Team 2

# Calculatorator

# User's Manual

## Version 1.0

*[Note: The following template is provided for use with the Unified Process for EDUcation. Text enclosed in square brackets and displayed in blue italics (style=InfoBlue) is included to provide guidance to the author and should be deleted before publishing the document. A paragraph entered following this style will automatically be set to normal (style=Body Text).]*

*[To customize automatic fields (which display a gray background when selected), select File>Properties and replace the Title, Subject and Company fields with the appropriate information for this document. After closing the dialog, automatic fields may be updated throughout the document by selecting Edit>Select All (or Ctrl-A) and pressing F9, or simply click on the field and press F9. This must be done separately for Headers and Footers. Alt-F9 will toggle between displaying the field names and the field contents. See Word help for more information on working with fields.]*

# Revision History

| Date | Version | Description | Author |
|---|---|---|---|
| <dd/mmm/yy> | <x.x> | <details> | <name> |
| 12/12/24 | 1.0 | Added section 1-5 | Riley Anderson |
| | | | |
| | | | |

# Table of Contents

# User Manual

## 1. Purpose

*The user manual for your software product should be an easy-to-understand guide on how to use the software. It should include the following sections.*

This manual lays out everything you will need to know to use the calculatorator.

## 2. Introduction

*This section should provide a brief overview of the software, including its purpose, features, and how to install and run it.*

The software is a command-line tool for evaluating arithmetic expressions. It can handle both positive and negative numbers, and all common operators (+, -, *, /, **, %), as well as grouping and operator precedence. This program requires a working installation of C++.

## 3. Getting started

*This section should provide a step-by-step guide on how to use the software to evaluate arithmetic expressions. It should include instructions on how to enter expressions, how to use the various operators and functions, and how to interpret the results.*

To download the software, simply clone the github repo to your machine in terminal with the command below.

git clone https://github.com/samkelemen/348GroupProject.git

After this, you can enter the folder with:

cd 348GroupProject

From within this folder, you can compile and start the program with:

make

Once the program is started, you may enter any expression you choose to evaluate with the previously outlined operators. Do not put spaces anywhere in the input.
To quit, simply enter Ctrl-C. To restart the program, you can either rebuild or enter:

./main

## 4. Advanced features

*This section should describe any advanced features of the software, such as the ability to save and load expressions, or to define custom variables and functions.*

**N/A**

## 5. Troubleshooting

*This section should provide a list of common problems, if any, that users may encounter, and how to solve them.*

If you are experiencing an error, your first course of action should be to restart the program, pull the newest version of the software, and rebuild. This usually fixes the issue. If you continue to experience errors, verify that your input does not contain any invalid characters, does not contain any spaces, and any parenthesis used are closed.

## 6. Examples

*This section should provide examples of how to use the software to evaluate different types of arithmetic expressions.*

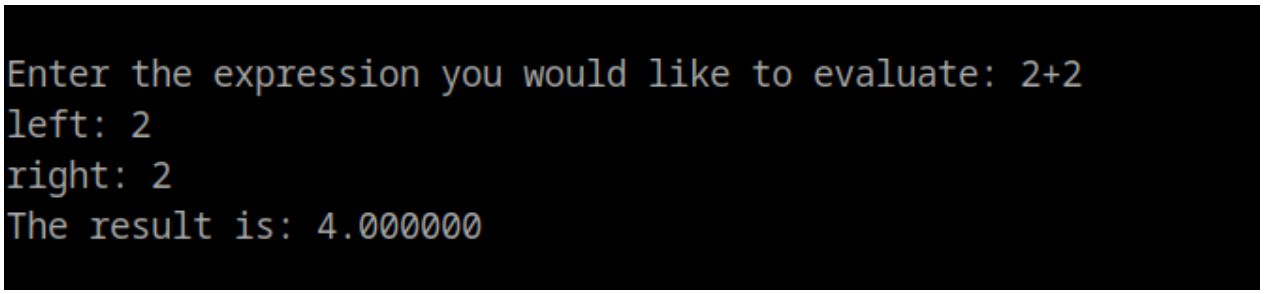


Fig1: User enters "2+2" without the output as 4

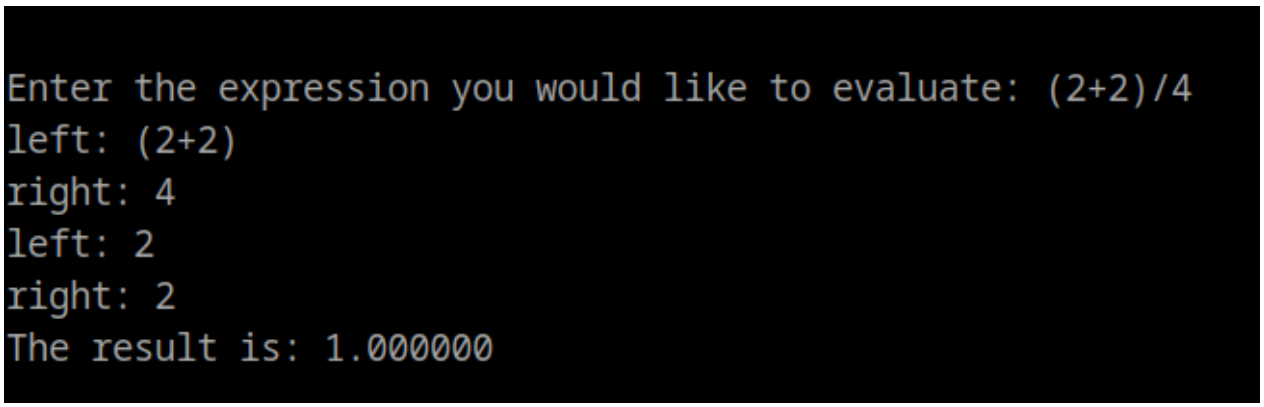


Fig2: User enters "(2+2)/4" without the output as 1

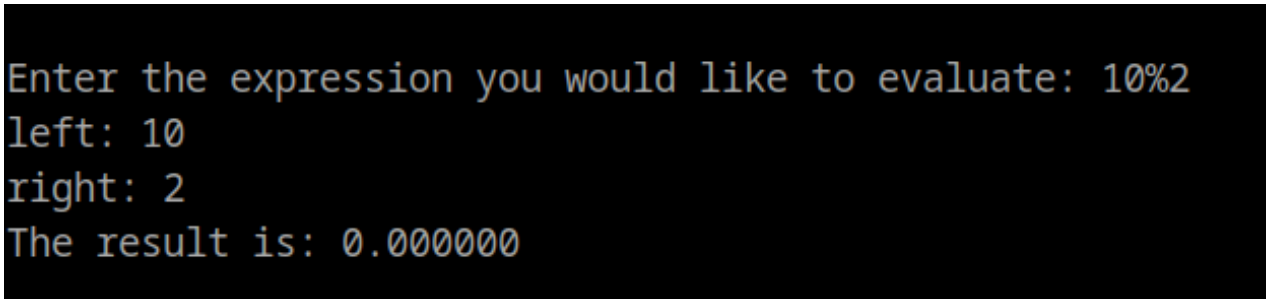


Fig3: User enters "10%2" without the output as 0

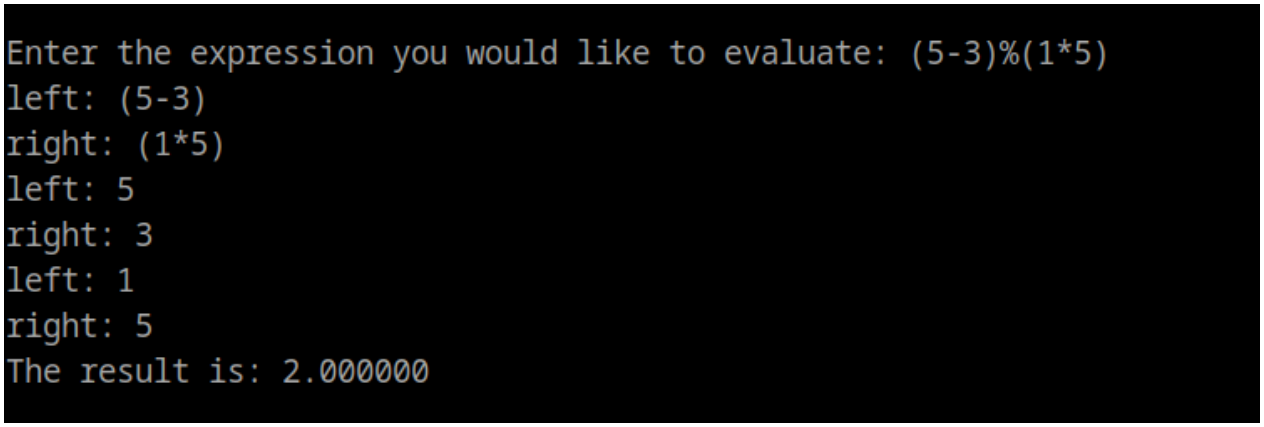


Fig4: User enters "(5-3)%(1*5)" without the output as 2

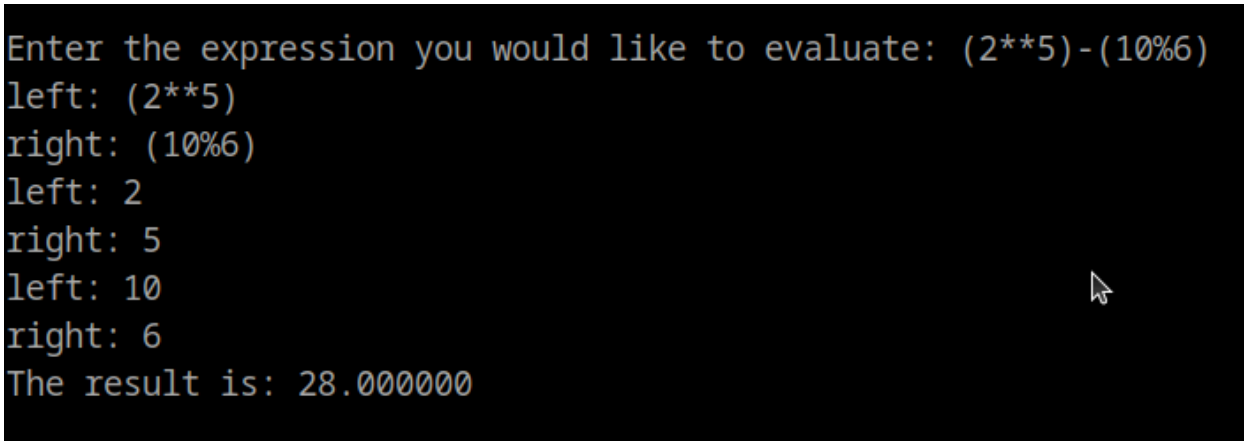


Fig5: User enters "(2**5)-(10%6)" with the output being 28

## 7. Glossary of terms

*This section should define any technical terms that are used in the manual.*

**Command-line tool:** You enter your "math problem" in the terminal as one line utilizing the syntax we provided

**Arithmetic expression:** A mathematical phrase that can include numbers, operators, and grouping symbols to represent a calculation.

**Operator:** A symbol (such as +, -, *, /) that tells the software to perform a specific mathematical operation.

**Precedence:** The rules that define the order in which different operations are performed within an arithmetic expression.

**Grouping symbols:** Parentheses () that indicate which parts of the expression should be evaluated first.

## 8. FAQ

*This section should answer frequently asked questions about the software.*

**Q: What should I do if the software doesn't start?** A: Ensure that you have a working installation of C++ on your machine. If the problem persists, try pulling the latest version from the GitHub repository and rebuilding the software.

**Q: How do I input an arithmetic expression?** A: Enter the expression directly into the command line without any spaces. For example, to evaluate "2 + 2," you would enter "2+2".

**Q: Can I use spaces in my input?** A: Yes, you can include as many spaces as you like in your input. The software will process the expression correctly regardless of spaces.

**Q: What if I enter an invalid character by mistake?** A: If you accidentally enter an invalid character, the software will handle the expression and return an error, and will not crash. However, your expression will not be saved and you will need to retype your entire expression (or copy-pasted). You will be allowed to re-enter the expression with the correct characters and ensure that your expression is syntactically correct.

**Q: Is there a way to save or load expressions?** A: Currently, the software does not support saving or loading expressions. You must re-enter each expression manually each time you start the program.