

Lab Assignment 5

CS 301 – Data Structures

Implementation

You are given a file **expressions.txt** (which you can download from canvas). The file contains some simple mathematical expressions. Write a program that converts those expressions from infix to postfix notation and evaluates the postfix notation. Your solution should be implemented in a file named **EvaluateExpressions.java**.

This file should have a minimum of two functions, **convertToPostFix** and **evaluatePostFix**. The function **convertToPostFix** should take in a single String as a parameter (an infix expression) and return a string (postfix expression). The function **evaluatePostFix** should take in a single String as a parameter and return a value of type double. This function should evaluate the postfix expression and return the result. You can also add additional helper functions as needed.

Also write a **Driver.java** class to test your solution. First you will need to produce a new text file named **postFixExpresions.txt**, by converting the infix expressions from the file **expressions.txt**. This needs to be done with the use of the function **convertToPostFix**. Then you should evaluate all of the postfix expressions using the function **evaluatePostFix**. Output to the console the expression and its result. For example "2 5 + evaluates to : 7".

Additional details

Your implementation must use a Stack. You may use the Stack implementation from the Java Collections or your own implementation of the stack. If you use your own implementation of the stack, then you need to upload it with your submission to Canvas.

Some of the expressions you are given have statements that divide by zero. Your solution needs to be able to handle this, and print out that it is undefined.

Submission

For your submission, upload the files **EvaluateExpressions.java** and **Driver.java** with your implementation to canvas.

This is an individual assignment. Therefore, a submission is required from each student.