

Bài 1. Danh sách tin nhắn**Message.cpp**

Huyền có một cái điện thoại, màn hình điện thoại của Huyền chỉ hiển thị được tối đa **k** tin nhắn. Màn hình của Huyền hiện thị như sau:

- Không hiện thị 2 tin nhắn của cùng một số điện thoại (SĐT) trên cùng một khung hình, nếu SĐT **a** gửi tin nhắn đến mà trên màn hình đã có tin nhắn của SĐT **a** thì màn hình không thay đổi gì.
- Khi SĐT **a** gửi tin nhắn đến mà trên màn hình chưa có tin nhắn của SĐT **a** thì:
 - Nếu màn hình chưa đủ **k** tin nhắn thì tin nhắn của SĐT **a** sẽ được chèn vào cuối màn hình.
 - Nếu màn hình đã có đủ **k** tin nhắn thì màn hình sẽ đẩy tin nhắn trên cùng ra và sau đó chèn tin nhắn của SĐT **a** vào cuối màn hình.

Cho dãy **a** là các SĐT sẽ gửi tin nhắn cho Huyền. Hỏi sau khi nhận được tin nhắn cuối cùng thì màn hình của Huyền đang hiển thị tin nhắn của các SĐT nào, đưa ra theo thứ tự từ trên xuống dưới của màn hình.

Giới hạn: $k \leq n \leq 10^5, a_i \leq 10^5$.

Ví dụ:

Message.inp	Message.out
5 1 2 1 3 4 3	2 3 4

Với $a = [1, 2, 1, 3, 4]$, $k = 3$. thì $\text{messagesPhone}(a) = [2, 3, 4]$;

Giải thích:

- Lúc đầu màn hình điện thoại rỗng [].
- Sau khi 1 nhắn tin đến thì màn hình hiển thị [1].
- Sau khi 2 nhắn tin đến thì màn hình hiển thị [1, 2].
- Sau khi 1 nhắn tin đến thì màn hình không thay đổi gì vì trên màn hình đã có tin nhắn của 1 rồi.
- Sau khi 3 nhắn tin đến màn hình hiển thị [1, 2, 3]
- Sau khi 4 nhắn tin đến, lúc này màn hình đã hiển thị đủ 3 tin nhắn rồi nên điện thoại sẽ đẩy tin nhắn đầu tiên là 1 đi và chèn tin nhắn của 4 vào cuối, cuối cùng màn hình hiển thị [2, 3, 4].

Bài 2. Đường đi ngắn nhất trên ma trận**matrix.cpp**

Cho một ma trận gồm m hàng, n cột, các hàng được đánh số từ 1 tới m từ trên xuống dưới, các cột được đánh số từ 1 tới n từ trái qua phải. Ô nằm trên giao của hàng i , cột j được gọi là ô (i, j) , trên ô đó có ghi số 0 hoặc 1 thể hiện có thể đi vào hoặc không thể đi vào nó. Bạn xuất phát từ ô (x, y) và mỗi bước di chuyển có thể đi sang một trong bốn ô trên, dưới, trái, phải.

Hãy tìm đường đi ngắn nhất từ ô (x_1, y_1) tới ô (x_2, y_2) trên ma trận? Biết rằng, độ dài đường đi giữa hai ô được tính bằng tổng số lượng ô trên đường đi, tính cả hai ô xuất phát và kết thúc.

Input:

- Dòng đầu tiên chứa hai số nguyên dương m, n .
- m dòng tiếp theo, mỗi dòng chứa một dãy gồm n số thuộc hai giá trị 0–1 thể hiện ma trận.
- Dòng cuối cùng chứa bốn số nguyên dương x_1, y_1, x_2, y_2 .

Output:

- Dòng đầu tiên in ra số nguyên duy nhất là độ dài đường đi ngắn nhất giữa hai ô (x_1, y_1) và (x_2, y_2) . Nếu không thể di chuyển được thì in ra -1 .

Ví dụ:

Matrix.inp	Matrix.out	Giải thích
5 5 0 1 0 0 0 0 0 1 0 1 0 0 0 1 0 0 1 0 0 0 0 1 0 0 1 1 1 5 3	7	Đường đi theo các ô được đánh dấu trên hình.

Bài 3. Tìm đường đi cho rô bốt

Robot.cpp

Trên một lưới ô vuông kích thước $M \times N$ ($M, N \leq 1000$), có hai robot:

- Robot A bắt đầu từ góc trái trên của lưới (ô $(1,1)$).
- Robot B bắt đầu từ góc phải dưới của lưới (ô (M,N)).

