

11th South African Regional ACM Collegiate Programming Contest

Sponsored by IBM

10 October 2009

Problem 4 - Purple Balloon Silly Sums

Problem Description

You have developed an automated system for evaluating the homework sums of a grade 1 class. As you can imagine, the sums are really quite simple at this level, and consist of only four entities: integers, the plus symbol, the minus symbol and the equals symbol.

The formal grammar of a valid equation is:

```
digit = "0" | "1" | "2" | "3" | "4" | "5" | "6" | "7" | "8" | "9" ;
posdigit = "1" | "2" | "3" | "4" | "5" | "6" | "7" | "8" | "9" ;
number = "0" | ["-"], posdigit, {digit} ;
operator = "+" | "-" ;
space = " " ;
expression = {space} , number , {space} , {operator} , {space} , number , {space} } ;
equation = expression , "=" , expression ;
```

where expressions between square brackets are optional (zero or one occurrence), and expressions between curly braces may occur zero or more times.

Examples of equations formed using this formal grammar include:

```
1--1=2
1 + 1 = 2 + 0
1- 1 = 2+ 0
```

The first two sums are correct, and the third is incorrect. Note that essentially all equations have a *left-hand side* and a *right-hand side*. Each side consists of numbers punctuated by either plus or minus operators, and the left- and right-hand sides are separated by an equals sign. A sum is correct if the left hand side is equal to the right hand side, and incorrect otherwise.

Your program must evaluate these expressions to determine whether they are correct.

Input

Your input will consist of an arbitrary number of equations (fewer than 100) conforming to the grammar specified above. Each line of your input will contain exactly one such equation, i.e., equations are separated by a newline character.

The end of input is indicated by a line containing only the value "0".

All numbers in the input equations will be in the range $[-2^{15}, 2^{15} - 1]$. The overall length of an expression will not exceed 1000 characters.

Output

For each input record, print out the line

`correct`

if the equation is correct, and

`incorrect`

if the equation is incorrect.

Sample Input

```
100+1000=999+101
100 + 1000 = 101 -1 +-1 +2 + 999
100 + 1000 = 101 +-1 +2 + 999
0
```

Sample Output

```
correct
correct
incorrect
```

Time Limit

3 seconds