Project 5 – Render Buffers

Name:	Mukunth Balaramachandran Srinivasan
uNID:	u1467270

What's implemented?

All requirements implemented. Rendered teapot.obj to a texture (a plane with a constant color) and implemented rotate, and zoom in/out for both the plane and the object. As instructed, ALT + left click is set to rotate the plane and ALT + right click is set to zoom in/out.

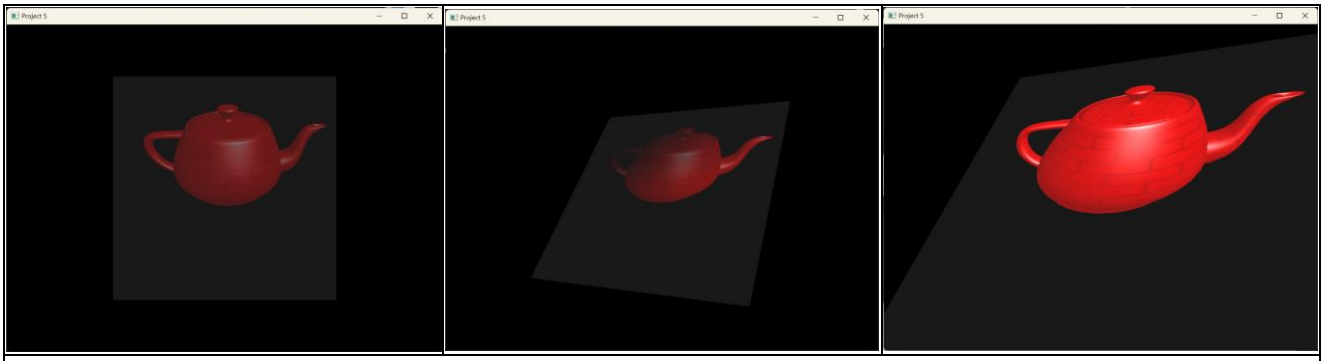

Fig 1. Different plane orientations (ALT + mouse click)

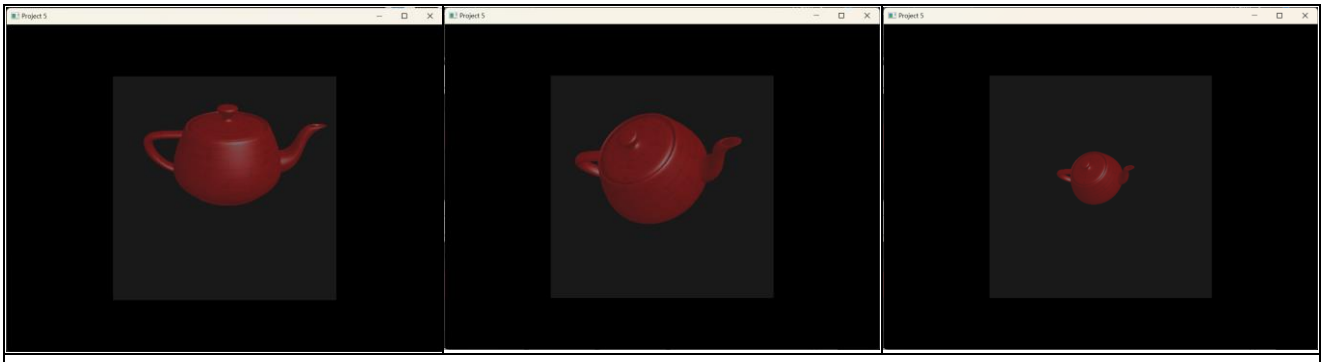

Fig 2. Different object orientations (mouse click)

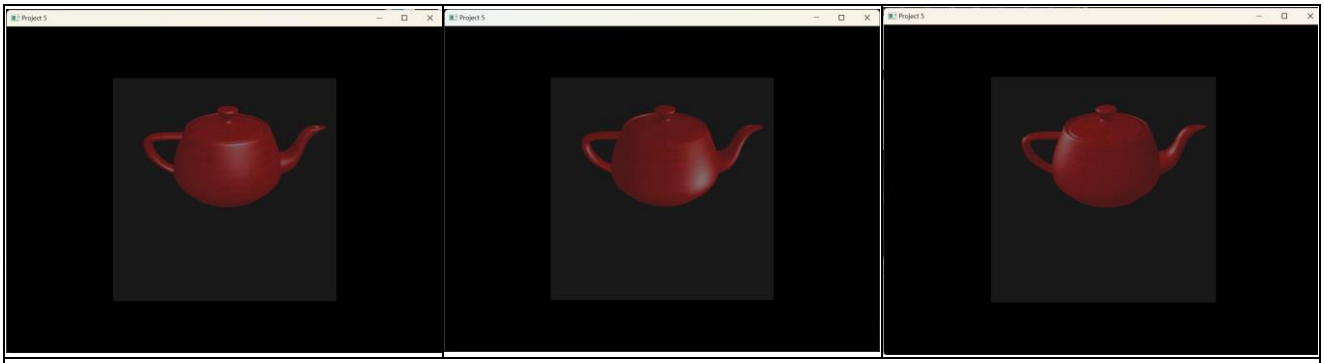

Fig 3. Different light orientations (CTRL + click)

What could not be implemented?

-

Additional functionalities

Previous projects' functionalities:

- Left mouse button to rotate and right mouse button to zoom in/out (click and drag).
- Centering the object on the window based on its boundary values.
- Re-compiling shaders on pressing F6 key.
- Ctrl + left mouse button to rotate the light source.

How to use implementation?

```
g++ main.cpp lodepng.cpp -o main -lfreeglut -lglu32 -lopengl32 -lglew32
```

This command will generate the output file "main" ("main.exe" in Windows) in the working directory. This command includes the GLEW 32-bit linker. I didn't use an IDE and had all the libraries and headers globally installed, so I didn't have to use `-I` and `-L` tags to specify paths to headers and DLLs. Place "lodepng.cpp" file in the same directory as the "main.cpp" file.

All texture maps, .obj files and .mtl files are expected to be in the same directory as the executable file.

The folder structure for the headers in `include` is as follows:

```
-> include
      -> GL / all FreeGLUT and GLEW headers
      -> cyCodeBase / all cyCodeBase headers
      -> lodepng.h
```

OS and Compiler

Operating System	Windows 11 (x64)
Compiler	g++

External libraries and additional requirements

Apart from FreeGLUT, GLEW, cyCodeBase and LodePNG have been used for this implementation.
