Simple Calc

Cyber Solutions Development - Georgia

September 3, 2020

Abstract

Your task is build a simple calculator application that will take an equation as an argument, and produce the result to standard out.

1 Requirements

In this assignment, you will build a simple calculator application that will compute simple equations and produce the result. The requirements are below:

1.1 Basic Requirements

1. Written in C
2. Take a single argument that is an equation of the form (operand1) (operator) (operand2)
3. Handle bad inputs such as bad format as described above, divide by zero, or interger overflow
4. Single binary with the usage statement ./simplecalc ¡equation¿

1.2 Specific Requirements

1.2.1 Required Operators

1. Addition
2. Subtraction
3. Multiplication
4. Division
5. Modulo
6. Left shift
7. Right shift

8. And
9. Or
10. XOR
11. Rotate Left
12. Rotate Right

2 Deliverables

Your code should have the following file structure:

```
SimpleCalc
├── src
│   └── source-files
├── hdrs
│   └── header-files
├── docs
│   └── documentation
└── CMakeLists.txt
```

Your code should build and compile with the following shell script ran from the SimpleCalc directory:

```
// build.sh
mkdir build
cd build
cmake ..
```

3 Notes to grader

The purpose of this assignment is NOT to write fancy C code. Use this assignment to achieve the following objectives:

1. Hammer down on the coding standard. By the end of the assignment your mentee should understand the coding standard.
2. Code organization. Ensure your mentee does not implement the solving functionality in main(). Encourage them to implement separate functions for each operator. This will help them throughout the next assignments.
3. Building projects with CMake. CMake is a powerful build system, but requires a learning curve. Use this project to introduce them to CMake.
4. Git tradecraft. Have your mentee branch off of devel, commit their work, submit a merge request to devel, etc. Make devel a protected branch.

4 JQR Sections Covered

- 3.1.3 (all)
- 3.1.5 (all)
- 3.1.6 (a, b)
- 3.1.8 (a, c, d, e)
- 3.1.9