CS 330 Project Reflection: Design Decisions

The 2D image that was used throughout the development of this project included one Fujifilm camera, one tangerine, one plastic measuring cup, and a sheet of paper towel. The four primitive shapes selected to represent these objects included the cylinder, the sphere, the cube, and the plane. The plane of course, being reminiscent of a flat surface, was used to depict the sheet of paper towel. The cube and cylinder were used to portray the body and lens of the camera respectively, as well as the handle and cup of the measuring cup. Due to the natural spherical shape of the tangerine, the sphere was used to represent it.

The input devices used with this 3D scene are the keyboard and mouse. Specifically, the keys W, A, S, D, Q, E, and P as well as the x-y planar movement of the mouse. The mouse movement and the keys manipulate the position, perspective, and orientation of the camera, allowing a user to easily navigate around the scene and view it from every angle. The keys W, A, S, and D deal with the forward, left, backward, and right movement relative to the camera’s current orientation. When the 'W' key is pressed, the camera moves forward in the direction it's facing, this is achieved by adding the cameraFront vector, modified by the camera offset to, the camera’s current position. The others work similarly, the ‘S’ key does the same thing as the ‘W’ key except that the modified cameraFront vector is subtracted from the camera position, moving the camera backwards. The A and D keys use cross products of the up and front vectors to move the camera left and right. The Q and E add and subtract the modified cameraUp from the camera position to move the camera up and down. When the P key is pressed, a global boolean value is toggled, this value is checked in the rendering loop to determine if the perspective or orthographic view should be used. Mouse input is taken as an x-y coordinate pair which inside a certain range, determines the orientation of the camera, allowing the user to rotate the direction of the camera.

To keep everything as orderly as possible, the various functions of several custom functions were made to be reused. The shape creation functions are able to be used to make multiple iterations with different applied textures and at different scales, translations, rotations, and transformed as needed. In the scene, the cube and cylinder creation functions are used twice to create both the camera and the measuring cup. Additionally the texture creation function is designed to handle multiple textures so that it can be reused as many times as necessary for each additional texture you want to include, the scene used 5 textures in this manner. The cleaning functions like the UDestroyTexture and UDestroyMesh functions are simple enough that they can be reused to destroy texture and mesh data for each texture and shape used. Modularization and reusability of code prevents redundancy and increases efficiency while making the purpose of the program easier to understand. This also makes refactoring and debugging easier because each essential function of the program is broken up into its own segment so that it is easy to find.

This program fulfilled the required functionality by using shape creation functions to generate four primitive shape types. These shapes were used to create the objects depicted in the original 2D image, all objects were textured and lit with multiple lights. Movement controls were implemented to allow for navigation around the scene. The result is an accurate 3D scene representing the original reference images.