COMPUTAÇÃO GRÁFICA

Transformações Geométricas
Parte I

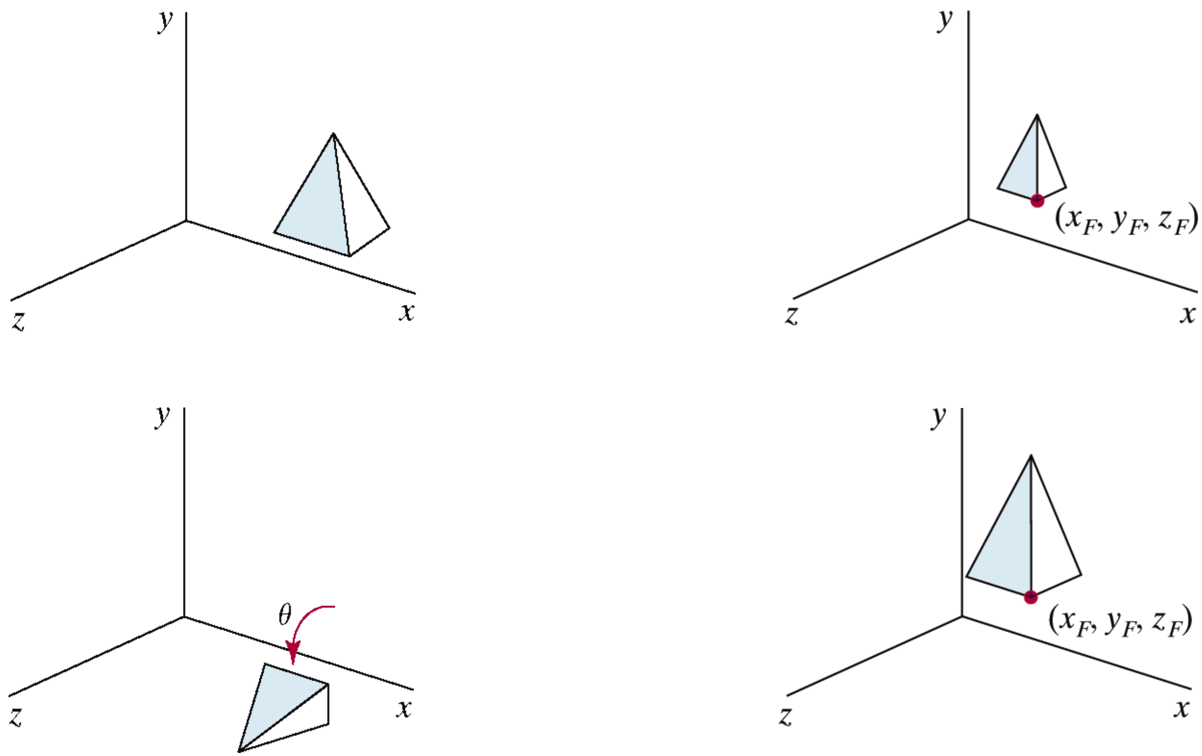
CURSO DE CIÊNCIA DA COMPUTAÇÃO
UNIPAC BARBACENA
PROFESSOR NAIRON NERI SILVA

Sumário

- Tópicos da aula de hoje:
  - Por que transformações?
  - Transformações
    - Translação
    - Escala
    - Rotação
  - Coordenadas Homogêneas
  - Combinando transformações

Por que transformações?

- Em CG, uma vez tendo um objeto definido, “transformações” são utilizadas para mover este objeto, mudar seu tamanho (escala) ou rotacioná-lo.

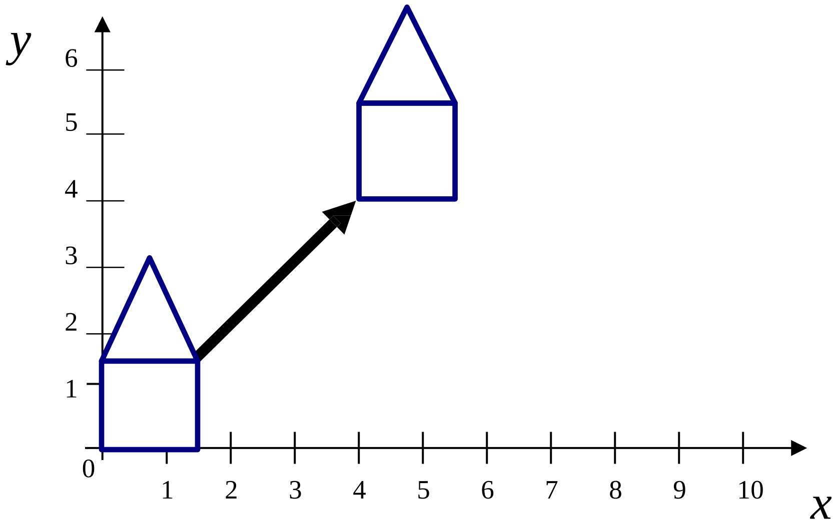


Images taken from Hearn & Baker, “Computer Graphics with OpenGL”(2004)

