CSE 8A Programming Assignment 8

Name should be formatted as (last, first)
If you are working solo you may leave the right column blank.

Name:	Darren Yeung	Name:	
PID:	A15943292	PID:	
Email:	dyeung@ucsd.edu	Email:	

Part 1 : Inplace Image Modifications

1.1 CODE FOR CUSTOM FILTER

```
// Copy and paste your custom image filter method here
// Make sure to set the font to Courier New
// IMPORTANT: Make sure your code is properly formatted and commented.
// Code that does not have correct indentation and comments will lose marks.

public void grayscale(int topLeftX, int topLeftY, int width, int height){

      int startingheight = this.getHeight();
      int startingwidth = this.getWidth();
      if(topLeftY < 0){
        height = height - (topLeftY * (-1));
        topLeftY = 0;
      }
      if(topLeftX < 0){
        width = width - (topLeftX * (-1));
        topLeftX = 0;
      }

      for(int row = topLeftY; row < height + topLeftY; row++){
        for(int col = topLeftX; col < width + topLeftX; col++){
          if(col > startingwidth-1){
            break;
          }

          Color rightnow = this.pixels[row][col];
          int red = rightnow.getRed();
          int green = rightnow.getGreen();
          int blue = rightnow.getBlue();
          int lumi = (int)(0.2126*red + 0.7152*green + 0.0722*blue);
          Color newcolor = new Color(lumi, lumi, lumi);

          this.pixels[row][col] = newcolor;
```

```
        }
        if(row > startingheight-1){
          break;
        }

      }

    }
```

1.2 TESTS:

```
// Copy and paste the code from both test methods (including comments!) here
```

Flip Horizontal

```
public static void testHorizontalFlip()
    {
        Image img1 = new Image("res/pixel-heart.png");
        img1.explore();
        int w = img1.getWidth();
        int h = img1.getHeight();
        img1.flipHorizontalRect(0,0,(int)w/2 ,(int)h);
        img1.explore();



        //TODO : Add another test case

        Color[][] image2 = new Color[3][3];
        image2[0][0] = new Color(255,255,255);
        image2[0][1] = new Color(255,255,255);
        image2[0][2] = new Color(255,255,255);
        image2[1][0] = new Color(255,255,255);
        image2[1][1] = new Color(255,255,255);
        image2[1][2] = new Color(255,255,255);
        image2[2][0] = new Color(0,0,0);
        image2[2][1] = new Color(0,0,0);
        image2[2][2] = new Color(0,0,0);
```

```
        Image img2 = new Image(image2);
        img2.explore();

        img2.flipHorizontalRect(0,0,3,3);
        img2.explore();


    }
```

