Name: Aryan Nagar
Roll No.: 20051479

Lab Assignment 10

Priority Scheduling (Non-Preemptive)

```c
#include <stdio.h>

int main()
{
    int bt[20], p[20], wt[20], tat[20], pr[20], i, j, n,
total = 0, pos, temp, avg_wt, avg_tat;
    printf("Enter Total Number of Process:");
    scanf("%d", &n);

    printf("\nEnter Burst Time and Priority\n");
    for (i = 0; i < n; i++)
    {
        printf("\nP[%d]\n", i + 1);
        printf("Burst Time:");
        scanf("%d", &bt[i]);
        printf("Priority:");
        scanf("%d", &pr[i]);
        p[i] = i + 1; // contains process number
    }

    // sorting burst time, priority and process number in
ascending order using selection sort
    for (i = 0; i < n; i++)
    {
        pos = i;
        for (j = i + 1; j < n; j++)
        {
            if (pr[j] < pr[pos])
                pos = j;
        }

        temp = pr[i];
        pr[i] = pr[pos];
        pr[pos] = temp;

        temp = bt[i];
```

Name: Aryan Nagar
Roll No.: 20051479

```
            bt[i] = bt[pos];
            bt[pos] = temp;

            temp = p[i];
            p[i] = p[pos];
            p[pos] = temp;
        }

        wt[0] = 0; // waiting time for first process is zero

        // calculate waiting time
        for (i = 1; i < n; i++)
        {
            wt[i] = 0;
            for (j = 0; j < i; j++)
                wt[i] += bt[j];

            total += wt[i];
        }

        avg_wt = total / n; // average waiting time
        total = 0;

        printf("\nProcess\t    Burst Time    \tWaiting
Time\tTurnaround Time");
        for (i = 0; i < n; i++)
        {
            tat[i] = bt[i] + wt[i]; // calculate turnaround
time
            total += tat[i];
            printf("\nP[%d]\t\t  %d\t\t    %d\t\t\t%d", p[i],
bt[i], wt[i], tat[i]);
        }

        avg_tat = total / n; // average turnaround time
        printf("\n\nAverage Waiting Time=%d", avg_wt);
        printf("\nAverage Turnaround Time=%d\n", avg_tat);

        return 0;
}
```

Name: Aryan Nagar
Roll No.: 20051479

Output

```
Enter Total Number of Process:5

Enter Burst Time and Priority

P[1]
Burst Time:4
Priority:1

P[2]
Burst Time:4
Priority:1

P[3]
Burst Time:3
Priority:2

P[4]
Burst Time:7
Priority:1

P[5]
Burst Time:4
Priority:3

Process     Burst Time          Waiting Time    Turnaround Time
P[1]              4                   0                4
P[2]              4                   4                8
P[4]              7                   8                15
P[3]              3                   15               18
P[5]              4                   18               22

Average Waiting Time=9
Average Turnaround Time=13
```

Name: Aryan Nagar
Roll No.: 20051479

Priority Scheduling (Preemptive)

```c
#include <stdio.h>
struct process
{
    int WT, AT, BT, TAT, PT;
};

struct process a[10];

int main()
{
    int n, temp[10], t, count = 0, short_p;
    float total_WT = 0, total_TAT = 0, Avg_WT, Avg_TAT;
    printf("Enter the number of the process\n");
    scanf("%d", &n);
    printf("Enter the arrival time , burst time and
priority of the process\n");
    printf("AT BT PT\n");
    for (int i = 0; i < n; i++)
    {
        scanf("%d%d%d", &a[i].AT, &a[i].BT, &a[i].PT);

        temp[i] = a[i].BT;
    }

    a[9].PT = 10000;

    for (t = 0; count != n; t++)
    {
        short_p = 9;
        for (int i = 0; i < n; i++)
        {
            if (a[short_p].PT > a[i].PT && a[i].AT <= t
&& a[i].BT > 0)
            {
                short_p = i;
            }
        }
```

Name: Aryan Nagar
Roll No.: 20051479

```
        a[short_p].BT = a[short_p].BT - 1;

        if (a[short_p].BT == 0)
        {

            count++;
            a[short_p].WT = t + 1 - a[short_p].AT -
temp[short_p];
            a[short_p].TAT = t + 1 - a[short_p].AT;

            total_WT = total_WT + a[short_p].WT;
            total_TAT = total_TAT + a[short_p].TAT;
        }
    }

    Avg_WT = total_WT / n;
    Avg_TAT = total_TAT / n;

    printf("ID\tAT\tWT\tTAT\tPR \n");
    for (int i = 0; i < n; i++)
    {
        printf("%d\t%d\t%d\t%d\t%d\n", i + 1, a[i].AT,
a[i].WT, a[i].TAT, a[i].PT);
    }

    printf("Avg waiting time of the process  is %f\n",
Avg_WT);
    printf("Avg turn around time of the process is %f\n",
Avg_TAT);

    return 0;
}
```

Name: Aryan Nagar
Roll No.: 20051479

Output

```
Enter the number of the process
5
AT BT PT
0 4 1
0 3 2
6 7 1
11 4 3
12 2 2
ID      AT      WT      TAT     PR
1       0       0       4       1
2       0       11      14      2
3       6       0       7       1
4       11      5       9       3
5       12      2       4       2
Avg waiting time of the process  is 3.600000
Avg turn around time of the process is 7.600000
PS E:\Mega Sync\Programming\C\Scheduling Algorithms>
PS E:\Mega Sync\Programming\C\Scheduling Algorithms>
```