CSE4203: Computer Graphics

Chapter – 6 (part - A)

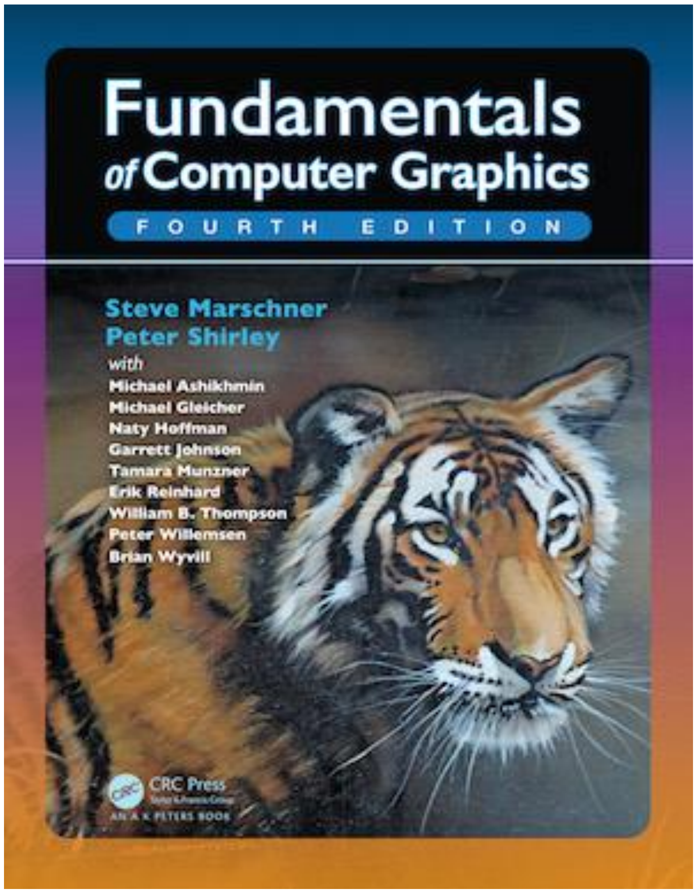
Transformation Matrices

Mohammad Imrul Jubair

Outline

– Transformation
– Linear Transformation
  - Scaling
  - Shearing
  - Rotation
– Composite Transformation

Credit



CS4620: Introduction to Computer Graphics

Cornell University

Instructor: Steve Marschner

http://www.cs.cornell.edu/courses/cs4620/2019fa/

2D Linear Transformations (1/1)

$$\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} a_{11}x + a_{12}y \\ a_{21}x + a_{22}y \end{bmatrix}$$

Linear Transformation: Operation of taking a **vector** and produces **another vector** by a **simple matrix multiplication**.

Credit: Fundamentals of Computer Graphics 3rd Edition by Peter Shirley, Steve Marschner | http://www.cs.cornell.edu/courses/cs4620/2019fa/

Scaling (1/6)

- The most basic transform is a scale along the coordinate axes.

$$\text{scale}(s_x, s_y) = \begin{bmatrix} s_x & 0 \\ 0 & s_y \end{bmatrix}$$

Credit: Fundamentals of Computer Graphics 3rd Edition by Peter Shirley, Steve Marschner | http://www.cs.cornell.edu/courses/cs4620/2019fa/

Scaling (2/6)

- The most basic transform is a scale along the coordinate axes.

$$\text{scale}(s_x, s_y) = \begin{bmatrix} s_x & 0 \\ 0 & s_y \end{bmatrix}$$

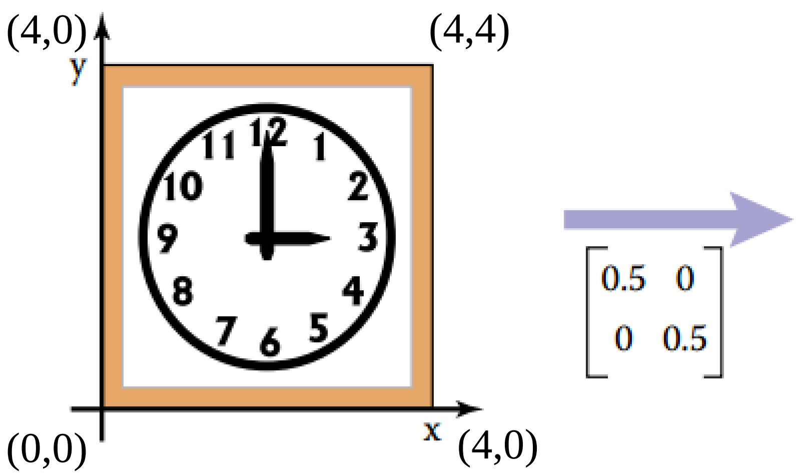
The matrix does to a vector with Cartesian components (x, y):

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

Credit: Fundamentals of Computer Graphics 3rd Edition by Peter Shirley, Steve Marschner | http://www.cs.cornell.edu/courses/cs4620/2019fa/

Scaling (3/6)

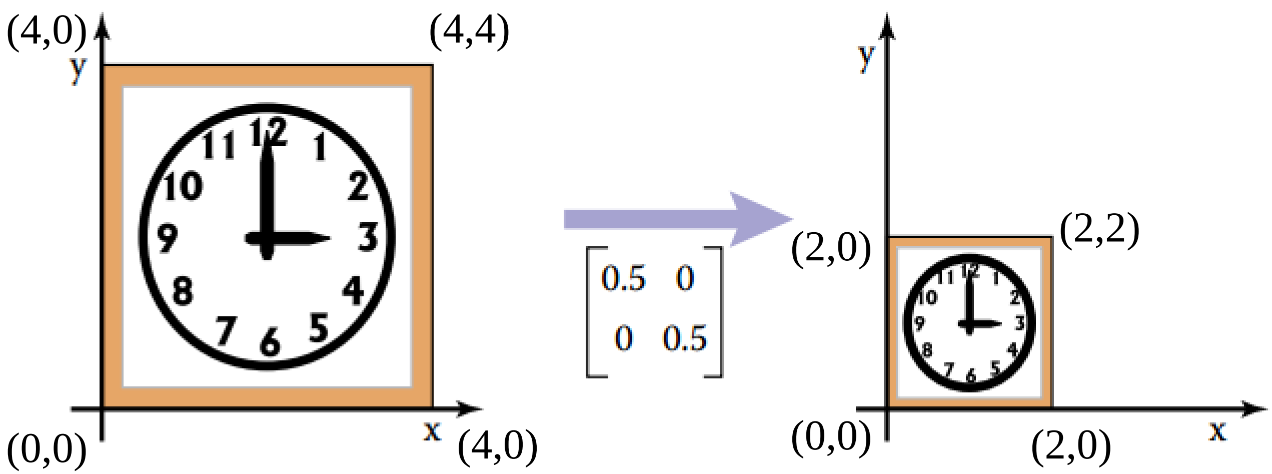
$$\mathrm{scale}(0.5, 0.5) = \begin{bmatrix} 0.5 & 0 \\ 0 & 0.5 \end{bmatrix}$$



Credit: Fundamentals of Computer Graphics 3rd Edition by Peter Shirley, Steve Marschner | http://www.cs.cornell.edu/courses/cs4620/2019fa/

Scaling (4/6)

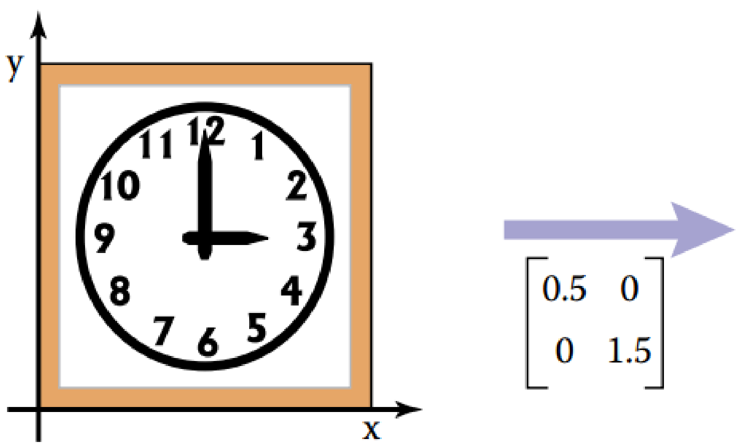
$$\text{scale}(0.5, 0.5) = \begin{bmatrix} 0.5 & 0 \\ 0 & 0.5 \end{bmatrix}$$



Credit: Fundamentals of Computer Graphics 3rd Edition by Peter Shirley, Steve Marschner | http://www.cs.cornell.edu/courses/cs4620/2019fa/

Scaling (5/6)

