Write a C program to simulate the following CPU scheduling algorithm to find turnaround time and waiting time.
 Priority (pre-emptive & Non-pre-emptive)
 Round Robin (Experiment with different quantum sizes for RR algorithm)

Code:

```
#include<stdio.h>
int at[10],t,pt[10],tat[10],wt[10],n,time=0,i,ready[10],pry[10],op=0, maxpr,x,p[10];
float atat=0,awt=0;
void main()
{
  printf("Enter number of processes \n");
  scanf("%d",&n);
  printf("Enter araival times: \n");
  for(i=0;i< n;i++)
  scanf("%d",&at[i]);
  printf("Enter process times: \n");
  for(i=0;i<n;i++)
  scanf("%d",&pt[i]);
  printf("Enter priority: \n");
  for(i=0;i<n;i++)
  scanf("%d",&pry[i]);
  for(i=0;i<n;i++)
  ready[i]=0;
  for(i=0;i<n;i++)
  p[i]=pt[i];
  for(i=0;i< n;i++)
  time+=pt[i];
  t=n;
  while(t--)
  {
     for(i=0;i<n;i++)
     if(op > = at[i])
     ready[i]=1;
```

```
for(i=0;i< n;i++)
  if(pt[i]==0)
   pry[i]=0;
  //finding index of max priority
   maxpr=pry[0];
  for(i=0;i< n;i++)
   if(ready[i]==1)
  if(pry[i]>maxpr)
   maxpr=pry[i];
   for(i=0;i< n;i++)
  if(maxpr==pry[i])
   x=i;
   //printing chart
  printf("%d p%d ",op,(x+1));
  op=op+pt[x];
   tat[x]=op;
  ready[x]=0;
  pry[x]=0;
}
printf("%d",op);
//finding avgtat and avg wt
for(i=0;i< n;i++)
{
  tat[i]=tat[i]-at[i];
}
for(i=0;i<n;i++)
{
   atat+=tat[i];
  wt[i]=tat[i]-pt[i];
}
for(i=0;i<n;i++)
awt+=wt[i];
awt=awt/n;
atat=atat/n;
```

```
//printing final values
printf("\n");
for(i=0;i<n;i++)
printf("P%d %d %d \n",(i+1),tat[i],wt[i]);
printf("ATAT=%f \nAWT=%f ",atat,awt);
}</pre>
```

Output:

```
PS D:\VS Code\OS> cd "d:\VS Code\OS\" ; if ($?) { gcc npp.c -o npp } ; if ($?) { .\npp } Enter number of processes

4
Enter araival times:
0 1 2 3
Enter process times:
4 3 3 5
Enter priority:
3 4 6 5
0 p1 4 p3 7 p4 12 p2 15
P1 4 0
P2 14 11
P3 5 2
P4 9 4
ATAT=8.000000
AUT=4.250000
PS D:\VS Code\OS>
```

