EXPERIMENT-3

Design a CPU scheduling program with C using First Come First Served technique with the following considerations.

a. All processes are activated at time 0.

b. Assume that no process waits on I/O devices

AIM:-

To design a CPU scheduling program using the **First Come First Served (FCFS)** technique in C with the following considerations:

- All processes are activated at time 0.
- No process waits on I/O devices.

.

ALGORITHM:-

1. Start the program.
2. Input the number of processes and their burst times.
3. Initialize the arrival time of all processes as 0 (as all are activated at time 0).
4. Calculate the completion time (CT) for each process:
    - CT = CT(previous process) + Burst Time
5. Calculate the turnaround time (TAT) for each process:
    - TAT = Completion Time - Arrival Time
6. Calculate the waiting time (WT) for each process:
    - WT = Turnaround Time - Burst Time
7. Display the results, including process IDs, burst time, completion time, turnaround time, and waiting time.
8. Calculate and display the average turnaround time and average waiting time.
9. End the program.

CODE:-

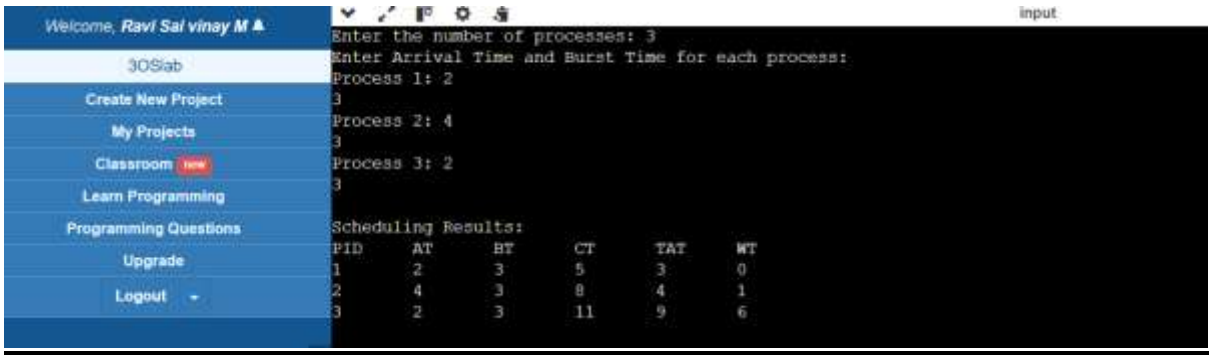
```
#include <stdio.h>

int main() {
    int n, i;
    int processes[100], burst_time[100], waiting_time[100], turnaround_time[100], completion_time[100];
    float avg_waiting_time = 0, avg_turnaround_time = 0;
    // Input number of processes
    printf("Enter the number of processes: ");
    scanf("%d", &n);
    // Input burst time for each process
    printf("Enter the burst time for each process:\n");
    for (i = 0; i < n; i++) {
        printf("Process %d: ", i + 1);
        scanf("%d", &burst_time[i]);
        processes[i] = i + 1; // Assign process IDs
    }
    // Calculate completion time
    completion_time[0] = burst_time[0];
    for (i = 1; i < n; i++) {
        completion_time[i] = completion_time[i - 1] + burst_time[i];
    }
    // Calculate turnaround time and waiting time
    for (i = 0; i < n; i++) {
```

```
        turnaround_time[i] = completion_time[i]; // Since arrival time = 0
        waiting_time[i] = turnaround_time[i] - burst_time[i];
        avg_turnaround_time += turnaround_time[i];
        avg_waiting_time += waiting_time[i];
    }
    // Calculate average times
    avg_turnaround_time /= n;
    avg_waiting_time /= n;
    // Display process details
    printf("\nProcess\tBurst Time\tCompletion Time\tTurnaround Time\tWaiting Time\n");
    for (i = 0; i < n; i++) {
        printf("%d\t%d\t\t%d\t\t%d\t\t%d\n", processes[i], burst_time[i], completion_time[i], turnaround_time[i], waiting_time[i]);
    }
    // Display average times
    printf("\nAverage Turnaround Time: %.2f\n", avg_turnaround_time);
    printf("Average Waiting Time: %.2f\n", avg_waiting_time);

    return 0;
}
```

OUTPUT:-



RESULT:-

The CPU scheduling program using the **First Come First Served (FCFS)** technique was successfully implemented in C, and the program calculated and displayed the completion time, turnaround time, and waiting time for all processes along with their averages..