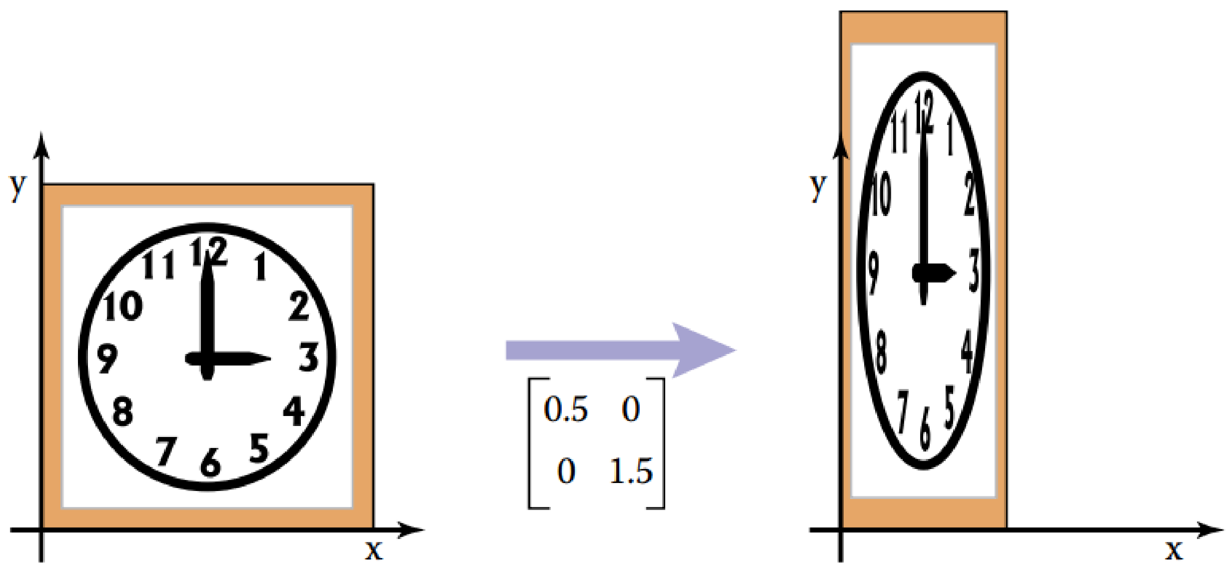
$$\mathrm{scale}(0.5, 1.5) = \begin{bmatrix} 0.5 & 0 \\ 0 & 1.5 \end{bmatrix}$$



Credit: Fundamentals of Computer Graphics 3rd Edition by Peter Shirley, Steve Marschner | http://www.cs.cornell.edu/courses/cs4620/2019fa/

Scaling (6/6)

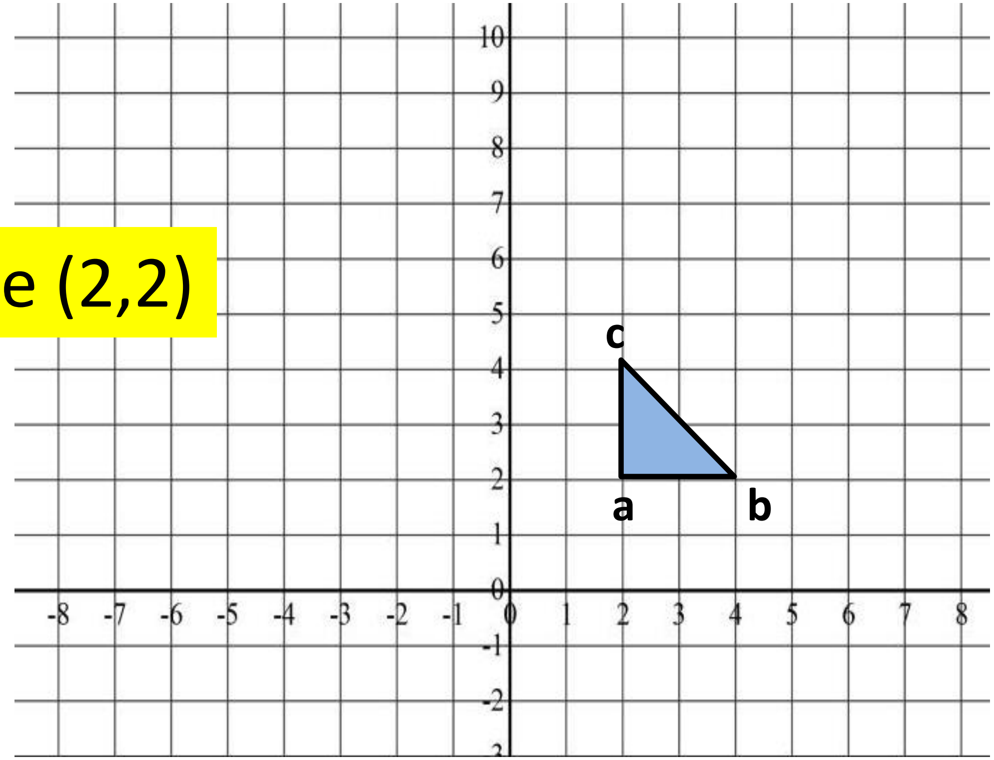
$$\text{scale}(0.5, 1.5) = \begin{bmatrix} 0.5 & 0 \\ 0 & 1.5 \end{bmatrix}$$



Credit: Fundamentals of Computer Graphics 3rd Edition by Peter Shirley, Steve Marschner | http://www.cs.cornell.edu/courses/cs4620/2019fa/

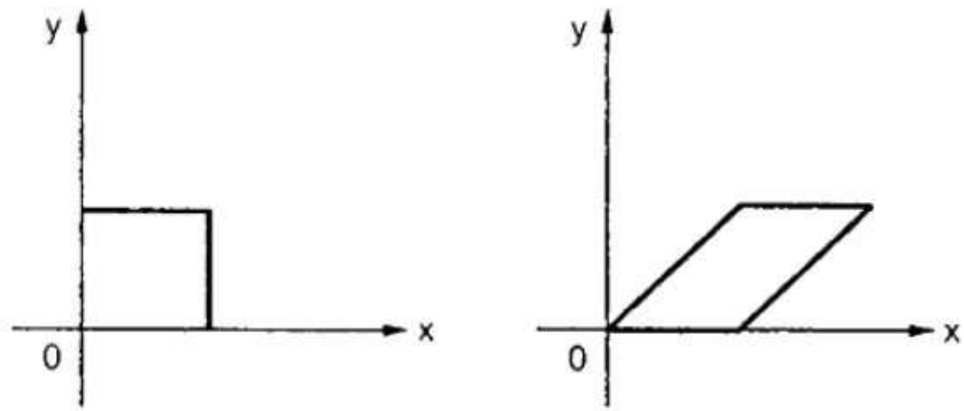
Scaling (6/6)

Scale (2,2)



Shearing (1/5)

- A shear is something that pushes things sideways

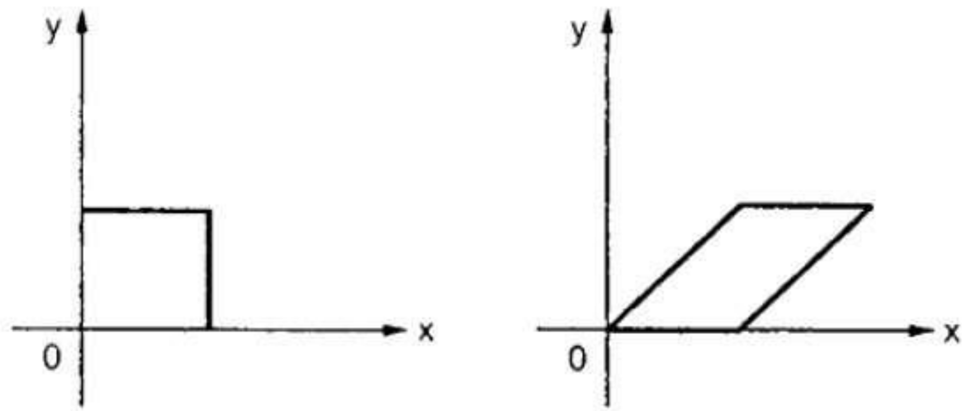


(a) Original object (b) Object after x shear

Credit: Fundamentals of Computer Graphics 3rd Edition by Peter Shirley, Steve Marschner | http://www.cs.cornell.edu/courses/cs4620/2019fa/

Shearing (2/5)

- A shear is something that pushes things sideways



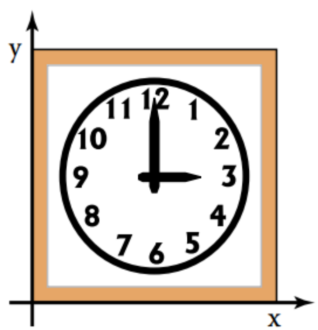
(a) Original object (b) Object after x shear

$$\text{shear-x}(s) = \begin{bmatrix} 1 & s \\ 0 & 1 \end{bmatrix}, \quad \text{shear-y}(s) = \begin{bmatrix} 1 & 0 \\ s & 1 \end{bmatrix}$$

Source: https://www.tutorialspoint.com/computer_graphics/2d_transformation.htm

Shearing (3/5)

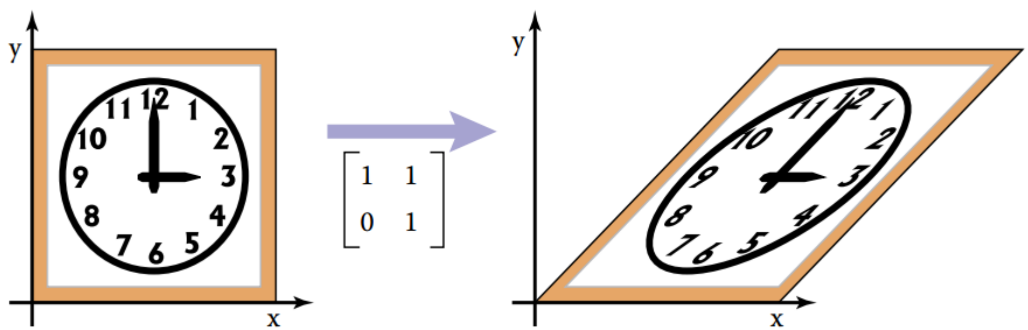
$$\text{shear-x}(1) = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}$$



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

Shearing (4/5)

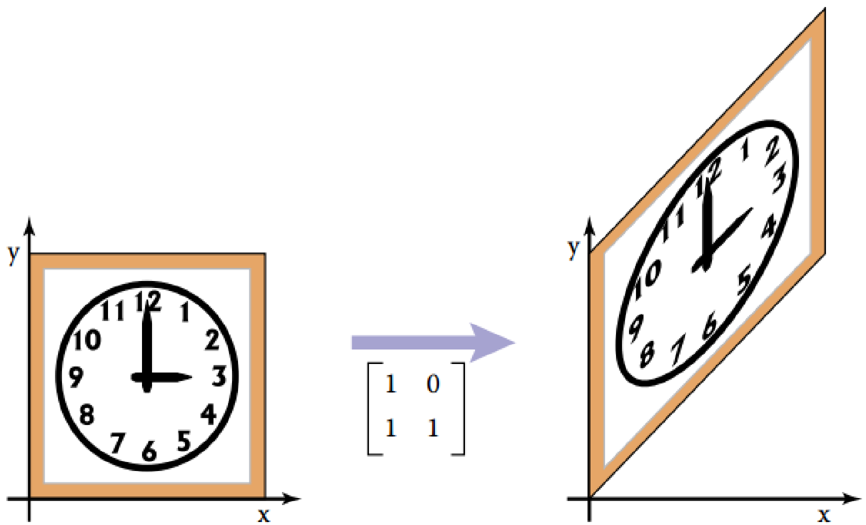
$$\text{shear-x}(1) = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}$$



Credit: Fundamentals of Computer Graphics 3rd Edition by Peter Shirley, Steve Marschner | http://www.cs.cornell.edu/courses/cs4620/2019fa/

Shearing (5/5)

$$\text{shear-y}(1) = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}$$

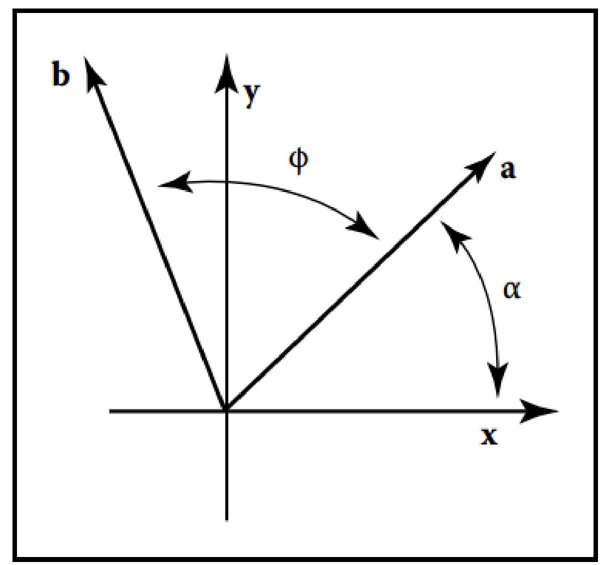


Credit: Fundamentals of Computer Graphics 3rd Edition by Peter Shirley, Steve Marschner | http://www.cs.cornell.edu/courses/cs4620/2019fa/

Rotation (1/11)

$$x_a = r\cos\alpha,$$

$$y_a =$$



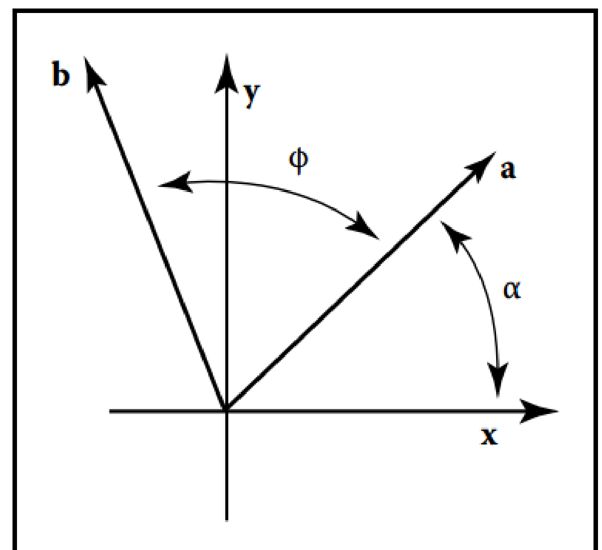
Rotation (2/11)

$$x_a = r\cos\alpha,$$

$$y_a = r\sin\alpha.$$

$$x_b = r\cos(\alpha + \phi) =$$

$$y_b =$$



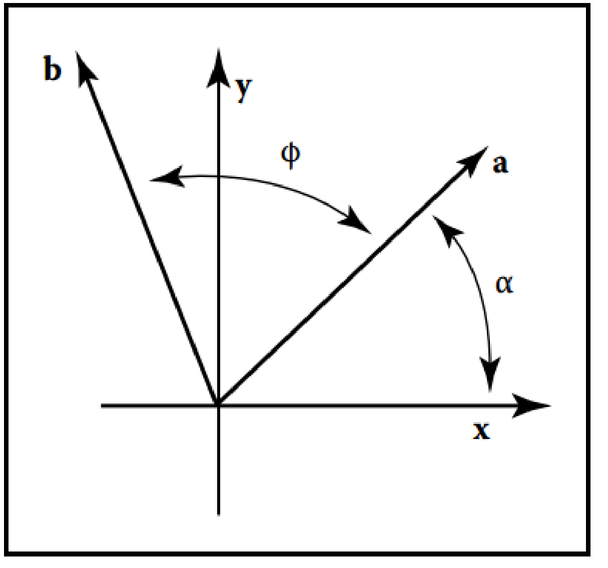
Rotation (3/11)

$$x_a = r\cos\alpha,$$

$$y_a = r\sin\alpha.$$

$$x_b = r\cos(\alpha+\phi) = r\cos\alpha\cos\phi - r\sin\alpha\sin\phi,$$

$$y_b =$$



Rotation (4/11)

$$x_a = r\cos\alpha,$$

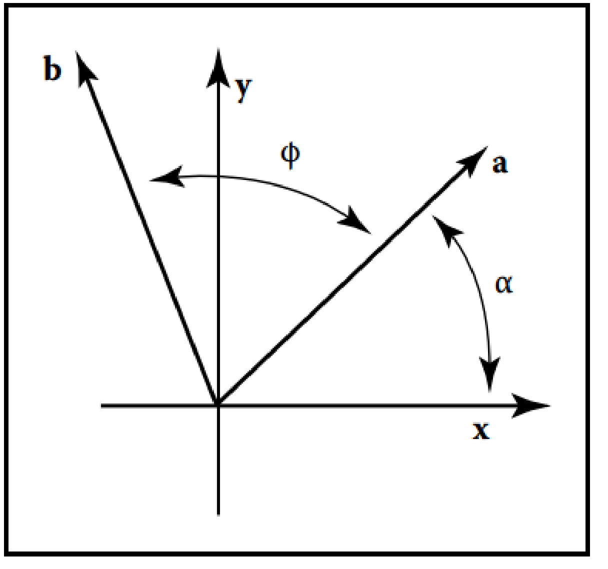
$$y_a = r\sin\alpha.$$

$$x_b = r\cos(\alpha+\phi) = r\cos\alpha\cos\phi - r\sin\alpha\sin\phi,$$

$$y_b = r\sin(\alpha+\phi) = r\sin\alpha\cos\phi + r\cos\alpha\sin\phi.$$

$$x_b =$$

$$y_b =$$



Rotation (5/11)

$$x_a = r\cos\alpha,$$

$$y_a = r\sin\alpha.$$

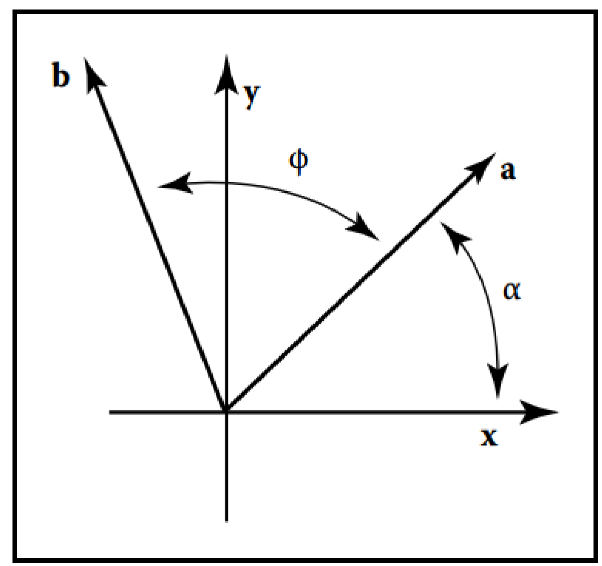
$$x_b = r\cos(\alpha+\phi) = r\cos\alpha\cos\phi - r\sin\alpha\sin\phi,$$

$$y_b = r\sin(\alpha+\phi) = r\sin\alpha\cos\phi + r\cos\alpha\sin\phi.$$

$$x_b = x_a\cos\phi - y_a\sin\phi,$$

$$y_b = y_a\cos\phi + x_a\sin\phi.$$

$$\text{rotate}(\phi) = \begin{bmatrix} \quad & \quad \\ \quad & \quad \end{bmatrix}$$



Rotation (6/11)

$$x_a = r\cos\alpha,$$

$$y_a = r\sin\alpha.$$

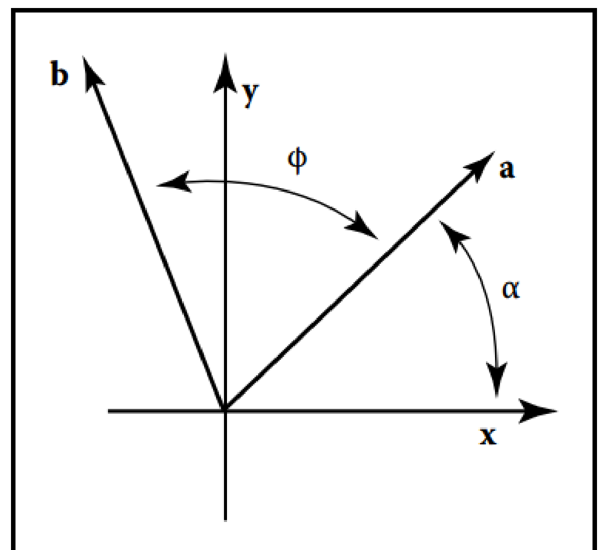
$$x_b = r\cos(\alpha+\phi) = r\cos\alpha\cos\phi - r\sin\alpha\sin\phi,$$

$$y_b = r\sin(\alpha+\phi) = r\sin\alpha\cos\phi + r\cos\alpha\sin\phi.$$

$$x_b = x_a\cos\phi - y_a\sin\phi,$$

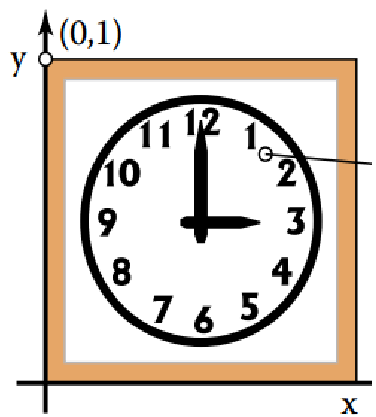
$$y_b = y_a\cos\phi + x_a\sin\phi.$$

$$\text{rotate}(\phi) = \begin{bmatrix}\cos\phi & -\sin\phi\\ \sin\phi & \cos\phi\end{bmatrix}$$



Rotation (7/11)

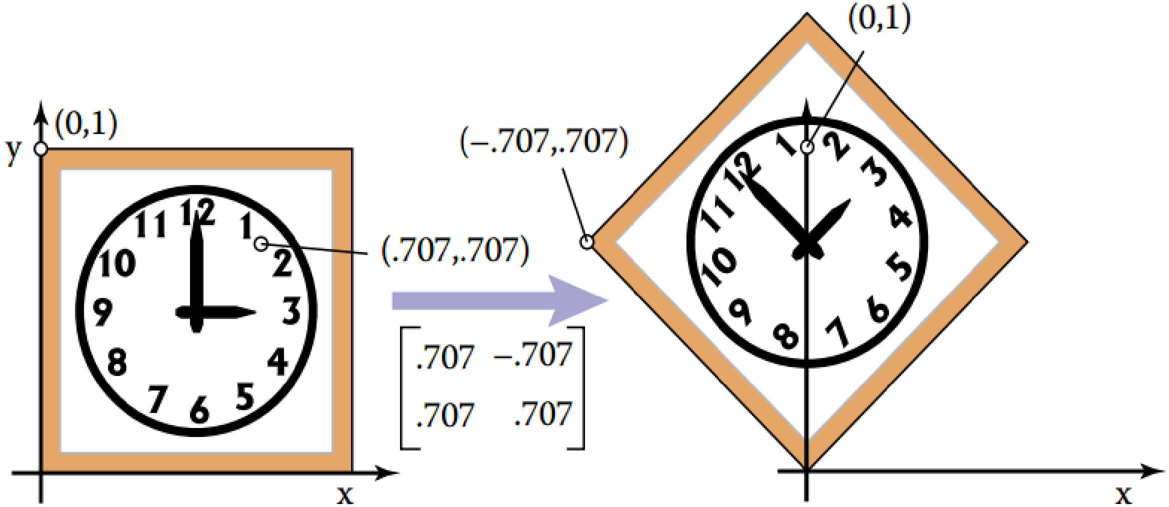
$$\begin{bmatrix} \cos\frac{\pi}{4} & -\sin\frac{\pi}{4} \\ \sin\frac{\pi}{4} & \cos\frac{\pi}{4} \end{bmatrix} = \begin{bmatrix} 0.707 & -0.707 \\ 0.707 & 0.707 \end{bmatrix}$$



Credit: Fundamentals of Computer Graphics 3rd Edition by Peter Shirley, Steve Marschner | http://www.cs.cornell.edu/courses/cs4620/2019fa/

Rotation (8/11)

$$\begin{bmatrix} \cos\frac{\pi}{4} & -\sin\frac{\pi}{4} \\ \sin\frac{\pi}{4} & \cos\frac{\pi}{4} \end{bmatrix} = \begin{bmatrix} 0.707 & -0.707 \\ 0.707 & 0.707 \end{bmatrix}$$

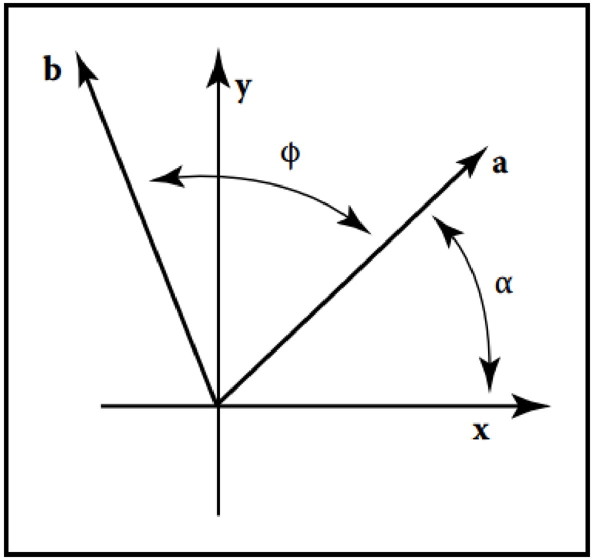


Credit: Fundamentals of Computer Graphics 3rd Edition by Peter Shirley, Steve Marschner | http://www.cs.cornell.edu/courses/cs4620/2019fa/

Rotation (9/11)

$$\text{rotate}(\phi) = \begin{bmatrix} \cos\phi & -\sin\phi \\ \sin\phi & \cos\phi \end{bmatrix}$$

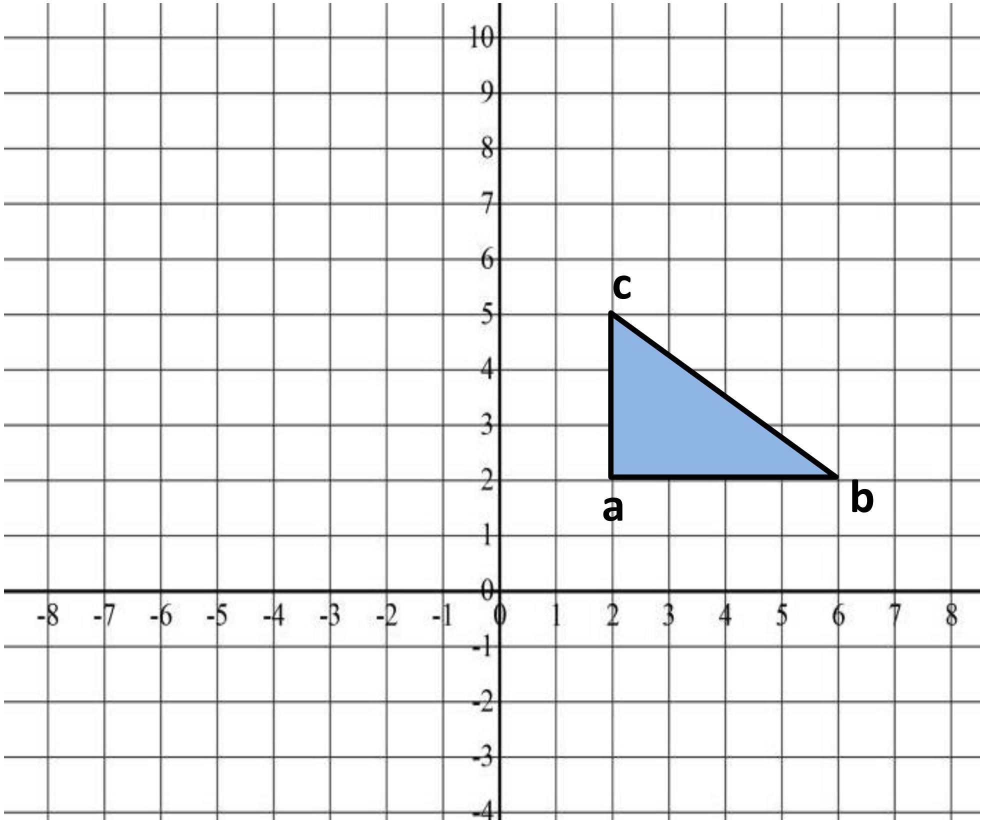
Q: What about – ve angle?