Observation:

```
28-06-23 vorite a c programto simulate CPU scheduling
ONOn-preemptive priority , (1) Round Robin.
                                                                 Per (1=0; 1er; 1+1)
 #include < Stallo. h >.
  winclude (Stello, h).

int attio, to ptio, toutlio), we tool, no time = 0, 1, randy(10), h
                                                                   it ( manyer = = pry
  06=0' would 1 x 1 blig 1, 100 0 =01.4
                                                                       M= 1)
                                                                  point (" " od prod
 float atoto, auto; 1 10
                                                                  op=op+p+(m);
  void maine
                 1,9 [9]
                                                                  tat [7] = 0p;
    print ("Enter no. of proces an");
                                                                   ready[2]=0;
                                                                    my [7] =0)
    Scant (" or od", don);
   print+("Enter arrival times: in");
                                                                 print+ (" yod", op);
    Por (1=0; 1cn; 1++)
                                                                  for (1-0) (cn) (++)
    Scout ( " ord", Lat (1));
                                                                  1 put (1)= tat(1)-
    private (" Enter process Hus: \");
                                                                  tor (1=0; ikn; it+)
    for (120; ian; i++)
    San + ( uy.du, bpt ()); . 8 assorg
                                                                    at at += fat(i)
    printf(" Enter priority: ("")"); and Loving
                                                                      wor (i) = tate
    for (i=0; icn; i++)
                                                                    for lied; icn;
    Scentin glad & Apry (1);
                                                                        aw+ += w+ 1
    for (1=0; icn; 1++)
                                                                    awt=awtin;
     ready (i) =0;
                                                                    atat = atat in:
     Por (1=0) (4) (+7)
     PCi) = PTCi];
                                                                    print+("IN");
                                                                     Por 11=0; icn;
                                                                      printf(" prod
    for ( i=0; ich; i++)
                                                                      Printfl" ATAT
     time += pr(F);
     t=n;
                                                                output,
    whire (+-)
                                                                 Enter number of
        forli= ofich; itT)
                                                                Enter arrival tin
      it (Opsout(1.7)
                                                                  0123
         read [13=1
                                                                Enter process tim
                                                                   4 3 3 5
         for (1-0) (cn; 1++1)
                                                                Enter priority:
          ( 0== (1) 7q) +i
          ory (1) =0',
                                 447 c 8003 6.33
        waxler = bed (0);
       furciso ; icn; int)
If (needly) = 1)
                                                                  P3
          i+ (pryci) > monpr)
                                                                   P4
                                                                       9
                                                                           4
                                                                   ATAT = 8.000
            Leished = Laken
```

```
Cer (1=0; 1cm; 1+1)
                   14 (manyr == pry(1)) mant tolly laby toller at
                      M= 1)
                 print(" " od prod", op, (n+1));
                 op=op+p+(a);
                  tat [7] = 0p;
                  ready[7]=0;
                   pry[7]=0;
                print(" god", op);
 TATTA
                for (1-0) (en) (++)
                 1 fut (i) = tat(i) - at(i);
                to (120; icn; i++)
                 of at at + = tatCi];
                   i [i]tq-(i]tal = [i]tw
                  for (1=0; 1 cm; 1++)
                     aw+ += w+ (i);
                 awt=awtn;
                 atat = atatin;
                 print#("w");
FATA
                  Por (1=0; 12n; ha)
                  printf(" prind you you and you, (iti), tout(i), wot(i));
                  printfl" ATAT = 40 + mAWT = 40 F 11 , atout, awt);
              4.
             output ,
             Enter number of processes
             4
            Enter arrowal time;
              0123
            Enter proces times:
              4 3 35
            Enter priority:
              3 4 65
                     4 P3 7 pu 12 p2 15
             PI
                  4
                      0
                  14
                      11
             P3
                   5
                      2
                      4
              P4
                   9
                              AWT = 4. 2700 .
             ATAT = 8.000
```

Code:

```
#include<stdio.h>
  int tq, at[10], pt[10], p[10], time=0, op=0, i,j,n, ready[10],q[100];
  int r=-1,f=0,tat[10],wt[10],z,fg,y=9999,ch;
  float atat, awt;
int rr(int x)
  if(pt[x]>tq)
  {
     pt[x]=tq;
     op+=tq;
  }
  else
  {
     op+=pt[x];
     pt[x]=0;
     tat[x]=op;
     ready[x]=0;
  }
  return x;
}
void main()
  printf("Enter number or processes \n");
  scanf("%d",&n);
  printf("Enter araival times: \n");
  for(i=0;i<n;i++)
  scanf("%d",&at[i]);
  printf("Enter process times: \n");
  for(i=0;i<n;i++)
  scanf("%d",&pt[i]);
  printf("Enter TQ \n");
  scanf("%d",&tq);
  for(i=0;i<n;i++)
  ready[i]=0;
```

```
for(i=0;i<n;i++)
q[i]=9999;
for(i=0;i< n;i++)
p[i]=pt[i];
for(i=0;i< n;i++)
time+=pt[i];
for(i=0;i< n;i++)
  if(op \ge at[i])
  ready[i]=1;
for(i=0;i< n;i++)
   if(ready[i]==1)
     q[++r]=i;
   }
while(op!=time)
  printf("%d ",op);
  if(z==y)
   q[++f];
   y=z;
  ch=q[f];
  if(pt[ch]!=0)
  z=rr(q[f]);
  printf("P%d ",(z+1));
  for(i=0;i<n;i++)
    if(op>=at[i] && pt[i]!=0)
    {
     fg=0;
     j=f;
     while(j<=r)
        if(i==q[j])
        fg=1;
```

```
j++;
     }
     if(fg==0)
        q[++r]=i;
     }
    }
  if(pt[z]!=0)
  q[++r]=z;
  }
  f++;
}
printf("%d ",op);
for(i=0;i< n;i++)
  tat[i]=tat[i]-at[i];
  wt[i]=tat[i]-p[i];
  atat+=tat[i];
  awt+=wt[i];
}
atat=atat/n;
awt=awt/n;
printf("\n");
for(i=0;i< n;i++)
printf("P%d %d %d \n",(i+1),tat[i],wt[i]);
printf("ATAT=%f \nAWT=%f ",atat,awt);
```

