**Project 2 Report**

**Infix Expression Parser**

Garrett Morgan

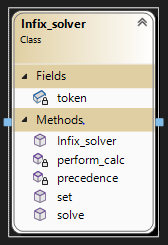
Kolton Kehoe

Reece Gillam

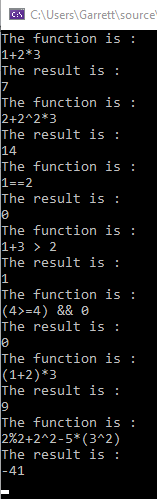
1). **System Design:**

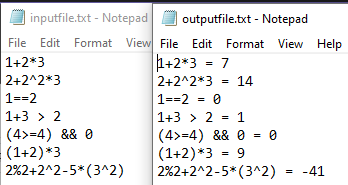
* The main method: The main method reads in the functions to use in the infix evaluation parser assignment. This class creates the infixparser object. It also sets the string token to the inserted value. After this, the solve function is called and it proceeds to solve the function.
* Infix\_solver class: This class takes in the string function and proceeds to run through it systematically. During this time, it does calculations to deduce the correct answer of the given function. The infix\_solver class uses stacks to store the operators and operands, so it knows what to use in each function.

3). **UML Diagram:**



4). **Test Cases:**





5). **Contributions:**

Garrett Morgan - I worked on the main file, the header file, and the infix\_solver class file.

Kolton Kehoe - Kolton dropped the class, so he wasn’t able to help throughout the assignment.

Reece Gillam -

6). **Improvements:**

Some of the code could be altered to make it simpler or easier to read.

Keeping in mind time complexities would also help. Some functions we initially started with were worse than what we ended with.