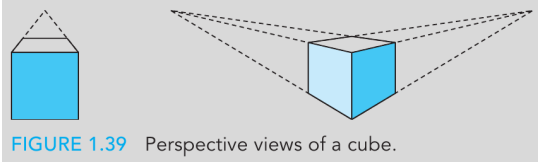
1.7 Consider the perspective views of the cube shown in Figure 1.39. The one on the left is called a one-point perspective because parallel lines in one direction of the cube - along the sides of the top - converge to a vanishing point in the image. In contrast, the image on the right is a two-point perspective. Characterize the particular relationship between the viewer, or a simple camera, and the cube that determines why one is a two-point perspective and the other a one-point perspective.



**One vanishing point when a face is parallel to the camera.**

**Two vanishing points when an edge is parallel to the camera.**

**Similarly, three vanishing points when no edge nor face is parallel to the camera.**

1.8 The memory in a frame buffer must be fast enough to allow the display to be refreshed at a rate sufficiently high to avoid flicker. A typical workstation display can have a resolution of 1280 X 1024 pixels. If it is refreshed 72 times per second, how fast must the memory be? That is, how much time can we take to read one pixel from memory?

**1280\*1024\*72 = 94,371,840 pixels per second**

What is this number for a 480 X 640 display that operates at 60 Hz but is interlaced?**480\*640\*60\*0.5(interlaced) = 9,216,000 pixels per second**

1.9 Movies are generally produced on 35 mm film that has a resolution of approximately 2000 X 3000 pixels. What implication does this resolution have for producing animated images for television as compared with film?

1.10 Answer the following questions.

i. What is the resolution and refresh rate of your screen/display?

1680\*1050 @ 60hertz

ii. Which browsers support WebGL? List four of the major ones.

Chrome, safari, firefox, Microsoft edge

iii. Is GLSL program platform dependent?

No.

iv. Are GLSL shaders stand-alone applications? That is, can they execute on their own like a stand-alone C application?

GLSL shaders not designed to be run as a standalone application

v. Where do you compile and link your GLSL programs?

webgl provides functions to compile shaders, we use specific libraries from the textbook to compile/link our shaders

vi. When a JS function is called?

static javascript runs on page load, but specific functions can be called with events

1.1 In representing a picture, one-bit and eight-bit-deep frame buffers allow only two and 256

colors, respectively. How many bits does a full-color system allow?

24 bits allows for 16,777,216 colours

1.2 In computer graphics, objects such as spheres are usually approximated by simpler objects constructed from flat polygons (polyhedra). Using lines of longitude and latitude, define a set of simple polygons that approximate a sphere centered at the origin. Can you use only quadrilaterals or only triangles?

You cannot use only quadrilaterals because polygons at the poles all share a vertex, resulting in triangles. You can use all triangles, however.

1.4 Consider the clipping of a point in two dimensions against a circular clipping window.

Show that you require only the coordinate of the point and the center and radius of the circle to determine whether the point is not clipped, or is clipped out completely.

1.5 For a line segment, show that clipping against the top of the clipping rectangle can be done independently of the clipping against the other sides. Use this result to show that a clipper can be implemented as a pipeline of four simpler clippers

2.9 We saw that a fundamental operation in graphics systems is to map a point (x, y) that lies within a clipping rectangle to a point (xs, ys) that lies in the viewport of a window on the screen. Assume that the two rectangles are defined by the viewport specified by glViewport(u, v, w, h); and a viewing rectangle specified by

xmin ≤ x ≤ xmax,

ymin ≤ y ≤ ymax.

Find the mathematical equations that map (x, y) into (xs, ys).

2.11 In practice, testing each point in a polygon to determine whether it is inside or outside the polygon is extremely inefficient. Describe the general strategies that you might pursue to avoid point-by-point testing.

2.14 In OpenGL we specify polygons using lists of vertices. Why might it be better to define polygons by their edges? Hint: Consider how you might represent a mesh efficiently.

Part B

Answer the following JavaScript related questions.

a. Do you need to compile JavaScript code in order to execute it?

No, javascript is interpreted

b. Variable type: Do you need to declare the type of a variable before using it? Why?

c. Variable scoping: What are global variables? How are they declared? What are the problems with using globals?

d. How are errors gracefully handled in JavaScript?

You can use try..catch statements in javascript, or throw an exception to catch an error

e. Does JavaScript support objects?

f. What is the relationship between canvas and WebGL?

4.1 Answer the following questions:

a. Can rotations and translations alter the shape of an object?

No

b. Can scaling alter the shape of an object?

c. Is there any point that remains unchanged by a rotation?

Any point on the axis of rotation

4.23 Given two nonparallel, three-dimensional vectors u and v, how can we form an orthogonal coordinate system in which u is one of the basis vectors?

4.21 We defined an instance transformation as the product of a translation, a rotation, and a scaling. Can we accomplish the same effect by applying these three types of transformations in a different order?

Check whether the following sequences commute:

a. A rotation and a uniform scaling

b. Two rotations about the same axis

c. Two translations

d. A rotation and a translation

4.4 If we are interested in only two-dimensional graphics, we can use three dimensional homogeneous coordinates by representing a point as p = [xy1]T and a vector as v = [ab0]T. Find the 3 X 3 rotation, translation, scaling, and shear matrices.

How many degrees of freedom are there in an affine transformation for transforming two dimensional points?

4.9 In two dimensions, we can specify a line by the equation y = mx + h. Find an affine transformation to reflect two-dimensional points about this line.

4.12 Find a homogeneous-coordinate representation of a plane, which passes through a point P0 and perpendicular to the vector n.

5.0 Not all projections are planar geometric projections. Give an example of a projection in which the projection surface is not a plane.

5.17 Find the projection of a point onto the plane ax + by + cz + d = 0 from a light source located at infinity in the direction (dx, dy, dz).

5.1 ALL perspective views are characterized by diminution of size. When objects moved farther from the viewer, their images becomes smaller. How does the classical perspective view work?

5.7 What is a shadow polygon?

5.21 Stereo images are produced by creating two images with the viewer in two slightly different positions. Consider a viewer who is at the origin but whose eyes are separated by dx units. What are the appropriate viewing specifications to create the two images?

5.25 What is the use of Projection Normalization?