# 

# CS 330 7-1 Final Project: Project Reflection

Justin M. Dougherty

justin.dougherty@snhu.edu

Southern New Hampshire University

Design Decisions for 3D Scene Development: Reflection

Rationale Behind Object Selection

In developing the 3D scene, I chose objects that would demonstrate a broad range of shapes, textures, and complexities, thereby showcasing the capabilities of the OpenGL rendering system. For instance, the transition from a simple white box to a Rubik's Cube adds a layer of complexity and visual interest. The cube, with its multiple colors and distinct segments, offers a more challenging texture mapping and geometry construction scenario. This change aims to highlight the engine's capability to handle intricate patterns and the programming logic required to replicate such a well-known object accurately.

Similarly, replacing the hair cream tube with a candle alters both the object's texture and shape. A candle, typically cylindrical with a simpler texture but distinct features like a wick and possibly melting wax, allows for a different demonstration of OpenGL's texture handling and lighting effects. It provides an opportunity to showcase how lighting and shadow can be used effectively to create a realistic rendering of a candle's smooth, slightly translucent surface and its interaction with light.

The scene is constructed using modular functions within the ShapeBuilder class, such as UBuildCylinder for the candle and custom methods for the Rubik's Cube. This modular approach not only streamlines the development process but also ensures that the code is reusable and extendable for future objects. The UTranslator function further centralizes transformations, applying consistent movements, rotations, and scaling across various objects, demonstrating effective code organization and efficiency.

User interaction within the scene is facilitated through intuitive camera controls, implemented via keyboard and mouse inputs. This setup simulates a natural walking and looking around experience within the virtual environment. The keyboard controls allow users to navigate through the scene, while the mouse input adjusts the camera angle, offering a comprehensive exploration experience of the 3D space.

The design and implementation heavily emphasize modularity and reusability. Each shape-building function in the ShapeBuilder class is designed to be self-contained and reusable for creating multiple instances of similar objects with varying dimensions or textures. The UTranslator function exemplifies this by providing a single, streamlined process for applying transformations to any object within the scene. Furthermore, the UBuildScene function in SceneBuilder assembles the scene methodically, calling upon ShapeBuilder functions, thus reinforcing the modular design approach.

By carefully selecting objects and designing the programming structure, this 3D scene serves as a testament to the capabilities of OpenGL in rendering realistic objects and the effectiveness of structured, modular programming in creating complex virtual environments.