ASSIGNMENT 3

First Come First Serve(FCFS)

Program:

```
class FCFS
{
public static void main(String a[])
{
String ar[]={"Procss "," Bus Time "," Waiting Time "," Tun Around Time"};
String p[]={"p1","p2","p3"};
int wt[]=new int[3];
int tat[]=new int[3];
int bt[]={24,3,4};
int i;
wt[0]=0;
for(i=1;i<3;i++)
{
wt[i]=wt[i-1]+bt[i-1];
}
for(i=0;i<3;i++)
{
tat[i]=wt[i]+bt[i];
}
System.out.print("");
System.out.println(" OUTPUT ");
System.out.print("");
System.out.println("Process BusTime waitingTime Turn Arount time");
int totaltat=0;
int totalwt=0;
for(i=0;i<3;i++)
{
System.out.println(p[i]+" "+bt[i]+" "+wt[i])+" "+tat[i];
}
for(i=0;i<3;i++)
{
```

totalwt=totalwt+wt[i];

}

totaltat=tat[0]+tat[1]+tat[2];

float avgwt=(float) totalwt/3;

float avgtat=(float) totaltat/3;

System.out.println("Total Tun around Time ="+totaltat);

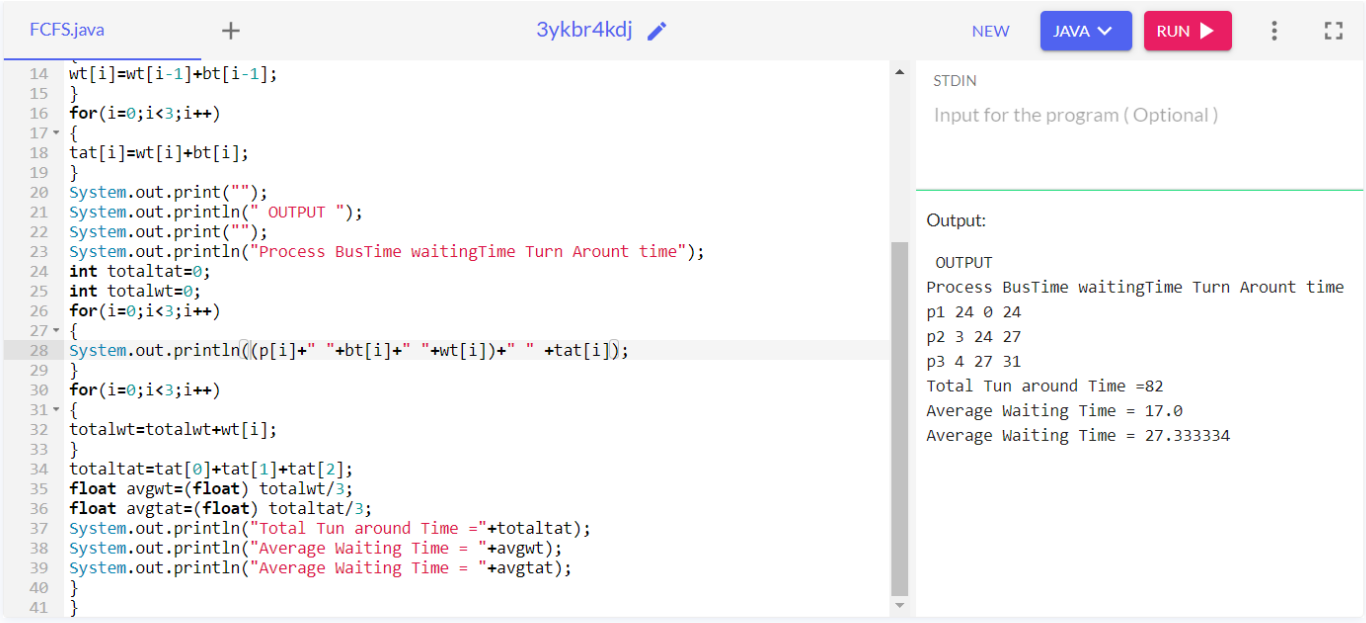
System.out.println("Average Waiting Time = "+avgwt);

System.out.println("Average Waiting Time = "+avgtat);

}

}

Output:



Shortest Job First(SJF)

Program:

public class SJF

{

public static void main(String[] args)

{

int n = 5;

