



Bài tập luyện tập

1. QBMAX - Đường đi có tổng lớn nhất

- Type: Array 2-D
- Level: Hard
- Href: <https://vn.spoj.com/problems/QBMAX/>

Cho một bảng A kích thước $m \times n$ ($1 \leq m, n \leq 100$), trên đó ghi các số nguyên a_{ij} ($|a_{ij}| \leq 100$). Một người xuất phát tại ô nào đó của cột 1, cần sang cột n (tại ô nào cũng được).

Quy tắc đi: Từ ô (i, j) chỉ được quyền sang một trong 3 ô $(i, j + 1)$; $(i - 1, j + 1)$; $(i + 1, j + 1)$

Input

Dòng 1: Ghi hai số m, n là số hàng và số cột của bảng.

M dòng tiếp theo, dòng thứ i ghi đủ n số trên hàng i của bảng theo đúng thứ tự từ trái qua phải

Output

Gồm 1 dòng duy nhất ghi tổng lớn nhất tìm được

- Example:

Input:

```
5 7
9 -2 6 2 1 3 4
0 -1 6 7 1 3 3
8 -2 8 2 5 3 2
1 -1 6 2 1 6 1
7 -2 6 2 1 3 7
```

Output: 41



2. Điều khiển máy bay

Cho một ma trận kích thước $N \times 7$ ($10 \leq N \leq 50$) chứa các số 0, 1, 2 với 0 tương ứng với vị trí trống, 1 tương ứng với vị trí có vàng, 2 tương ứng với vật cản.

Một chiếc máy bay được đặt ở vị trí $(N-1, 3)$, máy bay sẽ di chuyển theo lệnh cho trước, các lệnh được quy ước như sau:

Số 1: máy bay đi thẳng lên trên

Số 2: máy bay đi sang trái

Số 3: máy bay đi sang phải



2. Điều khiển máy bay

Ví dụ:

10 5 ← N=10, ma trận 10 hàng 7 cột, gồm có 5 câu lệnh di chuyển

2 1 3 2 1 ← các lệnh di chuyển

0 1 2 0 0 1 1

2 2 2 2 2 2 0

2 1 2 2 2 2 0

0 1 2 1 0 1 0

0 1 2 1 1 0 0 ← dữ liệu của ma trận

0 2 2 2 2 2 0

1 0 0 0 0 1 2

0 0 1 1 2 2 2

2 0 2 2 1 0 2

0 1 0 0 1 0 2



2. Điều khiển máy bay

0 1 2 0 0 1 1

2 2 2 2 2 2 0

2 1 2 2 2 2 0

0 1 2 1 0 1 0

0 1 2 1 1 0 0

0 2 2 2 2 2 0

1 0 0 0 0 1 2

0 0 1 1 2 2 2

2 0 2 2 1 0 2

0 1 0 0 1 0 2

Vị trí (9,3) được tô màu đỏ là vị trí xuất phát của máy bay.

Các lệnh đc thực hiện tuần tự là: 2 1 3 2 1

Đầu tiên, lệnh số 2 là rẽ trái, như vậy máy bay sau khi di chuyển sẽ ở vị trí là (8,2):

2. Điều khiển máy bay

0 1 2 0 0 1 1

2 2 2 2 2 2 0

2 1 2 2 2 2 0

0 1 2 1 0 1 0

0 1 2 1 1 0 0

0 2 2 2 2 2 0

1 0 0 0 0 1 2

0 0 1 1 2 2 2

2 0 2 2 1 0 2

0 1 0 0 1 0 2

Tuy nhiên do vị trí (8,2) là một vật cản nên máy bay không thể di chuyển đến vị trí này, vì vậy lệnh này bỏ qua.

