

# CMSI 371-01

## COMPUTER GRAPHICS

### Spring 2013

#### Assignment 0326 Feedback

For this assignment, outcomes *2a*, *2b*, *3d*, and *3e* max out at | because the requested functionality in this assignment do not yet reach the culmination of what these outcomes represent overall.

#### Quin Thames

*2a* — You’ve taken a few more concrete steps toward full 3D transform proficiency—now on to using these functions in your scene! (|)

*2b* — The mechanics of your ortho and frustum functions look good. The next test will be to use them in your scene. (|)

*3d* — Your matrix library is certainly moving in the right direction. Actual “field testing” in your 3D scene code is up next. (|)

*3e* — Your matrices represent additional progress toward 3D scene rendering, but as mentioned will not top out this outcome yet because we haven’t covered the full range of shader functionality yet. (|)

*4a* — The code that you have works well so far, and this is bolstered by having a unit test suite available to “keep it honest.” (+)

*4b* — Separation of concerns looks well taken care of in the code that you have so far. My only comment is that functions that are not object-oriented (i.e., they do not operate on `this`) do not have to be assigned to `Matrix4x4`’s prototype. Not a huge deal; more of a very JavaScript-specific fine point. (+)

*4c* — Your matrix code is decently readable, but as always has a tendency to be a little tight in places. Thankfully, you at least kept the `getRotationMatrix` spacing generally unscathed! Will we ever possibly be able to break you from this habit? (|)

*4d* — Your work shows fine resource use, including leveraging the rotation matrix code that is already in the sample programs and the projection matrices that are already in the handouts. (+)

*4e* — Your commit phasing can be finer grained than it is. You have a couple of “lump sum” commits where you did a bunch of work in one sitting then just committed when you’re done. Instead, for something like this you should really be working one function at a time. True, these functions were all relatively small and probably happened in short order for you. But it would still be good to establish this as a work habit: write the test; write the implementation; commit when the test succeeds. That gives your commit log a very logical, trackable evolutionary trajectory. (|)

*4f* — Submitted on time. (+)