Assignment 10 README

For our final assignment, we took full advantage of the code which was provided to us. By utilizing the view as well as the controller, we could easily implement the code necessary to run an animation in java. The only things that really remained were making functions to do the calculations.

As was said in previous parts of this assignment, our layout is very customizable. Since this is the case, we were able to extend our current interfaces to encompass the same necessary attributes that Vido did. Our motion classes, which were made early on, make brand new shapes after each change. Before this week, those shapes were not utilized. However, we chose this design style because it kept our old shapes from being mutated.

This paid off well this week. Since our motions were all separate subclasses, all we needed to do was save these shaped to an array list, and send it to Vido’s view. Besides this, we also made a scrollbar that will appear if any shape every extends our windows size. However, we also calculated the maximum size our window would need to be by finding the furthest coordinates ever used in each direction. Besides this, all we needed to do was update the existing code base with what is known as a tween method.

Tween is a method that finds the shapes motion-value at any tick in the animation. This means that a shape’s motion technically is elongated over several frames. In order to refresh our page and have our shape not instantly jump to its end value, we needed to make a method to find out what percentile of change should have occurred so far. This is exactly what tween does. After tween is called, the screen refreshes to a panel in which the shame is in this new location based on the tick-time.

Thank you for taking the time this semester to critique our work and provide positive encouragement. We hope you have a great break.

* Kraig and Tom