

CSC 413 Project Documentation

Spring 2022

Pragati Makani

920821704

CSC 413.02

<https://github.com/csc413-SFSU-Souza/csc413-p1-pragati-e>

Table of Contents

1	Introduction	3
1.1	Project Overview.....	3
1.2	Technical Overview	3
1.3	Summary of Work Completed	4
2	Development Environment.....	4
3	How to Build/Import your Project	4
4	How to Run your Project.....	4
5	Assumption Made.....	4
6	Implementation Discussion	5
6.1	Class Diagram.....	5
7	Project Reflection.....	5
8	Project Conclusion/Results	5

1 Introduction

Assignment 1 - Expression Evaluator and Calculator GUI

1.1 Project Overview

This Project is for us to Practice Object Oriented design. The requirement is to create an object to evaluate mathematical expressions in a GUI from the first part

1.2 Technical Overview

All the member variables I used are private so that its scope is restricted to a particular class. The main class of this assignment is the EvaluatorDriver and this class calls the evaluateExpression() method of the Evaluator class in order to compute the expression. We also have an operand class which evaluates whether the given string is an operand or not. Since the operator class is an abstract class, I wrote other classes such as AddOperator and the rest to do the function, when the operand is given. When the evaluateExpression() is called, I used the .trim() method of the String class to remove spaces on either sides and in between the expression. My next step was to check if the expression doesn't contain uneven number of brackets, consecutive operators and checking if the given expression contains non-numeric digits. (Well, these are the exception cases). Converted the obtained string to a char array and checked the number of operators present. Check for brackets and obtain the innermost bracket. If the innermost bracket has a valid expression, we will use substring from the start to the end of the bracket. If operatorCount is greater than 1, it would call the evaluateExpression() method again or else the value is computed using computeSingleExpr() method which has the actual implementation to evaluate an expression which has 2 operands (1 operator). If not since this method is called recursively in the bracketEvaluator(), the next check takes place. Here I have placed the evaluation according to the priority. If it satisfies the priority condition evalSubExpr() method is called, which will again call the computeSingleExpr() after manipulating the expression. I have implemented this assignment in a different way because I changed the starter code in such a way that it can evaluate an expression having 1 operation and I put this in the computeSingleExpr() method. I have also used additional methods to reuse the code again:

operatorCheck()

checkChar()

operatorCount()

bracketCount()

reverseString()

operatorList()

operandCount()

compareBracket()

1.3 Summary of Work Completed

I completed the assigned task in accordance with the specifications, and most of my test cases passed. My code will accept an expression in the form of a string and then utilize string manipulation and the base code provided to execute the required mathematical function.

2 Development Environment

A development environment is a set of tools used to create, test, and debug an application or software. IntelliJ was the IDE I used to create this project.

3 How to Build/Import your Project

A development environment is a set of techniques and tools used to create, test, and debug an application or software. The first thing I did was clone my git repository. Then, in IntelliJ, I created a java project and transferred the src into it. I used the build path option to add the necessary jar files to Junit to execute the test cases.

4 How to Run your Project

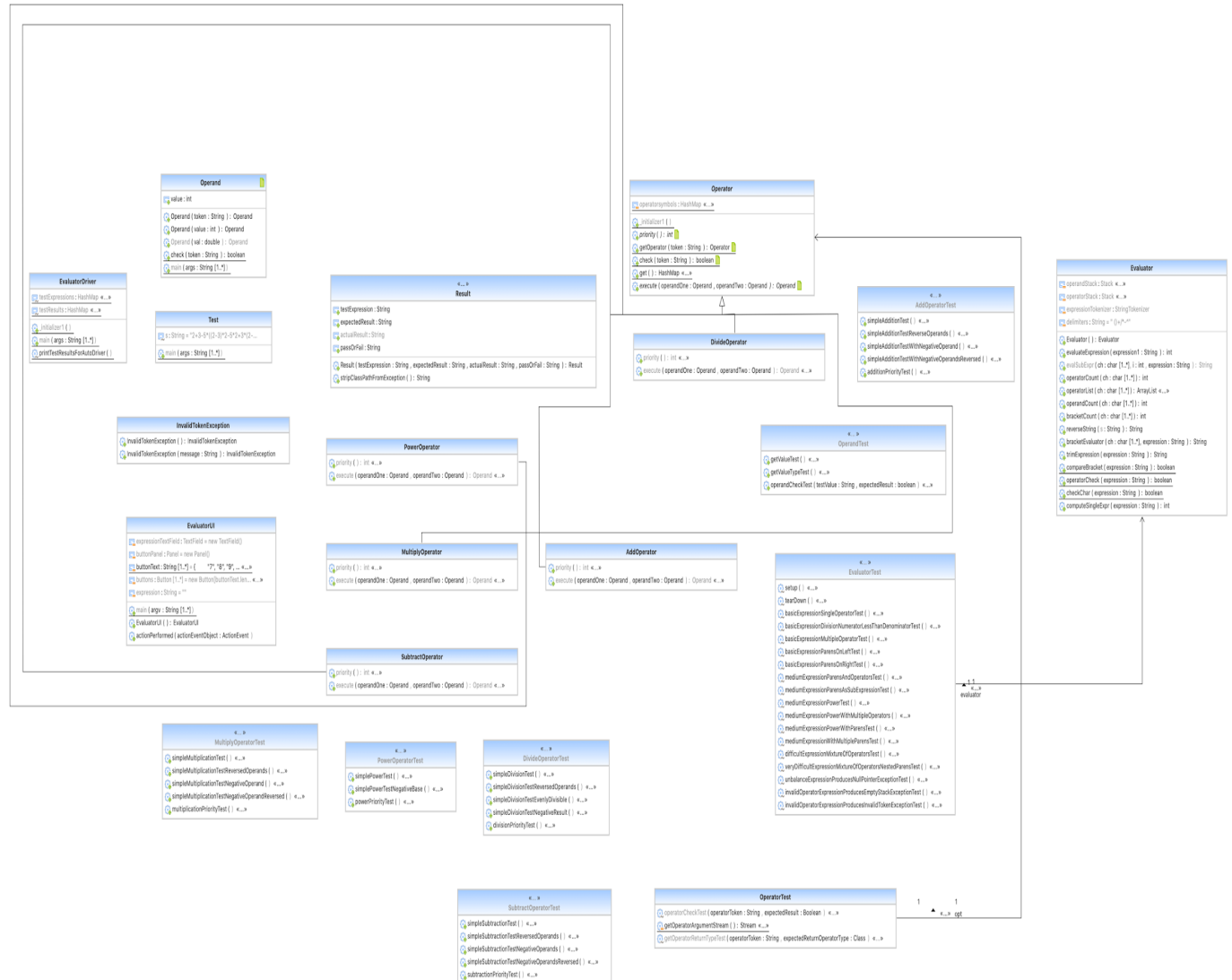
Any project should not have compile time errors to execute. So, in order to complete the job, I repaired such flaws.

5 Assumption Made

None

6 Implementation Discussion

6.1 Class Diagram



7 Project Reflection

This project helped me grasp Object Oriented Design and the benefits of utilizing objects and classes. This assignment also taught me to think more logically because I spent a lot of time considering how to approach the issue.

8 Project Conclusion/Results

Finally, I realized that solving an issue requires work and rational thought. Except for the longest one, I was able to solve all the offered equations. I completed the task flawlessly. I am aware that I changed the underlying implementation and relocated it to a different function.