

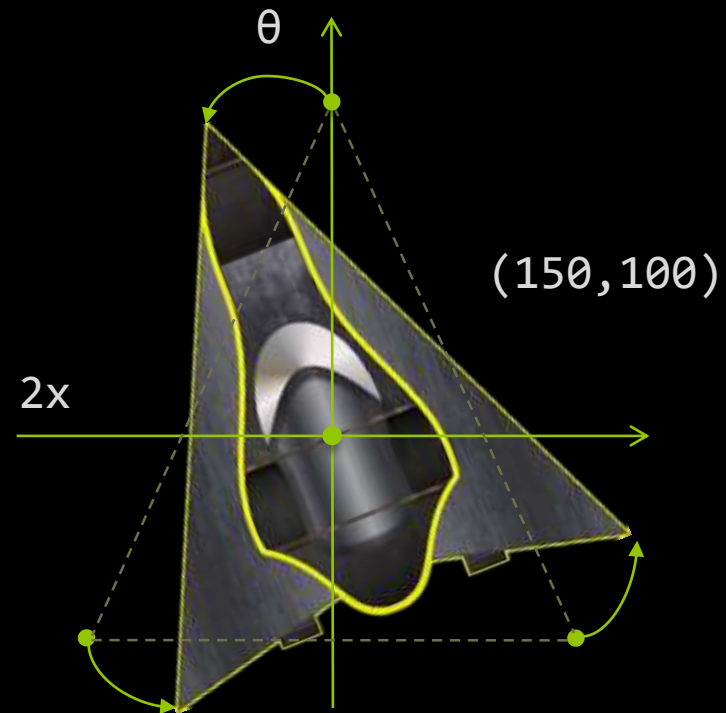
# Transformações

Programação de Jogos

Judson Santos Santiago

# Introdução

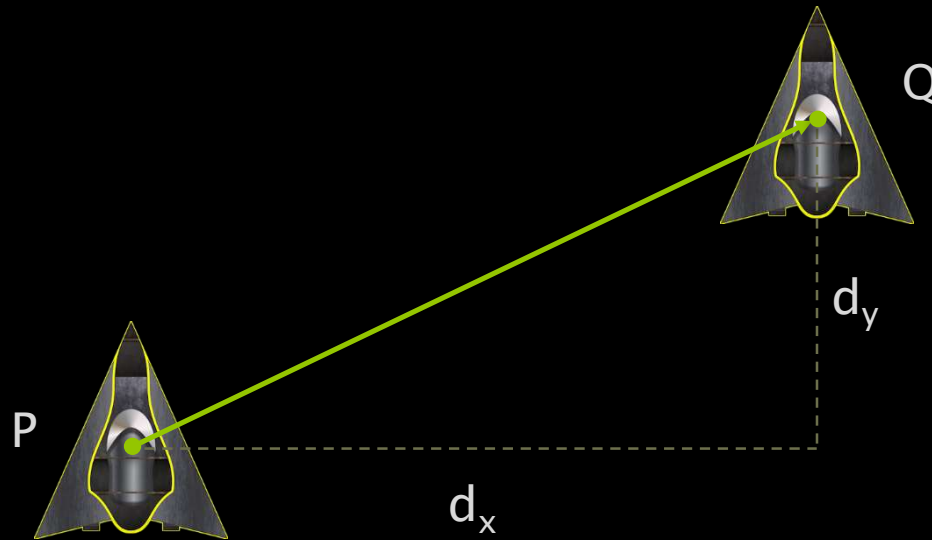
- ▶ **Transformações** são operações matemáticas aplicadas aos objetos do jogo
- ▶ As transformações modificam três propriedades dos objetos:
  - Posição
  - Rotação
  - Escala



