

533 Equation Solver

Write a program that can solve linear equations with one variable.

Input

The input file will contain a number of equations, each one on a separate line. All equations are strings of less than 100 characters which strictly adhere to the following grammar (given in EBNF):

```

Equation  := Expression '=' Expression
Expression := Term { ('+' | '-') Term }
Term      := Factor { '*' Factor }
Factor    := Number | 'x' | '(' Expression ')'
Number    := Digit | Digit Number
Digit     := '0' | '1' | ... | '9'
    
```

Although the grammar would allow to construct non-linear equations like “ $x*x = 25$ ”, we guarantee that all equations occurring in the input file will be linear in x . We further guarantee that all sub-expressions of an equation will be linear in x too. That means, there won't be test cases like $x*x - x*x + x = 0$ which is a linear equation but contains non-linear sub-expressions ($x*x$).

Note that all numbers occurring in the input are non-negative integers, while the solution for x is a real number.

Output

For each test case, print a line saying ‘Equation # i ’ (where i is the number of the test case) and a line with one of the following answers:

- If the equation has no solution, print ‘No solution.’
- If the equation has infinitely many solutions, print ‘Infinitely many solutions.’
- If the equation has exactly one solution, print ‘ $x = solution$ ’ where *solution* is replaced by the appropriate real number (printed to six decimals).

Print a blank line after each test case, but the last one.

Sample Input

```

x+x+x=10
4*x+2=19
3*x=3*x+1+2+3
(42-6*7)*x=2*5-10
    
```

Sample Output

```

Equation #1
x = 3.333333
    
```

```

Equation #2
    
```

$x = 4.250000$

Equation #3
No solution.

Equation #4
Infinitely many solutions.