Aula 05
Transformações Geométricas

Prof. Dr. Denis Henrique Pinheiro Salvadeo

Aula Anterior

- Preenchimento de Polígonos
  - Questões relacionadas
  - Algoritmo Scan-line

Aula de Hoje

- Transformações Geométricas 2-D
- Matrizes de Transformação e Coordenadas Homogêneas
- Combinação de Transformações
- Transformação de Janela de Desenho para ViewPort
- Transformações Geométricas 3-D

Transformações Geométricas

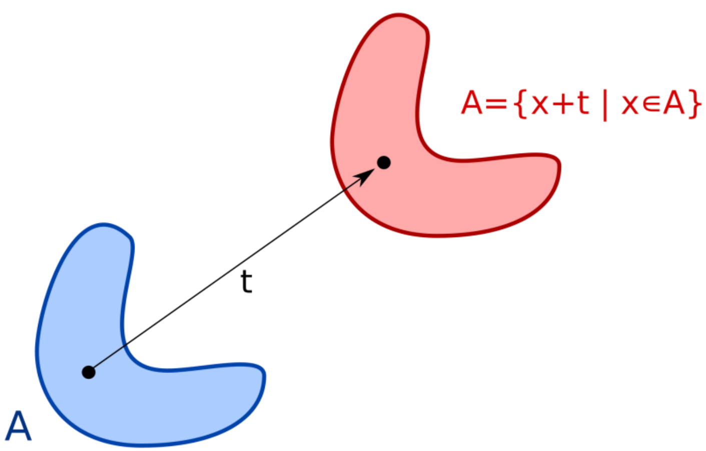
- Operações matemáticas que permitem alterar uniformemente o aspecto de um desenho
  - Desenho sendo apresentado em diferentes orientações e escalas, por exemplo
  - Não afetam a sua estrutura
- Agem sobre as primitivas gráficas
  - Transformam as coordenadas dos pontos que as definem, obtendo um desenho transformado
  - Assume que são definidas com relação à origem do sistema de referência
- Transformação Inversa
  - Sempre possível retornar ao desenho original a partir de um transformado
- Principais transformações
  - Escala, Translação e Rotação

Transformações Geométricas

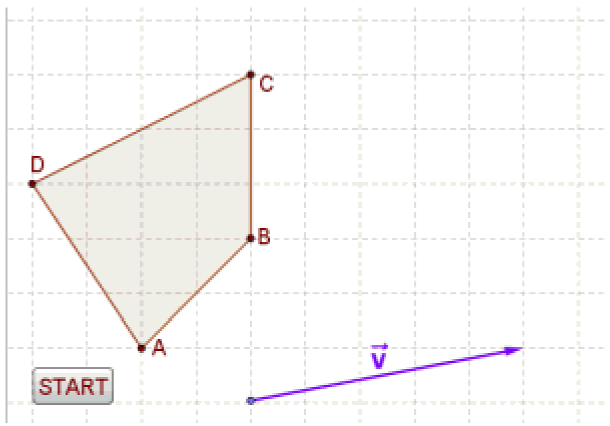
2-D

TRANSLAÇÃO

- É o deslocamento em linha reta de um objeto para uma outra posição
- Sejam T_x e T_y **distâncias de translação** nas direções x e y, respectivamente
  - Um **vetor de translação** ou **deslocamento** definido por $\vec{v}(T_x, T_y)$ é adicionado às coordenadas originais do desenho



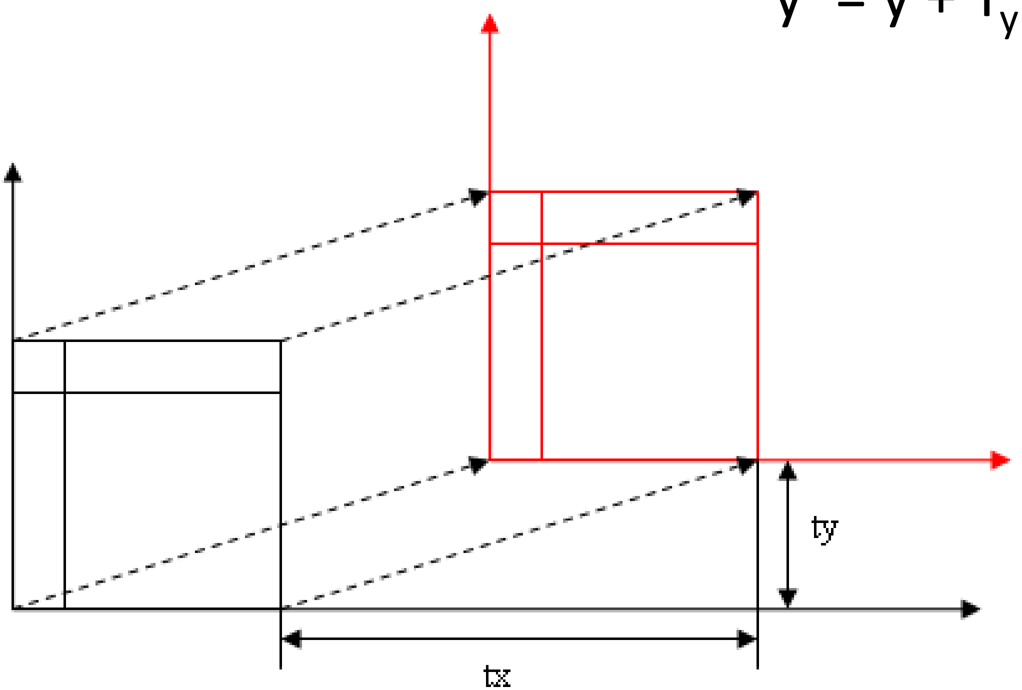
http://upload.wikimedia.org/wikipedia/commons/thumb/9/91/Translation_of_a_set.svg/2000px-Translation_of_a_set.svg.png



http://upload.wikimedia.org/wikipedia/commons/5/59/Translation_geometry.gif

- Sejam as coordenadas originais (x,y) e as coordenadas resultantes da transformação (x',y'), temos:

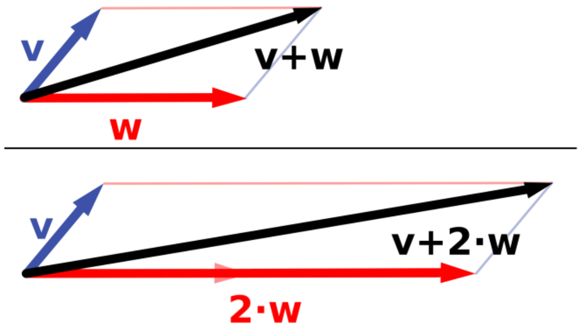
$$x' = x + T_x$$
$$y' = y + T_y$$



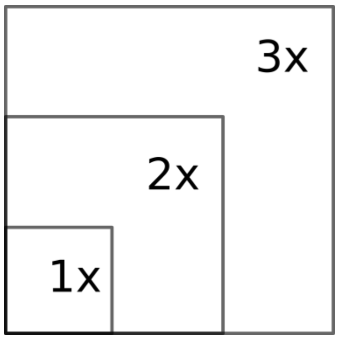
http://upload.wikimedia.org/wikipedia/commons/d/d5/CoordinateTranslation.png

ESCALA

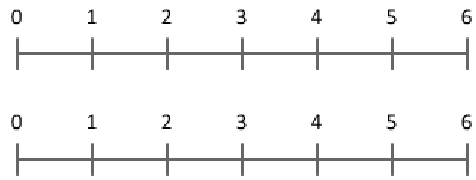
- Altera o tamanho do objeto
- Sejam S_x e S_y **fatores de escala** referentes aos eixos x e y, respectivamente



http://upload.wikimedia.org/wikipedia/commons/thumb/6/63/Vector_addition_and_scaling.svg/2000px-Vector_addition_and_scaling.svg.png



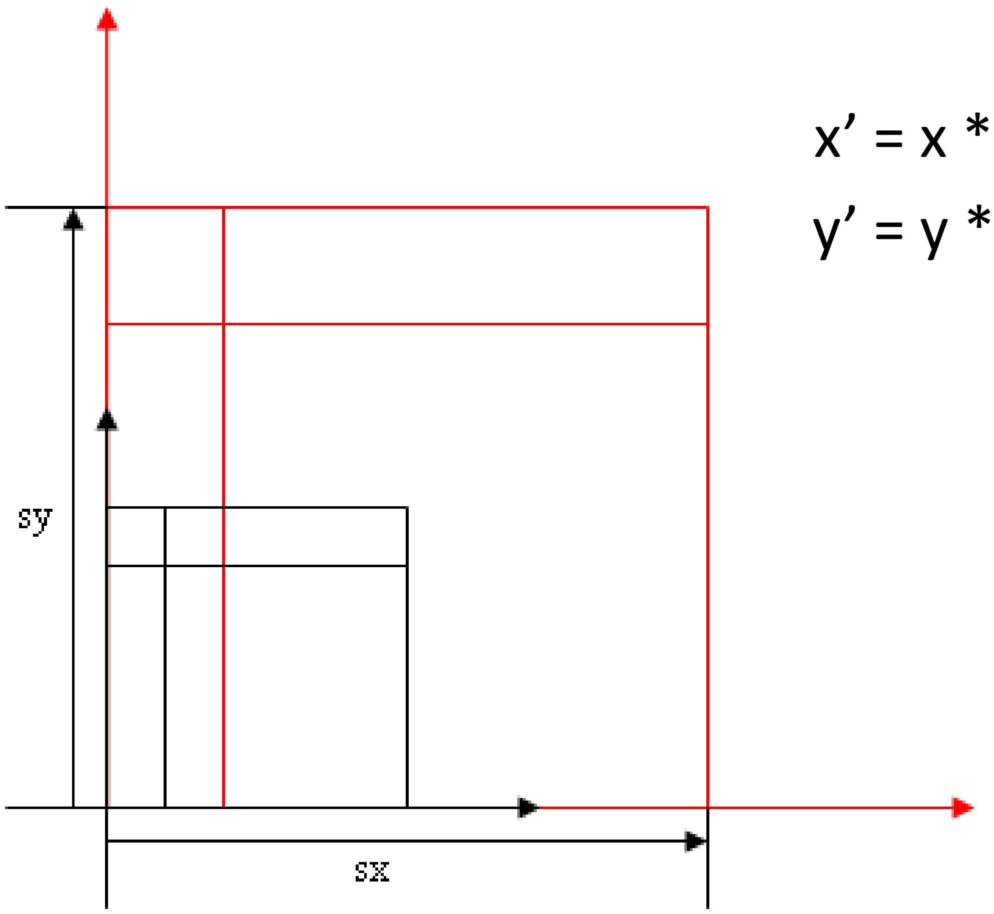
http://upload.wikimedia.org/wikipedia/commons/thumb/6/6a/Box_scaling.svg/2000px-Box_scaling.svg.png



http://upload.wikimedia.org/wikipedia/commons/0/02/Multiplication_as_scaling_integers.gif