# Transformação

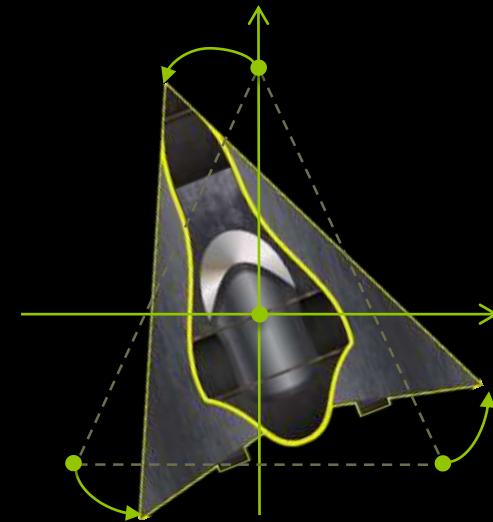
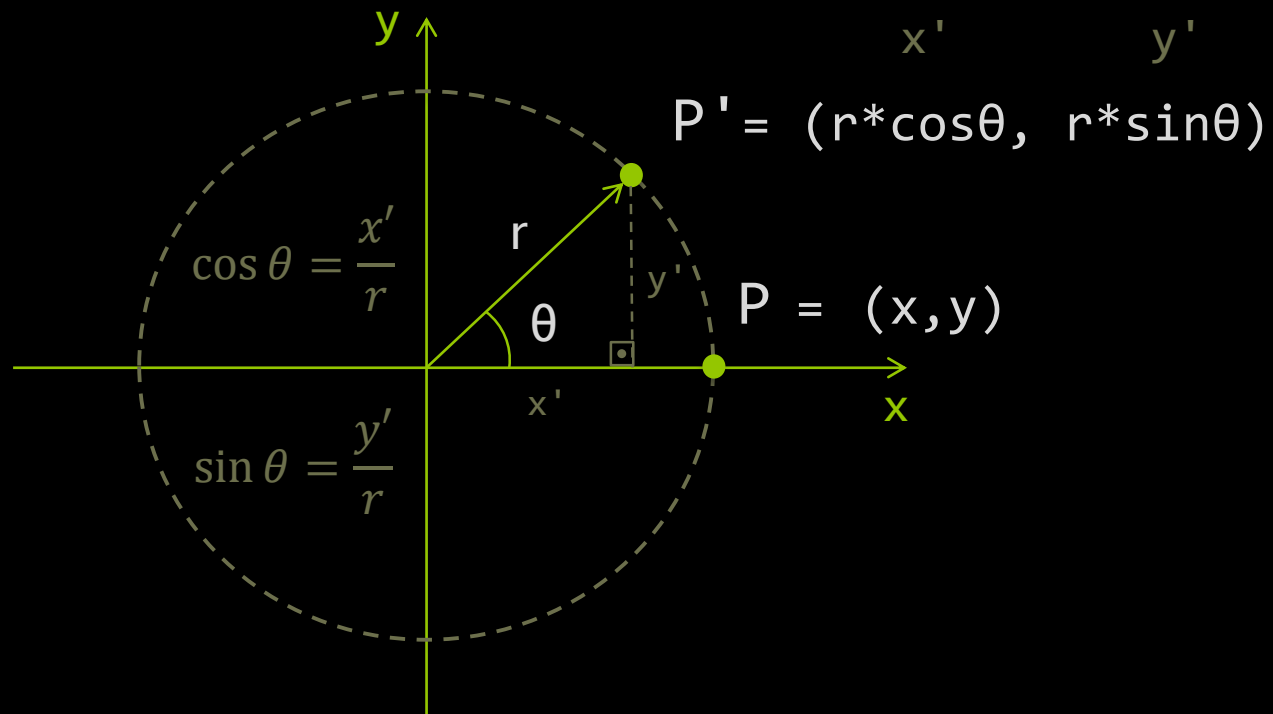
- **Translação:** move o objeto por um determinado valor delta (deltaX, deltaY) em relação a sua posição atual

