Anthony Borza

Project 1: Java 2D Graphics

CMSC 405 7980 Computer Graphics

University of Maryland University College

March 26, 2017

I. Brief Description of Project

This project required us to create three simple binary images of our choice. The size of each binary image was 25 X 25 pixels. We were then asked to use Java 2D graphic methods that include the following: rotate, scale and translate. Each image created had to show its original state, rotation state, scale state, and lastly translation state. The images are displayed in a user-friendly GUI, with a next graphic button at the bottom of the window, that is used to transition to the next graphic, and the state it is in. This document will contain screenshots of each shape and its state: original, rotation, scale, and translate.

II. Design and Functionality

The design of this project only required one java class called Project1.java. I found that multiple classes were unnecessary for this project, when it could all be done in one. The Project1.java class contains the following:

- Extends the JPanel class, and implements ActionListener.
- All variables are declared as private because it is more maintainable, and is good practice.
- Created three variables for Graphics2D to represent each image that will be created. The variable names are the following: graphicOne, graphicTwo, graphicThree.
- Created three BufferedImage variables: canvas1, canvas2, canvas3 that represents an image with 8-bit RGBA color components packed into integer pixels. The size of each image is 25 x 25 pixels, and they consist of binary numbers, 0's and 1's.
- Created a constructor that creates the GUI, the size of the GUI, and the placement of both the Label and button that will display in the GUI window. The constructor also implements the ActionListener, giving the button functionality.

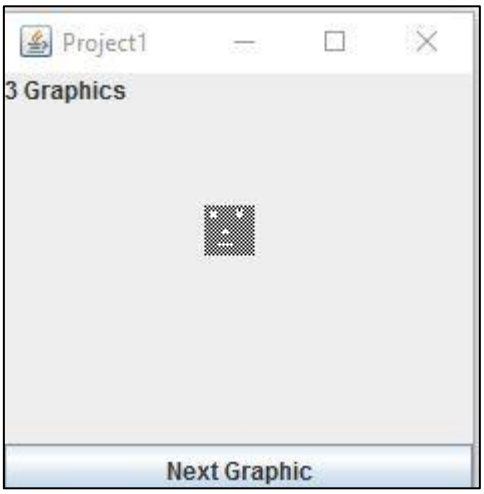
- Created a paint method that implements a switch statement with 15 case statements that handle creating the original image, rotating the image, translating the image, and lastly scaling the image. I used java documentation on creating a switch statement with Java 2D Graphics, and the reference can be found at the end of document (Oracle, 2017). I felt that this was the best way to transition between images. This was done for all three images, so that when you click the next graphic button on the GUI, it will transition to the next case and perform those assigned tasks.
- Created a drawArray, drawArray2, and drawArray3 method that generates each image and stores it in a 2D array. In those methods a for loop is used to display each transformation for each image.
- Created a actionPerformed method to had the buttons functionality. Each time the button is pressed, it moves to the next image, and calls the repaint method, so that it can display the next image.
- Created a main method that is used to execute and run the program.
- All code adhered to the Google Java style guide.

The following bullet points above accurately represent the layout and design of my Project1.java class, and its functionality.

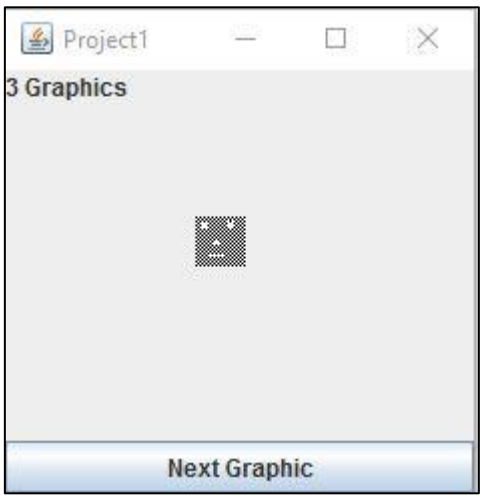
III. Screenshots

Image 1: Smiley Face:

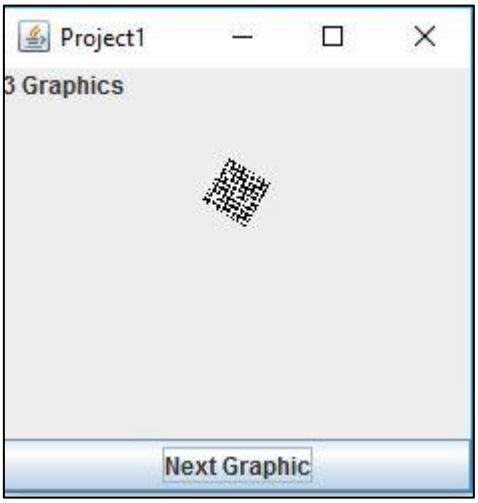
Original State:



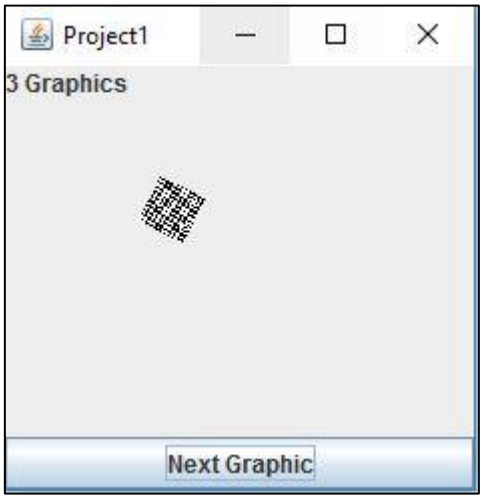
Translate State:



45 Degree Rotation:



90 Degree Rotation:



Scale 2 Times for X, and 0.5 for Y:

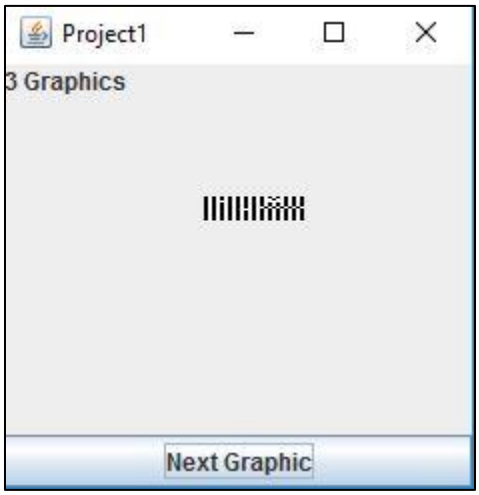
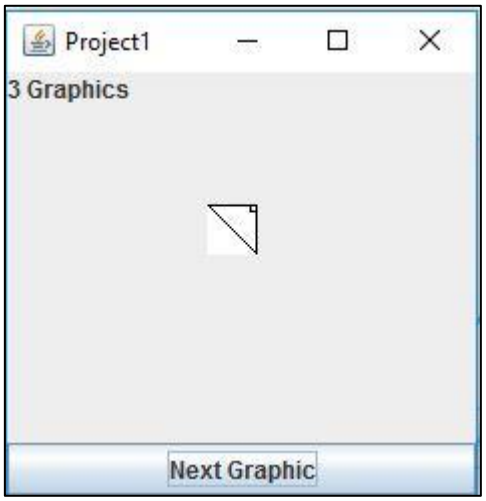
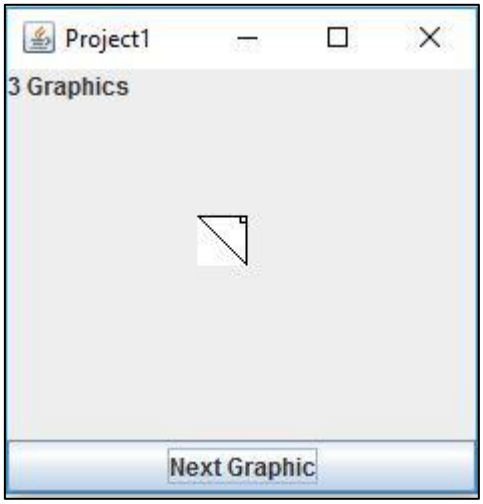


