CS370 - Assignment 1

1. What are the main advantages and main disadvantages to generating images using the graphics pipeline?

The main advantage to a pipeline architecture is performance. The hardware is optimized to operate on particular primitives in a parallel fashion thus significantly enhancing rendering speed. Also all the objects within the scene do not need to be stored in memory, but can be generated and processed as needed. Unfortunately this architecture does not allow for global aspects of the scene to be incorporated into the rendering, e.g. reflections, shadows, etc. Such global effects are necessary to produce truly photorealistic images.

2. Show that a rotation and a *uniform* scaling transformations (i.e. one where all the scale factors are identical) are commutative, i.e. that they can be applied in either order.

A rotation matrix by an angle θ about the z-axis is given by

$$R = \begin{bmatrix} cos(\theta) & -sin(\theta) & 0 & 0 \\ sin(\theta) & cos(\theta) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

A *uniform* scaling matrix (i.e. same in all dimensions) by a scale factor β is given by

$$S = \begin{bmatrix} \beta & 0 & 0 & 0 \\ 0 & \beta & 0 & 0 \\ 0 & 0 & \beta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

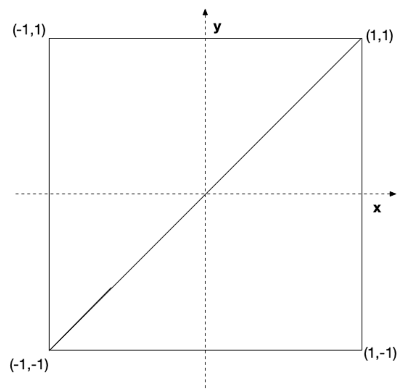
Thus the matrix products would be

$$RS = \begin{bmatrix} cos(\theta) & -sin(\theta) & 0 & 0 \\ sin(\theta) & cos(\theta) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \beta & 0 & 0 & 0 \\ 0 & \beta & 0 & 0 \\ 0 & 0 & \beta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} cos(\theta)\beta & -sin(\theta)\beta & 0 & 0 \\ sin(\theta)\beta & cos(\theta)\beta & 0 & 0 \\ 0 & 0 & \beta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

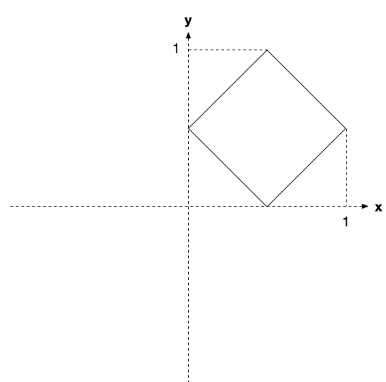
$$SR = \begin{bmatrix} \beta & 0 & 0 & 0 \\ 0 & \beta & 0 & 0 \\ 0 & 0 & \beta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} cos(\theta) & -sin(\theta) & 0 & 0 \\ sin(\theta) & cos(\theta) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \beta cos(\theta) & \beta(-sin(\theta)) & 0 & 0 \\ \beta sin(\theta) & \beta cos(\theta) & 0 & 0 \\ 0 & 0 & \beta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

And thus $RS = SR$ for rotation and *uniform* scaling. **Note:** This is **not** necessarily true for *non-uniform* scalings and other transformation matricies, e.g. translations.

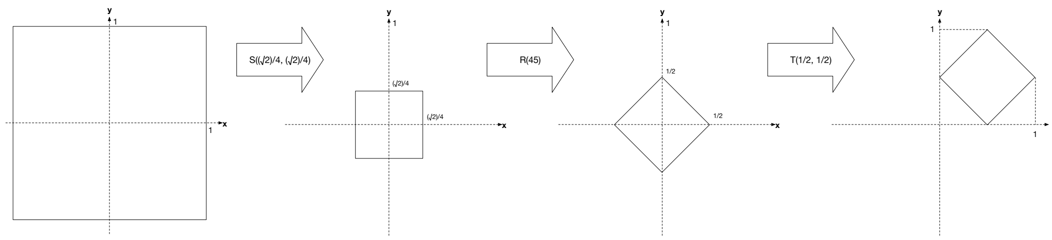
3. Given the following set of vertices that define a square



Sketch a set of *intermediate* transformations to produce the following end result with proper size, orientation, and location. Give the final product of your particular transformation sequence using **T(dx,dy)** for a translation by dx and dy, **R**(θ) for a rotation (about the z axis) by angle θ, and **S(sx,sy)** for a scaling by sx and sy.



First we see that the initial length of the sides of the original square is 2 units, yet the length of the sides of the final square is $\sqrt{2}/2$. Thus we can start by scaling both x and y by $\sqrt{2}/4$ (note since the original square's center is already at the origin, the scaled square's center will remain at the origin). Next we can rotate the square by 45° (i.e. counter-clockwise) again since the scaled square's center is at the origin it will rotate about it's center. Finally we can translate it into position by moving it 0.5 units in x and y. Thus the sequence can be shown graphically as

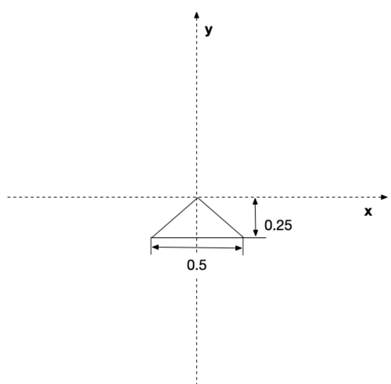


And thus the final transformation matrix would be given by

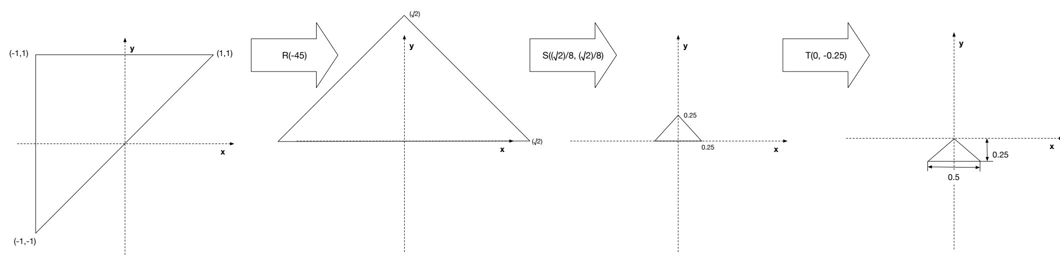
$$M = T(0.5, 0.5)R(45)S(\sqrt{2}/4, \sqrt{2}/4)$$

Note in this case since the scaling is *uniform* we could commute the scaling and rotation transformations, i.e. perform the rotation first.

Sketch a set of *intermediate* transformations using only the *upper left* triangle vertices to produce the following end result with proper size, orientation, and location. Give the final product of your particular transformation sequence using **T(dx,dy)** for a translation by dx and dy, $\mathbf{R}(\theta)$ for a rotation (about the z axis) by angle θ, and **S(sx,sy)** for a scaling by sx and sy.



First we see that the center of the diagonal edge is at the origin, thus if we rotate by $-45°$ (i.e. clockwise) the diagonal edge will align with the x-axis with its center still at the origin. We can then scale to the proper width of 0.5 and height 0.25 by $\sqrt{2}/8$ in x and $\sqrt{2}/8$ in y. Finally we can translate it into position by moving it 0 units in x and -0.25 units in y. Thus the sequence can be shown graphically as



And thus the final transformation matrix would be given by

$$M = T(0.0, -0.25)S(\sqrt{2}/8, \sqrt{2}/8)R(-45)$$

Note in this case since the scaling is *non-uniform* we cannot commute the scaling and rotation transformations without adjusting the parameters.

There are many other sequences (with different parameter values) that can achieve the same final results. However, the net transformation matrix product M MUST BE THE SAME regardless of the sequence.