

# Manual para o Trabalho Prático

Prof. Dr. Bianchi Serique Meiguins

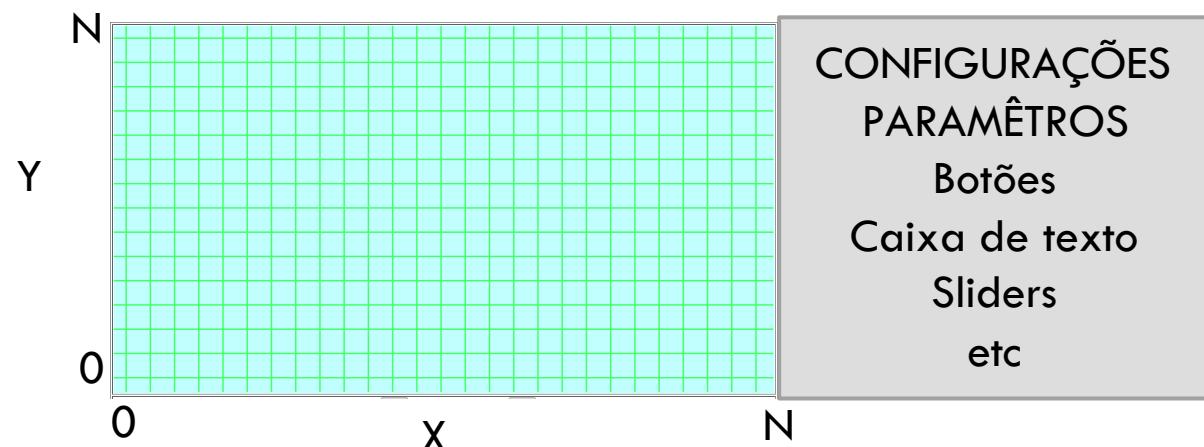
Prof. Dr. Carlos Gustavo Resque dos Santos ([gustavoresqueufpa@gmail.com](mailto:gustavoresqueufpa@gmail.com))

# Visão Geral

- O trabalho é composto de duas partes principais:
- Síntese de Imagem (rasterização)
- Processamento de Imagem

# Visão Geral – Síntese de Imagem

- Este modulo, como sugestão, deve conter duas partes principais:
- Uma área a onde será desenhado a saída dos algoritmos



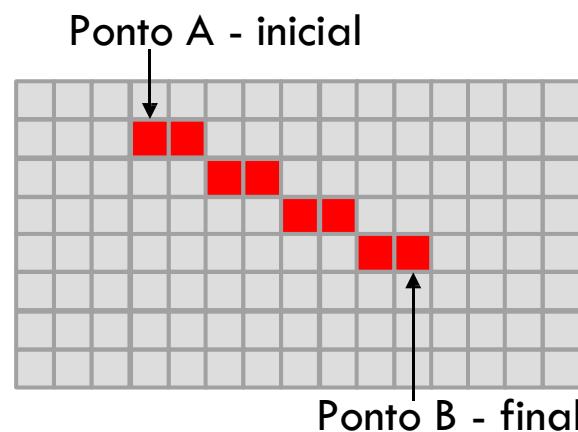
- Uma interface de usuário que permita modificar parâmetros para todos os algoritmos solicitados
- Obs: o sistema de coordenadas deverá considerar todos os quadrantes, incluindo coordenadas negativas.
- Valores de coordenadas  $-11 < x < 11$  e  $-11 < y < 11$

# Visão Geral – Síntese de Imagem

- Obs1: A seguir serão apresentados os algoritmos que devem ser implementados e sua pontuação
- Obs2: As imagens são apenas ilustrativas e não são a saída efetivamente calculada de um algoritmo
- Obs3: Ao apresentar o trabalho em formato de video, explique o código fonte e mostre a saída quando o algoritmo é executado através dessa interface gráfica

