

## ĐỀ LUYỆN TẬP KỲ THI ICPC 2024 – TUẦN 6

Ngày thi: Chủ nhật, ngày 25 tháng 08 năm 2024. Đề thi gồm có **10** bài, **07** trang

### BÀI A. BIỂU DIỄN SỐ BẰNG QUE DIÊM

Chỉ với các que diêm chúng ta cũng có thể tạo ra được những biểu thức số học. Trong bài toán này chúng ta sẽ quan tâm tới các biểu thức số học như sau:

- Một số được mô tả bằng đúng số lượng que diêm
- Phép cộng (+) được mô tả bằng 2 que diêm
- Phép nhân (x) cũng được mô tả bằng 2 que diêm

Chú ý rằng phép nhân có ưu tiên thực hiện hơn phép cộng.

Ví dụ: |||||x|||| có kết quả là 35 hoặc |||||x||||+| có kết quả là 37

**Yêu cầu:** Cho số  $N$ , tìm biểu thức biểu diễn bằng que diêm sử dụng ít que diêm nhất mà kết quả bằng đúng  $N$ .

#### Input:

Dòng đầu tiên là số lượng bộ test  $T$  ( $T \leq 6000$ ).

Mỗi test trên một dòng chứa một số nguyên dương  $N$  ( $N < 5001$ ).

#### Output:

Với mỗi test, hãy in ra số lượng que diêm dung ít nhất cần sử dụng.

#### Test ví dụ:

Input	Output
3	14
35	17
37	21
53	

Giới hạn thời gian: 2s

Giới hạn bộ nhớ: 524288 Kb

### BÀI B. XÂU AC

Cho một chuỗi AC được xây dựng một cách đệ quy vô hạn như sau:

Chuỗi  $S(0)$  là chuỗi đơn có 3 ký tự ACC. Chuỗi  $S(k)$  được xây dựng bằng công thức:

$S(k) = S(k-1) + AC..C + S(k-1)$ , trong đó chuỗi ở giữa  $AC..C$  có  $k+2$  ký tự C.

$S(0) = ACC$ ;

$S(1) = ACCACCCACC$

$S(2) = ACCACCCACCACCCACCACCCACC$

.....

Với cách này sẽ xây dựng một chuỗi với 2 ký tự A, C có độ dài vô hạn. Nhiệm vụ của bạn là hãy xác định xem ký tự thứ  $N$  là ký tự nào?

#### Input

Gồm một số nguyên  $N$  duy nhất ( $1 \leq N \leq 10^9$ ).

#### Output

In ra ký tự thứ  $N$  tìm được.

**Test ví dụ**

Input	Output
11	a
1	a
3	c

Giới hạn thời gian: 2s

Giới hạn bộ nhớ: 524288 Kb

**BÀI C. HOÁN VỊ**

Cho dãy số  $P[1], P[2], \dots, P[N]$  là một hoán vị của  $1, 2, \dots, N$ . Nhiệm vụ của bạn là hãy tìm số vị trí  $i$  thỏa mãn tính chất: với mọi  $j$  ( $1 \leq j \leq i$ ) thì  $P[i] \leq P[j]$ .

**Input:**

Dòng đầu tiên là số nguyên  $N$  ( $1 \leq N \leq 200\,000$ ).

Dòng tiếp theo là một hoán vị của  $1, 2, \dots, N$ .

**Output:**

In ra số lượng vị trí  $i$  thỏa mãn yêu cầu.

**Test ví dụ:**

Input	Output
4 4 2 5 1 3	3
5 5 4 3 2 1	5
3 1 2 3	1

Giải thích test 1:  $i = 1, 2, 4$  thỏa mãn, nhưng 3 và 5 thì không thỏa mãn.

Giới hạn thời gian: 2s

Giới hạn bộ nhớ: 524288 Kb

**BÀI D. NẤY MẦM**

Trong một vườn ươm đang có  $N$  hạt mầm được gieo tại các ô có vị trí  $(X[i], Y[i])$ . Mỗi loại hạt mầm sẽ có chỉ số tăng trưởng khác nhau, được kí hiệu là  $C[i]$ .

