

# Tập huấn APIO – Buổi 1-2

## DZI

### Vòng 1 | 256 MB | 1s

Cây là đồ thị liên thông  $k$  đỉnh,  $k-1$  cạnh. Bậc của đỉnh là số cạnh nối với một đỉnh. Cho dãy số  $a_1, a_2, \dots, a_n$ . Bạn được biết đó là danh sách bậc các đỉnh của một cây – tuy nhiên, danh sách này bị lỗi: một số số ngẫu nhiên được thêm vào, một số số bị ghi sai. Xóa bỏ một số số và thay đổi một số số để danh sách còn lại là danh sách bậc đỉnh hợp lệ.

Cụ thể, bạn phải sắp xếp lại dãy  $n$  số, xóa bỏ một số số, thay đổi một số số (trong những số còn lại). Với dãy thu được  $d_1, d_2, \dots, d_k$ , bạn phải chỉ ra cây có  $k \geq 2$  đỉnh (đánh số từ 1 đến  $k$ ) sao cho danh sách bậc các đỉnh của cây trùng với dãy  $d$ .

Bạn phải chỉ ra giải pháp với số lượng số thay đổi bé nhất (có thể là 0).

### INPUT

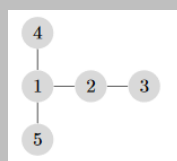
Dòng đầu ghi số  $n$  ( $2 \leq n \leq 10^6$ ). Dòng thứ 2 ghi  $n$  số  $a_1, a_2, \dots, a_n$  ( $1 \leq a_i \leq n-1$ ).

### OUTPUT

Ở dòng đầu tiên, in ra số  $x$  là số lượng số phải thay đổi (phải tối thiểu giá trị  $x$  này). Dòng thứ 2 in ra số đỉnh mới của  $k$  ( $2 \leq k \leq n$ ). Sau đó là  $k$  dòng, mỗi dòng liệt kê 2 đỉnh có nối bởi 1 cạnh  $u_i, v_i$ .

Sample Input	Sample Output
6 2 1 5 3 1 1	0 5 1 2 2 3 1 4 1 5
3 1 2 2	1 3 1 2 2 3

Giải thích ví dụ 1: Loại bỏ số 5, sắp xếp lại dãy số 3, 2, 1, 1, 1. Đồ thị như sau:



Chúng ta xóa 1 số, và không thay đổi số nào ( $x = 0$  ghi trên dòng đầu tiên).

Giải thích ví dụ 2: Sửa 1 số để dãy 1, 2, 2 thành 1, 2, 1. Đồ thị như sau:



# POL

## Vòng 1 | 256 MB | 0.25s

Từ có 3 nguyên âm liền nhau hoặc 3 phụ âm liền nhau là từ khó đọc. (Phụ âm: a, i, e, o, u, y). Cho 1 xâu, đếm xem có bao nhiêu từ khó đọc trong đó. Chú ý: cùng 1 từ ở 2 vị trí khác nhau được coi là khác nhau).

### INPUT

Dòng đầu ghi 1 xâu gồm chữ cái thường trong tiếng Anh có không quá 200000 ký tự.

### OUTPUT

In ra kết quả phải tìm

Sample Input	Sample Output
kobtkka	6
aaa	3

Giải thích ví dụ 1: 6 từ khó có thể là: btk, btka, obtk, kobtk, obtkka, kobtkka

# HER

## Vòng 2. 256 MB. 2-3 sec.

Cho đồ thị có hướng, phi chu trình có n đỉnh và m cạnh. Xóa đi k đỉnh và các cạnh liên quan. Vẽ đẹp của đồ thị còn lại là số đỉnh trên tuyến đường dài nhất đi qua các cạnh liên tiếp trên đồ thị. Hãy xóa đi k đỉnh sao cho vẽ đẹp của đồ thị còn lại bé nhất.

### INPUT

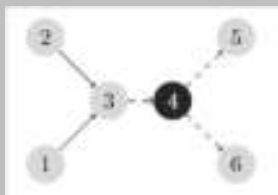
Dòng đầu ghi 3 số n, m và k ( $1 \leq n \leq 300$ ,  $0 \leq m \leq 400$ ,  $0 \leq k \leq \min(n, 4)$ ), các đỉnh đánh số từ 1 đến n.

### OUTPUT

In ra vẽ đẹp bé nhất sau khi xóa tối đa k đỉnh.

