

Program #6: Stack

For this assignment, I want you to create and submit a `Stack.java` file in which you implement a stack class. This stack should **ONLY** have the following operations as defined in our lectures.

1. Push: insert an item on top of the stack
2. Pop: delete an item from the top of the stack (and return it)
3. Peek: return (but do not delete) the item on top of the stack
4. Size: return the size of the stack
5. IsEmpty: return whether the stack is empty
6. IsFull: return whether the stack is full
7. Equals: compare two stacks
8. Add: concatenate two stacks

In addition, it should have a constructor, a copy constructor, and output capability i.e.

9. `Stack()`
10. `Stack(Stack s)`
11. `String toString()`

You are to build this stack **ON TOP** of your `List.java` from Assignment #5. In fact, the only **member variable** your Stack class will require is a generic list. To do this, make sure that you have a copy of `List.java` in the same folder as the `Stack.java` file you're creating.

- For this assignment, you will be provided with two files:
 - `StackTest.java` contains the `main` function. You may **NOT** modify this file!
 - `StackTest.out` contains the output from my implementation of the stack. You would be wise to ensure your output is **EXACTLY** the same (That includes spacing and formatting).
- Recall that the only file you will submit is (`Stack.java`). If you have implemented it correctly, It should work with my version of `List.java`
- Label the file by typing your name in a commented section at the top of the `Stack.java` file.

- Compiling `StackTest.java` should automatically use your `List.java` and `Stack.java` files (assuming they are in the same directory). After successful compilation, running the produced executable file SHOULD produce output identical to `StackTest.out`.
- Feel free to redirect the output to a file of your choosing, and then compare the two files (i.e. `StackTest.out`, and your output file) using the command prompt. For example, once completed, the three commands below should **NOT** produce any output.

Linux:

```
javac StackTest.java
java StackTest > myOutput
diff StackTest.out myOutput
```

Windows:

```
javac StackTest.java
java StackTest > myOutput
fc StackTest.out myOutput
```

- Just so that we are on the same page, your stack should be oriented such that its top is to the left i.e. the top of the stack is the left most item of the list.