Department Of Computer Science and Engineering

Course Title: Operating System Lab

Course Code: CSE 406

Title: CPU Scheduling Round Robin Algorithm

Submitted To
Atia Rahman Orthi
Lecturer
Department Of Computer Science & Engineering

Submitted By
Md. Atikul Islam Atik
Reg No:21201063
Roll:63
Sec:B1

Lab MID Exam

Submission Date: 15.4.2025

Problem Statement:

Input:

Process	Arrival	Burst
P1	0	7
P2	1	4
P3	2	15
P4	3	11
P5	4	20
P6	4	9

Output:

Process	AT	BT	CT	TAT	WT
P1	0	7	31	31	24
P2	1	4	9	8	4
P3	3	11	56	53	42
P4	3	11	56	53	42
P5	4	20	66	62	42
P6	4	9	50	46	37

Execution Steps (Gantt Chart Generation):

Time 0: P1 runs for 5 → Remaining = 2

Time 5: P2 runs for 4 → Finished

Time 9: P3 runs for 5 → Remaining = 10

Time 14: P4 runs for 5 → Remaining = 6

Time 19: P5 runs for 5 → Remaining = 15

Time 24: P6 runs for 5 → Remaining = 4

Time 29: P1 runs for 2 → Finished

Time 31: P3 runs for 5 → Remaining = 5

Time 36: P4 runs for 5 → Remaining = 1

Time 41: P5 runs for 5 → Remaining = 10

Time 46: P6 runs for 4 → Finished

Time 50: P3 runs for 5 → Finished

Time 55: P4 runs for 1 → Finished

Time 56: P5 runs for 5 → Remaining = 5

Time 61: P5 runs for 5 → Finished

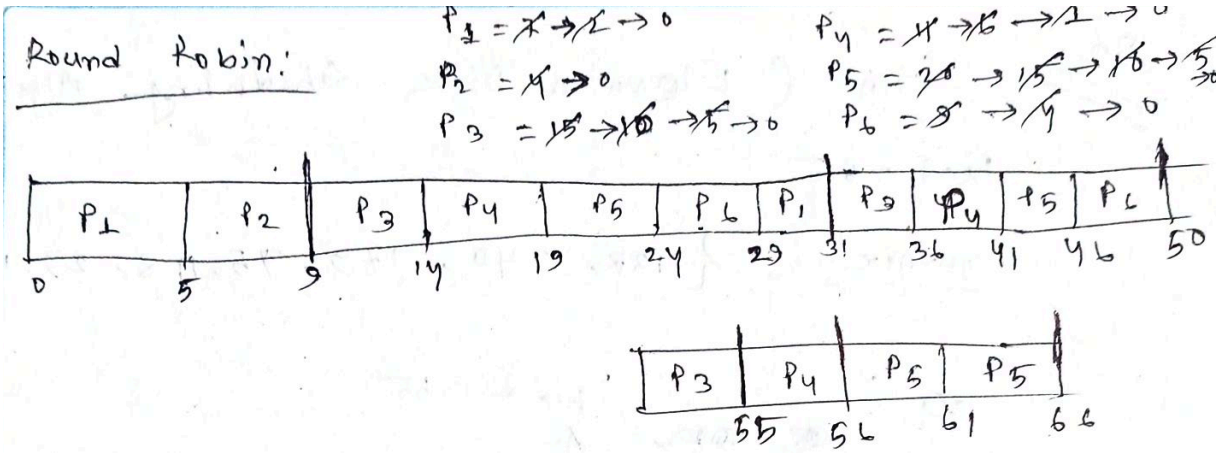
→ Final Time = 66

Gantt Chart (Execution Queue):

['p1', 'p2', 'p3', 'p4', 'p5', 'p6', 'p1', 'p3', 'p4', 'p5', 'p6', 'p3', 'p4', 'p5', 'p5']

Live Link of Code

Hand Calculation:



CT

$P_1 = 31$

$P_2 = 9$

$P_3 = 55$

$P_4 = 56$

$P_5 = 66$

$P_6 = 50$

Process	Arrival	Burst	CT	CT - Arrival (TAT)	TAT - Burst (WT)
P_1	0	7	31	31	24
P_2	1	4	9	8	4
P_3	2	15	55	53	38
P_4	3	11	56	53	42
P_5	4	20	66	62	42
P_6	4	9	50	46	37

Round Robin Algorithm:

- [ ] Sort all processes based on arrival time.
- [ ] Initialize time = 0, an empty queue, and tracking dictionaries.
- [ ] Add newly arrived processes to the queue.
- [ ] If the queue is empty, increment time (CPU remains idle).
- [ ] If the queue is not empty, dequeue the first process.
- [ ] Record the response time if it's the first execution of the process.
- [ ] Execute the process for at most time_quanta.
- [ ] Update time += execution_time, reduce the process's burst time.
- [ ] Add newly arrived processes to the queue during execution.
- [ ] If the process is not finished, re-add it to the queue; otherwise, calculate completion, turnaround, waiting, and response times.
- [ ] Repeat from step 3 until all processes are completed.

Conclusion:

The Round Robin (RR) scheduling algorithm was successfully implemented in this assignment. The program takes input for time quantum and the number of processes along with their arrival times and burst times . It then sorts the processes based on their arrival times and executes them in a time-sharing manner using a fixed time quantum.

The key metrics calculated include:

- **Completion Time (CT):** The time at which a process completes its execution.
- **Turnaround Time (TAT):** The total time taken from the arrival of the process to its completion.
- **Waiting Time (WT):** The time a process spends waiting in the queue before it starts executing.
- **Response Time (RT):** The time from process arrival to its first execution.