Assignment 2: OpenGL Lighting

# Changes in code

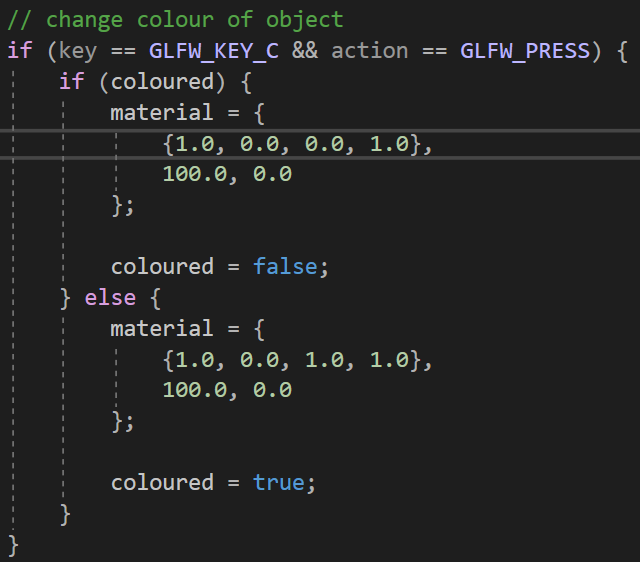
My solution uses Example6a as a base with example6c.vs and example6d.fs as the vertex and fragment shaders, respectively. The main change in the code is the addition of the uniform block material, which contains the material properties of the object. This uniform block is added to the fragment shader and main.cpp as follows:

|  |  |
| --- | --- |
| main.cpp | example6d.fs |

Next, I initialize a uniform buffer called materialBuffer. To use this buffer, it needs to be loaded in the same way as the lightBuffer. This is done in the init and display functions:

|  |
| --- |
| init() |
| display() |

Initially, the colour of the object is red. Toggle between two colours when C is pressed. Create a Boolean value, coloured, which signifies if the object’s color has changed. If false then change the value of Mcolour in the material structure, and revert Mcolour back to its original value if true. This is all done in the key\_callback function.



Initially, the object has a spotlight. Toggle between spotlight and point light when L is pressed. Create a Boolean value, spotlight, which signifies if we are using a spotlight. If false, then change the spotlight into a point light, and revert back to the spotlight if true. So, the values of the light structure are being changed. In order to change the lighting, we make changes in both the key\_callback function and in the fragment shader.

|  |  |
| --- | --- |
| key\_callback() | example6d.fs |

# Screenshots