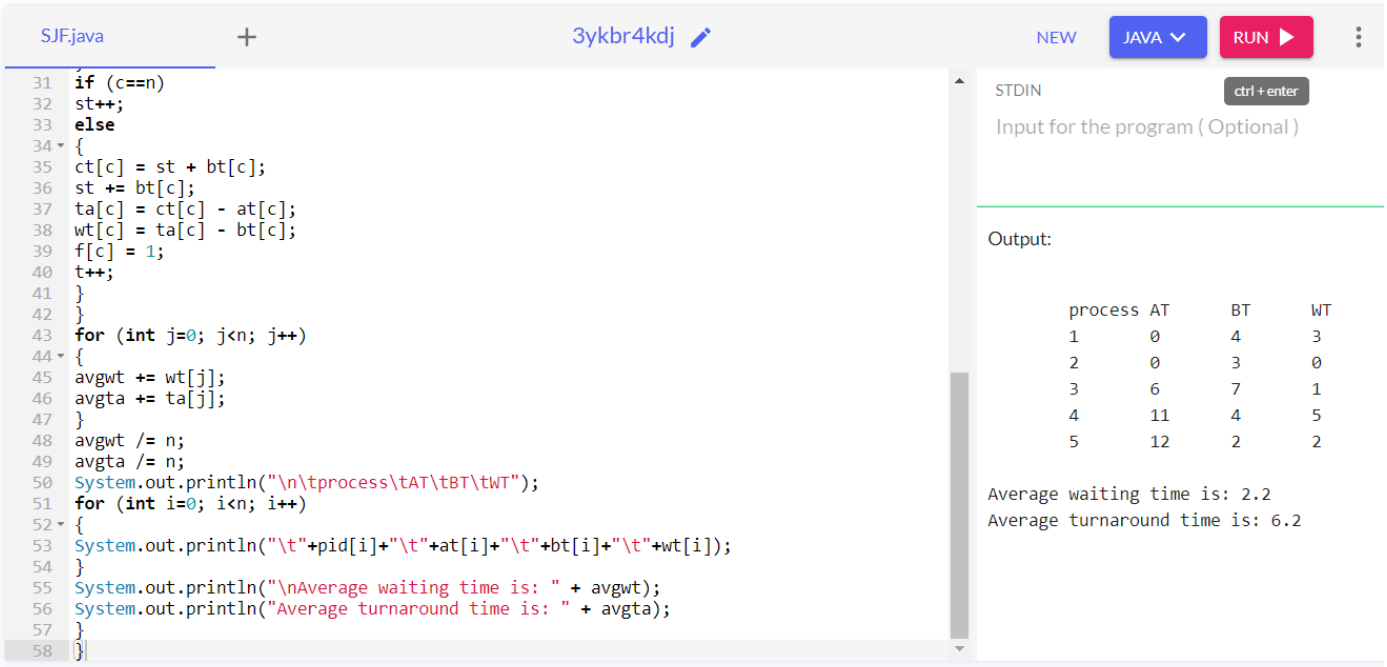
int t = 0;

int st = 0;

```
float avgwt = 0;
float avgta = 0;
int[] pid = {1, 2, 3, 4, 5};
int[] at = {0, 0, 6, 11, 12};
int[] bt = {4, 3, 7, 4, 2};
int[] f = new int[n];
int[] ct = new int[n];
int[] ta = new int[n];
int[] wt = new int[n];
for(int i=0; i<n; i++)
f[i] = 0;
while(true)
{
int c=n, min=999;
if(t==n) break;
for (int i=0; i<n; i++)
{
if ((at[i] <= st) && (f[i] == 0) && (bt[i]<min))
{
min = bt[i];
c = i;
}
}
if (c==n)
st++;
else
{
ct[c] = st + bt[c];
st += bt[c];
ta[c] = ct[c] - at[c];
wt[c] = ta[c] - bt[c];
f[c] = 1;
t++;
}
```

```
}
for (int j=0; j<n; j++)
{
avgwt += wt[j];
avgta += ta[j];
}
avgwt /= n;
avgta /= n;
System.out.println("\n\tprocess\tAT\tBT\tWT");
for (int i=0; i<n; i++)
{
System.out.println("\t"+pid[i]+"\t"+at[i]+"\t"+bt[i]+"\t"+wt[i]);
}
System.out.println("\nAverage waiting time is: " + avgwt);
System.out.println("Average turnaround time is: " + avgta);
}
}
```

Output:



Non-Preemptive Algorithm

Program:

```
class NPRMT
{
public static void main(String args[])
{
int st[]={0,4,7,16,14};
int bt[]={4,3,7,4,2};
int at[]={0,0,6,11,12};
int wt[]=new int[5];
int tat[]=new int[5];
for(int i=0;i<5;i++)
{
wt[i]=st[i]-at[i];
System.out.println("Waiting Time for Process "+(i+1)+" is : "+wt[i]);
}
int total_wt=wt[0]+wt[1]+wt[3]+wt[4];
float avg_wt=total_wt/5;
System.out.println("Total Waiting Time is : "+total_wt);
System.out.println("Average Waiting Time is : "+avg_wt);
for(int i=0;i<5;i++)
{
tat[i]=wt[i]+bt[i];
System.out.println("Turn Around Time for Process "+(i+1)+" is : "+tat[i]);
}
int total_tat=tat[0]+tat[1]+tat[2]+tat[3]+tat[4];
int avg_tat=total_tat/5;
System.out.println("Total Turn Around Time is : "+total_tat);
System.out.println("Average Turn Around Time is : "+avg_tat);
}
}
```

Output:

Java online compiler

Round Robin Scheduling(RRS)

Program:

```
class RRS
{
public static void main(String args[])
{
int bt[]={5,4,2,1};
int at[]={0,1,2,4};
int ct[]={12,11,6,9};
int wt[]=new int[5];
int tat[]=new int[5];
for(int i=0;i<4;i++)
{
tat[i]=ct[i]-at[i];
System.out.println("Turn Around Time for Process "+(i+1)+" is : "+tat[i]);
}
int total_tat=tat[0]+tat[1]+tat[2]+tat[3];
int avg_tat=total_tat/4;
System.out.println("Total Turn Around Time is : "+total_tat);
System.out.println("Average Turn Around Time is : "+avg_tat);
```

```
for(int i=0;i<4;i++)
{
wt[i]=tat[i]-bt[i];
System.out.println("Waiting Time for Process "+(i+1)+" is : "+wt[i]);
}
int total_wt=wt[0]+wt[1]+wt[2]+wt[3];
float avg_wt=total_wt/4;
System.out.println("Total Waiting Time is : "+total_wt);
System.out.println("Average Waiting Time is : "+avg_wt);
}
}
```

Output: