

COCI. Tháng 10/2008. Lần I

3. MRAVOJED

Các nhà khảo cổ vừa tìm thấy phần còn lại của kiến trúc Hy Lạp - La Mã. Nơi này có thể được mô hình hóa như một mạng lưới các $R \times C$ ô vuông. Đối với mỗi ô, các nhà khảo cổ đã xác định: hoặc có một tòa nhà ở đó hoặc là ô đó luôn luôn trống.

Sau khi xem xét các hiện vật cụ thể, họ kết luận rằng ở đây có **hai tòa nhà** từ 2 thời kì khác nhau và cấu trúc sàn của cả hai tòa nhà đều có dạng hình vuông. Bởi vì các khu nhà ở 2 thời kỳ khác nhau nên có thể cấu trúc sàn của chúng chồng chéo lên nhau. Xác định vị trí và kích thước (chiều dài của mặt sàn) cho mỗi tòa nhà.

INPUT

Dòng đầu tiên chứa hai số nguyên R ($1 \leq R \leq 100$) và C ($1 \leq C \leq 100$) - kích thước của khu vực. Mỗi dòng trong R dòng tiếp theo chứa C ký tự '.' (dấu chấm) hoặc 'x' (chữ thường). Ký tự '.' có nghĩa là không có gì được tìm thấy trong ô, trong khi 'x' chỉ ra rằng có một tòa nhà ở đó.

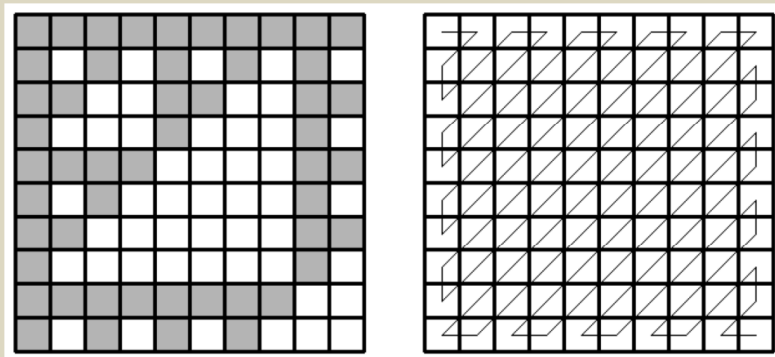
OUTPUT

Đối với mỗi một trong hai tòa nhà, in trên một dòng hàng và cột của góc trên bên trái của tòa nhà, và kích thước của tòa nhà. Lưu ý: test đảm bảo rằng luôn luôn tồn tại một giải pháp, mặc dù có thể không phải là duy nhất.

Sample Input	Sample Output
3 3 xx. xxx ...	1 1 2 2 3 1
4 6 xx.... xx.xxx ...xxx ...xxx	1 1 2 2 4 3
5 5 xxx.. xxxx. xxxx. .xxx.	2 1 3 3 2 3

4. JEZ

Luka tìm thấy một bàn cờ rất lạ trong căn gác của mình. Nó bao gồm $R \times C$ ô vuông và các hàng được đánh số từ 0 đến $R-1$ theo chiều từ trên xuống dưới và các cột đánh số từ 0 đến $C-1$ theo chiều từ trái sang phải. Các ô trong bàn đều được tô bằng một trong hai màu: xám hoặc trắng.



Màu trắng: nếu trong biểu diễn ở hệ nhị phân số hàng và số cột của ô này có ít nhất một chữ số 1 tại cùng một vị trí. Ví dụ, ô (4, 5) sẽ có màu trắng.

Màu xám, nếu ngược lại. Ví dụ, ô (2, 5) tô màu xám.

Chủ nhím của Luka thích đi bộ trên cái bảng đó và hiện nó đang đi một cách bất thường. Nó bắt đầu đi bộ từ ô (0, 0) và tiếp tục đi theo hình zig-zag, trong khi đó Luka đếm xem có bao nhiêu hình vuông màu xám mà chú nhím đã đi qua.

Sau khi nhím đi được K ô vuông thì nó đã mệt mỏi và ngủ thiếp đi. Luka sau đó cũng đi ngủ. Thật vui khi Luka đã có thể đếm được hết các ô vuông màu xám. Biết kích thước của bảng và số K cho trước. Nhiệm vụ của bạn là viết chương trình đếm.

INPUT

Dòng đầu tiên chứa hai số nguyên R ($1 \leq R \leq 1\,000\,000$) và C ($1 \leq C \leq 1\,000\,000$) là kích thước của bảng. Dòng thứ hai chứa số nguyên K ($1 \leq K \leq R \times C$) - tổng số ô vuông mà chú nhím đã đi qua. Lưu ý rằng con số này có thể không phù hợp với một số nguyên 32-bit.

