

CMSI 371-01

COMPUTER GRAPHICS

Spring 2015

Assignment 0430b

As advertised, this assignment (your last!) focuses on interaction and dynamic behavior. Specifics can't really be stipulated here because this will be very dependent on your scene. But just have *something* that isn't trivial. Definitely do more than what the sample code has!

Outcomes

This assignment will affect your proficiency measures for outcomes *1c*, *2a*, *2d*, *3a*, *3b*, *3d*, and *4a–4f*.

Not for Submission

The previously mentioned readings from the Angel textbook, if you have that, remain relevant to this assignment (and may actually be more meaningful now that you have a bigger picture of things). A closer focus on Chapter 3, in particular, may be helpful here, depending on the specific interaction that you are planning to do.

For Submission

Take the Red Pill

Build on a variety of sources to add animated, interactive, or dynamic elements to your scene. For animation, do *at least one* of the following:

- Adapt the keyframe tweener library from Assignment 0217 so that it can be used with your scene. *Hints:* The easing functions don't change; you will want a `keyframes` data structure that references your `Shape` objects; and finally the biggest changes you'll make code-wise will be in your periodically-called scene-updating function.
- Implement physics-based behavior like gravity, acceleration, or anything else. You may integrate a physics engine or roll your own simple one. *Hints:* The sample code already has a full-blown vector library—use it! Some of this work harkens back to assignments from CMSI 186, only now in 3D and displayed visually.

For interaction, do this:

- Set up JavaScript event handlers to enable interactive navigation, viewing, or any other user-driven manipulation of your scene. If this is not familiar to you, talk to me and/or read Chapter 6 of the JavaScript text and Chapter 3 in the Angel

textbook. 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-webkit* sample from CMSI 370 illustrates how you might do interactive mouse rotation.

Remember that these approaches are largely orthogonal and therefore can be mixed and matched. Note also that although some changes to your model and view code may be necessary, they should be quite distinct and separate from the actual dynamic behavior. In fact, these changes may have no effect at all on the view code, with the model affected mainly through additional properties or data.

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

Show Up

The remaining material for the course can be found in Angel Chapter 8 Sections 8.3–8.7 (clipping) and Section 8.11 (hidden surface removal). For another proficiency on the outcome pertaining to these topics (*2d*), please attend class on the days that these topics will be addressed; we will converge on the specific dates as the end of the semester approaches.