

# Lab-Report

Report No: 08

Course code: ICT-3110

Course title: Operating System Lab

Date of Performance:

Date of Submission:

## Submitted by

Name: Ashikur Rahman Miran

ID:IT-18014

3<sup>rd</sup> year 1<sup>st</sup> semester

Session: 2017-2018

Dept. of ICT

MBSTU.

## Submitted To

Nazrul Islam

Assistant Professor

Dept. of ICT

MBSTU.

## Experiment No: 08

**Experiment Name:** Implementation of SJF scheduling algorithm.

### Objectives:

- i) What is SJF scheduling algorithm?
- ii) How to implement SJF scheduling algorithm.

### Theory:

SJF algorithm is also known as Shortest-Job-First algorithm. It is a non-preemptive discipline in which waiting job with small burst time is run next. In other words, when CPU is available, it is assigned to the process that has smallest next CPU burst. The SJF scheduling is especially appropriate for batch jobs for which the run times are known in advance.

In this algorithm, we are sort the queue of process according to burst time and then we calculate waiting time and turnaround time by following rule:

Waiting time = start time –arrival time

Turnaround time = burst time + waiting time

### Source code:

```
#include<bits/stdc++.h>
using namespace std;
int main()
{
    int bt,p,i,wt=0,tat=0,n,bwt[100];
    double twt=0,ttat=0;
    vector<pair<int,int>>>v;
    cout<<"Enter the Number of process: ";
    cin>>n;
    cout<<"Enter Burst time: "<<endl;
```

```

for(i=1; i<=n; i++)
{
    cout<<"p"<<i<<": ";
    cin>>bt;
    v.push_back(make_pair(bt,i));
}
sort(v.begin(),v.end());
cout<<"Process\tBurst Time\tWaiting Time\tTurnaround Time"<<endl;
bwt[0]=0;
for(i=0;i<n;i++)
{
    cout<<"p"<<v[i].second<<"\t"<<v[i].first;
    cout<<"\t\t"<<wt;
    wt+=v[i].first;
    bwt[i+1]=wt;
    twt+=wt;
    tat=v[i].first+bwt[i];
    ttat+=tat;
    cout<<"\t\t"<<tat<<endl;
}
int tttt = twt-wt;
cout<<"Total wait time: "<<tttt<<endl;
cout<<"Average wait time: "<<(double)tttt/n<<endl;
cout<<"Total turnaround time: "<<ttat<<endl;
cout<<"Total average turnaround time: "<<(double)ttat/n<<endl;
}

```

## Output:

```
"D:\programming\c & c++ programming\algorithm\SJF algorithm.exe"
Enter the 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
Total wait time: 20
Average wait time: 4
Total turnaround time: 37
Total average turnaround time: 7.4

Process returned 0 (0x0)   execution time : 6.654 s
Press any key to continue.
```

## Conclusion:

In this lab I learn how to implement SJF scheduling algorithm and also run the code and shows the output and output is expected.