

- Dòng đầu tiên ghi hai số M, N - kích thước ô lưới ($M, N \leq 100$),
- M dòng tiếp theo, mỗi dòng một xâu N kí tự 0, 1 ứng với dòng tương ứng của A ,
- Tiếp theo là một dòng trống,
- M dòng cuối mỗi dòng 1 xâu N kí tự 0, 1 ứng với dòng tương ứng của B .

Dữ liệu ra trong file BIENDOI.OUT:

- Dòng đầu số nguyên k là số lượng phép biến đổi ít nhất cần áp dụng ($k = 0$ nếu không biến đổi được)
- Dòng thứ i trong số k dòng tiếp theo ghi hai số nguyên xác định ô cần chọn để thực hiện phép biến đổi.

Ví dụ:

BIENDOI.INP

```
4 5
1 0 0 0 0
1 0 0 0 0
0 1 0 0 0
0 1 0 0 0
```

```
0 0 0 0 0
0 0 0 0 0
0 0 1 0 0
0 0 0 0 0
```

BIENDOI.OUT

```
2
2 1
3 2
```

(Đề ra của bạn Nguyễn Văn Đức - Cần Thơ)

Bài 86/2001 - Dãy số tự nhiên logic

(Dành cho học sinh Tiểu học)

Đây là một chuỗi các số tự nhiên được sắp xếp theo một logic nào đó. Hãy tìm con số đầu tiên và cuối cùng của dãy số để thay thế cho dấu ?

? 12 14 15 16 18 20 21 22 ?

Bài 87/2001 - Ghi số trên bảng

(Dành cho học sinh THCS)

Trên bảng ghi số 0. Mỗi lần được tăng số đã viết lên bảng thêm 1 đơn vị hoặc tăng gấp đôi. Hỏi sau ít nhất là bao nhiêu bước sẽ thu được số nguyên dương N ?

Bài 88/2001 - Về các số đặc biệt có 10 chữ số

(Dành cho học sinh THCS và THPT)

Lập chương trình tính (và chỉ ra) tất cả các số có 10 chữ số $a_0a_1a_2\dots a_9$ thoả mãn các tính chất sau:

a_0 bằng số chữ số 0 của số trên;

a_1 bằng số chữ số 1 của số trên;

a_2 bằng số chữ số 2 của số trên;

.....

a_9 bằng số chữ số 9 của số trên;

Bài 89/2001 - Chữ số thứ N

(Dành cho học sinh THCS và THPT)

Khi viết các số tự nhiên tăng dần từ 1, 2, 3, ... liên tiếp nhau, ta nhận được một dãy các chữ số thập phân vô hạn, ví dụ: 1234567891011121314151617181920...

Yêu cầu: Hãy tìm chữ số thứ N của dãy số vô hạn trên.

Dữ liệu vào từ file 'Number.inp' gồm một số dòng, mỗi dòng ghi một số nguyên dương N ($N < 10^9$).

Kết quả ra file 'Number.out', với mỗi số N đọc được từ file Number.inp, ghi trên dòng tương ứng chữ số thứ N của dãy.

Ví dụ:

Number.inp	Number.out
5	5
10	1
54	3

Bài 90/2002 - Thay số trong bảng 9 ô

(Dành cho học sinh Tiểu học)

Cho một bảng vuông gồm 9 ô. Đầu tiên các ô được điền bởi các chữ cái I, S, M. Bạn hãy thay những số thích hợp vào các ô sao cho tổng các số trong các ô điền cùng chữ cái ban đầu là bằng nhau và là một số chia hết cho 4.

Chú ý: các ô cùng chữ cái phải thay bởi những số như nhau.

I	S	M
S	I	S
M	S	I

Bài 91/2002 - Các số lặp

(Dành cho học sinh THCS và THPT)

Cho dãy số nguyên gồm N phần tử. Lập chương trình in ra số được lặp nhiều nhất trong dãy.

Bài 92/2002 - Dãy chia hết*(Dành cho học sinh THPT)*

Xét một dãy gồm N số nguyên tùy ý. Giữa các số nguyên đó ta có thể đặt các dấu + hoặc - để thu được các biểu thức số học khác nhau. Ta nói dãy số là chia hết cho K nếu một trong các biểu thức thu được chia hết cho K . Hãy viết chương trình xác định tính chia hết của một dãy số đã cho.

Dữ liệu vào: Lấy từ một file văn bản có tên là DIV.INP có cấu trúc như sau:

- Dòng đầu là hai số N và K ($2 \leq N \leq 10\,000$, $2 \leq K \leq 100$), cách nhau bởi dấu trống.
- Các dòng tiếp theo là dãy N số có trị tuyệt đối không quá 10 000 cách nhau bởi dấu trống hoặc dấu xuống dòng.

Dữ liệu ra: Ghi ra file văn bản DIV.OUT số 1 nếu dãy đã cho chia hết cho K và số 0 nếu ngược lại.

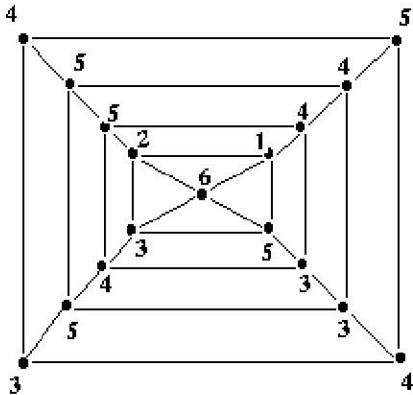
Ví dụ:

DIV.INP	DIV.OUT	DIV.INP	DIV.OUT
4 6	0	4 7	1
1 2 3 5		1 2 3 5	

(Đề ra của bạn Trần Đình Trung - Lớp 11A Tin - Khối PTCT - ĐH Vinh)

Bài 93/2002 - Trò chơi bắn bi*(Dành cho học sinh Tiểu học)*

Cho bảng bắn bi sau:



Bạn có thể bắn bi vào từ một trong số các đỉnh ở ngoài cùng. Khi được bắn vào trong, hòn bi chỉ có thể tiếp tục đi vào trong ở đỉnh gần đó nhất hoặc lăn theo nhiều nhất là một cạnh để đi vào ở đỉnh kề đó. Biết rằng khi đến hình chữ nhật trong cùng, hòn bi không được lăn trên một cạnh nào mà phải đi thẳng vào tâm.

Hãy tìm đường đi sao cho tổng số điểm mà nó đi qua là lớn nhất và có bao nhiêu đường đi để có được số điểm đó.

Bài 94/2002 - Biểu diễn tổng các số Fibonacci*(Dành cho học sinh THCS)*

Cho số tự nhiên N và dãy số Fibonacci: 1, 1, 2, 3, 5, 8,

Bạn hãy viết chương trình kiểm tra xem N có thể biểu diễn thành tổng của các số Fibonacci khác nhau hay không?

Bài 95/2002 - Dãy con có tổng lớn nhất

(Dành cho học sinh THPT)

Cho dãy gồm n số nguyên a_1, a_2, \dots, a_n . Tìm dãy con gồm một hoặc một số phần tử liên tiếp của dãy đã cho với tổng các phần tử trong dãy là lớn nhất.

Dữ liệu: Vào từ file văn bản SUBSEQ.INP

- Dòng đầu tiên chứa số nguyên dương n ($n < 10^6$).
- Dòng thứ i trong số n dòng tiếp theo chứa số a_i ($|a_i| \leq 1000$).

Kết quả: Ghi ra file văn bản SUBSEQ.OUT

- Dòng đầu tiên ghi vị trí của phần tử đầu tiên của dãy con tìm được.
- Dòng thứ hai ghi vị trí của phần tử cuối cùng của dãy con tìm được.
- Dòng thứ ba ghi tổng các phần tử của dãy con tìm được.

Ví dụ:

SUBSEQ.INP	SUBSEQ.OUT
8 12 -14 1 23 -6	3 6 40
22 -34 13	

Bài 96/2002 - Số chung lớn nhất

(Dành cho học sinh THPT)

Cho 2 xâu:

$X = x_1x_2\dots x_M$. (Với x_i là các kí tự số từ '0' đến '9')

$Y = y_1y_2\dots y_N$. (Với y_i là các kí tự số từ '0' đến '9')

($M, N \leq 250$)

Ta gọi: $Z = z_1z_2\dots z_k$ là xâu chung của 2 xâu X, Y nếu xâu Z nhận được từ xâu X bằng cách xoá đi một số kí tự và cũng nhận được từ xâu Y bằng cách xoá đi một số kí tự.

Yêu cầu: Tìm một xâu chung của 2 xâu X, Y sao cho xâu nhận được tạo thành một số lớn nhất có thể được.

Dữ liệu vào file: **String.inp**

Gồm 2 dòng, dòng 1 là xâu X , dòng 2 là xâu Y .

Kết quả ra file: **String.out**

Gồm 1 dòng duy nhất là số lớn nhất có thể nhận được.

Ví dụ:

String.inp	String.out
19012304	34
034012	

Bài 97/2002 - Thay số trong bảng*(Dành cho học sinh Tiểu học)*

Bảng dưới gồm 9 ô, ban đầu được điền bởi các chữ cái. Bạn hãy thay các chữ cái bởi các chữ số từ 0 đến 8 vào ô sao cho tất cả các số theo hàng ngang, hàng dọc đều là số có 3 chữ số (chữ số hàng trăm phải khác 0) và thoả mãn:

	1	2	3
4	a	b	c
5	d	e	f
6	g	h	i

Ngang

- 4 - Bội số nguyên của 8;
- 5 - Tích của các số tự nhiên liên tiếp đầu tiên;
- 6 - Tích các số nguyên tố kế nhau

Dọc

- 1 - Bội nguyên của 11;
- 2 - Tích của nhiều thừa số 2;
- 3 - Bội số nguyên của 11.

(Đề ra của bạn Đào Tuấn Anh - Lớp 10A Trường THPT Năng Khiếu Ngô Sĩ Liên - thị xã Bắc Giang)

Bài 98/2002 - Số phân nguyên tố*(Dành cho học sinh THCS và THPT)*

Một số n gọi là số phân nguyên tố nếu số ước số của nó là nhiều nhất trong n số tự nhiên đầu tiên. Cho số K ($K \leq 2$ tỷ). Hãy ghi ra số phân nguyên tố lớn nhất nhỏ hơn hoặc bằng K .

Dữ liệu vào trong file PNT.INP nội dung gồm:

- Dòng đầu tiên là số M ($1 < M \leq 100$) - số các số cần tìm số phân nguyên tố lớn nhất của nó;
- M dòng tiếp theo lần lượt là các số $K_1, K_2, K_3, \dots, K_M$;

Dữ liệu ra trong file PNT.OUT gồm M dòng: dòng thứ i là số phân nguyên tố lớn nhất nhỏ hơn hoặc bằng K_i .

Ví dụ:**PNT.INP**

1

1000

PNT.OUT

840

(Tác giả: Master - gửi bài qua Website của Tin học & Nhà trường)

Bài 99/2002 - Bài toán chúc Tết*(Dành cho học sinh THPT)*

Một người quyết định dành một ngày Tết để đến chúc Tết các bạn của mình. Để chắc

chấn, hôm trước anh ta đã điện thoại đến từng người để hỏi khoảng thời gian mà người đó có thể tiếp mình. Giả sử có N người được hỏi (đánh số từ 1 đến N), người thứ i cho biết thời gian có thể tiếp trong ngày là từ A_i đến B_i ($i = 1, 2, \dots, N$). Giả thiết rằng, khoảng thời gian cần thiết cho mỗi cuộc gặp là H và khoảng thời gian chuẩn bị từ một cuộc gặp đến một cuộc gặp kế tiếp là T . Bạn hãy xây dựng giúp một lịch chúc Tết để anh ta có thể chúc Tết được nhiều người nhất.

File dữ liệu vào trong file CHUCTET.INP gồm dòng đầu ghi số N , dòng thứ i trong số N dòng tiếp theo ghi khoảng thời gian có thể tiếp khách của người i gồm 2 số thực A_i và B_i (cách nhau ít nhất một dấu trắng). Dòng tiếp theo ghi giá trị H (số thực) và dòng cuối cùng ghi giá trị T (số thực). Giả thiết rằng các giá trị thời gian đều được viết dưới dạng thập phân theo đơn vị giờ, tính đến 1 số lẻ (thí dụ 10.5 có nghĩa là mười giờ rưỡi) và đều nằm trong khoảng từ 8 đến 21 (từ 8 giờ sáng đến 9 giờ tối). Số khách tối đa không quá 30.

Kết quả ghi ra file CHUCTET.OUT gồm dòng đầu ghi K là số người được thăm, K dòng tiếp theo ghi trình tự đi thăm, mỗi dòng gồm 2 số (ghi cách nhau ít nhất một dấu trắng): số đầu là số hiệu người được thăm, số tiếp theo là thời điểm gặp tương ứng.

Thí dụ:

CHUCTET.INP

20

```
10.5 12.6
15.5 16.6
14.0 14.1
17.5 21.0
15.0 16.1
10.5 10.6
19.0 21.0
10.5 13.6
12.5 12.6
11.5 13.6
12.5 15.6
16.0 18.1
13.5 14.6
12.5 17.6
13.0 13.1
18.5 21.0
 9.0 13.1
10.5 11.6
10.5 12.6
18.0 21.0
 0.5
 0.1
```

CHUCTET.OUT

16

```
17 9.0
 1 10.5
18 11.1
```

19 11.7
 8 12.3
 10 12.9
 11 13.5
 13 14.1
 5 15.0
 2 15.6
 12 16.2
 14 16.8
 4 17.5
 7 19.0
 16 19.6
 20 20.2

(Đề ra của bạn Đinh Quang Huy - ĐHKHTN - ĐHQG Hà Nội)

Bài 100/2002 - Mời khách dự tiệc

(Dành cho học sinh THPT)

Công ty trách nhiệm hữu hạn “Vui vẻ” có n cán bộ đánh số từ 1 đến n . Cán bộ i có đánh giá độ vui tính là v_i ($i = 1, 2, \dots, n$). Ngoại trừ Giám đốc Công ty, mỗi cán bộ có 1 thủ trưởng trực tiếp của mình.

Bạn chỉ cần giúp Công ty mời một nhóm cán bộ đến dự dạ tiệc “Vui vẻ” sao cho trong số những người được mời không đồng thời có mặt nhân viên và thủ trưởng trực tiếp và đồng thời tổng đánh giá độ vui tính của những người dự tiệc là lớn nhất.

Giả thiết rằng mỗi một thủ trưởng có không quá 20 cán bộ trực tiếp dưới quyền.

Dữ liệu: Vào từ file văn bản GUEST.INP

- Dòng đầu tiên ghi số cán bộ của Công ty: n ($1 < n < 1001$);
- Dòng thứ i trong số n dòng tiếp theo ghi hai số nguyên dương t_i, v_i ; trong đó t_i là số hiệu của thủ trưởng trực tiếp và v_i là độ vui tính của cán bộ i ($i = 1, 2, \dots, n$). Quy ước $t_i = 0$ nếu i là số hiệu của Giám đốc Công ty.

Kết quả: Ghi ra file văn bản GUEST.OUT

- Dòng đầu tiên ghi hai số m, v ; trong đó m là tổng số cán bộ được mời còn v là tổng độ vui tính của các cán bộ được mời dự tiệc;
- Dòng thứ i trong số m dòng tiếp theo ghi số hiệu của cán bộ được mời thứ i ($i = 1, 2, \dots, m$).

Ví dụ:

GUEST.INP	GUEST.OUT
3	2 7
0 3	1
1 6	3
2 4	

GUEST.INP	GUEST.OUT
7	3 63
0 1	3
1 1	4
1 12	5
2 50	
2 1	
3 1	
3 1	

(Đề ra của bạn Lưu Văn Minh)

Phần II: LỜI GIẢI

Bài 1/1999 - Trò chơi cùng nhau qua cầu

(Dành cho học sinh Tiểu học)

Đáp số: 17 phút. Cách đi như sau:

Lượt 1: 2 + 1 sang, 1 quay về thời gian: 3 phút

Lượt 2: 10 + 5 sang, 2 quay về thời gian: 12 phút

Lượt 3: 2 + 1 sang thời gian: 2 phút

Tổng thời gian: 17 phút

Bài 2/1999 - Tổ chức tham quan

(Dành cho học sinh THCS)

```

Program bai2;
uses crt;
const fi = 'P2.inp';
      fo = 'P2.out';
type _type=array[1..2] of integer;
      mang=array[1..200] of _type;

var f:text;
    d,v:mang;
    m,n:byte;

procedure input;
var i:byte;
begin
  assign(f,fi);
  reset(f);
  readln(f,n,m);
  for i:=1 to n do
  begin
    read(f,d[i,1]);
    d[i,2]:=i;
  end;
  readln(f);
  for i:=1 to m do
  begin
    read(f,v[i,1]);
    v[i,2]:=i;
  end;
  close(f);
end;

procedure sapxetang(var m:mang;n:byte);
var d:_type;
    i,j:byte;
begin
  for i:=1 to n-1 do

```

```

for j:=i+1 to n do
if m[j,1]m[i,1] then
begin
d:=m[j];
m[j]:=m[i];
m[i]:=d;
end;
end;

var i:byte;
tong:integer;
begin
input;
sapxep(m,d,n);
sapxep(m,v,m);
tong:=0;
for i:=1 to n do tong:=tong+v[n-i+1,1]*d[i,1];
for i:=1 to n do v[i,1]:=d[n-i+1,2];
xapxep(m,v,n);
assign(f,fo);
rewrite(f);
writeln(f,tong);
for i:=1 to n do writeln(f,v[i,2]);
close(f);
end.

```

Nhận xét: Chương trình trên sẽ chạy chậm nếu chúng ta mở rộng bài toán (chẳng hạn $n \leq m \leq 8000$). Sau đây là cách giải khác:

```

const
  Inp = 'P2.INP';
  Out = 'P2.OUT';
var
  n, m: Integer;
  Val, Pos: array[1..2, 1..8000] of Integer;
procedure ReadInput;
var
  i: Integer;
  hf: Text;
begin
  Assign(hf, Inp);
  Reset(hf);
  Readln(hf, n, m);
  for i := 1 to n do Read(hf, Val[1, i]);
  Readln(hf);
  for i := 1 to m do Read(hf, Val[2, i]);
  Close(hf);
  for i := 1 to m do
    begin
      Pos[1, i] := i;

```

```

    Pos[2, i] := i;
  end;
end;
procedure QuickSort(t, l, r: Integer);
var
  x, tg, i, j: Integer;
begin
  x := Val[t, (l + r) div 2];
  i := l; j := r;
  repeat
    while Val[t, i] < x do Inc(i);
    while Val[t, j] > x do Dec(j);
    if i <= j then
      begin
        Tg := Val[t, i]; Val[t, i] := Val[t, j]; Val[t, j] := Tg;
        Tg := Pos[t, i]; Pos[t, i] := Pos[t, j]; Pos[t, j] := Tg;
        Inc(i); Dec(j);
      end;
    until i > j;
    if i < r then QuickSort(t, i, r);
    if j > l then QuickSort(t, l, j);
  end;
procedure WriteOutput;
var
  i: Integer;
  Sum: LongInt;
  hf: Text;
begin
  Sum := 0;
  for i := 1 to n do Inc(Sum, Val[1, n - i + 1] * Val[2, i]);
  for i := 1 to n do Val[1, Pos[1, n - i + 1]] := Pos[2, i];
  Assign(hf, Out);
  Rewrite(hf);
  Writeln(hf, Sum);
  for i := 1 to n do Writeln(hf, Val[1, i]);
  Close(hf);
end;
begin
  ReadInput;
  QuickSort(1, 1, n);
  QuickSort(2, 1, m);
  WriteOutput;
end.

```

Bài 3/1999 - Mạng tế bào

(Dành cho học sinh THPT)

```

Program Bai3/1999;
uses crt;
const fi = 'P3.inp';

```

```

fo = 'P3.out';

type mang=array[0..201,0..201] of byte;

var m,n,t:byte;
    s:string;
    a:mang;
    f:text;
    b,c:^mang;

procedure input;
var i,j:byte;
begin
    assign(f,fi);
    reset(f);
    readln(f,m,n,t);
    readln(f,s);
    for i:=1 to m do
    begin
        for j:=1 to n do read(f,a[i,j]);
    end;
    close(f);
    new(b);
    new(c);
end;

procedure hien;
var i,j:byte;
begin
    for i:=1 to m do
    for j:=1 to n do
    begin
        gotoxy(j*2,i);
        write(b^[i,j]);
    end;
end;

procedure trans(ch:char);
var i,j,d:byte;
begin
    fillchar(c^,sizeof(mang),0);
    for i:=1 to m do
    for j:=1 to n do
    begin
        d:=b^[i,j];
        case a[i,j] of
            1:inc(c^[i,j-1],d);
            2:inc(c^[i,j+1],d);
            3:inc(c^[i-1,j],d);

```

```

4:inc(c^[i+1,j],d);
5:begin inc(c^[i-1,j],d);inc(c^[i+1,j],d); end;
6:begin inc(c^[i,j-1],d);inc(c^[i,j+1],d); end;
7:begin inc(c^[i,j-1],d);inc(c^[i-1,j],d); end;
8:begin inc(c^[i,j+1],d);inc(c^[i+1,j],d); end;
end;
end;
if ch<>'X' then b^[1,1]:=ord(ch)-48;
for i:=1 to m do
for j:=1 to n do
if (i<>1) or (j<>1) then b^[i,j]:=byte(c^[i,j]<>0);
hien;
readln;
end;

procedure output;
var i,j:byte;
begin
assign(f,fo);
rewrite(f);
for i:=1 to m do
begin
for j:=1 to n do write(f,' ',b^[i,j]);
writeln(f);
end;
close(f);
end;

var i:byte;
begin
clrscr;
input;
fillchar(b^,sizeof(mang),0);
fillchar(c^,sizeof(mang),0);
for i:=1 to t do trans(s[i]);
output;
end.

```

Bài 4/1999 - Trò chơi bốc sỏi

(Dành cho học sinh Tiểu học)

Huy sẽ là người thắng cuộc. Thật vậy số sỏi ban đầu là 101 là một số có dạng $5k+1$, nghĩa là số nếu chia 5 sẽ còn dư 1. Hoàng phải bốc trước, do số sỏi của Hoàng phải lấy là từ 1 đến 4 do đó sau lượt đi đầu tiên, số sỏi còn lại sẽ lớn hơn 96. Huy sẽ bốc tiếp theo sao cho số sỏi còn lại phải là 96, nghĩa là số dạng $5k+1$. Tương tự như vậy, Huy luôn luôn chủ động được để sau lần bốc của mình số sỏi còn lại là $5k+1$. Lần cuối cùng số sỏi còn lại chỉ là 1 và Hoàng bắt buộc phải bốc viên cuối cùng và ... thua.

Bài toán tổng quát: có thể cho số viên bi là $5k+1$ viên.

Bài 5/1999 - 12 viên bi*(Dành cho học sinh THCS)*

Ta sẽ chỉ ra rằng tồn tại 3 lần cân để chỉ ra được viên bi đặc biệt đó.

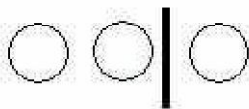


Gọi các viên bi này lần lượt là 1, 2, ..., 12. Trong khi mô tả thuật toán ta dùng ký hiệu

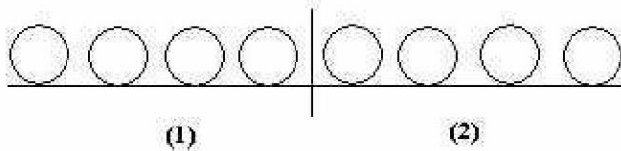
để mô tả quả hòn bi thứ n



để mô tả một hòn bi bất kỳ



Mô tả một phép cân.

Ta gọi viên bi có trọng lượng khác là **đđ**.**I. Lần cân thứ nhất.** Lấy ra 8 hòn bi bất kỳ và chia làm 2 phần để cân:

Có 2 trường hợp xảy ra:

1.1. Cân trên cân bằng. Suy ra viên bi **đđ** (không rõ nặng nhẹ) nằm trong 4 viên bi còn lại (không mang ra cân)**1.2.** Cân trên không cân bằng.**1.2.1.** Nếu (1) nhẹ hơn (2) suy ra hoặc **đđ** là nhẹ nằm trong (1) hoặc **đđ** là nặng nằm trong (2).**1.2.2.** Nếu (1) nặng hơn (2) suy ra hoặc **đđ** là nặng nằm trong (1) hoặc **đđ** là nhẹ nằm trong (2).

Để thấy các trường hợp 1.2.1. và 1.2.2. là tương tự nhau.

Trong mọi trường hợp ta có kết luận **đđ** nằm trong số 8 viên hoặc nhẹ trong 4 hoặc nặng trong 4 còn lại.**II. Xét trường hợp 1.1: Tìm được 4 viên bi chứa **đđ****

Gọi các hòn bi này là 1, 2, 3, 4

Lần cân thứ hai:

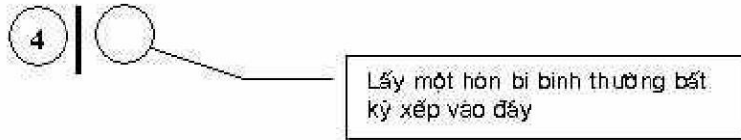
Xét các trường hợp sau:

2.1. Cân thăng bằng. Kết luận: viên bi 4 chính là **đđ**.**2.2.** Trường hợp cân trái nhẹ hơn phải (dấu <). Suy ra hoặc 3 là **đđ** nặng, hoặc 1 hoặc 2 là **đđ** nhẹ.**2.3.** Trường hợp cân trái nặng hơn phải (dấu >). Suy ra hoặc 3 là **đđ** nhẹ, hoặc 1 hoặc 2 là **đđ** nặng.

Để thấy rằng các trường hợp 2.2. và 2.3. là tương tự nhau.

III. Xét trường hợp 2.1: viên bi 4 chính là đđ

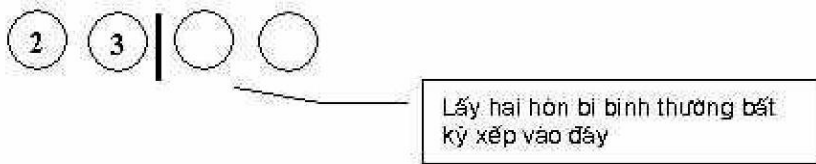
Lần cân thứ ba:



Nếu cân nghiêng < thì 4 là đđ nhẹ, nếu cân nghiêng > thì 4 là đđ nặng.

IV. Xét trường hợp 2.2. Hoặc 3 là đđ nặng, hoặc 1 hoặc 2 là đđ nhẹ.

Lần cân thứ ba:

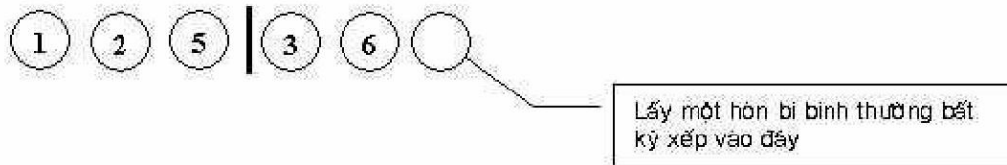


Nếu cân thăng bằng thì ta có 1 là hòn bi đđ nhẹ.

Nếu cân nghiêng > thì ta có 3 là hòn bi đđ nặng.

Nếu cân nghiêng < thì ta có 2 là hòn bi nhẹ.

V. Xét trường hợp 2.3. Hoặc 3 là đđ nhẹ, hoặc 1 hoặc 2 là đđ nặng.



Cách làm tương tự trường hợp 2.2 mô tả trong mục IV ở trên.

VI. Xét trường hợp 1.2.1.

Hoặc đđ là nhẹ trong 1, 2, 3, 4 hoặc đđ là nặng trong 5, 6, 7, 8.

Lần cân thứ hai:

6.1. Trường hợp cân thăng bằng. Suy ra đđ sẽ phải nằm trong 4, 7, 8, và do đó theo giả thiết của trường hợp này ta có hoặc đđ là 4 nhẹ, hoặc đđ là nặng trong 7, 8. Để nhận thấy trường hợp này hoàn toàn tương tự như 2.2. Bước tiếp theo làm tương tự như mô tả trong IV.

6.2. Trường hợp cân nghiêng <, suy ra hoặc đđ là nhẹ rơi vào 1, 2 hoặc đđ là 6 nặng. Trường hợp này cũng hoàn toàn tương tự như 2.2. Bước tiếp theo làm tương tự như mô tả trong IV.

6.3. Trường hợp cân nghiêng >, suy ra hoặc đđ là 5 nặng, hoặc đđ là nhẹ 3.

VII. Xét trường hợp 6.3.

Hoặc đđ là 5 nặng, hoặc đđ là 3 nhẹ.

Lần cân thứ ba:

Nếu cân thăng bằng, suy ra 5 là đđ nặng.

Nếu cân nghiêng < suy ra 3 là đđ nhẹ.

Tất cả các trường hợp của bài toán đã được xem xét.

Sau đây là chương trình chi tiết.

```

Program bai5;
Uses crt;
Const
st1=' nang hon.';
st2=' nhe hon.';
Var i, kq1: integer;
kq2: string;
ch: char;
(* Thủ tục Kq *)
Procedure kq(a: integer; b: string);
Begin
kq1:=a;
kq2:=b;
End;
(* Thủ tục Cân *)
Procedure can(lan: integer; t1, t2, t3, t4, p1, p2, p3, p4: string);
Begin
Writeln('Lần cân thứ', lan, ':');
Writeln;
Writeln(' ', t1, ' ', t2, ' ', t3, ' ', t4, ' ', p1, ' ', p2, ' ', p3, ' ', p4);
Writeln;
Write(' Bên nào nặng hơn? Trái(t)/Phải(p)/ Hay cân bằng(c)');
Repeat
ch:=readkey;
ch:=upcase(ch);
Until (ch in ['P', 'T', 'C']);
Writeln(ch);
Writeln('=====');
End;
(* Thủ tục Play *)
Procedure play;
Begin
Writeln('Có 12 quả cân: 1 2 3 4 5 6 7 8 9 10 11 12');
Writeln('Cho phép bạn chọn ra một quả cân nặng hơn hay nhẹ hơn những quả khác. ');
can(1, '1', '2', '3', '4', '5', '6', '7', '8');
If (ch='T') then {T}
Begin
can(2, '1', '2', '5', '3', '4', '6', ' ');
If (ch='T') then {TT}
Begin
can(3, '1', '6', '7', '8', ' ', ' ');
If ch='T' then kq(1, st1); {TTT}
If ch='P' then kq(6, st2); {TTP}
If ch='C' then kq(2, st1); {TTC}
End
Else If (ch='P') then {TP}

```



```

Begin
can(3, '3', '5', '1', '1', '7', '8', '1', '1');
If ch='T' then kq(3, st1); {TPT}
If ch='P' then kq(5, st2); {TPP}
If ch='C' then kq(4, st1); {TPC}
End
Else If (ch='C') then {TC}
Begin
can(3, '7', '1', '1', '1', '1', '8', '1', '1');
If ch='T' then kq(8, st2); {TCT}
If ch='P' then kq(7, st2); {TCP}
If ch='C' then
Begin
Writeln("Trả lời sai!"); kq2:=st2;
End;
End;
End
Else If (ch='P') then {P}
Begin
can(2, '5', '6', '1', '1', '7', '8', '2', '1');
If (ch='T') then {PT}
Begin
can(3, '5', '2', '1', '1', '3', '4', '1', '1');
If ch='T' then kq(5, st1);
If ch='P' then kq(2, st2);
If ch='C' then kq(6, st1);
End
Else If (ch='P') then {PP}
Begin
can(3, '7', '1', '1', '1', '3', '4', '1', '1');
If ch='T' then kq(7, st1);
If ch='P' then kq(1, st2);
If ch='C' then kq(8, st1);
End
Else If (ch='C') then {PC}
Begin
can(3, '3', '1', '1', '1', '1', '4', '1', '1');
If ch='T' then kq(4, st2);
If ch='P' then kq(3, st2);
If ch='C' then
Begin
Writeln("Trả lời sai !"); kq2:=st2;
End;
End;
End
Else If (ch='C') then {C}
Begin
can(2, '9', '10', '11', '1', '1', '2', '3', '1');
If (ch='T') then
{CT}

```

```

Begin
can(3, '9', ' ', ' ', ' ', '10', ' ', ' ', ' ');
If (ch='T') then kq(9, st1);
If (ch='P') then kq(10, st1);
If (ch='C') then kq(11, st1);
End
Else If (ch='P') then {CP}
Begin
can(3, '9', ' ', ' ', ' ', '10', ' ', ' ', ' ');
If (ch='T') then kq(10, st2);
If (ch='P') then kq(9, st2);
If (ch='C') then kq(11, st2); End
Else If (ch='C') then {CC}
Begin
can(3, '12', ' ', ' ', ' ', '1', ' ', ' ', ' ');
If (ch='T') then kq(12, st1);
If (ch='P') then kq(12, st2);
If (ch='C') then Writeln('Trả lời sai!');
kq1:=12;
End;
End;
End;
(* Chương trình chính*)
Begin
Clrscr;
play;
Writeln(' Quả thứ', kq1, kq2);
Writeln(' Nhấn Enter kết thúc...');
Readln;
End.

```

Bài 6/1999 - Giao điểm các đường thẳng

(Dành cho học sinh THPT)

```

Program Bai6;
(* Tinh so giao diem cua n duong thang 0 trung nhau *)
Uses Crt;
Const
fn = 'P6.INP';
fg = 'P6.OUT';
max = 100;
exp = 0.0001;
Var
a ,b ,c : array[1..max] of real;
n : integer;
sgd : integer;
Procedure Nhap;
Var
f: text;
i: integer;
Begin

```

```

Assign( f ,fn ); Reset( f );
Readln( f ,n );
For i := 1 to n do
Readln( f ,a[i] ,b[i] ,c[i] ); { ax + by = c }
Close( f );
End;
(*-----*)
Procedure Chuanbi;
Begin
sgd := 0;
End;
(*-----*)
Function GiaoDiem( i ,j : integer;Var x ,y : real ) : boolean;
Var
d ,dx ,
dy : real;
Begin
d := a[i] * b[j] - a[j] * b[i];
dx := c[i] * b[j] - c[j] * b[i];
dy := a[i] * c[j] - a[j] * c[i];
If d <> 0 then
begin
x := dx / d;
y := dy / d;
end;
giaodiem := d <> 0;
End;
(*-----*)
Function Giatri( i : integer;x ,y : real ) : real;
Begin
Giatri := a[i] * x + b[i] * y - c[i];
End;
(*-----*)
Function bang( a ,b : real ) : boolean;
Begin
bang := abs( a - b ) <= exp;
End;
(*-----*)
Function Thoaman( i ,j : integer;x ,y : real ) : boolean;
Var
ii: integer;
Begin
Thoaman := false;
For ii := 1 to i - 1 do
If (ii <> j) and bang( giatri( ii ,x ,y ) ,0 ) then
exit;
Thoaman := true;
End;
(*-----*)
Function Catrieng( i : integer ) : integer;

```

```

Var
ii , gt:integer;
x, y : real;
Begin
gt := 0;
For ii := 1 to i do
If giaodiem( i ,ii ,x ,y ) then
If thoaman( i ,ii ,x ,y ) then Inc( gt );
catrieng := gt;
End;
(*-----*)
Procedure Tinhsl;
Var
i : integer;
Begin
For i := 1 to n do
Inc( sgd ,catrieng( i ) );
End;
(*-----*)
Procedure GhiKQ;
Begin
WriteLn(So giao diem cua cac duong thang la: ' ,sgd );
End;
(*-----*)
BEGIN
ClrScr;
Nhap;
Chuanbi;
Tinhsl;
ghiKQ;
END.

```

Bài 7/1999 - Miền mặt phẳng chia bởi các đường thẳng

(Dành cho học sinh THPT)

```

Program Bai7;
(* Tinh so giao diem cua n duong thang ko trung nhau *)
Uses Crt;
Const
fn = 'P7.INP';
fg = 'P7.OUT';
max = 100;
exp = 0.0001;
Var
a ,b ,c : array[1..max] of real;
n : integer;
smien : integer;
Procedure Nhap;
Var
f : text;

```

```

i : integer;
Begin
Assign( f ,fn ); Reset( f );
Readln( f ,n );
For i := 1 to n do
Readln( f ,a[i] ,b[i] ,c[i] ); { ax + by = c }
Close( f );
End;
(*-----*)
Procedure Chuanbi;
Begin
smien := 1;
End;
(*-----*)
Function Giaodiem( i ,j : integer;Var x ,y : real ) : boolean;
Var
d ,dx ,dy :real;
Begin
d := a[i] * b[j] - a[j] * b[i];
dx:= c[i] * b[j] - c[j] * b[i];
dy := a[i] * c[j] - a[j] * c[i];
If d <> 0 then
begin
x := dx / d;
y := dy / d;
end;
Giaodiem := d <> 0;
End;
(*-----*)
Function Giatri( i : integer;x ,y : real ) : real;
Begin
Giatri := a[i] * x + b[i] * y - c[i];
End;
(*-----*)
Function bang( a ,b : real ) : boolean;
Begin
bang := abs( a - b ) <= exp;
End;
(*-----*)
Function Thoaman( i : integer;x ,y : real ) : boolean;
Var
ii : integer;
Begin
Thoaman := false;
For ii := 1 to i - 1 do
If bang( Giatri( ii ,x ,y ) ,0 ) then
exit;
Thoaman := true;
End;
(*-----*)

```

```

Function Cattruoc( i : integer ) : integer;
Var
ii , gt : integer;
x , y : real;
Begin
gt:= 0;
For ii := 1 to i - 1 do
If Giaodiem( i ,ii ,x ,y ) then
If Thoaman( ii ,x ,y ) then Inc( gt );
cattruoc := gt;
End;
(*-----*)
Procedure Tinhslmien;
Var
i : integer;
Begin
For i := 1 to n do
Inc( smien ,cattruoc( i ) + 1 );
End;
(*-----*)
Procedure GhiKQ;
Begin
WriteLn(So mien mat phang duoc chia la: ' ,smien );
End;
(*-----*)
BEGIN
Clrscr;
Nhap;
Chuanbi;
Tinhslmien;
GhiKQ;
END.

```

Bài 8/1999 - Cân táo

(Dành cho học sinh Tiểu học)

Số lần cân ít nhất là 3. Cách cân như sau:

Lần 1: Chia 27 quả táo thành 3 phần, mỗi phần 9 quả. Đặt 2 phần lên 2 đĩa cân. Nếu cân thăng bằng thì quả táo nhẹ nằm ở phần chưa cân, nếu cân lệch thì quả táo nhẹ nằm ở đĩa cân nhẹ hơn. Sau lần cân thứ nhất, ta chọn ra được 9 quả táo trong đó có quả táo nhẹ.

Lần 2: Chia 9 quả táo, chọn được ra thành 3 phần, mỗi phần 3 quả. Đặt 2 phần lên 2 đĩa cân. Nếu cân thăng bằng thì quả táo nhẹ nằm ở phần chưa cân, nếu cân lệch thì quả táo nhẹ nằm ở đĩa cân nhẹ hơn. Sau lần cân thứ 2, ta chọn ra được 3 quả táo trong đó có quả táo nhẹ.

Lần 3: Lấy 2 trong số 3 quả táo chọn đặt lên 2 đĩa cân. Nếu cân thăng bằng thì quả táo nhẹ là quả táo còn lại, nếu cân lệch thì quả táo nhẹ nằm ở đĩa cân nhẹ hơn. Sau ba lần cân ta chọn ra được quả táo nhẹ.

Bài 9/1999 - Bốc diêm*(Dành cho học sinh Tiểu học)*

Nếu số lượng que diêm của mỗi dãy là: 3, 5, 8 thì hai bạn Nga và An bạn nào bốc trước sẽ thắng. Có nhiều cách để người bốc trước sẽ thắng. Giả sử:

- Dãy thứ nhất có 8 que diêm.
- Dãy thứ hai có 5 que diêm.
- Dãy thứ hai có 3 que diêm.

Nếu Nga là người bốc trước để thắng, Nga sẽ làm như sau:

1. Bốc hết 8 que diêm ở dãy đầu tiên. Như vậy còn 2 dãy tổng cộng 8 que. An sẽ phải bốc một số que ở một trong hai dãy này.

2. Trong trường hợp sau khi An bốc số diêm chỉ còn ở trên một dãy, Nga sẽ bốc tất cả số diêm còn lại và sẽ thắng. Nếu sau khi An bốc mà số diêm vẫn còn ở trên hai dãy thì Nga cũng sẽ phải bốc sao cho đưa An vào thế bất lợi: mỗi dãy trong 2 dãy cuối cùng còn đúng một que diêm. Nếu chưa đưa An được vào thế bất lợi thì phải bốc sao cho mình không phải ở thế bất lợi. Chẳng hạn như:

- An bốc 3 que diêm ở dãy thứ 2. Nga sẽ bốc 1 que ở dãy cuối cùng.
- An bốc 1 que diêm tiếp theo cũng ở dãy đó. Nga cũng sẽ bốc 1 que ở dãy thứ 3.
- An bốc 1 que tiếp theo. Khi đó, Nga bốc que diêm cuối cùng và thắng cuộc.

Các bạn cũng có thể thử cho các trường hợp khác.

Bài 10/1999 - Dãy số nguyên*(Dành cho học sinh THCS)*

Dãy đã cho là dãy các số tự nhiên viết liên nhau:

123456789 101112...99 100101102...999 100010011002...9999
10000...

$$9 \times 1 = 9$$

$$90 \times 2 = 180$$

$$900 \times 3 = 2700$$

$$9000 \times 4 = 36000 \dots$$

Ta có nhận xét sau:

- Đoạn thứ 1 có 9 chữ số;
- Đoạn thứ 2 có 180 chữ số;
- Đoạn thứ 3 có 2700 chữ số;
- Đoạn thứ 4 có 36000 chữ số;
- Đoạn thứ 5 có $90000 \times 5 = 450000$ chữ số ...

Với $k = 1000$ ta có: $k = 9 + 180 + 3.270 + 1$.

Do đó, chữ số thứ k là chữ số đầu tiên của số 370, tức là chữ số 3.