ESCALA

- Sejam as coordenadas originais (x,y) e as coordenadas resultantes da transformação (x',y'), temos:



$$x' = x * S_x$$

$$y' = y * S_y$$

http://upload.wikimedia.org/wikipedia/commons/0/0f/CoordinateScaling.png

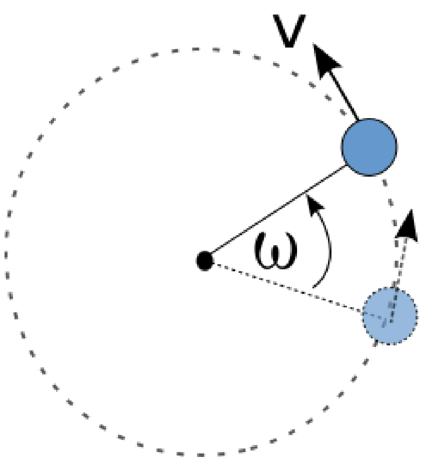
ESCALA

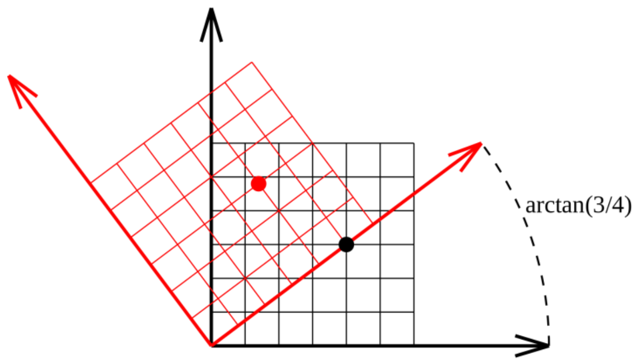
- Alguns casos:
  - $|Sx|, |Sy| = 1$ => não altera o tamanho
  - $|Sx|,|Sy| > 1$ => Ampliação no eixo afetado
  - $|Sx|,|Sy| < 1$ => Redução no eixo afetado
  - $Sx = Sy$ => produz transformação uniforme
  - $Sx < 0$, $Sy < 0$ => Espelha a imagem em relação ao eixo não afetado

http://upload.wikimedia.org/wikipedia/commons/0/0f/CoordinateScaling.png

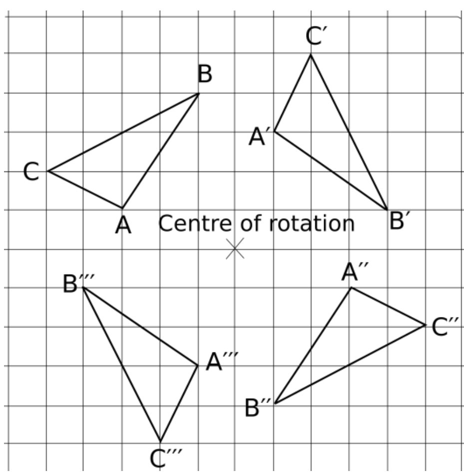
Rotação

- É o deslocamento de um objeto para uma outra posição em trajetória circular
  - Os pontos mantém a mesma distância da origem
  - O deslocamento angular θ é o único parâmetro da transformação
  - Sentido anti-horário





http://upload.wikimedia.org/wikipedia/commons/thumb/b/b1/Coordinate_rotation.svg/2000px-Coordinate_rotation.svg.png



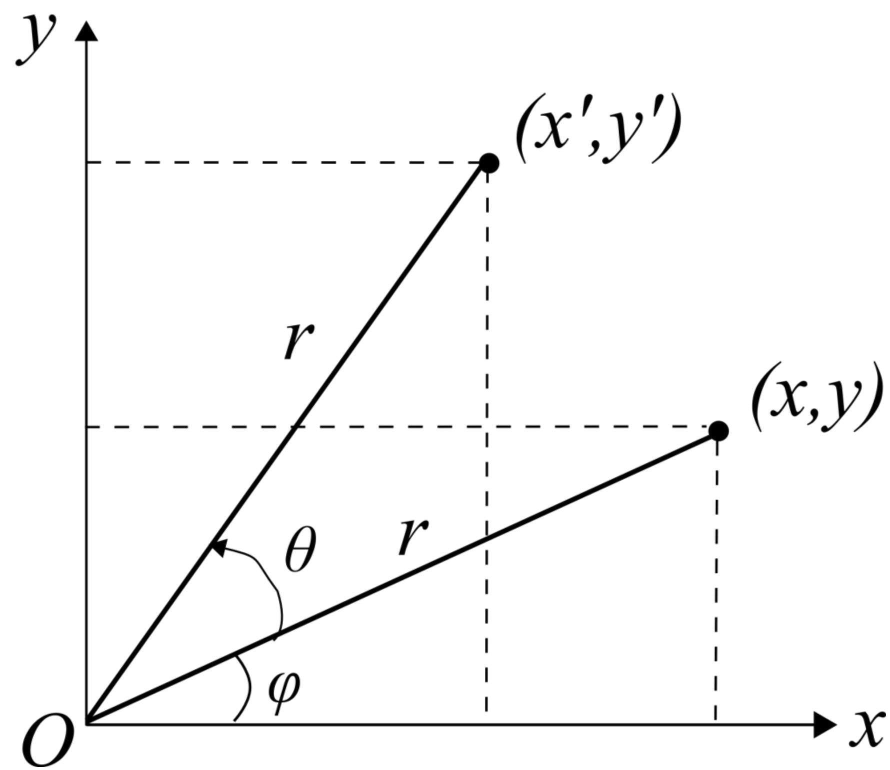
http://upload.wikimedia.org/wikipedia/commons/7/7b/Circular_motion_diagram.png

http://upload.wikimedia.org/wikipedia/commons/thumb/d/da/Rotational_transformation.svg/2000px-Rotational_transformation.svg.png

ROTAÇÃO

- As novas coordenadas são:
  - $x' = r * \cos(\phi+\theta) = r*\cos\phi \cos\theta - r*\text{sen}\phi\ \text{sen}\theta$
  - $y' = r * \text{sen}(\phi+\theta) = r*\text{sen}\phi \cos\theta + r*\cos\phi\ \text{sen}\theta$

- Como as coordenadas originais são $x = r * \cos\phi$ e $y = r * \text{sen}\phi$, temos:
  - $x' = x*\cos\theta - y*\text{sen}\theta$
  - $y' = y*\cos\theta + x*\text{sen}\theta$



Baseado em HEARN, D. Computer Graphics with OpenGL. 3rd ed. Upper Saddle River, NJ: Pearson Education, 2004. 857 p.

Matriz de Transformação

- Forma matricial de uma operação de transformação
- Para se calcular os pontos transformados de uma figura, basta multiplicar cada ponto (disposto em uma matriz coluna) por esta matriz

- Ex: Escala

$$x' = x * S_x$$
$$y' = y * S_y$$

  - Em forma matricial, temos: $$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} xS_x \\ yS_y \end{bmatrix} = \underbrace{\begin{bmatrix} S_x & 0 \\ 0 & S_y \end{bmatrix}}_{\text{Matriz de Escala}} \begin{bmatrix} x \\ y \end{bmatrix}$$

- E para a translação, será possível definirmos uma matriz de transformação 2x2?

Coordenadas Homogêneas

- Consiste em um método genérico para permitir todas as transformações
  - Consistência nos cálculos

- Tanto coordenadas, quando matrizes de transformação são expandidas para considerar uma dimensão adicional
  - Uma coordenada (x, y), será representada por (x_h, y_h, h) em coordenadas homogêneas, onde $x = x_h/h$ e $y = y_h/h$.
    - Se h = 1 (caso padrão), temos (x, y, 1) em coordenada homogênea
  - Após os cálculos, a terceira coordenada é descartada

Matriz de Escala em Coordenadas Homogêneas

$x' = x * S_x$

$y' = y * S_y$

$z' = 1$

$$\begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} = \begin{bmatrix} xS_x \\ yS_y \\ 1 \end{bmatrix} = \underbrace{\begin{bmatrix} S_x & 0 & 0 \\ 0 & S_y & 0 \\ 0 & 0 & 1 \end{bmatrix}}_{\text{Matriz de Escala}} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

*Onde a coordenada z' será descartada no ponto transformado

Matriz de Rotação em Coordenadas Homogêneas

x’ = x*cosθ – y*senθ

y’ = y*cosθ + x*senθ

z’ = 1

$$\begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} = \begin{bmatrix} x\cos\theta - y\,\mathrm{sen}\,\theta \\ x\,\mathrm{sen}\,\theta + y\cos\theta \\ 1 \end{bmatrix} = \underbrace{\begin{bmatrix} \cos\theta & -\mathrm{sen}\,\theta & 0 \\ \mathrm{sen}\,\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}}_{\text{Matriz de Rotação}} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

*Onde a coordenada z’ será descartada no ponto transformado

Matriz de Translação em Coordenadas Homogêneas

$x' = x + T_x$

$y' = y + T_y$

$z' = 1$

$$\begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} = \begin{bmatrix} x + T_x \\ y + T_y \\ 1 \end{bmatrix} = \underbrace{\begin{bmatrix} 1 & 0 & T_x \\ 0 & 1 & T_y \\ 0 & 0 & 1 \end{bmatrix}}_{\text{Matriz de Translação}} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

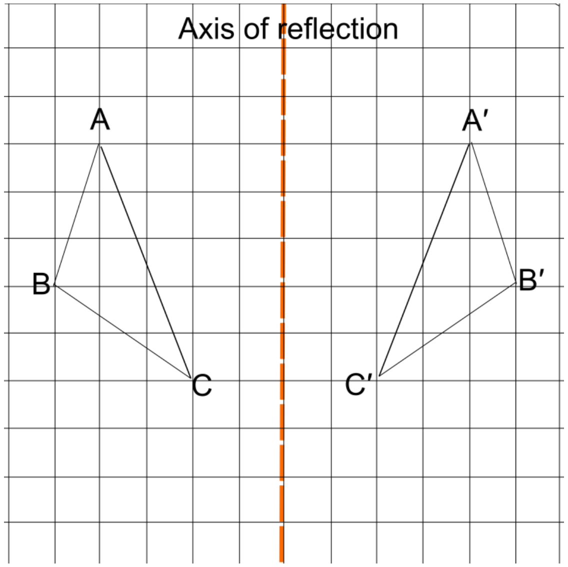
*Onde a coordenada z’ será descartada no ponto transformado

Outtras Transformações

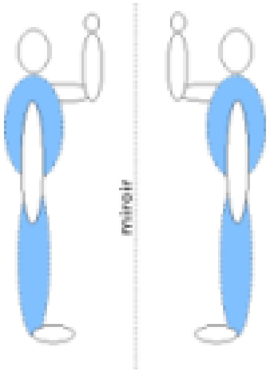
2-D

Espelhamento

- Também chamada de Reflexão
- O objeto resultante é um espelho do original
  - Corresponde a girar 180° o objeto com relação a algum eixo de reflexão



http://upload.wikimedia.org/wikipedia/commons/d/dd/Reflectional_transformation.svg



http://upload.wikimedia.org/wikipedia/commons/b/b3/Symplan.png

- No caso 2-D e com relação aos eixos O_x e O_y, pode ser obtida diretamente da Matriz de Escala

- Espelho com relação a O_x (S_y = -1)

$$E_{O_x} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

- Espelho com relação a O_y (S_x = -1)

$$E_{O_y} = \begin{bmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

- Espelho com relação a O_x e O_y ($S_x = S_y$ = -1)

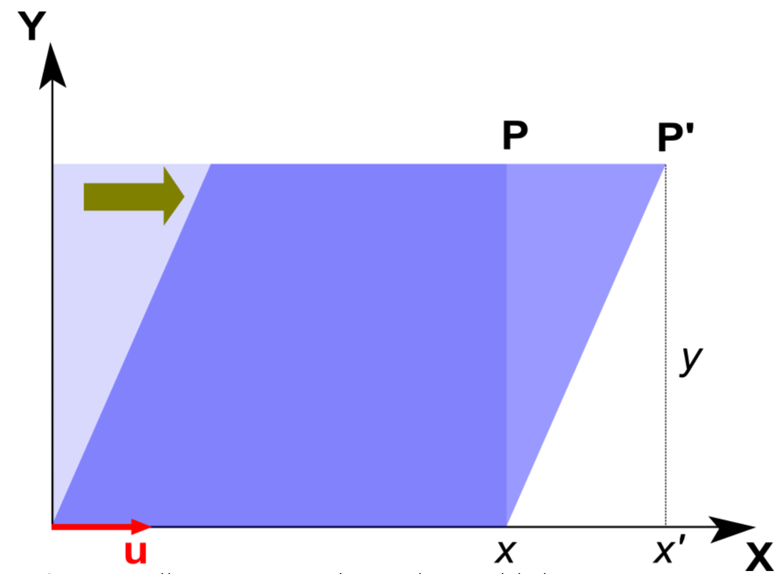
$$E_{O_{xy}} = \begin{bmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