Sample Input	Sample Output
6 5 1 1 3 2 3 3 4 4 5 4 6	2
3 3 3 1 2 1 3 2 3	0

Giải thích ví dụ 1: Sau khi xóa đỉnh 4 cùng các cạnh liên quan, tuyến đường dài nhất có độ dài 2 ( $1 \rightarrow 3$  hoặc  $2 \rightarrow 3$ ). Không thể xóa đỉnh nào khác để có tuyến đường có độ dài 1.



# KON

## Vòng 2. 256 MB. 2s.

Cho  $N$  số  $a_1, a_2, \dots, a_N$ . Hãy thêm vào cuối mỗi số này một số chữ số (có thể có số không cần thêm) để thu được một dãy tăng. Hãy tìm cách thêm sao cho tổng số chữ số thêm vào là bé nhất.

### INPUT

Dòng đầu ghi số  $N$  ( $1 \leq N \leq 200,000$ ). Dòng sau ghi  $N$  số nguyên dương  $a_i$  ( $1 \leq a_i \leq 10^9$ ).

### OUTPUT

In ra tổng số chữ số cần thêm ít nhất.

Sample Input	Sample Output
3 8 5 12	2

Giải thích ví dụ: có thể thêm như sau  $(8, 5, 12) \rightarrow (8, 5\underline{1}, 12\underline{2})$ .

# MAG

## Vòng 3 | 256M | 1s

Cho  $n$  đoạn thẳng. Một điểm  $P$  được gọi là an toàn nếu với mọi hướng xuất phát từ  $P$ , tồn tại 1 đoạn thẳng  $i$  sao cho hướng này đến gần cả 2 đầu mút của đoạn thẳng  $i$ . Vùng an toàn là tập hợp các điểm an toàn. Tính diện tích vùng an toàn.

### INPUT

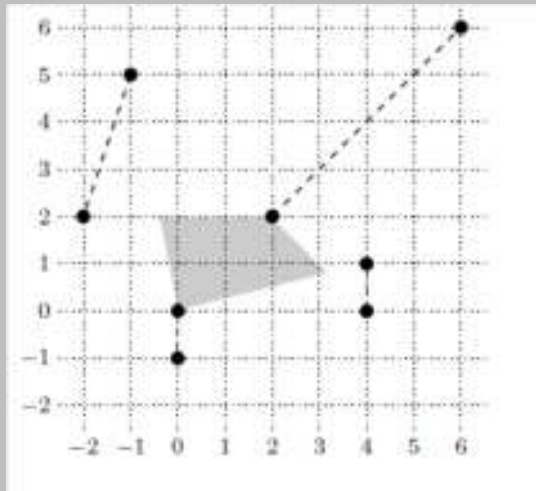
Dòng đầu ghi số đoạn thẳng  $n$  ( $3 \leq n \leq 100$ ). Sau đó là  $n$  dòng, mỗi dòng ghi 4 số nguyên  $ax_i, ay_i, bx_i, by_i$  ( $-500 \leq ax_i, ay_i, bx_i, by_i \leq 500$ ), thể hiện tọa độ 2 đầu mút của đoạn thẳng thứ  $i$  là  $(ax_i, ay_i)$  và  $(bx_i, by_i)$ . Đảm bảo  $2n$  đầu mút này phân biệt.

### OUTPUT

In ra diện tích vùng an toàn với độ chính xác  $10^{-8}$ .

Sample Input	Sample Output
4 0 0 0 -1 -1.5 -2 2 4 0 4 1 2 2 6 6	4.8000000000

Giải thích ví dụ: Vùng xám an toàn. Điểm  $(1, 1 \frac{1}{2})$  an toàn. Điểm  $(5, 5)$  không an toàn vì khi dịch chuyển lên trên, nó chỉ đến gần 1 đầu mút của đoạn thẳng thứ 4.



## PAL

### Vòng 3 | 4 MB | 5s.

Kiểm tra xem một xâu có phải xâu đối xứng hay không. Chú ý chỉ có 4MB bộ nhớ.

#### INPUT

Dòng đầu ghi số  $n$  (độ dài xâu). Nếu  $n > 0$  thì dòng thứ 2 ghi xâu có  $n$  ký tự. Nếu  $n = 0$  thì dòng thứ 2 ghi xâu (đọc không biết độ dài xâu). Xâu chứa các ký tự thường, có độ dài không quá 20000000.

#### OUTPUT

In ra TAK nếu xâu đối xứng, NIE nếu xâu không đối xứng.

