# LAB No: 2
# ROUND ROBIN AND PRIORITY BASED

**NAME:LAIBA NADEEM**

**ROLL NO:DT-22028**

**COURSE:OPERATING SYSTEM**

**COURSE CODE:CT-353**

**LAB#02**

**Objective: Write a C program to simulate the following CPU scheduling algorithms to find turnaround**

**time and waiting time for the above problem.**

**a) Round Robin b) Priority**

**EXERCISE:**

**a) OUTPUT:**

```
Round Robin Scheduling:
Process Burst Time  Waiting Time    Turnaround Time
P1  10      13      23
P2  5       10      15
P3  8       13      21

Average Waiting Time: 12.00
Average Turnaround Time: 19.67


=== Code Execution Successful ===
```

**b) OUTPUT:**

```
Priority Scheduling:
Process Burst Time  Priority    Waiting Time    Turnaround Time
P2  1       1       0       1
P5  5       2       1       6
P1  10      3       6       16
P3  2       4       16      18
P4  1       5       18      19

Average Waiting Time: 8.20
Average Turnaround Time: 12.00


=== Code Execution Successful ===
```

## QUESTION 5:

```
Round Robin Scheduling:
    PROCESS BURST TIME  WAITING TIME    TURNAROUND TIME
    P0      2       0       2
    P1      4       5       9
    P2      6       6       12

Average Waiting Time: 3.67
Average Turnaround Time: 7.67

Priority Scheduling:
    PROCESS PRIORITY    BURST TIME  WAITING TIME    TURNAROUND TIME
    P1      1       4       0       4
    P2      2       6       4       10
    P0      3       2       10      12

Average Waiting Time: 4.67
Average Turnaround Time: 8.67


=== Code Execution Successful ===
```

# OBSERVATION:

**Round Robin (RR) (Quantum = 3):**

- Processes are executed cyclically with equal CPU time slices.
- **Result:**
    - Balanced waiting times for all processes, ensuring fairness.
    - Higher average turnaround time compared to SJF due to context switching.

**Priority Scheduling:**

- Executes based on priority, with lower numerical values indicating higher priority.
- **Result:**
    - P1 (highest priority) completed first, minimizing its waiting time.
    - Starvation risk for lower-priority processes if longer.