Data Structures, Spring 2022

Assignment 02

Problem Description

In this problem, you have to implement a calculator like what has been taught in the classes. The calculator should support the following operator on double-precision floating points with correct precedence:

- arithmetic operators: +, -, *, /, ^, %, …
- parentheses: (,)

It is guaranteed that the divisor would not be zero during the process of the calculation.

Please do not try to solve this problem by calling other programs in your source code. If (and only if) you patiently, wholeheartedly code the homework out, you will gain a better coding skill and a deeper understanding of the data structure!

Input

The input contains *T* lines (multiple lines), each representing an arithmetic expression.

Output

For each test case print a double-precision floating point number in one line, which indicates the answer of the expression. Your solution will be considered correct if the absolute or relative error between the answer (a) and your output (b) is less than 10^{-8}, i.e., $\frac{|a-b|}{\max(1,|b|)} \leq 10^{-8}$.

Constraints

- $0 <$ the length of each line $L < 10^6$
- $0 < a_i < 10^8$ for each number a_i in the expression
- $L \cdot T \leq 10^6$
- Every number in the input will be an integer containing only of digits (no decimal points). We expect the final output to be a floating-point number, though.

Tasks

(a) (20%) **Implementation.** The operators include only {+, -}.

(b) (20%) **Implementation.** The operators include only {+, -, *, /}.

(c) (30%) **Implementation.** All operators are possible.

(d) (30%) **Discussion**. Describe the data structures used in your code. How does your program work in terms of performance? Is it possible to be improved using other strategies?

Sample Cases

Input	Output
1+3-2	2.0000000000000
1+2*3	7.0000000000000
1-2*3	-5.0000000000000
(1+2)*3	9.0000000000000