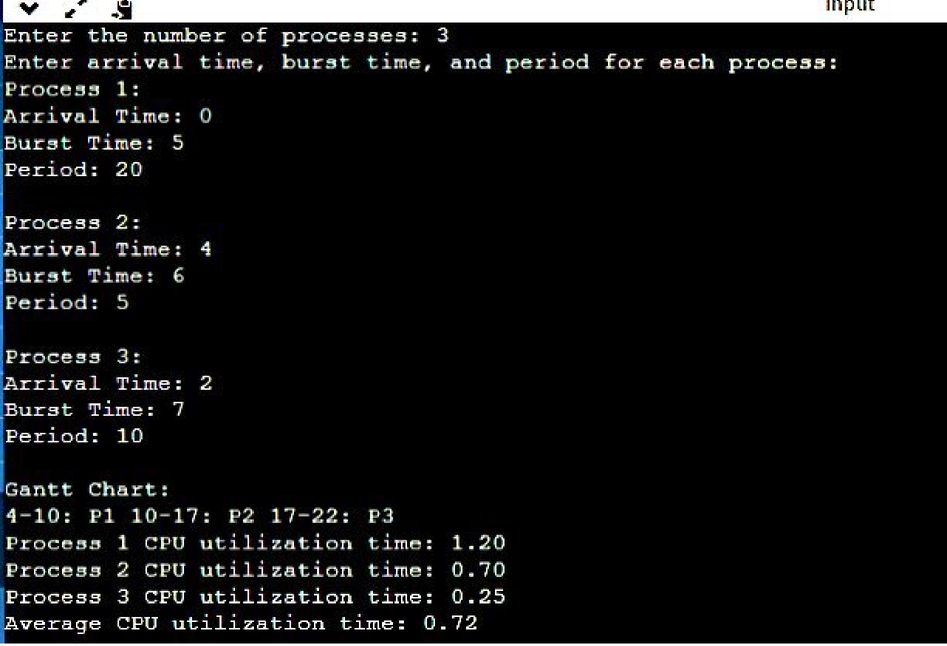
Write a C-program to simulate Real-Time CPU scheduling algorithms: Rate-Monotonic.

```
  #include <stdio.h>
  #define Max-processes 10
 typedef struct {
     int arrival Time;
     int burst Time;
     int period;
     int priority;
 } Process;
void scheduleRM (Process processes [], int n) {
     int time=0;
     int i, j;
 for (i=0; i<n; i++) {
    Processes [i].priority = processes [i].period;
 }
 for (i=0; i<n-1; i++) {
 for (j=0; j<n-i-1; j++){
 if (processes [j].priority >processes [j+1].priority){
process temp= processes [j];
processes [j] = processes [j+1];
processes [j+1] = temp;
}
}
}
Printf (" Gantt chart: \n");
for (i=0; i<n; i++) {
if (time < processes [i].arrivalTime) {
(time = processes [i].arrivalTime};
printf ("%d - %d : P%d", time, time + processes [i].burst
Time, i+1);
time += processes [i].burst Time;
}
Printf ("\n");
}
int main(){
int n, i;
Process processes [Max-processes];
float totalUtilization Time =0;
Printf ("Enter the number of processes :");
Scanf ("%d", &n);
Printf ("Enter arrival time, burst time and period for each
process: \n");
for (i=0; i<n; i++){
Printf ("Process %d :\n", i+1);
Printf ("Arrival Time:");
scanf ("%d", &processes [i].arrivalTime );
Print f ("Burst Time :");
Scanf ("%d", & processes [i].burst Time);
Printf ("Period: ");
scanf ("%d", & processes [i].period);
print f("\n");
}
Schedule RM (processes, n);
for (i=0; i<n; i++){
float utilization Time = (float) processes [i]. burstTime /
            processes [i]. period;
totalUtilization Time+=utilization Time;
Printf ("Process %d CPU utilization time: %.2f \n", i+1,
utilization Time);
}
float averageUtilization Time = totalUtilizationTime/n;
Printf ("Average CPU ~~util~~ utilization time: %..2f \n",
average Utilization Time);
return0;
}
```

```
Enter the number of processes: 3
Enter arrival time, burst time, and period for each process:
Process 1:
Arrival time: 0
Burst time: 5
Period: 20
Process 2:
Arrival time: 4
Burst time: 6
Period: 5
Process 3:
Arrival time: 2
Burst time: 7
Period: 10
Gantt chart:
 4-10: P1   10-17: P2   17-22: P3
Process 1 CPU utilization time: 1.20
Process 2 CPU utilization time: 0.70
Process 3 CPU utilization time: 0.25
Average CPU utilization time: 0.72
```

Input



```
Enter the number of processes: 3
Enter arrival time, burst time, and period for each process:
Process 1:
Arrival Time: 0
Burst Time: 5
Period: 20

Process 2:
Arrival Time: 4
Burst Time: 6
Period: 5

Process 3:
Arrival Time: 2
Burst Time: 7
Period: 10

Gantt Chart:
4-10: P1 10-17: P2 17-22: P3
Process 1 CPU utilization time: 1.20
Process 2 CPU utilization time: 0.70
Process 3 CPU utilization time: 0.25
Average CPU utilization time: 0.72
```