Project 1
CSCI-4962: Three-Dimensional Computer Graphics

Due: Friday, September 20, 2002, 11:59:59pm

1 Overview

This project is intended to familiarize you with OpenGL and its 3D modeling, viewing, and projection transformations. You are to write a program to display an image of the "Utah teapot" on a table created from cubes. You will use both OpenGL and GLUT functions such as `glTranslatef()` and `glutWireTeapot()`, and use mouse and keyboard input events. You will also need functions such as `gluLookAt()`, `gluPerspective()`, and `glOrtho()` to set up the viewing and projection transformations.

Please see the course web page at `www.cs.rpi.edu/~sakella/graphics/` for an example piece of OpenGL code, and for project updates and additional information.

2 Project Tasks

You are to write an OpenGL program to draw a teapot on a table, where the table has four legs and is supported on a planar floor surface. The table consists of five cuboids, with one serving as the tabletop and the other four serving as table legs. You can use `glutWireTeapot()` to draw the teapot, and `glutWireCube()` to draw a cube.

1. Draw the teapot with its center at the origin (0, 0, 0). Note that the height of the teapot is 1.6 units when you call `glutWireTeapot(1.0)`. Select the initial camera location to be at (0, 0, 5). Let the default projection mode be perspective projection.

2. Create a table with a cuboidal tabletop (4 units long along the X direction, 2 units long along the Z direction, and 0.2 units long along the Y direction) and four cuboids (0.2 units long along the X and Z directions and 1 unit long along the Y direction) serving as its four legs. Each leg should be flush with one of the corners of the tabletop.

3. Draw a rectangle with dimensions 6 units (in X) and 4 units (in Z) to serve as the supporting planar floor.

   The centers of the teapot, the table, and the floor are on the Y axis. Use the `glColor3f()` function to display the teapot, table, and the floor plane in different colors.

4. To enable user interaction using both the mouse and keyboard, write functions to rotate the viewpoint around the objects, to translate away from and towards the objects, and to toggle between perspective and orthographic projection modes.

   The main keyboard events and the corresponding desired behavior are:

   - `x/X` keys: The objects are rotated 10 degrees counterclockwise/clockwise about the X axis.
   - `y/Y` keys: The objects are rotated 10 degrees counterclockwise/clockwise about the Y axis with each key press.
   - `z/Z` keys: The viewpoint moves 0.1 unit closer/further to the objects.
   - `i` key: The camera and objects are returned to the initial configuration.
   - `q` key: The program quits.

   The main mouse event is:

   - A simultaneous click of both the left and right mouse buttons inside the display window should toggle the projection transformation mode between perspective projection and orthographic projection.

5. Finally, add the ability to rotate the teapot at a constant angular velocity. The teapot should rotate by 1 degree about its vertical axis at each frame. The rotation should be turned on by pressing the `r` key and should be stopped by pressing the `R` key.

3 Grading

Your project will be graded for a total of 100 points as follows:

- 45 points for displaying the image of the teapot, table, and floor.
- 25 points for keyboard event functions (`x/X`, `y/Y`, `z/Z`, `i`, and `q` keys).
- 10 points for mouse function to toggle projection modes.
- 10 points for animating rotation of the teapot.
- 10 points for code structure, clarity, and documentation.

Extra credit: You can earn up to 10 additional points for special features and creative enhancements to the project requirements. Describe your enhancements in your README file, and ensure that they do not interfere with the required project functionality.

Lateness policy: Please read the lateness policy in the course syllabus. Use your late days carefully, if at all.

4 Submission

The code must be submitted no later than **11:59:59 pm on September 20, 2002**. Specific instructions for handing in your code will be provided on the course web page. **You are responsible for ensuring that your code can compile and run on the PCs in VCC North or South, or the Sun Ultras in Amos Eaton 217.** Your source code must be readable and commented. The comments need not be extremely long, just explain clearly the purpose of each block of code.

You must hand in a README file, your source code (source and header files), and if necessary, a Makefile to compile it. Your submission should NOT include any object files or executable files. Every file should have your name in a comment line at the top. The README file should contain the following information: your name, instructions on how to compile the code and run it, known bugs or limitations, any extra credit enhancements, and any other relevant information.

Proper submission is entirely **your responsibility**. Contact the TA if you have any doubts whatsoever about your submission. Do **NOT** submit your project via email. Be sure to keep copies of your files and **do not change them after submitting**. After grades are posted, you have exactly one week to resolve all problems. All grades are final two weeks after they are posted.

A project that does not follow the submission guidelines will receive a **10 point deduction**.