**Jeremy Cooley**

**IC5, Java 2150, Summer 2018**

**Practice IT! problems**

Work 5 of 6. If all 6, 1 will count as a bonus.

* ~~BJP4 12.2: writeNums~~
* ~~BJP4 12.6: writeSquares~~
* ~~BJP4 12.7: writeChars~~
* ~~BJP4 12.8: multiplyEvens~~
* ~~BJP4 12.9: sumTo~~
* ~~BJP4 12.11: repeat~~

**BJP4 Exercise 12.2: writeNums**

void writeNums(int n) {

if (n == 1) {

System.out.print(1);

} else if (n > 1) {

writeNums(n-1);

System.out.print(", " + n);

} else if (n == 0) {

throw new IllegalArgumentException();

}

}

**BJP4 Self-Check 12.6: mysteryXY**

mysteryXY(4, 1); 4

mysteryXY(4, 2); 8, 4, 8

mysteryXY(8, 2); 16, 8, 16

mysteryXY(4, 3); 12, 8, 4, 8, 12

mysteryXY(3, 4); 12, 9, 6, 3, 6, 9, 12

**BJP4 Exercise 12.7: writeChars**

void writeChars(int n) {

if (n < 1) {

throw new IllegalArgumentException ();

} else if (n == 1) {

System.out.print("\*");

} else if (n == 2) {

System.out.print("\*\*");

} else {

System.out.print("<");

writeChars(n-2);

System.out.print(">");

}

}

**BJP4 Exercise 12.8: multiplyEvens**

int multiplyEvens(int n) {

    if (n <= 0) {

         throw new IllegalArgumentException();

    } else if (n == 1) {

         return 2;

    } else {

         return multiplyEvens(n-1) \* (2\*n);

    }

}

**BJP4 Exercise 12.9: sumTo**

double sumTo(int n) {

if (n < 0) {

throw new IllegalArgumentException();

} else if (n == 0) {

return 0.0;

} else {

double divided = 1.0/n;

return divided + sumTo(n-1);

}

}

**BJP4 Exercise 12.11: repeat**

public static String repeat(String s, int n) {

if (n < 0) {

throw new IllegalArgumentException();

} else if (n == 0) {

return "";

} else {

String base = "";

for (int i = 0; i < n; i++) {

base = base + s;

}

return base;

}

}