

1. Ghép số.

Cho N số nguyên dương a_1, a_1, \dots, a_n ($1 < n \leq 100$), mỗi số không vượt quá 10^9 . Từ các số này người ta tạo ra số nguyên bằng cách ghép liên tiếp tất cả các số đã cho tạo thành một số. Hãy xác định số lớn nhất có thể tạo ra khi ghép tất cả các số đã cho tạo thành số mới.

Dữ liệu vào file NUMJOIN.INP: Dòng đầu là số n ; dòng tiếp theo là n số nguyên dương a_1, a_1, \dots, a_n .

Kết quả ra file NUMJOIN.OUT: Gồm một dòng là số lớn nhất đã ghép được.

Ví dụ:

NUMJOIN.INP	NUMJOIN.OUT
4	9056124123
123 124 56 90	

2. Tìm kiếm nhị phân

Cho dãy số A gồm n phần tử ($1 < n < 10^9$) nguyên có giá trị tuyệt đối nhỏ hơn 10^9 và giá trị x nguyên có giá trị tuyệt đối nhỏ hơn 10^9 . Hãy kiểm tra xem x có trong A hay không? Dữ liệu vào file TKNP.INP: Dòng đầu số n và số x ; dòng thứ hai là các giá trị của A .

Kết quả ra file TKNP.OUT: Ghi vị trí xuất hiện tương ứng với câu trả lời theo yêu cầu trên; nếu không có giá trị x thì ghi 0;

TKNP.INP	TKNP.OUT
10 -2	4
-20 -19 -12 -2 7 17 23 29 34 41	

3. Nối xích

Người ta có N đoạn dây xích ($N \leq 20000$), mỗi đoạn dây xích là một chuỗi các mắt xích được nối nhau. Các đoạn dây xích này tách rời nhau. Một đoạn xích có không quá 20000 mắt xích.

Bằng cách cắt ra một mắt xích, sau đó hàn lại, ta có thể nối hai dây xích thành một đoạn. Thời gian để cắt và hàn mỗi mắt xích là 1 đơn vị thời gian và được xem là bằng nhau với mọi mắt xích.

Nhiệm vụ của bạn là phải nối chúng lại thành một đoạn xích duy nhất với thời gian ít nhất (Hay số mắt xích bị cắt và hàn là ít nhất).

Noixich.inp	Noixich.out
3	2
4 7	

Dữ liệu vào file Noixich.inp: Dòng đầu là số N số đoạn xích; Những dòng tiếp theo ghi N số nguyên dương, số thứ i là số mắt xích có trong đoạn xích thứ i.

Kết quả ra file Noixich.out: Ghi ra một số duy nhất là số đơn vị thời gian cần để nối N đoạn xích đã cho.

6	
6	3
1 4 3	
5 1 6	

4. Kết quả giải bóng đá.

Bảng kết quả của một giải vô địch bóng đá thi đấu 1 lượt trận được cho bởi ma trận vuông cấp n ($1 < n < 100$), trong đó tất cả các phần tử thuộc đường chéo chính bằng 0, các phần tử khác bằng 3, 1 hoặc 0 (đó là số điểm của trận đấu: 3 - thắng, 1 - hoà, 0 - thua). Dòng thứ i là kết quả thi đấu của đội thứ i.

a. In ra các đội có tổng điểm lớn nhất với các thông tin số thứ tự đội, tổng số điểm, số trận thắng, số trận hoà và số trận thua?

b. In ra thứ tự các đội có số trận thắng nhiều hơn số trận thua? Nếu không có in ra số 0.

c. In ra thứ tự các đội không thua trận nào? Nếu không có in ra số 0.

Dữ liệu vào file 'KQBONGDA.INP': Dòng đầu là số n; n dòng tiếp theo là bảng kết quả bóng đá.

Kết quả ra file 'KQBONGDA.OUT': Mỗi kết quả ghi trên một dòng theo thứ tự các phần đã cho.

Ví dụ:

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

5. Trò chơi.

Ngoài đam mê về lập trình, Tuấn Anh rất thích chơi game, nhất là game Line98 huyền thoại. Ở đó, có 4 quả bóng cùng màu sẽ nổ khi nó đứng cạnh nhau theo chiều dọc hoặc chiều ngang hoặc theo đường chéo theo một đường thẳng. Với khả năng lập trình của mình, Tuấn Anh muốn phát triển game này lên với cách chơi mới.

Cũng với hình chữ nhật kích thước mxn được chia thành lưới ô vuông. Ở mỗi ô có một quả bóng mà trên nó có ghi một số nguyên. Người chơi sẽ được cầm một chiếc búa, mỗi lần đập vào quả bóng nào thì quả bóng đó vỡ và tất cả các quả bóng khác có số nguyên bằng

số nguyên ở quả bóng đầu tiên bị đập vào thì cũng vỡ theo. Mỗi ván chơi, một người chơi được đập búa tối đa K lần. Tất nhiên, khi các quả bóng đã vỡ hết thì không phải đập búa nữa.

Ví dụ, với các quả bóng như hình bên. $M=3$, $n=6$, $k=2$ thì người chơi có thể chơi như sau:

```

1  2  1  3  1  1
2  1  4  1  4  3
1  2  1  4  1  1

```

Dùng búa đập vào quả 1 và quả 3 sẽ có 12 quả bóng vỡ.

Dùng búa đập vào quả 1 và quả 4 sẽ có 13 quả bóng vỡ.

Yêu cầu: Hãy giúp Tuấn Anh tìm cách đập bóng không quá K lần sao cho vỡ được nhiều bóng nhất.