- Distorce o formato do objeto, em geral, deslocando as coordenadas x ou y em função do eixo não selecionado

- Ex: Com relação ao eixo x

$$x' = x + Sh_x * y$$
$$y' = y$$
$$z' = 1$$

$$\begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} = \begin{bmatrix} x + Sh_x y \\ y \\ 1 \end{bmatrix} = \underbrace{\begin{bmatrix} 1 & Sh_x & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}}_{\substack{\text{Matriz de Cisalhamento} \\ \text{(Eixo x)}}} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$



Modificado de http://upload.wikimedia.org/wikipedia/commons/2/2a/Shear.svg

Combinação de Transformações

- Considere o seguinte:
  1. Um ponto (x_1,y_1) sofre a transformação M_1, obtendo o ponto (x_2,y_2)

$$\begin{bmatrix} x_2 \\ y_2 \\ z_2 \end{bmatrix} = M_1 \begin{bmatrix} x_1 \\ y_1 \\ 1 \end{bmatrix}$$

  2. Em seguida, o ponto (x_2,y_2) sofre uma transformação M_2, obtendo o ponto (x_3,y_3)

$$\begin{bmatrix} x_3 \\ y_3 \\ z_3 \end{bmatrix} = M_2 \begin{bmatrix} x_2 \\ y_2 \\ 1 \end{bmatrix}$$

- **Existe uma matriz M_3 que aplicada sobre (x_1,y_1), resulta diretamente em (x_3,y_3)?**

- Substituindo (1) em (2), temos:

$$\begin{bmatrix} x_3 \\ y_3 \\ z_3 \end{bmatrix} = M_2 \left(M_1 \begin{bmatrix} x_1 \\ y_1 \\ 1 \end{bmatrix} \right)$$

- Pela propriedade associativa, temos:

$$\begin{bmatrix} x_3 \\ y_3 \\ z_3 \end{bmatrix} = \left(M_2 M_1 \right) \begin{bmatrix} x_1 \\ y_1 \\ 1 \end{bmatrix}$$

- Substituindo-se $M_3 = M_2M_1$, temos

$$\begin{bmatrix} x_3 \\ y_3 \\ z_3 \end{bmatrix} = M_3 \begin{bmatrix} x_1 \\ y_1 \\ 1 \end{bmatrix}$$

Combinação de Transformações

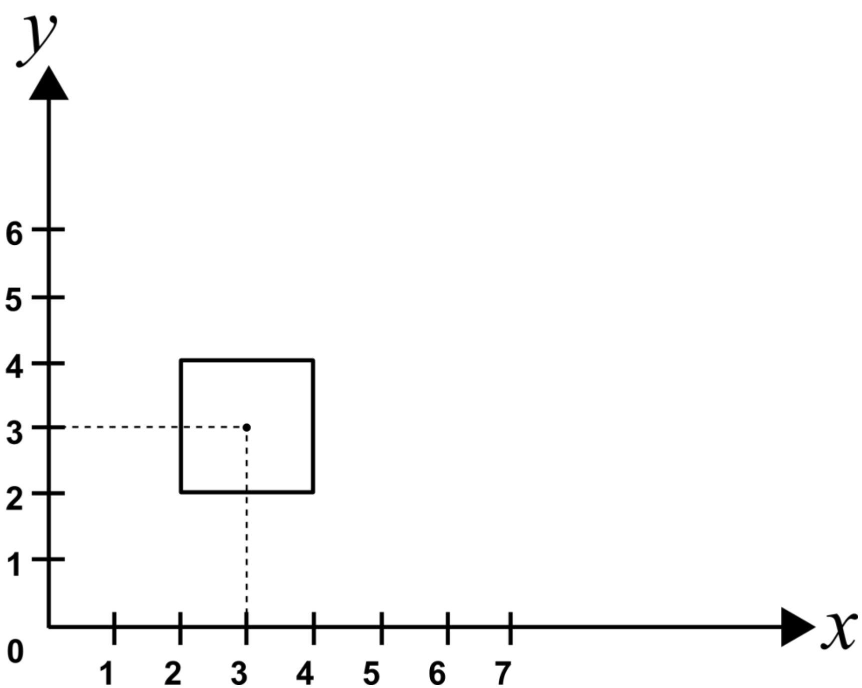
- Resumindo:
  - Várias transformações podem ser **combinadas em uma única matriz**
  - As matrizes devem ser **multiplicadas na ordem oposta** em que se deseja aplicar as transformações.
    - Relembre: $M_3 = M_2M_1$

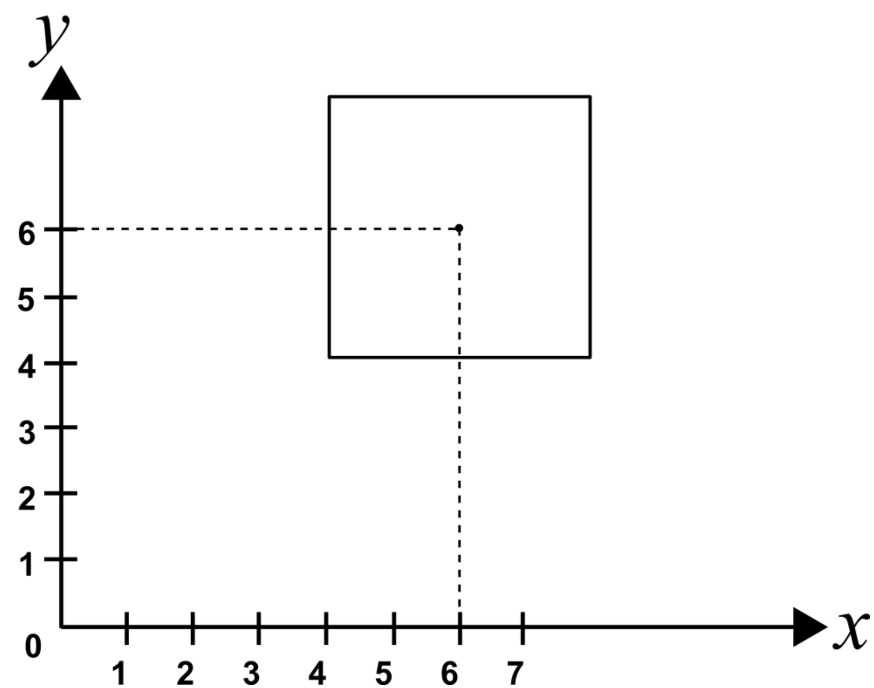
- Observações:
  - Estamos assumindo o padrão de matriz coluna.
  - Multiplicação de matrizes não tem propriedade comutativa
    - Portanto, **a ordem das transformações importa!!!**
    - Ex: Aplicar uma translação e depois uma rotação, não é o mesmo que aplicar uma rotação e depois uma translação.

- Isto nos permitirá obter outras transformações

- Problema da Escala
  - Em geral, desloca o objeto (distancia ou aproxima da origem – ponto de referência)

$S_x = S_y = 2$

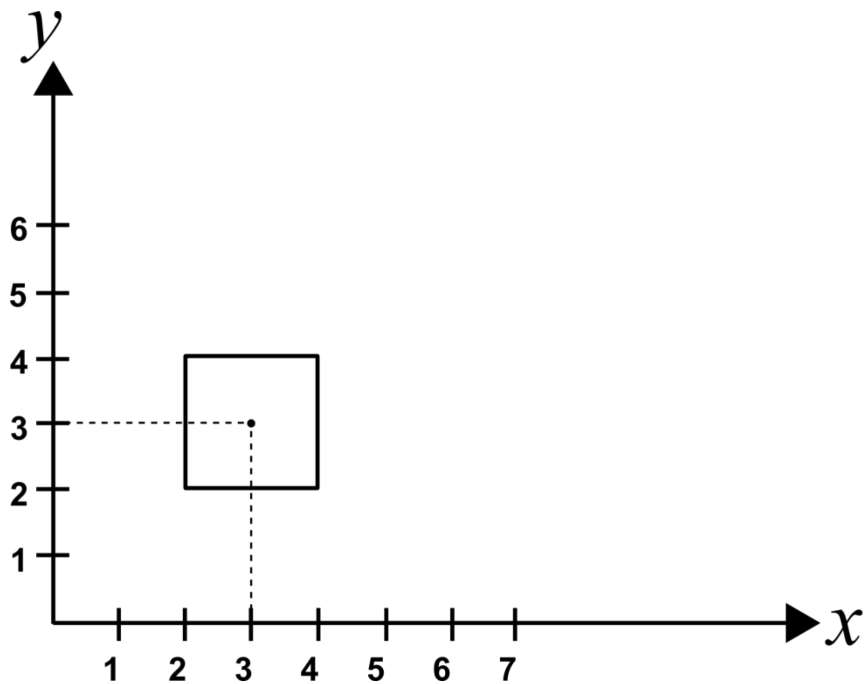


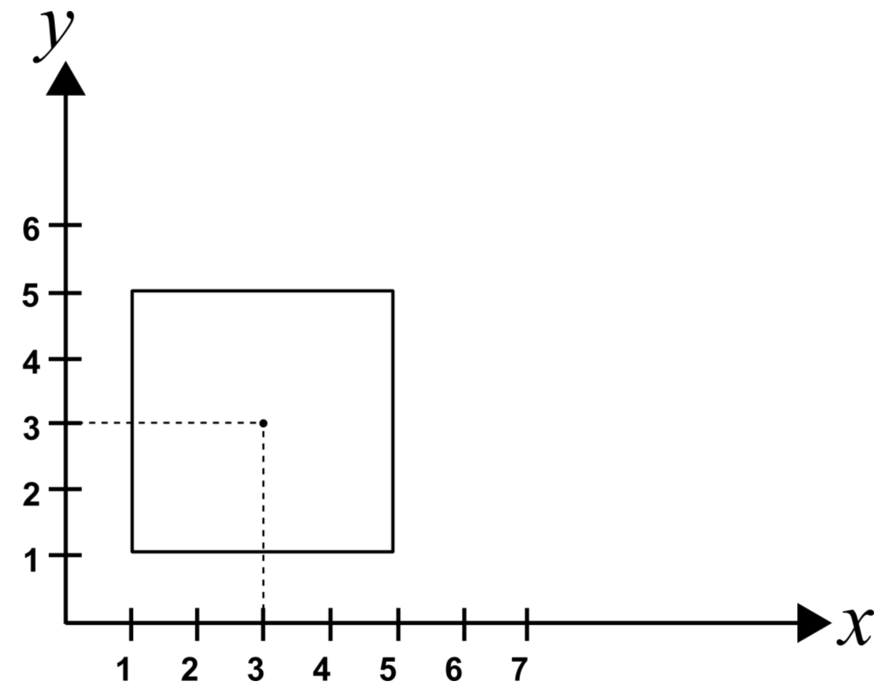


Baseados em ROMERO, T. et al. Fundamentos de Computação Gráfica, LTC, 1987.

