Sweets, lots of sweets

Problem Statement

Georg has a new profession: he now works in a candy factory. The job is way more boring than it sounds: he is responsible for quality assurance. And that means taking a box of candies, counting them and checking that none are missing. Help poor Georg! Write a program that will do the job for him.

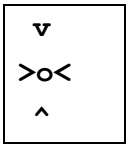
You will be given a matrix with *r* rows and *c* columns. This matrix represents the top view of a box of candies. The matrix will only contain the following characters:

- ".": a free spot
- "o": the edible part of the candy
- "<>v^": candy wrapper

There are exactly two ways how a whole piece of candy looks like:

1. >o<
2. v
   o
   ^

Whenever you see three characters arranged in this way, you see a whole piece of candy. You may assume that the following configuration will never appear in the input



Input

The first line of the input contains an integer *t* specifying the number of test cases. Each test case is preceded by a blank line.

The first line of each test case contains two integers *r* and *c* ($1 \le r,c \le 400$): the dimensions of the matrix. Each of the next *r* lines contains one row of the matrix: *c* characters from the set ".o<>v^". (Their ASCII values are 46, 111, 60, 62, 118, and 94.)

Output

For each test case, output a single line with one integer – the number of whole candies in the box.

Sample

Input	Output
2 5 4 .>o< v.^. ooo. ^.^. >o<< 1 3 >o<	3 1