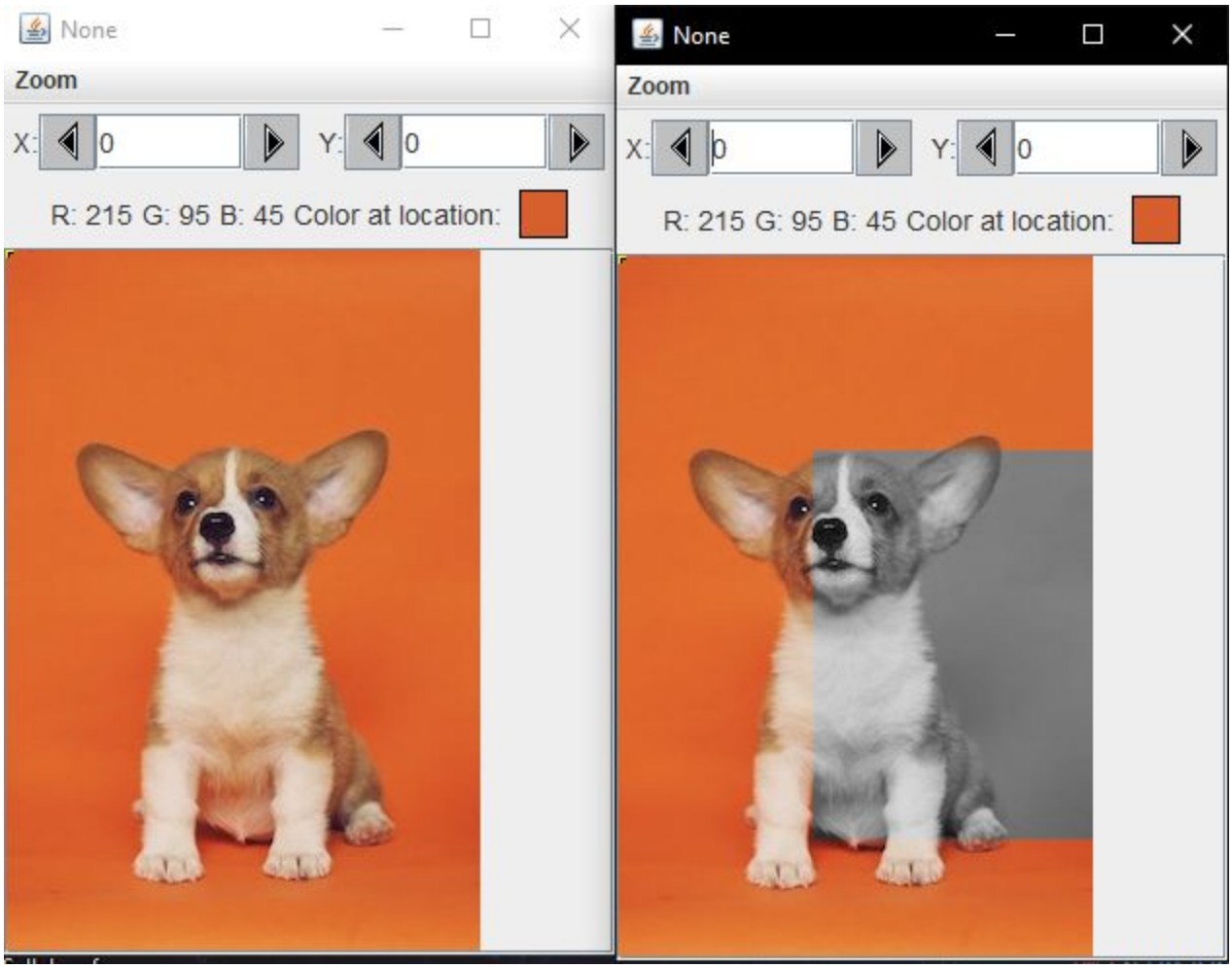
OwnMethod Test

```
public static void testown(){
      Image img1 = new Image("res/dog.jpg");

      img1.explore();
      //goes off to the right
      img1.grayscale(100,100,10000,200);

      img1.explore();

      Image img2 = new Image("res/cat.jpg");
      img2.explore();

      //starting from negative
      img2.grayscale(-100,-100,200,200);
      img2.explore();

      Image img3 = new Image("res/crane.jpg");

      img3.explore();
      //starting from within the image and staying in bound
      img3.grayscale(50,50,100,100);

      img3.explore();

    }
```

