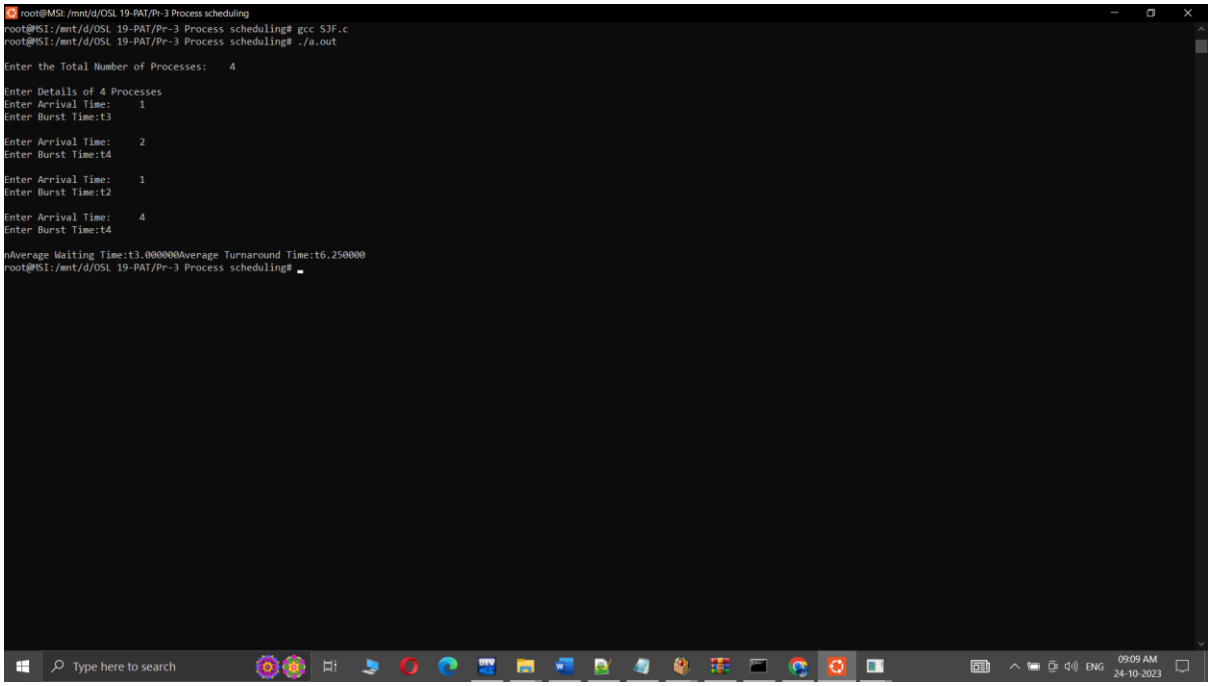
Shortest Job Frist

```
#include <stdio.h>
int main()
{

    int arrival_time[10], burst_time[10], temp[10];
    int i, smallest, count = 0, time, limit;
    double wait_time = 0, turnaround_time = 0, end;
    float average_waiting_time, average_turnaround_time;
    printf("\nEnter the Total Number of Processes:\t");
    scanf("%d", &limit);
    printf("\nEnter Details of %d Processes", limit);
    for(i = 0; i < limit; i++)
    {
            printf("\nEnter Arrival Time:\t");
            scanf("%d", &arrival_time[i]);
            printf("Enter Burst Time:t");
            scanf("%d", &burst_time[i]);
            temp[i] = burst_time[i];
    }
    burst_time[9] = 9999;
    for(time = 0; count != limit; time++)
    {
            smallest = 9;
            for(i = 0; i < limit; i++)
            {
                    if(arrival_time[i] <= time && burst_time[i] < burst_time[smallest] && burst_time[i]
> 0)
                    {
                            smallest = i;
                    }
```

```
            }
            burst_time[smallest]--;
            if(burst_time[smallest] == 0)
            {
                  count++;
                  end = time + 1;
                  wait_time = wait_time + end - arrival_time[smallest] - temp[smallest];
                  turnaround_time = turnaround_time + end - arrival_time[smallest];
            }
      }
      average_waiting_time = wait_time / limit;
      average_turnaround_time = turnaround_time / limit;
      printf("\nnAverage Waiting Time:t%lf", average_waiting_time);
      printf("Average Turnaround Time:t%lf\n", average_turnaround_time);
      return 0;
}
```

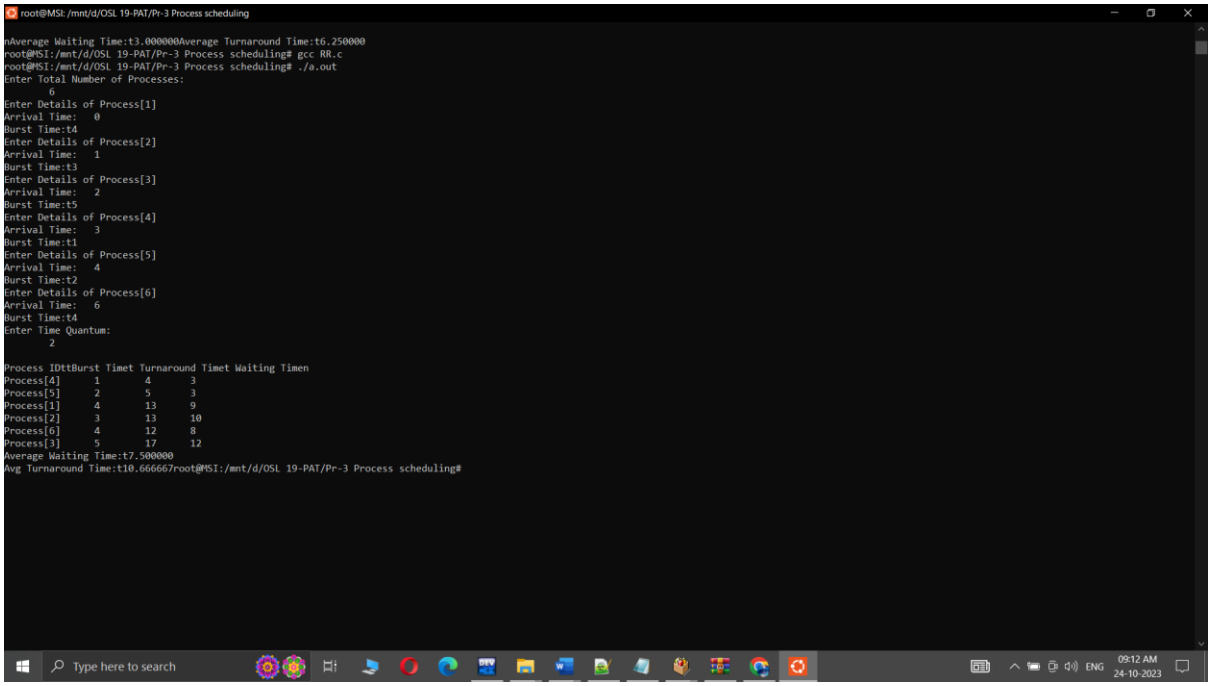

