Homework 4 - Viewing & Projection
Solution

1. **What matrix is created by LookAt(eye=[0,0,2], at=[0,0,0], up=[0,1,0])? Is there a way to get this matrix using translate and/or rotate and/or scale? Explain.**

$$= \text{setLookAt}(0,\ 0,\ 2,\ 0,\ 0,\ 0,\ 0,\ 1,\ 0) = \begin{vmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & -2 \\ 0 & 0 & 0 & 1 \end{vmatrix}$$

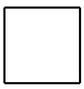
Notice that this is just a translate matrix. The goal of LookAt is to transform the world such that the camera is at the origin looking down the negative z-axis, while keeping everything else in relative position. The matrix is the one that accomplishes that action. We could achieve that with Rotate and Translate matrices just as well. How can we get EYE onto the origin? Well Translate(-eye) will do that. Think of multiplying a vertex at the camera location by the matrix. We dont actually every multiply the camera point, but we do multiply all the real vertices with the view matrix. Once we have eye in position, how to get AT onto the Z-axis? A couple of rotates will do that. In this case its already pointed the right way, so those are just identity matrices. No need to actually rotate. The resulting matrix is exactly the same whether produced with LookAt or produced with Translate and Rotate.

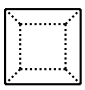
In this case. You can use:

$$R(pitch) * R(yaw) * T(-eye) = \begin{vmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{vmatrix} * \begin{vmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{vmatrix} * \begin{vmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & -2 \\ 0 & 0 & 0 & 1 \end{vmatrix} = \begin{vmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & -2 \\ 0 & 0 & 0 & 1 \end{vmatrix}$$

2. **Illustrate the difference between orthographic (parallel) and perspective projection.**

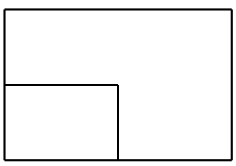
Imagine a cube:

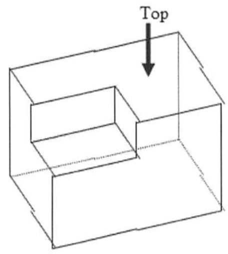
Orthogonal: 

Parallel Projection: 

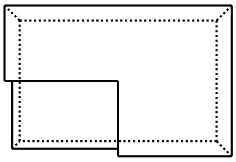
3. **Consider the 3D solid cube, with a cube-shaped notch removed from one corner.**

   (a) **Sketch a wire frame of the top view of the object, using orthogonal projection.**



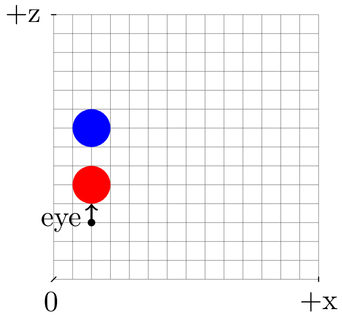


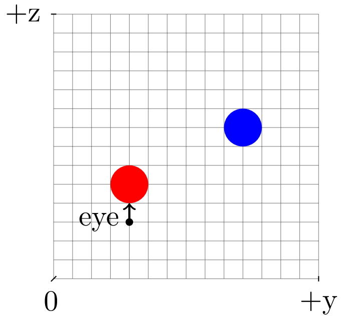
(b) **Sketch a wire frame of the top view of the object, using perspective projection.**



4. **Suppose we set up a camera using LookAt(eye=[2,4,3], at=[2,4,4], up=[0,1,0]) and render a scene with two spheres of radius 1, centered at [2,4,5] and [2,10,8].**

   (a) **Draw this scene in 2D both on the x-z plane (left) and on the y-z plane (right). Label the eye and the at points as well as the two spheres.**

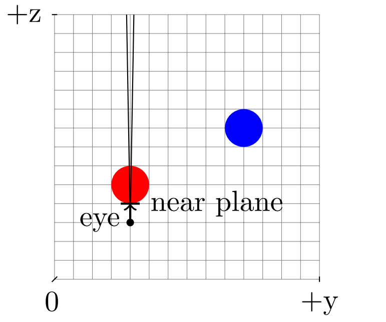
   

   (b) **What is the distance to the near and far plane if they are set to bound the objects as closely as possible? Describe the distance in numerical values and explain it.**

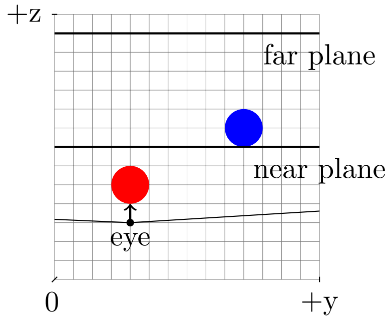
   The maximum *near* distance is 1 and the minimum *far* distance is 6. This is the distance to the closest point of the first object and the distance to the farthest point of the farthest object.

   (c) **If we render this scene with a projection matrix, how many spheres will be seen in each of the following cases? Why?**

   i. perspective(fovy=1°, aspect=1, zNear=1, zFar=100)
   We would see the red sphere. We can see the sphere in front of the camera but the fov is too narrow to see the whole thing



ii. perspective(fovy=175°, aspect=1, zNear=4, zFar=10)
We can see one spheres as the fov is wide enough to capture both spheres, however, the near plane is past the red sphere.



iii. perspective(fovy=60°, aspect=1, zNear=1, zFar=10)
We can see only the red sphere as the fov is too narrow to see both.