Grupo II – Preenchimento e Recorte

3. Algoritmo Varrimento (*scan-line*)

Recorra ao algoritmo *scan-line* para calcular as coordenadas dos *pixels* de preenchimento da área bidimensional definida pelo polígono constituído pelos vértices $\boldsymbol{A}(4,1)$, $\boldsymbol{B}(7,4)$, $\boldsymbol{C}(7,7)$ e $\boldsymbol{D}(1,4)$

3.1. Apresente o estado da tabela de arestas (ET - *Edge Table*) e tabela de arestas activas (AET - *Active Edge Table*) no início do algoritmo. (1.0)

3.2. Calcule as coordenadas dos pixels de preenchimento, apresentando cada iteração do algoritmo separadamente, indicando o estado da ET e AET, e apresente no final, de forma gráfica, o preenchimento. (3.0)

R:

3.1.)

Antes de iniciar as iterações do algoritmo é necessário obter para cada aresta do polígono que vai ser preenchido um conjunto de elementos informativos sobre os mesmos, tais como: declive m e $\frac{1}{m}$, $x_{mínimo}$, $y_{mínimo}$ e $y_{máximo}$ por forma a podermos construir as tabelas ET e AET.

Nota: $x_{mínimo}$ é a coordenada que acompanha $y_{mínimo}$.

Para cada aresta do polígono, vista como segmento de recta, obtemos os valores de $m = \frac{dy}{dx}$ tendo em conta que cada aresta $\overline{A_1A_2}$ é obtida através dos pontos sucessivos do polígono ligando o último ponto ao primeiro.

Temos assim:

$$\overline{AB} = [(4,1),(7,4)]: m = \frac{4-1}{7-4} = \frac{3}{3}; \frac{1}{m} = 1; y_{min} = 1; x_{min} = 4;\ y_{máx} = 4;$$

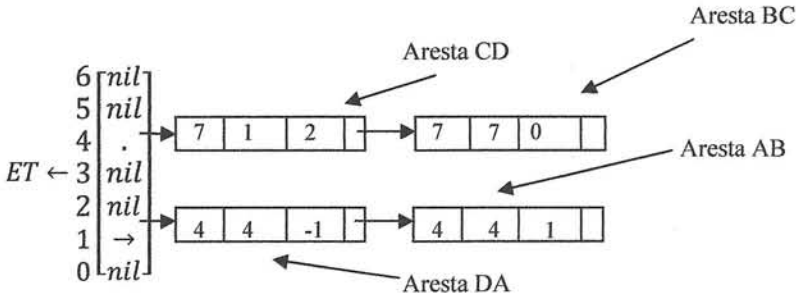
$$\overline{BC} = [(7,4),(7,7)]: m = \frac{7-4}{7-7} = \frac{3}{0} = \infty; \frac{1}{m} = 0; y_{min} = 4; x_{min} = 7;\ y_{máx} = 7;\ \text{(seg. vertical)}$$

$$\overline{CD} = [(7,7),(1,4)]: m = \frac{4-7}{1-7} = \frac{-3}{-6} = \frac{1}{2}; \frac{1}{m} = 2; y_{min} = 4; x_{min} = 1;\ y_{máx} = 7;$$

$$\overline{DA} = [(1,4),(4,1)]: m = \frac{1-4}{4-1} = \frac{-3}{3} = -1; \frac{1}{m} = -1; y_{min} = 1; x_{min} = 4;\ y_{máx} = 4;$$

No arranque do algoritmo as tabelas ET e AET tem o seguinte estado:

$AET \leftarrow nil$



3.2)

Seja a linha de varrimento (*scan-line*) definida por *SL* sendo que o varrimento se faz na vertical, percorrendo o eixo Y desde os valores mais pequenos até à maior coordenada em Y do polígono.

$SL \leftarrow 1$ ou seja, a linha de varrimento recebe o valor do primeiro índice da entrada da tabela ET que não esteja vazia, neste caso a entrada 1, que corresponde à coordenada y menor do polígono partilhada pelas arestas $\overline{AB}$ e $\overline{DA}$.

