California State University, Sacramento
College of Engineering and Computer Science

CSc 20: Programming Concepts and Methodology II

Lab 07

Objective:

The objective of this lab is to get you some experience in **testing**, **debugging** and **exception handling**.

The lab assignment:

1. Test your infix evaluator program with the following test cases.

    a. 12 + (34 – 56) * 78)
    b. 12 + ((34-56) * 78
    c. 12 + 34 56
    d. 12 + * 34

2. Obviously, you will either get an exception or wrong output for each of the test cases.

3. If the test case produces an exception, you can locate the error from the **call stack trace** generated by the JVM. Example,

    ```
    Exception in thread "main" java.lang.ArrayIndexOutOfBoundsException: -1
        at StackQueue.Stack.pop(Stack.java:10)
        at infix.evaluePostfix(infix.java:59)
        at infix.main(infix.java:83)
    ```

4. Insert code into your program to catch the exception and, throw, catch, and print your own exceptions.

5. If the test case doesn't produce any exception, use jdb to step through you program in order to locate the error.

6. Insert some error checking code to throw, catch, and print your own exceptions

7. Your new program should produce outputs similar to the following.

    ```
    D:\lab05\>java infix2
    Enter infix: 12 + (34 - 56) * 78 )
    ***** Too many ) *****
    Enter infix: 12 + (( 34-56) * 78
    ***** Too many ( *****
    Enter infix: 12 + 34 56
    Postfix is 12 34 56 +
    ***** Too many operands *****
    Enter infix: 12 + * 34
    Postfix is 12 34 * +
    ***** Too many operators *****
    ```

Notes:

1. You must use jdb for at least one test case.

    a. Open two athena windows.
    b. In one window, do
    java -Xdebug -Xrunjdwp:transport=dt_socket,server=y,address=20000 infix
    c. In the other window, do
    jdb -attach 20000

2. Use command script to record a jdb session.

3. To get whole credit for this lab:

    a. Demo with the four test cases and "12+34*(56-7)-18/9".
    b. Turn in a hardcopy of infix.java.
    c. Email a softcopy of *infix.java* and the file *typescript*.