

# A - Seinfeld

**Input:** standard input

**Output:** standard output

I'm out of stories. For years I've been writing stories, some rather silly, just to make simple problems look difficult and complex problems look easy. But, alas, not for this one.

You're given a non empty string made in its entirety from opening and closing braces. Your task is to find the minimum number of "*operations*" needed to make the string *stable*. The definition for being stable is as follows:

1. An empty string is stable.
2. If S is stable, then {S} is also stable.
3. If S and T are both stable, then ST (the concatenation of the two) is also stable.

All of these strings are stable: {}, {}, and {{{}}}; But none of these: }, {{{, nor {}. The only operation allowed on the string is to replace an opening brace with a closing brace, or visa-versa.

## Input

Your program will be tested on one or more data sets. Each data set is described on a single line. The line is a non-empty string of opening and closing braces and nothing else. No string has more than 2000 braces. All sequences are of even length.

The last line of the input is made of one or more '-' (minus signs).

## Output

For each test case, print the following line:

k. N

Where k is the test case number (starting at one,) and N is the minimum number of operations needed to convert the given string into a balanced one.

## Sample Input

```
}{
{}{}{}
{{{
---
```

## Sample Output

1. 2
2. 0
3. 1