```java
import components.sequence.Sequence;
import components.sequence.Sequence1L;
import components.simplewriter.SimpleWriter;
import components.simplewriter.SimpleWriter1L;
import components.stack.Stack;
import components.stack.Stack1L;

/**
 * HW20
 *
 * @author Sam Espanioly
 */
public final class practice {

    /**
     * Default constructor--private to prevent instantiation.
     */
    private practice() {
        // no code needed here
    }

    /**
     * Reverses ("flips") {@code this}.
     *
     * @updates this
     * @ensures this = rev(#this)
     */
    //this method works though its no an instance method
    public void flipSequenceRecursive() {
        // temporary stack
        Sequence<String> tempo = new Sequence1L<>();

        if (this.length() > 0) {
            tempo.add(0, this.remove(0));
            this.flipSequenceRecursive();
        }
        this.transferFrom(tempo);
    }

    // this method should work but it gives an error
    public void flipStack() {
        // temporary stack
        Stack<String> tempo = new Stack1L<>();

        for (String x : this) {
            tempo.push(x);
        }
        int len = this.length();
        for (int i = 0; i < len; i++) {
            this.transferFrom(tempo);
        }
    }

    /**
     * Reverses ("flips") {@code this}.
     *
     * @updates this
     * @ensures this = rev(#this)
     */
    public void flipSequence() {
        Sequence<String> tempis = new Sequence1L<>();
        int a = 0;
        for (String x : this) {
            tempis.add(a, x);
            a++;
        }
        int len = this.length();
        for (int i = 0; i < len; i++) {
            this.transferFrom(tempis);
        }
    }

    /**
     * Main method.
     *
     * @param args
     *            the command line arguments; unused here
     */
    public static void main(String[] args) {
        SimpleWriter out = new SimpleWriter1L();
    }

}

```