2D Transformation and Matrices

By
Poonam Saini
Dept. of Computer Science & Engineering
Sir Padampat Singhania University
Udaipur

Matrices

$$A = \begin{bmatrix} a_{1,1} & a_{1,2} & a_{1,3} \\ a_{2,1} & a_{2,2} & a_{2,3} \\ a_{3,1} & a_{3,2} & a_{3,3} \end{bmatrix}$$

- A matrix is a rectangular array of numbers.
- A general matrix will be represented by an upper-case italicised letter.
- The element on the *i*th row and *j*th column is denoted by $a_{i,j}$. Note that we start indexing at 1, whereas *C* indexes arrays from 0.

Matrices – Addition

- Given two matrices A and B if we want to add B to A (that is form $A+B$) then if A is $(n \times m)$, B must be $(n \times m)$, Otherwise, $A+B$ is not defined.

- The addition produces a result, $C = A+B$, with elements: $C_{i,j} = A_{i,j} + B_{i,j}$

$$\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} + \begin{bmatrix} 5 & 6 \\ 7 & 8 \end{bmatrix} = \begin{bmatrix} 1+5 & 2+6 \\ 3+7 & 4+8 \end{bmatrix} = \begin{bmatrix} 6 & 8 \\ 10 & 12 \end{bmatrix}$$

Matrices – Multiplication

- Given two matrices *A* and *B* if we want to multiply *B* by *A* (that is form *AB*) then if *A* is ($n \times m$), *B* must be ($m \times p$), i.e., the number of columns in *A* must be equal to the number of rows in *B*. Otherwise, *AB* is not defined.
- The multiplication produces a result, *C* = *AB*, with elements:

$$C_{i,j} = \sum_{k=1}^{m} a_{ik} b_{kj}$$

(Basically we multiply the first row of *A* with the first column of *B* and put this in the $c_{1,1}$ element of *C*. And so on…).

Matrices – Multiplication (Examples)

$$2\times6+6\times3+7\times2=44$$

$$\begin{bmatrix} 2 & 6 & 7 \\ 4 & 5 & 8 \\ 9 & 2 & 3 \end{bmatrix} \times \begin{bmatrix} 6 & 8 \\ 3 & 3 \\ 2 & 6 \end{bmatrix} = \begin{bmatrix} 44 & 76 \\ 55 & 95 \\ 66 & 96 \end{bmatrix}$$

$$\begin{bmatrix} 2 & 6 \\ 4 & 5 \end{bmatrix} \times \begin{bmatrix} 6 & 8 \\ 3 & 3 \\ 2 & 6 \end{bmatrix}$$

Undefined!

2x2 x 3x2   2!=3

2x2 x 2x4 x 4x4 is allowed. Result is 2x4 matrix

Matrices -- Basics

- Unlike scalar multiplication, $AB \neq BA$
- Matrix multiplication distributes over addition:

$$A(B+C) = AB + AC$$

- Identity matrix for multiplication is defined as I.
- The transpose of a matrix, A, is either denoted A^T or A' is obtained by swapping the rows and columns of A:

$$A = \begin{bmatrix} a_{1,1} & a_{1,2} & a_{1,3} \\ a_{2,1} & a_{2,2} & a_{2,3} \end{bmatrix} \Rightarrow A' = \begin{bmatrix} a_{1,1} & a_{2,1} \\ a_{1,2} & a_{2,2} \\ a_{1,3} & a_{2,3} \end{bmatrix}$$

2D Geometrical Transformations

Transformation

- The manipulation of objects in space is referred to as transformation.
- It is useful for placing the independently defined objects into a common scene in a master coordinate system.
- Types of object Transformations:
  - → Geometric transformation
  - → Coordinate transformation

Transformation

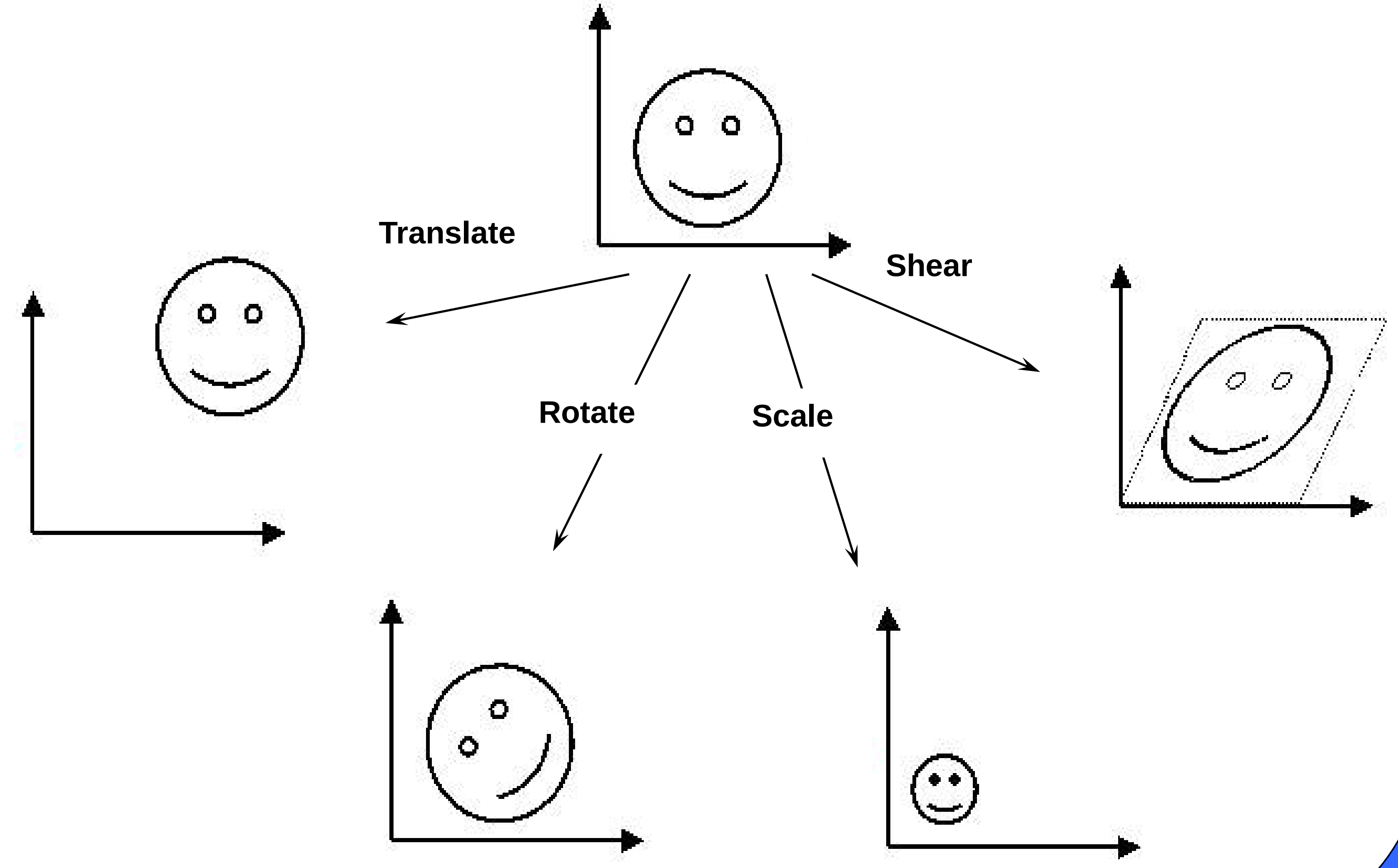
- **Geometric Transformation**

In this type of transformation object is transformed relative to a stationary coordinate system or background.

- **Coordinate transformation**

In this type of transformation, we keep the object stationary while the coordinate system is transformed relative to the object.

2D Geometrical Transformations

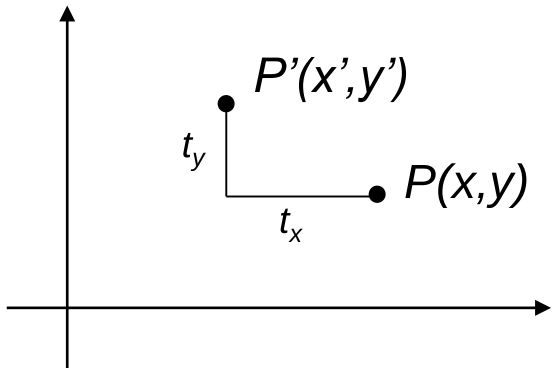


Translation

- A translation is applied to an object by repositioning it along a straight line path from one coordinate location to another.

Translate Points

We can translate points in the (*x, y*) plane to new positions by adding translation amounts to the coordinates of the points. For each point *P*(*x, y*) to be moved by t_x units parallel to the *x* axis and by t_y units parallel to the *y* axis, to the new point *P'*(*x'*, *y'*). The translation has the following form:



$$x' = x + t_x$$

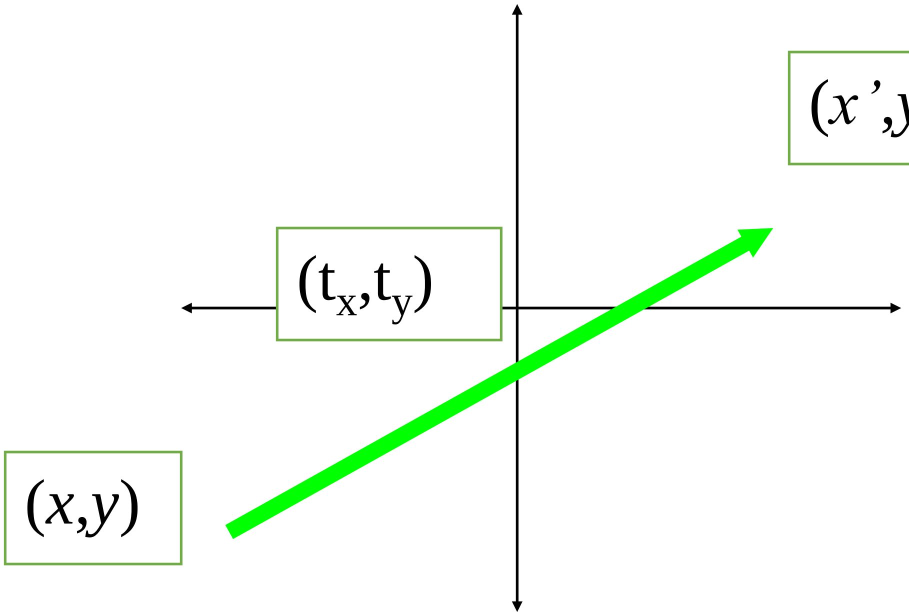
$$y' = y + t_y$$

In matrix format:

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

If we define the translation matrix $T = \begin{bmatrix} t_x \\ t_y \end{bmatrix}$ **, then we have *P' =P + T.***

Translation contd.



- Given:

$$P = (x, y)$$
$$T = (t_x, t_y)$$

$$P = (-3.7, -4.1)$$
$$T = (7.1, 8.2)$$

- We want:

$$x' = x + t_x$$
$$y' = y + t_y$$

$$x' = -3.7 + 7.1$$
$$y' = -4.1 + 8.2$$

- Matrix form:

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \end{bmatrix}$$
$$P' = P + T$$

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} -3.7 \\ -4.1 \end{bmatrix} + \begin{bmatrix} 7.1 \\ 8.2 \end{bmatrix}$$
$$x' = 3.4$$
$$y' = 4.1$$

Translation Exercises

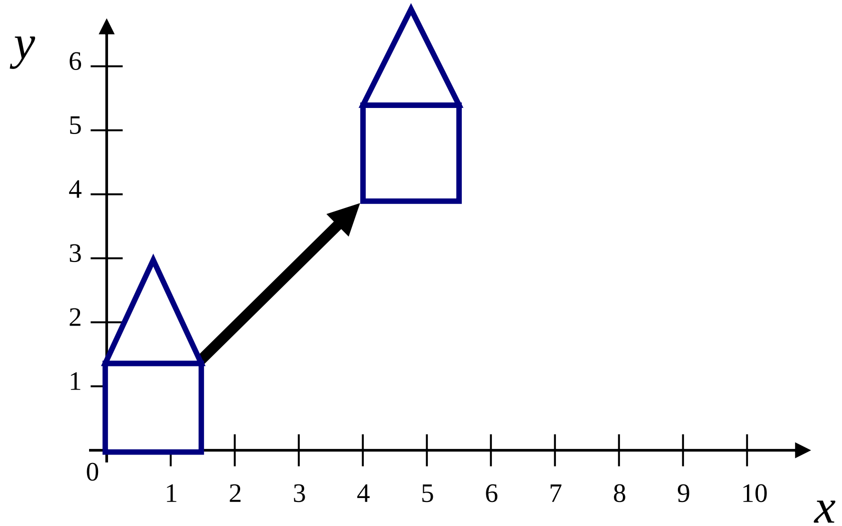
Problem 1: **Translate the point P=(2,4) with the translational value T=(-1,14). Find the coordinate of the point P after translation i.e. P’=(?,?).**

Problem 2: **Translate the point P=(8.6,-1), with the translational value T=(0.4,-0.2). Find the coordinate of the point P after translation i.e. P’=(?,?).**

Problem 3: **Translate the point P=(0,0), with the translational value T=(1,0). Find the coordinate of the point P after translation i.e. P’=(?,?).**

Translation contd. With objects

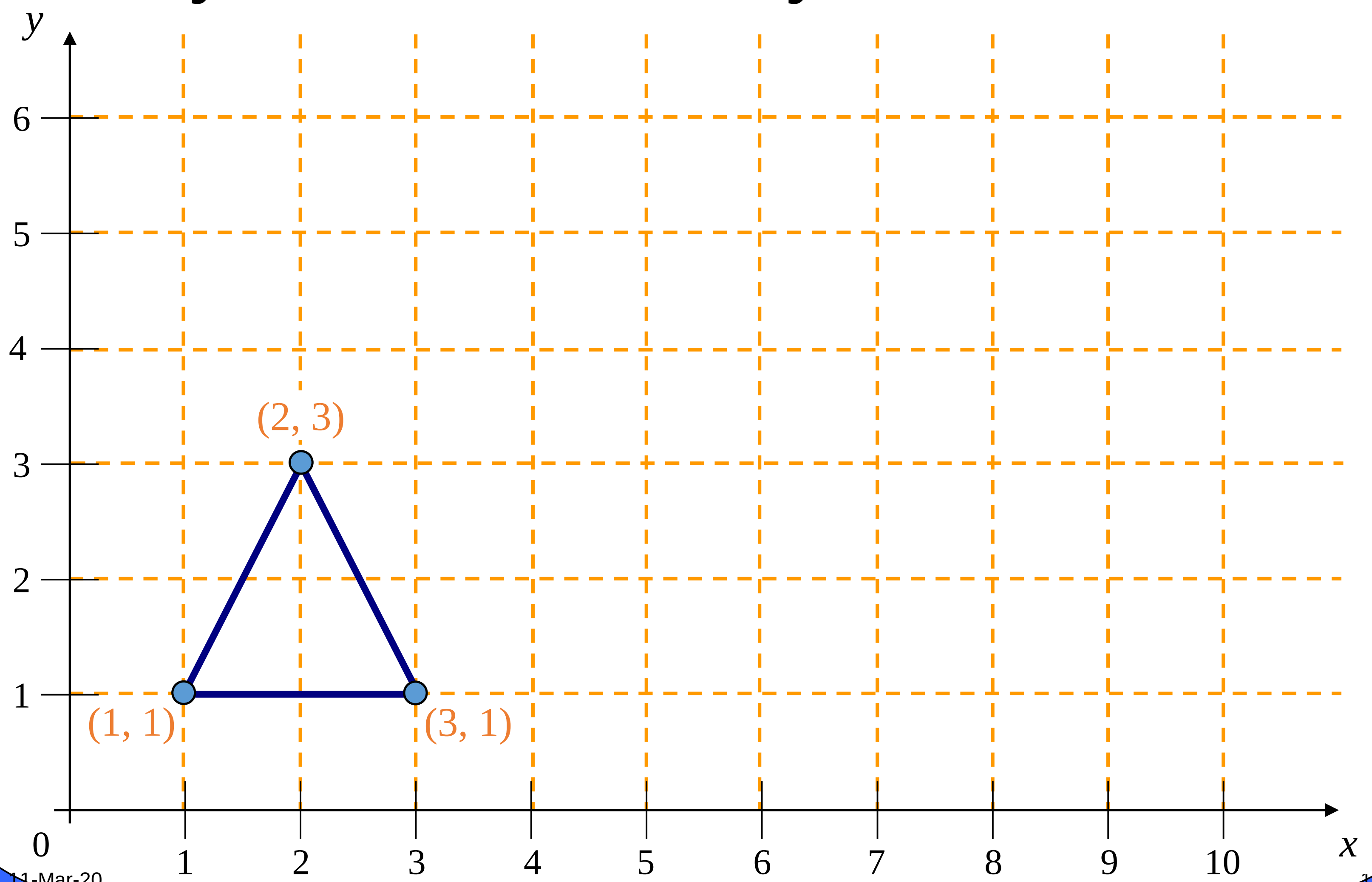
- Simply moves an object from one position to another
- $x_{new} = x_{old} + \mathrm{t_x}$ $y_{new} = y_{old} + \mathrm{t_y}$



Note: House shifts position relative to origin

Problem:

Solve yourself for tx=2,ty=3 units



Scaling

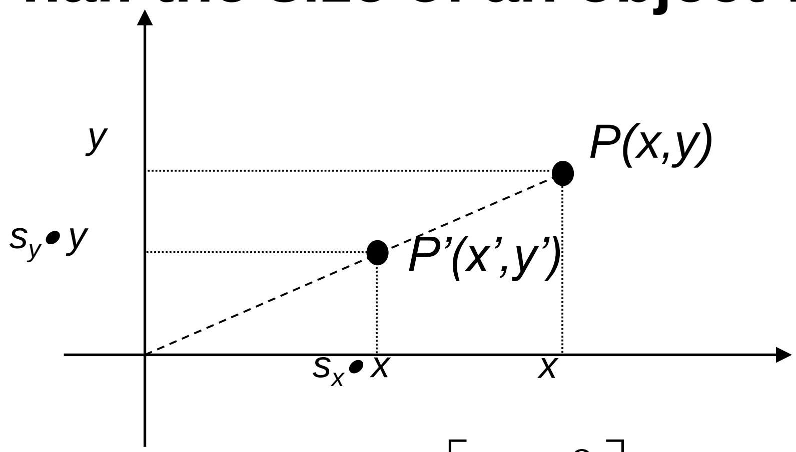
- The process of changing the size of an object such that we can magnify the size or reduce it, is known as scaling.

Scaling Points

Points can be scaled (stretched) by s_x along the x axis and by s_y along the y axis into the new points by the multiplications:

We can specify how much bigger or smaller by means of a "scale factor"

To double the size of an object we use a scale factor of 2, to half the size of an object we use a scale factor of 0.5



$$x' = s_x x$$

$$y' = s_y y$$

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} s_x & 0 \\ 0 & s_y \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

If we define $S = \begin{bmatrix} s_x & 0 \\ 0 & s_y \end{bmatrix}$, then we have $P' = SP$

Scaling contd.

- Any positive numeric values can be assigned to the scaling factors s_x and s_y
- Values less than 1 reduce the size of objects; values greater than 1 produce an enlargement.
- Specifying a value of 1 for both s_x and s_y leaves the size of objects unchanged.
- When both s_x and s_y are assigned the same value, a uniform scaling is produced
- Unequal values for s_x and s_y result in a differential scaling

Scaling w.r.t. a Fixed point

- We can control the location of a scaled object by choosing a position, called the fixed point, that is to remain unchanged after the scaling transformation.
- Coordinates for the fixed point (x_f, y_f) can be chosen as one of the vertices, the object centroid, or any other position .A polygon is then scaled relative to the fixed point by scaling the distance from each vertex to the fixed point. For a vertex with coordinates (x,y) the scaled coordinates (x',y') are calculated as:

$$\boldsymbol{x' = x_f + (x - x_f)S_x}$$
$$\boldsymbol{y' = y_f + (y - y_f)S_y}$$

Scaling w.r.t. Origin

- Given:

$$P = (x, y)$$
$$S = (s_x, s_y)$$

$$P = (1.4, 2.2)$$
$$S = (3, 3)$$

- We want:

$$x' = s_x x$$
$$y' = s_y y$$

$$x' = 3 * 1.4$$
$$y' = 3 * 2.2$$

- Matrix form:

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} s_x & 0 \\ 0 & s_y \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$
$$P' = S \cdot P$$

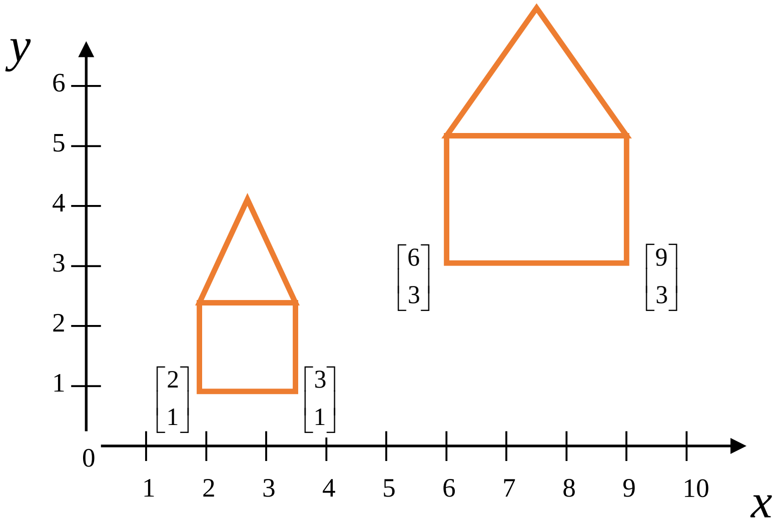
$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} 3 & 0 \\ 0 & 3 \end{bmatrix} \begin{bmatrix} 1.4 \\ 2.2 \end{bmatrix}$$
$$x' = 4.2$$
$$y' = 6.6$$

Scaling contd.

Scalar multiplies all coordinates

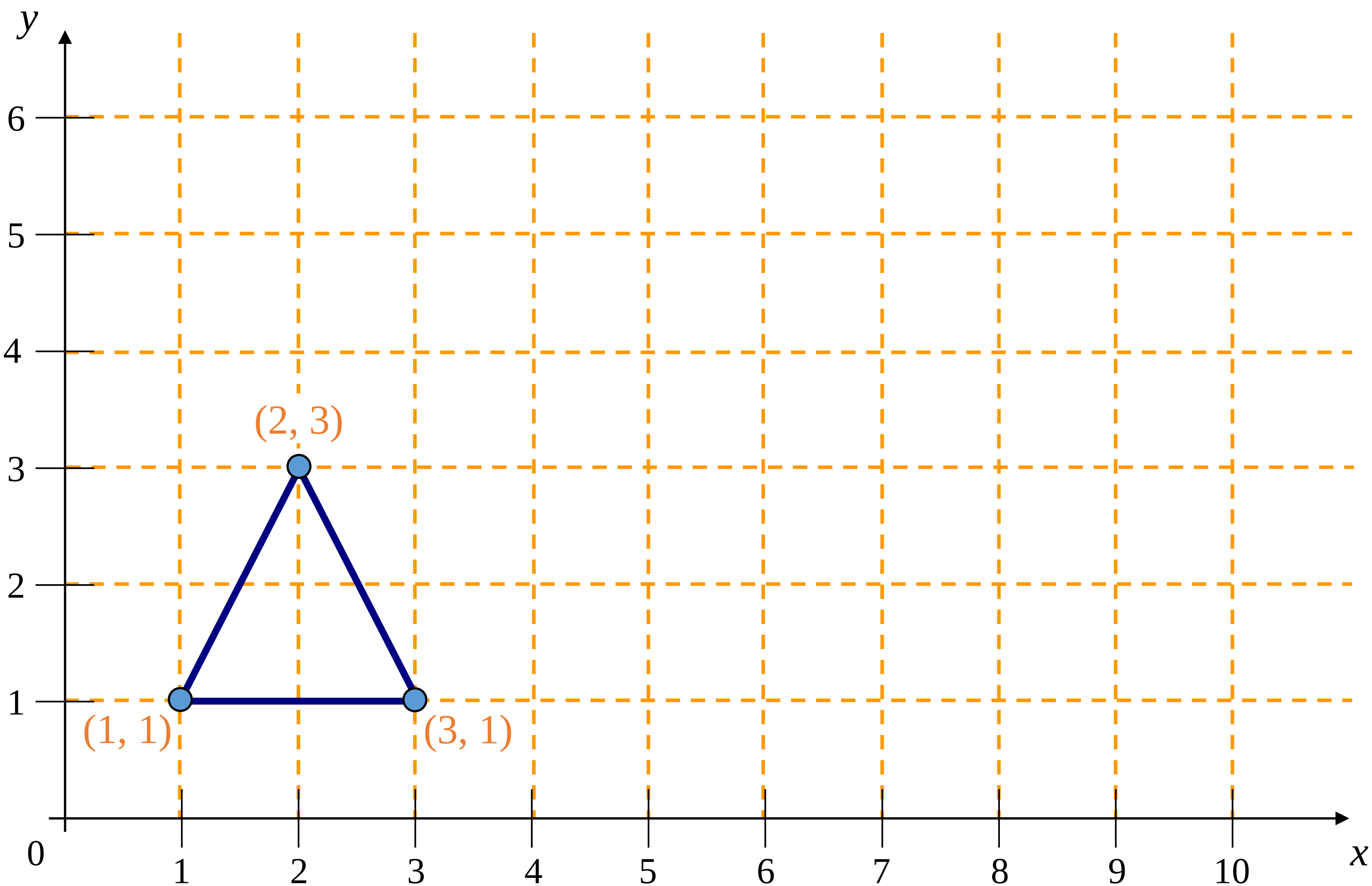
WATCH OUT: Objects grow and move!

$$x_{new} = Sx \times x_{old} \qquad y_{new} = Sy \times y_{old}$$



Note: House shifts position relative to origin

Solve yourself for s_x=4, s_y=2 units



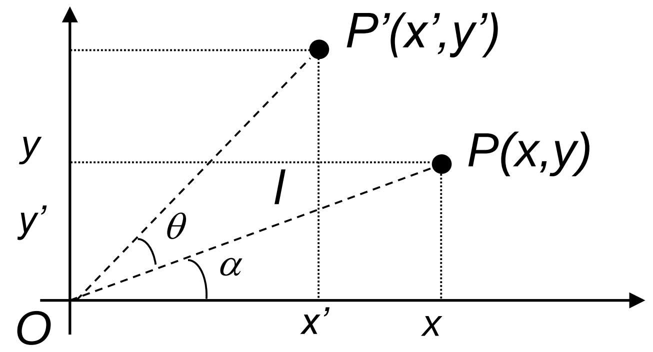
Rotation

- Repositioning of an object along a circular path in xy plane.
- To generate rotation, specify a rotation angle ϴ and position (x_r, y_r) of the rotation point or pivot point about which the object is to be rotated.

Rotate Points

Points can be rotated through an angle θ about the origin:

$$|OP'| = |OP| = l$$



$$x' = |OP'|\cos(\alpha+\theta) = l\cos(\alpha+\theta)$$
$$= l\cos\alpha\cos\theta - l\sin\alpha\sin\theta$$
$$= x\cos\theta - y\sin\theta$$

$$y' = |OP'|\sin(\alpha+\theta) = l\sin(\alpha+\theta)$$
$$= l\cos\alpha\sin\theta + l\sin\alpha\cos\theta$$
$$= x\sin\theta + y\cos\theta$$

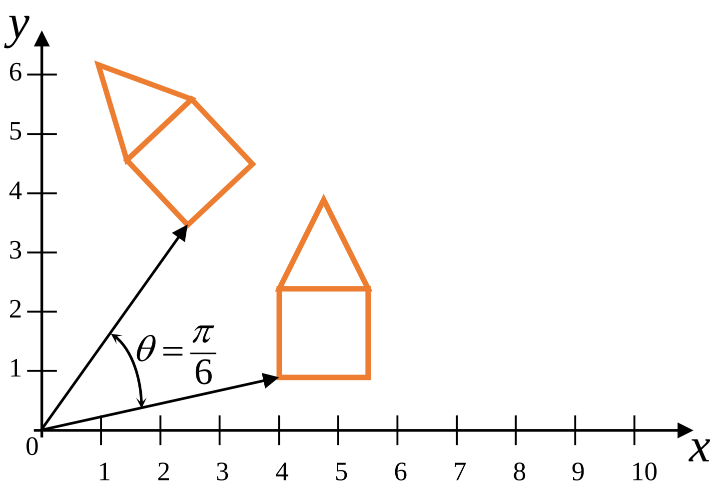
$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

