This is a closed-book, closed-note exam! You are allowed to bring one-page one-sided cheat-sheet and a calculator.

- Print your name in the box above and print your name at the top of every page.
- Please write your answers on the front of the exam pages. You can use the back of the pages as scratch paper. Let us know if you need more paper.
- Read the entire question before writing anything. Make sure you understand what the questions are asking. If you give a beautiful answer to the wrong question, you’ll get no credit. If any question is unclear, please ask us for clarification.
- Don’t spend too much time on any single problem. If you get stuck, move on to something else and come back later.
- To get the credit, sufficient and appropriate intermediate steps are required in your answers.
- Don’t panic and erase large chunks of work. It may be worth partial credit.
Useful equations:

Dot product of two n-dimensional vectors: \( \mathbf{a} \cdot \mathbf{b} = \sum_{i=1}^{n} a_i b_i \)

Cross product of two 3-dimensional vectors: \( \mathbf{c} = \begin{bmatrix} c_1 \\ c_2 \\ c_3 \end{bmatrix} = \mathbf{a} \times \mathbf{b} = \begin{vmatrix} \mathbf{i} & \mathbf{j} & \mathbf{k} \\ a_1 & a_2 & a_3 \\ b_1 & b_2 & b_3 \end{vmatrix} \),

where \( c_1 = \begin{vmatrix} a_2 & a_3 \\ b_2 & b_3 \end{vmatrix} \), \( c_2 = -\begin{vmatrix} a_1 & a_3 \\ b_1 & b_3 \end{vmatrix} \), and \( c_3 = \begin{vmatrix} a_1 & a_2 \\ b_1 & b_2 \end{vmatrix} \)

Matrix-matrix multiplication: \( \mathbf{C}^{n \times l} = \mathbf{A}^{n \times m} \mathbf{B}^{m \times l} \), where \( c_{ij} = \sum_{k=1}^{m} a_{ik} b_{kj} \)
1. (30%) Briefly answer the following questions:
1) What is the role of $\alpha$ in the RGB $\alpha$ color system?

   **Solution:**
   $\alpha$ is an extra channel introduced to RGB color system. It represents transparency of the pixel it represents. When $\alpha = 0$, it’s totally transparent, and $\alpha = 1$ means it’s opaque.
   RGB value of $(r, g, b) \Rightarrow (r\alpha, g\alpha, b\alpha, \alpha)$

2) What are the four key elements of image formation?

   **Solution:**
   The four key elements of image formation are illumination sources, objects, viewer, and attributes of materials.

3) How do you model a ball in OpenGL? You can draw a figure to help illustrate it.

   **Solution:**
   The surface of the ball can be approximated by triangles which can be obtained by bisecting quadrilateral formed by lines of longitude and latitude.
2. (15pts) Consider two colors \( A = \begin{bmatrix} r_A \\ g_A \\ b_A \\ \alpha_A \end{bmatrix} = \begin{bmatrix} 0 \\ 0.5 \\ 1 \\ 0.7 \end{bmatrix} \) and \( B = \begin{bmatrix} r_B \\ g_B \\ b_B \\ \alpha_B \end{bmatrix} = \begin{bmatrix} 1 \\ 0.3 \\ 0 \\ 0.8 \end{bmatrix} \). Compute the composition color \( C = A \) over \( B \).

Solution:

First, we pre-multiply \( \alpha \) value and get \( A' = \begin{bmatrix} 0 \\ 0.35 \\ 0.7 \\ 0.7 \end{bmatrix} \) and \( B' = \begin{bmatrix} 0.8 \\ 0.24 \\ 0 \\ 0.8 \end{bmatrix} \).

