CMSI 371-01
COMPUTER GRAPHICS
Spring 2016

Assignment 0329a Feedback

This is the last assignment where *3a* tops out at **|** as we head toward the full expected functionality of your library code. Meanwhile, *3d* also stays at **|** until we get sufficient functionality in our shaders.

Matt Flickner *mwflickner / mflickne@lion.lmu.edu*

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

- Scene looks good, if still somewhat random.
- And of course the shape test suite with the legendary child sadness names…

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

1. Child/grouping functionality is straightforwardly done and demonstrated directly in the scene. Not much to say except maybe the vertex-buffering and drawing routines can become `Shape` prototype functions instead, since they will pretty much be the same regardless of what the specific scene might be. (*+1c*, *+3a*, *4b*)
2. Shape library looks good, including that sphere. (*+1b*, *+3a*)

1b — **+**

1c — **+**

3a (max **|**) — **|**

3d (max **|**) — **|**

4a — **+**

4b — **|** …`Shape` object can use a second pass to consolidate additional functionality. Put yourself in the position of having to create a new, separate scene with the libraries you've written. What code gets copied over? Those are likely good candidates for additional functions to fold into your libraries. Ultimately, only the code specific to a particular scene (the "app"), like its actual objects, or how they are updated during animation and interaction, should ideally be in the "app-specific" code.

4c — **+**

4d — **+**

4e — **+** …Good frequency and descriptive messages…with character!

4f — **+** …Sufficient work done close to due date.