



100 đề Toán Tin

Tin học & Nhà trường

Hà Nội - 2002

Phần 1: ĐỀ BÀI

Bài 1/1999 - Trò chơi cùng nhau qua cầu

(Dành cho học sinh Tiểu học)

Bốn người cần đi qua một chiếc cầu. Do cầu yếu nên mỗi lần đi không quá hai người, và vì trời tối nên phải cầm đèn mới đi được. Bốn người đi nhanh chậm khác nhau, qua cầu với thời gian tương ứng là 10 phút, 5 phút, 2 phút và 1 phút. Vì chỉ có một chiếc đèn nên mỗi lần qua cầu phải có người mang đèn trở về cho những người kế tiếp. Khi hai người đi cùng nhau thì qua cầu với thời gian của người đi chậm hơn. Ví dụ sau đây là một cách đi:

- Người 10 phút đi với người 5 phút qua cầu, mất 10 phút.
- Người 5 phút cầm đèn quay về, mất 5 phút.
- Người 5 phút đi với người 2 phút qua cầu, mất 5 phút.
- Người 2 phút cầm đèn quay về, mất 2 phút.
- Người 2 phút đi với người 1 phút qua cầu, mất 2 phút.

Thời gian tổng cộng là $10+5+5+2+2 = 24$ phút.

Em hãy tìm cách đi khác với tổng thời gian càng ít càng tốt, và nếu dưới 19 phút thì thật tuyệt vời! Lời giải ghi trong tệp văn bản có tên là P1.DOC

Bài 2/1999 - Tổ chức tham quan

(Dành cho học sinh THCS)

Trong đợt tổ chức đi tham quan danh lam thắng cảnh của thành phố Hồ Chí Minh, Ban tổ chức hội thi Tin học trẻ tổ chức cho N đoàn (đánh từ số 1 đến N) mỗi đoàn đi thăm quan một địa điểm khác nhau. Đoàn thứ i đi thăm địa điểm ở cách Khách sạn Hoàng Đế d_i km ($i=1,2,\dots, N$). Hội thi có M xe taxi đánh số từ 1 đến M ($M \geq N$) để phục vụ việc đưa các đoàn đi thăm quan. Xe thứ j có mức tiêu thụ xăng là v_j đơn vị thể tích/km.

Yêu cầu: Hãy chọn N xe để phục vụ việc đưa các đoàn đi thăm quan, mỗi xe chỉ phục vụ một đoàn, sao cho tổng chi phí xăng cần sử dụng là ít nhất.

Dữ liệu: File văn bản P2.INP:

- Dòng đầu tiên chứa hai số nguyên dương N, M ($N \leq M \leq 200$);
- Dòng thứ hai chứa các số nguyên dương d_1, d_2, \dots, d_N ;
- Dòng thứ ba chứa các số nguyên dương v_1, v_2, \dots, v_M .
- Các số trên cùng một dòng được ghi khác nhau bởi dấu trắng.

Kết quả: Ghi ra file văn bản P2.OUT:

- Dòng đầu tiên chứa tổng lượng xăng dầu cần dùng cho việc đưa các đoàn đi thăm quan (không tính lượt về);
- Dòng thứ i trong số N dòng tiếp theo ghi chỉ số xe phục vụ đoàn i ($i=1, 2, \dots, N$).

Ví dụ:

P2.INP
3 4
7 5 9
17 13 15 10

P2.OUT
256
2
3
4

Bài 3/1999 - Mạng tế bào

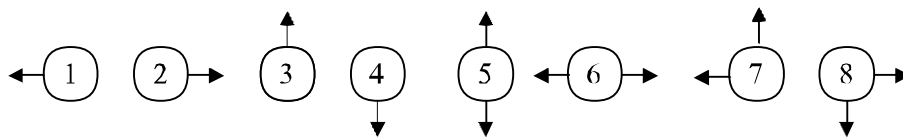
(Dành cho học sinh THPT)

Mạng tế bào có dạng một lưới ô vuông hình chữ nhật. Tại mỗi nhịp thời gian: mỗi ô của lưới chứa tín hiệu là 0 hoặc 1 và có thể truyền tín hiệu trong nó cho một số ô kề cạnh theo một qui luật cho trước. Ô ở góc trên bên trái có thể nhận tín hiệu từ bên ngoài đưa vào. Sau nhịp thời gian đó, tín hiệu ở một ô sẽ là 0 nếu tất cả các tín hiệu truyền đến nó là 0, còn trong trường hợp ngược lại tín hiệu trong nó sẽ là 1. Một ô không nhận được tín hiệu nào từ các ô kề cạnh với nó sẽ giữ nguyên tín hiệu đang có trong nó. Riêng đối với ô trên trái, sau khi truyền tín hiệu chứa trong nó đi, nếu có tín hiệu vào thì ô trên trái sẽ *chỉ* nhận tín hiệu này, còn nếu không có tín hiệu nào thì ô trên trái cũng hoạt động giống như các ô khác, ở trạng thái đầu tín hiệu trong tất cả các ô là 0.

Yêu cầu: Cho trước số nhịp thời gian T và dãy tín hiệu vào S là một dãy gồm T ký hiệu S_1, \dots, S_T , trong đó S_i là 0 hoặc 1 thể hiện có tín hiệu vào, ngược lại S_i là X thể hiện không có tín hiệu vào tại nhịp thời gian thứ i ($1 \leq i \leq T$), hãy xác định trạng thái của lưới sau nhịp thời gian thứ T .

Dữ liệu: vào từ file văn bản P3.INP:

- Dòng đầu tiên chứa 3 số nguyên M, N, T theo thứ tự là số dòng, số cột của lưới và số nhịp thời gian ($1 < M, N \leq 200; T \leq 100$);
- Dòng thứ hai chứa xâu tín hiệu vào S ;
- M dòng tiếp theo mô tả qui luật truyền tin. Dòng thứ i trong số M dòng này chứa N số $a_{i1}, a_{i2}, \dots, a_{iN}$, trong đó giá trị của a_{ij} sẽ là 1, 2, 3, 4, 5, 6, 7, 8 tương ứng lần lượt nếu ô (i, j) phải truyền tin cho ô kề cạnh bên trái, bên phải, bên trên, bên dưới, bên trên và bên dưới, bên trái và bên phải, bên trên và bên trái, bên dưới và bên phải (xem hình vẽ); còn nếu ô (i, j) không phải truyền tín hiệu thì $a_{ij} = 0$.



Kết quả: Ghi ra file văn bản P3.OUT gồm M dòng, mỗi dòng là một xâu gồm N ký tự 0 hoặc 1 mô tả trạng thái của lưới sau nhịp thời gian thứ T .

Ví dụ:

P3.INP
2 2 5
101XX
2 4
2 1

P3.OUT
11
01

Quá trình biến đổi trạng thái được diễn tả trong hình dưới đây:

0	0
0	0

1	0
0	0

0	1
0	0

1	0
0	1

1	1
1	0

1	1
0	1

Bài 4/1999 - Trò chơi bốc sỏi

(Dành cho học sinh Tiểu học)

Trên mặt đất có một đống sỏi có 101 viên. Hai em học sinh Hoàng và Huy chơi trò chơi như sau: Mỗi em đến lượt đi phải bốc ra từ đống sỏi trên tối thiểu là 1 viên và tối đa là 4 viên. Người thua là người phải bốc viên sỏi cuối cùng. Giả sử Hoàng là người được bốc trước, Huy bốc sau. Các em thử nghĩ xem ai là người thắng cuộc, Hoàng hay Huy? Và người thắng cuộc phải suy nghĩ gì và thực hiện các bước đi của mình ra sao?

Bài 5/1999 - 12 viên bi

(Dành cho học sinh THCS)

Có 12 hòn bi giống hệt nhau về kích thước, hình dáng và khối lượng. Tuy nhiên trong chúng lại có đúng một hòn bi kém chất lượng: hoặc nhẹ hơn hoặc nặng hơn bình thường. Dùng một cân bàn hai bên, bạn hãy dùng 3 lần cân để tìm ra được viên bi đó. Cần chỉ rõ rằng viên bi đó là nặng hơn hay nhẹ hơn.

Viết chương trình mô phỏng việc tổ chức cân các hòn bi trên. Dữ liệu về hòn bi kém chất lượng do người sử dụng chương trình nhập giữ. Yêu cầu trình bày chương trình đẹp và mỹ thuật.

Bài 6/1999 - Giao điểm các đường thẳng

(Dành cho học sinh THPT)

Trên mặt phẳng cho trước n đường thẳng. Hãy tính số giao điểm của các đường thẳng này. Yêu cầu tính càng chính xác càng tốt.

Các đường thẳng trên mặt phẳng được cho bởi 3 số thực A, B, C với phương trình $Ax + By + C = 0$, ở đây các số A, B không đồng thời bằng 0.

Dữ liệu vào của bài toán cho trong tệp B6.INP có dạng sau:

- Dòng đầu tiên ghi số n
- n dòng tiếp theo, mỗi dòng ghi 3 số thực A, B, C cách nhau bởi dấu cách.

Kết quả của bài toán thể hiện trên màn hình.

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

(Dành cho học sinh THPT)

Xét bài toán tương tự như bài 6/1999 nhưng yêu cầu tính số miền mặt phẳng được chia bởi n đường thẳng này:

Trên mặt phẳng cho trước n đường thẳng. Hãy tính số miền mặt phẳng được chia bởi các đường thẳng này. Yêu cầu tính càng chính xác càng tốt.

Các đường thẳng trên mặt phẳng được cho bởi 3 số thực A, B, C với phương trình $Ax + By + C = 0$, ở đây các số A, B không đồng thời bằng 0.

Dữ liệu vào của bài toán cho trong tệp B7.INP có dạng sau:

- Dòng đầu tiên ghi số n
- n dòng tiếp theo, mỗi dòng ghi 3 số thực A, B, C cách nhau bởi dấu cách.

Kết quả của bài toán thể hiện trên màn hình.

Bài 8/1999 - Cân táo

(Dành cho học sinh Tiểu học)

Mẹ đi chợ về mua cho Nga 27 quả táo giống hệt nhau về kích thước và khối lượng. Tuy nhiên người bán hàng nói rằng trong số các quả táo trên có đúng một quả có khối lượng nhẹ hơn. Em hãy dùng một chiếc cân bàn hai bên để tìm ra quả táo nhẹ đó. Yêu cầu số lần cân là nhỏ nhất.

Các em hãy giúp bạn Nga tìm ra quả táo nhẹ đó đi. Nếu các em tìm ra quả táo đó sau ít hơn 5 lần cân thì đã là tốt lắm rồi.

Bài 9/1999 - Bốc diêm

(Dành cho học sinh Tiểu học)

Trên bàn có 3 dãy que diêm, số lượng que diêm của các dãy này lần lượt là 3, 5 và 8. Hai bạn Nga và An chơi trò chơi sau: Mỗi bạn đến lượt mình được quyền (và phải) bốc một số que diêm bất kỳ từ một dãy trên. Người thắng là người bốc được que diêm cuối cùng.

Ai là người thắng cuộc trong trò chơi trên? Và bạn đó phải bốc diêm như thế nào? Các bạn hãy cùng suy nghĩ với Nga và An nhé.

Bài 10/1999 - Dãy số nguyên

(Dành cho học sinh THCS)

Dãy các số tự nhiên được viết ra thành một dãy vô hạn trên đường thẳng:

1234567891011121314..... (1)

Hỏi số ở vị trí thứ 1000 trong dãy trên là số nào?

Em hãy làm bài này theo hai cách: Cách 1 dùng suy luận logic và cách 2 viết chương trình để tính toán và so sánh hai kết quả với nhau.

Tổng quát bài toán trên: Chương trình yêu cầu nhập số K từ bàn phím và in ra trên màn hình kết quả là số nằm ở vị trí thứ K trong dãy (1) trên. Yêu cầu chương trình chạy càng nhanh càng tốt.

Bài 11/1999 - Dãy số Fibonacci

(Dành cho học sinh THCS)

Như các bạn đã biết dãy số Fibonacci là dãy 1, 1, 2, 3, 5, 8, Dãy này cho bởi công thức đệ qui sau:

$$F_1 = 1, F_2 = 1, F_n = F_{n-1} + F_{n-2} \text{ với } n > 2$$

1. Chứng minh khẳng định sau:

Mọi số tự nhiên N đều có thể biểu diễn duy nhất dưới dạng tổng của một số số trong dãy số Fibonacci.

$$N = a_k F_k + a_{k-1} F_{k-1} + \dots + a_1 F_1$$

Với biểu diễn như trên ta nói N có biểu diễn Fibonacci là $a_k a_{k-1} \dots a_2 a_1$.

2. Cho trước số tự nhiên N , hãy tìm biểu diễn Fibonacci của số N .

Input:

Tệp văn bản P11.INP bao gồm nhiều dòng. Mỗi dòng ghi một số tự nhiên.

Output:

Tệp P11.OUT ghi kết quả của chương trình: trên mỗi dòng ghi lại biểu diễn Fibonacci của các số tự nhiên tương ứng trong tệp P11.INP.

Bài 12/1999 - N-mino

(Dành cho học sinh THPT)

N -mino là hình thu được từ N hình vuông 1×1 ghép lại (cạnh kề cạnh). Hai n -mino được gọi là đồng nhất nếu chúng có thể đặt chồng khít lên nhau.

Bạn hãy lập chương trình tính và vẽ ra tất cả các N -mino trên màn hình. Số n nhập từ bàn phím.

Ví dụ: Với $N=3$ chỉ có hai loại N -mino sau đây:



3-mino thẳng

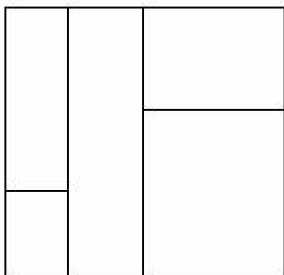
3-mino hình thước thợ

Chú ý: Gọi M_n là số các n -mino khác nhau thì ta có $M_1=1, M_2=1, M_3=2, M_4=5, M_5=12, M_6=35, \dots$

Yêu cầu bài giải đúng và trình bày đẹp.

Bài 13/1999 - Phân hoạch hình chữ nhật*(Dành cho học sinh THPT)*

Một hình vuông có thể chia thành nhiều hình chữ nhật có các cạnh song song với cạnh hình vuông (xem Hình vẽ). Xây dựng cấu trúc dữ liệu và lập chương trình mô tả phép chia đó. Tính xem có bao nhiêu cách chia như vậy.

**Input**

Dữ liệu nhập vào từ tệp P13.INP bao gồm hai số tự nhiên là n, m - kích thước hình chữ nhật.

Output

Dữ liệu ra nằm trong tệp P13.OUT có dạng sau:

- Dòng đầu tiên ghi số K là tổng số các phép phân hoạch.
- Tiếp theo là K nhóm, mỗi nhóm cách nhau bằng một dòng trống.

- Mỗi nhóm dữ liệu bao gồm các cặp tọa độ của các hình chữ nhật nằm trong phân hoạch.

Bài 14/2000 - Tìm số trang sách của một quyển sách*(Dành cho học sinh Tiểu học)*

Để đánh số các trang sách của 1 quyển sách cần tất cả 1392 chữ số. Hỏi quyển sách có tất cả bao nhiêu trang?

Bài 15/2000 - Hội nghị đội viên*(Dành cho học sinh Tiểu học)*

Trong một hội nghị liên chi đội có một số bạn nam và nữ. Biết rằng mỗi bạn trai đều quen với N các bạn gái và mỗi bạn gái đều quen với đúng N bạn trai. Hãy lập luận để chứng tỏ rằng 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)*

Bạn hãy chia N^2 số $1, 2, 3, \dots, N^2-1, N^2$ thành N nhóm sao cho mỗi nhóm có số các số hạng như nhau và có tổng các số này cũng bằng nhau.

Bài 17/2000 - Số nguyên tố tương đương*(Dành cho học sinh THCS)*

Hai số tự nhiên được gọi là *Nguyên tố tương đương* nếu chúng có chung các ước số nguyên tố. Ví dụ các số 75 và 15 là nguyên tố tương đương vì cùng có các ước nguyên tố là 3 và 5. Cho trước hai số tự nhiên N, M . Hãy viết chương trình kiểm tra xem các số này có là nguyên tố tương đương với nhau hay không.

Bài 18/2000 - Sên bò*(Dành cho học sinh THCS và THPT)*

Trên lưới ô vuông một con sên xuất phát từ đỉnh $(0,0)$ cần phải đi đến điểm kết thúc tại $(N,0)$ (N là số tự nhiên cho trước).

Qui tắc đi: Mỗi bước $(x_1, y_1) \rightarrow (x_2, y_2)$ thoả mãn điều kiện (sên bò):

- $x_2 = x_1 + 1$,
- $y_1 - 1 \leq y_2 \leq y_1 + 1$

Tìm một cách đi sao cho trong quá trình đi nó có thể lên cao nhất trên trục tung (tức là tọa độ y đạt cực đại). Chỉ cần đưa ra một nghiệm.

Input

Số N được nhập từ bàn phím.

Output

Output ra file P5.OUT có dạng:

- Dòng đầu tiên ghi 2 số: m, h . Trong đó m là số các bước đi của con sên để đến được vị trí đích, h ghi lại độ cao cực đại đạt được của con sên.
- m dòng tiếp theo, mỗi dòng ghi ra lần lượt các tọa độ (x,y) là các bước đi của sên trên lưới.

Yêu cầu kỹ thuật

Các bạn có thể mô tả các bước đi của con sên trên màn hình đồ họa. Để đạt được mục đích đó số N cần được chọn không vượt quá 50. Mặc dù không yêu cầu nhưng những lời giải có mô phỏng đồ họa sẽ có điểm cao hơn nếu không mô phỏng đồ họa.

Bài 19/2000 - Đa giác

(Dành cho học sinh THPT)

Hãy tìm điều kiện cần và đủ để N số thực dương a_1, a_2, \dots, a_N tạo thành các cạnh liên tiếp của một đa giác N cạnh trên mặt phẳng. Giả sử cho trước N số a_1, a_2, \dots, a_N thỏa mãn điều kiện là các cạnh của đa giác, bạn hãy lập chương trình biểu diễn và vẽ đa giác trên.

Input

Input của bài toán là tệp P6.INP bao gồm 2 dòng, dòng đầu tiên ghi số N , dòng thứ hai ghi N số thực cách nhau bởi dấu cách.

Output

Đầu ra của bài toán thể hiện trên màn hình.

Chú ý: Phần lý thuyết của bài toán cần được chứng minh một cách chặt chẽ.

Bài 20/2000 - Bạn Lan ở căn hộ số mấy?

(Dành cho học sinh Tiểu học)

Nhà Lan ở trong một ngôi nhà 8 tầng, mỗi tầng có 8 căn hộ. Một hôm, các bạn trong lớp hỏi Lan:

"Nhà bạn ở căn hộ số mấy?".

"Các bạn hãy thử hỏi một số câu, mình sẽ trả lời tất cả câu hỏi của các bạn, nhưng chỉ nói "đúng" hoặc "không" thôi. Qua các câu hỏi đó các bạn thử đoán xem mình ở căn hộ số bao nhiêu"- Lan trả lời.

Bạn Huy nói:

"Mình sẽ hỏi, có phải bạn ở căn hộ số 1, số 2,..., số 63 không. Như vậy với nhiều nhất 63 câu hỏi mình sẽ biết được bạn căn hộ nào."

Bạn Nam nói:

"Còn mình chỉ cần đến 14 câu, 7 câu đủ để biết bạn ở tầng mấy và 7 câu có thể biết chính xác bạn ở căn hộ số mấy".

Còn em, em phải hỏi nhiều nhất mấy lần để biết được bạn Lan ở căn hộ số bao nhiêu?

Bài 21/2000 - Những trang sách bị rơi

(Dành cho học sinh Tiểu học)

Một cuốn sách bị rơi mất một mảng. Trang bị rơi thứ nhất có số 387, còn trang cuối cũng gồm 3 chữ số 3, 8, 7 nhưng được viết theo một thứ tự khác.

Hỏi có bao nhiêu trang sách bị rơi ra?

Bài 22/2000 - Đếm đường đi

(Dành cho học sinh THCS)

Cho hình sau:

- Bạn hãy đếm tất cả các đường đi từ A đến B. Mỗi đường đi chỉ được đi qua mỗi đỉnh nhiều nhất là 1 lần.
- Bạn hãy tìm tất cả các đường đi từ A đến D, sao cho đường đi đó qua mỗi cạnh đúng một lần.
- Bạn hãy tìm tất cả các đường đi qua tất cả các cạnh của hình, mỗi cạnh đúng một lần, sao cho:
 - Điểm bắt đầu và điểm kết thúc trùng nhau.
 - Điểm bắt đầu và điểm kết thúc không trùng nhau

Bài 23/2000 - Quay Rubic

(Dành cho học sinh THPT)

Rubic là một khối lập phương gồm $3 \times 3 \times 3 = 27$ khối lập phương con. Mỗi mặt rubic gồm $3 \times 3 = 9$ mặt của một lớp 9 khối lập phương con. ở trạng thái ban đầu, mỗi mặt rubic được tô một màu. Các mặt khác nhau được tô các màu khác nhau. Giả sử ta đang nhìn vào một mặt trước của rubic. Có thể kí hiệu màu các mặt như sau: F: màu mặt trước là mặt ta đang nhìn; U: màu mặt trên; R: màu mặt phải; B: màu mặt sau; L: màu mặt bên trái; D: màu mặt dưới.

Một lớp gồm 3×3 khối lập phương con có thể quay 90 độ nhiều lần, trục quay đi qua tâm và vuông góc với mặt đang xét. Kết quả sau khi quay là khối lập phương $3 \times 3 \times 3$ với các màu mặt đã bị đổi khác.

Một xâu vòng quay liên tiếp rubic có thể mô tả bằng xâu các chữ cái của U, R, F, D, B, L, trong đó mỗi chữ cái là kí hiệu một vòng quay cơ sở: quay mặt tương ứng 90 độ theo chiều kim đồng hồ. Hãy viết chương trình giải 3 bài toán dưới đây:

1. Cho 2 xâu INPUT khác nhau, kiểm tra xem liệu nếu áp dụng với trạng thái đầu có cho cùng một kết quả hay không?
2. Cho một xâu vào, hãy xác định số lần cần áp dụng xâu vào đó cho trạng thái đầu rubic để lại nhận được trạng thái đầu đó.

Bài 24/2000 - Sắp xếp dãy số

(Dành cho học sinh Tiểu học)

Cho dãy số: 3, 1, 7, 9, 5

Cho phép 3 lần đổi chỗ, mỗi lần được đổi chỗ hai số bất kỳ. Em hãy sắp xếp lại dãy số trên theo thứ tự tăng dần.

Bài 25/2000 - Xây dựng số

(Dành cho học sinh THCS)

Cho các số sau: 1, 2, 3, 5, 7

Chỉ dùng phép toán cộng hãy dùng dãy trên để tạo ra số: 43, 52.

Ví dụ để tạo số 130 bạn có thể làm như sau: $123 + 7 = 130$.

Bài 26/2000 - Tô màu

(Dành cho học sinh THCS)

Cho lưới ô vuông 4×4 , cần phải tô màu các ô của lưới. Được phép dùng 3 màu: Xanh, đỏ, vàng. Điều kiện tô màu là ba ô bất kỳ liên nhau theo chiều dọc và ngang phải khác màu nhau. Hỏi có bao nhiêu cách như vậy, hãy liệt kê tất cả các cách.

Bài 27/2000 - Bàn cờ

(Dành cho học sinh THPT)

Cho một bàn cờ vuông 8×8 , trên đó cho trước một số quân cờ. Ví dụ hình vẽ sau là một bàn cờ như vậy:

	×		×		×		
×			×	×			×
×		×				×	×
			×		×		
		×					
	×		×				×
×			×	×			
	×				×	×	

Dữ liệu nhập được ghi trên tệp BANCO.TXT bao gồm 8 dòng, mỗi dòng là một sô nhị phân có độ dài bằng 8. Vị trí các quân cờ ứng với số 1, các ô trống ứng với số 0. Ví dụ tệp BANCO.TXT ứng với bàn cờ trên:

01010100

10011001

10100011

00010100

00100000

01010001

10011000

01000110

Hãy viết chương trình tính số quân cờ liên tục lớn nhất nằm trên một đường thẳng trên bàn cờ. Đường thẳng ở đây có thể là đường thẳng đứng, đường nằm ngang hoặc đường chéo. Kết quả thể hiện trên màn hình.

Với ví dụ nêu trên, chương trình phải in trên màn hình kết quả là 4.

Bài 28/2000 - Đổi tiền

(Dành cho học sinh Tiểu học)

Giả sử bạn có nhiều tờ tiền loại 1, 2 và 3 ngàn đồng. Hỏi với các tờ tiền đó bạn có bao nhiêu cách đổi tờ 10 ngàn đồng? Hãy liệt kê các cách đổi.

