LAB-02

Exercise:

1) Implement the First Come First Serve code and paste the output below.

CODE:

```c
#include <stdio.h>
int main()
{
    int at[20], bt[20], wt[20], tat[20], ct[20], i, n;
    float wtavg = 0, tatavg = 0;
    printf("Enter the number of processes: ");
    scanf("%d", &n);
    for (i = 0; i < n; i++)
    {
        printf("Enter Arrival Time and Burst Time for Process P%d: ", i + 1);
        scanf("%d%d", &at[i], &bt[i]);
    }
    ct[0] = at[0] + bt[0];
    tat[0] = ct[0] - at[0];
    wt[0] = tat[0] - bt[0];

    for (i = 1; i < n; i++)
    {
        if (at[i] > ct[i - 1])
        {
            ct[i] = at[i] + bt[i];
        }
        else
        {
            ct[i] = ct[i - 1] + bt[i];
        }
        tat[i] = ct[i] - at[i];
        wt[i] = tat[i] - bt[i];
    }
    for (i = 0; i < n; i++)
    {
        wtavg += wt[i];
        tatavg += tat[i];
    }
    wtavg /= n;
    tatavg /= n;
    printf("\nProcess\tArrival Time\tBurst Time\tCompletion Time\tTurnaround Time\tWaiting Time\n");
    for (i = 0; i < n; i++)
    {
        printf("P%d\t\t%d\t\t%d\t\t%d\t\t%d\t\t%d\n", i + 1, at[i], bt[i], ct[i], tat[i], wt[i]);
    }
    printf("\nAverage Turnaround Time: %.2f\n", tatavg);
    printf("Average Waiting Time: %.2f\n", wtavg);

    return 0;
}
```

OUTPUT:

```
PS D:\OS labs> cd "d:\OS labs\" ; if ($?) { gcc Lab_2.c -o Lab_2 } ; if ($?) { .\Lab_2 }
Enter the number of processes: 4
Enter Arrival Time and Burst Time for Process P1: 3 3
Enter Arrival Time and Burst Time for Process P2: 0 4
Enter Arrival Time and Burst Time for Process P3: 2 2
Enter Arrival Time and Burst Time for Process P4: 1 4

Process Arrival Time    Burst Time      Completion Time Turnaround Time Waiting Time
P1              3               3               6               3               0
P2              0               4               10              10              6
P3              2               2               12              10              8
P4              1               4               16              15              11

Average Turnaround Time: 9.50
Average Waiting Time: 6.25
```

2) Implement the Shortest Job First code and paste the output below.

CODE:

```c
#include <stdio.h>
int main()
{
    int p[20], at[20], bt[20], wt[20], tat[20], ct[20], i, k, n, temp;
    float wtavg = 0, tatavg = 0;
    printf("Enter the number of processes: ");
    scanf("%d", &n);
    for (i = 0; i < n; i++)
    {
        p[i] = i + 1;
        printf("Enter Arrival Time for Process P%d: ", i + 1);
        scanf("%d", &at[i]);
        printf("Enter Burst Time for Process P%d: ", i + 1);
        scanf("%d", &bt[i]);
    }
    for (i = 0; i < n; i++)
    {
        for (k = i + 1; k < n; k++)
        {
            if (at[i] > at[k] || (at[i] == at[k] && bt[i] > bt[k]))
            {
                temp = at[i];
                at[i] = at[k];
                at[k] = temp;

                temp = bt[i];
                bt[i] = bt[k];
                bt[k] = temp;

                temp = p[i];
                p[i] = p[k];
                p[k] = temp;
            }
        }
    }
    ct[0] = at[0] + bt[0];
    tat[0] = ct[0] - at[0];
```

```
    wt[0] = tat[0] - bt[0];
    wtavg = wt[0];
    tatavg = tat[0];
    for (i = 1; i < n; i++)
    {
        if (ct[i - 1] < at[i])
        {
            ct[i] = at[i] + bt[i];
        }
        else
        {
            ct[i] = ct[i - 1] + bt[i];
        }
        tat[i] = ct[i] - at[i];
        wt[i] = tat[i] - bt[i];
        wtavg += wt[i];
        tatavg += tat[i];
    }
    wtavg /= n;
    tatavg /= n;
    printf("\nProcess\tArrival Time\tBurst Time\tCompletion Time\tTurnaround Time\tWaiting
Time\n");
    for (i = 0; i < n; i++)
    {
        printf("P%d\t\t%d\t\t%d\t\t%d\t\t%d\t\t%d\n", p[i], at[i], bt[i], ct[i], tat[i],
wt[i]);
    }
    printf("\nAverage Turnaround Time: %.2f\n", tatavg);
    printf("Average Waiting Time: %.2f\n", wtavg);
    return 0;
}
```

OUTPUT:

```
Enter the number of processes: 4
Enter Arrival Time for Process P1: 3
Enter Burst Time for Process P1: 3
Enter Arrival Time for Process P2: 0
Enter Burst Time for Process P2: 4
Enter Arrival Time for Process P3: 2
Enter Burst Time for Process P3: 2
Enter Arrival Time for Process P4: 1
Enter Burst Time for Process P4: 4

Process Arrival Time    Burst Time      Completion Time Turnaround Time Waiting Time
P2              0               4               4               4               0
P4              1               4               8               7               3
P3              2               2               10              8               6
P1              3               3               13              10              7

Average Turnaround Time: 7.25
Average Waiting Time: 4.00
```