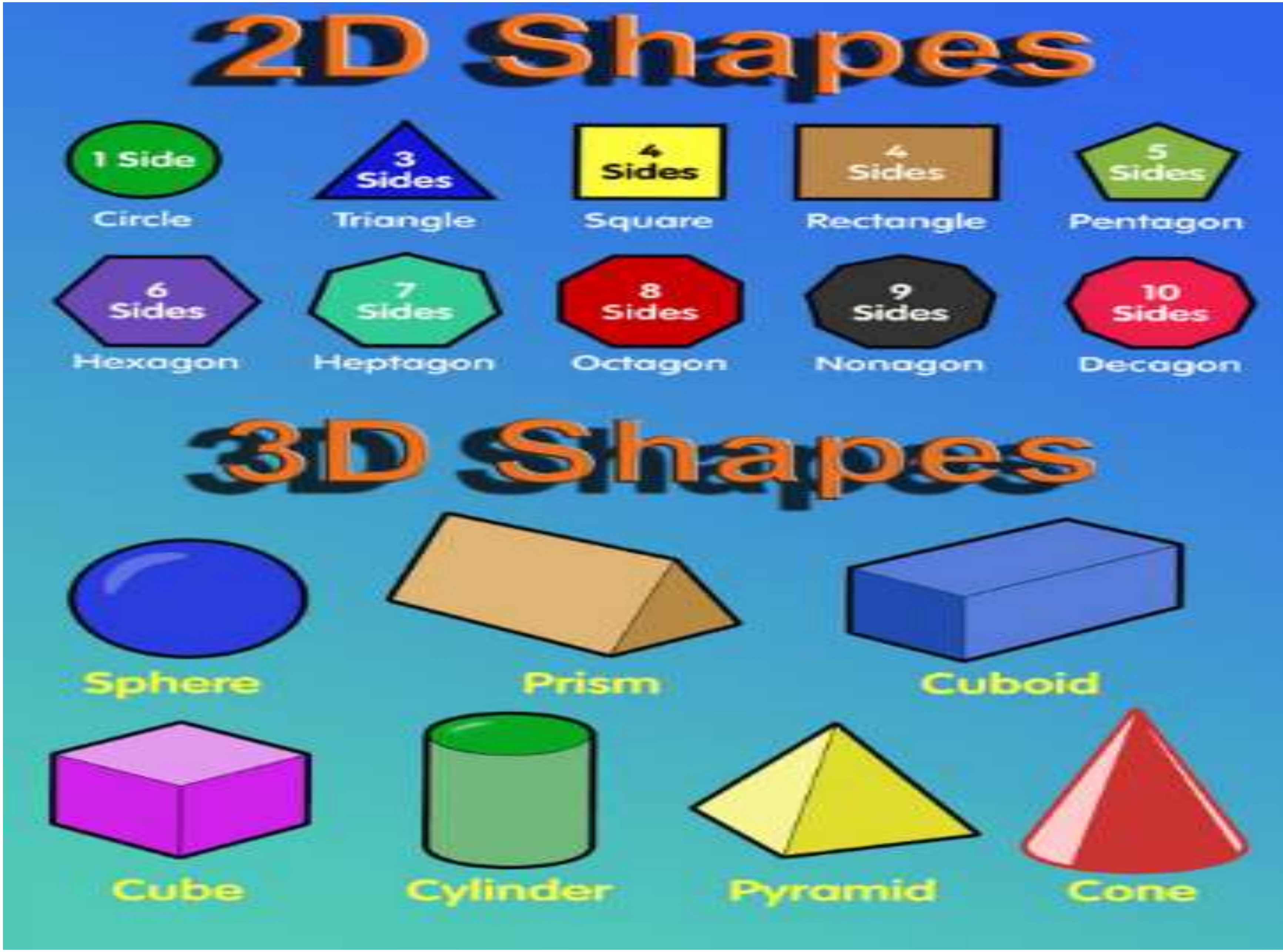
COMPUTER GRAPHICS

2D TRANSFORMATIONS


2D Shapes
1 Side
Circle
3 Sides
Triangle
4 Sides
Square
4 Sides
Rectangle
5 Sides
Pentagon
6 Sides
Hexagon
7 Sides
Heptagon
8 Sides
Octagon
9 Sides
Nonagon
10 Sides
Decagon
3D Shapes
Sphere
Prism
Cuboid
Cube
Cylinder
Pyramid
Cone

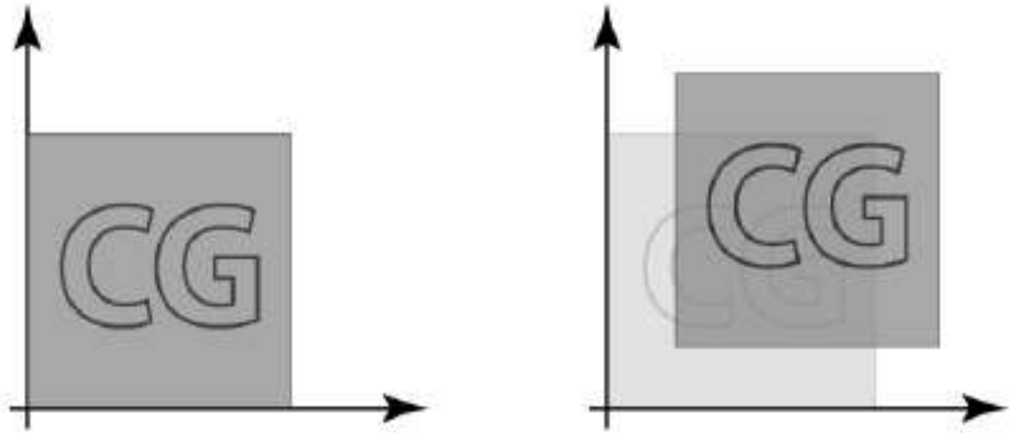
2D Transformations

"Transformations are the operations applied to geometrical description of an object to change its position, orientation, or size are called geometric transformations".

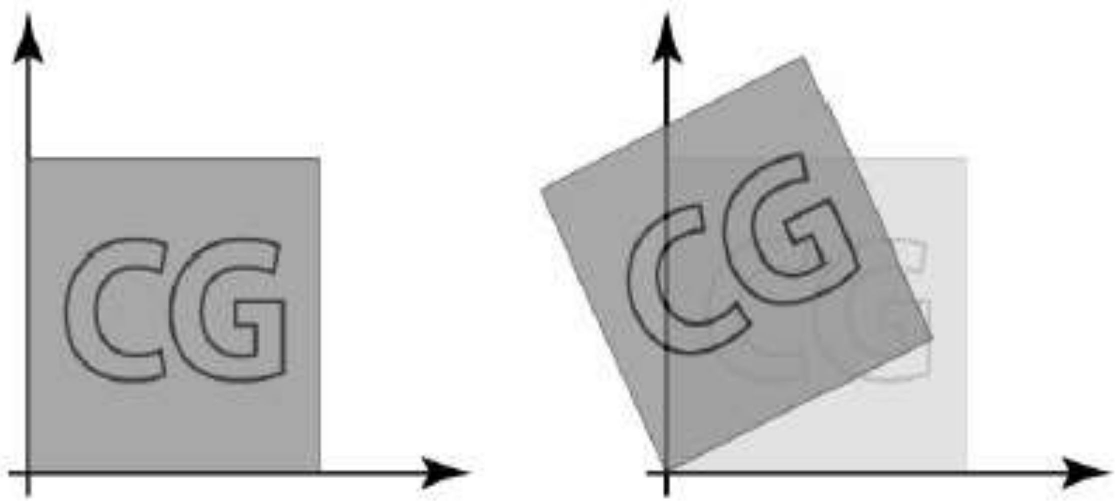
Why Transformations ?

"Transformations are needed to manipulate the initially created object and to display the modified object without having to redraw it."

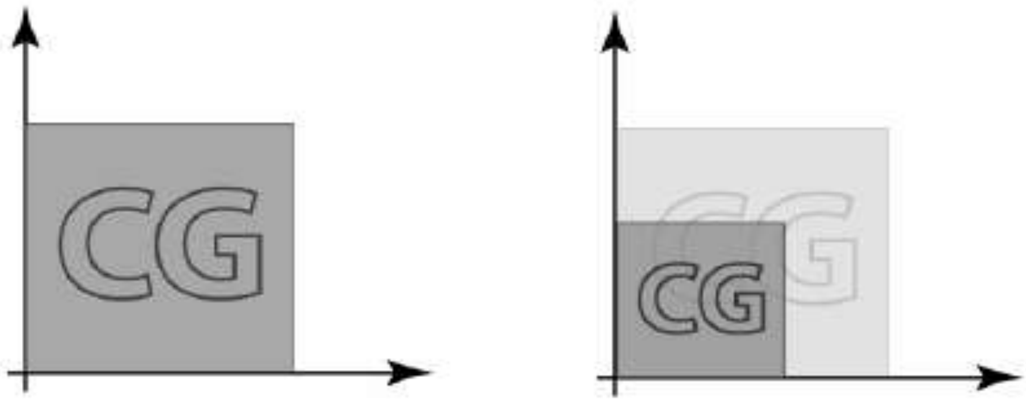
- Translation



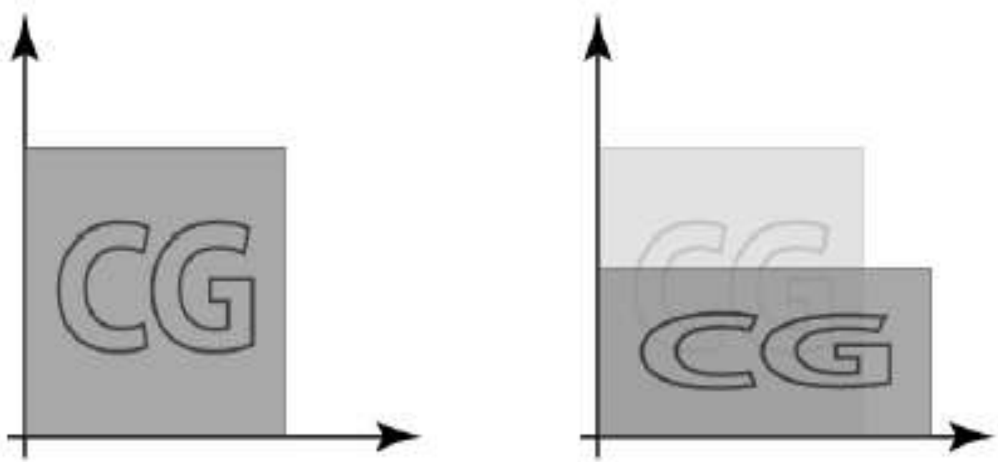
- Rotation



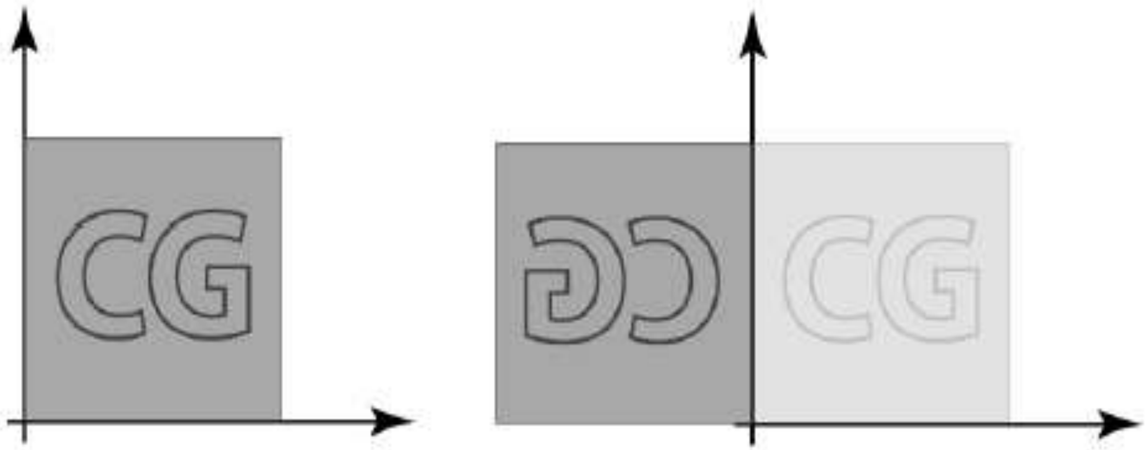
- Scaling
- Uniform Scaling



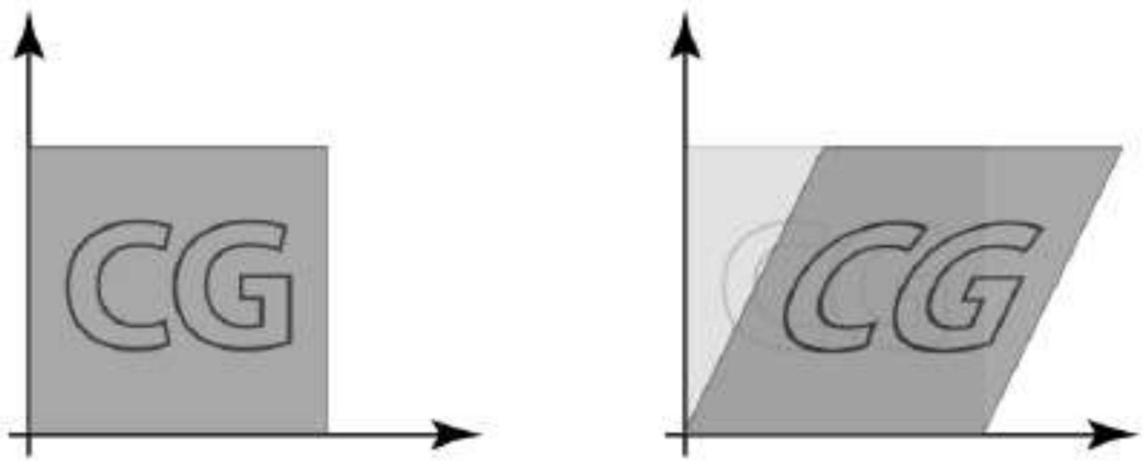
- Un-uniform Scaling



- Reflection



- Shear



Translation

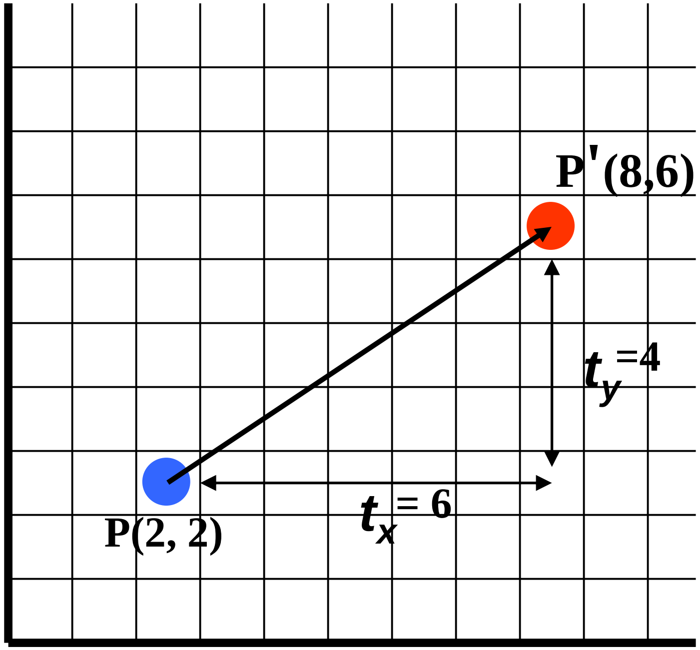
- A translation moves all points in an object along the same straight-line path to new positions.
- The path is represented by a vector, called the translation or shift vector.
- We can write the components:

$$p'_x = p_x + t_x$$
$$p'_y = p_y + t_y$$

- or in matrix form:

$$P' = P + T$$

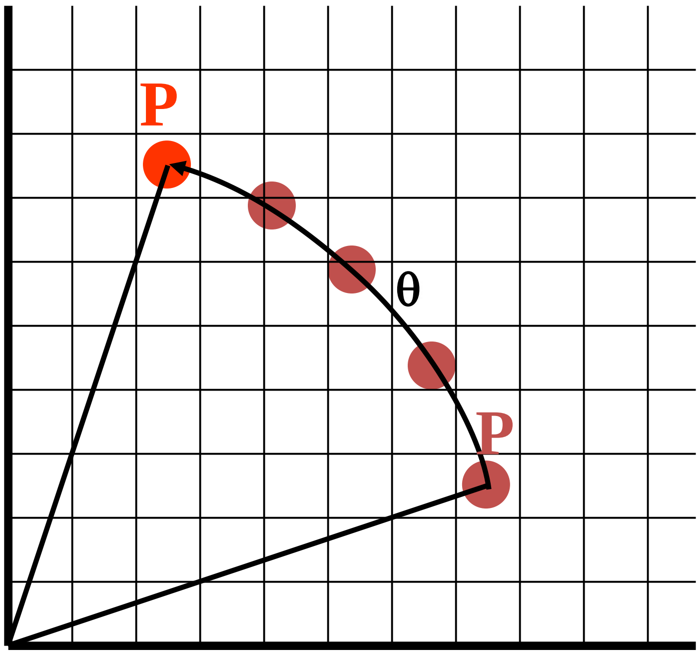
$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} t_x \\ t_y \end{pmatrix}$$



Rotation

- A rotation repositions all points in an object along a circular path in the plane centered at the pivot point.

- First, we'll assume the pivot is at the origin.



Rotation

- Review Trigonometry

=> $\cos \phi = \mathbf{x/r}$, $\sin \phi = \mathbf{y/r}$

- $x = r.\cos \phi$, $y = r.\sin \phi$

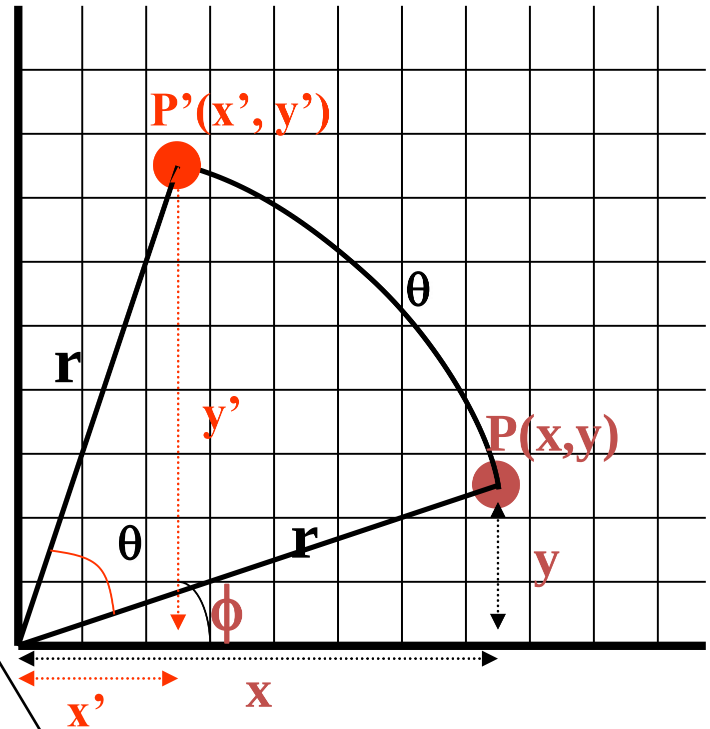
=> $\cos(\phi + \theta) = x'/r$

- $x' = r.\cos(\phi + \theta)$
- $x' = r.\cos\phi\cos\theta - r.\sin\phi\sin\theta$
- $x' = x.\cos\theta - y.\sin\theta$

=> $\sin(\phi + \theta) = y'/r$

$y' = r.\sin(\phi + \theta)$

- $y' = r.\cos\phi\sin\theta + r.\sin\phi\cos\theta$
- $y' = x.\sin\theta + y.\cos\theta$



Identity of Trigonometry

Rotation

- We can write the components:

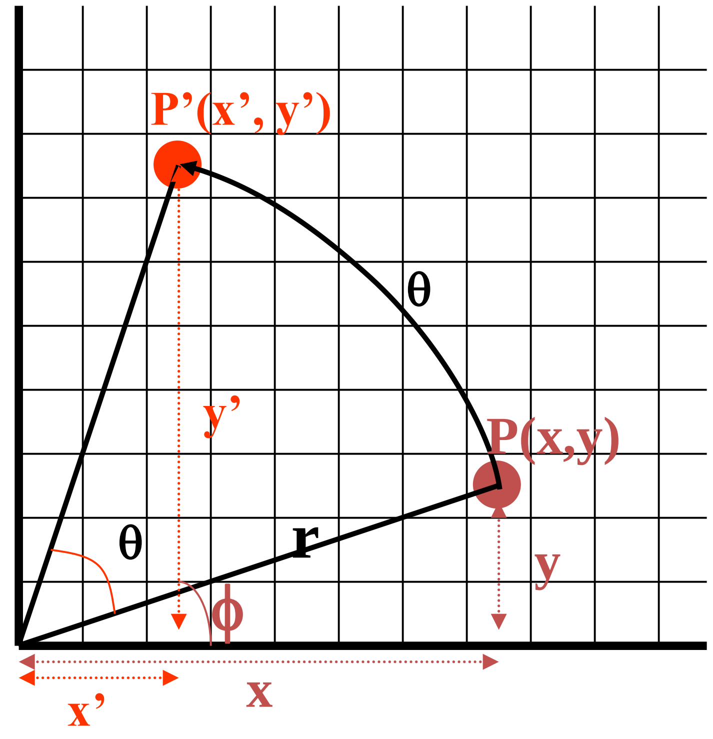
$$p'_x = p_x \cos\theta - p_y \sin\theta$$
$$p'_y = p_x \sin\theta + p_y \cos\theta$$

- or in matrix form:

P' = R • P

- θ can be clockwise (-ve) or counterclockwise (+ve as our example).
- Rotation matrix

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



Scaling

- **Scaling changes the size of an object and involves two scale factors, S_x and S_y for the x- and y- coordinates respectively.**
- **Scales are about the origin.**
- **We can write the components:**

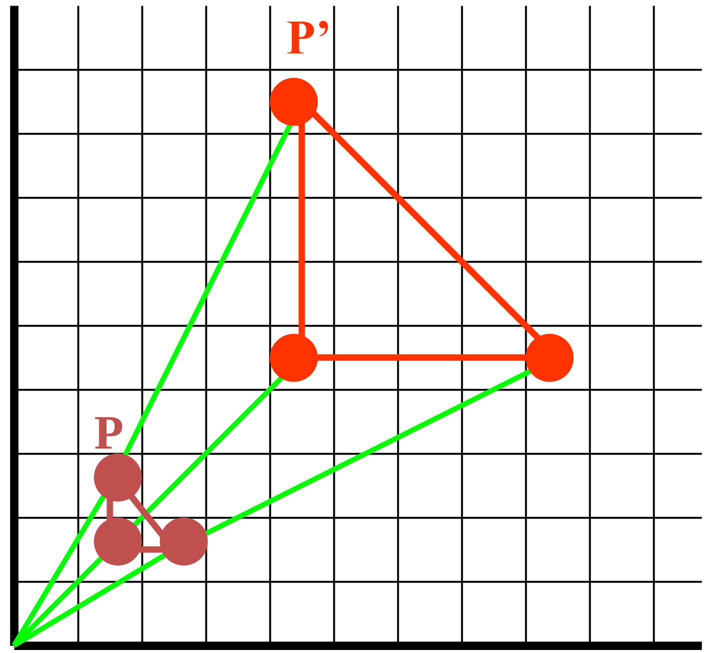
$$p'_x = s_x \bullet p_x$$
$$p'_y = s_y \bullet p_y$$

or in matrix form:

$$\mathbf{P' = S \bullet P}$$

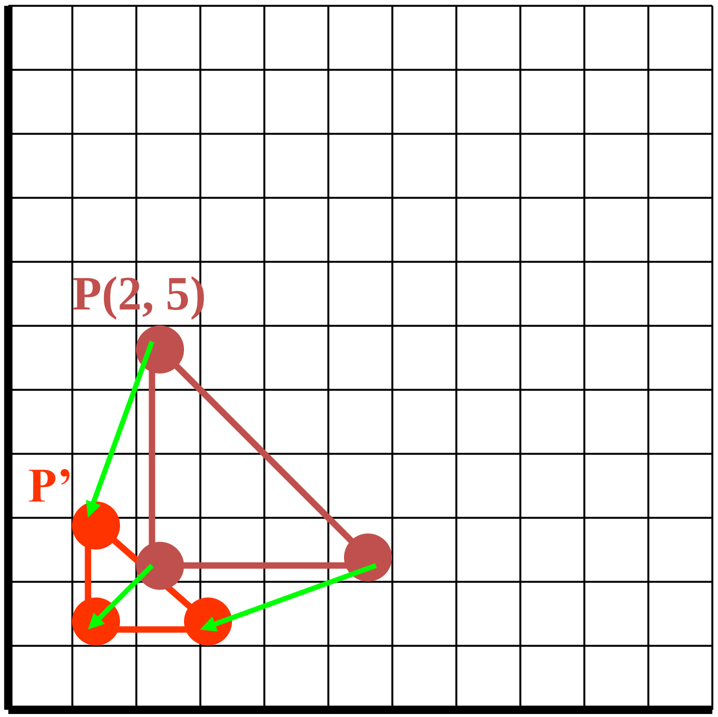
Scale matrix as:

$$S = \begin{bmatrix} s_x & 0 \\ 0 & s_y \end{bmatrix}$$



Scaling

- If the scale factors are in between 0 and 1:----
- ➔ the points will be moved closer to the origin
- ➔ the object will be smaller.

