Problem E. Frogger

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	256 megabytes

You may have played the classic video game Frogger. In this game, a frog must cross a busy street without getting run over by one of the many fast cars. Then the frog must cross a river by jumping onto floating logs without falling in the water.

In this problem, we focus on the second part of the game, in which the frog crosses a river using floating logs. The game is played on a rectangular grid. Each log is the size of one square of the grid. Unlike in the real game, logs can float vertically in addition to horizontally.

This means that logs can also collide with each other. In this case, multiple logs can just be on the same grid square at the same time, and they just pass through each other without changing their direction and speed. When this happens, this is the only time that the frog can switch from one log to another. At any time, the frog is always on some log and travels with that log. If at any point in time, the log that the frog is on is on the same grid square as another log, the frog may choose to switch and continue travelling on the other log.

The input describes the state of the river at time $t = 0$. Specifically, the input defines the position of each log and and the direction that it travels in, either up (`^`), down (`v`), left (`<`), or right (`>`). It is guaranteed that at time $t = 0$, there is some log in the top-left corner of the grid, where the frog starts. At each time step, all the logs move simultaneously, at the same speed, by one grid square in their specified directions. The grid wraps around: when a log reaches an edge of the grid, its next position is at the opposite edge of the grid.

Determine the earliest time that the frog reaches the bottom-right corner of the grid, or that it can never reach it.

Input

The first line contains two space-separated integers $2 \leq R, C \leq 50$, the number of rows and columns in the grid. The next R lines each contain C characters each, either a period (`.`) indicating no log on that grid square, or one of `^v<>` indicating a log and the direction in which it travels.

Output

Output one line containing the minimum number of time steps for the frog to reach the bottom-right corner of the grid, or `-1` if it can never reach the bottom-right corner.

Examples

standard input	standard output

`3 5`
`>^..v`
`...<.`
`.v..^`
	`5`

`3 5`
`<....`
`.v...`
`...>.`
	`31`

`2 2`
`><`
`><`
	`-1`

Note

In the first sample the frog can go right-down-left-left-down to reach the bottom-right corner.