

ĐỀ THI CHÍNH THỨC

Môn thi: TIN HỌC  
Buổi thi thứ hai: Chiều 12/01/2020

**HƯỚNG DẪN CHẤM THI**  
(Bản hướng dẫn này gồm 07 trang)

**I. Hướng dẫn chung**

- 1) Nếu thí sinh làm bài không theo cách nêu trong đáp án nhưng đúng thì Tổ chấm thi trao đổi, thống nhất thang điểm và vẫn cho đủ số điểm từng phần như hướng dẫn quy định;
- 2) Việc chi tiết hoá (nếu có) thang điểm trong hướng dẫn chấm phải đảm bảo không làm sai lệch hướng dẫn chấm và phải được thống nhất thực hiện trong Tổ chấm thi.

**II. Đáp án và thang điểm**

BÀI	ĐÁP ÁN		ĐIỂM
	<b>Dãy số</b> Lời giải: Duyệt nâng cao bằng cách chuẩn bị trước dữ liệu - Chuẩn bị trước mảng Sum[i] là tổng của dãy $a_1, \dots, a_i$ . - Tìm đoạn i,j dài nhất sao cho $\text{Sum}[j] - \text{Sum}[i-1] = K$ - Độ phức tạp $O(n^2)$		<b>7,0</b>
<b>Test</b>	DAYSO.INP	DAYSO.OUT	
1	10 5 1 2 2 3 2 1 1 1 1 2	4	Mỗi Test 0.5 điểm
2	5 3 1 4 4 5 6	- 1	
3	10 552 -167 -442 -389 552 -253 -617 122 373 -539 -943	1	
4	10 20 2 2 2 2 2 2 2 2 2 2	10	
5	File dayso05.inp	File dayso05.out	
6	File dayso06.inp	File dayso06.out	
7	File dayso07.inp	File dayso07.out	
8	File dayso08.inp	File dayso08.out	
9	File dayso09.inp	File dayso09.out	
10	File dayso10.inp	File dayso10.out	
11	File dayso11.inp	File dayso11.out	
12	File dayso12.inp	File dayso12.out	
13	File dayso13.inp	File dayso13.out	
14	File dayso14.inp	File dayso14.out	
<b>Bài 6</b>	<b>Vườn cây</b> Lời giải: Xử lý trên mảng 2 chiều		<b>7,0</b>
<b>Test</b>	VUONCAY.INP	VUONCAY.OUT	
1	4 4	10	Mỗi

	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 3 0 2 2 7 1 1 1 2 1 2 2 3	15	Test 0.5 điểm
2	4 3 1 1 1 1 1 1 1 1 1 1 1 1 4 0 2 2 2 0 1 1 2 0 3 3 2 1 1 1 3	12	
3	4 5 1 1 1 1 1 2 2 2 2 2 3 3 3 3 3 4 4 4 4 4 4 0 1 5 2 1 1 4 2 0 3 3 4 1 2 3 3	7 28	
4	5 5 1 1 1 1 1 2 2 2 2 2 3 3 3 3 3 4 4 4 4 4 5 5 5 5 5 6 0 2 2 4 1 2 2 4 0 3 3 4 1 1 1 5 0 5 5 20 1 5 5 1	58 78 20	
5	File vuoncay05.inp	File vuoncay05.out	
6	File vuoncay06.inp	File vuoncay06.out	
7	File vuoncay07.inp	File vuoncay07.out	
8	File vuoncay08.inp	File vuoncay08.out	
9	File vuoncay09.inp	File vuoncay09.out	
10	File vuoncay10.inp	File vuoncay10.out	
11	File vuoncay11.inp	File vuoncay11.out	
12	File vuoncay12.inp	File vuoncay12.out	
13	File vuoncay13.inp	File vuoncay13.out	
14	File vuoncay14.inp	File vuoncay14.out	

<b>Bài 7</b>	<b>Du lịch</b> Lời giải: Dijkstra <ul style="list-style-type: none"> <li>- Tìm đường đi ngắn nhất theo chi phí di chuyển. Giả sử đường đi ngắn nhất là <math>S_{min}</math></li> <li>- So sánh <math>2 * S_{min}</math> với Z để đưa ra kết quả</li> <li>- Độ phức tạp <math>O(n^2)</math></li> </ul>		<b>6,0</b>
<b>Test</b>	<b>GIAOHANG.INP</b>	<b>GIAOHANG.OUT</b>	
1	4 5 2 3 17 1 2 5 1 3 4 1 4 9 2 4 3 3 4 5	YES 16	Mỗi Test 0,5 điểm
2	3 2 1 3 10 1 2 5 2 3 4	NO	
3	5 7 1 3 240 1 2 70 1 5 60 2 3 50 2 4 40 2 5 20 3 4 40 4 5 30	YES 240	
4	5 5 1 4 25 1 2 2 2 3 3 2 5 7 3 4 8 4 5 2	YES 22	
5	File dulich05.inp	File dulich05.out	
6	File dulich06.inp	File dulich06.out	
7	File dulich07.inp	File dulich07.out	
8	File dulich08.inp	File dulich08.out	
9	File dulich09.inp	File dulich09.out	
10	File dulich10.inp	File dulich10.out	
11	File dulich11.inp	File dulich11.out	
12	File dulich12.inp	File dulich12.out	

