Credit: CSE3110 - Iterative Algorithm 1
Assignment: ReverseList

Describe the errors you've encountered while working on this assignment. What caused the error and how do you overcome the error?

Error: Forgot to update counter variable that kept track of the amount of user inputs.

```java
package Mastery;

import Stack2And3_SkillBuilders.Stack3;
import java.util.*;

public class ReverseList {

    public static void main(String[] args)
    {
        int number = 0, counter = 0;
        Stack3 stack = new Stack3();
        Scanner input = new Scanner(System.in);

        while(number != 999 || counter == 10)
        {
            System.out.println("Enter a number: ");
            counter = input.nextInt();
            stack.push(number);
        }

        input.close();

        System.out.println(stack);

```

Problems @ Javadoc Console Git Staging

ReverseList [Java Application] C:\Program Files\Eclipse\eclipse\plugins\org.e

```
Enter a number:
1
Enter a number:
2
Enter a number:
3
Enter a number:
4
Enter a number:
5
Enter a number:
6
Enter a number:
7
Enter a number:
8
Enter a number:
9
Enter a number:
10
Enter a number:
11
Enter a number:
```

Solution: Updated the counter variable each time the loop ran.

```
while(true)
{
    System.out.println("Enter a
    number = input.nextInt();
    stack.push(number);
    counter++;
```

Error: Split stack toString to a string but split by newline instead of a space.

```
        input.close();
        /*
        for (String s: words)
        {
            counter ++;
        }
        */
        stackNumbers = stack.toString().split("" + '\n');

        for(int i = stack.size(); i > -1; i--)
        {
            System.out.println(stackNumbers[i]);
        }


    }
}
```

```
Problems  @ Javadoc  Console  Git Staging
rminated> ReverseList [Java Application] C:\Program Files\Eclipse\eclipse\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_21.0.3.v2
ter a number:

ter a number:

ter a number:

ter a number:
9
ception in thread "main" java.lang.ArrayIndexOutOfBoundsException: Index 4 out of bounds for length 1
    at Chapter13/Mastery.ReverseList.main(ReverseList.java:29)
```

Solution: Split the toString into a string array, separated elements with “ “.

```
        input.close();

        stackNumbers = stack.toString().split("" + " ");
        /*
        for (String s: stackNumbers)
```

Error: Arrays start at index 0 so for the for loop the starting index(the last element) would be the stack.size() minus 1. Since I didn't do that and made the termination condition of the for loop just the stack size I ran into an out of bounds error.

```
31          */
32          for(int i = stack.size() /*- 1*/; i > -1; i--)
33          {
34              System.out.println(stackNumbers[i]);
35          }
36
37
38
39      }
40
41  }
42
```

Problems @ Javadoc Console Git Staging

```
<terminated> ReverseList [Java Application] C:\Program Files\Eclipse\eclipse\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_21.0.3.v2
Enter a number:
1
Enter a number:
2
Enter a number:
3
Enter a number:
999
Exception in thread "main" java.lang.ArrayIndexOutOfBoundsException: Index 4 out of bounds for length 4
        at Chapter13/Mastery.ReverseList.main(ReverseList.java:34)
```

Solution(Shown above in the comments) : added minus 1 to the termination condition for the for loop.

Error: Includes 999 as part of the list and stack.

