PRIORITY SCHELDULING

AIM:

PROGRAM TO IMPLEMENT PRIORITY SCHEDULING USING C LANGUAGE.

In priority scheduling algorithm each process has a priority associated with it and as each process hits the queue, it is stored in based on its priority so that process with higher priority are dealt with first.

It should be noted that equal priority processes are scheduled in FCFS order.

ALGORITHM:

1- First input the processes with their burst time and priority.

2- Sort the processes, burst time and priority according to the priority.

3- Now simply apply FCFS algorithm.

Write a C Program to implement priority scheduling?

Input 1 : Total no.of Process (EX:5)

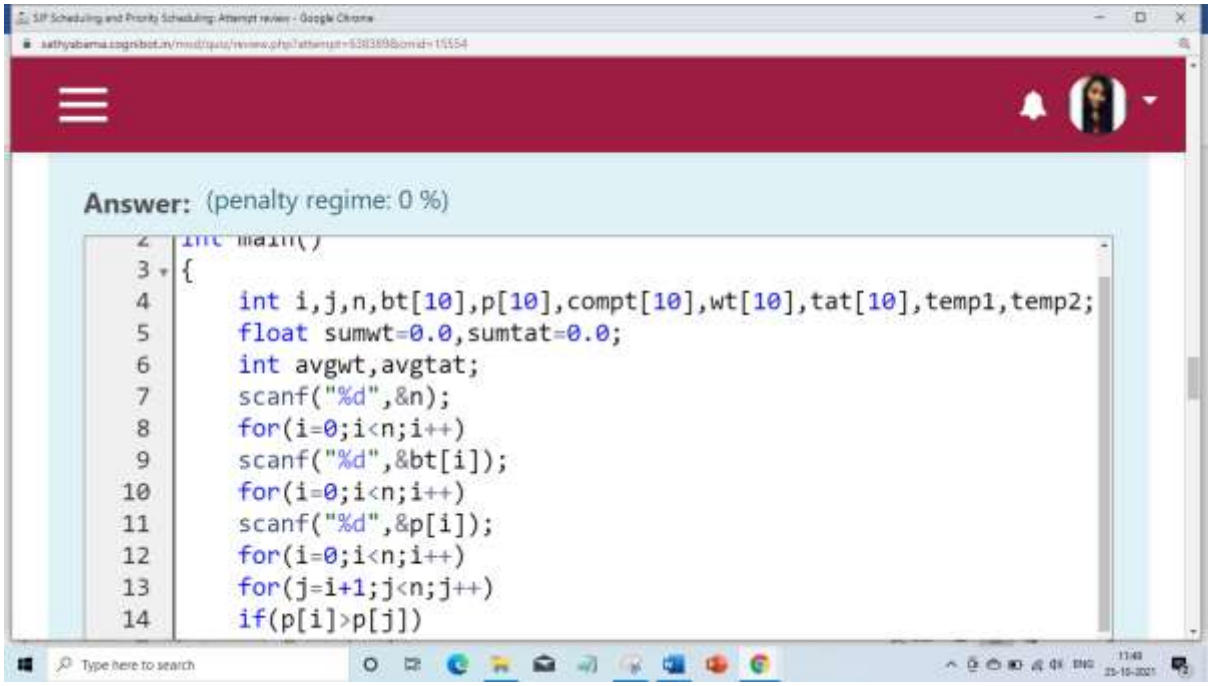
Input 2: Burst time of all five process (EX: 10,1,2,1,5)

Input 3: Priority of all process (Ex: 3,1,4,5,2)

For example:

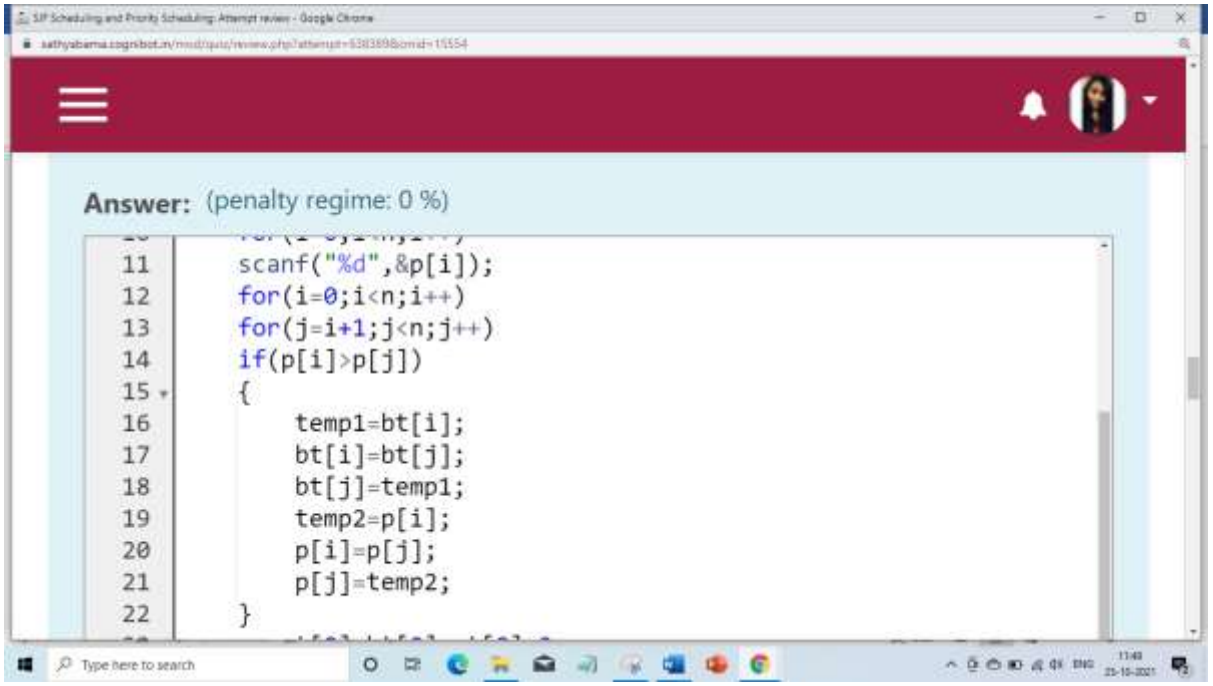
Test	Input	Result
T1	5 10 1 2 1 5 3 1 4 5 2	8 12

PROGRAM:



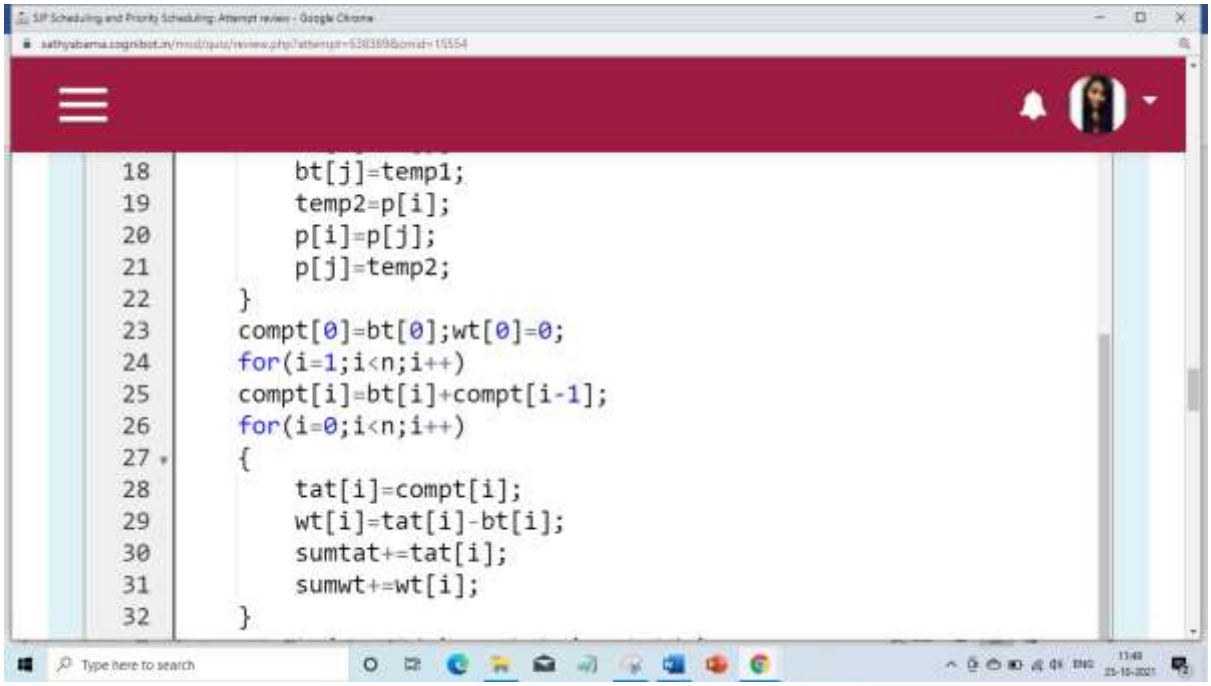
Answer: (penalty regime: 0 %)

```
int main()
{
    int i,j,n,bt[10],p[10],compt[10],wt[10],tat[10],temp1,temp2;
    float sumwt=0.0,sumtat=0.0;
    int avgwt,avgtat;
    scanf("%d",&n);
    for(i=0;i<n;i++)
    scanf("%d",&bt[i]);
    for(i=0;i<n;i++)
    scanf("%d",&p[i]);
    for(i=0;i<n;i++)
    for(j=i+1;j<n;j++)
    if(p[i]>p[j])
```

