CS174A Assignment 1 - Part 1
Written Section: Transformations

Out: Fri 6 Oct 2017   **Value:** 5% of final grade
Due: Fri 20 Oct 2017 12pm   **Total Points**: 137

1. (45 pts) The point coordinate **p** can be expressed as $\mathbf{p} = 0+3*\mathbf{i} + 1*\mathbf{j}$, where **i** and **j** are basis vectors of unit length along the **x** and **y** axes, respectively.
   a. (6pts) Describe the point **p** in terms of the 3 other coordinate systems given below.
   b. (18pts) How many points does one need to describe a vector? Describe the vector **v** in terms of the 3 other coordinate systems given below.
   c. (3pts) How is a coordinate frame represented?
   d. (18pts) Give the three coordinate frame representations, $\mathbf{M_A}$, $\mathbf{M_B}$, $\mathbf{M_C}$, that relate each of the coordinate frames, A, B, C to the world coordinate frame W
   e. (6pts) Given the position of **p** in each of the three coordinate frames computed in (a), show that each of the coordinate frame representations from (d) can be used as transforms to compute the position of **p** in the world coordinate frame.
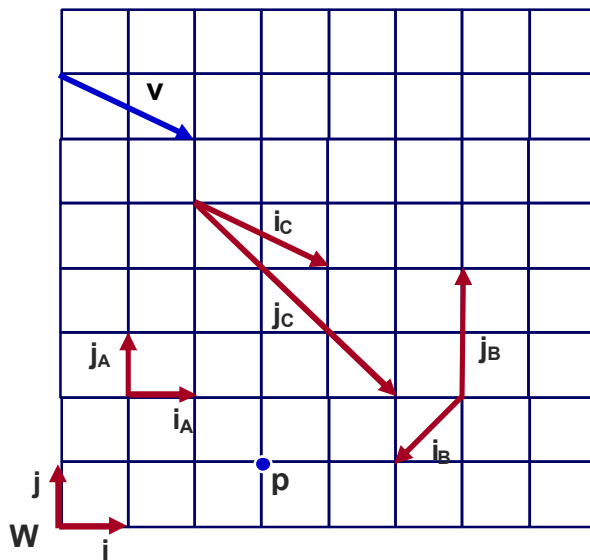
2. (3 pts) Write down the affine matrix for scaling an object by 2 in **y** and 3 in **z**.

3. (10 pts) Give the OpenGL Shader commands required to create **M** using elementary transformations . You may assume the matrix stack has been initialized with `modelMatrix.setAsIdentity();`.

$$\begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 2 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

4. (6 pts) Homogenize the point $[2\ 10\ 8\ 4]^T$.

5. (20 pts) Write down the 4x4 OpenGL modelview matrix at the four lines A, B, C, and D below.

```
modelMatrix.setAsIdentity();
modelMatrix = modelMatrix * Translate(2,3,0);
//A
modelMatrix = modelMatrix * RotateY(90);
//B
matrixStack.push(modelMatrix);
modelMatrix = modelMatrix * Scale(1,.5,1);
modelMatrix = modelMatrix * Translate(1,1,0);
//C
modelMatrix = matrixStack.pop();
modelMatrix = modelMatrix * Scale(2,1,1);
//D
```

6. (24pts) Write the composite 2D matrix **M** that performs a reflection about an arbitrary tilted line using the elementary transformations and write the equivalent OpenGl Shader code that generates **M**.
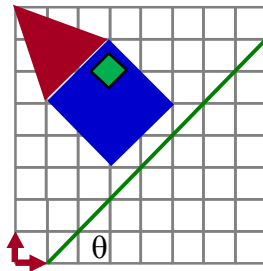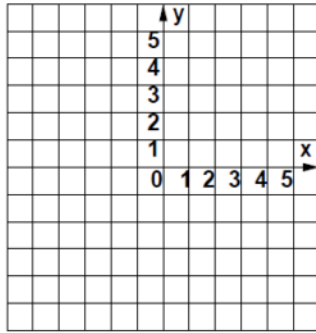

7. (26 pts) For each equation below,
   - sketch the new location L' of the L shape on the grid and
   - provide the sequence of commands needed to carry out those operations.

Use the function drawL(), which draws an L shape with the lower left corner at the current origin as shown below. You may assume the matrix mode is matrixStack.push(modelMatrix) and that the stack has been initialized with modelMatrix.setAsIdentity().
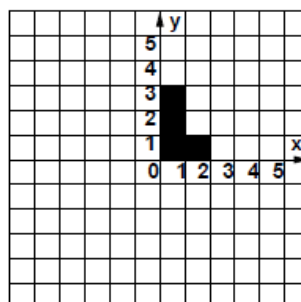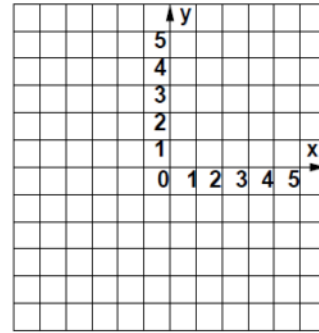
$$
A = \begin{bmatrix} 2 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, B = \begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, C = \begin{bmatrix} 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, D = \begin{bmatrix} -1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}
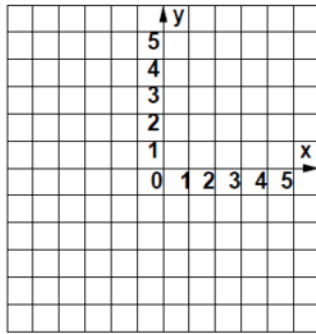$$

a) L' = ABC L

b) L' = CAD L

c) L' = CBD L

d) L' = DCCAD L