

Problem F. Antichamber (Division 1 Only!)

Input file: `input.txt`
Output file: `output.txt`
Time limit: 5 seconds
Memory limit: 256 mebibytes

In the Antichamber video game the player has a special tool called the Brick Tool. You must model its work in a simple two-dimensional variant.

The game takes place in an infinite grid. Each cell is either white or black. Two cells are considered *adjacent* if they share a side. A maximum connected set of black cells is called a *component*. The size of a component is the number of cells in it.

The tool allows the player to repaint any cell to the opposite color. Straight afterwards checks are performed which can change the state of the grid.

If a cell has been repainted to white and the number of components has increased, then some component has split into several *subcomponents*. In this case the newly formed subcomponents are removed except for, possibly, the largest of them. The largest subcomponent is not removed only in the case when its size is strictly greater than the sum of the sizes of all the other subcomponents. Removing a component means repainting all its cells to white.

Next regardless of the color of the repainted cell, «holes» in components are removed: If any white cell is located inside any cycle of black cells, it is repainted to black. Every two neighboring cells in a cycle must be adjacent in the sense defined above.

Initially all cells are white. A sequence of operations is given. Each operation is either repainting of a cell or a query. All the operations must be executed in the given order.

There can be two types of queries. A query of type **Comp**: is it true that the given two cells are black and belong to the same component? A query of type **Size**: find the size of the component that contains the given cell. If this cell is white, the answer must be zero.

Input

The first line contains a single integer N — the number of repaintings and queries ($1 \leq N \leq 10^5$). Each of the following N lines contains a description of a single operation.

The operation formats are presented below:

- “**Draw** x y ” — repaint a cell with the coordinates $(x; y)$ to the opposite color;
- “**Size** x y ” — find the size of the component containing the cell with the coordinates $(x; y)$;
- “**Comp** x_1 y_1 x_2 y_2 ” — determine whether the cells with the coordinates $(x_1; y_1)$ and $(x_2; y_2)$ belong to the same component.

All coordinates are positive integers not exceeding 10^6 .

Output

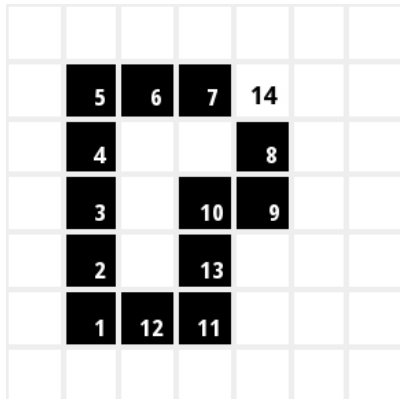
Print answers to the queries in order of their appearance in the input file, one answer per line. An answer to a query of the type **Size** must be an integer. An answer to a query of the type **Comp** must be either YES or NO.

Examples

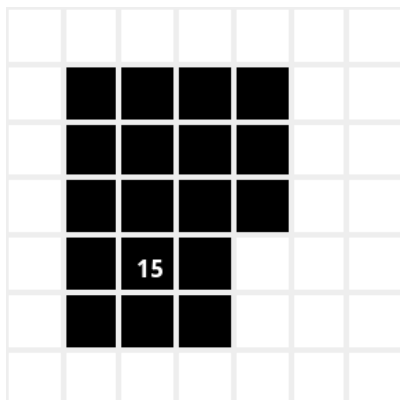
input.txt	output.txt
38	1
Draw 2 2	7
Draw 2 3	3
Draw 2 4	0
Draw 2 5	YES
Draw 2 6	NO
Draw 3 6	NO
Draw 4 6	13
Draw 5 5	18
Draw 5 4	18
Draw 4 4	YES
Draw 4 2	0
Size 4 2	9
Size 2 4	9
Size 4 4	0
Size 3 3	0
Comp 2 2 2 4	0
Comp 2 2 4 4	
Comp 3 3 4 4	
Draw 3 2	
Draw 4 3	
Size 2 2	
Draw 5 6	
Size 2 2	
Draw 3 3	
Size 3 3	
Comp 3 3 4 4	
Draw 2 4	
Draw 3 4	
Draw 4 4	
Size 3 2	
Size 3 6	
Draw 4 7	
Draw 3 5	
Size 2 5	
Draw 4 6	
Size 2 5	
Size 4 5	
Size 4 7	

Commentary

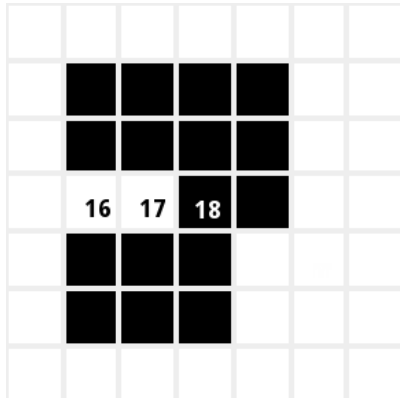
Numbers on the pictures below denote number of Brick tool usage for the corresponding number.



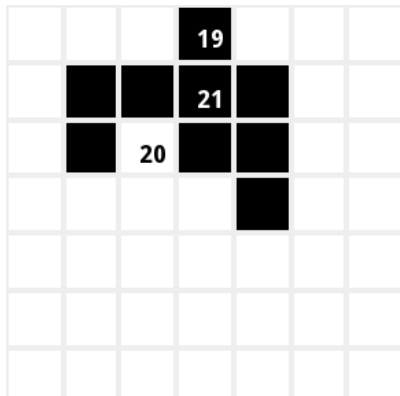
Playfield after 14'th usage of the Brick tool



15'th usage of the Brick tool does not cause changes in the playfield.



Playfield after 18'th usage of Brick tool.



After 21'th usage of Brick tool all the cells in the playfield will become white again.