

1.

```
package week02.lab;

Windsurf: Refactor | Explain
public class RecursionStar {
    Run | Debug | Windsurf: Refactor | Explain | Generate Javadoc | X
    public static void main(String[] args) {
        System.out.println(starString(n: 3));
    }

    Windsurf: Refactor | Explain | Generate Javadoc | X
    public static String starString(int n) {
        if (n < 0) {
            throw new IllegalArgumentException();
        }
        if (n == 0) {
            return "*";
        }
        else {
            return starString(n - 1) + starString(n - 1);
            //2^n = 2 * 2^(n-1), so double the starString
        }
    }
}
```

2.

```
1 package week02.lab;
2
Windsurf: Refactor | Explain
3 public class RecursionNums {
    Run | Debug | Windsurf: Refactor | Explain | Generate Javadoc | X
4     public static void main(String[] args) {
5         writeNums(n: 5);
6     }
7
    Windsurf: Refactor | Explain | Generate Javadoc | X
8     public static void writeNums(int n) {
9         if (n < 1) {
10             throw new IllegalArgumentException();
11         }
12         if (n == 1) {
13             System.out.print(1);
14         }
15         else {
16             writeNums(n - 1); // recursive call first so numbers print in ascending order
17             System.out.print(", " + n); // if you want descending order, this line goes first
18         }
19     }
20 }
21
```

PROBLEMS 27 OUTPUT DEBUG CONSOLE TERMINAL TEST RESULTS PORTS GITLENS Run: RecursionNums

```
duey@MacBook-Air-Duy CS211 % /usr/bin/env /Library/Internet\ Plug-Ins/JavaAppletPlugin.plugin/Contents/Home
ders/w3/0x9k0j1s65gczs9x9ztngbw0000gn/T/cp_blgly17jggzf1hn5ifl2kqlxy.jar week02.lab.RecursionNums
1, 2, 3, 4, 5
duey@MacBook-Air-Duy CS211 %
```

3.

```

1 package week02.lab;
2
3 public class RecursionChars {
4     public static void main(String[] args) {
5         writeChars(n: 8);
6     }
7
8     public static void writeChars(int n) {
9         if (n < 1) {
10             throw new IllegalArgumentException();
11         }
12         if (n == 1) {
13             System.out.print('*');
14         }
15         else if (n == 2) {
16             System.out.print("**");
17         }
18         else {
19             System.out.print('<'); // '<' prints before recursion
20             writeChars(n - 2); // n goes down by 2 instead of 1
21             System.out.print('>'); // '>' prints after recursion returns
22         }
23     }
24 }

```

Windsurf: Refactor | Explain | Generate Javadoc | X

PROBLEMS 29 OUTPUT DEBUG CONSOLE TERMINAL TEST RESULTS PORTS GITLENS

ars

duey@MacBook-Air-Duy CS211 % cd /Users/duey/CS211/CS211 ; /usr/bin/env /Library/Internet\ Plug-Ins/JavaAppletPlugin.plugin/Contents/Home/bin/java -cp /var/folders/w3/0x9k0j1s65gczs9x9ztngbw0000gn/T/cp_24312wgvg5vmxwc56j5.jar week02.lab.RecursionChars

4.

```

1 package week02.lab;
2
3 public class RecursionEvents {
4     public static void main(String[] args) {
5         int product = multiplyEvens(n: 4);
6         System.out.println(product);
7     }
8
9     public static int multiplyEvens(int n) {
10         if (n <= 0) {
11             throw new IllegalArgumentException();
12         }
13         if (n == 1) {
14             return 2;
15         }
16         else {
17             return multiplyEvens(n - 1) * (2 * n);
18             //product(n) = product(n-1) * 2n
19         }
20     }
21 }
22

