RR :

```
#include <stdio.h>

struct process {
    int at, st, ft, status;
} ready[10];

int n, t, completed = 0;

int Dispatch(int cur_time) {
    for (int i = 0; i < n; i++) {
        if (ready[i].status != 2 && ready[i].at <= cur_time) {
            return i;
        }
    }
    return -1;
}

int main() {
    printf("Enter number of processes and time slice: ");
    scanf("%d %d", &n, &t);

    int original_st[10];
    for (int i = 0; i < n; i++) {
        printf("Enter Arrival and Service Time for Process %d: ", i + 1);
        scanf("%d %d", &ready[i].at, &ready[i].st);
        original_st[i] = ready[i].st;
        ready[i].status = 0;
    }

    int cur_time = 0;
    while (completed < n) {
        int pid = Dispatch(cur_time);
        if (pid == -1) {
            cur_time++;
            continue;
        }

        if (ready[pid].st <= t) {
            cur_time += ready[pid].st;
            ready[pid].ft = cur_time;
            ready[pid].status = 2;
            completed++;
        } else {
            cur_time += t;
            ready[pid].st -= t;
        }
    }
```

```
    printf("Process\tAT\tST\tFT\tTT\tWT\n");
    for (int i = 0; i < n; i++) {
        int tt = ready[i].ft - ready[i].at;
        int wt = tt - original_st[i];
        printf("%d\t%d\t%d\t%d\t%d\t%d\n", i + 1, ready[i].at, original_st[i], ready[i].ft, tt, wt);
    }

    return 0;
}
```