FCFS CODE:-

import java.util.\*;

public class FCFS {

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]; // process ids

int ar[] = new int[n]; // arrival times

int bt[] = new int[n]; // burst or execution times

int ct[] = new int[n]; // completion times

int ta[] = new int[n]; // turn around times

int wt[] = new int[n]; // waiting times

int temp;

float avgwt=0,avgta=0;

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

{

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

ar[i] = sc.nextInt();

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

bt[i] = sc.nextInt();

pid[i] = i+1;

}

//sorting according to arrival times

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

{

for(int j=0; j < n-(i+1) ; j++)

{

if( ar[j] > ar[j+1] )

{

temp = ar[j];

ar[j] = ar[j+1];

ar[j+1] = temp;

temp = bt[j];

bt[j] = bt[j+1];

bt[j+1] = temp;

temp = pid[j];

pid[j] = pid[j+1];

pid[j+1] = temp;

}

}

}

// finding completion times

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

{

if( i == 0)

{

ct[i] = ar[i] + bt[i];

}

else

{

if( ar[i] > ct[i-1])

{

ct[i] = ar[i] + bt[i];

}

else

ct[i] = ct[i-1] + bt[i];

}

ta[i] = ct[i] - ar[i] ; // turnaround time= completion time- arrival time

wt[i] = ta[i] - bt[i] ; // waiting time= turnaround time- burst time

avgwt += wt[i] ; // total waiting time

avgta += ta[i] ; // total turnaround time

}

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

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

{

System.out.println(pid[i] + " \t " + ar[i] + "\t" + bt[i] + "\t" + ct[i] + "\t" + ta[i] + "\t" + wt[i] ) ;

}

sc.close();

System.out.println("\naverage waiting time: "+ (avgwt/n)); // printing average waiting time.

System.out.println("average turnaround time:"+(avgta/n)); // printing average turnaround time.

}

}

SJF CODE:-

import java.util.Scanner;

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]; // process ids

int ar[] = new int[n]; // arrival times

int bt[] = new int[n]; // burst or execution times

int ct[] = new int[n]; // completion times

int ta[] = new int[n]; // turn around times

int wt[] = new int[n]; // waiting times

int f[] = new int[n];

int k[] = new int[n];

int temp, tot=0,st=0;

float avgwt=0,avgta=0;

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

{

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

ar[i] = sc.nextInt();

System.out.println("enter process " + (i+1) + " burst time: ");

bt[i] = sc.nextInt();

k[i] = bt[i];

pid[i] = i+1;

}

while(true){

int min = 99 , c =n;

if(tot == n)

break;

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

if(ar[i]<=st && f[i] == 0 && bt[i]<min ){

min = bt[i];

c = i;

}

}

if (c == n) st++;

else{

bt[c]--;

st++;

if(bt[c] == 0){

ct[c] =st;

f[c]=1;

tot++;

}

}

}

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

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

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

avgwt += wt[i];

avgta += ta[i];

}

System.out.println("pid arrival burst complete turn waiting");

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

{

System.out.println(pid[i] +"\t"+ ar[i]+"\t"+ k[i] +"\t"+ ct[i] +"\t"+ ta[i] +"\t"+ wt[i]);

}

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

System.out.println("average wt is "+ (float)(avgwt/n));

sc.close();

}

}