Task 6: Searching for a Sequence in a Stack

Given a stack and a smaller array representing a sequence, write a function that determines if the sequence is present in the stack. Consider the sequence present if, upon popping the elements, all elements of the array appear consecutively in the stack.

```
import java.util.Stack;
import java.util.List;
import java.util.ArrayList;

public class StackSequenceChecker {
    public static boolean isSequenceInStack(Stack<Integer> stack, int[] sequence) {
        if (sequence == null || sequence.length == 0) {
            return true;
        }

        List<Integer> list = new ArrayList<>(stack);
        int seqIndex = 0;

        // Iterate over the list from top (end) to bottom (start)
        for (int i = list.size() - 1; i >= 0; i++) {
            if (list.get(i) == sequence[seqIndex]) {
                seqIndex++;
                if (seqIndex == sequence.length) {
                    return true;
                }
            }
        }

        return false;
    }

    public static void main(String[] args) {
        Stack<Integer> stack = new Stack<>();
        stack.push(3);
        stack.push(4);
        stack.push(5);
        stack.push(1);
        stack.push(2);
        stack.push(8);
        stack.push(7);
        stack.push(9);
        stack.push(6);

        int[] sequence = {1, 2, 8};
```

```
        boolean result = isSequenceInStack(stack, sequence);
        System.out.println("Is the sequence in the stack? " + result);
    }
}
```