- Mas e se quisermos obter um aumento ou redução do objeto com relação a um ponto arbitrário (por exemplo, centro do objeto)?

$$S_x = S_y = 2$$



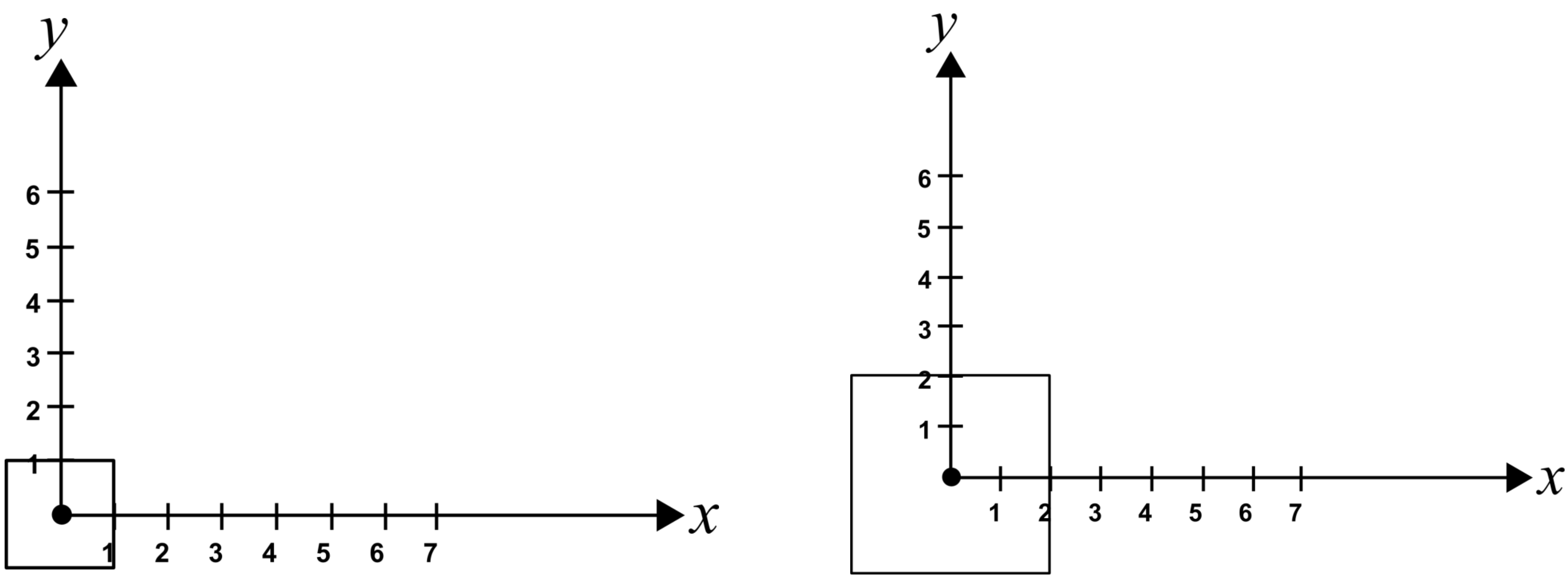


Baseados em ROMERO, T. et al. Fundamentos de Computação Gráfica, LTC, 1987.

ESCALA EM RELAÇÃO A UM PONTO ARBITRÁRIO

- Vejamos o que ocorre quando o centro do objeto é a Origem?

$S_x = S_y = 2$



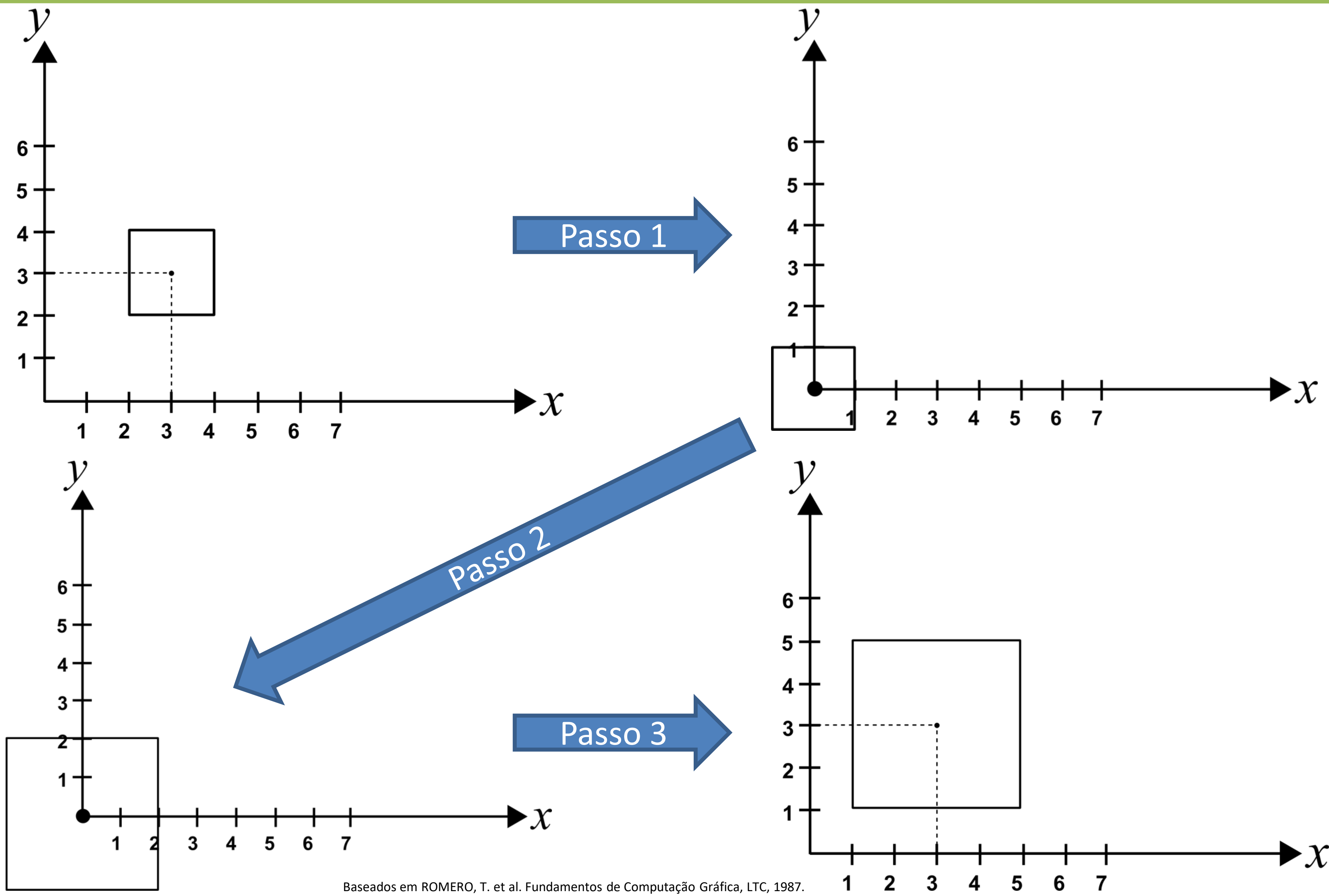
Baseados em ROMERO, T. et al. Fundamentos de Computação Gráfica, LTC, 1987.

Escala em Relação a um Ponto Arbitrário

- Solução:
  - Seja o ponto (x_F,y_F) o nosso ponto de referência para a transformação
    1. Transladar o objeto de modo que (x_F,y_F) caia na origem ($T_x = -x_F$, $T_y = -y_F$)
    2. Aplicar a escala deseja (S_x e S_y)
    3. Transladar o objeto de modo que o ponto de referência retorna para a sua posição original ($T_x = x_F$, $T_y = y_F$)

$$\underbrace{\begin{bmatrix} 1 & 0 & x_F \\ 0 & 1 & y_F \\ 0 & 0 & 1 \end{bmatrix}}_{\substack{\text{Matriz de Translação} \\ \text{(Passo 3)}}} \underbrace{\begin{bmatrix} S_x & 0 & 0 \\ 0 & S_y & 0 \\ 0 & 0 & 1 \end{bmatrix}}_{\substack{\text{Matriz de Escala} \\ \text{(Passo 2)}}} \underbrace{\begin{bmatrix} 1 & 0 & -x_F \\ 0 & 1 & -y_F \\ 0 & 0 & 1 \end{bmatrix}}_{\substack{\text{Matriz de Translação} \\ \text{(Passo 1)}}} = \underbrace{\begin{bmatrix} S_x & 0 & x_F(1-S_x) \\ 0 & S_y & y_F(1-S_y) \\ 0 & 0 & 1 \end{bmatrix}}_{\substack{\text{Matriz de Escala} \\ \text{com Relação a um} \\ \text{Ponto Arbitrário}}}$$

Escala em Relação a um Ponto Arbitrário

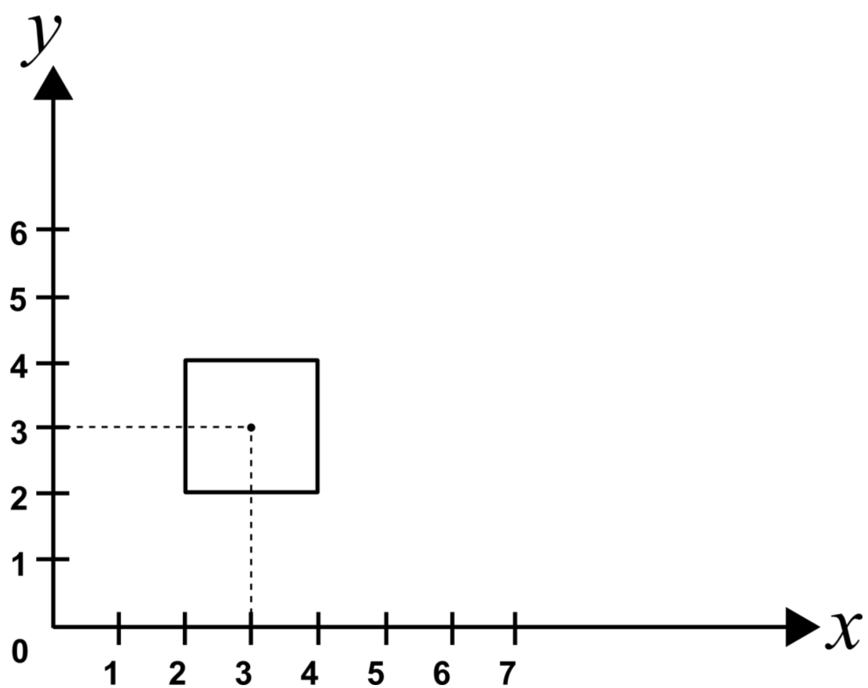


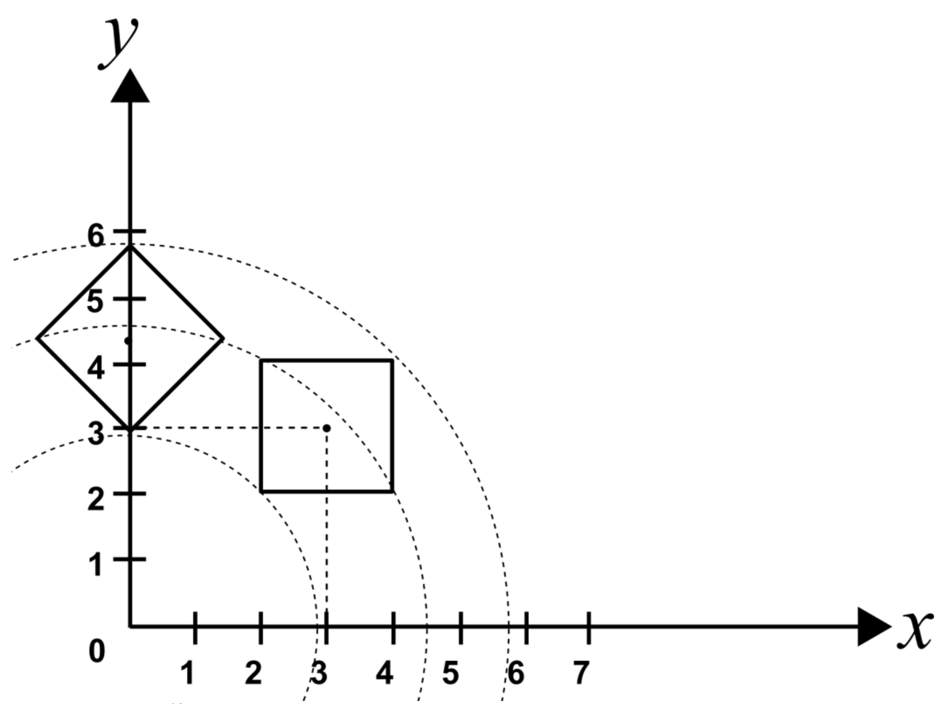
Baseados em ROMERO, T. et al. Fundamentos de Computação Gráfica, LTC, 1987.