Iteração do algoritmo

Até que *AET* fique vazia repete

<u>Iteração 1</u>

1.1) Mover de *ET* para *AET* as arestas que passam a ser intersectadas por *SL*, ou seja, o valor de entrada com índice *SL*.
A tabela *AET* recebe então os valores da(s) aresta(s) de *ET* com entrada *SL*.

$$AET \leftarrow \overbrace{|4|4|-1|}^{\overline{DA}} \rightarrow \overbrace{|4|4|-1|}^{\overline{AB}}$$

1.2) Eliminar de *AET* todas as arestas que deixam de ser intersectadas pela linha de varrimento *SL*. Basta verificar se existe alguma aresta em *AET* cujo $y_{máximo} \leq SL$

Neste caso temos $SL = 1$ e $\overline{EA}\ y_{máximo} = 4$ e $\overline{AB}\ y_{máximo} = 4$ logo $SL <$ que ambos. Não existem arestas em condições de ser eliminadas de *AET*, ficando esta tabela inalterada.

1.3) Identificar pontos a desenhar no ecrã entre pares de arestas de *AET* (estas são as intersectadas por *SL*).
A coordenada em y é o valor de *SL* e temos de calcular os valores da abcissa x mais à esquerda e mais à direita (daqui a importância do valor de $x_{mínimo}$ a abcissa que acompanha a ordenada do ponto de intersecção de SL com cada aresta).

Ou seja, vamos considerar os pontos entre $x_{mínimo}$ de pares de arestas em *AET*.
Neste caso, $\overline{EA}\ x_{mínimo} = 4$ e $\overline{AB}\ x_{mínimo} = 4$ logo
Pontos a activar: (4,1)

1.4) Incrementamos SL ou seja $SL \leftarrow SL + 1; SL = 2$

1.5) Actualizamos AET de $x_{mínimo}$ tendo em conta o incremento *SL* vem

$x_{i+1} \leftarrow x_i + \frac{1}{m}$ para qualquer aresta.

Temos então:

$\overline{DA}$: $x_{i+1} \leftarrow 4 + (-1) = 3$; $\overline{AB}$: $x_{i+1} = 4 + 1 = 5$

Ficando $AET \leftarrow \overbrace{|4|3|-1|}^{\overline{DA}'} \rightarrow \overbrace{|4|5|1|}^{\overline{AB}'}$

<u>Iteração 2</u>

2.1) Existem arestas na entrada *SL (==2)* de *ET*? Não.

2.2) Existem arestas em *AET* tal que $y_{máximos} \leq SL$? Não.

2.3) Pontos a activar no ecrã entre pares de arestas de *AET*.
Abcissas entre $x = 3\ (\overline{DA'}\ x_{mínimo})$ e $x = 5\ (\overline{AB'}\ x_{mínimo})$

Pontos a activar: (3,2), (4,2), (5,2)

2.4) Incrementamos SL ou seja $SL \leftarrow SL + 1; SL = 3$
2.5) Actualizamos AET tal que
$\overline{DA'}: x_{i+1} \leftarrow\ 3 + (-1) = 2;\ \overline{AB'}: x_{i+1} = 5 + 1 = 6$

Ficando $AET \leftarrow \overbrace{|4|2|-1|}^{\overline{DA''}} \rightarrow \overbrace{|4|6|1|}^{\overline{AB''}}$

Iteração 3

3.1) Existem arestas na entrada *SL (==3)* de *ET*? Não.

3.2) Existem arestas em *AET* tal que $y_{máximos} \leq SL$? Não.

3.3) Pontos a activar no ecrã entre pares de arestas de *AET*.
Abcissas entre $x = 2\ (\overline{DA''}\ x_{mínimo})$ e $x = 6\ (\overline{AB''}\ x_{mínimo})$

Pontos a activar: (2,3), (3,3), (4,3), (5,3), (6,3)

3.4) Incrementamos SL ou seja $SL \leftarrow SL + 1; SL = 4$
3.5) Actualizamos AET tal que
$\overline{DA''}: x_{i+1} \leftarrow\ 2 + (-1) = 1;\ \overline{AB''}: x_{i+1} = 6 + 1 = 7$

Ficando $AET \leftarrow \overbrace{|4|1|-1|}^{\overline{DA'''}} \rightarrow \overbrace{|4|7|1|}^{\overline{AB'''}}$

