# *CSC 413 Project Documentation*

***Summer 2020***

***Javier Gonzalez***

***916582412***

***CSC413.01***

***https://github.com/csc413-01-SU2020/csc413-p1-gonzalezjavier***

## Table of Contents

# 1 Introduction

## 1.1 Project Overview

This project is a simple calculator that allows the user to input an expression to be evaluated. The input of the user is stored and ran though the correct mathematical functions required to achieve the expected result.

## 1.2 Technical Overview

This project is a calculator that also has a GUI for a clean and easy expression input. There is an evaluator class that is what takes the user input and splits the expression into tokens in order for the expression to be stored into stacks. From theses stacks, we are able to pop out operands and operators to complete the calculation needed and in the correct order. An operator abstract class was used, and the child classes were created in order for each individual operator to have the mathematical instructions specific to them. HashMaps were used in the abstract class to create instances of each operator, making it easier to add more operators in the future.

## 1.3 Summary of Work Completed

The entire project runs and provides the correct calculations when an expression is entered by the user. I was able to complete the project by following the requirements in the assignment printout and by following the suggestions in the comments within the code. I was able to finish the operand class quite fast and from there went to the operator class. The subclasses were then created which was fairly easy as well. I was able to get the evaluator class complete by following the basic ideas of completing an equation with the given operators. The GUI was also completed by adding the actions for the pressed buttons and by adding text to the text field that would show the expression and result.

# 2 Development Environment

During the development of this project I used the IntelliJ IDEA with the version 13 of Java.

# 3 How to Build/Import your Project

Using the IntelliJ IDEA, to import the project you can open the IDE and click ‘Open or Import’ button. From there you can go the project folder and click on the ‘Calculator’ folder. Select this folder to import. If the IDE is already open, click from the file menu and use the ‘New Project from Existing Resources’ option. From here you can import the project by clicking the same ‘Calculator’ folder. It then will show the required libraries needed and include them in the project. A JDK will also need to be installed and you will be given the choice to install one if needed. At this point you should be able to complete the import of the project.

# 4 How to Run your Project

In order to run the calculator, you should be in the ‘EvaluatorUI’ class and press the ‘Run’ button. This launches the GUI for you to put in an expression and get the result.

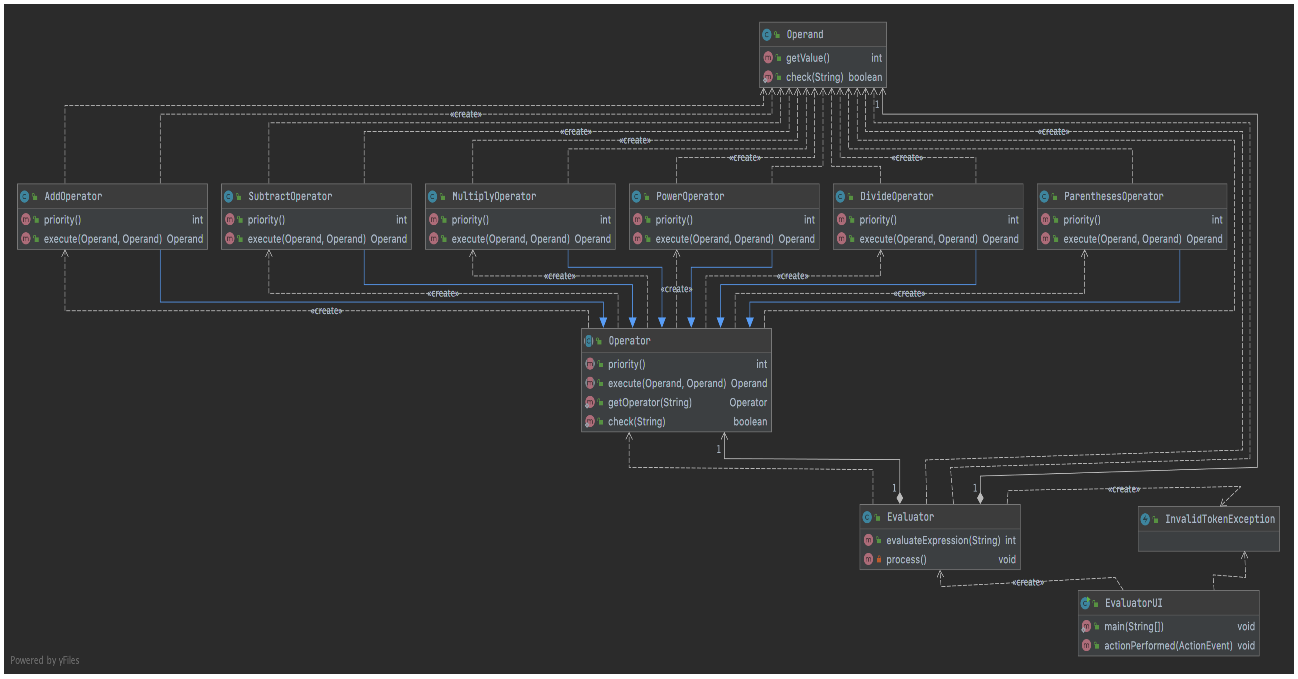
## 5 Assumption Made

The assumption that was made included not having the user input an improper equation. If the user inputs an invalid equation, the answer given will not be correct. The program will not crash but the text that appears may be the last number input or the same equation until it is fixed.

## 6 Implementation Discussion

For this project, there is a class that represents an operand in an equation. There was also an abstract class to represent an operator, which had many child classes that represented their own operator and its attributes. Within the project there is an evaluator class that takes in an expression and stores the operands and operators in their own data structure in order to process the equation. The Evaluator class is the class that does the processing of the equation passed in by the user. The EvaluatorUI class is the GUI that allows the user to type in an expression and passed that input to the Evaluator class to get the result. This design follows OOP which allows each class to fulfill specific tasks related to their characteristics. The following is the UML diagram for this project.

### 6.1 Class Diagram



## 7 Project Reflection

This project showcased my ability to solve the problems of using data structures to manipulate user input to get the required result. I was able to add any new classes that were needed to complete the project and write the necessary code required to complete the main objective of evaluating an expression. The UI was completed and could be improved in any future iterations by adding more operators.

## 8 Project Conclusion/Results

I was able to produce a successful project based on what was required and needed to complete the project. The calculator UI works as intended and allows a user to calculate their desired expression.