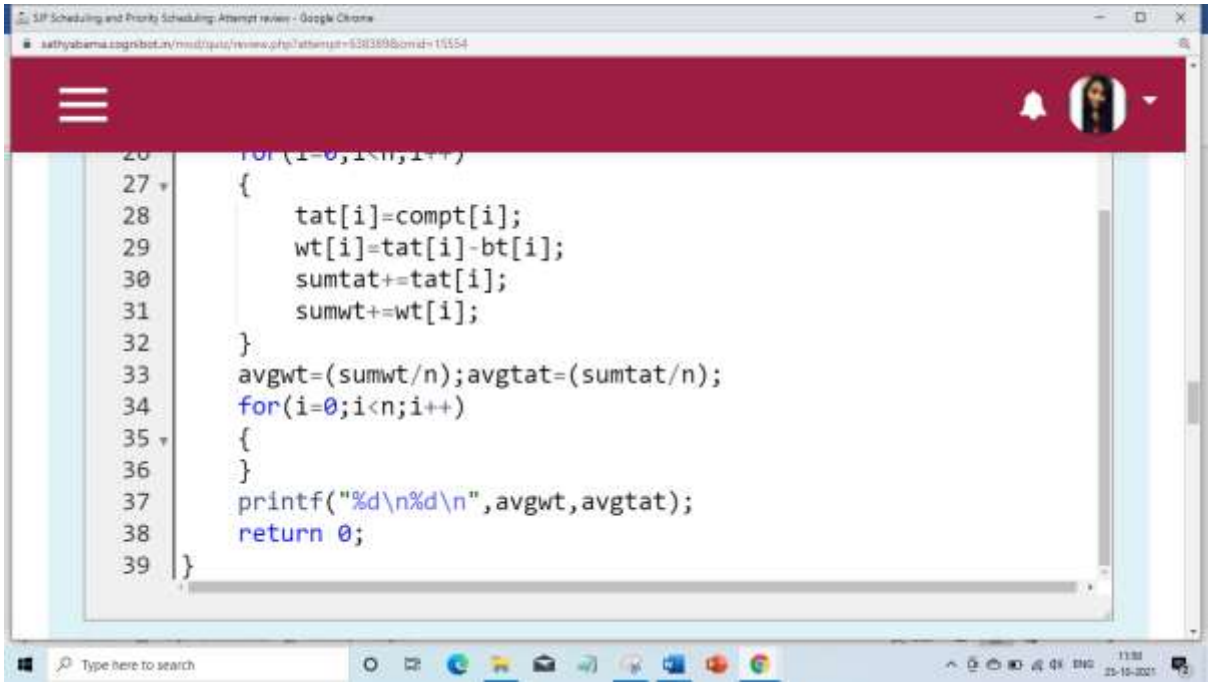


Answer: (penalty regime: 0 %)

```
    scanf("%d",&p[i]);
    for(i=0;i<n;i++)
    for(j=i+1;j<n;j++)
    if(p[i]>p[j])
    {
        temp1=bt[i];
        bt[i]=bt[j];
        bt[j]=temp1;
        temp2=p[i];
        p[i]=p[j];
        p[j]=temp2;
    }
```

