

**ĐỀ CHÍNH THỨC**  
(Đề thi gồm 02 trang)

**MÔN: TIN HỌC 12 – THPT (VÒNG 1)**  
(Thời gian làm bài 180 phút, không kể giao đề)

Bài	File bài làm	Dữ liệu vào	Kết quả
Bài 1 – Tính tổng	BAI1.*	BAI1.INP	BAI1.OUT
Bài 2 – Bảng số	BAI2.*	BAI2.INP	BAI2.OUT
Bài 3 – Tuyến đường	BAI3.*	BAI3.INP	BAI3.OUT

Dấu \* được thay thế bởi PAS hoặc CPP của ngôn ngữ lập trình được sử dụng tương ứng là PASCAL hoặc C++.

**Bài 1 (6 điểm) – Tính tổng**

Cho dãy số nguyên  $a_1, a_2, \dots, a_N$  ( $|a_i| < 10^9, N < 10^5$ ). Một tập hợp khác rỗng các số hạng liên tiếp  $\{a_i, a_{i+1}, \dots, a_k\}$  ( $i \leq k$ ) gọi là một đoạn con của dãy số. Với mỗi đoạn con ta tính tổng tất cả các số hạng của nó.

**Yêu cầu:** Tìm giá trị lớn nhất trong số các tổng của các đoạn con từ dãy số đã cho.

**Dữ liệu vào** từ file BAI1.INP gồm:

- Dòng đầu chứa số N.
- Dòng thứ i trong N dòng tiếp theo chứa số  $a_i$ .

**Kết quả** ghi vào file BAI1.OUT giá trị lớn nhất trong số các tổng của các đoạn con.

**Ví dụ:**

BAI1.INP	BAI1.OUT
6	8
1	
-2	
5	
-1	
4	
-3	

**Bài 2 (7 điểm) – Bảng số**

Trò chơi sau đây do Hirofumi Fujiwara đề xuất: Trong một lưới ô vuông kích thước  $9 \times 9$  cần điền các số từ 1 đến 9 sao cho trong mỗi dòng, mỗi cột và mỗi lưới ô vuông con kích thước  $3 \times 3$  đều có tất cả các số từ 1 đến 9 (lưới ô vuông được chia thành 9 lưới ô vuông kề nhau mỗi lưới ô vuông có kích thước  $3 \times 3$ ).

**Yêu cầu:** Cho trước một lưới ô vuông kích thước  $9 \times 9$ , trong đó có một số ô đã được điền số, cần xác định có thể điền tiếp các số vào các ô còn lại để thu được bảng số đúng luật của trò chơi hay không?

**Dữ liệu vào** từ file BAI2.INP gồm 9 dòng mô tả bảng số, mỗi dòng gồm 9 ký tự, ký tự '.' để chỉ ô chưa được điền số.

**Kết quả** ghi vào file BAI2.OUT bảng số tìm được gồm 9 dòng, mỗi dòng 9 ký tự số hoặc thông báo 'That bai' nếu không tìm được lời giải.

**Ví dụ:**

BAI2.INP	BAI2.OUT
.....2.3.	185742936
374..6.2.	374916528
.....8.1.	692358714
258.....	258493167
.....	439671852
.....493	716285493
.4.1.....	543167289
.2.8..675	921834675
.6.5.....	867529341

### Bài 3 (7 điểm) – Tuyến đường

Để phát triển ngành du lịch, lãnh đạo của tỉnh A đầu tư xây dựng N địa điểm du lịch được đánh số từ 1 đến N và các tuyến đường đi giữa các địa điểm đó. Lãnh đạo tỉnh A chọn ra 4 địa điểm du lịch đặc biệt để làm trung tâm đầu tư phát triển và 4 địa điểm này phải liên thông với nhau. Lãnh đạo tỉnh muốn xây dựng các tuyến đường đi sao cho các địa điểm du lịch liên thông với nhau với chi phí ít nhất.

**Yêu cầu:** Bạn hãy viết chương trình để chọn một số tuyến đường đi cần xây dựng nhằm đảm bảo các địa điểm du lịch liên thông với nhau sao cho chi phí ít nhất.

**Dữ liệu vào** từ file BAI3.INP gồm:

- Dòng đầu chứa số nguyên dương N.
- Dòng thứ hai ghi 4 số nguyên là số hiệu của 4 địa điểm du lịch đặc biệt, kí hiệu từ 1 đến 4.
- Các dòng tiếp theo, mỗi dòng ghi 3 số nguyên u, v và c với ý nghĩa muốn xây dựng một tuyến đường hai chiều nối trực tiếp giữa hai địa điểm du lịch u và v với chi phí là c.

**Kết quả** ghi vào file BAI3.OUT gồm:

- Dòng đầu ghi tổng chi phí nhỏ nhất để xây dựng các tuyến đường.
- Các dòng tiếp theo, mỗi dòng ghi 2 số nguyên u và v với ý nghĩa cần xây dựng tuyến đường hai chiều nối giữa hai địa điểm du lịch u và v.

**Ví dụ:**

BAI3.INP	BAI3.OUT
5	5
2 3 4 1	1 4
1 2 10	1 5
1 5 1	5 2
5 2 1	2 3
1 4 1	
4 3 3	
3 2 2	

----- HẾT -----

Thí sinh không được sử dụng tài liệu. Giám thị không giải thích gì thêm.  
Họ và tên.....Số báo danh:.....

**ĐÁP ÁN VÀ HƯỚNG DẪN CHẤM VÒNG 1****I. Phần chương trình nguồn****BÀI 1: (6đ)**

```
const nmax = 100000;
var n:longint; i:byte;
    S:array[0..nmax] of int64;
    kq:int64;

procedure nhapdulieu;
var f:text;
    i,a:longint;
begin
    s[0]:=0;
    assign(f,'BAI1.INP');
    reset(f);
    readln(f,n);
    for i:= 1 to n do
        begin
            readln(f,a);
            s[i]:=s[i-1]+a;
        end;
    close(f);
end;

procedure xuly;
var i,j:longint;
    max:int64;
    f:text;
begin
    max:=s[n];
    kq:=-2000000000000000;
    for i:= n-1 downto 0 do
        begin
            if kq<max-s[i] then kq:=max-s[i];
            if s[i]>max then max:=s[i];
        end;
    assign(f,'BAI1.OUT');
    rewrite(f);
    writeln(f,kq);
    close(f);
end;

begin
    nhapdulieu;
    xuly;
end.
```

**BÀI 2: (7đ)**

```
{ $A+,B-,D+,E+,F-,G-,I+,L+,N+,O-,P-,Q-,R-,S+,T-,V+,X+ }
{ $M 65000,0,655360 }
```

uses crt;

```
{const
  tfi      = 'BAI2.INP';
  tfo      = 'BAI2.OUT';}
```

```
var
  fi,fo    : text;
  x        : array[1..9,1..9] of byte;
  Good     : boolean;
  sol      : array[1..9,1..9] of integer;
  cn       : array[1..9,1..9,1..9] of byte;
```

procedure Docdl;

```
var i,j: integer;
    ch: char;
begin
  fillchar(x,sizeof(x),0);
  for i:=1 to 9 do
    begin
      for j:=1 to 9 do
        begin
          read(fi,ch);
          if ch<>'.' then x[i,j]:=ord(ch)-48;
        end;
      readln(fi);
    end;
end;
```

procedure Dat(i,j,k: integer; var ok: boolean);

```
var u,v,id,ic,jd,jc: integer;
begin
  ok:=false;
  for u:=1 to 9 do
    begin
      if (u<>i) and (x[u,j]=k) then exit;
      if cn[u,j,k]=0 then
        begin
          sol[u,j]:=sol[u,j]-1;
          cn[u,j,k]:=100;
          if (x[u,j]=0) and (sol[u,j]=0) then exit;
        end;
    end;
  for v:=1 to 9 do
    begin
      if (v<>j) and (x[i,v]=k) then exit;
      if cn[i,v,k]=0 then
        begin
          sol[i,v]:=sol[i,v]-1;
          cn[i,v,k]:=100;
          if (x[i,v]=0) and (sol[i,v]=0) then exit;
        end;
    end;
end;
```



```

id:=(i-1) div 3*3+1; ic:=id+2;
jd:=(j-1) div 3*3+1; jc:=jd+2;
for u:=id to ic do
  for v:=jd to jc do
    begin
      if ((u<>i) or (v<>j)) and (x[u,v]=k) then exit;
      if cn[u,v,k]=0 then
        begin
          sol[u,v]:=sol[u,v]-1;
          cn[u,v,k]:=100;
          if (x[u,v]=0) and (sol[u,v]=0) then exit;
        end;
      end;
    ok:=true;
  end;
end;

```

```

procedure Chuanbi;
var i,j,k: integer;
    ok: boolean;
begin
  Good:=false;
  for i:=1 to 9 do
    for j:=1 to 9 do sol[i,j]:=9;
  fillchar(cn,sizeof(cn),0);
  for i:=1 to 9 do
    for j:=1 to 9 do
      if x[i,j]>0 then
        begin
          k:=x[i,j];
          Dat(i,j,k,ok);
          if not ok then exit;
        end;
      end;
    for i:=1 to 9 do
      for j:=1 to 9 do
        if (x[i,j]=0) and (sol[i,j]=1) then
          for k:=1 to 9 do
            if cn[i,j,k]=0 then
              begin
                x[i,j]:=k;
                Dat(i,j,k,ok);
                if not ok then exit;
              end;
            end;
          Good:=true;
        end;
      end;
end;

```

```

procedure Thu(u,v: integer);
var k,i,j,ud,uc,vd,vc: integer;
    tiep: boolean;
begin
  if u=10 then
    begin
      Good:=true;
      exit;
    end;
end;

```

```

if v=10 then
  begin
    Thu(u+1,1);
    exit;
  end;
if x[u,v]>0 then
  begin
    Thu(u,v+1);
    exit;
  end;
for k:=1 to 9 do
  if cn[u,v,k]=0 then
    begin
      x[u,v]:=k;

      (* Dat trang thai *)
      tiep:=true;
      for i:=u+1 to 9 do
        if (x[i,v]=0) and (cn[i,v,k]=0) then
          begin
            cn[i,v,k]:=(u-1)*9+v;
            sol[i,v]:=sol[i,v]-1;
            tiep:=(sol[i,v]>0);
          end;
      for j:=v+1 to 9 do
        if (x[u,j]=0) and (cn[u,j,k]=0) then
          begin
            cn[u,j,k]:=(u-1)*9+v;
            sol[u,j]:=sol[u,j]-1;
            tiep:=(sol[u,j]>0);
          end;
      ud:=(u-1) div 3*3+1; uc:=ud+2;
      vd:=(v-1) div 3*3+1; vc:=vd+2;
      for j:=v+1 to vc do
        if (x[u,j]=0) and (cn[u,j,k]=0) then
          begin
            cn[u,j,k]:=(u-1)*9+v;
            sol[u,j]:=sol[u,j]-1;
            tiep:=(sol[u,j]>0);
          end;
      for i:=u+1 to uc do
        for j:=vd to vc do
          if (x[i,j]=0) and (cn[i,j,k]=0) then
            begin
              cn[i,j,k]:=(u-1)*9+v;
              sol[i,j]:=sol[i,j]-1;
              tiep:=(sol[i,j]>0);
            end;

      (* thu tiep *)
      if Tiep then Thu(u,v+1);
      if Good then exit;

      (* Quay lui *)

```

```

x[u,v]:=0;
for i:=u+1 to 9 do
  if cn[i,v,k]=(u-1)*9+v then
    begin
      cn[i,v,k]:=0;
      sol[i,v]:=sol[i,v]+1;
    end;
  for j:=v+1 to 9 do
    if cn[u,j,k]=(u-1)*9+v then
      begin
        cn[u,j,k]:=0;
        sol[u,j]:=sol[u,j]+1;
      end;
    for j:=v+1 to vc do
      if cn[u,j,k]=(u-1)*9+v then
        begin
          cn[u,j,k]:=0;
          sol[u,j]:=sol[u,j]+1;
        end;
      for i:=u+1 to uc do
        for j:=vd to vc do
          if cn[i,j,k]=(u-1)*9+v then
            begin
              cn[i,j,k]:=0;
              sol[i,j]:=sol[i,j]+1;
            end;
          end;
        end;
      end;
    end;
  end;
end;

procedure Duyet;
begin
  if not Good then exit;
  Good:=false;
  Thu(1,1);
end;

procedure Inkq;
var i,j: integer;
begin
  if not Good then
    begin
      writeln(fo,'That bai');
      exit;
    end;
  for i:=1 to 9 do
    begin
      for j:=1 to 9 do write(fo,x[i,j]);
      writeln(fo);
    end;
  end;
end;

BEGIN
  assign(fi,'BAI2.INP'); reset(fi);

```

```

assign(fo,'BAI2.OUT'); rewrite(fo);
Docdl;
Chuanbi;
Duyet;
Inkq;
close(fi); close(fo);
END.
BÀI 3: (7đ)
const
    maxN                =    101;
    Unseen: longint      =    600000;
    H: array[1..6,1..4] of integer =    ((1,2,3,4),
                                           (1,3,2,4),
                                           (1,4,2,3),
                                           (2,3,1,4),
                                           (2,4,1,3),
                                           (3,4,1,2));

type
    mang1=array[1..maxN,1..maxN] of longint;
    mang2=array[1..maxN,1..maxN] of integer;

var
    fi, fo: text;
    N: integer;
    t: array[1..4] of integer;
    a: mang1;
    Tr: mang2;
    min: longint;
    slx: integer;
    x: array[1..2,1..maxN] of integer;

procedure Nhap;
var i,j,u,v,c: integer;
begin
    assign(fi,'BAI3.INP'); reset(fi);
    readln(fi,N);
    readln(fi,t[1],t[2],t[3],t[4]);
    for i:=1 to n do
        for j:=1 to n do a[i,j]:=Unseen;
    while not seekeof(fi) do
        begin
            readln(fi,u,v,c);
            a[u,v]:=c;
            a[v,u]:=c;
        end;
    close(fi);
end;

procedure Floy;
var k,i,j: integer;
begin
    fillchar(Tr,sizeof(tr),0);
    for k:=1 to n do

```



```

for i:=1 to n do
  for j:=1 to n do
    if a[i,j]>a[i,k]+a[k,j] then
      begin
        a[i,j]:=a[i,k]+a[k,j];
        tr[i,j]:=k;
      end;
  for i:=1 to n do a[i,i]:=0;
end;

```

```

procedure Tim(u,v: integer);
begin
  if Tr[u,v]=0 then
    begin
      inc(slx);
      x[1,slx]:=u;
      x[2,slx]:=v;
      exit;
    end;
  Tim(u,Tr[u,v]);
  Tim(Tr[u,v],v);
end;

```

```

procedure GhiNhan(i,j,k: integer);
begin
  slx:=0;
  if i<>t[H[k,1]] then Tim(i,t[H[k,1]]);
  if i<>t[H[k,2]] then Tim(i,t[H[k,2]]);
  if i<>j then Tim(i,j);
  if j<>t[H[k,3]] then Tim(j,t[H[k,3]]);
  if j<>t[H[k,4]] then Tim(j,t[H[k,4]]);
end;

```

```

procedure Solve;
var i,j,k: integer;
    Tong: longint;
begin
  Min:=Unseen;
  for i:=1 to n do
    for j:=1 to n do
      if (a[i,j]<Unseen) then
        for k:=1 to 6 do
          if (a[i,t[H[k,1]]]<Unseen) and (a[i,t[H[k,2]]]<Unseen) and
            (a[j,t[H[k,3]]]<Unseen) and (a[j,t[H[k,4]]]<Unseen) then
            begin
              Tong:=a[i,j]+a[i,t[H[k,1]]]+a[i,t[H[k,2]]]+
                a[j,t[H[k,3]]]+a[j,t[H[k,4]]];
              if Tong<Min then
                begin
                  Min:=Tong;
                  GhiNhan(i,j,k);
                end;
            end;
        end;
    end;
  end;
end;

```

```
procedure Xuat;  
var i: integer;  
begin  
  assign(fo,'BAI3.OUT'); rewrite(fo);  
  writeln(fo,Min);  
  for i:=1 to slx do writeln(fo,x[1,i],#32,x[2,i]);  
  close(fo);  
end;
```

BEGIN

Nhap;  
Floy;  
Solve;  
Xuat;

END.

## **II. Hướng dẫn chấm.**

**Bài 1: 3 Test, mỗi test đúng cho 2 điểm**

**Bài 2: 3 Test và test ví dụ, mỗi test đúng cho 2 điểm, riêng test ví dụ đúng được 1 điểm.**

**Bài 3: 3 Test và test ví dụ, mỗi test đúng cho 2 điểm, riêng test ví dụ đúng được 1 điểm.**

*Chú ý: Kết quả có thể có nhiều phương án khác nhau do đó yêu cầu giám khảo xem xét kỹ khi chấm bài.*

---- Hết ----

**ĐỀ CHÍNH THỨC**  
(Đề thi gồm 02 trang)

**MÔN: TIN HỌC 12 – THPT (VÒNG 2)**  
(Thời gian làm bài 180 phút, không kể giao đề)

Bài	File bài làm	Dữ liệu vào	Kết quả
Bài 1 – Đội cờ	BAI1.*	BAI1.INP	BAI1.OUT
Bài 2 – Xe Buýt	BAI2.*	BAI2.INP	BAI2.OUT

Dấu \* được thay thế bởi **PAS** hoặc **CPP** của ngôn ngữ lập trình được sử dụng tương ứng là **PASCAL** hoặc **C++**.

**Bài 1. (10 điểm) – Đội cờ**

Có hai đội cờ vua A và B thi đấu với nhau. Mỗi đội cờ cử ra n kỳ thủ, mỗi kỳ thủ của đội B chỉ đấu một trận và chỉ đấu với một kỳ thủ của đội A và ngược lại. Vậy có tất cả n trận đấu. Đội nào thắng được 2 điểm, hòa được 1 điểm và thua được 0 điểm. Cho đội B được quyền chọn cặp thi đấu.

**Yêu cầu:**

Lập trình để đội B chọn được các cặp thi đấu sao cho tổng số điểm của đội B là cao nhất. Cho biết trình độ của cầu thủ thứ i của đội A và B lần lượt là a[i] và b[i] (i=1,2,...,n) và giả sử trong thi đấu, hai kỳ thủ có trình độ ngang nhau sẽ hòa và kỳ thủ nào có trình độ cao hơn sẽ thắng.

**Dữ liệu:**

Ghi trên file văn bản BAI1.INP gồm n+1 dòng:

Dòng đầu tiên ghi số nguyên dương n,  $1 \leq n \leq 1000$

Dòng thứ i+1 ghi 2 số nguyên a[i], b[i] ( $1 \leq a[i]$ ,  $b[i] \leq 100$ ) cách nhau ít nhất một khoảng trắng.

**Kết quả:**

Ghi ra file văn bản BAI1.OUT gồm n+1 dòng

Dòng đầu ghi số nguyên T là tổng số điểm cao nhất mà đội B có thể đạt được.

Dòng thứ i+1 ( $1 \leq i \leq n$ ) ghi số nguyên dương x[i] ( $1 \leq x[i] \leq n$ ) với x[i] là số thứ tự của kỳ thủ đội B phải thi đấu với kỳ thủ thứ i của đội A để tổng số điểm của đội B đạt được là T.

Ví dụ:

**BAI1.INP**

4  
7 8  
5 6  
4 3  
9 4

**BAI1.OUT**

5  
1  
2  
4  
3

## Bài 2. (10 điểm) – Xe Buýt

Một xe buýt của công ty có nhiệm vụ đón nhân viên đến trụ sở làm việc. Trên hành trình xe buýt sẽ tiếp nhận nhân viên đứng chờ ở các điểm hẹn nếu như xe còn chỗ trống. Xe buýt có thể sẽ đỗ lại để chờ những công nhân còn chưa kịp đến điểm hẹn. Cho biết thời điểm mà mỗi nhân viên đến điểm hẹn của mình và thời điểm qua mỗi điểm hẹn của xe buýt. Giả thiết rằng xe buýt đến điểm hẹn đầu tiên tại thời điểm 0, thời gian xếp khách lên xe coi như bằng 0. Hãy xác định khoảng thời gian ngắn nhất để xe buýt có thể chở một số lượng các nhân viên đến trụ sở làm việc lớn nhất có thể được.

Dữ liệu vào từ file BAI2.INP:

Dòng đầu tiên chứa hai số nguyên dương  $N$  và  $M$  theo thứ tự là số điểm hẹn và số chỗ ngồi của xe buýt.

Dòng thứ  $i$  trong số  $N$  dòng tiếp theo chứa số nguyên  $t_i$  là thời gian cần thiết để xe buýt di chuyển từ điểm hẹn  $i$  đến điểm hẹn  $i+1$  (điểm hẹn thứ  $N+1$  sẽ là trụ sở làm việc của công ty), số nguyên  $K$  là số lượng nhân viên đến điểm hẹn  $i$  và tiếp đến là  $K$  số nguyên là các thời điểm đến điểm hẹn của  $K$  nhân viên.

Giới hạn:  $1 \leq M \leq 2000$ ,  $1 \leq N \leq 200000$

Kết quả ghi ra file văn bản BAI2.OUT thời gian ngắn nhất tìm được.

Ví dụ:

BAI2.INP

BAI2.OUT

3 5

4

1 2 0 1

1 1 2

1 4 0 2 3 4

----- HẾT -----

Thí sinh không được sử dụng tài liệu. Giám thị coi thi không giải thích gì thêm.

Họ và tên.....Số báo danh:.....



**ĐÁP ÁN VÀ HƯỚNG DẪN CHẤM VÒNG 2****I. Phần chương trình nguồn****BÀI 1:**

```
{ *****
Thuat toan:  Sap xep
***** }

uses crt;
const
  tfi      = 'BAI1.INP';
  tfo      = 'BAI1.OUT';
  maxN     = 1000;
var
  fi, fo   : text;
  N        : integer;
  dd,x,a,b,csA,csB : array[1..maxN] of integer;
  DiemMax  : integer;
procedure Docdl;
var i: integer;
begin
  assign(fi,tfi); reset(fi);
  readln(fi,N);
  for i:=1 to N do read(fi,a[i],b[i]);
  close(fi);
end;
procedure Doi(var u,v: integer);
var w: integer;
begin
  w:=u;
  u:=v;
  v:=w;
end;
procedure SortA(k,l: integer);
var i,j,mid: integer;
begin
  mid:=a[(k+l) div 2];
  i:=k; j:=l;
  repeat
    while a[i]<mid do inc(i);
    while a[j]>mid do dec(j);
    if i<=j then
      begin
        Doi(a[i],a[j]);
        Doi(csA[i],csA[j]);
        inc(i);
```



```

        dec(j);
    end;
until i>j;
if j>k then SortA(k,j);
if i<l then SortA(i,l);
end;
procedure SapxepA;
var i: integer;
begin
    for i:=1 to N do csA[i]:=i;
    SortA(1,N);
end;
procedure SortB(k,l: integer);
var i,j,mid: integer;
begin
    mid:=b[(k+l) div 2];
    i:=k;
    j:=l;
    repeat
        while b[i]>mid do inc(i);
        while b[j]<mid do dec(j);
        if i<=j then
            begin
                Doi(b[i],b[j]);
                Doi(csB[i],csB[j]);
                inc(i);
                dec(j);
            end;
    until i>j;
    if j>k then SortB(k,j);
    if i<l then SortB(i,l);
end;
procedure SapXepB;
var i: integer;
begin
    for i:=1 to N do csB[i]:=i;
    SortB(1,n);
end;
function TimA(r: integer): integer;
var i: integer;
begin
    for i:=n downto 1 do
        if (dd[csA[i]]=0) and (a[i]<r) then
            begin
                TimA:=i;
                exit;
            end;
    for i:=n downto 1 do