Bài 29/2000 - Chọn bạn

(Dành cho học sinh THCS)

Trong một trại hè người ta tình cờ chọn ra một nhóm 6 học sinh. Chứng minh rằng sẽ tìm được 3 trong số 6 bạn đó sao cho 3 bạn này hoặc đã quen nhau (đôi một) từ trước hoặc chưa hề quen nhau. Em hãy chỉ ra cách tìm 3 bạn đó.

Bài 30/2000 - Phần tử yên ngựa

(Dành cho học sinh THCS)

Cho bảng A kích thước $M \times N$. Phần tử A_{ij} được gọi là phần tử yên ngựa nếu nó là phần tử nhỏ nhất trong hàng của nó đồng thời là phần tử lớn nhất trong cột của nó. Ví dụ trong bảng số sau đây:

15	3	9
55	4	6
76	1	2

thì phần tử A_{22} chính là phần tử yên ngựa.

Bạn hãy lập chương trình nhập từ bàn phím một bảng số kích thước $M \times N$ và kiểm tra xem nó có phần tử yên ngựa hay không?

Bài 31/2000 - Biểu diễn phân số

(Dành cho học sinh PTTH)

Một phân số luôn luôn có thể được viết dưới số thập phân hữu hạn hoặc vô hạn tuần hoàn. Ví dụ:

$$\begin{aligned} 23/5 &= 4.6 \\ 3/8 &= 0.375 \\ 1/3 &= 0.(3) \\ 45/56 &= 0.803(571428) \end{aligned}$$

....

Trong các ví dụ trên thì các chữ số đặt trong dấu ngoặc chỉ phần tuần hoàn của số thập phân.

Nhiệm vụ của bạn là viết một chương trình nhập từ số (N) và nhập mẫu số (D), sau đó đưa ra kết quả là dạng thập phân của phân số N/D.

Ví dụ chạy chương trình:

$$\begin{aligned} \text{Nhap N, D: } &1 \quad 7 \\ 1/7 &= 0.(142857)_ \end{aligned}$$

Bài 32/2000 - Bài toán 8 hậu

(Dành cho học sinh Tiểu học)

Trên bàn cờ vua hãy sắp xếp đúng 8 quân Hậu sao cho không còn con nào có thể ăn được con nào. Hãy tìm ra nhiều cách sắp nhất?

Bài 33/2000 - Mã hoá văn bản

(Dành cho học sinh THCS)

Bài toán sau mô tả một thuật toán mã hoá đơn giản (để tiện ta lấy ví dụ tiếng Anh, các bạn có thể mở rộng cho tiếng Việt):

Tập hợp các chữ cái tiếng Anh bao gồm 26 chữ cái được đánh số thứ tự từ 0 đến 25 như sau:

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	Z

Quy tắc mã hoá một ký tự như sau (lấy ví dụ ký tự X):

- Tìm số thứ tự tương ứng của ký tự ta được 23
- Tăng giá trị số này lên 5 ta được 28
- Tìm số dư trong phép chia số này cho 26 ta được 2
- Tra ngược bảng chữ cái ta thu được C.

a. Sử dụng quy tắc trên để mã hoá các dòng chữ sau:

PEACE
HEAL THE WORLD
I LOVE SPRING

b. Hãy tìm ra quy tắc giải mã các dòng chữ sau:

N FR F XYZIJSY
NSKTVRFYNHX

MFSTN SFYNTSFQ ZSNBJVXNYD

Bài 34/2000 - Mã hoá và giải mã*(Dành cho học sinh THCS)*

Theo quy tắc mã hoá ở bài trên (33/2000), hãy viết chương trình cho phép:

- Nhập một chuỗi ký tự và in ra chuỗi ký tự đã được mã hóa
- Nhập một chuỗi ký tự đã được mã hoá và in ra chuỗi ký tự đã được giải mã.

Ví dụ khi chạy chương trình:

Nhập chuỗi ký tự:

PEACE ↵

Chuỗi ký tự trên được mã hoá là:

UJFHJ

Nhập chuỗi ký tự cần giải mã:

FR ↵

Chuỗi ký tự trên được giải mã là:

AM_

Bài 35/2000 - Các phân số được sắp xếp*(Dành cho học sinh THPT)*Xét tập $F(N)$ tất cả các số hữu tỷ trong đoạn $[0,1]$ với mẫu số không vượt quá N .Ví dụ tập $F(5)$:

0/1 1/5 1/4 1/3 2/5 1/2 3/5 2/3 3/4 4/5 1/1

Hãy viết chương trình cho phép nhập số nguyên N nằm trong khoảng từ 1 đến 100 và xuất ra theo thứ tự tăng dần các phân số trong tập $F(N)$ cùng số lượng các phân số đó.

Ví dụ khi chạy chương trình:

Nhập số N : 5 ↵

0/1 1/5 1/4 1/3 2/5 1/2 3/5 2/3 3/4 4/5 1/1

Tất cả có 11 phân số_

Bài 36/2000 - Anh chàng hà tiện*(Dành cho học sinh Tiểu học)*

Một chàng hà tiện ra hiệu may quần áo. Người chủ hiệu biết tính khách nên nói với anh ta: “Tôi tính tiền công theo 2 cách: cách thứ nhất là lấy đúng 11700 đồng. Cách thứ hai là lấy theo tiền cúc: chiếc cúc thứ nhất tôi lấy 1 đồng, chiếc cúc thứ 2 tôi lấy 2 đồng gấp đôi chiếc thứ nhất, chiếc cúc thứ 3 tôi lấy 4 đồng gấp đôi lần chiếc cúc thứ 2 và cứ tiếp tục như thế cho đến hết. áo của anh có 18 chiếc cúc. Nếu anh thấy cách thứ nhất là đắt thì anh có thể trả tôi theo cách thứ hai.”

Sau một hồi suy nghĩ chàng hà tiện quyết định chọn theo cách thứ hai. Hỏi anh ta phải trả bao nhiêu tiền và anh ta có bị “hớ” hay không?

Bài 37/2000 - Số siêu nguyên tố*(Dành cho học sinh THCS)*

Số siêu nguyên tố là số nguyên tố mà khi bỏ một số tùy ý các chữ số bên phải của nó thì phần còn lại vẫn tạo thành một số nguyên tố.

Ví dụ 7331 là một số siêu nguyên tố có 4 chữ số vì 733, 73, 7 cũng là các số nguyên tố. Nhiệm vụ của bạn là viết chương trình nhập dữ liệu vào là một số nguyên N ($0 < N < 10$) và đưa ra kết quả là một số siêu nguyên tố có N chữ số cùng số lượng của chúng.

Ví dụ khi chạy chương trình:

Nhap so N: 4 ↵

Cac so sieu nguyen to co 4 chu so la: 2333 2339 2393 2399 2939 3119 3137
3733 3739 3793 3797 5939 7193 7331 7333 7393

Tat ca co 16 so_

Bài 38/2000 - Tam giác số

(Dành cho học sinh THPT)

Hình sau mô tả một tam giác số có số hàng $N=5$:

```

      7
     3 8
    8 1 0
   2 7 4 4
  4 5 2 6 5

```

Đi từ đỉnh (số 7) đến đáy tam giác bằng một đường gấp khúc, mỗi bước chỉ được đi từ số ở hàng trên xuống một trong hai số đứng kề bên phải hay bên trái ở hàng dưới, và cộng các số trên đường đi lại ta được một tổng.

Ví dụ: đường đi 7 8 1 4 6 có tổng là $S=26$, đường đi 7 3 1 7 5 có tổng là $S=23$

Trong hình trên, tổng $S_{\max}=30$ theo đường đi 7 3 8 7 5 là tổng lớn nhất trong tất cả các tổng.

Nhiệm vụ của bạn là viết chương trình nhận dữ liệu vào là một tam giác số chứa trong text file INPUT.TXT và đưa ra kết quả là giá trị của tổng S_{\max} trên màn hình.

File INPUT.TXT có dạng như sau:

Dòng thứ 1: có duy nhất 1 số N là số hàng của tam giác số ($0 < N < 100$).

N dòng tiếp theo, từ dòng thứ 2 đến dòng thứ $N+1$: dòng thứ i có $(i-1)$ số cách nhau bởi dấu trống (space).

Ví dụ: với nội dung của file INPUT.TXT là

```

5
7
3 8
8 1 0
2 7 4 4
4 5 2 6 5

```

thì kết quả chạy chương trình sẽ là: $S_{\max}=30$.

Bài 39/2000 - Ô chữ*(Dành cho học sinh THCS và THPT)*

Trò chơi ô chữ thông dụng 30 năm trước của trẻ em gồm một khung ô chữ kích thước 5x5 chứa 24 hình vuông nhỏ kích thước như nhau. Trên mặt mỗi hình vuông nhỏ có in một chữ cái trong bảng chữ cái. Vì chỉ có 24 hình vuông trong ô chữ nên trong ô chữ còn thừa ra một ô trống, có kích thước đúng bằng kích thước các hình vuông. Một hình vuông có thể đẩy trượt vào ô trống đó nếu nó nằm ngay sát bên trái, bên phải, bên trên hay bên dưới ô trống. Mục tiêu của trò chơi là trượt các hình vuông vào ô trống sao cho cuối cùng các chữ cái trong ô chữ được xếp theo đúng thứ tự của chúng trong bảng chữ cái. Hình sau đây minh họa một ô chữ với cấu hình ban đầu và cấu hình của nó sau 6 nước đi sau:

1. Trượt hình vuông phía trên ô trống.
2. Trượt hình vuông bên phải ô trống.
3. Trượt hình vuông bên phải ô trống.
4. Trượt hình vuông phía dưới ô trống.
5. Trượt hình vuông phía dưới ô trống.
6. Trượt hình vuông bên trái ô trống.

T	R	G	S	J
X	D	O	K	I
M		V	L	N
W	P	A	B	E
U	Q	H	C	F

Cấu hình ban đầu của ô chữ

TT	RR	GG	SS	JJ
XX	OO	KK	LL	II
MM	DD	VV	BB	NN
WW	PP		AA	EE
UU	QQ	HH	CC	FF

Cấu hình của ô chữ sau 6

Bạn hãy viết một chương trình của bạn chứa cấu hình ban đầu của ô chữ cùng các nước đi để vẽ ra ô chữ kết quả.

Input

Đầu vào của chương trình của bạn chứa cấu hình ban đầu của một ô chữ và một dãy các nước đi trong ô chữ đó.

Năm dòng đầu tiên mô tả cấu hình ban đầu của ô chữ, mỗi dòng tương ứng với một hàng của ô chữ và chứa đúng 5 ký tự tương ứng với 5 hình vuông của ô chữ trên hàng đó. Ô trống được diễn tả bằng một dấu cách.

Các dòng tiếp theo sau là dãy các nước đi. Dãy các nước đi được ghi bằng dãy các chữ A, B, R và L để thể hiện hình vuông nào được trượt vào ô trống. A thể hiện hình vuông phía trên ô trống được trượt vào ô trống, tương ứng: B-phía dưới, R-bên phải, L-bên trái. Có thể có những nước đi không hợp cách, ngay cả khi nó được biểu thị bằng những chữ cái trên. Nếu xuất hiện một nước đi không hợp cách thì ô chữ coi như không có cấu hình kết quả. Dãy các nước đi có thể chiếm một số dòng, nhưng nó sẽ được xem là kết thúc ngay khi gặp một số 0.

Out put

Nếu ô chữ không có cấu hình kết quả thì thông báo 'This puzzle has no final configuration.'; ngược lại thì hiển thị cấu hình ô chữ kết quả. Định dạng mỗi dòng kết quả bằng cách đặt một dấu cách vào giữa hai kí tự kế tiếp nhau. Ô trống cũng được xử lý như vậy. Ví dụ nếu ô trống nằm bên trong hàng thì nó được xuất hiện dưới dạng 3 dấu cách: một để ngăn cách nó với kí tự bên trái, một để thể hiện chính ô trống đó, còn một để ngăn cách nó với kí tự bên phải.

Chú ý: Input mẫu đầu tiên tương ứng với ô chữ được minh hoạ trong ví dụ trên.

Sample Input 1

```
TRGSJ
XDOKI
M VLN
WPABE
UQHCF
ARRBBL0
```

Sample Output 1

```
T R G S J
X O K L I
M D V B N
W P   A E
U Q H C F
```

Sample Input 2

```
AB C DE
F G H I J
KLMNO
PQRS
TUVWX
AAA
LLLL0
```

Sample Output 2

```
  A B C D
F G H I E
K L M N J
P Q R S O
T U V W X
```

Sample Input 3

```
ABCDE
FGHIJ
KLMNO
PQRS
TUVWX
AAAAABBRRLL0
```

Sample Output 3

This puzzle has no final configuration.

Bài 40/2000 - Máy định vị Radio

Một con tàu được trang bị ăng-ten định hướng có thể xác định vị trí hiện thời của mình nhờ các lần đọc đèn hiệu địa phương. Mỗi đèn hiệu được đặt ở một vị trí đã biết và phát ra một tín hiệu đơn nhất. Mỗi khi bắt được tín hiệu, tàu liền quay ăng-ten của mình cho đến khi đạt được tín hiệu cực đại. Điều đó cho phép xác định được phương vị tương đối của đèn hiệu. Cho biết dữ liệu của lần đọc trước (thời gian, phương vị tương đối, vị trí của đèn), một lần đọc mới đủ để xác định vị trí hiện thời của tàu. Bạn phải viết một chương trình xác định vị trí hiện thời của tàu từ hai lần đọc đèn hiệu.

Vị trí của các đèn hiệu và các con tàu được cho trong hệ tọa độ vuông góc, trục Ox hướng về phía đông, còn Oy hướng về phía bắc. Hướng đi của con tàu được đo bằng độ, theo chiều kim đồng hồ tính từ hướng bắc. Như vậy, hướng bắc sẽ là 0^0 , hướng đông là 90^0 , hướng nam là 180^0 và hướng tây là 270^0 . Phương vị tương đối của đèn hiệu cũng được đo bằng độ, tương đối với hướng đi của tàu và theo chiều kim đồng hồ. ăng ten không thể chỉ ra đèn hiệu nằm ở hướng nào trên phương vị. Như vậy, một phương vị 90^0 có nghĩa là đèn hiệu có thể nằm ở hướng 90^0 hoặc 270^0 .

Input

Dòng đầu tiên của input là một số nguyên chỉ số lượng các đèn hiệu (nhiều nhất là 30). Mỗi dòng tiếp theo cho một đèn hiệu. Mỗi dòng bắt đầu bằng tên đèn (là một chuỗi kí tự không vượt quá 20 kí tự), sau đó là vị trí của đèn cho bằng hoành độ và tung độ. Các trường này phân cách bởi một dấu cách.

Dòng tiếp theo ngay sau các dữ liệu về đèn hiệu là một số nguyên chỉ số lượng các kịch bản đường đi của tàu. Mỗi kịch bản chứa 3 dòng gồm một dòng cho biết hướng đi của tàu so với hướng Bắc và vận tốc vận tốc thực của tàu, và hai dòng chỉ hai lần đọc đèn hiệu. Thời gian được đo bằng phút, tính từ lúc nửa đêm trong vòng 24 giờ. Vận tốc đo bằng đơn vị độ dài (như các đơn vị của hệ trục tọa độ) trên đơn vị thời gian. Dòng thứ hai của kịch bản là lần đọc thứ nhất gồm thời gian (là một số nguyên), tên đèn và góc phương vị tương đối với hướng đi của tàu. Ba trường được ngăn cách nhau bởi một dấu cách. Dòng thứ ba của kịch bản là lần đọc thứ hai. Thời gian của lần đọc này luôn lớn hơn lần đọc thứ nhất.

Output

Với mỗi kịch bản, chương trình của bạn phải chỉ ra được số thứ tự của kịch bản (Scenario 1, Scenario 2,...), và một thông báo về vị trí của con tàu (được làm tròn đến hai chữ số thập phân) tại thời điểm của lần đọc *thứ hai*. Nếu vị trí của tàu không thể xác định thì thông báo: "Position cannot be determined."

Mẫu input và output chính xác tương ứng được cho như sau:

Sample Input

4

First 2.0 4.0

Second 6.0 2.0

Third 6.0 7.0

Fourth 10.0 5.0

2

0.0 1.0

1 First 270.0

2 Fourth 90.0

116.5651 2.2361

4 Third 126.8699

5 First 319.3987

Sample Output

Scenario 1: Position cannot be determined

Scenario 2: Position is (6.00, 5.00)

Bài 41/2000 - Cờ Othello

(Dành cho học sinh THCS và THPT)

Cờ Othello là trò chơi cho 2 người trên một bàn cờ kích thước 8×8 ô, dùng những quân tròn một mặt đen, một mặt trắng. Các đấu thủ sẽ được lần lượt đi một quân vào ô còn trống trên bàn cờ. Khi đi một quân, đấu thủ phải lật được ít nhất một quân của đấu thủ kia. Các quân sẽ lật được nếu chúng nằm liên tiếp trên cùng một đường thẳng (ngang, dọc hoặc chéo) mà ở hai đầu của đường đó là hai quân có màu của đấu thủ đang đi. Khi xong một lượt đi, tất cả các quân đã bị lật đã được đổi sang màu của đấu thủ vừa đi. Trong một lượt đi có thể lật được nhiều hàng.

Ví dụ: Nếu thế cờ hiện thời ở bàn cờ bên trái và lượt đi là của đấu thủ trắng, thì anh ta có thể đi được một trong các nước sau: (3,5) (4,6) (5,3) (6,4). Nếu anh ta đi nước (3,5) thì sau nước đi thế cờ sẽ như ở bàn cờ bên phải.

Vẽ bàn cờ

Bạn hãy viết một chương trình để đọc một ván cờ từ một text file có qui cách:

8 dòng đầu tiên là bàn cờ thế, mỗi dòng chứa 8 kí tự, mỗi kí tự có thể là:

'.' thể hiện một ô trống,

'B' thể hiện một ô có quân đen,

'W' thể hiện một ô có quân trắng.

Dòng thứ 9 chứa một trong hai kí tự 'B' hoặc 'W' để chỉ nước đi thuộc về đấu thủ nào.

Các dòng tiếp theo là các lệnh. Mỗi lệnh có thể là: liệt kê tất cả các nước đi có thể của đấu thủ hiện thời, thực hiện một nước đi, hay thôi chơi ván cờ đó. Mỗi lệnh ghi trên một dòng theo qui cách sau:

Liệt kê tất cả các nước đi có thể của đấu thủ hiện thời:

Lệnh là một chữ 'L' ở cột đầu tiên của dòng. Chương trình phải kiểm tra cả bàn cờ và in ra tất cả các nước đi hợp lệ của đấu thủ hiện thời theo dạng (x,y) trong đó x là hàng và y là cột của nước đi. Các nước đi này phải được in theo qui cách:

+ Mọi nước đi trên hàng i sẽ được in trước mỗi nước đi trên hàng j nếu $j > i$.

+ Nếu trên hàng i có nhiều hơn 1 nước đi thì các nước đi được in theo thứ tự của cột.

Mọi nước đi hợp lệ phải in trên một dòng. Nếu không có nước đi nào hợp lệ vì đấu thủ hiện thời không thể lật bất cứ một quân nào thì phải in ra thông báo 'No legal move'.

Thực hiện một nước đi

Lệnh là một chữ 'M' ở cột đầu tiên của dòng, tiếp theo sau là 2 chữ số ở cột thứ hai và thứ ba của dòng. Các chữ số chỉ ra hàng và cột của ô trống trên bàn cờ nơi đấu thủ hiện thời sẽ đặt quân của mình, trừ phi anh ta không có nước đi hợp lệ nào. Nếu đấu thủ hiện thời không có nước đi hợp lệ nào thì anh ta được thay bởi đấu thủ kia và bây giờ nước đi là của đấu thủ mới. Chương trình phải kiểm tra khi đó nước đi là hợp lệ. Bạn sẽ phải ghi nhận sự thay đổi trên bàn cờ, kể cả việc thêm các quân mới lẫn việc thay đổi màu sắc quân cờ bị lật. Cuối mỗi nước đi hãy in ra số lượng tất cả các quân cờ mỗi màu trên bàn cờ theo qui cách 'Black - xx White - yy, trong đó xx là số lượng các quân đen còn yy là số lượng các quân trắng. Sau một nước đi, đấu thủ hiện thời được thay bởi đấu thủ kia.

Thoại chơi ván cờ đó

Lệnh là một chữ 'Q' ở cột đầu tiên của dòng, dòng lệnh này kết thúc Input cho ván cờ đang xét. Chương trình phải in thế cờ cuối cùng của ván cờ theo qui cách được dùng ở input.

Bạn phải kiểm tra tính chính xác của các lệnh. Không được để dòng trắng ở bất cứ nơi nào trong output.

Bài 42/2000 - Một chút về tư duy số học

(Dành cho học sinh Tiểu học)

Tìm số tự nhiên nhỏ nhất khi chia cho 2, 3, 4, 5, 6, 7, 8, 9, 10 cho phần dư tương ứng là 1, 2, 3, 4, 5, 6, 7, 8, 9.

Bài 43/2000 - Kim giờ và phút gặp nhau bao nhiêu lần trong ngày

(Dành cho học sinh Tiểu học)

Đồng hồ quả lắc có 2 kim: giờ và phút. Tính xem trong vòng 1 ngày đêm (từ 0h - 24h) có bao nhiêu lần 2 kim gặp nhau và đó là những lúc nào.

Bài 44/2000 - Tạo ma trận số

(Dành cho học sinh THCS)

Cho trước số nguyên dương N bất kỳ. Hãy viết thuật toán và chương trình để tạo lập bảng NxN phần tử nguyên dương theo quy luật được cho trong ví dụ sau:

1 2 3 4 5 6

2 4 6 8 10 12

3 6 9 12 2 4

4 8 12 2 4 6

5 10 2 4 6 8

6 12 4 6 8 10

Thực hiện chương trình đó trên máy với N=12, đưa ra màn hình ma trận kết quả (có dạng như trong ví dụ).

Bài 45/2000 - Các vòng tròn Olympic

(Dành cho học sinh THPT)

Có 5 vòng tròn Olympic chia mặt phẳng thành 15 phần (không kể phần vô hạn) (hình vẽ). Hãy đặt vào mỗi phần đó một số sao cho tổng số các số trong mỗi vòng tròn bằng 39.

Lập chương trình giải quyết bài toán trên và cho biết có bao nhiêu cách xếp như vậy.

Bài 46/2000 - Đảo chữ cái

(Dành cho học sinh THCS và THPT)

Bạn phải viết chương trình đưa ra tất cả các từ có thể có phát sinh từ một tập các chữ cái.

Ví dụ: Cho từ "abc", chương trình của bạn phải đưa ra được các từ "abc", "acb", "bac", "bca", "cab" và "cba" (bằng cách khảo sát tất cả các trường hợp khác nhau của tổ hợp ba chữ cái đã cho).

Input

Dữ liệu vào được cho trong tệp input.txt chứa một số từ. Dòng đầu tiên là một số tự nhiên cho biết số từ được cho ở dưới. Mỗi dòng tiếp theo chứa một từ. Trong đó, một từ có thể chứa cả chữ cái thường hoặc hoa từ A đến Z. Các chữ thường và hoa được coi như là khác nhau. Một chữ cái nào đó có thể xuất hiện nhiều hơn một lần.

Output

Với mỗi từ đã cho trong file Input.txt, kết quả nhận được ra file Output.txt phải chứa tất cả các từ khác nhau được sinh từ các chữ cái của từ đó. Các từ được sinh ra từ một từ đã cho phải được đưa ra theo thứ tự tăng dần của bảng chữ cái.

Sample Input

2

abc

acba

Sample Output

abc

acb

bac

bca

cab

cba

aabc

aacb

abac

abca

acab

acba

baac

baca

bcaa

caab

caba

cbaa

Bài 47/2000 - Xoá số trên vòng tròn

(Dành cho học sinh THCS và PTTH)

Các số từ 1 đến 2000 được xếp theo thứ tự tăng dần trên một đường tròn theo chiều kim đồng hồ. Bắt đầu từ số 1, chuyển động theo chiều kim đồng hồ, cứ bước qua một số lại xoá đi một số. Công việc đó tiếp diễn cho đến khi trên vòng tròn còn lại đúng một số. Lập chương trình tính và in ra số đó.

Bài 48/2000 - Những chiếc gậy

(Dành cho học sinh THCS và THPT)

George có những chiếc gậy với chiều dài như nhau và chặt chúng thành những đoạn có chiều dài ngẫu nhiên cho đến khi tất cả các phần trở thành đều có chiều dài tối đa là 50 đơn vị. Bây giờ anh ta muốn ghép các đoạn lại như ban đầu nhưng lại quên mất nó như thế nào và chiều dài ban đầu của chúng là bao nhiêu. Hãy giúp George thiết kế chương trình để ước tính nhỏ nhất có thể của chiều dài những cái gậy này. Tất cả chiều dài được biểu diễn bằng đơn vị là những số nguyên lớn hơn 0.

Input

Dữ liệu vào trong file Input.txt chứa các khối mỗi khối 2 dòng. Dòng đầu tiên chứa số phần của chiếc gậy sau khi cắt. Dòng thứ 2 là chiều dài của các phần này cách nhau bởi một dấu cách. Dòng cuối cùng kết thúc file Input là số 0.

Output

Kết quả ra trong file Output.txt chứa chiều dài nhỏ nhất có thể của những cái gậy, mỗi chiếc trong mỗi khối trên một dòng.

Sample Input

```
9
5 2 1 5 2 1 5 2 1
4
1 2 3 4
0
```

Sample Output

```
6
5
```

Bài 49/2001 - Một chút nhanh trí

(Dành cho học sinh Tiểu học)

Số tự nhiên A có tính chất là khi chia A và lập phương của A cho một số lẻ bất kỳ thì nhận được số dư như nhau. Tìm tất cả các số tự nhiên như vậy.

Bài 50/2001 - Bài toán đổi màu bi

(Dành cho học sinh THCS và THPT)

Trên bàn có N1 hòn bi xanh, N2 hòn bi đỏ và N3 hòn bi vàng. Luật chơi như sau:

Nếu 2 hòn bi khác màu nhau chạm nhau thì chúng sẽ cùng biến thành màu thứ 3 (ví dụ: xanh, vàng --> đỏ, đỏ).

Tìm thuật toán và lập chương trình cho biết rằng có thể biến tất cả các hòn bi đỏ thành một màu đỏ có được không?

Bài 51/2001 - Thay thế từ

(Dành cho học sinh THCS và PTTH)

Hai file INPUT1.TXT và INPUT2.TXT được cho như sau: File INPUT1.TXT chứa một đoạn văn bản bất kì. File INPUT2.TXT chứa không quá 50 dòng, mỗi dòng gồm hai từ: từ đầu là từ đích và từ sau là từ nguồn. Hãy tìm trong file INPUT1.TXT tất cả các từ là từ đích và thay thế chúng bằng các từ nguồn tương ứng. Kết quả ghi vào file KQ.OUT (sẽ là một đoạn văn bản tương tự như trong file INPUT1.TXT nhưng đã được thay thế từ đích bởi từ nguồn).

Sample INPUT

❖ File INPUT1.TXT chứa đoạn văn bản sau:

Nam moi sap den roi, ban co zui khong?

Chuc cac ban don mot cai Tet that vui ve va hanh phuc.

Chuc ban luon hoc gioi!

❖ File INPUT2.TXT chứa các dòng sau:

ban em

zui vui

Sample OUTPUT

❖ File KQ.OUT sẽ chứa đoạn văn bản sau:

Nam moi sap den roi, em co vui khong?

Chuc cac em don mot cai Tet that vui ve va hanh phuc.

Chuc em luon hoc gioi!

Bài 52/2001 - Xác định các tứ giác đồng hồ trong ma trận

(Dành cho học sinh THCS và THPT)

Cho ma trận vuông $A[i,j]$ ($i, j = 1, 2 \dots n$). Các phần tử của A được đánh số từ 1 đến $n \times n$.

Gọi S là số lượng các "tứ giác" có bốn đỉnh là: $A[i,j]$; $A[i,j+1]$; $A[i+1,j]$; $A[i+1,j+1]$ sao cho các số ở đỉnh của nó xếp theo thứ tự tăng dần theo chiều kim đồng hồ (tính từ một đỉnh nào đó).

1) Lập chương trình tính số lượng S.

2) Lập thuật toán xác định A sao cho số S là:

a. Lớn nhất.

b. Nhỏ nhất.

Bài 53/2001 - Lập lịch tháng kỳ ảo*(Dành cho học sinh THCS và THPT)*

Lịch của các tháng được biểu diễn bằng một ma trận có số cột bằng 7 và số hàng nhỏ hơn hoặc bằng 6.

		1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30			

Ví dụ: Trong hình vẽ, lịch này thỏa mãn tính chất sau: Mọi ma trận con 3×3 không có ô trống đều là ma trận "kỳ ảo" theo nghĩa: Tổng các số của mỗi đường chéo bằng tổng của trung bình cộng của tất cả các cột và hàng. Hãy xây dựng tất cả các lịch tháng có tính chất như trên. Lập chương trình mô tả tất cả các khả năng xảy ra.

Bài 54/2001 - Bạn hãy gạch số*(Dành cho học sinh Tiểu học và THCS)*

Chúng ta viết liên tiếp 10 số nguyên tố đầu tiên theo thứ tự tăng để tạo thành một số có nhiều chữ số. Trong số này hãy gạch đi một nửa số chữ số để số còn lại là:

- Nhỏ nhất
- Lớn nhất

Trong từng trường hợp phải nêu cụ thể thuật giải (tại sao lại gạch như vậy)?

Bài 55/2001 - Bài toán che mắt mèo*(Dành cho học sinh THCS và THPT)*

Trên bàn cờ ô vuông $N \times N$ tại mỗi ô có thể xếp hoặc một con mèo con, hoặc một quân cờ. Hai con mèo trên bàn cờ sẽ nhìn thấy nhau nếu trên đường thẳng nối chúng theo hàng ngang, hàng dọc hay đường chéo không có quân cờ nào cả.

Hãy tìm cách xếp mèo và quân cờ như trên sao cho số mèo lớn nhất mà không có hai con mèo nào nhìn thấy nhau?

Bài 56/2001 - Chia lưới*(Dành cho học sinh THPT)*

Cho lưới $M \times N$ ($m, n \leq 20$) ô vuông, trong mỗi ô cho trước một số tự nhiên. Hãy tìm cách chia lưới trên làm hai phần (chia theo cạnh lưới) sao cho trị tuyệt đối hiệu số của tổng các số trong mỗi phần có giá trị nhỏ nhất (như hình dưới đây).

				7	
	1	3	5		
	12	2	5		
	9	2	10		

Dữ liệu được cho trong file LUOI.INP, được cho như sau:

- Dòng đầu tiên gồm 2 số m, n là kích thước của ô lưới.
- m dòng tiếp theo, mỗi dòng gồm n số cách nhau bởi dấu cách, ô nào không có giá trị được cho bằng 0.

Dữ liệu ra trong file LUOI.OUT miêu tả lưới sau khi chia thành hai phần: là một ma trận kích thước $m \times n$ gồm các số 0 và 1 (số 0 kí hiệu cho các ô tương ứng với phần thứ nhất, và số 1 kí hiệu cho các ô tương ứng với phần thứ hai).

Sample Input:

Dữ liệu cho sau đây tương ứng với hình trên:

```
5 6
0 0 0 0 7 0
0 1 3 5 0 0
0 12 2 5 0 0
0 9 2 10 0 0
0 0 0 0 0 0
```

Sample Output:

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

Bài 57/2001 - Chọn số

(Dành cho học sinh Tiểu học và THCS)

Cho 2000 số $a_1, a_2, \dots, a_{2000}$ mỗi số là +1 hoặc -1. Hỏi có thể hay không từ 2000 số đó chọn ra các số nào đó để tổng các số được chọn ra bằng tổng các số còn lại? Giả sử cho 2001 số, liệu có thể có cách chọn không? Nêu cách giải tổng quát.

Bài 58/2001 - Tổng các số tự nhiên liên tiếp

(Dành cho học sinh THCS và THPT)

Cho trước số tự nhiên n . Lập thuật toán cho biết n có thể biểu diễn thành tổng của hai hoặc nhiều số tự nhiên liên tiếp hay không?

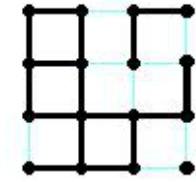
Trong trường hợp có, hãy thể hiện tất cả các cách có thể có.

Bài 59/2001 - Đếm số ô vuông

(Dành cho học sinh THCS và THPT)

Cho một bảng vuông gồm $N \times N$ điểm nằm trên các mắt lưới ô vuông. Các điểm kề nhau trên một hàng hay một cột có thể được nối với nhau bằng một đoạn thẳng hoặc không

được nối. Các đoạn đó sẽ tạo ra các ô vuông trên bảng. Ví dụ với bảng sau đây thì $n = 4$ và có 3 ô vuông:



Trên mỗi hàng có thể có nhiều nhất $n-1$ đoạn thẳng nằm ngang và có tất cả n hàng như vậy. Tương tự như vậy có tất cả $n-1$ hàng các đoạn thẳng nằm dọc và trên mỗi hàng có thể có nhiều nhất n đoạn.

Để mô tả người ta dùng hai mảng nhị phân: một mảng ghi các đoạn nằm ngang kích thước $n \times (n-1)$, và một mảng ghi các đoạn nằm dọc kích thước $(n-1) \times n$. Trong mảng, số 1 dùng để mô tả đoạn thẳng nối giữa 2 điểm, còn số 0 miêu tả giữa hai điểm không có đoạn thẳng nối. Trong ví dụ trên thì ma trận "ngang" là:

$$\begin{bmatrix} 1 & 0 & 1 \\ 1 & 0 & 0 \\ 1 & 1 & 1 \\ 1 & 1 & 0 \end{bmatrix}$$

và ma trận "dọc" là:

$$\begin{bmatrix} 1 & 1 & 1 & 0 \\ 1 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix}$$

Cho trước ma trận "ngang" và ma trận "dọc", dữ liệu nhập từ các tệp văn bản có tên là NGANG.INP và DOC.INP. Hãy lập trình đếm số các ô vuông trên bảng.

Bài 60/2001 - Tìm số dư của phép chia

(Dành cho học sinh Tiểu học)

Một số nguyên khi chia cho 1976 và 1977 đều dư 76. Hỏi số đó khi chia cho 39 dư bao nhiêu?

Bài 61/2001 - Thuật toán điền số vào ma trận

(Dành cho học sinh THCS và THPT)

lập thuật toán điền các phần tử của ma trận $N \times N$ các số 0, 1 và -1 sao cho:

- Tổng các số của mọi hình vuông con 2×2 đều bằng 0.
- Tổng các số của ma trận trên là lớn nhất.

Bài 62/2001 - Chèn Xâu

(Dành cho học sinh THCS và THPT)

Cho một xâu $S = '123456789'$ hãy tìm cách chèn vào S các dấu '+' hoặc '-' để thu được số M cho trước (nếu có thể). Số M nguyên được nhập từ bàn phím. Trong file Output Chenxau.Out ghi tất cả các phương án chèn (nếu có) và ghi "Khong co" nếu như không thể thu được M từ cách làm trên.

Ví dụ: Nhập $M = 8$, một trong các phương án đó là: $'-1+2-3+4+5-6+7'$;

$M = -28$, một trong các phương án đó là: $'-1+2-34+5'$;

(Đề ra của bạn: Lê Nhân Tâm - 12 Tin Trường THPT Lam Sơn)

Bài 63/2001 - Tìm số nhỏ nhất

(Dành cho học sinh Tiểu học)

Hãy viết ra số nhỏ nhất bao gồm tất cả các chữ số 0, 1, 2, 3, ... 9 mà nó:

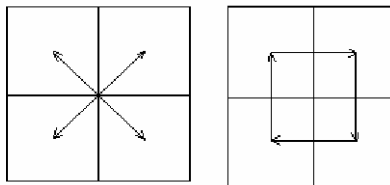
- Chia hết cho 9
- Chia hết cho 5
- Chia hết cho 20

Có giải thích cho từng trường hợp?

Bài 64/2001 - Đôi ma trận số

(Dành cho học sinh THCS và THPT)

Cho mảng số thực vuông A kích thước $2n \times 2n$. Hãy lập các mảng mới bằng cách đổi chỗ các khối vuông kích thước $n \times n$ của A theo các cách sau:



a.

b.

Bài 65/2001 - Lưới ô vuông vô hạn

(Dành cho học sinh THCS và THPT)

Cho lưới ô vuông vô hạn về hai phía (trên và phải). Các ô của lưới được đánh số theo quy tắc sau:

- Ô trái dưới - vị trí $(0,0)$ - được đánh số 0.
- Các ô còn lại được đánh số theo nguyên tắc lan tỏa từ vị trí $(0,0)$ và theo quy tắc: tại một vị trí số được điền vào là số nguyên không âm nhỏ nhất chưa được điền trên hàng và cột chứa ô hiện thời. Ví dụ, ta có hình dạng của một số ô của lưới như sau:

3	2	1	0	
2	3	0	1	
1	0	3	2	
0	1	2	3	

Cho trước cặp số tự nhiên M, N - kích thước ô lưới. Hãy viết chương trình mô tả lưới trên, kết quả được ghi vào file KQ.TXT.

Bài 66/2001 - Bảng số 9 x 9

(Dành cho học sinh Tiểu học và THCS)

Hãy xếp các số 1, 2, 3, ..., 81 vào bảng 9 x 9 sao cho:

- Trên mỗi hàng các số được xếp theo thứ tự tăng dần (từ trái qua phải).
- Tổng các số ở cột 5 là lớn nhất.

Yêu cầu:

- + Đối với các bạn học sinh khối Tiểu học chỉ cần viết ra bảng số thoả mãn tính chất trên.
- + Các bạn học sinh khối THCS thì phải lập trình hiển thị kết quả ra màn hình.

Bài 67/2001 - Về các phép biến đổi "Nhân 2 trừ 1"

(Dành cho học sinh THCS và THPT)

Cho ma trận A kích thước $M \times N$, A_{ij} - là các số tự nhiên. Các phép biến đổi có thể là:

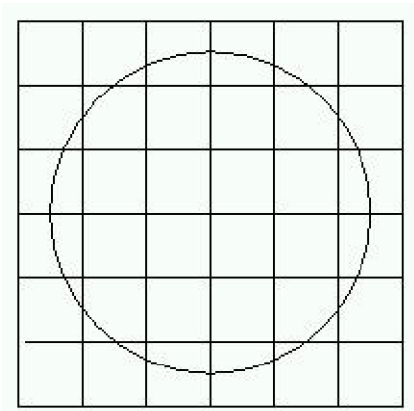
- Nhân tất cả các số của một hàng với 2.
- Trừ tất cả các số của một cột cho 1.

Tìm thuật toán sao cho sau một số phép biến đổi trên ma trận A trở thành toàn số 0.

Bài 68/2001 - Hình tròn và bảng vuông

(Dành cho học sinh THPT)

Một đường tròn đường kính $2n - 1$ đơn vị được vẽ giữa bàn cờ $2n \times 2n$. Với $n = 3$ được minh hoạ như dưới đây:



Viết chương trình xác định số ô vuông của bảng bị cắt bởi hình tròn và số ô vuông nằm hoàn toàn trong hình tròn.

Dữ liệu vào trong file Input.txt bao gồm: Mỗi dòng là một số nguyên dương không lớn hơn 150 - là các giá trị của n .

Dữ liệu ra trong file Output.txt: Với mỗi giá trị vào n , kết quả ra phải tính được số ô vuông bị cắt bởi hình tròn và số ô vuông nằm hoàn toàn trong hình tròn, mỗi số trên một dòng. Mỗi kết quả tương ứng với một giá trị n phải cách nhau một dòng.

Sample Input

3
4

Sample Output

20
12

28
24

Bài 69/2001 - Bội của 36*(Dành cho học sinh Tiểu học)*

Tìm số tự nhiên nhỏ nhất chia hết cho 36 mà trong dạng viết thập phân của nó có chứa tất cả các chữ số từ 1 tới 9.

Bài 70/2001 - Mã hoá theo khoá*(Dành cho học sinh THCS và THPT)*

Cho trước khoá là một hoán vị của n số $(1, 2, \dots, n)$. Khi đó để mã hoá một xâu kí tự ta có thể chia xâu thành từng nhóm n kí tự (riêng nếu nhóm cuối cùng không đủ n kí tự thì ta có thể thêm các dấu cách vào sau cho đủ) rồi hoán vị các kí tự trong từng nhóm. Sau đó, ghép lại theo thứ tự các nhóm ta được một xâu đã mã hoá.

Chẳng hạn: với khoá 3241 ($n=4$) thì ta có thể mã hoá xâu 'english' thành 'gnlehs i'.

Hãy viết chương trình mã hoá một xâu kí tự cho trước.

Bài 71/2001 - Thực hiện phép nhân*(Dành cho học sinh THCS và THPT)*

Bạn hãy lập chương trình nhập 2 số nguyên dương a và b . Sau đó thực hiện phép nhân $(a \times b)$ như cách nhân bằng tay thông thường. Ví dụ:

$$\begin{array}{r}
 \times 37 \\
 45 \\
 \hline
 185 \\
 148 \\
 \hline
 1665
 \end{array}$$

Bài 72/2001 - Biến đổi trên lưới số*(Dành cho học sinh THCS và THPT)*

Trên một lưới $N \times N$ các ô được đánh số 1 hoặc -1. Lưới trên được biến đổi theo quy tắc sau: một ô nào đó được thay thế bằng tích của các số trong các ô kề nó (kề cạnh). Lập chương trình thực hiện sao cho sau một số bước toàn lưới còn lại chữ số 1.

Bài 73/2001 - Bài toán chuỗi số*(Dành cho học sinh Tiểu học và THCS)*

Cho một chuỗi số có quy luật. Bạn có thể tìm được hai số cuối của dãy không, thay thế chúng trong dấu hỏi chấm (?). Bài toán không dễ dàng lắm đâu, vì chúng được tạo ra bởi một quy luật rất phức tạp. Bạn thử sức xem?

5 8 11 14 17 23 27 32 35 41 49 52 ? ?

Bài 74/2001 - Hai hàng số kỳ ảo*(Dành cho học sinh THCS và THPT)*

Hãy xếp $2N$ số tự nhiên $1, 2, \dots, 2N$ thành 2 hàng số:

A1, A2 ... An

B1, B2 ... Bn

Thỏa mãn điều kiện: tổng các số theo n cột bằng nhau, tổng các số theo các hàng bằng nhau.

Bài 75/2001 - Trò chơi Tích - Tắc vuông*(Dành cho học sinh THCS và THPT)*

Trên một lưới kẻ ô vuông có 2 người chơi như sau: người thứ nhất mỗi lần chơi sẽ đánh dấu x vào 1 ô trống. Người thứ hai được đánh dấu 0 vào 1 ô trống. Người thứ nhất muốn đạt được mục đích là đánh được 4 dấu x tạo thành 4 đỉnh của 1 hình vuông. Người thứ hai có nhiệm vụ ngăn cản mục đích đó của người thứ nhất.

Lập chương trình tìm thuật toán tối ưu cho người thứ nhất (người thứ nhất có thể luôn thắng).

Chú ý: Lưới ô vuông được coi là vô hạn về cả hai phía.

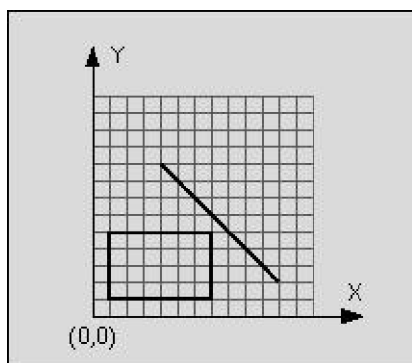
Bài 76/2001 - Đoạn thẳng và hình chữ nhật*(Dành cho học sinh THPT)*

Hãy viết một chương trình xác định xem một đoạn thẳng có cắt hình chữ nhật hay không?

Ví dụ:

Cho tọa độ điểm bắt đầu và điểm kết thúc của đường thẳng: (4,9) và (11,2);

Và tọa độ đỉnh trái trên và đỉnh phải dưới của hình chữ nhật: (1,5) và (7,1);



Hình 1: Đoạn thẳng không cắt hình chữ nhật

Đoạn thẳng được gọi là cắt hình chữ nhật nếu đoạn thẳng và hình chữ nhật có ít nhất một điểm chung.

Chú ý: mặc dù tất cả dữ liệu vào đều là số nguyên, nhưng tọa độ của các giao điểm tính ra chưa chắc là số nguyên.

Input

Dữ liệu vào trong file Input.Inp kiểm tra N trường hợp ($N \leq 1000$). Dòng đầu tiên của file dữ liệu vào là số N . Mỗi dòng tiếp theo chứa một trường hợp kiểm tra theo quy cách sau:

xstart ystart xend yend xleft ytop xright ybottom

trong đó: $(xstart, ystart)$ là điểm bắt đầu và $(xend, yend)$ là điểm kết thúc của đoạn thẳng. Và $(xleft, ytop)$ là đỉnh trái trên, $(xright, ybottom)$ là đỉnh phải dưới của hình chữ nhật. 8 số này được cách nhau bởi một dấu cách.

Output

Với mỗi một trường hợp kiểm tra trong file Input.txt, dữ liệu ra trong file Output.out phải đưa ra một dòng gồm hoặc là chữ cái "T" nếu đoạn thẳng cắt hình chữ nhật, hoặc là "F" nếu đoạn thẳng không cắt hình chữ nhật.

Ví dụ

Input.Inp

1

4 9 11 2 1 5 7 1

Output.out

F

Bài 77/2001 - Xoá số trên bảng

(Dành cho học sinh Tiểu học)

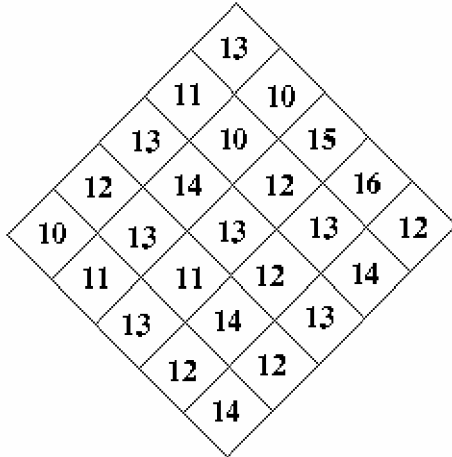
Trên bảng đen cô giáo ghi lên 23 số tự nhiên: 1, 2, 3, ..., 23

Các bạn được phép xoá đi 2 số bất kỳ trên bảng và thay vào đó một số mới là hiệu của chúng.

- Hỏi có thể thực hiện sau một số bước trên bảng còn lại toàn số 0 hay không? Nếu được hãy chỉ ra một cách làm cụ thể.
- Bài toán còn đúng không nếu thay số 23 bằng 25.

Bài 78/2001 - Cà rốt và những chú thỏ*(Dành cho học sinh Tiểu học)*

Các số ở mỗi ô trong hình thoi dưới đây biểu thị số lượng củ cà rốt. Chú thỏ đi từ góc dưới với 14 củ cà rốt và đi lên đỉnh trên với 13 củ cà rốt, chỉ được đi theo đường chéo, đi đến đâu ăn hết tổng số cà rốt trong ô đó. Hỏi rằng chú thỏ có thể ăn được nhiều nhất bao nhiêu củ cà rốt?

**Bài 79/2001 - Về một ma trận số***(Dành cho học sinh THCS)*

Mô tả thuật toán, lập chương trình xây dựng ma trận $A[10,10]$ thoả mãn các tính chất:

- + $A[i,j]$ là các số nguyên từ $0..9$ ($1 \leq i, j \leq 10$),
- + Mỗi số từ $0..9$ được gặp 10 lần trong ma trận A ,
- + Mỗi hàng và mỗi cột của A chứa không quá 4 số khác nhau.

Bài 80/2001 - Xếp số 1 trên lưới*(Dành cho học sinh THCS)*

Hãy xếp 16 số 1 lên ma trận 10×10 sao cho nếu xoá đi bất kỳ 5 hàng và 5 cột thì vẫn còn lại ít nhất là một số 1. Nêu thuật toán và lập trình hiển thị ra màn hình kết quả ma trận thoả mãn tính chất trên.

Bài 81/2001 - Dãy nghịch thế*(Dành cho học sinh THPT)*

Cho dãy số $(a_1, a_2, a_3, \dots, a_n)$ là một hoán vị bất kỳ của tập hợp $(1, 2, 3, \dots, n)$. Dãy số $(b_1, b_2, b_3, \dots, b_n)$ gọi là nghịch thế của dãy a nếu b_i là các phân tử đứng trước số i trong dãy a mà lớn hơn i .

Ví dụ:

Dãy a là: 3 2 5 7 1 4 6

Dãy b là: 4 1 0 2 0 1 0

a. Cho dãy a , hãy xây dựng chương trình tìm dãy b .

b. Cho dãy b, xây dựng chương trình tìm dãy a.

Dữ liệu vào trong file NGICH.INP với nội dung:

Dòng đầu tiên là số n ($1 \leq n \leq 10\,000$).

Các dòng tiếp theo là n số của dãy a, mỗi số cách nhau một dấu cách,

Các dòng tiếp theo là n số của dãy b, mỗi số cách nhau bởi một dấu cách.

Dữ liệu ra trong file NGHICH.OUT với nội dung:

n số đầu tiên là kết quả của câu a,

Tiếp đó là một dòng trống và sau đó là n số kết quả của câu b (nếu tìm được dãy a).

Bài 82/2001 - Gặp gỡ

(Dành cho học sinh THPT)

Trên một lưới ô vuông kích thước $M \times N$ (M dòng, N cột) người ta đặt k rôbốt. Rôbốt thứ i được đặt ở ô (x_i, y_i) . Mỗi ô của lưới có thể đặt một vật cản hay không. Tại mỗi bước, mỗi rôbốt chỉ có thể di chuyển theo các hướng lên, xuống, trái, phải - vào các ô kề cạnh không có vật cản. k rôbốt sẽ gặp nhau nếu chúng cùng đứng trong một ô. k rôbốt bắt đầu di chuyển đồng thời và mỗi lượt cả k rôbốt đều phải thực hiện việc di chuyển (nghĩa là không cho phép một rôbốt dừng lại một ô nào đó trong khi rôbốt khác thực hiện bước di chuyển). Bài toán đặt ra là tìm số bước di chuyển ít nhất mà k rôbốt phải thực hiện để có thể gặp nhau. Chú ý rằng, tùy trạng thái của lưới, k rôbốt có thể không khi nào gặp được nhau.

