enter any integer as a process ID. Your program must check the appropriate range (0 to 10). Also, user entered value should not be a duplicate of already created IDs. Any of these violated, user must be notified and given chance to enter a new ID.
  - User may enter any integer as burst length. Your program must check the appropriate range (20 to 100). If violated, user gets chances until an acceptable value is entered.
  - User may enter any integer as a priority value. Your program must check the appropriate range (1 to 10). If violated, user gets chances until an acceptable value is entered.

- Requirement 3: Display updated snapshot of the system (with 7 processes now) in the same format as above (Requirement 1).

  Now, assuming all processes are ready for scheduling, calculate individual waiting time for each process using the following algorithms: (i) Non-preemptive SJF, (ii) Non-preemptive priority, and (iii) round robin with time quantum 20.

- Requirement 4: Display the result as follows:
  ProcessID | Priority | Burst-length | Scheduling Algorithm | Waiting time

  The above snapshot display should be grouped by algorithm. That is, all processes' SJF waiting times first, followed by all priority waiting times, and finally all RR waiting times.

- Requirement 5: Based on your snapshot and calculations, also display average waiting time for each algorithm and order the algorithms from lowest to highest average waiting time.

Notes: You may work with absolute values ignoring the time unit. That is, a process may have burst length 46 – you may ignore whether it is 46 seconds or 46 milliseconds.

Submission: You can use Java or C to solve the problem. Include your name, course name, semester, and assignment identifier (Assignment 3) as program comments at the top of your code. Submit the source file(s) through BlazeVIEW submission box.