INF2110 - Introduction à l'Informatique Graphique **M1 Informatique**
TP8 **Fall 2018**
Prof. Caroline Larboulette

Lighting and Shading

What you need to turn in at the end of the lab session
At the end of the lab session, you should upload a .zip file containing the source code .cpp (+ eventually other classes you have used) of each exercice as well as a .pdf shortly describing what you have done for each question with screenshots to illustrate the results you have obtained. Remember the assignment may be graded.

Program using the *libqglviewer* library. Start from TP6 (you need to be able to load a polygonal model). Read the entire practice before you start: a light without a material will not do much !

1. Light Sources

Specify one light source of each openGL type and switch them on/off to see the effect on your objects (see the lecture and OpenGL documentation for the necessary parameters).

`void` **`glLight`**`{if}(GLenum` *`light`*`, GLenum` *`pname`*`, TYPE` *`param`*`);`
`void` **`glLight`**`{if}`**`v`**`(GLenum` *`light`*`, GLenum` *`pname`*`, TYPE *`*`param`*`);`

Creates the light specified by *`light`*, which can be `GL_LIGHT0`, `GL_LIGHT1`, ..., `GL_LIGHT7`. The characteristic of the light being set is defined by *`pname`* (`GL_AMBIENT`, `GL_DIFFUSE`, `GL_SPECULAR`, `GL_POSITION`, `GL_SPOT_DIRECTION`, `GL_SPOT_EXPONENT`, `GL_SPOT_CUTOFF`, `GL_CONSTANT_ATTENUATION`, `GL_LINEAR_ATTENUATION`, `GL_QUADRATIC_ATTENUATION`), and the desired value by *`param`*.

Before you draw, you need to enable openGL lighting and each light: `glEnable(GL_LIGHTING); glEnable(GL_LIGHT0); glEnable(GL_DEPTH_TEST);`

Use `void` **`drawLight`**`(GLenum light, qreal scale = 1.0) const;` from the *libqglviewer* to draw a representation of the light (see documentation for details).

2. Material

Change the parameters of your object material and observe the effects.
To be able to see the lighting in effect, you also need to define a behavior for your material. You can load it from a *`.mtl`* file or specify it once for the entire object.

`void` **`glMaterial`**`{if}{GLenum` *`face`*`, GLenum` *`pname`*`, TYPE` *`param`*`);`
`void` **`glMaterial`**`{if}`**`v`**`(GLenum` *`face`*`, GLenum` *`pname`*`, TYPE *`*`param`*`);`

Specifies a current material property for use in lighting calculations. *`face`* can be `GL_FRONT`, `GL_BACK` or `GL_FRONT_AND_BACK` to indicate which face of the object the material should be applied to. The particular material property being set is identified by *`pname`* (`GL_AMBIENT`, `GL_DIFFUSE`, `GL_AMBIENT_AND_DIFFUSE`, `GL_SPECULAR`, `GL_SHININESS`, `GL_EMISSION`, `GL_COLOR_INDEXES`), and the desired value by *`param`*.

3. Lighting Model

In openGL, the lighting model has 3 components: global ambient light intensity, whether the viewpoint location is local to the scene or at an infinite distance away, and whether lighting calculations should be performed differently for front and back faces.

`void` **`glLightModel`**`{if}(GLenum` *`pname`*`, TYPE` *`param`*`);`
`void` **`glLightModel`**`{if}`**`v`**`(GLenum` *`pname`*`, TYPE *`*`param`*`);`

Parameters can be `GL_LIGHT_MODEL_AMBIENT`, `GL_LIGHT_MODEL_LOCAL_VIEWER`, `GL_LIGHT_MODEL_TWO_SIDE`.