Bookmarks

- How To?
- Week 1
- Week 2
  - Computational Complexity. Linear Data Structures
  - **2nd Week Problems** due Nov 14, 2016 22:00 CET
  - 2nd Week Problems: Training
  - 2nd Week: Editorials
- Week 3
- Week 4
- Week 5

Week 2 > 2nd Week Problems > Postfix Notation

Postfix Notation

Bookmark this page

Postfix Notation

2.0/2.0 points (graded)

Input file:	postfix.in
Output file:	postfix.out
Time limit:	2 seconds
Memory limit:	256 megabytes

In *postfix notation* (or *reverse Polish notation*) an operation is written directly after its operands. For instance, a sum of two numbers, A + B, is written as A B +. The expression B C + D ∗ means (B + C) ∗D, and A B C + D ∗ + means A + (B + C) ∗D. (Here, we denote the multiplication operation as ∗). An advantage of this notation is that it does not require neither parentheses nor agreements on operators' priority.

Given an expression in postfix notation, evaluate it.

Input

The first and only line of the input file contains an expression in postfix notation, which can contain single-digit non-negative numbers and operations +, -, ∗. The line contains at most 10^6 numbers and/or operations. Every two subsequent entities are separated by a single whitespace. It is guaranteed that the expression is correct and is evaluated to a single number.

Output

Output the result of evaluation of this expression. It is guaranteed that the result, as well as any intermediate value, will be less than 2^{31} by the absolute value.

Example

postfix.in	postfix.out
8 9 + 1 7 - *	-102

Download Download

Choose Files No file chosen

Accepted

Submit You have used 1 of 200 attempts

Discussion
Topic: 04: 2nd Week Problems / Postfix Notation

Show Discussion