Home assignment

Instructions

Below you will find a set of problems that you can solve in any language, though Java is preferred. The only requirements is that you need to provide a README file on how to build and run your solutions. The end result should be provided in the form of a single public GitHub repository that contains the solutions to all the problems.

Unless explicitly stated otherwise; input will be provided through `stdin` and output should be printed to `stdout`. Lines should be delimited by the line feed (`\n`) character only.

This is your time to shine and show off your coding skills. Try to use best practices where applicable, to test your code and remember that readability matters.

Good luck and have fun!

Problem 1

You are given an array of n unique integers `a = a[0], a[1], ... , a[n - 1]` and an integer value `k`. Find and print the number of pairs `(a[i], a[j])` where `i < j` and `a[i] + a[j] = k`.

Input

The values `k, a[0], a[1], ... a[n - 1]`, one value per line. Empty lines should be ignored.

Output

Print the number of pairs matching the criteria.

Constraints

- All values are 32-bit signed integers
- `2 <= n < 10e6`

Example

Given `k = 6` and `a = [2, 1, 4, 5, 3]`. The pairs matching the criteria are `(2, 4)` and `(1, 5)`.

Input
6 2 1 4 5 3

The expected output is:

Output
2

Problem 2

Create a program that evaluates arithmetic expressions written in Polish notation. Expressions can contain double-precision floating point numbers and the following operations: addition, subtraction, division and multiplication.

Input

A list of expressions to evaluate, one expression per line.

Output

Print the result of the evaluation for each expression on separate lines. The result should be given with 2 digits of precision or "`error`" in the case that the expression was invalid.

Constraints

- Expressions can contain up to 100k operations

Example

Given the following list of expressions:

Input

```
+ + 0.5 1.5 * 4 10
- 2e3 - 700 + 7 * 2 15
- -1.5 * 3.1415 / -7 -2
100500
1 2
+ 1
```

The expected output is:

Output

```
42.00
1337.00
-12.50
100500.00
error
error
```

Problem 3

Implement a REST API for the solution to problem 2 that evaluates expressions that are supplied in the requests. Even though it may be a tiny API, be sure to provide everything you would expect from a production grade API.