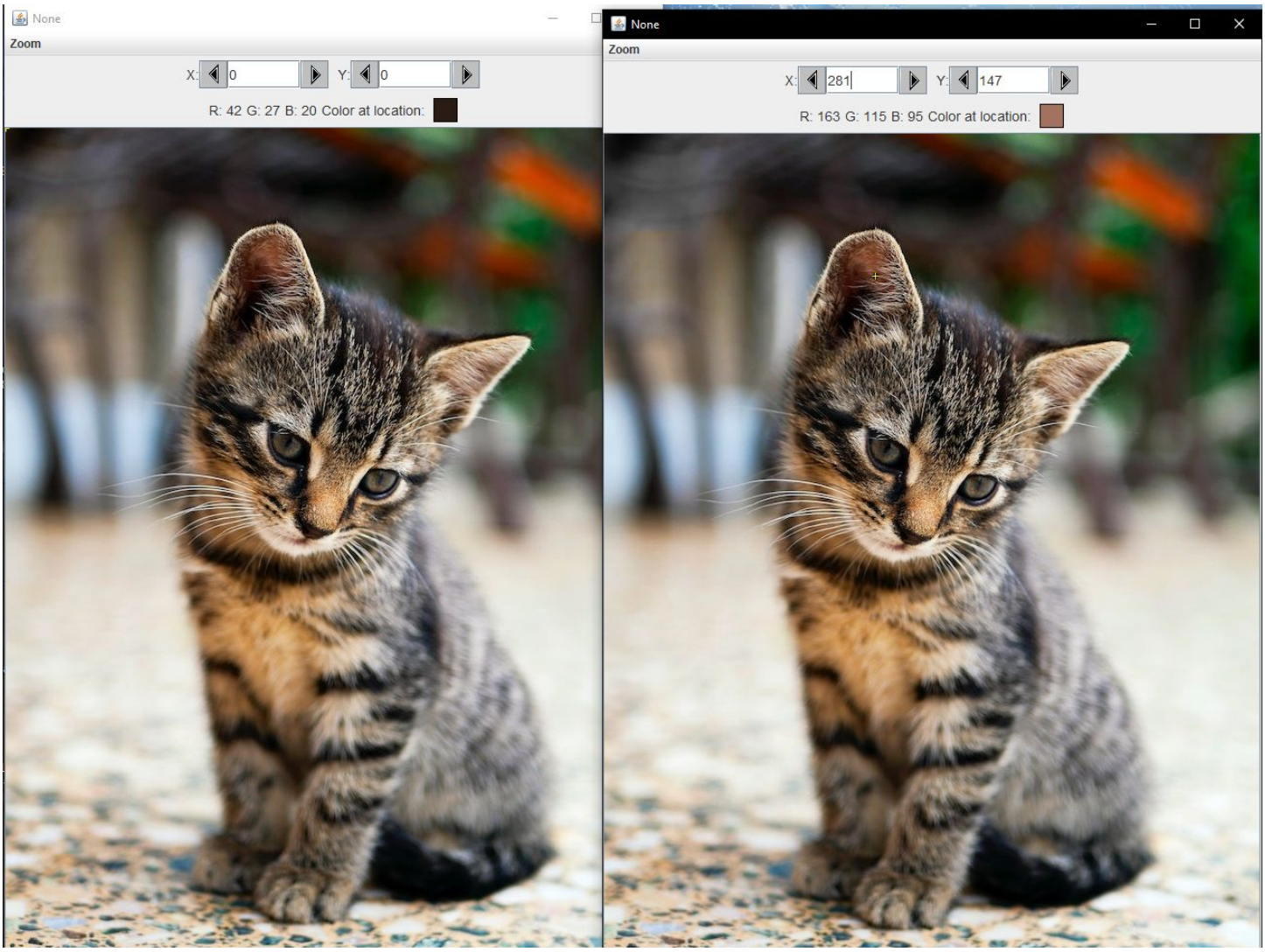
1.3 OUTPUTS

For the tests of your custom filter only, show the output from running each test.