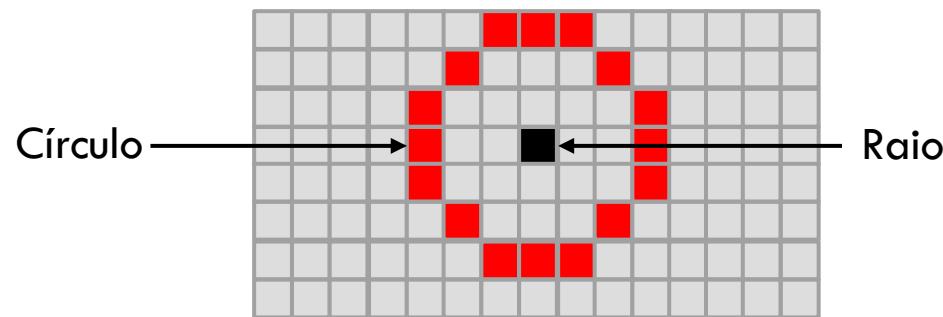
# Visão Geral – Síntese de Imagem

- **Bresenham** [0.5],
- Entrada: especificar dois pontos quaisquer (X,Y) na área de configurações, ou diretamente na área de desenho, no escopo do sistema de coordenadas da grade.
- Saída:



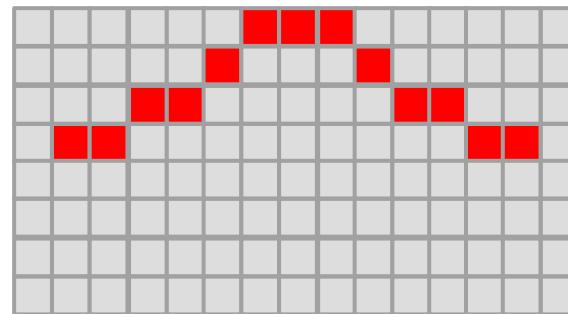
# Visão Geral – Síntese de Imagem

- **Círculos [0,.5] e Elipses [0.5]**
- Entrada: especificar raio e centro (X,Y) na área de configuração, ou diretamente na área de desenho, no escopo do sistema de coordenadas da grade.
- Saída



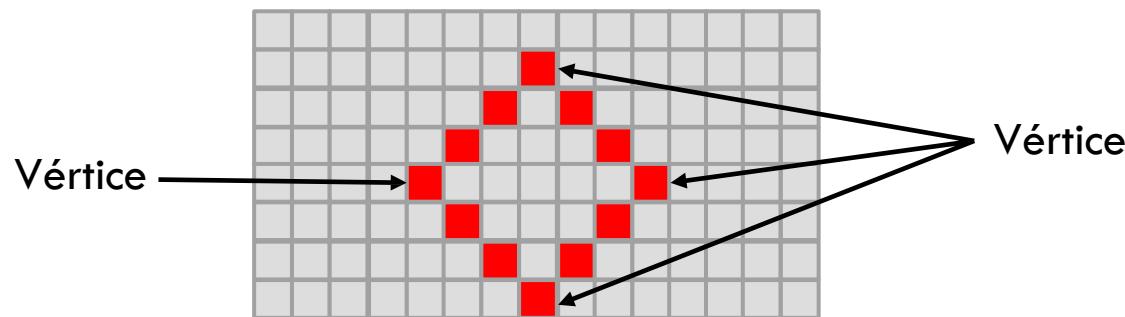
# Visão Geral – Síntese de Imagem

- **Curvas de Bezier graus 2 [0.5] e 3 [0.5]**
- Entrada: parâmetros para calcular uma curva Bezier, pontos inicial, final e controles – rasterizar resultados com bresenham
- Saída



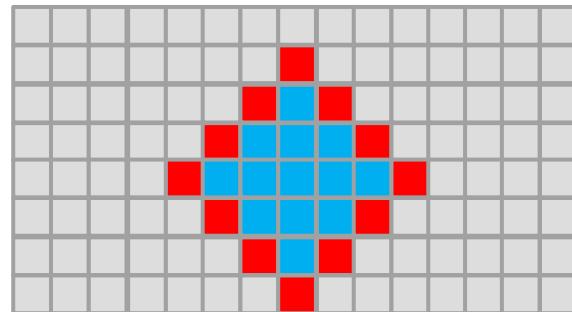
# Visão Geral – Síntese de Imagem

- **Polilinha [0,5],**
- Entrada: especificar na área de configuração um conjunto de  $N > 3$  pontos  $(X, Y)$  quaisquer no escopo do sistema de coordenadas da grades
- Saída:



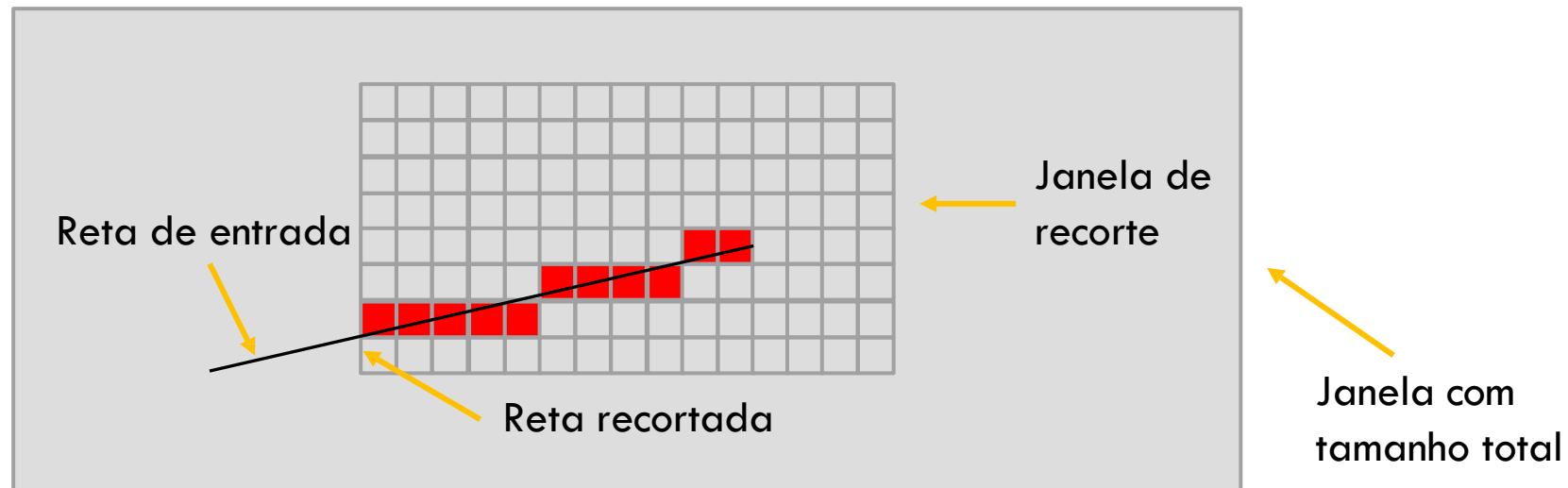
# Visão Geral – Síntese de Imagem

- **Preenchimento Recursivo [0.25] e Varredura [0.25]**
- Entrada 1: especificar um ponto aleatório qualquer (X,Y) no interior de polígono qualquer.
- Entrada 2: Especificar um polígono não regular e aplicar o algoritmo de varredura para preenchimento
- Saída:



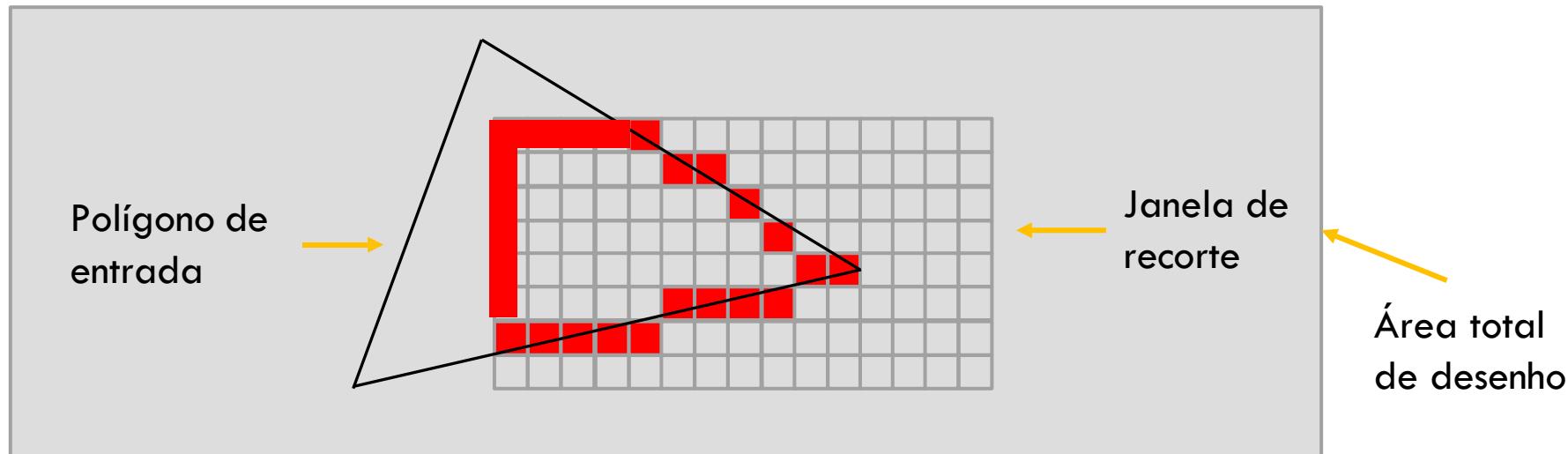
# Visão Geral – Síntese de Imagem

- **Recorte de linha [0.25]**
- Entrada: especificar uma janela de recorte menor que área de desenho.
- Saída: qualquer linha desenhada que esteja fora da janela ou tenha parte fora, será recortada.



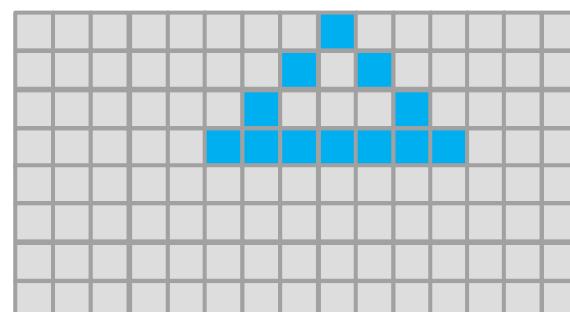
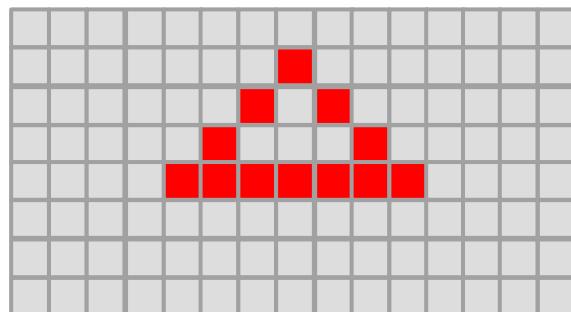
# Visão Geral – Síntese de Imagem

- **Recorte de polígono [0.5]**
- Entrada: especificar uma janela de recorte menor que área de desenho.
- Saída: qualquer polígono desenhado que esteja fora da janela ou tenha parte fora, será recortado.



# Visão Geral – Síntese de Imagem

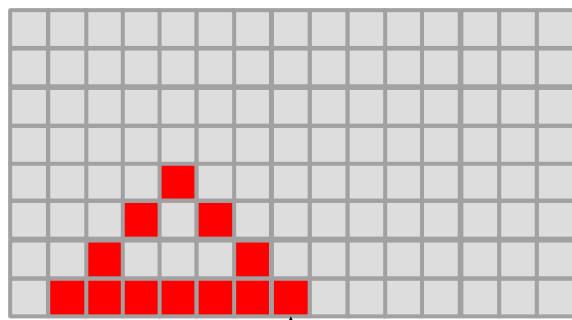
- **Transformações** (3 básicas [3x0.5])
- Entrada: um polígono de vértices quaisquer dentro do sistema de coordenadas da grade
- Rotação entrada - ângulo de rotação e ponto de pivô
- Translação entrada – deslocamento na direção X e Y
- Escala entrada – fator de escala para X e Y e ponto fixo