Rotação em Torno de um Ponto Arbitrário

- Até agora, sabemos a rotação com relação à origem
  - Desloca o objeto em trajetória (mantendo a distância da origem)

$\theta = 45°$

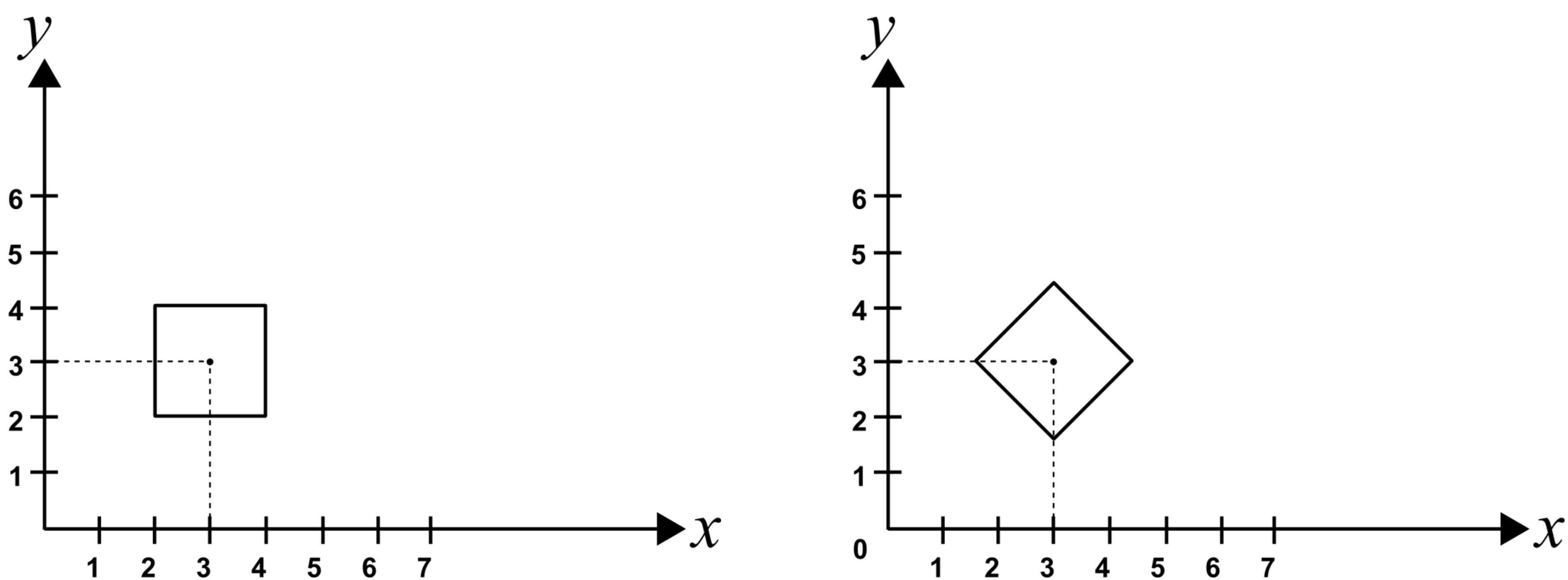




Baseados em ROMERO, T. et al. Fundamentos de Computação Gráfica, LTC, 1987.

- Mas e se quisermos obter uma rotação com relação a um ponto arbitrário (por exemplo, centro do objeto)?

$\theta = 45°$

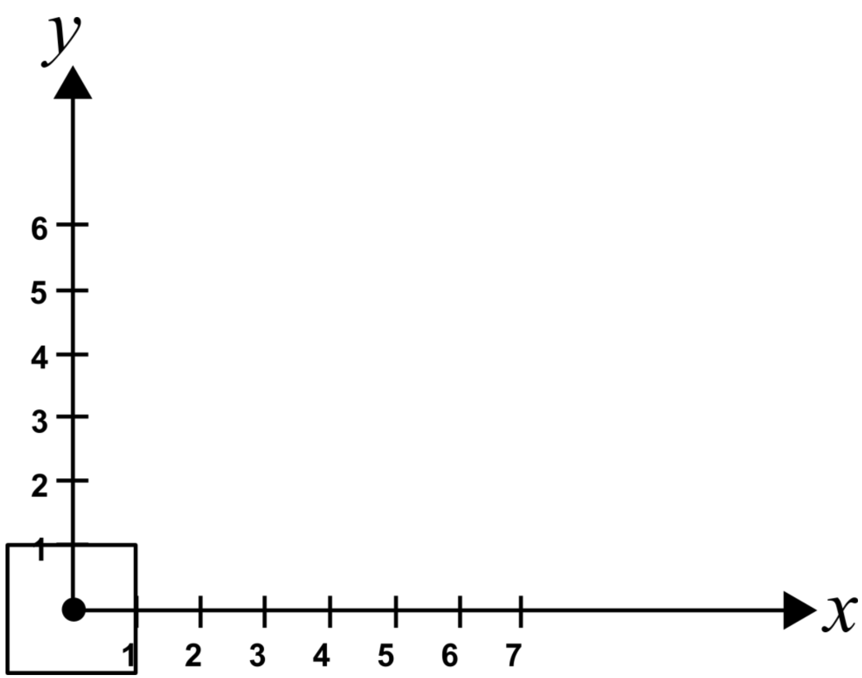


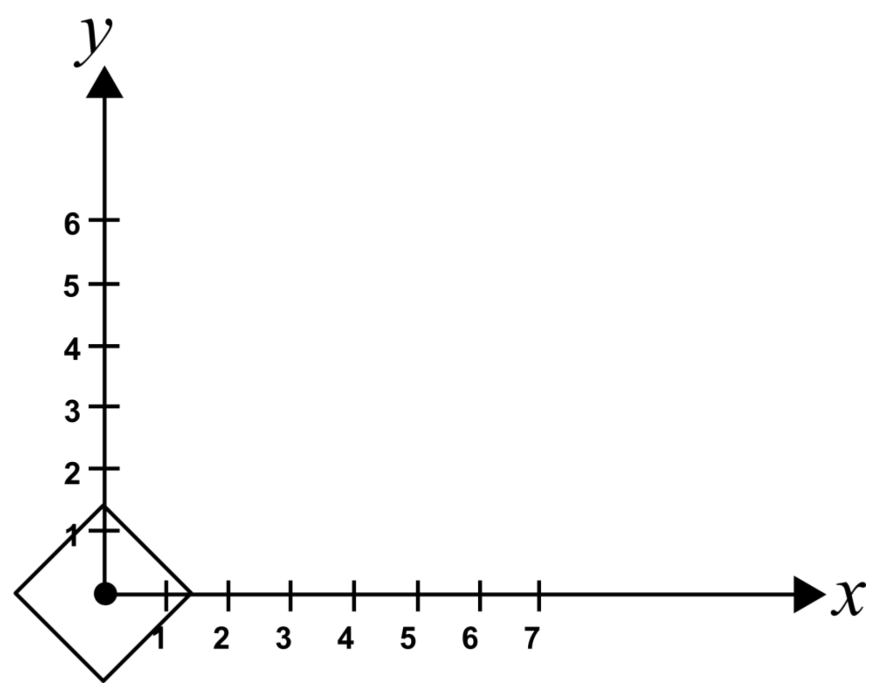
Baseados em ROMERO, T. et al. Fundamentos de Computação Gráfica, LTC, 1987.

ROTAÇÃO EM TORNO DE UM PONTO ARBITRÁRIO

- Novamente, vejamos o que ocorre quando o centro do objeto é a Origem?

θ = 45°





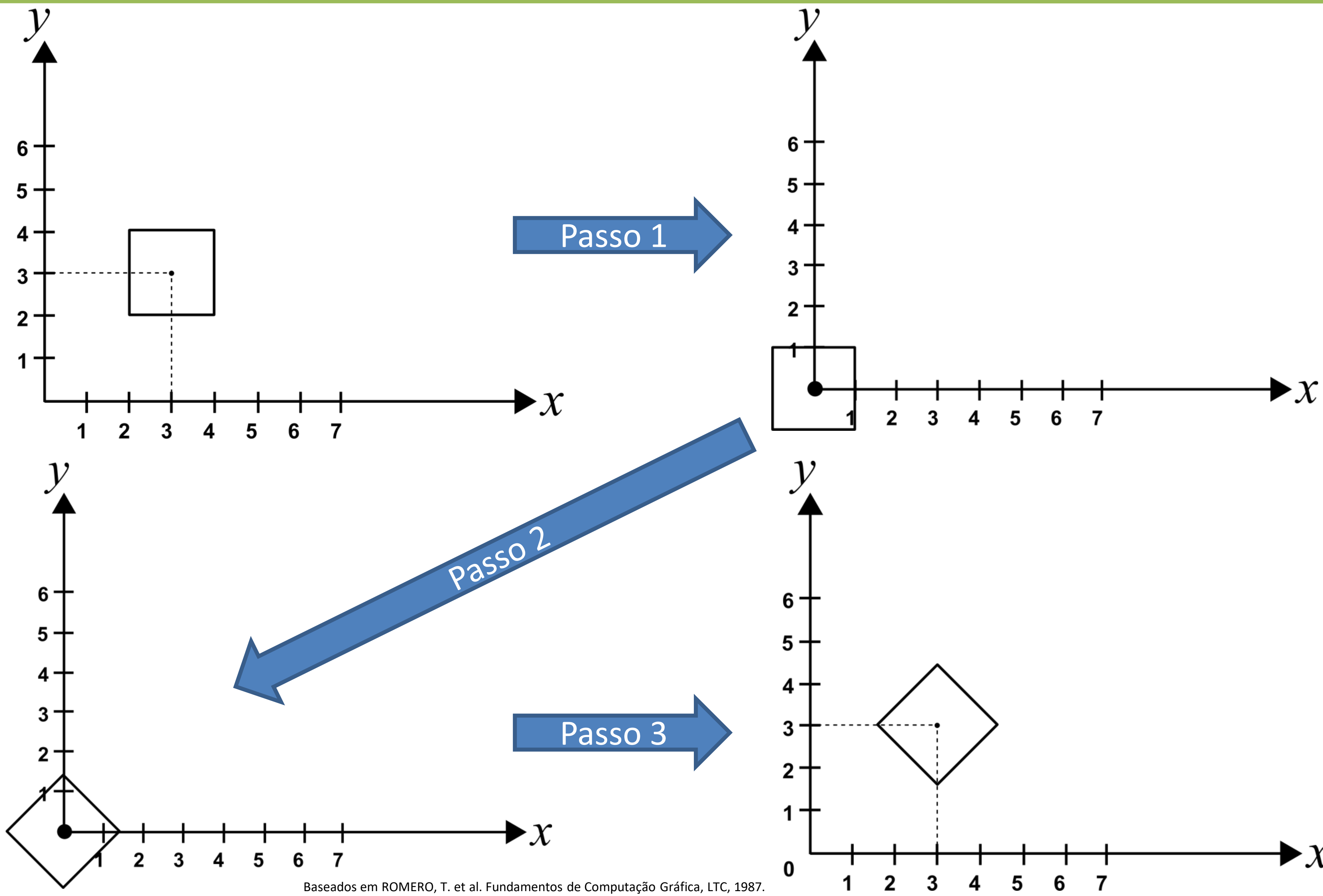
Baseados em ROMERO, T. et al. Fundamentos de Computação Gráfica, LTC, 1987.

