SHORTEST JOB FIRST(SJF) SCHELDULING.

AIM:

PROGRAM TO IMPLEMENT SHORTEST JOB FIRST SCHELDULING USING C LANGUAGE.

This is an approach which considers the next CPU burst. Each process posses its next CPU burst.

When CPU is available, the process having the smallest next CPU burst is allocated CPU.

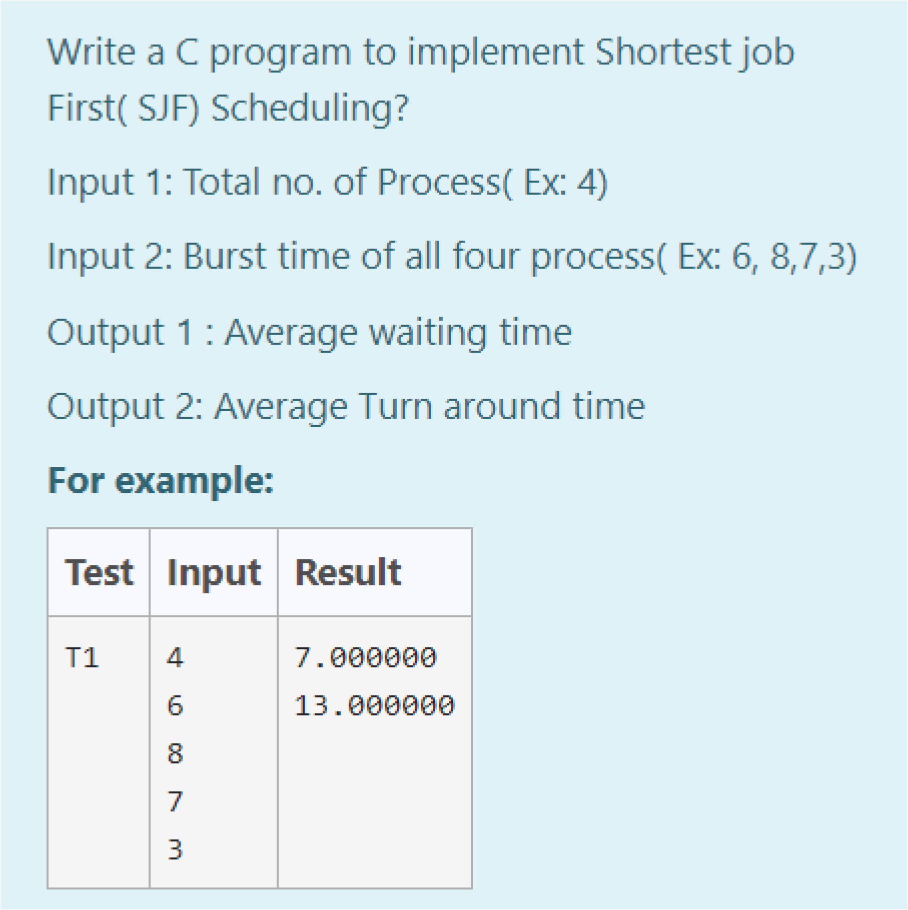
Now it may happen that two or more processes have the same next CPU burst. Then which process to allocate will be decided as per FCFS scheduling.

ALGORITHM:

1. Sort all the process according to the arrival time.

2. Then select that process which has minimum arrival time and minimum Burst time.

3. After completion of process make a pool of process which after till the completion of previous process and select that process among the pool which is having minimum Burst time.



Write a C program to implement Shortest job First(SJF) Scheduling?

Input 1: Total no. of Process(Ex: 4)

Input 2: Burst time of all four process(Ex: 6, 8,7,3)

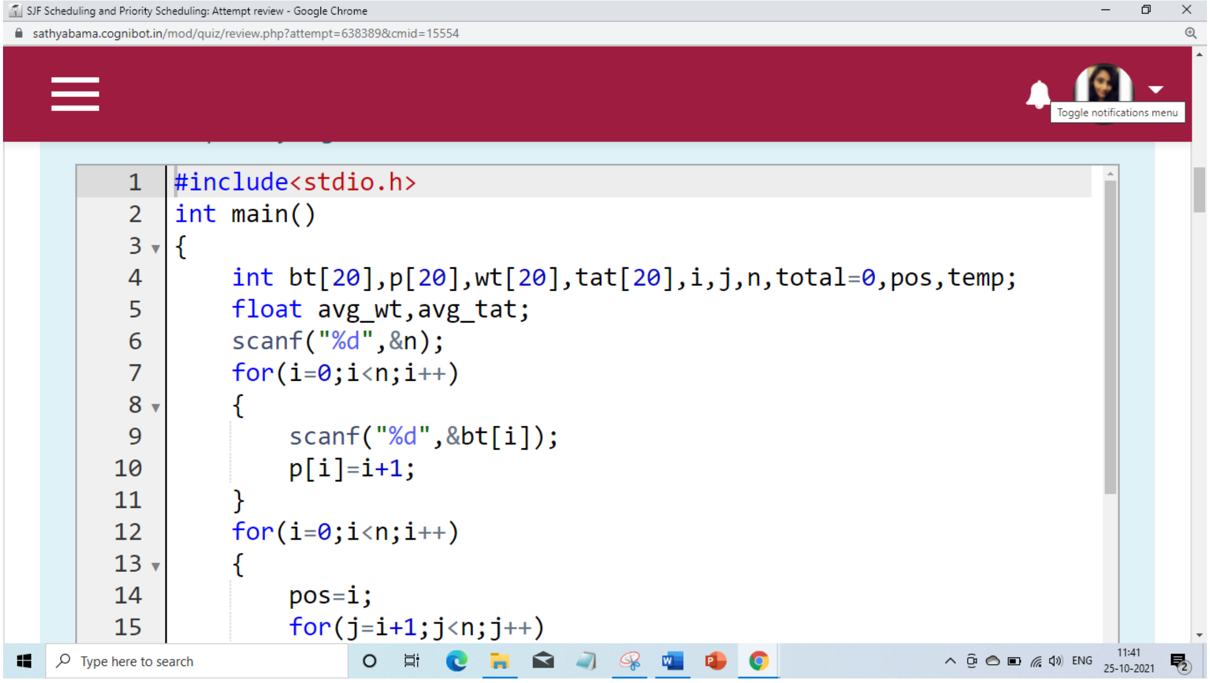
Output 1 : Average waiting time

Output 2: Average Turn around time

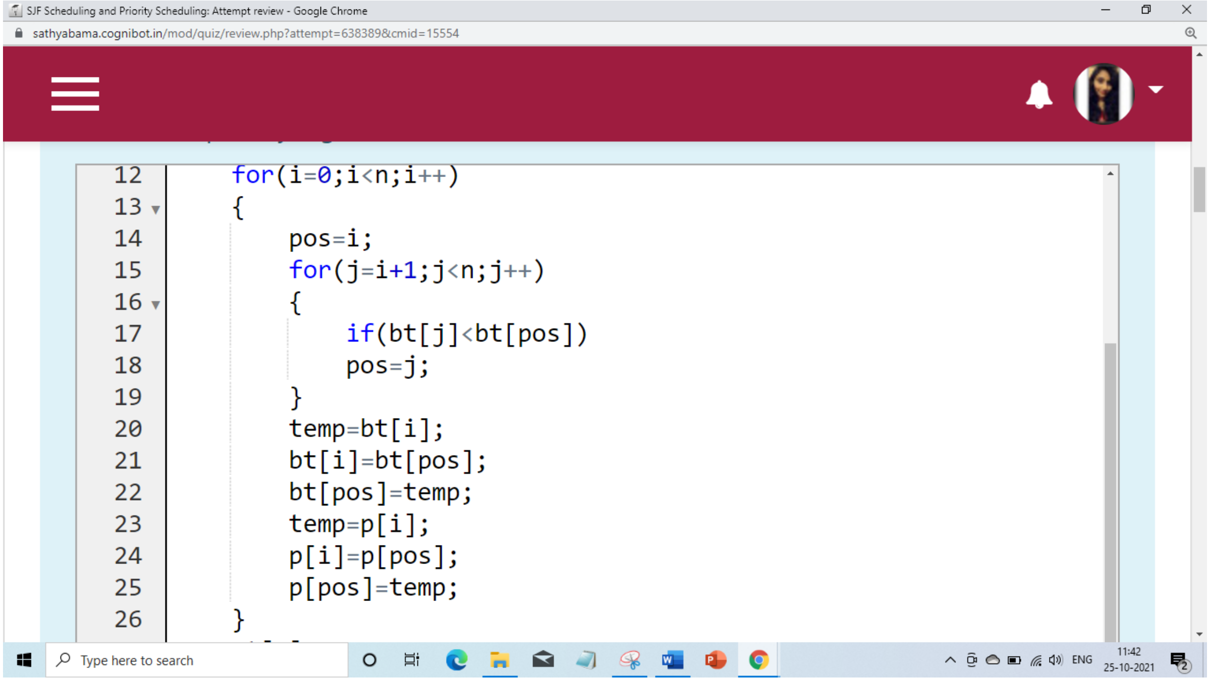
For example:

Test	Input	Result
T1	4 6 8 7 3	7.000000 13.000000

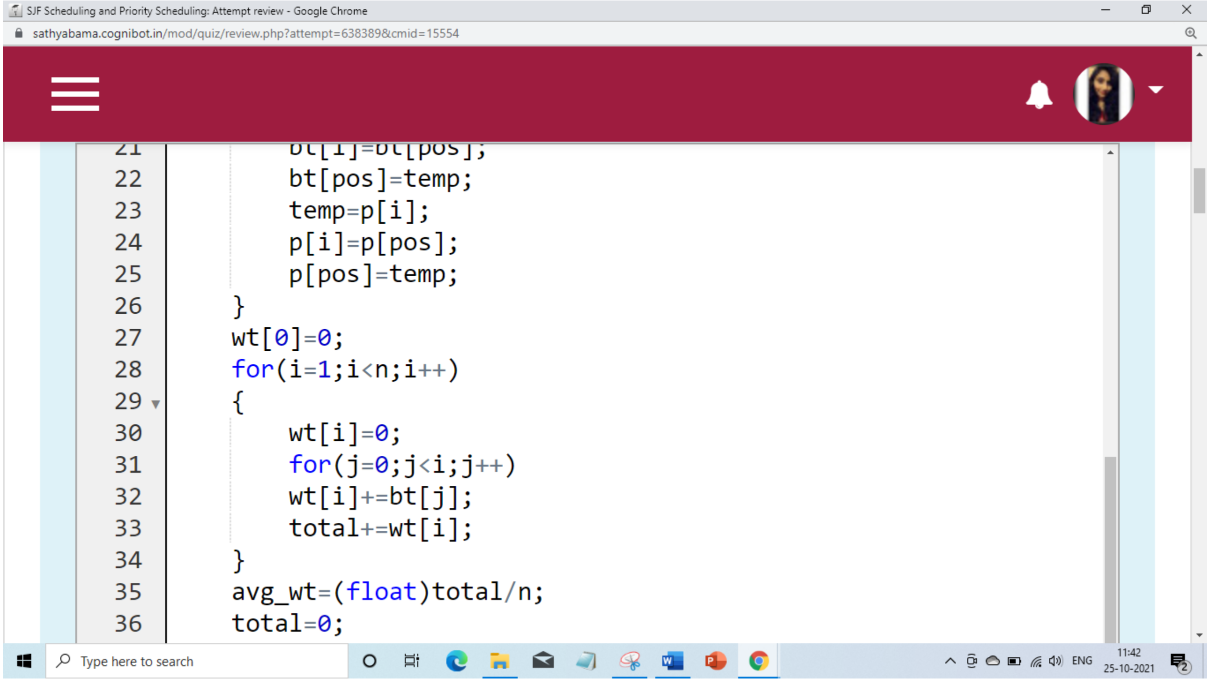
PROGRAM:


SJF Scheduling and Priority Scheduling: Attempt review - Google Chrome
sathyabama.cognibot.in/mod/quiz/review.php?attempt=638389&cmid=15554
Toggle notifications menu
Type here to search
ENG
11:41
25-10-2021

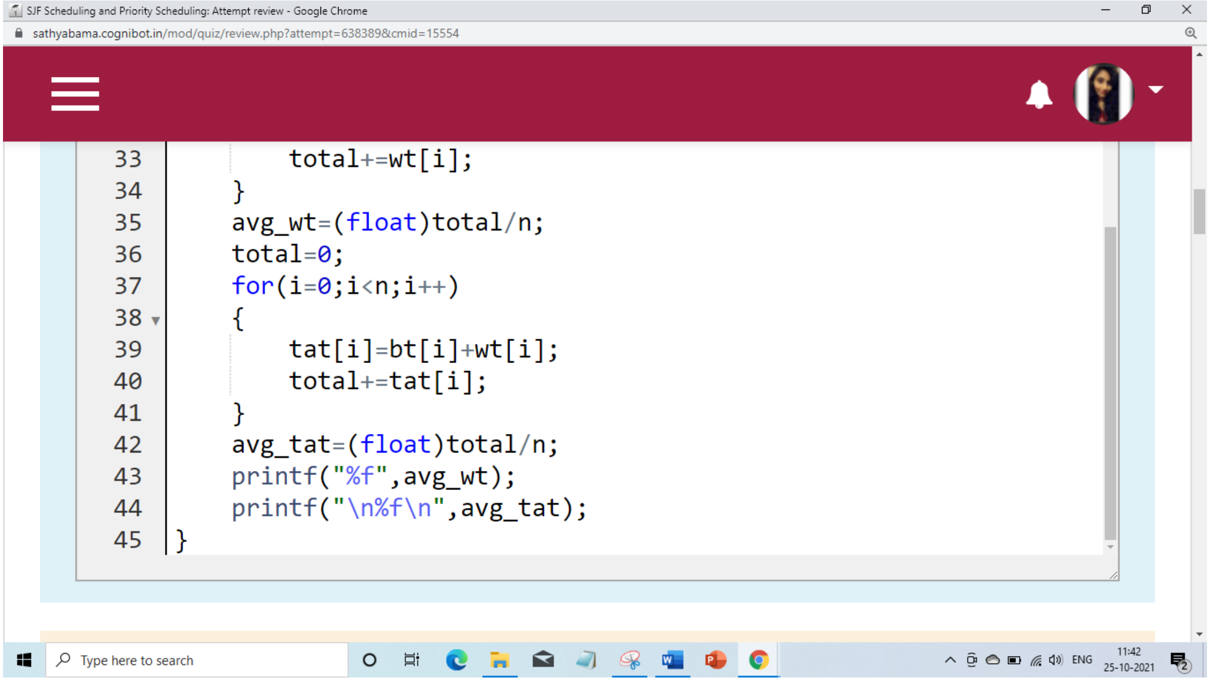
```
#include<stdio.h>
int main()
{
    int bt[20],p[20],wt[20],tat[20],i,j,n,total=0,pos,temp;
    float avg_wt,avg_tat;
    scanf("%d",&n);
    for(i=0;i<n;i++)
    {
        scanf("%d",&bt[i]);
        p[i]=i+1;
    }
    for(i=0;i<n;i++)
    {
        pos=i;
        for(j=i+1;j<n;j++)
```


SJF Scheduling and Priority Scheduling: Attempt review - Google Chrome
sathyabama.cognibot.in/mod/quiz/review.php?attempt=638389&cmid=15554
Type here to search
ENG
11:42
25-10-2021

```
    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;
    }
```

