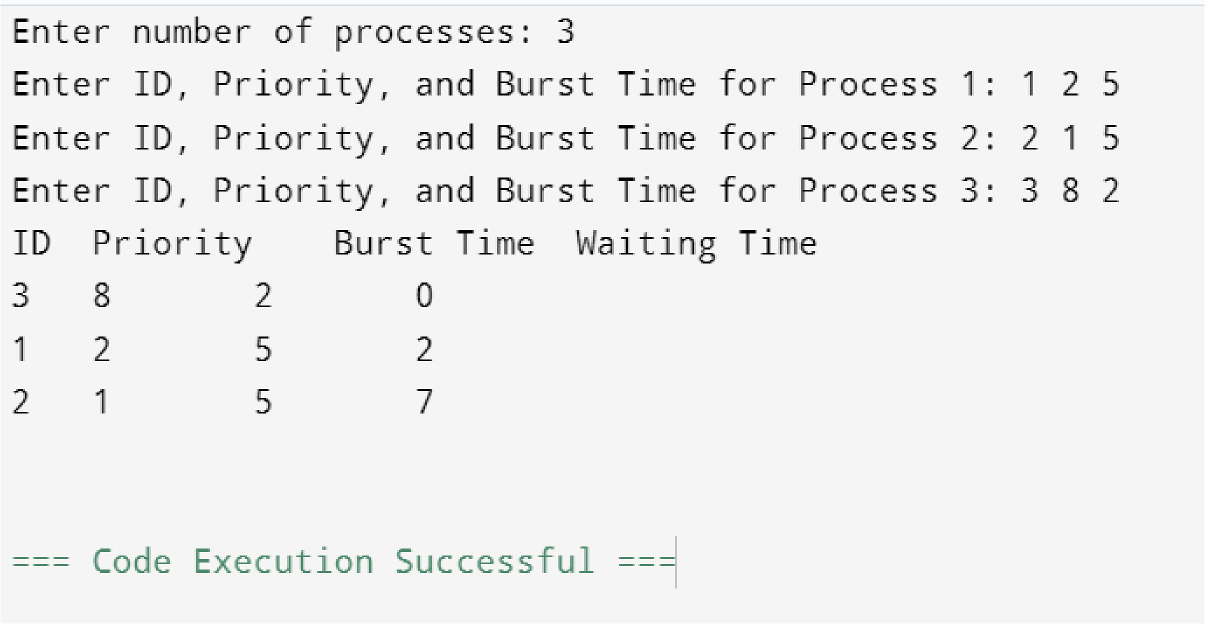
EXP-5 :-

PROGRAM :-

```
#include <stdio.h>
typedef struct {
    int id, priority, burst_time, waiting_time;
} Process;
void scheduleProcesses(Process processes[], int n) {
    for (int i = 0; i < n - 1; i++)
        for (int j = i + 1; j < n; j++)
            if (processes[i].priority < processes[j].priority) {
                Process temp = processes[i];
                processes[i] = processes[j];
                processes[j] = temp;
            }
}
void calculateAndPrintSchedule(Process processes[], int n) {
    int waiting_time = 0;
    printf("ID\tPriority\tBurst Time\tWaiting Time\n");
    for (int i = 0; i < n; i++) {
        processes[i].waiting_time = waiting_time;
        printf("%d\t%d\t\t%d\t\t%d\n", processes[i].id, processes[i].priority, processes[i].burst_time, waiting_time);
        waiting_time += processes[i].burst_time;
    }
}
int main() {
    Process processes[10];
    int n;
    printf("Enter number of processes: ");
    scanf("%d", &n);
    for (int i = 0; i < n; i++) {
        printf("Enter ID, Priority, and Burst Time for Process %d: ", i + 1);

        scanf("%d %d %d", &processes[i].id, &processes[i].priority, &processes[i].burst_time);

    }

    scheduleProcesses(processes, n);

    calculateAndPrintSchedule(processes, n);

    return 0;

}
```

OUTPUT :-



```
Enter number of processes: 3
Enter ID, Priority, and Burst Time for Process 1: 1 2 5
Enter ID, Priority, and Burst Time for Process 2: 2 1 5
Enter ID, Priority, and Burst Time for Process 3: 3 8 2
ID  Priority    Burst Time  Waiting Time
3   8       2       0
1   2       5       2
2   1       5       7



=== Code Execution Successful ===
```