Rotação em Torno de um Ponto Arbitrário

- Solução:
  – Seja o ponto (x_R,y_R) o nosso ponto de referência para a transformação
    1. Transladar o objeto de modo que (x_R,y_R) caia na origem (T_x = $-x_R$, T_y = $-y_R$)
    2. Aplicar a rotação deseja (θ)
    3. Transladar o objeto de modo que o ponto de referência retorna para a sua posição original (T_x = x_R, T_y = y_R)

$$\underbrace{\begin{bmatrix} 1 & 0 & x_R \\ 0 & 1 & y_R \\ 0 & 0 & 1 \end{bmatrix}}_{\substack{\text{Matriz de Translação} \\ \text{(Passo 3)}}} \underbrace{\begin{bmatrix} \cos\theta & -\operatorname{sen}\theta & 0 \\ \operatorname{sen}\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}}_{\substack{\text{Matriz de Rotação} \\ \text{(Passo 2)}}} \underbrace{\begin{bmatrix} 1 & 0 & -x_R \\ 0 & 1 & -y_R \\ 0 & 0 & 1 \end{bmatrix}}_{\substack{\text{Matriz de Translação} \\ \text{(Passo 1)}}} = \underbrace{\begin{bmatrix} \cos\theta & -\operatorname{sen}\theta & x_R(1-\cos\theta)+y_R\operatorname{sen}\theta \\ \operatorname{sen}\theta & \cos\theta & y_R(1-\cos\theta)-x_R\operatorname{sen}\theta \\ 0 & 0 & 1 \end{bmatrix}}_{\text{Matriz de Rotação em torno de um Ponto Arbitrário}}$$

Rotação em Torno de um Ponto Arbitrário



Baseados em ROMERO, T. et al. Fundamentos de Computação Gráfica, LTC, 1987.

Espelhamento em Relação a Uma Reta Qualquer

- Até agora, sabemos espelhar com relação aos eixos coordenados
  - Através de valores negativos para os fatores de escala Sx e Sy

- Mas e se desejamos fazer o espelhamento com relação a qualquer reta?

- Ideia
  - Fazer algo semelhante ao que foi realizado para a escala e rotação, mas agora considerando os eixos coordenados

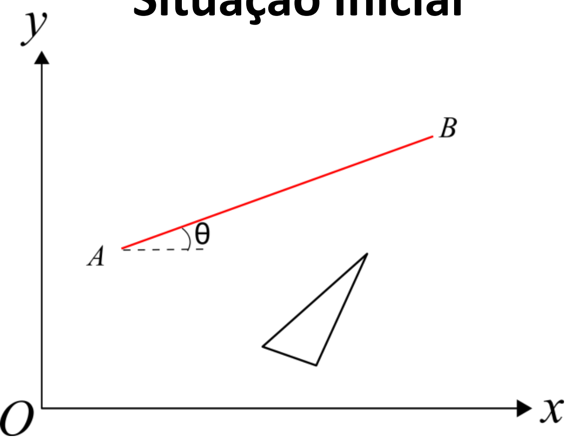
Espelhamento em Relação a Uma Reta Qualquer

- Solução:
  – Seja o segmento de reta AB definido pelos pontos (x_A,y_A) e (x_B,y_B) a nossa referência para a transformação
    1. Transladar o objeto de modo que um dos pontos A ou B caia na origem
       – Ex: Vamos assumir que A ficará na origem ($T_x = -x_A$, $T_y = -y_A$)
    2. Rotacionar a figura de modo que a reta de simetria se torne paralela a um dos eixos coordenados
       – Ex: Vamos assumir a rotação deseja $-\theta$
    3. Espelhar a figura em relação ao eixo coincidente com a reta de simetria.
       – Ex: Assumindo que coincide com o eixo Ox, usamos $Sx = 1$ e $Sy = -1$ na matriz de escala
    4. Rotacionar a figura de ângulo oposto ao realizado no passo (2)
       – Ex: Como em (2) usamos $-\theta$, agora usaremos θ
    5. Transladar o objeto com deslocamentos opostos aos realizados em (1), retornando o segmento de reta para a sua posição original
       – Ex: Como assumimos A no passo (1), agora usaremos ($T_x = x_A$, $T_y = y_A$)

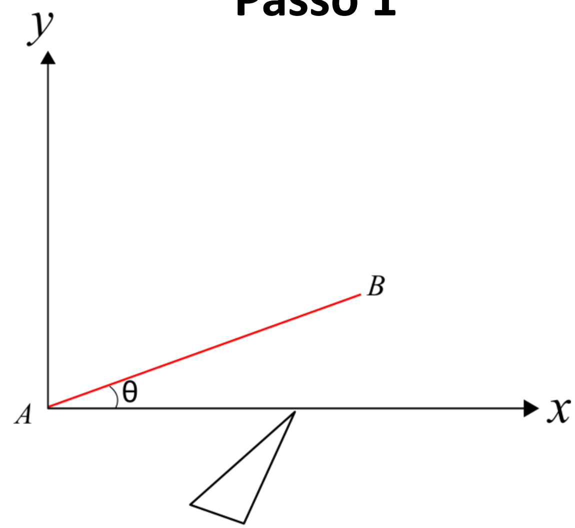
- $M = T(T_x = x_A, T_y = y_A) \,.\, R(\theta) \,.\, S(Sx = 1, Sy = -1) \,.\, R(-\theta) \,.\, T(T_x = -x_A, T_y = -y_A)$

Espelhamento em Relação a Uma Reta Qualquer

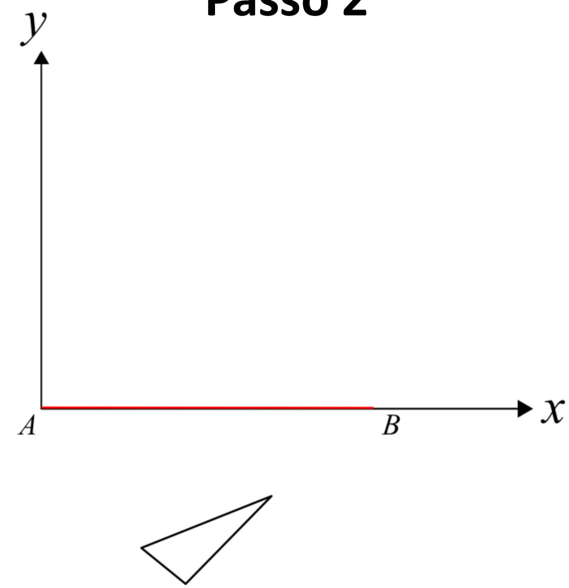
Situação Inicial



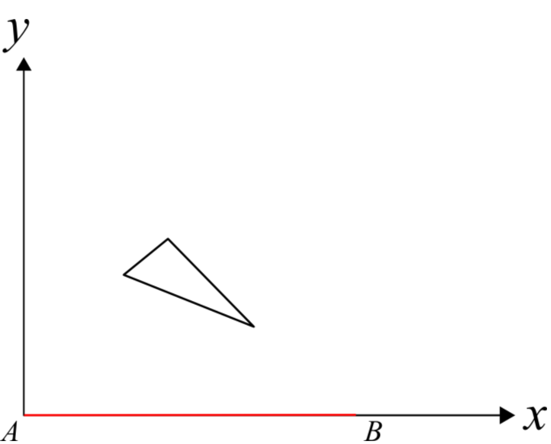
Passo 1



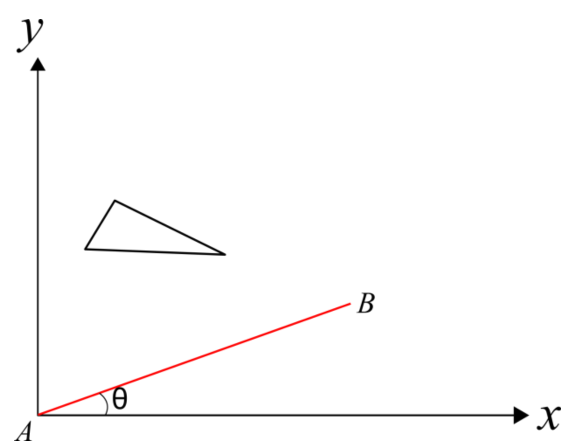
Passo 2



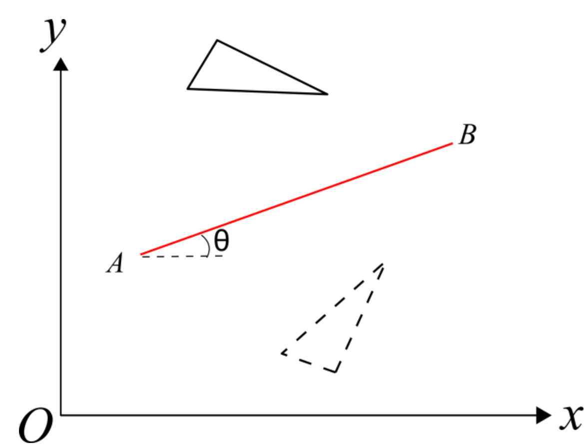
Passo 3



Passo 4



Passo 5



Baseado em ROMERO, T. et al. Fundamentos de Computação Gráfica, LTC, 1987.

Observação Básica e Essencial!

Na origem do sistema de coordenadas (ou em seus eixos coordenados) as transformações funcionam como esperado...

E sabemos as transformações básicas para trabalhar quando a origem ou os eixos coordenados são a referência...

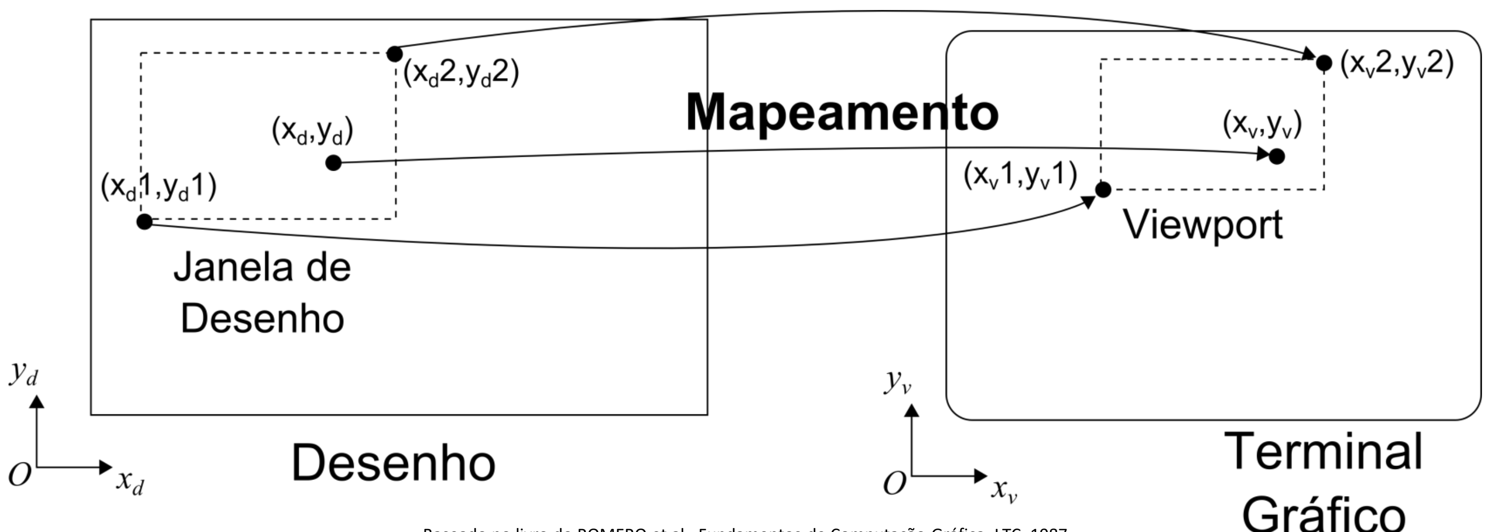
Observação Básica e Essencial!

Portanto, resolva o seu problema principal na origem (ou nos eixos coordenados) e depois retorne à posição ou ao eixo de referência original com as transformações inversas!