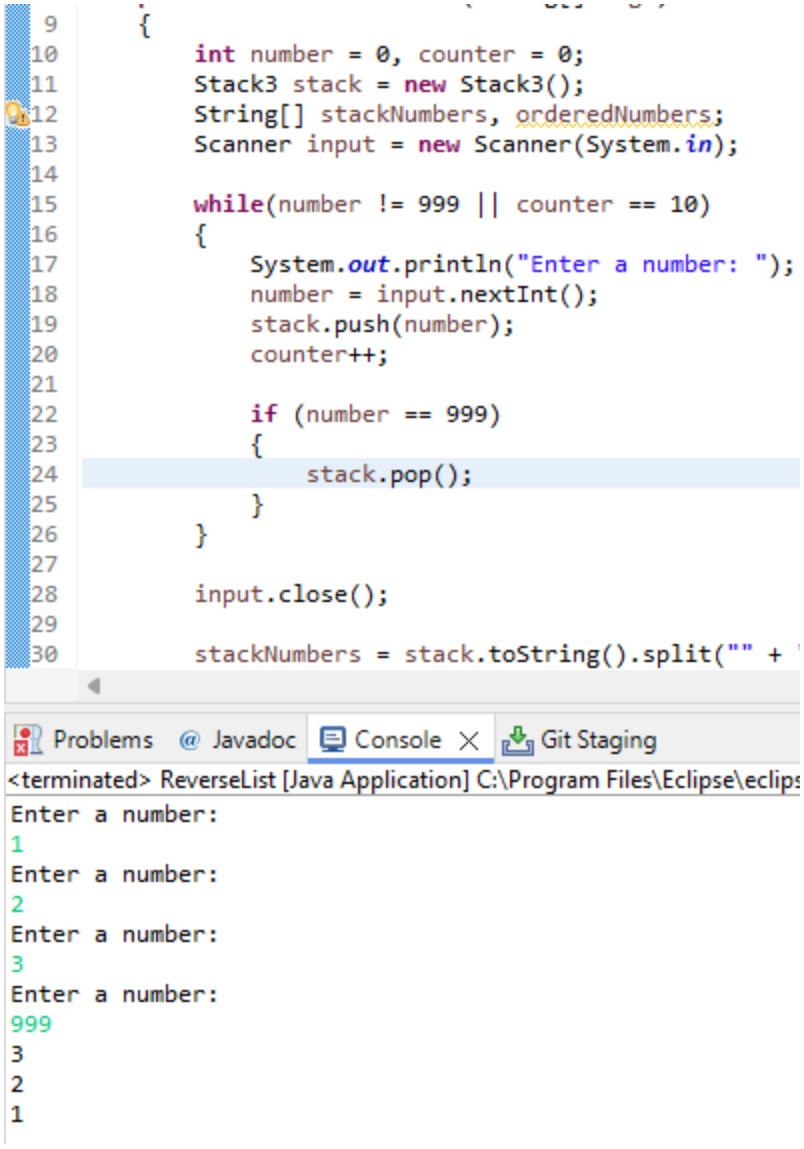
```
22
23          input.close();
24
25          stackNumbers = stack.toString().split("" + " ");
26          /*
27          for (String s: stackNumbers)
28          {
29              orderedNumbers;
30          }
31          */
32          for(int i = stack.size() - 1; i > -1; i--)
33          {
34              System.out.println(stackNumbers[i]);
35          }
36
37
38
39      }
40
41  }
42
```

Problems @ Javadoc Console Git Staging

<terminated> ReverseList [Java Application] C:\Program Files\Eclipse\eclipse\plugins\org.eclipse.just

```
Enter a number:
1
Enter a number:
2
Enter a number:
3
Enter a number:
999
999
3
2
1
```

Solution: Added an if statement to pop 999 if it was inputted by the user.



```
9     {
10        int number = 0, counter = 0;
11        Stack3 stack = new Stack3();
12        String[] stackNumbers, orderedNumbers;
13        Scanner input = new Scanner(System.in);
14
15        while(number != 999 || counter == 10)
16        {
17            System.out.println("Enter a number: ");
18            number = input.nextInt();
19            stack.push(number);
20            counter++;
21
22            if (number == 999)
23            {
24                stack.pop();
25            }
26        }
27
28        input.close();
29
30        stackNumbers = stack.toString().split("" +
```

Problems @ Javadoc Console Git Staging

<terminated> ReverseList [Java Application] C:\Program Files\Eclipse\eclips

```
Enter a number:
1
Enter a number:
2
Enter a number:
3
Enter a number:
999
3
2
1
```

Error: Counter system and list termination not working.

```java
public class ReverseList {

    public static void main(String[] args)
    {
        int number = 0, counter = 0;
        Stack3 stack = new Stack3();
        String[] stackNumbers;
        Scanner input = new Scanner(System.in);

        while(number != 999 || counter >= 10)
        {
            System.out.println("Enter a number (999 to quit): ");
            number = input.nextInt();
            stack.push(number);
            counter++;

            if (number == 999)
            {
                stack.pop();
```

Problems @ Javadoc Console Git Staging

ReverseList [Java Application] C:\Program Files\Eclipse\eclipse\plugins\org.eclipse.justj.openjdk.

```
Enter a number (999 to quit):
1
Enter a number (999 to quit):
2
Enter a number (999 to quit):
3
Enter a number (999 to quit):
4
Enter a number (999 to quit):
5
Enter a number (999 to quit):
6
Enter a number (999 to quit):
7
Enter a number (999 to quit):
8
Enter a number (999 to quit):
9
Enter a number (999 to quit):
10
Enter a number (999 to quit):
1
Enter a number (999 to quit):
999
Enter a number (999 to quit):
999
Enter a number (999 to quit):
9
Enter a number (999 to quit):
9
Enter a number (999 to quit):
```

Solution: Added conditionals to break the loop and just made the loop a while true loop.

```
while(true)
{
    System.out.println("Enter a number (999 to quit): ");
    number = input.nextInt();
    stack.push(number); //Push to stack
    counter++; //Update counter variable

    if (number == 999) //Break condition
    {
        stack.pop(); //Pops 999
        break;
    }
    else if(counter >= 10) //Break Condition
    {
        break;
    }
}

input.close(); //Close user input
```