$$\Downarrow$$

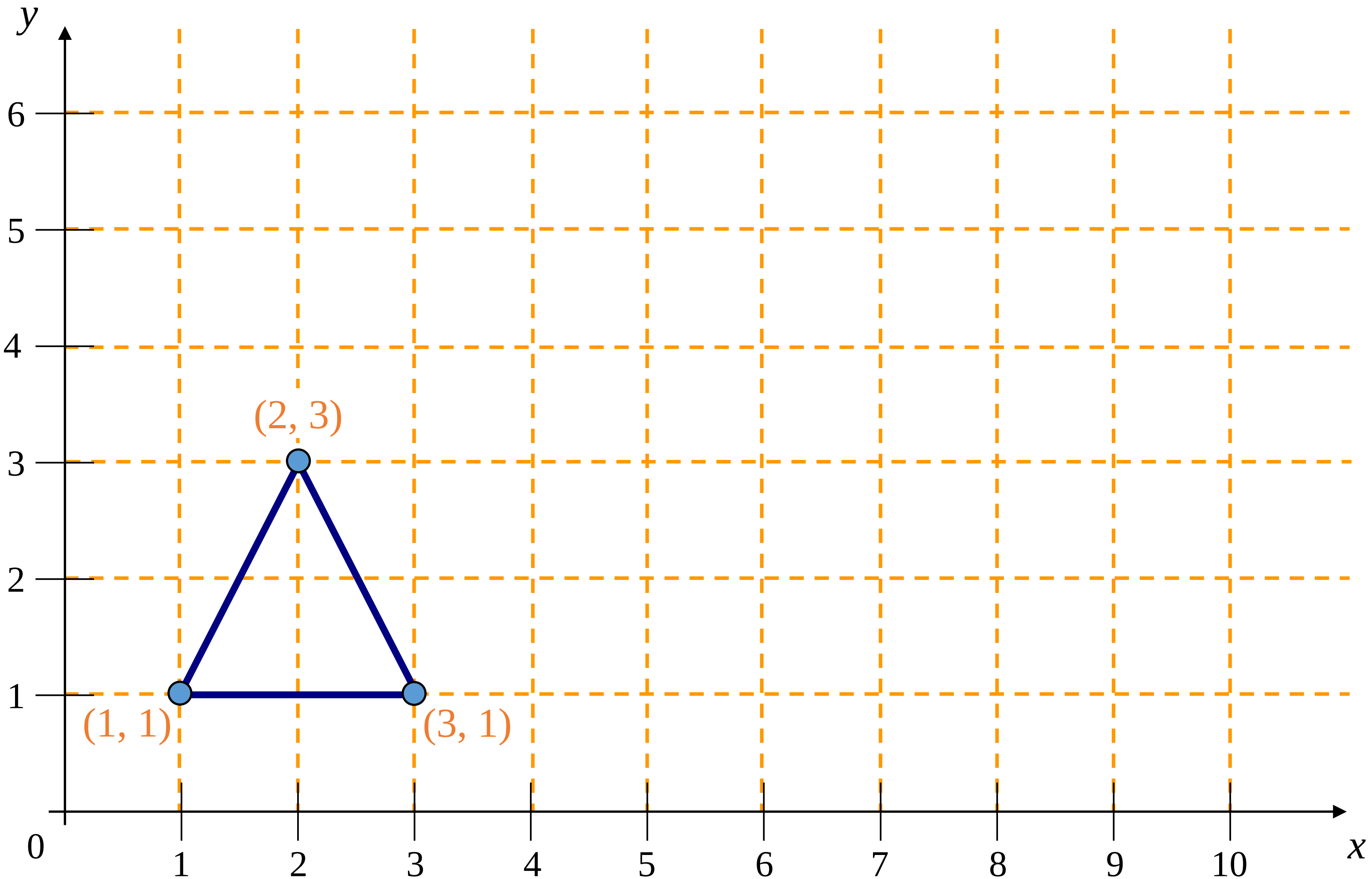
$$\boxed{P' = RP}$$

Rotation contd.

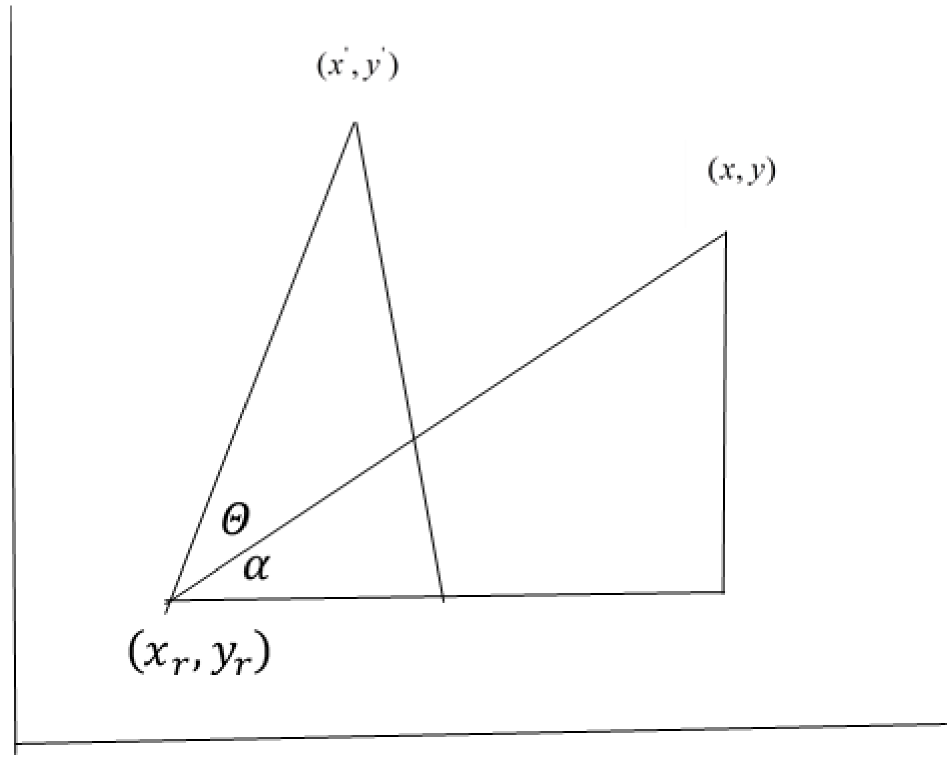
- Rotates all coordinates by a specified angle
- $$x_{new} = x_{old} \times \cos\theta - y_{old} \times \sin\theta$$
- $$y_{new} = x_{old} \times \sin\theta + y_{old} \times \cos\theta$$
- Points are always rotated about the origin



Solve yourself for theta=45 deg



Rotation w.r.t. external point (x_r, y_r)



Rotation of a point about an arbitrary pivot position

obtain the transformation equations for rotation of a point about any specified transformation position x_r and y_r

$$x' = x_r + (x - x_r)\cos\theta - (y - y_r)\sin\theta$$
$$y' = y_r + (x - x_r)\sin\theta + (y - y_r)\cos\theta$$

Review…

- Translate: $P' = P+T$
- Scale: $P' = SP$
- Rotate: $P' = RP$

Homogenous Coordinates

Homogenous Coordinates

- A point (x, y) can be re-written in **homogeneous coordinates** as (x_h, y_h, h)
- The **homogeneous parameter** h is a non-zero value such that:

$$x = \frac{x_h}{h} \qquad\qquad y = \frac{y_h}{h}$$

- We can then write any point (x, y) as (x_h, y_h, h)
- We can conveniently choose $h = 1$ so that (x, y) becomes $(x, y, 1)$

Homogeneous Translation

- The translation of a point by (t_x, t_y) can be written in matrix form as:

$$\begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix}$$

- Representing the point as a homogeneous column vector we perform the calculation as:

$$\begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} 1*x+0*y+t_x*1 \\ 0*x+1*y+t_y*1 \\ 0*x+0*y+1*1 \end{bmatrix} = \begin{bmatrix} x+t_x \\ y+t_y \\ 1 \end{bmatrix}$$

Remember Matrix Multiplication

- Recall how matrix multiplication takes place:

$$\begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix} \times \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} a*x+b*y+c*z \\ d*x+e*y+f*z \\ g*x+h*y+i*z \end{bmatrix}$$

Homogenous Coordinates

- To make operations easier, 2-D points are written as homogenous coordinate column vectors

Translation: $\begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} x + t_x \\ y + t_y \\ 1 \end{bmatrix}$ $\quad$ $\boldsymbol{or \quad P' = T(t_x, t_y)P}$

Scaling: $\begin{bmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} s_x \times x \\ s_y \times y \\ 1 \end{bmatrix}$ $\quad$ $\boldsymbol{or \quad P' = S(sx, sy)P}$

Homogenous Coordinates Contd.

Rotation:

$$\begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} \cos\theta \times x - \sin\theta \times y \\ \sin\theta \times x + \cos\theta \times y \\ 1 \end{bmatrix}$$

or

$$\boldsymbol{P' = R(\theta)P}$$

Inverse Transformations

- Transformations can easily be reversed using inverse transformations
- Inverse Translation

$$T^{-1} = \begin{bmatrix} 1 & 0 & -t_x \\ 0 & 1 & -t_y \\ 0 & 0 & 1 \end{bmatrix}$$

Inverse Scaling

$$S^{-1} = \begin{bmatrix} \frac{1}{s_x} & 0 & 0 \\ 0 & \frac{1}{s_y} & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

- Inverse Rotation

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

Composite Transformation

- A matrix for any sequence of transformation is called composite transformation matrix and can be obtained by calculating the matrix product of the individual transformations.

Composition of 2D Transformations

1. Additivity of successive translations

We want to translate a point *P* to *P'* by $T(t_{x1}, t_{y1})$ and then to *P''* by another $T(t_{x2}, t_{y2})$

$$P'' = T(t_{x2}, t_{y2})P' = T(t_{x2}, t_{y2})[T(t_{x1}, t_{y1})P]$$

On the other hand, we can define $T_{21} = T(t_{x1}, t_{y1})\, T(t_{x2}, t_{y2})$ first, then apply T_{21} to *P*:

$$\boxed{P'' = T_{21}P}$$

where $T_{21} = T(t_{x2}, t_{y2})T(t_{x1}, t_{y1})$

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

Composition of 2D Transformations

2. Multiplicativity of successive scalings

$$P'' = S(s_{x2}, s_{y2})[S(s_{x1}, s_{y1})P]$$
$$= [S(s_{x2}, s_{y2})S(s_{x1}, s_{y1})]P$$
$$= S_{21}P$$

where

$$S_{21} = S(s_{x2}, s_{y2})S(s_{x1}, s_{y1})$$
$$= \begin{bmatrix} s_{x2} & 0 & 0 \\ 0 & s_{y2} & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} s_{x1} & 0 & 0 \\ 0 & s_{y1} & 0 \\ 0 & 0 & 1 \end{bmatrix}$$
$$= \begin{bmatrix} s_{x2} * s_{x1} & 0 & 0 \\ 0 & s_{y2} * s_{y1} & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Composition of 2D Transformations

3. **Additivity of successive rotations**

$$
\begin{aligned}
P'' &= R(\theta_2)[R(\theta_1)P] \\
&= [R(\theta_2)R(\theta_1)]P \\
&= R_{21}P
\end{aligned}
$$

where

$$
\begin{aligned}
R_{21} &= R(\theta_2)R(\theta_1) \\
&= \begin{bmatrix} \cos\theta_2 & -\sin\theta_2 & 0 \\ \sin\theta_2 & \cos\theta_2 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos\theta_1 & -\sin\theta_1 & 0 \\ \sin\theta_1 & \cos\theta_1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \\
&= \begin{bmatrix} \cos(\theta_2+\theta_1) & -\sin(\theta_2+\theta_1) & 0 \\ \sin(\theta_2+\theta_1) & \cos(\theta_2+\theta_1) & 0 \\ 0 & 0 & 1 \end{bmatrix}
\end{aligned}
$$

Composition of 2D Transformations

4. **Different types of elementary transformations discussed above can be concatenated as well.**

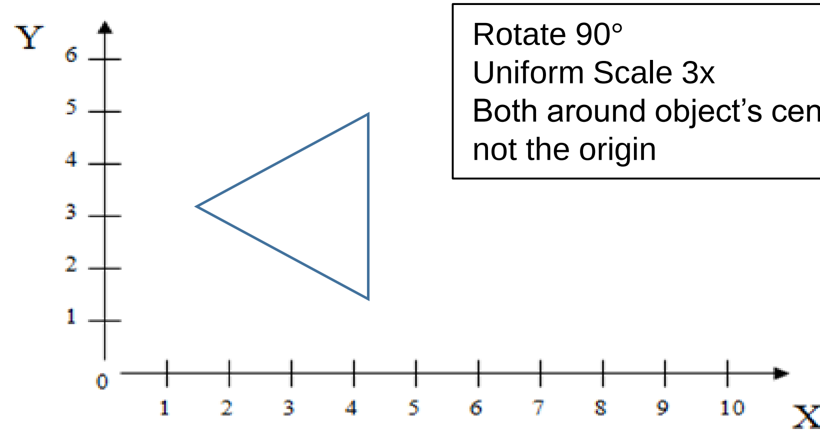
$$
\begin{aligned}
P' &= R(\theta)[T(d_x, d_y)P] \\
&= [R(\theta)T(d_x, d_y)]P \\
&= MP
\end{aligned}
$$

where $M = R(\theta)T(d_x, d_y)$

Composition (an example) (2D)

- Start:

Goal:



Rotate 90°
Uniform Scale 3x
Both around object's center, not the origin

- Important concept: Make the problem simpler
- Translate object to origin first, scale , rotate, and translate back

$$\boldsymbol{T^{-1}RST} = \begin{bmatrix} 1 & 0 & -3 \\ 0 & 1 & -3 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos 90 & -sin90 & 0 \\ sin90 & cos90 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 3 & 0 & 0 \\ 0 & 3 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 3 \\ 0 & 1 & 3 \\ 0 & 0 & 1 \end{bmatrix}$$

- Apply to all vertices

Order Matters!

As we said, the order for composition of 2D geometrical transformations matters, because, in general, matrix multiplication is not commutative. However, it is easy to show that, in the following four cases, commutativity holds:

1). Translation + Translation
2). Scaling + Scaling
3). Rotation + Rotation
4). Scaling (with $s_x = s_y$) + Rotation

just to verify case 4:

$$M_1 = S(s_x, s_y)R(\theta)$$

$$= \begin{bmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} s_x * \cos\theta & -s_x * \sin\theta & 0 \\ s_y * \sin\theta & s_y * \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$M_2 = R(\theta)S(s_x, s_y)$$

$$= \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} s_x * \cos\theta & -s_y * \sin\theta & 0 \\ s_x * \sin\theta & s_y * \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

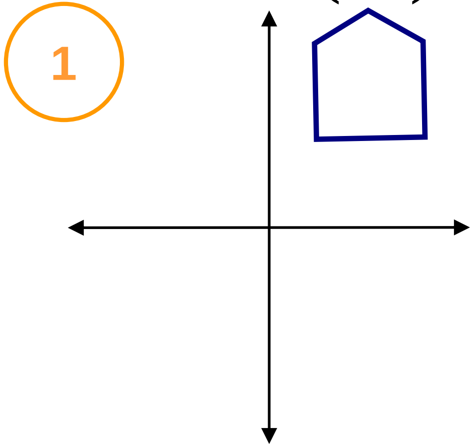
if $s_x = s_y$, $M_1 = M_2$.

Composite Transformations

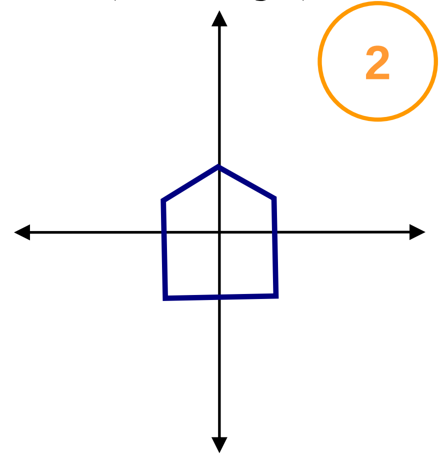
- Example: Imagine rotating a polygon around a point other than the origin
  - Transform to centre point to origin
  - Rotate around origin
  - Transform back to centre point

Combining Transformations (cont…)

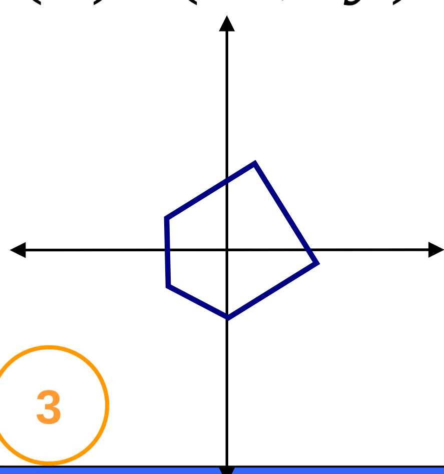
$House\ (H)$



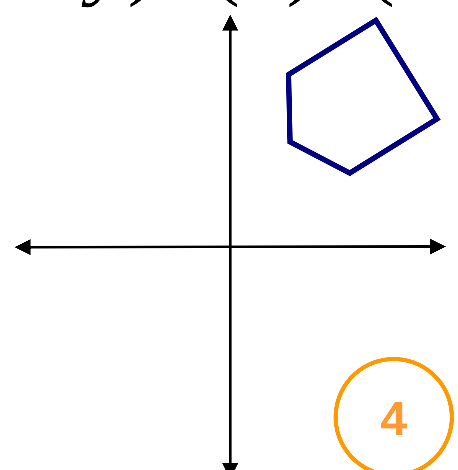
$T(dx, dy)H$



$R(\theta)T(dx, dy)H$



$T(-dx, -dy)R(\theta)T(dx, dy)H$



Combining Transformations (cont…)

- The three transformation matrices are combined as follows

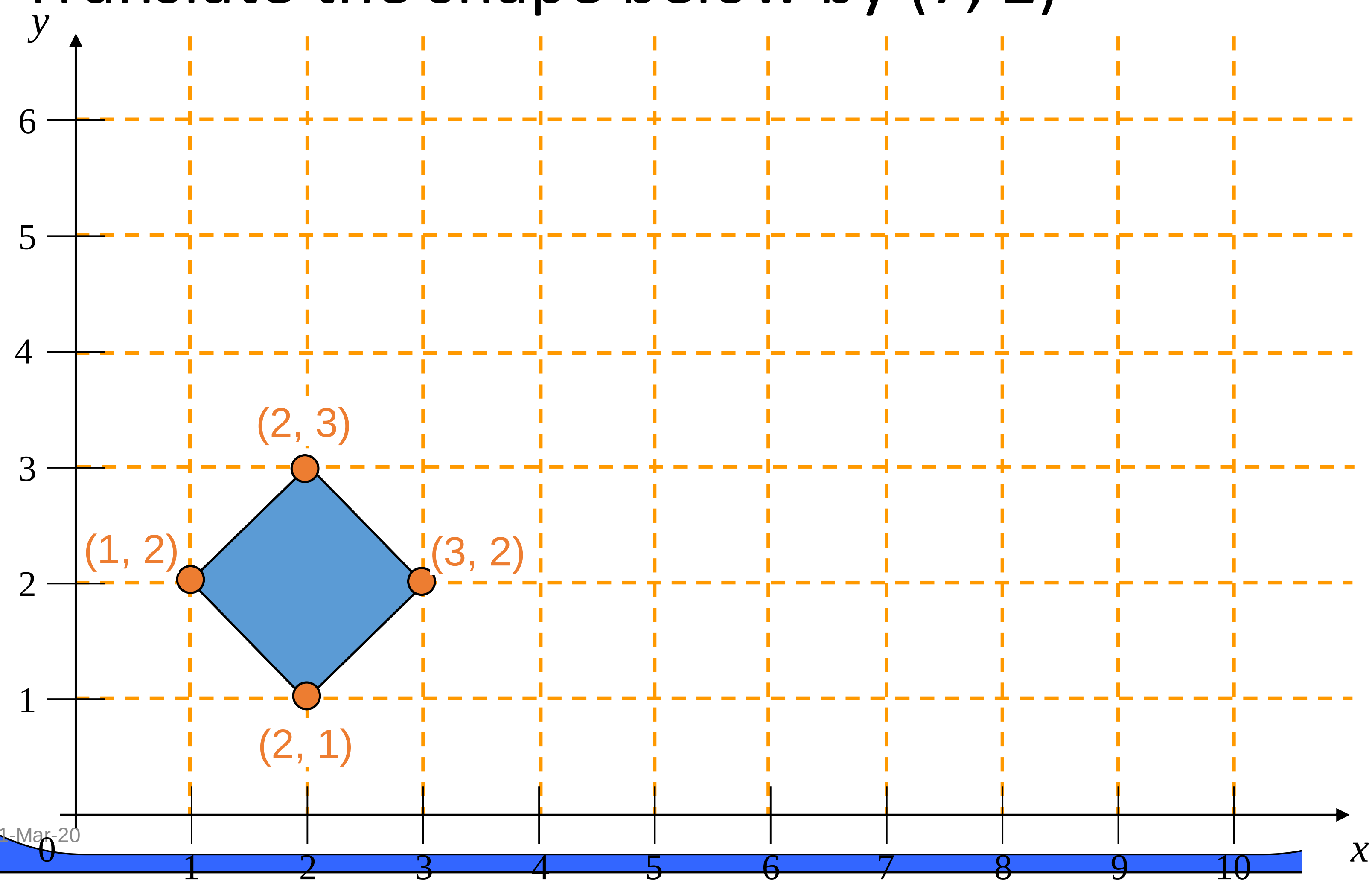
$$\begin{bmatrix} 1 & 0 & -t_x \\ 0 & 1 & -t_y \\ 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

$$P' = T(-t_x, -t_y) R(\theta) T(t_x, t_y) P$$

REMEMBER: Matrix multiplication is not commutative so order matters

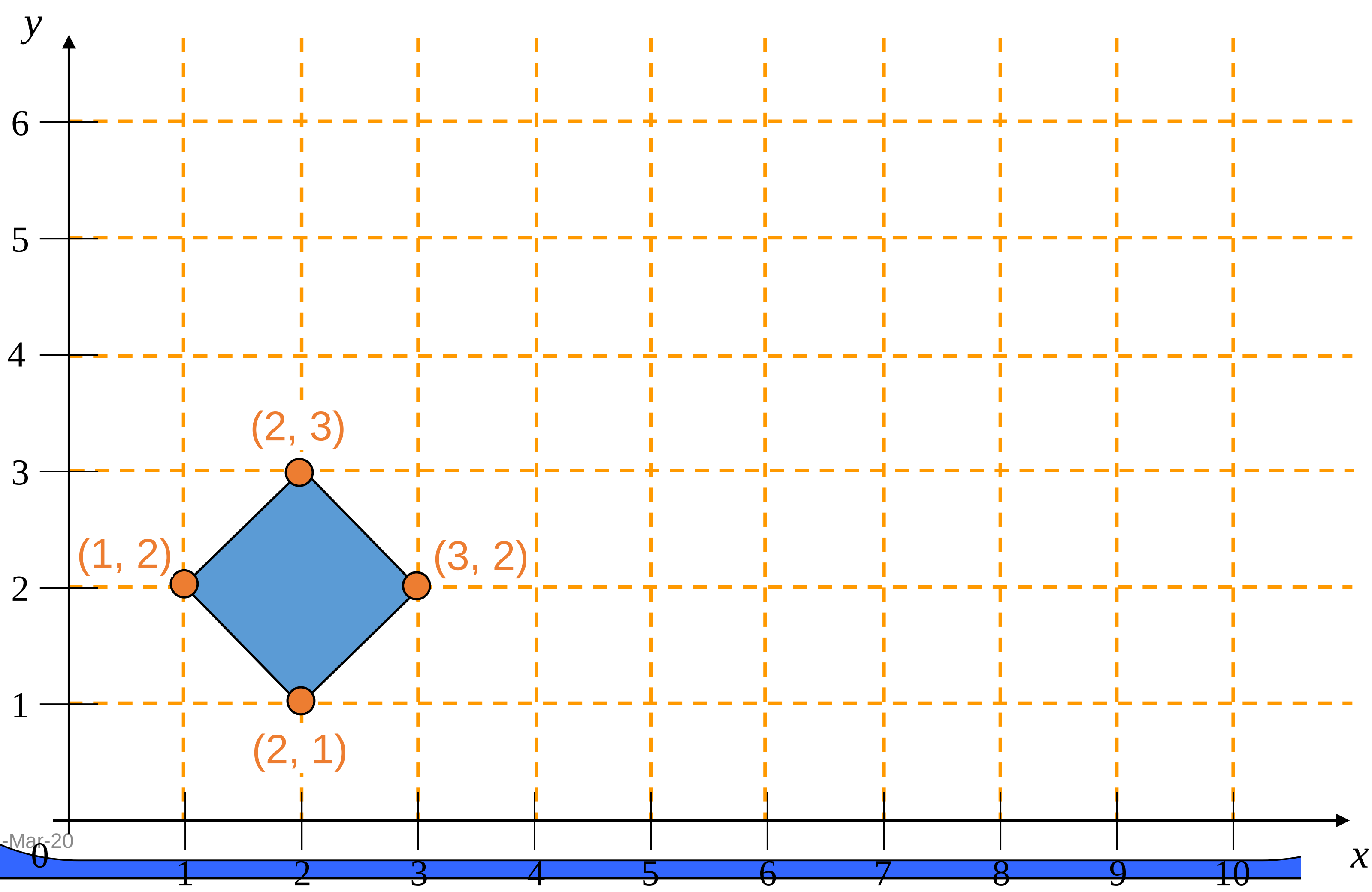
Exercises 1

- Translate the shape below by (7, 2)



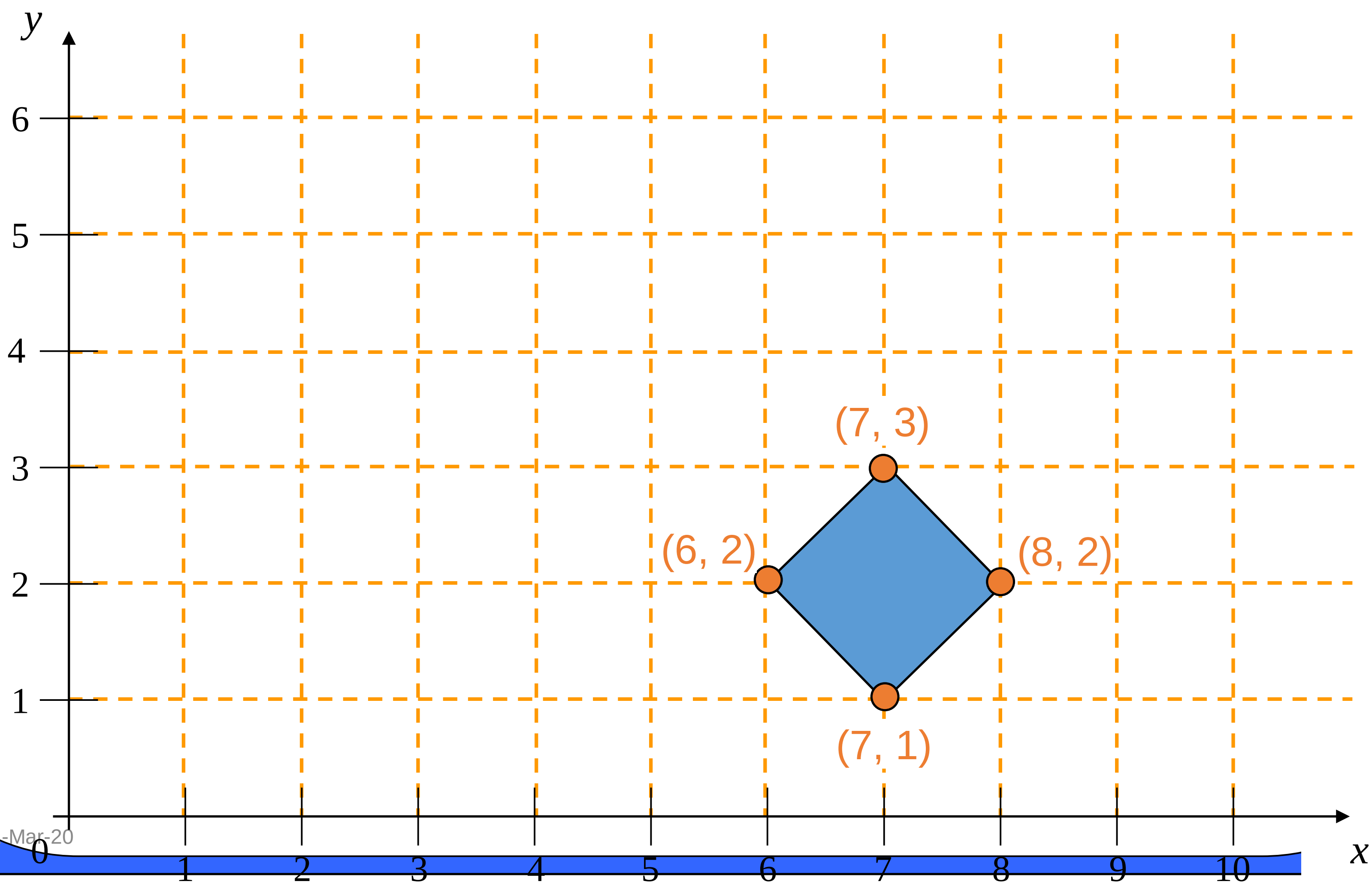
Exercises 2

- Scale the shape below by 3 in x and 2 in y



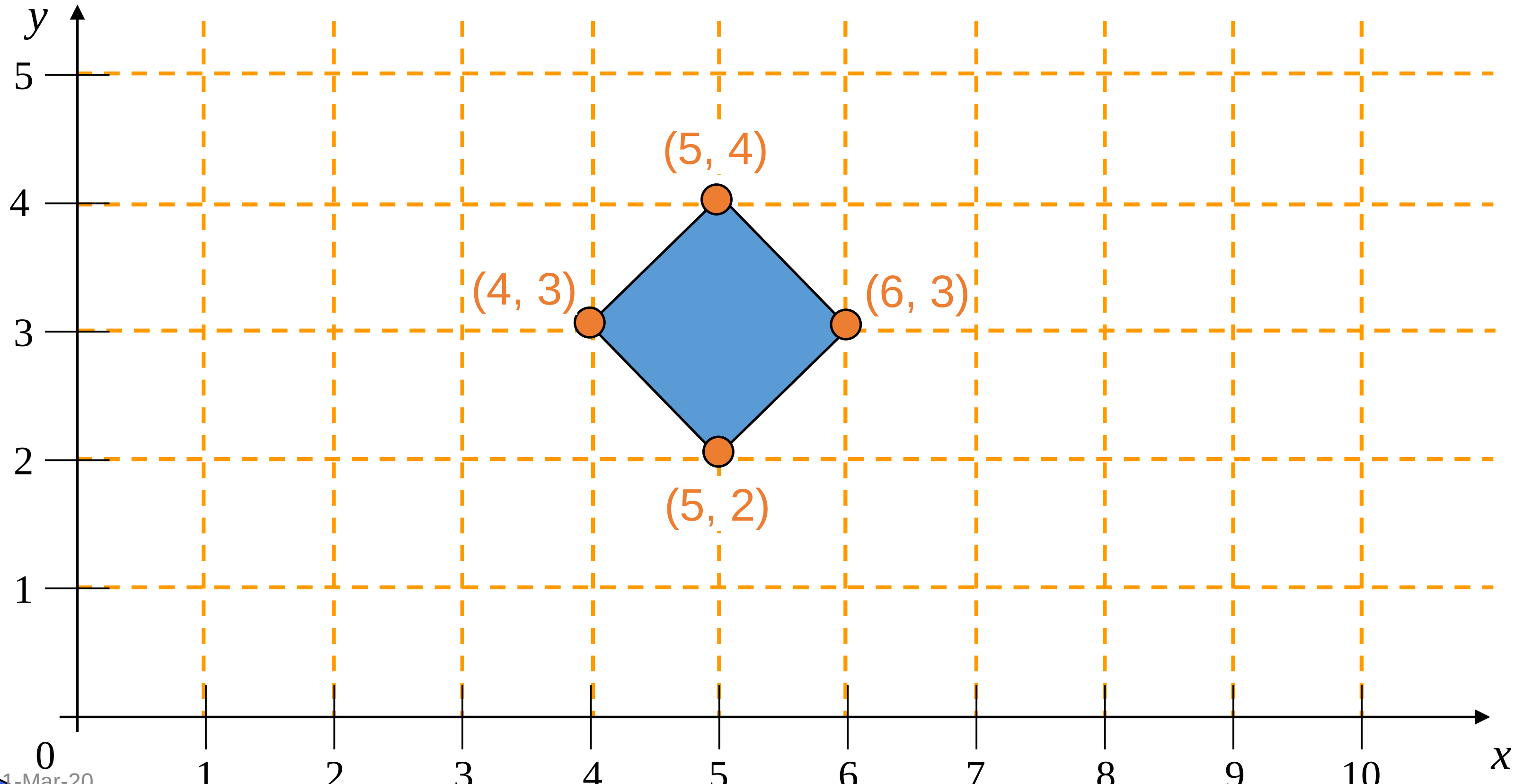
Exercises 3

- Rotate the shape below by 30° about the origin



Exercises 5

- Using matrix multiplication calculate the rotation of the shape below by 45° about its centre (5, 3)



