Experiement No. 2
Experiment Name : Study of SJF(Shortest Job First) scheduling in operating systems with considering the arrival time and without considering the arrival time of the processes.

Theory :
Shortest job first(SJF) is a scheduling algorithm that is used to schedule processes in an operating system.We know that Shortest job first is a scheduling algorithm in which the process with the smallest execution time is selected for execution next. It is the easiest and simplest CPU scheduling algorithm. In this type of algorithm, the process which has the smallest burst time or execution time gets the CPU allocation first. There are two types of SJF
1) Preemptive SJF
2) Non-Preemptive SJF
These algorithms schedule processes in the order in which the shortest job is done first. It has a minimum average waiting time.There are 3 factors to consider while solving SJF, they are
1. BURST Time
2. Average waiting time
3. Average turnaround time
For some given processes with their arrival time and burst time, the waiting time and the turnaround time can be calculated. For that purpose there some equations are needed (considering arrival time) and they are:

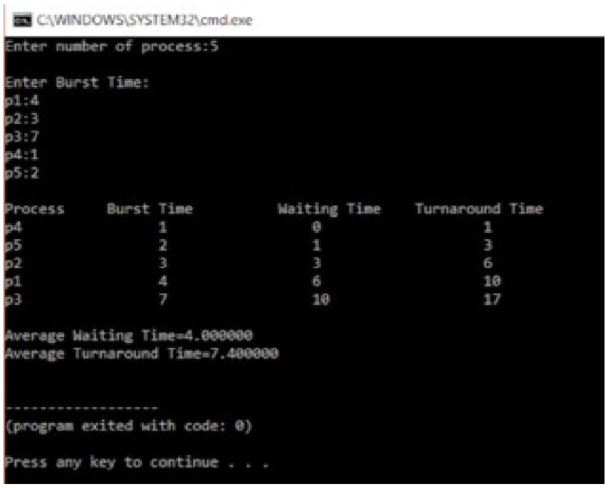
Turnaround Time = Completion Time – Arrival Time
Waiting Time = Turnaround Time – Arrival Time

Code:

```
#include<bits/stdc++.h>
using namespace std;
int main()
{
int bt[20],p[20],wt[20],tat[20];
float avg_wt,avg_tat;
cout<<"Enter number of process:";
cin>>("%d",&n);
printf("Enter Burst Time:n");
for(int i=0;i<n;i++)
{
cout<<"p%d:",i+1;
cin>>"%d",&bt[i];
p[i]=i+1;
}
// sorting burst time
for(int i=0;i<n;i++)
{
pos=i;
for(int 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(int i=1;i<n;i++)
{
wt[i]=0;
for(int j=0;j<i;j++)
wt[i]+=bt[j];
total+=wt[i];
}
avg_wt=(float)total/n;
total=0;
cin>>"nProcesst Burst Time tWaiting TimetTurnaround Time";
for(int i=0;i<n;i++)
{
tat[i]=bt[i]+wt[i];
total+=tat[i];
cout<<"\n\nProcess \tBurst Time \tWaiting Time \tTurnaround Time"<<endl;
}
avg_tat=(float)total/n;
printf("nnAverage Waiting Time=%f",avg_wt);
printf("nAverage Turnaround Time=%fn",avg_tat);
}
```

Output:



```
Enter number of process:5

Enter Burst Time:
p1:4
p2:3
p3:7
p4:1
p5:2

Process     Burst Time          Waiting Time    Turnaround Time
p4              1                   0                1
p5              2                   1                3
p2              3                   3                6
p1              4                   6                10
p3              7                   10               17

Average Waiting Time=4.000000
Average Turnaround Time=7.400000


------------------
(program exited with code: 0)

Press any key to continue . . .
```

Discussion:

In the above program, we calculate the average waiting and average turn around times of the jobs. We first ask the user to enter the number of processes and store it in n. We then accept the burst times from the user. It is stored in the bt array. After this, the burst times are sorted in the next section so the shortest one can be executed first. Here selection sort is used to sort the array of burst time *bt.* Waiting time of the first element is zero, the remaining waiting time is calculated by using two for loop that runs from 1 to in that controls the outer loop and the inner loop is controlled by another for loop that runs from j=0 to j<i. Inside the loop, the waiting time is calculated by adding the burst time to the waiting time**.**

Conclusion:

After doing this experiment, we learned about the Shortest Job First (SJF) scheduling algorithm.Then we learned how to calculate the waiting time and the turnaround time also average waiting and turnaround time for each process without considering the arrival time.And last we have implemented the coding of this algorithm successfully.

Submission Date :
Course No: ECE 3222
Course Title: Operating system sessional

Submitted To:
Hafsa Binte Kibria
Lecturer, Dept of ECE
RUET

Submitted From:
Tamim Hasan
Roll: 1810044
Dept of ECE
Session 2018-19
RUET