Dữ liệu vào: Từ tệp văn bản 'B_LINE.INP' gồm

Dòng 1: Ghi số nguyên dương m , n , k . ($m \leq 300$, $n \leq 300$, $k \leq m*n$)

M dòng tiếp theo, mỗi dòng ghi N số nguyên dương $a[i,j]$ là số ghi trên quả bóng có tọa độ (i,j) . $a[i,j] < 100000$.

Kết quả: Ghi ra tệp văn bản 'B_LINE.OUT' số lượng bóng vỡ nhiều nhất tìm được.

B_LINE.INP	B_LINE.OUT
3 6 2	13
1 2 1 3 1 1	
2 1 4 1 4 3	
1 2 1 4 1 1	

Subtask #1 60% số điểm của câu với $m < 100$; $n < 100$; $a[i,j] < 300$.

Subtask#2 40% số điểm của câu với $m > 290$; $n > 290$; $0 < a[i,j] < 100000$

6. Số dư.

Cho dãy A gồm n số nguyên dương $a[1], a[2], \dots, a[n]$. Hãy tìm số nguyên dương d lớn nhất sao cho $a[1] \bmod d = a[2] \bmod d = a[3] \bmod d = \dots = a[n] \bmod d$.

Dữ liệu vào file **remain.inp**:

- Dòng đầu chứa số nguyên dương n ($n \leq 500$).
- Dòng thứ hai chứa n số nguyên dương $a[1], a[2], \dots, a[n]$ ($a[i]$ đôi một khác nhau, $a[i] \leq 10^{16}$).

Dữ liệu ra file **remain.out** :

- Chứa một số nguyên dương d duy nhất là kết quả bài toán.

Ví dụ:

remain.inp	remain.out
3	2
3 7 9	

7. Hoán vị xâu

Cho xâu S (độ dài tối đa không vượt quá 10) chỉ gồm các kí tự 'A'-'Z' (các kí tự trong xâu không nhất thiết phải khác nhau). Hãy liệt kê tất cả các hoán vị khác nhau của xâu S.

Dữ liệu vào file **HVXAU.INP**: Chứa duy nhất xâu S.

Kết quả ra file **HVXAU.OUT**: Mỗi dòng ghi một hoán vị đã tạo được.

Ví dụ:

HVXAU.INP	HVXAU.OUT
ABA	AAB
	ABA
	BAA

8. Phân tích số

Cho số N ($1 \leq N \leq 300$) nguyên dương, hãy phân tích N thành các số nguyên tố khác nhau có tổng bằng N và tìm ra cách phân tích sao cho tích của các số đó là lớn nhất?

Dữ liệu vào file **PHANTICH.INP**: Chỉ một dòng chứa số N.

Dữ liệu ra file **PHANTICH.OUT**: Nếu phân tích được thì dòng đầu ghi ra số lượng các cách phân tích và dòng thứ hai ghi ra tích lớn nhất trong các cách phân tích, còn nếu không phân tích được thì ghi ra 0.

Ví dụ:

PHANTICH.INP	PHANTICH.OUT
10	2
	30

9. Tạo ngoặc đúng

Cho số nguyên dương n ($n \leq 10$), hãy liệt kê tất cả các cách khác nhau đặt n dấu ngoặc mở và n dấu ngoặc đóng đúng đắn?

Dữ liệu vào file **DATNGOAC.INP**: Chứa duy nhất số n.

Kết quả ra file **DATNGOAC.OUT**: Mỗi dòng ghi một cách đặt ngoặc đúng đã tạo được.

Ví dụ:

DATNGOAC.INP	DATNGOAC.OUT
3	((()))
	((()()))

	(())()
	()(())
	()()()

10. Chọn số

Cho ma trận A có kích thước $n \times n$ ($1 < n \leq 10$) gồm các số nguyên 2 byte. Hãy chọn ra n số sao cho mỗi hàng, mỗi cột có đúng một số được chọn đồng thời tổng các số được chọn là lớn nhất.

Dữ liệu vào file **CHONSO.INP**: Dòng đầu là số n; n dòng tiếp theo biểu diễn ma trận A.

Kết quả ra file **CHONSO.OUT**: Ghi ra tổng lớn nhất tìm được.

Ví dụ:

CHONSO.INP	CHONSO.OUT
3	25
7 9 6	
3 5 8	
8 6 9	

11. Đặt phép toán

Cho chuỗi S (chỉ gồm các ký tự '0' đến '9', độ dài nhỏ hơn 10) và số nguyên M, hãy tìm tất cả các cách chèn vào S các dấu '+' hoặc '-' để thu được số M cho trước?

Dữ liệu vào file **DATPT.INP**: Dòng đầu là chuỗi S, dòng thứ hai là số M.

Kết quả ra file **DATPT.OUT**: Nếu có các cách ghi số lượng các phép toán, còn ngược lại ghi số 0.

Ví dụ:

DATPT.INP	DATPT.OUT	Giải thích
1234	4	1+2+3
6		1-2+3+4
		-1+3+4
		2+4

12. Điền phép toán

Cho dãy số A gồm n phần tử ($n \leq 15$), hãy điền các phép toán '+', '-', '*', '/' vào giữa các số của dãy A để thu được kết quả gần số M cho trước nhất?

Dữ liệu vào file **DIENPT.INP**: Dòng đầu là 2 số M, n; dòng thứ hai là các phần tử của dãy A.

Kết quả ra file **DIENPT.OUT**: Ghi ra một biểu thức gần với M nhất (biểu thức này lớn nhất là bằng M).

Ví dụ:

DIENPT.INP	DIENPT.OUT
10 5	1+2+3*4-5
1 2 3 4 5	