Alice đang cần xác định một vị trí tối ưu nào đó để đặt nguồn nước, rồi nước từ vị trí này sẽ lan sang các vùng lân cận, cung cấp dinh dưỡng cho các hạt mầm. Giả sử nguồn nước được đặt vào vị trí  $(A, B)$ , thì hạt mầm thứ  $i$  sẽ nảy mầm sau  $C[i] * \sqrt{(A - X[i])^2 + (B - Y[i])^2}$  đơn vị thời gian.

Mục tiêu của Alice là cần có được  $K$  hạt giống đã nảy mầm và mang chúng đến lớp cho cô giáo chấm điểm. Các bạn hãy xác định xem Alice cần sử dụng ít nhất bao nhiêu đơn vị thời gian

**Input:**

Dòng đầu tiên gồm hai số nguyên dương  $N$  và  $K$  ( $1 \leq K \leq N \leq 80$ ).

$N$  dòng tiếp theo, mỗi dòng gồm 3 số nguyên  $X[i], Y[i], C[i]$ .

Input đảm bảo các tọa độ là riêng biệt. ( $-1000 \leq X[i], Y[i] \leq 1000, 1 \leq C[i] \leq 100$ ).

**Output:**

In ra một số thực là khoảng thời gian ít nhất mà Alice cần chờ đợi nếu như anh ấy chọn được vị trí đặt nguồn nước tối ưu.

**Test ví dụ:**

Input	Output
2 2 1000 1000 1 -1000 -1000 1	1414.213562
4 3 0 0 3 -1 0 3 1 0 2 1 1 30	2.400000

Giải thích test 2: Đặt nguồn nước ở tọa độ  $(-0.2, 0)$ , hạt giống 1, 2 và 3 sẽ nảy mầm được sau 2.4 giây.

Giới hạn thời gian: 2s

Giới hạn bộ nhớ: 524288 Kb

**BÀI E. XÂU ANAGRAM**

Một chuỗi A được gọi là chuỗi Anagram của chuỗi B nếu như ta có thể thu được chuỗi A bằng một số phép đổi chỗ các ký tự trên chuỗi B.

Ví dụ: “listen” là một anagram của “silent”.

Cho N chuỗi phân biệt  $S[1], S[2], \dots, S[N]$ . Nhiệm vụ của bạn là hãy đếm số lượng cặp chỉ số  $(i, j)$  thỏa mãn  $S[i]$  là chuỗi Anagram của  $S[j]$ .

**Input:**

Dòng đầu tiên là số lượng chuỗi N ( $1 \leq N \leq 100000$ ).

N dòng tiếp theo, mỗi dòng gồm một chuỗi có đúng 10 ký tự, tất cả các ký tự đều là chữ cái thường.

**Output:**

In ra một số nguyên duy nhất là đáp số của bài toán.

**Test ví dụ:**

Input	Output
3 listenaaaa aaaasilent bbbbbbbbbb	1
2 aaaaaaaaaa bbbbbbbbbb	0
5 abaaaaaaaa ccccccxccc aaaaaaaaaba ccccccccccx aaaabaaaaa	4

## BÀI F. DÃY FIBONACCI TUẦN HOÀN

Dãy số Fibonacci được định nghĩa như sau:

- $T(0) = 0, T(1) = 1$
- $T(n) = T(n-1) + T(n-2)$ , với  $n > 1$ .

Như vậy, các số đầu tiên của dãy sẽ là: 0 1 1 2 3 5 8 13 21 34 ...

Bây giờ ta thay đổi biểu diễn của dãy bằng cách chia lấy phần dư các số trong dãy cho một số B cho trước. Khi đó người ta nhận thấy các biểu diễn của dãy số Fibonacci thành một dãy mới và dãy này sẽ gồm các đoạn lặp đi lặp lại. Ví dụ với  $B = 2, 3, 4, 5$  ta có bảng sau:

Số B	Dãy Fibonacci khi đã chia lấy phần dư với B	Độ dài đoạn lặp ngắn nhất
2	0 1 1 0 1 1 0 1 1 0 1 1 0 1 1 0 1 1 0 1	3
3	0 1 1 2 0 2 2 1 0 1 1 2 0 2 2 1 0 1 1 2 0 2 2	8
4	0 1 1 2 3 1 0 1 1 2 3 1 0 1 1 2 3 1 0 1 1 2 3	6
5	0 1 1 2 3 0 3 3 1 4 0 4 4 3 2 0 2 2 4 1 0 1 1	20

Cho trước số nguyên B. Hãy xác định chu kì tuần hoàn nhỏ nhất tìm được.

### Input:

Dòng đầu tiên là số lượng bộ test T ( $1 \leq T \leq 50$ ).

Mỗi test gồm một số nguyên B ( $2 \leq B \leq 2^{24}$ ).

### Output:

In ra một số nguyên duy nhất là đáp số của bài toán.

### Test ví dụ:

Input	Output
6	3
2	8
3	6
4	20
5	24
6	25165824
16777216	

Giới hạn thời gian: 2s

Giới hạn bộ nhớ: 524288 Kb

## BÀI G. PHƯƠNG TRÌNH

Xét phương trình có dạng  $(A_1x^2 + B_1x + C_1)(A_2x^2 + B_2x + C_2) \dots (A_nx^2 + B_nx + C_n) = 0$

Ví dụ, phương trình  $(2x^2 - 4x + 2)(3x^2 + 2x + 1) = 0$  có duy nhất một nghiệm  $x_1 = 1$

phương trình  $(2x^2 + 5x + 3)(3x^2 + 4x + 1) = 0$  có 3 nghiệm  $x_1 = \frac{-3}{2}, x_2 = -1, x_3 = \frac{-1}{3}$

còn phương trình  $(0x^2 + 1x - 1)(0x^2 + 1x - 2)(0x^2 + 2x - 2) = 0$  có 2 nghiệm  $x_1 = 1, x_2 = 2$ .

**Yêu cầu:** Cho n bộ số nguyên  $(A_i, B_i, C_i)$  ( $i = 1, 2, \dots, n$ ) đếm số nghiệm của phương trình  $(A_1x^2 + B_1x + C_1)(A_2x^2 + B_2x + C_2) \dots (A_nx^2 + B_nx + C_n) = 0$ .

### Input

- Dòng thứ nhất chứa số nguyên dương n ( $n < 1000$ )
- Tiếp theo là n dòng, dòng thứ i chứa ba số nguyên  $(A_i, B_i, C_i)$  ( $i = 1, 2, \dots, n$ )
- Các số có giá trị tuyệt đối không quá 1000

## Output

Ghi một số nguyên duy nhất là số nghiệm của phương trình. Nếu phương trình có vô số nghiệm ghi -1.

Input	Output	Input	Output
1 2 5 3	2	2 0 1 1 1 2 1	1

Giới hạn thời gian: 1s

Giới hạn bộ nhớ: 524288 Kb

## BÀI H. DÃY SỐ ĐẸP

Cho dãy số A gồm N số nguyên không âm. Dãy số B gồm N phần tử được tạo từ dãy số A bằng cách sau:

- Nếu  $A[i] = 0$  thì  $B[i] = X$ ,
- Ngược lại  $B[i] = A[i]$  với  $1 \leq i \leq N$

Dãy số B được gọi là đẹp nếu tất cả phần tử của B có giá trị lớn hơn 1 và ước số chung lớn nhất của hai phần tử liên tiếp  $B[i]$  và  $B[i + 1]$  bằng 1.

Hãy tìm số nguyên dương X nhỏ nhất có thể để dãy số B là dãy số đẹp.

**Input:** Dòng đầu tiên ghi N ( $2 \leq N \leq 10^6$ ).

Dòng thứ hai chứa N phần tử của dãy số A ( $0 \leq A[i] \leq 2 \cdot 10^7$ ).

**Output:** Ghi ra X. Hoặc nếu không thể tạo ra dãy số B đẹp thì ghi ra -1

**Ví dụ:**

Input	Output
5 20 0 15 0 30	7

Giới hạn thời gian: 3s

Giới hạn bộ nhớ: 524288 Kb

## BÀI I. THU HOẠCH TÁO

Một Khu vườn rộng N x M mét vuông được chia thành N x M ô đất mỗi ô kích thước 1 mét x 1 mét. Khu vườn này có một vài ô đất trồng cây táo, Lộc sẽ đặt một Robot ở một ô đất bất kỳ sau đó điều khiển Robot di chuyển đến từng ô đất có cây táo để thu hoạch táo. Vì pin của Robot có hạn nên Lộc muốn biết để Robot có thể đến mỗi cây táo ít nhất một lần thì Robot cần ít nhất bao nhiêu năng lượng?