```

Windsurf: Refactor | Explain | Generate Javadoc | X

PROBLEMS 29 OUTPUT DEBUG CONSOLE TERMINAL TEST RESULTS PORTS GITLENS Run: RecursionEvents

duey@MacBook-Air-Duy CS211 % /usr/bin/env /Library/Internet\ Plug-Ins/JavaAppletPlugin.plugin/Contents/Home/bin/java -cp /var/folders/w3/0x9k0j1s65gczs9x9ztngbw0000gn/T/cp_24312wgvg5vmxwc56j5.jar week02.lab.RecursionEvents

duey@MacBook-Air-Duy CS211 %

5.

```
Windsurf: Refactor | Explain
3 public class RecursionReverse {
  Run | Debug | Windsurf: Refactor | Explain | Generate Javadoc | X
4   public static void main(String[] args) {
5     System.out.println(isReverse("CSE143", "341esc"));
6   }
7
  Windsurf: Refactor | Explain | Generate Javadoc | X
8   public static boolean isReverse(String s1, String s2) {
9     if (s1.length() != s2.length()) { // if lengths don't match
10      return false;
11    }
12    if (s1.length() <= 1) { // empty or single character
13      return true;
14    }
15
16    char first = Character.toLowerCase(s1.charAt(0)); // first character of s1 lowercased
17    char last = Character.toLowerCase(s2.charAt(s2.length() - 1)); // last character of s2 lowercased
18    if (first != last) { // if outer character don't match
19      return false;
20    }
21
22    return isReverse(s1.substring(1), // remove first character of s1
23                    s2.substring(0, s2.length() - 1) // remove last character of s2
24                    );
25  }
26 }
27

PROBLEMS 29 OUTPUT DEBUG CONSOLE TERMINAL TEST RESULTS PORTS GITLENS Run: RecursionReverse
duey@MacBook-Air-Duy CS211 % /usr/bin/env /Library/Internet\ Plug-Ins/JavaAppletPlugin.plugin/Contents/Home/bin/java -cp /var/folders/w3/0x9k0j1s65gczs9x9ztngbw0000gn/T/cp_24312wgvg5vmxwc5gf71z7ej5.jar week02.lab.RecursionReverse
true
duey@MacBook-Air-Duy CS211 %
```

6.

```
3 public class RecursionDigits {
  Run | Debug | Windsurf: Refactor | Explain | Generate Javadoc | X
4   public static void main(String[] args) {
5     System.out.println(evenDigits(-123456789));
6   }
7
  Windsurf: Refactor | Explain | Generate Javadoc | X
8   public static int evenDigits(int n) {
9     boolean isNegative = n < 0;
10    n = Math.abs(n);
11
12    int result = evenDigitsHelper(n); // call evenDigitHelper (recursive method)
13
14    // if statement makes sure -0 is not returned as -0 but as 0
15    // e.g. evenDigits(-7), isNegative = true, evenDigitsHelper(7) returns 0
16    // return isNegative ? -result : result would return -0
17    if (result == 0) {
18      return 0;
19    }
20
21    return isNegative ? -result : result; // if isNegative true, return -result, else return result
22  }
23
  Windsurf: Refactor | Explain | Generate Javadoc | X
24   private static int evenDigitsHelper(int n) {
25     if (n == 0) {
26       return 0;
27     }
28
29     int digit = n % 10; // get last digit, e.g. 1234 % 10 = 4
30     int remainingDigits = evenDigitsHelper(n / 10);
31     // get remaining digits, e.g. 1234 / 10 = 123
32     // recursively call evenDigitHelper(n / 10)
33
34     if (digit % 2 == 0) { // if digit is even
35       return remainingDigits * 10 + digit; // keeps digit & appends to right of remainingDigits
36       // remainingDigits = 123, digit = 4
37       // 123 * 10 + 4 = 1234
38       // if we did 123 + 4, would we get 127, 4 wouldn't be appended
39     }
40     else { // if digit is odd, return but don't append
41       return remainingDigits;
42     }
43   }
44 }

PROBLEMS 29 OUTPUT DEBUG CONSOLE TERMINAL TEST RESULTS PORTS GITLENS
duey@MacBook-Air-Duy CS211 % /usr/bin/env /Library/Internet\ Plug-Ins/JavaAppletPlugin.plugin/Contents/Home/bin/java -cp /var/folders/w3/0x9k0j1s65gczs9x9ztngbw0000gn/T/cp_24312wgvg5vmxwc5gf71z7ej5.jar week02.lab.RecursionDigits
-2468
duey@MacBook-Air-Duy CS211 %
```

I'll make sure to do 7 and 8 later in the week when I have the time but I want to try to get this week's assignment finished asap :)