**Final Project Description Update:** The previous description is pretty spot-on, besides the fact that there is no increasing difficulty feature. I added on two features. The first is a local multiplayer. With this, I added another buggy that Player 2 plays with – this one is from SDC, another buggy team. What’s interesting about this multiplayer is that both buggies drive and interact on the exact same track. Buggies can drive through each other without harm, but are competing for power-ups, and if one player uses a power-up like a pothole remover, it removes all potholes for both users. The objective of this mode is now to beat the other player. I like how I implemented this feature because it adds a lot more strategy and incentives to the multiplayer, instead of just two people trying to survive. The second feature is a global scoring system. Score is now indicated for single-player game mode only, and the top five best scores are stored globally, in a separate file. If one of the scores on this list is matched or beaten, the user can put in their name in the Python Shell running the program, and will be included in the score list. This list can be accessed by clicking the high score button on the start screen, and prints in the Python Shell. I also do not have squirrels implemented as a power-up source, as I thought it would be more important to put time into incorporating other features.

**Final Structural Plan Update:** The UI screens that I have included are a start screen, a game over screen (for single player), a player one wins screen (for multiplayer), and a player two wins screen. The file system of my project has changed a bit. My main file is now the one that has all of the animation on it, and the start and end screens are now individual objects with their own respective files. There is also an object for the score list, which is in the start screen file, as it can be accessed in the start screen.

**Final Algorithmic Plan Update:** 2.5D has been implemented. Another pretty large area of complexity, which I didn’t mention before and doesn’t show on the code, is all of the art that I did. I have 16 (?) different sprites in my TP\sprites directory, which were all hand-drawn, and a lot of them were not trivial and took a decent amount of time to make. The pedestrian and some of the screens especially took a while to do. I did this all on piskel, which is a free pixel art application. I was able to give my game a pixelated arcade-like graphics effect while making the game still run and look really clean, which I was shooting for, and am proud of. Finally, I added a global high score list. I started to implement this one at 11am on the Thursday that the TP was due so has very low graphics quality, but I used pickle to read and write from a global file, scoreboard.pkl, to maintain scores. I also accessed this file to check if a score makes it onto the high score list, and updated it accordingly.