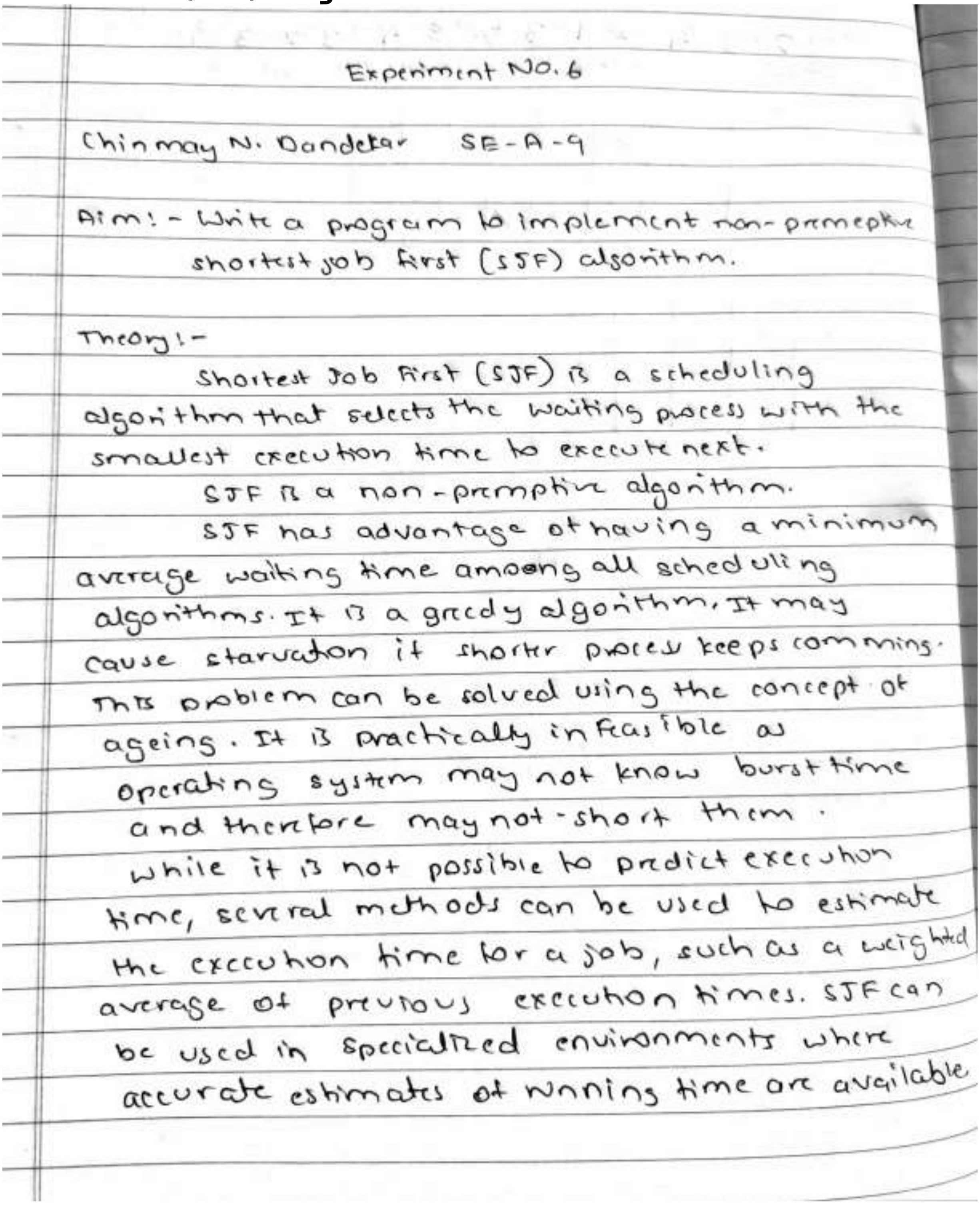
Chinmay Dandekar
SE-A-9

Experiment no. 4

Aim: Write a program to implement Non-Premeptive Shortest Job First (SJF) Algorithm.



Experiment No. 6

Chinmay N. Dandekar SE-A-9

Aim:- Write a program to implement non-premeptive shortest job first (SJF) algorithm.

Theory:-

Shortest Job First (SJF) is a scheduling algorithm that selects the waiting process with the smallest execution time to execute next.

SJF is a non-premptive algorithm.

SJF has advantage of having a minimum average waiting time amoong all scheduling algorithms. It is a greedy algorithm, It may cause starvation if shorter process keeps comming. This problem can be solved using the concept of ageing. It is practically infeasible as operating system may not know burst time and therefore may not - short them. while it is not possible to predict execution time, several methods can be used to estimate the execution time for a job, such as a weighted average of previous execution times. SJF can be used in specialized environments where accurate estimates of running time are available

Code:

```
import java.io.*;

import java.util.*;


class sjf

{


    public static void main(String args[])

    {

        int bt[]=new int[10];

        int ct[]=new int[10];

        int tat[]=new int[10];

        int wt[]=new int[10];

        int p[]=new int[10];

        int i,n,j,temp;


        Scanner s=new Scanner(System.in);


        System.out.println("Enter no of Processes");

        n=s.nextInt();


        for(i=0;i<n;i++)

        {

            System.out.println("Enter the Burst Time for
Process"+(i+1));

            bt[i]=s.nextInt();

            p[i]=i+1;

        }


        for(i=0;i<n;i++)
        {
            for(j=0;j<n-1;j++)
            {
                if(bt[j]>bt[j+1])
                {
                    temp=bt[j];
                    bt[j]=bt[j+1];
                    bt[j+1]=temp;
                    temp=p[j];
                    p[j]=p[j+1];
                    p[j+1]=temp;
                }
            }
        }

        ct[0]=bt[0];

        for(i=1;i<=n;i++)
        {
            ct[i]=ct[i-1]+bt[i];
        }

        for(i=0;i<n;i++)
        {
            tat[i]=ct[i];
            wt[i]=tat[i]-bt[i];
        }


    System.out.println("Process\t\t"+"BT\t\t"+"CT\t\t"+"TAT\t\t"+"W
T\t\t");
```

```
        for(i=0;i<n;i++)
        {

    System.out.println("P"+p[i]+"\t\t"+bt[i]+"\t\t"+ct[i]+"\t\t"+tat[i]+"\t\t"+wt[i]+"\t\t");
        }

        float atat=0,awt=0;

        for(i=0;i<n;i++)
        {
            atat=atat+tat[i];
            awt=awt+wt[i];
        }

        System.out.println("Average Turn Around Time ="+(atat/n));
        System.out.println("Average Waiting Time ="+(awt/n));
    }
```

Output:

```
Enter no of Processes
4
Enter the Burst Time for Process1
6
Enter the Burst Time for Process2
8
Enter the Burst Time for Process3
7
Enter the Burst Time for Process4
3
Process   BT        CT        TAT       WT
P4        3         3         3         0
P1        6         9         9         3
P3        7         16        16        9
P2        8         24        24        16
Average Turn Around Time =13.0
Average Waiting Time =7.0
```

Conclusion:

Hence, we have implemented Non-Premeptive Shortest Job First (SJF) Algorithm.