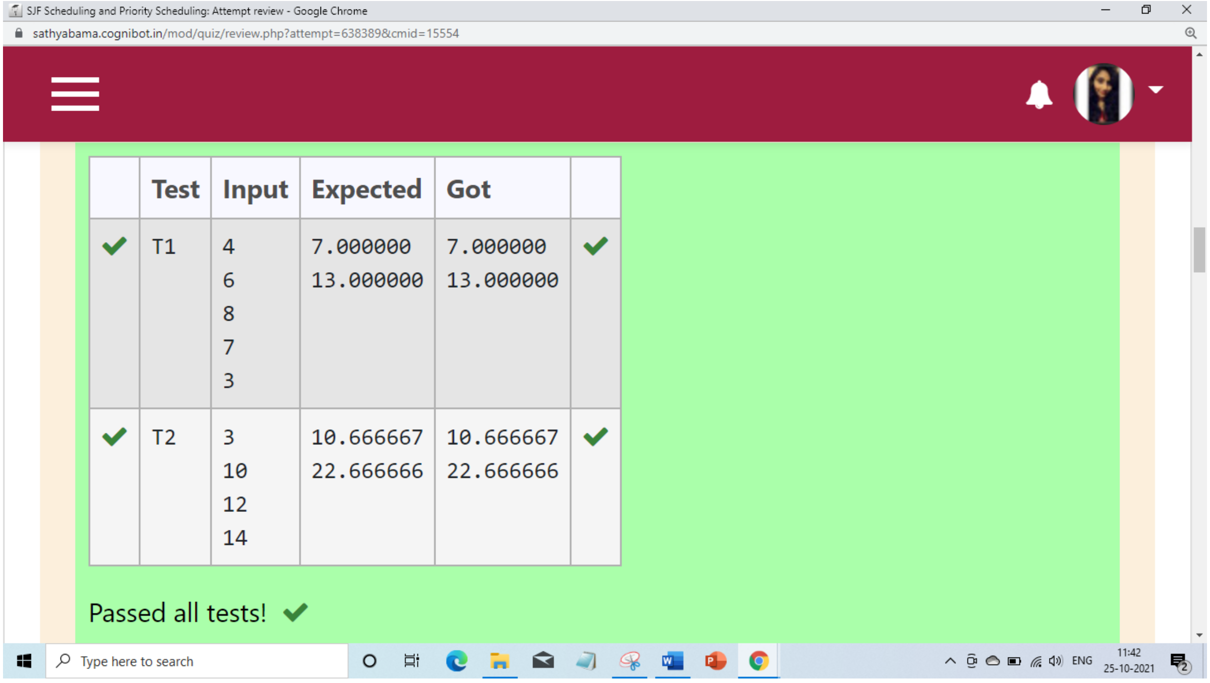

SJF Scheduling and Priority Scheduling: Attempt review - Google Chrome
sathyabama.cognibot.in/mod/quiz/review.php?attempt=638389&cmid=15554
Type here to search
ENG
11:42
25-10-2021

```
        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;
```


SJF Scheduling and Priority Scheduling: Attempt review - Google Chrome
sathyabama.cognibot.in/mod/quiz/review.php?attempt=638389&cmid=15554
Type here to search
ENG
11:42
25-10-2021

```
        total+=wt[i];
    }
    avg_wt=(float)total/n;
    total=0;
    for(i=0;i<n;i++)
    {
        tat[i]=bt[i]+wt[i];
        total+=tat[i];
    }
    avg_tat=(float)total/n;
    printf("%f",avg_wt);
    printf("\n%f\n",avg_tat);
}
```

RESULT:


SJF Scheduling and Priority Scheduling: Attempt review - Google Chrome
sathyabama.cognibot.in/mod/quiz/review.php?attempt=638389&cmid=15554
Type here to search
ENG
11:42
25-10-2021

	Test	Input	Expected	Got	
✓	T1	4 6 8 7 3	7.000000 13.000000	7.000000 13.000000	✓
✓	T2	3 10 12 14	10.666667 22.666666	10.666667 22.666666	✓

Passed all tests! ✓

RESULT:

SJF WAS SUCCSSFULLY IMPLEMENT USING C LANGUAGE.