Image 2: Right Triangle:

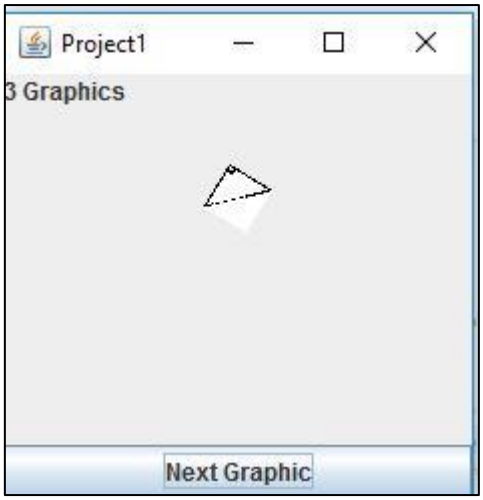
Original State:



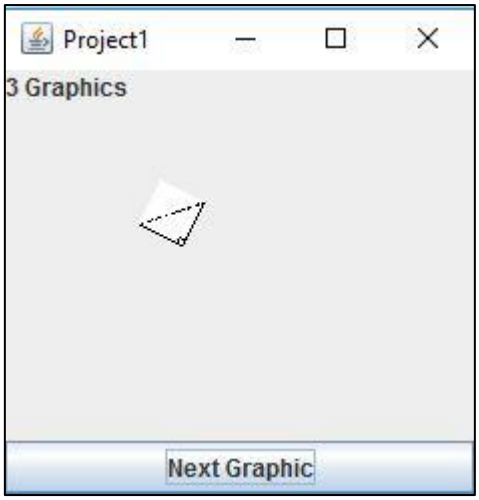
Translate State:



45 Degree Rotation:



90 Degree Rotation:



Scale 2 Times for X, and 0.5 for Y:

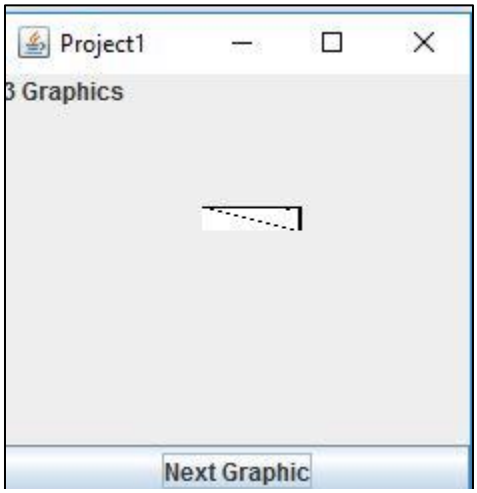
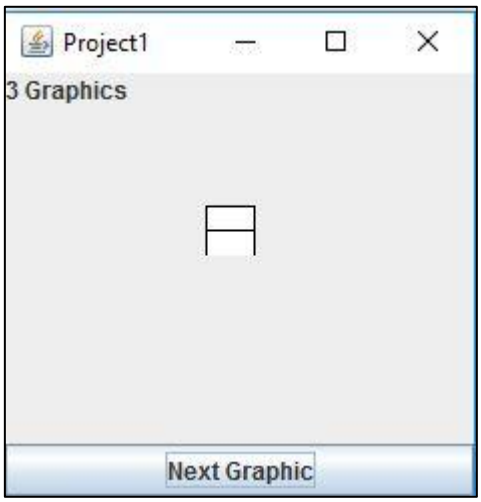
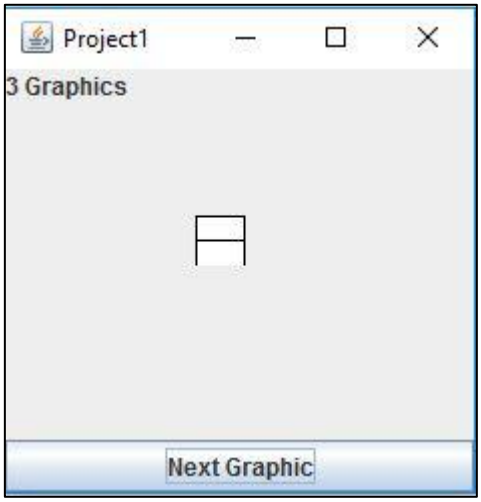


