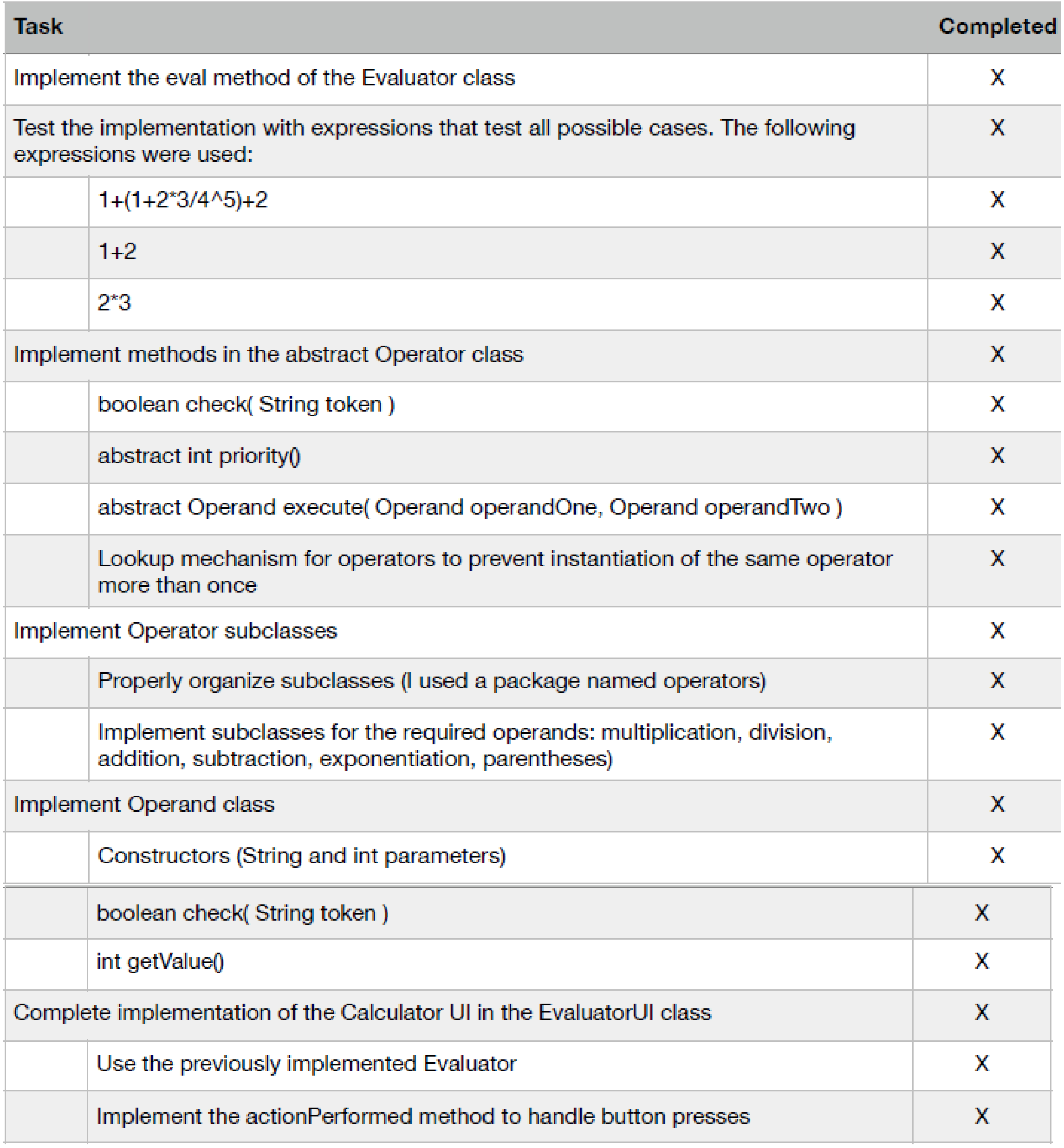
Dylan Shwan
917262260
September 24, 2019

CSC 413 Assignment 1 – GUI Calculator

GitHub Repository: https://github.com/sfsu-csc-413-spring-2018/assignment-1-expression-evaluator-fall-2019-dshwan

Introduction & Overview: The assignment required me to implement an infix expression evaluator using a code skeleton provided as well as use the evaluator to create an implemented calculator user interface.

Scope of Work:



Task		Completed
Implement the eval method of the Evaluator class		X
Test the implementation with expressions that test all possible cases. The following expressions were used:		X
	1+(1+2*3/4^5)+2	X
	1+2	X
	2*3	X
Implement methods in the abstract Operator class		X
	boolean check(String token)	X
	abstract int priority()	X
	abstract Operand execute(Operand operandOne, Operand operandTwo)	X
	Lookup mechanism for operators to prevent instantiation of the same operator more than once	X
Implement Operator subclasses		X
	Properly organize subclasses (I used a package named operators)	X
	Implement subclasses for the required operands: multiplication, division, addition, subtraction, exponentiation, parentheses)	X
Implement Operand class		X
	Constructors (String and int parameters)	X
	boolean check(String token)	X
	int getValue()	X
Complete implementation of the Calculator UI in the EvaluatorUI class		X
	Use the previously implemented Evaluator	X
	Implement the actionPerformed method to handle button presses	X

Execution & Development Environment: I used the NetBeans IDE on my HP Laptop to complete this assignment as well as tested in both the IDE and shell.

