Week - 2
(15 June 2023)
Experiment - 2

Question:

Write a C program to simulate the following non-pre-emptive CPU scheduling algorithm to find turnaround time and waiting time:
(a) FCFS
(b) SJF (Non-pre-emptive)

Program:

```
#include<stdio.h>

   int n, i, j, pos, temp, choice, Burst_time[20], Waiting_time[20], Turn_around_time[20], process[20],
   total=0;
   float avg_Turn_around_time=0, avg_Waiting_time=0;

int FCFS()
{
   Waiting_time[0]=0;

   for(i=1;i<n;i++)
   {
      Waiting_time[i]=0;
      for(j=0;j<i;j++)
         Waiting_time[i]+=Burst_time[j];
   }

   printf("\nProcess\t\tBurst Time\t\tWaiting Time\t\tTurnaround Time");

   for(i=0;i<n;i++)
   {
      Turn_around_time[i]=Burst_time[i]+Waiting_time[i];
      avg_Waiting_time+=Waiting_time[i];
      avg_Turn_around_time+=Turn_around_time[i];
      printf("\nP[%d]\t\t%d\t\t\t%d\t\t\t\t%d",i+1,Burst_time[i],Waiting_time[i],Turn_around_time[i]);
   }

   avg_Waiting_time =(float)(avg_Waiting_time)/(float)i;
   avg_Turn_around_time=(float)(avg_Turn_around_time)/(float)i;
   printf("\nAverage Waiting Time:%.2f",avg_Waiting_time);
   printf("\nAverage Turnaround Time:%.2f\n",avg_Turn_around_time);

   return 0;
}

int SJF()
{
   //sorting
   for(i=0;i<n;i++)
   {
      pos=i;
      for(j=i+1;j<n;j++)
      {
```

```
        if(Burst_time[j]<Burst_time[pos])
            pos=j;
      }

      temp=Burst_time[i];
      Burst_time[i]=Burst_time[pos];
      Burst_time[pos]=temp;

      temp=process[i];
      process[i]=process[pos];
      process[pos]=temp;
   }
      Waiting_time[0]=0;


   for(i=1;i<n;i++)
   {
      Waiting_time[i]=0;

      for(j=0;j<i;j++)
         Waiting_time[i]+=Burst_time[j];

      total+=Waiting_time[i];
   }

   avg_Waiting_time=(float)total/n;
   total=0;

   printf("\nProcess\t\tBurst Time\t\tWaiting Time\t\tTurnaround Time");

   for(i=0;i<n;i++)
   {
      Turn_around_time[i]=Burst_time[i]+Waiting_time[i];
      total+=Turn_around_time[i];
      printf("\nP[%d]\t\t%d\t\t\t%d\t\t\t\t%d",process[i],Burst_time[i],Waiting_time[i],Turn_around_time[i]);
   }

   avg_Turn_around_time=(float)total/n;
   printf("\n\nAverage Waiting Time=%f",avg_Waiting_time);
   printf("\nAverage Turnaround Time=%f\n",avg_Turn_around_time);
}

int main()
{
   printf("Enter the total number of processes:");
   scanf("%d",&n);

   printf("\nEnter Burst Time:\n");
   for(i=0;i<n;i++)
   {
      printf("P[%d]:",i+1);
      scanf("%d",&Burst_time[i]);
      process[i]=i+1;
   }
```

```
   while(1)
   {   printf("\n-----MAIN MENU-----\n");
       printf("1. FCFS Scheduling\n2. SJF Scheduling\n");
       printf("\nEnter your choice:");
       scanf("%d", &choice);
       switch(choice)
       {
          case 1: FCFS();
          break;

          case 2: SJF();
          break;

          default: printf("Invalid Input!!!");
       }
   }
   return 0;
}
```

Output:

```
Enter the total number of processes:3

Enter Burst Time:
P[1]:5
P[2]:12
P[3]:19

-----MAIN MENU-----
1. FCFS Scheduling
2. SJF Scheduling

Enter your choice:1

Process         Burst Time              Waiting Time            Turnaround Time
P[1]            5                       0                               5
P[2]            12                      5                               17
P[3]            19                      17                              36
Average Waiting Time:7.33
Average Turnaround Time:19.33

-----MAIN MENU-----
1. FCFS Scheduling
2. SJF Scheduling

Enter your choice:2

Process         Burst Time              Waiting Time            Turnaround Time
P[1]            5                       0                               5
P[2]            12                      5                               17
P[3]            19                      17                              36

Average Waiting Time=7.333333
Average Turnaround Time=19.333334
```

```
Enter the total number of processes:3

Enter Burst Time:
P[1]:19
P[2]:5
P[3]:12

-----MAIN MENU-----
1. FCFS Scheduling
2. SJF Scheduling

Enter your choice:1

Process         Burst Time              Waiting Time            Turnaround Time
P[1]            19                      0                               19
P[2]            5                       19                              24
P[3]            12                      24                              36
Average Waiting Time:14.33
Average Turnaround Time:26.33

-----MAIN MENU-----
1. FCFS Scheduling
2. SJF Scheduling

Enter your choice:2

Process         Burst Time              Waiting Time            Turnaround Time
P[2]            5                       0                               5
P[3]            12                      5                               17
P[1]            19                      17                              36

Average Waiting Time=7.333333
Average Turnaround Time=19.333334
```

