CSE 287 Final

Name: ____

Write legibly. Ask if you do not understand something. To get partial credit on calculations, show your work. You may use a calculator to complete the exam. The calculator can either be on the PC at your desk or handheld.

1. Assume that **v** is three dimensional vector that would be of type glm::vec3 if represented in a program.

   a. Write code or a mathematical derivation that shows how you would create a vector called vPrime (**v'**) that is three times as long as **v** and points in the same direction as **v**. (3 points)

      v' = 3v

   b. Write code or a mathematical derivation that shows how you would create a vector called vPrime (**v'**) that points in a direction that is the opposite of **v** and has a length of two. (4 points)

      v' = -2 * v / ||v||

2. If the Z buffer or depth buffer algorithm is being used for hidden surface elimination, under what conditions will the color buffer and depth buffer be updated when a particular fragment is being processed? (4 points)

   If the fragment's depth is less than that currently saved in the buffer, the color and depth buffer will be updated to match the fragment.

3. Write out a 4 x 4 homogenous transformation matrix that will cause a non-uniform scale of 2 in the X direction, 6 in the Y direction, and 1 in the Z direction. (4 points).

$$\begin{pmatrix} 2 & 0 & 0 & 1 \\ 0 & 6 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

4. The matrix created by executing the following GLM command

```
glm::perspective(glm::radians(45.0f), 2.0f, 1.0f, 21.0f);
```

would be approximately

$$\begin{bmatrix} 1.2 & 0 & 0 & 0 \\ 0 & 2.4 & 0 & 0 \\ 0 & 0 & -1.1 & -2.1 \\ 0 & 0 & -1.0 & 0 \end{bmatrix}$$

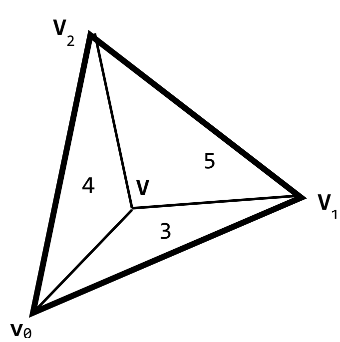
a. Use the matrix and *perspective division* to transform the following vertex to clip coordinates. (4 points)

$$\begin{bmatrix} 1 \\ 3 \\ -5 \\ 1 \end{bmatrix} \quad \begin{pmatrix} 1.2 & 0 & 0 & 0 \\ 0 & 2.4 & 0 & 0 \\ 0 & 0 & -1.1 & -2.1 \\ 0 & 0 & -1 & 0 \end{pmatrix}\begin{pmatrix} 1 \\ 3 \\ -5 \\ 1 \end{pmatrix} = \begin{pmatrix} 1.2 \\ 7.2 \\ 3.4 \\ 5 \end{pmatrix} \qquad \frac{1}{5}\begin{pmatrix} 1.2 \\ 7.2 \\ 3.4 \\ 5 \end{pmatrix} = \begin{pmatrix} 0.24 \\ 1.44 \\ 0.68 \\ 1 \end{pmatrix}$$

b. Indicate whether or not the vertex would be clipped based on the six planes of the normalized view volume. If it is outside the view volume, state why. (4 points)

If x,y, or z are not in the range [-1, 1], the vertex will be clipped. The y coordinate is 1.44 > 1, so this vertex will be clipped.

5. The vertices **v₀**, **v₁**, and **v₂** contain the xyz window coordinates of the corners of the triangle depicted below. The "front face" of the triangle is depicted. For the purposes of writing code, you can assume the vertices are of type glm::vec3. **v** contains the xyz window coordinates of a fragment within the triangle and can also be assumed to be a glm::vec3. The areas of three sub-triangles that include the vertex **v** are also shown.



a. Write C++ code or a mathematical derivation that shows how you find a unit normal vector that that comes out of the front face of the triangle. (4 points)

$$\frac{(v2 - v0)\ x\ (v1 - v0)}{|(v2 - v0)\ x\ (v1 - v0)|}$$

b. Assume the viewing direction is described by the vector **d**. Write a C++ code fragment or a mathematical derivation that shows how you determine whether or not the triangle faces towards or away from the viewing direction. (4 points)

- If the polygon is facing the viewing direction

$$\mathbf{v} \bullet \mathbf{n} \leq 0$$

- If the polygon is facing away from the viewing direction

$$\mathbf{v} \bullet \mathbf{n} > 0$$

c. Write a code fragment or a mathematical derivation that shows how you would determine the area of the sub-triangle with and area of 3. (4 points)

I don't understand this question at all. The area of a sub-triangle with an area of 3 is 3.

d. Based on the areas of the sub-triangles, what will the Barycentric weights for use in interpolating the vertex attributes of $\mathbf{v_0}$, $\mathbf{v_1}$, and $\mathbf{v_2}$. (6 points)

