CMSI 371-01

COMPUTER GRAPHICS

Spring 2016

Assignment 0329b Feedback

All caps are released with the outcomes in this assignment because a sufficient amount of functionality will have been reached here.

Dustin Kane *ranneyd / dustinpkane@gmail.com*

*Notes while running (high-priority notes are marked with ***):*

- Nice, robust matrix test suite there!
- The full matrix functionality is only very subtly used in your scene, but presumably that will get used in more overt ways later.

Code review (refer to http://lmucs.github.io/hacking-guidelines/ for code-review abbreviations):

1. Damn those *qunit-close-enough.js* tabs! (*4c*)
2. Matrix code all looks good, in terms of both design and functionality. +(*2a*, *3a*, *4a*, *4b*)
3. I'm generally not picking out much code presentation issues to save time, but I really must harp on *http://lmucs.github.io/hacking-guidelines/curly/#curly-mult* because that may be the most divergent presentation choice (in terms of how it corresponds to the code's meaning) being made here. (*4c*)
4. Instance transformation and propagation to children looks alive and well here, and nicely is already being used by your shapes to "complete" themselves. +(*2a*, *3a*, *4a*)
5. What we *do* need to see is a new viewing volume. Time to break out of that cube! (*2b*)

2a — **+**

2b — **|** …The matrix code is there—just need to use it!

3a — **+**

3d — **|** …Just add `projectionMatrix`.

4a — **+**

4b — **+**

4c — **|** …More disconnected `else`'s than a **+** could tolerate… **:-**

4d — **+**

4e — **+**

4f — **+** …Same notes for *4e* and *4f* as in HW 0329a.