Sample Input	Sample Output
4 abba	TAK
0 abc	NIE

## GRA

### Vòng 5 | 256MB | 5s

Đây là trò chơi sưu tập vàng trên lưới ô vuông và mang về căn cứ. Quy tắc chơi như sau:

Kích thước bảng  $n \times n$ . Các ô được đánh số từ trên xuống dưới (từ 0 đến  $n-1$ ), các cột được đánh số từ trái qua phải (từ 0 đến  $n-1$ ). Tọa độ ô ở hàng  $r$ , cột  $c$  là  $(r, c)$ . 2 ô được gọi là kề nhau nếu chung cạnh.

Căn cứ ở ô  $(0, 0)$  – góc trên bên trái. Bạn có thể tạo mới Xe ở đây, đồng thời phải mang vàng về đây. Mỗi một trong  $n^2-1$  ô còn lại hoặc chứa vàng, hoặc chứa đá.

Có 2 loại Xe: Xe Gặt để nhặt vàng - không được đi vào ô có đá và Xe Lu để nhặt đá – đi được vào mọi ô.

Trò chơi chia thành lượt. Ở mỗi lượt, Xe đi đến một trong các ô kề. Không có 2 Xe ở cùng 1 ô. Mọi di chuyển diễn ra ngay lập tức.

Nếu Xe Gặt đi qua ô chứa vàng, sẽ lấy 10 đồng vàng và đặt vào cốp (Nếu có ít hơn, sẽ lấy hết). Cốp xe cực to nên chứa bao nhiêu đồng vàng cũng được. Khi quay trở về căn cứ, Xe Gặt đổ hết vàng.

Nếu Xe Lu đi vào ô có đá, nó sẽ lấy 10 viên (hoặc lấy hết nếu có ít hơn 10 viên).

Ban đầu có 200 đồng vàng ở căn cứ. Chi phí mua mỗi Xe là 100 đồng (bạn phải có đủ từng đó ở căn cứ)

Cho trước bảng, hãy đưa ra 1 dãy lệnh để mang hết vàng về căn cứ. Các lệnh như sau:

R FARMER : Mua 1 Xe Gặt ở căn cứ.

R TANK : Mua 1 Xe Lu ở căn cứ.

M r1 c1 r2 c2 : Chuyển xe từ ô (r1, c1) sang ô kề (r2, c2).

= : Kết thúc một lượt

=== : Kết thúc game

Chương trình chạy trên nhiều test. Mỗi test có giới hạn k là số lượt ở mỗi test

## INPUT

Dòng đầu ghi số test T và giới hạn k (trừ test ví dụ, mọi test có  $T = 10$ ). Trong mỗi test: dòng đầu ghi số n. Sau đó là n dòng, mỗi dòng ghi n số. Số 0 là ô căn cứ. Số nguyên dương a thể hiện có a đồng vàng. Số nguyên âm b thể hiện có |b| viên đá ở ô tương ứng.

## OUTPUT

I ra chuỗi lệnh, mỗi lệnh một dòng.

Sample Input	Sample Output

## RYK

### Vòng 5 | 256MB | 4s

Bearland là lưới ô vuông vô tận, các hàng được đánh số từ dưới lên trên, các cột đánh số từ trái qua phải. Gọi ô (r, c) là ô ở hàng r, cột c. Hai ô được gọi là kề nếu có chung đỉnh (như vậy 1 ô có 8 ô kề). Khoảng cách giữa 2 ô A ( $R_A, C_A$ ) và B ( $R_B, C_B$ ) là khoảng cách Euclid  $\sqrt{(R_A^2 - R_B^2)^2 + (C_A^2 - C_B^2)^2}$ .

Có n con gấu, con thứ i ở ô ( $r_i, c_i$ ). Khi một con gấu hù, những con gấu còn lại sẽ di chuyển sang ô kề nào có khoảng cách đến ô có con gấu hù bé nhất (luôn tồn tại duy nhất 1 ô như vậy). Ví dụ có 2 con gấu ở ô (2, 1) và (4, 8). Khi con gấu 1 hù, con gấu 2 sẽ di chuyển đến ô (3, 7) (khoảng cách là  $\sqrt{(3-2)^2 + (7-1)^2} = \sqrt{37}$ ). Nếu ở cùng ô với con gấu hù, con gấu sẽ đứng im không di chuyển.

Các con gấu 1, 2, ..., n lần lượt hù – ngoại trừ con gấu Bear (đang bị viêm họng). Với mỗi k ( $1 \leq k \leq n$ ), xác định vị trí cuối cùng của các con gấu nếu gấu k là gấu Bear.

