Nice Binary Trees

Binary trees can sometimes be very difficult to work with. Fortunately, there is a class of trees with some really nice properties. A rooted binary tree is called "nice", if every node is either a leaf, or has exactly two children.

For example, the following tree is nice,

but the following tree is not.

The leaves of a nice binary tree are labeled by the letter 'l', and other nodes are labeled by the letter 'n'.

Given the pre-order traversal of a nice binary tree, you are required to find the depth of the tree.

Notes:

1. The depth of a tree is defined as the length of the longest path with one end at the root.
2. The pre-order traversal of the tree in the first image above produces the string "nlnnlll".

Input

The first line contains the number of test cases T. T lines follow. Each line contains a string, which represents the pre-order traversal of a "nice" binary tree. Leaves are represented by the letter 'l' and other nodes by the letter 'n'. The input is guaranteed to be the preorder traversal of a nice binary tree.

Output

Output one line for each test case, containing a single integer, the depth of tree.

Constraints

$0 < T < 20$

Length of the input string in each test case is at most 10000.

Example

Input:
3
l
nlnll
nlnnlll

Output:
0
2
3