Everybody in the Casa Mare

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	1024 megabytes

Listen, brother, the whip may be cracking,
Yet the carriage never stops advancing.

Zdob și Zdub - "Everybody in the Casa Mare"

In the cartesian plane, there are initially n active points, with known coordinates.

You are given q queries, which you will have to process in order. Each query can be of either one of the following two types:

1. `! x y` — Toggle the state of the point at coordinates (x, y).

   If the point at (x, y) used to be active, it will now be inactive, and vice-versa.

2. `? x1 y1 x2 y2` — Determine the number of ways to get from an active point at coordinates (x_1, y_1) to another active point at coordinates (x_2, y_2) by only teleporting through active points.

   It is possible to teleport from an active point (a, b) to another active point (c, d) if and only if $a < c$. Since the answer can be very large, print its remainder modulo $10^9 + 7$.

Input

The first line of input contains a single integer n ($1 \leq n \leq 3 \cdot 10^5$) — the number of points which are initially active.

Each of the next n lines of input contain two integers x_i and y_i ($-10^9 \leq x_i, y_i \leq 10^9$) — the coordinates of the i-th active point.

It is guaranteed that all n active points are **pairwise distinct**.

The next line of input contains a single integer q ($1 \leq q \leq 3 \cdot 10^5$) — the number of queries.

The next q lines contain the descriptions of the queries, one line per query:

Each query can have one of the following two formats:

1. `! x y`. In this case, it is guaranteed that $-10^9 \leq x, y \leq 10^9$.
2. `? x1 y1 x2 y2`. In this case, it is guaranteed that $-10^9 \leq x_1, y_1, x_2, y_2 \leq 10^9$ and that both (x_1, y_1) and (x_2, y_2) are active points.

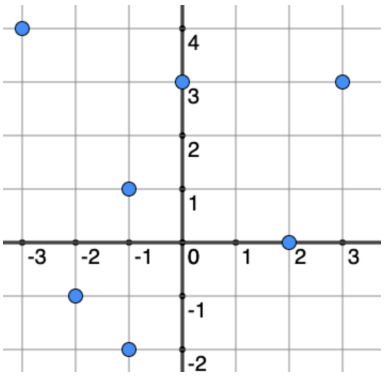
Output

For every query of the second type, print one integer, the number of ways to get from (x_1, y_1) to (x_2, y_2) by only teleporting through active points. Since this number can be very large, print its remainder modulo $10^9 + 7$.

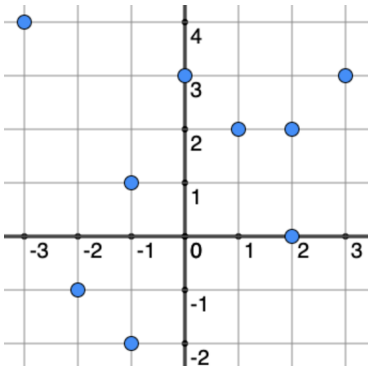
Example

standard input	standard output
7 -3 4 -2 -1 -1 -2 -1 1 0 3 2 0 3 3 9 ? -1 -2 3 3 ? 2 0 2 0 ? 3 3 -1 -2 ! 2 2 ! 1 2 ? -1 1 -1 -2 ? -2 -1 0 3 ! -2 -1 ? -3 4 3 3	4 1 0 0 3 36

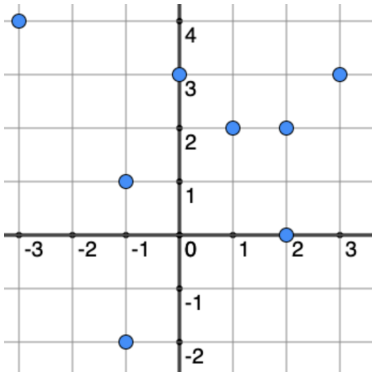
Note



The 7 initially active points



The active points after the fifth query



The active points after the eighth query

For the first query, the 4 paths from $(-1, -2)$ to $(3, 3)$ are:

1. $(-1, -2) \to (3, 3)$;
2. $(-1, -2) \to (0, 3) \to (3, 3)$;
3. $(-1, -2) \to (2, 0) \to (3, 3)$;
4. $(-1, -2) \to (0, 3) \to (2, 0) \to (3, 3)$

For the second query, the only way to get from $(2, 0)$ to $(2, 0)$ is to not teleport at all.

For the third query, it is impossible to get from $(3, 3)$ to $(-1, 2)$.

The fourth query toggles the state of point $(2, 2)$, making it active. Similarly, the fifth query activates point $(1, 2)$ and the eighth query deactivates point $(-2, -1)$.