**VAISHNAVI E 231901059**

**Ex. No.: 6D**

**Date:22.03.2025**

**ROUND ROBIN SCHEDULING**

**Aim:**

To implement the Round Robin (RR) scheduling technique

**Algorithm:**

1. Declare the structure and its elements.
2. Get number of processes and Time quantum as input from the user.
3. Read the process name, arrival time and burst time
4. Create an array **rem\_bt[]** to keep track of remaining burst time of processes which is initially copy of bt[] (burst times array)
5. Create another array **wt[]** to store waiting times of processes. Initialize this array as 0. 6. Initialize time : t = 0

7. Keep traversing the all processes while all processes are not done. Do following for i'th process if it is not done yet.

1. If rem\_bt[i] > quantum (i) t = t + quantum (ii) bt\_rem[i] -= quantum;
2. Else // Last cycle for this process
3. t = t + bt\_rem[i];
4. wt[i] = t - bt[i]
5. bt\_rem[i] = 0; // This process is over
6. Calculate the waiting time and turnaround time for each process.
7. Calculate the average waiting time and average turnaround time.
8. Display the results.

**Program Code:** #include <stdio.h> int main() {

int n, quantum, i, t = 0, x, done; printf("Enter number of processes and time quantum: ");

scanf("%d %d", &n, &quantum);

int at[n], bt[n], rem\_bt[n], wt[n], tat[n];

x = n;

printf("Enter arrival time and burst time for each process:\n"); for (i = 0; i < n; i++) { scanf("%d %d", &at[i], &bt[i]);

rem\_bt[i] = bt[i]; wt[i] = 0;

}

printf("\nProcess\tAT\tBT\tWT\tTAT\n");

int total\_wt = 0, total\_tat = 0; for (t = 0, i = 0; x != 0;) { if (rem\_bt[i] > 0 && at[i] <= t) { if (rem\_bt[i] > quantum) { t += quantum; rem\_bt[i] -= quantum;

} else { t += rem\_bt[i]; wt[i] = t - at[i] - bt[i]; tat[i] = t - at[i]; total\_wt += wt[i]; total\_tat += tat[i]; rem\_bt[i] = 0; x--; printf("P%d\t%d\t%d\t%d\t%d\n", i + 1, at[i], bt[i], wt[i], tat[i]);

}

}

i = (i + 1) % n;

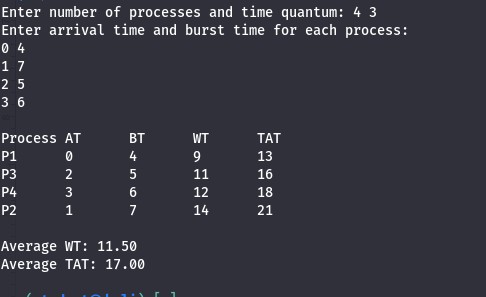
}

printf("\nAverage WT: %.2f\nAverage TAT: %.2f\n", (float)total\_wt / n, (float)total\_tat / n);

return 0;

}

**OUTPUT:**



**RESULT:**

Hence, RoundRobin CPU Scheduling has been executed successfully.