

CS 3305A: Operating Systems
Department of Computer Science
Western University
Assignment 4
Fall 2021
Due Date: November 8th

Purpose:

The goals of this assignment are the following:

- Gain more experience with the C programming language from an OS's *CPU scheduling* perspective.
- Get hands-on experience with the *CPU scheduling algorithms*.

Performance Evaluation of CPU Scheduling Algorithms

You will be applying *CPU Scheduling Algorithms* in the C programming language. A sample input file *rr_input.txt* is provided with the assignment, which must be used to develop the **Round Robin (RR)** *CPU Scheduling Algorithm*. You must use the same file name (i.e., *rr_input.txt*) in your code.

Format of the Input File:

The input file *rr_input.txt* contains test cases for this assignment. Every line of the input file represents one individual test case. For example, if there are four lines inside the input file, it means your program must execute four different test cases. Every input line consists of several processes and their corresponding information such as *process name*, *arrival time (art₁)*, *burst time (brt₁)*, and *quantum time (q_{time})* using the following format:

Input format: process1 art₁ brt₁ process_name2 art₂ brt₂ q_{time}

Input example: P1 0 23 P2 1 3 P3 2 3 4

In the example input given above, there are three processes such as P1, P2, and P3. The arrival and burst time of process P1 is 0 and 23, process P2 is 1 and 3, and process P3 is 2 and 3. The value of the quantum time is 4. The individual value in the input line is separated by space.

What you need to do:

You must parse the given input file *rr_input.txt* before applying the RR algorithm. You must not use any other format, such as user input from the terminal, command-line argument, etc., for providing input to the program. Also, you must not modify the format of the input file. Any changes to the input file or input-taking procedure will not be considered a successful submission for this assignment. CPU scheduling algorithm RR must be applied to each input test case. The output of your program must follow the format of the sample output given below. You must consider two decimal points for printing the fractional number. Any other format will be regarded as for deducting marks.

Sample Input File (see the *rr_input.txt* file):

P1 0 24 P2 1 3 P3 2 3 4

P1 0 10 P2 1 5 P3 2 8 P4 3 9 2

Sample Output:

Process: P1 Arrival Time: 0 Burst time: 24 Waiting Time: 6 Turnaround Time: 30

Process: P2 Arrival Time: 1 Burst time: 3 Waiting Time: 3 Turnaround Time: 6

Process: P3 Arrival Time: 2 Burst time: 3 Waiting Time: 5 Turnaround Time: 8

Total Turnaround Time: 44

Average waiting time = 4.67s

Average turnaround time = 14.67

Process: P1 Arrival Time: 0 Burst time: 10 Waiting Time: 21 Turnaround Time: 31

Process: P2 Arrival Time: 1 Burst time: 5 Waiting Time: 13 Turnaround Time: 18

Process: P3 Arrival Time: 2 Burst time: 8 Waiting Time: 17 Turnaround Time: 25

Process: P4 Arrival Time: 3 Burst time: 9 Waiting Time: 20 Turnaround Time: 29

Total Turnaround Time: 103

Average waiting time = 17.75

Average turnaround time = 25.75

Computing Platform for Assignments

You are responsible for ensuring that your program compiles and runs without error on the computing platform mentioned below. **Marks will be deducted** if your program fails to compile or runs into errors on the specified computing platform (see below).

- Students have virtual access to the MC 244 lab, which contains 30 Fedora 28 systems. Linux machines available to you are **linux01.gaul.csd.uwo.ca** through **linux30.gaul.csd.uwo.ca**.
- It is your responsibility to ensure that your code compiles and runs on the above systems. You can SSH into MC 244 machines (please see the Assignment 1 file transfer tutorial).
- If you are off-campus, you have to SSH to **compute.gaul.csd.uwo.ca** first (this server is also known as **sylvia.gaul.csd.uwo.ca**, in honor of Dr. Sylvia Osborn), and then to one of the MC 244 systems (**linux01.gaul.csd.uwo.ca** through **linux30.gaul.csd.uwo.ca**) (please see the Assignment 1 file transfer tutorial).
- <https://wiki.sci.uwo.ca/sts/computer-science/gaul>

Assignment Submission

You need to submit only one C file. The name of your submitted C file must be **“assignment4.c”**. Marks will be deducted if your submitted C file name is different. You must submit your assignment through OWL. Be sure to test your code on one of MC 244 systems (see “Computing Platform for Assignments” section above). **Marks will be deducted** if your program fails to compile or runs into errors on the computing platform mentioned above.

Assignment 4 FAQ will be made available on OWL as needed. Also, consult TAs and the Instructor for any questions you may have regarding this assignment.