Lab 3

CSE 385

Create a new project in Eclipse, import the `Application.java` file as a source file, run the application and make sure that you receive no error.

I. Balance Checker

The class `ExpressionChecker` includes a method for checking whether or not an expression is balanced. Copy the following lines of code into the body of `main` method and run the application:

```
String input;

input="{7-(348+[1*9]+15)-3}";

ExpressionChecker theChecker = new ExpressionChecker(input);
System.out.println(theChecker.check());
```

The output is printed below:

```
true
```

Using the `check()` method, specify whether the following expressions are balanced on unbalanced:

- `(93{82}738)`
- `{93[8{}2]738}`
- `{[4{44(85)31}22]75}(14)`
- `{( {(29[3]) ([351]1{9[2]7)})}`

II. Receiving an Input from the User

Erase the body of the `main` method and write a piece of code in the body of `main` method which keeps receiving expressions from the user and prints `true/false` in the output for balanced/unbalanced expressions. This piece of code should `return` once the user provides an empty input. You can use the provided `GetString()` method which receives an input from the user and `return` it as an output.

III. A Generic Stack Class

In this section, we develop a `Stack` class which has exactly the same features and functionality of `StackC` class with only one difference: The `Stack` class is generic and can be used to create stacks of `Character`, `Integer`, `String`, `Long` and other types of data. A draft of `Stack` class is provided in the `Application.java` file. Develop the methods `push`, `pop`, `isEmpty`, and `isFull` for the `Stack` class. Please set proper type of input and output for these methods.

After developing the above methods, find the following line of code in the ExpressionChecker class:

```
StackC theStack = new StackC(stackSize);
```

Replace the above line with the below code:

```
Stack<Character> theStack = new Stack<Character> (stackSize);
```

Erase the body of the `main` method, copy the following lines of code into the body of the `main` method and see if the correct output is being produced:

```
String input= "(38[72{53}])";
ExpressionChecker theChecker = new ExpressionChecker(input);
System.out.println(theChecker.check());
```

The correct output is printed below:

```
true
```

IV. A Bug in Balance Checker

Erase the body of the `main` method, copy the following lines of code into the body of the `main` method, and see the output:

```
String input= "{(38[72{53}])";
ExpressionChecker theChecker = new ExpressionChecker(input);
System.out.println(theChecker.check());
```

The provided `String` in the above code is unbalanced, however a `true` output is produced. It seems that there is a bug in the `check` method of `ExpressesionChecker` class. To find this bug, review the algorithm that we studied for checking balanced expressions and see if the `check` method implements the algorithm correctly.