\[
C' = A' + (1 - \alpha_A)B' = \begin{bmatrix} 0 \\ 0.35 \\ 0.7 \\ 0.7 \end{bmatrix} + 0.3 \begin{bmatrix} 0.8 \\ 0.24 \\ 0 \\ 0.8 \end{bmatrix} = \begin{bmatrix} 0.24 \\ 0.422 \\ 0.7 \\ 0.94 \end{bmatrix}
\]

\[\Rightarrow C = A \) over \( B = \begin{bmatrix} 0.255 \\ 0.449 \\ 0.745 \\ 0.94 \end{bmatrix}\]
3. (25pts) Suppose a triangle consisting of points $P_1 = [0 \ 0 \ 0]^T$, $P_2 = [1 \ 0 \ 0]^T$, and $P_3 = [0 \ 1 \ 0]^T$. Rotate the triangle by 30 degrees about the z-axis and then translate the rotated triangle by $t_x = 1$, $t_y = 1$, and $t_z = -1$. Give the three vertices of the triangle after the two transformations.

**Solution:**

First, let’s build the rotation matrix $R_z = \begin{bmatrix} \frac{\sqrt{3}}{2} & -\frac{1}{2} & 0 & 0 \\ \frac{1}{2} & \frac{\sqrt{3}}{2} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$.

Then, let’s build the translation matrix $T = \begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & -1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$.

The combined matrix is $M = TR_z = \begin{bmatrix} \frac{\sqrt{3}}{2} & -\frac{1}{2} & 0 & 1 \\ \frac{1}{2} & \frac{\sqrt{3}}{2} & 0 & 1 \\ 0 & 0 & 1 & -1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$.

The new vertices are

$$P'_1 = MP_1 = \begin{bmatrix} \frac{\sqrt{3}}{2} & -\frac{1}{2} & 0 & 1 \\ \frac{1}{2} & \frac{\sqrt{3}}{2} & 0 & 1 \\ 0 & 0 & 1 & -1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ -1 \\ 1 \end{bmatrix}$$

$$P'_2 = MP_2 = \begin{bmatrix} \frac{\sqrt{3}}{2} & -\frac{1}{2} & 0 & 1 \\ \frac{1}{2} & \frac{\sqrt{3}}{2} & 0 & 1 \\ 0 & 0 & 1 & -1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} \frac{\sqrt{3}}{2} + 1 \\ 3 \\ -1 \\ 1 \end{bmatrix}$$

$$P'_3 = MP_3 = \begin{bmatrix} \frac{\sqrt{3}}{2} & -\frac{1}{2} & 0 & 1 \\ \frac{1}{2} & \frac{\sqrt{3}}{2} & 0 & 1 \\ 0 & 0 & 1 & -1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ \frac{\sqrt{3}}{2} + 1 \\ -1 \\ 1 \end{bmatrix}.$$
4. (30pts) For undergraduate students only:

A camera located at $\text{eye} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$ is looking at the origin of the object frame $\text{at} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$ with the up vector defined as $\text{up} = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}$. The frustum is defined by left = -5, right =5, bottom = -4, top = 4, near = 1, and far = 3.

For a point $P = \begin{bmatrix} 2 \\ 1 \\ 0 \\ 1 \end{bmatrix}$ in the object frame, where its projection $Q$ on the projection plane $z= -1$?

Solution:

First, we need to calculate model view matrix, which is the same for both graduate and undergraduate questions.

Step 1: Calculate the normalized view plane normal

$$\text{vpn} = \text{at} - \text{eye} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} - \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} -1 \\ 0 \\ 0 \end{bmatrix}$$

$$\text{n} = \frac{\text{vpn}}{|\text{vpn}|} = \begin{bmatrix} -1 \\ 0 \\ 0 \end{bmatrix}$$

Step 2: Calculate the other two vectors

$$\text{u} = \frac{\text{up} \times \text{n}}{|\text{up} \times \text{n}|} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

$$\text{v} = \frac{\text{n} \times \text{u}}{|\text{n} \times \text{u}|} = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}$$

Step 3: Construct the model view matrix
\[
M_v = \begin{bmatrix}
-u_x & -u_y & -u_z & -u \cdot v
\end{bmatrix}
\begin{bmatrix}
v_x & v_y & v_z \v \cdot v
-n_x & -n_y & -n_z \n \cdot v
0 & 0 & 1
\end{bmatrix}
= \begin{bmatrix}
0 & 0 & -1 & 0
0 & 1 & 0 & 0
1 & 0 & 0 & -1
0 & 0 & 0 & 1
\end{bmatrix}
\]

Then, we need to calculate projection matrix
\[
M_p = M_{orthNSH} = \begin{bmatrix}
\begin{array}{c c c}
\text{2near}\ & 0 & \text{left + right}\ \\
\text{right - left}\ & \text{2near}\ & \text{right - left}\ \\
0 & 0 & 0
\end{array}
\begin{array}{c c c}
\text{top - bottom}\ & \text{2near}\ & \text{top - bottom}\ \\
0 & 0 & 0
\end{array}
\begin{array}{c c c}
0 & 0 & -1
\end{array}
\end{bmatrix}
\]

For undergraduate,
\[
M_p = \begin{bmatrix}
\frac{2 \times 1}{10} & 0 & 0 & 0
0 & \frac{2 \times 1}{8} & 0 & 0
0 & 0 & 0 & 0
0 & 0 & 0 & 0
\end{bmatrix}
\]

\[
Q = M_p M_v P = \begin{bmatrix}
\frac{1}{5} & 0 & 0 & 0
0 & \frac{1}{4} & 0 & 0
0 & 0 & 0 & 0
0 & 0 & 1 & 0
\end{bmatrix}
\begin{bmatrix}
0 & 0 & -1 & 0
0 & 1 & 0 & 0
1 & 0 & 0 & -1
0 & 0 & 0 & 1
\end{bmatrix}
= \begin{bmatrix}
0 & 0 & -1 & 0
\frac{1}{4} & 0 & 0 & -1
\end{bmatrix}
= \begin{bmatrix}
0
-1
\end{bmatrix}
\]

For graduate,
\[
M_p = \begin{bmatrix}
\frac{2 \times 4}{8} & 0 & -2
0 & \frac{2 \times 4}{8} & 2
0 & 0 & 0
0 & 0 & 0
\end{bmatrix}
\]

\[
Q = M_p M_v P = \begin{bmatrix}
1 & 0 & \frac{1}{4} & 0
0 & 1 & \frac{1}{4} & 0
0 & 0 & 0 & 0
0 & 0 & 0 & 0
\end{bmatrix}
\begin{bmatrix}
0 & 0 & -1 & 0
0 & 1 & 0 & 0
1 & 0 & 0 & -1
0 & 0 & 0 & 1
\end{bmatrix}
= \begin{bmatrix}
-1 & 0
\frac{5}{4} & 0
-\frac{5}{4}
0
\end{bmatrix}
= \begin{bmatrix}
1
\frac{5}{4}
-\frac{5}{4}
0
\end{bmatrix}
\]
Bonus question: (15pts) Design a shader-based method to realize an animation effect: a cube moves from left side of the window to the right side with a speed of $v_x$ and $v_y$; and the size of the cube increases such that $s = s + 0.01t$. Describe your method briefly. You can use pseudo code or draw a figure to illustrate your method.

Solution:

The cube is transformed by a scaling and a translation such that

$$T = \begin{bmatrix} 1 & 0 & 0 & v_x t \\ 0 & 1 & 0 & v_y t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{and} \quad S = \begin{bmatrix} 1 + 0.01t & 0 & 0 & 0 \\ 0 & 1 + 0.01t & 0 & 0 \\ 0 & 0 & 1 + 0.01t & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

You can handle the transformation either in the display callback function or within the shader. For the first case, you can set the model view matrix as a “uniform” variable in the shader.

For the second case, you need to set $v_x$, $v_y$, and $t$ or $T$ and $S$ as “uniform” variables in the shader. But you cannot define them as “in” variables.