جامعة المستقبل
AL MUSTAQBAL UNIVERSITY

كلية العلوم

قسم الانظمة الطبية الذكية

Intelligent Medical Systems Department

Lab (3)

CPU scheduling algorithms

Subject: **OPERATING SYSTEMS**

Class: Second

Lecturer: Dr.Ahmad almhanna

Eng .Jumana altahier

Programar :Ahyab hashim

Study Year: 2023-2024

Write a C prgram simulate the following CPU scheduling algorithms:

b) SJF

SJF CPU SCHEDULING ALGORITHM

For SJF scheduling algorithm, read the number of processes/jobs in the system, their CPU burst times. Arrange all the jobs in order with respect to their burst times. There may be two jobs in queue with the same execution time, and then FCFS approach is to be performed. Each process will be executed according to the length of its burst time. Then calculate the waiting time and turnaround time of each of the processes accordingly. HARDWARE REQUIREMENTS: Intel based Desktop Pc RAM of 512 MB SOFTWARE REQUIREMENTS: Turbo C/ Borland C.

THEORY:

Example of Non Preemptive SJF

Process	Arrival Time	Burst Time
P_1	0.0	7
P_2	2.0	4
P_3	4.0	1
P_4	3.0	4

P1	**P3**	**P2**	**P4**

0 7 8 12 16

Example of Preemptive SJF

Process	Arrival Time	Burst Time
P_1	0.0	7
P_2	2.0	4
P_3	4.0	1
P_4	3.0	4

P1	P2	P3	P2	P4	P1

Average waiting time = (9 + 1 + 0 +2)/4 = 3

ALGORITHM

1. Start

2. Declare the array size

3. Read the number of processes to be inserted

4. Read the Burst times of processes

5. sort the Burst times in ascending order and process with shortest burst time is first executed.

6. calculate the waiting time of each process wt[i+1]=bt[i]+wt[i]

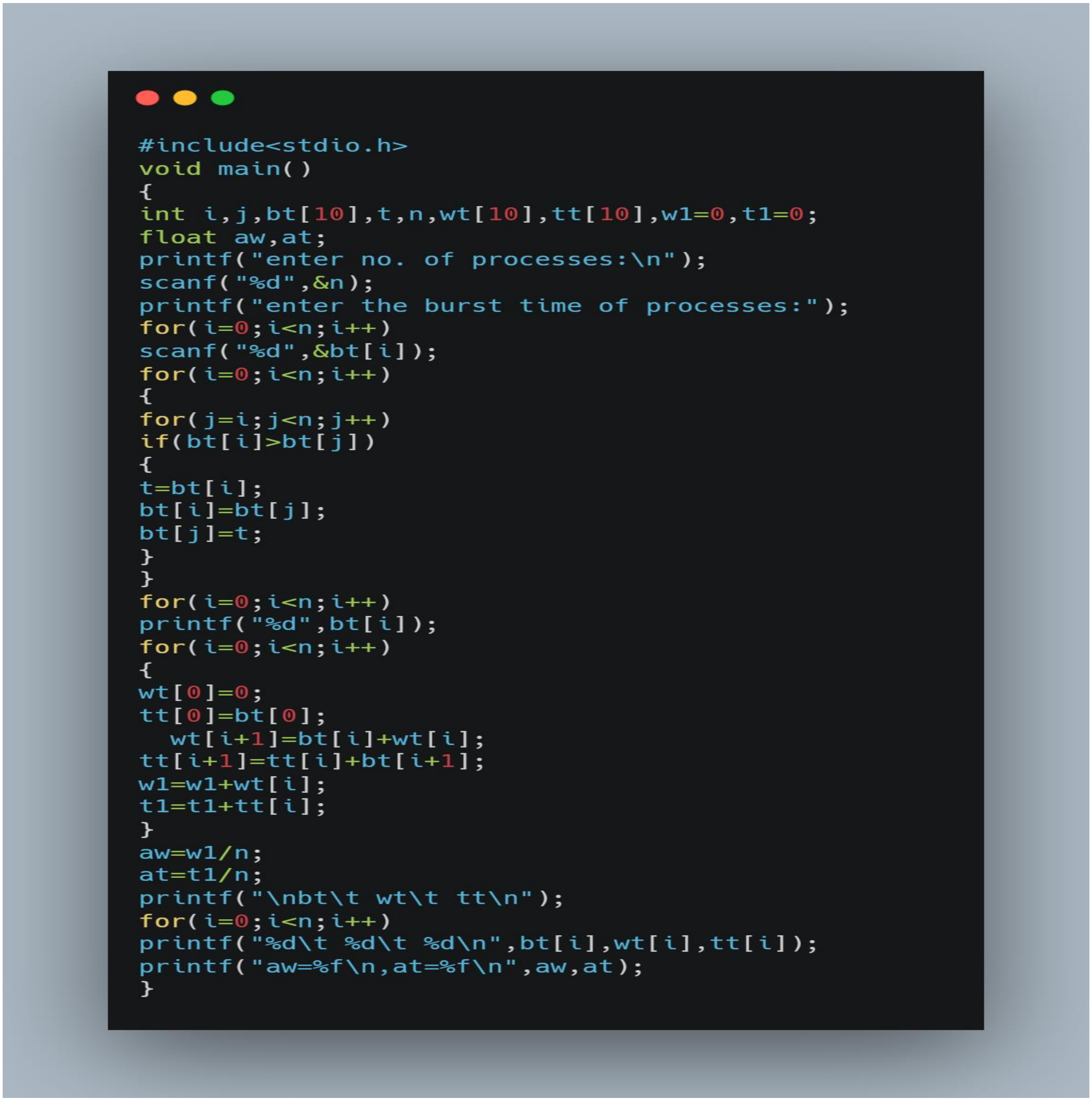
7. calculate the turnaround time of each process tt[i+1]=tt[i]+bt[i+1]

