COSC4370 HW2 - OpenGL

Due: March 9 at 23:59, 2021

1   Introduction

In this assignment, we will practice some of the basics of OpenGL.

2   Setup

We will be using the GLUT/freeglut library for our OpenGL code. If you are on Windows, you can find a freeglut 3.0.0 package for Visual Studio (MSVC) at http://www.transmissionzero.co.uk/software/freeglut-devel/. On OS X, GLUT should be installed by default. Finally, if you are on Linux and don't have GLUT already installed, you should be able to install it by running sudo apt-get install freeglut3-dev or sudo yum install freeglut-devel in a terminal. We have written the included Makefile so that the homework should compile with it on Linux or OS X. For Windows, you will need to create a project in Visual Studio rather than using the Makefile.

Please make sure you try compiling the starter code ASAP so that there are no last-minute problems with getting your environment setup correctly.

3   The Main Assignment

For this homework, you will be using your newfound OpenGL skills from class, as well as your artistic creativity, to create several 3D scenes with OpenGL.
The first part of the assignment is to write code to reproduce each of the following three images:
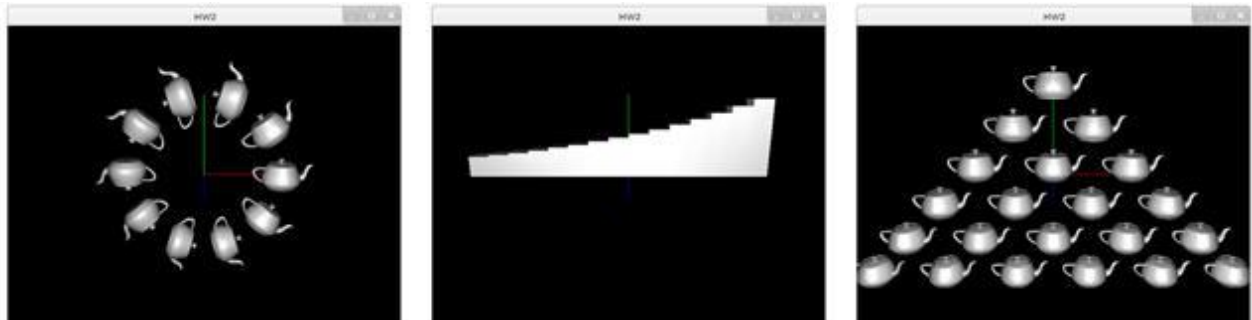
The second part of the assignment is to create, using similar techniques, a scene of your own imagination. All the code for this homework lives in main.cpp; you need to fill in the functions problem1, problem2, problem3, and problem4. You can switch between the different examples while the program is running by pressing the 1,2, . . . keys. Hence you don't need to recompile in order to run different examples. Additionally, you can quit the program at any time by pressing 'q', 'Q', or the Escape key.

1

4   Tips and Requirements

For each of the three reproductions, you should be able to create the image using only glutSolidTeapot, glutSolidCube, and OpenGL's transformation mechanisms like glPushMatrix, glPopMatrix, glTranslatef, etc. Note that you should not need any custom geometry, just the teapot and cube, to reproduce the images.

Your reproductions do not need to match exactly. However, please try to make them match the examples as closely as possible. We used nice numbers in the reference solutions, so if you find yourself using strange fractions etc. to reproduce the examples, you may be trying too hard!

For the open-ended image/scene, we require the following to make sure your image is interesting:

•Make use of OpenGL's transformation mechanisms in a nontrivial way, with at least one instance of nested applications of glPushMatrix (i.e., a glPushMatrix within another glPushMatrix).

•Render at least one triangle by feeding in its coordinates directly (OpenGL immediate mode is okay here, even though it's deprecated)

As an example, you could attempt to create a very rough approximation of an articulated hand:

5   Deliverables

Submit all deliverables to your Github repository.

    a.   Code for generating each of your four images (main.cpp)

    b.   Screenshots (preferably .png) for each of your four images

    c.   You need to write a detailed report(50 percent points of the assignment, pdf format), you should state

       the assignment problem, explain the algorithm or method you use, explain details of implementation,

       discuss your results and etc.

Upload your source code contained in a single file: hw2.cpp and other necessary files

a. In your Github readme file, put your name and student ID, and also coding environment, compiling method (commands).
b. You can only use the libraries we provided.
c. You will lose points if violate any requirement above