Write a C program to simulate the following CPU scheduling algorithm to find turnaround time and waiting time.

d) Round Robin

```
#include <stdio.h>

#define MAX 100

void roundRobin(int n, int at[], int bt[], int quant) {
    int ct[n], tat[n], wt[n], rem_bt[n];
    int queue[MAX], front = 0, rear = 0;
    int time = 0, completed = 0, visited[n];

    for (int i = 0; i < n; i++) {
        rem_bt[i] = bt[i];
        visited[i] = 0;
    }

    queue[rear++] = 0;
    visited[0] = 1;

    while (completed < n) {
        int index = queue[front++];

        if (rem_bt[index] > quant) {
            time += quant;
            rem_bt[index] -= quant;
        } else {
            time += rem_bt[index];
            rem_bt[index] = 0;
            ct[index] = time;
            completed++;
        }

        for (int i = 0; i < n; i++) {
            if (at[i] <= time && rem_bt[i] > 0 && !visited[i]) {
                queue[rear++] = i;
                visited[i] = 1;
            }
        }

        if (rem_bt[index] > 0) {
            queue[rear++] = index;
        }
```

```
        if (front == rear) {
            for (int i = 0; i < n; i++) {
                if (rem_bt[i] > 0) {
                    queue[rear++] = i;
                    visited[i] = 1;
                    break;
                }
            }
        }
    }

    float total_tat = 0, total_wt = 0;
    printf("P#\tAT\tBT\tCT\tTAT\tWT\n");
    for (int i = 0; i < n; i++) {
        tat[i] = ct[i] - at[i];
        wt[i] = tat[i] - bt[i];
        total_tat += tat[i];
        total_wt += wt[i];
        printf("%d\t%d\t%d\t%d\t%d\t%d\n", i + 1, at[i], bt[i], ct[i], tat[i], wt[i]);
    }

    printf("Average TAT: %.2f\n", total_tat / n);
    printf("Average WT: %.2f\n", total_wt / n);
}

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

    int at[n], bt[n];
    for (int i = 0; i < n; i++) {
        printf("Enter AT and BT for process %d: ", i + 1);
        scanf("%d %d", &at[i], &bt[i]);
    }

    printf("Enter time quantum: ");
    scanf("%d", &quant);

    roundRobin(n, at, bt, quant);
    return 0;
}
```

Output :

```
Enter number of processes: 5
Enter AT and BT for process 1: 0 5
Enter AT and BT for process 2: 1 3
Enter AT and BT for process 3: 2 1
Enter AT and BT for process 4: 3 2
Enter AT and BT for process 5: 4 3
Enter time quantum: 2
P#      AT      BT      CT      TAT     WT
1       0       5       13      13      8
2       1       3       12      11      8
3       2       1       5       3       2
4       3       2       9       6       4
5       4       3       14      10      7
Average TAT: 8.60
Average WT: 5.80
```