Observation Book Pictures:

PAGE NO :
DATE : 15/06/2023

Experiment - 2

Write a C program to simulate the following non-pre-emptive CPU scheduling algorithm to find turnaround time and waiting time.

a) FCFS
b) SJF

Program:

(a)
```
#include <stdio.h>
int main()
{ int i, j, n, Burst_time[10], Waiting_time[10],
  Turn_around_time[10];
float avg_Turn_around_time = 0,
      avg_Waiting_time = 0;

printf("Enter the total no. of processes : ");
scanf("%d", &n);

printf("Enter the Process Burst Time : \n");
for(i=0; i<n; i++)
  { printf("P[%d] : ", i+1);
    scanf("%d", &Burst_time[i]);
  }
Waiting_time[0] = 0;

for (i=1; i<n; i++)
  { Waiting_time[i] = 0;
    for (j=0; j<i; j++)
      Waiting_time[i] += Burst_time[j];
  }
```

```
printf("\n Process \t\t Burst Time \t\t
        waiting time \t\t Turn around time");

for (i=0; i<n; i++)
{ Turnaround_time[i] = Burst_time[i] +
                       waiting_time[i];
avg_waiting_time += waiting_time[i];
avg_Turn_around_time += Turn_around_time[i];
printf("\n P[%d] \t\t %d \t\t\t %d \t\t\t
        %d", i+1, Burst_time[i], waiting_time[i],
        Turn_around_time[i]);
}

avg_waiting_time = (float)(avg_waiting_time)/(float)(i);
avg_Turn_around_time = (float)(avg_Turn_around_time)/(float)(i);

printf("\nAverage waiting Time : %.2f",
        avg_waiting_time);
printf("\n Average Turn Around Time : %.2f",
        avg_Turn_around_time);

return 0;

}
```

PAGE NO :
DATE :

Output:

Enter the total no. of processes : 3

Enter the process Burst time:

P[1] : 5

P[2] : 12

P[3] : 19

Process	Burst time	Waiting time	Turnaround time
P[1]	5	0	5
P[2]	12	5	17
P[3]	19	17	36

Average Waiting Time : 7.33

Average Turnaround Time : 19.33.

N
22/6/23

(b)

```
#include<stdio.h>
{ int Burst_time[20], process[20],
    Waiting_time[20], Turn_around_time[20],
    i, j, n, total = 0, pos, temp ;

  float avg_Waiting_time, avg_Turn_around_time;

printf("Enter no. of processes: ");
scanf("%d", &n);

printf("\n Enter Burst time:\n");
for(i=0; i<n; i++)
  { printf("p[%d]:" i+1)
    scanf("%d", &Burst_time[i]);
    process[i] = i+1;
  }

//Sorting
for(i=0; i<n; i++)
{ pos = i;
  for(j= i+1; j<n; j++)
    { if (Burst_time[j] < Burst_time[pos])
        pos=j;
    }
  temp = Burst_time[i]
  Burst_time[i] = Burst_time[pos];
  Burst_time[pos] = temp;

  temp = process[i];
  process[i] = process[pos];
  process[pos] = temp;

  Waiting_time[0] = 0;
```

→

```
for (i = 1; i < n; i++)
{ waiting_time[i] = 0;
    for (j = 0; j < i; j++)
        waiting_time[i] += Burst_time[j];
    total += waiting_time[i];
}

avg_waiting_time = (float) total / n;
total = 0;

printf("\n Process \t Burst Time \t Waiting time
        \t Turn around time");
for (i=0; i<n; i++)
{ Turnaround_time[i] = Burst_time[i] +
                       waiting_time[i];

  total += Turn_around_time[i];
printf("\n P[%d] \t \t %d \t \t %d \t \t \t %d",
        process[i], Burst_time[i], waiting_time[i],
        Turn_around_time[i]);
}

avg_Turn_around_time = (float) total/n;
printf("\n\n Average Waiting Time = %f",
        avg_waiting_time);
printf("\n Average Turnaround Time = %f\n",
        avg_Turn_around_time);

}
```

Output:

Enter the no of processes : 3

Enter Burst Time:

P[1] : 19

P[2] : 5

P[3] : 12

Process	Burst Time	Waiting Time	Turnaround Time
P[2]	5	0	5
P[3]	12	5	17
P[1]	19	17	36

Average Waiting Time : 7.333

Average Turnaround Time : 19.333

22/6/23