

## Problem D. Weighting stones

Input file:               stones.in  
Output file:              stones.out  
Time limit:               1 second  
Memory limit:            256 megabytes  
Detailed Feedback:       none

Jack somehow found  $N$  stones and arranged them in increasing order of their weights. No two weights are equal. The lightest stone is given the rank 1, the next lightest — 2, and so on, the heaviest stone gets the rank  $N$ .

He has a balance scale and decided to put all the stones on it's sides in some order. It's known in which order he is going to put those stones on the scale and on which side each stone gets.

You have to determine the state of scale after each stone is added. Jack doesn't tell the exact weights of those stones.

### Input

The first line contains integer number  $N$  ( $1 \leq N \leq 100000$ ).

Each of the next  $N$  lines contains two integer numbers:  $R$  ( $1 \leq R \leq N$ ) and  $S$  ( $1 \leq S \leq 2$ ).  $R$  is the rank of the next stone which is put on side  $S$ . All  $R$ 's will be distinct.

### Output

Output  $N$  lines — one for each added stone. If after adding the corresponding stone side 1 is heavier, output "<". If side 2 is heavier, output ">". If it's not clear in which state the scale will be, output "?".

### Examples

stones.in	stones.out
5	<
1 2	>
3 1	>
2 1	?
4 2	>
5 1	