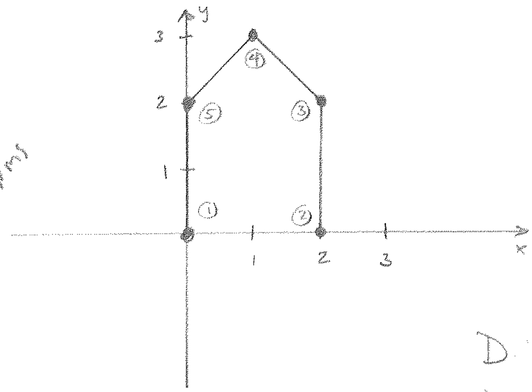
2.7 Applications to computer graphics

In computer graphics, particularly interactive computer graphics, objects are usually represented by a set of vertices.

Explain why we only need to transform vertices (Transforms are linear)



$$D = \begin{bmatrix} 0 & 2 & 2 & 1 & 0 \\ 0 & 0 & 2 & 3 & 2 \end{bmatrix}$$
$$\quad ① \quad ② \quad ③ \quad ④ \quad ⑤$$

D is the <u>data matrix</u> for this house.

Now, suppose that we wanted to scale the house so that it was twice as tall. Recall that we've seen a scale matrix before: $\begin{bmatrix} S_x & 0 \\ 0 & S_y \end{bmatrix}$

To make the house twice as tall but the same width we need $S_x = 1$ and $S_y = 2$. So

$$\begin{bmatrix} 1 & 0 \\ 0 & 2 \end{bmatrix} \begin{bmatrix} 0 & 2 & 2 & 1 & 0 \\ 0 & 0 & 2 & 3 & 2 \end{bmatrix} = \begin{bmatrix} 0 & 2 & 2 & 1 & 0 \\ 0 & 0 & 4 & 6 & 4 \end{bmatrix}$$

To invert the house we would use

$$\begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \begin{bmatrix} 0 & 2 & 2 & 1 & 0 \\ 0 & 0 & 2 & 3 & 2 \end{bmatrix} = \begin{bmatrix} 0 & 2 & 2 & 1 & 0 \\ 0 & 0 & -2 & -3 & -2 \end{bmatrix}$$

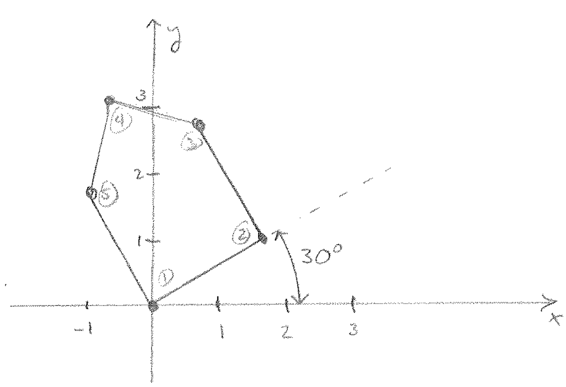
We can also rotate the house. The matrix that causes a <u>counterclockwise</u> rotation of angle θ about the origin is

$$\begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}$$

So to rotate the house by $30°$ we would use

$$\begin{bmatrix} \frac{\sqrt{3}}{2} & -\frac{1}{2} \\ \frac{1}{2} & \frac{\sqrt{3}}{2} \end{bmatrix}\begin{bmatrix} 0 & 2 & 2 & 1 & 0 \\ 0 & 0 & 2 & 3 & 2 \end{bmatrix} = \begin{bmatrix} 0 & \sqrt{3} & \sqrt{3}-1 & \frac{\sqrt{3}}{2}-\frac{3}{2} & -1 \\ 0 & 1 & 1+\sqrt{3} & \frac{1}{2}+3\frac{\sqrt{3}}{2} & \sqrt{3} \end{bmatrix}$$

$$\cong \begin{bmatrix} 0 & 1.73 & 0.73 & -0.63 & -1 \\ 0 & 1 & 2.73 & 3.10 & 1.73 \end{bmatrix}$$



Notice that the point at the origin stays at the origin

Question: How can we translate the house? i.e., we want each point $\begin{bmatrix} x \\ y \end{bmatrix}$ to be mapped to $\begin{bmatrix} x+h \\ y+k \end{bmatrix}$.

If this is a linear transformation from $\mathbb{R}^2$ to $\mathbb{R}^2$ then we must be able to find a matrix A that implements this transformation.

$$\begin{bmatrix} x+h \\ y+k \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}\begin{bmatrix} x \\ y \end{bmatrix} \quad ? \quad \text{NO!}$$

We need $\begin{bmatrix} x+h \\ y+k \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} h \\ k \end{bmatrix}$

For reasons we'll see in a few minutes, we'd really like to be able to implement every transformation (scaling, shearing, rotating, translating) with a matrix multiplication.

We say that the <u>Homogeneous coordinates</u> form of $\begin{bmatrix} x \\ y \end{bmatrix}$ is $\begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$. Thus, we are going to consider $\mathbb{R}^2$ as a plane in $\mathbb{R}^3$ located one unit above the xy-plane.

In homogeneous coordinates we can translate with a linear transformation:

$$\begin{bmatrix} x+h \\ y+k \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & h \\ 0 & 1 & k \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

We now have a 3x3 transformation matrix.

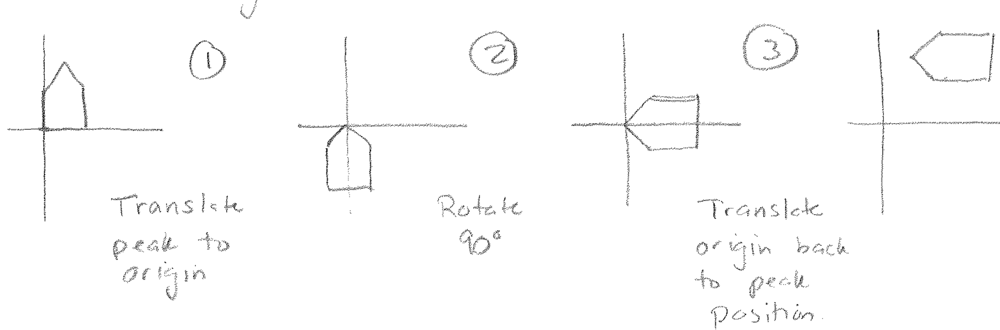
We can also describe the other transformations:

Scaling	Shear	Rotation
$\left[\begin{array}{cc:c} S_x & 0 & 0 \\ 0 & S_y & 0 \\ \hdashline 0 & 0 & 1 \end{array}\right]$	$\left[\begin{array}{cc:c} 1 & S_x & 0 \\ S_y & 1 & 0 \\ \hdashline 0 & 0 & 1 \end{array}\right]$	$\left[\begin{array}{cc:c} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ \hdashline 0 & 0 & 1 \end{array}\right]$
$\begin{bmatrix} S & 0 \\ 0 & 1 \end{bmatrix}$	$\begin{bmatrix} Sh & 0 \\ 0 & 1 \end{bmatrix}$	$\begin{bmatrix} R & 0 \\ 0 & 1 \end{bmatrix}$

partitioned form

Now, consider our house once again. Suppose we want to rotate it 90° <u>about the peak of the roof</u>?

We can't just use our rotation matrix since it always rotates about the origin.



$$T_1 = \begin{bmatrix} 1 & 0 & -1 \\ 0 & 1 & -3 \\ 0 & 0 & 1 \end{bmatrix} \qquad R = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \qquad T_2 = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 3 \\ 0 & 0 & 1 \end{bmatrix}$$

If $D = \begin{bmatrix} 0 & 2 & 2 & 1 & 0 \\ 0 & 0 & 2 & 3 & 2 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix}$ Then we have

$D_1 = T_1 D$ translate peak to origin

$D_2 = R D_1$ rotate 90°

$D_3 = T_2 D_2$ translate origin to peak position

Notice $D_3 = T_2 D_2 = T_2 (R D_1) = T_2 (R (T_1 D)) = T_2 R T_1 D$

$\therefore \; D_3 = (T_2 R T_1) D$

↑ composite transformation matrix

2.7

By using the composite transformation matrix we can save operations: If D has n columns then:

To compute $D1$ we do $(9\text{ mult} + 6\text{ add})n = 15n$

$D2$ " " $15n$

$D3$ " " $15n$

Summing we find that we do $45n$ operations to compute $D3$.

However, to compute $T = T_2 R T_1$ we do
$3 \cdot 15 + 3 \cdot 15 = 6 \cdot 15 = 90$ operations

Then to compute $D3$ we do $90 + 15n$ operations