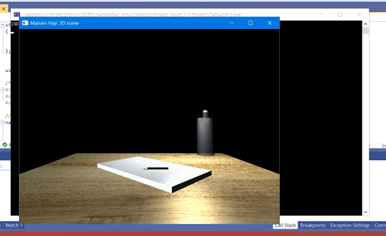
CS330-Final Project:

By Mariam Haji

For my openGL project, I chose to construct a 3D representation of a table, bottle, pen, and paper. I selected this scene for its straightforward nature and widespread familiarity. My goal was to strike a balance between realism and visual appeal. Please refer to the accompanying image for a detailed view.



The table, bottle, pen, and paper were chosen as they are commonplace items. I aspired to challenge myself in creating a setting that was both true-to-life and visually stunning. In my modeling, I favored low-polygon 3D designs for their elegance and efficiency. The basic geometries I utilized included cubes for the table, cylinders for the bottle and pen, and planes for the paper, ensuring compliance with object generation guidelines. The textures I selected resonate with the authentic feel of these items and are aligned closely with the reference image. To ensure consistent lighting and to prevent obscure shadows during viewpoint changes, I incorporated dual light sources, providing a flawless presentation of the models.

Users can navigate the 3D environment using the WASD keys for forward, backward, and side-to-side movements. Vertical adjustments can be made using the QE keys, offering a comprehensive navigation experience. The camera's viewpoint adjusts with the mouse movement, allowing users to view the scene from any desired angle. An added feature also allows users to toggle between orthographic (2D) and perspective (3D) views with a specific key, enhancing the overall visual experience.

To maintain a streamlined code, I employed several functions. Here are some of the key functions integrated into my code:

**UResizeWindow:** This function is triggered upon window resizing, adjusting the OpenGL viewport to align with the new window size. It is essential for ensuring proper content scaling in any OpenGL application.

**MoveCursor:** Monitors cursor movement since the last frame, useful for detecting mouse movement events in various applications.

**UscrollCallBack:** Handles mouse scroll events, updating variables that monitor the direction of the scroll wheel movement. This function can be adopted in applications that require scroll wheel input, such as zoom functionality in a 3D environment.

**Uinput:** Processes inputs from both mouse scroll and cursor movement, influencing properties like viewer movement speed. Its foundational structure can be tailored for applications that require multi-source input processing.

These functions have been instrumental in enhancing the code's clarity, ease of maintenance, and reusability.