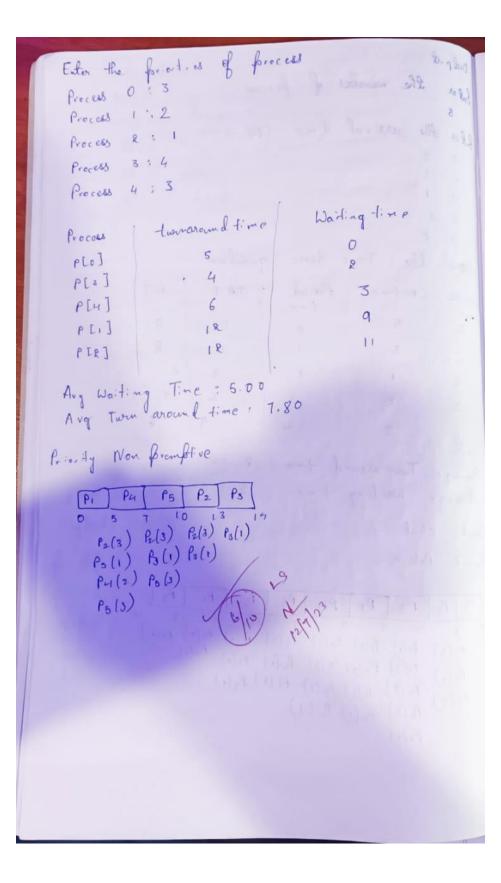
```
write a C foregram to stimulate the following of CPU scheduling algor than to find Lura around
  time & waiting lime.
   -> Paiority (Pre emptive & Non foremptive)
-> Round Robin
  # include < stdio. h >
  # include < conio. h >
  int main () &
   and n. i. bu[10], wa[10]. tat[10]. t. ct[10], at[10]
   float aut=0, att=0. temp=0:
  friend ("Enter the number of process; "):
   Scanf ("1.d" . & n):
  for (:= 0; i < m; i++) {
   frintf ("Enter arrival time for p[".d]: ".i+1):
  Sconf (" '/.d", & alli]):
  fourtf (" Enter burst time for p[:/d]: ", i+1):
  Scanf (" 7.d". & bt [i]).
  frintf ("Enter fronty of p[1/d]: i+1):
  Scarf (" Yod" & priority [:]).
for (i=0; i < n-1; i++) {
   for (j=0; j < n-1-i; j++) {
    if ( providy [ j ] > providy (j+1) }
         temps : prior ty [ ] :
    forority [j] : provity [j+1]
        forority [j+1] = temp?
```

tens ofly) btlil: btlitil: biljil : temp temp at [i] atly] atly. 7: alljil. temp: for (:=0; i < an; i+1) { wtli]: sem-atli]; tat [i] = wt[j + bt[i]: for of (" /d / will): augut + wt [:] : avg tod + = tat [:]: lum + = bt [:]: flood argulf = (flood) argulln : fleat augital f = (float) ovg-lat In: for of (In Total average waiting time: 14 avg wtf) finds ("In Total average turnaround time: "If gdch(): Action 0:

```
e) # include < Stdio. h >
    # include < sonio. h >
       int n. i. pt [RO] - formerly [RO] at [RO] is temp. wt[or] that [RO] sum: 0. avgust = 0. avgst at = 0:
       formal ( Enter number of forces : "):
Scarf (" "/od", &n):
          Brintf ("Enter assival time for P[1/d]:", i+1):
           Scanf ( 1.d., & at [:]).
          formif ("Enter dourst time for p[://d]:",i+1);
           Scarf ( " 1.d , & bt[:]):
           for alf ("Enter for orty of p [-/.d]: ". :+v):
            & canf (" 1.d", & boiorfy [:]):
       for (1:03 i < n-13 j+1) {
          for (j=0: j < m-1-1: j++) {
              if (formity (j) > frienty (j+1)) }
              formy by by by by
               brionfylitil: temp;
f D
             temp : pt [j]
              bt GJ: pt [j+1]:
              btlj + 1] : temp:
               temp = odlij:
                attij] = attij+17;
                atlij +1] = temp;
```

```
temp st Ej]
                                   of include comes
       61 [] 1 6 [j + 1];
                                         7 1) mission (1) 7
                                                          Ens.
      btlj 11]: dempi
        demp at [j]:
                                                          8.8
        atlij] atlij+1],
       ally 11]: temp;
   fa (i=0; i<n; i+) f
      wtli]: Sum - all ]:
      totli] = wtli] = ptli];
      for if (" ". d \m" , w(fi]):
        avgut + : wtli]:
        argital + = +allil;
         sum t = btlil;
    3
 float argut f = (float) arguntles;
  floot any tool f = Glood) any tool I m:
Brindf ("In Total any waiting time: "If", any will brindf ("In Total any turn around time: "If", any total
 g & ch ().
 ret un 0;
```

```
Output
       the number of process
Ender
   5
Ent ea
        5
    0
         3
                                                       WT
  Enter
                                         TA T
                          Arrival
                                                        9
 Process
                                             14
                                0
                                                        8
                5
                                             11
    0
                                1
                                2
                                               4
                                  3
       3
                                   4
                  3
                                         8.20
   Average Turn around time
    Avorage Waiting
             chat
    Round Rob. n
                                           P2
                                    P5
                                                12
         Pr(3) Po(1) Pr(3) P4(2) P5(3) P2(1) Pr(1) P5(1)
         P_{3}(1) P_{4}(2) P_{3}(3) P_{2}(1) P_{1}(1) P_{5}(1) P_{1}(2) P_{1}(3) P_{2}(1) P_{1}(1) P_{2}(1) P_{3}(1)
                 B(3) P2(1) P, (1)
                  P2(1)
```



Output:

```
Enter the number of processes
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
Enter the time quantum
ProcessCpu Time Arrival Time
                                  Turnaround Time Waiting Time
        5
                          Θ
                                           14
                                                            8
        3
                          1
                                           11
        1
                          2
                                           3
                                                            2
        2
                          3
                                           4
        3
                          4
                                           9
                                                            6
Average Turnaround Time: 8.20
Average Waiting Time: 5.40
Enter the priorities of processes
Process 0: 3
Process 1: 2
Process 2: 1
Process 3: 4
Process 4: 3
         P[0]
P[3]
P[4]
P[1]
                                            3
                           6
                           12
                                            9
                                            11
          P[2]
                           12
Average Waiting Time -- 7.800000
Average Turnaround Time -- 5.000000
```