Translação deslocamento X=1 e Y =1

# Visão Geral – Síntese de Imagem

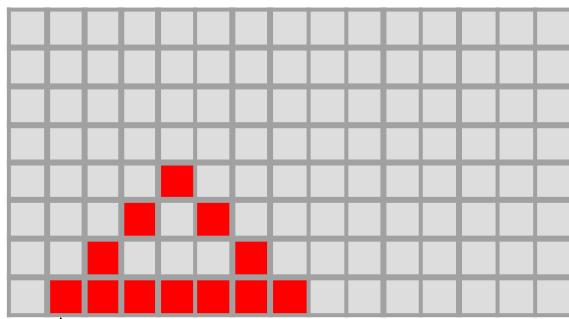
Rotação



Pivô

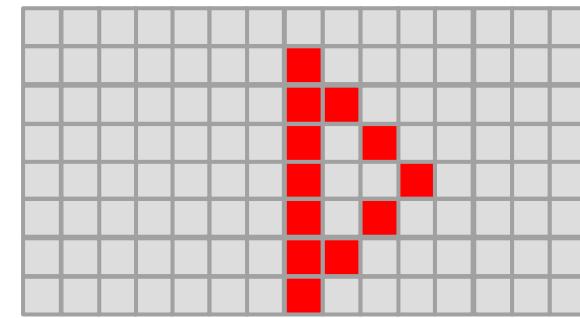
Ângulo = - 90 graus

Escala



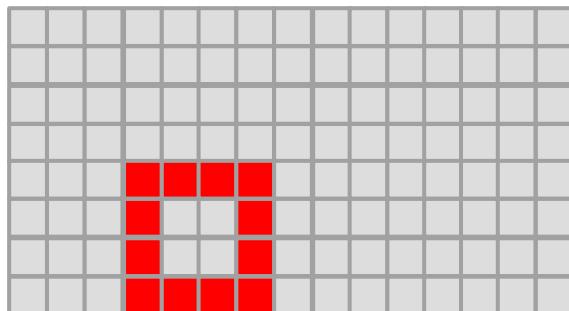
fixo

Fator X = 1 fator Y = 2

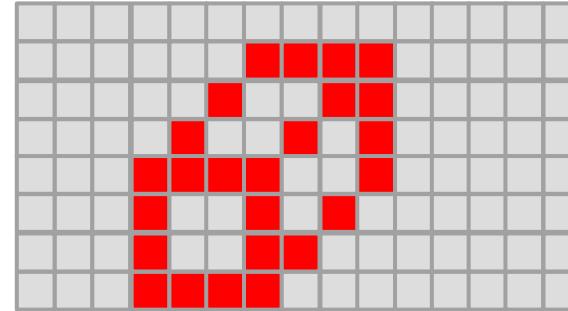


# Visão Geral – Síntese de Imagem

- **Projeções Orto.** [0.5], **Oblíqua**[0,5] e **Perspec.** [0.5]).
- Entrada: especificar(vértices) um sólido (3D) qualquer, especificar a projeção a ser utilizada, rasterizar o resultado com bresenham
- Saída: exemplo de entrada Cubo



Ortogonal - Frontal



Perspectiva

# Visão Geral – Processamento de Imagem

## □ Interface básica



# Visão Geral – Processamento de Imagem

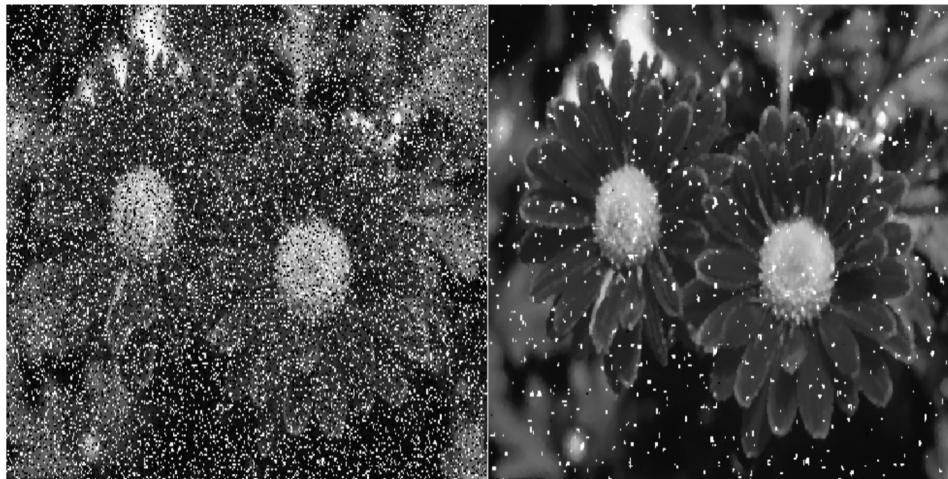
- Conversão entre tipos de imagem (1,0)
- RGB → Tons de Cinza → Binária
- Binária → Tons de Cinza → RGB



- Média (0.25), Mediana(0.25), Guassiano(0.5)

# Visão Geral – Processamento de Imagem

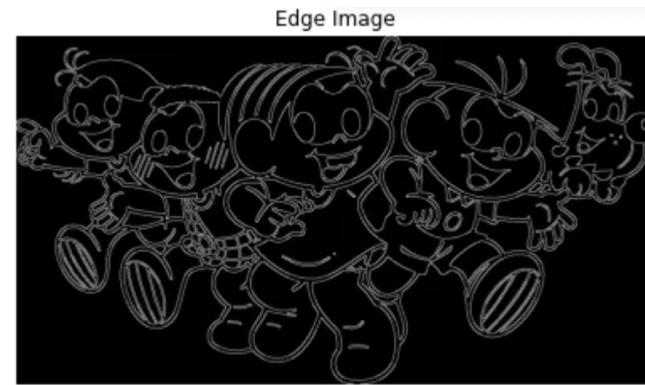
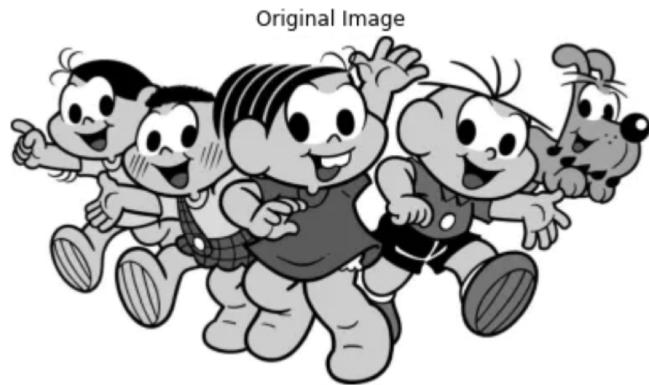
- implementação de um algoritmo para remoção de ruído em imagens



- Média (0.25), Mediana(0.25), Guassiano(0.5)

# Visão Geral – Processamento de Imagem

- implementação de um algoritmo para destaque de bordas em imagens



- Sobel (0.25), Prewitt(0.25), Canny (0.5)

# Resumo de Entrega

Algoritmos de síntese de imagem:

- Bresenham;
  - Círculo e elipse;
  - Curva de Bézier;
  - Polilinha;
  - Preenchimento recursivo e varredura;
  - Recorte de linha e de polígono;
  - Transformações;
  - Projeções.
- 
- Vídeo explicativo e código fonte

# Resumo de Entrega

Algoritmos processamento de imagem:

- Conversão entre tipos de imagens;
  - Remoção de ruídos;
  - Destaque de bordas.
- 
- Vídeo explicativo e código fonte

# Data de Entrega

- Data limite: 31/10/2024 até 23:59 via SIGAA
- Deve-se anexar o código ou link do GitHUB ou similar, e o link do vídeo no youtube;
- Caso a postagem no SIGAA seja feita até dia 28/10/2024 23:59, haverá 1.0 ponto extra na terceira avaliação;
- No máximo em dupla, nomes devem estar explícitos no SIGAA, bastando apenas uma postagem, e só será permitido postar 1 vez;
- Códigos iguais serão descontados 50% da nota na segunda ocorrência.