Shearing of 2D objects

- Shearing enjoys the property that all points along a given line *l* remain fixed, while other points are shifted parallel to *l* by a distance that is proportional to their perpendicular distance from *l*.
- Note that shearing an object in the plane does not change its area at all.
- As a margin note, let us say that shearing can easily be generalized to three dimensions, where planes are translated instead of lines.

Shearing of 2D objects

Shearing a point (*x*, *y*) by a factor *hx* along the *x* axis and *hy* along the *y* axis is given by the following equations:

$$\boldsymbol{x' = x + h_x . y}$$
$$\boldsymbol{y' = y + h_y . x}$$

It can be represented in Matrix form as follows:

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & h_x & 0 \\ h_y & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Shearing of 2D objects

The effect of a shearing looks like "pushing" an object in a direction that is parallel to a coordinate axis in 2D (or coordinate plane in 3D). Note that we can do this only in the x-direction as follows

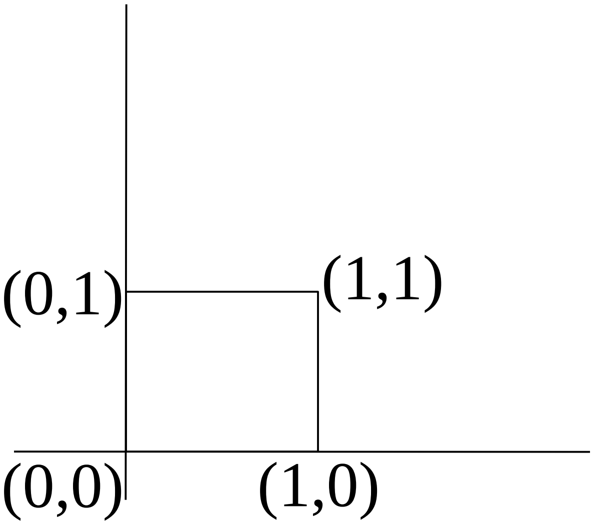
$$x' = x + h_x . y$$
$$y' = y$$

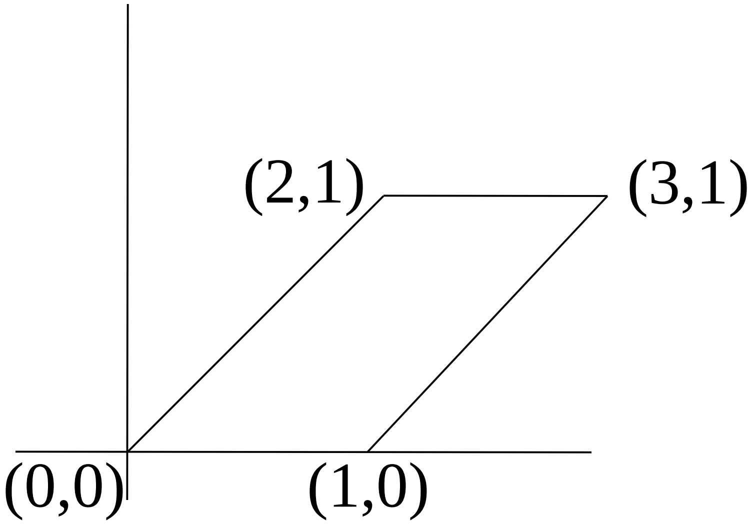
or in the y-direction

$$y' = x + h_y . x$$
$$x' = x$$

An X- direction Shear

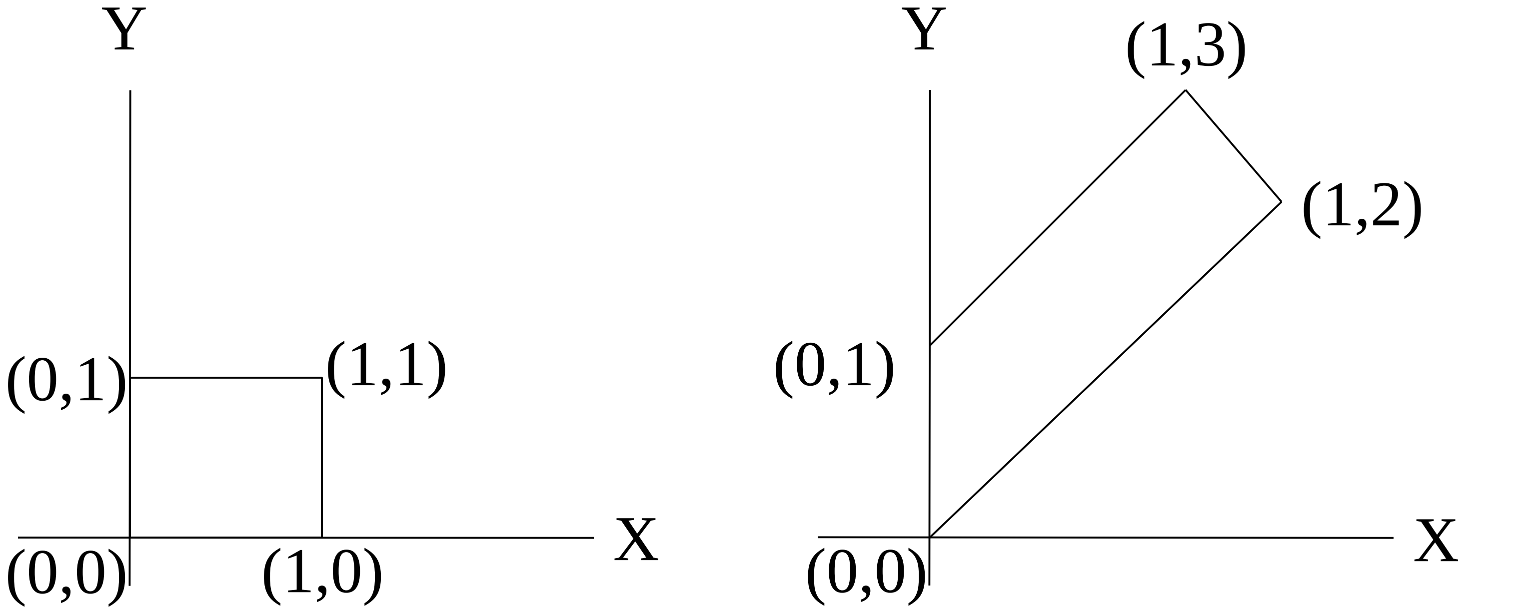
For example, Sh_x=2





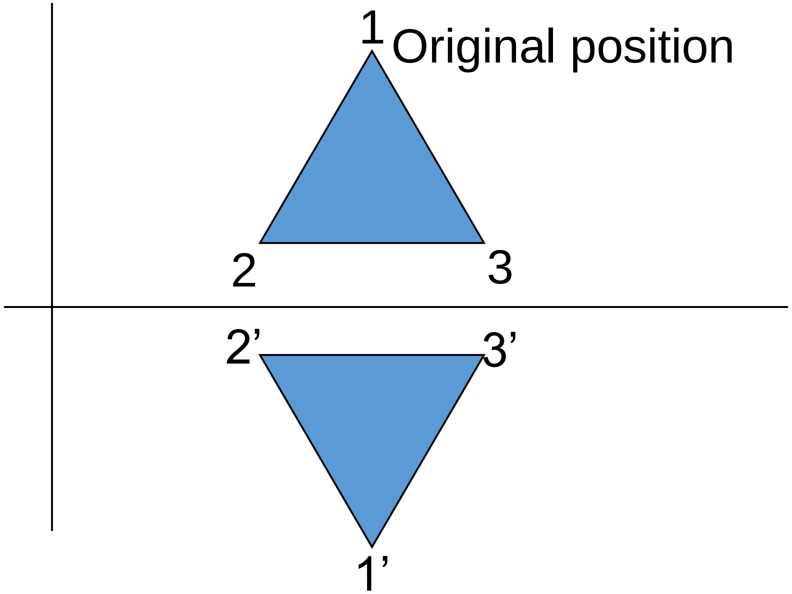
An Y- direction Shear

For example, Sh_y=2



Other transformations

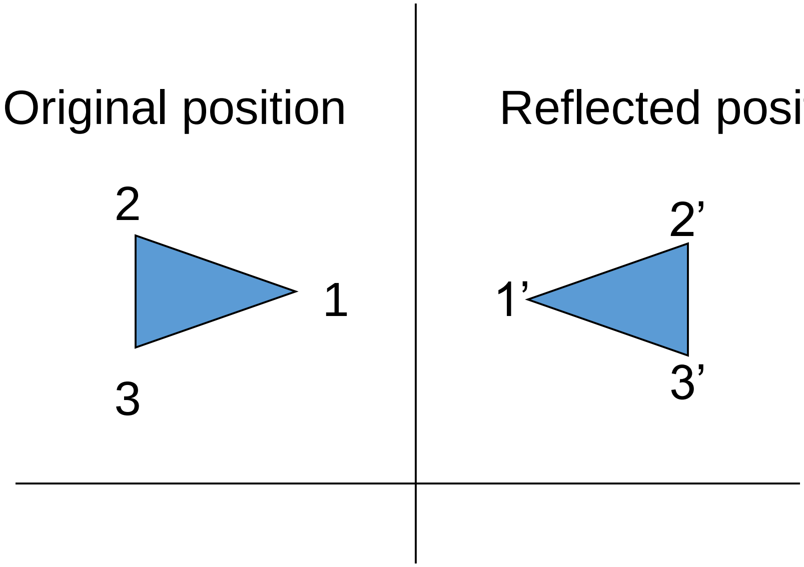
- **Reflection** is a transformation that produces a mirror image of an object. It is obtained by rotating the object by 180 deg about the reflection axis



