```java
import static org.junit.Assert.assertEquals;

import org.junit.Test;

import components.sequence.Sequence;
import components.sequence.Sequence1L;

/**
 * Sample JUnit test fixture for SequenceSmooth.
 *
 * @author Put your name here
 *
 */
public final class SequenceSmoothTest {

    /**
     * Constructs and returns a sequence of the integers provided as arguments.
     *
     * @param args 0 or more integer arguments
     * @return the sequence of the given arguments
     * @ensures createFromArgs= [the sequence of integers in args]
     */
    private Sequence<Integer> createFromArgs(Integer... args) {
        Sequence<Integer> s = new Sequence1L<Integer>();
        for (Integer x : args) {
            s.add(s.length(), x);
        }
        return s;
    }

    /**
     * Test smooth with s1 = <2, 4, 6> and s2 = <-5, 12>.
     */
    @Test
    public void test1() {
        /*
         * Set up variables and call method under test
         */
        Sequence<Integer> seq1 = this.createFromArgs(2, 4, 6);
        Sequence<Integer> expectedSeq1 = this.createFromArgs(2, 4, 6);
        Sequence<Integer> seq2 = this.createFromArgs(-5, 12);
        Sequence<Integer> expectedSeq2 = this.createFromArgs(3, 5);
        SequenceSmooth.smooth(seq1, seq2);
        /*
         * Assert that values of variables match expectations
         */
        assertEquals(expectedSeq1, seq1);
        assertEquals(expectedSeq2, seq2);
    }

    /**
     * Test smooth with s1 = <7> and s2 = <13, 17, 11>.
     */
    @Test
    public void test2() {
        /*
         * Set up variables and call method under test
         */
        Sequence<Integer> seq1 = this.createFromArgs(7);
        Sequence<Integer> expectedSeq1 = this.createFromArgs(7);
        Sequence<Integer> seq2 = this.createFromArgs(13, 17, 11);
        Sequence<Integer> expectedSeq2 = this.createFromArgs();
        SequenceSmooth.smooth(seq1, seq2);
        /*
         * Assert that values of variables match expectations
         */
        assertEquals(expectedSeq1, seq1);
        assertEquals(expectedSeq2, seq2);
    }

    /**
     * Test smooth with s1 = <2, 6, 14, 18> and s2 = <17, 11>.
     */
    @Test
    public void test3() {
        /*
         * Set up variables and call method under test
         */
        Sequence<Integer> seq1 = this.createFromArgs(2, 6, 14, 18);
        Sequence<Integer> expectedSeq1 = this.createFromArgs(2, 6, 14, 18);
        Sequence<Integer> seq2 = this.createFromArgs(17, 11);
        Sequence<Integer> expectedSeq2 = this.createFromArgs(4, 10, 16);
        SequenceSmooth.smooth(seq1, seq2);
        /*
         * Assert that values of variables match expectations
         */
        assertEquals(expectedSeq1, seq1);
        assertEquals(expectedSeq2, seq2);
    }

    /**
     * Test smooth with s1 = <4, 6, 8> and s2 = <6, 9>.
     */
    @Test
    public void test4() {
        /*
         * Set up variables and call method under test
         */
        Sequence<Integer> seq1 = this.createFromArgs(4, 6, 8);
        Sequence<Integer> expectedSeq1 = this.createFromArgs(4, 6, 8);
        Sequence<Integer> seq2 = this.createFromArgs(6, 9);
        Sequence<Integer> expectedSeq2 = this.createFromArgs(5, 7);
        SequenceSmooth.smooth(seq1, seq2);
        /*
         * Assert that values of variables match expectations
         */
        assertEquals(expectedSeq1, seq1);
        assertEquals(expectedSeq2, seq2);
    }

    /**
     * Test smooth with s1 = <8, 12> and s2 = <4, 20>.
     */
    @Test
    public void test5() {
        /*
         * Set up variables and call method under test
         */
        Sequence<Integer> seq1 = this.createFromArgs(8, 12);
        Sequence<Integer> expectedSeq1 = this.createFromArgs(8, 12);
        Sequence<Integer> seq2 = this.createFromArgs(4, 20);
        Sequence<Integer> expectedSeq2 = this.createFromArgs(10);
        SequenceSmooth.smooth(seq1, seq2);
        /*
         * Assert that values of variables match expectations
         */
        assertEquals(expectedSeq1, seq1);
        assertEquals(expectedSeq2, seq2);
    }

    /**
     * Test smooth with s1 = <69> and s2 = <100>.
     */
    @Test
    public void test6() {
        /*
         * Set up variables and call method under test
         */
        Sequence<Integer> seq1 = this.createFromArgs(69);
        Sequence<Integer> expectedSeq1 = this.createFromArgs(69);
        Sequence<Integer> seq2 = this.createFromArgs(100);
        Sequence<Integer> expectedSeq2 = this.createFromArgs(69);
        SequenceSmooth.smooth(seq1, seq2);
        /*
         * Assert that values of variables match expectations
         */
        assertEquals(expectedSeq1, seq1);
        assertEquals(expectedSeq2, seq2);
    }

}
```