#include<stdio.h>

#include<stdlib.h>

int no;

void roundrobin(int,int,int[],int[]);

void srtf();

main()

{

int n,tq,choice;

int bt[10], st[10], i,j,k;

for(;;)

{

printf("Enter choice\n");

printf("1.Round Robin \n2.str \n3.Exit\n");

scanf("%d",&choice);

switch(choice)

{

case 1: printf("Round robin scheduling algorithm\n");

printf("Enter the number of process\n");

scanf("%d",&n);

printf("Enter burst time for sequences\n");

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

{

scanf("%d",&bt[i]);

st[i]=bt[i];

}

printf("Enter time quantum\n");

scanf("%d",&tq);

roundrobin(n,tq,st,bt);

break;

case 2: printf("\n\n-------------------Shortest remaining

time------");

srtf();

break;

case 3: exit(0);

break;

}

}

}

void roundrobin(int n, int tq, int st[], int bt[])

{

int time=0;

int tat[10],wt[10],i,count=0,swt=0,stat=0,temp1,sq=0,j,k;

float awt=0.0, atat=0.0;

while(1)

{

for(i=0,count=0;i<n;i++)

{

temp1=tq;

if(st[i]==0)

{

count++;

continue;

}

if (st[i]>tq)

st[i]=st[i]-tq;

else if(st[i]>=0)

{

temp1=st[i];

st[i]=0;

}

sq=sq+temp1;

tat[i]=sq;

}

if(n==count)

break;

}

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

{

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

swt=swt+wt[i];

stat=stat+tat[i];

}

awt=(float)swt/n;

atat=(float)stat/n;

printf("process no burst time waiting time turnaround time\n");

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

printf("%d \t\t %d \t\t %d \t\t %d \t\t\n",i+1,bt[i],wt[i],tat[i]);

printf("average waiting time is%f\n avg turnaround time is

%f\n",awt,atat);

}

void srtf()

{

int

n,j=0,st[10],bt[10],rt[10],remain=0,smallest,time=0,i,endtime,swt=0,stat=0;

printf("enter the no. of process:");

scanf("%d",&n);

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

{

printf("enter the arrivaltime for p[%d]:",i+1);

scanf("%d",&st[i]);

printf("enter the burst time for p[%d]:",i+1);

scanf("%d",&bt[i]);

rt[i]=bt[i];

}

rt[100]=999;

printf("process\t|writing time\t|turnarround time\n");

for(time=0;remain!=n;time++)

{

smallest=100;

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

{

if(st[i]<=time && rt[i]<rt[smallest] && rt[i]>0)

{

smallest=i;

}

}

rt[smallest]--;

if(rt[smallest]==0)

{

remain++;

endtime=time+1;

j=smallest;

printf("p[%d]\t|\t%d\t|\t%d\n",smallest+1,endtime-bt[j]-st[j],endtime-st[j]);

swt +=endtime-bt[j]-st[j];

stat +=endtime-st[j];

}

}

float awt=0.0,atat=0.0;

awt=(float)swt/n;

atat=(float)stat/n;

printf("average waiting time:%f\n",awt);

printf("average turnarround time:%f\n",atat);

}