# **CS 240 - Data Structures and Algorithms I**

# **Programming Assignment #2**

Due date: 5/6/16 11:59 hrs

# Statement

The main aim of this project is to make you conversant with building and implementing stacks. You should use the given *Stack* to implement an infix calculator.

# Details

Implement a Calculator using the provided framework and correctly test your program to ensure that it responds to all cases as expected. You can add extra methods or fields as you see fit, but the framework provided is mandatory. You will be using the Stack in the framework provided (the LinkedList is simply to implement the stack for you).The calculator will need to be able to solve equations in infix notation, and it should be able to follow the correct order of operations (includes: parentheses, multiplication, division, addition, subtraction). You need to implement the following methods for Calculator.java, for example

* public Double solve(String equation)
  + This method should return the answer to an equation passed to it. The equation passed will only be the operands and operators (Ex: “5+4”). No equals sign will be passed.
* public double calc(double operand1, double operand2, char operator)
  + This method should return the result of operand1 operator operand2

# Sample method invocation

These are examples of the types of equations you should be able to pass to your calculator to solve. They will not include an equals sign.

test.txt

2.5+0.01 //returns 2.51

100\*2.3 //returns 230.0

100-­(73+360/60) //returns 21.0

1400/2\*(20) //returns 14000.0

**NOTE**: Please limit all returning answers to 3 decimal places using java.text.DecimalFormat. Here is an example:

DecimalFormat formatter = new DecimalFormat(“#.###”);

Double example = 4.11111;

return Double.parseDouble(formatter.format(example)); //returns 4.111

# **Code**

Download PA2.zip from the Blackboard under Assignments → Programming Assignments → PA2.

## Code Structure

data/ → This directory contains different TESTCASE data, which will be used for this assignment. test.txt is the input file with different expressions

java/ → This directory contains all the JAVA code. You only need to modify Calculator.java; **Please do not modify other code in this folder.** It is mandatory that you use the Stack.java provided.

run.py -> A tool to compile, run & test your code.

## Extract and Run

* Download the file PA2.zip. Extract it.
* Copy the directory to your ZFS system by following the basic linux guide provided.
* Let’s say you extracted into ‘PA2’ directory. Now, from the terminal:

$cd PA2

# To run and evaluate your code

$./run.py

## Compatibility

Note that, all assignments will be tested under Linux environment with Python and Oracle Java is installed. Given code might work on other platforms (like Windows, etc.) but has not been tested. Hence, it is encouraged to develop and test your code in a Linux based environment.

# **Submission**

You should only modify and upload Calculator.java to Blackboard. **Any change in other files will not be accepted and you will not be evaluated in that case**.

# **Evaluation**

There is some held out data set against which your code will be tested and evaluated. Your main aim is to write a Stack class following the specifications provided.

# **Honor Code**

I encourage students to discuss the programming assignments including specific algorithms and data structures required for the assignments. However, students should not share any source code for solution.

Code exists on the web for many problems including some that we may pose in problem sets or assignments. Students are expected to come up with the answers on their own, rather than extracting them from code on the web. This also means that we ask that you do not share your solutions to any of the homework - programming assignments, or problem sets - with any other students. This includes any sort of sharing, whether face-to-face, by email, uploading onto public sites, etc. Doing so will drastically detract from the learning experience of your fellow students.