

Input File: MathSyntaxIn.txt

Nothing annoys Professor Plumb more than students who write improperly formed infix math expressions on his examinations. To avoid this annoyance, he has asked you to grade the exams, and you have decided to automate the grading.

According to the professor, infix math expressions always begin and end with a possibly signed numeric value (e.g., $-1.17.5$). In between these two numerics, there must be one or more pairs of a math operator followed by a numeric value (e.g., $+ -2.34$), and the numeric value can be a signed value. The four allowable operators are $+$, $-$, $*$ and $/$. All entities (numerics and operators) must be separated by *exactly one* space. For example:

$$-1.1 * -2.34 - 7.5 \quad \text{and} \quad 3 * 3.14159 - 18 + -6.6$$

are both valid infix expressions. The following four expressions are all invalid, and a carat (^) has been shown below the first invalid character in the expression when examining the expression from left to right.

1.1 + 7.5 **and** 1 + - 2 **and** 3.14 * 6.8 811 **and** / 2 + 3

extra space extra extra numeric non-numeric

Inputs:

The first line of input will be the number of math expression on the examination, **n**, followed by the math expressions to grade, each one on a separate line.

Outputs:

There will be one line output per math expression that contains the word `correct` for valid math expressions. For an invalid expression, output the character position (e.g., 1 for the first character, 2 for the second character, ...) of the first invalid character in the math expression when examining the expression from left to right. However, when the invalid expression ends with an operator, output the integer that is one more than the number of characters in the math expression (see the last sample input and output below.).

Sample Inputs

8
-1.1 * -2.34 - 7.5
3 * 3.14159 - 18 + -6.6
1.1 + 7.5
1 + - 2
3.14 * 6.8 811
3.14 x 6 / 6.845 8
/ 2 + 3
2 + 3 +

Sample Outputs:

correct
correct
7
5
12
6
1
8