Saram Nadeem

CS-330

2/25/24

Final Reflection

**Justify development choices for your 3D scene**.

For my three-dimensional scene I have a desk, pencil, mug filled with water, and a computer with a base. My scene primarily consists of the items located at my desk while I’m working on the project. I also thought it would be an interesting challenge to try and create both a pencil with all its features as well as a handle for a mug. The pencil includes four different primitive objects which are a cylinder for the body and eraser, a torus for the eraser connector, a tapered cylinder for the sharpened head and then a cone for the actual graphite section of the pencil. This one object was able to meet the requirements of using at least four primitive objects for the project.

**Explain how a user can navigate your 3D scene**.

A user can navigate through the 3D scene with the use of both their keyboard and mouse. The keyboard functions as it would in a computer video game with the use of (WASD) as the front back and side to side. We also have (Q&E) as our up and down. The mouse functions as a viewer, so the user can move their mouse to get a complete 360-degree view but must use their keyboard to move in the coordinate plane.

**Explain the custom functions in your program that you are using to make your code more modular and organized**.

Thanks to Professor Battersby’s meshes, my code is a lot more modular and organized. The code has become a lot cleaner and easier to read now that all I have to do is call upon our meshes file and render in the actual object. This saves us the hassle of having long lengthy lines of code that take up tons of space only to create the positions of for our desired shapes. Another step I took to make the code more organized was to separate and label each object respective meshes. For example, our pencil mesh was composed of four different primitive objects, so I labeled each individual mesh according to what part of the pencil it was. This greatly assisted in organizing each individual component, as well as whole objects from one another within our source file.