

IA725 Computação Gráfica I

1º semestre de 2006

Profª. Wu, Shin-Ting
ting@dca.fee.unicamp.br
Bloco A – sala 317

Prof. José Mario De Martino
martino@dca.fee.unicamp.br
Bloco A – sala 317-A

Agenda

Aula	Dia	Tema	Projeto
17	09/05/06	Cor	
18	12/05/06	Cor	
19	16/05/06	Iluminação	
20	19/05/06	Iluminação	
21	23/05/06	Iluminação	
22	26/05/06	Textura	
23	30/05/06	Textura	Versão 0.2
24	02/06/06	Rasterização	
25	06/06/06	Rasterização	
26	09/06/06	Recorte	
27	13/06/06	Recorte	
-	16/06/06		
28	20/06/06	Visibilidade	
29	23/06/06	Visibilidae	
30	27/06/06	Prova	Versão 0.3
	30/junho		Versão Final

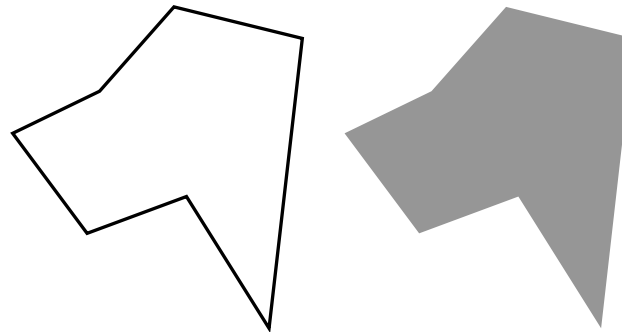
Rasterização

- Referências

- Computer Graphics Principles and Practice (2nd Edition)
- J. D. Foley, A. van Dam, S. K. Feiner, J. F. Hughes
- Addison-Wesley – 1990
- Procedural Elements for Computer Graphics
- D. F. Rogers
- McGraw-Hill – 1988
- 3D Computer Graphics (3rd Edition)
- Alan Watt
- Addison-Wesley – 2000

Algoritmos de preenchimento de polígonos

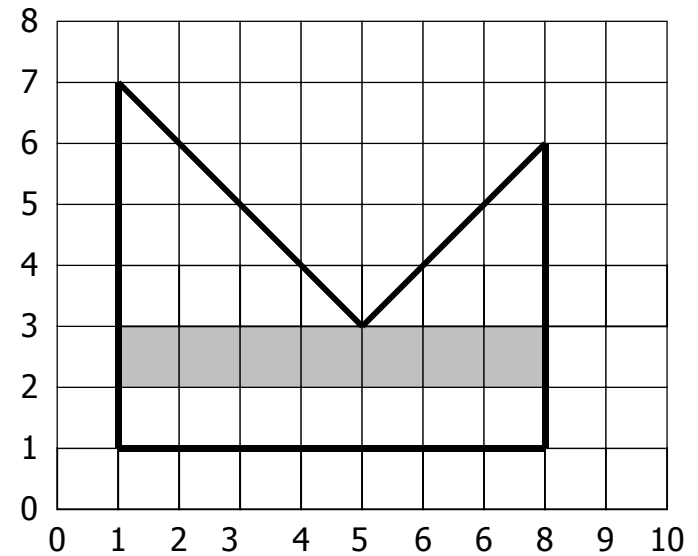
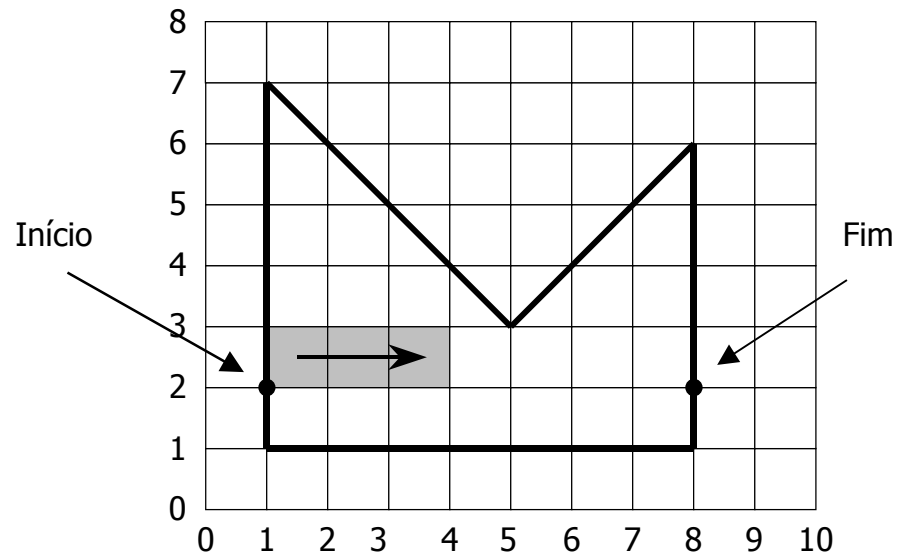
- Apenas traçar as arestas dos polígonos que compõem um objeto pode não ser indicado para todas aplicações.
- Uma forma alternativa de apresentação é a apresentação dos polígonos como área preenchidas.



Algoritmos de preenchimento de polígonos

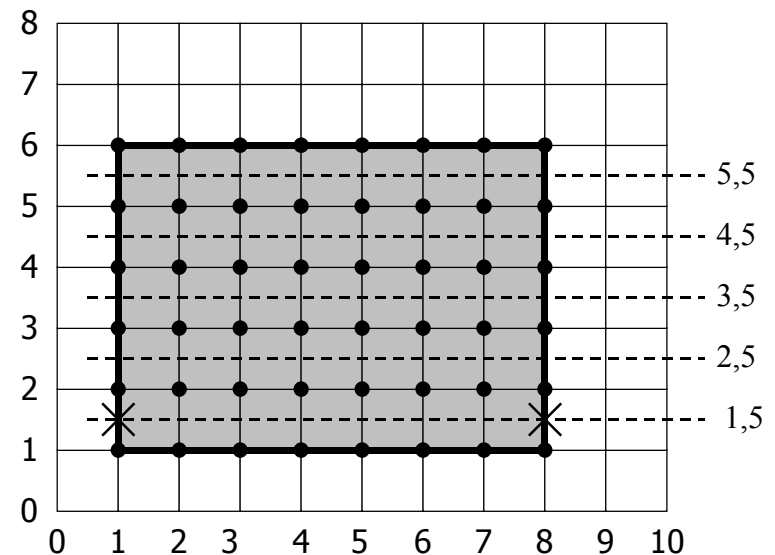
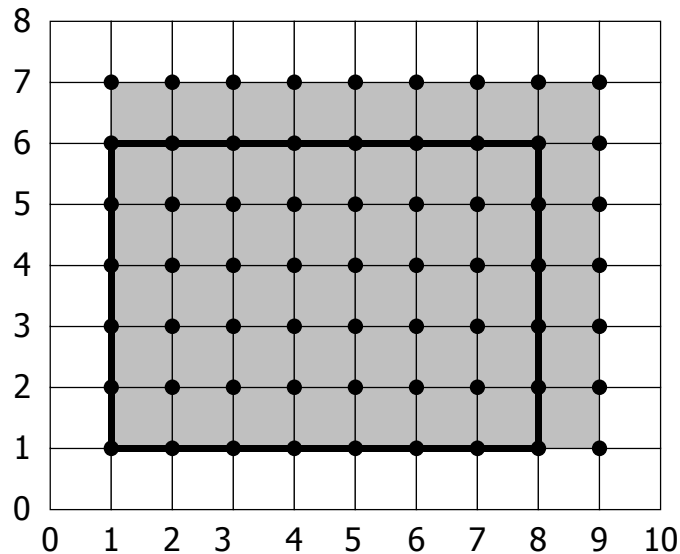
- Estratégia sobre o preenchimento de áreas de polígonos.
 - Calcular interseção das arestas com as linhas de varredura (*scanlines*). Observar que se tem um número par de interseções.
 - Ordenar em x crescente.
 - Preencher entre duas interseções consecutivas.

Algoritmos de preenchimento de polígonos



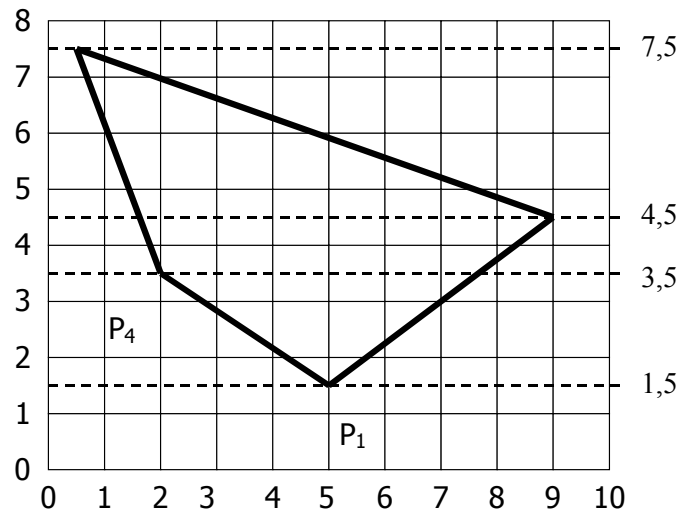
Algoritmos de preenchimento de polígonos

- A determinação dos *pixels* a serem ativados exige a interseção com as linhas de varredura intermediárias e correção para as bordas da direita.



Algoritmos de preenchimento de polígonos

- Vértices localizados exatamente nas linhas de varreduras intermediárias introduzem singularidades.
 - Por exemplo: Na linha 3.5 teríamos 3 interseções.



- Solução:
 - Se ponto de máximo/mínimo $\Rightarrow$ 2 interseções.
 - Caso contrário apenas 1.
 - Ponto máximo/mínimo: os dois outros vértices das arestas possuem ambos coordenadas y maiores/menores que a atual (estão de mesmo lado da linha de varredura).

Algoritmos de preenchimento de polígonos

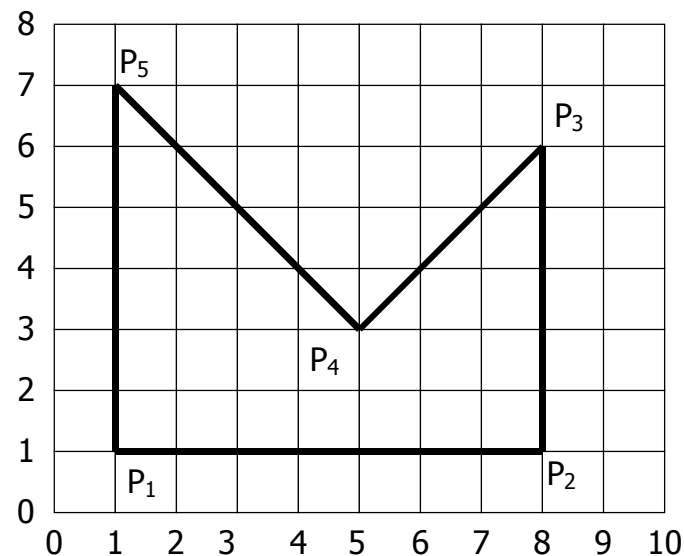
- Algoritmo Lista Ordenada de Arestas (*Ordered Edge List*) - versão simplificada
 - Passos:
 - Determinar para cada arestas a interseção com as linhas de varreduras intermediárias (utilizar DDA ou Bresenham). Arestas horizontais são ignoradas. Armazenar as interseções em uma lista.
 - Ordenar lista por linha de varredura e x crescente. Ou seja, (x_1, y_1) vem antes de (x_2, y_2) se:
 - $y_1 > y_2$ ou
 - $y_1 = y_2$ e $x_1 \leq x_2$.
 - Extrair, aos pares, os elementos da lista ordenada. Ativar os *pixels* da linha de varredura y para todos os valores de x que satisfaçam:

$$x_1 \leq x + \frac{1}{2} \leq x_2$$

Algoritmo de Lista Ordenada de Arestas - simplificado

- Exemplo

- $P_1(1,1)$ $P_2(8,1)$ $P_3(8,6)$ $P_4(5,3)$ $P_5(1,7)$



Algoritmo de Lista Ordenada de Arestas - simplificado

- Interseções com as linhas de varreduras intermediárias

Varredura

1,5	(8; 1,5)	(1; 1,5)		
2,5	(8; 2,5)	(1; 2,5)		
3,5	(8; 3,5)	(5,5; 3,5)	(4,5; 3,5)	(1; 3,5)
4,5	(8; 4,5)	(6,5; 4,5)	(3,5; 4,5)	(1; 4,5)
5,5	(8; 5,5)	(7,5; 5,5)	(2,5; 5,5)	(1; 5,5)
6,5	(1,5; 6,5)	(1; 6,5)		
7,5	-			

Algoritmo de Lista Ordenada de Arestas - simplificado

- Lista ordenada:

(1; 6,5) (1,5; 6,5) (1; 5,5) (2,5; 5,5) (7,5; 5,5) (8; 5,5)

(1; 4,5) (3,5; 4,5) (6,5; 4,5) (8; 4,5) (1; 3,5) (4,5; 3,5) (5,5; 3,5)

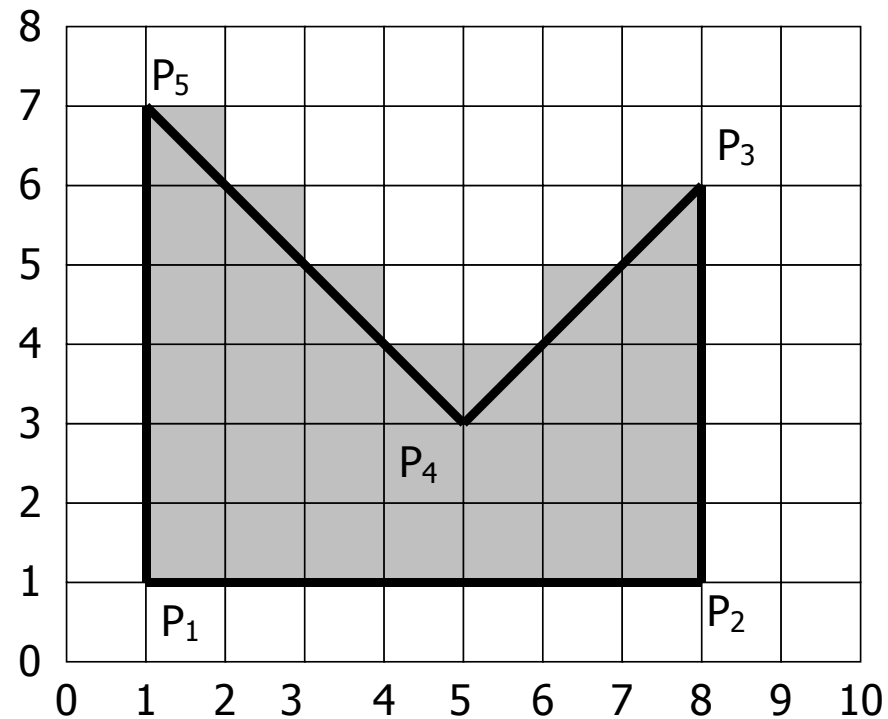
(8; 3,5) (1; 2,5) (8; 2,5) (1; 1,5) (8; 1,5)

Algoritmo de Lista Ordenada de Arestas - simplificado

- Extração das interseções aos pares e ativação dos *pixels*

(1; 6,5)	(1,5; 6,5)	(1,6)
(1; 5,5)	(2,5; 5,5)	(1,5) (2,5)
(7,5; 5,5)	(8; 5,5)	(7,5)
(1; 4,5)	(3,5; 4,5)	(1,4) (2,4) (3,4)
(6,5; 4,5)	(8; 4,5)	(6,4) (7,4)
(1; 3,5)	(4,5; 3,5)	(1,3) (2,3) (3,3) (4,3)
(5,5; 3,5)	(8; 3,5)	(5,3) (6,3) (7,3)
(1; 2,5)	(8; 2,5)	(1,2) (2,2) (3,2) (4,2) (5,2) (6,2) (7,2)
(1; 1,5)	(8; 1,5)	(1,1) (2,1) (3,1) (4,1) (5,1) (6,1) (7,1)

Algoritmo de Lista Ordenada de Arestas - simplificado



Algoritmo de Lista Ordenada de Arestas - simplificado

- Observação final: O algoritmo, como proposto, gera uma lista que precisa ser ordenada. Dependendo da complexidade da cena a lista pode crescer demasiadamente, implicando em grande consumo de memória e um processo de ordenação lento.
- Solução: Algoritmo Lista Ordenada de Arestas com lista de arestas ativas (*Ordered Edge List with Active Edge List*)

Algoritmo de Lista Ordenada de Arestas – lista ativa

- O algoritmo “Lista Ordenada de Arestas com lista de arestas ativas” se apoia em duas estruturas de dados principais: y-bucket e lista ativa
- O y-bucket é uma estrutura com uma entrada para cada linha de varredura. Cada entrada contém as informações das arestas que se iniciam naquela linha. Por exemplo, para um imagem 640x480, teríamos uma estrutura com 480 entradas.
- Cada entrada do y-bucket é uma lista ligada contendo:
 - x - valor inicial da aresta na direção x .
 - Δx - valor do incremento em x .
 - Δy - número de incrementos em y .

Algoritmo de Lista Ordenada de Arestas – lista ativa

- A lista ativa contém, durante o processamento, a lista das aresta que interceptadas pela linha de varredura atual (a que está sendo processada).
- Cada aresta da lista ativas é representada por 3 parâmetros:
 - x - valor da interseção em x com a próxima linha (intermediária) de varredura.
 - Δx - valor do incremento em x
 - Δy - número de incrementos em y que ainda faltam.

Algoritmo de Lista Ordenada de Arestas – lista ativa

- Algoritmo
 1. Determinar para cada aresta a linha de varredura mais alta (maior y), cuja linha de varredura intermediária ($y + 1/2$) intercepta a aresta. Colocar a aresta na posição y do y-bucket $(x, \Delta x, \Delta y)$.
 2. Para cada linha de varredura em ordem e começando com a mais alta, verifique se há aresta na posição y-bucket associada. Se houver, adicione a aresta na lista de arestas ativas.
 3. Ordenar em x crescente a lista de arestas ativas.

Algoritmo de Lista Ordenada de Arestas – lista ativa

4. Extrair arestas da lista de aresta ativas aos pares e ativar os *pixels*

$$x_1 \leq x + \frac{1}{2} \leq x_2$$

5. Para cada aresta decrementar Δy da lista ativa.

- Se $\Delta y = 0 \Rightarrow$ retirar aresta da lista de arestas ativas
- Caso contrário, atualizar x na lista de arestas ativas:

$$x += \Delta x$$

6. Se ainda houver linha de varredura a processar, inserir , se houver, arestas da linha de varredura ativa e voltar ao passo 3.

Algoritmo de Lista Ordenada de Arestas – lista ativa

- Exemplo:

