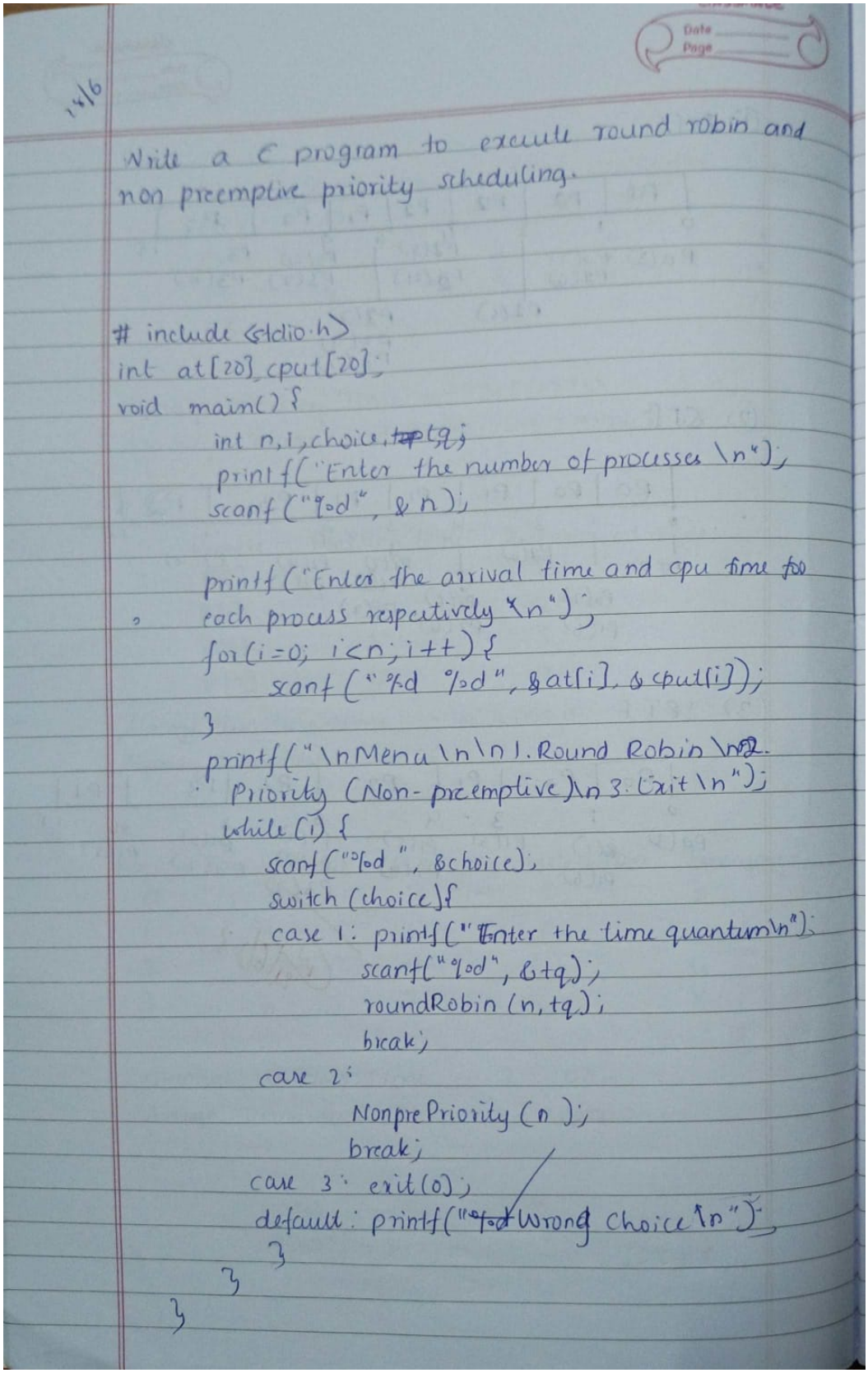
Q)Write a C program to execute round robin and non preemptive priority scheduling.



