**CSCI 4250/5250** **Homework 5 (Due beginning of class, Tuesday Oct 31st)**

***You are required to type your answers. Submit to the D2L Dropbox labeled “homework 5”***

1. Given the 3D cube example in programs: ortho.js and ortho.html (available on the course web page), if the view position and the orthographic viewing volume is changed into each of the following situations, how will the final 2D image change from its original image? Justify your answer.

a. mvMatrix=lookAt(vec3(-4, 0, 0), at, up); // pMatrix does not change

The eye will be positioned facing one of the back (because the X value is negated from before) panels straight on (because the other eye values are 0), so that back panel will appear as if it is a 2d rectangle.

b. mvMatrix=lookAt(vec3(3, 3, 3), at, up); // pMatrix does not change

The model will be rotated 90 degrees so that it will look basically the same, but the face that was visible on the left before will now be visible on the right.

1. mvMatrix=lookAt(vec3(3, 3, 3, at, up); pMatrix=ortho(-3, 3, -3, 3, -1, 1);

The object will not be visible, because the view volume is too narrow

d. pMatrix= ortho(-6, 6, -3, 3, 2, 10); // mvMatrix does not change

The object will appear squished inward from the sides, because the view is stretched out wider.

* 1. pMatrix=ortho(0, 4, 0, 3, 2, 10); // mvMatrix does not change

A zoomed in view of just the top right corner of the object will be visible, because the view is set to only show the top right area.

1. Given: mvMatrix=lookAt(vec3(4, 4, - 4), at, up);

pMatrix=ortho(-2, 2, -4, 4, -10, 10);

show:

* + the mvMatrix

0 4 0 0

0 16 0 -64

4 4 -4 -16



* + the pMatrix

0.5 0 0 0

0 0.25 0 0

0 0 -0.1 0

0 0 0 1

* + the coordinates of a point F(1, 1, -1) when converted into the final clip coordinates. (show intermediate steps in deriving the results)

mvMatrix \* pMatrix \* F

0 1 0 0 1

0 4 0 -64 1

2 1 0.4 -16 \* -1

0 0 0 1 1

F(1, -60, -13.4, 1)

1. Changing the orthographic viewing volume in problem 2) to a frustum with left=-2, right=2, bottom=-4, top=4 for the near plane, and the near plane at distance 4 and far plane at distance 10 from the eye/camera. How would you call the perspective function to set up the corresponding pMatrix in the .js program?

pMatrix = frustrum(-2, 2, -4, 4, 4, 10);

1. With the perspective viewing volume defined in problem 3), what will be the x and y coordinates of the two points F(1, 1, -1) and B(1, 1, 1) when projected onto the near plane?

F(4, 4)

B(-4, -4)