Salahaddin University-Erbil/College of Science
Department of Computer Science & IT

Computer Graphics

Lecture 12

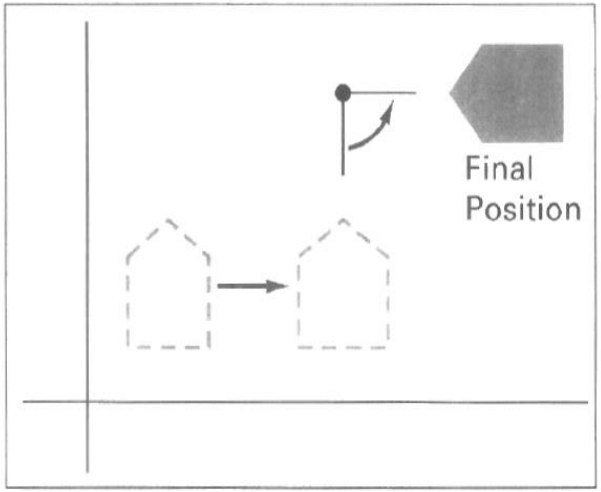
Tarza Hasan

tarza.abdullah@su.edu.krd

Contents

- ❖Matrix Representations and Homogeneous Coordinates

- ❖Composite Transformations
  - ❖ **Rotation** about **a pivot point**
  - ❖ **Scaling** about a **pivot point**



Matrix Representations and Homogeneous Coordinates

- Many graphics applications involve sequences of geometric transformations
  - Animations
  - Design and picture construction applications
- We will now consider matrix representations of these operations
  - Sequences of transformations can be efficiently processed using matrices

Matrix Representations and Homogeneous Coordinates (cont.)

- To produce a sequence of operations, such as scaling followed by rotation then translation, we could calculate the transformed coordinates one step at a time
- A more efficient approach is to combine transformations, without calculating intermediate coordinate values

Matrix Representations and Homogeneous Coordinates (cont.)

- **Multiplicative** and **translational** matrices of a 2D geometric transformation can be combined into a single matrix if we expand the representations to 3 by 3 matrices
  - We can use the third column for translation terms, and all transformation equations can be expressed as matrix multiplications

Matrix Representations and Homogeneous Coordinates (cont.)

- Expand each 2D coordinate (x,y) to three element representation (x_h,y_h,h) called **homogeneous coordinates**
- h is the **homogeneous parameter** such that $x = x_h/h, \quad y = y_h/h,$
- A convenient choice is to choose h = 1

Matrix Representations and Homogeneous Coordinates (cont.)

- 2D Translation Matrix

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

or, $\mathbf{P}' = \mathbf{T}(t_x, t_y) \cdot \mathbf{P}$

Matrix Representations and Homogeneous Coordinates (cont.)

- 2D Rotation Matrix

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} \cos\Theta & -\sin\Theta & 0 \\ \sin\Theta & \cos\Theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

or, **P’** = **R**(θ)·**P**

Matrix Representations and Homogeneous Coordinates (cont.)

- 2D Scaling Matrix

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

or, $\boldsymbol{P'} = \boldsymbol{S}(s_x, s_y) \cdot \boldsymbol{P}$

Inverse Transformations

- 2D Inverse Translation Matrix

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

- By the way: $T^{-1} * T = I$

Inverse Transformations

- 2D Inverse Rotation Matrix

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

- And also:

$$R^{-1} * R = I$$

Inverse Transformations (cont.)

- 2D Inverse Scaling Matrix

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

- Of course: $S^{-1} * S = I$

2D Composite Transformations

- We can setup a sequence of transformations as a **composite transformation matrix** by calculating the product of the individual transformations
- $P' = M_2 \cdot M_1 \cdot P$

  $= M \cdot P$

2D Composite Transformations (cont.)

- Composite 2D Translations
  - If two successive translation are applied to a point P, then the final transformed location P' is calculated as

$$\mathbf{P}' = \mathbf{T}(t_{x_2}, t_{y_2}) \cdot \mathbf{T}(t_{x_1}, t_{y_1}) \cdot \mathbf{P} = \mathbf{T}(t_{x_1} + t_{x_2}, t_{y_1} + t_{y_2}) \cdot \mathbf{P}$$

$$\begin{bmatrix} 1 & 0 & t_{2x} \\ 0 & 1 & t_{2y} \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & t_{1x} \\ 0 & 1 & t_{1y} \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & t_{1x} + t_{2x} \\ 0 & 1 & t_{1y} + t_{2y} \\ 0 & 0 & 1 \end{bmatrix}$$

2D Composite Transformations (cont.)

