**1)**

import java.util.\*;

public class SJF {

public static void main(String args[])

{

Scanner sc = new Scanner(System.in);

System.out.println ("Enter total number of process:");

int m = sc.nextInt();

int p\_id[] = new int[m];

int arrive\_t[] = new int[m];

int burs\_t[] = new int[m];

int ct[] = new int[m];

int tat[] = new int[m];

int wt[] = new int[m];

int flag[] = new int[m];

int start=0, total=0;

float avg\_wt=0, avg\_tat=0;

for(int i=0;i<m;i++)

{

System.out.println ("Enter process " + (i+1) + " arrival time:");

arrive\_t[i] = sc.nextInt();

System.out.println ("Enter process " + (i+1) + " brust time:");

burs\_t[i] = sc.nextInt();

p\_id[i] = i+1;

flag[i] = 0;

}

boolean x = true;

while(true)

{

int p=m, min=888;

if (total == m)

break;

for (int i=0; i<m; i++)

{

if ((arrive\_t[i] <= start) && (flag[i] == 0) && (burs\_t[i]<min))

{

min=burs\_t[i];

p=i;

}

}

if (p==m)

start++;

else

{

ct[p]=start+burs\_t[p];

start+=burs\_t[p];

tat[p]=ct[p]-arrive\_t[p];

wt[p]=tat[p]-burs\_t[p];

flag[p]=1;

total++;

}

}

System.out.println("\npid arrival brust complete turn waiting");

for(int i=0;i<m;i++)

{

avg\_wt+= wt[i];

avg\_tat+= tat[i];

System.out.println( p\_id[i] + "\t" + arrive\_t[i] + "\t" + burs\_t[i] + "\t" + ct[i] + "\t" + tat[i] + "\t" + wt[i] );

}

System.out.println ("\naverage tat is "+ (float)(avg\_tat/m));

System.out.println ("average wt is "+ (float)(avg\_wt/m));

sc.close();

}

}

**2)**

import java.util.\*;

public class Priority\_Scheduling {

public static void main(String args[])

{

Scanner sc = new Scanner(System.in);

System.out.println ("Enter number of total process:");

int n = sc.nextInt();

int process\_id[] = new int[n];

int arrival\_t[] = new int[n];

int burst\_t[] = new int[n];

int prior[]= new int[n];

int comp\_t[] = new int[n];

int t\_around[] = new int[n];

int wait\_t[] = new int[n];

int f[] = new int[n];

int start=0, total=0;

float avg\_wait\_t=0, avg\_t\_around=0;

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

{

System.out.println ("Enter process " + (i+1) + " priority:");

prior[i] = sc.nextInt();

System.out.println ("Enter process " + (i+1) + " arrival time:");

arrival\_t[i] = sc.nextInt();

System.out.println ("Enter process " + (i+1) + " brust time:");

burst\_t[i] = sc.nextInt();

process\_id[i] = i+1;

f[i] = 0;

}

boolean a = true;

while(true)

{

int p=n, min=888;

if (total == n)

break;

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

{

if ((arrival\_t[i] <= start) && (f[i] == 0) && (prior[i]<min))

{

min=prior[i];

p=i;

}

}

if (p==n)

start++;

else

{

comp\_t[p]=start+burst\_t[p];

start+=burst\_t[p];

t\_around[p]=comp\_t[p]-arrival\_t[p];

wait\_t[p]=t\_around[p]-burst\_t[p];

f[p]=1;

total++;

}

}

System.out.println("pid" +" " +"arrival" +" " + "brust" + " "+"priority" + " " + "complete" + " " + "turn" +" " + "waiting");

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

{

avg\_wait\_t+= wait\_t[i];

avg\_t\_around+= t\_around[i];

System.out.println( process\_id[i] + "\t" + arrival\_t[i] + "\t" + burst\_t[i] +"\t"+ prior[i]+ "\t" + comp\_t[i] + "\t" + t\_around[i] + "\t" + wait\_t[i] );

}

System.out.println ("\naverage tat is "+ (float)(avg\_t\_around/n));

System.out.println ("average wt is "+ (float)(avg\_wait\_t/n));

sc.close();

}

}

**3)**

import java.util.\*;

public class Robin\_round{

public static void WaitingTime(int process[],int wait\_t[],int m ,int burs\_t[],int quantum,int comp\_t[],int arrive\_t[]){

int rt[] = new int[m];

for(int i=0;i<wait\_t.length;i++){

rt[i]= burs\_t[i];

}

int t=0;

int arrival=0;

while(true){

boolean done = true;

for(int i=0;i<m;i++){

if(rt[i]>0){

done =false;

if(rt[i]>quantum && arrive\_t[i]<=arrival){

t +=quantum;

rt[i]-=quantum;

arrival++;

}

else{

if(arrive\_t[i]<=arrival){

arrival++;

t+=rt[i];

rt[i]=0;

comp\_t[i]=t; }

}

}

}

if(done==true)

{

break;

}

}

}

public static void TurnAroundTime(int process[] ,int wt[],int m,int bt[],int ta[],int ct[],int at[]){

for(int i=0;i<m;i++){

ta[i]= ct[i]-at[i];

wt[i] = ta[i]-bt[i];

}

}

public static void AvgTime(int process[],int m,int bt[],int quantum,int at[]){

int wt[] = new int[m];

int ta[] = new int[m];

int ct[] = new int[m];

WaitingTime(process,wt,m,bt,quantum,ct,at);

TurnAroundTime(process,wt,m,bt,ta,ct,at);

int total\_wt = 0, total\_ta= 0;

System.out.println("Processes " +" Arrival Time\t"+ " Burst time " +" completion time"+

" Turn Around Time " + " Waiting time");

for (int i=0; i<m; i++)

{

total\_wt = total\_wt + wt[i];

total\_ta = total\_ta + ta[i];

System.out.println(" " + (i+1) + "\t\t"+ at[i]+"\t\t"+ + bt[i] +"\t " +ct[i]+"\t\t"

+ta[i] +"\t\t " + wt[i]);

}

System.out.println("Average waiting time = " +

(float)total\_wt / (float)m);

System.out.println("Average turn around time = " +

(float)total\_ta / (float)m);

}

public static void main(String []agrs){

Scanner scan = new Scanner(System.in);

int quan\_time = 4;

int arrival\_time[] = new int[]{0,0,0};

int processes[] = new int[]{1,2,3};

int burs\_time[] = new int[]{24,3,3};

int m = processes.length;

AvgTime(processes,m,burs\_time,quan\_time,arrival\_time);

scan.close();

}

}