-----HẾT-----

## Code tham khảo

### Bài 4. Dãy số

```
const nmax=100000;
var n:longint;
    K:int64;
    A:array[1..nmax] of longint;
    Sum:array[0..nmax] of int64;
//-----
procedure nhap;
var i:longint;
begin
    readln(n,K);
    for i:=1 to n do read(A[i]);
end;
//-----
procedure xuli; //O(n3)
//-----
function tongK(x,y:longint):boolean;
var p,tmp:longint;
begin
    tmp:=0;
    for p:=x to y do tmp:=tmp+A[p];
    if tmp=K then exit(true) else exit(false);
end;
//-----
var i,j,len:longint;
begin
    len:=0;
    for i:=1 to n do
        for j:=i to n do
            if tongK(i,j) then if len<j-i+1 then len:=j-i+1;
        if len=0 then write(-1) else write(len);
    end;
//-----
procedure xuli1; //O(n2)
var i,j,len:longint;
begin
    Sum[0]:=0; len:=0;
    for i:=1 to n do Sum[i]:=Sum[i-1]+A[i];

    for i:=1 to n do
        for j:=i to n do
            if sum[j]-sum[i-1]=K then if len<j-i+1 then len:=j-i+1;

    if len=0 then write(-1) else write(len);
end;
//-----
BEGIN
```

```

assign(input,'dayso.inp'); reset(input);
assign(output,'dayso.out'); rewrite(output);
nhap;
// xuli;
xuli1;
close(input); close(output);
END.

```

### **Bài 5. Vườn cây**

```

const nmax=10000;
var n,m,p:longint;
    A:array[1..nmax,1..nmax] of longint;
//-----
procedure nhap;
var i,j:longint;
begin
    readln(m,n);
    for i:=1 to m do
        for j:=1 to n do read(A[i,j]);
end;
//-----
procedure tinhtong(x,y,c:longint);
var i,j: longint; S:int64;
begin
    S:=0;
    for i:=x to x+c-1 do
        for j:=y to y+c-1 do S:=S+A[i,j];
    writeln(S);
end;
//-----
procedure xuli; //O(n3)
var i,p,ct,x,y,t:longint;
begin
    readln(p);
    for i:=1 to p do
        begin
            readln(ct,x,y,t);
            if ct=0 then A[x,y]:=t;
            if ct=1 then tinhtong(x,y,t);
        end;
end;
//-----
BEGIN
    assign(input,'vuoncay.inp'); reset(input);
    assign(output,'vuoncay.out'); rewrite(output);
    nhap;
    xuli;
    close(input); close(output);
END.

```

**Bài 6. Dulich**

```
const nmax=1000;
    maxC=trunc(1e9);
var n,m,s,t,X:longint;
    C:array[1..nmax,1..nmax] of longint;
    D:array[1..nmax] of int64;
    Fr:array[1..nmax] of boolean;

//-----
procedure nhap;
var i,j,u,v,cp:longint;
begin
    readln(n,m,s,t,X);
    for i:=1 to n do
        for j:=1 to n do C[i,j]:=maxC;
    for i:=1 to m do
        begin
            readln(u,v,cp);
            C[u,v]:=cp; C[v,u]:=cp;
        end;
end;
//-----
procedure Dijkstra;
var i,u,v,min:longint;
begin
    fillchar(Fr,sizeof(Fr),true);
    for i:=1 to n do D[i]:=C[s,i];
    D[s]:=0; Fr[s]:=false;
    repeat
        u:=0; min:=maxC;
        for i:=1 to n do
            if Fr[i] and (min > D[i]) then
                begin
                    min:=D[i];
                    u:=i;
                end;
        if (u=0) or (u=t) then break;
        Fr[u]:=false;
        for v:=1 to n do
            if Fr[v] and (D[v]>D[u]+C[u,v]) then D[v]:=D[u]+C[u,v];
        until false;
    end;
//-----
procedure xuli;
begin
    Dijkstra;
    if 2*D[t]>X then write('NO')
    else begin writeln('YES'); write(D[t]*2); end;
```

```
end;  
//-----  
BEGIN  
  assign(input,'dulich.inp'); reset(input);  
  assign(output,'dulich.out'); rewrite(output);  
  nhap;  
  xuli;  
  close(input); close(output);  
END.
```