Mỗi ô trong lưới có thể chứa một vật cản hoặc không:

- 0: ô trống (có thể di chuyển vào).
- 1: ô có vật cản (không thể đi vào).

Hai robot di chuyển đồng thời với tốc độ như nhau. Trong một bước, mỗi robot có thể di chuyển sang một ô liền kề theo 4 hướng (lên, xuống, trái, phải). Bắt buộc phải di chuyển, không được đứng yên; chúng có thể quay lại ô trước đó đã đi qua. Hai robot sẽ gặp nhau khi chúng **đứng cùng một ô tại một thời điểm** nào đó.

Yêu cầu: Xác định số bước ít nhất để hai robot gặp nhau.

Dữ liệu vào:

Dòng đầu tiên chứa hai số nguyên M, N (số hàng và số cột của lưới).

M dòng tiếp theo, mỗi dòng gồm N số 0 hoặc 1, cách nhau một khoảng trắng, mô tả trạng thái của các ô trong lưới.

Ô $(1,1)$ và (M,N) đảm bảo có giá trị 0 (không có vật cản).

Dữ liệu ra:

Nếu hai robot có thể gặp nhau, in ra số bước ít nhất để chúng gặp nhau.

Nếu không thể gặp nhau, in ra ký tự "#".

Ví dụ:

Robot.inp	Robot.out	Giải thích
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	4	Đi theo các ô được đánh dấu, gặp nhau tại ô: (2,4)
5 5 0 1 1 1 1 0 1 0 0 0 0 1 0 1 0 0 1 0 1 0 0 0 0 1 0	7	Gặp nhau tại vị ô (4,3)
1 10 0 0 0 0 0 0 0 0 0 0	#	Robot không gặp nhau được

Bài 4. Đếm số phòng**ROOM.cpp**

Bạn được cung cấp một bản đồ của một tòa nhà, và nhiệm vụ của bạn là đếm số lượng phòng trong tòa nhà đó. Kích thước của bản đồ là $n \times m$ ô vuông, trong đó mỗi ô vuông có thể là sàn hoặc tường. Bạn có thể di chuyển sang trái, phải, lên và xuống qua các ô sàn.

Dữ liệu vào: Dòng đầu tiên chứa hai số nguyên nnn và mmm : chiều cao và chiều rộng của bản đồ.

Tiếp theo có nnn dòng, mỗi dòng gồm mmm ký tự mô tả bản đồ.

Ký tự `.` đại diện cho sàn.

Ký tự `#` đại diện cho tường.

Dữ liệu ra: In ra một số nguyên: số lượng phòng trong tòa nhà.

Ràng buộc: $1 \leq n, m \leq 1000$.

Ví dụ:

Room.inp	Room.out	Giải thích
<pre> 5 8 ##### #...#...# ####.#.# #...#...# ##### </pre>	3	Ở ví dụ trên, có 3 phòng được tạo bởi các vùng <code>.</code> kết nối nhau

Bài 5. Bồn hoa**FlowerPot.cpp**

Có một mảnh đất hình chữ nhật được chia thành các ô vuông, trên mỗi dòng có n ô và trên mỗi cột có m ô. Trên đó người ta xây dựng các bồn hoa, mỗi bồn hoa bao gồm các ô liên kết với nhau. Hai ô cùng nằm trên một bồn hoa nếu chúng có cùng chung cạnh.

Hãy xác định diện tích và chu vi của bồn hoa lớn nhất (mỗi ô vuông là một đơn vị đo diện tích, mỗi cạnh của ô vuông là một đơn vị đo chiều dài)?

Input:

- Dòng đầu tiên chứa hai số nguyên dương m, n – kích thước mảnh đất.
- m dòng tiếp theo, mỗi dòng ghi n số nguyên dương 0 hoặc 1, giữa hai số cách nhau một khoảng trống. Tại dòng i , số thứ j được ghi:
 - Số 0 nếu ô ở dòng i , cột j không nằm trong bồn hoa nào.
 - Số 1 nếu ô ở dòng i , cột j nằm trong bồn hoa nào đó.

Ràng buộc: $1 \leq m, n \leq 1000$.

Output: Gồm hai số nguyên trên hai dòng lần lượt là diện tích của bồn hoa có diện tích lớn nhất và chu vi của bồn hoa có chu vi lớn nhất.