Translação

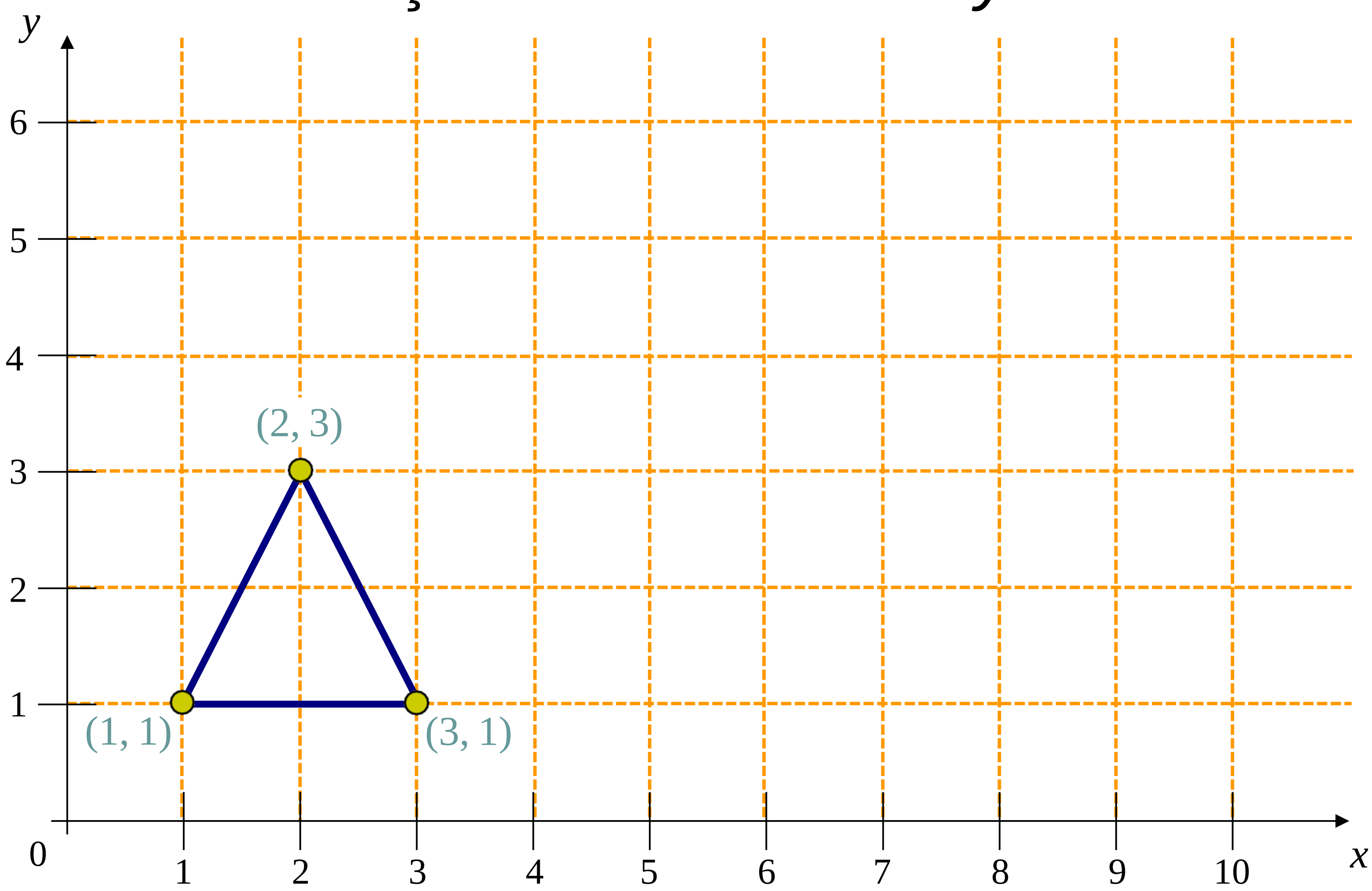
- Move um objeto de uma posição para outra

$$x_{new} = x_{old} + dx \qquad y_{new} = y_{old} + dy$$



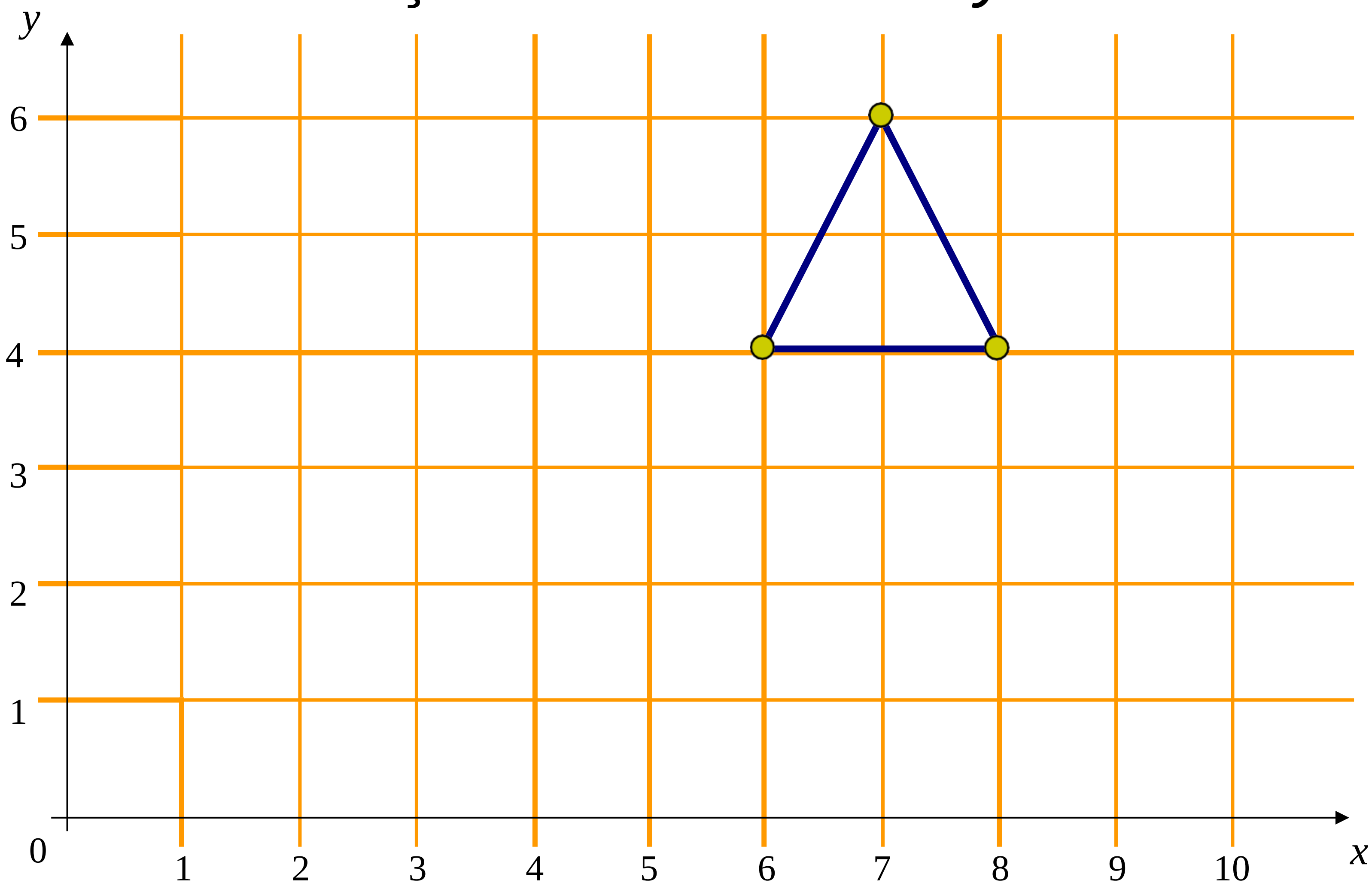
Exemplo de Translação

- O que acontecerá ao objeto se aplicarmos uma translação com *dx* = 5 e *dy* = 3