Cụ thể, giả sử vị trí cuối cùng của con gấu thứ i là ( $r'_i$  và  $c'_i$ ), bạn phải tính  $\sum_{i=1}^n r'_i \times c'_i$

## INPUT

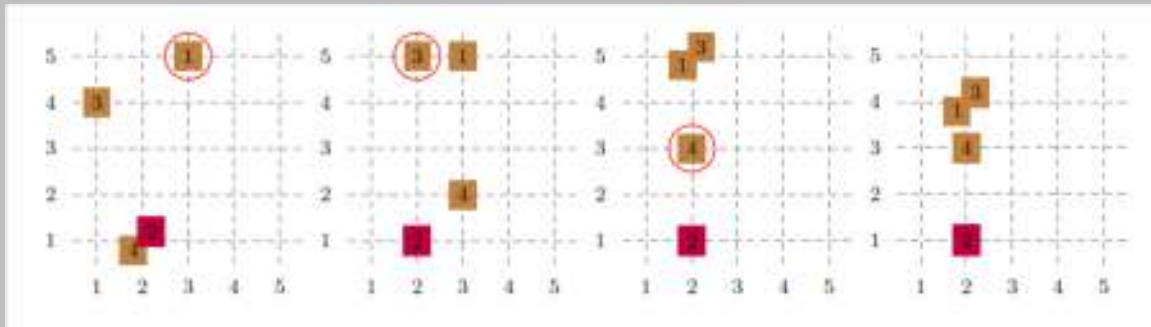
Dòng đầu ghi số gấu n ( $2 \leq n \leq 250,000$ ). Sau đó là n dòng, dòng thứ i ghi tọa độ ô ban đầu của gấu i, gồm 2 số  $r_i, c_i$  ( $1 \leq r_i, c_i \leq 10^6$ ).

## OUTPUT

In ra n dòng, dòng thứ k là giá trị cần tính trong trường hợp gấu k không hù.

Sample Input	Sample Output
4	27
3 5	24
2 1	25
1 4	35
2 1	

Giải thích dòng 2: Giả sử gấu 2 không hù. Gấu 1, 3, 4 lần lượt hù. Tổng cuối cùng là  $2 \cdot 4 + 2 \cdot 1 + 2 \cdot 4 + 2 \cdot 3 = 24$ .



## SKW

### Vòng 5 | 256MB | 3s

Cho một hoán vị  $(a_1, a_2, \dots, a_N)$  của các số từ 1 đến  $N$ . Cứ mỗi giây, số  $a_i$  biến mất nếu số đứng bên cạnh nó lớn hơn. Dĩ nhiên cuối cùng chỉ còn lại số  $N$ . Ví dụ với hoán vị  $(1, 2, 5, 3, 4)$ , các số 1, 2, 3 sẽ mất, còn lại  $(5, 4)$ .

Có bao nhiêu hoán vị mà sau đúng  $K$  giây sẽ còn lại đúng số  $N$ .

Vì đáp số rất lớn, nên lấy theo modulo  $P$  ( $P$  nguyên tố).

### INPUT

Số  $N, M, K$ . ( $1 \leq M \leq N \leq 1000, N \geq 2, 10^8 \leq P \leq 10^9$ )

### OUTPUT

Đáp số cần tìm.

Sample Input	Sample Output
5 3 100000007	4

4 hoán vị là:  $(4, 1, 3, 2, 5), (4, 2, 3, 1, 5), (5, 1, 3, 2, 4), (5, 2, 3, 1, 4)$

## WIE

### Vòng 5 | 256MB | 3s

Đếm số lượng các đa giác lồi có đỉnh nằm ở nút có tọa độ nguyên trên lưới điểm nguyên, trên cạnh và bên trong đa giác không chứa điểm nguyên.

Nếu đỉnh đa giác là  $(x_i, y_i)$  thì phải thỏa mãn các điều kiện sau đây::

$$1 \leq x_i \leq X \text{ và } 1 \leq y_i \leq Y$$

Độ dài mỗi cạnh không vượt quá  $K$ .

Hình dưới minh họa các đa giác không thỏa mãn. Đa giác 1, 2 có điểm nguyên trên cạnh và bên trong đa giác, đa giác 3 không phải đa giác lồi.

In kết quả theo mod  $2^{32}$ . 2 đa giác khác nhau nếu có 1 điểm là đỉnh của đúng một đa giác.

### INPUT

Dòng đầu ghi 3 số nguyên  $X, Y$  và  $K$  ( $1 \leq X, Y \leq 10^9, 1 \leq K \leq 250$ )

### OUTPUT

In ra kết quả phải tính

Sample Input	Sample Output
--------------	---------------

Đa giác 4 đỉnh sau thỏa mãn  $(2, 1)$ ,  $(2, 2)$ ,  $(6, 5)$ ,  $(3, 1)$ ,

