**Student Name:** Mohammed Nihad K
**Student ID:** 11802815
**Email Address**: nihadkaipalli@gmail.com
**Code:** Question No: 12

## Description

This is a pre-emptive version of Longest Job First (LJF) scheduling algorithm. In this scheduling algorithm, we find the process with maximum remaining time and then process it. We check for the maximum remaining time after some interval of time to check if another process having more Burst Time arrived up to that time.

## Algorithm

- **Step-1**: Create a structure of process containing all necessary fields like Arrival Time, Burst Time, Completion Time, Turn Around Time, Waiting Time.
- **Step-2**: Sort according to the Arrival Time.
- **Step-3**: Find the process having Largest Burst Time and execute for each single unit. Increase the total time by 1 and reduce the Burst Time of that process with 1.
- **Step-4**: When any process has 0 Burst Time left, then update the Completion Time (Completion Time of that process will be Total Time at that time).
- **Step-5**: After calculating the Completion Time for each process, find Turn Around Time and Waiting Time.

## Procedure

There are 3 students (a, b, c) to be served food. Since they arrive at the same time, we need to use an algorithm which solves how the students are to be served food.

As given in the question, we must use LRTF (Longest Remaining Time Algorithm). Also given that if two students have same remaining time, then the student with the lowest ID number is served first.

As LRTF is used, Student C has the longest remaining time. So, the serving starts with C.

After C is served for 4 minutes, the remaining time of B and C are equal. B has the lowest ID number. So, the serve moves onto B.

After B is served for 2 minutes, the reaming time of A and B are equal. But still, B has the lowest ID number. So, B is served.

Since his “food taken time” is 4 minutes, he is done with the job.

Comparing the remaining students, A and C, C has a longer waiting time i.e., 4 min (he was already served 4 minutes, total time being 8 min). So, C is continued with the service.

After serving him for 2 minutes, the remaining time of C and A are equal. A has the lowest ID number. So, the serve moves onto A.

After serving A for 2 minutes, he is done with the job since his “food taken time” is 2 minutes.

The serve moves onto C again to process the remaining 2 min of his “food taken time”.

| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| c | c | c | c | b | b | b | b | c | c | a | a | c | c | |

Turn Around Time (TAT) = Completion Time (CT) - Arrival Time (AT)

Wait Time (WT) = Turn Around Time (TAT) - Burst Time (BT)

A completes the job in 12 min, arrival time is 0.

B completes the job in 8 min, arrival time is 0.

C completes the job in 14 min, arrival time is 0.

Turn Around Time (TAT) = Completion Time (CT) - Arrival Time (AT)

TAT(A) = 12 - 0 = 12 min

TAT(B) = 8 - 0 = 8 min

TAT(C) = 14 - 0 = 14 min

Given;

A’s Burst Time (BT) = 2 min

B’s Burst Time (BT) = 4 min

C’s Burst Time (BT) = 8 min

Here, Burst Time (BT) = Food Taken Time

Wait Time (WT) = Turn Around Time (TAT) - Burst Time (BT)

WT(A) = 12 - 2 = 10

WT(B) = 8 - 4 = 4

WT(C) = 14 - 8 = 6

Therefore,

The required Average Turn Around Time = (12+8+14)/3 = 11.33 minutes

The required Average Wait Time = (10+4+6)/3 = 6.67 minutes

## General Terms

- **Arrival Time**: Time at which the process arrives in the ready queue.
- **Completion Time**: Time at which process completes its execution.
- **Burst Time**: Time required by a process for CPU execution.
- **Turn Around Time**: Time Difference between completion time and arrival time.
- **Waiting Time**: Time Difference between turnaround time and burst time.

## Formulas

Turn Around Time (TAT) = (Completion Time) - (Arrival Time)

Waiting Time (WT) = (Turn Around Time) - (Burst Time)