

CMSI 371-01

COMPUTER GRAPHICS

Spring 2012

Assignment 0410

This assignment seeks to wrap up the geometry so that we can move on to the fragment shader.

Outcomes

This assignment will affect your proficiency measures for outcomes *2b–2e*, *3c–3e*, and *4a–4f*.

Not for Submission

At this point, with the exceptions of Sections 2.11, 2.12, 3.14, and 4.10, we have pretty much covered Angel Chapters 1–4. And of course, any C or C++ code in those chapters is superseded by or adapted into JavaScript and WebGL.

Catching up on this reading by April 10 will be helpful to you in accomplishing this assignment's tasks for submission.

For Submission

For the following tasks, keep building on *homework/pipeline* on your git repository. Do rename files, however, to better reflect what you have going now (yes, that means no *hello-webgl.htmls* etc. anymore—you're way past "hello" now!).

Shape Up

Add one more shape to your shape library. Your choice: possibilities include a tetrahedron, cylinder (approximated), cone (approximated), etc.

Make a Scene

Use your library of shapes to construct a 3D scene that interests you. Use transforms liberally to position, rotate, and scale objects. Use projection and camera/view matrices to get full flexibility in terms of framing and displaying your scene.

You are "The Architect"

In accomplishing the above, you will need to fill out your matrix library (see what I did there?) with the remaining useful transforms. Building the camera matrix will also require vector functions. Implement what you need.

Take the Red Pill

Set up JavaScript event handlers (if these are not familiar to you, talk to me and/or read Chapter 6 of the JavaScript text) to enable interactive navigation or viewing through your scene. There are many ways to do this; Angel Sections 3.13 (page 180) and 4.9.3 (page 247) offer some ideas, but you don't have to restrict yourself to those. In the GitHub bazaar, the *sierpinski-webgl* sample from CMSI 370 illustrates how you might do interactive mouse rotation.

Commit and push your work to your git repository under *homework/pipeline*.