- Composite 2D Rotations

$$\mathbf{P'} = \mathbf{R}(\theta_1 + \theta_2) \cdot \mathbf{P}$$

$$\begin{bmatrix} \cos\Theta_2 & -\sin\Theta_2 & 0 \\ \sin\Theta_2 & \cos\Theta_2 & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \cos\Theta_1 & -\sin\Theta_1 & 0 \\ \sin\Theta_1 & \cos\Theta_1 & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \cos(\Theta_1 + \Theta_2) & -\sin(\Theta_1 + \Theta_2) & 0 \\ \sin(\Theta_1 + \Theta_2) & \cos(\Theta_1 + \Theta_2) & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

2D Composite Transformations (cont.)

- Composite 2D Scaling

$$\mathbf{S}\left(s_{x_2}, s_{y_2}\right) \cdot \mathbf{S}\left(s_{x_1}, s_{y_1}\right) = \mathbf{S}\left(s_{x_1} \cdot s_{x_2}, s_{y_1} \cdot s_{y_2}\right)$$

$$\begin{bmatrix} s_{2x} & 0 & 0 \\ 0 & s_{2y} & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} s_{1x} & 0 & 0 \\ 0 & s_{1y} & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} s_{1x} \cdot s_{2x} & 0 & 0 \\ 0 & s_{1y} \cdot s_{2y} & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

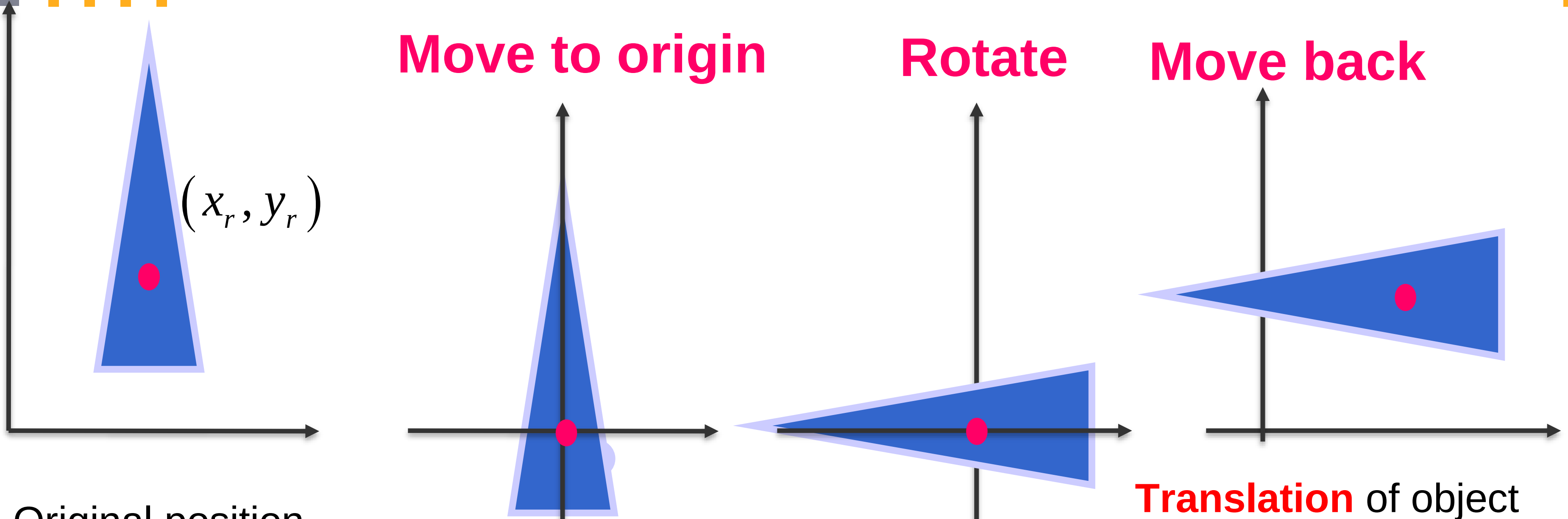
2D Composite Transformations (cont.)

- Don’t forget:
- **Successive translations** are **additive**
- **Successive scalings** are **multiplicative**
  - For example: If we **triple** the size of an object **twice**, the final size is nine (9) times the original
  - 9 times?
  - Why?☺

General Pivot Point Rotation

- Steps:
  1. **Translate** the object so that the pivot point is **moved** to the **coordinate origin.**
  2. **Rotate** the object about **the origin**.
  3. **Translate**T the object so that the pivot point is **returned** to its **original position**.

General Pivot Point Rotation



Original position of object and pivot point

Translation of object so that pivot point (xr,yr) is at origin

Rotation about origin

Translation of object so that the pivot point is returned to position(xr,yr)

19

2D Composite Transformations (cont.)

- **General 2D Pivot-Point Rotation**

$$\begin{bmatrix} 1 & 0 & x_r \\ 0 & 1 & y_r \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \cos\Theta & -\sin\Theta & 0 \\ \sin\Theta & \cos\Theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & -x_r \\ 0 & 1 & -y_r \\ 0 & 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} \cos\Theta & -\sin\Theta & x_r(1-\cos\Theta) + y_r\sin\Theta \\ \sin\Theta & \cos\Theta & y_r(1-\cos\Theta) - x_r\sin\Theta \\ 0 & 0 & 1 \end{bmatrix} * \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$$

X'= xcos θ-ysin θ+xr(1-cos θ)+yrsin θ

Y'=xsin θ+ycos θ+yr(1-cos θ)-xrsin θ

General 2D Pivot-Point Rotation

Consider a triangle ABC A(4,6) B(2,2) C(6,2). Rotate the object 90 degree anticlockwise about the point (3,3).

Xr=3,yr=3,theta=90,cos90=0,sin90=1

$$P' = \begin{bmatrix} \cos\Theta & -\sin\Theta & x_r(1-\cos\Theta) + y_r \sin\Theta \\ \sin\Theta & \cos\Theta & y_r(1-\cos\Theta) - x_r \sin\Theta \\ 0 & 0 & 1 \end{bmatrix} * \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

$$X' = x\cos\theta - y\sin\theta + x_r(1-\cos\theta) + y_r\sin\theta$$

$$Y' = x\sin\theta + y\cos\theta + y_r(1-\cos\theta) - x_r\sin\theta$$

A(4,6) B(2,2) C(6,2)   Xr=3,Yr=3,theta=90   cos90=0,sin90=1

$X' = x\cos\theta - y\sin\theta + xr(1-\cos\theta) + yr\sin\theta$

$Y' = x\sin\theta + y\cos\theta + yr(1-\cos\theta) - xr\sin\theta$

$X' = x.0 - y.1 + xr(1-0) + yr.1 = -y+xr+yr$

$Y' = x.1 + y.0 + yr(1-0) – xr.1 = x+yr-xr$

X'=-y+xr+yr=-6+3-3=0

Y'=x+yr-xr=4+3-3=4

A'(0,4)

X'=-y+xr+yr=-2+3+3=4

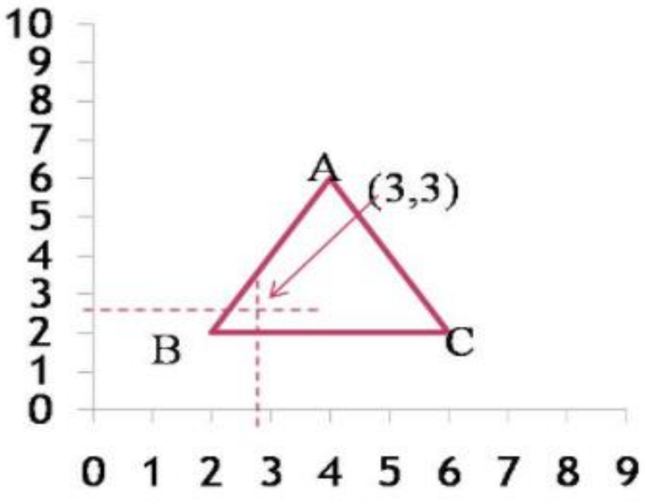
Y'=x+yr-xr=2+3-3=2

B'(4,2)

