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
      View position changed so that only the yellow quadrant is visible
      Eye translates 8 points to the right of the image (x axis)
      Eye translates 4 points downward (y axis)
      Eye translates 4 points along (z axis)
   b. mvMatrix=lookAt(vec3(3, 3, 3), at, up); // pMatrix does not change
      View position changed to the left making red quadrant visible on left, blue quad on right, and cyan top remains same
   c. mvMatrix=lookAt(vec3(3, 3, 3, at, up);
      pMatrix=ortho(-3, 3, -3, 3, -1, 1);
      2D Image disappears from view.
      View volume is reduced to a 1 and the view position is outside of the clipping distance of the viewing volume.
   d. pMatrix= ortho(-6, 6, -3, 3, 2, 10); // mvMatrix does not change
      The image becomes more narrow
      The viewing volume translates -4 along the x axis
      The viewing volume translates 4 along the y axis
   e. pMatrix=ortho(0, 4, 0, 3, 2, 10); // mvMatrix does not change
      The top right corner is visible on the bottom left corner of the canvas.
      The viewing volume translates along x, y, and z

2) Given: mvMatrix=lookAt(vec3(4, 4, - 4), at, up);
   pMatrix=ortho(-2, 2, -4, 4, -10, 10);

show:

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

mvMatrix:

n = eye(4,4,-4) – look(1,1,-1) = (3, 3, -3) = (normalized) (0.5774, 0.5774, -0.5774)
u = (0,1,0) * n = (3, 0, 3) = (normalized) (0.7071, 0, 0.7071)
v = n * u = (9, -18, -9) = (normalized) (0.4082, -0.8165, -0.4082)
dx = -eye . u = 0
dy = -eye . v = -0.0004
dz = -eye . n = -6.9288

mvMatrix = $$\begin{pmatrix} 0.7071 & 0 & 0.7071 & 0 \\ 0.4082 & -0.8165 & -0.4082 & -0.0004 \\ 0.5774 & 0.5774 & -0.05774 & -6.9288 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

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

v = (0,0,0) – (0, 0, 0) = (0, 0, 0)

T = $$\begin{pmatrix} 1 & 0 & 0 & -\frac{-2+2}{2} \\ 0 & 1 & 0 & \frac{-2+2}{2} \\ 0 & 0 & 1 & \frac{-10+10}{2} \\ 0 & 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

S = $$\begin{pmatrix} \frac{2}{2-(-2)} & 0 & 0 & 0 \\ 0 & \frac{2}{4-(-4)} & 0 & 0 \\ 0 & 0 & \frac{2}{10-(-10)} & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 0.5 & 0 & 0 & 0 \\ 0 & 0.25 & 0 & 0 \\ 0 & 0 & 0.1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

pMatrix = S * T = $$\begin{pmatrix} \frac{1}{2} & 0 & 0 & 0 \\ 0 & \frac{1}{4} & 0 & 0 \\ 0 & 0 & \frac{1}{10} & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

P*F = $$\begin{pmatrix} \frac{1}{2} & 0 & 0 & 0 \\ 0 & \frac{1}{4} & 0 & 0 \\ 0 & 0 & \frac{1}{10} & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} * \begin{pmatrix} 1 \\ 1 \\ -1 \end{pmatrix} = \begin{pmatrix} \frac{1}{2} & 0 & 0 & 0 \\ 0 & \frac{1}{4} & 0 & 0 \\ 0 & 0 & -\frac{1}{10} & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

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

4) 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?