CS 3410 : Data Structures

Programming 2 (Fall 2017)

Due date: Saturday, 30 September, 2017 by 11:00 p.m.

A stack can be used as a balanced parenthesis checker. In this programming assignment you are required to implement a stack using array. The purpose of the stack is to check an expression involving parentheses, braces, & brackets and determine whether they are balanced. That is, whether they are in proper order and all open parentheses (or, braces, or brackets) have corresponding closing ones in the proper order. Write a Java program to solve this problem. The input to your program is a *.java* file that contain parentheses, braces, and brackets. You may assume that no parenthesis, brace, or bracket is commented out.

- **Programming:**

Create a separate class for your stack and implement three stack related methods (*push()*, *pop()*, and *peek()*) in it. YOU ARE NOT ALLOWED TO USE STACK CLASS FROM JAVA API. You need to build your own stack class. The class should be reusable, that is, do not make it type specific. Use of Generic could be helpful. You can use this stack class in future to solve stack-related problems involving any data type. Safely assign relatively large size (may be 1000) for the stack (*yes, we are compromising the space. It is a trade-off as space is relatively less costly but this trade-off would allow easier development*). Implement the stack class in a separate .java file.

Once the stack class is built use the class to solve the problem mentioned above. This should be a separate .java file containing the *main()* method. The *main()* method is responsible to capture the input (file name with .java extension), parse it, use the stack (as defined in the other file), and then produce the output displaying a message indicating whether the expression is balanced (parentheses, braces, and brackets) or not.

Program should not suffer from runtime exceptions. All runtime errors should be properly handled (for example, filename specified not existing).

Submission instruction: You are required to submit your files through BlazeVIEW submission dropbox. It is easier to upload them as a .zip file. Name the .zip file as ***lastname.zip*** by replacing *lastname* by your last name.

Commenting: Your program should contain the following information: your name, course (CS3410), semester (Fall 2017), and assignment identifier (Program 2).

Caution: Try to avoid errors in the main program due to incorrectness of the stack class built.