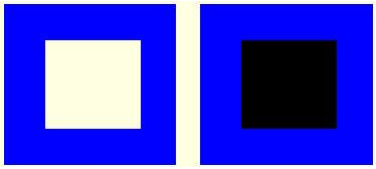
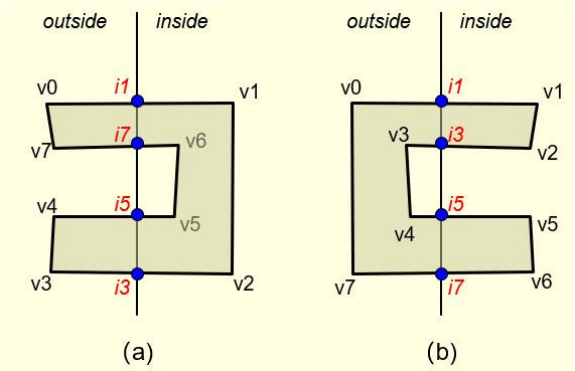
**CS 535 Computer Graphics**
**Homework Assignment 3 (40 points)**
**Due: 9/27/2024**

1. Based on example programs 1-6 in the notes “OpenGL and Shaders”, write a simple C++/openGL program to animate 2D rotation of **one** of the following hollow blue squares of dimension 100x100 (pixels) at the center of your glfw window, the white (black) portion is of dimension 50x50 (pixels). Use “HW3 – Q1” as the title of your glfw window and use white as the background color of your glfw window. Your vertex shader and fragment shader should be written in separate glsl files.



Turn in a screen shot of the rotating hollow square and your program with the submission of your HW3. (10 points)

2. If the two polygons shown in (a) and (b) are clipped against the left bounding edge of a 2D window using Sutherland-Hodgman and Weiler-Atherton algorithms, respectively, what would the outputs be in each case? In each case, the output by each algorithm is supposed to be a sequence of points (vertices of the original polygon or intersection points of the edges of the polygon with the left bounding edge of the window) or several sequences of points. For your convenience, intersection points of the polygon with the left bounding edge of the window are also shown below. (6 points)




3. In perspective projection, if center of projection (COP) is at (-3, 0, -1) and the projection plane passes through the origin and is perpendicular to the vector (-3, 0,

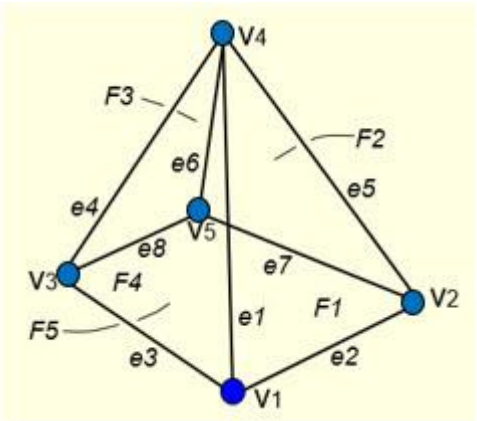
-1) then what (4x4) matrix should we use to compute the projection of a given point (x, y, z) on the projection plane in homogeneous coordinates? (5 points)

4. In perspective projection, if the projection plane is perpendicular to *z* axis at z = -4 and **vanishing point** of a line passing through A=(-5, 0, 0) and B=(0, 0, -5) is at (5, 0, -4), then where is the **center of projection**? (5 points)

5. In 3D graphics, for perspective projection, we do clipping in homogeneous coordinates, i.e, before the perspective division step. Why? (5 points)

6. For the alternative data structure given in slides 3-6 of the notes "3D Data Structures" for 3D objects with bounding faces (polygons) represented as sequences of bounding edges, which of the following queries can be answered in constant time? (4 points)
   (a) Given a face, which faces are adjacent to it?
   (b) Given an edge, which faces are sharing it as a bounding edge?
   (c) Given a vertex, which faces are adjacent to it?
   (d) Given a vertex, which edges are adjacent to it?

7. To use the winged-edge data structure to represent the following pyramid, we need an Edge Table, a Vertex-Edge Table and a Face-Edge Table. To find all the **adjacent edges** of vertex v4, how many times do we have to access the Edge Table, the Vertex-Edge Table and the Face-Edge Table, respectively? Justify your answer. (5 points)




- Solutions must be typed (word processed) and emailed to me both as a pdf file and a word document before 23:59 on 9/27/2024.
- Please name your files as:
  CS535_HW3_2024f_LastName.docx / CS535_HW3_2024f_LastName.pdf