Joseph Collett

CS-330

February 17, 2023

**7-1 Project Reflection**

For the 3D scene we were tasked to perpetually complete integrative milestones to an ultimate final project, I chose objects that I use on an almost daily basis. Since I use these objects regularly, I thought it might help me design the approximate size ratios in relation to each object more accurately.

Although with more time, I could have developed my code much more decoupled and modular, I created several functions to aid with code reusability. These functions include framebuffer\_size\_callback, mouse\_callback, scroll\_callback, processInput, and loadTexture. Framebuffer\_size\_callback aids with proportioning the scene as the user resizes the window. Mouse\_callback is a callback that executes when the user moves the mouse cursor in the scene. It acts directly with the direction of the mouse cursor. The scroll\_callback is called when the user scrolls through the scene with their mouse scroller. Since the method of applying texture to objects is the same on an object-to-object basis, the loadTexture function handles the logic for that repetitive process. ProcessInput is the most important callback for scene navigation. Consider the following keyboard navigation button handlers:

1. W is forward
2. S is backward
3. Q is up
4. E is down
5. A is left
6. D is right
7. P changes the projection from orthographic to perspective
8. ESC closes the scene window
9. Y moves the mobile light cube up
10. U moves the mobile light cube down
11. J moves the mobile light cube right
12. H moves the mobile light cube left
13. M moves the mobile light cube backwards
14. N moves the mobile light cube forward

To handle the first-person navigation through the 3D scene, as the user hits the appropriate keys, the processInput callback is executed. Then, the cameraPos variable is incremented or decremented by the product of the cameraFront, cameraUp, or the normalized cross-value of the cameraFront and cameraUp value.

All in all, this was a very interesting project. Although I'm not that interested in developing computer graphics in the future professionally, this course and final project helped me increase my C++ programming knowledge and skill set, as just about every aspect of programming was inherent to this project, such as object-oriented programming, structures, namespaces, pointers, functional programming, and memory management.