8. Calculate the average waiting time and average turnaround time.

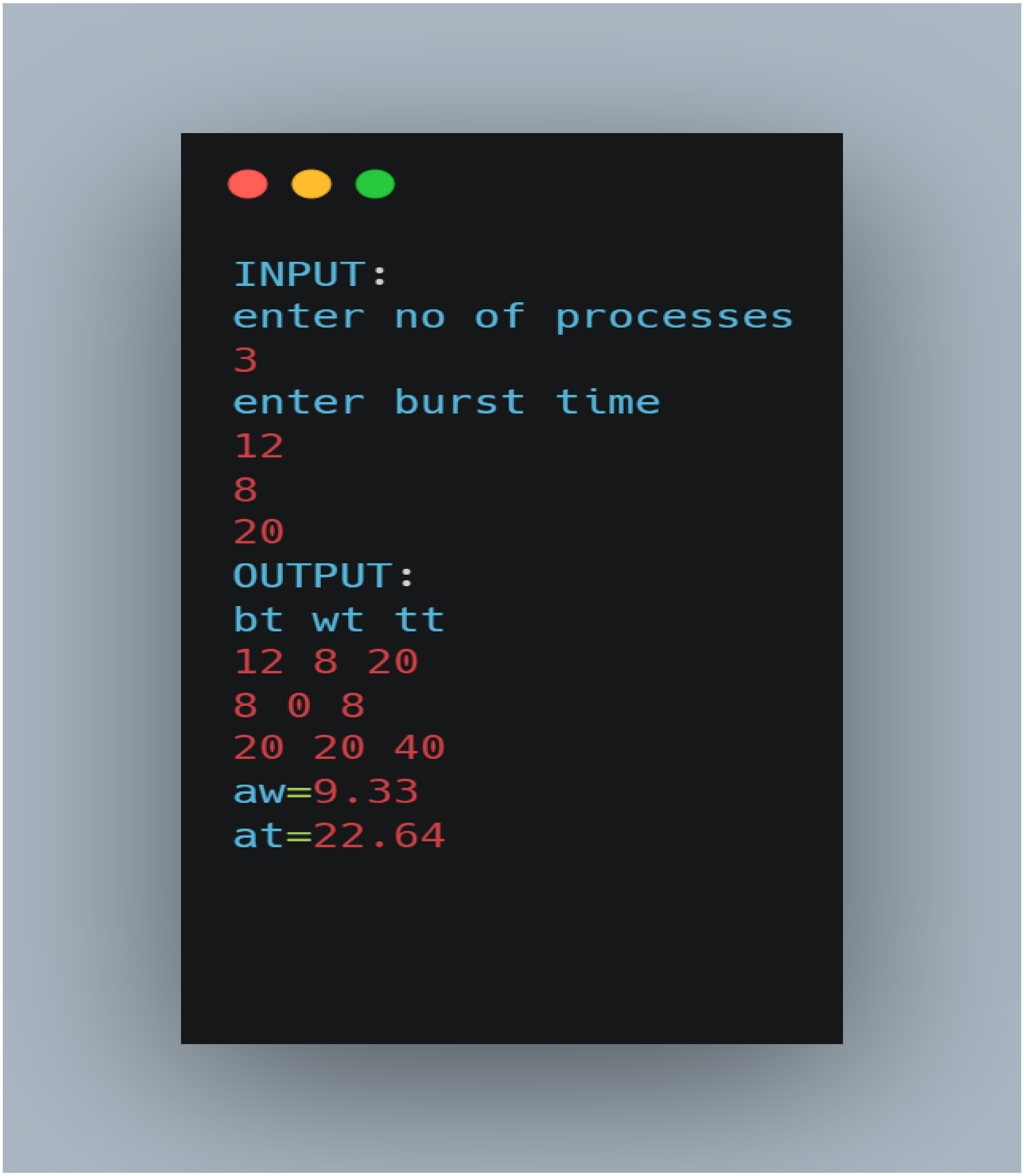
9. Display the values

10. Stop

PROGRAM:



```
#include<stdio.h>
void main()
{
int i,j,bt[10],t,n,wt[10],tt[10],w1=0,t1=0;
float aw,at;
printf("enter no. of processes:\n");
scanf("%d",&n);
printf("enter the burst time of processes:");
for(i=0;i<n;i++)
scanf("%d",&bt[i]);
for(i=0;i<n;i++)
{
for(j=i;j<n;j++)
if(bt[i]>bt[j])
{
t=bt[i];
bt[i]=bt[j];
bt[j]=t;
}
}
for(i=0;i<n;i++)
printf("%d",bt[i]);
for(i=0;i<n;i++)
{
wt[0]=0;
tt[0]=bt[0];
  wt[i+1]=bt[i]+wt[i];
tt[i+1]=tt[i]+bt[i+1];
w1=w1+wt[i];
t1=t1+tt[i];
}
aw=w1/n;
at=t1/n;
printf("\nbt\t wt\t tt\n");
for(i=0;i<n;i++)
printf("%d\t %d\t %d\n",bt[i],wt[i],tt[i]);
printf("aw=%f\n,at=%f\n",aw,at);
}
```



```
INPUT:
enter no of processes
3
enter burst time
12
8
20
OUTPUT:
bt wt tt
12 8 20
8 0 8
20 20 40
aw=9.33
at=22.64
```

THANK YOU!