EXP-6 :-

PROGRAM :-

```
#include <stdio.h>
struct Process {
   int id;
   int burst_time;
   int remaining_time;
};
void roundRobin(struct Process processes[], int n, int quantum) {
   int time = 0;
   int completed = 0;
   while (completed < n) {
      for (int i = 0; i < n; i++) {
         if (processes[i].remaining_time > 0) {
            if (processes[i].remaining_time > quantum) {
               time += quantum;
               processes[i].remaining_time -= quantum;
            } else {
               time += processes[i].remaining_time;
               processes[i].remaining_time = 0;
               completed++;
               printf("Process %d completed at time %d\n", processes[i].id, time);
            }
         }
      }
   }
}
int main() {
   int n, quantum;
   printf("Enter number of processes: ");
   scanf("%d", &n);
```

```
    struct Process processes[n];
    for (int i = 0; i < n; i++) {
        processes[i].id = i + 1;
        printf("Enter burst time for process %d: ", processes[i].id);
        scanf("%d", &processes[i].burst_time);
        processes[i].remaining_time = processes[i].burst_time;
    }
    printf("Enter time quantum: ");
    scanf("%d", &quantum);
    roundRobin(processes, n, quantum);
    return 0;
}
```

OUTPUT :-

```
Enter number of processes: 4
Enter burst time for process 1: 10
Enter burst time for process 2: 4
Enter burst time for process 3: 25
Enter burst time for process 4: 12
Enter time quantum: 5
Process 2 completed at time 9
Process 1 completed at time 24
Process 4 completed at time 41
Process 3 completed at time 51


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