root@MSI: /mnt/d/OSL 19-PAT/Pr-3 Process scheduling
root@MSI:/mnt/d/OSL 19-PAT/Pr-3 Process scheduling# gcc SJF.c
root@MSI:/mnt/d/OSL 19-PAT/Pr-3 Process scheduling# ./a.out
Enter the Total Number of Processes: 4
Enter Details of 4 Processes
Enter Arrival Time: 1
Enter Burst Time:t3
Enter Arrival Time: 2
Enter Burst Time:t4
Enter Arrival Time: 1
Enter Burst Time:t2
Enter Arrival Time: 4
Enter Burst Time:t4
nAverage Waiting Time:t3.000000Average Turnaround Time:t6.250000
root@MSI:/mnt/d/OSL 19-PAT/Pr-3 Process scheduling#
Type here to search
ENG
09:09 AM
24-10-2023

Round Robin

```
#include<stdio.h>


int main()
{
    int i, limit, total = 0, x, counter = 0, time_quantum;
    int wait_time = 0, turnaround_time = 0, arrival_time[10], burst_time[10], temp[10];
    float average_wait_time, average_turnaround_time;
    printf("Enter Total Number of Processes:\n\t");
    scanf("%d", &limit);
    x = limit;
    for(i = 0; i < limit; i++)
    {
        printf("Enter Details of Process[%d]\n", i + 1);


        printf("Arrival Time:\t");


        scanf("%d", &arrival_time[i]);


        printf("Burst Time:t");


        scanf("%d", &burst_time[i]);


        temp[i] = burst_time[i];
    }


    printf("Enter Time Quantum:\n\t");
    scanf("%d", &time_quantum);
    printf("\nProcess IDttBurst Timet Turnaround Timet Waiting Timen");
    for(total = 0, i = 0; x != 0;)
    {
```

```
        if(temp[i] <= time_quantum && temp[i] > 0)
        {
            total = total + temp[i];
            temp[i] = 0;
            counter = 1;
        }
        else if(temp[i] > 0)
        {
            temp[i] = temp[i] - time_quantum;
            total = total + time_quantum;
        }
        if(temp[i] == 0 && counter == 1)
        {
            x--;
            printf("\nProcess[%d]\t%d\t %d\t %d", i + 1, burst_time[i], total - arrival_time[i], total -
arrival_time[i] - burst_time[i]);
            wait_time = wait_time + total - arrival_time[i] - burst_time[i];
            turnaround_time = turnaround_time + total - arrival_time[i];
            counter = 0;
        }
        if(i == limit - 1)
        {
            i = 0;
        }
        else if(arrival_time[i + 1] <= total)
        {
            i++;
        }
        else
        {
            i = 0;
```

```
        }
    }

    average_wait_time = wait_time * 1.0 / limit;
    average_turnaround_time = turnaround_time * 1.0 / limit;
    printf("\nAverage Waiting Time:t%f", average_wait_time);
    printf("\nAvg Turnaround Time:t%f", average_turnaround_time);
    return 0;
}
```


root@MSI: /mnt/d/OSL 19-PAT/Pr-3 Process scheduling
nAverage Waiting Time:t3.000000Average Turnaround Time:t6.250000
root@MSI:/mnt/d/OSL 19-PAT/Pr-3 Process scheduling# gcc RR.c
root@MSI:/mnt/d/OSL 19-PAT/Pr-3 Process scheduling# ./a.out
Enter Total Number of Processes:
6
Enter Details of Process[1]
Arrival Time: 0
Burst Time:t4
Enter Details of Process[2]
Arrival Time: 1
Burst Time:t3
Enter Details of Process[3]
Arrival Time: 2
Burst Time:t5
Enter Details of Process[4]
Arrival Time: 3
Burst Time:t1
Enter Details of Process[5]
Arrival Time: 4
Burst Time:t2
Enter Details of Process[6]
Arrival Time: 6
Burst Time:t4
Enter Time Quantum:
2
Process IDttBurst Timet Turnaround Timet Waiting Timen
Process[4] 1 4 3
Process[5] 2 5 3
Process[1] 4 13 9
Process[2] 3 13 10
Process[6] 4 12 8
Process[3] 5 17 12
Average Waiting Time:t7.500000
Avg Turnaround Time:t10.666667root@MSI:/mnt/d/OSL 19-PAT/Pr-3 Process scheduling#
Type here to search
ENG
09:12 AM
24-10-2023