Dữ liệu vào cho trong file văn bản MEET.INP, bao gồm:

+ Dòng đầu tiên chứa 3 số M, N và k ($M, N \leq 50; k \leq 10$)

+ k dòng sau, dòng thứ i gồm 2 số x_i, y_i là vị trí của rôbốt thứ i.

+ M dòng tiếp theo, mỗi dòng ghi N số gồm 0 và 1 mô tả trạng thái dòng tương ứng của lưới, trong đó mỗi số mô tả một ô với quy ước: 0 - không có vật cản, 1 - có vật cản.

Các số trên cùng một dòng của file dữ liệu được ghi cách nhau ít nhất một dấu trắng.

Dữ liệu ra ghi lên file văn bản MEET.OUT: nếu k rôbốt không thể gặp nhau thì ghi một dòng gồm một ký tự #, trái lại ghi k dòng, mỗi dòng là một dãy các ký tự viết liền nhau mô tả các bước đi của rôbốt: U-lên trên, D-xuống dưới, L-sang trái, R-sang phải.

Ví dụ:

MEET.INP

4 6 2

1 1

4 6

0 1 1 0 0 0

0 0 0 0 0 1

0 0 1 0 0 1

0 1 0 1 0 0

MEET.OUT

DRRR

LUUL

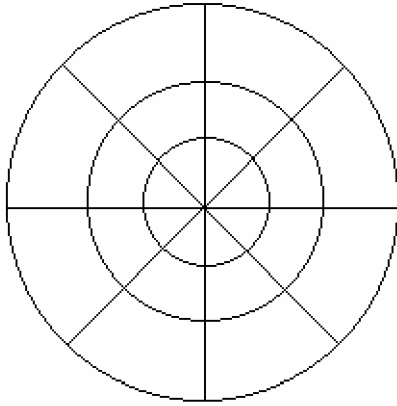
Bài 83/2001 - Các đường tròn đồng tâm*(Dành cho học sinh Tiểu học)*

Ba đường tròn đồng tâm, mỗi hình được chia thành 8 phần (như hình dưới).

Hãy đặt các số trong danh sách dưới đây vào các phần trong các hình tròn sao cho: mỗi đường tròn gồm 8 số trong tám phần có tổng bằng 80, mỗi phần của hình tròn ngoài gồm 3 số (mỗi phần của hình tròn ngoài chứa cả phần của hai hình tròn trong) có tổng bằng 30.

Các số bạn được sử dụng là:

14, 11, 10, 12, 7, 9, 9, 8, 9, 9, 11, 11, 10, 10, 10, 10, 14, 9, 7, 11, 10, 8, 12, 9.

**Bài 84/2001 - Cùng một tích***(Dành cho học sinh THCS và THPT)*

Cho n số x_1, x_2, \dots, x_n chỉ nhận một trong các giá trị $-1, 0, 1$. Và cho một số nguyên P .

Hãy tính số lượng tất cả các cách gán giá trị khác nhau của n số trên sao cho:

$\sum x_i x_j = P$ (với $i=1..n, j=1..n, i \neq j$). Hai cách gán được gọi là khác nhau nếu số lượng các số $x_i = 0$ là khác nhau.

Input: gồm 2 số n, P .

Output: số các cách chọn khác nhau.

Giới hạn: $2 \leq n \leq 10^{10}$; $|P| \leq 10^{10}$.

(Đề ra của ban Lý Quốc Vinh - Tp. Hồ Chí Minh)

Bài 85/2001 - Biến đổi 0 - 1*(Dành cho học sinh THPT)*

Cho 2 lưới ô vuông A và B cùng kích thước $M \times N$, mỗi ô có chỉ nhận các giá trị 0 hoặc 1 (A khác B). Các ô lưới được đánh số từ trên xuống dưới, từ trái qua phải bắt đầu từ 1.

Cho phép thực hiện phép biến đổi sau đây với lưới A :

- Chọn ô (i, j) và đảo giá trị của ô đó và các ô chung cạnh với nó (0 thành 1, 1 thành 0).

Hãy xác định xem bằng cách áp dụng dãy biến đổi trên có thể đưa A về B được hay không? Nếu có hãy chỉ ra cách sử dụng một số ít nhất phép biến đổi.

Dữ liệu nhập vào từ file văn bản BIENDO1.INP:

- 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.

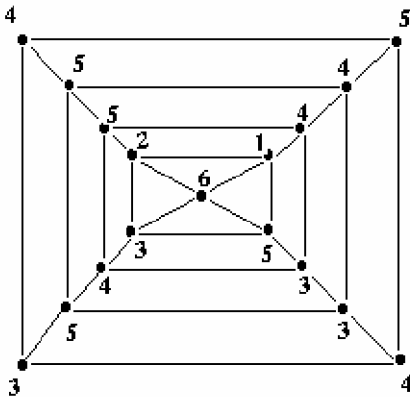
Ví dụ:

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 chong trình kiểm tra xem N có thể biểu diễn thành tổng của 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:

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

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

($M, N \leq 250$)

Ta gọi: $Z = z_1z_2..z_k$ 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 phản nguyên tố*(Dành cho học sinh THCS và THPT)*

Một số n gọi là số phản 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 phản 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 phản 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 phản 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 sapxep(ang(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;
sapxepang(d,n);
sapxepang(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];
xapxepang(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 đề 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)*

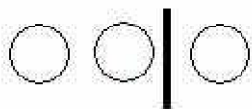
Ta sẽ chỉ ra rằng tồn tại 3 lần cân để chỉ ra được viên bi đặc biệt đó.



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



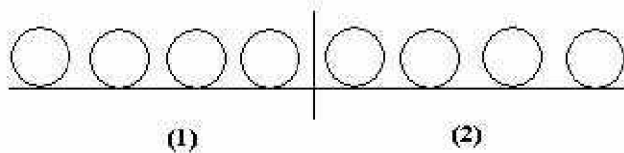
để mô tả một hòn bi bất kỳ



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**.

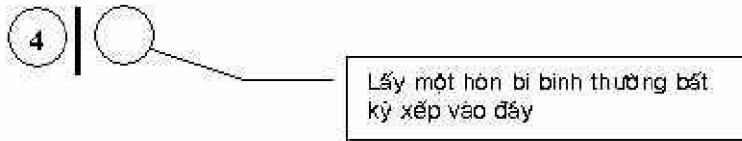
2.2. Trường hợp cân trái nhẹ hơn phải (dấu $<$). Suy ra hoặc 3 là **đđ** nặng, hoặc 1 hoặc 2 là **đđ** nhẹ.

2.3. Trường hợp cân trái nặng hơn phải (dấu $>$). Suy ra hoặc 3 là **đđ** nhẹ, hoặc 1 hoặc 2 là **đđ** nặng.

Dễ 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

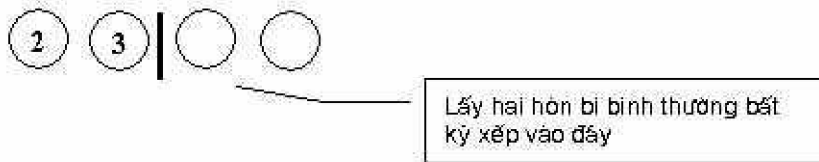
Lần cân thứ ba:



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:

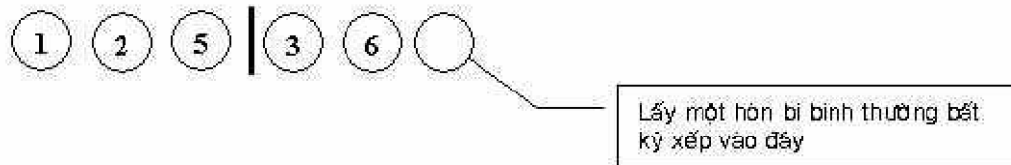


Nếu cân thăng bằng thì ta có 1 là hòn bi đđ nhẹ.

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=' nhẹ hơn.';
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
Writeh(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 Catruoc( 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 );
catruoc := gt;
End;
(*-----*)
Procedure Tinhsmien;
Var
i : integer;
Begin
For i := 1 to n do
Inc( smien ,catruoc( i ) + 1 );
End;
(*-----*)
Procedure GhiKQ;
Begin
WriteLn(So mien mat phang duoc chia la: ' ,smien );
End;
(*-----*)
BEGIN
Clrscr;
Nhap;
Chuanbi;
Tinhsmien;
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;
  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 Xetlaihangcot( 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 duyet(i+1,ii);
Xetlaihangcot(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 := lommin 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;
{-----}
BEGEN
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	...	$N^2 - (N-3)$
			...	$N^2 - (N-4)$
				...

Trong N hàng trên, tổng i số trong hàng thứ i là:

$$\begin{aligned} & i + [N + (i-1)] + [2N + (i-2)] + \dots + [(i-1)N + 1] \\ &= N[1 + 2 + \dots + (i-1)] + [i + (i-1) + (i-2) + \dots + 1] \\ &= Ni(i-1)/2 + i(i+1)/2 \\ &= (Ni^2 - Ni + i^2 + i)/2 \end{aligned}$$

Trong N hàng dưới, tổng $(N-i)$ số trong hàng thứ $N+i$ là

$$\begin{aligned} & (i+1)N + [(i+2)N-1] + [(i+3)N-2] + \dots + [N^2 - (N-i-1)] \\ &= N[(i+1) + (i+2) + \dots + N] - [1 + 2 + \dots + (N-i-1)] \\ &= N(N+i+1)(N-i)/2 - (N-i-1)(N-i)/2 \\ &= (N^2 + Ni + i + 1)(N-i)/2 \\ &= (N^3 + Ni + N - Ni^2 - i^2 - i)/2 \end{aligned}$$

Cắt đôi bảng ở chính giữa theo đường kẻ đậm và ghép lại thành một bảng vuông như sau:

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

Khi đó tổng các số trong hàng thứ i là

$$(Ni^2 - Ni + i^2 + i)/2 + (N^3 + Ni + N - Ni^2 - i^2 - i)/2 = (N^3 + N)/2 = N(N^2 + 1)/2$$

Rõ ràng trong mỗi hàng có N số và tổng các số trong mỗi hàng là như nhau.

Bài 17/2000 - Số nguyên tố tương đương

(Dành cho học sinh THCS)

Có thể viết chương trình như sau:

```
Program Nttđ;
Var M,N,d,i: integer;
{-----}
Function USCLN(m,n: integer): integer;
Var r: integer;
Begin
  While n<>0 do
  begin
    r:=m mod n; m:=n; n:=r;
  end;
  USCLN:=m;
End;
{-----}
BEGIN
  Write('Nhập M,N: '); Readln(M,N);
  d:=USCLN(M,N); i:=2;
  While d<>1 do
  begin
    If d mod i =0 then
    begin
      While d mod i=0 do d:=d div i;
      While M mod i=0 do M:=M div i;
    end;
  end;
```

```

While N mod i=0 do N:=N div i;
end;
Inc(i);
end;
If M*N=1 then Write('M va N nguyen to tuong duong.')
```

Else Write('M va N khong nguyen to tuong duong.');

```

Readln;
END.
```

Bài 18/2000 - Sân bò*(Dành cho học sinh THCS và THPT)*

Ta có thể thấy ngay là con sân phải đi N bước (vì $x_{i+1} = x_i + 1$), và nếu đi lên k bước thì lại đi xuống k bước (vì $y_N = y_0 = 0$). Do đó, $h = N \text{ div } 2$;
Chương trình có thể viết như sau:

```

Program Senbo;
Uses Crt, Graph;
Var f:Text;
    gd, gm, N, W,xo,yo:Integer;
Procedure Nhap;
Begin
    Write('Nhap so N<50:');Readln(N);
    If N>50 Then N:=50;
End;
Procedure Veluoi;
Var i,j,x,y:Integer;
Begin
    W:=(GetMaxX -50) Div N;
    yo:=GetMaxY-100;
    xo:=(GetMaxX-W*N) Div 2-25;
    For i:=0 To N Do
        For j:=0 To N Div 2 Do
            Begin
                x:=i*W+xo;
                y:=yo-j*W;
                Bar(x-1,y-1,x+1,y+1);
            End;
        End;
End;

Procedure Bo
Var i,j,xo,yo,x,y:Integer;
    Sx,Sy,S:String;
Begin
    j:=0;xo:=xo;y:=yo;
    Writeln(f,N:2,N Div 2:3);
    SetColor(2);
    OutTextXY(xo,yo+5,'(0,0)');
    For i:=1 To N Do
        Begin
            If i<=N-i Then Inc(j)
            Else If j>0 Then Dec(j);
```



```

    WriteLn(f,i:2,j:3);
    x:=i*W+xo;y:=yo-j*W;
    Line(xo,yo,x,y);
    Str(i,sx);str(j,sy);
    S:='(+sx+','+sy+')';
    OutTextXY(x,y+5,s);
    Delay(10000);
    xo:=x;yo:=y;
  End;
End;

Begin
  Nhap;
  Assign(F,'P5.Out');
  ReWrite(F);
  Dg:=Detect;
  InitGraph(Gd,Gm,"");
  VeLuoi;
  Bo;
  ReadLn;
  Close(F);
  CloseGraph;
End.

```

Bài 19/2000 - Đa giác

(Dành cho học sinh THPT)

Ta sẽ chứng minh khẳng định sau cho $n \geq 3$:

Các số thực dương $a_1, a_2, a_3, \dots, a_n$ lập thành các cạnh liên tiếp của một đa giác n cạnh khi và chỉ khi với mọi $k=1, 2, \dots, n$ ta có các bất đẳng thức sau:

$$a_1 + a_2 + \dots + a_{n-k} + a_n > a_k \quad (1)$$

(tổng của $n-1$ cạnh bất kỳ phải lớn hơn độ dài cạnh còn lại)

Chứng minh

Chứng minh được tiến hành qui nạp theo n . Với $n = 3$ thì (1) chính là bất đẳng thức tam giác quen thuộc.

Giả sử (1) đúng đến n . Xét (1) cho trường hợp $n+1$.

Trước tiên ta có nhận xét sau: Các số $a_1, a_2, \dots, a_n, a_{n+1}$ lập thành một đa giác $n+1$ cạnh khi và chỉ khi tồn tại một số g sao cho $a_1, a_2, a_3, \dots, a_{n-1}, g$ tạo thành một đa giác n cạnh và g, a_n, a_{n+1} tạo thành một tam giác.

Giả sử $a_1, a_2, a_3, \dots, a_n, a_{n+1}$ lập thành một đa giác $n+1$ cạnh. Khi đó theo nhận xét trên thì tồn tại đa giác n cạnh $a_1, a_2, a_3, \dots, a_{n-1}, g$ và tam giác g, a_n, a_{n+1} . Do đó ta có các bất đẳng thức sau suy từ giả thiết qui nạp và bất đẳng thức tam giác:

$$a_1 + a_2 + a_3 + \dots + a_{n-1} > g \quad (2)$$

$$a_n + a_{n+1} > g > |a_n - a_{n+1}| \quad (3)$$

Do vậy ta có

$$a_1 + a_2 + a_3 + \dots + a_{n-1} > |a_n - a_{n+1}| \quad (4)$$

từ (4) suy ra ngay các khẳng định sau:

$$a_1 + a_2 + a_3 + \dots + a_{n-1} + a_n > a_{n+1} \quad (5)$$

$$a_1 + a_2 + a_3 + \dots + a_{n-1} + a_{n+1} > a_n \quad (6)$$

Mặt khác từ giả thiết qui nạp cho đa giác n cạnh $a_1, a_2, a_3, \dots, a_{n-1}, g$, tương tự như (2) ta có các bất đẳng thức sau với $k < n$:

$$a_1 + a_2 + \dots (\text{thiếu } k) \dots + a_{n-1} + g > ak$$

thay thế vế trái của (3) ta phải có với $k < N: < p >$

$$a_1 + a_2 + \dots (\text{thiếu } k) \dots + a_{n-1} + a_n + a_{n+1} > ak \quad (7)$$

Các bất đẳng thức (5), (6) và (7) chính là (1). Điều kiện cần được chứng minh.

Giả sử ngược lại, hệ bất đẳng thức (1) thỏa mãn, ta có

$$a_1 + a_2 + \dots + a_{n-1} + a_n > a_{n+1} \quad (8)$$

$$a_1 + a_2 + \dots + a_{n-1} + a_{n+1} > a_n \quad (9)$$

và với mọi $k < n$ ta có:

$$a_1 + a_2 + \dots (\text{thiếu } k) \dots + a_{n-1} + a_n + a_{n+1} > ak \quad (10)$$

Từ (8) và (9) ta có ngay:

$$a_1 + a_2 + \dots + a_{n-1} > |a_n - a_{n+1}| \quad (11)$$

Từ (10) suy ra với mọi $k < n$ ta có:

$$a_n + a_{n+1} > ak - a_1 - a_2 - \dots (\text{thiếu } k) \dots - ak \quad (12)$$

Từ các bất đẳng thức (11) và (12) suy ra tồn tại một số dương g thỏa mãn đồng thời các điều kiện sau:

$$a_n + a_{n+1} > g > |a_n - a_{n+1}| \quad (13)$$

$$a_1 + a_2 + \dots + a_{n-1} > g \quad (14)$$

$$g > ak - a_1 - a_2 - \dots (\text{thiếu } k) \dots - ak \quad (15)$$

Các bất đẳng thức (13), (14) và (15) chính là điều kiện để tồn tại đa giác n cạnh $a_1, a_2, a_3, \dots, a_{n-1}, g$ và tam giác g, a_n, a_{n+1} . Điều kiện đủ đã được chứng minh.

Chương trình:

```

Program Dagiac;
Uses Crt;
Const fn = 'P6.INP';
Var i,j,N: integer;
    a: array[1..100] of real;
    s: real;
    Kq: boolean;
{-----}
Procedure Nhap;
Var f: text;
Begin
  Assign(f,fn); Reset(f);
  Readln(f,N);
  For i:=1 to N do Read(f,a[i]);
  Close(f);
End;
{-----}
BEGIN
  Nhap;
  Kq:=true;
  For i:=1 to N do
  begin
    s:=0;
    For j:=1 to N do If j<>i then s:=s+a[j];
    If s<=a[i] then Kq:=false;
  end;
  If Kq then Write('Co.') Else Write('Khong.');
```

END.

Bài 20/2000 - Bạn Lan ở căn hộ số mấy?

(Dành cho học sinh Tiểu học)

Ta coi như các căn hộ được đánh số từ 1 đến 64 (vì ngôi nhà có 8 tầng, mỗi tầng có 8 căn hộ). Ta có thể hỏi như sau:

- Có phải số nhà bạn lớn hơn 32?

Sau khi Lan trả lời, dù "đúng" hay "không" ta cũng biết chính xác căn hộ của Lan ở trong số 32 căn hộ nào. Giả sử câu trả lời là "không" ta cũng biết chính xác căn hộ của Lan ở trong số 32 căn hộ nào. Giả sử câu trả lời là "không", ta hỏi tiếp:

- Có phải số nhà bạn lớn hơn 16?

Sau câu hỏi này ta biết được 16 căn hộ trong đó có căn hộ Lan đang ở.

Tiếp tục hỏi như vậy đối với số đứng giữa trong các số còn lại. Sau mỗi câu trả lời khoảng cách giữa các số giảm đi một nửa. Cứ như vậy, chỉ cần 6 câu hỏi, ta sẽ biết được căn hộ Lan ở.

Bài 21/2000 - Những trang sách bị rơi

(Dành cho học sinh Tiểu học)

Nếu trang bị rơi đầu tiên đánh số 387 thì trang cuối cùng sẽ phải đánh số lớn hơn và phải là số chẵn. Do vậy trang cuối cùng phải là 738.

Như vậy, có $738 - 387 + 1 = 352$ trang sách (176 tờ) bị rơi.

Bài 22/2000 - Đếm đường đi

(Dành cho học sinh THCS)

a) Có tất cả 8 đường đi từ A đến B sao cho mỗi đường đi qua một đỉnh nào đó chỉ đúng một lần. Cụ thể:

A B

A E B

A E F B

A E D F B

A E F C B

A E D C B

A E F D C B

A E D F C B

b). Có tất cả 8 đường đi từ A đến D, sao cho đường đi đó qua mọi cạnh nào đó chỉ đúng một lần, cụ thể:

A B C D

A B E D

A B F D

A E D

A E B F D

A E B C D

A E F D

A E F C D

c). Các đường đi qua tất cả các cạnh của hình, qua mỗi cạnh đúng một lần (điểm bắt đầu và điểm kết thúc trùng nhau):

-

+ Các đường đi qua tất cả các cạnh của hình, qua mỗi cạnh đúng một lần (điểm bắt đầu và điểm kết thúc không trùng nhau):

- Điểm bắt đầu là C và điểm kết thúc là D:

CFBCDFEBAED

CFBCDFEABED

CDFCBFEBAED

....

Tương tự như thế với điểm bắt đầu là D và điểm kết thúc là C ta cũng tìm được các đường thoả mãn tính chất này.

Bài 23/2000 - Quay Rubic

(Dành cho học sinh THPT)

Khai triển mặt rubic và đánh số các mặt như hình vẽ sau:

Khi đó ta có thể xây dựng thủ tục Quay (mặt thứ i) để đổi màu 8 mặt con của mặt này và 12 mặt con kề với mặt này. Trên cơ sở đó giải được 2 bài toán này. Chương trình có thể viết như sau:

Program Rubic;

uses Crt;

Type Arr= array[0..5, 0..7] of byte;

const color: Array [0..5] of char=('F', 'U', 'R', 'B', 'L', 'D');

Var

A1, A2, A0, A: Arr;

X, X1, X2: String;

k: byte;

Procedure Nhap;

Var i, j: byte;

Begin

Clrscr;

Writeln ('Bai toan 1. So sanh hai xau:');

Writeln ('Nhap xau X1:');

Readln (X1);

Writeln (' Nhap xau X2:');

Readln (X2);

Writeln ('Bai toan 2. Tinh so lan xoay:');

Write ('Nhap xau X:');

Readln (X);

For i:= 0 to 5 do

For j:= 0 to 7 do A[i, j]:= i;

A:=A0; A1:=A0; A2:=A0;

End;

Procedure Quay (Var A: Arr; k: byte);

Const Dir : array

[0.. 5, 0.. 3, 0.. 3] of byte = (((1,2,5,4), (6,0,2,4), (5,7,1,3), (4,6,0,2)),
 ((0,4,3,2), (0,0,4,0), (1,1,5,1), (2,2,6,2)),
 ((0,1,3,5), (4,4,4,4), (3,3,3,3), (2,2,2,2)),
 ((1,4,5,2), (2,0,6,4), (1,7,5,3), (0,6,4,2)),
 ((0,5,3,1), (0,0,0,0), (7,7,7,7), (6,6,6,6)),
 ((0,2,3,4), (6,6,2,6), (5,5,1,5), (4,4,0,4)));

var i,j,tg: byte;

Begin

```

tg:=A[k,6];
for i:=3 downto 1 do A[k,0] := A[k,2*i-2];
A[k,0]:=tg;
tg:=A[k,7];
for i:=3 downto 1 do A[k,2*i] := A[k,2*i -2];
A[k,1]:=tg;
for i:=1 to 3 do
begin
tg:=A[dir[k,0,3], Dir[k,i,3];
for j:=3 downto 1 do A[ dir[k,0,j], Dir[k,i,j] ]:= A[ dir[k,0,j-1], Dir[k,i,j-1] ];
A[ [dir[k,0,0], Dir[k,i,0] ]]:=tg;
end;
End;
Function Eq(A,B:Arr):Boolean;
Var i,j,c:byte;
Begin
c:=0;
for i:=1 to 5 do
for j:=1 to 7 do
If A[i,j] <> B[i,j] then inc(c);
If c=0 then Eq:=true else Eq:=false;
End;
Procedure QuayXau(x:string; var A: arr);
Var i,j:byte;
Begin
for i:=1 to length(X) do
begin
for j:= 1 to 5 do
If Color[j] = X[i] then Quay(A,j);
end;
End;
Procedure Bai1;
Begin
QuayXau(X1,A1);
QuayXau(X2,A2);
End;
Procedure Bai2;
Begin
k:=0;
Repeat
QuayXau(X,A);
Inc(k);
Until Eq(A,A0);
End;
Procedure Xuat;
Var i,j:byte;
Begin
writeln;
writeln('Ket qua:');
writeln('Bai toan 1. So sanh 2 xau:');

```

```

If Eq(A1,A2) then writeln('Hai xau X1 va X2 cho cung mot ket qua. ');
writeln('Can ap dung xau X ',k,' lan de Rubic quay ve trang thai ban dau. ');
Readln;
End;
Begin
  Nhap;
  Bai1;
  Bai2;
  Xuat;
END.

