

ROUND ROBIN CPU SCHEDULING – INTERACTIVE SIMULATION AND VIDEO DEMONSTRATION

TEAM MEMBERS

| <u>NAME</u> | <u>REGISTRATION NUMBER</u> |
|--------------------|-----------------------------------|
| AALIF HASSAN | RA2411026010893 |
| JOSHIK R | RA2411026010885 |
| ADITYA RAJ | RA2411026010888 |

INTRODUCTION ABOUT THE CONCEPT

The Round Robin Scheduling Algorithm is one of the most widely used preemptive scheduling techniques, especially in time-sharing systems. It assigns each process a fixed time quantum, allowing every process to get an equal share of CPU time in a cyclic manner.

This project focuses on creating an interactive web-based toolkit and a demonstrative video that visually explain how the Round Robin method works and how process metrics like Completion Time (CT), Turnaround Time (TAT), and Waiting Time (WT) are calculated.

DESCRIPTION ABOUT THE CONCEPT

Algorithm Description

- Each process is placed in a ready queue.
- The CPU executes the first process for a fixed time quantum (q).
- If the process completes before the quantum expires, it is removed from the queue.

- If not, it is preempted and added to the end of the queue for the next cycle.
- This process repeats until all processes are completed.

Key Features of Our Toolkit

- User can enter the number of processes, arrival times, burst times, and time quantum.
- The toolkit dynamically computes:
 - Completion Time (CT)
 - Turnaround Time (TAT)
 - Waiting Time (WT)
- Displays results in a tabular form.
- Generates a Gantt Chart showing CPU execution order.

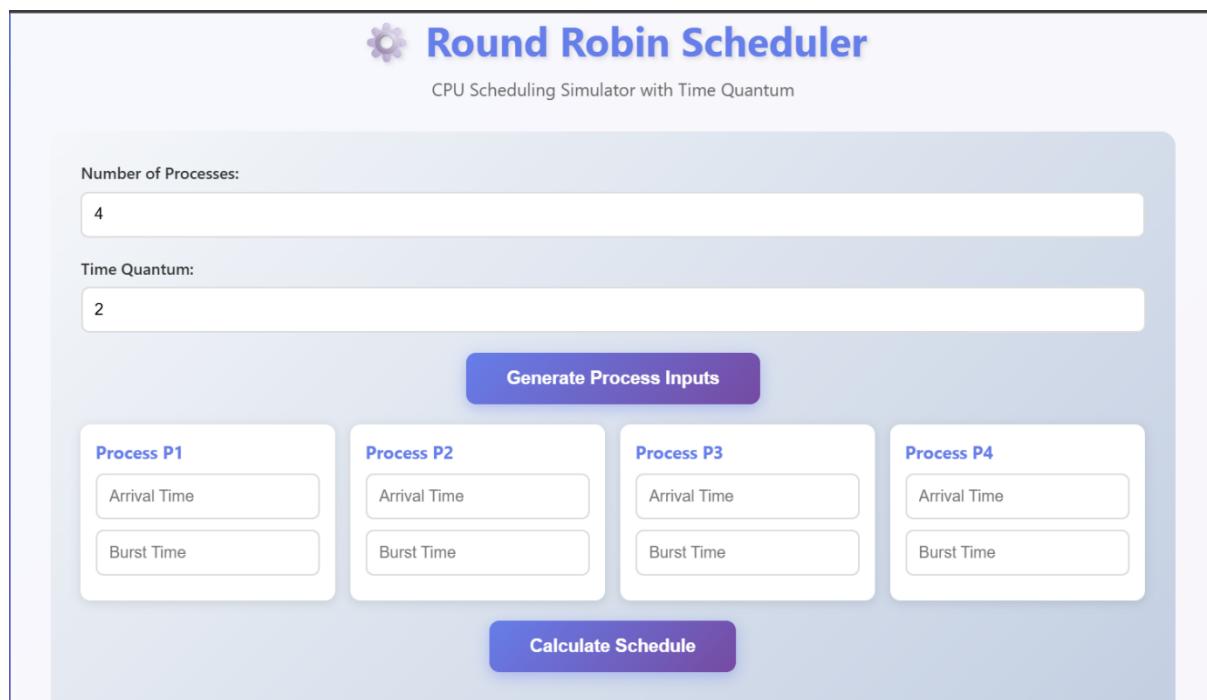
Technology Stack

Frontend → HTML

Logic Implementation → Integrated In HTML

Presentation → PowerPoint + Screen Recording for video demo

SCREENSHOT AND DRIVE LINK OF THE DEMO



The screenshot shows a web-based CPU Scheduling Simulator titled "Round Robin Scheduler". The title bar includes a gear icon and the text "Round Robin Scheduler" followed by "CPU Scheduling Simulator with Time Quantum".

The input section contains two fields: "Number of Processes" (set to 4) and "Time Quantum" (set to 2). Below these is a blue button labeled "Generate Process Inputs".

Four process input boxes are displayed below the button:

- Process P1:** Arrival Time and Burst Time input fields.
- Process P2:** Arrival Time and Burst Time input fields.
- Process P3:** Arrival Time and Burst Time input fields.
- Process P4:** Arrival Time and Burst Time input fields.

At the bottom center is a blue button labeled "Calculate Schedule".

Gantt Chart



Process Details

| Process | Arrival Time | Burst Time | Completion Time | Turnaround Time | Waiting Time |
|---------|--------------|------------|-----------------|-----------------|--------------|
| P1 | 0 | 2 | 2 | 2 | 0 |
| P2 | 2 | 4 | 10 | 8 | 4 |
| P3 | 3 | 6 | 16 | 13 | 7 |
| P4 | 4 | 8 | 20 | 16 | 8 |

Average Turnaround Time

9.75

Average Waiting Time

4.75

LOGICAL IMPLEMENTATION

```
while (1) {
    done = 1;
    for (i = 0; i < n; i++) {
        if (remaining[i] > 0) {
            done = 0; // at least one process is not finished
            if (remaining[i] > time_quantum) {
                t += time_quantum;
                remaining[i] -= time_quantum;
            } else {
                t += remaining[i];
                completion[i] = t;
                remaining[i] = 0;
            }
        }
    }
    if (done) break;
}

// Calculate Turnaround Time and Waiting Time
for (i = 0; i < n; i++) {
    tat[i] = completion[i] - arrival[i];
    wt[i] = tat[i] - burst[i];
}
```

GOOGLE DRIVE LINK FOR VIDEO PRESENTATION AND HTML FILE

https://drive.google.com/drive/folders/1teUlhUxAP7t_qnEu2YIGNeJ0LDDmPh0K?usp=drive_link

CONCLUSION

The Round Robin CPU Scheduling Project provided a clear understanding of how preemptive scheduling works in real systems.

By developing a video explanation and interactive simulation toolkit, we made the concept more visual, engaging, and easy to understand.

The project enhanced our skills in:

- Implementing scheduling logic using C and HTML
- Visualizing real-time process execution
- Presenting technical concepts effectively through video

Overall, this project bridges the gap between theoretical understanding and practical implementation of Round Robin scheduling algorithms.

REFERENCES

1. ChatGPT – For logical and Concept understanding
2. TutorialsPoint – *CPU Scheduling Algorithms*
https://www.tutorialspoint.com/operating_system/os_process_scheduling.htm
3. GeeksforGeeks – *Round Robin CPU Scheduling*
<https://www.geeksforgeeks.org/round-robin-scheduling/>