CS-330

Professor Graham

Kyunghoon Lee

17 October 2024

**Module 7 – Design Decisions**

The main reasoning for the objects I selected in this scene was finding a balance of complexity and varied objects that would not be too difficult to derive from the basic shape meshes we were given. During the initial planning of the scene, I realized that most of the objects on my desk were really comprised of variations of boxes with different dimensions, which led me to adding objects like headphones and the mouse as an opportunity to manipulate some of the other shapes, such as a torus or sphere. To approach programming each of these objects, I first tried to visualize the object as a low-polygon model with fine details removed. I was then able to break down the different parts of each object into examples of the basic shape meshes and begin planning how to combine them to achieve the final object I desired. For example, when modeling my monitor stand (comprised of 2 articulated sections connected by joints) I was able to break it down into cylinders to make up each of the joints, and elongated boxes to make up each arm portion of the stand.

The navigation system for my 3D scene is based on a combination of keyboard and mouse inputs, not unlike those commonly used in PC games. The WASD keys act as binds for moving the camera (forward, back, left, right) in the same format as a typical arrow cluster. The Q and E keys can be used to move the camera up and down. The mouse is locked to the rendered window and is used to pan the camera, or “look around”. Scrolling up and down can be used to modify the movement speed of the camera, and the P and Q keys can be used to switch between perspective and orthographic views.

The primary change I made to my code in working on this final project was to the overall RenderScene() method. After adding and assembling all the shape meshes for the objects in my scene, I realized that the method was just a list of shapes and their parameters, differentiated only by comments I added myself. For an object such as the monitor stand or headphones, it would be difficult to make modifications to the individual shapes if anyone other than myself were to look at this code. I decided to break each object down into its own method, with individual calls to draw each shape mesh within the methods. This way, RenderScene() only had to call the method for each distinct object, making modifications easier and my code more digestible.