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

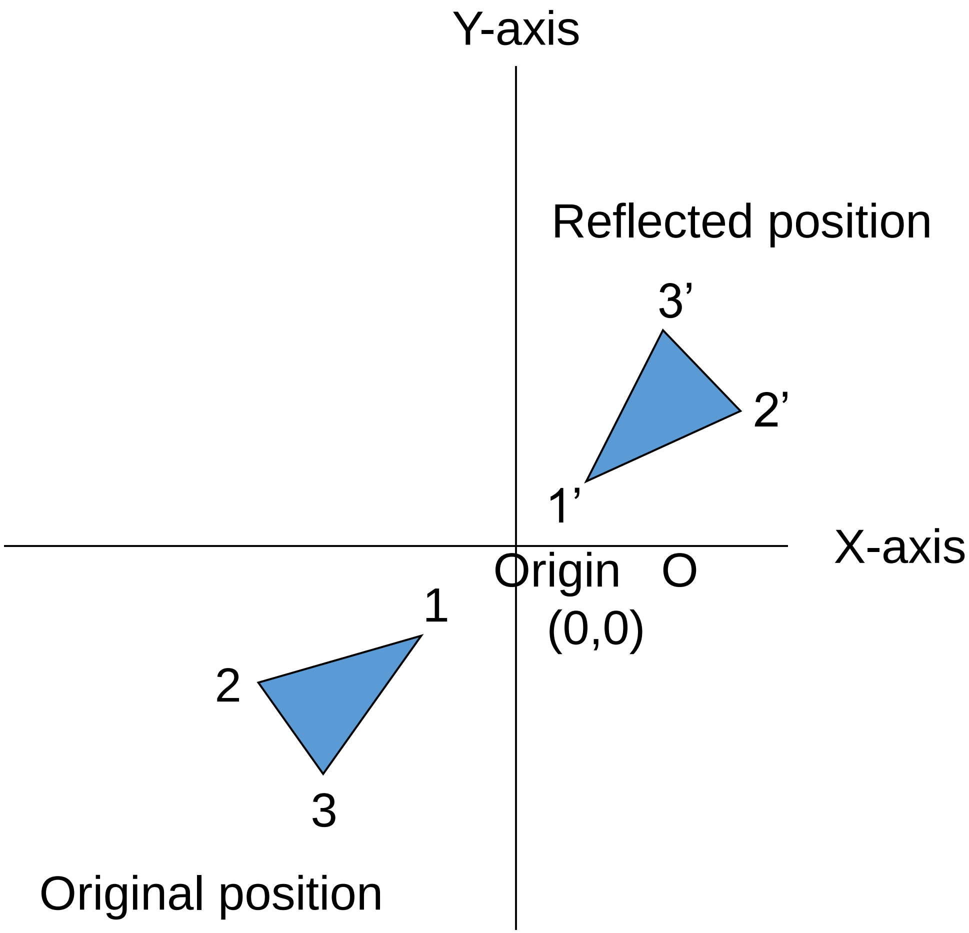


Reflection about the line x=0, the Y- axis , is accomplished with the transformation matrix

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

Reflection

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

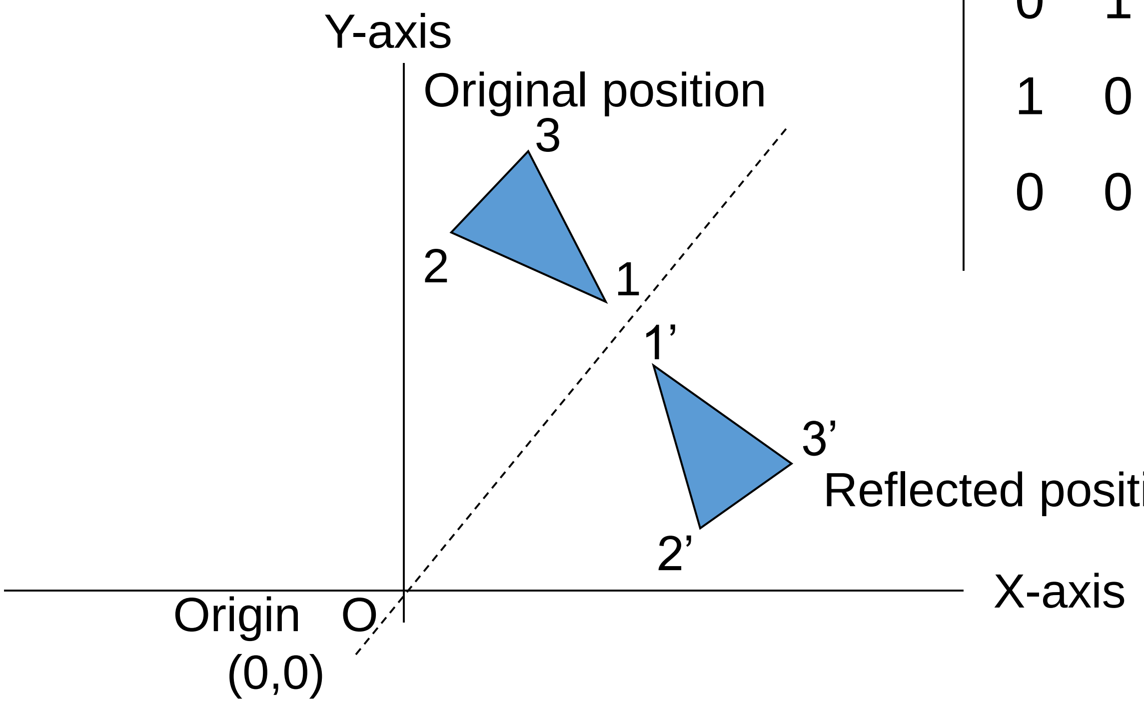


$$\begin{vmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{vmatrix}$$

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

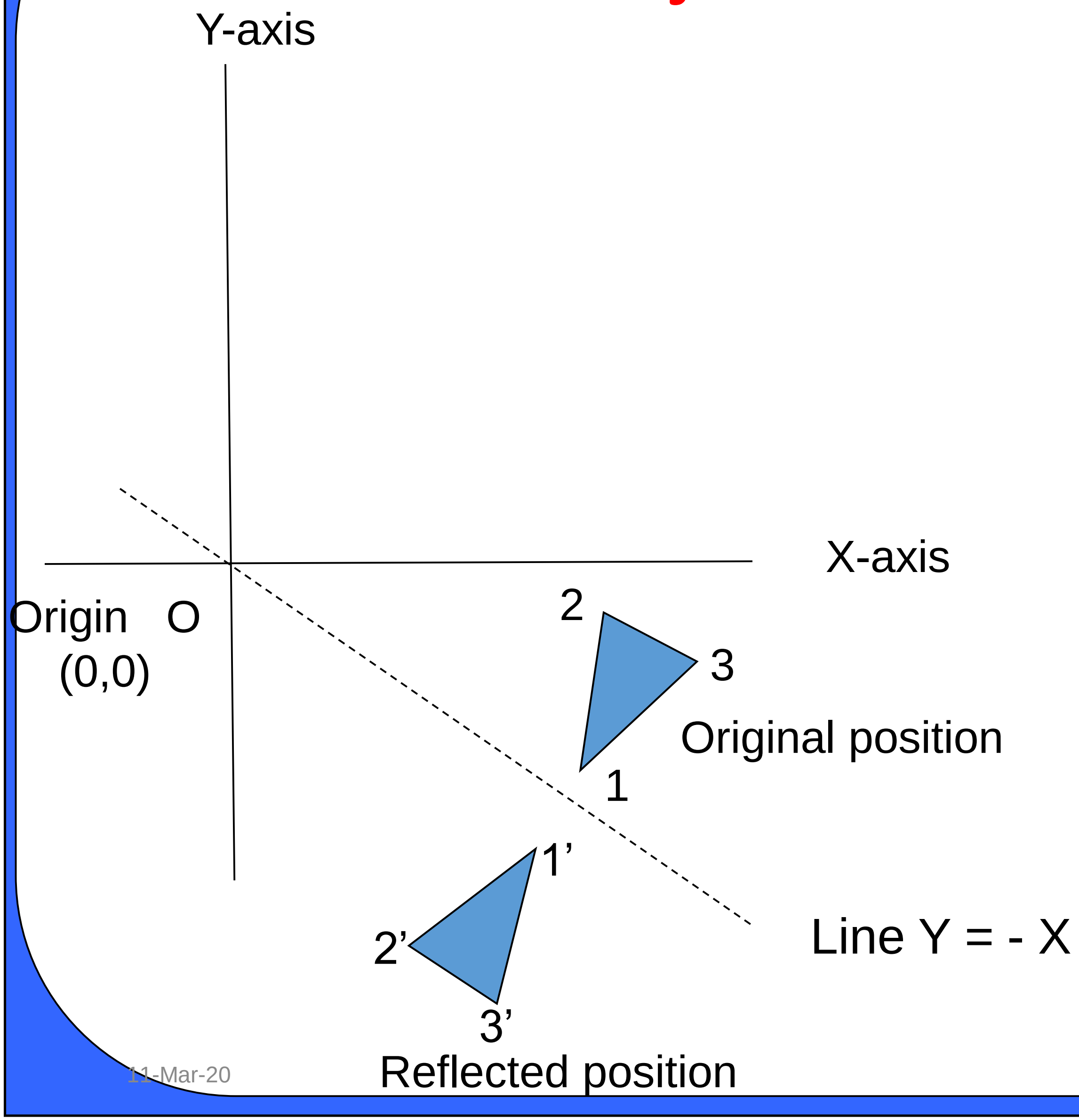
This can be generalized to any reflection point in the xy plane. This reflection is the same as a 180 degree rotation in the xy plane using the reflection point as the pivot point.

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 Problem