# Attempt by Hand:

I attempted to complete #1 without using all the helper libraries, but ran into some problems getting the texture to actually show up. I have placed this code in a zip file in this directory.

# Description of Changes:

The code used here is pretty much the same as in p2. The shader program is simplified in this version, commenting out the 2nd and 3rd textures.

CMakeLists.txt

Changed reference location to Texture.cpp and Texture.h

Not sure whether this will work correctly on your end or not. This was an attempt to get the build to take my modified versions of the files, though I don’t seem to have the CMakeLists working correctly, though the underlying code does work correctly.

Texture.cpp

Start of Program

Added

string GLMultiTexture::\_glsl\_names[4] = { "texture\_sunset", "texture\_face", "texture\_colorwheel", "texture\_blend"};

// Handles 3rd texture (texture\_colorwheel)

GLMultiTexture::GLMultiTexture()

Added

\_texture\_3 = 0;

\_textureIdx3 = -1;

int GLMultiTexture::loadAndCreateTextures

Added

unsigned int channels3;

unsigned int width3;

unsigned int height3;

unsigned char\* data3 = loadBitmapFile(path\_and\_file\_texture\_3, channels3, width3, height3 );

Handling for 3rd texture (labeled with #pragma region Texture 3)

Modified

if(data1 == NULL || data2 == NULL || data3 == NULL)return -1;

// Accepts 3rd image

bool GLMultiTexture::addVariablesToProgram

Added

\_textureIdx3 = glGetUniformLocation(program, \_glsl\_names[2].c\_str());

checkUniform(\_textureIdx3, \_glsl\_names[2]);

// glActiveTexture tells OpenGL which texture unit we want to use. GL\_TEXTURE0 is the first texture unit, so we will just use that.

glActiveTexture(GL\_TEXTURE2);

//We use glBindTexture bind our texture into the active texture unit.

glBindTexture(GL\_TEXTURE\_2D, \_texture\_3);

// Then we set the tex uniform of the shaders to the index of the texture unit. We used texture unit zero, so we set the tex uniform to the integer value 0.

glUniform1i(\_textureIdx3, 2);

// Handles 3rd image texturing

Modified

\_textureBlendModelIdx = glGetUniformLocation(program, \_glsl\_names[3].c\_str() );

checkUniform(\_textureBlendModelIdx, \_glsl\_names[3]);

Index changed from 2 to 3

Texture.h

class GLMultiTexture : public GLTextureBase

Added

const string \_glsl\_names[4] = { "texture\_sunset", "texture\_face", "texture\_colorwheel", "texture\_blend" };

// Added parameter for texture\_colorwheel

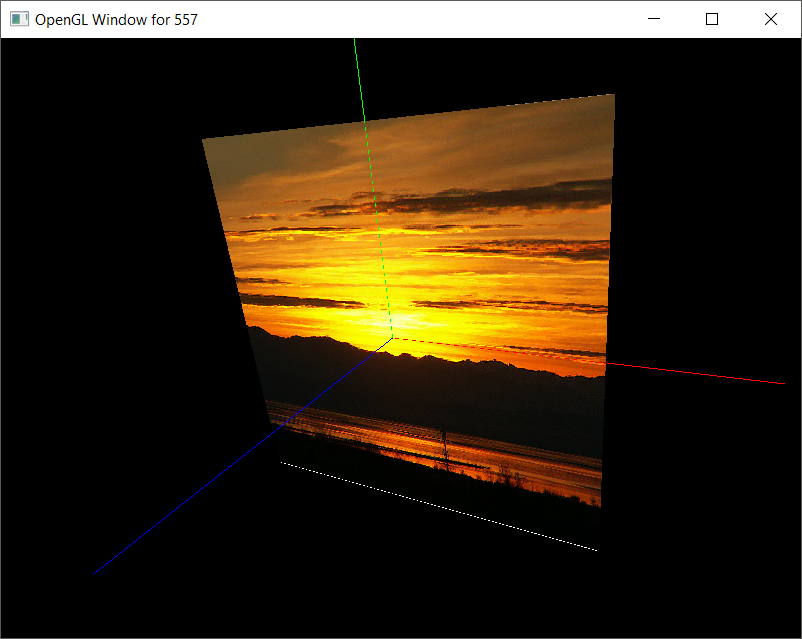
private

Added

GLuint \_texture\_3

int \_textureIdx3

int \_textureBlendModelIdx2 // Not used



color = tex\_sunset;