(150 pts) due: December 4, 2013 11:59pm

Important Notes

- **This assignment is to be done on your own.** If you need help, see the instructor or TA.
- Please start the assignment as soon as possible and get your questions answered early.
- Read through this specification completely before you start.
- Some aspects of this specification are subject to change, in response to issues detected by students or the course staff.

1 Description

This assignment is for you to work on stacks and queues. There are two problems:

- Implement a queue using two stacks. More specifically, use ArrayBasedStack or LinkedStack that we have implemented in class. The definition for this class (using ArrayBasedStack) is as follows:

```
public class MyQueue<E> {
   ArrayBasedStack<E> s1, s2;

   public MyQueue() {//constructor
      ...
   }

   public int size() {
      ...
   }

   public boolean add(E e) {
     //Inserts the specified element into this queue if it is possible to
     //do so immediately without violating capacity restrictions,
     //returning true upon success and throwing an IllegalStateException
     //if no space is currently available
      ...
   }

   public E element() {
     //Retrieves, but does not remove, the head of this queue.
     //Throws NoSuchElementException if this queue is empty
      ...
   }

   public E peek() {
     // Retrieves, but does not remove, the head of this queue,
     //or returns null if this queue is empty
      ...
   }

   public boolean offer(E e) {
```

```
    //Inserts the specified element into this queue if it is possible to
    //do so immediately without violating capacity restrictions.
      ...
  }

  public E remove() {
    //Retrieves and removes the head of this queue.
    //Throws NoSuchElementException if this queue is empty.
      ...
  }

  public E pull() {
    //Retrieves and removes the head of this queue,
    //or returns null if this queue is empty.
      ...
  }

}
```

- Implement the general case of converting an infix expression to a postfix expression. Refer to the slides titled "Stack Applications" page 8-10.
- Given a string containing just the characters '(' and ')', find the length of the longest valid (well-formed) parentheses substring.

  For "(()", the longest valid parentheses substring is "()", which has length = 2.

  Another example is ")()())", where the longest valid parentheses substring is "()()", which has length = 4.

  The interface for this method is defined as below:

```
public class StackUtil {
  public int longestValidParentheses(String s) {
      ...
  }
}
```

2 Submission

Provide working code (as well as JUnit tests) for the class and method required for this assignment (50pts each problem). Turn in a zip file named LastnameFirstname-Assignment7.zip, containing all your source code. The package name for the assignment must be edu.brynmawr.cs206.assignment7. Include the Javadoc tag @author in each class source file. **Do not turn in class files**.