50% số test có $K < 1000000$.

OUTPUT

Số lượng các ô màu xám mà chú nhím đã đi qua.

Sample Input	Sample Output
10 10 6	5
3 5 11	8
10 10 100	51

5. SKAKAVAC

Một con châu chấu ở trong một cánh đồng hoa. Trong cánh đồng có $N \times N$ bông hoa sắp xếp trong N dòng và N cột. Đối với mỗi bông hoa, chúng ta đều biết số cánh hoa nó có. Con châu chấu ban đầu ở trên bông hoa trên dòng R và cột C. Mục tiêu của nó là nhảy đến nhiều bông hoa nhất có thể trong khi tuân theo các quy tắc:

1. Nó chỉ có thể nhảy sang hàng hoặc cột liền kề. Nếu nó nhảy vào hàng liền kề, nó phải nhảy ít nhất hai cột, và nếu nó nhảy vào cột liền kề, nó phải nhảy ít nhất hai hàng. Nói cách khác, nó có thể nhảy từ hoa (r1, c1) đến hoa (r2, c2) nếu:

- $|r1-r2| = 1$ và $|c1-c2| > 1$ hoặc
- $|c1-c2| = 1$ và $|r1-r2| > 1$

2. Số lượng cánh hoa trên bông hoa tiếp theo phải lớn hơn so với số cánh hoa trên bông hoa trước.

Hãy viết một chương trình tính toán số lượng lớn nhất các hoa mà châu chấu có thể nhảy đến.

INPUT

Dòng đầu tiên chứa số nguyên N ($1 \leq N \leq 1500$)- kích thước của cánh đồng. Dòng thứ hai chứa số nguyên R ($1 \leq R \leq N$) và C ($1 \leq C \leq N$), vị trí ban đầu của châu chấu. N dòng tiếp theo chứa N số nguyên dương cách nhau bằng dấu cách, mỗi số ít hơn 1000000- số cánh hoa trên những bông hoa.

OUTPUT

In ra số lượng lớn nhất của các bông hoa mà châu chấu có thể nhảy đến.

Trong 50% số test, $N < 100$.

Trong 80% số test, $N < 1000$.

Sample Input	Sample Output
4	4

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

6. KRTICA

Chuột chũi là động vật gọn gàng và làm việc chăm chỉ. Chúng luôn giữ cho tổ dưới lòng đất gọn gàng nhất có thể, do đó tất cả số chuột chũi sống ở đó đều biết nơi để tìm những thứ đã cất đi. Để đạt được điều này, chúng phải kết nối các phòng bằng các đường hầm để chỉ có một cách duy nhất đi được từ phòng này đến phòng khác. Khoảng cách giữa hai phòng là số đường kết nối chúng. Mặc dù nỗ lực nhiều, một số vị khách vẫn phàn nàn rằng phải mất quá nhiều thời gian để đi bộ giữa một số cặp phòng.

Do đó chuột chũi đã quyết định xây dựng lại nơi cư trú của mình, đóng cửa một đường hầm và xây dựng một đường hầm mới để khoảng cách giữa hai phòng xa nhất là nhỏ nhất có thể, nhưng vẫn có thể đi từ bất kỳ phòng nào sang bất kỳ phòng nào khác.

Viết chương trình xác định khoảng cách giữa hai phòng xa nhất sau khi tái thiết, đường hầm nào phải đóng và đường hầm nào được mở.

INPUT

Dòng đầu tiên chứa một số nguyên $1 \leq N \leq 300000$ là số phòng. Các phòng đều được đánh số từ 1 đến N. Mỗi dòng trong số N-1 dòng tiếp theo ghi hai số nguyên, là số thứ tự của phòng là đầu mút một đường hầm.

OUTPUT

In lần lượt trên mỗi dòng

- Khoảng cách giữa hai phòng xa nhất sau khi xây dựng lại.
- Một cặp số nguyên đại diện cho một đường hầm được đóng lại.
- Một cặp số nguyên, thể hiện đầu mút của đường hầm mới được mở ra.

Lưu ý: Giải pháp này sẽ không nhất thiết phải là duy nhất. In ra bất kỳ kế hoạch tái thiết nào mà khoảng cách giữa hai phòng xa nhất là nhỏ nhất.

Sample Input	Sample Output
4 1 2 2 3 3 4	2 3 4 4 2
7 1 3 2 3 2 7 4 3 7 5 3 6	3 2 3 7 3

