Experiment 6

Subject: CSL403 Operating System Lab

NAME: GINI CHACKO

ROLL: 8942

CLASS: SE COMPS B

BATCH: B

Aim: Study Process Scheduling

Objectives: Implement various Process scheduling algorithm and evaluate their performance

Problem Statement:
Write a program to demonstrate Process Scheduling Algorithms.

Batch (A): First Come First Serve (FCFS) ,NonPreemptive Shortest Job First (SJF)

Batch (B): Non Preemptive Shortest Job First (SJF) ,Shortest Remaining Time First (SRTF)

Batch (C):Round Robin Algorithm (RR),Non Preemptive Priority (NPP)

Batch (D): Non Preemptive Priority (NPP),Premptive Priority (PP)

1. Calculate WT, AWT, TAT, ATAT.

2. Compare the result of algorithms for a problem and find which algorithm is performing better.

Answer:

1. **Calculate WT, AWT, TAT, ATAT.**

A.] Process Scheduling – Non-Pre-Emptive(SJF)

```
Process     Burst Time          Waiting Time    Turnaround Time
p2                1                   0                  1
p1                2                   1                  3
p4                4                   3                  7
p5                5                   7                  12
p3                8                   12                 20

Average Waiting Time = 4.600000 ms

Average Turnaround Time = 8.600000 ms
```

B.] Process Scheduling – Shortest Remaining Time First (SRTF)

```
Process   BT      AT      WT      TAT     CT
p1        8       0       15      23      23
p2        4       2       3       7       9
p3        1       1       0       1       2
p4        3       2       0       3       5
p5        7       0       8       15      15

Average waiting time = 5.200000 ms

Average Turnaround time = 9.800000 ms
```

2. **Compare the result of algorithms for a problem and find which algorithm is performing better.**

Ans: SRTF is a pre-emptive variant of SJF. And SJF gives minimum average waiting time for a given set of processes. For minimum waiting time SRTF Is better than SJF. If arrival time and burst time is same then also SJF and SRTF will give same waiting time. Hence there should not be any comparison between SJF and SRTF.

Program Section:

A.] Process Scheduling – Non-Pre-Emptive(SJF)

CODE:

```
#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_tat;
   printf("*****PROCESS SCHEDULING - NON-PRE-EMPTIVE(SJF)*****\n\n");
   printf("Enter number of process : ");
   scanf("%d",&n);

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

  //sorting of burst times
   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=1;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("\nProcesst    Burst Time  \tWaiting Time \tTurnaround Time");
   for(i=0;i<n;i++)
   {
      tat[i]=bt[i]+wt[i];
      total+=tat[i];
      printf("\np%d\t\t  %d\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\n",avg_wt);
   printf("\nAverage Turnaround Time=%f\n\n",avg_tat);
}
```

B.] Process Scheduling – Shortest Remaining Time First (SRTF)

CODE:

```
#include<stdio.h>
int main()
{
   int a[10],b[10],x[10];
   int waiting[10],turnaround[10],completion[10];
   int i,j,smallest,count=0,time,n;
   double avg=0,tt=0,end;
   printf("*****PROCESS SCHEDULING - SHORTEST REMAINING TIME
FIRST(SRTF)*****\n");
   printf("\nEnter the number of Processes : ");  //input
   scanf("%d",&n);
   for(i=0; i<n; i++)
   {
      printf("\nEnter arrival time of process : ");  //input
      scanf("%d",&a[i]);
   }
   for(i=0; i<n; i++)
   {
      printf("\nEnter burst time of process : ");  //input
      scanf("%d",&b[i]);
   }
   for(i=0; i<n; i++)
      x[i]=b[i];

   b[9]=9999;
   for(time=0; count!=n; time++)
   {
      smallest=9;
      for(i=0; i<n; i++)
      {
         if(a[i]<=time && b[i]<b[smallest] && b[i]>0 )
```

```
            smallest=i;
        }
        b[smallest]--;

        if(b[smallest]==0)
        {
           count++;
           end=time+1;
           completion[smallest] = end;
           waiting[smallest] = end - a[smallest] - x[smallest];
           turnaround[smallest] = end - a[smallest];
        }
    }
    printf("Process   BT\t  AT\t  WT\t  TAT\t  CT");
    for(i=0; i<n; i++)
    {
       printf("\np%d\t  %d\t  %d\t  %d\t  %d\t  %d ",(i+1), x[i], a[i], waiting[i], turnaround[i],
completion[i]);

       avg = avg + waiting[i];
       tt = tt + turnaround[i];
    }
    printf("\n\nAverage waiting time = %f ms", avg/n);
    printf("\n\nAverage Turnaround time = %f ms",tt/n);
}
```

Output Section:

A.] Process Scheduling – Non-Pre-Emptive(SJF)

OUTPUT:

```
gini@gini:~/Practicals/OS_LAB_3$ gcc exp_3.c
gini@gini:~/Practicals/OS_LAB_3$ ./a.out
*****PROCESS SCHEDULING - NON-PRE-EMPTIVE(SJF)*****

Enter number of process : 5

Enter Burst Time :
p1 : 2
p2 : 1
p3 : 8
p4 : 4
p5 : 5

Process     Burst Time          Waiting Time    Turnaround Time
p2                1                   0                   1
p1                2                   1                   3
p4                4                   3                   7
p5                5                   7                   12
p3                8                   12                  20

Average Waiting Time = 4.600000 ms

Average Turnaround Time = 8.600000 ms
```

B.] Process Scheduling – Shortest Remaining Time First (SRTF)

OUTPUT:

```
gini@gini:~/Practicals/OS_LAB_3$ gcc exp_3b.c
gini@gini:~/Practicals/OS_LAB_3$ ./a.out
*****PROCESS SCHEDULING - SHORTEST REMAINING TIME FIRST(SRTF)*****

Enter the number of Processes : 5

Enter arrival time of process : 0

Enter arrival time of process : 2

Enter arrival time of process : 1

Enter arrival time of process : 2

Enter arrival time of process : 0

Enter burst time of process : 8

Enter burst time of process : 4

Enter burst time of process : 1

Enter burst time of process : 3

Enter burst time of process : 7
Process   BT      AT      WT      TAT     CT
p1        8       0       15      23      23
p2        4       2       3       7       9
p3        1       1       0       1       2
p4        3       2       0       3       5
p5        7       0       8       15      15

Average waiting time = 5.200000 ms

Average Turnaround time = 9.800000 ms
```