**Herman Mann**

**CMSC 204**

**Assignment 2**

**Notation Implementation**

**GUI SCREENSHOTS OF ASSIGNMENT 2 Execution (Notation)**

**Graphical user interface, application

Description automatically generatedGraphical user interface, application

Description automatically generated Startup Conversion of Infix to Postfix**

**Graphical user interface, application

Description automatically generatedGraphical user interface, application

Description automatically generatedConversion of Postfix to Infix Evaluating Postfix**

**Evaluating Infix**

**Graphical user interface

Description automatically generated**

**Github Screenshot of Assignment 2 (Notation in directory)**

A screenshot of a computer

Description automatically generated with medium confidence

Table

Description automatically generated

Reflection Paragraphs

Throughout the completion of the Assignment 2 which was on Notation, I got to learn and experience a lot of different object-oriented programming skills such as exceptions, and generic classes, and how to handle them in different methods of the assignment’s implementation. Using the try/catch for catching the exceptions of InvalidNotationFormatException, StackOverflowException, QueueUnderflowException, etc. was very useful in the assignment to catch the exceptions after I tried to check if the convertPostfixtoInfix method throwed the appropriate exception as needed (ex: InvalidNotationFormatException etc.). I really enjoyed how the assignment is getting me introduced to the new and unique data structures of object-oriented programming, and I am interested to look at these different data structure algorithms as the class keeps goes on. Also, I got to experience more in-depth knowledge and reasoning in Exception handling, using the static keyword, and using the concept of testing and planning using Junit tests and using Javadoc to make the appropriate commenting on each of the assignment’s implementation methods. The learning of Exceptions, the generic queue, and generic stack and using static served as a great learning experience for this assignment and for future assignments throughout the class.

In the completion of this assignment, I struggled with a couple of things towards the end of things. First, I struggled with theconvertInfixtoPostfix method which it was supposed to get the string representation of an infix expression and return a postfix expression converted into a string, but that did not meet a single expression’s requirements. My program came to an error towards the center of the Junit test due to me not throwing the StackOverflowException in my earlier methods which was the evaluatePostfix method. I solved the issue by using the java precedence characters, and which had the highest precedence went first (ex: +, -, \*, /) to return the correct format of the post fix expression entered by any user to include the special character within their infix expression which was to be converted. Another problem I struggled with were creating the student Junit tests, some methods towards the end of this assignment’s student tests had me thinking about how to approach it. So, my solution to the problem was looking at the other methods of the student test that I had done previously and from there I solved the issue successfully by changing things here and there to make the test work according to the certain test I was doing at the time by following the way the other tests were made and solved to be working successfully after the student test ran through.

When I had completed the entirety of this project, I found out a lot of things to be useful for me and would help me out for the completing of future assignments/projects. I learned and experienced that the use of exceptions will serve me a big deal and help to know for the future and will be a great thing to be well-experienced in the concept of using try/catch for exception handling and things of this sort. Also, Javadoc will help me in the future not just for the upcoming assignments but also in my career of Computer Science, for accurate and important commenting of various programs that I will be coding for bigger and widespread company projects. Also, critical analysis especially learned from the completion of this project implementation will be so useful and demanding in the field for advancement to write larger scope programs with various object-oriented programming techniques. Also, the importance of stacks, queues, and the different notation requirements served such an important purpose of furthering my knowledge on Java and its object-oriented principles/techniques. This assignment implementation of Notation helped me in so many great ways its satisfying to know I chose a great career path majoring in Computer Science.