Reese Thurman  
CS 330  
Module 7 Design Decisions  
6/22/2024

1.) The 3D scene was developed based on a still life, two-dimensional picture that depicts a table with a brown wicker basket placed on top. The wicker basket is filled with red apples and apple tree leaves. Some of the fruit and leaves are also strown onto the table surface. The objects that were included in the 3D scene were fruit, leaves, stems, a wicker basket, and a window. These objects were selected for the 3D scene because they can be converted into basic shapes that are easily rendered using OpenGL. For example, the apples were represented by spheres, whereas the stems and leaves were represented by cylinders and prisms.

The conversion of the objects into 3D shapes was the first step in the developmental process for the 3D scene. This step was followed by the addition of colors and textures to the 3D shapes. Finally, lighting was incorporated to fully capture the essence of each object depicted in the still life picture.

2.) The user can navigate the 3D scene using several camera controls that were created using the ViewManager class in the final project. The keyboard allows the user to move the camera up, down, left, right, as well as forward and back. The corresponding keys for these camera controls are Q, E, W, A, S, and D, and these keys are enabled using the ProcessKeyboardEvents function within the ViewManager class. Additionally, the mouse curser can be used to pan the camera left, right, up, and down. The mouse can also be used to control the movement speed of the camera using the scroll wheel. These features were implemented using the Scroll\_Callback and Mouse\_Position\_Callback functions.

3.) There were several custom functions in the program that were used to organize the code. These functions included Mouse\_Position\_Callback, Scroll\_Callback, and ProcessKeyboardEvents. Most camera movement features were relegated to these functions in order to organize the code using a modular approach. Maintaining modular code allows the programmer to reuse the same code within a single project or in future projects. For example, the camera functions used in this project can be used in future applications that require a customizable camera view system, such as a video game with an immersive world or a airplane simulation program.