CMSI 371-01
COMPUTER GRAPHICS
Spring 2013

Assignment 0404 Feedback

The "cuffs" are off outcomes *1c*, *2a*, *2b*, and *3d* with this assignment, so I have started giving **+**'s for those outcomes if the submitted work calls for it. *3a* awaits full scene interaction before it can max out. *2c* (not part of this assignment) and *3e* need proficiency in lighting and fragment shaders in order to go **+**.

Chris Whiting

1c — Instance transform functionality is partially there, but not completely. The expectation is that *individual* objects can have the full gamut of transforms; that is not the case with your cod right now. The individual capability will give you a new level of flexibility in composing and arranging objects in your scene. (**/**)

2a — You are certainly using transforms more, but without the full-fledged instance transform capability we can't max this out. (**|**)

2b — Projection has been successfully implemented. Do remember to clean up any experimental projection code once you're done here. (**+**)

3a — As of the assignment due date, you had not gone beyond the rotation already given by the sample code. You had the beginnings of another event handler, but that was it. (**/**)

3d — Your library was completed nicely within a day after the assignment due date. Your unit tests are a little thin, but otherwise I think this library is ready to be exploited. (**+**)

3e — You have successfully extended your vertex shader to use some additional transforms, but not quite full-blown instance transforms yet. (**/**)

4a — Your code appears to function correctly from the outside, but the source itself shows a bunch of bothersome glitches. The inline comments spell more things out. Something that really sticks out is the odd way by which you access the `subshapes` property. What you have is surprisingly roundabout. You've also defined, perhaps unknown to you, some global variables which really should be local. There are a few more things here and there—scan the inline comments. But that `subshapes` things alone—phew! (**/**)

4b — Your code looks properly structured and separated. (**+**)

4c — Format-wise, your code has decent readability though I had some indentation suggestions. The big knock on your readability is some badly chosen names (e.g., `checker`, `checkSub`). It's very hard to fully understand code if its variable and function names aren't sufficiently descriptive. (**/**)

4d — Your work certainly shows some information and resource use, but could use a bit more with regard to the instance transform. (**|**)

4e — Your commit frequency seems to be a tad on the not-often-enough side, given the amount of work that I saw leading up to the due date and a day or so after that. Your messages are nicely descriptive however. (**+**)

4f — Some functionality submitted on time, some a day later, some unfinished. (**|**)