Lauryn C. Hansen - u0264212 - member of Group 2
CS6018 - Phase 2 Code Review Summary

For the code review I met with Reshma Raghavan from Group 1.

- Each person picks a piece of code from your own project (class, method, etc) that you think can be improved. Discuss that code in detail, improve it, and commit those improvements (ie, refactor something with a helper that has fresh eyes).
    - Team 1 included an action bar at the top of their UI which functions similar to a back button rather than just using the built in Android Studio IDE back button to navigate the app. Team 2 plans to implement an actual back button in their UI. I was able to complete this upgraded implementation. I had to update some of our xml files to make it so that all UI components were still displayed properly. This is commit #21 titled “Lauryn Phase 2 Code Review Implementation” in our group repo.
- Pick a piece of functionality that both of your projects include. Discuss the similarities/differences between your implementations
    - Team 1 is using a color picker while Team 2 has buttons for 3 different color options.
    - A similarity is that we both implemented this within our DrawFragment.
    - Team 1 used the kotlin path library for the actual drawing. And the kotlin strokeWidth method for pen tip size
    - Team 2 had color button listeners whose callbacks manually set the color of interest. And we had buttons for increasing and decreasing size whose callback methods increment a member variable of our view. We used an enum for defining pen tip shape.
    - Pro for team 1 method - beautiful UI/functionality
    - Pro for team 2 method – more accessible, less beautiful
- Pick part of your code that you don't think is well tested. Work with your partner to add more tests to exercise that part of your project and commit them to your repo
    - Team 1 tested buttons in their composables, team 2 was not doing this - this will be an improvement that team 2 refactors. I was unable to get this functionality to work properly. I was unable to get the following code to compile - I will continue to debug this issue.

      ```
      @get:Rule
          val composeTestRule = createComposeRule()
      ```