Output:

}

```
PROBLEMS OUTPUT DEBUGCONSOLE TERMINAL

PS D:\VS Code> cd "d:\VS Code\OS\\"; if ($?) { gcc RR1.c -o RR1 }; if ($?) { .\RR1 }

Enter number or processes
5
Enter araival times:
0 1 2 3 4
Enter process times:
5 3 1 2 3
Enter TQ
2
0 P1 2 P3 3 P1 5 P2 7 P4 9 P5 11 P1 12 P2 13 P5 14
P1 12 7
P2 12 9
P3 1 0
P4 6 4
P5 10 7
ATAT=8.2000000
ANT=5.4000000
ANT=5.4000000
PS D:\VS Code\OS>
```

Observation:

```
int to, at (10), pt(10), p(10), time =0, op=0, isish, ready(10), 110
int ==1, f=0; tot(10), wt(10), Ze, to, y=00000, ch;
(1) #include LStallo.h >
                                                                     fosti=0; ich
                                                                       it (ready [i
                                                                        1 2[+++]
   float alod, aust;
    int re(int n)
                                                                      white (00) =
                                                                          print + ("
       1+(p+(x)>+2)
                                                                           11 (200
        1 pt(x)-= +2;
                                                                            9[++
        y op+=+2;
                                                                             y = Z
      elsed
                                                                            cn= 9
                                                                            i+cpt
          Opt= pt(D);
          P+[n]=0;
          tat [n]= op;
                                                                              ber,
        y ready[n]=0;
                                                                                to
       return n'
  void maines
     print # (" Enter number of processes ("");
     Scanflu dodu, Con ?;
     private (" Enter arrival time: (");
      fosc1=01 ( LN : 1++)
     Scantinoloda, Gat(ij);
print f(" Enter proces time: ");
    for 11=0; (cn; i++)
      Scont (" olude, 6 pt[i])
      printf(" Enter TQ");
      Sant (" 110d", 6t2);
                                                                           for (1=0
                                                                            tat [1]
                                                                             W+C1"
      for (1 =0; icn; i++)
                                                                              atat
       ready [i] =0.
                                                                              aw.
      for (1=0; (2n; 1++)
                                                                          Print # (4 p
                                                                          pout to "
       92[1]= 9999;
      forci=o; (cn; i++)
                                                                      output,
       : [i) tq =[i] q
      for (i=o;ikn;i+n)
                                                                              mor
       time+= pt(i);
       tor (100; 1 cm; 120)
        it-cop>= at(i))
         ready[i)=1;
```

```
٢) ١١٠٠/ ١٤٠٥].
                 forti=0; lengitt)
                   ; + (ready [i] == 1)
                    1 2[+++]=i;
                  white (0p; = time)
                     print *(" yod", op);
                      1+(200 Z=-4)
                       9[4+4] 700
                       4=2;
                      cn=9[+];
                       1+(p) [Ch] =0)
                        1 S=01(d[+));
                        print + (" p 4 od", (2+17);
                         toolisolicn; itt)
                          1 it (opsecut(i) 46 pt(1), 20)
                             d to =0',
                                3=4;
                               while (jx= x)
                                ([i]p==i) +i }
                                   f g=1;
                                   5++;
                               it (+g==0)
                                  2[+*>]=i;
                           it(p+(2];20)
                           2[++~]=Z;
                        f+1;
                   for (1=0) (cn) (++)
                   ([[]] - [[]] tot = []] tot | |
                     atate = tat(i);
                   1 aut+= aut(1);
                  about = about (n) anot = anot (n);

Anint # (" paid blood blood ) n" (it 1); bort(i), wa (i)));
               protti ATAT = 917 IN ANT colof ", atod, auxt);
              output,
              Buter not of process 5.
              Enter arrival times 0 12 3 4
              Enter process times 5 31 23
               0 P1 2 P3 3 P1 5 P2 7 P4 9 P5 11 P1 12 PL 13 P5 14
              Enter
                   12
                PI
                                   ATAT =802
                PL
                         0
                                   AWT =504.
```