14/6

Write a C program to execute round robin and non preemptive priority scheduling.

```
# include <stdio.h>
int at[20], cput[20];
void main(){
    int n,i,choice,tq;
    printf("Enter the number of processes \n");
    scanf("%d", &n);

    printf("Enter the arrival time and cpu time for
    each process respatively \n");
    for(i=0; i<n; i++){
        scanf("%d %d", &at[i], &cput[i]);
    }
    printf("\nMenu\n\n1. Round Robin\n2.
    Priority (Non-preemptive)\n3. Exit\n");
    while(1){
        scanf("%d", &choice);
        switch(choice){
        case 1: printf("Enter the time quantum\n");
                scanf("%d", &tq);
                roundRobin(n, tq);
                break;
        case 2:
                NonprePriority(n);
                break;
        case 3: exit(0);
        default: printf("Wrong choice\n");
        }
    }
}
```

```
void roundRobin(int n, int tq){
    int i, remaining_time[20], wt[20], tat[20],
    completed=0, time=0;
    float awt=0, atat=0;

    for(i=0; i<n; i++){
        remaining_time[i]= cput[i];
    }

    while(completed <n){
        for(int i=0; i<n; i++){
            if(remaining_time[i]>0 && at[i]<=time){
            if(remaining_time[i]<=tq){
                time += remaining_time[i];
                remaining_time[i]=0;
                completed ++;
                tat[i] = time - at[i];
                wt[i] = tat[i] - cput[i];
            }
            else{
                time += tq;
                remaining_time[i]= tq;
            }
        }
    }
    for(i=0; i<n; i++){
        atat += tat[i];
        awt += wt[i];
    }
    atat /= n;
    awt /= n;
```

```
printf("\nProcess \t cpu time \t Arrivaltime \t
turn around time \t waiting time \n");
for(int i=0; i<n, i++){
    printf("%d \t %d \t \t %d \t \t %d
    \t \t %d \n", i, cput[i], at[i],
    tat[i], wt[i]);
}
printf("Average Turnaround time: %.2f\n", atat);
printf("Average Waiting time: %.2f\n", awt);
}

void Nonpre Priority(int n){
    int priority[20], wt[20], tat[20], hp=0,
    cmpt[20], cput1[20], sum=0, i, sum_burst_time=0;
    float awt=0, atat=0;

    printf("Enter the priorities of processes\n");
    for(int i=0; i<n; i++){
        printf("Process %d: ", i);
        scanf("%d", &priority[i]);
    }
    for(i=0; i<n; i++){
        cput1[i] = cput[i];
        sum_burst_time += cput[i];
    }
    printf("\tProcess \t turnaround time \t Waiting
        time \n");
    cput1[9] = -1;
    while(sum < sum_burst_time){
        hp=9;
        for(i=0; i<; i++){
        if(at[i]<=sum && cput1[i]>0 &&
        priority[i] > priority[hp])
            hp=i;
    }
```

```
printf("\t P[%d] \t\t %d \t\t\t %d \n", hp,
    sum + cputl[hp] - at[hp], sum - at[hp]);
awt += sum + cputl[hp] - at[hp];
atat += sum - at[hp];
sum += cputl[hp];
cput[hp] = 0;
}
awt = awt / n;
atat = atat / n;
printf("\n Average Waiting time --%f ", awt);
printf("\n Average Turnaround time --%f \n",
                                    atat);
}
```

Output

```
Enter the number of processes
5
Enter the arrival time & CPU time for each process
respectively
0  5
1  3
2  1
3  2
4  2
Menu
1. Round Robin
2. Priority (Non Preemptive)
3. Exit
1
Enter the time quantum
2
```

Process	cpu time	Arrival Time	Turnaround time	waiting time
0	5	0	14	9
1	3	1	11	8
2	1	2	3	2
3	2	3	4	2
4	3	4	9	6

Average turn around Time : 8.20
Average waiting Time : 5.00

2
Enter the priorities of processes
Process 0 : 3
process 1 : 2
Process 2 : 4
Process 3 : 4
Process 4 : 5

Process	turn around time	waiting time
P[0]	5	0
P[3]	4	2
P[4]	6	3
P[1]	12	9
P[2]	12	11

Average waiting Time : 5.00
Average Turn Around time : 7.80

Gantt chart
Round Robin

P1	P2	P3	P1	P4	P5	P2	P1	P5
0	2	4	5	7	9	11	12	13	14

P2(3)	P3(1)	P1(2)	P4(2)	P5(3)	P2(1)	P1(1)	P5(1)
P3(1)	P1(5)	P4(2)	P5(3)	P2(1)	P1(1)	P5(1)	
P1(3)	P4(2)	P5(3)	P2(1)	P1(1)	P1(1)		
	P5(3)	P2(1)	P1(1)	P1(1)	P5(1)		
	P2(1)						

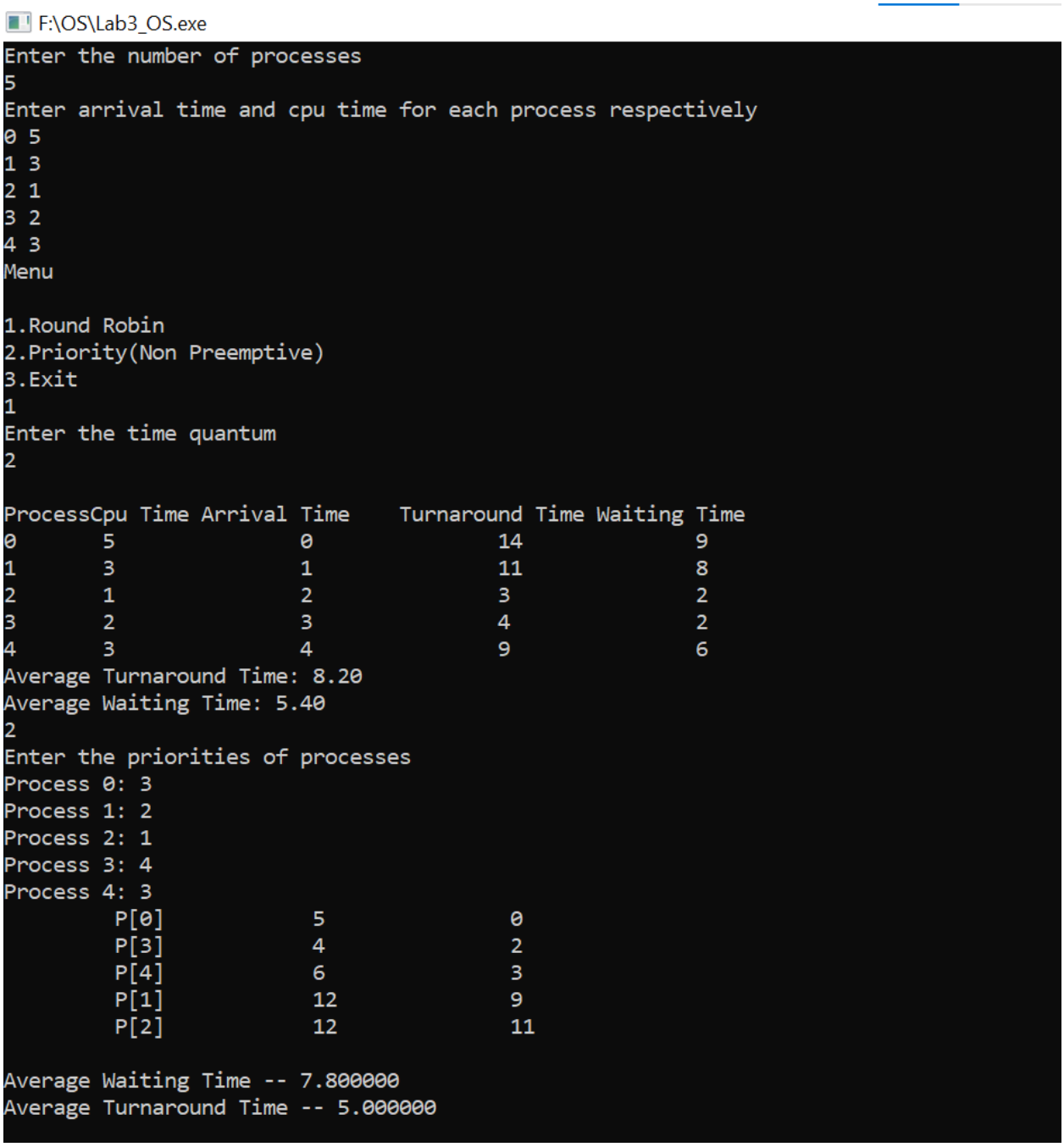
Priority Non preemptive

| P1 | P4 | P5 | P2 | P3 |
0 5 7 10 13 14

P2(3) P2(3) P2(3) P3(1)
P3(1) P3(1) P3(1)
P4(2) P5(3)
P5(2)

28/6/2023

Output:


F:\OS\Lab3_OS.exe
Enter the number of processes
5
Enter arrival time and cpu time for each process respectively
0 5
1 3
2 1
3 2
4 3
Menu
1.Round Robin
2.Priority(Non Preemptive)
3.Exit
1
Enter the time quantum
2
ProcessCpu Time Arrival Time Turnaround Time Waiting Time
0 5 0 14 9
1 3 1 11 8
2 1 2 3 2
3 2 3 4 2
4 3 4 9 6
Average Turnaround Time: 8.20
Average Waiting Time: 5.40
2
Enter the priorities of processes
Process 0: 3
Process 1: 2
Process 2: 1
Process 3: 4
Process 4: 3
P[0] 5 0
P[3] 4 2
P[4] 6 3
P[1] 12 9
P[2] 12 11
Average Waiting Time -- 7.800000
Average Turnaround Time -- 5.000000