EXP 6: Construct a C program to implement pre-emptive priority scheduling algorithm.

```
#include <stdio.h>

int main() {
    int n, i, time = 0, completed = 0;
    int at[20], bt[20], pr[20], rt[20]; // Arrival, Burst, Priority, Remaining Time
    int wt[20], tat[20];
    float total_wt = 0, total_tat = 0;

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

    // Input arrival time, burst time, priority
    for (i = 0; i < n; i++) {
        printf("Enter arrival time, burst time and priority for process %d: ", i + 1);
        scanf("%d%d%d", &at[i], &bt[i], &pr[i]);
        rt[i] = bt[i]; // Set remaining time = burst time
    }

    int smallest, minPriority;
    int finish_time[20];
    int is_completed[20] = {0};

    while (completed != n) {
        smallest = -1;
        minPriority = 9999;

        // Find process with highest priority (lowest number)
        for (i = 0; i < n; i++) {
```

```
        if (at[i] <= time && rt[i] > 0 && pr[i] < minPriority) {
            minPriority = pr[i];
            smallest = i;
        }
    }

    if (smallest == -1) {
        time++;
        continue;
    }

    rt[smallest]--; // Run process for 1 unit time
    time++;

    // If process is finished
    if (rt[smallest] == 0) {
        completed++;
        finish_time[smallest] = time;
        tat[smallest] = finish_time[smallest] - at[smallest];
        wt[smallest] = tat[smallest] - bt[smallest];

        total_wt += wt[smallest];
        total_tat += tat[smallest];
    }
}

// Display results
printf("\nProcess\tAT\tBT\tPR\tWT\tTAT\n");
for (i = 0; i < n; i++) {
    printf("P%d\t%d\t%d\t%d\t%d\t%d\n", i + 1, at[i], bt[i], pr[i], wt[i], tat[i]);
}
```

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

    return 0;
}
```

Sample Input

Enter number of processes: 3

Enter arrival time, burst time and priority for process 1: 0 5 1

Enter arrival time, burst time and priority for process 2: 2 3 4

Enter arrival time, burst time and priority for process 3: 2 9 0

Sample Output

```
Process AT      BT      PR      WT      TAT
P1      0       5       1       9       14
P2      2       3       4       12      15
P3      2       9       0       0       9

Average Waiting Time = 7.00
Average Turnaround Time = 12.67
```