Iteração 4

4.1) Existem arestas na entrada *SL (==4)* de *ET*? Sim, as aresta $\overline{BC}$ e $\overline{CD}$ (passam a ser intersectadas pela linha de varrimento *SL*)
Fica $AET \leftarrow \overbrace{|4|1|-1|}^{\overline{DA'''}} \rightarrow \overbrace{|7|1|2|}^{\overline{CD}} \rightarrow \overbrace{|4|7|1|}^{\overline{AB'''}} \rightarrow \overbrace{|7|7|0|}^{\overline{BC}}$ (ordenada por x_{min})

4.2) Existem arestas em *AET* tal que $y_{máximos} \leq SL$? Sim, as arestas $\overline{DA'''}$ e $\overline{AB'''}$ pelo que são eliminadas da tabela AET (estas arestas deixam de ser intersectadas)
Fica $AET \leftarrow \overbrace{|7|1|2|}^{\overline{CD}} \rightarrow \overbrace{|7|7|0|}^{\overline{BC}}$

4.3) Pontos a activar no ecrã entre pares de arestas de *AET*.
Abcissas entre $x = 1\ (\overline{CD}\ x_{mínimo})$ e $x = 7\ (\overline{BC}\ x_{mínimo})$

Pontos a activar: (1,4), (2,4), (3,4), (4,4), (5,4), (6,4), (7,4)

4.4) Incrementamos SL ou seja $SL \leftarrow SL + 1; SL = 5$
4.5) Actualizamos AET tal que
$\overline{CD}: x_{i+1} \leftarrow 1 + 2 = 3;\ \overline{BC}: x_{i+1} = 7 + 0 = 7$

Ficando $AET \leftarrow \overbrace{|7|3|2|}^{\overline{CD'}} \rightarrow \overbrace{|7|7|0|}^{\overline{BC'}}$

Iteração 5

5.1) Existem arestas na entrada *SL (==5)* de *ET*? Não
5.2) Existem arestas em *AET* tal que $y_{máximos} \leq SL$? Não

5.3) Pontos a activar no ecrã entre pares de arestas de *AET*.
Abcissas entre $x = 3\ (\overline{CD'}\ x_{mínimo})$ e $x = 7\ (\overline{BC'}\ x_{mínimo})$

Pontos a activar: (3,5), (4,5), (5,5), (6,5), (7,5)

5.4) Incrementamos SL ou seja $SL \leftarrow SL + 1; SL = 6$
5.5) Actualizamos AET tal que
$\overline{CD'}: x_{i+1} \leftarrow 3 + 2 = 5;\ \overline{BC'}: x_{i+1} = 7 + 0 = 7$

Ficando $AET \leftarrow \overbrace{|7|5|2|}^{\overline{CD''}} \rightarrow \overbrace{|7|7|0|}^{\overline{BC''}}$

Iteração 6

6.1) Existem arestas na entrada *SL (==6)* de *ET*? Não.
6.2) Existem arestas em *AET* tal que $y_{máximos} \leq SL$? Não.

6.3) Pontos a activar no ecrã entre pares de arestas de *AET*.
Abcissas entre $x = 5\ (\overline{CD''}\ x_{mínimo})$ e $x = 7\ (\overline{BC''}\ x_{mínimo})$