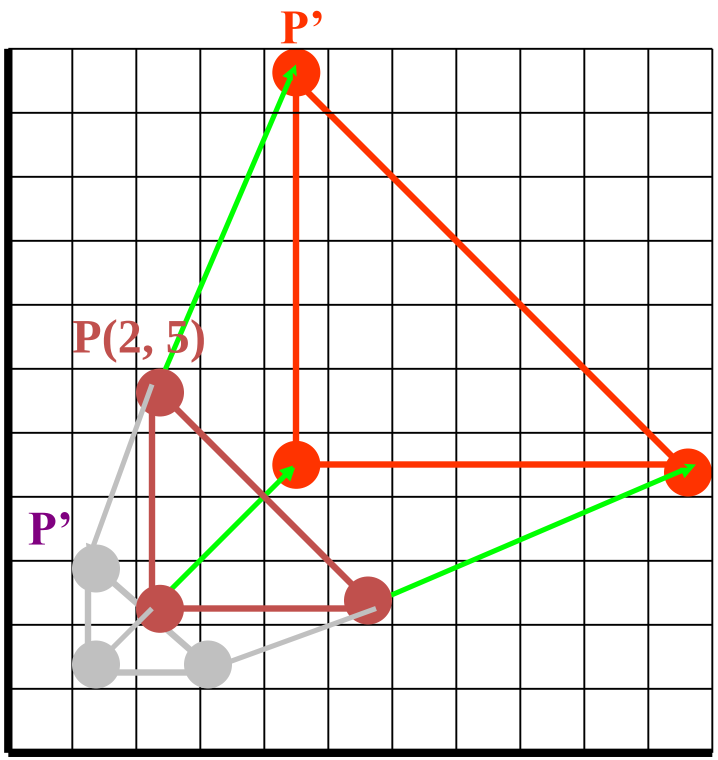


- Example :
  - P(2, 5), $S_x = 0.5$, $S_y = 0.5$

Scaling

- If the scale factors are in between 0 and 1 ➔ the points will be moved closer to the origin ➔ the object will be smaller.

- Example :
  - P(2, 5), $S_x = 0.5$, $S_y = 0.5$

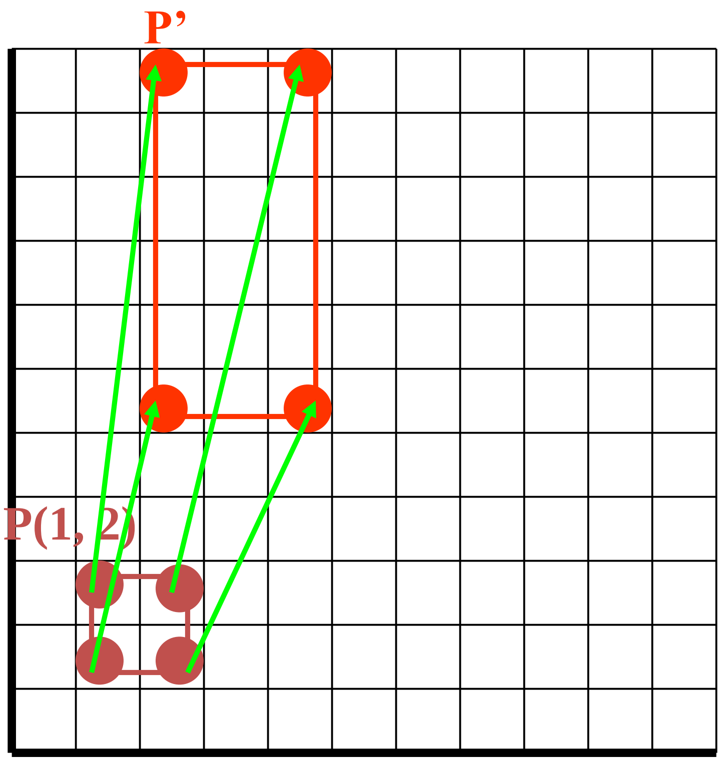


- If the scale factors are larger than 1 ➔ the points will be moved away from the origin ➔ the object will be larger.

- Example :
  - P(2, 5), $S_x = 2$, $S_y = 2$

Scaling

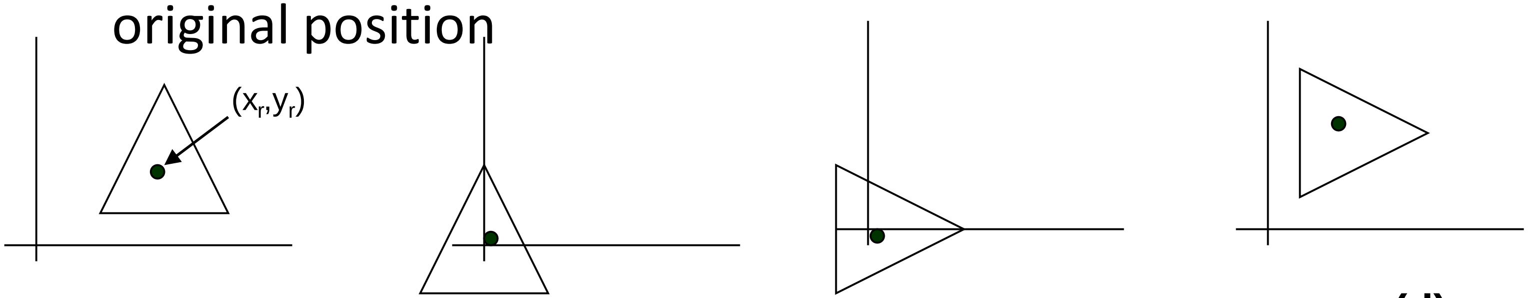
- If the scale factors are the same, $S_x = S_y$ ➔ uniform scaling
- Only change in size (as previous example)



- If $S_x \neq S_y$ ➔ differential scaling.
- Change in size and shape
- Example : square → rectangle
    - P(1, 3), $S_x = 2$, $S_y = 5$

General pivot point rotation

- Translate the object so that pivot-position is moved to the coordinate origin
- Rotate the object about the coordinate origin
- Translate the object so that the pivot point is returned to its original position



(a) **Original Position of Object and pivot point**

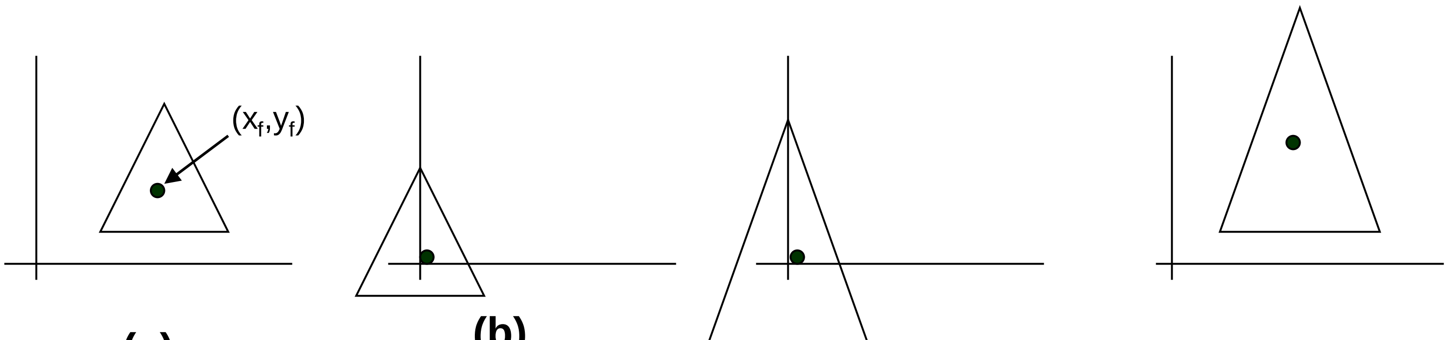
(b) **Translation of object so that pivot point (x_r,y_r) is at origin**

(c) **Rotation was about origin**

(d) **Translation of the object so that the pivot point is returned to position (x_r,y_r)**

General fixed point scaling

- Translate object so that the fixed point coincides with the coordinate origin
- Scale the object with respect to the coordinate origin
- Use the inverse translation of step 1 to return the object to its original position



(a) **Original Position of Object and Fixed point**

(b) **Translation of object so that fixed point (x_f,y_f) is at origin**

(c) **scaling was about origin**

(d) **Translation of the object so that the Fixed point is returned to position (x_f,y_f)**

Composite Transformations

(A) Translations

If two successive translation vectors (t_{x1},t_{y1}) and (t_{x2},t_{y2}) are applied to a coordinate position P, the final transformed location P' is calculated as: -

$$P' = T(t_{x2},t_{y2}) \,.\, \{T(t_{x1},t_{y1})\,.P\}$$
$$= \{T(t_{x2},t_{y2}) \,.\, T(t_{x1},t_{y1})\}\,.P$$

Where P and P' are represented as homogeneous-coordinate column vectors. We can verify this result by calculating the matrix product for the two associative groupings. Also, the composite transformation matrix for this sequence of transformations is: -

$$\begin{vmatrix} 1 & 0 & t_{x2} \\ 0 & 1 & t_{y2} \\ 0 & 0 & 1 \end{vmatrix} \cdot \begin{vmatrix} 1 & 0 & t_{x1} \\ 0 & 1 & t_{y1} \\ 0 & 0 & 1 \end{vmatrix} = \begin{vmatrix} 1 & 0 & t_{x1}+t_{x2} \\ 0 & 1 & t_{y1}+t_{y2} \\ 0 & 0 & 1 \end{vmatrix}$$

Or, $\mathbf{T(t_{x2},t_{y2}) \,.\, T(t_{x1},t_{y1}) = T(t_{x1}+t_{x2},\, t_{y1}+t_{y2})}$

Which demonstrate that two successive translations are additive.

(B) Rotations

Two successive rotations applied to point P produce the transformed position: -

$$P' = R(\Theta_2) \,.\, \{R(\Theta_1\ ) \,.\, P\}$$

$$= \{R(\Theta_2) \,.\, R(\Theta_1)\} \,.\, P$$

By multiplication the two rotation matrices, we can verify that two successive rotations are additive:

$$\mathbf{R(\Theta_2) \,.\, R(\Theta_1) = R\ (\Theta_1 + \Theta_2)}$$

So that the final rotated coordinates can be calculated with the composite rotation matrix as: -

$$\mathbf{P' = R(\Theta_1 + \Theta_2) \,.\, P}$$

(C) Scaling

Concatenating transformation matrices for two successive scaling operations produces the following composite scaling matrix: -

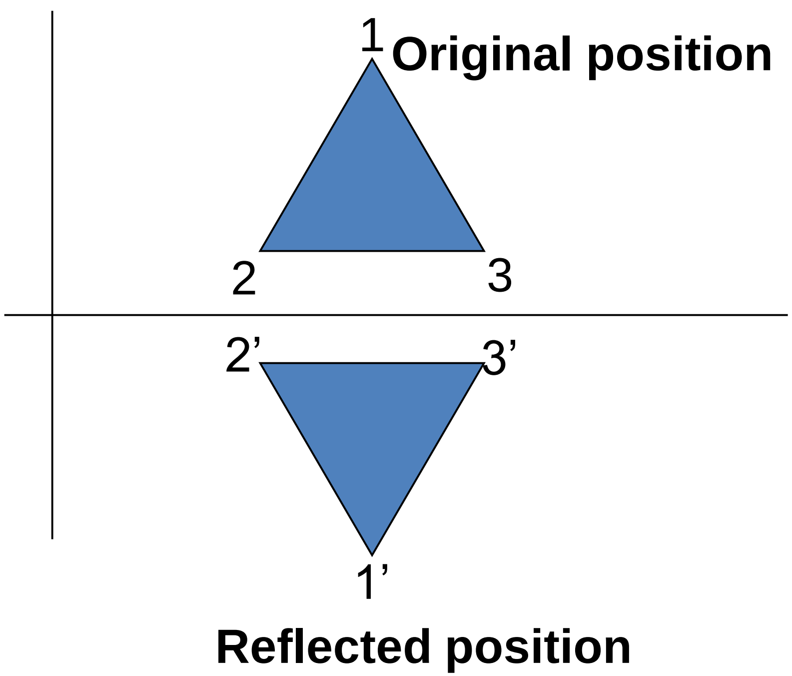
$$\begin{vmatrix} S_{x2} & 0 & 0 \\ 0 & S_{y2} & 0 \\ 0 & 0 & 1 \end{vmatrix} \cdot \begin{vmatrix} S_{x1} & 0 & 0 \\ 0 & S_{y1} & 0 \\ 0 & 0 & 1 \end{vmatrix} = \begin{vmatrix} S_{x1} \cdot S_{x2} & 0 & 0 \\ 0 & S_{y1} \cdot S_{y2} & 0 \\ 0 & 0 & 1 \end{vmatrix}$$

Or, $\mathbf{S(S_{x2}, S_{y2}) \cdot S(S_{x1}, S_{y1}) = S(S_{x1} \cdot S_{x2}, S_{y1} \cdot S_{y2})}$

The resulting matrix in this case indicates that successive scaling operations are multiplicative.

Other transformations

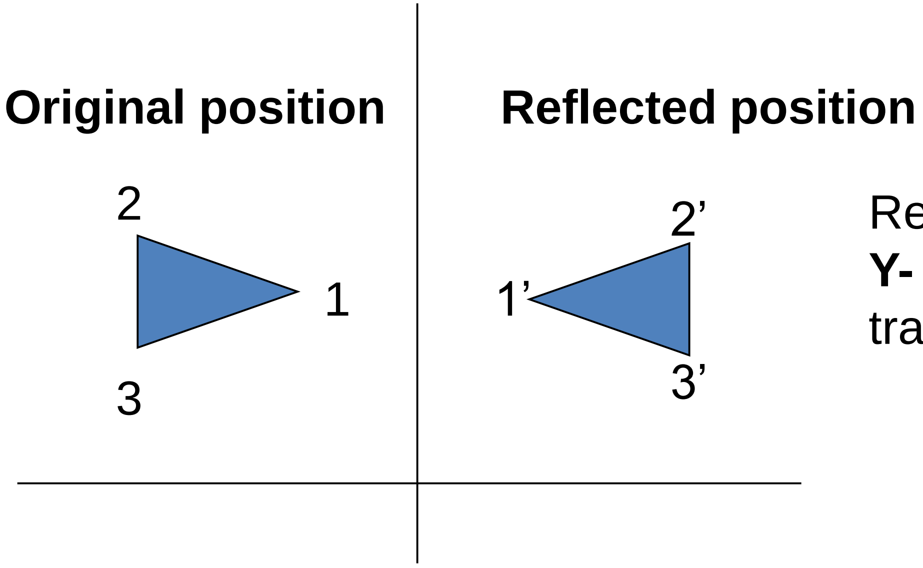
- **Reflection** is a transformation that produces a mirror image of an object. It is obtained by rotating the object by 180 deg about the reflection axis



Reflection about the line y=0, the **X- axis** , is accomplished with the transformation matrix

$$\begin{vmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{vmatrix}$$

Reflection

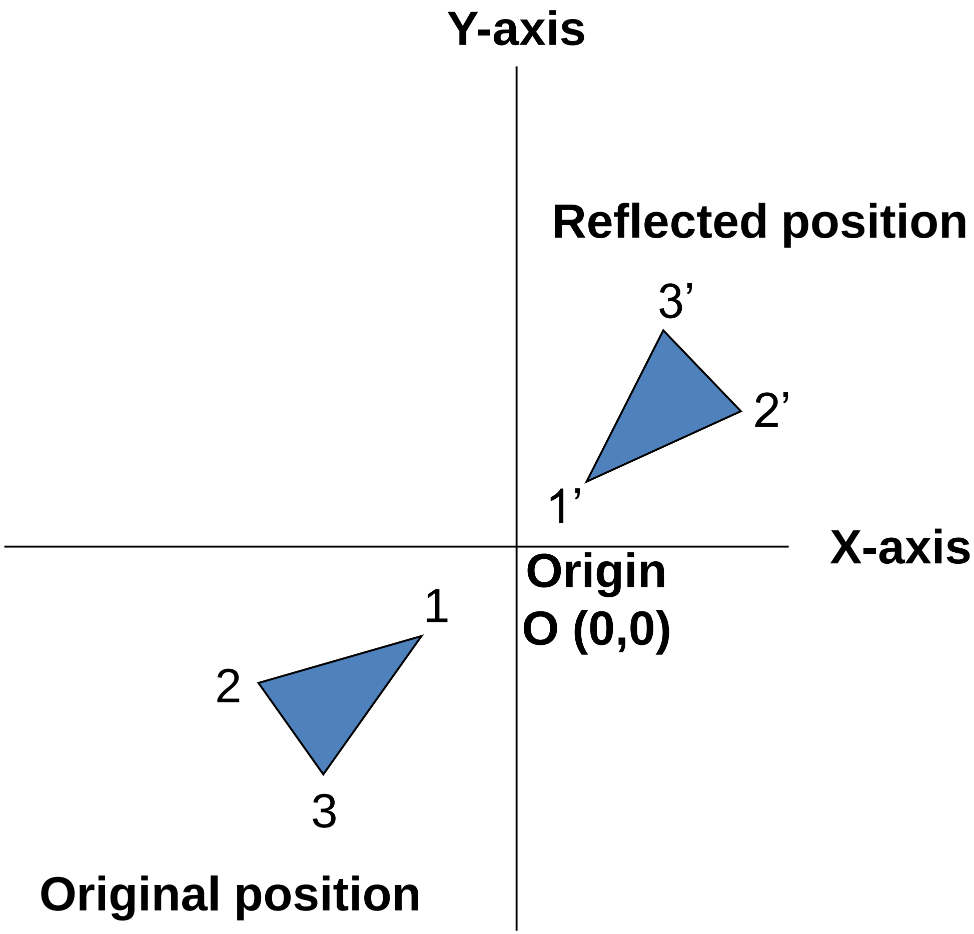


Reflection about the line x=0, the **Y- axis** , is accomplished with the transformation matrix

$$\begin{vmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{vmatrix}$$

Reflection

Reflection of an object relative to an axis perpendicular to the xy plane and passing through the coordinate origin



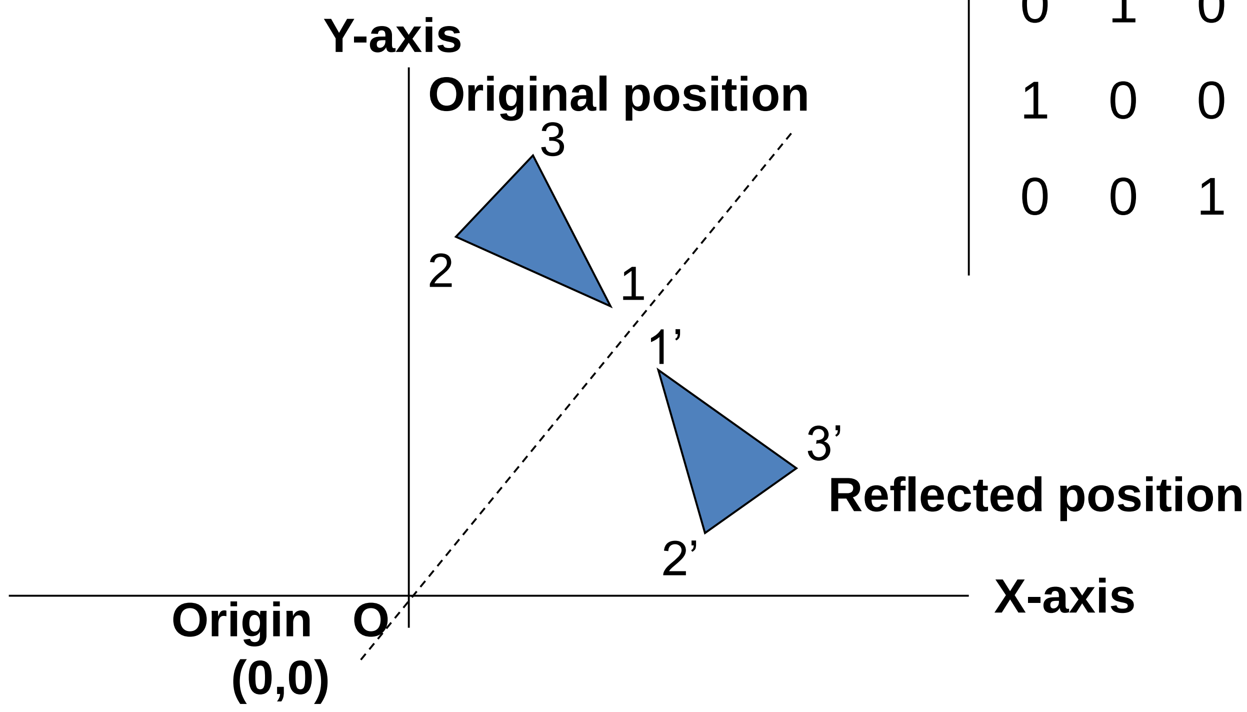
$$\begin{vmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{vmatrix}$$

The above reflection matrix is the rotation matrix with angle=180 degree.

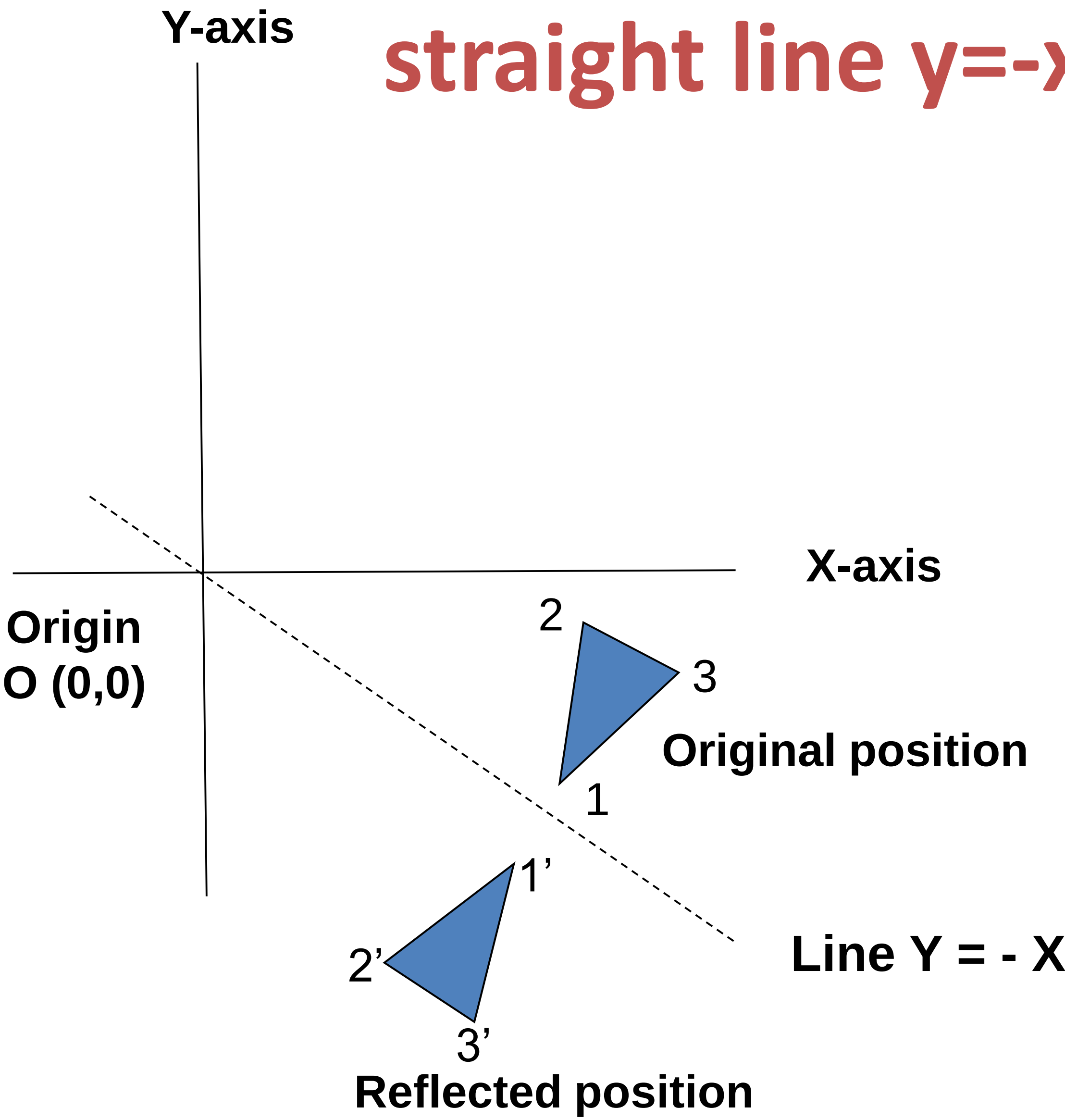
This can be generalized to any reflection point in the xy plane. This reflection is the same as a 180 degree rotation in the xy plane using the reflection point as the pivot point.

Reflection of an object w.r.t the straight line y=x

$$\begin{vmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{vmatrix}$$

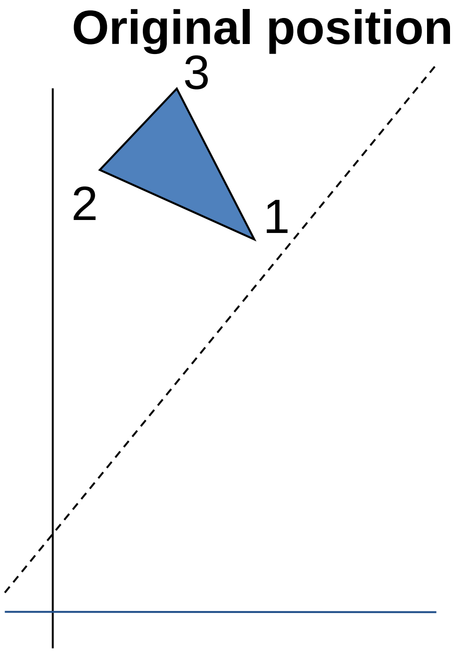


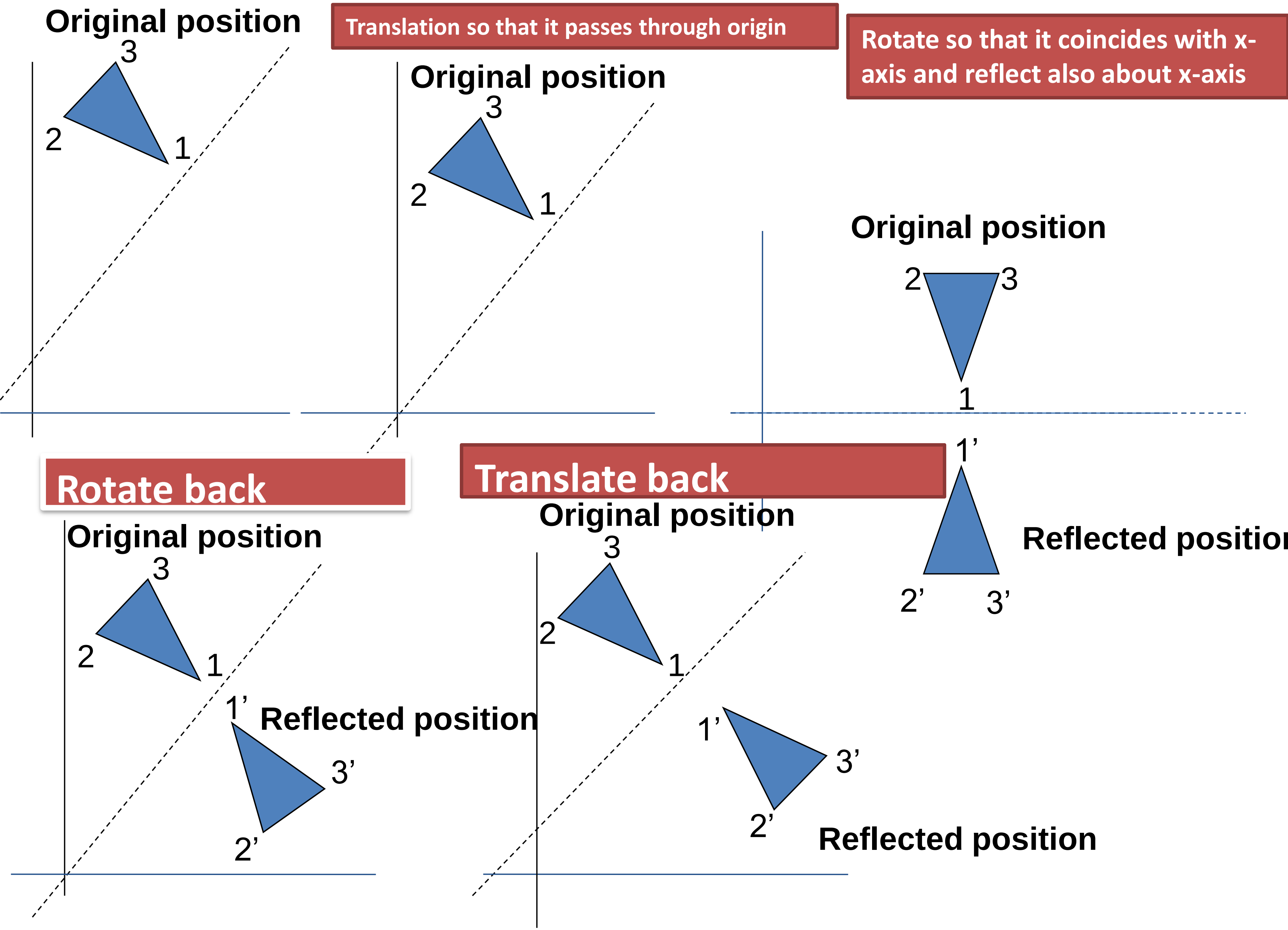
Reflection of an object w.r.t the straight line y=-x



$$\begin{vmatrix} 0 & -1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 1 \end{vmatrix}$$

Reflection of an arbitrary axis y=mx+b



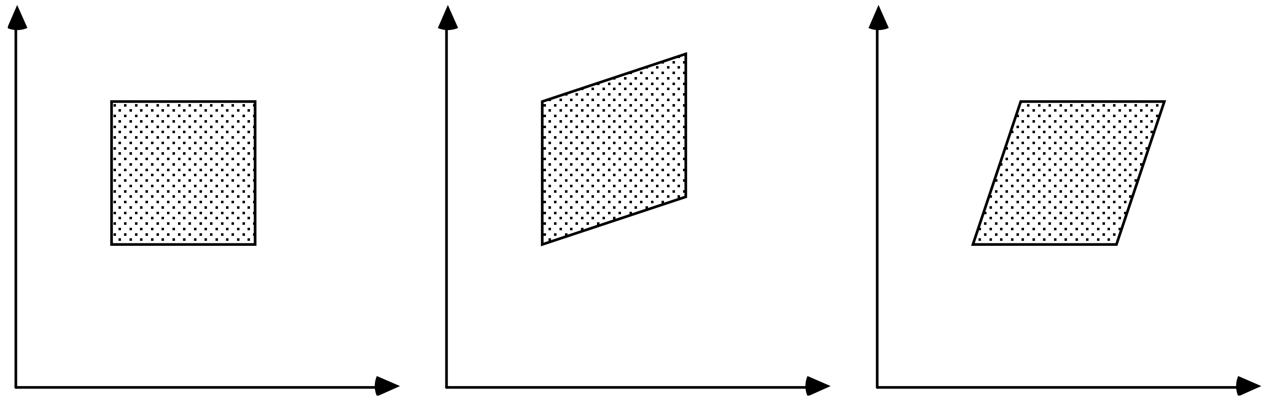


Original position
3
2
1
Translation so that it passes through origin
Original position
3
2
1
Rotate so that it coincides with x-axis and reflect also about x-axis
Original position
2
3
1
1’
Reflected positio
2’
3’
Rotate back
Original position
3
2
1
1’
Reflected position
3’
2’
Translate back
Original position
3
2
1
1’
3’
2’
Reflected position

Shear Transformations

- Shear is a transformation that distorts the shape of an object such that the transformed shape appears as if the object were composed of internal layers that had been caused to slide over each other
- Two common shearing transformations are those that shift coordinate x values and those that shift y values

Shears

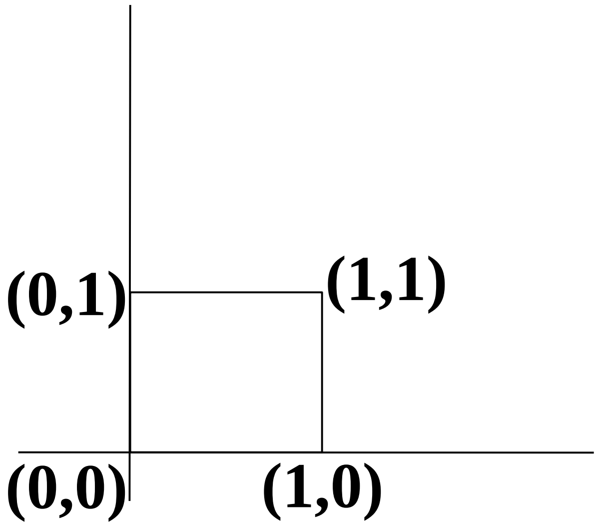


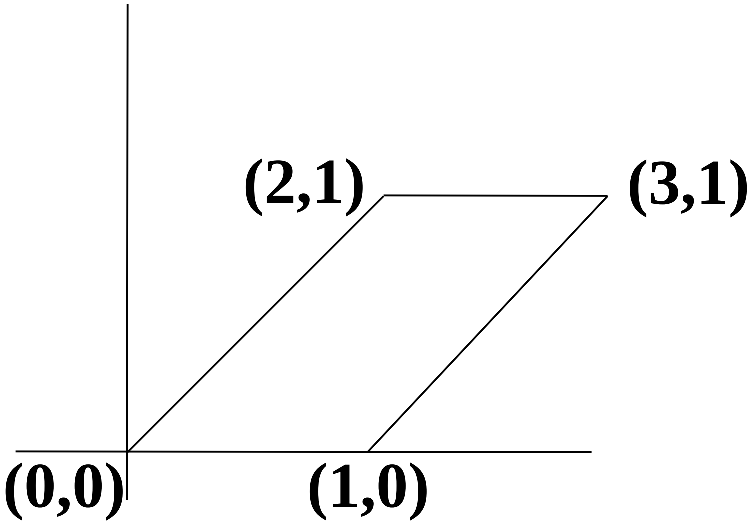
Original Data **y Shear** **x Shear**

$$\begin{vmatrix} 1 & 0 & 0 \\ sh_y & 1 & 0 \\ 0 & 0 & 1 \end{vmatrix} \qquad \begin{vmatrix} 1 & sh_x & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{vmatrix}$$

An X- direction Shear

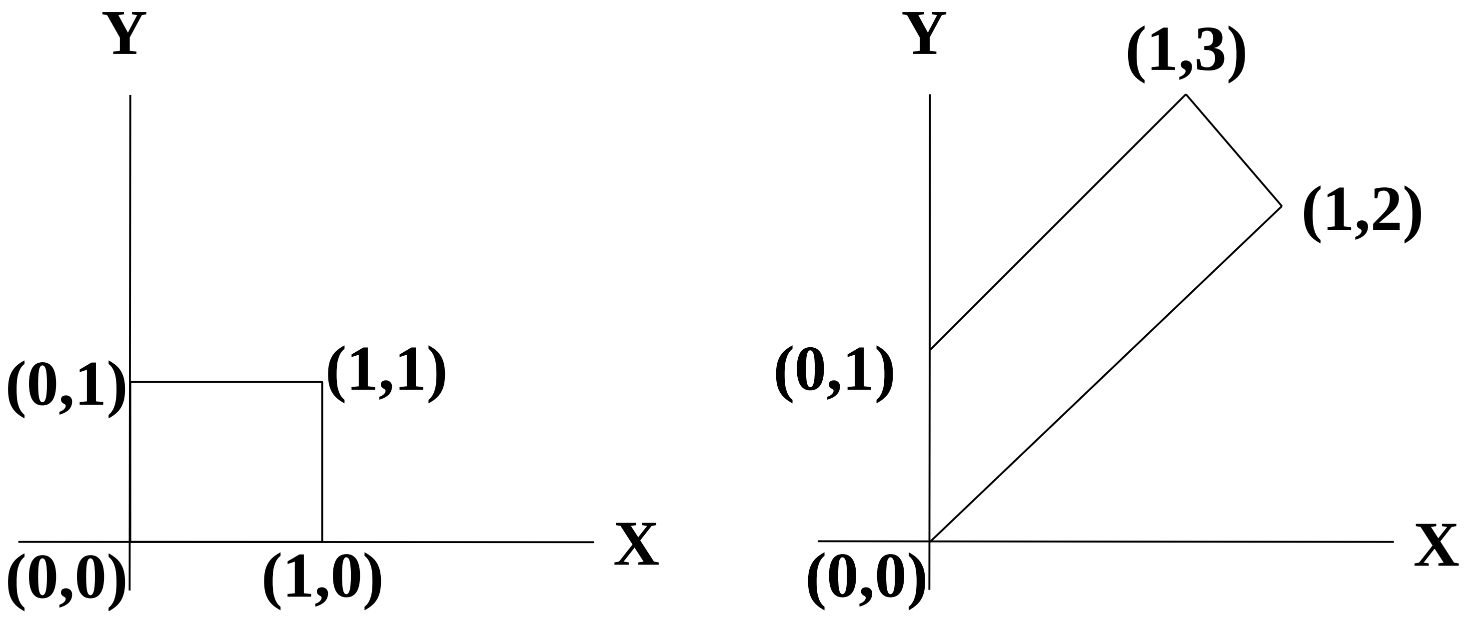
For example, Sh_x=2





An Y- direction Shear

For example, $Sh_y=2$

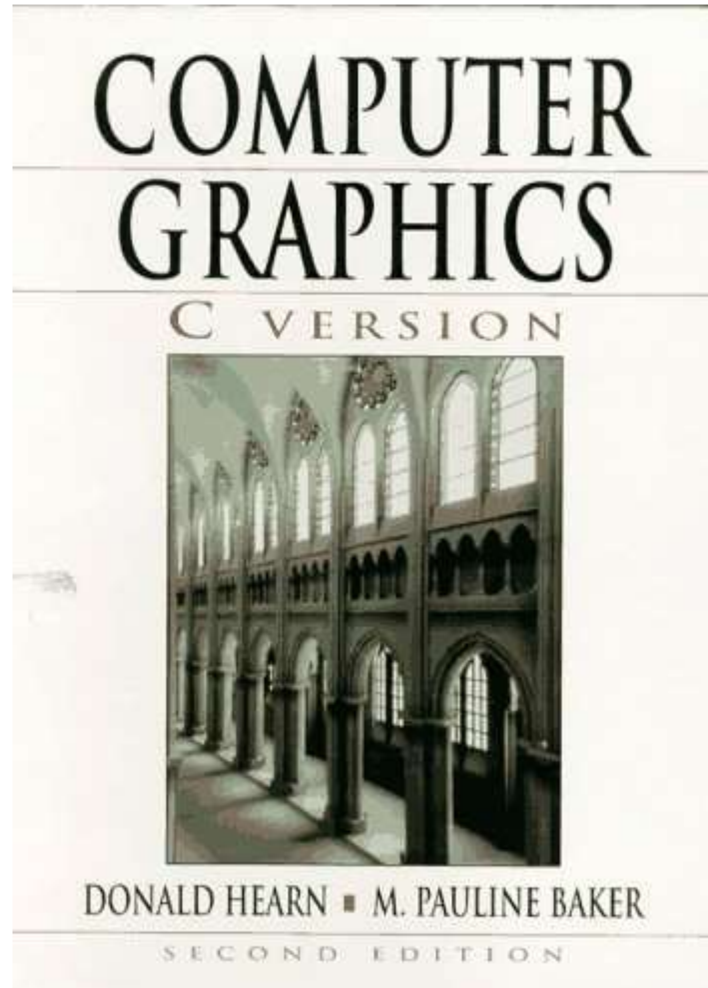


CONCLUSION

To manipulate the initially created object and to display the modified object without having to redraw it, we use Transformations.

Textbook

- Computer Graphics
  C Version
  - D. Hearn and M. P. Baker
  - 2nd Edition
  - PRENTICE HALL



QUERY ?

THANK YOU ALL