Name : Aakash A. Joshi

Roll no. : 0077

Subject : SPOS

Class : TE Computer

Batch: T4

Assignment no. 5.1(FCFS)

Code:

```
import java.util.*;
import java.io.*;
public class FCFS
{

   public static void main(String args[])
   {
      int n,sum=0;
      float total_tt=0,total_waiting=0;

       Scanner p=new Scanner(System.in);
       System.out.println("Enter Number Of Process");
       n=p.nextInt();
       int arrival[]=new int[n];
       int cpu[]=new int[n];
       int finish[]=new int[n];
       int turntt[]=new int[n];
       int wait[]=new int[n];
       int process[]=new int[n];

      // int pro[][]=new int[3][3];
       for(int i=0;i<n;i++)
       {
            System.out.println("Enter arrival time of "+(i+1)+" Process : ");
            arrival[i]=p.nextInt();//input arrival time
            System.out.println("Enter CPU time of "+(i+1)+" Process : ");
            cpu[i]=p.nextInt(); // input execution time

            process[i]=i+1;
      }

       for(int i=0;i<n;i++)
      {
           sum=sum+cpu[i];
           finish[i]=sum; //calculate finish time of each process
      }

      for(int i=0;i<n;i++)
      {
           turntt[i]=finish[i]-arrival[i]; // turnaround time = finish time - arrival time
```

```
            total_tt=total_tt+turntt[i];// calculate total turnaround time

            wait[i]=turntt[i]-cpu[i]; // waiting time = turnaround time - execution time

            total_waiting+=wait[i]; // calculate total waiting time
        }
System.out.println("\n\nProcessId\tArrivalTime\tBurstTime\tWaitingTime\tTurnAroundTime"
);
        for(int i=0;i<n;i++)
        {

System.out.println(process[i]+"\t\t"+arrival[i]+"\t\t"+cpu[i]+"\t\t"+wait[i]+"\t\t"+turntt[i]);
        }
        System.out.println("\n\n");
        System.out.println("Total turn around time is : "+(total_tt/n));
        System.out.println("Total waiting time is : "+(total_waiting/n));
    }
}
```

Output:

```
Enter Number Of Process
5
Enter arrival time of 1 Process :
0
Enter CPU time of 1 Process :
4
Enter arrival time of 2 Process :
1
Enter CPU time of 2 Process :
3
Enter arrival time of 3 Process :
2
Enter CPU time of 3 Process :
1
Enter arrival time of 4 Process :
3
Enter CPU time of 4 Process :
2
Enter arrival time of 5 Process :
4
Enter CPU time of 5 Process :
5

ProcessId     ArrivalTime     BurstTime     WaitingTime     TurnAroundTime
1             0               4             0               4
2             1               3             3               6
3             2               1             5               6
4             3               2             5               7
5             4               5             6               11

Total turn around time is : 6.8
```

Total waiting time is : 3.8

Name : Aakash A. Joshi

Roll no. : 0077

Subject : SPOS

Class : TE Computer

Batch: T4

Assignment no. 5.2(SJF)

Code:

```
import java.util.*;

public class SJF {
public static void main(String args[])
{
Scanner sc = new Scanner(System.in);
System.out.println ("Enter no of process:");
int n = sc.nextInt();
int pid[] = new int[n];
int at[] = new int[n]; // at means arrival time
int bt[] = new int[n]; // bt means burst time
int ct[] = new int[n]; // ct means complete time
int ta[] = new int[n]; // ta means turn around time
int wt[] = new int[n];  //wt means waiting time
int f[] = new int[n];  // f means it is flag it checks process is completed or not
int st=0, tot=0;
float avgwt=0, avgta=0;

for(int i=0;i<n;i++)
{
System.out.println ("Enter process " + (i+1) + " arrival time:");
at[i] = sc.nextInt();
System.out.println ("Enter process " + (i+1) + " brust time:");
bt[i] = sc.nextInt();
pid[i] = i+1;
f[i] = 0;
}
boolean a = true;
while(true)
{
int c=n, min=999;
if (tot == 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;
tot++;
}
}
System.out.println("\npid \t arrivaltime \t brusttime \t turnaroundtime \t waitingtime");
for(int i=0;i<n;i++)
{
avgwt+= wt[i];
avgta+= ta[i];
System.out.println(pid[i]+"\t\t"+at[i]+"\t\t\t"+bt[i]+"\t\t\t"+ta[i]+"\t\t\t\t"+wt[i]);
}
System.out.println ("\nAverage tat is "+ (float)(avgta/n));
System.out.println ("Average wt is "+ (float)(avgwt/n));
sc.close();
}
}
```

Output:

```
Enter no of process:
3
Enter process 1 arrival time:
0
Enter process 1 brust time:
3
Enter process 2 arrival time:
0
Enter process 2 brust time:
1
Enter process 3 arrival time:
0
Enter process 3 brust time:
2

pid     arrivaltime     brusttime       turnaroundtime          waitingtime
1               0               3               6                       3
2               0                       1               1                       0
3               0                       2               3                       1

Average tat is 3.3333333
Average wt is 1.3333334
```

Name : Aakash A. Joshi

Roll no. : 0077

Subject : SPOS

Class : TE Computer

Batch: T4

Assignment no. 5.3(Priority)

Code:

```
import java.util.ArrayList;
import java.util.Collections;
import java.util.Comparator;
import java.util.List;
import java.util.Scanner;

class Process {
    int pid;
    int burstTime;
    int priority;
    int waitingTime;
    int turnaroundTime;

    public Process(int pid, int burstTime, int priority) {
        this.pid = pid;
        this.burstTime = burstTime;
        this.priority = priority;
        this.waitingTime = 0;
        this.turnaroundTime = 0;
    }
}

public class priority {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        List<Process> processes = new ArrayList<>();

        System.out.print("Enter the number of processes: ");
        int numProcesses = scanner.nextInt();

        for (int i = 1; i <= numProcesses; i++) {
            System.out.print("Enter burst time for Process " + i + ": ");
            int burstTime = scanner.nextInt();

            System.out.print("Enter priority for Process " + i + ": ");
            int priority = scanner.nextInt();

            processes.add(new Process(i, burstTime, priority));
        }
```

```
        // Sort processes by priority
        Collections.sort(processes, Comparator.comparingInt(p -> p.priority));

        int currentTime = 0;

        for (Process process : processes) {
            process.waitingTime = currentTime;
            currentTime += process.burstTime;
            process.turnaroundTime = process.waitingTime + process.burstTime;
        }

        // Print the schedule
        System.out.println("Process\t\tBurst Time\t\tPriority\t\tWaiting Time\t\tTurnaround
 Time");
        for (Process process : processes) {
            System.out.println(process.pid + "\t\t" + process.burstTime + "\t\t\t" +
 process.priority + "\t\t\t"
                    + process.waitingTime + "\t\t\t" + process.turnaroundTime);
        }

        double avgWaitingTime = processes.stream().mapToDouble(p ->
 p.waitingTime).average().orElse(0);
        double avgTurnaroundTime = processes.stream().mapToDouble(p ->
 p.turnaroundTime).average().orElse(0);

        System.out.println("\nAverage Waiting Time: " + avgWaitingTime);
        System.out.println("Average Turnaround Time: " + avgTurnaroundTime);
    }
}
```

Output:

```
Enter the number of processes: 5
Enter burst time for Process 1: 10
Enter priority for Process 1: 3
Enter burst time for Process 2: 1
Enter priority for Process 2: 1
Enter burst time for Process 3: 2
Enter priority for Process 3: 4
Enter burst time for Process 4: 1
Enter priority for Process 4: 5
Enter burst time for Process 5: 5
Enter priority for Process 5: 2
```

Process	BurstTime	Priority	WaitingTime	TurnaroundTime
2	1	1	0	1
5	5	2	1	6
1	10	3	6	16
3	2	4	16	18
4	1	5	18	19

```
Average Waiting Time: 8.2
Average Turnaround Time: 12.0
```

Name : Aakash A. Joshi

Roll no. : 0077

Subject : SPOS

Class : TE Computer

Batch: T4

Assignment no. 5.4(Round Robbin)

Code:

```
import java.util.*;
import java.io.*;

public class robbin
{

   public static void main(String args[])
   {
      int n,sum=0;
      float total_tt=0,total_waiting=0;

       Scanner s=new Scanner(System.in);
       System.out.println("Enter Number Of Process");
       n=s.nextInt();
       int arrival[]=new int[n];
       int cpu[]=new int[n];
       int ncpu[]=new int[n];
       int pri[]=new int[n];
       int finish[]=new int[100];
       int turntt[]=new int[n];
       int wait[]=new int[n];
       int process[]=new int[n];
       int t_quantum,difference,temp_sum=0,k=0;
       int seq[]=new int[100];


       for(int i=0;i<n;i++)
       {
           System.out.println("Enter arrival time of "+(i+1)+" Process : ");
           arrival[i]=s.nextInt();//input arrival time
           System.out.println("Enter CPU time of "+(i+1)+" Process : ");
           ncpu[i]=cpu[i]=s.nextInt();// input execution time
           process[i]=i+1;
       }

      System.out.println("Enter time quantum : ");
      t_quantum = s.nextInt();// input time slice

       int tv=0;
       for(int i=0;i<n;i++)
```

```
    {
     temp_sum=temp_sum+cpu[i];//calculate total execution time
}
     while(sum!=temp_sum)
     {
          for(int i=0;i<n;i++)
          {
               if(ncpu[i]<t_quantum)
                    {
                         difference=ncpu[i];
                         tv=ncpu[i];
                         ncpu[i]=0;
                    }
               else
                    {
                         difference = ncpu[i]-t_quantum;
                         tv=t_quantum;
                         ncpu[i]=difference;//calculate remaining time for process
                    }
               if(tv > 0)
               {
              sum=sum+tv;
              finish[k]=sum;
              seq[k]=i;// calculate the sequence of process of execution
              k++;
              }
          }
    }
    System.out.println();
    for(int i=0;i<n;i++)
    {
     int carr=0,tt=0;
     carr=arrival[i];

     for(int j=0;j<k;j++)
     {
          if(seq[j]==i)
          {
               tt=tt+(finish[j]-carr);
               carr=finish[j];
          }
       }

       turntt[i]=tt;
       total_tt=total_tt+turntt[i];
       wait[i]=turntt[i]-cpu[i];
       total_waiting+=wait[i];
    }

System.out.println("\n\nProcessId\tArrivalTime\tBurstTime\tWaitingTime\tTurnAroundTime"
);
     for(int i=0;i<n;i++)
```

```
        {
          System.out.println(process[i]+"\t\t"+arrival[i]+"\t\t"+cpu[i]+"\t\t"+wait[i]+"\t\t"+turnttt[i]);
        }

        System.out.println("\n\n");
        System.out.println("Total turn around time is : "+(total_tt/n));
        System.out.println("Total waiting time is : "+(total_waiting/n));
    }
}
```

Output:

```
Enter Number Of Process
4
Enter arrival time of 1 Process :
1
Enter CPU time of 1 Process :
4
Enter arrival time of 2 Process :
2
Enter CPU time of 2 Process :
3
Enter arrival time of 3 Process :
3
Enter CPU time of 3 Process :
5
Enter arrival time of 4 Process :
4
Enter CPU time of 4 Process :
7
Enter time quantum :
2
```

ProcessId		ArrivalTime	BurstTime	WaitingTime	TurnAroundTime
1	1	4	5	9	
2	2	3	6	9	
3	3	5	8	13	
4	4	7	8	15	

```
Total turn around time is : 11.5
Total waiting time is : 6.75
```