

```

sinh(dmay,d1,p,k);
sinh(g2,d2,p,k);
If (A[d1.hg,d1.cot]='.')and(A[g.hg,g.cot]='.')and(A[d2.hg,d2.cot]='.')then
  begin A[g.hg,g.cot]='x'; HienA(-10,10,-10,10); exit; end;
sinh(g1,g,-p,k);
sinh(dmay,d1,-p,k);
sinh(g2,d2,-p,k);
If (A[d1.hg,d1.cot]='.')and(A[g.hg,g.cot]='.')and(A[d2.hg,d2.cot]='.')then
  begin A[g.hg,g.cot]='x'; HienA(-10,10,-10,10); exit; end;
kt:=false;
End;
Procedure Ngdi;
Begin
  Repeat
    Write('Nhap toa do diem (hang,cot): '); Readln(dng.hg,dng.cot);
  Until
    (dng.hg>=Min)and(dng.hg<=Max)and(dng.cot>=Min)and(dng.cot<=Max)and(A[dng.h
    g,dng.cot]='. ');
    A[dng.hg,dng.cot]='1'; HienA(-10,10,-10,10);
  End;
Function Hgchan:Integer;
Var Hgc:Integer;
Begin
  If dmay.cot<dng.cot then
    Begin
      Hgc:=2;
      If Hgc=hgdi then Begin Hgchan:=Hgc; Exit; End;
    End;
  If dmay.cot>dng.cot then
    Begin
      Hgc:=-2;
      If Hgc=hgdi then Begin Hgchan:=Hgc; Exit; End;
    End;
  If dmay.hg<dng.hg then
    Begin
      Hgc:=-1;
      If Hgc=hgdi then Begin Hgchan:=Hgc; Exit; End;
    End;
  If dmay.hg>dng.hg then
    Begin
      Hgc:=1;
      If Hgc=hgdi then Begin Hgchan:=Hgc; Exit; End;
    End;
  Hgchan:=Hgc;
End;
Procedure Nap(Var Q:Qu; d1:diem; hgdi,k:Integer);
Var h,c:Integer;
    d2:diem;
Begin
  Sinh(d1,Q[cq],hgdi,k);

```

```

End;
Procedure Maydi;
Begin
  Inc(dq);
  if not ok then
    Begin
      If Q[dq].hg<dmay.hg then hgdi:=1
      Else If Q[dq].hg>dmay.hg then hgdi:=-1
        Else If Q[dq].cot<dmay.cot then hgdi:=-2
          Else If Q[dq].cot>dmay.cot then hgdi:=2;
    End;
  dmay:=Q[dq];
  A[q[dq].hg,q[dq].cot]:='x';
  HienA(-10,10,-10,10)
End;
Procedure Process;
Var Hgc,p,i,ntt:Integer;
Begin
  ok:=true; ntt:=0;
  Ngdi;
  Hgc:=Hgchan; Hgdi:=-Hgc;
  Inc(cq); Nap(Q,dmay,hgdi,1); Maydi; Inc(dlt);
  Repeat
    Ngdi; Hgc:=Hgchan;
    If ntt=1 then
      If A[dc1.hg,dc1.cot]='.' then finish(dc1)
      Else finish(dc2);
    If ntt=0 then If (dlt>=3) and (kt(dc1,dc2)) then ntt:=1;
    If (Hgc=Hgdi) then
      If ok then
        Begin
          p:=2 div abs(Hgc);
          For i:=1 to dlt-1 do
            Begin
              Inc(cq); Nap(Q,dmay,p,i); Nap(Qc,Q[cq],-hgdi,i);
              Inc(cq); Nap(Q,dmay,-p,i);Nap(Qc,Q[cq],-hgdi,i);
            End;
          ok:=false;
          dlt:=1;
        End
      Else
        Begin
          hgdi:=-hgdi; Inc(cq); Nap(Q,dmay,hgdi,dlt);
        End;
    If ntt=0 then
      Begin
        If dq=cq then Begin Inc(cq); Nap(Q,dmay,hgdi,1); End;
        If A[Qc[dq].hg,Qc[dq].cot]='.' then finish(Qc[dq]);
        Maydi; Inc(dlt);
      End;
  End;

```

```

Until fin;
End;
BEGIN
Init;
Process;
END.

```

Bài 76/2001 - Đoạn thẳng và hình chữ nhật

(Dành cho học sinh PTTH)

Thuật toán:

- Xét đoạn thẳng cắt với từng cạnh của hình chữ nhật, điều kiện cắt của đoạn thẳng với một đoạn thẳng khác (cạnh của hình chữ nhật) là:

- + Hai đầu của đoạn thẳng khác phía với đoạn thẳng của hình chữ nhật;
- + Hai đầu của đoạn thẳng hình chữ nhật khác phía với đoạn thẳng.

Chương trình:

```

Program Bai76;
const inp= 'input.txt';
      out= 'output.txt';
function cat (xs, ys, xe, ye, xl, yt, xr, yb: real): boolean;
var a, b, x, y: real;
      lg1, lg2: boolean;
Begin
if xs=xe then
begin
lg1:=(xs<xl) or (xs>xr) or ((ys>yt) and (ye>yt)) or ((ys<yb) and (ye<yb));
lg2:=(xs>xl) and (xs<xr) and (ys<yt)and (ye<yt) and (ys>yb) and (ye>yb);
cat:=not (lg1 or lg2);
end
else begin
if ys=ye then
begin
lg1:=((xs<xl) and (xe<xl)) or ((xs>xr) and (xe>xr)) or (ys>yt) or (ys<yb));
lg2:=(xs>xl) and (xe>xl) and (xs<xr)and (xe<xr) and (ys<yt) and (ys>yb);
cat:=not (lg1 or lg2);
end
else begin
cat:=false;
a:=(ys-ye)/(xs-xe);
b:=ys-a*xs;
y:= a*xl+b;
if(y<=yt)and(y>=yb)then cat:= true;
y:= a*xr+b;
if(y<=yt)and(y>=yb)then cat:=true;
x:=(yt-b)/a;
if (x>=xl)and (x<=xr)then cat:=true;
x:=(yb-b)/a;
if (x>=xl)and (x<=xr)then cat:=true;
end;
end;
end;

```

```

procedure xuly;
var n, i: word; xs, ys, xe, ye, xl, yt, xr, yb: real;
fi, fo: text;
Begin
assign(fi, imp); reset (fi);
assign (fo, out); rewrite(fo);
readln(fi, n);
for i:=1 to n do begin
readln (fi, xs, ys, xe, ye, xl, yt, xr, yb);
if cat (xs, ys, xe, ye, xl, yt, xr, yb) then writeln (fo, 'T')
else writeln(fo, 'F');
end;
close (fi);
close (fo);
end;
BEGIN
xuly;
END.

```

(Lời giải của bạn Lê Mạnh Hà - Lớp 10A Tin - Khối PTCTT - ĐHKHTN - ĐHQG Hà Nội)

Bài 77/2001 - Xoá số trên bảng

(Dành cho học sinh Tiểu học)

1. Có thể thực hiện được.

Sau đây là một cách làm cụ thể: ta lần lượt xoá từng nhóm hai số một từ cuối lên: (23 - 22); (21 - 20);; (5 - 4); (3 - 2). Như vậy, sau 11 bước này trên bảng sẽ còn lại 12 số 1. Do đó, ta chỉ việc nhóm 12 số 1 này thành 6 nhóm có hiệu bằng 0. Khi đó, trên bảng sẽ chỉ còn lại toàn số 0.

2. Nếu thay 23 số bằng 25 số thì bài toán trên sẽ không thực hiện được.

Giải thích:

Ta có tổng các số từ 1 đến 25 = $(1 + 25) \times 25 : 2$ sẽ là một số lẻ.

Giả sử, khi xoá đi hai số bất kỳ thì tổng các số trên bảng sẽ giảm đi là: $(a + b) - (a - b) = 2b =$ một số chẵn.

Như vậy, sau một số bước xoá hai số bất kỳ thì tổng các số trên bảng vẫn còn lại là một số lẻ (số lẻ - số chẵn = số lẻ) và do đó trên bảng sẽ không phải là còn toàn số 0.

Bài 78/2001 - Cà rốt và những chú thỏ

(Dành cho học sinh Tiểu học)

Chú thỏ có thể ăn được nhiều nhất 120 củ cà rốt. Đường đi của chú thỏ như sau:

14->12->13->14->13->16->15->10->13

Do đó, số củ cà rốt chú thỏ ăn được khi đi theo đường này là:

$$14 + 12 + 13 + 14 + 13 + 16 + 15 + 10 + 13 = 120 \text{ (củ)}$$

Bài 79/2001 - Về một ma trận số

(Dành cho học sinh THCS)

Bài này có rất nhiều nghiệm, để liệt kê tất cả các nghiệm thì phải sử dụng thuật toán duyệt. Do không gian tìm kiếm là cực kì lớn nên nếu duyệt tầm thường thì không thể giải được, thậm chí còn không ra nghiệm nào cả. Vì vậy bài giải này duyệt bằng cách

xây dựng một mảng ban đầu thoả mãn tính chất: dùng đúng 10 số 0, 10 số 1, ..., 10 số 9 và mỗi dòng không có quá 4 số khác nhau. Sau đó bằng cách hoán vị vòng các dòng để thoả mãn tính chất của đề bài.

Chọn mảng ban đầu như thế giảm đi rất nhiều khả năng và cũng làm mất đi rất nhiều nghiệm. Mảng ban đầu có thể có rất nhiều cách chọn, số nghiệm tìm ra phụ thuộc rất nhiều vào cách chọn này.

Ví dụ có thể chọn mảng ban đầu là:

(0,0,1,1,2,2,2,3,3,3)

(1,1,2,2,3,3,3,4,4,4)

(2,2,3,3,4,4,4,5,5,5)

(3,3,4,4,5,5,5,6,6,6)

(4,4,5,5,6,6,6,7,7,7)

(5,5,6,6,7,7,7,8,8,8)

(6,6,7,7,8,8,8,9,9,9)

(7,7,8,8,9,9,9,0,0,0)

(8,8,9,9,0,0,0,1,1,1)

(9,9,0,0,1,1,1,2,2,2)

Vì số nghiệm rất nhiều nên ta muốn ghi ra bao nhiêu nghiệm thì thay đổi biến sn để thay đổi số nghiệm cần ghi ra. Bài giải này in ra 100 nghiệm.

Các bạn chú ý rằng nếu có 1 bảng thoả mãn tính chất của bài thì tráo 2 dòng hoặc tráo 2 cột bất kì với nhau, hoặc quay 90° bảng ta có thể có các bảng cũng thoả mãn.

{A+,B-,D+,E+,F-,G-,I+,L+,N-,O-,P-,Q+,R+,S+,T-,V+,X+}

{M 65384,0,655360}

uses crt;

```
type    MG      = array[1..10,1..10]of integer;
        mglc    = array[1..10]of integer;
```

```
const   N       =10;
        p       = 4;
        sn      =100; {số nghiệm muốn ghi ra}
        fo      ='out.txt';
        h       :MG= {một cách chọn khác}
                ((0,0,0,1,1,1,2,2,2,3),
                 (1,1,1,2,2,2,3,3,3,4),
                 (2,2,2,3,3,3,4,4,4,5),
                 (3,3,3,4,4,4,5,5,5,6),
                 (4,4,4,5,5,5,6,6,6,7),
                 (5,5,5,6,6,6,7,7,7,8),
                 (6,6,6,7,7,7,8,8,8,9),
                 (7,7,7,8,8,8,9,9,9,0),
                 (8,8,8,9,9,9,0,0,0,1),
                 (9,9,9,0,0,0,1,1,1,2));
```

```
var     a,dx    : MG;
        lap     : mglc;
        dem     : longint;
        f       : text;
```

```
procedure init;
var k :integer;
```

```

begin
  dem:=0;
  a:=h;
  fillchar(dx,sizeof(dx),0);
  fillchar(lap,sizeof(lap),0);
  for k:=1 to N do lap[k]:=1;
  for k:=1 to N do dx[k,a[1,k]+1]:=1;
end;

```

```

procedure ghikq(w:mg);
var i,j,ds:integer;
begin
  inc(dem);
  writeln('***** :',dem,'*****');
  writeln(f,'***** :',dem,'*****');
  for i:=1 to N do
    begin
      for j:=1 to N do
        begin
          write(w[i,j]:2);
          write(f,w[i,j]:2);
        end;
      writeln;writeln(f);
    end;
end;

```

```

function doi(k:integer):integer;
begin
  if k mod N=0 then doi:=N
  else doi:=k mod N;
end;

```

```

procedure try(k:byte;w:MG);
var i,j :byte;
    luu :mg1c;
    ldx :mg;
    ok :boolean;
begin
  luu:=lap;ldx:=dx;
  for i:=1 to N do
    begin
      lap:=luu;dx:=ldx;
      for j:=1 to N do w[k,j]:=a[k,doi(i+j-1)];

      ok:=true;
      for j:=1 to N do
        begin
          inc(lap[j],1-dx[j,w[k,j]+1]);
          dx[j,w[k,j]+1]:=1;
          if lap[j]>4 then

```

```

begin
  ok:=false;
  break;
end;
end;

if ok then
begin
  if k=N then
    ghikq(w)
  else try(k+1,w);
  end;
  if dem=sn then exit;
end;
lap:=luu;dx:=ldx;
end;

```

BEGIN

```

clrscr;
init;
assign(f,fo);
rewrite(f);
try(2,a);
close(f);

```

END.

(Lời giải của Vũ Anh Quân)

Bài 80/2001 - Xếp số 1 trên lưới

(Dành cho học sinh THCS)

Bài toán có rất nhiều nghiệm, để liệt kê các nghiệm thì ta phải sử dụng thuật toán duyệt. Song duyệt thì rất lớn, mặt khác để ra được một cách điền thoả mãn thì không đơn giản chút nào (thời gian chạy sẽ rất lâu, thậm chí còn có thể bế tắc). Bài giải này duyệt theo một hướng tham lam có thể hiện ra được khá nhiều cách điền thoả mãn, tuy nhiên hướng giải này không hiện ra hết tất cả các nghiệm.

Hướng duyệt tham lam:

+ Mỗi dòng, mỗi cột có ít nhất một số 1.

+ Chia ma trận 10×10 thành 4 ma trận 5×5 , mỗi ma trận 5×5 này sẽ được điền 4 số 1.

Cách kiểm tra tốt một ma trận sau khi điền có thoả mãn tính chất của bài không?

Duyệt cách chọn 5 hàng bất kì rồi xoá các số ở hàng đó, sau khi xoá xong ta tìm cách xoá 5 cột. Nếu sau khi xoá hàng xong mà cột nào còn số 1 thì phải xoá cột đó.

Nếu trong tất cả các cách xoá hàng, cột như vậy đều không xoá hết được thì bảng đó thoả mãn tính chất của bài.

Chương trình sau hiện ra 100 nghiệm.

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

```
{M 16384,0,655360}
```

```
uses crt;
```

```

const   N =10;
        p  =16;
        sn =100; {số nghiệm muốn hiện ra}
        fo ='output.txt';

```

```

type    MG =array[1..5,1..5] of byte;
var     a : array[1..N,1..N] of integer;
        w : array[1..600] of MG;
        d : array[1..5] of integer;
        c,dong,cc,ddd : array[0..N] of integer;
        ok : boolean;
        dem,sl : longint;
        s : MG;
        f : text;
procedure nap;
var i,j,k : integer;
begin
  for i:=1 to 5 do
    begin
      k:=0;
      inc(dem);
      for j:=1 to 5 do
        if i<>j then
          begin
            inc(k);
            w[dem,j]:=s[k];
          end;
      end;
    end;
end;

procedure try(i:byte);
var j :byte;
begin
  for j:=1 to 5 do
    if d[j]=0 then
      begin
        s[i,j]:=1;
        d[j]:=1;
        if i=4 then nap
        else try(i+1);
        d[j]:=0;
        s[i,j]:=0;
      end;
    end;
end;

procedure kiemtra;
var i,j,use,k :integer;
begin
  cc:=c;
  for i:=1 to 5 do
    for j:=1 to N do dec(cc[j],a[dong[i],j]);
  use:=0;
  for k:=1 to N do inc(use,ord(cc[k]>0));
  if use<=5 then ok:=false;
end;

procedure thu(i:integer);

```



```

var j :integer;
begin
  for j:=dong[i-1]+1 to N-5+i do
    begin
      dong[i]:=j;
      if i=5 then kiemtra
      else thu(i+1);
      if ok=false then exit;
    end;
  end;
procedure lam;
var i,j,x,y,u,v,k :integer;
begin
  for i:=1 to dem do
    for j:=dem downto 1 do
      for x:=1 to dem do
        for y:=dem downto 1 do
          begin
            for u:=1 to 5 do
              for v:=1 to 5 do a[u,v]:=w[i,u,v];
            for u:=1 to 5 do
              for v:=1 to 5 do a[u,5+v]:=w[j,u,v];
            for u:=1 to 5 do
              for v:=1 to 5 do a[5+u,v]:=w[x,u,v];
            for u:=1 to 5 do
              for v:=1 to 5 do a[5+u,5+v]:=w[y,u,v];

            fillchar(c,sizeof(c),0);
            fillchar(ddd,sizeof(ddd),0);
            fillchar(dong,sizeof(dong),0);
            for u:=1 to N do
              for v:=1 to N do
                begin
                  inc(c[v],a[u,v]);
                  inc(ddd[u],a[u,v]);
                end;
            ok:=true;
            for k:=1 to N do
              if (c[k]=0)or(ddd[k]=0) then ok:=false;
            if ok then thu(1);
            if ok then
              begin
                inc(sl);
                writeln('*****:',sl,'*****');
                writeln(f,'*****:',sl,'*****');
                for u:=1 to N do
                  begin
                    for v:=1 to N do
                      begin
                        write(a[u,v],#32);

```

```

        write(f,a[u,v],#32);
        end;
        writeln;writeln(f);
        end;
        if sn=sl then exit;
        end;
        end;
end;
BEGIN
  clrscr;
  fillchar(d,sizeof(d),0);
  fillchar(w,sizeof(w),0);
  fillchar(s,sizeof(s),0);
  dem:=0;sl:=0;
  try(1);
  assign(f,fo);
  rewrite(f);
  lam;
  close(f);
END.

```

(Lời giải của Đỗ Đức Đông)

Bài 81/2001 - Dãy nghịch thế

(Dành cho học sinh PTTH)

```

Program day_nghich_the;
uses crt;
const   fn = 'nghich.inp';
        gn = 'nghich.out';
        nmax=10000;
var     f,g:text;
        n,i,j,dem:0..nmax;
        a,b,luu:array[1..nmax] of 0..nmax;
procedure nhap;
begin
  fillchar(a,sizeof(a),0); b:=a;
  assign(f,fn); reset(f);
  readln(f,n);
  for i:=1 to n do read(f,a[i]); write(f);
  for i:=1 to n do read(f,b[i]);
  close(f);
end;
procedure tim_b;
begin
  fillchar(luu,sizeof(luu),0);
  for i:=1 to n do
  begin
    dem:=0;
    for j:=i-1 downto 1 do
      if a[i]<a[j] then inc(dem);

```

```

    luu[a[i]]:=dem;
  end;
  for i:=1 to n do write(g,luu[i]:2);
  writeln(g); writeln(g);
end;
procedure tim_a;
begin
  fillchar(luu,sizeof(luu),0);
  for i:=1 to n do
    if b[i]>n-i then exit else
      begin
        j:=0;
        dem:=0;
        repeat
          inc(dem);
          if luu[dem]=0 then j:=j+1;
        until j>b[i];
        luu[dem]:=i;
      end;
  for i:=1 to n do write(g,luu[i]:2);
end;
BEGIN
  nhap;
  assign(g,gn);rewrite(g);
  tim_b;
  tim_a;
  close(g);
END.

```

(Lời giải của bạn Lê Thị Thu Thủy - Lớp 11A2 PTTH chuyên Vĩnh Phúc - thị xã Vĩnh Yên - tỉnh Vĩnh Phúc)

Bài 82/2001 - Gặp gỡ

(Dành cho học sinh PTTH)

Bài này có thể giải dễ dàng nhờ nhận xét sau:

- Nếu k robot ở các vị trí mà tổng tọa độ của chúng $(x+y)$ có tính chẵn lẻ khác nhau thì chúng không bao giờ gặp nhau (vì chúng luôn luôn di chuyển, không có robot đứng yên). Như vậy, sau khi loại trường hợp trên, gọi $A[t, i, j]$ là số bước di chuyển ít nhất để robot t di chuyển từ vị trí ban đầu đến ô (i, j) . Khi đó, số bước di chuyển ít nhất mà k robot phải di chuyển để gặp nhau là:

$\text{Min} (\max(A(t, i, j))$ với $1 \leq t \leq k, 1 \leq i \leq M, 1 \leq j \leq N$. Loang ngược lại, ta có đường đi của những robot này.

Cài đặt chương trình:

{ \$A+,B-,D+,E+,F-,G-,I+,L+,N-,O-,P-,Q-,R-,S+,T-,V+,X+,Y+ }

{ \$M 16384,0,655360 }

Program MEET;

Uses crt;

Type point = record

 x,y:integer;

End;

```

Const P:array[1..4,1..2] of integer=((0,1),(0,-1),(-1,0),(1,0));
      Q:string='LRDU';
      inp = 'MEET.INP';
      out = 'MEET.OUT';
Var v: array[1..10] of point;
    A: array[1..10,0..51,0..51] of integer;
    B: array[0..51,0..51] of byte;
    t: array[0..1,1..750] of point;
    M,N,K,c,d,e,g,h,l,i,j,Min,Max:integer;
    s,st:string;
    f:text;
Procedure NoSolution;
Begin
  Write(' # ');Readln;Halt;
End;
Procedure Input;
Begin
  Assign(f,inp);Reset(f);
  Readln(f,m,n,k);
  If k>0 then
    Begin
      Readln(f,v[1].x,v[1].y);
      e:=(v[1].x+v[1].y) mod 2;
    End;
  For c:=2 to k do
    Begin
      Read(f,v[c].x,v[c].y);
      If (v[c].x+v[c].y) mod 2<>e then NoSolution;
    End;
  Fillechar(b,sizeof(b),1);
  For c:=1 to m do
    For d:=1 to n do read(f,B[c,d]);
  Close(f);
End;
Procedure Solve;
Var Stop:boolean;
    z:array[0..1] of integer;
Begin
  For c:=0 to m+1 do
    For d:=0 to n+1 do
      If b[c,d]=0 then
        For e:=1 to k do a[e,c,d]:=MaxInt else
        For e:=1 to k do a[e,c,d]:=-1;
  For c:=1 to k do
    Begin
      l:=1;g:=0;h:=1;z[0]:=1;z[1]:=0;
      t[0,1]:=v[c];a[c,v[c].x,v[c].y]:=0;
      Stop:=false;
      While not Stop do
        Begin

```

```

    Stop:=true;
    For d:=1 to z[g] do
      For e:=1 to 4 do
        Begin
          i:=P[e,1]+t[g,d].x;
          j:=P[e,2]+t[g,d].y;
          If a[c,i,j]>1 then
            Begin
              a[c,i,j]:=1;inc(z[h]);
              t[h,z[h]].x:=i;
              t[h,z[h]].y:=j;
              Stop:=false;
            End;
          End;
        l:=l+1;g:=1-g;h:=1-h;z[h]:=0;
      End;
    End;
  Min:=MaxInt;
  For c:=1 to m do
    For d:=1 to n do
      If b[c,d]<>1 then
        Begin
          max:=a[1,c,d];
          For e:=2 to k do
            If Max<a[e,c,d] then Max:=a[e,c,d];
          If Min>Max then
            Begin
              Min:=Max;
              i:=c;j:=d;
            End;
          End;
        If Min=MaxInt then NoSolution;
        Assign(f,out);Rewrite(f);
        For e:=1 to k do
          Begin
            c:=i;d:=j;s:="";
            While A[e,c,d]>0 do
              Begin
                l:=1;
                While a[e,c+P[l,1],d+P[l,2]]+1<>a[e,c,d] do l:=l+1;
                s:=Q[l]+s;
                c:=c+P[l,1];d:=d+P[l,2];
              End;
            l:=l-1+2*(1 mod 2);
            st:=s[1]+Q[l];
            For g:=1 to (min-a[e,i,j]) div 2 do s:=st+s;
            Writeln(f,s);
          End;
        Close(f);
      End;
    End;
  End;

```

BEGIN

```

  Clrscr;
  Input;
  Solve;
  Write('Complete - Open file ',out,' to view the result!');
  Readln

```

END.

(Lời giải của bạn Vũ Lê An - Lớp 12T2 - Lê Khiết - Quảng Ngãi)

Nhân xét: Bài làm của bạn Vũ Lê An phân kết quả còn thiếu trường hợp. Sau đây là một cách cài đặt khác song thuật toán cũng giống với Vũ Lê An.

Mở rộng bài toán: Cho một đồ thị gồm N đỉnh, có k con robot ở k đỉnh V_1, V_2, \dots, V_k . Sau mỗi đơn vị thời gian tất cả các con robot đều phải chuyển động sang các đỉnh kề với đỉnh nó đang đứng. Hãy tìm cách di chuyển các con robot để chúng gặp nhau tại một điểm.

a. Trong đồ thị vô hướng

b. Trong đồ thị có hướng ($k = 2$ - Đề thi chọn đội tuyển Quốc gia)

{A+,B-,D+,E+,F-,G-,I+,L+,N-,O-,P-,Q+,R+,S+,T-,V+,X+}

{SM 65384,0,655360}

program Bai82_gap_go; {Author : Đỗ Đức Đông}

uses crt;

```

const   max           =50;
        max_robot     =10;
        fi            ='meet.inp';
        fo            ='meet.out';
        tx            :array[1..4]of integer=(0,-1,1,0);
        ty            :array[1..4]of integer=(-1,0,0,1);
        h            :string='LUDR';

var     a              :array[1..max,1..max]of byte;
        robot         :array[1..max_robot,1..2]of byte;
        l             :array[1..max,1..max,1..max_robot]of integer;
        q             :array[1..max*max,1..2]of byte;
        dau,cuoi,m,n,r :integer;
        best,mx,my    :integer;
        ok            :boolean;

```

procedure docf;

```

var f :text;
    k,i,j:integer;
begin
  assign(f,fi);
  reset(f);
  readln(f,m,n,r);
  for k:=1 to r do readln(f,robot[k,1],robot[k,2]);
  for i:=1 to m do
    for j:=1 to n do read(f,a[i,j]);
  close(f);
end;

```

```

procedure loang(k:integer);
var x,y,s,u,v :integer;
begin
  fillchar(q,sizeof(q),0);
  dau:=1;cuoi:=1;
  q[1,1]:=robot[k,1];
  q[1,2]:=robot[k,2];
  l[robot[k,1],robot[k,2],k]:=1;
  while dau<=cuoi do
    begin
      x:=q[dau,1];y:=q[dau,2];
      for s:=1 to 4 do
        begin
          u:=x+tx[s];
          v:=y+ty[s];
          if (u>0)and(v>0)and(u<=m)and(v<=n)and(a[u,v]=0)and(l[u,v,k]=0) then
            begin
              inc(cuoi);q[cuoi,1]:=u;q[cuoi,2]:=v;
              l[u,v,k]:=l[x,y,k]+1;
            end;
          end;
        inc(dau);
      end;
    end;
end;

```

```

procedure lam;
var k,i,j :integer;
    meet :boolean;
begin
  fillchar(l,sizeof(l),0);
  ok:=true;
  for k:=2 to r do
    if (robot[1,1]+robot[1,2]+robot[k,1]+robot[k,2]) mod 2=1 then ok:=false;

  if ok then
    begin
      best:=maxint;
      for k:=1 to r do loang(k);
      for i:=1 to m do
        for j:=1 to n do
          begin
            meet:=true;
            for k:=1 to r do meet:=meet and (l[i,j,k]>0) and (l[i,j,k]<best);

            if meet then
              begin
                best:=0;
                for k:=1 to r do
                  if l[i,j,k]>best then
                    begin

```

```

        best:=l[i,j,k];
        mx:=i,my:=j;
    end;
    end;
    end;
    ok:=best<maxint;
end;
end;

procedure ghif;
var f      :text;
    k,kk   :byte;
    lap    :string;

procedure viet(x,y:byte);
var u,v,s :byte;
begin
    for s:=1 to 4 do
        begin
            u:=x+tx[s];
            v:=y+ty[s];
            if (u>0)and(v>0)and(u<=m)and(v<=n)and(l[u,v,k]=l[x,y,k]-1) then
                begin
                    if l[u,v,k]>1 then viet(u,v);
                    write(f,h[5-s]);
                    break;
                end;
            end;
        end;
    end;

begin
    assign(f,fo);
    rewrite(f);
    if ok=false then write(f,'#')
    else
        begin
            for k:=1 to 4 do
                if (mx+tx[k]>0)and(my+ty[k]>0)and(mx+tx[k]<=m)and(my+ty[k]<=n) then
                    if (a[mx+tx[k],my+ty[k]]=0) then kk:=k;
                lap:=h[kk]+h[5-kk];

            for k:=1 to r do
                begin
                    if l[mx,my,k]>1 then viet(mx,my);
                    for kk:=1 to (best-l[mx,my,k]) div 2 do write(f,lap);
                    writeln(f);
                end;
            end;
        close(f);
    end;
end;

```



```

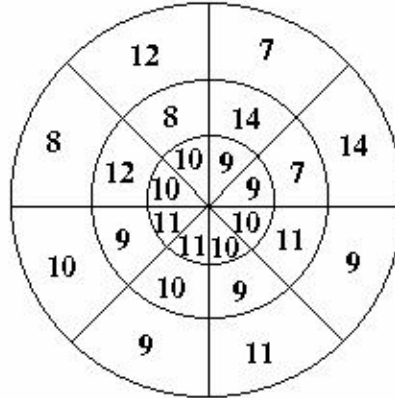
BEGIN
  docf;
  lam;
  ghif;
END.

```

Bài 83/2001 - Các đường tròn đồng tâm

(Dành cho học sinh Tiểu học)

Đáp số: Các số được điền như sau:



Bài 84/2001 - Cùng một tích

(Dành cho học sinh THCS và THPT)

Thuật toán: Gọi số lượng số $x_i = 1$ là a , số lượng số $x_i = -1$ là b , số lượng số $x_i = 0$ là c .
Ta có: $a + b + c = N$.

Với mỗi giá trị c khác nhau ta có tương ứng một nghiệm. Nên số nghiệm bằng số giá trị mà c có thể nhận được. Nếu duyệt theo biến c thì có rất nhiều khả năng nên thay vì duyệt theo biến c ta duyệt theo a và b . Vai trò của các số bằng 1 và các số bằng -1 là như nhau nên ta có thể giả sử số lượng số bằng 1 lớn hơn số lượng bằng -1 ($a \geq b$).

Vậy $\sum x_i = a - b$ và $\sum x_i^2 = a + b$ ($i = 1, \dots, N$)

$\sum x_i x_j = P$ ($i = 1, \dots, N; j = 1, \dots, N; i < j$) suy ra $P = 2 * \sum x_i x_j$ ($i = 1, \dots, N - 1; j = 1, \dots, N; i < j$)

Ta có phương trình: $(a+b) + p = (a-b)^2$

suy ra $0 \leq (a-b) \leq \sqrt{a+b+p} \leq \sqrt{N+p} < [\sqrt{2*10^{10}}] = 44721$.

Vậy ứng với mỗi giá trị $(a-b)$ ta có một giá trị $(a+b)$ và một giá trị c . Lần lượt thử với từng giá trị của $(a-b)$ rồi kiểm tra xem a , b và c thoả mãn các tính chất không?

{ \$A+,B-,D+,E+,F-,G-,I+,L+,N-,O-,P-,Q+,R+,S+,T-,V+,X+,Y+ }

{ \$M 16384,0,655360 }

uses crt;

```

const fi  = 'input.txt';
      fo  = 'output.txt';

```

```

var n,p, h :longint;
    dem :longint;
    t :real;

```

```

procedure docf;

```

```

var f :text;

```

```

begin

```

```

    assign(f,fi);
    reset(f);
    read(f,n,p);
    close(f);
    dem:=0;
end;
procedure lam;
var can :longint;
begin
    can:=trunc(sqrt(2*n));
    for h:=0 to can do
        begin
            t:=h;
            t:=sqrt(t)-p;
            if (t>=h)and(t<=n) then inc(dem);
        end;
    end;
end;
procedure ghif;
var f :text;
begin
    assign(f,fo);
    rewrite(f);
    writeln(f,dem);
    close(f);
end;
BEGIN
    docf;
    if p mod 2=0 then lam;
    ghif;
END.

```

(Lời giải của Đỗ Đức Đông)

Bài 85/2001 - Biến đổi 0 - 1

(Dành cho học sinh THPT)

Thuật toán: Bài này sử dụng thuật toán duyệt nhưng có một vài chú ý sau:

- Với 1 ô ta chỉ tác động nhiều nhất một lần.
- Thứ tự tác động là không quan trọng.
- Với một ô có nhiều nhất 5 ô ảnh hưởng được tới nó, vì vậy nếu với một ô ta biết 4 ô ảnh hưởng của nó có được tác động hay không thì ô còn lại ta sẽ biết là có nên tác động hay không tác động.

Từ các chú ý trên ta sẽ duyệt một dòng 1 (hoặc một cột 1) được tác động như thế nào khi đó các ô ở dòng 1 (hoặc cột 1) sẽ chỉ còn 1 ô ảnh hưởng tới nó. Ta sẽ biết được rằng các ô dòng 2 (hoặc cột 2) cũng sẽ được tác động như thế nào, cứ như vậy cho các dòng tiếp theo.

Bài sẽ phải duyệt 2N nếu duyệt theo dòng 1 (2M nếu duyệt theo cột 1) vì vậy để giảm độ phức tạp của bài bạn nên chọn duyệt theo chiều nào tùy thuộc vào M,N.

{A+,B-,D+,E+,F-,G-,I+,L+,N-,O-,P-,Q-,R+,S+,T-,V+,X+}

{M 16384,0,655360}

uses crt;

```

const   max =100;
        fi  ='biendoi.inp';
        fo  ='biendoi.out';
        tx : array[0..4]of integer=(0,0,-1,0,1);
        ty : array[0..4]of integer=(0,-1,0,1,0);
type    mg = array[1..max,1..max]of byte;
var     a,b,td,lkq,c:mg;
        m,n,dem,best:integer;
procedure docf;
var     f :text;
        i,j :byte;
begin
    assign(f,fi);
    reset(f);
    readln(f,m,n);
    for i:=1 to m do
        for j:=1 to n do read(f,a[i,j]);
    for i:=1 to m do
        for j:=1 to n do read(f,b[i,j]);
    close(f);
end;
procedure tacdong(i,j:byte);
var     u,v,k :integer;
begin
    for k:=0 to 4 do
        begin
            u:=i+tx[k];
            v:=j+ty[k];
            if (u>0)and(v>0)and(u<=m)and(v<=n) then a[u,v]:=1-a[u,v];
        end;
        inc(dem);
    end;
procedure process;
var     i,j,k :byte;
        w : mg;
begin
    c:=a;dem:=0;w:=td;
    for i:=1 to n do
        if td[1,i]=1 then tacdong(1,i);
    for i:=2 to m do
        for j:=1 to n do
            if a[i-1,j]<>b[i-1,j] then
                begin
                    tacdong(i,j);
                    td[i,j]:=1;
                end;
        for k:=1 to n do
            if a[m,k]<>b[m,k] then begin a:=c;td:=w;exit;end;
        if dem<best then
            begin

```

```

    best:=dem;
    lkq:=td;
  end;
  a:=c;td:=w;
end;
procedure try(i:byte);
var j :byte;
begin
  for j:=0 to 1 do
    begin
      td[1,i]:=j;
      if i=n then process
        else try(i+1);
      end;
    end;
  end;
procedure ghif;
var f :text;
    i,j :integer;
begin
  assign(f,fo);
  rewrite(f);
  if best<>maxint then
    begin
      writeln(f,best);
      for i:=1 to m do
        for j:=1 to n do
          if lkq[i,j]=1 then writeln(f,i,#32,j);
        end
      else writeln(f,'No solution');
      close(f);
    end;
end;
begin
  clrscr;
  best:=maxint;
  docf;
  try(1);
  ghif;
end.

```

(Lời giải của Đinh Quang Huy)

Bài 86/2001 - Dãy số tự nhiên logic

(Dành cho học sinh Tiểu học)

Số đầu và số cuối cần tìm của dãy số logic đã cho là: 10 và 24.

Giải thích: dãy số đó là dãy các số tự nhiên liên tiếp không nguyên tố.

Bài 87/2001 - Ghi các số trên bảng

(Dành cho học sinh THCS)

Procedure bai87;

```

uses crt;
var d, N:integer;
begin
  clrscr;
  write('Nhap so nguyen duong N: '); readln(N);
  repeat
    if N mod 2 = 0 then N:= div 2 else N:=N-1;
    d:=d+1;
  until N=0;
  write('So lan ghi so len bang: ', d);
  readln;
End.

```

(Lời giải của bạn Cao Le Thang Long)

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

(Dành cho học sinh THCS và THPT)

Thuật toán: mảng $a[0..9]$ lưu kết quả, $t[i]$ là số các chữ số i trong a . Theo bài ta có thể suy ra: $a[0] + a[1] + \dots + a[9] = \text{số các chữ số } 0 + \text{số các chữ số } 1 + \dots + \text{số các chữ số } 9 = 10$. Như vậy, ta dùng phép sinh đệ quy có nhánh cận để giải bài toán: ở mỗi bước sinh $a[i]$, ta tính tổng các chữ số $a[0]..a[i]$ (lưu vào biến s), nếu $s > 10$ thì không sinh tiếp nữa. Sau đây là toàn bộ chương trình:

```

Procedure bai88;
const fo='bai88.out';
var a,t:array[0..9] of integer;
    i,s:integer;
    f:text;
procedure save;
var i:integer;
begin
  for i:=0 to 9 do if a[i] <> t[i] then exit;
  for i:=0 to 9 do write(f,a[i]); writeln(f);
end;
procedure try(i:integer);
var j:integer;
begin
  for j:= 0 to 9 do
    if ((i<j) or ((i>=j) and (t[j] + 1 <=a[j]))) and (s<=10) then
      begin
        a[i]:=j;
        inc(t[j]);
        s:=s+j;
        if i<9 then try(i+1) else save;
        dec(t[j]);
        s:=s-j;
      end;
end;
BEGIN
  assign(f,fo);rewrite(f);
  for i:=1 to 9 do
    begin
      fillchar(t,sizeof(t),0);

```

```

s:=0;
a[0]:=i;
s:=s+i;
t[i]:=1;
try(1);
end;
close(f);
END.

```

(Lời giải của bạn Nguyễn Chí Thức - Lớp 11A1 khối PTCTT - ĐHSP Hà Nội)

Bài 89/2001 - Chữ số thứ N

(Dành cho học sinh THCS và THPT)

Thuật toán: từ nhận xét rằng có 9 số có 1 chữ số, 90 số có 2 chữ số, ... Ta sẽ xác định xem chữ số thứ N thuộc số có mấy chữ số và nó là số nào? Sau đó xem nó ở vị trí thứ mấy trong số đó.

Program bai89;

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

```
{SM 16384,0,655360}
```

Uses crt;

```
Const fi ='number.inp';
```

```
fo ='number.out';
```

```
cs:array[1..8] of longint = (9, 180, 2700, 36000, 450000, 5400000, 63000000,
720000000);
```

```
Var n : longint;
```

```
f,g :text;
```

```
Function num(n:longint):char;
```

```
var k, so, mu : longint;
```

```
s : string;
```

```
Begin
```

```
k:=1; mu:=1;
```

```
while (k<9)and(cs[k]<n) do
```

```
begin
```

```
n:=n-cs[k];
```

```
inc(k); mu:=mu*10;
```

```
end;
```

```
if mu=1 then so:=n div k
```

```
else so:=n div k+mu+ord(n mod k>0)-1;
```

```
str(so,s);s:=s[k]+s;
```

```
num:=s[n mod k+1];
```

```
End;
```

```
BEGIN
```

```
assign(f,fi); reset(f);
```

```
assign(g,fo); rewrite(g);
```

```
while not seekeof(f) do
```

```
begin
```

```
readln(f,n);
```

```
writeln(g,num(n));
```

```
end;
```

```
close(f);
```

close(g);

END.

(Lời giải của bạn Lê Văn Đức - Nguyễn Huệ - Hà Đông - Hà Tây)

Bài 90/2002 - Thay số trong bảng 9 ô

(Dành cho học sinh Tiểu học)

Do tổng các số trong các ô điền cùng chữ cái ban đầu là bằng nhau nên ta suy ra: $2M = 3I = 4S$. Vì $4S$ chia hết cho 4, do đó $2M$ và $3I$ cũng chia hết cho 4.

Suy ra: I chia hết cho 4; $M = 2S$; $3I = 4S$.

Đặt $I = 4k$ ($k = 1, 2, \dots$), ta suy ra tương ứng: $S = 3k$, và $M = 6k$.

Ví dụ, với $k = 1$ ta có đáp số sau: $I = 4$, $S = 3$, $M = 6$;

Với $k = 2$, ta có: $I = 8$, $S = 6$, $M = 12$; ...

Bài 91/2002 - Các số lặp

(Dành cho học sinh THCS và THPT)

Program bai91;

{Thuat toan lua bo vao chuong}

{S,A+,B-,D+,E+,F-,G-,I+,L+,N-,O-,P-,Q-,R+,S+,T-,V+,X+}

{SM 16384,0,655360}

USES crt;

CONST M1 = MaxInt div 4 + 1;

M2 = MaxInt;

fi = 'Bai91.Inp';

TYPE MA = Array[0..M1] of LongInt;

Var A: Array[0..3] of ^MA;

d,l :LongInt;

Procedure Init;

Var i:Byte;

Begin

For i:=0 to 3 do

begin

New(A[i]);

Fillchar(A[i]^,sizeof(A[i]^),0);

end;

End;

Procedure ReadF(k:ShortInt);

Var f:Text;

x:LongInt;

i,j:Integer;

Begin

Init;

Assign(f,fi);

Reset(f);

While Not SeekEof(f) do

begin

Read(f,x);

x:=x*k;

If x>=0 then

begin

i:=x div M1;

```

        j:=x mod M1;
        If i=4 then begin i:=3; j:=M1; end;
        Inc(A[i]^j);
        If A[i]^j>d then begin d:=A[i]^j; l:=x*k; end;
    end;
end;
Close(f);
For i:=0 to 3 do Dispose(A[i]);
End;
BEGIN
    Clrscr;
    d:=0; l:=0;
    ReadF(-1);
    ReadF(1);
    Writeln('So lap nhieu nhat la: ',l,#10#13,'Voi so lan lap : ',d);
    Readln;
END.

```

(Lời giải của Nguyễn Toàn Thắng *)

Bài giải của bạn Nguyễn Toàn Thắng dùng thuật toán lùa bò vào chuồng. Sau đây là cách giải khác dùng thuật toán đếm số lần lặp.

Thuật toán: Tư tưởng thuật toán là dùng mảng đánh dấu có nghĩa là số x thì $Lap[x]$ sẽ là số lần xuất hiện của số x trong mảng. Vì số phần tử của mảng nhỏ hơn hoặc bằng 10^6 nên phần tử của mảng Lap phải là kiểu dữ liệu để có thể lưu trữ được 10^6 . Số x là số nguyên kiểu integer và do giới hạn bộ nhớ là 64K nên ta dùng ba mảng động như sau:

MG = array[-maxint..maxint] of byte;

L[1..3] of ^MG;

Xử lý trong hệ cơ số 100.

Chương trình.

{ \$A+,B-,D+,E+,F-,G-,I+,L+,N-,O-,P-,Q+,R+,S+,T-,V+,X+,Y+ }

{ \$M 16384,0,655360 }

program bai91; {Đỗ Đức Đông}

uses crt;

```
const    fi      ='input.txt';
```

```
        fo      ='output.txt';
```

```
        coso    =100;
```

```
type     mg     =array[-maxint..maxint]of byte;
```

```
var      L      :array[1..3]of ^mg;
```

```
        n,lap   :longint;
```

```
        kq      :integer;
```

```
        time    :longint;
```

```
        clock   :longint absolute $00:$0046c;
```

```
procedure tao_test;
```

```
var f    :text;
```

```
    k    :longint;
```

```
begin
```

```
    n:=1000000;
```



```

assign(f,fi);
rewrite(f);
writeln(f,n);
for k:=1 to N do
  if random(2)=1 then write(f,random(maxint),#32)
  else write(f,-random(maxint),#32);
close(f);
end;

procedure danhdao(x:integer);
var i :integer;
begin
  for i:=3 downto 1 do
    if L[i]^x<coso then
      begin
        inc(L[i]^x);
        break;
      end
    else L[i]^x:=0;
  end;

procedure lam;
var f :text;
  k :longint;
  x :integer;
begin
  for k:=1 to 3 do
    begin
      new(L[k]);
      fillchar(L[k]^,sizeof(L[k]^),0);
    end;
  assign(f,fi);
  reset(f);
  read(f,n);
  for k:=1 to n do
    begin
      read(f,x);
      danhdao(x);
    end;
  close(f);

  lap:=0;
  for k:=-maxint to maxint do
    if L[1]^k*sqr(coso)+L[2]^k*coso+L[3]^k>lap then
      begin
        lap:=L[1]^k*sqr(coso)+L[2]^k*coso+L[3]^k;
        kq:=k;
      end;

  for k:=1 to 3 do dispose(L[k]);

```

```

end;

procedure ghif;
var f :text;
begin
  assign(f,fo);
  rewrite(f);
  write(f,kq);
  writeln('So lan lap :',lap);
  close(f);
end;

BEGIN
  {tao_test;}
  time:=clock;
  lam;
  ghif;
  writeln((clock-time)/18.2:10:10);
END.

```

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

```

program DayChiaHet;
uses crt;
const inp='div.inp';
      out='div.out';
var a:array[0..1] of set of byte;
    g:text;
    k,n,t,i,j,l:longint;
function f(x:longint):byte;
begin
  x:=x mod k;
  if x<0 then f:=x+k else f:=x;
end;
begin
  clrscr;
  assign(g,inp);reset(g);
  readln(g,n,k);
  t:=0;
  read(g,j);
  a[0]:=[f(j)];
  for i:=2 to n do
  begin
    t:=1-t;
    a[t]:=[];
    read(g,j);
    for l:=0 to k-1 do
      if l in a[1-t] then
        begin
          a[t]:=a[t]+[f(l+j)];

```

```

    a[t]:=a[t]+[f(1-j)];
end;
end;
close(g);
assign(g,out);rewrite(g);
if 0 in a[t] then write(g,1) else write(g,0);
close(g);
write('Complete - Open file ',out,' to view the result');
readln;
End.

```

(Lời giải của bạn Vũ Lê An - 12T2 - Lê Khiết - Quảng Ngãi)

Mở rộng bài toán:

1. Tìm dãy con liên tiếp có tổng bé nhất.
2. Tìm dãy con liên tiếp các phần tử thuộc dãy bằng nhau dài nhất.
3. Cho ma trận $M \times N$ hãy tìm hình chữ nhật có tổng lớn nhất (nhỏ nhất) với $M, N \leq 100$
4. Cho ma trận $M \times N$ hãy tìm hình chữ nhật có diện tích lớn nhất có các phần tử bằng nhau.

Cách giải bài toán 2 giải giống với bài toán 1, bài toán 3 và 4 giải giống nhau dựa trên cơ sở bài 1,2.

Cách giải bài toán 3: Xét hình các hình chữ nhật có tọa độ cột trái là i tọa độ cột phải là j (mất $O(N^2)$). Coi mỗi dòng như một phần tử, để tìm hình chữ nhật có diện tích lớn nhất ta phải mất $O(N)$ nữa. Như vậy độ phức tạp là $O(N^3)$.

Bài 93/2002 - Trò chơi bắn bi

(Dành cho học sinh Tiểu học)

Có 3 đường đi đạt số điểm lớn nhất là: 32.

Bài 94/2002 - Biểu diễn tổng các số Fibonacci

(Dành cho học sinh THCS)

Cách giải: Ta sẽ tìm số Fibonacci gần với số N nhất. Đây sẽ chính là số hạng đầu tiên nằm trong dãy kết quả. Sau đó, lấy hiệu của số N và số Fibonacci gần với số N nhất, tiếp tục tìm số Fib gần với hiệu trên và cứ thế cho đến khi hiệu đó là một số Fib. Kết quả các số Fibonacci sẽ được liệt kê theo thứ tự từ lớn đến nhỏ.

Chương trình:

```

Program BdFib; {Bai 94/2002: Bieu dien tong cac so Fibonacci}
uses crt;
var n:longint;
    f:array[1..1000] of longint;
function fib(k:integer): longint;
begin
    f[1]:=1;
    f[2]:=1;
    f[3]:=2;
    if f[k]=-1 then f[k]:=fib(k-1)+fib(k-2);
    fib:=f[k];
end;
procedure xuly;
var i,j:longint;
begin
    for i:=1 to 1000 do f[i]==-1;
    while n>0 do
        begin

```

```

        i:=1;
        while fib(i)<=n do
            inc(i);
            j:=fib(i-1);
            write(j, ' + ');
            n:=n-j;
        end;
        gotoxy(wherex-2,wherey);
        writeln(' ');
    end;
    procedure test;
    begin
        clrscr;
        write('Nhap n='); readln(n);
        clrscr;
        write('n=');
        xuly;
    end;
    BEGIN
        test;
        readln;
    END.

```

(Lời giải của bạn Cao Lê Thăng Long - Lớp 8E Nguyễn Trường Tộ - Hà Nội)

Bài 95/2002 - Dãy con có tổng lớn nhất
(Dành cho học sinh THPT)

```

Program subseq;
const inp = 'subseq.inp';
      out = 'subseq.out';
var n, dau, cuoi, d:longint;
    max, T:longint;
    f, g:text;
Procedure input;
begin
    assign(f,inp); reset(f);
    assign(g,out); rewrite(g);
    Readln(f,n);
End;
Procedure solve;
    var i,j:longint;
begin
    dau:=1; cuoi:=1; d:=1;
    max:=-maxlongint; T:=0;
    for i:=1 to n do
        begin
            readln(f,j); T:=T + j ;
            If T > max then
                begin
                    max:=T;
                    dau:=d; cuoi:=i;
                end;
            If T<0 then begin T:=0; d:=i+1; end;
        end;
    end;

```

```

end;
End;
Procedure output;
Begin
  writeln(g,dau);
  writeln(g,cuoi);
  writeln(g,max);
  Close(f); Close(g);
End;
BEGIN
  input;
  solve;
  output;
END.

```

(Lời giải của bạn Võ Xuân Sơn - Lớp 11A2 THPT Phan Bội Châu - Nghệ An)

Bài 96/2002 - Số chung lớn nhất

(Dành cho học sinh THPT)

{A+,B-,D+,E+,F-,G-,H+,L+,N-,O-,P-,Q+,R+,S+,T-,V+,X+}
 {\$M 16384,0,655360}

```

uses crt;
const maxn = 251;
      fi = 'string.inp';
      fo = 'string.out';
var pa : array[0..maxn,0..maxn] of byte;
    s1,s2,skq : string;
    max : byte;
procedure docf;
var f : text;
begin
  assign(f,fi);
  reset(f);
  readln(f,s1);
  read(f,s2);
  close(f);
end;
function maxxso(a,b:byte) : byte;
begin
  maxxso := (abs(a-b)+a+b) div 2;
end;
procedure ldonotknow;
var i,j : byte;
begin
  for i := length(s1) downto 1 do
    for j := length(s2) downto 1 do
      if s1[i] = s2[j] then pa[i,j] := pa[i+1,j+1] + 1
      else pa[i,j] := maxxso(pa[i+1,j] , pa[i,j+1] );
    max := pa[1,1];
  end;
end;
procedure wastingtime;
var ch : char;
    i,j,so,is,js : byte;
begin
  is := 1; js := 1;
  so := 0;

```

```

repeat
  for ch := '9' downto '0' do
    begin
      i := is; j := js;
      while (s1[i] <> ch) and (i <= length(s1)) do inc(i);
      while (s2[j] <> ch) and (j <= length(s2)) do inc(j);
      if pa[i,j] = max - so then
        begin
          skq := skq + ch;
          is := i+1; js := j+1;
          break;
        end;
      end;
      inc(so);
    until max=so;
    while (skq[1] = '0') and (skq <> '0') do delete(skq,1,1);
  end;
procedure ghif;
var f : text;
begin
  assign(f,fo);
  rewrite(f);
  if max = 0 then write(f,' Khong co xau chung !!!...')
  else
    begin
      wastingttime;
      write(f,skq);
    end;
  close(f);
end;
BEGIN
  docf;
  idonotknow;
  ghif;
END.

```

Bài 97/2002 - Thay số trong bảng

(Dành cho học sinh Tiểu học)

	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 kề nhau

Đọc

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

Giải:

Từ (5) - Tích của các số tự nhiên đầu tiên cho kết quả là một số có 3 chữ số chỉ có thể là 120 hoặc 720 ($1 \times 2 \times 3 \times 4 \times 5 = 120$; $1 \times 2 \times 3 \times 4 \times 5 \times 6 = 720$).

Do đó, (5) có thể là 120 hoặc 720. Suy ra: $f = 0$; $e = 2$; $d = 1$ hoặc $d = 7$.

Tương tự, ta tìm được (6) có thể là 105 hoặc 385 ($3 \times 5 \times 7 = 105$; $5 \times 7 \times 11 = 385$). Suy ra: $i = 5$; $h = 0$ hoặc $h = 8$; $g = 1$ hoặc $g = 3$.

Từ (4) suy ra c chỉ có thể là số chẵn. Do $f = 0$, $i = 5$, từ (3) ta tìm được $c = 6$.

Từ (2) - tích của nhiều thừa số 2 cho kết quả là một số có 3 chữ số chỉ có thể là một trong các số: 128, 256, 512. Mà theo trên $e = 2$ nên ta tìm được (2) là 128. Vậy $b = 1$, $h = 8$, $g = 3$.

Từ (4) - Bội số nguyên của 8, do đó ta có thể tìm được (4) có thể là một trong các số: 216, 416, 616, 816.

Tức là, a có thể bằng 2, 4, 6, hoặc 8. Kết hợp với (1), giả sử $d = 1$, như vậy ta không thể tìm được số nào thoả mãn (1).

Với $d = 7$, ta tìm được $a = 4$ thoả mãn (1).

Vậy $a = 4$, $b = 1$, $c = 6$, $d = 7$, $e = 2$, $f = 0$, $g = 3$, $h = 8$, $i = 5$.

Và ta có kết quả như sau:

4	1	6
7	2	0
3	8	5

Bài 100/2002 - Mời khách dự tiệc

(Dành cho học sinh THPT)

```

program Guest;
const
  Inp = 'Guest.inp';
  Out = 'Guest.out';
var
  n: Integer;
  lSum: LongInt;
  t, v, p, Pred, Ind: array[0..1005] of Integer;
  Value: array[0..1005] of LongInt;
  Ok: array[0..1005] of Boolean;
procedure ReadInput;
var
  hFile: Text;
  i: Integer;
begin
  Assign(hFile, Inp);
  Reset(hFile);
  Readln(hFile, n);
  for i := 1 to n do Readln(hFile, t[i], v[i]);
  Close(hFile);
end;
procedure QuickSort(l, r: Integer);
var
  i, j, x, tg: Integer;
begin
  i := l; j := r; x := p[(l + r) div 2];
  repeat
    while t[p[i]] < t[x] do Inc(i);
    while t[p[j]] > t[x] do Dec(j);
    if i <= j then
      begin

```

```

    tg := p[i]; p[i] := p[j]; p[j] := tg;
    Inc(i); Dec(j);
  end;
until i > j;
if i < r then QuickSort(i, r);
if j > l then QuickSort(l, j);
end;
procedure Prepare;
var
  i, j: Integer;
begin
  FillChar(Value, SizeOf(Value), 0);
  FillChar(Ok, SizeOf(Ok), False);
  lSum := 0;
  for i := 1 to n + 1 do p[i] := i;
  t[n + 1] := n + 1;
  QuickSort(1, n);
  j := 2; Ind[0] := 1;
  for i := 1 to n do
  begin
    while t[p[j]] = i do Inc(j);
    Ind[i] := j - 1;
  end;
end;
function View(n: Integer): LongInt;
var
  i, j: Integer;
  lSum1, lSum2: LongInt;
begin
  lSum1 := 0; lSum2 := v[n];
  for i := Ind[n - 1] + 1 to Ind[n] do
  begin
    if Value[p[i]] = 0 then Value[p[i]] := View(p[i]);
    lSum1 := lSum1 + Value[p[i]];
    for j := Ind[p[i] - 1] + 1 to Ind[p[i]] do
    begin
      if Value[p[i]] = 0 then Value[p[i]] := View(p[j]);
      lSum2 := lSum2 + Value[p[j]];
    end;
  end;
  end;
  if lSum1 > lSum2 then
  begin
    View := lSum1;
    Pred[n] := n - 1;
  end
  else
  begin
    View := lSum2;
    Pred[n] := n - 2;
  end;
end;

```



```

end;
procedure Calculator(n: Integer);
var
  i, j: Integer;
begin
  if Pred[n] = n - 2 then
    begin
      Ok[n] := True; Inc(lSum);
      for i := Ind[n - 1] + 1 to Ind[n] do
        for j := Ind[p[i] - 1] + 1 to Ind[p[i]] do Calculator(p[j])
        end
      else for i := Ind[n - 1] + 1 to Ind[n] do Calculator(p[i])
    end;
  procedure WriteOutput;
  var
    hFile: Text;
    i: Integer;
    sView: LongInt;
  begin
    Assign(hFile, Out);
    Rewrite(hFile);
    sView := View(p[1]);
    Calculator(p[1]);
    Writeln(hFile, lSum, ' ', sView);
    for i := 1 to n do
      if Ok[i] then Writeln(hFile, i);
    Close(hFile);
  end;
begin
  ReadInput;
  Prepare;
  WriteOutput;
end.

```

===== The End =====