```

```

    if (dd[csA[i]]=0) and (a[i]=r) then
        begin
            TimA:=i;
            exit;
        end;
    TimA:=0;
end;
procedure Ghep;
var j,limit,i,k: integer;
begin
    fillchar(dd,sizeof(dd),0);
    fillchar(x,sizeof(x),0);
    i:=1;
    DiemMax:=0;
    limit:=n;
    repeat
        k:=TimA(b[i]);
        if k>0 then
            begin
                x[csA[k]]:=csB[i];
                dd[csA[k]]:=1;
                if b[i]>a[k] then inc(DiemMax,2) else inc(DiemMax,1);
                limit:=k-1;
                inc(i);
            end;
        until k=0;
        for j:=i to N do
            begin
                repeat inc(k) until dd[k]=0;
                x[k]:=csB[j];
            end;
    end;
procedure inkq;
var i: integer;
begin
    assign(fo,tfo); rewrite(fo);
    writeln(fo,diemMax);
    for i:=1 to N do writeln(fo,x[i]);
    close(fo);
end;
BEGIN
    Docdl;
    SapXepA;
    SapXepB;
    Ghep;
    Inkq;
END.

```

**BÀI 2:**

```

{$r-}
const
    fi = 'BAI2.INP';
    fo = 'BAI2.OUT';
    nmax = 2000;

var
    f      : text;
    sl     : array [0..10000] of real;
    n, max, t : longint;
    m      : real;
procedure open_file;
begin
    assign(f, fi); reset(f);
    readln(f, n, m); {n: so diem hen, m: so cho ngoi}
    fillchar(sl, sizeof(sl), 0);
end;
function getmax(i, j : longint) : longint;
begin
    if i > j then getmax := i
    else getmax := j;
end;
procedure solve;
var ii, j, i, time, k, cs : longint;
begin
    t := 0; max := 0;
    for ii := 1 to n do
        begin
            read(f, time);
            read(f, k);
            for i := 1 to k do
                begin
                    read(f, j);
                    cs := getmax(0, j - t);
                    sl[cs] := sl[cs] + 1;
                    if cs > max then max := cs;
                end;
            t := t + time;
        end;
    end;
procedure print;
var
    i : longint;
    kq : real;
begin
    assign(f, fo); rewrite(f);
    kq := sl[0];
    i := 0;

```

```
while (kq < m) and (i < max) do
begin
    inc(i);
    kq := kq + sl[i];
    t := t + 1;
end;
writeln(f, t);
close(f);
end;
begin
    open_file;
    solve;
    print;
end.
```

## **II. Hướng dẫn chấm.**

**Bài 1: 5 Test, mỗi test đúng cho 2 điểm**

**Bài 2: 5 Test, mỗi test đúng cho 2 điểm**

**Chú ý: Kết quả có thể có nhiều phương án khác nhau do đó yêu cầu giám khảo xem xét kỹ khi chấm bài.**

---- Hết ----