```

Bài 24/2000 - Sắp xếp dãy số*(Dành cho học sinh Tiểu học)*

Có thể sắp xếp dãy số đã cho theo cách sau:

Lần thứ	Cách đổi chỗ	Kết quả
0	Dãy ban đầu	3, 1, 7, 9, 5
1	Đổi chỗ 1 và 3	1, 3, 7, 9, 5
2	Đổi chỗ 5 và 7	1, 3, 5, 9, 7
3	Đổi chỗ 7 và 9	1, 3, 5, 7, 9

Bài 25/2000 - Xây dựng số*(Dành cho học sinh THCS)*

Có thể làm như sau:

$$1+35+7 = 43$$

$$17+35 = 52$$

Bài 26/2000 - Tô màu*(Dành cho học sinh THCS)*

Ký hiệu màu Xanh là x, màu Đỏ là d, màu Vàng là v. Ta có 12 cách tô màu được liệt kê như sau:

x	d	v	x
d	v	x	d
v	x	d	v
x	d	v	x

xx	dd	vv	xx
vv	xx	dd	vv
dd	vv	xx	dd
xx	dd	vv	xx

xx	dd	vv	xx
dd	xx	vv	dd
vv	dd	xx	vv
xx	vv	dd	xx

xx	dd	vv	xx
vv	dd	xx	vv
dd	xx	vv	dd
xx	vv	dd	xx

dd	vv	xx	dd
xx	dd	vv	xx
vv	xx	dd	vv
dd	vv	xx	dd

dd	vv	xx	dd
vv	xx	dd	vv
xx	dd	vv	xx
dd	vv	xx	dd

dd	xx	vv	dd
xx	vv	dd	xx
vv	dd	xx	vv
dd	xx	vv	dd

vv	xx	dd	vv
xx	dd	vv	xx
dd	vv	xx	dd
vv	xx	dd	vv

vv	xx	dd	vv
dd	vv	xx	dd
xx	dd	vv	xx
vv	xx	dd	vv

vv	dd	xx	vv
dd	xx	vv	dd
xx	vv	dd	xx
vv	dd	xx	vv

vv	dd	xx	vv
xx	vv	dd	xx
dd	xx	vv	dd
vv	dd	xx	vv

dd	xx	vv	dd
vv	dd	xx	vv
xx	vv	dd	xx
dd	xx	vv	dd

Bài 27/2000 - Bàn cờ*(Dành cho học sinh THPT)*

Chương trình của bạn Nguyễn Tiến Dũng lớp 8A2 trường PTTH chuyên Bến Tre, tỉnh Bến Tre.

Program Ban_co;

Uses Crt;

```

  Var   a: array [1..8, 1..8] of 0..1;
        b, c, d, p: array [0..8,0..8] of integer;
        max:integer;

```

Procedure Input;

```

  Var   f: text;  i, j: integer;
        st: string[8];

```

Begin

```

  Assign (f, 'banco2.txt');
  Reset (f);
  For i:=1 to 8 do
  begin
    Readln(f,st);
    For j:=1 to 8 do If st[j]= 0 then  a[i,j]:=0 else a[i,j]:=1;
  end;
  Close(f);

```

End;

Procedure Init;

Begin

```

  Input;
  Fillchar(b,sizeof(b),0);
  c:=b; d:=b; p:=b;

```

End;

Function Get_max(x, y, z, t: integer): integer;

```

  Var   k: integer;
  Begin
    k:=x;
    If k < y then k:=y;
    If k < z then k:=z;
    If k < t then k:=t;
    Get_max:=k;
  End;

```

Procedure Find_max;

```

  Var
    i, j, k: integer;
  Begin
    max:=0;
    For i:=1 to 8 do
    For j:=1 to 8 do
    If a[i, j]= 1 then
    begin
      b[i, j]:=b[i-1,j]+1;
      c[i, j]:=c[i,j-1]+1;
      d[i,j]:=d[i-1,j-1]+1;
    end;
  end;

```

```

    p[i,j]:=p[i-1,j+1]+1;
    k:=get_max(b[i,j], c[i,j], d[i,j], p[i,j]);
    If max < k then max:=k;
  end;
  Writeln (max);
  Readln;
End;
BEGIN
  Clrscr;
  Init;
  Find_max;
END.

```

Bài 28/2000 - Đổi tiền*(Dành cho học sinh Tiểu học)*

Có 10 cách đổi từ 10 ngàn đồng bằng các đồng tiền 1, 2 và 5 ngàn đồng.

Số tờ 1 ngàn	Số tờ 2 ngàn	Số tờ 5 ngàn
0	0	2
1	2	1
3	1	1
5	0	1
0	5	0
2	4	0
4	3	0
6	2	0
8	1	0
10	0	0

Bài 29/2000 - Chọn bạn*(Dành cho học sinh THCS)*

Gọi một bạn học sinh nào đó trong 6 bạn là A. Chia 5 bạn còn lại thành 2 nhóm: Nhóm 1 gồm những bạn quen A, nhóm 2 gồm những bạn không quen A (đĩ nhiên A không nằm trong 2 nhóm đó). Vì tổng số các bạn trong 2 nhóm bằng 5 nên chắc chắn có 1 nhóm có từ 3 bạn trở lên. Có thể xảy ra hai khả năng:

Khả năng 1. Nhóm 1 có từ 3 bạn trở lên: Khi đó nếu các bạn trong nhóm đó không ai quen ai thì bản thân nhóm đó chứa 3 bạn không quen nhau cần tìm. Ngược lại nếu có 2 bạn trong nhóm đó quen nhau thì hai bạn đó cùng với A chính là 3 bạn quen nhau cần tìm.

Khả năng 2. Nhóm 2 có từ 3 bạn trở lên: Khi đó nếu các bạn trong nhóm 2 đã quen nhau đôi một thì nhóm đó chứa 3 bạn quen nhau đôi một cần tìm; ngược lại nếu có 2 bạn trong nhóm không quen nhau thì 2 bạn đó cùng với A chính là 3 bạn không quen nhau cần tìm.

Bài 30/2000 - Phân tử yên ngựa*(Dành cho học sinh THCS)*

```

const
  Inp = 'Bai30.INP';
  Out = 'Bai30.OUT';
  MaxLongInt = 2147483647;

```



```

var
  Min, Max: array[1..5000] of LongInt;
  m, n: Integer;
procedure ReadInput;
var
  i, j, k: Integer;
  hf: Text;
begin
  Assign(hf, Inp);
  Reset(hf);
  Readln(hf, m, n);
  for i := 1 to m do Min[i] := MaxLongInt;
  for j := 1 to n do Max[j] := -MaxLongInt;
  for i := 1 to m do
  begin
    for j := 1 to n do
    begin
      Read(hf, k);
      if Min[i] > k then Min[i] := k;
      if Max[j] < k then Max[j] := k;
    end;
    Readln(hf);
  end;
  Close(hf);
end;
procedure WriteOutput;
var
  i, j: Integer;
  Result: Boolean;
  hf: Text;
begin
  Result := False;
  Assign(hf, Out);
  Rewrite(hf);
  Writeln(hf, 'Các phần tử yên ngựa là: ');
  for i := 1 to m do
    for j := 1 to n do
      if Min[i] = Max[j] then
        begin
          Result := True;
          Write(hf, '( ' , i , ' , ' , j , ' );
        end;
  if not Result then
  begin
    Rewrite(hf);
    Write(hf, 'Không có phần tử yên ngựa');
  end;
  Close(hf);
end;
begin
  ReadInput;
  WriteOutput;
end.
3 3
15 3 9
55 4 6
76 1 2

```

Bài 32/2000 - Bài toán 8 hậu
(Dành cho học sinh Tiểu học)

Có rất nhiều cách xếp. Sau đây là một vài cách để các bạn tham khảo:

0 1 0 0 0 0 0 0	0 0 0 1 0 0 0 0
0 0 0 0 0 1 0 0	0 0 0 0 0 0 0 1
0 0 1 0 0 0 0 0	1 0 0 0 0 0 0 0
0 0 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 1 0 0 0
0 0 0 0 0 0 1 0	0 0 0 1 0 0 0 0
1 0 0 0 0 0 0 0	0 0 0 0 0 0 0 1
0 0 0 0 0 1 0 0	0 0 1 0 0 0 0 0
0 1 0 0 0 0 0 0	0 0 0 0 1 0 0 0
0 0 0 0 0 0 1 0	1 0 0 0 0 0 0 0
0 0 1 0 0 0 0 0	0 0 0 0 0 0 0 1
0 0 0 0 0 1 0 0	0 0 0 1 0 0 0 0
0 1 0 0 0 0 0 0	0 0 0 0 0 1 0 0
1 0 0 0 0 0 0 0	0 0 0 0 0 0 1 0
0 0 0 1 0 0 0 0	0 0 0 0 0 0 0 1
0 0 1 0 0 0 0 0	0 0 0 0 1 0 0 0

Để tìm hết nghiệm của bài này chúng ta phải sử dụng thuật toán Đệ quy - Quay lui. Sau đây là chương trình, chạy ra 92 nghiệm và ghi các kết quả đó ra file HAU.OUT.

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

```
const fo = 'hau.out';
      n = 8;
```

```
var A : array[1..n,1..n] of byte;
    c : array[1..n] of byte;
    dc1 : array[2..2*n] of byte;
    dc2 : array[1-n..n-1] of byte;
    sn : integer;
    f : text;
```

```
procedure ghino;
var i,j : byte;
begin
  inc(sn);
  writeln(f,'Nghiem thu ',sn,' la :');
  for i := 1 to n do
  begin
    for j := 1 to n do
      write(f,A[i,j],#32);
    writeln(f);
  end;
  writeln(f);
end;
```

```
procedure vet(i : byte);
var j : byte;
begin

  if i = n+1 then
    begin
```

```

    ghino;
    exit;
end;

for j := 1 to n do
  if (c[j]=0)and(dc1[i+j]=0) and (dc2[i-j]=0) then
    begin
      A[i,j] := 1; c[j] := 1; dc1[i+j] := 1; dc2[i-j] := 1;
      vet(i+1);
      A[i,j] := 0; c[j] := 0; dc1[i+j] := 0; dc2[i-j] := 0;
    end;
end;

BEGIN
  assign(f,fo);
  rewrite(f);
  vet(1);
  close(f);
END.

```

Bài 33/2000 - Mã hoá văn bản

(Dành cho học sinh THCS)

a. Mã hoá:

PEACE thành UJFHJ

HEAL THE WORLD thành MJFQ YMJ BTWQI

I LOVE SPRING thành N QTAJ XUWNSL.

b. Qui tắc giải mã các dòng chữ đã được mã hoá theo quy tắc trên: (lấy ví dụ ký tự X):

-Tìm số thứ tự tương ứng của kí tự, ta được 23.

-Tăng giá trị số này lên 21 (thực ra là giảm giá trị số này đi 5 rồi cộng với 26), ta được 44.

-Tìm số dư trong phép chia số này cho 26 ta được 18.

-Tra ngược bảng chữ cái ta thu được S.

Giải mã:

N FRF XYZIJSY thành I AM A STUDENT

NSKTVRFYNHX thành INFOQMATIC.S

MFSTN SFYNTSFQ ZSNBJVXNYD thành HANOI NATIONAL UNIWEQSITY.

Sau đây là chương trình mô tả thuật toán giải quyết bài 33/2000, gồm 2 thủ tục chính là: *mahoatu* (chuyển xâu thành xâu mã hoá) và *giaimatu* (chuyển xâu thành xâu giải mã).

Các bạn có thể xem kết quả sau khi chạy chương trình bằng cách ấn Alt + F5.

```

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

function mahoatu(x : char) : char;
var vtri : byte;
begin
  if upcase(x) in ['A'..'Z'] then
    begin
      vtri := ord(upcase(x))-ord('A');
      vtri := vtri+5;
      mahoatu := char( vtri mod 26+ord('A'));
    end
  else mahoatu := x;
end;

```

```

function giaima(x : char) : char;
var vtri : byte;
begin
  if upcase(x) in ['A'..'Z'] then
    begin
      vtri := ord(upcase(x))-ord('A');
      vtri := vtri-5+26;
      giaima := char( vtri mod 26 + ord('A'));
    end
  else giaima := x;
end;

procedure mahoatu(s : string);
var i : byte;
begin
  write(s, ' -> ');
  for i := 1 to length(s) do write(mahoa(s[i]));
  writeln;
end;

procedure giaimatu(s : string);
var i : byte;
begin
  write(s, ' <- ');
  for i := 1 to length(s) do write(giaima(s[i]));
  writeln;
end;

BEGIN
  clrscr;
  mahoatu('PEACE');
  mahoatu('HEAL THE WORLD');
  mahoatu('I LOVE SPRING');
  giaimatu('N FR F XYZIJSY');
  giaimatu('NSKTVRFYNHX');
  giaimatu('MFSTN SFYNTSFQ ZSNBJVXNYD');
END.

```

Bài 34/2000 - Mã hoá và giải mã
(Dành cho học sinh THCS)

```

Program bai34;
Uses crt;
Const
Ord : array['A'..'Z'] of byte =(0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17,
18, 19, 20, 21, 22, 23, 24, 25);
chr : array[0..25] of char = ('A', 'B', 'C', 'D', 'E', 'F', 'G', 'H', 'I', 'J', 'K', 'L', 'M', 'N', 'O', 'P',
'Q', 'R', 'S', 'T', 'U', 'V', 'W', 'X', 'Y', 'Z');
Var s:string;
    i, j:integer; ch:char;
Begin
  S:="";
  Writeln('Nhap xau ki tu:');
  Repeat
    ch:= ReadKey;
    If (ch in ['a'..'z', 'A'..'Z']) then
      Begin

```

```

    ch := Upcase(ch); Write(ch);
    S := S + ch;
  End;
Until ch = #13; Writeln;
For i := 1 to length(s) do
  If S[i] <> ' ' then S[i] := chr[(ord{s[i]} + 5) mod 26];
  Writeln('Xau ki tu tren duoc ma hoa la:'); write(s); Readln;
  S := ' ';
  Writeln('Nhap xau ki tu can giai ma:');
  Repeat
    ch := Readkey;
    If (ch in ['a'..'z', 'A'..'Z']) then
      Begin
        ch := Upcase(ch); Write(ch);
        s := s + ch;
      End;
  Until ch = #13; Writeln;
  for i := 1 to length{S} do
  If S[i] <> ' ' then S[i] := chr[(Ord[S[i]] + 21) mod 26];
  writeln('Xau ki tu tren duoc giai ma la:'); write(s);
  Readln;
End.

```

Các bạn cũng có thể sử dụng lại 2 thủ tục *mahoatu* và *giaimatu* ở bài 33/2000 để giải bài này. Việc thiết kế giao diện khi nhập xâu từ bàn phím xin dành cho các bạn.

Bài 35/2000 - Các phân số được sắp xếp

(Dành cho học sinh THPT)

```

Program bai35;
Uses crt;
Type Phanso = (tu, mau);
  Var F: array[1..4000, phanso] of integer;
      N, dem : Integer;
Procedure nhap;
Begin
  Write('Nhap so N:'); Readln(N);
  F[1,tu] := 0; F[1,mau] := 1; dem := 2;
  F[dem, tu] := 1; F[dem,mau] := 1;
End;
Procedure Chen(t,m,i:Integer);
  Var j:integer;
Begin
  Inc(dem);
  For j := dem downto i + 1 do
  begin
    F[j,tu] := F[j-1,tu];
    F[j,mau] := F[j-1,mau];
  end;
  F[i,tu] := t; F[i,mau] := m;
End;
Program xuli;

```

```

Var t,m,i:integer;
Begin
  for m:=2 to N do
    for t:=1 to m-1 do
      begin
        i:=1;
        While (F[i,tu]*m < F[i,mau]*t) do inc(i);
        If (F[i,tu]*m > F[i,mau]*t) then chen(t,m,i);
      end;
    End;
  Procedure xuat;
    var i:integer;
  Begin
    for i:=2 to dem do
      begin
        If WhereX > 75 then writeln;
        If WhereY > 24 then
          begin
            Write('Nhan Enter de tiep tuc');
            Readln;
          end;
        write('Tat ca co', dem, 'phan so. ');
        Readln;
      End;
    BEGIN
      nhap;
      xuli;
      Xuat;
    END.

```

Bài 36/2000 - Anh chàng hà tiện*(Dành cho học sinh Tiểu học)*

Liệt kê số tiền phải trả cho từng chiếc cúc rồi cộng lại, ta được bảng sau:

Thứ tự	Số tiền	Cộng dồn
1	1	1
2	2	3
3	4	7
4	8	15
5	16	31
6	32	63
7	64	127
8	128	255
9	256	511
10	512	1023
11	1024	2047
12	2048	4095
13	4096	8191
14	8192	16383

15	16384	32767
16	32768	65535
17	65536	131071
18	131072	262143 (= $2^{18} - 1$)

Như vậy anh ta phải trả 262143 đồng và anh ta rõ ràng là bị "hớ" nặng do phải trả gấp hơn 20 lần so với cách thứ nhất.

Bài 37/2000 - Số siêu nguyên tố

(Dành cho học sinh THCS)

Program Bai37;

{SuperPrime};

var a,b: array [1..100] of longint;

 N,i,k,ka,kb,cs: byte;

Function Prime(N: longint): boolean;

Var i: longint;

Begin

 If (N=0) or (N=1) then

 Prime:=false

 Else

 Begin

 i:=2;

 While (N mod i <> 0) and (i <= Sqrt(N)) do Inc(i);

 If i > Sqrt(N) then

 Prime:=true Else Prime:=false;

 End;

End;

BEGIN

 Write ('Nhap N: ');

 Readln (N);

 ka:=1; a[ka]:=0;

 For i:=1 to N do

 Begin

 Kb:=0;

 For k:=1 to ka do

 For cs:=0 to 9 do

 If Prime(a[k]*10+cs) then

 Begin

 Inc(kb);

 b[kb]:=a[k]*10+cs;

 end;

 ka:=kb;

 For k:=1 to ka do

 a[k]:=b[k]; end;

 For k:=1 to ka do

 Write(a[k]:10);

 Writeln;

 Writeln('Co tat ca',ka,'so sieu nguyen to co',N,'chu so.');

 Readln;

END.

Bài 38/2000 - Tam giác số

```

Uses Crt;
Const inp='INPUT.TXT';
Var N,Smax: integer;
    a: array [1..100,1..100] of integer;
{-----}
Procedure Nhap;
Var f: text;
    i,j: integer;
Begin
Assign(f,inp);
Reset(f);
Readln(f,n);
For i:=1 to N do
begin
    For j:=1 to i do Read(f,a[i,j]);
    Readln(f);
end;
Close(f);
End;
{-----}
Procedure Thu(S,i,j: integer);
Var k,S_new: integer;
Begin
S_new:=S+a[i,j];
If i=N then
begin
    If S_new>Smax then Smax:=S_new;
end
else
    For k:=j to j+1 do Thu(S_new, i+1, k);
End;
{-----}
BEGIN
Nhap;
Smax:=0;
Thu(0,1,1);
Write('Smax = ',Smax);
Readln;
END.

```

Dưới đây các bạn có thể tham khảo lời giải của bạn Phạm Đức Thanh dùng phương pháp quy hoạch động trên mảng hai chiều:

```

Program bai38;
Uses crt;
Type mang = array[1..100,1..100] of integer;
Var
    f:text;
    i,j,n:integer;

```



```

    a,b:mang;
Procedure Input;
Begin
  clrscr;
  Assign(f,'input.txt');
  reset(f);
  readln(f,n);
  for j:=1 to n do
    begin
      for i:=2 to j+1 do
        read(f,a[j,i]);
      end;
    end;
  close(f);
end;
{-----}
Function Max(m,n:integer):integer;
Begin
  if n>m then Max:=n
  else Max:=m;
end;
{-----}
Procedure MakeArrayOfQHD;
Begin
  b[1,2]:=a[1,2];
  for j:=1 to n do b[j,1]:=-maxint;
  for i:=3 to n do b[1,i]:=-maxint;
  for j:=2 to n do
    begin
      for i:=2 to j+1 do
        b[j,i]:=a[j,i]+max(b[j-1,i],b[j-1,i-1]);
      end;
    end;
end;
{-----}
Procedure FindMax;
var max:integer;
Begin
  max:=b[n,1];
  for i:=2 to n do
    if b[n,i]>max then max:=b[n,i];
  writeln('Smax:=',max);
  readln;
end;
{-----}
BEGIN
  Input;
  makearrayofQHD;
  FindMax;
END.

```

Nhận xét: Lời giải dùng thuật toán quy hoạch động của Phạm Đức Thanh tốt hơn rất nhiều so với thuật toán đệ quy quay lui.

Bài 39/2000 - Ô chữ

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

{M 16384,0,655360}

```
uses crt;

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

var A : array[1..5,1..5] of char;
    new,blank : record x,y : integer end;

procedure no_no_and_no;
var f : text;
begin
  assign(f,fo);
  rewrite(f);
  write(f,'This puzzle has no final configuration. ');
  close(f);
  halt;
end;

procedure yes_yes_and_yes;
var f : text;
    i,j : byte;
begin
  assign(f,fo);
  rewrite(f);
  for i := 1 to 5 do
    begin
      for j := 1 to 5 do
        write(f,a[i,j]);
        writeln(f);
      end;
    end;
  close(f);
end;

procedure swap(px,py : integer);
var coc : char;
begin
  new.x := blank.x + px;
  new.y := blank.y + py;
  if (new.x > 5) or (new.y > 5) or (new.x < 1) or (new.y < 1) then
    no_no_and_no;

  coc := A[new.x,new.y];
  A[new.x,new.y] := A[blank.x,blank.y];
  A[blank.x,blank.y] := coc;
  blank := new;
end;

procedure chuyen(ch : char);
begin
  case ch of
    'A' : swap(-1,0);
    'B' : swap( 1,0);
    'R' : swap( 0, 1);
```

```

'L' : swap( 0,-1);
end;
end;

procedure docf;
var   f   :   text;
      i,j :   byte;
      s   :   string[5];
      ch  :   char;
begin
  assign(f,fi);
  reset(f);
  for i :=1 to 5 do
    begin
      readln(f,s);
      if length(s) = 4 then s := s+ #32;
      for j := 1 to 5 do
        begin
          A[i,j] := s[j];
          if A[i,j] = #32 then
            begin
              blank.x := i;
              blank.y := j;
            end;
          end;
        end;
      end;
    while not seekeof(f) do
      begin
        read(f,ch);
        if ch = '0' then exit;
        chuyen(ch);
      end;
    close(f);
  end;

BEGIN
  clrscr;
  docf;
  yes_yes_and_yes;
END.

```

Bài 40/2000 - Máy định vị Radio

```

Uses crt;
Const nmax = 30;
      Output = 'P27.out';
      Input = 'P27.inp';
Type
  str20 = string[20];
Var
  Toado : Array[1..nmax,1..2] of real;
  TenDen,TenDen1,TenDen2 : Array[1..nmax] of str20;
  n,j,i,k:integer;
  Td1,Td2:array[1..2] of integer;
  goc,g1,g2,v,l:array[1..2] of real;
  t1,t2:array[1..2] of integer;
  xd,yd,x,y, x1,x2,y1,y2:array[1..2] of real;
  f:text;

```

```

Function tg(x: real): real;
Begin
  if cos(x) <> 0 then tg:=sin(x)/cos(x);
End;
Procedure DocDen(var s:str20);
Var d:char;
Begin
  repeat
    read(f,d);
  Until (d <> ' ');
  s:="";
  While (d <> ' ') do
    begin
      s:=s+d;
      Read(f,d);
    End;
  End;
End;
Function Xdtoado(s:str20):Integer;
Var i:integer;
Begin
  i:=1;
  While (i<=n) and (s <> tenden[i]) do inc(i);
  Xdtoado:=i;
End;

```

```

Procedure InputDen;
Var i:integer;
Begin
  Assign(f,input);
  Reset(f);
  Readln(f,n);
  For i:=1 to n do
    Begin
      DocDen(TenDen[i]);
      Readln(f,Toado[i,1],Toado[i,2]);
    End;
  End;
End;

```

```

Procedure Inputkichban;
Begin
  Readln(f,k);
  For i:=1 to k do
    Begin
      Readln(f, goc[i],v[i]);
      Read(f,t1[i]);
      Docden(tenden1[i]);
      Td1[i]:=Xdtoado(tenden1[i]);
      Readln(f,g1[i]);
      Read(f,t2[i]);
      Docden(tenden2[i]);
      Td2[i]:=Xdtoado(tenden2[i]);
    End;
  End;

```

```

    Readln(f,g2[i]);
  End;
Close(f);
End;
Procedure Doi;
Begin
  For j:=1 to k do
    Begin
      goc[j]:=goc[j]*pi/180;
      g1[j]:=g1[j]*pi/180;
      g2[j]:=g2[j]*pi/180;
      l[j]:=(t2[j]-t1[j])*v[j];
    End;
  End;
Procedure TinhToan;
Begin
  Assign(f,output);Rewrite(f);
  For j:=1 to k do
    Begin
      x1[j]:=Toado[td1[j],1];
      y1[j]:=Toado[td1[j],2];
      x2[j]:=Toado[td2[j],1];
      y2[j]:=Toado[td2[j],2];
      xd[j]:=x1[j]+l[j]*sin(goc[j]);
      yd[j]:=y1[j]+l[j]*cos(goc[j]);
      If (cos(goc[j]+g2[j])=0) or (cos(goc[j]+g1[j])=0) then
        Writeln(f,'Scenario ',j,': Position cannot be determined')
      else
        Begin
          y[j]:=(xd[j] - x2[j] - yd[j]*tg(goc[j] + g1[j]) + y2[j]*tg(goc[j] + g2[j]))/(tg(goc[j]
+ g2[j]) - tg(goc[j] + g1[j]));
          x[j]:= x2[j] - (y2[j] - y[j])*tg(goc[j] + g2[j]);
          Writeln(f,'Scenario ',j,': Positino is (' , x[j]:6:2, y[j]:6:2,')' );
        End;
    End;
  End;
BEGIN
  InputDen;
  Inputkichban;
  Doi;
  TinhToan;
  Close(f);
END.

```

Bài 41/2000 - Cờ Othello

Program bai41; {Co Othello}

Uses Crt ;

Const Inp = 'othello.Inp' ;

Out = 'othello.out' ;

nmax = 50;

```

huongi:array[1..8] of integer = (-1,-1,-1,0,0,1,1,1);
huongj:array[1..8] of integer = (-1,0,1,-1,1,-1,0,1);
Type
  Mang1 = Array [1..nmax] of string[3] ;
  Mang2 = Array [1..8,1..8] of char ;
Var  f: text;
     a: mang2; l:mang1;
     c: char; n, k, code:integer;
     di:array[1..8,1..8] of boolean;
     x0,y0:array[1..nmax] of integer;
{=====}
Procedure nhap;
Var  i,j : Byte ;
Begin
  Assign(f,inp) ;
  Reset(f) ;
  for i:=1 to 8 do
    begin
      for j:=1 to 8 do Read(f,a[i,j]) ;
      Readln(f) ;
    end;
  Readln(f,c) ;
i:=0;
  while not eof(f) do
    begin
      inc(i);
      Readln(f,l[i]);
    end;
  n:=i;
End ;
{=====}
Procedure kiemtra(i,j:integer);
Var m:integer;
Begin
  Case c of
    'B': If a[i,j] = 'B' then
      Begin
        m:= 1;
        repeat
          if (a[i+huongi[m],j+huongj[m]] = 'W')
            and(i+huongi[m]>0)and(j+huongj[m]>0)
            and(i+2*huongi[m]>0)and(j+2*huongj[m]>0)
            and(i+huongi[m]<9)and(j+huongj[m]<9)
            and(i+2*huongi[m]<9)and(j+2*huongj[m]<9)
            and(A [i+2*huongi[m],j+2*huongj[m]] = '-')
          then
            di [i+2*huongi[m],j+2*huongj[m]] := True;
          m:=m+1;
        until m>8;
      End;

```

```

'W': If (a[i,j] = 'W') then
  Begin
    m:= 1;
    repeat
      if (a [i+huongi[m],j+huongj[m]] = 'B')
        and(i+huongi[m]>0)and(j+huongj[m]>0)
        and(i+2*huongi[m]>0)and(j+2*huongj[m]>0)
        and(i+huongi[m]<9)and(j+huongj[m]<9)
        and(i+2*huongi[m]<9)and(j+2*huongj[m]<9)
        and(a[i+2*huongi[m],j+2*huongj[m]] = '-')
          then
            di[i+2*huongi[m],j+2*huongj[m]] := True;
    m:=m+1;
  until m>8;
  end;
End;{of Case}
End;
{=====}
Procedure lietke;
Var
  i,j,m: Integer;
  t: Boolean;
Begin
  t:= false;
  for i:=1 to 8 do
    for j:= 1 to 8 do
      di[i,j]:=false;
  for i:=1 to 8 do
    for j:= 1 to 8 do kiemtra(i,j);
      for i:= 1 to 8 do
        for j:= 1 to 8 do
          If di[i,j] then
            Begin
              t:= True;
              Write (f,'(,i,',',j,')');
            End;
  If t=false then Write (f, 'No legal move. ');
  Writeln(f);
End;
{=====}
Procedure latco(x0,y0:integer);
Var m:integer;
Begin
  Case c of
    'B': if a[x0,y0] = '-'then
      begin
        m:= 1;
        repeat
          If (a[x0-2*huongi[m],y0-2*huongj[m]] = 'B')
            and(a[x0-huongi[m],y0-huongj[m]] = 'W')

```

```

    then
      begin
        a[x0,y0]:='B';
        a[x0-huongi[m],y0-huongj[m]] := 'B';
      end;

    m:=m+1;
    until m>8;
  end;
'W': if a[x0,y0] = '-' then
  begin
    m:= 1;
    repeat
      If (a[x0-2*huongi[m],y0-2*huongj[m]] = 'W')
      and(a[x0-huongi[m],y0-huongj[m]] = 'B')
      then
        begin
          a[x0,y0]:='W';
          a[x0-huongi[m],y0-huongj[m]] := 'W';
        end;
      m:=m+1;
      until m>8;
    end;
  end;
End;
{=====}
Procedure Thuchien(k:integer);
Var
  i,j,xx,yy,xx1,yy1: Integer;
  code,m: Integer;
Begin
  for i:= 1 to 8 do
    for j:= 1 to 8 do
      begin
        if a[i,j]='W'then yy1 :=yy1+1;
        if a[i,j]='B'then xx1 :=xx1+1;
      end;
  xx:= 0; yy:= 0;
  for i:= 1 to 8 do
    for j:= 1 to 8 do kiemtra(i,j);
    If not di[x0[k],y0[k]] then
      begin
        Case c Of
          'W':c:= 'B';
          'B':c:= 'W';
        End;
        for i:= 1 to 8 do
          for j:= 1 to 8 do kiemtra(i,j);
          If not di[x0[k],y0[k]] then

```



```

        Case c Of
            'W':c:= 'W';
            'B':c:= 'B';
        End;
    end;
latco(x0[k],y0[k]);
for i:= 1 to 8 do
    for j:= 1 to 8 do
        begin
            if a[i,j]='W'then yy:=yy+1;
            if a[i,j]='B'then xx:=xx+1;
        end;
WriteLn (f,'Black - ',xx, ' White - ',yy);
if (xx<>xx1)and(yy<>yy1) then
    Case c Of
        'W':c:= 'B';
        'B':c:= 'W';
    End;
End;
{=====}
Procedure ketthuc;
Var
    i,j:Integer;
Begin
    for i:= 1 to 8 do
        begin
            for j:= 1 to 8 do Write (f,a [i,j]);
            Writeln(f);
        end;
End;
{=====}
Begin
    clrscr;
    nhap;
    Assign(f,out);
    Rewrite(f);
    for k:=1 to n do
        Case l[k][1] of
            'L': Lietke;
            'M':begin
                Val(l[k][2],x0[k],code);
                Val(l[k][3],y0[k],code);
                Thuchien(k);
            end;
            'Q': ketthuc;
        End;
    Close(f);
End.

```

Bài 42/2000 - Một chút về tư duy số học

(Dành cho học sinh Tiểu học)

Giả sử A là số phải tìm, khi đó A phải có dạng:

$$A = 2k_1 + 1 = 3k_2 + 2 = \dots = 10k_9 + 9 \quad (k_1, k_2, \dots, k_9 - \text{là các số tự nhiên}).$$

$$\text{Khi đó } A + 1 = 2(k_1 + 1) = 3(k_2 + 1) = \dots = 10(k_9 + 1).$$

Vậy A+1 phải là BSCNN (bội số chung nhỏ nhất) của (2, 3, ..., 10) = 2520.

Do đó số phải tìm là A = 2519.

Bài 43/2000 - Kim giờ và kim phút gặp nhau bao nhiêu lần trong ngày

(Dành cho học sinh Tiểu học)

Ta có các nhận xét sau:

+ Kim phút chạy nhanh gấp 12 lần kim giờ. Giả sử gọi v là vận tốc chạy của kim giờ, khi đó vận tốc của kim phút là 12v.

+ Mỗi giờ kim phút chạy một vòng và gặp kim giờ một lần. Như vậy trong 24 giờ, kim giờ và kim phút sẽ gặp nhau 24 lần. Tất nhiên những lần gặp nhau trong 12 giờ đầu cũng như các lần gặp nhau trong 12 giờ sau. Và các lần gặp nhau lúc 0 giờ, 12 giờ và 24 giờ là trùng nhau và gặp nhau vào chính xác các giờ đó.

Do đó, ở đây ta chỉ xét trong chu kì một vòng của kim giờ (tức là từ 0 giờ đến 12 giờ).

Giả sử kim giờ và kim phút gặp nhau lúc h giờ (h = 0, 1, 2, 3, ..., 10, 11) và s phút. Và giả sử xét quãng đường được đo theo đơn vị là phút. Do thời gian chạy là như nhau nên ta có:

$$\frac{60h + s}{12h} = \frac{s}{h}$$

$$\Rightarrow 60h = 11s \Rightarrow s = \frac{60h}{11}.$$

Thay lần lượt h = 0, 1, 2, 3, ..., 10, 11 vào ta sẽ tính được s.

Ví dụ:

Với h = 0, $\Rightarrow s = 0 \Rightarrow$ Kim giờ và kim phút gặp nhau đúng vào lúc 0 giờ.

$$h = 1, \Rightarrow s = \frac{60}{11} = 5\frac{5}{11} \Rightarrow \text{Kim giờ và kim phút gặp nhau lúc 1 giờ } 5\frac{5}{11} \text{ phút.}$$

$$h = 2, \Rightarrow s = 10\frac{10}{11} \Rightarrow \text{Kim giờ và kim phút gặp nhau lúc 2 giờ } 10\frac{10}{11} \text{ phút.}$$

....

h = 11, $\Rightarrow s = 60; 11 \text{ giờ } 60 \text{ phút} = 12 \text{ giờ} \Rightarrow$ Kim giờ và kim phút gặp nhau đúng vào lúc 12 giờ.

Bài 44/2000 - Tạo ma trận số

(Dành cho học sinh THCS)

Program mang;

uses crt;

const n=9;

var a:array[1..n,1..n] of integer;

i,j,k:integer; t:boolean;

Begin

clrscr;

for j:=1 to n do

Begin

a[1,j]:=j;

a[j,1]:=a[1,j];

end;

```

i:=1;
repeat
  i:=i+1;
  for j:=i to n do
    begin
      t:= false;
      for k:= 2 to j-1 do if (a[k-1,i]>a[k,i]) then t:=true;
      if t then
        begin
          if a[j-1,i]+2 > n*2 then a[j,i]:=2 else a[j,i]:=a[j-1,i]+2;
          a[i,j]:=a[j,i];
        end
      else
        begin
          if a[j-1,i]+i>2*n then a[j,i]:=2 else a[j,i]:=a[j-1,i]+i;
          a[i,j]:=a[j,i];
        end;
      end;
    end;
until i=n;
for i:=1 to n do
  begin
    for j:=1 to n do write(a[i,j]:4);
    writeln;
  end;
readln;
end.

```

Bài 45/2000 - Các vòng tròn Olympic

(Dành cho học sinh THCS và PTTH)

```

{$Q-}
{$M 65000 0 655360}
Program Vong_Tron;
Uses  Crt,Dos;
Const Max = 39;
      Fileout = 'VTron.out';
      Dvt : array [1 .. 5,0 .. 8] of byte = ((8,1,2,3 ,4 ,5 ,6 ,7,8),
                                             (6,2,3,4 ,9 ,10,11,0,0),
                                             (6,4,5,6 ,11,12,13,0,0),
                                             (4,6,7,13,14,0 ,0 ,0,0),
                                             (4,1,2,9 ,15,0 ,0 ,0,0));
      D0 : array [1 .. 5] of byte = (8,11,13,14,15);
Type  Limt = 0 .. Max;
      Mang = array [Limt] of byte;
Var   A,B : Mang;
      dm : longint;
      fout : text;
      {-----}
Procedure Time;
  Var   h,k,i,j : word;
Begin

```

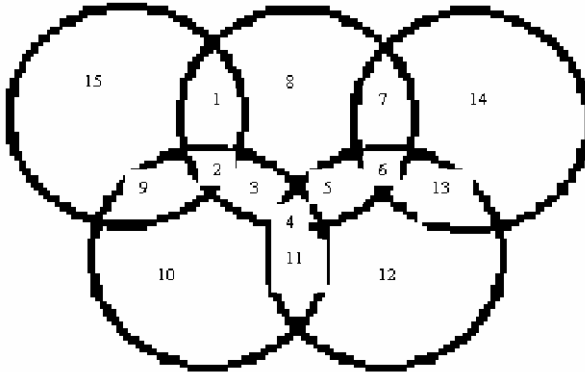
```

    Gettime(h,k,i,j);
    writeln(h,' : ',k,' : ',i,' ',j);
End;
{-----}
Procedure Output;
    Var   i,j : byte;
Begin
    Inc(dm);
    For i := 1 to 15 do write(fout,A[i],' ');
    writeln(fout);
End;
{-----}
Function GT(j0,count : shortint) : byte;
    Var   s,i0 : shortint;
Begin
    s := 0;
    For i0 := 1 to Dvt[j0,0] do
        if Dvt[j0,i0] <= count then Inc(s,A[Dvt[j0,i0]]);
    GT := s;
End;
{-----}
Procedure Try(s0,count,k0 : shortint);
    Var   i0 : shortint;
Begin
    if (count <= D0[k0]) and (s0 <= Max) then
        For i0 := 1 to Max-s0 do if B[i0] = 0 then
            Begin
                B[i0] := 1;
                A[count] := i0;
                if (count = D0[k0]) and (s0 + i0 = Max) then
                    Begin
                        if k0 = 5 then Output else Try(gt(k0 + 1,count),count + 1,k0 + 1);
                    End
                else Try(s0 + i0,count + 1,k0);
                B[i0] := 0;
            End;
End;
{-----}
Procedure Process;
Begin
    clrscr;
    Time;
    Assign(fout,fileout);rewrite(fout);
    Fillchar(A,sizeof(A),0);
    B:= A; dm := 0;
    Try(0,1,1);
    writeln(fout,'So cach : ',dm);
    close(fout); Time;
End;
{-----}
BEGIN

```

Process;
END.

Cách ghi kết quả trong file **Vtron.out** như sau: trong mỗi dòng ghi một cách đặt các số theo thứ tự từ 1 đến 15 theo cách đánh số như trên hình vẽ. Số cách xếp được ghi ở cuối tệp.



(Lời giải của bạn Đỗ Thanh Tùng - Lớp 12 Tin - PTH chuyên Thái Bình)

Bài 46/2000 - Đảo chữ cái

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

{SM 16384,0,655360}

(*Du lieu vao: file 'inp.txt' voi cac tu khac nhau, moi tu ghi o mot dong;

Du lieu ra: file 'out.txt' *)

PROGRAM Sinh_hoan_vi;

USES Crt;

CONST

MAX = 100;

INP = 'inp.txt';

OUT = 'out.txt';

TYPE

STR = array[0..max] of char;

VAR

s :str;

f,g :text;

n :longint; { so luong tu }

time:longint ;

PROCEDURE Nhap_dl;

Begin

Assign(f,inp);

Assign(g,out);

Reset(f);

Rewrite(g);

Readln(f,n);

End;

PROCEDURE DocDay(var s:str);

```

Begin
  Fillchar(s,sizeof(s),chr(0));
  While not eoln(f) do
    begin
      s[0]:=chr(ord(s[0])+1);
      read(f,s[ord(s[0])]);
    end;
End;

PROCEDURE VietDay(s:str);
Var i :word;
Begin
  For i:=1 to ord(s[0]) do Write(g,s[i]);
End;

PROCEDURE Sap_xep(l,r:word); { giai thuat Quicksort}
Var i,j :word;
    tg,tam :char;
Begin
  i:=l;j:=r;
  tg:=s[(l+r) div 2];
  Repeat
    While ord(s[i]) < ord(tg) do inc(i);
    While ord(s[j]) > ord(tg) do dec(j);
    If i<=j then
      begin
        tam:=s[i];
        s[i]:=s[j];
        s[j]:=tam;
        inc(i);
        dec(j);
      end;
  Until i>j;
  If j>l then Sap_xep(l,j);
  If i<r then Sap_xep(i,r);
End;

PROCEDURE Sinh_hv(s:str);
Var vti,vtj,i,j:word;
    stop :boolean;
    tam :char;
Begin
  Writeln(g);
  VietDay(s);
  Repeat
    Stop:=true;
    For i:= ord(s[0]) downto 2 do
      If s[i] > s[i-1] then
        begin
          vti:=i-1;

```

```

stop:=false;
For j:=ord(s[0]) downto vti+1 do
begin
  If (ord(s[j])>ord(s[vti])) then
    begin
      vtj:=j;
      break;
    end;
end;
tam:=s[vtj];
s[vtj]:=s[vti];
s[vti]:=tam;
For j:=1 to ((ord(s[0]) - (vti+1))+1) div 2 do
begin
  tam:=s[vti+j];
  s[vti+j]:=s[ord(s[0])-j+1];
  s[ord(s[0])-j+1]:=tam;
end;
Writeln(g);
VietDay(s);
break;
end;
Until stop;
End;

```

```

PROCEDURE Xu_ly;
Var i:longint;
Begin
  For i:=1 to n do
    begin
      DocDay(s);
      readln(f);
      Sap_xep(1,ord(s[0]));
      Sinh_hv(s);
      Writeln(g);
    end;
  Close(f);
  Close(g);
End;

```

```

BEGIN
  Nhap_dl;
  Xu_ly;
END.

```

(Lời giải của bạn Nguyễn Việt Bằng - Lớp 10 Tin - Trường phổ thông Năng Khiếu - ĐHQG TPHCM)

Bài 47/2000 - Xoá số trên vòng tròn

Lời giải 1:

Program vd;

```

Uses crt;
Var s:array[1..2000] of integer;
      i:integer;
Begin
  Clrscr;
  for i:=0 to 1999 do s[i]:=i+1;
  s[2000]:=1;
  i:=1;
  repeat
    s[i]:=s[s[i]];
    i:=s[i];
  until
    s[i]=i;
  writeln(i);
  readln;
End.

```

(Lời giải của bạn: Hà Huy Luân)

Lời giải 2:

```

Program xoa_so;
Const N=2000;
Var x:integer;

Function topow(x:integer):integer;
Var P:integer;
Begin
  P:=1;
  Repeat
    p:=p*2;
  Until p>x;
  topow:=p div 2;
End;

```

```

BEGIN
  x:=1+2*(N-topow(N));
  write(x);
END.

```

(Lời giải của bạn: Nguyễn Quang Trung - Lớp 12A Trường HERMANN GMEINER, Đà Lạt, Lâm Đồng)

Lời giải 3:

```

(* Thuật Giải Xu ly Bit *)
USES Crt;
CONST
  Max = 2000;
VAR
  A: array[0..(MAX div 8)] of byte;
  so: word;
FUNCTION Laybit(i:word):byte;
Var k:word;

```


Begin

```
k:=i div 8;
i:=i mod 8;
Laybit:=(a[k] shr (7-i)) and 1;
```

End;

PROCEDURE Tatbit(i:word);

Var k:word;

Begin

```
k:=i div 8;
i:=i mod 8;
a[k]:=a[k] and (not (1 shl (7-i)));
```

End;

FUNCTION Tim(j:word):word;

Begin

```
While (laybit(j+1)=0) do
  begin
    If j=max-1 then j:=0
    else inc(j);
  end;
  Tim:=j+1;
```

End;

PROCEDURE Xuly;

Var j,dem,i :word;

Begin

```
j:=1;dem:=0;
Fillchar(a,sizeof(a),255);
Tatbit(0);
Repeat
  If j=max then j:=0;
  j:=tim(j);
  Tatbit(j);
  inc(dem);
  If j=max then j:=0;
  j:=tim(j);
Until dem=max-1;
For i:=0 to (max div 8) do
  If a[i]<>0 then break;
  so:=i * (1 shl 3);
  For i:=so to so+7 do
  If Laybit(i)=1 then break;
  so:=i;
  Writeln(' SO TIM DUOC LA !',SO:4);
  Writeln(' Press Enter to Stop.....');
  readln;
```

End;

BEGIN

Clrscr;

Xuly;

END.

(Lời giải của bạn: Nguyễn Việt Bằng - Lớp 10 Tin - Phổ thông Năng Khiếu - ĐHQG.TPHCM)

Bài 48/2000 - Những chiếc gậy

(Dành cho học sinh THPT)

Program bai48;

Var x:array[0..10000] of word;

 d,a:array[1..1000] of byte;

 n,p,s,gtmax:word;

 fi,fo:text;

 ok:boolean;

Procedure Q_sort(l,k:word);

Var h,i,j,t:word;

Begin

 h:=a[(l+k)div 2];i:=l;j:=k;

 Repeat

 While a[i]>h do inc(i);

 While a[j]<h do dec(j);

 If i<=j then

 Begin

 t:=a[i];a[i]:=a[j];a[j]:=t;

 inc(i);dec(j);

 End;

 Until i>j;

 if i<k then Q_sort(i,k);

 if j>l then Q_sort(l,j);

End;

Procedure phan(var ok:boolean);

Var i,p1,j:word;

Begin

 Fillchar(x,sizeof(x),0);x[0]:=1;

 For i:=1 to n do

 If (d[i]=0) then

 For j:=p downto a[i] do

 If (x[j]=0) and(x[j-a[i]]<>0) then

 Begin

 x[j]:=i;

 if j=p then

 Begin

 j:=a[i];

 i:=n;

 End;

 End;

 ok:=(x[p]<>0);

 if ok then

 Begin

 p1:=p;

 Repeat

 d[x[p1]]:=1;

 p1:=p1-a[x[p1]];

 Until p1=0;

 End;

End;

```

Procedure chat(Var ok:boolean);
Var i:word;
Begin
  Fillchar(d,sizeof(d),0);
  Repeat
    phan(ok);
  Until not ok;
  ok:=true;
  for i:= n downto 1 do
  if d[i]=0 then
  Begin
    ok:=false;
    break;
  End;
End;

```

```

Procedure Tinh;
Begin
  For p:=gtmax to s div 2 do
  Begin
    chat(ok);
    if ok then
      Begin
        writeln(fo,p);
        break;
      End;
  End;
  If not ok then
    Writeln(fo,s);
End;

```

```

Procedure Start;
Var i:word;
Begin
  assign(fi,'input.txt');reset(fi);
  assign(fo,'output.txt');rewrite(fo);
  While not seekeof(fi) do
  Begin
    Readln(fi,n);
    if n<>0 then
      Begin
        gtmax:=0;s:=0;
        for i:=1 to n do
          Begin
            Read(fi,a[i]);
            s:=s+a[i];
            if a[i]> gtmax then
              gtmax:=a[i];
          End;
        Q_sort(1,n);
        Tinh;
      End;
  End;
  Close(fi);Close(fo);
End;

```

```

Begin
  Start;
End.

```

9
5 2 1 5 2 1 5 2 1
4
1 2 3 4
0

(Lời giải của bạn Tăng Hải Anh - Hải Dương - TP. Hải Phòng)

Bài 49/2001 - Một chút nhanh trí

(Dành cho học sinh Tiểu học)

Theo giả thiết khi chia A và lập phương của A cho một số lẻ bất kỳ thì nhận được số dư như nhau, tức là: $A^3 \pmod{N} = A \pmod{N}$, ở đây N số lẻ bất kỳ, chọn N lẻ sao cho $N > A^3$ thì ta phải có $A^3 = A$ suy ra $A=1$.

Vậy chỉ có số 1 thoả mãn điều kiện của bài toán.

Bài 50/2001 - Bài toán đổi màu bi

(Dành cho học sinh THCS và PTTH)

Program ba_bi;

Uses crt;

var v,x,d:integer;

BEGIN

 Clrscr;

 writeln('v x d ?(>=0)');

 readln(v,x,d);

 if ((v-x) mod 3 = 0) and ((x+d)*(v+d) <> 0) then

 while (v+x) <> 0 do

 begin

 d:=d-1+3*((3*v*x) div (3*v*x-1));

 x:=x+2-3*((3*x) div (3*x-1));

 v:=v+2-3*((3*v) div (3*v-1));

 writeln('>> ',v,' ',x,' ',d);

 end

 else writeln('Khong duoc !');

 readln;

END.

