# Postfix Notation

**Implementation Practice 1**

We are familiar with infix notation for representing expressions, where the operator is placed infix between the operands, as in `5 * 5`. Given an expression in postfix notation, such as `5 5 *`, evaluate the expression and print it to standard output, rounded to one decimal place.

Valid operands are `*` (multiplication), `/` (division), `+` (addition), `-` (subtraction), `%` (mod), and `^` (exponentiation).

## Input Format

A valid postfix expression. The input will have no more than $50$ characters, and the value of each number in the input and each intermediate result will be less than or equal to $1000$.

## Output Format

The result of the evaluation. The answer will be considered correct if its absolute or relative error does not exceed $10^{-1}$.

## Sample Input

```
5 5 +
```

## Sample Output

```
10.0
```

## Sample Input

```
5 5 + 6 * 8 -
```

## Sample Output

```
52.0
```