Transformação de Janela de Desenho para ViewPort

- Podemos criar uma matriz para realizar o mapeamento das coordenadas de mundo (real e infinita) para as coordenadas de tela (inteira e limitada)

- Ideia:
  - Basear-se no que já conhecemos até aqui



Baseado no livro do ROMERO et al.. Fundamentos de Computação Gráfica. LTC, 1987

Transformação de Janela de Desenho para ViewPort

- Solução:
  - Sejam os pontos (x_D1,y_D1) e (x_D2,y_D2) as coordenadas da janela de desenho mínima e máxima, respectivamente.
  - De maneira semelhante (x_V1,y_V1) e (x_V2,y_V2) definem a Viewport
    1. Transladar o objeto de modo que (x_D1,y_D1) caia na origem ($T_x = -x_D1$, $T_y = -y_D1$)
    2. Aplicar a escala deseja (S_x = fator_vis_x e S_y =fator_vis_y) para a janela tornar-se do tamanho da Viewport

$$\text{fator\_vis\_x} = (x_v2 - x_v1) / (x_d2 - x_d1)$$

$$\text{fator\_vis\_y} = (y_v2 - y_v1) / (y_d2 - y_d1)$$

    3. Transladar o objeto de modo que a Viewport fique na sua posição correta na tela ($T_x = x_V1$, $T_y = y_V1$)

Transformação de Janela de Desenho para ViewPort

$$\underbrace{\begin{bmatrix} 1 & 0 & x_V 1 \\ 0 & 1 & y_V 1 \\ 0 & 0 & 1 \end{bmatrix}}_{\substack{\text{Matriz de Translação} \\ \text{(Passo 3)}}} \underbrace{\begin{bmatrix} \dfrac{x_V 2 - x_V 1}{x_D 2 - x_D 1} & 0 & 0 \\ 0 & \dfrac{y_V 2 - y_V 1}{y_D 2 - y_D 1} & 0 \\ 0 & 0 & 1 \end{bmatrix}}_{\substack{\text{Matriz de Escala} \\ \text{(Passo 2)}}} \underbrace{\begin{bmatrix} 1 & 0 & -x_D 1 \\ 0 & 1 & -y_D 1 \\ 0 & 0 & 1 \end{bmatrix}}_{\substack{\text{Matriz de Translação} \\ \text{(Passo 1)}}}$$

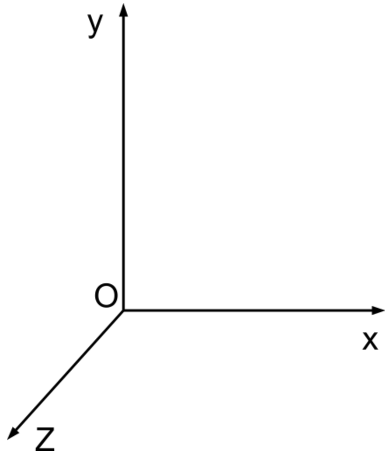
$$\underbrace{\begin{bmatrix} \dfrac{x_V 2 - x_V 1}{x_D 2 - x_D 1} & 0 & -x_D 1\left(\dfrac{x_V 2 - x_V 1}{x_D 2 - x_D 1}\right) + x_V 1 \\ 0 & \dfrac{y_V 2 - y_V 1}{y_D 2 - y_D 1} & -y_D 1\left(\dfrac{y_V 2 - y_V 1}{y_D 2 - y_D 1}\right) + x_V 1 \\ 0 & 0 & 1 \end{bmatrix}}_{\text{Matriz Mapeamento (Window-to-Viewport)}}$$

TRANSFORMAÇÕES 3-D

- Podemos estender os conceitos vistos para 2-D para o 3-D

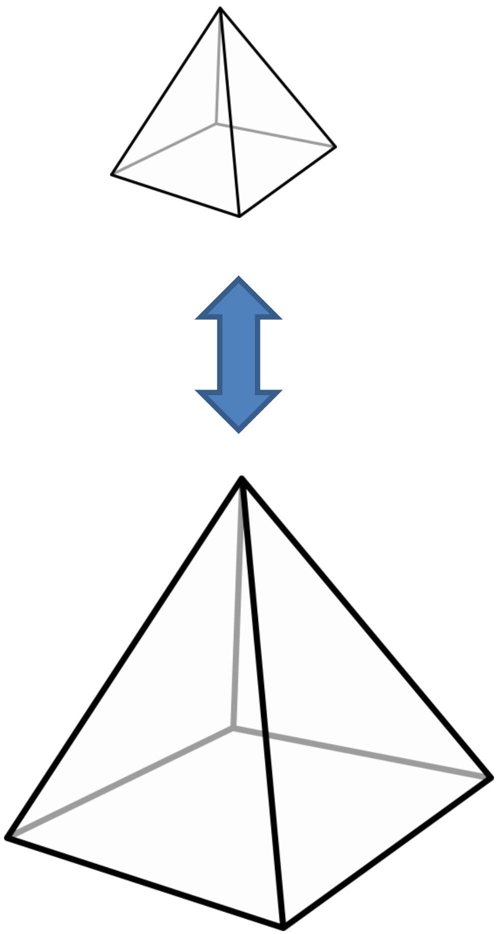
- Em coordenadas homogêneas:
  - Um ponto P de coordenadas (x, y, z), será representado por (x, y, z, 1)
  - A matriz de transformação será de tamanho 4 x 4

- Utilizaremos a Regra da Mão Direita para definir o sistema de coordenadas
  - O eixo Z está apontado (cresce) para a direção do observador



Modificado de http://upload.wikimedia.org/wikipedia/commons/f/fb/Spherical_coordinates.svg

Matriz de Escala 3-D



Modificado de http://upload.wikimedia.org/wikipedia/commons/9/91/Pyramid.svg

$$x' = x * S_x$$
$$y' = y * S_y$$
$$z' = z * S_z$$

$$\begin{bmatrix} x' \\ y' \\ z' \\ w \end{bmatrix} = \begin{bmatrix} xS_x \\ yS_y \\ zS_z \\ 1 \end{bmatrix} = \underbrace{\begin{bmatrix} Sx & 0 & 0 & 0 \\ 0 & Sy & 0 & 0 \\ 0 & 0 & Sz & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}}_{\text{Matriz de Escala 3-D}} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

*Onde a coordenada w será descartada no ponto transformado

Matriz de Escala 3-D com um Ponto Fixo (x_F, y_F, z_F)

$$x' = x * S_x + (1- S_x)x_F$$

$$y' = y * S_y + (1- S_y)y_F$$

$$z' = z * S_z + (1- S_z)z_F$$

$$\begin{bmatrix} x' \\ y' \\ z' \\ w \end{bmatrix} = \begin{bmatrix} xS_x + (1-S_x)x_F \\ yS_y + (1-S_y)y_F \\ zS_z + (1-S_z)z_F \\ 1 \end{bmatrix} = \underbrace{\begin{bmatrix} Sx & 0 & 0 & (1-S_x)x_F \\ 0 & Sy & 0 & (1-S_y)y_F \\ 0 & 0 & Sz & (1-S_z)z_F \\ 0 & 0 & 0 & 1 \end{bmatrix}}_{\text{Matriz de Escala 3-D com relação a um Ponto Fixo}} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

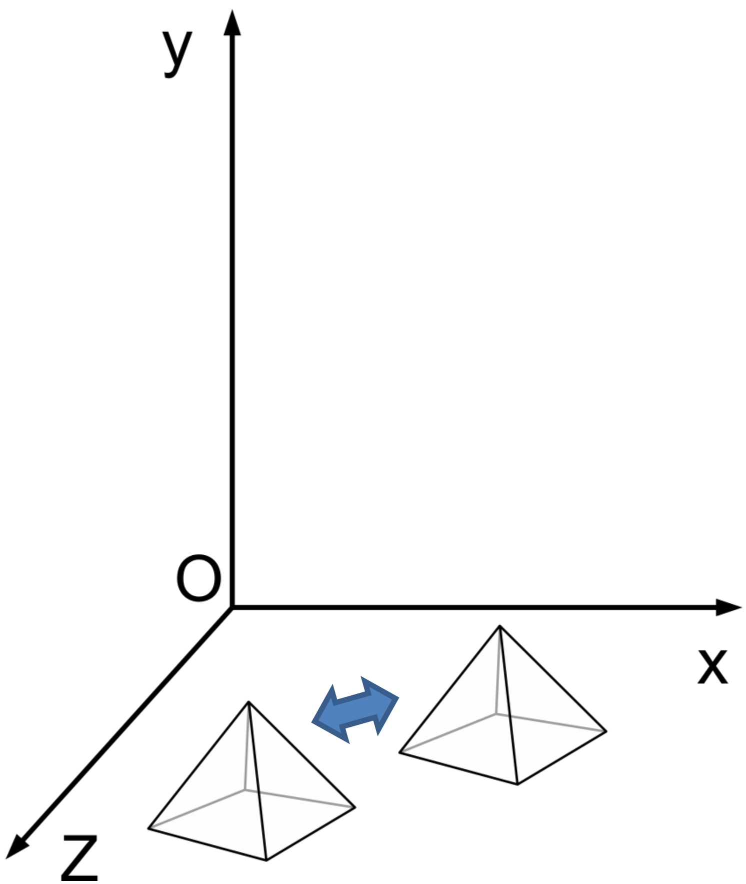
*Onde a coordenada w será descartada no ponto transformado

Matriz de Translação 3-D

$x' = x + T_x$

$y' = y + T_y$

$z' = z + T_z$



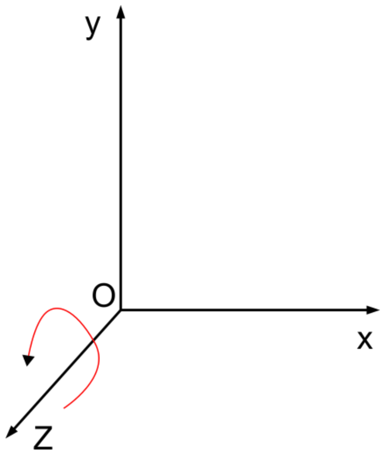
Modificado de http://upload.wikimedia.org/wikipedia/commons/9/91/Pyramid.svg
http://upload.wikimedia.org/wikipedia/commons/f/fb/Spherical_coordinates.svg

$$\begin{bmatrix} x' \\ y' \\ z' \\ w \end{bmatrix} = \begin{bmatrix} x + T_x \\ y + T_y \\ z + T_z \\ 1 \end{bmatrix} = \underbrace{\begin{bmatrix} 1 & 0 & 0 & T_x \\ 0 & 1 & 0 & T_y \\ 0 & 0 & 1 & T_z \\ 0 & 0 & 0 & 1 \end{bmatrix}}_{\text{Matriz de Translação 3-D}} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

*Onde a coordenada w será descartada no ponto transformado

MATRIZ DE ROTAÇÃO 3-D (R_Z)

Modificado de
http://upload.wikimedia.org/wikipedia/commons/f/fb/Spherical_coordinates.svg



$$x' = x*\cos\theta - y*\text{sen}\theta$$

$$y' = y*\cos\theta + x*\text{sen}\theta$$

$$z' = z$$

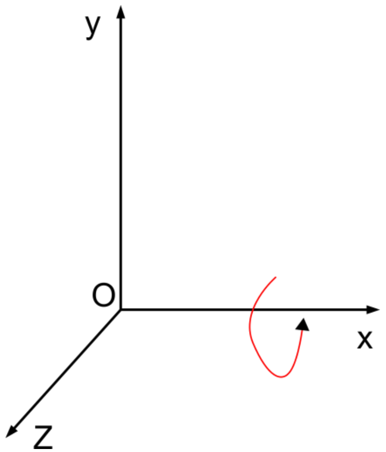
$$\begin{bmatrix} x' \\ y' \\ z' \\ w \end{bmatrix} = \begin{bmatrix} x\cos\theta - y\,\text{sen}\,\theta \\ x\,\text{sen}\,\theta + y\cos\theta \\ z \\ 1 \end{bmatrix} = \underbrace{\begin{bmatrix} \cos\theta & -\text{sen}\,\theta & 0 & 0 \\ \text{sen}\,\theta & \cos\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}}_{\text{Matriz de Rotação 3-D em torno de } O_Z\ (R_z)} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

– **Mas e para Rx e Ry?**

- Fazer permuta cíclica: x->y->z->x

Matriz de Rotação 3-D (R_x)

Modificado de http://upload.wikimedia.org/wikipedia/commons/f/fb/Spherical_coordinates.svg



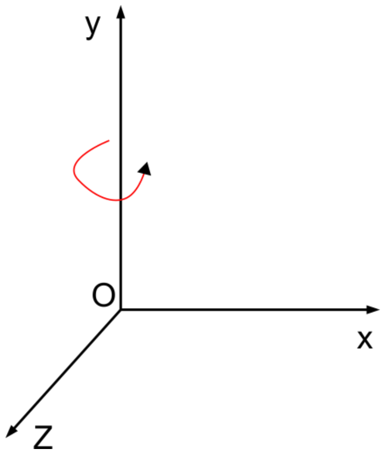
$$y' = y*\cos\theta - z*\text{sen}\theta$$

$$z' = z*\cos\theta + y*\text{sen}\theta$$

$$x' = x$$

$$\begin{bmatrix} x' \\ y' \\ z' \\ w \end{bmatrix} = \begin{bmatrix} x \\ y\cos\theta - z\,\text{sen}\,\theta \\ y\,\text{sen}\,\theta + z\cos\theta \\ 1 \end{bmatrix} = \underbrace{\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\theta & -\text{sen}\,\theta & 0 \\ 0 & \text{sen}\,\theta & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}}_{\substack{\text{Matriz de Rotação 3-D em torno de } O_x \\ (R_x)}} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

Modificado de
http://upload.wikimedia.org/wikipedia/commons/f/fb/Spherical_coordinates.svg



z’ = z*cosθ – x*senθ

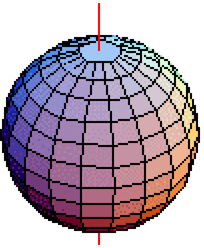
x’ = x*cosθ + z*senθ

y’ = y

$$\begin{bmatrix} x' \\ y' \\ z' \\ w \end{bmatrix} = \begin{bmatrix} x\cos\theta + z\,\mathrm{sen}\,\theta \\ y \\ z\cos\theta - x\,\mathrm{sen}\,\theta \\ 1 \end{bmatrix} = \underbrace{\begin{bmatrix} \cos\theta & 0 & \mathrm{sen}\,\theta & 0 \\ 0 & 1 & 0 & 0 \\ -\mathrm{sen}\,\theta & 0 & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}}_{\substack{\text{Matriz de Rotação 3-D em torno de } O_y \\ (R_y)}} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

Rotação em Torno de Eixo Arbitrário

- Seguindo o que foi feito para o espelhamento 2-D em reta arbritrária, temos:
- Solução:
  - Seja o segmento de reta AB definido pelos pontos (x_A,y_A,z_A) e (x_B,y_B,z_B) a nossa referência para a transformação
    1. Transladar o objeto de modo que um dos pontos A ou B caia na origem
       - Ex: Vamos assumir que A ficará na origem ($T_x = -x_A$, $T_y = -y_A$, $T_z = -z_A$)
    2. Rotacionar em torno do eixo x (eixo de rotação vai para o plano xz)
       - Ex: Vamos assumir a rotação α
    3. Rotacionar em torno do eixo y, coincidindo o eixo de rotação com o eixo O_x
       - Ex: Vamos assumir a rotação β
    4. Rotacionar em torno do eixo z, com o ângulo desejado
       - Ex: Vamos assumir a rotação desejada θ
    5. Rotacionar em torno do eixo y, com o ângulo oposto do realizado no passo (3)
       - Ex: Como em (3) usamos β, agora usaremos $-\beta$
    6. Rotacionar em torno do eixo x, com o ângulo oposto do realizado no passo (2)
       - Ex: Como em (2) usamos α, agora usaremos $-\alpha$
    7. Transladar o objeto com deslocamentos opostos aos realizados em (1), retornando o segmento de reta para a sua posição original
       - Ex: Como assumimos A no passo (1), agora usaremos ($T_x = x_A$, $T_y = y_A$, $T_z = z_A$)



http://upload.wikimedia.org/wikipedia/commons/0/02/Rotating_Sphere.gif

$$M = T(T_x = x_A, T_y = y_A , T_z = z_A) \,.\, R_x(-\alpha) \,.\, R_y(-\beta) \,.\, R_z(\theta) \,.\, R_y(\beta) \,.\, R_x(\alpha) \,.\, T(T_x = -x_A, T_y = -y_A , T_z = -z_A)$$

- Como calcular senos e cossenos de α e β?

- Seja o vetor do eixo de rotação **V**

$$\mathbf{V} = \mathbf{B} - \mathbf{A} = (x_B - x_A, y_B - y_A, z_B - z_A)$$

- Seja **u** o eixo de rotação unitário

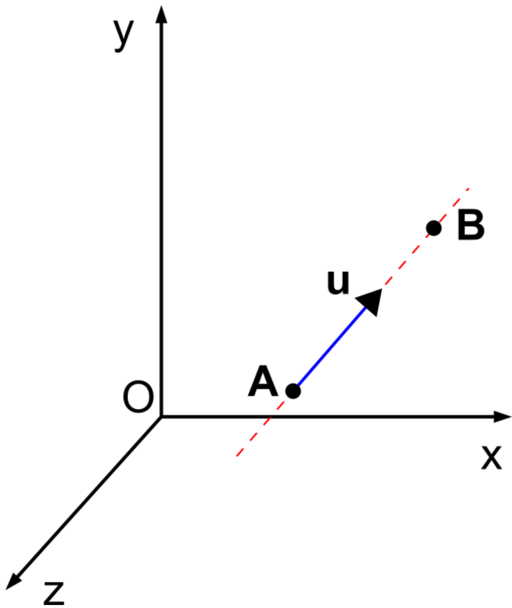
$$\mathbf{u} = \mathbf{V} / |\mathbf{V}| = (a, b, c)$$

$$a = (x_B - x_A) / |V|$$

$$b = (y_B - y_A) / |V|$$

$$c = (z_B - z_A) / |V|$$

$$|V| = \text{sqrt}(a\text{^}2+b\text{^}2+c\text{^}2)$$



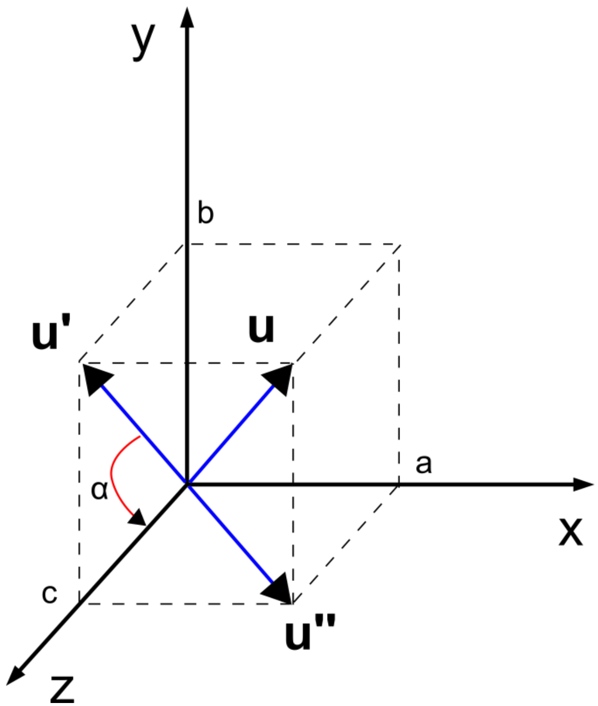
Baseado em HEARN, D. Computer Graphics with OpenGL. 3rd ed. Upper Saddle River, NJ: Pearson Education, 2004. 857 p.

Rotação em Torno de Eixo Arbitrário

- Considere o resultado após a translação

- Seja o **u'**(0, b, c) a projeção de **u** no plano yz

$$|\mathbf{u'}| = \text{sqrt}(b\text{^}2 + c\text{^}2) = d$$



$$\text{sen } \alpha = b / d$$

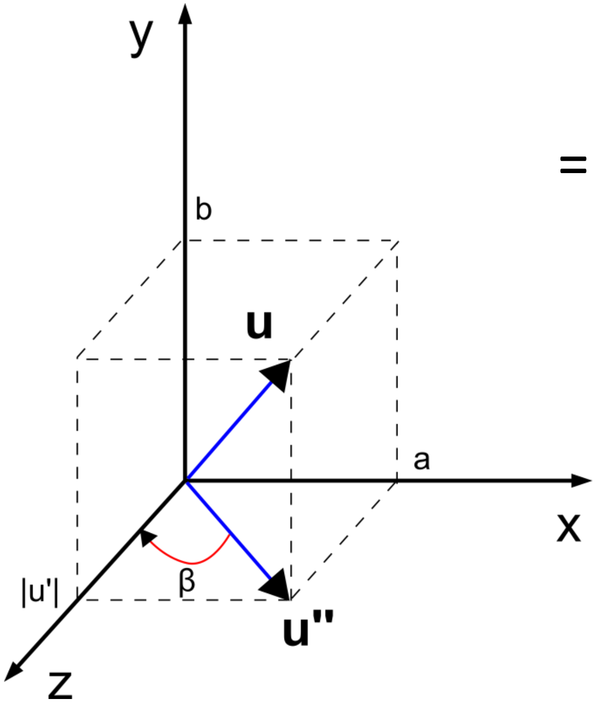
$$\cos \alpha = c / d$$

Baseado em ROMERO, T. et al. Fundamentos de Computação Gráfica, LTC, 1987.
HEARN, D. Computer Graphics with OpenGL. 3rd ed. Upper Saddle River, NJ: Pearson Education, 2004. 857 p.

Rotação em Torno de Eixo Arbitrário

- Considere o resultado após a rotação ao redor de X
  - Lembrar que **u'** será rotacionado para ficar sobre O_z

- Seja o **u''**(a, 0, d) obtido a partir da rotação de u ao redor de x



$$|\mathbf{u''}| = sqrt(a\hat{}2 + d\hat{}2)$$
$$= sqrt(a\hat{}2 + sqrt(b\hat{}2 + c\hat{}2)\hat{}2)$$
$$= sqrt(a\hat{}2 + b\hat{}2 + c\hat{}2)$$
$$= |\mathbf{u}| = 1$$

$$\text{sen } \beta = a / |\mathbf{u''}| = a$$
$$\cos \beta = d / |\mathbf{u''}| = d$$

Baseado em ROMERO, T. et al. Fundamentos de Computação Gráfica, LTC, 1987.
HEARN, D. Computer Graphics with OpenGL. 3rd ed. Upper Saddle River, NJ: Pearson Education, 2004. 857 p.

ESPELHAMENTO

- Como definido anteriormente, o espelhamento consiste em uma rotação de 180° ao redor de um eixo qualquer.

- Desta forma, podemos obter a sua matriz de transformação a partir da Rotação 3-D em torno de eixo arbitrário

- Basta usar θ = 180° na expressão anterior, obtendo:

$$M = T(T_x = x_A, T_y = y_A, T_z = z_A) \cdot R_x(\alpha) \cdot R_y(\beta) \cdot R_z(180°) \cdot R_y(-\beta) \cdot R_x(-\alpha) \cdot T(T_x = -x_A, T_y = -y_A, T_z = -z_A)$$

Exercício

- Considere um polígono com os seguintes vértices: V1(1,4); V2(3,7); V3(4,5); V4(6,9); V5(7,4); V6(4,1). Desejamos transladar a figura de tal forma que o vértice V1 fique no ponto (5,5), depois aumentar a figura em 40% em relação ao ponto E (6,8) e em seguida rotacioná-lo de 30° em torno do ponto R (8,10). Qual seria a transformação composta, neste caso? Quais seriam as novas coordenadas dos vértices? Faça o desenho inicial e final.

IMPLEMENTAÇÃO DOS ALGORITMOS VISTOS