$\mathbf{v_0}$ weight: $\frac{3}{3+4+5}$ $\mathbf{v_1}$ weight: $\frac{4}{3+4+5}$ $\mathbf{v_2}$ weight: $\frac{5}{3+4+5}$

e. If the values for a scalar vertex attribute are $i_0 = 10$, $i_1 = 7$, and $i_2 = 4$ for the vertices $\mathbf{v}_0$, $\mathbf{v}_1$, and $\mathbf{v}_2$ respectively, what will the interpolated value be for the point $\mathbf{v}$. (4 points)

$$\frac{3}{3+4+5}(10) + \frac{4}{3+4+5}(7) + \frac{5}{3+4+5}(4) = \frac{13}{2}$$

f. Write a parametric description of a ray that runs from vertex $\mathbf{v}_0$ towards vertex $\mathbf{v}_1$. The direction vector in the description should be unit length. (4 points)

$$p(t) = v0 + t(v1 - v0)$$

6. Below are material properties for a surface, position and normal vector for a fragment, characteristics of a directional light source, and an eye position. Assume the one light source gives off ambient, diffuse, and specular light.

Surface Color RGB [0.5, 0.4, 0.2]
Surface Specular Color RGB [0.7, 0.7, 0.7,]
Shininess 3.0

Ambient Light RGB [0.2, 0.2, 0.2,]
Diffuse Light RGB [0.6, 0.6, 0.6,]
Specular Light RGB [1.0, 1.0, 1.0]

Fragment Position xyz [-4.0, 0.0, 0.0]
Eye Position xyz [5.0, 5.0, 0.0]
Light direction xyz [-0.87, 0.5, 0.0]
Normal vector xyz [0.0, 1.0, 0.0]
Reflection vector xyz [0.87, 0.5, 0.0]

a. Calculate the view vector for use in lighting calculations. (6 points)

$$\frac{[5.0, 5.0, 0.0] - [-4.0, 0.0, 0.0]}{\left|[5.0, 5.0, 0.0] - [-4.0, 0.0, 0.0]\right|}$$

[0.875, 0.486, 0]

b. Calculate the green component of ambient reflection for the fragment. (3 points)

Alpha value of Green * Ambient Green Intensity = 0.4(0.2) = 0.08

c. Calculate the green component of diffuse reflection for the fragment. (6 points)

max(0, n *l)diffuse color light * diffuse color surface
n = surface normal
l = (p - i) / (||p - i ||) where p = position of light source, i = intersection point

$$[0.0, 1.0, 0.0] \cdot [-0.87, 0.5, 0.0] \cdot 0.6 \cdot 0.4 = 0.12$$

DRAFT

d. Calculate the green component of specular reflection for the fragment. (6 points)

max(0, r * v) ^ shininess * specular color light * specular color surface

$$([0.87, 0.5, 0.0] \cdot [0.875, 0.486, 0])^{3.0} \cdot 1.0 \cdot 0.7 = 0.70896298498$$

e. Calculate the total green component for this light source. (4 points)

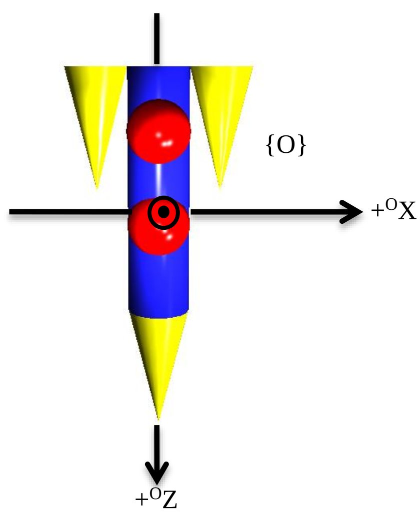
Total Green = 0.71 + 0.12 +0.08 = 0.91 green

7. Consider the following variables.

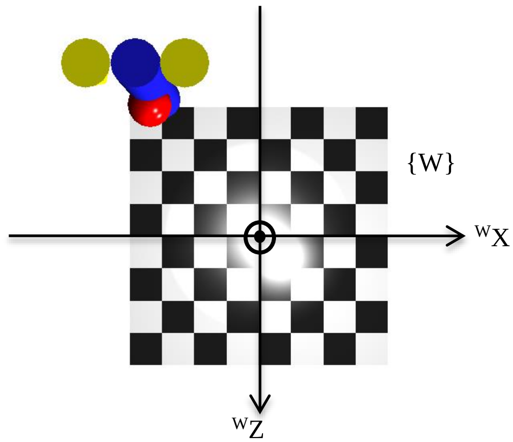
```
glm::vec3 position;          // The position of a vertex or a fragment
glm::vec3 spotDirection;     // Direction in which a spot light is shinning
glm::vec3 spotPosition;      // Position of a spot light
float spotCutOffCosine;      // Cosine of half angle of the spot light beam
```

Assume the variable names reflect what is held in each and that each of the positions and directions are expressed relative to the World coordinate reference frame. Write a code fragment that would determine if the location held in `position` would be in the beam of the spot light. (5 points)

8. The spaceship model is depicted below relative to object coordinates i.e. the picture shows the position of the spaceship without any modeling transformations applied. The object coordinate Y axis is coming out of the page. The "nose" is pointing the positive Z direction. The "top" of the model is represented by the two red spheres and is oriented in the positive Y direction.

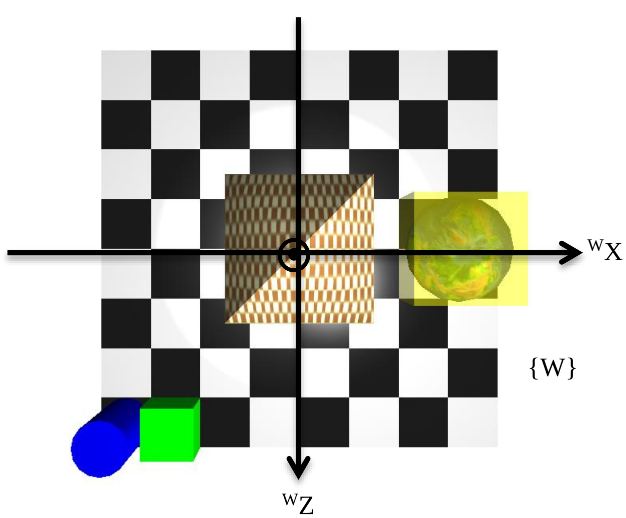


Write a series of statements to create translation and rotation matrices and multiply them together to create the composite modeling transformation that will position and orient the spaceship as depicted below. Relative to World coordinates, the nose of the spaceship is pointing in the negative Y direction.. The position of the ship is X = -2.5, Y = 3, Z = -3.5 relative to the World coordinate origin. (9 points)




```
glm::mat4 modelMatrix =
```

9. The picture belows depict the locations and orientation of objects in a scene relative to the World coordinate origin when no viewing transformations applied. The Y axis is coming out of the page. The tip of the pyramid is at the World coordinate origin. The floor/board is 3 units below the World coordinate origin. (Recall that without viewing transformations, the eye is always at the World coodinate origin looking in the negative Z direction.)



a. Write a series of statements to create translation and rotation matrices and multiply them together to create the viewing transformation for the view of the same scene that is shown below. The tip of the pyramid is centered in the view at a distance of 20. (9 points)

```
glm::mat4 viewMatrix =
```

b. Use the GLM lookAt function to create a viewing transformation that is the same as that which is shown in part a. (6 points)

```
viewMatrix = lookat( vec3(________________, ________________, ________________),
                     vec3(________________, ________________, ________________),
                     vec3(________________, ________________, ________________));
```

10. Which will render a given scene more quickly? Ray tracing (image order algorithm) or a graphics pipeline (object order algorithm)? Why? (6 Points)

11. Below are *source* and *destination* colors, use the *source alpha* value to blend the two colors in order to create a transparency effect. (6 points)

$$source = \begin{bmatrix} 0.8 \\ 0.3 \\ 0.4 \\ 0.4 \end{bmatrix} \qquad destination = \begin{bmatrix} 0.6 \\ 0.3 \\ 0.3 \\ 1.0 \end{bmatrix}$$

(0.4)(0.8) + (1 - 0.4)(0.6) = R Component
(0.4)(0.3) + (1 - 0.4)(0.3) = G Component
(0.4)(0.4) + (1 - 0.4)(0.4) = B Component

12. Consider the following matrices and the corresponding transformations that they describe:

    [M] modeling transformation
    [V] viewing transformation
    [P] projection transformation
    [VP] viewport transformation

    The vertex ${}^{O}v$ describes the coordinates of a vertex in Object coordinates, {O}.

    For each of the coordinate frames below, write an expression that shows the required transformations need to transform ${}^{O}\mathbf{v}$ from Object coordinates to that frame. The matrices should be in the correct order. Include the words, *pd* in the sequence, if perspective division is needed to produce the desired composite transformation. (8 points)

    a. Write and expression that would transform ${}^{O}\mathbf{v}$ to Eye coordinates {E}

       ${}^{eye}\mathbf{v}$ = [M] * oV

    b. Write and expression that would transform ${}^{O}\mathbf{v}$ to Normalized Device Coordinates {NDC}

       ${}^{NDC}\mathbf{v}$ = [P][V][M] * oV

    c. Write and expression that would transform ${}^{O}\mathbf{v}$ to World Coordinates, {W}.

       ${}^{world}\mathbf{v}$ = [V][M] * oV

    d. Write and expression that would transform ${}^{O}\mathbf{v}$ to Window Coordinates, {window}.

       ${}^{window}\mathbf{v}$ = [VP]* pd *[P][V][M] * oV