Chương trình:

Program Bai10;

Uses crt;

Var k: longInt;

(*-----*)

Function chuso(NN: longInt):char;

```

Var st:string[10];
    dem,M:longInt;
Begin
    dem:=0;
    M:=1;
    Repeat
        str(M,st);
        dem := dem+length(st);
        inc(M);
    Until dem >= NN;
    chuso := st[length(st) - (dem - NN)]
    (*-----*)
BEGIN
    clrscr;;
    write('Nhap k:');
    Readln(k);
    Writeln('Chu so thu', k,'cua day vo han cac so nguyen khong am');
    write('123456789101112... la:', chu so(k));
    Readln;
END.

```

Cách giải khác:

```

var n, Result: LongInt;

procedure ReadInput;
begin
    Write('Ban hay nhap so K: '); Readln(n);
end;

procedure Solution;
var
    i, Sum, Num, Digits: LongInt;
begin
    Sum := 9; Num := 1; Digits := 1;
    while Sum < n do
        begin
            Num := Num * 10; Inc(Digits);
            Inc(Sum, Num * 9 * Digits);
        end;
    Dec(Sum, Num * 9 * Digits); Dec(n, Sum);
    Num := Num + (n - 1) div Digits;
    n := (n - 1) mod Digits + 1;
    for i := 1 to Digits - n do Num := Num div 10;
    Result := Num mod 10;
end;

procedure WriteOutput;
begin
    Writeln('Chu so can tim la: ', Result);
    Readln;
end;
begin

```



```

ReadInput;
Solution;
WriteOutput;
end.

```

Bài 11/1999 - Dãy số Fibonacci

(Dành cho học sinh THCS)

```

{$R+}
const
  Inp = 'P11.INP';
  Out = 'P11.OUT';
  Ind = 46;
var
  n: LongInt;
  Fibo: array[1..Ind] of LongInt;
procedure Init;
var
  i: Integer;
begin
  Fibo[1] := 1; Fibo[2] := 1;
  for i := 3 to Ind do Fibo[i] := Fibo[i - 1] + Fibo[i - 2];
end;
procedure Solution;
var
  i: LongInt;
  hfi, hfo: Text;
begin
  Assign(hfi, Inp);
  Reset(hfi);
  Assign(hfo, Out);
  Rewrite(hfo);
  while not Eof(hfi) do
  begin
    Readln(hfi, n);
    Write(hfo, n, ' = ');
    i := Ind; while Fibo[i] > n do Dec(i);
    Write(hfo, Fibo[i]);
    Dec(n, Fibo[i]);
    while n > 0 do
    begin
      Dec(i);
      if n >= Fibo[i] then
      begin

```

```

    Write(hfo, ' + ', Fibo[i]);
    Dec(n, Fibo[i]);
end;
end;
Writeln(hfo);
end;
Close(hfo);
Close(hfi);
end;
begin
    Init;
    Solution;
end.

```

Bài 12/1999 - N-mino*(Dành cho học sinh THPT)*

```

Program Bai12; {Tinh va ve ra tat ca Mino}
Uses Crt;
Const  fn = 'NMINO.INP';
       fg = 'NMINO.OUT';
       max = 16;
Type   bang = array[0..max+1,0..max+1] of integer;
Var    n : integer;
       lonmin : integer;
       hinh ,hinh1 ,xet ,dd : bang;
       hang ,cot: array[1..max] of integer;
       sl : integer;
       qi,qj : array[1..max*max] of integer;
       sh ,sc :integer;
       hangthieu , cotthieu:integer;
       slch : longint;
       f : text;

```

```

Procedure Nhap;
Var f:text;
Begin
    Assign(f,fn); Reset(f);
    Readln(f,n);
    Close(f);
End;

```

```

Procedure Chuanbi;
Begin
    lonmin:= trunc(sqrt(n));
    If n <> sqr(lonmin) then Inc(lonmin);
    slch := 0;

```

```

End;

Function min2( a ,b : integer ) : integer;
Begin
  If a < b then min2 := a Else min2 := b;
End;

Procedure Taobien( i ,j : integer );
Var ii ,jj : integer;
Begin
  FillChar(dd ,SizeOf(dd),1);
  FillChar(xet,SizeOf(xet),1);
  For ii := 1 to i do
    For jj := 1 to j do
      begin
        dd[ii,jj] := 0;
        xet[ii,jj] := 0;
      end;
  End;

Procedure Ghinhancauhinh;
Var i ,j : integer;
Begin
  Inc(slch);
  Writeln(f,sh ,' ' ,sc);
  For i := 1 to sh do
    begin
      For j := 1 to sc do Write(f,(dd[i,j] mod 2):2);
      Writeln(f)
    end;
  End;

Procedure Quaytrai;
Var hinh1 : bang;
    i ,j : integer;
Begin
  hinh1:= hinh;
  For i := 1 to sh do
    For j := 1 to sc do hinh[i,j] := hinh1[sc-j+1,i];
  End;

Procedure Lathinh;
Var hinh1 : bang;
    i ,j : integer;
Begin
  hinh1:= hinh;
  For i := 1 to sh do
    For j := 1 to sc do hinh[i,j] := hinh1[sh-i+1,sc-j+1];
  End;

```

```

Procedure Daohinh;
Var hinh1 : bang;
    i,j : integer;
Begin
    hinh1 := hinh;
    For i := 1 to sh do
        For j := 1 to sc do hinh[i,j] := hinh1[sh-i+1,j];
    End;

```

```

Function Bethat : boolean;
Var ii,jj :integer;
Begin
    Bethat := false;
    For ii := 1 to sh do
        For jj := 1 to sc do
            If hinh[ii,jj] <> hinh1[ii,jj] then
                begin
                    Bethat:= hinh[ii,jj] < hinh1 [ii,jj];
                    exit;
                end;
        End;
    End;

```

```

Function Behon : boolean;
Begin
    Behon := Bethat;
End;

```

```

Function Xethinhvuong : boolean;
Begin
    Xethinhvuong := false;
    Quaytrai;
    If Behon then exit; Quaytrai;
    If Behon then exit; Quaytrai;
    If Behon then exit; Daohinh;
    If Behon then exit; Quaytrai;
    If Behon then exit; Quaytrai;
    If Behon then exit; Quaytrai;
    If Behon then exit; Xethinhvuong := true;
End;

```

```

Function Xetchunhat : boolean;
Begin
    Xetchunhat := false;
    Lathinh;
    If Behon then exit; Daohinh;
    If Behon then exit; Lathinh;
    If Behon then exit; Xetchunhat := true;
End;

```

```

Procedure Chuyensang( a : bang;Var b : bang );
Var   i,j:integer;
Begin
For i := 1 to sh do
  For j := 1 to sc do b[i,j] := a[i,j] mod 2;
End;

```

```

Procedure Thughinhancauhinh;
Begin
  Chuyensang(dd ,hinh);
  hinh1:= hinh;
  If sh = sc then begin If not Xethinhvuong then exit; end
  Else If not Xetchunhat then exit;
  Ghinhancauhinh;
End;

```

```

Procedure Xetthem( i ,j : integer );
Begin
  Inc(xet[i,j]);
  If xet[i,j] = 1 then
    begin
      Inc(sl);
      qi[sl] := i;
      qj[sl] := j
    end;
End;

```

```

Procedure Xetbot( i ,j : integer );
Begin
  If xet[i,j] = 1 then Dec(sl);
  Dec( xet[i,j] );
End;

```

```

Procedure Themdiem( ii : integer );
Var i ,j : integer;
Begin
  i := qi[ii];
  j := qj[ii];
  dd[i,j] := 1;
  If dd[i,j-1] = 0 then Xetthem(i ,j-1);
  If dd[i,j+1] = 0 then Xetthem(i ,j+1);
  If dd[i-1,j] = 0 then Xetthem(i-1 ,j);
  If dd[i+1,j] = 0 then Xetthem(i+1 ,j);
End;

```

```

Procedure Bodiem( ii : integer );
Var i ,j : integer;
Begin
  i := qi[ii];
  j := qj[ii];

```

```

dd[i,j] := 0;
If dd[i,j-1] = 0 then Xetbot(i,j-1);
If dd[i,j+1] = 0 then Xetbot(i,j+1);
If dd[i-1,j] = 0 then Xetbot(i-1,j);
If dd[i+1,j] = 0 then Xetbot(i+1,j);
End;

```

```

Procedure Xethangcot( ii : integer );
Var i,j :integer;
Begin
  i := qi[ii];
  j := qj[ii];
  Inc(hang[i]);
  If hang[i] = 1 then Dec(hangthieu);
  Inc(cot[j]);
  If cot[j] = 1 then Dec(cotthieu);
End;

```

```

Procedure Xetlaihancot( ii : integer );
Var i,j : integer;
Begin
  i := qi[ii];
  j := qj[ii];
  If hang[i] = 1 then Inc(hangthieu);
  Dec(hang[i]);
  If cot[j] = 1 then Inc(cotthieu);
  Dec(cot[j]);
End;

```

```

Procedure Duyet( i : integer;last : integer );
Var ii :integer;
Begin
  If i > n then
    begin thughinhancauhinh; exit; end;
  For ii := last + 1 to sl do
    begin
      themdiem(ii);
      xethangcot(ii);
      If hangthieu + cotthieu <= n - i then duyetcot(ii);
      Xetlaihancot(ii);
      bodiem(ii);
    end;
End;

```

```

Procedure Duyetcauhinh( i ,j : integer );
Var jj : integer;
Begin
  sh := i;
  sc := j;
  FillChar(hang ,SizeOf(hang),0);

```

```

FillChar(cot,SizeOf(cot),0);
hangthieu := sh;
cotthieu := sc;
taobien(i ,j);
For jj := 1 to j do
  begin
    sl:= 1;
    qi[1] := 1;
    qj[1] := jj;
    duyet(1,0);
    dd[1,jj] := 2;
  end;
End;

Procedure Duyethinhbao;
Var i ,j : integer;
    minj ,maxj : integer;
Begin
For i := lonmin to n do
  begin
    minj := (n-1) div i + 1;
    maxj := min2(n+1-i,i);
    For j := minj to maxj do duyetcauhinh(i,j);
  end;
End;

Procedure Ghicuoi;
Var f : file of char;
    s : string;
    i : integer;
Begin
  str(slch,s);
  Assign(f,fg); reset(f);
  Seek(f,0);
  For i := 1 to length(s) do Write(f,s[i]);
  Close(f);
End;

BEGIN
  Clrscr;
  Assign(f,fg); Rewrite(f);
  Writeln(f, ' ');
  Nhap;
  Chuanbi;
  duyethinhbao;
  Close(f);
  ghicuoi;
END.

```

Bài 13/1999 - Phân hoạch hình chữ nhật

(Dành cho học sinh THPT)

```
{Recommend:m,n<5}
const m=4;n=4;max=m*n;
var
  a: array[1..m,1..n] of byte;
  i1,j1,dem,daxep,tg: integer;
  f: text;
  time: longint absolute $0:$46C;
  save: longint;
{-----}
procedure init;
begin
  for i1:=1 to m do
    for j1:=1 to n do a[i1,j1]:=0;
  dem:=0; daxep:=0; tg:=0;
end;
{-----}
procedure kq;
begin
  for i1:=1 to m do
    begin
      for j1:=1 to n do write(f,a[i1,j1],');
      writeln(f);
    end;
end;
{-----}
procedure try(i,j: integer);
var i2,j2,flag: integer;
begin
  if (daxep=max) then begin kq; writeln(f); tg:=tg+1; end
  else
  begin
    flag:=j;
    while (flag
    if (a[i,flag]<>0) then flag:=flag-1;
    for i2:=i to m do for j2:=j to flag do
    begin
      dem:=dem+1;
      for i1:=i to i2 do for j1:=j to j2 do a[i1,j1]:=dem;
      daxep:=daxep+(i2-i+1)*(j2-j+1);
      i1:=i;j1:=j2;
      while (a[i1,j1]<>0) do
      begin
        j1:=j1+1;
        if j1=n+1 then begin j1:=1; i1:=i1+1; end;
      end;
      try(i1,j1);
      daxep:=daxep-(i2-i+1)*(j2-j+1);
      for i1:=i to i2 do
        for j1:=j to j2 do a[i1,j1]:=0;
```



```

    dem:=dem-1;
  end;
end;
end;
{-----}
BEGIN
  init;
  assign(f,'kq.dat'); rewrite(f);
  save:=time;
  try(1,1);
  write(f,tg);
  close(f);
  write('Time is about:',(time-save)/18.2);
  readln;
END.

```

Bài 14/2000 - Tìm số trang sách của một quyển sách

(Dành cho học sinh Tiểu học)

Để tiện tính toán, ta sẽ đánh số lại quyển sách bằng các số 001, 002, 003, ..., 009, 010, 011, 012, 013, ..., 098, 099, 100, 101, ... tức là mỗi số ghi bằng đúng 3 chữ số. Như vậy ta phải cần thêm $9 \times 2 = 18$ chữ số cho các số trước đây chỉ có 1 chữ số và 90 chữ số cho các số trước đây chỉ có 2 chữ số, tổng cộng ta phải dùng thêm 108 chữ số. Với cách đánh số mới này, ta phải cần tới $1392 + 108 = 1500$ chữ số. Vì mỗi số có đúng 3 chữ số nên có tất cả $1500 : 3 = 500$ số, bắt đầu từ 001. Vậy quyển sách có 500 trang.

Bài 15/2000 - Hội nghị đội viên

(Dành cho học sinh Tiểu học)

Để tiện tính toán, cứ mỗi một cặp bạn trai-bạn gái quen nhau ta sẽ nói lại bằng một sợi dây. Như vậy mỗi bạn sẽ bị "buộc" bởi đúng N sợi dây vì quen với N bạn khác giới. Gọi số bạn trai là T thì tính được số dây nói là $T \times N$. Gọi số bạn gái là G thì tính được số dây nói là $G \times N$. Nhưng vì 2 cách tính cho cùng kết quả là số dây nói nên $T \times N = G \times N$, suy ra $T = G$. Vậy trong hội nghị đó số các bạn trai và các bạn gái là như nhau.

Bài 16/2000 - Chia số

(Dành cho học sinh THCS)

Lập một bảng $2N \times N$ ô. Lần lượt ghi N^2 số 1, 2, 3, ..., $N^2 - 1, N^2$ vào N cột, mỗi cột N số theo cách sau:

1				
2	$N+1$			
3	$N+2$	$2N+1$		
...
N	$2N-1$	$3N-2$...	$(N-1)N+1$
	$2N$	$3N-1$...	$N^2-(N-2)$
		$3N$...	$N^2-(N-3)$
			...	$N^2-(N-4)$
				...