Andrew Petriccione
CSCI 373
Professor Sheaffer
Fall 2016

Project 2 Change-Log and Notes

1. Begin work and make a Triangle.

As I started on Project 2 I started by copying my Project 1 code and made a class for Triangle. It took me a little while just to get a regular triangle to show up on the screen. Initially I just made a triangle 'from scratch' even though later I would make them from Line objects and just use the draw method in my Line class. I didn't have the forethought to do that at the time though.

2. Got bresenham's algorithm working so I can do lines in any direction now--now I can begin the actual assignment.

I had some trouble getting the lines to go in any direction until I got a hold of the Bresenham algorithm. Once I could draw lines in any direction I could start work on the real assignment.

3. Merged my testing branch with the master branch since I'm the only working on this and extra branches didn't seem necessary.

4. Made some progress; have triangles, arrows, circles, still experimenting with how Bres alg works to learn from it to apply to other shapes.

5. Add parallelogram class and change logic for line draw method.

Made the parallelogram class in a pretty straightforward way and updated some of the control flow for the Bresenham algorithm as I came to understand what it was really doing.

6. Finish all shapes, make Diamond with static factory method, begin experimenting with event listener for window size with a while loop but that might be defeating the point of listener...?

I was trying to wrap my head around the event listener and thought I needed a while loop to be running to use it. That didn't feel right though, and I didn't need that. The static factory method worked, but I took it out since it made things more complicated when I needed to call the constructor for Diamond. If I had more time I would have liked to use it.

7.Add all shapes to Shape abstract class, add event listener print messages, & add array list for Shape objects instead of just Box objects.

In project 1 I had an array list to track my Box objects so it was easy to adapt that to track Shape objects instead. Making an abstract Shape class to hold all the Shapes let me access their methods from that array list and I used some print messages for debugging purposes.

8. Get erase to work from interface, need to bugs but it's erasing from the array list when it erases from the palette--need to test that.

This was when I got the erase to work at first. I wasn't sure that it was doing everything I needed it to, but was able to figure it out with debugging. I wish I had realized sooner that I didn't need to erase and delete from the array list if I was going to redraw the array list since it won't draw things that are removed.

9. add String modifier to all draw method calls to allow up and down etc shapes but may be a better way to do that with a sub class or something. Box now erases by having the line objects call erase on themselves which might be the best way for all erases to work. The lines just need to be instance variables to do this.

I was still trying to find a way to select different drawing directions (up or down) and was trying out passing a String to determine the direction. Later I realized that I could have used an enum type to handle this sort of thing, but I ended up removing the String for everything except my Triangle class to say up, down, left, or right.

10. Found call to set colors, can make methods to make calls easier or not, everything is erasing if a little liberal with the clicking area to do so. Haven't cracked the resizing problem.

I had some trouble understanding the resizing and how to handle it. After some trial and error and discussion I saw that I could just redraw everything. Later I also realized I needed to erase everything first too.

11. Fix erase issue, still getting phantom arrow redraws on window re-size. Fix erase arraylist out of bounds error with if not empty list test. Add addShape() method instead of trying to add shapes in their own draw method since that was adding the shape and the lines it was made up of.

I really wanted to add Shapes to my array list when they were being drawn but that caused problems with the resize. I also had some code in my Arrow class that was causing the end coordinates to be recalculated and cause them to grow on resize. That was pretty easy to spot though.

12. Identify several bugs including erase coords being off, redrawing arrows and adding them to arraylist happening in a loop somewhere, and redrawing things that shouldn't be on resize. Need fixing but looking okay before the resize. Also pulled in new jar file Sheaffer updated yesterday.

Added the updated jar file and updated my erase method to use the erase in the Line class of the object being erased.

13. Fix resize loop. Was adding boxes to array list in the Box draw method causing the issue. Still not erasing as quickly as I would like, and triangle boundary issue and getting the reflection redrawn on resize as well which all need to be fixed.

Had an issue where my Triangles were getting reflections because of my weird direction String. As much as I liked the idea it had a lot of issues, but ultimately it did work out. I just needed to take into account a need to update the coordinates being passed up to the abstract class for erasing. I also had to track down some code that had Boxes getting added to the array list in their draw method as I had had it previously.

14. managed to break erase on click--not erasing lines or arrows although boxes and stuff are erasing despite being MADE OF LINES! And despite all my efforts with git I didn't do a push before making changes to my semi-working code so I have to figure it out.

Made some changes and couldn’t erase lines for a day. Despite all my efforts of using git I did not do a push before I made some key changes and was left without a backup of my working code. Never again.

15. Going through and adding each shape by uncommenting one shape at a time and printing out the contents of the array list to see where things are going screwy b/c it seems to have trouble with erasing the last couple shapes with a bunch of stuff on the screen. Or it could be b/c of overlapping stuff.

Further debugging.

16. Reached a point where things aren't perfect but it's working and I have something to deliver. Added some wiggle room to erasing vertical and horizontal lines/arrows.

Made sure to back up and push my code to github when I reached a point where everything was working alright. I added some wiggle room to the Line class’s erase method to make it easier to click on them.

17. Add getWidth and getHeight so I can call them in the connector class to connect the generic Shape class. Also want to change erase by adding getLines() method to interface so I can erase by finding the lines instead of using a bounding box for diagonal lines etc that take up a lot of space.

Added a couple more methods to the interface to facilitate my Connector class and connect any Shape instead of just boxes. Tried to make it so erase could work off just the lines in a shape, but I kind of ran out of time. There are also better ways to do it instead of accessing all the Line objects inside each shape.

18. Wrapping it up for shipment. Most things seem to be mostly working.

It's not perfect but it works and I have to turn it in. I could keep working on it and make several improvements including the way the erase works, and to make it faster. I could analyze if I need all the classes I have to try and streamline it. I could see about removing the String modifier I'm using and using an enum type instead. I could do a lot more testing, and it would be cool to add more interactivity. Overall I learned a lot working on this iteration of the project and would like to keep improving it.