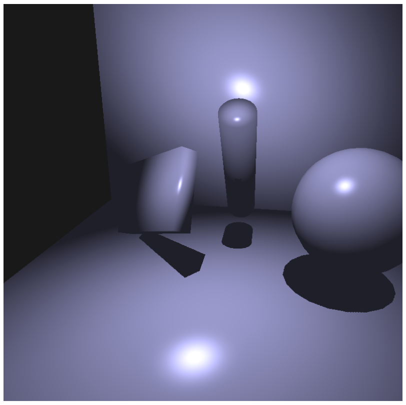
Programming Assignment 2

Date of announcement: 14^{th} May 2018
Submission deadline: 28^{th} May 2018

Description

In this programming assignment you will learn about shadow mapping. In particular, you will learn how to render a depth map to texture, and use that to compute realistic shadows like the one shown below:



Implementation Specifications - Grading Criteria

Develop an OpenGL application with the following functionalities and features:

- the minimum version of OpenGL should be 3.1 and up
- OpenGL is used in retained mode
- include comments explaining each step
- load an OBJ file containing the scene. You can use the OBJParser project provided. Alternatively you can load a heightmap image and convert that into a mesh
- create a GLFW window of size 800x800 with double buffering support
- render the scene on display
- position a point(spot) light on top of the scene looking at the center of origin.
- the application should use a perspective view to display the scene and use the depth buffer for hidden surface removal
- all vertices must be colored. You can select whatever color you like
- handle the following input:
  - the user can rotate the camera using keyboard input i.e. left arrow → R_z, right arrow → R_{-z}, up arrow → R_x, down arrow → R_{-x}
  - the user can move the camera using the mouse i.e. moving forward/backward while left button is pressed → move into/out of the scene
  - the user can move the light using keyboard input i.e. key 'a' → moves left, key 'd' → moves right, key 's' → moves up, key 'd' → moves down
  - the user should be able to change the bias i.e. key '+' → increases bias, key '-' → decreases bias with a minimum value of 0

Submission (electronic submission through Moodle only)

Please create a zip file containing your C/C++ code, vertex shader, fragment shader, a readme text file (.txt). In the readme file document the features and functionality of the application, and anything else you want the grader to know i.e. control keys, keyboard/mouse shortcuts, etc.

Additional Information

- You can use the skeleton code provided during the lab sessions to get started.
- A video demonstrating the functionality is posted on YouTube: `https://youtu.be/y7tHG3OZTrY`

Evaluation Procedure

You MUST demonstrate your solution program to the lab instructor during lab hours. You will be asked to download and run your submitted code, demonstrate its full functionality and answer questions about the OpenGL programming aspects of your solution. Major marking is done on the spot during demonstration. Your code will be further checked for structure, non-plagiarism, etc. However, ONLY demonstrated submissions will receive marks. Other submissions will not be marked.