Exemplo de Translação

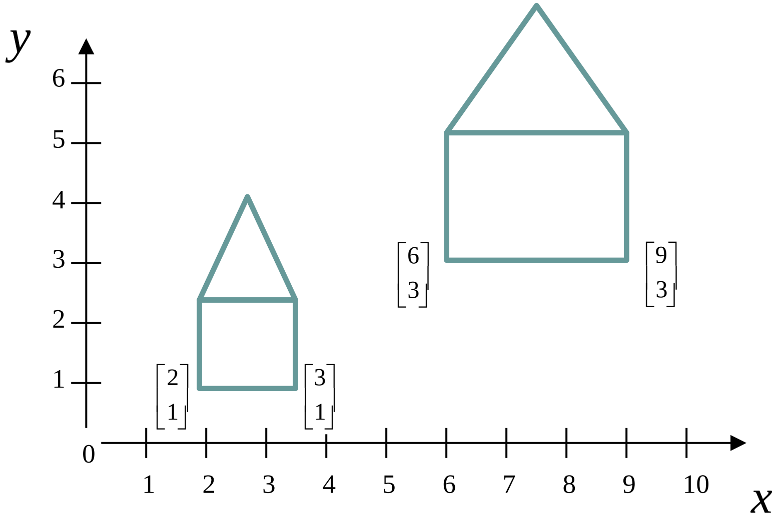
- O que acontecerá ao objeto se aplicarmos uma translação com $dx = 5$ e $dy = 3$



Escala

- Escalas multiplicam todas as coordenadas
- **Atenção:** Objetos podem mudar de lugar ao aplicarmos operações de escala

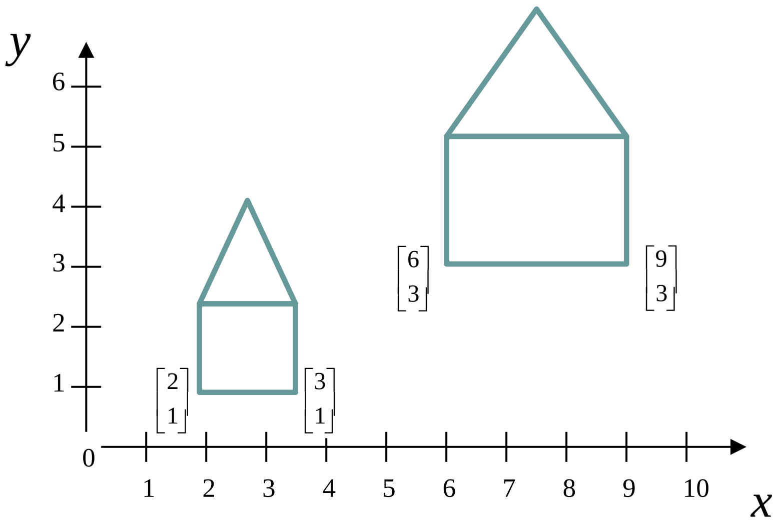
$$x_{new} = Sx \times x_{old} \qquad y_{new} = Sy \times y_{old}$$



Escala

- Escalas multiplicam todas as coordenadas
- **Atenção:** Objetos podem mudar de lugar ao aplicarmos operações de escala

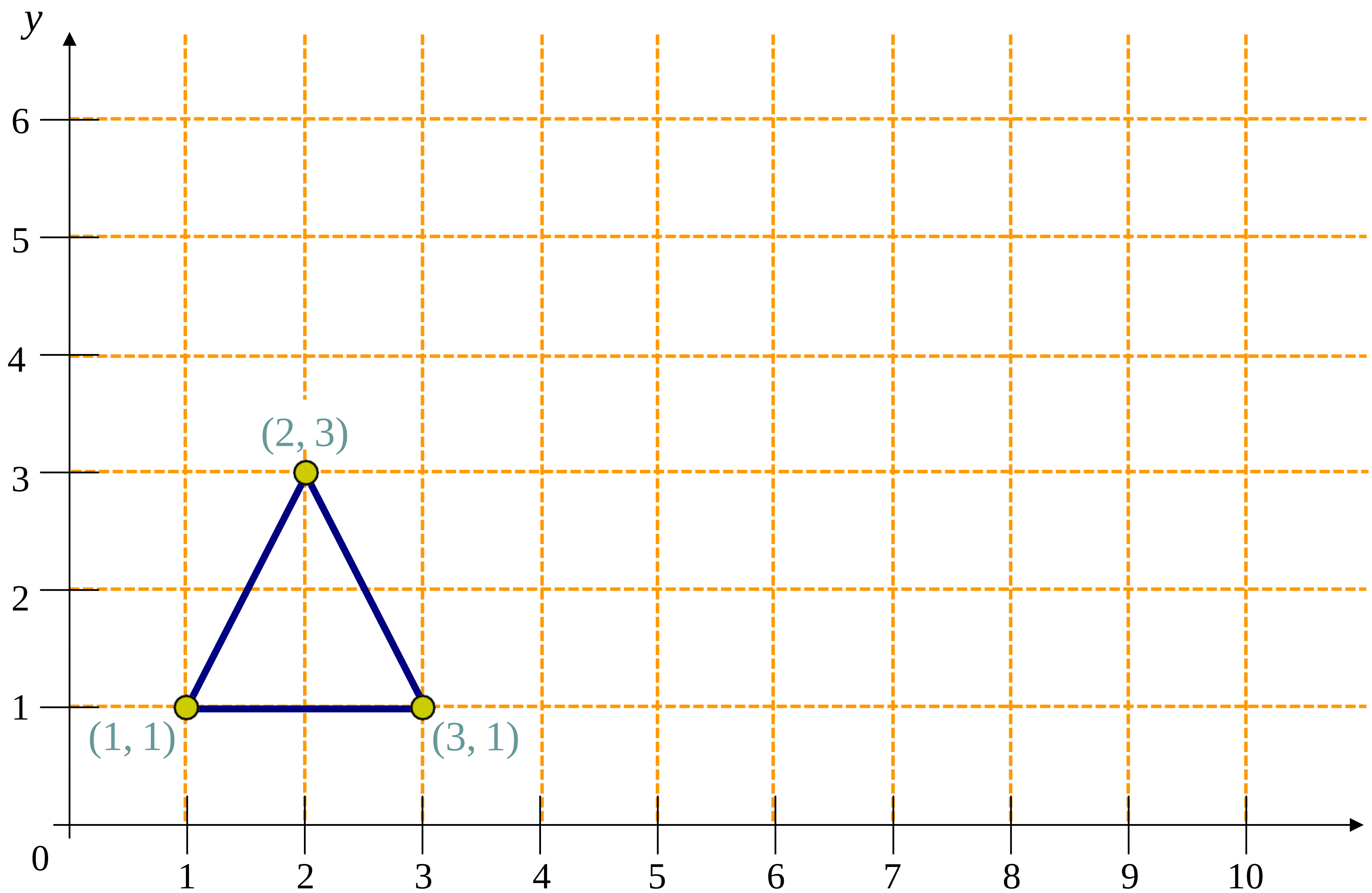
$$x_{new} = Sx \times x_{old} \qquad y_{new} = Sy \times y_{old}$$



Como podemos fazer para que objetos não sofram alteração de lugar?

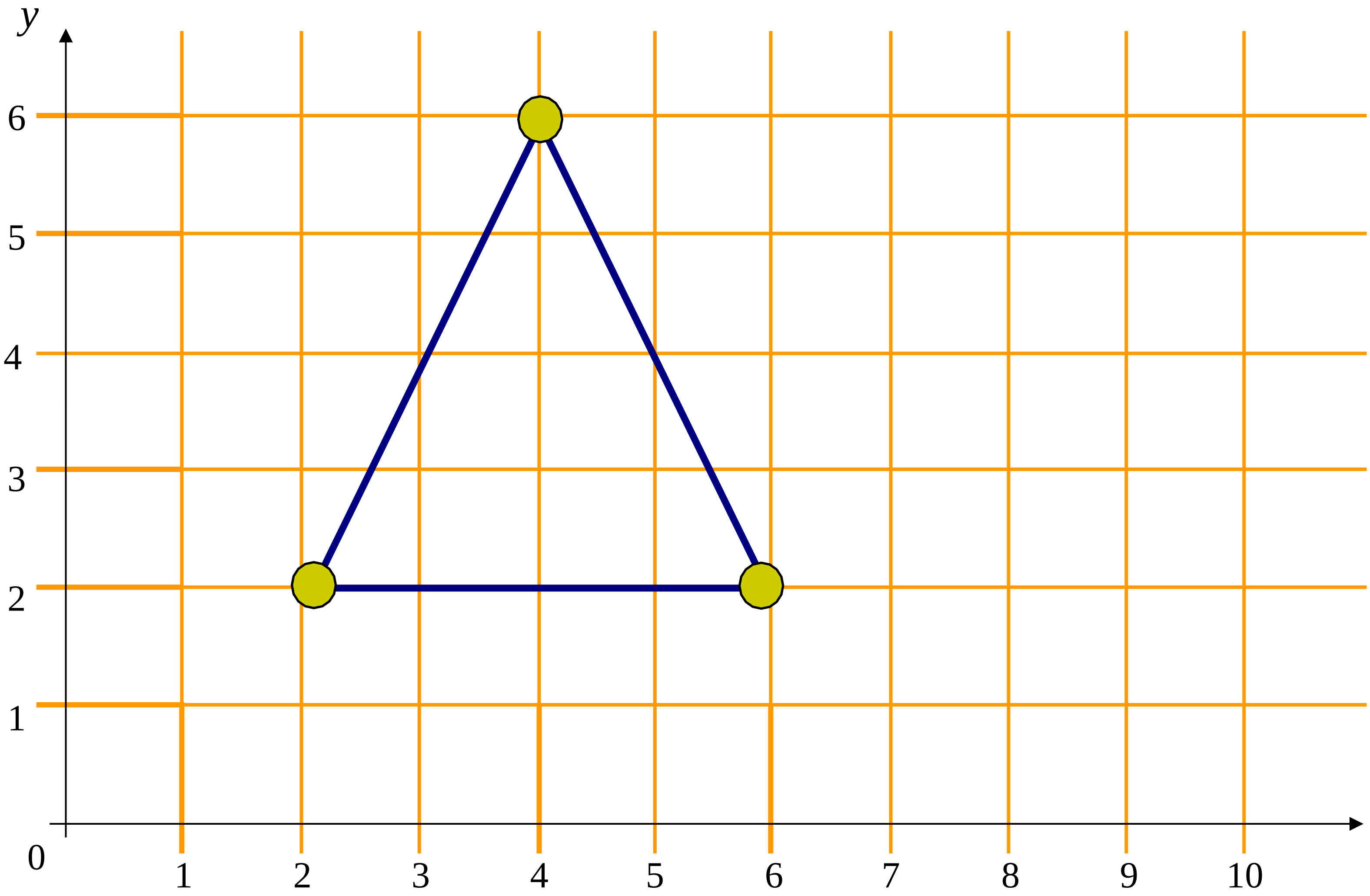
Exemplo de Escala

- O que acontecerá ao objeto se aplicarmos uma escala com sx = 2 e sy = 2



Exemplo de Escala

- O que acontecerá ao objeto se aplicarmos uma escala com $sx = 2$ e $sy = 2$



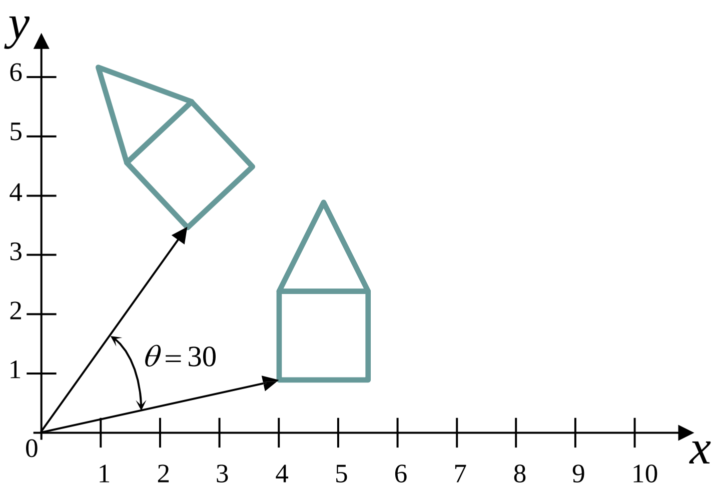
Rotação

- Rotaciona todas as coordenadas por um ângulo específico

$$x_{new} = x_{old} \times \cos\theta - y_{old} \times \sin\theta$$

$$y_{new} = x_{old} \times \sin\theta + y_{old} \times \cos\theta$$

- Pontos sempre serão rotacionados em relação a origem



Rotação