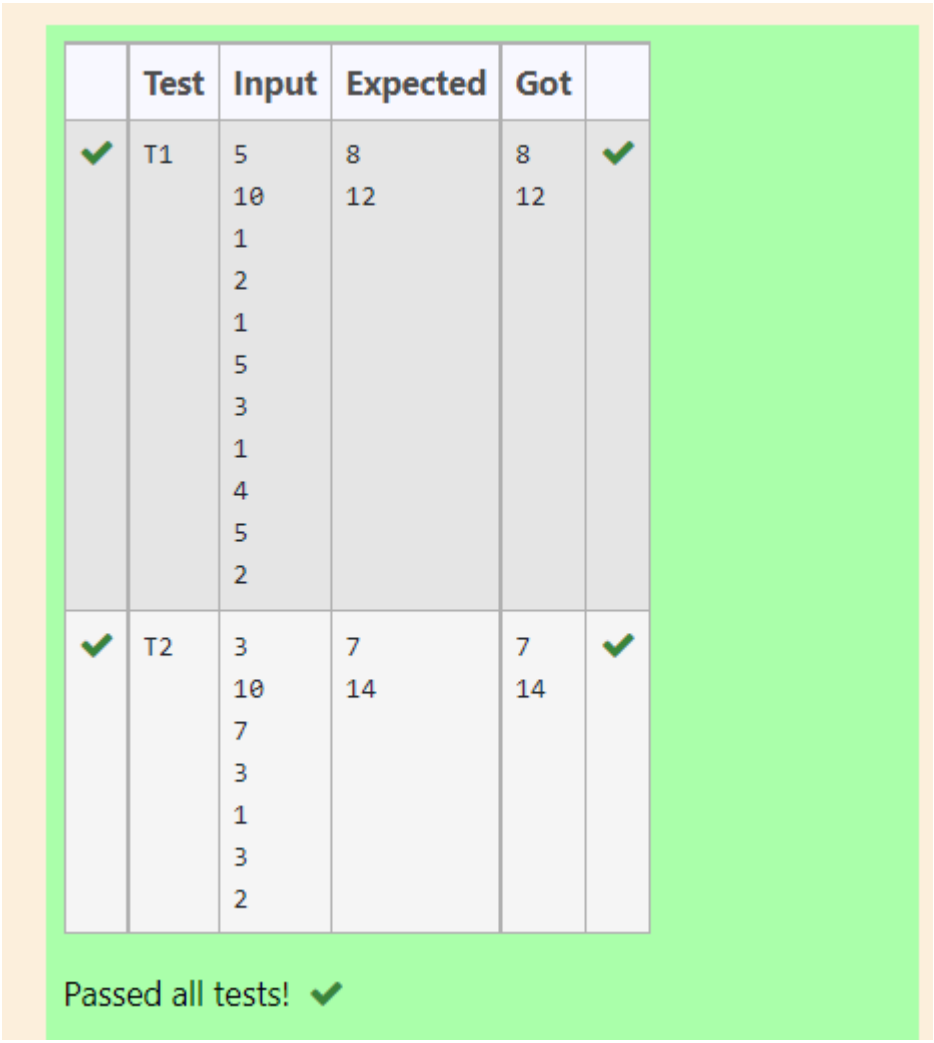


```
        bt[j]=temp1;
        temp2=p[i];
        p[i]=p[j];
        p[j]=temp2;
    }
    compt[0]=bt[0];wt[0]=0;
    for(i=1;i<n;i++)
    compt[i]=bt[i]+compt[i-1];
    for(i=0;i<n;i++)
    {
        tat[i]=compt[i];
        wt[i]=tat[i]-bt[i];
        sumtat+=tat[i];
        sumwt+=wt[i];
    }
```



```
    {
        tat[i]=compt[i];
        wt[i]=tat[i]-bt[i];
        sumtat+=tat[i];
        sumwt+=wt[i];
    }
    avgwt=(sumwt/n);avgtat=(sumtat/n);
    for(i=0;i<n;i++)
    {
    }
    printf("%d\n%d\n",avgwt,avgtat);
    return 0;
}
```

RESULT;



	Test	Input	Expected	Got	
✔	T1	5 10 1 2 1 5 3 1 4 5 2	8 12	8 12	✔
✔	T2	3 10 7 3 1 3 2	7 14	7 14	✔

Passed all tests! ✔

RESULT:

PRIORITY SCHELDULING WAS SUCCESSSFULLY IMPLEMENT USING C LANGUAGE.