December 20, 2008 Name _____

CS445 Final

Fall 2008

1.	(max = 5)	5.	(max = 5)
2.	(max = 5)	6.	(max = 15)
3.	(max = 5)	7.	(max = 18)
4.	(max = 5)	8.	(max = 42)
Final Score _____(max=100)			

a) (5 pts total) Discuss the meaning of and motivation for homogeneous coordinates.

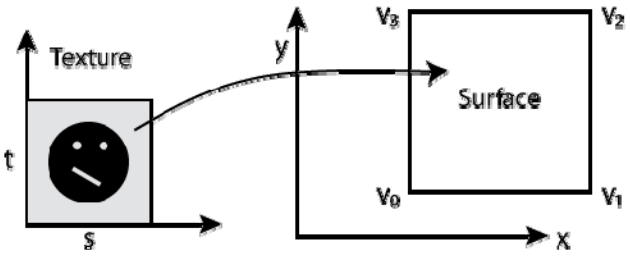
2) (5 pts) What is Gouraud shading and how is it different from Phong shading? What is the advantage/disadvantage of each?

3) (5 pts) Given a triangle in 3D space with vertices V_0, V_1, and V_2. How do you calculate the normal to the triangle surface?

4) (5 pts) How are shadows computed in ray tracing? Please use complete sentences and be sure to define terms. Include picture.

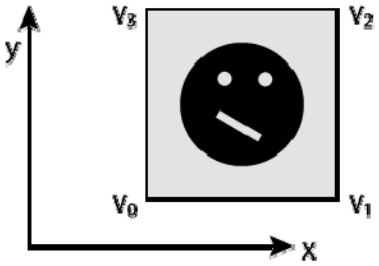
5) (5 pts) If you apply the convolution filter $\begin{pmatrix} -2 & 0 & 0 \\ 0 & 4 & 0 \\ 0 & 0 & -2 \end{pmatrix}$ to an image, what will be the characteristics of the resulting image? Explain your answer.

6) (5 pts each, 15 pts total) Suppose you want to apply the following texture to the square surface on the right.



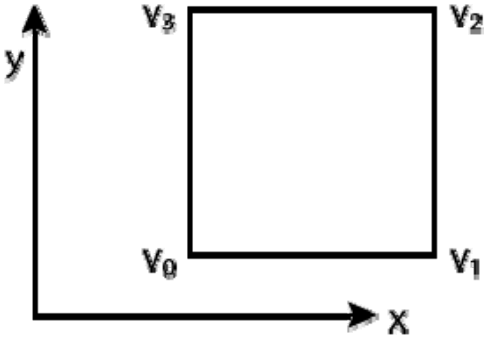
If the code below is applied, the square will be textured as shown below on the right.

```
gl.glTexParameteri(GL.GL_TEXTURE_2D,
   GL.GL_TEXTURE_WRAP_S.GL.GL_REPEAT);
gl.glTexParameteri(GL.GL_TEXTURE_2D,
   GL.GL_TEXTURE_WRAP_T.GL.GL_REPEAT);
gl.glBegin(GL.GL_POLYGON);
  gl.glTexCoord3f(0,0);   gl.glVertex3f(v0x,v0y,0)
  gl.glTexCoord3f(1,0);   gl.glVertex3f(v1x,v1y,0);
  gl.glTexCoord3f(1,1);   gl.glVertex3f(v2x,v2y,0);
  gl.glTexCoord3f(0,1);    gl.glVertex3f(v3x,v3y,0);
gl.glEnd();
```

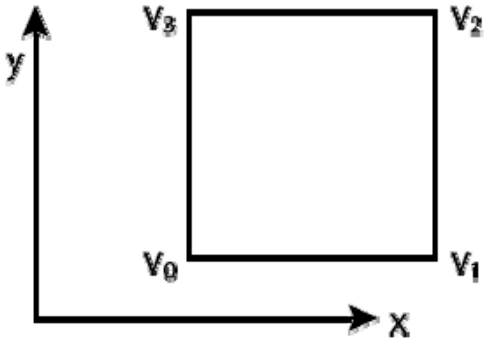


In the rectangle on the right, sketch how the texture would appear if the following texture coordinates are used instead?

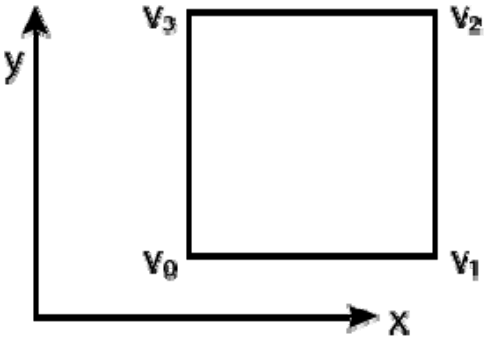
a. gl.glTexCoord3f(0.5,0); gl.glVertex3f(v0x,v0y,0);
   gl.glTexCoord3f(1,0); gl.glVertex3f(v1x,v1y,0);
   gl.glTexCoord3f(1,1); gl.glVertex3f(v2x,v2y,0);
   gl.glTexCoord3f(0.5,1); gl.glVertex3f(v3x,v3y,0);



b. gl.glTexCoord3f(0,1); gl.glVertex3f(v0x,v0y,0);
   gl.glTexCoord3f(1,1); gl.glVertex3f(v1x,v1y,0);
   gl.glTexCoord3f(1,0); gl.glVertex3f(v2x,v2y,0);
   gl.glTexCoord3f(0,0); gl.glVertex3f(v3x,v3y,0);

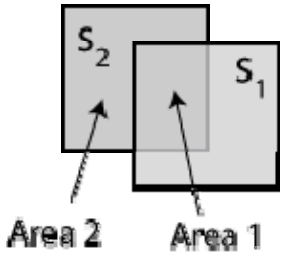


c. gl.glTexCoord3f(0,0); gl.glVertex3f(v0x,v0y,0);
   gl.glTexCoord3f(2,0); gl.glVertex3f(v1x,v1y,0);
   gl.glTexCoord3f(2,1); gl.glVertex3f(v2x,v2y,0);
   gl.glTexCoord3f(0,1); gl.glVertex3f(v3x,v3y,0);



7) (3pts each, 18 pts total) Blending in OpenGL: Suppose that you draw two rectangles S_1 and S_2, using OpenGL where

- S_1 is closer to the camera than S_2.
- S_1 is drawn first followed by S_2.
- The background color is set to white=(1,1,1).
- the glColor is set to green=(0,1,0, α_1) when S_1 is drawn.
- the glColor is set to red=(1,0,0,α_2) when S_2 is drawn.



What is the *<u>resulting (r,g,b) color in the frame buffer</u>* corresponding to each of the following. *Show how you calculated your answer* (you will not get full credit for just an answer written down)

a. **Area 2** if α_1=0.5, α_2=0.25 when z-buffering is turned on.

(r,g,b) = _____

b. **Area 1** if α_1=1.0, α_2=1.0 when z-buffering is turned on.

(r,g,b) = _____

c. **Area 1** if α_1=1.0, α_2=1.0 when z-buffering is turned off.

(r,g,b) = _____

d. **Area 1** if α_1=0.25, α_2=1.0 when z-buffering is turned on.

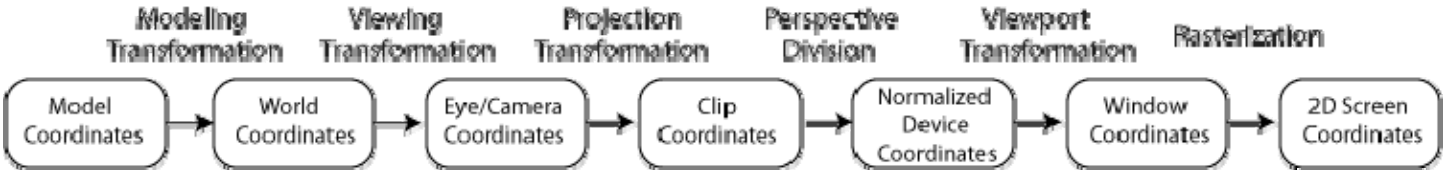
(r,g,b) = _____

e. **Area 1** if α_1=1.0, α_2=0.25 when z-buffering is turned off.

(r,g,b) = _____

f. **Area 1** if α_1=0.5, α_2=0.5 when z-buffering is turned off.
(r,g,b) = _____

8) (6 pts each, 42 pts total) For each of the coordinate systems in the rendering pipeline: 1) what is the meaning and motivation of the coordinate system and 2) describe the transformations or process required to obtain it from the previous coordinate system. Please use complete sentences and include as many specifics as you can remember. Include pictures to clarify the explanation. (Don't rush, you have lots of time!)



a. Modeling Coordinates:

b. World Coordinates & Modeling Transformation:

c. Eye/Camera Coordinates & Viewing Transformation:

d. Clip Coordinates & Projection Transformation:

e. Normalized Device Coordinates & Perspective Division:

f. Window Coordinates & Viewport Transformation:

g. 2D Screen Coordinates & Rasterization: