CSE 142 Review Solutions – Monday Oct. 3, 2016

0. `sameDashes` (`Strings`)

```
public static boolean sameDashes(String s1, String s2) {
    while (!(s1.isEmpty() || s2.isEmpty())) {
        char first = s1.charAt(0);
        char second = s2.charAt(0);
        if ((first == '-' || second == '-') && first != second) {
               return false;
        }
        s1 = s1.substring(1);
        s2 = s2.substring(1);
    }
    if (!s1.isEmpty()) {
        return s1.indexOf('-') == -1;
    } else if (!s2.isEmpty()) {
        return s2.indexOf('-') == -1;
    } else {
        return true;
    }
}
```

1. `flipLines` (File Processing with Scanners)

```
public static void flipLines(Scanner input) {
    while (input.hasNextLine()) {
        String line1 = input.nextLine();
        if (input.hasNextLine()) {
            System.out.println(input.nextLine());
        }
        System.out.println(line1);
    }
}
```

2. `repeatedSequence` (Arrays)

```
public static boolean repeatedSequence(int[] a1, int[] a2) {
     if (a2.length % a1.length == 0) {
        for (int i = 0; i < (a2.length / a1.length); i++) {
            for (int j = 0; j < a1.length; j++) {
                if (a2[i * a1.length + j] != a1[j]) {
                    return false;
                }
            }
        }
        return true;
    }
    return false;
}
```

3. Implementing the `Point` class (Objects)

```
public class Point {
     private int x;
     private int y;

     public Point(Point other) {
         this.x = other.x;
         this.y = other.y;
     }

     public double distance(Point other) {
          double deltaX = other.x - this.x;
          double deltaY = other.y - this.y;
          return Math.sqrt(deltaX * deltaX + deltaY * deltaY);
     }

     public int getX() {
          return this.x;
     }

     public int getY() {
         return this.y;
     }

     public void setX(int x) {
         this.x = x;
     }

     public void setY(int y) {
         this.y = y;
     }

     public String toString() {
         return "Point[x=" + this.x + ",y=" + this.y + "]";
     }
}
```

4. `interleave` (ArrayLists)

```
public void interleave(ArrayList<Integer> a1,
                       ArrayList<Integer> a2) {
    for (int i = 0; i < a2.size(); i++) {
        if (a1.size() > i * 2) {
            a1.add(i * 2 + 1, a2.get(i));
        } else {
            a1.add(a2.get(i));
        }
    }
}
```