

Laser Maze

Standard mazes lose their mystery as one grows older. But throw in some lasers, and suddenly you've got yourself a recipe for cross-generational appeal. The object in any maze is to find your way from your starting point to some goal. In a *Laser Maze* you must additionally contend with laser turrets.

A laser turret is a stationary pillar that both blocks your movement and fires lasers from one side. Every time you take a step (either up, down, left, or right), every laser turret in the maze then rotates 90 degrees clockwise, and then shoots a momentary laser blast in the direction that it is facing. Needless to say, if you find yourself in the path of one of these lasers, you won't be around long enough to find a way out. A wall is a stationary pillar that blocks your movement, but does not fire lasers.

Lasers are powerful, but they do not pass through walls or laser turrets. The laser turrets respond to your movements, so you can't stand still and wait for the turrets to turn. If you reach the goal, but are immediately shot by a laser, your efforts will have been in vain, so make sure you reach the goal safely.

Input

Input begins with an integer **T**, the number of mazes you'll explore. For each maze, there is first a line containing two integers, **M** and **N**, the height and width of the maze, respectively. The next **M** lines contain **N** characters each, describing the maze:

. (empty space)
(wall)
S (starting position)
G (goal)
< > ^ v (laser turrets)

The four symbols for laser turrets signify turrets that are initially pointing left, right, up, or down respectively before you take your first step.

Output

For the *i*th maze, print a line containing "Case #*i*:" followed by the smallest number of steps necessary to get to the exit without being hit by a laser, or the string "impossible" if there is no way to reach the goal safely.

Constraints

$1 \leq T \leq 100$
 $1 \leq M, N \leq 100$
Each maze will contain exactly one 'S' and exactly one 'G'.

Example

Input:

```
5
2 5
##^##
S...G
2 5
##v##
S...G
1 5
S..G<
1 6
S...G<
5 5
S....
.....
.>v..
.^<..
```

....G

Output:

Case #1: 6

Case #2: 4

Case #3: 3

Case #4: impossible

Case #5: 8