CSC 413 Project Documentation

Spring 2019

Kevin Fung

915857298

413.03

https://github.com/csc415-03-spring2019/csc413-p1-kefung2

Introduction

This project is the first assignment from CSC 413 as a quick refresher on Java. The project require us to finish the half complete Expression Evaluator and the Calculator GUI. Most of the class in the file are half complete and some only have the skeleton code, and we have to fill in the code in between.

In this project we have to finish implementing the part where it it check if the token is operator or operand and push it into the corresponding stack, and if it encounter the open and close parenthesis it will do that first, and all this is implemented in the Evaluator.java file. Next, we have to create the operator subclasses for the operator class. And lastly the GUI for the calculator.

Development Environment

This project is done IntelliJ Ultimate Edition, with Language level 8 and SDK 1.8.

Link:

https://www.jetbrains.com/idea/download/#section=windows

How to Build or Import and Run the Project

To get this project, you first go to the github link, and use the SSH link to clone the repo, or you can download the file it self, then to import this project, you press import project and choice calculator as the root of the project. To run this project go to the EvaluatorGUI.java file and run that file, and calculator interface will pop up, and you can use it from there.

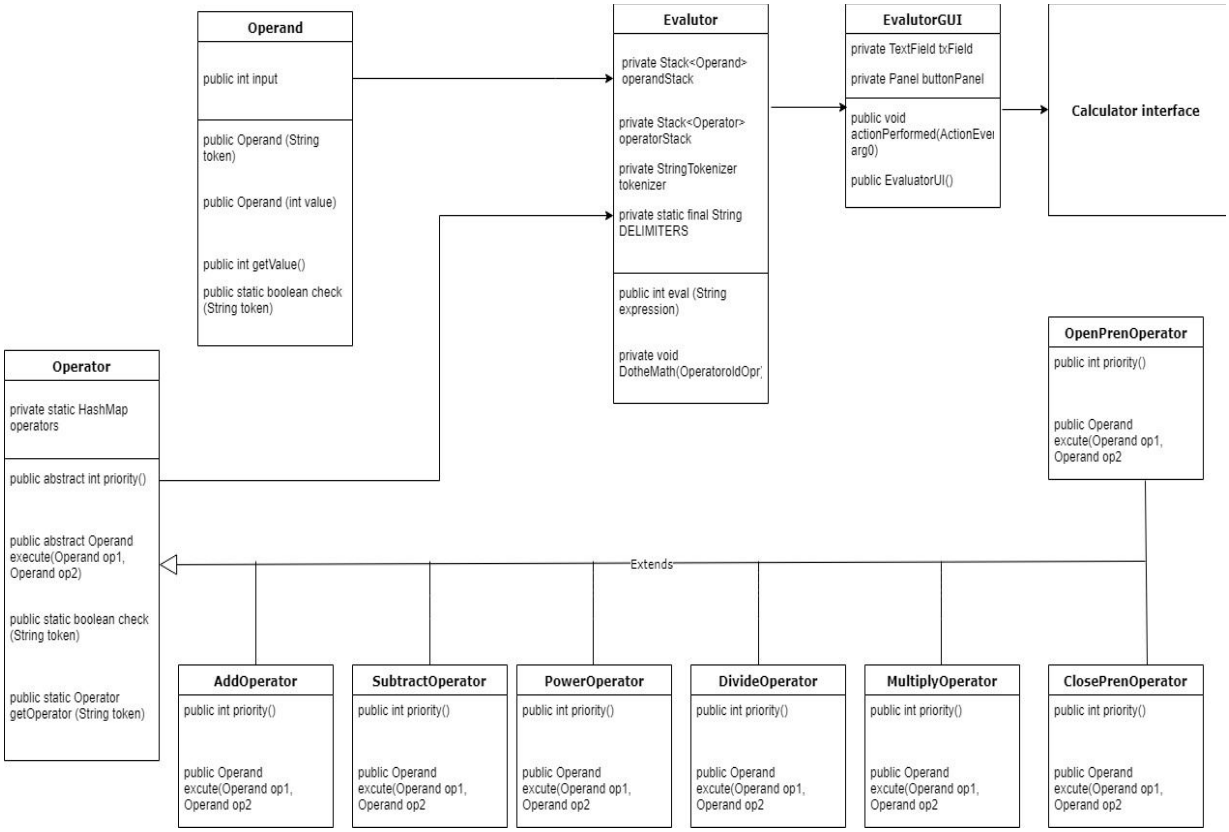
Assumptions Made when designing and Implementing you Project

When I first read the assignment, some assumption is I had was that the only part we have to implement are the part that are mentioned in the pdf file, and we have to add the subclasses ourselves. I was also told that there will not be any invalid expression, so I can safely assume that only normal expression will be entered. When I got to the GUI part, I was not sure what I had to do, since this is my first time seeing it, so my first thought was it would be hard.

Implementation Discussion

In this project, the main part that we have to implement is the Evaluator part, in the assignment pdf, it give us a list of if statement, so the first thing I think of was using if-else to do this. This program mainly revolve around 2 stack, the operator stack and operand stack. Inside the long list of if-else statement, we have to check if the token is a operand or operator, and if it is a operator, we have to if it is a valid one and the priority of it. Next is the 5 subclasses of the operator class. Inside each of the subclass there are 2 function, priority(), which return the priority of that operation, the other one is execute(Operand op1, Operand op2), which does what the operator suppose to do. Inside the Operator, there are the function getOperator(), which return the operator. A hashmap to map the key to each operator subclass, and a check function to see if the entered token is a valid operation or not, by matching the entered key with the hashmap's key. Inside the Operand function there is the function Operand, which change String to int. A getValue() function to return the value, and lastly a check function to see if you enter a number or not. The last part of this project is the GUI part, which give out a user interface. The EvaluatorGUI file was mostly done for you, and to only part we have to do is to get the text to display on the calculator, and make sure that it won't show the =, C or CE keys. Whenever C is press it will clear the whole line, and CE will clear the last character entered. When the = button

is press it call the eval function in the Evaluator class, and evaluate the expression you entered. To put all this into in to a tree diagram it will look something like this:



(chart created with https://www.draw.io/)

Project Reflection

When I got the assignment I decided to start on the next class so I can see if there will be more information given to us, but that was not a good choice, because we have to use github to submit our work, and I am not really familiar with it, so I end up spending a long time setting it up, which delay me from starting the project. Next time I think I should start it on the day that is given to us, or at the very least carefully read the assignment first, set up the project and look

over all the file that are given to us, so when I have time to do the project I don't have to waste time on it and start right on the project.

Project Conclusion and Results

This project overall went well, the function work as expected. I ran into some problem when I got empty stack exception, or it put in a extra operator, but using the debug tool I was able to find where the problem is at and fix it. The calculator itself also run fine, the number are able to display on the textbox when press and execute it when = is press. The C and CE button will not be shown on the textbox, since they will not be counted as expression. The main problem I have is resetting the textbox after pressing =, I tried to use a reset boolean statement but that end up not showing the answer, another thing I tried was to add another if-else to see if the displaying answer match with the txField, and if it does reset the text, but none of it worked.