Tiếp theo là lệnh số 1: máy bay đi thẳng tức là từ vị trí (9,3) lên vị trí (8,3), tuy nhiên cũng giống như trường hợp trên, vị trí (8,3) là một vật cản nên lệnh được bỏ qua

2. Điều khiển máy bay

Tiếp theo là lệnh số 3: máy bay sẽ rẽ phải

0 1 2 0 0 1 1

2 2 2 2 2 2 0

2 1 2 2 2 2 0

0 1 2 1 0 1 0

0 1 2 1 1 0 0

0 2 2 2 2 2 0

1 0 0 0 0 1 2

0 0 1 1 2 2 2

2 0 2 2 1 0 2

0 1 0 0 1 0 2

Máy bay sẽ di chuyển từ vị trí (9,3) lên vị trí (8, 4), vị trí này là vị trí có vàng vì vậy số vàng mà máy bay thu được là 1.

Lệnh tiếp theo là lệnh 2: rẽ trái, máy bay từ vị trí (8,4) sẽ lên vị trí (7,3)

2. Điều khiển máy bay

0 1 2 0 0 1 1

2 2 2 2 2 2 0

2 1 2 2 2 2 0

0 1 2 1 0 1 0

0 1 2 1 1 0 0

0 2 2 2 2 2 0

1 0 0 0 0 1 2

0 0 1 1 2 2 2

2 0 2 2 1 0 2

0 1 0 0 1 0 2

Vì vị trí (7,3) là vị trí có vàng nên số vàng của máy bay tăng lên 2.

Lệnh cuối cùng là 1: đi thẳng.

Từ vị trí (7,3) lên vị trí (6,3) là một vị trí trống, vì vậy máy bay chỉ di chuyển đến vị trí đó.



2. Điều khiển máy bay

Kết thúc chuỗi lệnh di chuyển, tổng cộng máy bay thu được 2 vàng, vì vậy in kết quả là 2

Cho biết số hàng N của ma trận, số lệnh di chuyển, các lệnh di chuyển và dữ liệu ma trận.

Hãy viết chương trình đưa ra số vàng mà máy bay thu được sau khi thực hiện tất cả các lệnh đã cho.

Sample input:



2. Điều khiển máy bay

1 ← số test case

10 5 ← N=10, ma trận 10 hàng 7 cột, gồm có 5 câu lệnh di chuyển

2 1 3 2 1 ← các lệnh di chuyển

0 1 2 0 0 1 1

2 2 2 2 2 2 0

2 1 2 2 2 2 0

0 1 2 1 0 1 0

0 1 2 1 1 0 0 ← dữ liệu của ma trận

0 2 2 2 2 2 0

1 0 0 0 0 1 2

0 0 1 1 2 2 2

2 0 2 2 1 0 2

0 1 0 0 1 0 2

Sample output

#1 2



3. Chạy đua

Cho một ma trận $N \times M$ ($5 \leq N$, $M \leq 100$) chứa các số 0, 1, 2, 3.

Các vận động viên bắt đầu xuất phát từ hàng thứ nhất của ma trận và đích là hàng cuối cùng của ma trận.

Các vận động viên sẽ chạy đua theo quy tắc như sau:

Khi chạy nếu vận động viên gặp vị trí có giá trị là 0 thì tiếp tục chạy thẳng vì vậy mất 1 giây để qua vị trí tiếp theo, nếu gặp vị trí có giá trị là 1 thì vận động viên bắt buộc phải rẽ trái vì vậy sẽ mất 2 giây để sang vị trí tiếp theo, nếu gặp vị trí có giá trị là 2 thì vận động viên bắt buộc phải rẽ phải vì vậy sẽ mất 2 giây để sang vị trí tiếp theo, nếu gặp vị trí có giá trị là 3 thì vận động viên phải vượt chướng ngại vật sẽ mất thời gian là 3 giây sau đó chạy thẳng tiếp.

Trong trường hợp phải rẽ phải hoặc rẽ trái mà gặp biên không thể rẽ được thì vận động viên tiếp tục chạy thẳng và tốn 1 giây.

Cho kích thước ma trận $N \times M$ và dữ liệu ma trận, hãy xác định vận động viên nào về đích đầu tiên, nếu có nhiều hơn 1 vận động viên về đích sớm nhất, in ra vận động viên có số thứ tự cao nhất




3. Chạy đua

Ví dụ ta có ma trận 7x9:

```
3 1 1 0 0 1 2 2 0
0 0 2 2 3 3 3 2 1
1 3 1 2 2 2 2 2 1
1 0 0 0 2 1 2 3 2
3 0 0 0 1 1 1 0 0
0 2 2 2 3 1 0 3 0
3 1 2 1 1 0 2 1 2
```

Các vận động viên sẽ được đánh số theo số cột mà vận động viên đó xuất phát bắt đầu từ 1.



3. Chạy đua

Ví dụ vận động viên số 1 sẽ chạy theo đường được đánh dấu đỏ:

3 1 1 0 0 1 2 2 0

0 0 2 2 3 3 3 2 1

1 3 1 2 2 2 2 2 1

1 0 0 0 2 1 2 3 2

3 0 0 0 1 1 1 0 0

0 2 2 2 3 1 0 3 0

3 1 2 1 1 0 2 1 2

Ban đầu người đó xuất phát gặp ngay chướng ngại vật nên sẽ mất 3s để vượt qua, chạy thẳng tiếp gặp số 0 sẽ mất 1s, tiếp tục gặp số 1 là phải rẽ trái, tuy nhiên không thể rẽ trái trong trường hợp này nên vận động viên tiếp tục chạy thẳng và tốn 1s, tương tự gặp số 1 tiếp theo cũng mất 1s, tiếp theo gặp số 3 là chướng ngại vật mất 3s vượt qua, tiếp đến là số 0 mất 1s vượt qua, tiếp theo gặp số 3 là chướng ngại vật nên mất 3s.

Vì vậy tổng thời gian vận động viên số 1 chạy về đích là: $3 + 1 + 1 + 1 + 3 + 1 + 3 = 13s$

3. Chạy đua

Tương tự vận động viên số 2 sẽ có đường chạy như sau:

3 1 1 0 0 1 2 2 0

0 0 2 2 3 3 3 2 1

1 3 1 2 2 2 2 2 1

1 0 0 0 2 1 2 3 2

3 0 0 0 1 1 1 0 0

0 2 2 2 3 1 0 3 0

3 1 2 1 1 0 2 1 2

Tổng thời gian mà vận động viên 2 chạy về đích là: $2 + 1 + 1 + 1 + 3 + 1 + 3 = 12$ s

Tính tương tự với các vận động viên khác sẽ có thời gian về đích của các vận động viên lần lượt là: 12s, 13s, 14s, 15s, 15s, 14s, 9s

Như vậy, vận động viên số 9 là người về đích sớm nhất, kết quả in ra là 9.



3. Chạy đua

Sample input:

1

7 9

3 1 1 0 0 1 2 2 0

0 0 2 2 3 3 3 2 1

1 3 1 2 2 2 2 2 1

1 0 0 0 2 1 2 3 2

3 0 0 0 1 1 1 0 0

0 2 2 2 3 1 0 3 0

3 1 2 1 1 0 2 1 2

Sample output:

#1 9






4. Tìm ma trận

Cho một ma trận vuông A kích thước $N \times N$ ($5 \leq N \leq 100$) chứa các số nguyên nằm trong khoảng từ 1 đến 1000, số nguyên M ($2 \leq M \leq N-1$) là kích thước ma trận con của ma trận A và số nguyên K .

Hãy tìm vị trí bắt đầu của ma trận con trong ma trận A sao cho tổng tất cả các số trong ma trận con có giá trị gần nhất với số nguyên K . Nếu có 2 vị trí thỏa mãn thì lấy vị trí ma trận con có tổng nhỏ hơn. Nếu có 2 ma trận con có cùng tổng thì lấy ma trận con xuất hiện trước theo thứ tự duyệt từ trái qua phải từ trên xuống dưới của ma trận A .

Ví dụ: số nguyên $K = 50$, ta có 2 tổng của ma trận con là 25 và 75 thì ta sẽ in vị trí của ma trận có tổng là 25.

Nếu tổng của cả 2 ma trận đều là 25, nhưng một ma trận có chỉ số tọa độ điểm bắt đầu là (3,4), một ma trận có chỉ số tọa độ bắt đầu là (3,5) thì in vị trí (3,4).





4. Tìm ma trận

Input:

Dòng đầu tiên của input chứa số test case.

Các dòng tiếp theo sẽ chứa thông tin của test case, trong đó:

- Dòng đầu tiên của mỗi test case là 3 số lần lượt là N , M , K .
- N dòng tiếp theo là dữ liệu của ma trận A .

Output:

In ra kết quả là vị trí bắt đầu của ma trận thỏa mãn (vị trí bắt đầu là vị trí thuộc hàng đầu tiên và cột đầu tiên) theo dạng “#tc x y”.

Trong đó:

tc: số thứ tự của test case.

x: là số thứ tự của hàng trong ma trận A .

y: là số thứ tự của cột trong ma trận A .





4. Tìm ma trận

Sample input:

1 ← số test case

6 2 31500 ← N, M, K

616 610 65 167 894 75

510 301 696 574 590 162

173 969 359 32 269 427

511 423 775 780 911 553

183 392 496 765 875 365

903 256 461 475 973 822

Sample output:

#1 3 3



5 . QMAX

QMAX - Giá trị lớn nhất

Cho một dãy gồm n phần tử có giá trị ban đầu bằng 0.

Cho m phép biến đổi, mỗi phép có dạng (u, v, k) : tăng mỗi phần tử từ vị trí u đến vị trí v lên k đơn vị.

Cho q câu hỏi, mỗi câu có dạng (u, v) : cho biết phần tử có giá trị lớn nhất thuộc đoạn $[u, v]$

Giới hạn

$n, m, q \leq 50000$

$k > 0$

Giá trị của một phần tử luôn không vượt quá $2^{31}-1$

5.

Input

Dòng 1: n, m

m dòng tiếp theo, mỗi dòng chứa u, v, k cho biết một phép biến đổi

Dòng thứ $m+2$: p

p dòng tiếp theo, mỗi dòng chứa u, v cho biết một phép biến đổi

Output

Gồm p dòng chứa kết quả tương ứng cho từng câu hỏi.

Example

Input:

7 2

1 4 2

3 7 3

2

3 4

5 7

Output:

5

3

6. Mountain Walking

<https://vn.spoj.com/problems/MTWALK/>

MTWALK - Mountain Walking

Cho một bản đồ kích thước $N \times N$ ($2 \leq N \leq 100$), mỗi ô mang giá trị là độ cao của ô đó ($0 \leq \text{độ cao} \leq 110$). Bác John và bò Bessie đang ở ô trên trái (dòng 1, cột 1) và muốn đi đến cabin (dòng N , cột N). Họ có thể đi sang phải, trái, lên trên và xuống dưới nhưng không thể đi theo đường chéo. Hãy giúp bác John và bò Bessie tìm đường đi sao cho chênh lệch giữa điểm cao nhất và thấp nhất trên đường đi là nhỏ nhất.

Dữ liệu

Dòng 1: N

Dòng 2.. $N+1$: Mỗi dòng chứa N số nguyên, mỗi số cho biết cao độ của một ô.

Kết quả

Một số nguyên là chênh lệch cao độ nhỏ nhất.

6.

Ví dụ

Dữ liệu

5

1 1 3 6 8

1 2 2 5 5

4 4 0 3 3

8 0 2 3 4

4 3 0 2 1

Kết quả

2