

# JOI '20 Open P1 - Furniture

---

**Time limit:** 2.5s    **Memory limit:** 512M

---

JOI-kun's room is rectangular shaped. It is a grid of  $N \times M$  blocks. There are  $N$  horizontal rows. Each row is parallel to the east-west direction. There are  $M$  vertical columns. Each column is parallel to the south-north direction. The block in the  $i$ -th row from the north and the  $j$ -th column from the west ( $1 \leq i \leq N, 1 \leq j \leq M$ ) is denoted by the block  $(i, j)$ . Pieces of furniture are located in some of the blocks. For  $i, j$  ( $1 \leq i \leq N, 1 \leq j \leq M$ ), if  $C_{i,j} = 1$ , there is a piece of furniture in the block  $(i, j)$ . If  $C_{i,j} = 0$ , there is no furniture in the block  $(i, j)$ .

We say the arrangement of furniture is nice if we can travel from the block  $(1, 1)$  to the block  $(N, M)$  without passing through blocks with furniture by repeating travels from one block to another block which is in the south or the east direction. It is guaranteed that the initial arrangement of furniture is nice.

JOI-kun will perform  $Q$  operations. The  $k$ -th operation ( $1 \leq k \leq Q$ ) will be performed in the following way.

If the arrangement of furniture remains nice if a new piece of furniture is located in the block  $(X_k, Y_k)$ , then he locates a new piece of furniture in the block  $(X_k, Y_k)$ . Otherwise, he performs nothing.

Note that he will not perform an operation to a block where a piece of furniture is located initially, or where he already performed an operation. There is no furniture in the block  $(1, 1)$  and the block  $(N, M)$ , and he will not perform an operation in these blocks. Write a program which, given the size of the room, the initial arrangement of furniture, and the blocks where he will perform the operations, determines whether a piece of furniture is located or not for each operation.

## Input Specification

---

Read the following data from the standard input. All the values in the input are integers.

$N$   $M$

$C_{1,1}$   $C_{1,2}$   $\dots$   $C_{1,M}$

$C_{2,1}$   $C_{2,2}$   $\dots$   $C_{2,M}$

$\vdots$

$C_{N,1}$   $C_{N,2}$   $\dots$   $C_{N,M}$

$Q$

$X_1$   $Y_1$

$X_2$   $Y_2$

$\vdots$

$X_Q$   $Y_Q$

## Output Specification

---

Write  $Q$  lines to the standard output. The  $k$ -th line ( $1 \leq k \leq Q$ ) should contain  $\boxed{1}$  if JOI-kun locates a new piece of furniture in the block  $(X_k, Y_k)$  in the  $k$ -th operation. Otherwise, it should contain  $\boxed{0}$ .

## Constraints

---

- $1 \leq N \leq 1\,000$ .
- $1 \leq M \leq 1\,000$ .
- $0 \leq C_{i,j} \leq 1$  ( $1 \leq i \leq N, 1 \leq j \leq M$ ).
- $C_{1,1} = 0$ .
- $C_{N,M} = 0$ .
- The initial arrangement of furniture is nice.
- $1 \leq Q \leq N \times M$ .
- $1 \leq X_k \leq N$  ( $1 \leq k \leq Q$ ).
- $1 \leq Y_k \leq M$  ( $1 \leq k \leq Q$ ).
- $(X_k, Y_k) \neq (1, 1)$  ( $1 \leq k \leq Q$ ).
- $(X_k, Y_k) \neq (N, M)$  ( $1 \leq k \leq Q$ ).
- $C_{X_k, Y_k} \neq 1$  ( $1 \leq k \leq Q$ ).
- $(X_k, Y_k) \neq (X_l, Y_l)$  ( $1 \leq k < l \leq Q$ ).

## Subtasks

---

1. (5 points)  $N \leq 100, M \leq 100$ .
2. (95 points) No additional constraints.

## Sample Input 1

---

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

## Sample Output 1

---

```
0
1
0
```

## Explanation for Sample 1

---

The first operation is performed to the block (2, 2). If a new piece of furniture is located at the block (2, 2), then the arrangement will not remain nice. Therefore, JOI-kun does not actually locate a piece of furniture at the block (2, 2). The output should be 0.

The second operation is performed to the block (2, 1). If a new piece of furniture is located at the block (2, 1), we can travel the blocks (1, 1), (1, 2), (2, 2), (2, 3) in this order, for example, and the arrangement will remain nice. Therefore, JOI-kun locates a new piece of furniture at the block (2, 1). The output should be 1.

The third operation is performed to the block (1, 2). Since a piece of furniture is already located at the block (2, 1), the arrangement will not remain nice if a new piece of furniture is located at the block (1, 2). Therefore, the output should be 0.

## Sample Input 2

---

```
2 5
0 0 0 0 0
0 0 0 1 0
2
1 2
2 2
```

## Sample Output 2

---

```
0
1
```

## Explanation for Sample 2

---

The first operation is performed to the block (1, 2). If a new piece of furniture is located at the block (1, 2), then the arrangement will not remain nice. Therefore, JOI-kun does not actually locate a piece of furniture, and the output should be 0. Note that if we travel the blocks (1, 1), (2, 1), (2, 2), (2, 3), (1, 3), (1, 4), (1, 5), (2, 5) in this order, then we do not pass through blocks with furniture. However, since the travel from the block (2, 3) to the block (1, 3) is toward the north direction, it does not satisfy the condition of nice arrangements.

The second operation is performed to the block  $(2, 2)$ . If a new piece of furniture is located at the block  $(2, 2)$ , then the arrangement will remain nice since we can travel the blocks  $(1, 1)$ ,  $(1, 2)$ ,  $(1, 3)$ ,  $(1, 4)$ ,  $(1, 5)$ ,  $(2, 5)$  in this order. Therefore, JOI-kun locates a new piece of furniture at the block  $(2, 2)$ , and the output should be 1.