FlowerPot.inp	FlowerPot.out	Giải thích
<pre> 7 10 0 1 1 1 0 0 0 0 1 1 1 1 1 0 0 0 0 0 0 1 0 1 1 1 1 0 1 1 1 0 0 0 0 0 0 0 0 1 0 0 0 0 1 1 0 0 0 0 0 0 0 1 1 1 0 1 1 1 0 0 0 0 1 0 0 1 0 0 0 1 </pre>	<pre> 10 18 </pre>	Chú ý chưa chắc bồn hoa có diện tích lớn nhất đã là bồn hoa có chu vi lớn nhất

Bài 6. Bảo vệ nông trang**GUARD.cpp**

John có một nông trang với nhiều ngọn đồi. Để bảo vệ nông trang, anh ta muốn đặt người canh gác trên các đỉnh đồi.

John cần xác định số lượng đỉnh đồi trên bản đồ. Bản đồ được thể hiện dưới dạng ma trận kích thước $N \times M$, trong đó mỗi ô chứa một số nguyên biểu thị độ cao của vị trí đó so với mực nước biển.

Định nghĩa đỉnh đồi:

- Một đỉnh đồi là một hoặc nhiều ô liên kề nhau có cùng độ cao.
- Đỉnh đồi phải được bao quanh bởi mép bản đồ hoặc bởi các ô có độ cao thấp hơn.
- Hai ô được coi là liên kề nếu chúng có độ chênh lệch tọa độ xxx không quá 1 và độ chênh lệch tọa độ yyy không quá 1 (tức là có thể tiếp xúc theo 8 hướng).

Dữ liệu vào:

Dòng đầu tiên chứa hai số nguyên N và M ($1 < N, M \leq 700$) - số hàng và số cột của bản đồ.

N dòng tiếp theo, mỗi dòng gồm M số nguyên ($0 \leq H_{ij} \leq 10000$), biểu thị độ cao của từng ô trong ma trận.

Dữ liệu ra: Một số nguyên duy nhất: số lượng đỉnh đồi trong bản đồ.

Ví dụ:

GUARD.inp	GUARD.out	Giải thích
8 7 4 3 2 2 1 0 1 3 3 3 2 1 0 1 2 2 2 2 1 0 0 2 1 1 1 1 0 0 1 1 0 0 0 1 0 0 0 0 1 1 1 0 0 1 2 2 1 1 0 0 1 1 1 2 1 0	3	Bản đồ trên có 3 đỉnh đồi, mỗi đỉnh là một vùng liên thông có cùng độ cao và không bị bao quanh bởi điểm cao hơn.

Bài 7. Di chuyển trên sao hỏa

Move.cpp

Một nhà du hành vũ trụ đang thám hiểm một vùng đất lạ trên sao Hỏa. Thông qua máy quét radar, bản đồ của vùng đất được mô tả như một bảng gồm m dòng (đánh số từ 1 tới m , từ trên xuống dưới) và n cột (đánh số từ 1 tới n , từ trái sang phải), mỗi ô (i, j) của bảng thể hiện một khu vực của vùng đất và chứa một số nguyên $h_{i,j}$ là chiều cao của khu vực đó (do thiết kế đặc biệt của máy dò nên chiều cao của khu đất có thể hiển thị là một số nguyên âm). Nhà du hành muốn di chuyển từ khu vực (x_1, y_1) tới khu vực (x_2, y_2) của vùng đất. Để đảm bảo tính an toàn, nhà du hành chỉ di chuyển từ một khu vực tới bốn khu vực chung cạnh xung quanh nó, và ông ta sẽ di chuyển sao cho chênh lệch độ cao lớn nhất giữa hai khu vực liên tiếp trên đường đi là nhỏ nhất có thể.

Biết rằng, chênh lệch độ cao giữa hai khu vực được tính bằng giá trị tuyệt đối của hiệu chiều cao giữa hai nơi. Hãy xác định hành trình của nhà du hành?

Input:

- Dòng đầu tiên chứa hai số nguyên m và n .
- Dòng thứ hai chứa bốn số nguyên dương x_1, y_1, x_2, y_2 .
- mm dòng cuối cùng, dòng thứ ii chứa nn số nguyên $h_{i,j}$, mỗi số cách nhau một dấu cách.

Ràng buộc:

- $1 \leq m, n \leq 1000$.
- $|h_{i,j}| \leq 10^9; \forall i, j: 1 \leq i \leq m, 1 \leq j \leq n$.

Output:

- Một số nguyên duy nhất là độ chênh lệch lớn nhất giữa hai khu vực liên tiếp trên hành trình của nhà du hành.

Ví dụ:

Move.inp	Move.out	Giải thích
3 4 1 1 3 3 -2 9 3 -4 0 3 7 -5 8 9 9 10	4	Di chuyển từ ô (1,1) đến (3,3) như sau: (1,1)→(2,1)→(2,2)→(2,3)→(3,3)

Bài 8. Nông Trại

MTNTRAI.cpp

Trong một trang trại hình chữ nhật kích thước $m \times n$ ($m, n \leq 1000$), có các loại ô như sau:

- "." : Ô trống.
- "#" : Hàng rào (không thể đi qua).
- "c" : Ô có gà.
- "f" : Ô có cáo.

Hai ô thuộc cùng một chuồng nếu có thể đi từ ô này đến ô kia theo 4 hướng (trái, phải, lên, xuống) mà không bị vướng vào hàng rào (#).

Luật sinh tồn:

- Nếu trong một chuồng có nhiều gà hơn cáo, thì tất cả cáo trong chuồng sẽ bị mổ chết.
- Nếu cáo nhiều hơn hoặc bằng gà, thì tất cả gà trong chuồng sẽ bị ăn hết.
- Hàng rào (#) phân tách các chuồng, các con vật không thể vượt qua hàng rào.

Yêu cầu: Hãy tính số gà và số cáo còn lại sau khi tất cả các cuộc chiến kết thúc.

Dữ liệu vào:

- Dòng đầu chứa hai số nguyên m, n ($1 \leq m, n \leq 1000$) - số hàng và số cột của trang trại.
- m dòng tiếp theo, mỗi dòng có n ký tự mô tả trạng thái của các ô trong trang trại.

Dữ liệu ra: - Một dòng duy nhất chứa hai số nguyên lần lượt là số cáo còn lại và số gà còn lại sau cuộc chiến.

Ví dụ:

MTNTRAI.inp	MTNTRAI.out	Giải thích
8 8 .###### #...c...# #.##### #.f.#.# #.#.c#c# #c.##...# #.f...f.# .#####.	1 3	<ul style="list-style-type: none"> - Có nhiều chuồng trong trang trại. - Trong mỗi chuồng, nếu gà nhiều hơn cáo → gà tiêu diệt hết cáo. - Nếu cáo nhiều hơn hoặc bằng gà → cáo ăn hết gà. - Sau tất cả các cuộc chiến, còn lại 1 cáo và 3 gà.

Bài 9.

Tí tham gia Hội chợ xuân và giành được phần thưởng. Ban tổ chức đưa ra một bảng số được chia thành lưới ô vuông đơn vị kích thước $M \times N$. Các hàng của bảng được đánh số từ 1 đến M , từ trên xuống dưới. Các cột của bảng được đánh số từ 1 đến N , từ trái qua phải. Ô nằm trên hàng i , cột j của bảng được gọi là ô (i, j) và có ghi giá trị a_{ij} ($1 \leq i \leq M, 1 \leq j \leq N$). Với 2 số nguyên dương P, Q cho trước, Tí phải chọn ra một bảng con kích thước $P \times Q$ (P dòng, Q cột) của bảng số, phần thưởng của Tí sẽ là ô có giá trị nhỏ nhất trong bảng con này.

Yêu cầu: Em hãy giúp Tí chọn bảng con để đạt được phần thưởng có giá trị lớn nhất.

Input

- Dòng thứ nhất chứa bốn số nguyên dương M, N, P, Q ($1 < M \leq 1000, 1 < N \leq 1000, P \leq M, Q \leq N$).
- Trong M dòng tiếp theo, mỗi dòng chứa N số nguyên dương a_{ij} là số ghi trên ô ở dòng thứ i và cột thứ j . ($a_{ij} \leq 10^6; 1 \leq i \leq M; 1 \leq j \leq N$)

Output: ghi một số nguyên duy nhất là giá trị phần thưởng lớn nhất Tí có thể đạt được.

Ví dụ:

Input	Output	Giải thích
5 5 2 3 1 11 4 3 3 9 9 4 6 3 2 5 5 5 2 1 2 2 5 6 4 10 2 7 8	4	Bảng con thỏa đề có góc trái trên là (2, 2), ô nhỏ nhất có giá trị là 4 1 11 4 3 3 9 9 4 6 3 2 5 5 5 2 1 2 2 5 6 4 10 2 7 8