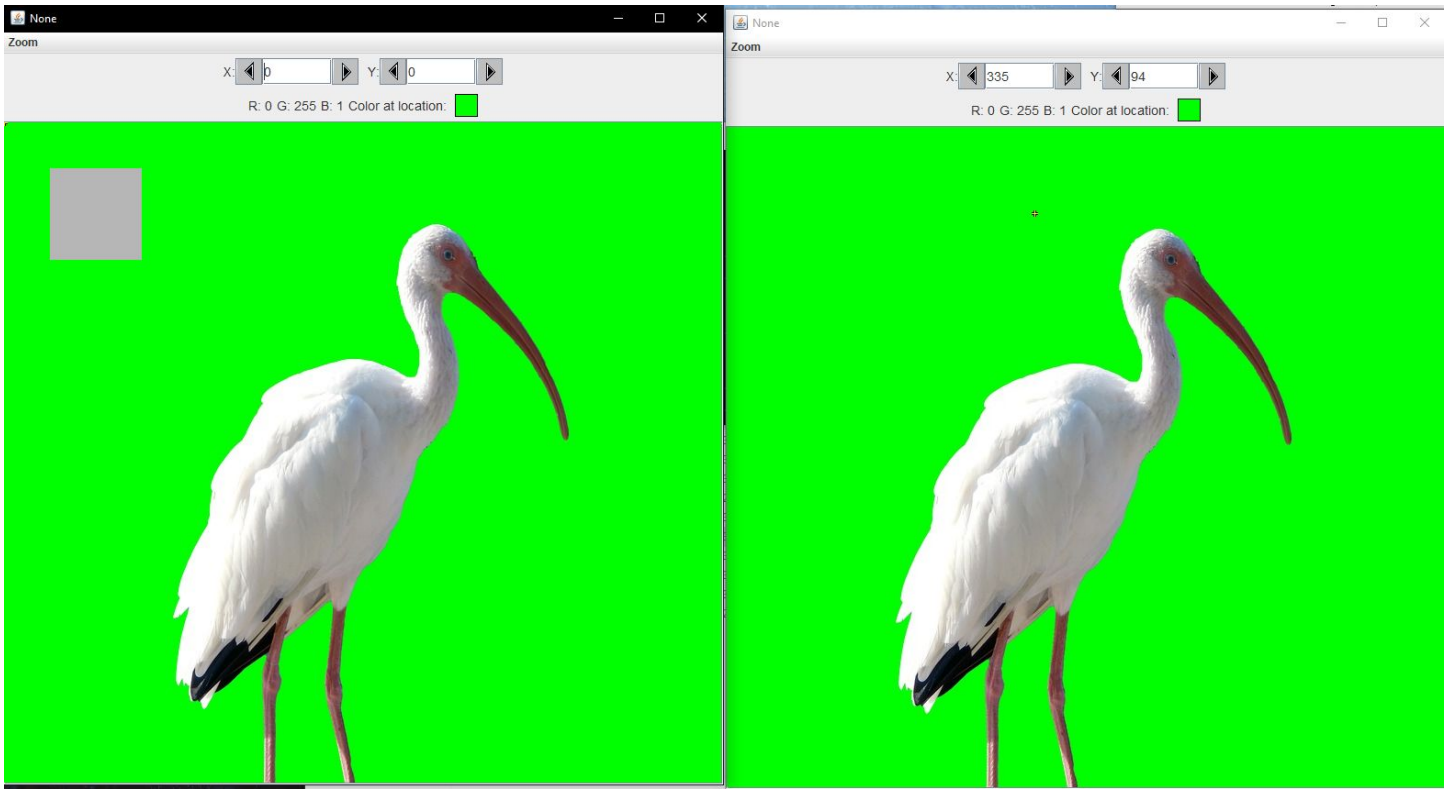
GOING OFF TO THE RIGHT



STARTING FROM NEGATIVE



STARTING FROM INSIDE AND NOT GOING OUT OF BOUNDS



1.4 EXPLANATION

For the tests of your custom filter only, briefly explain why you chose each test and how you know your code is working correctly (or not). If there are known bugs with your method, explain what they are.

Test 1 (going out of bounds to the right)

I chose this test because I needed to test if my method would work if I started somewhere in the middle of the image but provided a width or height that was too big that it went out of bounds of the picture. I know this code works because I started from 100,100, which is somewhere inside the picture. But I chose a width that was way out of bounds but my height was still in bound. The height was 200 so it should have went down to 299 which was what happened. However, for the width, it just kept going until it reached the end which was what was supposed to happen since the width was so big.

Test 2 (starting from a negative value)

I chose this test because I needed to test if my method can work if it started from a negative pixel. I know my method works because I started from -100 ,-100 and gave it a height and width of 200 . So it should have went down to 99 for both row and col. This is exactly what happened in my cat picture

Test 3 (starting in middle and NOT going out of bounds)

I chose this test because I had to make sure my method works if I started from the middle and did not out bounds of the picture like a normal person. I know this works because I started from 50 50 on my crane picture

and gave it a height and width of 100. It should of went to 149 for both col and row which was what happened. This is how I know my methods work.

There are no bugs in my methods.

Part 2 : Slideshow class

2.1 DESCRIPTION

Provide a brief description of the slideshow. What is the topic of the slideshow ? Describe what individual slides in you slideshow represent.

The slideshow is basically a movie with images and sounds that play together and when the sound ends, the next image is shown with its own sound. The topic of the slide show is different animals with random sounds. For the first slide, it is a cat with a broom sweeping sound. For my second slide, it is a dog picture with upbeatfunk playing. For my last slide, it is a picture of a crane with an owl sound in the background.

2.2 SLIDESHOW CODE: play() method only

```
// Copy and paste the code from your play() method in the Slideshow class here
// Make sure to set the font to Courier New
// IMPORTANT: Make sure your code is properly formatted and commented.
// Code that does not have correct indentation and comments will lose marks.


public void play(){

    for(int i = 0; i < (this.arraySounds).length; i++){
      (this.arrayImages[i]).show();
      (this.arraySounds[i]).blockingPlay();
    }
  }
```

2.2 SLIDESHOW PROGRAM

```
// Copy and paste the code from your main method in the Slideshow class here
// Make sure to set the font to Courier New
// IMPORTANT: Make sure your code is properly formatted and commented.
// Code that does not have correct indentation and comments will lose marks.

public static void main(String[] args){
    Scanner scan = new Scanner(System.in);

    Slideshow slideshow = new Slideshow();
    //first slide
```

```
    Image cat = new Image("res/cat.jpg");
    Sound broom = new Sound("res/broomsweep.wav");

    slideshow.addSlide(cat, broom);

    Image dog = new Image("res/dog.jpg");
    Sound funk = new Sound("res/UpbeatFunk.wav");

    slideshow.addSlide(dog, funk);

    Image crane = new Image("res/crane.jpg");
    Sound owl = new Sound("res/owl.wav");

    slideshow.addSlide(crane, owl);

    slideshow.play();


    System.out.println("Would you like to play the slide show again?");
    String answer = scan.nextLine();

    if(answer.equals("yes")){
      slideshow.play();
    }else{
      System.exit(0);

    }

  }
```

Known bugs or issues

*I*f you have known bugs or issues with your slideshow, let us know here. If you think your slideshow program works correctly, justify why.

My slideshow is working correctly because the sound plays with the picture. Whenever the sound stops, the next picture is shown and the next sound is played. When it ends, the terminal also asks the user if the slideshow should be played again. I typed yes and the slideshow did indeed play again. I typed something else like no and the program ended.