

# CS559: Computer Systems Lab

**READ THE FOLLOWING INSTRUCTIONS CAREFULLY. FAILING TO ADHERE TO IT WILL LEAD TO DEDUCTION OF MARKS.**

1. Make sure to give the correct information in your submission link. You will receive a response email for every submission. **You must save it for future reference for the whole semester as you will be asked to show it.**
2. Every assignment will have multiple questions. **For every question, you will save your solution (i.e. program) as a .c file.** So if “Assignment 01” has 4 questions (marked as 1, 2, 3 and 4), then you will save your solution program as 1.c, 2.c, 3.c and 4.c.

**Note:** Your program will be compiled to check if your solution is correct and code is executed successfully and you will be marked accordingly. If any solution file is missing or not submitted, the marks for that solution will be **zero**.

3. Submit your files in a compressed folder (rar or zip) as per “submission instructions” given in your assignment question paper.

**Note:** You must make sure to upload in .zip or .rar. **Folders submitted with any other extension or wrong extension will not be evaluated and awarded zero.**

4. The student must ensure that the submitted file is not corrupted and can be unzipped properly. **Corrupted files that cannot be opened will be given zero. Similarly bad filenames having any other extension than “.c” will not be evaluated and given zero.**
5. You will be able to upload your assignment multiple times within the deadline span, but only your last submission will be treated as final and considered for evaluation. **No plea/request to consider intermediate submissions for evaluation will be entertained.** So, make sure to carefully check that you are submitting the correct assignment.
6. The submission link will be automatically deactivated after the deadline, and no request/plea for extension will be accepted. **Non-submission /wrong submission will be automatically awarded zero.**
7. Your code will be checked for similarity and you will be penalized according to the following rule:

**Similarity above 75%: 50% deduction**

**Similarity of 100%: full deduction**

8. **Compiler Information: Program must be compiled using online gdb compiler. If your program can not be compiled using gdb compiler then marks will be reduced.**  
**Compiler Link :** [https://www.onlinegdb.com/online\\_c\\_compiler](https://www.onlinegdb.com/online_c_compiler)
9. **Marks for every assignment (other than practice assignments) will be mailed to you with remarks. You will be allowed a deadline for rebuttal, after which no rebuttal will be entertained.**

## CS559: Computer Systems Lab

Date: November 03,2022

### Assignment\_9

Time : 3 hours(9.00-1200 Hrs.)

Submission deadline: November 03, 2022 - 1200 Hrs. IST

**Submission Instruction:** Store your assignments in folder and compress it as a rar/zip file (filename should be in this format: **roll-number\_Assignment9.rar or roll-number\_Assignment9.zip**). For example, if your roll number is 2211CS01, store your assignment as 2211CS01\_Assignment9.rar or 2211CS01\_Assignment9.zip. Also, save each program in the format given beside each question. Upload the same at the below link:  
<https://forms.gle/PTuoPoyC2jV3BN6S7>

**Problem 1:(Save as 1.c) :-** Write a program in C/C++ to implement Pre-emptive SJF Scheduling algorithm. The program output must show the following: [10 point]

1. Gantt Chart.
2. Arrival Time (AT), Burst Time (BT), Turnaround Time (TAT) and Waiting Time (WT) for each process in a tabular format.
3. Average Waiting Time.
4. Average Turnaround Time.

#### Sample Input:

Enter Number of test cases : 1

Enter Number Of Processes: 6

Enter arrival time and Burst time for Process P0: 0 7

Enter arrival time and Burst time for Process P1: 1 5

Enter arrival time and Burst time for Process P2: 2 3

Enter arrival time and Burst time for Process P3: 3 1

Enter arrival time and Burst time for Process P4: 4 2

Enter arrival time and Burst time for Process P5: 5 1

#### Expected Output:

Expected Output in accordance to Sample Input

Gantt Chart:

0 → [P1] ← 1 → [P2] ← 2 → [P3] ← 3 → [P4] ← 4 → [P3] ← 5 → [P3] ← 6 → [P6] ← 7 → [P5] ← 9 → [P2] ← 13 → [P1] → 19

P_No	AT	BT	TAT	WT
P2	0	7	19	12
P1	1	5	12	7
P0	2	3	4	1
P3	3	1	1	0
P4	4	2	5	3
P5	5	1	2	1

Average Turnaround Time: 7.10

Average Waiting Time: 4.00