District 99

August 2022
C++ — 2 SEC — 512 MB

The latest Town Council meeting in Silhouettown produced a lot of food for thought. It was decided that the blue bins need a new shade, the lampposts need shortening, and, most importantly, the innumerable potholes need filling in.

Nobody in Silhouettown likes potholes, but nobody likes filling them in either. To solve this problem, the Town Council has formed a Pothole Action Team.

The main hurdle to filling in the potholes is the town's nebulous bureaucracy. Silhouettown is divided into many different rectangular districts (which are all parallel, by decree). Bylaw 534 states that each district must fill in all the potholes within its boundaries.

Unhelpfully, some of the districts overlap. Thus, some of the potholes lie within multiple different districts. For each pothole, the Action Team needs to know within how many districts it lies.

INPUT First, you will be given integers **p** and **d**, denoting the number of potholes and the number of districts respectively. This will be followed by **p** lines of 2 integers, **x y**, defining the location of each pothole. Finally, you will be given **d** lines of 4 integers, **x1 y1 x2 y2**, defining the bottom-left (**x1**, **y1**) and top-right (**x2**, **y2**) points of each district.

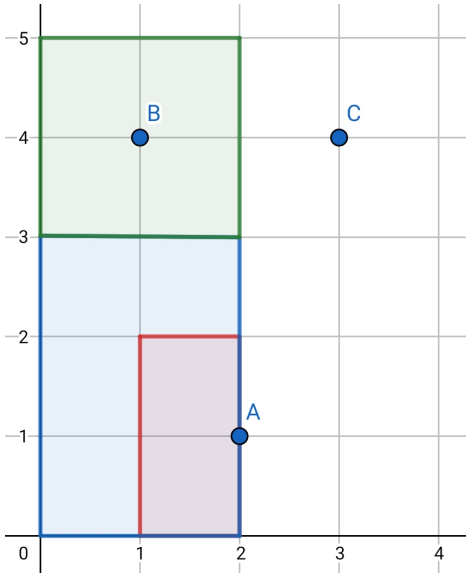
Note that the locations of potholes and districts are not necessarily unique.

$1 \leq \mathbf{p}, \mathbf{d} \leq 10{,}000$
$0 \leq \mathbf{x}, \mathbf{y} \leq 2^{30}$

OUTPUT For each pothole (in the order they have been given), output the number of districts that contain it, on a new line.

SAMPLE For example, suppose there are 3 potholes at (2, 1), (1, 4), and (3, 4); and there are 3 districts defined as (1, 0) – (2, 2), (0, 0) – (2, 3), and (0, 3) – (2, 5).

The first pothole is contained in 2 districts, the second in 1, and the third in 0. A diagram of the situation is given to the right.



INPUT

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

5 3
4 9
1 5
9 5
2 8
6 6
0 0 10 5
0 4 5 4
1 3 10 6

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

OUTPUT

```
1
3

0
2
2
0
1

1
3
1
2
```