Credit: Fundamentals of Computer Graphics 3rd Edition by Peter Shirley, Steve Marschner | http://www.cs.cornell.edu/courses/cs4620/2019fa/

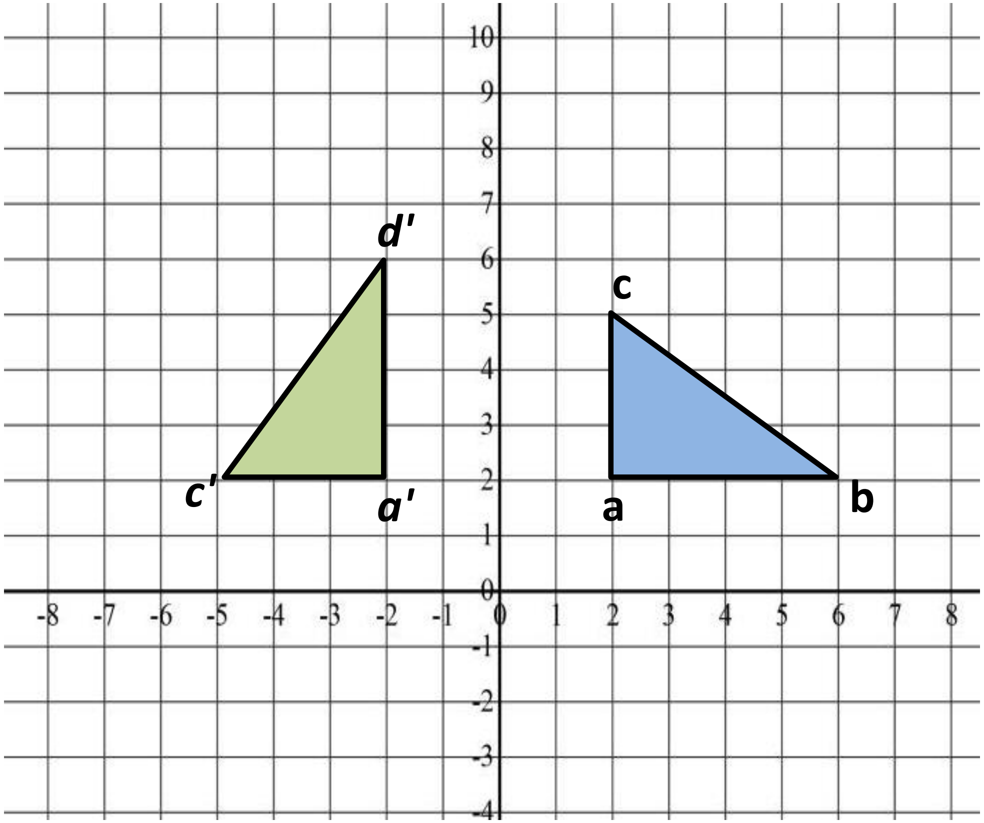
Rotation (10/11)

Rotate(90)



Rotation (11/11)

Rotate(90)



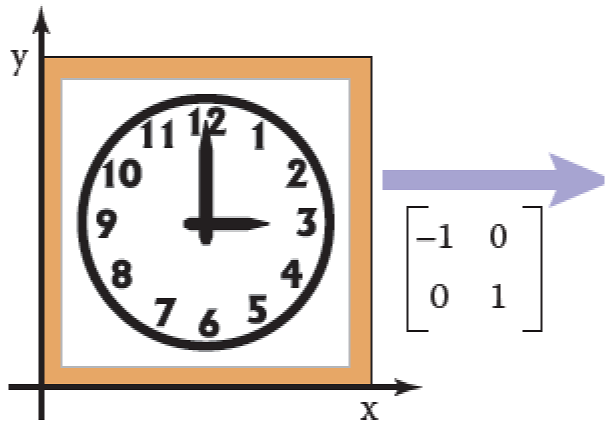
Reflection (1/5)

$$\text{reflect-y} = \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix}, \quad \text{reflect-x} = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$$

Credit: Fundamentals of Computer Graphics 3rd Edition by Peter Shirley, Steve Marschner | http://www.cs.cornell.edu/courses/cs4620/2019fa/

Reflection (2/5)

$$\text{reflect-y} = \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix}, \quad \text{reflect-x} = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$$

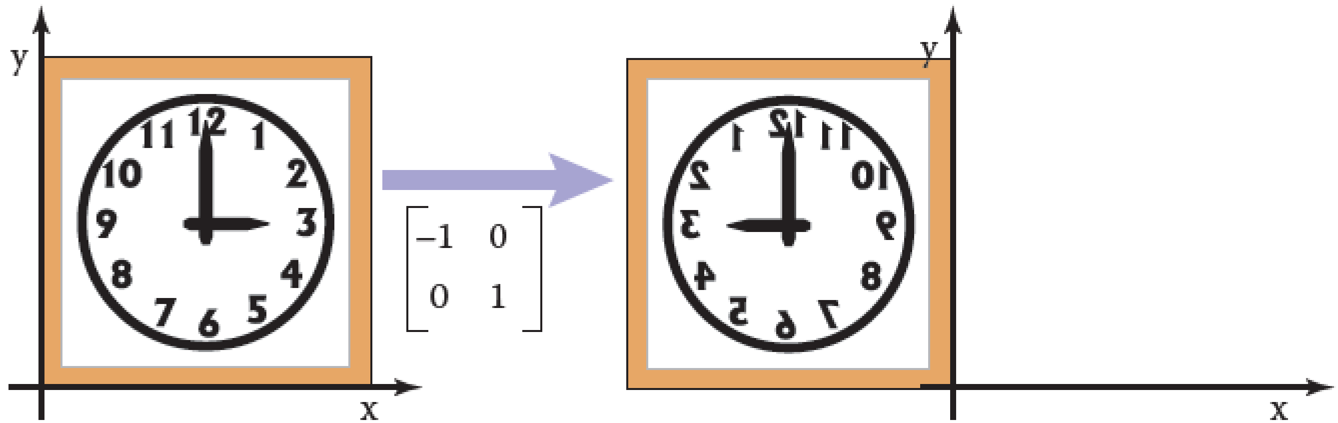


$$\begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix}$$

Credit: Fundamentals of Computer Graphics 3rd Edition by Peter Shirley, Steve Marschner | http://www.cs.cornell.edu/courses/cs4620/2019fa/

Reflection (3/5)

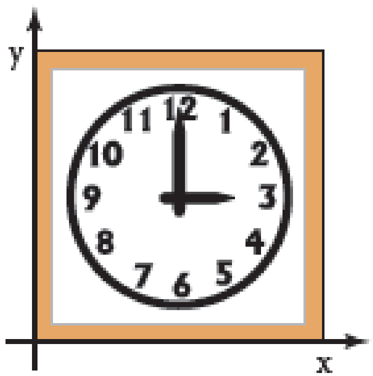
$$\text{reflect-y} = \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix}, \quad \text{reflect-x} = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$$



Credit: Fundamentals of Computer Graphics 3rd Edition by Peter Shirley, Steve Marschner | http://www.cs.cornell.edu/courses/cs4620/2019fa/

Reflection (5/5)

$$\text{reflect-y} = \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix}, \quad \text{reflect-x} = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$$



$$\xrightarrow{\begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}}$$

?

Credit: Fundamentals of Computer Graphics 3rd Edition by Peter Shirley, Steve Marschner | http://www.cs.cornell.edu/courses/cs4620/2019fa/

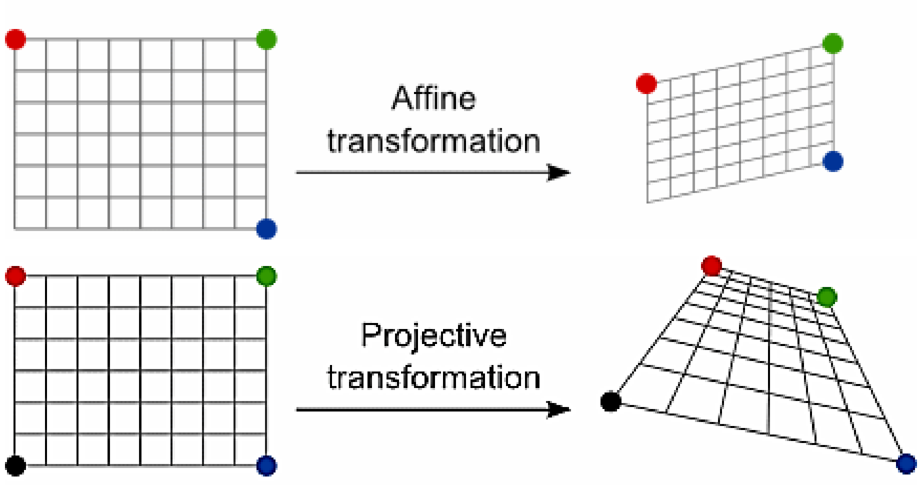
Affine transformation (1/2)

- Maps points to points, lines to lines, planes to planes.
- Preserves the ratio of lengths of parallel line segments.
- Sets of parallel lines remain parallel.
- Does not necessarily preserve angles between lines or distances between points.

