Final Project Reflection

For my final project I chose a few objects I could make easily from rectangles since I already had the desk. I found my weakest points in this class were when I had to mess with vectors and indices on three-dimension objects so I figured it was better to find objects I could build by modifying the same model rather than trying to build new models for each object. Theoretically with a square or rectangle this could be used to build any object I wished since a computer uses a bunch of square pixels to render objects but technically, I probably should’ve figured out how to build new objects with new vertices and indices because it would be much more difficult to use this technique to build some of the other objects in my scene. Since I used the same vertices for every object I had to use scaling, positioning, and rotating to build the object the way I wanted it to look, and in the case of the laptop I had to use three such objects positioned perfectly to account for the top, bottom, and laptop screen. In addition, I figured out how to apply textures including applying different textures to different objects within a multi-object program which is how I was able to give my laptop a screen. I was also able to modify my texture, so each object looks more realistic than it had in some of my past milestone submissions.

Users can navigate my three-dimension scene by using either the mouse to look around, since the camera moves when the mouse moves, and by using the “wsad” keys to move around. It is possible to navigate my scene using just the “wsad” keys or to get a quick look using just the mouse, but it is better to use both to get the best view of the objects in the program. The camera is completely controlled by callbacks, OpenGL is no different from other programs in that it recognizes information gathered from hardware when focused on the program and the callbacks simply catch that information and utilize it to perform functions like moving the camera, changing textures, or whatever the programmer programs. These calls are performed through GLFW which creates and controls the programs window.

I will admit my code is not as organized as I would like it to be. A lot of my code for my programs and objects is similar but it seems when I start messing with things my code starts to break so I have not gone through and tried deleting some of the code I believe could be unnecessary, for fear of having to re-do a good portion of the project if I do. I did put all of the code regarding programs, necessary variables, and each object together making sure that it was easy to read what code went to what program and where it is used. But I think I probably could go through and delete a bunch of code that was mostly re-written because it worked the first time and I did not want to mess with it. Some of my variables could probably just be written once and used the entire time, rather than being re-written for each program and I probably should’ve made functions to build each object in order to make URender() a little easier to read. But I did add spacing and comments to the top of each object, so it is easier for another programmer to see what parts of the program render each object. I also named each variable appropriately, so it was easy to tell what program a variable belonged to and what it was used for. I think the program would look cleaner if I had added more similar variables as class variables and helped to reduce the lines of code I wrote, but once again the program works and I do not want to mess with anything that might make the program crash or cause an object to not render correctly because it might make it more difficult for me to re-build the project in the same way.