Senior Software Engineer
Assignment:

Using the programming language of your choice among C, C++, and Python, implement a simple calculator that satisfies these conditions:

- The calculator is run without any command-line arguments.

- The calculator reads lines from the standard input. Each line is an arithmetic expression (see below). The calculator evaluates the expression and prints the result to the standard output on a single line.

- The calculator exits with a zero exit code upon reading an end-of-file.

- An arithmetic expression consists of ASCII characters and has this BNF syntax:

```
Expression ::= Term
          | Term AddOp Term

AddOp ::= '+'
       | '-'

Term ::= Factor
      | Factor MulOp Factor

MulOp ::= '*'
       | '/'
       | '%'

Factor ::= '(' Expression ')'
        | '-' Factor
        | Number

Number ::= Digit
        | Digit Number

Digit ::= '0' | '1' | '2' | '3' | '4' | '5' | '6' | '7' | '8' | '9'
```

- Whitespace is not allowed in the expression.

- The division operation ('/') rounds down to the nearest integer; eg,
8/3 => 2, -8/3 => -3. The '%' sign is the remainder; eg, 8%3 => 2,
-8%3 => 1.

- If an input line contains a syntax or arithmetic error, the calculator prints out a short error message to the diagnostic output and continues running.

An example session:

```
==========================================================================
$ calc
(1+3)*(9/3)
12
0-123
-123
^D
$ echo $?
0
$
==========================================================================
```