(Lời giải của bạn: Nguyễn Quang Trung)

Bài 51/2001 - Thay thế từ

(Dành cho học sinh THCS và PTTH)

program thaythetu;

var

 source,des:array[1..50]of string;

 n:byte;

procedure init;

var

 i:byte;

 s:string;

 f:text;

begin

 assign(f,'input2.txt');

 reset(f);

 n:=0;

 while not eof(f) do

```

begin
  readln(f,s);
  inc(n);
  while (s<>")and(s[1]=' ') do
    delete(s,1,1);
  if i>0 then
    begin
      i:=pos(' ',s);
      des[n]:=copy(s,1,i-1);
      while (i<=length(s))and(s[i]=' ') do
        i:=i+1;
      source[n]:=copy(s,i,length(s)-i+1);
    end;
  end;
end;

procedure replace;
var
  f,g:text;
  s:string;
  i,k:byte;
begin
  assign(f,'input1.txt');
  reset(f);
  assign(g,'kq.out');
  rewrite(g);
  while not eof(f) do
    begin
      readln(f,s);
      for k:=1 to n do
        for i:=1 to length(s)-length(des[k])+1 do
          if des[k]=copy(s,i,length(des[k])) then
            begin
              delete(s,i,length(des[k]));
              insert(source[k],s,i);
              i:=i+length(source[k]);
            end;
        writeln(g,s);
      end;
    close(f);
    close(g);
  end;

begin
  init;
  replace;
end.

```

Bài 52/2001 - Xác định các tứ giác đồng hồ trong ma trận
(Dành cho học sinh THCS và PTTH)

```

uses crt;
var s,n,i,k,j,a1,a2,b1,b2:integer;
    chon,mau:byte;
    a:array[1..100,1..100]of integer;
    {-----}
procedure nhap;
begin

```

```

write('nhap n>=2:');readln(n);
for i:=1 to n do
  for j:=1 to n do
    begin
      write('nhap a['i,j]:');
      readln(a[i,j]);
    end;
  end;
  {-----}
procedure tinh;
begin
  clrscr;
  nhap;
  s:=0;
  for i:=1 to n-1 do
    for j:=1 to n-1 do
      if ((a[i,j]<a[i,j+1])and(a[i,j+1]<a[i+1,j+1])and(a[i+1,j+1]<a[i+1,j]))
        or((a[i,j+1]<a[i+1,j+1])and(a[i+1,j+1]<a[i+1,j])and(a[i+1,j]<a[i,j]))
        or((a[i+1,j+1]<a[i+1,j])and(a[i+1,j]<a[i,j])and(a[i,j]<a[i,j+1]))
        or((a[i+1,j]<a[i,j])and(a[i,j]<a[i,j+1])and(a[i,j+1]<a[i+1,j+1]))
          then inc(s);
    end;
  writeLn;
  writeLn;
  writeLn;
  writeLn('So luong tu giac dong ho la:',s);
  readln;
  end;
  {-----}
procedure max;
var t:integer;
begin
  writeLn('Nhap n>=2:');readln(n);
  i:=1;
  a1:=1;a2:=n;
  b1:=1;b2:=n;
  mau:=0;
  t:=0;
  while i<=n*n do
    begin
      for k:=a1 to a2 do
        begin
          a[b1,k]:=i;
          gotoxy(5*k,b1);
          inc(mau);
          if mau>15 then mau:=1;
          textcolor(mau);
          write(i);
          delay(70);inc(i);
        end;
      for k:=b1+1 to b2+t do

```

```

begin
  a[k,a2]:=i;
  gotoxy(5*(a2),k);
  inc(mau);
  if mau>15 then
    mau:=1;
  textcolor(mau);
  write(i);
  delay(70);
  inc(i);
end;
for k:=b2+t downto b1+1 do
begin
  a[k,b2]:=i;
  gotoxy(5*(b2-1),k);
  inc(mau);
  if mau>15 then mau:=1;
  textcolor(mau);
  write(i);
  delay(70);
  inc(i);
end;
for k:=a2-2 downto a1 do
begin
  a[b1+1,k]:=i;
  gotoxy(5*k,b1+1);
  inc(mau);
  textcolor(mau);
  write(i);
  delay(70);
  inc(i);
end;
dec(a2,2);
dec(b2,2);
inc(t,2);
inc(b1,2);
end;
if n>2 then s:=3*(n-2) else s:=1;
writeln;writeln;
writeln('Bang dong ho max');writeln;
writeln('Voi ma tran vuong cap ',n,'thi so luong tu giac dong ho lon nhat la:',s);
readln;
End;
{-----}
procedure min;
begin
  clrscr;
  writeln('n>=2:');readln(n);
  i:=1;
  b1:=1;

```

```

while i<=n*n do
begin
  for k:=1 to n do
  begin
    a[b1,k]:=i;
    inc(mau);
    if mau>15 then mau:=1;
    textcolor(mau);
    gotoxy(5*k,b1);
    write(i);
    delay(70);
    inc(i);
  end;
  inc(b1);
end;
writeln;writeln;writeln('Bang tren s co gia tri=0');
readln;
End;
{-----}
BEGIN
Clrscr;
repeat
  textcolor(white);
  writeln('1:cau a (Tinh so luong S)');
  writeln('2:cau b (Lap bang co S lon nhat)');
  writeln('3:cau c (Lap bang co S nho nhat)');
  writeln('4:thoat!');
  writeln('Chon chuc nang:');readln(chon);
  case chon of
    1: begin
        clrscr;
        tinh;
      end;
    2: begin
        clrscr;
        max;
      end;
    3: begin
        clrscr;
        min;
      end;
  end; {of Case}
  clrscr;
until chon=4;
END.

```

(Lời giải của bạn: Nguyễn Việt Hoà)

Bài 53/2001 - Lập lịch tháng kỳ ảo
(Dành cho học sinh THCS và PTTH)
 (* Tất cả các lịch đều là lịch ki ao *)


```

Program bai 53;
uses crt;
Const out='lichao.out';
Type mang=array[1..6,1..7] of integer;
Var a:mang;
    i,j,dem:integer;
    s:real;
    f:text;
(*-----*)
PROCEDURE Viet;
Var i,j:integer;
Begin
    inc(dem);
    writeln(f,'Kha nang thu ',dem);
    for i:=1 to 6 do
        begin
            for j:=1 to 7 do
                if a[i,j]<>0 then write(f,a[i,j]:3)
                    else write(f,":3");
            writeln(f);
        end;
    writeln(f);
End;
(*-----*)
PROCEDURE Laplich(k,t:integer);
Var i,j,i1:integer;
Begin
    for i1:=k to t+k-1 do
        begin
            j:=i1 mod 7;
            i:=i1 div 7;
            if j=0 then
                begin
                    j:=7;
                    dec(i);
                end;
            a[i+1,j]:=i1-k+1;
        end;
    viet;
End;
(*-----*)
PROCEDURE Xuli;
Var i,j,k,t:integer;
Begin
    for k:=1 to 7 do
        for t:=28 to 31 do
            begin
                fillchar(a,sizeof(a),0);
                Laplich(k,t);
            end;
        end;
    end;

```

```

End;
(*-----*)
BEGIN
  clrscr;
  assign(f,out);
  rewrite(f);
  dem:=0;
  Xuli;
  close(f);
END.

```

(Lời giải của bạn: Đỗ Ngọc Sơn)

Bài 54/2001 - Bạn hãy gạch số

(Dành cho học sinh Tiểu học và THCS)

Chúng ta viết ra 10 số nguyên tố đầu tiên:

2 3 5 7 11 13 17 19 23 29

là số có 16 chữ số, có thể chứng minh không khó khăn lắm rằng sau khi gạch đi 8 chữ số thì số nhỏ nhất có thể được là: 11111229; còn số lớn nhất có thể được là: 77192329. Thật vậy:

a. Gạch đi 8 chữ số, để số còn lại là một số có 8 chữ số là nhỏ nhất (giữ nguyên thứ tự ban đầu). Nhìn vào dãy số ở trên ta thấy số 1 là nhỏ nhất, có năm chữ số 1 và sau chữ số 1 thứ năm này lại còn nhiều hơn 3 chữ số khác nữa. Do đó, 5 chữ số đầu của số cần tìm chắc chắn phải là 5 chữ số 1. Lí luận tương tự, để tìm được 3 chữ số còn lại.

b. Tương tự như thế: chữ số 9 là lớn nhất, nhưng sau chữ số 9 đầu tiên lại chỉ còn lại 4 chữ số (mà ta cần giữ lại số có 8 chữ số), nên ta không thể chọn số 9 là chữ số đứng đầu trong 8 chữ số cần tìm. Chữ số lớn thứ hai là 7, có hai chữ số 7, tất nhiên ta chọn chữ số 7 đầu tiên (vì sau chữ số 7 thứ 2 chỉ còn lại 6 chữ số). Lí luận tương tự, ta tìm được chữ số thứ hai trong 8 chữ số cần tìm cũng là chữ số 7, và 6 chữ số còn lại phải tìm tất nhiên là 6 chữ số sau chữ số 7 này.

Bài 55/2001 - Bài toán che mắt mèo

(Dành cho học sinh THCS và PTTH)

Program Che_Mat_meo;

Uses crt;

Const td=200;

Var i,j,n:integer;

out:string;

f:text;

Procedure Xuli;

Begin

for i:=1 to n do

begin

gotoxy(15,i+3);

for j:=1 to n do

begin

if (odd(i))and(odd(j)) then

begin

textcolor(11);

if out<>" then write(f,'M ')

```

    else
      begin
        write('M ');
        delay(td);
      end;
    end
  else
    begin
      textcolor(14);
      if out<>" then write(f,'o ')
        else
          begin
            write('o ');
            delay(td);
          end;
        end;
      end;
    end;
  writeln(f);
end;
End;

```

BEGIN

```

  Clrscr; textcolor(2);
  Write('Nhap n= ');
  Readln(n);
  if n<=20 then out:=""
  else
    begin
      out:='matmeo.inp';
      writeln('Mo File meo.inp de xem ket qua');
    end;
  Assign(f,out);
  Rewrite(f);
  writeln(f,'(Chu M Ki hieu cho con meo, chu o ki hieu cho quan co)');
  Xuli; writeln(f);
  Writeln(f,'Tong cong co ',sqr((n+1) div 2),' con meo');
  Close(f);
  Readln;
END.

```

(Lời giải của bạn Đỗ Ngọc Sơn - Quảng Ninh)

Bài 56/2001 - Chia lưới

(Dành cho học sinh PTTH)

```

Program Chia_luoi ;
Uses Crt ;
Const Fi = 'LUOI.INP';
      Fo = 'LUOI.OUT';
Var A : Array[1..20,1..20]Of Integer ;

```

```

    B : Array[1..20,1..20]Of 0..1 ;
    Px,Py: Array[1..4] Of ShortInt ;
    M,N,S,S1,S2 : LongInt ;
    F : Text ;
Procedure Read_Input ;
    Var i,j :Integer;
Begin
    Clrscr ; S:= 0 ;
    Assign(F,Fi) ;Reset(F) ;
    Readln(F,M,N);
    For i:=1 to M do
    Begin
    For j:=1 to N do
    Begin
    Read(F,A[i,j]);
    S:=S+A[i,j];
    End;
    Readln(F);
    End;
    Close(F);
End;

Procedure Inmit ;
Begin
    S1 := S div 2;
    Px[1]:= 0 ;Px[2]:= 0 ;Px[3]:=1 ;Px[4]:=-1 ;
    Py[1]:= 1 ;Py[2]:=-1 ;Py[3]:=0 ;Py[4]:= 0 ;
End ;

Procedure Write_Output ;
    Var i,j :Integer;
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);Halt;
End;

Function Ktra(x,y : Integer) : Boolean ;
Begin
    Ktra:= False ;
    If (x in [1..M]) And (y in [1..N]) And
    (B[x,y] = 0 ) Then Ktra := True ;
End;

Procedure Try(x,y:Integer ;Sum :LongInt);

```

```

Var i :Integer ;
Begin
  For i:=1 to 4 do
    If Ktra(x+Px[i],y+Py[i]) Then
      Begin
        x := x + Px[i] ;
        y := y + Py[i] ;
        Sum := Sum + A[x,y];
        B[x,y] := 1;
        If Sum = S2 Then Write_Output ;
        Try(x,y,Sum) ;
        Sum := Sum - A[x,y];
        B[x,y] := 0;
        x := x - Px[i] ;
        y := y - Py[i] ;
      End ;
    End;

```

```

Procedure Run ;
  Var i,j : Integer ;
Begin
  Read_Input ;Imit ;
  For i:=1 to M do
    For j:=1 to N do
      If A[i,j]>= S1 Then
        Begin
          Fillchar(B,SizeOf(B),0);
          B[i,j]:=1;
          Write_Output;
        End ;
      For S2 := S1 downto 1 do
        Begin
          Fillchar(B,SizeOf(B),0);
          B[1,1]:=1;
          Try(1,1,A[1,1]);
        End;
    End;
End;

```

BEGIN

Run;

END.

(Lời giải của bạn Lê Sơn Tùng - Vĩnh Phúc)

Bài 57/2001 - Chọn số

(Dành cho học sinh Tiểu học và THCS)

Giả sử có m số 1, n số -1 (m, n nguyên dương) theo giả thiết:

a) $m + n = 2000$, suy ra m, n cùng tính chẵn lẻ.

+ Nếu m chẵn, do đó n cũng chẵn, ta chọn ra $m/2$ số 1 và $n/2$ số -1.

+ Nếu m lẻ, n lẻ:

$$m = 2k + 1 = k + (k + 1)$$

$$n = 2q + 1 = q + (q + 1)$$

Luôn có: $k - q = (k+1) - (q+1)$, do đó ta sẽ chọn k số 1 và q số -1.

Vậy ta luôn có thể chọn ra các số thỏa mãn điều kiện của bài toán.

b) $m + n = 2001 \rightarrow m$ và n không cùng tính chẵn lẻ.

+ Nếu m chẵn $\rightarrow n$ phải là lẻ:

$$m = 2k = i + j \text{ (giả sử chọn } i \text{ số 1, giữ lại } j \text{ số 1)}$$

$$n = 2q + 1 = t + s \text{ (giả sử chọn } t \text{ số -1, giữ lại } s \text{ số -1)}$$

Theo cách chọn này $\rightarrow i, j$ phải cùng tính chẵn lẻ; t, s không cùng tính chẵn lẻ.

Giả sử i chẵn, j chẵn, t lẻ, s chẵn, do đó: $i + t \neq j + s$, như vậy cách chọn này không thỏa mãn. Các trường hợp còn lại xét tương tự.

Do đó, với trường hợp này không thể có cách chọn nào thỏa mãn điều kiện của bài toán.

Bài 58/2001 - Tổng các số tự nhiên liên tiếp

(Dành cho học sinh THCS và PTTH)

Program bai58;

Uses crt;

var N:longint;

m,i,dem,a,limit:longint;

procedure Solve;

begin

Writeln('Chia so ',N,':');

limit:=trunc(sqrt(1+8*N)+1) div 2;

for m:=2 to limit-1 do

if ((N-m*(m-1) div 2) mod m =0) then

begin

a:=(N-m*(m-1) div 2) div m;

inc(dem);

writeln('+ Cach thu ',dem,':');

for i:=a to a+m-1 do

begin

write(' ',i);

if (i-a+1) mod 10=0 then writeln;

end;

writeln;

end;

end;

BEGIN

chrscr;

writeln('Nhap N: ');readln(N);

Solve;

if dem=0 then writeln('Khong the chia!')

else writeln('Co tat ca', dem, ' cach chia!');

readln;

END.

(Lời giải của bạn Nguyễn Quốc Quân - Lớp 11 T₂ - Trường PTTH Lê Viết Thuật - Vinh)

Bài 59/2001 - Đếm số ô vuông*(Dành cho học sinh THCS và PTTH)***Uses** crt;**Const** Ngang = 'ngang.inp';

Doc = 'doc.inp';

Max = 100;

n: integer = 0;

count: integer = 0;

Var f1,f2:text;

o,i,j:integer;

a,b,c:array[1..max] of boolean;

BEGIN

clrscr;

Assign(f1,ngang); Assign(f2,doc);

Reset(f1); Reset(f2);

While not eoln(f1) **do****begin**

Read(f1,o);

Inc(n);

If o=1 **then** a[n]:=true**else** a[n]:=false**end;**

Readln(f1);

for i:= 1 **to** n **do****begin****for** j:= 1 **to** n **do****begin**

Read(f1,o);

If o=1 **then** b[j]:=true**else** b[j]:=false;**end;**

Readln(f1);

for j:=1 **to** n+1 **do****begin**

Read(f2,o);

If o=1 **then** c[j]:=true**else** c[j] := false**end;**

Readln(f2);

for j:=1 **to** n **do****begin****If** (a[j] **and** b[j] **and** c[j] **and** c[j+1]) **then**

inc(count);

end;

a:=b;

end;

Close(f1); Close(f2);

Write('Co', count, 'hinh vuong!');

Readln;

END.

(Lời giải của bạn Nguyễn Chí Thức - Lớp 10A1 - Khối chuyên Toán Tin - ĐH Sư phạm Hà Nội)

Bài 60/2001 - Tìm số dư của phép chia

(Dành cho học sinh Tiểu học)

Vì 1976 và 1977 là 2 số nguyên liên tiếp nên nguyên tố cùng nhau, do đó số thoả mãn điều kiện của bài toán phải có dạng:

$$n = 1976 * 1977 * k + 76 \quad (k \text{ là số nguyên})$$

nhưng $1976 * 1977$ lại chia hết cho 39 nên phần dư của n khi chia cho 39 sẽ là 37 (= 76 - 39).

Bài 61/2001 - Thuật toán điền số vào ma trận

(Dành cho học sinh THCS và PTTH)

Program Bai61;

Uses crt;

Var a:array[2..250,2..250] of -1..1;

n,i,j:integer;

BEGIN

Write('Doc vao n:'); Readln(n);

Fillchar(a, sizeof(a), 0);

for i:=1 to n do

for j:=1 to n do

begin

If (i mod 2 <> 0) and (j mod 2 <> 0) then a[i,i] := 1;

If (i mod 2 = 0) and (j mod 2 = 0) then a[i,i] := -1;

end;

Writeln('Mang da dien la: ');

for i:=1 to n do

begin

for j:=1 to n do Write(a[i,j]:3);

Writeln;

end;

Write('Tong lon nhat la:');

If n mod 2 = 0 then Write(0) else Write(n);

Readln;

END.

(Lời giải của bạn Trương Đức Hạnh - 12 Toán Năng Khiếu - Hà Tĩnh)

Bài 62/2001 - Chèn Xâu

(Dành cho học sinh THCS và PTTH)

Do sơ xuất khi ra đề nên trong số các lời giải của bạn đọc gửi đến toà soạn, có thể các bạn đã hiểu đề bài theo 2 cách sau đây, ta coi như hai bài toán:

1. Nếu theo ví dụ, thì ta cần chèn dấu vào xâu (không cần đủ 9 số như trong xâu S, có thể bớt một số số cuối của xâu, nhưng phải theo thứ tự) để phép tính nhận được bằng M cho trước.

2. Ta không để ý đến ví dụ của đề ra, yêu cầu cần chèn dấu vào giữa các số trong xâu '123456789' để nhận được kết quả M cho trước.

Sau đây là lời giải của bạn Nguyễn Chí Thức (hiểu theo bài toán 1):


```

Program Bai62;
Uses crt;
Const fo = 'chenxau.out';
  dau: array[1..3] of String[1]= ('-', '-', '+');
  s:array[1..9] of char=('1','2','3','4','5','6','7','8','9');
Var d:array[1..9] of String[1];
  m:longInt;
  f:text;
  k:integer;
  found:boolean;
Procedure Init;
Begin
  Write('Cho M=');
  Readln(m);
  found:=false;
end;
Function tinh(s:string):longint;
Var i,t:longint;
  code:integer;
Begin
  i:=length(s);
  While not(s[i] in ['-','+']) and (i>0) do dec(i);
  val(copy(s,i+1,length(s)-i),t,code);
  If i=0 then begin tinh:=t; exit; end
  else
  begin
    delete(s,i,length(s)-i+1);
    If s[i]='+' then tinh:=t+tinh(s);
    If s[i]='-' then tinh:=tinh(s)-t;
  end;
End;
Procedure Test(i:integer);
Var st:string; j:integer;
Begin
  st:="";
  For j:=1 to i do st:=st+d[j]+s[j];
  If Tinh(st) = m then begin writeln(f,st); found:=true; end;
End;
Procedure Try(i:integer);
Var j:integer;
Begin
  for j:=1 to 3 do
  begin
    d[i]:=dau[j]; Test(i);
    If i<9 then try(i+1);
  end;
End;
BEGIN
  Chrscr;
  Init;

```

```

Assign(f,fo);Rewrite(f);
for k:=1 to 2 do
begin
  d[1]:=dau[k];
  Try(2);
end;
If not found then write(f,'khong co ngiem');
Close(f);
END.

```

Từ lời giải trên của bạn Thức, để thoả mãn yêu cầu của bài toán 2, trong thủ tục Try cần sửa lại như sau:

```

Procedure Try(i:integer);
Var j:integer;
Begin
  for j:=1 to 3 do
begin
  d[i]:=dau[j];
  If i<9 then try(i+1);
  If i=9 then Test(i);
end;
End;

```

Bài 63/2001 - Tìm số nhỏ nhất

(Dành cho học sinh Tiểu học)

a. Số đó chia hết cho 9 nên tổng các chữ số của nó phải chia hết cho 9. Ta thấy tổng $0 + 1 + 2 + 3 + 4 + 5 + 6 + 7 + 8 + 9 = 45$ chia hết cho 9. Vậy số nhỏ nhất bao gồm tất cả các chữ số 0, 1, 2, ..., 9 mà chia hết cho 9 là: 1023456789.

b. Số này chia hết cho 5 nên tận cùng phải là 0 hoặc 5. Nếu tận cùng là 5 thì số nhỏ nhất sẽ là 1023467895 còn nếu số đó tận cùng là 0 thì số nhỏ nhất sẽ là 123457890.

So sánh hai số trên, suy ra số nhỏ nhất phải tìm là: 1023467895

c. Một số chia hết cho 20, do đó phải chia hết cho 10. Suy ra số đó phải là số nhỏ nhất tận cùng là 0. Mặt khác, chữ số hàng chục của số đó phải là một số chẵn. Vì vậy ta tìm được số phải tìm là 1234567980.

Bài 64/2001 - Đổi ma trận số

(Dành cho học sinh THCS và PTTH)

```

Program DoiMT;
Uses Crt;
Const nmax=50;
  inp='INPUT.TXT'; {Du lieu duoc nhap vao file input.txt}
Type Mang=array [1..nmax,1..nmax] of real;
Var a,b,c: Mang;
  n,i,j: integer;

Procedure Nhap;
Var i,j: integer;
  f: text;
Begin
  Assign(f,inp); Reset(f);

```

```

Readln(f,n);
For i:=1 to 2*n do
begin
  For j:=1 to 2*n do Read(f,c[i,j]);
  Readln(f);
end;
Close(f);
End;
Procedure Xuat(a: Mang);
Var i,j: integer;
Begin
  For i:=1 to 2*n do
  begin
    For j:=1 to 2*n do Write(a[i,j]:8:2);
    Writeln;
  end;
End;
BEGIN
Nhap;
For i:=1 to n do
  For j:=1 to n do
  begin
    a[i+n,j+n]:=c[i,j];
    a[i,j+n]:=c[i+n,j];
    a[i,j]:=c[i+n,j+n];
    a[i+n,j]:=c[i,j+n];
    b[i,j]:=c[i+n,j];
    b[i,j+n]:=c[i,j];
    b[i+n,j+n]:=c[i,j+n];
    b[i+n,j]:=c[i+n,j+n];
  end;
ChrScr;
Xuat(c); {mang ban dau}
Writeln;
Xuat(a);
Writeln;
Xuat(b);
Readln;
END.

```

(Lời giải của bạn Lê Thanh Tùng - Vĩnh Yên - Vĩnh Phúc)

Bài 65/2001 - Lưới ô vuông vô hạn
(Dành cho học sinh THCS và PTTH)

```

Program bai65;
uses crt;
var
  a:array[1..100,1..100] of integer;
  b,i,j,n,m,k:integer;
  f:text;

```

```

t:boolean;
Begin
clrscr;
write('Nhap so n: '); readln(n);
write('Nhap so m: '); readln(m);
for i:=1 to m do
  for j:=1 to n do a[i,j]:=-1;
for i:=m downto 1 do
  for j:=1 to n do
    begin
      b:=-1;
      repeat
        inc(b); t:=true;
        for k:=1 to n do if a[i,k]=b then t:=false; {kt hang}
        for k:=1 to m do if a[k,j]=b then t:=false; {kt cot}
      until t;
      a[i,j]:=b;
    end;
assign(f,'KQ.TXT');
rewrite(f);
for i:=1 to m do
  begin
    for j:=1 to n do write(f,a[i,j]:5);
    writeln(f);
  end;
close(f);
write('Mo file KQ.TXT de xem ket qua!');
readln;
END.

```

(Lời giải của bạn Nguyễn Trường Đức Trí)

Bài 66/2001 - Bảng số 9 x 9

(Dành cho học sinh Tiểu học và THCS)

Ta sẽ điền vào các ô ở cột thứ năm các số lớn nhất có thể được. Nếu số lớn nhất trong các cột còn lại (chưa điền vào bảng) là a , thì số lớn nhất có thể điền vào cột thứ năm là $a-4$ vì các số phải điền theo thứ tự tăng dần theo hàng mà sau cột thứ 5 còn có 4 cột nữa. Ta thực hiện điền các số giảm dần từ 81 vào nửa phải của bảng trước, sau đó dễ dàng điền vào nửa còn lại với nhiều cách khác nhau:

1	2	3	4	77	78	79	80	81
5	6	7	8	72	73	74	75	76
9	10	11	12	67	68	69	70	71
13	14	15	16	62	63	64	65	66
17	18	19	20	57	58	59	60	61
21	22	23	24	52	53	54	55	56
25	26	27	28	47	48	49	50	51
29	30	31	32	42	43	44	45	46
33	34	35	36	37	38	39	40	41

Program bai66;

Uses ctr ;

```

Var i,j : integer ;
Begin
  ClsScr;
  for i:= 1 to do
    begin
      for j:= 1 to 4 do write (4*(i-1) + j :3);
      for j:= 0 to 4 do write (81-4*i-(i-1)+j :3) ;
      Writeln;
    end ;
  Write ('tong cac so o cot 5: ',(37+77)*9div2);
  Readln
End.

```

(Lời giải của bạn Nguyễn Chí Thức - Lớp 11A1 - Khối PTCTT - ĐHSPhN - Thôn Đại Đồng - xã Thụy Phương - Từ Liêm - Hà Nội)

Bài 67/2001 - Về các phép biến đổi "Nhân 2 trừ 1"

(Dành cho học sinh THCS và PTTH)

Để biến đổi ma trận A thành 0, ta biến đổi từng cột thành 0

Xét một cột bất kì có n số a_1, \dots, a_n ($a_i \geq 0$)

Đặt $X = \max(a_1, \dots, a_n)$.

- Bước 1:

+ Nếu dãy a_1, \dots, a_n có một số 0 và một số khác 0, dừng ở đây vì không thể đưa A về 0;

- Bước 2:

+ Nếu dãy a_1, \dots, a_n có $a_i = 0$ ($i = 1..n$) thì cột này đã được biến đổi xong, qua cột tiếp theo,

+ Nếu không thì $a_i = 2a_i$ nếu $2a_i \leq X$ (nhân hàng có chứa số a_i lên 2), tiếp tục thực hiện đến khi không nhân được nữa, qua bước 3;

- Bước 3:

$X := X - 1;$

$a_i := a_i - 1;$

Quay lại bước 2.

Đây không phải là lời giải tốt nhất nhưng rất đơn giản, dễ dàng cài đặt (việc viết chương trình tương đối đơn giản)

Nhân xét: Bài này thực sự dễ nếu chỉ dừng lại ở mức tìm thuật toán? Nếu đặt lại điều kiện là có thể nhân hàng, cột cho 2, trừ hàng, cột cho 1, tìm lời giải tối ưu với giới hạn của M, N thì hay hơn nhiều.

(Lời giải của bạn Vũ Lê An - Lớp 11T2 - Lê Khiết - Quảng Ngãi)

Thuật toán của bạn Vũ Lê An rất đúng. Song trên thực tế thuật toán này còn một điểm chưa chuẩn vì nếu các số của mảng số thì nhỏ, số thì lớn thì thuật toán này mất rất nhiều bước. Việc nhân có thể gây ra tràn số.

Ví dụ:

2 3

1 100 1

100 1 100

số bước sẽ rất lớn.

Nhưng thuật toán này trên lý thuyết là giải được. Chương trình theo thuật toán trên.

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

{SM 16384,0,655360}

program bai67_bien_doi_mang; {Author : Nguyen Van Chung}

```

uses crt;
const    max      =100;
        fi       ='bai67.inp';
        fo       ='bai67.out';
var      a        :array[1..max,1..max]of longint;
        m,n      :integer;

```

```

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

```

```

procedure lam;
var f      :text;
    i,j,ma,mi,k :longint;
begin
  assign(f,fo);
  rewrite(f);
  for j:=1 to n do
    begin
      ma:=0;mi:=maxlongint;
      for i:=1 to m do
        begin
          if a[i,j]>ma then ma:=a[i,j];
          if a[i,j]<mi then mi:=a[i,j];
        end;
      if (ma>0)and(mi=0) then
        begin
          rewrite(f);
          writeln(f,'No solution');
          break;
        end;
    repeat
      for i:=1 to m do
        begin
          while a[i,j]*2<=ma do
            begin
              for k:=1 to n do a[i,k]:=a[i,k]*2;
              writeln(f,'nhân 2 dòng :',i);
            end;
          a[i,j]:=a[i,j]-1;
        end;
      dec(ma);
    until

```

```

    writeln(f,'tru 1 cot :',j);
    until ma=0;
  end;
  close(f);
end;

```

```

BEGIN
  docf;
  lam;
END.

```

Bài 68/2001 - Hình tròn và bảng vuông

(Dành cho học sinh PTTH)

+ Tính số ô vuông bị cắt bởi hình tròn:

Nếu trục tọa độ là (0,0) thì tâm vòng tròn có tọa độ (n,n). Xét 1 phần 4 vòng tròn từ 6 giờ đến giờ ô bị cắt là ô có đỉnh (i,j) nằm ngoài vòng tròn và 1 đến 3 đỉnh (i+1, j), (i, j+1), (i+1, j+1) trong vòng tròn. Do tính đối xứng ta chỉ cần tính số ô của 1 phần 4 vòng tròn rồi nhân với 4. Tuy nhiên nếu nhận xét kỹ hơn ta thấy với $n = 2$, số ô bị cắt là 12, khi n tăng 1 đơn vị, số ô bị cắt tăng lên 8 ô. Do đó ta có thể tính thẳng số ô bị cắt bằng công thức : Số ô bị cắt = $12 + (n-2)*8$

+ Tính số ô nằm trong vòng tròn:

Cũng do tính đối xứng ta chỉ cần tính số ô nằm trong 1 phần 4 vòng tròn rồi nhân với 4, ô nằm trong vòng tròn khi tất cả 4 đỉnh nằm trong vòng tròn.

Chương trình Pascal

Uses Ctr;

Const S1 ='INPUT.TXT';

S2='OUTPUT.TXT';

Var F1F2: text;

I,J,N : word;

Dem :longint;

FunctionTrong(X,Y: longint): boolean;

Begin

Trong:= $4*(\text{sqr}(X-N)+\text{sqr}(Y-N))\leq\text{sqr}(2*N-1)$;

End

BEGIN

Clrscr;

Assign(F1,S1);

Reset(F1);

Assign(F2,S2);

Rewrite(F2);

While not eof(F1) do

Begin

Readln(F1,N);

Write(F2,'N=', '>', $12+((N-2)*8)$);

Dem:= 0;

For I:= 0 to N-1 do

For J:= 0 to J-1 do

If Trong (I,J) and Trong (I+1,J) and Trong (I,J+1) and Trong (I+1, J+1) then(Dem)

Writeln(F2, '', Dem*4);

End;

Close(F1);

Close(F2);

End.

(Lời giải của bạn Lâm Tấn Minh Tâm - 12 Tin trường PTH Chuyên Tiền Giang- **Tiền Giang**)

Bài 69/2001 - Bội số của 36

(Dành cho học sinh Tiểu học)

Một số đồng thời chia hết cho 4 và 9 thì sẽ chia hết cho 36 (vì 4 và 9 nguyên tố cùng nhau: $(4, 9) = 1$).

Ta thấy, tổng của tất cả các số từ 1 đến 9 $= 1 + 2 + \dots + 9 = 45$ chia hết cho 9.

Một số chia hết cho 4 khi và chỉ khi hai chữ số cuối cùng của nó chia hết cho 4. Mà ta cần tìm số nhỏ nhất chia hết cho 36, do đó số đó phải là **số nhỏ nhất** có đầy đủ các chữ số từ 1 đến 9 và **hai số cuối cùng** của nó phải là một số chia hết cho 4. Vậy số phải tìm là: 123457896

Bài 70/2001 - Mã hoá theo khoá

(Dành cho học sinh THCS và THPT)

```
{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 MaxVal=256;
```

```
Var
```

```
n:Integer; S,KQ:String;
```

```
a:array[0..MaxVal] of Integer;
```

```
Procedure InPut;
```

```
Var i:Integer;
```

```
Begin
```

```
CLrscr;
```

```
Write('Nhap N=');Readln(n);
```

```
For i:=1 to n do
```

```
Begin Write('a['i,']=');Readln(a[i]); End;
```

```
Write('Nhap Xau:');Readln(S);
```

```
End;
```

```
Procedure Main;
```

```
Var i,j:Integer;
```

```
Begin
```

```
if (Length(S) Mod n) <>0 then
```

```
For i:=1 to n-(Length(S) Mod n) do S:=S+' ';
```

```
KQ:='';
```

```
For i:=0 to (Length(S) Div n)-1 do
```

```
For j:=(n*i)+1 to n*(i+1) do
```

```
KQ:=KQ+S[a[j-(n*i)]+(n*i)];
```

```
Writeln('Xau Ma Hoa: ',KQ);
```

```
End;
```

```
Begin
```

```
InPut;
```

```
Main;
```

```
Readln;
```

```
End.
```


(Lời giải của bạn Nguyễn Cao Thắng - Lớp 12A2 chuyên Vĩnh Phúc - tỉnh Vĩnh Phúc)

Bài 71/2001 - Thực hiện phép nhân

```

Program Thuc_hien_phep_nhan;
Uses Crt;
Type so = 0..9;
Var a,b,c,d: string;
    can,i: byte;
Procedure Nhap;
Begin
    Clrscr;
    Write('Nhap so a : '); Readln(a);
    Write('Nhap so b : '); Readln(b);
    Writeln('Phep nhan a va b : ');
    can:=length(a)+length(b)+1;
    Writeln(a:can);
    Writeln('X');
    Writeln(b:can);
    For i:=1 to can do Write('-');
    Writeln;
End;
Procedure Nhan(a: string; k: so);
Var nho: so;
    x,i: byte;
Begin
    nho:=0;
    c:="";
    For i:=length(a) downto 1 do
        Begin
            x:=(ord(a[i])-48)*k+nho;
            nho:=x div 10;
            c:=chr((x mod 10)+48)+c;
        End;
    If nho>0 then c:=chr(nho+48)+c;
    Writeln(c:can);
    can:=can-1;
End;
Procedure Cong(var c,d: string; z:byte);
Var nho: so;
    x,i: byte;
Begin
    for i:=1 to length(b)-z do c:=c+'0';
    If length(c) > length(d) then
        For i:=1 to length(c)-length(d) do d:='0'+d
    Else
        For i:=1 to length(d)-length(c) do c:='0'+c;
    nho:=0;
    For i:=length(d) downto 1 do
        Begin
            x:=ord(d[i])+ord(c[i])-96+nho;

```

```

    d[i]:=chr((x mod 10)+48);
    nho:=x div 10;
  End;
  If nho>0 then d:='1'+d;
End;
Begin
  Nhap;
  d:='';
  For i:=length(b) downto 1 do
    Begin
      Nhan(a,ord(b[i])-48);
      Cong(c,d,i);
    End;
  can:=length(a)+length(b)+1;
  For i:=1 to can do Write('-');
  Writeln;
  Writeln(d:can);
  Readln;
End.

```

(Lời giải của bạn Đặng Trung Thành - PTH Nguyễn Du - **Buôn Mê Thuật**)

Bài 72/2001 - Biến đổi trên lưới số

```

const Inp ='bai72.inp';
      Out ='bai72.out' ;
      maxn=100;
Var dem, n, i, j, d:integer; f:text;
    a:array[0..maxn+1,0..maxn+1] of Boolean;
Procedure Init;
Var t:integer;
Begin
  Fillchar(a, Sizeof(a), true);
  Assign(f, inp); reset(f);
  dem:=0;
  Readln(f, n);
  for i:= 1 to n do
    for j:=1 to n do
      begin
        read(f, t);
        If t=1 then a[i,j]:=true else begin a[i,j]:=false;inc(dem); end;
        If j=n then readln(f);
      end;
  Close(f);
End;
Procedure Solve1;
Begin
  for i:=1 to n do
    for j:=1 to n do
      begin
        If not a[i,j] then
          begin

```

```

    a[i,j]:= not (a[i,j-1] xor a[i,j+1] xor a[i-1,j] xor a[i+1,j]);
    If a[i,j] then begin dec(dem);writeln(f,i,' ',j) end
  end;
end;
End;
Procedure Solve2;
Begin
  for i:=1 to n do
    for j:=1 to n do
      If not a[i,j] then
        begin
          If i >1 then
            begin
              a[i-1,j]:=false;
              inc(dem);
              writeln(f, i-1, ' ', j);
            end
          else
            If i <n then
              begin
                a[i+1,j]:=false;
                inc(dem);
                writeln(f, i+1, ' ', j);
              end
            else
              If j >1 then
                begin
                  a[i,j-1]:=false;
                  inc(dem);
                  writeln(f, i, ' ', j-1);
                end
              else
                begin a[i,j+1]:=false; inc(dem); writeln(f, i, ' ', j+1) end;
        end;
    end;
  end;
End;
BEGIN
  Init;
  Assign(f,out); rewrite(f);
  While dem >0 do
    begin
      writeln(dem); d:=dem; solve1;
      If (d=dem) and (dem >0) then solve2;
    end; Close(f);
END.

```

(Lời giải của bạn Nguyễn Chí Thức - khối PTCTT - ĐHSPT - Hà Nội)

Bài 73/2001 - Bài toán chuỗi số

(Dành cho học sinh Tiểu học và THCS)

Hai số cuối là 59 và 65.

Giải thích: Chuỗi số được tạo ra từ việc cộng các số nguyên tố (ở hàng trên) với các số không phải là nguyên tố (hàng dưới), cụ thể như sau:

+	1	2	3	5	7	11	13	17	19	23	29	31	37	41
	4	6	8	9	10	12	14	15	16	18	20	21	22	24
	5	8	11	14	17	23	27	32	35	41	49	52	59	65

Bài 74/2001 - Hai hàng số kỳ ảo

(Dành cho học sinh THCS và PTTH)

Tổng các số từ 1 đến $2n$: $1 + 2 + \dots + 2n = (2n \cdot (2n+1))/2 = n \cdot (2n+1)$.

Do đó, để hai hàng có tổng bằng nhau thì tổng của mỗi hàng phải là $(n \cdot (2n+1))/2$, như vậy n phải là số chẵn thì mới tồn tại hai hàng số kỳ ảo.

Tổng của n cột bằng nhau nên tổng của mỗi cột sẽ là $2n+1$.

ứng với một số $A[i]$ ($A[i] = 1, 2, \dots, 2n$) chỉ tồn tại duy nhất một số $B[i] = 2n - (A[i] - 1)$

sao cho: $A[i] + B[i] = 2n + 1$;

Toàn bộ chương trình lời giải:

Program bai74;

uses crt;

var n:byte;

 a:array[1..100]of 0..1;

 th:array[0..50]of byte;

 ok:boolean;

 s:integer;

Procedure xet;

var i,j,tong:integer;

 duoc:boolean;

Begin

tong:=0;

for j:=1 to n do tong:=tong+th[j];

if tong=s div 2 then

begin

 duoc:=true;

 for j:=1 to n-1 do

 for i:=j+1 to n do

 if th[j]+th[i]=(s div n) then duoc:=false;

 if duoc then

 begin

 for i:=1 to n do write(th[i]:3);

 writeln;

 for i:=1 to n do write(((s div n)-th[i]):3);

 ok:=true;

 end;

 end;

end;

Procedure try(i:byte);

var j:byte;

Begin

 if i>n then xet

 else if not ok then

```

    for j:=th[i-1]+1 to 2*n do
    begin
        th[i]:=j;
        try(i+1);
    end;
End;
Procedure xuli;
var i:byte;
Begin
    th[0]:=0;
    ok:=false;
    s:=n*(2*n)+1;
    try(1);
    if ok=false then write('Khong the sap xep');
End;
BEGIN
    clrscr;
    write('Nhap n:');readln(n);
    if n mod 2 =1 then writeln('Khong the sap xep')
    else xuli;
    readln;
END.

```

(Lời giải của bạn Hoàng Phương Nhi - PTTH chuyên Lý Tự Trọng - **Cần Thơ**)

Nhân xét: Cách làm của bạn Hoàng Phương Nhi - PTTH chuyên Lý Tự Trọng - **Cần Thơ** dùng thuật toán duyệt nên chạy không được lớn. Với $N = 20$ thì chương trình chạy rất lâu, nếu N lớn hơn nữa thì không thể ra được kết quả. Bạn có thể cải tiến chương trình này bằng cách kiểm tra các điều kiện ngay trong quá trình duyệt để giảm bớt thời gian duyệt.

Cách làm khác dùng thuật toán chia kẹo chạy rất nhanh với $N < 35$.

Tổng các số từ 1 đến $2n$: $1 + 2 + \dots + 2n = (2n*(2n+1))/2 = n*(2n+1)$.

Do đó, để hai hàng có tổng bằng nhau thì tổng của mỗi hàng phải là $(n*(2n+1))/2$, như vậy n phải là số chẵn thì mới tồn tại hai hàng số kì ảo.

Tổng của n cột bằng nhau nên tổng của mỗi cột sẽ là: $2n+1$.

ứng với một số $A[i]$ ($A[i] = 1, 2, \dots, 2n$) chỉ tồn tại duy nhất một số $B[i] = 2n - (A[i] - 1)$ sao cho: $A[i] + B[i] = 2n + 1$

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

{SM 16384,0,655360}

uses crt;

```
const max =35;
```

```
    fi = 'bai74.inp';
```

```
    fo = 'bai74.out';
```

```
var    d : array[0..max*(2*max+1) div 2] of byte;
```

```
    tr : array[1..max,0..max*(2*max+1) div 2] of byte;
```

```
    kq : array[1..max] of integer;
```

```
    n,sum : integer;
```

```
    ok : boolean;
```

```

procedure docf;
var f :text;
begin
  ok:=false;
  assign(f,fi);
  reset(f);
  read(f,n);
  close(f);
end;

procedure lam;
var i,j :integer;
begin
  sum:=n*(2*n+1) div 2;
  fillchar(d,sizeof(d),0);
  fillchar(tr,sizeof(tr),0);
  d[0]:=1;
  for i:=1 to n do
    begin
      for j:=sum-i downto 0 do
        if d[j]=1 then
          begin
            d[j+i]:=2;
            tr[i,j+i]:=1;
          end;

      for j:=sum-(2*n+1-i) downto 0 do
        if d[j]=1 then
          begin
            d[j+2*n+1-i]:=2;
            tr[i,j+2*n+1-i]:=2;
          end;
      for j:=0 to sum do
        if d[j]>0 then dec(d[j]);
      end;
      ok:=(d[sum]=1);
    end;

procedure ghif;
var f :text;
    i,j :integer;
begin
  assign(f,fo);
  rewrite(f);
  if ok=false then write(f,'No solution')
  else
    begin
      i:=sum;j:=n;
      while i>0 do
        begin

```

```

    if tr[j,i]=1 then kq[j]:=j else kq[j]:=2*n+1-j;
    i:=i-kq[j];
    dec(j);
  end;
  for j:=1 to n do write(f,kq[j]:6);
  writeln(f);
  for j:=1 to n do write(f,(2*n+1-kq[j]):6);
  end;
  close(f);
end;

```

```

BEGIN
  docf;
  if n mod 2=0 then lam;
  ghif;
END.

```

Bài 75/2001 - Trò chơi Tích - Tắc vuông

(Dành cho học sinh THCS và PTTH)

(* Thuật toán:

Chia bàn cờ làm 4 hướng: Đông, Tây, Nam, Bắc. Ta có cách đi sau:

- i) Luôn đi theo ô liền cạnh với ô trước
- ii) Đi theo hướng không bị chặn. Ví dụ: ô bước 1 nếu bị chặn ở hướng Đông thì đi theo hướng ngược lại là hướng Tây. Đi theo hướng Tây đến khi hướng Tây bị chặn thì đi theo hướng Bắc hoặc Nam.

Trong khi đi ta luôn để ý 2 điều kiện sau:

1. Nếu có 3 ô đã lấp thành 3 đỉnh của 1 hình vuông mà ô thứ 4 chưa bị đi thì ta sẽ đi ô thứ 4 và giành được thắng lợi.
2. Nếu có $2k+1$ ($k \geq 1$) ô liền cạnh liên tiếp thì kiểm tra có thể giành thắng lợi bằng nước đi không? Nước đi là nước ta đánh vào 1 ô nhưng có thể có được 2 hình vuông. ví dụ: có 3 ô (1,1);(1,2);(1,3) thì ta có thể đánh nước đi bằng cách đánh vào ô (2,2) như vậy ta có khả năng hình thành 2 ô vuông. Nhưng sau 1 nước đi đôi thì chỉ duy nhất chặn được 1 ô vuông, ta có thể đánh nước tiếp theo để hình thành ô vuông còn lại và giành được thắng lợi.

Bảng cách đánh như vậy bạn có thể chiến thắng trong vòng tối đa là 10 nước.*)

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

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

```
CONST Min=-50;
```

```
    Max=50;
```

```
TYPE Ma=Array[Min..Max,Min..Max] of char;
```

```
    diem= Record
```

```
        hg,cot:Integer;
```

```
    End;
```

```
    Qu=Array[1..Max] of diem;
```

```
VAR dmay,dng,dc1,dc2:diem;
```

```
    hgd:Integer; (*1:B ; 2:D ; -1:N ; -2:T*)
```

```
    fin,ok:Boolean;
```

```
    A:Ma;
```

```
    Q,Qc:Qu;
```

```
    dlt,dq,cq:Integer;
```

```

Procedure HienA(hgd,hgc,cotd,cotc:Integer);
Var i,j:Integer;
Begin
  For i:=hgd to hgc do
    Begin
      For j:=cotd to cotc do Write(A[i,j], ' ');
      Writeln;
    End;
  End;
Procedure finish(d:diem);
Begin
  A[d.hg,d.cot]:='x';
  HienA(-10,10,-10,10);
  Writeln('Ban da thua! An ENTER de ket thuc chuong trinh');
  Readln;
  Halt;
End;
Procedure Init;
Begin
  Fillchar(A,sizeof(A),' ');
  fin:=false;
  Writeln('Gia thiet bang o vuong co: 101 hang (-50 -> 50)');
  Writeln('          101 cot (-50 -> 50)');
  Writeln('Gia thiet may luon di nuoc dau tien tai o co toa do (0:0)');
  dmay.hg:=0; dmay.cot:=0; A[dmay.hg,dmay.cot]:='X';
  HienA(-10,10,-10,10);
  dlt:=1;
End;
Procedure Sinh(d1:diem; Var d2:diem; hgdi,k:integer);
Var h,c:Integer;
Begin
  h:=d1.hg; c:=d1.cot;
  Case hgdi of
    1: Dec(h,k);
    2: Inc(c,k);
   -1: Inc(h,k);
   -2: Dec(c,k);
  End;
  d2.hg:=h; d2.cot:=c;
End;
Function kt(Var d1,d2:diem):boolean;
Var g1,g2:diem;
    k,p:integer;
Begin
  kt:=true;
  k:=(dlt-1) div 2;
  p:=2 div abs(hgdi);
  sinh(dmay,g1,-hgdi,k);
  sinh(dmay,g2,-hgdi,2*k);
  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;
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, yr): 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<yr) and (ye<yr));
  lg2:=(xs>xl) and (xs<xr) and (ys<yt)and (ye<yt) and (ys>yr) and (ye>yr);
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<yr));
  lg2:=(xs>xl) and (xe>xl) and (xs<xr)and (xe<xr) and (ys<yt) and (ys>yr);
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>=yr)then cat:= true;
y:= a*xr+b;
if(y<=yt)and(y>=yr)then cat:=true;
x:=(yt-b)/a;
if (x>=xl)and (x<=xr)then cat:=true;
x:=(yr-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, inp); 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 90^0 bảng ta có thể có các bảng cũng thoả mãn.

{S,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 :mglc;
    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.

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

```
{SM 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 kiểmtra;
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 kiểmtra
      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
  chrscr;
  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:

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

```
{SM 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;
  Fillchar(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]:=l;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;
        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*(l 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;
        End;
      Close(f);
    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)

Nhận xét: Bài làm của bạn Vũ Lê An phần 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)

```
{S+,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;

```

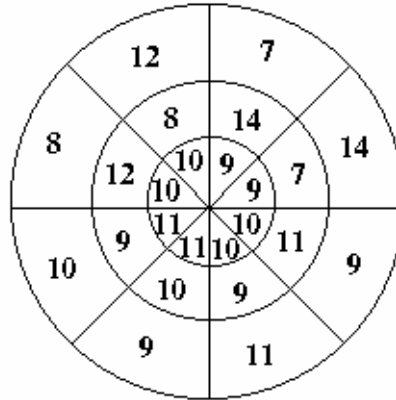


```
BEGIN
  docf;
  lam;
  ghif;
END.
```

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?

{S,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:=sqr(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.

{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   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
      fillechar(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 - ĐHSPT 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 lặ

(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;
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(l-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 phân 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 phân 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 phân 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 Idonotknow;
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,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
      wastingtime;
      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;
  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
    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 =====