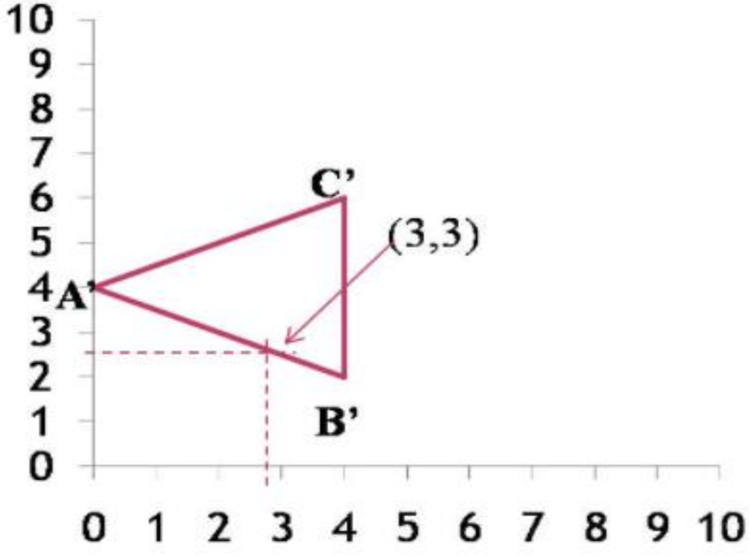
X'=-y+xr+yr=-2+3+3=4

Y'=x+yr-xr=6+3-3=6

C'(4,6)



ORIGINAL TRIANGLE ABC



FINAL TRIANGLE A'B'C'

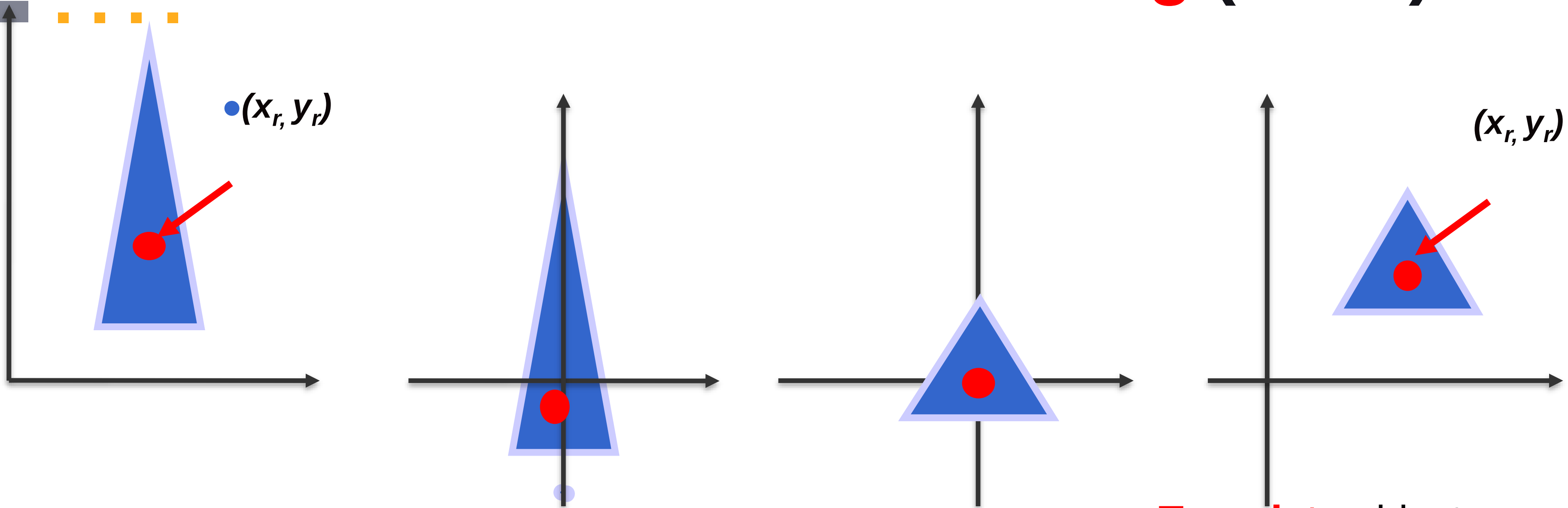
A=(4,6)
B=(2,2)
C=(6,2)

A'=(0,4)
B'=(4,2)
C'=(4,6)

General **Fixed Point Scaling**

- Steps:
  1. **Translate** the object so that the fixed point **coincides** with the **coordinate origin**.
  2. **Scale** the object about the **origin.**
  3. **Translate** the object so that the pivot point is **returned** to its **original position**.

General **Fixed Point Scaling** (cont.)



Original position of object and pivot point

Translation of object so that pivot point (xr,yr) is at origin

Scale with respect to origin

Translate object so that the pivot point is returned to position(xr,yr)

General Fixed Point Scaling (cont.)

- General 2D **Fixed-Point Scaling**:

$$\begin{bmatrix} 1 & 0 & x_f \\ 0 & 1 & y_f \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & -x_f \\ 0 & 1 & -y_f \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} s_x & 0 & x_f(1-s_x) \\ 0 & s_y & y_f(1-s_y) \\ 0 & 0 & 1 \end{bmatrix} * \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$$

$$\mathbf{T}(x_f, y_f) \cdot \mathbf{S}(s_x, s_y) \cdot \mathbf{T}(-x_f, -y_f) = \mathbf{S}(x_f, y_f, s_x, s_y)$$

➡ **X'= x.sx+xf(1-sx)**

Y'=y.sy+yf(1-sy)

Consider a triangle ABC A(4,6) B(2,2)C(6,2). Scale the object about the point (4,4) when sx=3, and sy=3.

xf=yf=4

X'= x.sx+xf(1-sx)

Y'=y.sy+yf(1-sy)

X'=x.sx+xf(1-sx)=12-8=4

Y'=y.sy+yf(1-sy)=18-8=10

A'(4,10)

X'=x.sx+xf(1-sx)=6-8=-2

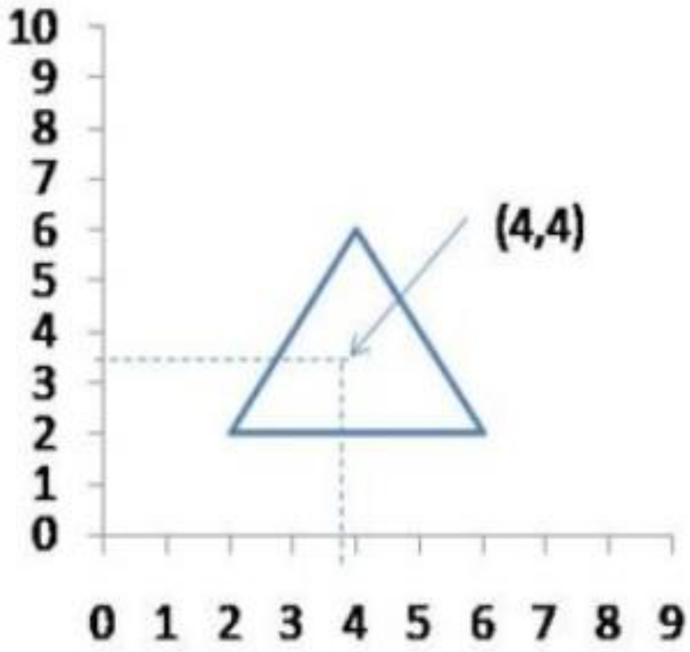
Y'=y.sy+yf(1-sy)=6-8=-2

B'(-2,-2)

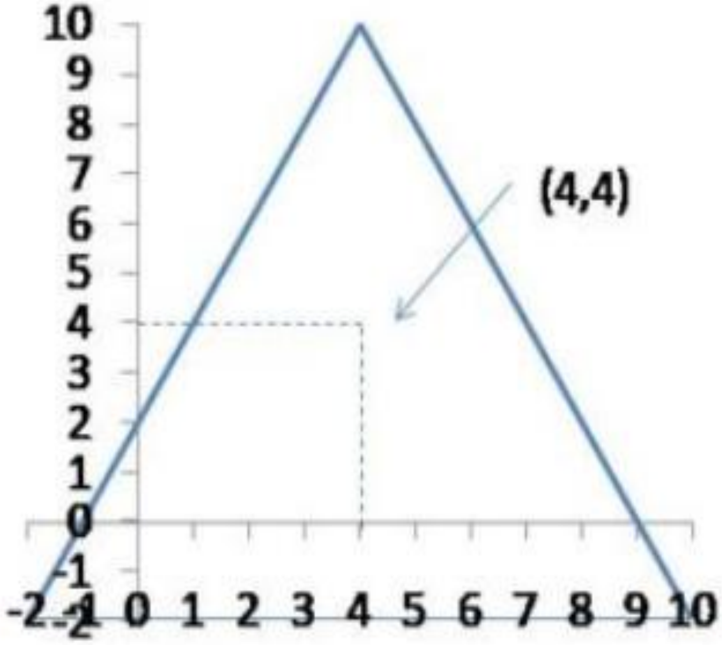
X'=x.sx+xf(1-sx)=18-8=10

Y'=y.sy+yf(1-sy)=6-8=-2

C'(10,-2)



Original Triangle ABC



FINAL TRIANGLE A'B'C'

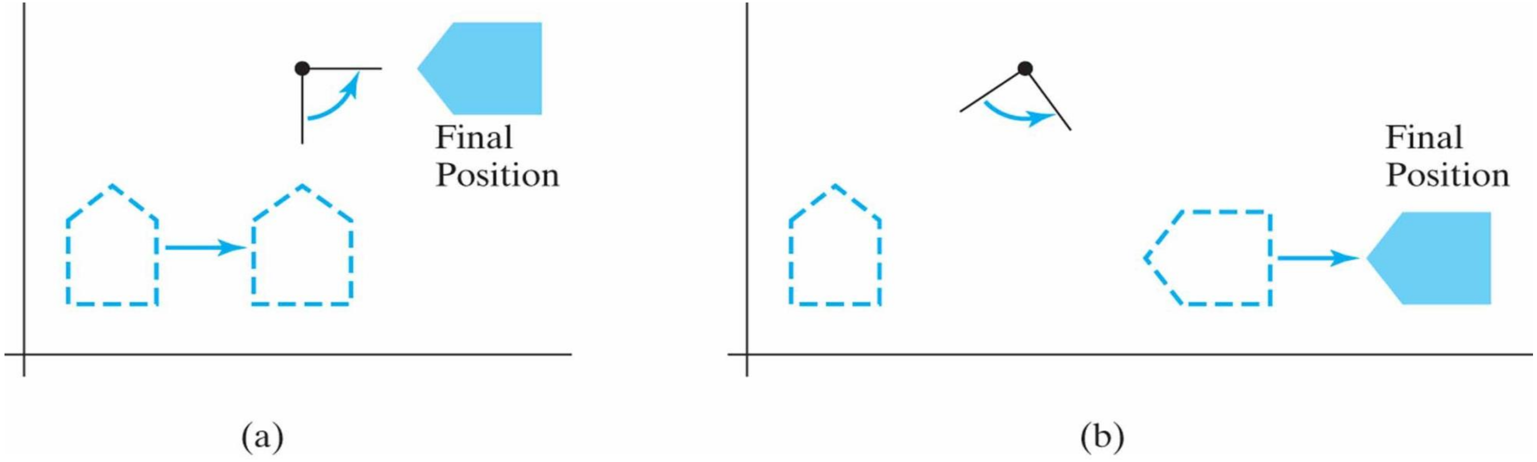
2D Composite Transformations (cont.)

- Matrix Concatenation Properties:
  - Matrix multiplication is **associative !**
    - $M_3 \cdot M_2 \cdot M_1 = (M_3 \cdot M_2) \cdot M_1 = M_3 \cdot (M_2 \cdot M_1)$
    - A composite matrix can be created by multiplicating **left-to-right (premultiplication)** or **right-to-left (postmultiplication)**
  - Transformation matrix multiplication may not be **commutative** !
    - $M_2 \cdot M_1 \neq M_1 \cdot M_2$

2D Composite Transformations (cont.)

- Matrix Concatenation Properties:
    - Two **successive rotations**
    - Two **successive translations**
    - Two **successive scalings**
  - ***are*** commutative!
- Why?☺ ☺

Reversing the order



A sequence of transformations is performed may affect the transformed position of an object.

In **(a),** an object **is first translated in the x direction**, then **rotated counterclockwise through an angle of 45°.**

In **(b),** the object **is first rotated 45° counterclockwise**, **then translated in the *x* direction**