Name: Anika Jallipalli  
Date: 3-12-2020

**15.07 Picture Lab Worksheet**

**Directions**: Make note of your responses to the following questions as you work through the activities and exercise in the lesson.

**Activity 5 Questions**

|  |  |  |
| --- | --- | --- |
| **Question** | **Yes** | **No** |
| 1. Is the method getPixels2D in the Picture.java class? |  | x |
| 1. Is the method getPixels2D in the SimplePicture.java class? | x |  |
| 1. Will the following code compile?  DigitalPicture p = new DigitalPicture(); |  | x |
| 1. Assuming a no-argument constructor exists for SimplePicture, will the following code compile?  DigitalPicture p = new SimplePicture(); | x |  |
| 1. Assuming a no-argument constructor exists for Picture, will the following code compile?  DigitalPicture p = new Picture(); | x |  |
| 1. Assuming a no-argument constructor exists for Picture, will the following code compile?  SimplePicture p = new Picture(); | x |  |
| 1. Assuming a no-argument constructor exists for SimplePicture, will the following code compile?  Picture p = new SimplePicture(); |  | x |

**Activity 5 Exercise Results**

1. Describe your method for keepOnly red, blue, or green.   
   pixelObj.setRed(pixelObj.getRed() - 255);

pixelObj.setGreen(pixelObj.getGreen() - 255);

pixelObj.setBlue(pixelObj.getBlue() - 255);

1. For the negate method, paste your code related to calculating and setting the values for red, blue, and green.
2. Paste a copy of the image that is the result of calling the grayscale.
3. For the method fixUnderwater, describe the algorithm you'd propose to accomplish the task.

The algorithm I’d propose to accomplish the task would do the following

1. Obtain a sample of the fish, find the avg. value and range of colors.

2. Calculate the range

3. Calculate distance from inputted values to sample values, and adjust image based on how close sample value is to inputted value.

**Activity 6 Exercise Results**

1. Paste the image that is the result of calling the method mirrorVerticalRightToLeft.

|  |
| --- |
| public void mirrorVerticalRightToLeft() |
|  | { |
|  | Pixel[][] pixels = this.getPixels2D(); |
|  | Pixel leftPixel = null; |
|  | Pixel rightPixel = null; |
|  | int width = pixels[0].length; |
|  | for (int row = 0; row < pixels.length; row++) |
|  | { |
|  | for (int col = 0; col < width / 2; col++) |
|  | { |
|  | leftPixel = pixels[row][col]; |
|  | rightPixel = pixels[row][width - 1 - col]; |
|  | leftPixel.setColor(rightPixel.getColor()); |
|  | } |
|  | } |
|  | } |

1. Describe the algorithm for the method mirrorHorizontal works.

|  |
| --- |
| public void mirrorHorizontal() |
|  | { |
|  | Pixel[][] pixels = this.getPixels2D(); |
|  | Pixel topPixel = null; |
|  | Pixel bottomPixel = null; |
|  | int height = pixels.length; |
|  | for (int row = 0; row < height; row++) |
|  | { |
|  | for (int col = 0; col < pixels[0].length; col++) |
|  | { |
|  | topPixel = pixels[row][col]; |
|  | bottomPixel = pixels[height - 1 - row][col]; |
|  | bottomPixel.setColor(topPixel.getColor()); |
|  | } |
|  | } |

1. Paste the image that is the result of calling the method mirrorHorizontalBotToTop.

|  |
| --- |
| public void mirrorHorizontalBottomToTop() |
|  | { |
|  | Pixel[][] pixels = this.getPixels2D(); |
|  | Pixel topPixel = null; |
|  | Pixel bottomPixel = null; |
|  | int height = pixels.length; |
|  | for (int row = 0; row < height; row++) |
|  | { |
|  | for (int col = 0; col < pixels[0].length; col++) |
|  | { |
|  | topPixel = pixels[row][col]; |
|  | bottomPixel = pixels[height - 1 - row][col]; |
|  | topPixel.setColor(bottomPixel.getColor()); |
|  | } |
|  | } |
|  | } |

**Activity 7 Questions**

* 1. How many times would the body of this nested for loop execute? \_\_70\_\_\_

for(int row = 7; row < 17; row++)

for(int col = 6; col < 15; col++)

* 1. How many times would the body of this nested for loop execute? \_\_90\_\_\_

for(int row = 5; row <= 11; row++)

for(int col = 3; col <= 18; col++)

**Activity 7 Exercise Results**

1. What value is displayed for count after the nested loop ends in the mirrorTemple method? 112
2. Paste the image that is the result of calling the method mirrorArms.

|  |
| --- |
| public void mirrorArms() |
|  | { |
|  | int mirrorPoint = 193; |
|  | Pixel topPixel = null; |
|  | Pixel bottomPixel = null; |
|  | Pixel[][] pixels = this.getPixels2D(); |
|  |  |
|  | // Left arm |
|  | for (int row = 158; row < mirrorPoint; row++) |
|  | { |
|  | // loop from 13 to just before the mirror point |
|  | for (int col = 103; col < 170; col++) |
|  | { |
|  | topPixel = pixels[row][col]; |
|  | bottomPixel = pixels[mirrorPoint - row + mirrorPoint][col]; |
|  | bottomPixel.setColor(topPixel.getColor()); |
|  | } |
|  | } |
|  |  |
|  | int mirrorPoint2 = 198; |
|  | Pixel topPixel2 = null; |
|  | Pixel bottomPixel2 = null; |
|  |  |
|  | // Right arm |
|  | for (int row = 171; row < mirrorPoint2; row++) |
|  | { |
|  | // loop from 13 to just before the mirror point |
|  | for (int col = 239; col < 294; col++) |
|  | { |
|  | topPixel2 = pixels[row][col]; |
|  | bottomPixel2 = pixels[mirrorPoint2 - row + mirrorPoint2][col]; |
|  | bottomPixel2.setColor(topPixel2.getColor()); |
|  | } |
|  | } |
|  | } |

1. Paste the image that is the result of calling the method mirrorGull.

|  |
| --- |
| public void mirrorGull() |
|  | { |
|  | int mirrorPoint = 345; |
|  | Pixel rightPixel = null; |
|  | Pixel leftPixel = null; |
|  | Pixel[][] pixels = this.getPixels2D(); |
|  |  |
|  | // Seagull |
|  | for (int row = 235; row < 323; row++) |
|  | { |
|  | for (int col = 238; col < mirrorPoint; col++) |
|  | { |
|  | rightPixel = pixels[row][col]; |
|  | leftPixel = pixels[row][mirrorPoint - col + mirrorPoint/3]; |
|  | leftPixel.setColor(rightPixel.getColor()); |
|  | } |
|  | } |
|  | } |
|  |  |