Command Line Instructions to Compile & Execute:

- Download the contents within the folder CSC413 GUI Calculator and place it where it is easily accessible.
- To use the .jar file, enter the directory via command line and type

  java -jar CSC413_GUI_Calculator.jar.

  The calculator GUI will pop up and then may be used to test arguments.
- Alternatively, you may access the EvaluatorUI.java within the directory CSC413 GUI Calculator\src via command line and type

  javac EvaluatorUI.java.

  This will compile EvaluatorUI.java along with all the linked files.

  Afterwards, type

  java EvaluatorUI

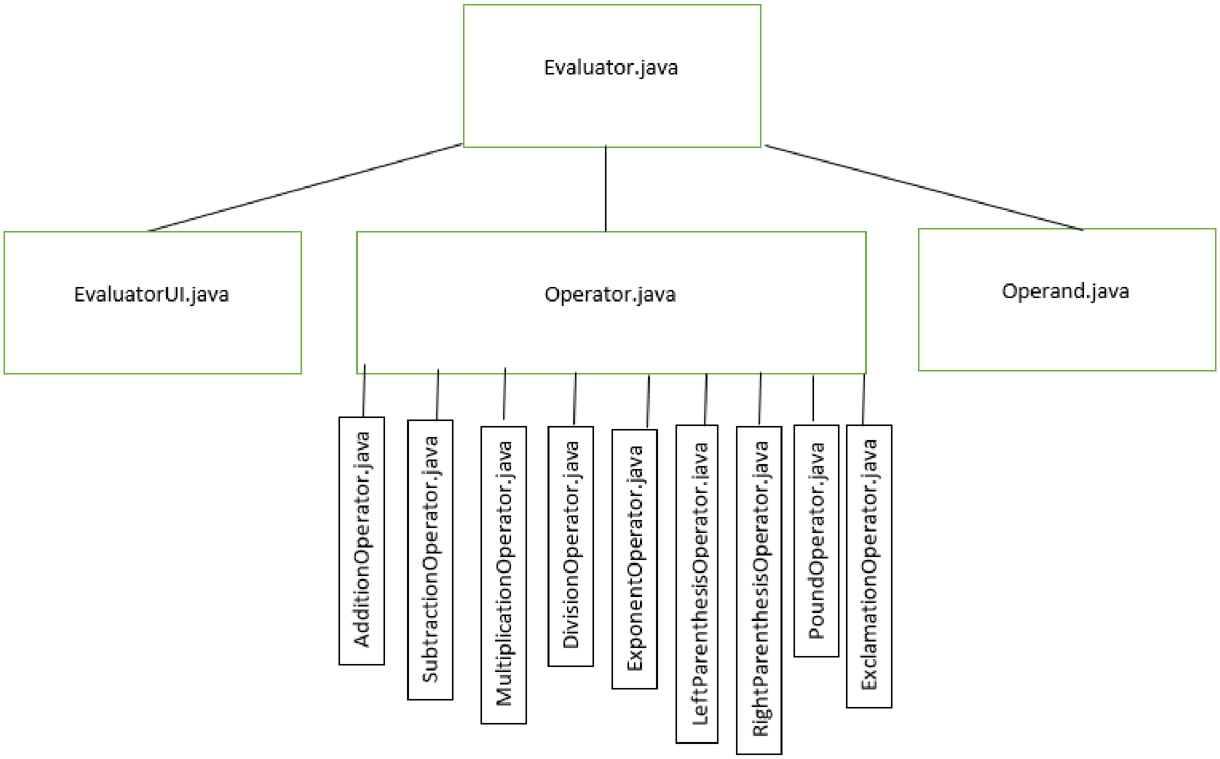
  to run the calculator.

Assumptions: The main assumption I had is that each operator used two operands. In addition, I assumed it would be best to work on implementing Evaluator.java prior to working on the GUI. The only unknowns to me were how the parentheses would be implemented

Implementation: Going into this project I had the goal of changing the framework as little as possible if I can help it. I had the Operator and derived classes in a package, operators, in order to better organize the code.

For parentheses, I made the left parenthesis have a higher priority than the right to keep pushing more operands and operators until it reaches the final corresponding right parenthesis. The PoundOperator has a priority of -1 to prevent the user from operating with a bogus operator and to keep the program running while looking for the final right parenthesis.

Class Diagram:



Conclusion: All required features were successfully implemented. This assignment was an excellent exercise to get me up to speed with implementing Abstract Classes in Java. The last time I ever used programmed involving polymorphism and inheritance was almost three years ago, and most of it was in C++. This assignment helped with making me more comfortable with Java and its concepts. If given the opportunity to continue further updates to this program, I would like to add other operands and operators such as changing the sign, random number, percentages and more.