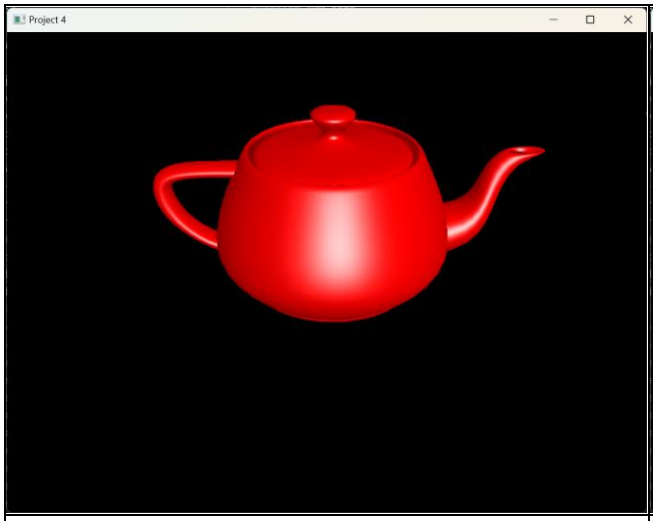
Project 4 - Textures

Name:	Mukunth Balaramachandran Srinivasan
uNID:	u1467270

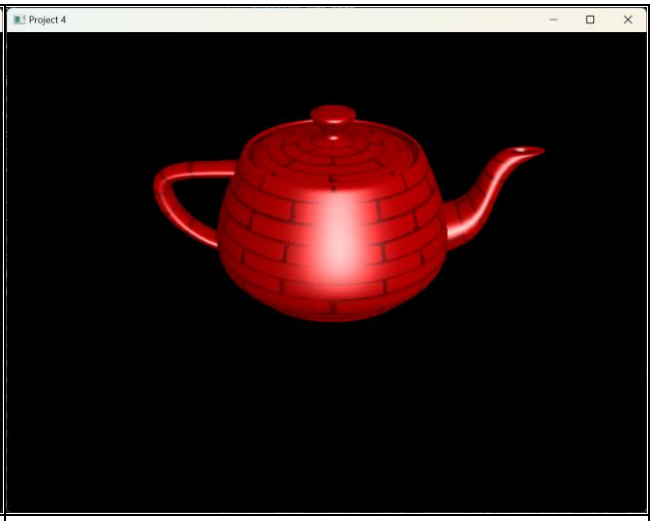
What's implemented?

All requirements implemented. Textures have been added to introduce color variations to the object. Ambient, diffuse and specular textures have been successfully added and displayed.

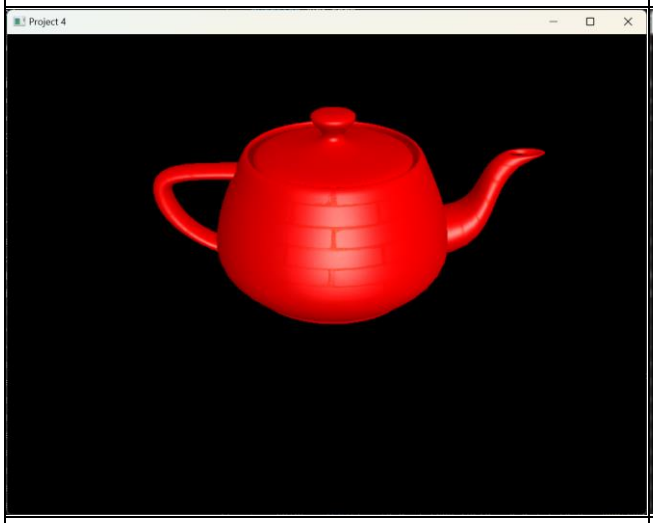
Optional - Support for objects with multiple materials has also been implemented.



No Texture



Ambient + Diffuse



Specular

Ambient + Diffuse + Specular

Fig 1. Different texture components

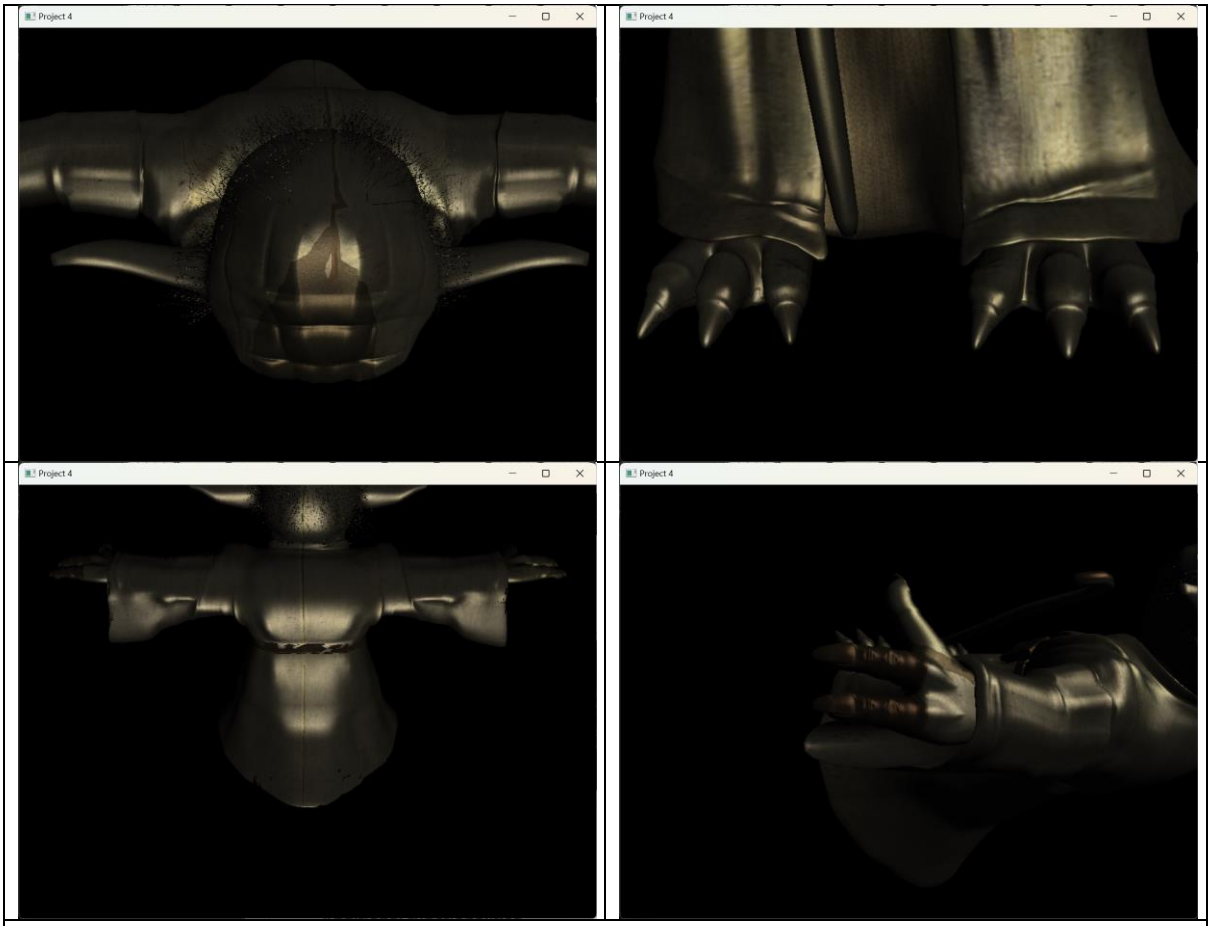


Fig 2. Different textures on “yoda.obj”

What could not be implemented?

-

Additional functionalities

Window resizing:

I’ve also implemented a resize function and mapped it to the `glutReshapeFunc()` callback. Whenever the window is resized, the viewport size is changed, and the object’s size is preserved by adjusting the field-of-view (FOV) and the aspect ratio.

View big objects:

To view big objects such as “yoda.obj”, I have mapped the B and S keys to change the zoom speed to view big and small objects respectively.

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
```

Note: While testing with “yoda.obj” file, I found the object to be much larger than “teapot.obj”. Therefore, I had to change the zoom in/out speed to comfortably view the object. Use B key to increase the zoom speed and S key to decrease it.

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.
