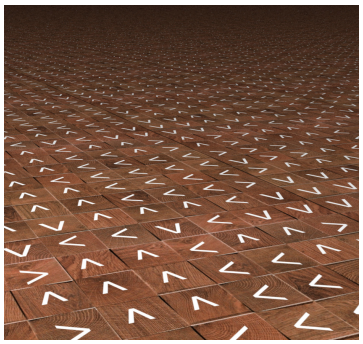
G Gridlock

Time limit: 6s



And fingers crossed that it is a Finite Plane of Characters...

The Foolish Puzzle Company posted an advertisement showcasing a new puzzle along with a very inefficient gameplay. Being taunted by the "99.99% cant solve these puzle" caption, you decided to download this puzzle. After wasting a good amount of time, you notice that the puzzles become larger and you are not having any kind of fun. You then decide to write a program to automatically solve the levels of this puzzle and skip the levels that are impossible.

A level of the puzzle gives you a rectangular grid with blocks. Each block contains an arrow that points in one of the cardinal directions (up, down, left, right). You can take out a block by sliding it in the direction that it points to. When sliding a block, you must slide it all the way out of the grid. You cannot slide it partially, slide another block, and go back to the same block to slide it out.

Given a grid, find a way to remove all the blocks, or state that it is impossible to take out all blocks.

Input

The input consists of:

- One line with two integers h and w ($1 \leq h, w \leq 2000$), the number of rows and the number of columns of the grid.
- h lines that contain strings of length w consisting of the characters "<", "^", ">", and "v", each character representing the direction written on the block.

Output

If it is possible to solve the grid, print $h \cdot w$ pairs of numbers y and x ($1 \leq y, x \leq n$), each pair representing a block that you take out from the grid, ordered from the first one to the last one. The two numbers in each pair represent the *row* and the *column* of the corresponding block, respectively.

If it is impossible to solve the grid, output "`impossible`".

If there are multiple valid solutions, you may output any one of them.

Sample Input 1	Sample Output 1
2 2 >v ^<	impossible

Sample Input 2	**Sample Output 2**
3 3 <<< <^< >>^	1 1 2 1 1 2 1 3 2 2 2 3 3 3 3 2 3 1