Source: https://www.graphicsmill.com/docs/gm/affine-and-projective-transformations.htm | https://en.wikipedia.org/wiki/Affine_transformation

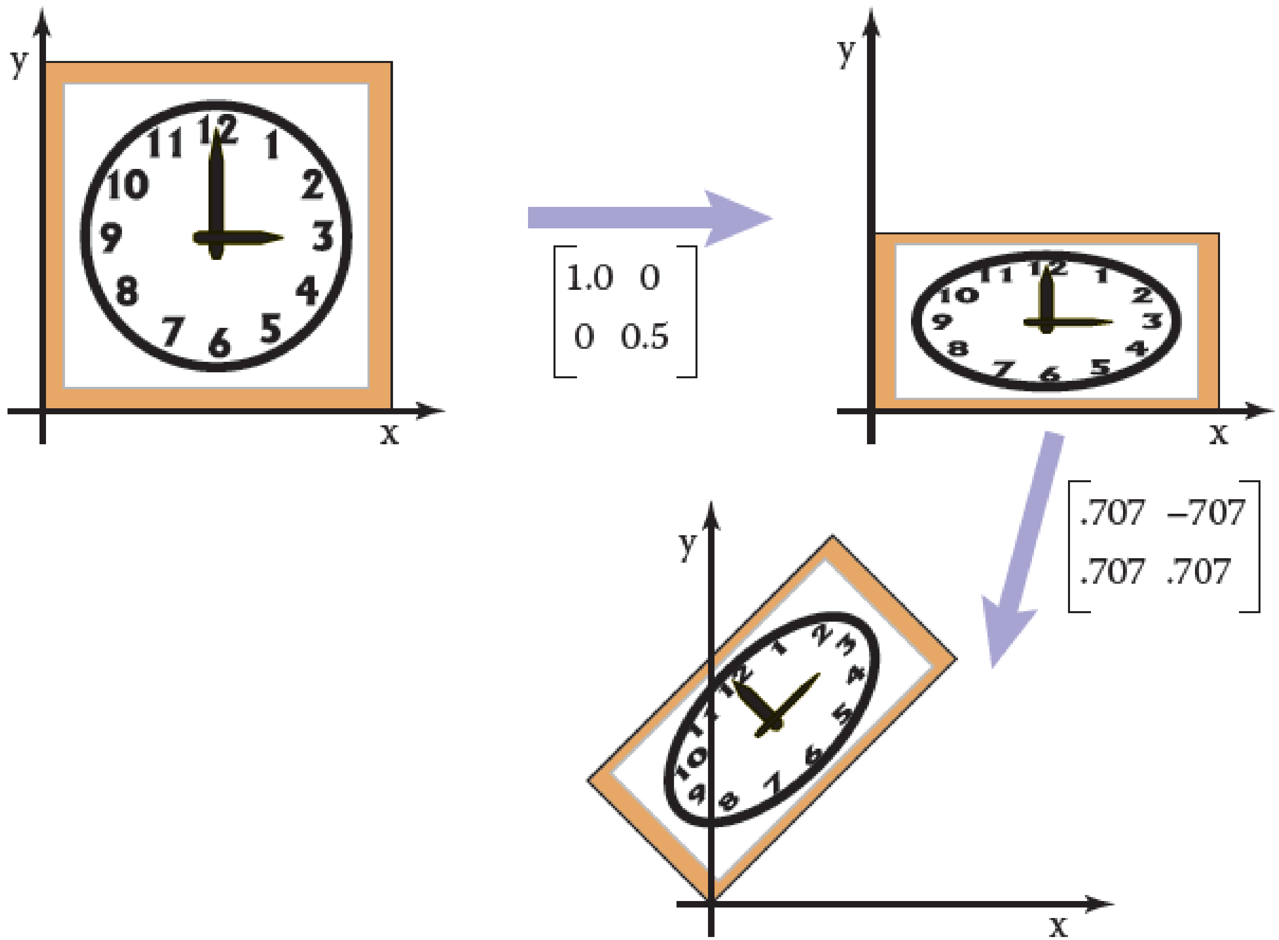
Affine transformation (2/2)

- Maps points to points, lines to lines, planes to planes.
- Preserves the ratio of lengths of parallel line segments.
- Sets of parallel lines remain parallel.
- Does not necessarily preserve angles between lines or distances between points.



Source: https://www.graphicsmill.com/docs/gm/affine-and-projective-transformations.htm | https://en.wikipedia.org/wiki/Affine_transformation

Composition of Transformations (1/11)



Credit: Fundamentals of Computer Graphics 3rd Edition by Peter Shirley, Steve Marschner | http://www.cs.cornell.edu/courses/cs4620/2019fa/

Composition of Transformations (2/11)

- Apply more than one transformation:
  - i.e., for a 2D point v_1 we might want to –
  1. first apply a scale S
  2. then a rotation R.
- This would be done in two steps:
  1. **first, $v_2 = S\ v_1$**
  2. **then, $v_3 = R\ v_2$.**

Credit: Fundamentals of Computer Graphics 3rd Edition by Peter Shirley, Steve Marschner | http://www.cs.cornell.edu/courses/cs4620/2019fa/

Composition of Transformations (3/11)

Therefore –

1. $v_2 = S\ v_1$
2. $v_3 = R\ v_2$
3. $v_3 = R\ (Sv_1)$
4. $v_3 = (RS)\ v_1$ *[matrix multiplication is associative]*
5. $v_3 = Mv_1$ *[Where M=RS]*

Credit: Fundamentals of Computer Graphics 3rd Edition by Peter Shirley, Steve Marschner | http://www.cs.cornell.edu/courses/cs4620/2019fa/

Composition of Transformations (4/11)

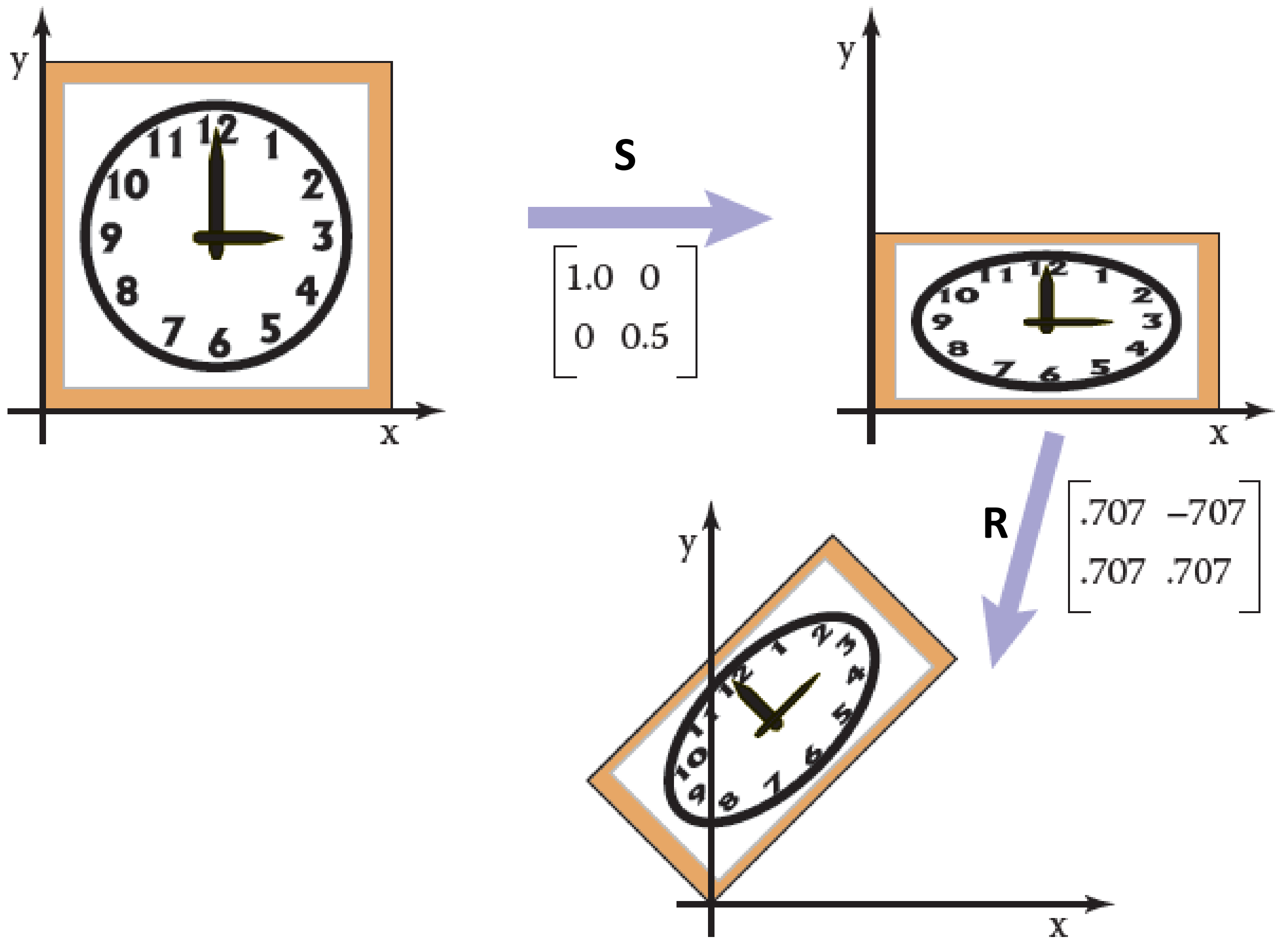
$$\mathbf{v}_{\mathbf{out}} = \mathbf{M}\,\mathbf{v}_{\mathbf{in}}$$

[Where M = R S]

- We can represent the effects of transforming a vector by two matrices in sequence using a single matrix of the same size
  - which we can compute by multiplying the two matrices: **M = RS**

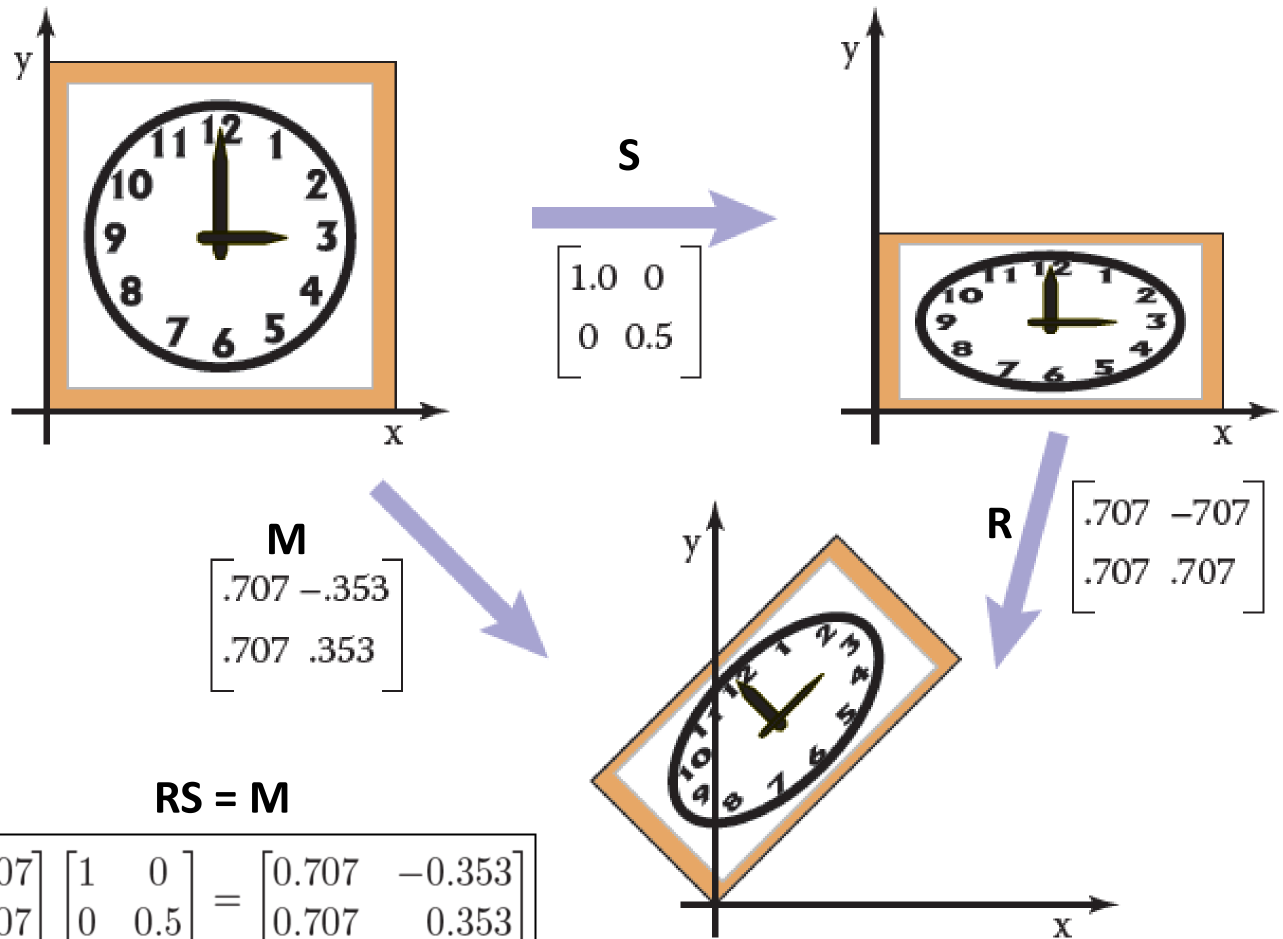
Credit: Fundamentals of Computer Graphics 3rd Edition by Peter Shirley, Steve Marschner | http://www.cs.cornell.edu/courses/cs4620/2019fa/

Composition of Transformations (6/11)



Credit: Fundamentals of Computer Graphics 3rd Edition by Peter Shirley, Steve Marschner | http://www.cs.cornell.edu/courses/cs4620/2019fa/

Composition of Transformations (7/11)



RS = M

$$\begin{bmatrix}0.707 & -0.707\\ 0.707 & 0.707\end{bmatrix}\begin{bmatrix}1 & 0\\ 0 & 0.5\end{bmatrix} = \begin{bmatrix}0.707 & -0.353\\ 0.707 & 0.353\end{bmatrix}$$

Credit: Fundamentals of Computer Graphics 3rd Edition by Peter Shirley, Steve Marschner | http://www.cs.cornell.edu/courses/cs4620/2019fa/

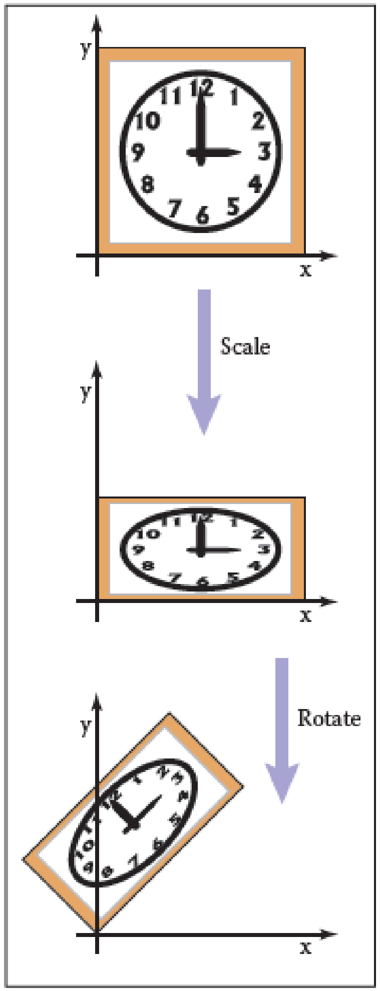
Composition of Transformations (8/11)

- It is very important to remember that these transforms are applied :
  - **from the right side first.**
  - So the matrix M= RS
    - first applies S and then R.

Credit: Fundamentals of Computer Graphics 3rd Edition by Peter Shirley, Steve Marschner | http://www.cs.cornell.edu/courses/cs4620/2019fa/

Composition of Transformations (9/11)

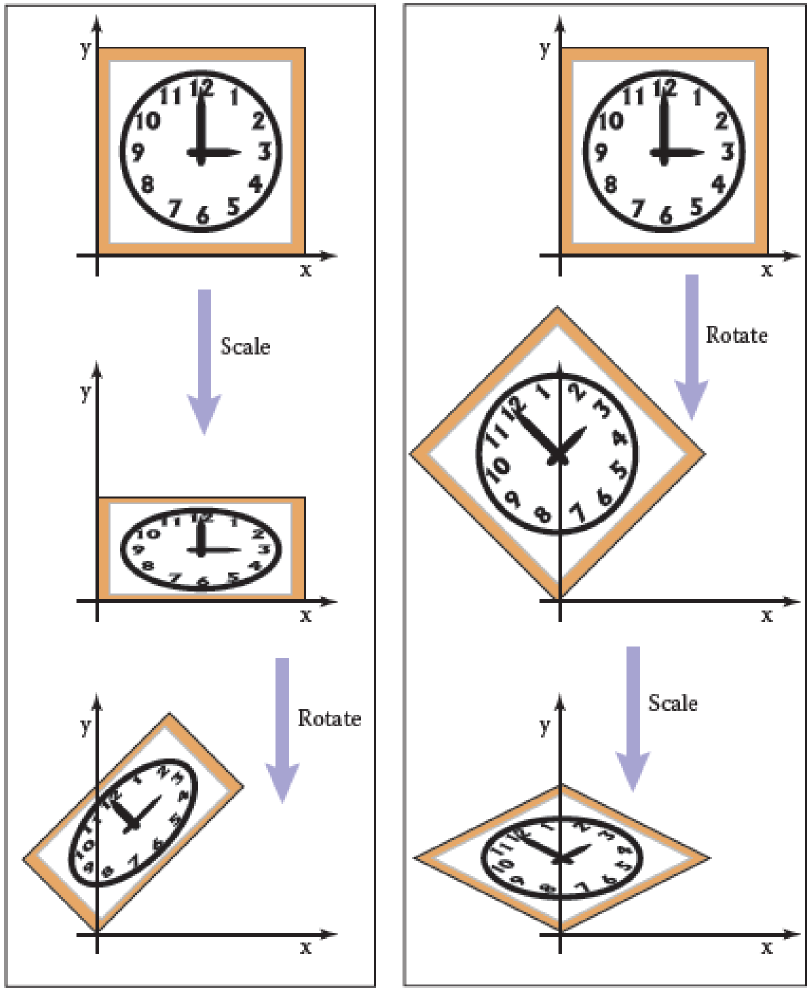
M= RS



Credit: Fundamentals of Computer Graphics 3rd Edition by Peter Shirley, Steve Marschner | http://www.cs.cornell.edu/courses/cs4620/2019fa/

Composition of Transformations (10/11)

M= RS

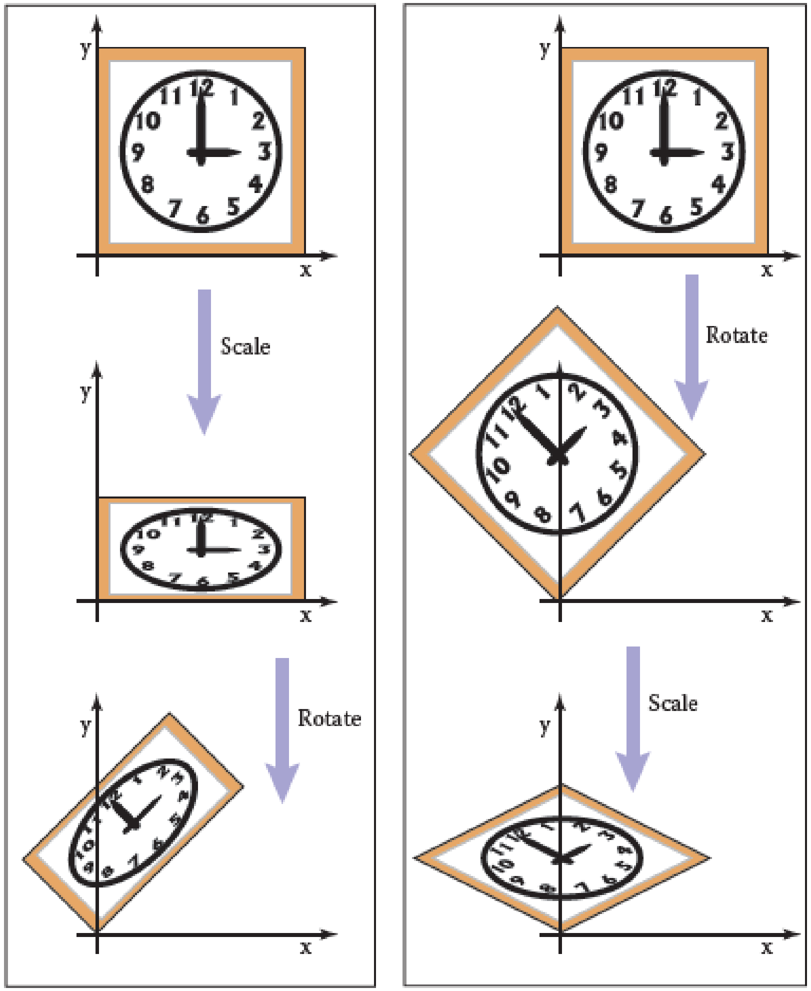


M = **?**

Credit: Fundamentals of Computer Graphics 3rd Edition by Peter Shirley, Steve Marschner | http://www.cs.cornell.edu/courses/cs4620/2019fa/

Composition of Transformations (11/11)

M= RS



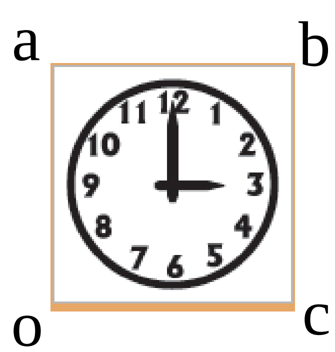
M=

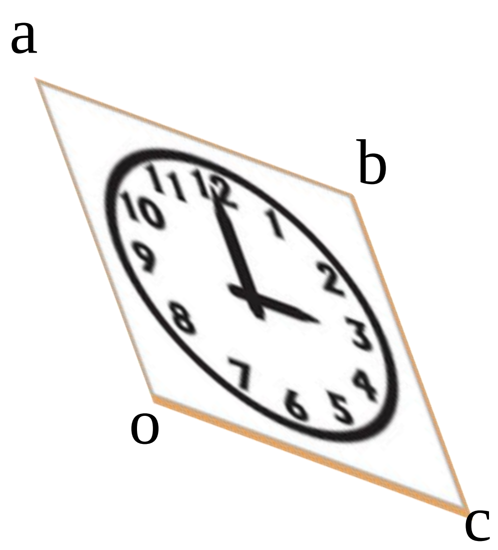
Q: What about more than two transformations: $T_1 \rightarrow T_2 \rightarrow T_3 \dots \rightarrow T_n$.

Credit: Fundamentals of Computer Graphics 3rd Edition by Peter Shirley, Steve Marschner | http://www.cs.cornell.edu/courses/cs4620/2019fa/

Practice Problem - 1

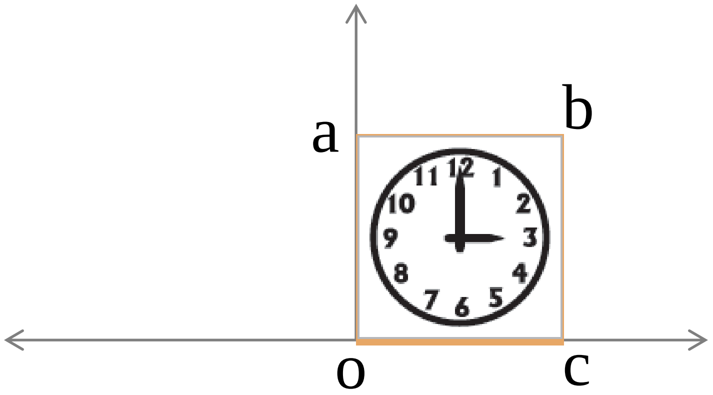
- Stretch the clock by 50% along one of its diagonals
  - so that 8:00 through 1:00 move to the northwest and 2:00 through 7:00 move to the southeast.



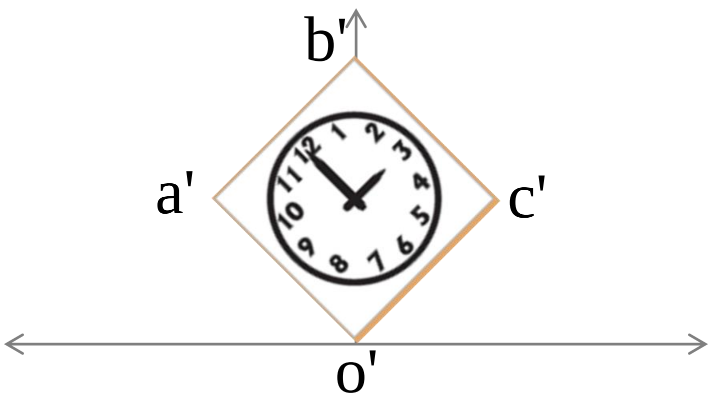


Credit: Fundamentals of Computer Graphics 3rd Edition by Peter Shirley, Steve Marschner | http://www.cs.cornell.edu/courses/cs4620/2019fa/

Practice Problem – 1 (Sol.)

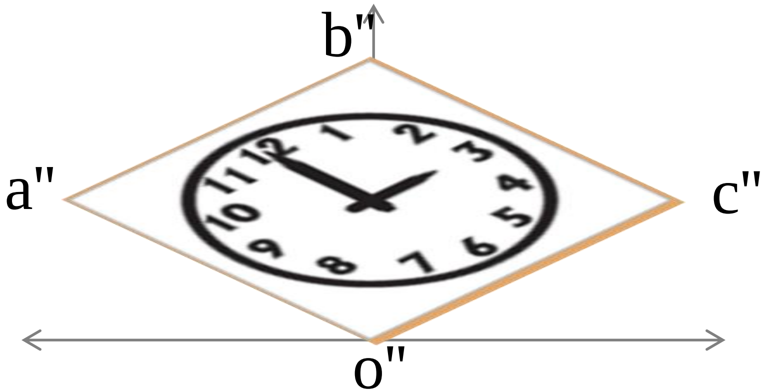


Practice Problem – 1 (Sol.)



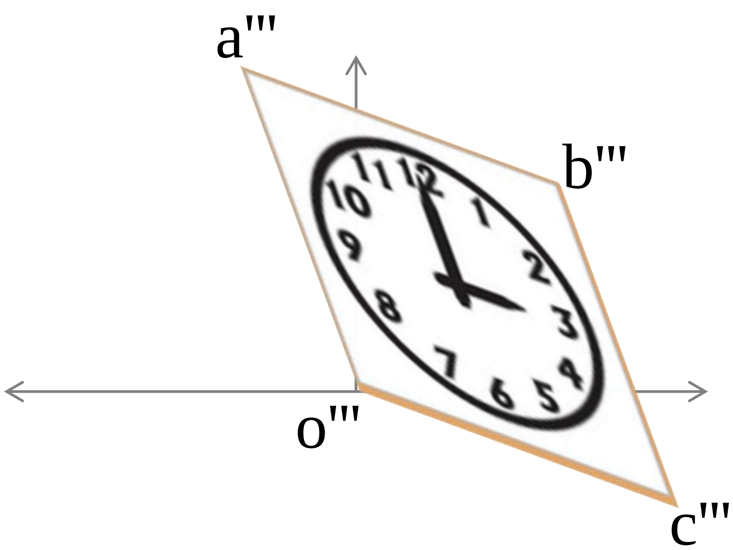
rotate(?)

Practice Problem – 1 (Sol.)



scale(?)

Practice Problem – 1 (Sol.)



rotate(?)

Practice Problem – 1 (Sol.)

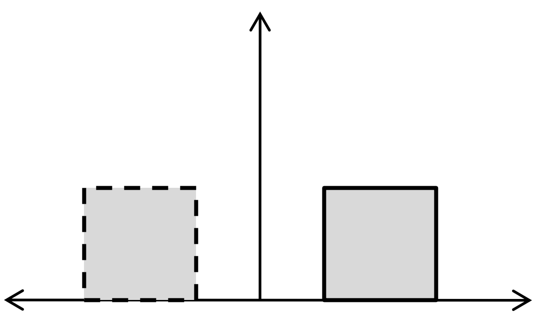
- rotate(45°) → scale(1.5, 1) → rotate(-45°).
  - *Q: Draw the steps*
- $M = R_{(-45^\circ)}\, S_{(1.5,\, 1)}\, R_{(45^\circ)}$

  $= R^T\, S\, R$
  - *Q: Calculate the matrix*

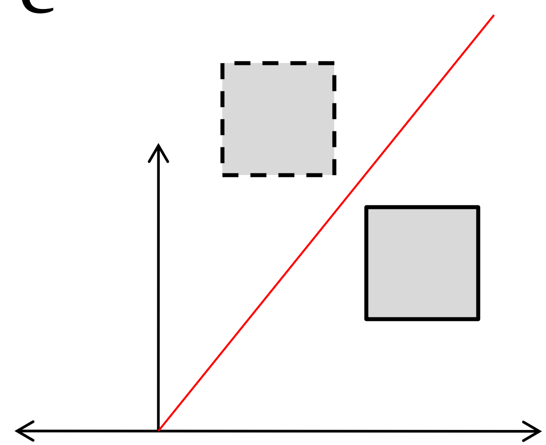
Practice Problem – 2

- Reflect the clock along a line goes through origin:

$$y = mx + c$$



w.r.t y-axis



w.r.t arbitrary line

Thank you