CSCI 4250/5250 Take Home Test 2 (100 pts) **Name _____**

Due: midnight, Thursday, Nov 21st

You are required to type your answers for these questions.

Save the file as a PDF file and submit through D2L dropbox.

Open Book & Notes. You may use computer program to assist your calculations where needed.

You are on the honor system and do this exam with NO help from any other person. When you type your name below, you are indicating that you have adhered to these restrictions. Turn in this page with your answers to the questions. Turn in your typed answers in pdf file to the D2L Dropbox named "Test 2".

I, _____, worked all of the problems on this test completely on my own without any assistance from any other person. My only resources were the textbook, online course material, my notes, my programs, and Internet sources.

1. **(8 pts) Fill in the blanks:**

   a. The _____ matrix transforms the object drawn from the world coordinate into the camera coordinate.

   b. The _____ matrix is saved and restored using Push and Pop operations in WebGL during 2D and 3D drawing involving transformations.

   c. The _____ buffer is used for hidden surface removal.

   d. In texture mapping, the s and t values of the texel coordinates (s, t) is limited to the range between _____ and _____.

   e. In perspective projection, the vertices further away from the viewer appear to have smaller x and y coordinate values, this is the result of applying the step: P _____ D_____, in the graphics pipeline.

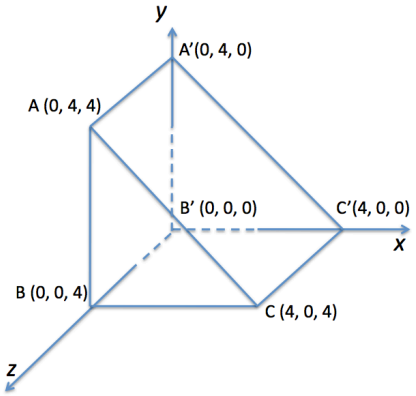
   f. When specifying the points on each face of a 3D shape, the points should be specified in _____ order.

2. (7 pts) Given the light specified as $l_d = l_a = (0.8, 0, 0)$, and $l_s=(1, 0.5, 0.5)$, the location of the light source is at (2, 1, 3, 1), and the object material property specified as: $\rho_d =(1, 0, 0.5)$, $\rho_s =(0.9, 0.9, 0.9)$, $\rho_a=(0.5, 0.5, 1)$, shininess value = 30. Applying the lighting model studied in class to compute the **specular light component** of the color of a vertex p located at (1, 0.5, 0, 1) on the object. The eye location is (0, 0, 0, 1). The normal vector at the vertex is (1, 1, 0.5, 1). Use the simplified halfway-vector approach.

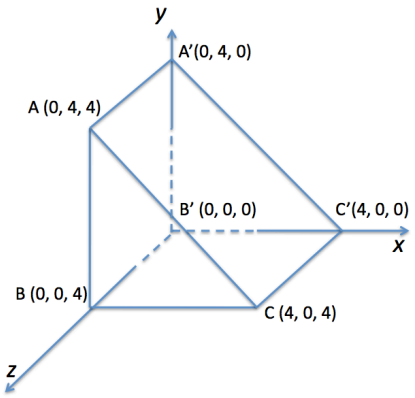
3. (30 pts) Given:
   - the vertices describing the 3D shape in the world coordinates,
   - the camera location, look at position and up direction specified in the lookAt function, and
   - the orthographic projection setup,

   answer the following questions:
   a. What is the view matrix generated at line 1 ? Show all steps in how the view matrix is derived.
   b. What is the modelviewMatrix matrix computed at line 4 ? Show the matrix t, matrix r, and the matrix modelviewMatrix after the multiplication is performed.
   c. What is the projection matrix generated at line 5?
   d. After the modelviewMatrix and projectionMatrix have been sent to the vertex shader, they are used to transform the individual vertices in each object into their final coordinates in the clip-coordinates. Compute the coordinates of vertices B in the clip coordinates.

```
modelviewMatrix = lookAt(1, 0, 5, 0, 1, 0, 0, 1, 0);      // line 1
t=translate(1, 0, 1);                                      // line 2
r=rotate(30, 0, 0, 1);                                     // line 3
modelviewMatrix = mult(mult(modelviewMatrix, r), t);   // line 4
projectionMatrix = ortho(-6, 6, -6, 6, 2, 10);             // line 5
```



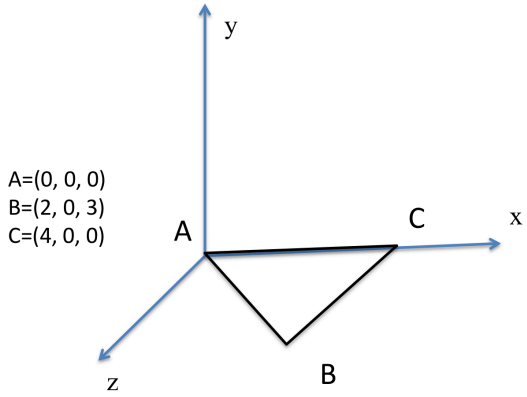
4. (10 pts) Assuming a perspective projection is used to view the object defined below:



modelviewMatrix = **lookAt(1, 0, 5, 0, 1, 0, 0, 1, 0);**
projectionMatrix = frustum(-6, 6, -8, 8, 2, 10);

Compute the x and y coordinates of the points A and A' as they are projected on the near plane.

5. (30 points) An extruded shape is formed from the base triangle shown below. The triangle extruded in the direction <0, 1, 0>, i.e., along the y-axis. The height of the extruded shape is 3.
    a. Define the extruded shape in terms of the vertex list, normal list, and face list. When normal to a face is not readily computable, apply Newell's method for computation. Show each list in a table format as discussed in class.
    b. Suppose all the relevant data, e.g., the vertex positions, the faces, and the normals have all been stored in the appropriate arrays and pushed onto the vertex shader, show all the relevant WebGL code (in .js file) to setup proper lighting and object material properties to display a blue and shiny extruded triangle. Use a white directional light with light direction set to [4, 2, 4, 0].
    c. Show WebGL code needed in .js file to put an image "label.jpg" (both the length and width of the image is a power of two, so no special steps is needed for texture mapping) onto 3 sides of the extruded shape as 2D texture. Show ALL the additional code needed for the adding the texture.



6. (15 pts) Apply the Liang Barsky 3D edge/line clipping algorithm to clip the line V_2V_3 against the Canonical View Volume (CVV) as shown below. That is to compute the end points of the line segment that is inside the CVV. Assuming the homogeneous coordinates for V_2 is (8, 4, -8, 5) and for V_3 is (2, 3, 2, 4). Show step-by-step computations in the process.