

Test 2 Review

1. Coordinate system and frame

- a. What is the significance of having homogeneous coordinates for defining points and vectors in computer graphics?
- b. A frame is a coordinate system (defined with 3 vectors) plus the origin point. In general, when converting from one frame to another frame, i.e., from the world frame to the eye frame, what transformation matrix is applied to transform the points and vectors from within the first frame to the second?

2. Transformations

- Programming involving 2D and 3D transformations
 - For example, given description of a 2D or 3D object and the final desired 2D or 3D configuration, write program to draw that configuration using translate, rotate, scale transformations as well as push and pop `modelViewMatrix` operations.
 - What is the purpose of the push and pop `modelViewMatrix` operations?
- Understand the transformation matrices corresponding to individual transformations. For example, show the matrix for a rotation of 35 degrees around vector (0, 0, 1).
- `ModelView` (VM)
 - What are the Model and View parts of the matrix?
 - How to compute the V matrix given a `lookAt` function?
 - How to move camera in the camera coordinate system? (move the camera to the left, right, up, down; and around a scene)
 - How to compute the M matrix given the transformation functions?
- Coordinate systems
 - Change from the world coordinates to the eye/camera coordinates
 - Understand how to compute the new coordinates of the objects drawn in the world coordinates
 - Coordinate system transformations involved in 2D and 3D transformations.
- Projection matrix
 - What is the main purpose of performing projection in the graphics pipeline?
 - What is a CVV?
 - Compute the matrix P in Orthographic projection given javascript code setting up this type of projection
 - Compute the matrix P in Perspective projection given javascript code setting up this type of projection
 - Given a perspective projection specified in frustum, convert it to perspective, and vice versa.
 - Given object drawn in world coordinate, and the perspective project setup, show the image displayed in the view plane
 - How does the perspective matrix achieve the effect where objects further away from the eye looks shorter than those closer to the eye?
 - What is the perspective transformation? What does it achieve in transforming vertices?
 - What is the perspective division?
 - How is the pseudo depth of each vertex defined?
 -

3. Polygonal Meshes

- Understand how to define a polygon mesh in terms of vertex list, normal list, and face list. Compute normal using vector cross product or using the Newell's method.
- What are the advantages of Newell's method when compared with computing normal using cross product of two vectors on the plane?

- What is an extruded shape? How to define an extruded shape? What to do when the extruded shape you want to draw is non-convex?
- Be able to write code to draw a 3D object in terms of surface of revolution

4. Lighting

- What are the differences between ambient light, diffuse light, and specular light sources?
- Understand how is the lighting/material effect computed for each vertex or each fragment of a 3D object
- What is the
- Understand how reflected light from diffuse light and specular light source are computed
 - Why is the angle between **light direction** and **normal** used for diffuse light computation, and the angle between **viewer direction** and **normal** used for specular computation?
 - What is the purpose of performing half way vector computation for specular light computation? How is it computed?
- Given description of the desired color of an object in a scene, understand how to specify light source (color, intensity, and position), the material properties of the object, as well as global factors to achieve the desired effect.
 - Understand the different factors that contribute to color of objects under lighting.

5. Texture

- What are the texture coordinates?
- How to specify the portion of a 2D texture for texture mapping using the texture coordinates, i.e., the (s, t) pairs
- Be able to attach textures to objects: given a 2D texture, be able to apply it to the face of a 2D or 3D object
- What is the problem of minification and magnification in the context of texture mapping?
- Understand the difference between clamping and repeating texture wrapping methods?
- Understand the difference between “Nearest”, “Linear” filtering methods used for minification and magnification, as well as their usage with mipmap.

6. Color blending

- a. Understand how color maybe blended using different blending methods/modes
- b. Be able to compute the blended code with javascript/WebGL code