LAB PROGRAM - 02

Write a C program to simulate the following CPU scheduling algorithm to find turnaround time and waiting time.

a) FCFS

```
#include<stdio.h>
void sort(int processes[],int n,int at[],int bt[])
{
    for(int i =0;i<n-1;i++)
    {
        for(int j=0;j<n-i-1;j++)
        {
            if(at[j]>at[j+1])
            {
                int temp=at[j];
                at[j]=at[j+1];
                at[j+1]=temp;


                temp=bt[j];
                bt[j]=bt[j+1];
                bt[j+1]=temp;

                temp = processes[j];
                processes[j]=processes[j+1];
                processes[j+1]=temp;

            }
        }
    }
}
void calculateTimes(int processes[], int n, int bt[], int at[], int wt[],
int tat[], int rt[]) {
    int service_time[n];
    service_time[0] = at[0];
    wt[0] = 0;
    rt[0] = 0;
    for (int i = 1; i < n; i++) {
```

```
        service_time[i] = service_time[i - 1] + bt[i - 1];
        wt[i] = service_time[i] - at[i];


        if (wt[i] < 0)
            wt[i] = 0;

        rt[i] = service_time[i] - at[i];
    }


    for (int i = 0; i < n; i++) {
        tat[i] = bt[i] + wt[i];
    }
}


void displayResults(int processes[], int n, int at[], int bt[], int wt[],
int tat[], int rt[]) {
    int total_wt = 0, total_tat = 0, total_rt = 0;

    printf("\nProcess\tArrival Time\tBurst Time\tWaiting Time\tTurnaround
Time\tResponse Time\n");
    for (int i = 0; i < n; i++) {
        total_wt += wt[i];
        total_tat += tat[i];
        total_rt += rt[i];
        printf("%d\t%d\t\t%d\t\t%d\t\t%d\t\t%d\n", processes[i], at[i],
bt[i], wt[i], tat[i], rt[i]);
    }

    printf("\nAverage Waiting Time = %.2f", (float)total_wt / n);
    printf("\nAverage Turnaround Time = %.2f", (float)total_tat / n);
    printf("\nAverage Response Time = %.2f\n", (float)total_rt / n);
}


int main() {
    int n;
```

```
    printf("Enter the number of processes: ");
    scanf("%d", &n);

    int processes[n], arrival_time[n], burst_time[n], waiting_time[n],
turnaround_time[n], response_time[n];

    printf("Enter Process IDs, Arrival Time, and Burst Time:\n");
    for (int i = 0; i < n; i++) {
        printf("Process %d: ", i + 1);
        scanf("%d %d %d", &processes[i], &arrival_time[i],
&burst_time[i]);
    }
     sort(processes,n,arrival_time,burst_time);

    calculateTimes(processes, n, burst_time, arrival_time, waiting_time,
turnaround_time, response_time);


    displayResults(processes, n, arrival_time, burst_time, waiting_time,
turnaround_time, response_time);

    return 0;
}
```

Output

```
dt-ba002h3m.p11   --stderr=Microsoft-MIEngine-Error-tv3034wf.u5g   --pid=Microsoft-MIEngin
Enter the number of processes: 3
Enter Process IDs, Arrival Time, and Burst Time:
Process 1: 1 2 5
Process 2: 3 5 1
Process 3: 2 3 6

Process Arrival Time    Burst Time      Waiting Time    Turnaround Time Response Time
1       2               5               0               5               0
2       3               6               4               10              4
3       5               1               8               9               8

Average Waiting Time = 4.00
Average Turnaround Time = 8.00
Average Response Time = 4.00
PS C:\Users\Admin>
```