Biết rằng khi Robot đang ở ô đất (x, y) thì Robot có thể di chuyển đến một trong bốn ô kề cạnh (x-1, y), (x+1, y), (x, y-1), (x, y+1) tốn 1 đơn vị năng lượng. Khu vườn có thể có các ô đất chứa tảng đá, Robot không thể di chuyển vào ô đất này. Điều đặc biệt nhất ở khu vườn là có thể có Cánh Cửa Thần Kỳ được đặt ở một số ô đất, đây là những lối tắt xuyên không thời gian nên Robot có thể di chuyển từ Cánh Cửa này đến Cánh Cửa khác mà không tiêu tốn bất cứ năng lượng nào.

### Input

Dòng đầu tiên chứa hai số nguyên N, M ( $1 \leq N, M \leq 1000$ ). N dòng tiếp theo, dòng thứ i chứa một xâu có độ dài M trong đó ký tự thứ j là một trong 4 ký tự sau:

- 'A' - mô tả ô đất (i, j) có một cây táo
- 'D' - mô tả ô đất (i, j) có một Cánh Cửa Thần Kỳ
- '#' - mô tả ô đất (i, j) có một tảng đá
- '.' - mô tả ô đất (i, j) là một ô đất bình thường

Khu vườn có ít nhất 1 cây táo và tối đa 9 cây

**Output:** Ghi ra năng lượng ít nhất Robot cần có để thực hiện thu hoạch táo. Nếu không có cách nào điều khiển Robot để thu hoạch táo thì ghi ra -1.

**Ví dụ**

Input	Output
4 6 D...D DA.#.. ...#.A .A....	5
1 9 AD.A....D	3
2 7 A..#... .A.#..A	-1

*Giải thích:*

Test ví dụ 1:  $(4,2) \rightarrow (3,2) \rightarrow (2,2) \rightarrow (2,1) \rightarrow (1,6) \rightarrow (2,6) \rightarrow (3,6)$   $1+1+1+0+1+1=5$

Test ví dụ 2:  $(1,1) \rightarrow (1,2) \rightarrow (1,3) \rightarrow (1,4)$   $1+1+1=3$

Giới hạn thời gian: 1s

Giới hạn bộ nhớ: 524288 Kb

## BÀI J. LAZER

Trong một mê cung có kích thước  $N \times M$ , mỗi ô đơn vị nhận một trong những giá trị sau:

# một bức tường

. một ô trống

S vị trí bắt đầu

G vị trí kết thúc

<, >, v, ^ mô tả một đèn laze theo hướng mũi tên. Mỗi giây, đèn laze sẽ quay 90 độ theo chiều kim đồng hồ. Tia laze sẽ không thể xuyên qua được bức tường hoặc vị trí đặt một đèn laze khác.

Nhiệm vụ của bạn là hãy di chuyển từ vị trí S đến vị trí G với số bước ít nhất, sao cho không bị phát hiện bởi bất kì chiếc đèn laze nào. Mỗi bước di chuyển tốn chi phí thời gian là 1 s.

Khi đến vị trí kết thúc G, nếu bạn bị đèn laze chiếu, trường hợp này sẽ không hợp lệ.

**Input:**

Dòng đầu tiên là số lượng bộ test  $T$  ( $T \leq 100$ ).

Mỗi test bắt đầu bởi 2 số nguyên dương  $N$  và  $M$  ( $N, M \leq 100$ ).

$N$  dòng tiếp, mỗi dòng gồm một xâu dài  $M$  kí tự, mô tả một hàng của mê cung.

**Output:**

Với mỗi test, in ra đáp án tìm được là số bước di chuyển ít nhất. Nếu không có đáp án, in ra impossible.

**Ví dụ:**

INPUT	OUTPUT
5	
2 5	
##^##	
S...G	
2 5	
##v##	
S...G	6
1 5	4
S..G<	3
1 6	impossible
S...G<	8
4 5	
S....	
.>v..	
.^<..	
....G	

**HẾT**