Jutge.org

The Virtual Learning Environment for Computer Programming

How many sentences after a question with more a's than b'sX49928_en

The input of this exercise consists of several non-empty sentences made of a's and b's and ending in a punctuation sign, either . or ? or !. Each sentence will be in a different line to make the presentation clearer. For instance, this would be a possible input:

```
bbb?
aabba.
b.
bbbba?
abbaa?
aaabbb!
babbabb.
bbabb!
aa!
```

Sentences ending in . are called statements.
Sentences ending in ? are called questions.
Sentences ending in ! are called exclamations.
Implement a program that reads this input and outputs how many sentences there are which appear immediately after a question and also having strictly more as than bs.
In the previous example, the answer would be 2. Note that after question bbb? the next sentence is aabba., which has strictly more as than bs. Also, after the question bbbba? the next sentence abbaa? has more as than bs. Therefore, we count 2. Finally, after question abbaa? the sentence aaabbb! appears, but this one does not have strictly more as than bs.

Input

The input contains an arbitrary number of lines. Each line has one or more characters a or b, having as the last character a punctuation sign . or ? or !.

Output

For each input line, write an integer to the terminal, which is the total number of sentences that appear immediately after questions, and that have strictly more as than bs.

Sample input 1

```
bbb?
aabba.
b.
bbbba?
abbaa?
aaabbb!
babbabb.
bbabb!
aa!
```

Sample output 1

```
2
```

Sample input 2

```
abbbbaabbababb?
a?
ba.
aaabbbbaaa!
bababbbbabaab.
abaabaaab.
a.
babbbababababa!
abaa!
abbabaaaa?
aaabbaaa!
babaaab!
bbbababbbbbb!
bbabbababb!
babbbaaabababa?
bbbbaaababaa?
ababbbbbaa!
abb.
ab!
bbaabbab!
bbbaa?
aaabb!
abababa!
abbaa.
aab.
a!
aabbaaaababb?
a!
a?
abaaaaa.
```

Sample output 2

```
5
```

Observation

Massive storage solutions are not accepted (like `string`s or `vector`s). Read the input character by character. In particular, using `getline` or similar is forbidden, and the reason is obvious: in normal conditions, the sentences would not be separated by newline characters. In fact, the newlines are there only for presentation reasons. If you read characters with `cin >> c`, it is irrelevant if newline characters appear since `cin` ignores them.
Evaluation over 10 points:

- Slow solution: 5 points.
- Fast solution: 10 points.

We understand as fast solution one which is correct, has linear cost and passes the public and private tests. We understand as slow solution one which is not fast, but it is correct and passes the public tests.

Problem information

Author : PRO1
Generation : 2023-11-01 16:39:34