Image 3: Letter A:

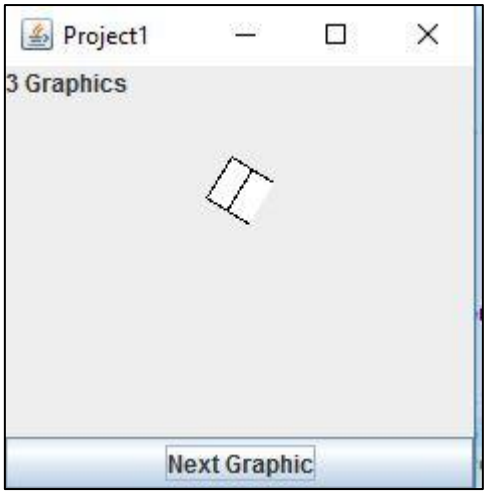
Original State:



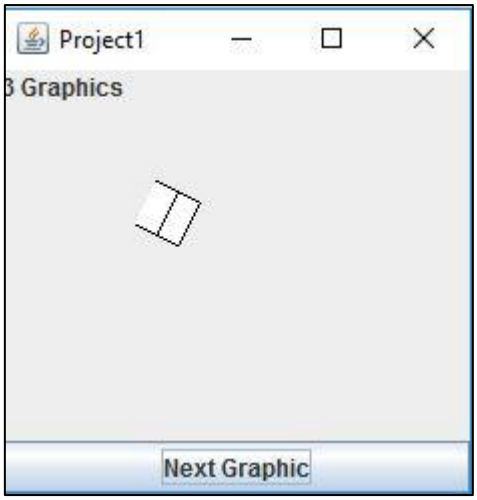
Translate State:



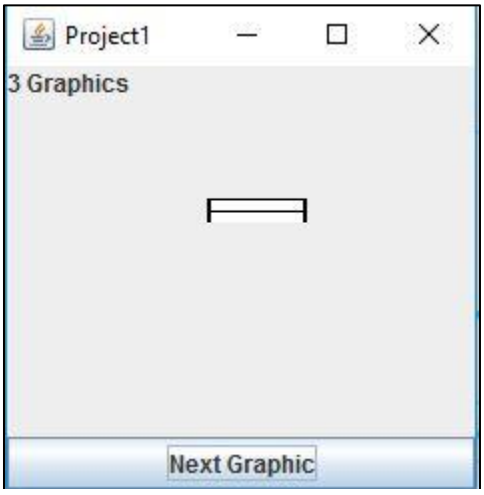
45 Degree Rotation:



90 Degree Rotation:



Scale 2 Times for X, and 0.5 for Y:



IV. Lessons Learned:

After completing project 1, there are many things that I learned throughout the process of creating 3 simple, binary 25 X 25 images using Java 2D graphic methods to rotate, scale, and translate each image. The first thing I did, like with all programing projects I am given, was to first read over the instructions, and understand what is being asked. Since this is my first time using Java 2D graphics, I found it challenging, but not too challenging since we were provided with detailed lectures. The hardest part was understanding how to create the images, and then implementing the 3 methods to: rotate, scale, and translate each image. I found the example code that the professor provided in the ask professor section of the course to be very helpful with getting started. The example provided us with code to create one of the images that then made creating the code for the remaining two images easy. Overall, I am satisfied with how my program functions, and believe by only creating one java class, it simplified the program and made it more efficient.

Reference:

Oracle. (2017). Transforming Shapes, Text, and Images. Retrieved from https://docs.oracle.com/javase/tutorial/2d/advanced/transforming.html