



Problem C: Food Restrictions

Those who study the moral traditions of society often find themselves adding to the moral code in an effort to help the less informed keep from becoming unacceptable. One of the more dynamic areas of moral codes is dietary guidelines. As new food items are introduced to a society, it is often necessary to instruct the faithful regarding the efficacy of various food combinations.

Write a program to determine how many, if any, dietary restrictions are being violated by a given combination of foods.

Input

The input will consist of a number of test cases, t , followed by the definition of each test case. Each test case contains a series of one or more food prohibitions followed by a line beginning with period (indicating end of prohibitions). An unknown number of test meals will then follow (one meal per line). A line beginning with a “#” indicates the end of the test case (also the end of the meals). The next test case begins on the following line.

The food prohibitions are expressed as follows:

```
hotdogs + frenchfries + -catsup
```

which indicates that hotdogs must be eaten with french-fries and catsup is not permitted, while

```
milk + cheese + -eggs
```

indicates that cheese must be eaten with milk, but no eggs can be served. The logic operators are “+”, and “-” meaning “and” and “not” respectively. The “not” has highest precedence, while “and” operators should be considered left to right.

Output

For each meal, print the number of restrictions violated, one number per line. If a rule contains a single non-negative food, containing the food in a meal, does NOT violate the rule. The rule is not able to be violated. For a rule "a + b + -c" if neither a nor b is present, and c is present in a meal, this does not violate the rule. Neither a rule nor recipe can include the same food more than once. If a rule contains only negative entries, any meal that has any foods from that rule has violated the rule.



2010 ACM ICPC South Central USA Regional Scripting Contest

Sample Input

```
3
hotdogs + frenchfries + -catsup
milk + cheese + - eggs
-chocolate
.
hotdogs hamburgers frenchfries milk cheese icecream
#
hamburgers+frenchfries+catsup
milk+cheese+-icecream+-buttermilk
.
frenchfries catsup buttermilk cheese
#
-eggs+milk+cheese
milk+cheese
.
eggs milk cheese
#
```

Sample Output

```
0
2
1
```