$$\begin{aligned}Q_x &= P_x + d_x \\ Q_y &= P_y + d_y\end{aligned}$$



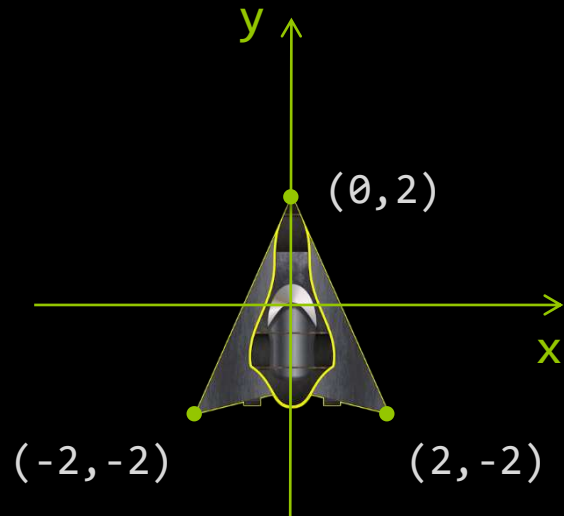
# Transformação

- **Rotação:** rotaciona o objeto por um determinado ângulo, normalmente expresso em radianos

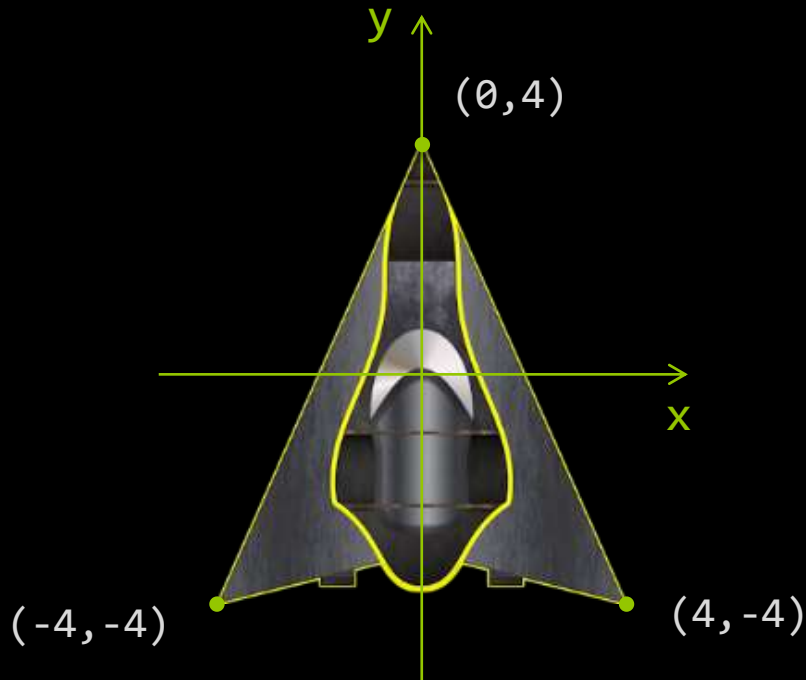


# Transformação

- **Escala:** expande ou reduz o objeto por um determinado fator, chamado fator de escala



2x



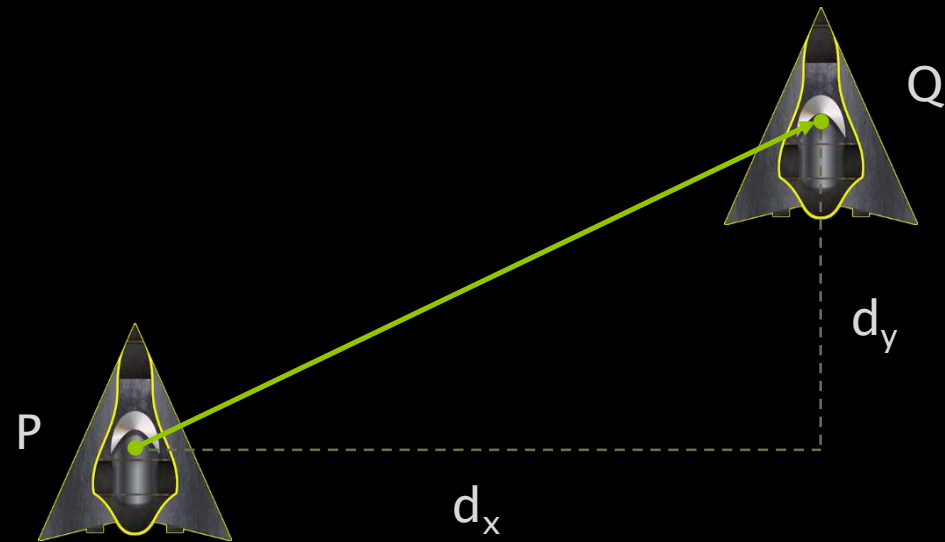
# Matriz de Transformação

- ▶ Por uma questão de desempenho as transformações normalmente são feitas usando **matrizes**
  - **Translação**: move o objeto por um  $(d_x, d_y)$

$$P = [x \ y \ 1] \quad T = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ d_x & d_y & 1 \end{bmatrix}$$

$$Q = P * T$$

$$Q = [x+d_x \ y+d_y \ 1]$$



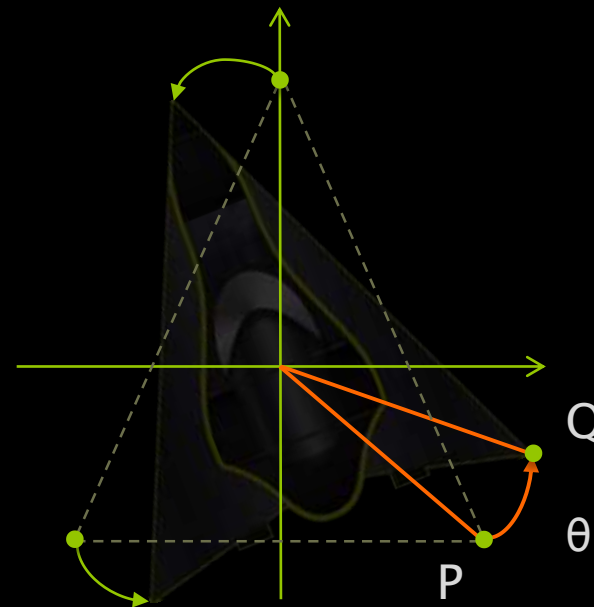
# Matriz de Transformação

- ▶ Por uma questão de desempenho as transformações normalmente são feitas usando **matrizes**
  - **Rotação**: rotaciona o objeto por um ângulo  $\theta$

$$P = [x \ y \ 1] \quad R = \begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$Q = P * R$$

$$Q = [x\cos\theta - y\sin\theta \quad x\sin\theta + y\cos\theta \quad 1]$$



# Matriz de Transformação

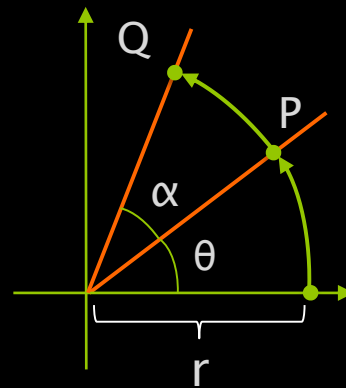
- A equação da rotação é obtida a partir de:

$$Q_x = r \cos(\theta + \alpha)$$

$$Q_y = r \sin(\theta + \alpha)$$

$$Q_x = r \cos\theta \cos\alpha - r \sin\theta \sin\alpha$$

$$Q_y = r \sin\theta \cos\alpha + r \cos\theta \sin\alpha$$



$$P_x = r \cos\theta$$

$$P_y = r \sin\theta$$

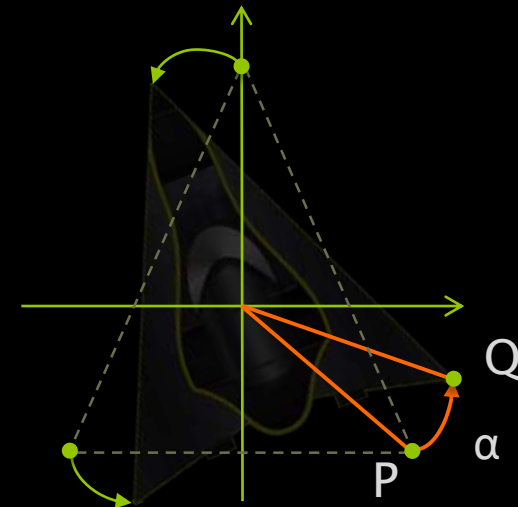
$$Q_x = P_x \cos\alpha - P_y \sin\alpha$$

$$Q_y = P_y \cos\alpha + P_x \sin\alpha$$

Identidades trigonométricas:

$$\cos(\theta + \alpha) = \cos(\theta) \cos(\alpha) - \sin(\theta) \sin(\alpha)$$

$$\sin(\theta + \alpha) = \sin(\theta) \cos(\alpha) + \cos(\theta) \sin(\alpha)$$





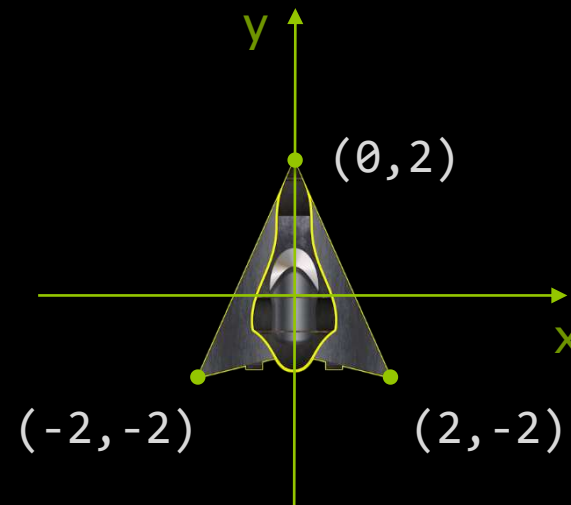
# Matriz de Transformação

- ▶ Por uma questão de desempenho as transformações normalmente são feitas usando **matrizes**
  - **Escala**: expande ou reduz o objeto

$$P = [x \ y \ 1] \quad S = \begin{bmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$Q = P * S$$

$$Q = [xS_x \ yS_y \ 1]$$

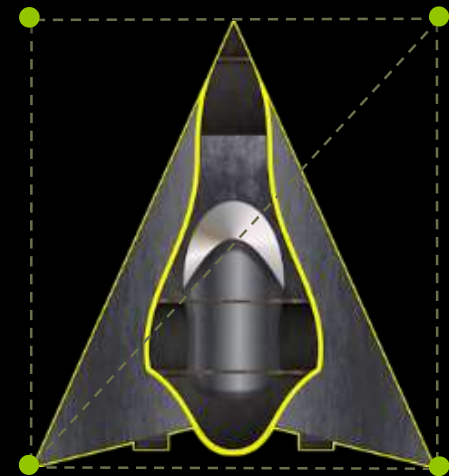


# Implementação no Motor

- ▶ Uma **matriz de transformação única** pode ser obtida pela combinação das matrizes de escala, rotação e translação:

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

- ▶ Os objetos são representados por Sprites
  - Sprites são **Quads Texturizados**
  - Transformar um objeto significa aplicar a matriz de transformação sobre cada vértice do objeto



# Resumo

- ▶ As transformações alteram as propriedades de:

- Posição
- Escala
- Rotação

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

- ▶ Tudo que o programador precisa fazer é criar uma **matriz de transformação** e aplicá-la aos vértices que definem os objetos

