

Intersection Query

Considering a set of segments. At the beginning, the set is totally empty.

There are **N** operations: insert or erase a *vertical* (or *horizontal*) segment and then you have to compute how many intersections are there.

There is no two same type of segments that share a common point in the set.

Input

The first line contains **N** ($1 \leq N \leq 10^5$) - the number of operations.

The following **N** lines describe the operations:

- **1 X₁ Y₁ X₂ Y₂**: insert a segment whose two endpoints are (**X₁**, **Y₁**) and (**X₂**, **Y₂**) to the set. ($|X_1|, |Y_1|, |X_2|, |Y_2| \leq 10^9$)
- **2 K**: erase the **K-th** oldest segment from the set. ($1 \leq K \leq$ current size of the set)

Output

For each query, print the number of intersections of the line segments in the set after processing the operation in one line.

Example

Input:

```
10
1 -1 0 -2 0
1 2 -2 0 -2
1 1 1 1 -3
2 1
1 6 -2 5 -2
2 3
1 2 -2 2 -6
1 -4 -3 -5 -3
1 3 -4 -1 -4
2 2
```

Output:

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

UPD (1/30/2015): Increase time limit from 2s to 5s. My C++ solution (unoptimized) runs in about 7 seconds.

