PROGRAM NUMBER :1

AIM:
Write a C program to simulate then non-pre-emptive CPU scheduling algorithms for finding turnaround time and waiting time.
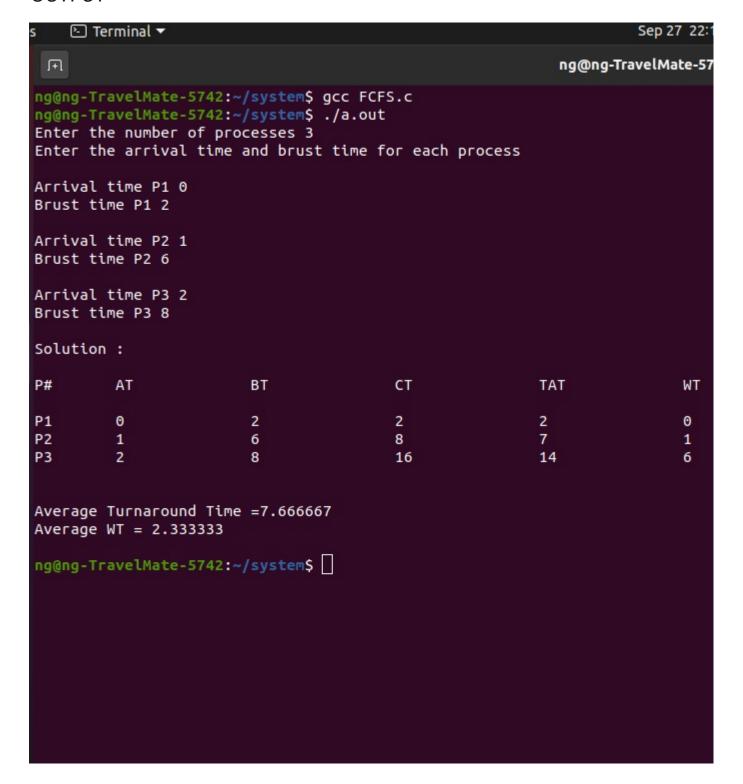1.First Come First Serve(FCFS)
2.Shortest Job First(SJF)
PROGRAM

1.FCFS

```c
#include<stdio.h>
int main()
{ int bt[10]={0},at[10]={0},tat[10]={0},wt[10]={0},ct[10]={0};
  int n,i=0,j=0,k=0,sum=0;
  float totalTAT=0,totalWT=0;
  printf("Enter the number of processes ");
  scanf("%d",&n);
  printf("Enter the arrival time and brust time for each process \n\n");
  for(i=0;i<n;i++)
  { printf("Arrival time P%d ",i+1);
    scanf("%d",&at[i]);
    printf("Brust time P%d ",i+1);
    scanf("%d",&bt[i]);
    printf("\n");
  }
  for(j=0;j<n;j++)
  {    sum+=bt[j];
      ct[j]+=sum;
  }
  for(k=0;k<n;k++)
  {   tat[k]=ct[k]-at[k];
      totalTAT+=tat[k];
  }
  for(k=0;k<n;k++)
  {   wt[k]=tat[k]-bt[k];
      totalWT+=wt[k];
  }
  printf("Solution :\n\n");
  printf("P#\t AT\t\tBT\t\tCT\t\tTAT\t\tWT\t\n\n");
  for(i=0;i<n;i++)
  {printf("P%d\t %d\t\t%d\t\t%d\t\t%d\t\t%d\n",i+1,at[i],bt[i],ct[i],tat[i],
  }
  printf("\n\nAverage Turnaround Time =%f\n",totalTAT/n);
  printf("Average WT = %f\n\n",totalWT/n);
  return 0;
}

ng@ng-TravelMate-5742:~/system$ 
```

OUTPUT

ng@ng-TravelMate-57

```
ng@ng-TravelMate-5742:~/system$ gcc FCFS.c
ng@ng-TravelMate-5742:~/system$ ./a.out
Enter the number of processes 3
Enter the arrival time and brust time for each process

Arrival time P1 0
Brust time P1 2

Arrival time P2 1
Brust time P2 6

Arrival time P3 2
Brust time P3 8

Solution :

P#         AT              BT              CT            TAT           WT

P1         0               2               2             2             0
P2         1               6               8             7             1
P3         2               8               16            14            6


Average Turnaround Time =7.666667
Average WT = 2.333333

ng@ng-TravelMate-5742:~/system$
```

PROGRAM

2.SJF

```c
#include<stdio.h>
int main()
{ int bt[20],p[20],wt[20],tat[20],i,j,n,total=0,pos,temp;float avg_wt,avg_t
  printf("Enter number of process:");
  scanf("%d",&n);
  printf("\n Enter Brust Time:\n");
  for(i=0;i<n;i++)
  { printf("p%d:",i+1);
    scanf("%d",&bt[i]);
    p[i]=i+1;}
  for(i=0;i<n;i++)
  { pos=i;
    for(j=i+1;j<n;j++)
    { if(bt[j]<bt[pos])
          pos=j;}
    temp=bt[i];
    bt[i]=bt[pos];
    bt[pos]=temp;
    temp=p[i];
    p[i]=p[pos];
    p[pos]=temp;}
  wt[0]=0;
  for(i=0;i<n;i++)
   { wt[i]=0;
    for(j=0;j<i;j++)
        wt[i]+=bt[j];
    total+=wt[i];}
  avg_wt=(float)total/n;
  total=0;
  printf("\nProcess\t\tBrust Time\t\tWaiting Time\tTrunaround Time");
  for(i=0;i<n;i++)
  { tat[i]=bt[i]+wt[i];
    total+=tat[i];
    printf("\np%d\t\t%d\t\t\t%d\t\t\t%d",p[i],bt[i],wt[i],tat[i]);}
  avg_tat=(float)total/n;
  printf("\n\nAverage Waiting Time=%f",avg_wt);
  printf("\n\nAverage Turnaround Time=%f",avg_tat)}
ng@ng-TravelMate-5742:~/system$ 
```

OUTPUT

```
ng@ng-TravelMate-5742:~/system$ gcc SJF.c
ng@ng-TravelMate-5742:~/system$ ./a.out
Enter number of process:3

 Enter Brust Time:
p1:2
p2:4
p3:1

Process          Brust Time            Waiting Time      Trunaround Time
p3               1                     0                              1
p1               2                     1                              3
p2               4                     3                              7

Average Waiting Time=1.333333

Average Turnaround Time=3.666667
ng@ng-TravelMate-5742:~/system$ ▯
```

RESULT

Program is executed successfully and output is obtained.

BY NIVEA GIGEN
   S5 C
   CHN18CS092