PROJECT (Group work) **DUE:** 14th NOVEMBER

Instructions: FORMULATE GROUPS OF BETWEEN 5-8 MEMBERS
In your groups, You should create a program that accepts a list of processes (Process ID, Arrival Time, Burst Time, and optionally Priority/Time Quantum) and calculates the required metrics for the specified scheduling algorithm, outputting the results.:

To help you put into practice the concepts, of CPU Utilization

General Implementation Requirements

1. The program must handle multiple processes.
2. The output must display the calculated **Turn Around Time** (Completion Time – Arrival Time) and **Waiting Time** (Turn Around Time – Burst Time) for each process.

The program must calculate and display the **Average Waiting Time** and **Average Turn Around Time**.

Task 1: First Come First Serve (FCFS) Scheduling

FCFS is defined as a non-preemptive scheduling algorithm where the process that arrives first in the ready queue is assigned the CPU first. Its implementation uses a FIFO queue.

Programming Challenge: Implement the FCFS scheduling algorithm.

Consider four processes arriving for execution.

Process ID	Arrival Time (ms)	Burst Time (ms)
P1	0	8
P2	1	4
P3	2	9
P4	3	5

Your program should be able to automatically:-

i. Calculate the waiting time and Turn around Time
ii. The average waiting time for the above scheduling algorithm

TASK 2: Shortest Remaining Time First (SRTF) Scheduling

RTF is the preemptive mode of Shortest Job First (SJF). SJF aims to schedule the process with the shortest burst time first, which is the best approach to minimize waiting time. Preemptive scheduling

means a running process can be interrupted if a new process arrives with a shorter remaining burst time.

Programming Challenge: Implement the SRTF scheduling algorithm, which requires tracking the remaining burst time and allowing preemption.

Consider five processes arriving with the following data:

Process ID	Arrival Time (unit)	Burst Time (unit)
P1	3	1
P2	1	4
P3	4	2
P4	0	6
P5	2	3

Your program should be able to automatically:-

iii. Calculate the waiting time and Turn around Time
iv. The average waiting time for the above scheduling algorithm

TASK 3 : ROUND ROBIN SCHEDULING

Round Robin scheduling is always preemptive in nature, where the CPU is assigned to processes based on FCFS for a fixed time quantum (or time slice). After the quantum expires, the process is preempted and sent back to the ready queue.

Programming Challenge: Implement the Round Robin scheduling algorithm, correctly handling preemption and queuing based on the time quantum.

Consider five processes. The time quantum is 2 units.

Process ID	Arrival Time (unit)	Burst Time (unit)
P1	0	5
P2	1	3
P3	2	1
P4	3	2
P5	4	3

Your program should be able to automatically:-

v. Calculate the waiting time and Turn around Time
vi. The average waiting time for the above scheduling algorithm

Task 4 : Priority Scheduling (Preemptive)

In Priority Scheduling, the CPU is assigned to the process having the highest priority. If priorities are equal, FCFS breaks the tie. In preemptive mode, a higher priority process arriving mid-execution can interrupt a lower priority process.

Programming Challenge: Implement Preemptive Priority Scheduling.

Consider five processes (Note: Higher number represents higher priority).

Process ID	Arrival Time (unit)	Burst Time (unit)	Priority
P1	0	4	2
P2	1	3	3
P3	2	1	4
P4	3	5	5
P5	4	2	5

Your program should be able to automatically:-

vii. Calculate the waiting time and
viii. The average waiting time for the above scheduling algorithm