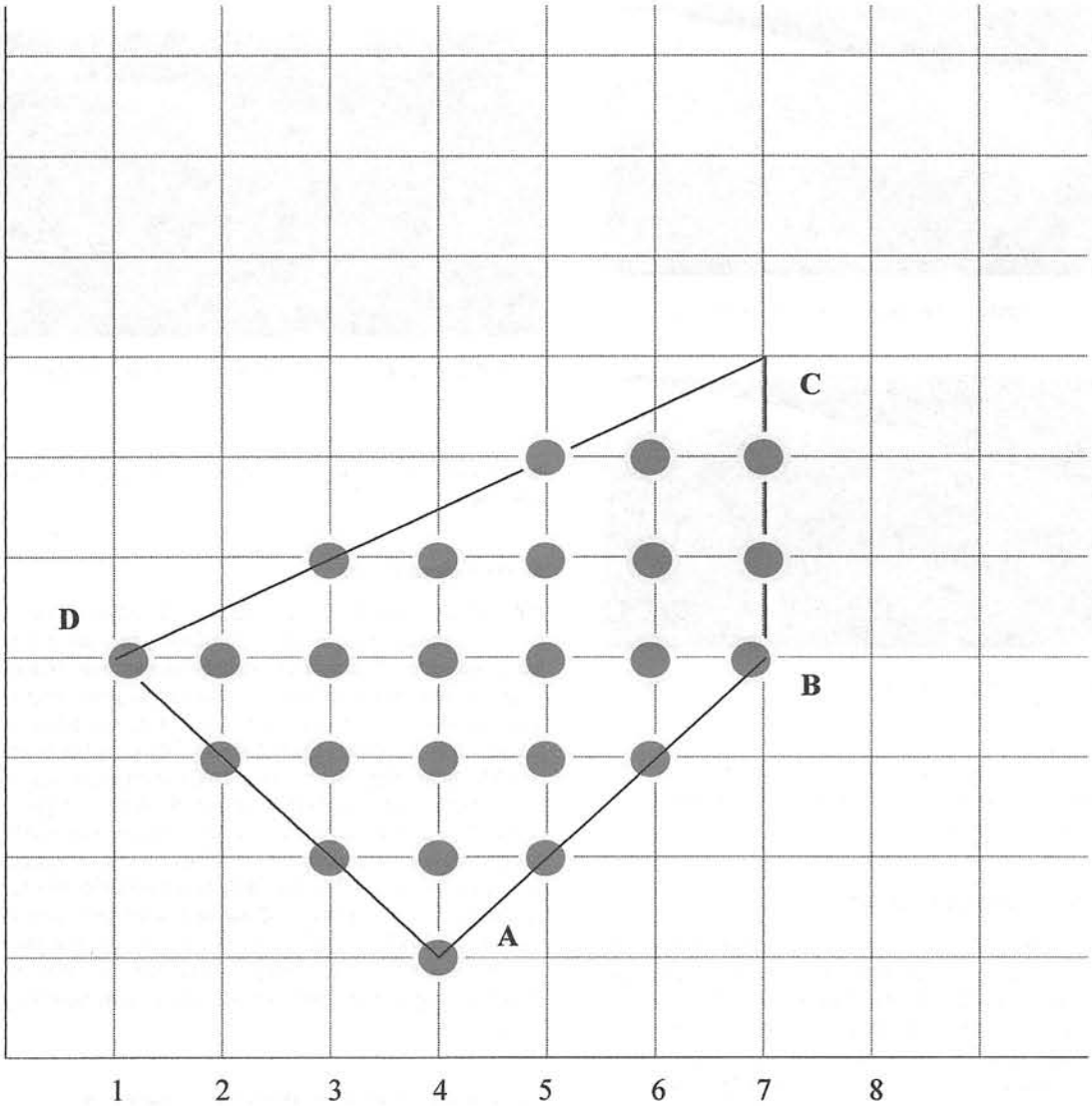
Pontos a activar: (5,6), (6,6), (7,6)

6.4) Incrementamos SL ou seja $SL \leftarrow SL + 1; SL = 7$
6.5) Actualizamos AET tal que
$\overline{CD''}: x_{i+1} \leftarrow 5 + 2 = 7;\ \overline{BC''}: x_{i+1} = 7 + 0 = 7$

Ficando $AET \leftarrow \overbrace{|7|7|2|}^{\overline{CD'''}} \rightarrow \overbrace{|7|7|0|}^{\overline{BC'''}}$

Iteração 7

7.1) Existem arestas na entrada *SL (==7)* de *ET*? Não.
7.2) Existem arestas em *AET* tal que $y_{máximos} \leq SL$? Sim, as aresta $\overline{BC''}$ e $\overline{CD'''}$ (deixam de ser intersectadas pela linha de varrimento *SL*) sendo eliminadas de *AET*.
Ficando $AET \leftarrow nil$ = condição para terminar o algoritmo.


C
D
B
A
1
2
3
4
5
6
7
8

4. Algorithmo Surtherland-Hodgman

Ao polígono da questão 3 aplique o algoritmo Surtherland-Hodgman com a janela de recorte $[(x_{min}, y_{min}), (x_{max}, y_{max})] = [(2,2),(6,8)]$

Mostre como os vértices são processados ao longo das várias arestas de recorte (*clippers*) da fronteira da janela e apresente o conjunto final de vértices de saída (*output*). Apresente todos os cálculos, por aresta de recorte, separadamente, e de forma gráfica e o resultado final do recorte.

