Assignment 5

Arrays & 2D arrays

Arrays vs. ArrayLists

- fixed size (arrays) vs. can change the size (ArrayLists)
- can use primitive types (arrays) vs. can't use primitive types (ArrayLists)
- different syntax
- both are 0-indexed!

declare an array:

```
int[] intArray = new int[10]; //declares an int array of size 10
```

set index 3 of the array (the 4th position) to 6

```
intArray[3] = 6;
```

iterate over the array and print out each value:

```
for (int i = 0; i < intArray.length; i++) {
        int value = intArray[i];
        println(value);
}
```

declare a 10x10 2D array:

```
int[][] intGrid = new int[10][10];
```

iterate over it and print out each value

```
for (int row = 0; row < intGrid.length; row++) {
        for (int col = 0; col < intGrid[0].length; col++) {
                int value = intGrid[row][col];
                println(value);
        }
}
```

Steganography

Premise: hide messages in pictures by altering with the red channel of the image
Two methods

- recover message (go from pixel array --> boolean array)
- encode message (use info from boolean 2D array to alter the picture and hide the message)

Helpful hints:

```
int[][] pixels = source.getPixelArray(); //store the pixels from an image in a 2D int array
```

int redComponent = GImage.getRed(pixels[row][col]); //get the red channel for a pixel at row, col

Histogram Equalization

1. create histogram from 2D array of luminances
   public static int[] histogramFor(int[][] luminances)
2. create cumulative histogram array
   public static int[] cumulativeSumFor(int[] histogram)
3. get the total number of pixels
   public static int totalPixelsIn(int[][] luminances)
4. equalize
   use formula from handout: (MAX_LUMINANCE * cumulativeHistogram[L]) / total pixels
   public static int[][] equalize(int[][] luminances)

Additional test class included to help you out :) Make sure all the tests pass!

ToneMatrix

See the demo online

Implement public static double[] matrixToMusic(boolean[][] toneMatrix, int column, double[][] samples) {...} in ToneMatrixLogic

Not too many lines of code! Just think carefully.

You get:

- boolean[][] toneMatrix
  - boolean matrix representing lights on/off on display
- int column
  - represents the currently selected column on the tone matrix
- double[][] samples
  - The sound samples associated with each row

Steps:
Figure out which lights in a column are on and populate the sounds array appropriately
Normalize sounds array
Return sounds array