CMPT 276 Group Project: Assignment 3

Owen Liu, 301369997
Yoobin Nam, 301538106

Smells Identified in StartFrame.Java

- Unclear Variable Names
    - We found that variable names for Images and Labels were often too short and abbreviated in a way that could lead to misinterpretations or failure to understand in some cases. Although refactoring the names made them longer, it will be easier for future coders to understand and leaves little to no room for misunderstanding. These changes can be found in the commit SHA: 6b069b395b1f25114eb5f0a83bf8473802ef2c88 and 437798c00d42f89ce45c8dcae2e6f1c423f8f3c1

- Removing hard-coded paths to files
    - Before refactoring, the paths to files were hard coded so that you have to find the exact spot where it was being instantiated and then manually type the string path at the location. To increase the ease of changing the path, we created two variables for the images used in StartFrame to hold the string of the path to the image. This allows for fast changes to images, as the variables lie at the top of the class and any methods/classes that may ever need the path of the images can easily access them with the variable name. These changes can be found in the commit SHA: 6b069b395b1f25114eb5f0a83bf8473802ef2c88

- Unsound creation of filling Labels with images
    - Looking through the code for setting the images for labels, we found that the images did not actually fill the entire Label space created and thus making unproportionate images in comparison to the Label size. This also means that when aligning the labels, labels could intersect and overlap but visually the images could be nowhere near close to each other since the images don't resemble the size of the area they are put into. So We created a function called returnImage that takes in the path to the image, and a label to size it to. The function turns the image specified at the path into a buffered image to scale it to the size of the label by using label.getWidth and height respectively. This should allow for better alignment of items in the frame when dealing with each label and other interactions with newly added components to the frame, as visually it will be more accurately represented/precise. Shown in commit SHA: 6b069b395b1f25114eb5f0a83bf8473802ef2c88 and 437798c00d42f89ce45c8dcae2e6f1c423f8f3c1

- Added More Documentation to play button and newly added function returnImage()
    - We added extra documentation to startFrame as the play button was missing some documentation to its functionality and adding the new getImage function also needed new documentation as it will be easier to understand and perceive just by looking at the documentation provided by the developers of it.

Smells Identified in gamePanel.Java

- Removed unnecessary memory usage for maze creation
    - To create a maze for our game, we need to use a Barrier array to save temporary walls. Before changing it, it generated 10 Barrier arrays but we only need 4. Six memory for array spaces were useless and it was one bad smell. After changing it, it is expected that the execution time of the Game Panel would have been slightly reduced. In commit SHA: b2b54060e0352b3b3b6733dc59064e217900f750.

- Changed maze cells layout
    - It was using rows and columns to create a maze. It made long if condition statements and was hard to read and understand. Therefore, we decided to create a 2d-array to handle the maze, and add a comment to make it more clear. In commit SHA: b2b54060e0352b3b3b6733dc59064e217900f750.

After refactoring our code, the tests for each section still pass without any errors.