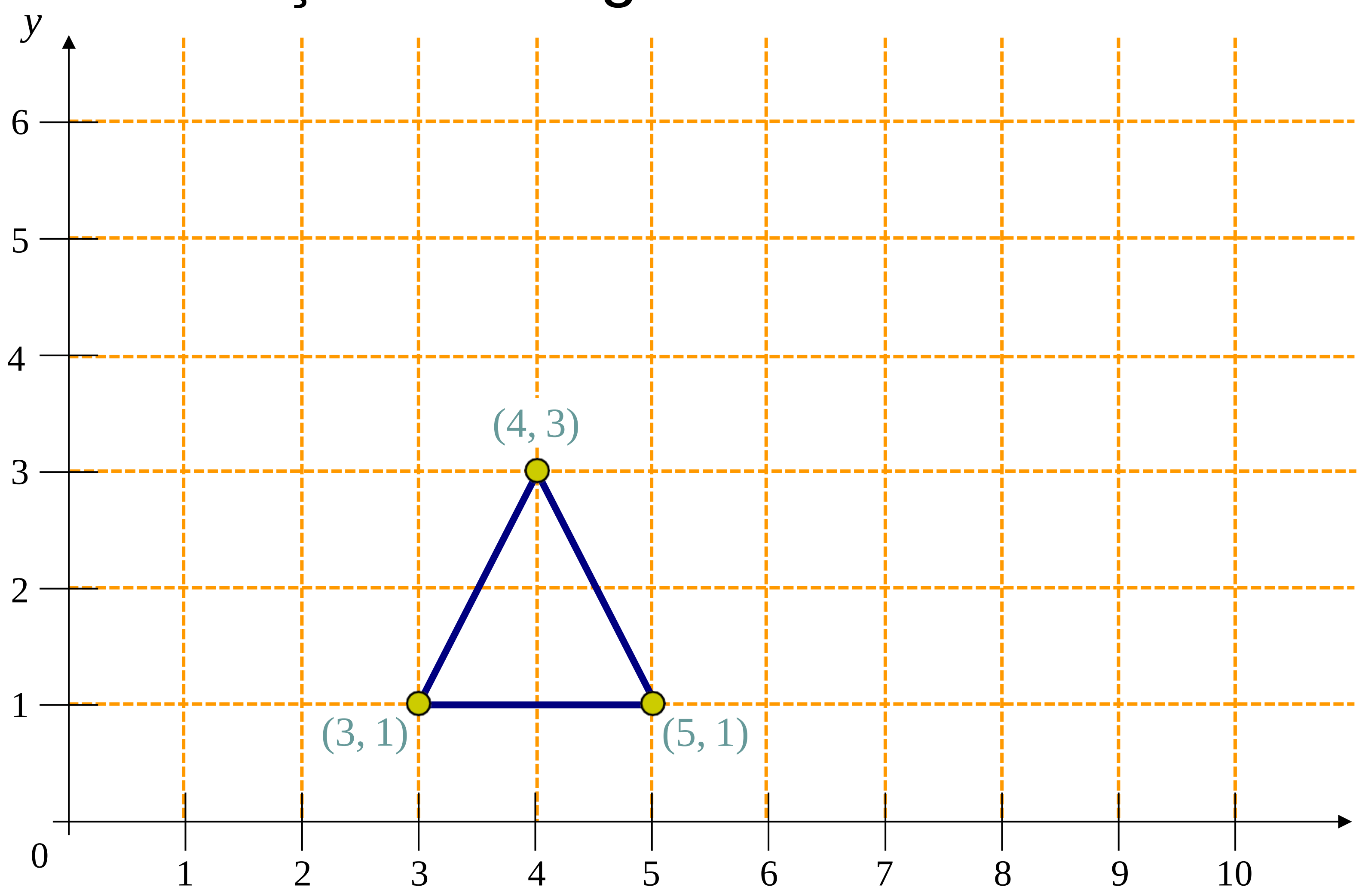
- **Atenção!**
  Muitas linguagens (como C e C++) já possuem funções trigonométricas implementadas. Nestas funções, normalmente, o ângulo a ser passado como parâmetro deve estar em **radianos** e não em **graus**.

- Para converter de graus (g) para radianos (r), utilize a seguinte fórmula:

$$r = \frac{g \times \pi}{180}$$

Exemplo de Rotação

- O que acontecerá ao objeto se aplicarmos uma rotação de 30 graus



Exemplo de Rotação

Primeiro, vamos calcular o ângulo em radianos (no exercício *g = 30*)

$$r = g * PI / 180 = 0,5235$$

Vamos agora calcular as funções trigonométricas

$$cos(0.5235)=0.866$$

$$sin(0.5235)=0.5$$

Recapitulando a fórmulas

$$x_{new} = x_{old} \times \cos\theta - y_{old} \times \sin\theta$$

$$y_{new} = x_{old} \times \sin\theta + y_{old} \times \cos\theta$$

Exemplo de Rotação

Basta agora calcularmos os novos pontos

*x’(3, 1) = 3*0.866 – 1*0.5 = 2.598 – 0.5 =* ***2.098***

*y’(3, 1) = 3*0.5 + 1*0.866 = 1.5 + 0.866 =* ***2.366***

*x’(5, 1) = 5*0.866 – 1*0.5 = 4.33 – 0.5 =* ***3.83***

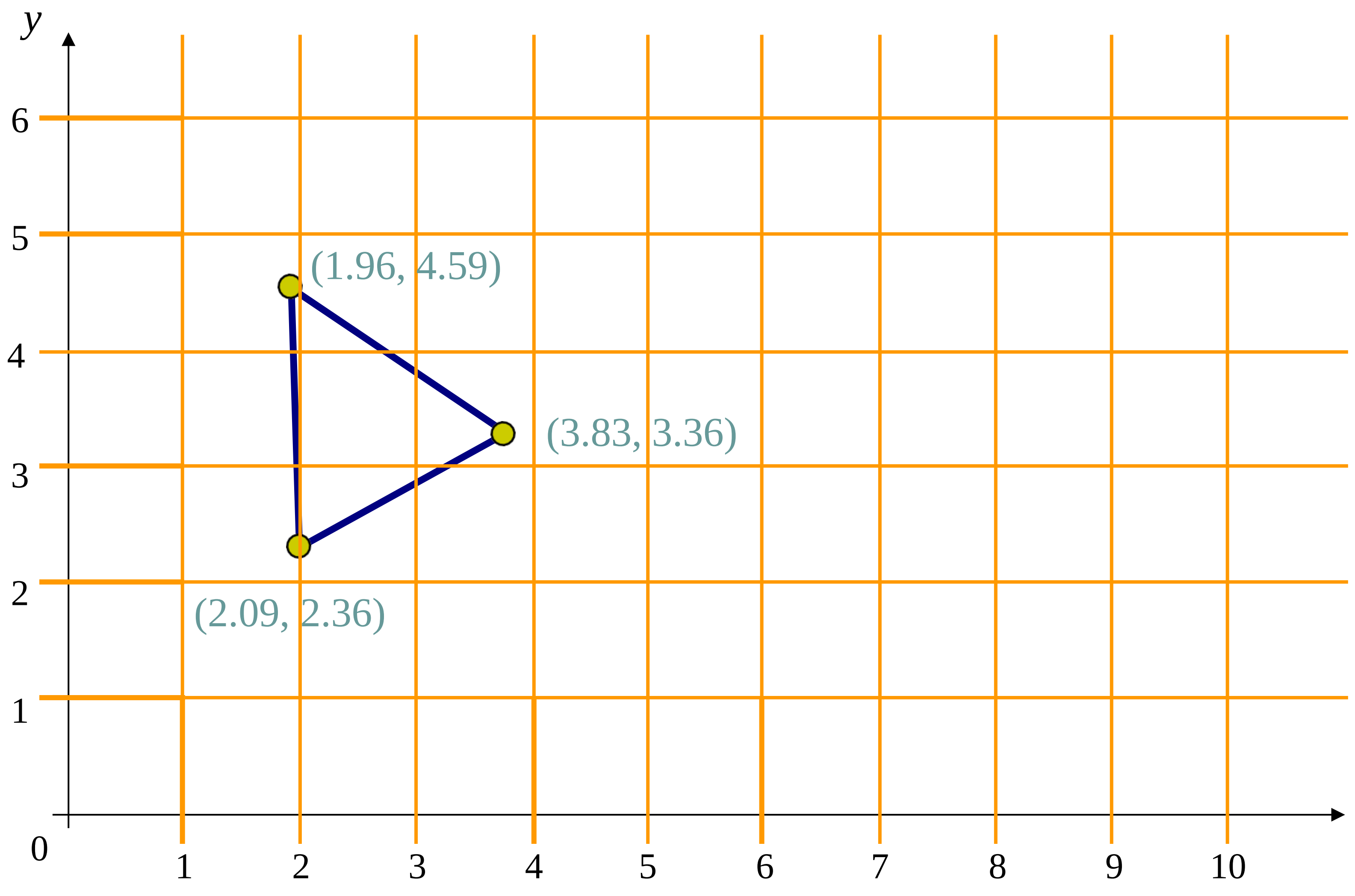
*y’(5, 1) = 5*0.5 + 1*0.866 = 2.5 + 0.866 =* ***3.366***

*x’(4, 3) = 4*0.866 – 3*0.5 = 3.46 – 1.5 =* ***1.96***

*y’(4, 3) = 4*0.5 + 3*0.866 = 2.0 + 2.598 =* ***4.598***

Exemplo de Rotação

- Resultado



Coordenadas Homogêneas

- Um ponto (x, y) pode ser reescrito em **coordenadas homogeneas** como (x_h, y_h, h)
- O **parâmetro** h é um valor diferente de zero tal que:

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

- Deste modo, o ponto cartesiano (x, y) corresponde à uma infinidade de triplas (hx, hy, h), incluindo o caso particular de $(x, y, 1)$

Por que usar coordenadas homogêneas?

- As operações de translação, rotação e escala podem ser facilmente executadas com o uso de matrizes.

- No entanto, enquanto as operações de rotação e escala podem ser concatenadas numa única matriz (pela multiplicação prévia de suas respectivas matrizes) as operações de translação ainda têm de ser conduzidas em separado, uma vez que sua aplicação depende de uma soma matricial...

Por que usar coordenadas homogêneas?

- Usando coordenadas homogêneas, representaremos transformações em 2D através de matrizes 3 x 3 e transformações em 3D através de matrizes 4 x 4.

- Com isso, poderemos concatenar todas as matrizes de transformação numa única matriz global de transformação que será aplicada a todos os pontos do desenho.

Lembrando de Multiplicação de Matrizes

- Lembrando como a multiplicação de matrizes é calculada…

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

Escala

- A escala de um ponto por (s_x, s_y) pode ser reescrita matricialmente como:

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

Rotação

- A rotação de um ponto em relação a origem teria a seguinte representação:

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

Translação

- Finalmente a translação de um ponto por *(dx, dy)* pode ser reescrita matricialmente como:

$$\begin{bmatrix} 1 & 0 & dx \\ 0 & 1 & dy \\ 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} x+dx \\ y+dy \\ 1 \end{bmatrix}$$

Transformações Inversas

- Transformações podem ser facilmente revertidas através de operações inversas

$$T^{-1} = \begin{bmatrix} 1 & 0 & -dx \\ 0 & 1 & -dy \\ 0 & 0 & 1 \end{bmatrix}$$

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

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

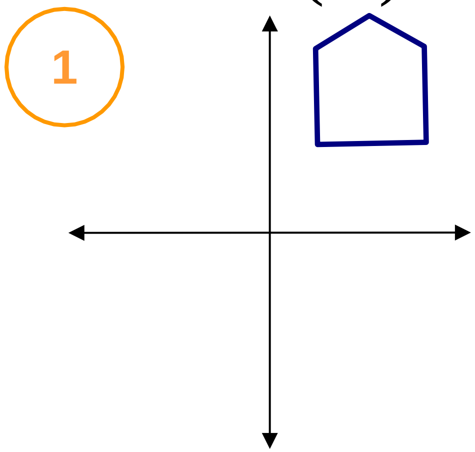
Combinando Transformações

- Como já visto, várias transformações podem ser combinadas em uma única matriz para facilitar o processamento
  - Permitido pelo fato de usarmos coordenadas homogêneas

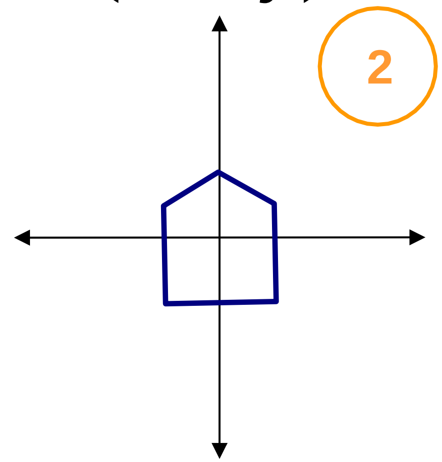
- Imagine rotacionar um polígono em volta de um ponto que não seja a origem:
  - Translade o centro do polígono para a origem
  - Rotacione normalmente em relação a origem
  - Faça a translação inversa