R:

O recorte deve ser realizado aresta a aresta de recorte, seguindo o sentido do ponteiro dos relógios (ou inverso) e mantendo a sequência. É indiferente qual a aresta de recorte inicial escolhida.

Em cada passo do recorte um novo conjunto de vértices é gerado que serve de entrada para o passo seguinte. O conjunto de vértices resultante do último passo configura o polígono final recortado.

Conjunto de Vértice $V = \{A, B, C, D\}$ e janela de recorte = [(2,2), (6,8)]

Vamos recortar por $x_{min} = 2$

Para cada aresta do polígono que se vai recortar verificar em qual dos casos se deixa definir (ver notas do docente, página 15 de "Tema-III-Recorte2D.pdf"). Note-se que a verificação se deve fazer por cálculo vectorial, embora por simplificação se omita essa parte da resolução.

Arestas	Ponto 1: fora/dentro da aresta/área de recorte ?	Ponto 2: fora/dentro da aresta/área de recorte ?	Caso do algoritmo	Resultado
$\overline{AB}$	A: dentro	B: dentro	Caso 1	B
$\overline{BC}$	B: dentro	C: dentro	Caso 1	C
$\overline{CD}$	C: dentro	D: fora	Caso 2	D_1
$\overline{DA}$	D: fora	A: dentro	Caso 4	D_2, A

Conjunto de Vértice resultante $V_1 = \{B, C, D_1, D_2, A\}$

Vamos recortar por $y_{min} = 2$

Arestas	Ponto 1: fora/dentro da aresta/área de recorte ?	Ponto 2: fora/dentro da aresta/área de recorte ?	Caso do algoritmo	Resultado
$\overline{BC}$	B: dentro	C: dentro	Caso 1	C
$\overline{CD_1}$	C: dentro	D_1: dentro	Caso 1	D_1

$\overline{D_1D_2}$	D_1: dentro	D_2: dentro	Caso 1	D_2
$\overline{D_2A}$	D_2: dentro	A: fora	Caso 2	A_1
$\overline{AB}$	A: fora	B: dentro	Caso 4	A_2 , B

Conjunto de Vértice resultante $V_2 = \{C, D_1, D_2, A_1, A_2, B\}$

Vamos recortar por $x_{max} = 6$

Arestas	Ponto 1: fora/dentro da aresta/área de recorte ?	Ponto 2: fora/dentro da aresta/área de recorte ?	Caso do algoritmo	Resultado
$\overline{CD_1}$	C: fora	D_1: dentro	Caso 4	C_1, D_1
$\overline{D_1D_2}$	D_1: dentro	D_2: dentro	Caso 1	D_2
$\overline{D_2A_1}$	D_2: dentro	A_1: dentro	Caso 1	A_1
$\overline{A_1A_2}$	A_1: dentro	A_2: dentro	Caso 1	A_2
$\overline{A_2B}$	A_2: dentro	B: fora	Caso 2	B_1
$\overline{BC}$	B: fora	C: fora	Caso 3	nil

Conjunto de Vértice resultante $V_3 = \{C_1, D_1, D_2, A_1, A_2, B_1\}$

Vamos recortar por $y_{max} = 8$

Arestas	Ponto 1: fora/dentro da aresta/área de recorte ?	Ponto 2: fora/dentro da aresta/área de recorte ?	Caso do algoritmo	Resultado
$\overline{C_1D_1}$	C_1: dentro	D_1: dentro	Caso 1	D_1
$\overline{D_1D_2}$	D_1: dentro	D_2: dentro	Caso 1	D_2
$\overline{D_2A_1}$	D_2: dentro	A_1: dentro	Caso 1	A_1
$\overline{A_1A_2}$	A_1: dentro	A_2: dentro	Caso 1	A_2
$\overline{A_2B_1}$	A_2: dentro	B_1: dentro	Caso 1	B_1
$\overline{B_1C_1}$	B_1: dentro	C_1: dentro	Caso 1	C_1

Conjunto de Vértice resultante e final

$V_4 = \{D_1, D_2, A_1, A_2, B_1, C_1\}$

Graficamente: