Group 4

Arithmetic Expression Evaluator
Software Requirements Specifications

Version 1.0

Revision History

Date	Version	Description	Author
9/25/2024	0.0	Initial meeting, discussion of meeting times, and assigning roles for the project	Alexander Carrillo
10/16/2024	1.0	Software requirements discussion, reviewing rough code, talking about next steps	Alexander Carrillo

Table of Contents

Software Requirements Specifications

1. Introduction

The Arithmetic Expression Evaluator aims to be a user interface calculator that can handle operators for addition, subtraction, multiplication, division, exponents, modulus, and order of operations. Users will be able to input an equation using understandable arithmetic and then receive an output. PEMDAS will be used in the document and further ones to describe the process of the order of operations.

1.1 Purpose

This software requirements specifications document will show the requirements, functional and nonfunctional, and constraints for the project.

1.2 Scope

The goal is to develop an easy-to-use arithmetic calculator that can do multiple operations, handle exceptions, and do the correct order of operations.

Use cases would be the user wanting to perform an arithmetic problem.

Requirements would be operators, user interface, error handling, and order of operations.

Nonessential requirements would be multiple types of parentheses, and rounding up decimals to the second place.

1.3 Definitions, Acronyms, and Abbreviations

PEMDAS: describes the process of the order of operations, parentheses, exponents, multiplication, division, addition, and subtraction. in that order

Operators: the symbols that the program will read from the user input, +, -, *, **, /, %

1.4 References

"Other documents" refers to the previous software development plan, or project plan

in future documents it will refer to previous documents which can be found here (github link)

1.5 Overview

The rest of the document describes requirements, functionality, and use case specifications.

2. Overall Description

1.6 Product perspective

1.6.1 System Interfaces

1.6.1.1 Arithmetic calculator code
1.6.1.2 user interface code

1.6.2 User Interfaces

1.6.2.1 console input (x + y)

1.6.3 Hardware Interfaces

1.6.3.1 Computer

1.6.4 Software Interfaces

1.6.4.1 computer

1.6.5 Communication Interfaces

1.6.6 Memory Constraints

1.6.6.1 Integer limit of C++: 2147483647

1.6.7 Operations

1.6.7.1 addition
1.6.7.2 subtraction
1.6.7.3 multiplication
1.6.7.4 division
1.6.7.5 exponents
1.6.7.6 modulation

1.7 Product functions

1.7.1 arithmetic operations
1.7.2 parentheses
1.7.3 PEMDAS

1.8 User characteristics

1.8.1 Computer accessible, keyboard usable

1.9 Constraints

1.9.1 Limited operators, no trig calculations
1.9.2 working within c++

1.10 Assumptions and dependencies

1.10.1 working computer
1.10.2 able to run c++
1.10.3 user limited to operators given (add, sub, multi, etc)

1.11 Requirements subsets

1.11.1 Essential
1.11.1.1 Operators
1.11.1.2 error handling
1.11.1.3 user interface
1.11.2 Desirables
1.11.2.1 Rounding up for decimals
1.11.2.2 Different parentheses

3. Specific Requirements

1.12 Functionality

1.12.1 Operations

- able to recognize and then perform operations such as addition, subtraction, multiplication, division, modulation, and exponentes

1.12.2 User Interface

- User is able to input arithmetic problems such as 2 + 2 (most basic) or 2+4*2/((5-2)/4) (more advanced) and then be shown the correct output

1.12.3 Order of operations

- arithmetic problems are done in the PEMDAS order of operations, parentheses > exponent > multiplication

& division > addition and subtraction

1.12.4 Exception handling

- If the user inputs something that can not be computed the program should not crash and should instead give an error message and the give the user another chance for an input

1.12.5 multiple kinds of parentheses

- Allows for the use of [] and {} as substitutes for parentheses

1.13 Use-Case Specifications

Basic Example: The user inputs 7+4, the system splits into two ints, and the operator reads and calls the addition function. the function returns the sum of 7 and 4, 11, then shows it in the console

Parentheses and Pemdas example: The user inputs 2+4*5+(2-1), the system breaks it into parts starting with what is in the parentheses, calls subtract, and returns 1. Next is multiplication, call multiply, and returns 20. next, we just go from left to right, doing: 2+20 = 22, 22+ 1 =23. shows 23.

you could break down use cases to each operator (addition, multiplication, etc.) and combinations of them, the most basic use case for the calculator would be, “do an arithmetic problem”, then branching off into specifics from there.

1.14 Supplementary Requirements

Desirable: Rounding up to two decimal places, Multiple types of parentheses

4. Classification of Functional Requirements

Functionality	Type
Operators (add, sub, mult, div, mod, ex)	Essential
User interfacing	Essential
Order of Operations (PEMDAS)	Essential
Multiple types of parentheses ([], {}, etc.)	Desirable
Exception handling (dividing by zero, unmatched parentheses)	Essential
rounding up decimals to the second place	Desirable

5. Appendices

(1) Project Plan