Combinando Transformações (cont…)

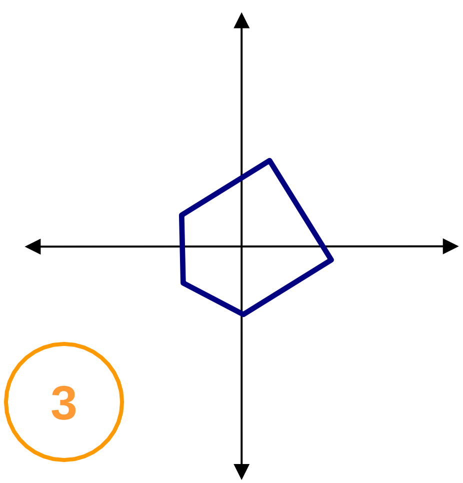
$Casa\,(C)$



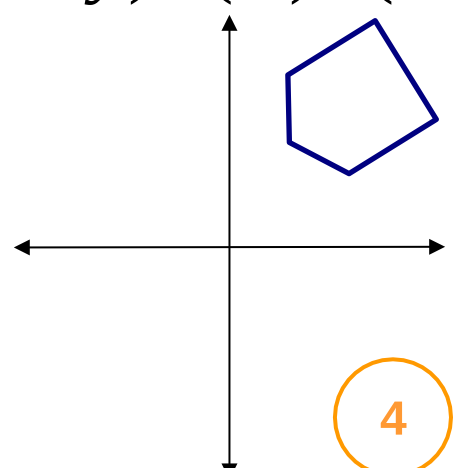
$T(dx, dy)C$



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

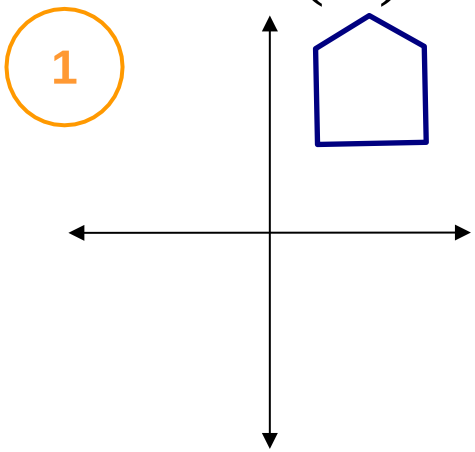


$T(-dx, -dy)R(\theta)T(dx, dy)C$



Combinando Transformações (cont…)

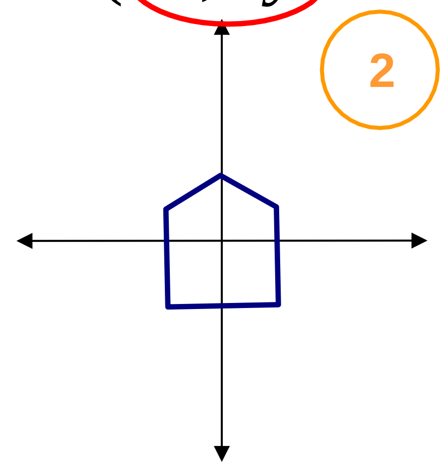
$Casa\,(C)$



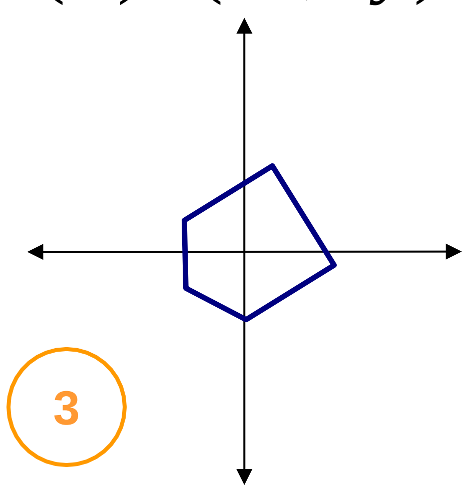
$T(dx, dy)C$

Exemplo:

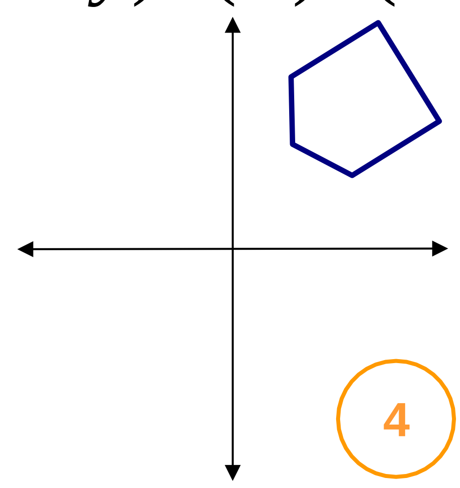
$T(-2,-4)$



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

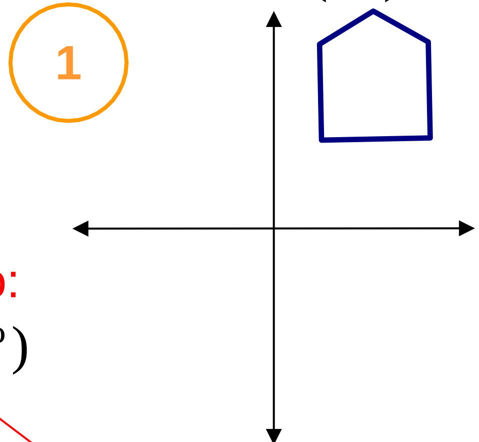


$T(-dx,-dy)R(\theta)T(dx, dy)C$

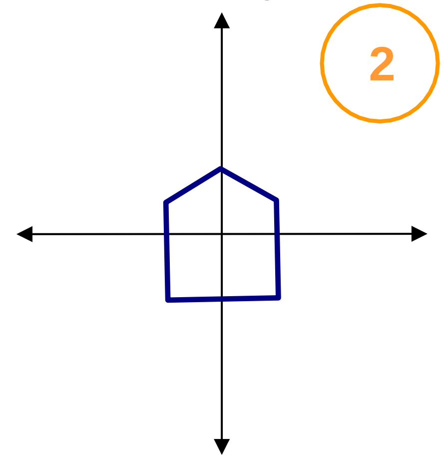


Combinando Transformações (cont…)

$$Casa\,(C)$$



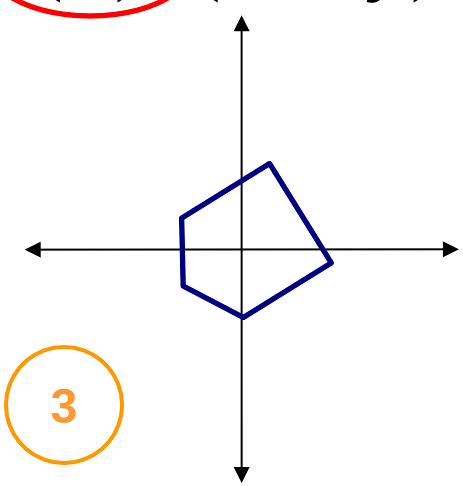
$$T(dx, dy)C$$



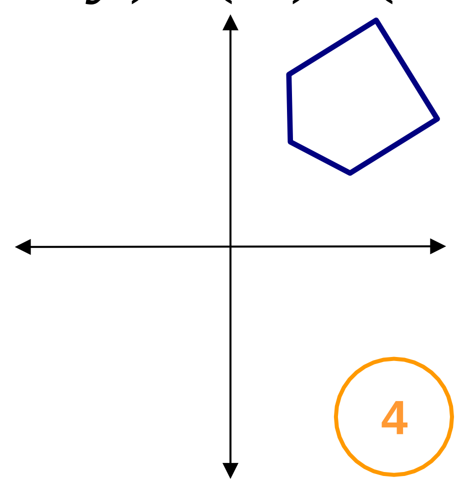
Exemplo:

$R(135^{o})$

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



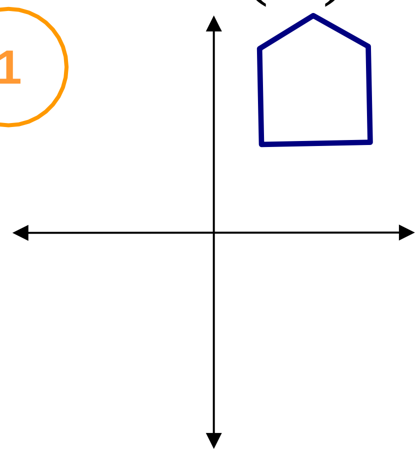
$$T(-dx, -dy)R(\theta)T(dx, dy)C$$



Combinando Transformações (cont...)

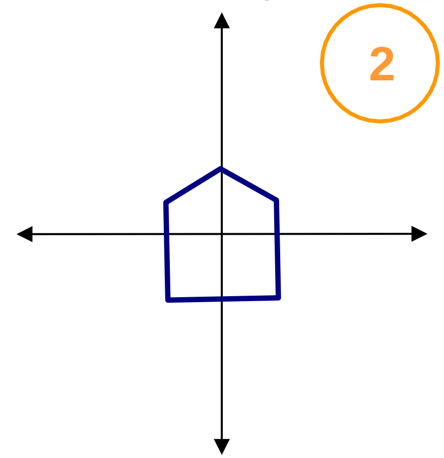
$Casa\ (C)$

1



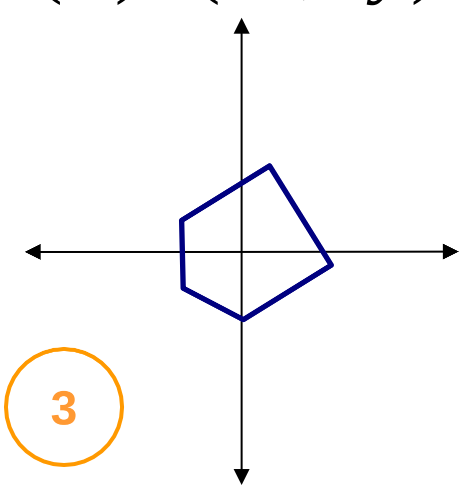
$T(dx, dy)C$

2



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

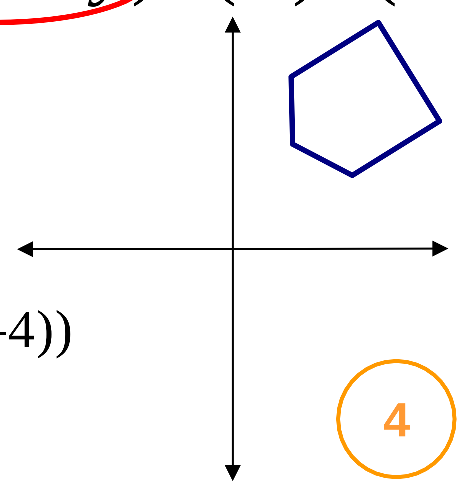
3



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

Exemplo:

$T(-(-2), -(-4))$



4

Combinando Transformações (cont…)

- As três transformações seriam combinadas da seguinte forma

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

$$v' = T(-dx, -dy)R(\theta)T(dx, dy)v$$

Lembre-se: Multiplicação de matrizes não é uma operação comutativa, então a ordem IMPORTA!

Combinando Transformações (cont…)

- A ordem de execução é a seguinte:

$$v' = \underbrace{(T(-dx,-dy)\underbrace{(R(\theta)\underbrace{(T(dx,dy)v)}_{1^{\text{o}}})}_{2^{\text{o}}})}_{3^{\text{o}}}$$

Transformações Geométricas com OpenGL

- Os principais comandos em OpenGL para realizar as transformações geométricas são as seguintes:

```
glTranslatef(dx, dy, dz);

glScalef(sx, sy, sz);

glRotatef(ângulo, x, y, z);
```

Resumo das operações

- Translação

$$x_{new} = x_{old} + dx \qquad y_{new} = y_{old} + dy$$

- Escala

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

- Rotação

$$x_{new} = x_{old} \times \cos\theta - y_{old} \times \sin\theta$$
$$y_{new} = x_{old} \times \sin\theta + y_{old} \times \cos\theta$$

Resumo das operações

Translação: $$\begin{bmatrix} 1 & 0 & dx \\ 0 & 1 & dy \\ 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} x+dx \\ y+dy \\ 1 \end{bmatrix} : v' = T(dx, dy)v$$

Escala: $$\begin{bmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} s_x \times x \\ s_y \times y \\ 1 \end{bmatrix} : v' = S(s_x, s_y)v$$

Rotação: $$\begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} \cos\theta \times x - \sin\theta \times y \\ \sin\theta \times x + \cos\theta \times y \\ 1 \end{bmatrix} : v' = R(\theta)v$$

Fonte dos Slides

Slides do material do Prof. Rodrigo Luis de Souza da Silva - UFJF