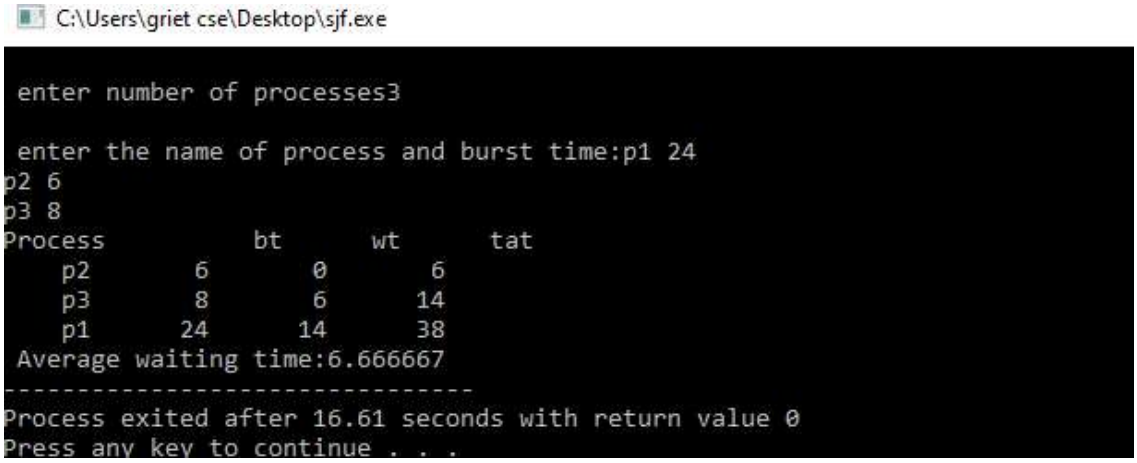
Task 3b: Simulate the SJF Scheduling algorithms using C program

Program:

```
#include<stdio.h>
struct sa
{
char pro[10];
int bt,wt,tat;
}p[10],temp[10];
void main()
{
int i,j,n,temp1=0;
float awt=0,atat=0;
printf("\n enter number of processes");
scanf("%d",&n);
printf("\n enter the name of process and burst time:");
for(i=0;i<n;i++)
{
scanf("%s %d",p[i].pro,&p[i].bt);
}
for(i=0;i<n;i++)
{
for(j=i+1;j<n;j++)
{
if(p[i].bt>p[j].bt)
{
temp[i]=p[i];
```

```
p[i]=p[j];
p[j]=temp[i];
}
}
}
for(i=0;i<n;i++)
{
p[i].wt=temp1;
p[i].tat=p[i].bt+p[i].wt;
temp1=p[i].bt+temp1;
}
for(i=0;i<n;i++)
{
awt=awt+p[i].wt;
}
awt=awt/n;
printf("Process \t bt \t wt \t tat");
for(i=0;i<n;i++)
{
printf("\n %5s \t %5d \t %5d \t %5d",p[i].pro,p[i].bt,p[i].wt,p[i].tat);
}
printf("\n Average waiting time:%f",awt);
}
```

Output:


C:\Users\griet cse\Desktop\sjf.exe
enter number of processes3
enter the name of process and burst time:p1 24
p2 6
p3 8
Process bt wt tat
p2 6 0 6
p3 8 6 14
p1 24 14 38
Average waiting time:6.666667
Process exited after 16.61 seconds with return value 0
Press any key to continue . . .

Task 3c: Simulate the Priority Scheduling algorithms using C program

Program:

```
#include<stdio.h>
struct sq
{
char pro[10];
int bt,wt,prior,tat;
}
P[10],temp;
main()
{
int i,j,n,temp1=0;
float awt=0,atat=0;
printf("Enter no. of processes\n");
scanf("%d",&n);
printf("enter name, burst time, priority\n");
for(i=0;i<n;i++)
{
scanf("%s%d%d",P[i].pro,&P[i].bt,&P[i].prior);
}
for(i=0;i<n;i++)
{
for(j=i+1;j<n;j++)
{
if(P[i].prior>P[j].prior)
{
temp=P[i];
P[i]=P[j];
P[j]=temp;
}
}
}
for(i=0;i<n;i++)
{
P[i].wt=temp1;
P[i].tat=P[i].wt+P[i].bt;
temp1+=P[i].bt;
}
for(i=0;i<n;i++)
{
awt+=P[i].wt;
atat+=P[i].tat;
}
printf("process\tbt\twt\ttat\n");
awt/=n;
atat/=n;
for(i=0;i<n;i++)
{
printf("%s\t%d\t%d\t%d\n",P[i].pro,P[i].bt,P[i].wt,P[i].tat);
```

```
}
printf("awt=%f\n,atat=%f\n",awt,atat);
}
```

Output: