



Faculty of Science

<b>Course:</b>	CSCI 4110U: Advanced Computer Graphics
<b>Lab Assignment:</b>	7
<b>Topic:</b>	Lighting and shading

## Overview

In this lab, you will build upon last week's lab. You are going to add a light source, and diffuse shading, to the scene.

*Note: If you were not able to get Lab 06 working, simply create a project (using the Draw Sphere example as a base) that draws two motionless spheres.*

## Instructions

Lab 06 should have two celestial bodies animated (if not, at least rendered) with just a solid colour for each. In this lab, we will add a light source, representing the sun, in two different ways:

1. A point source light. You can use the actual distance from the sun to the Earth to determine the light position.
2. A directional light. This does not need a position, but does need a direction vector.

Write your program such that toggling a Boolean flag (e.g. `usePointLight`) in the code will switch between these two light sources.

In order to see these light sources in action, we'll need to add their information, and basic Lambertian (diffuse) shading to the shaders for our spheres.

Use the formula for Lambertian shading, provided in the lecture notes. The precise diffuse colours for each of the objects are up to you, but I suggest making the Earth a shade of deep blue, and the moon a light gray.

One way to implement this would be to have two sets of shaders, one that calculates the shading for a point light source, by calculating a vector from the light source to the fragment, and the other that uses the directional light source's vector for the same purpose. Most of the code of these shaders will be identical. In your draw code, you can pass both values (as uniforms), and the shader will ignore any uniform values passed that are not declared in the shader. This will mean that your draw code won't change, just the code that activates the shader program.

## Lab Report

To demonstrate to the lab instructor your completion of this laboratory assignment, merely show them the modified OpenGL program.