Task 5: `Field`

Stuart the Snail lives on a field. The field can be described as an infinite plane (two-dimensional space). There are edible plants on every point of the field with integer coordinates. Stuart's house is located on the origin.

It is raining, so Stuart is able to move around easily. In one hour, he can choose one of the four edible plants adjacent to his current location, move there and have a snack. Formally, if he is at the coordinates (x, y), he may move to $(x + 1, y)$, $(x, y + 1)$, $(x - 1, y)$ or $(x, y - 1)$. He can continue his trip as long as it rains, but may also choose to stop at any time at the location of any edible plant, including simply staying at home.

However, it is known that there are n deep puddles in the field. Each puddle covers a rectangular region and is too deep for Stuart to pass through safely. Puddle i prevents Stuart from moving to any location with integer coordinates (x, y) satisfying both $a[i] \le x \le b[i]$ and $c[i] \le y \le d[i]$. Puddles may overlap.

It is not known how long the rain will last. Answer q queries of the same type defined by t.

- If $t = 1$, Stuart would like to visit the plant at the coordinates $(x[j], y[j])$. Find the minimum amount of time (in hours) required for him to reach $(x[j], y[j])$, assuming it never stops raining. If he cannot reach his destination, output -1 instead.
- If $t = 2$, suppose that the rain will last for $m[j]$ hours. Calculate the number of distinct locations that Stuart can end his trip in after at most $m[j]$ hours.

Input format

Your program must read from standard input.

The first line of input contains three integers n, t and q, representing the number of puddles, the type of query and the number of queries respectively.

Each of the next n lines contains four integers $a[i]$, $b[i]$, $c[i]$ and $d[i]$, describing a puddle.

If $t = 1$, each of the next q lines contains two integers $x[j]$ and $y[j]$, describing a destination.

Otherwise, if $t = 2$, each of the next q lines contains a single integer $m[j]$, describing the duration of the rain for one query.

Output format

Your program must print to standard output.

For each of the q queries, output a single integer on a new line representing the answer.

Subtasks

For all testcases, the input will satisfy the following bounds:

- $1 \leq n \leq 400$
- $1 \leq t \leq 2$
- $1 \leq q \leq 200\,000$
- $-10^9 \leq a[i] \leq b[i] \leq 10^9$
- $-10^9 \leq c[i] \leq d[i] \leq 10^9$
- $(0, 0)$ is not covered by any puddle.
- If $t = 1$, then $-10^9 \leq x[j], y[j] \leq 10^9$
- If $t = 2$, then $1 \leq m[j] \leq 10^9$

Your program will be tested on input instances that satisfy the following restrictions:

<table>
<tr><th>Subtask</th><th>Marks</th><th>Value of t</th><th>Additional Constraints</th></tr>
<tr><td>0</td><td>0</td><td colspan="2">Sample testcases</td></tr>
<tr><td>1</td><td>5</td><td rowspan="3">$t = 1$</td><td>$n \leq 100$
$-400 \leq a[i] \leq b[i] \leq 400$
$-400 \leq c[i] \leq d[i] \leq 400$
$-400 \leq x[j], y[j] \leq 400$</td></tr>
<tr><td>2</td><td>17</td><td>$n \leq 100$
$a[i] \equiv c[i] \equiv 0 \pmod{10^7}$
$b[i] \equiv d[i] \equiv -1 \pmod{10^7}$</td></tr>
<tr><td>3</td><td>8</td><td>$n \leq 100$</td></tr>
</table>

Subtask	Marks	Value of t	Additional Constraints
4	8	$t = 2$	$n \le 100, q \le 400$ $-400 \le a[i] \le b[i] \le 400$ $-400 \le c[i] \le d[i] \le 400$ $m[j] \le 400$
5	21		$n \le 100, q \le 400$ $a[i] \equiv c[i] \equiv 0 \pmod{10^7}$ $b[i] \equiv d[i] \equiv -1 \pmod{10^7}$
6	10		$n \le 100, q \le 400$
7	13		$n \le 100, q \le 5000$
8	14		$n \le 100$
9	4		No additional constraints

Sample Testcase 1

This testcase is valid for subtasks 1 and 3.

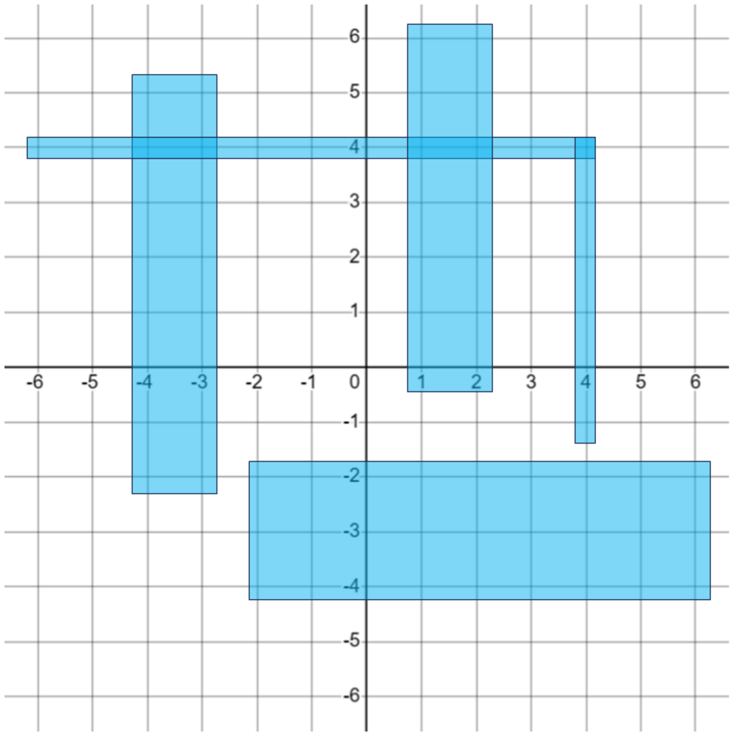
Input	Output

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

Sample Testcase 1 Explanation

The diagram on the next page describes the area around the origin. The blue rectangles represent puddles which cannot be entered or passed through.

It is not possible to reach the last two destinations without entering puddles. Note that the last destination is covered by a puddle.



Sample Testcase 2

This testcase is valid for subtasks 2 and 3.

Input	Output

`2 1 4`
`-1000000000 -1 0 999999999`
`0 999999999 -1000000000 -1`
`1 1`
`-1 1`
`-1 -1`
`1 -1`
	`2`
	`-1`
	`4000000002`
	`-1`

Sample Testcase 3

This testcase is valid for subtasks 4, 6, 7, 8 and 9.

Input	Output
2 2 6 -2 5 1 1 0 1 -3 -2 1 2 3 4 5 6	4 8 13 21 32 48

Sample Testcase 4

This testcase is valid for subtasks 5, 6, 7, 8 and 9.

Input	Output
2 2 4 0 9999999 -10000000 -1 -10000000 -1 10000000 29999999 12 1234 123456 12345678	235 2285986 22862261089 231374765559370