



Computação Gráfica:

Aula 1.3.3:

Exercícios sobre Clipping

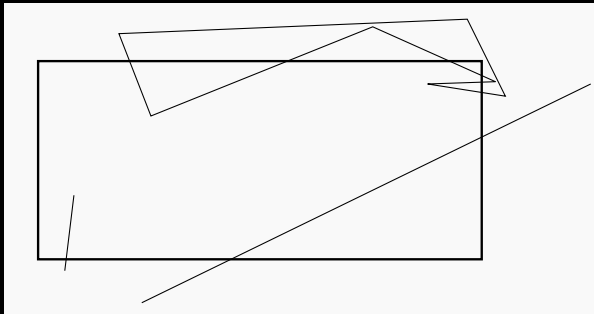
Prof. Dr. rer.nat. Aldo von Wangenheim

1.3.3. Revisão: Clipping de Retas e Polígonos



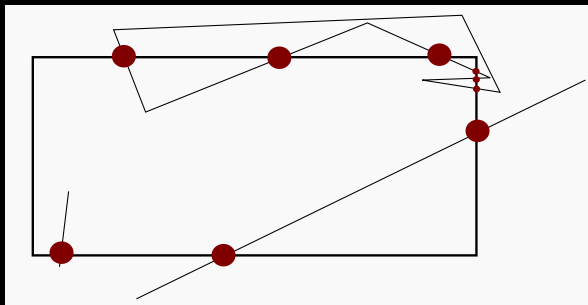


Clipping



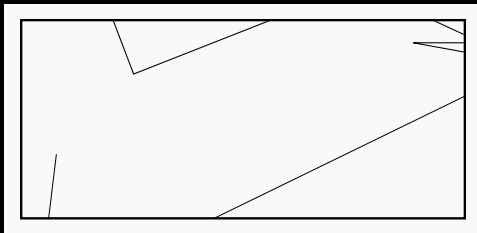


Clipping





Clipping



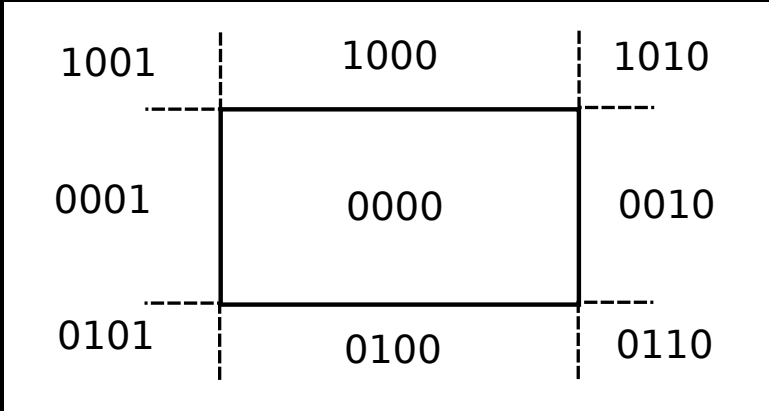


Cohen-Sutherland Clipping





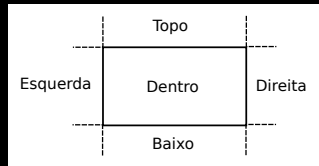
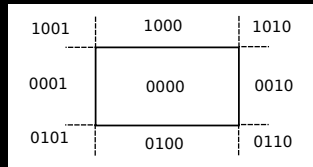
Cohen-Sutherland Clipping





Cohen-Sutherland Clipping

- Esquerda
 - 0001 (1)
- Direita
 - 0010 (2)
- Baixo
 - 0100 (4)
- Cima
 - 1000 (8)





Cohen-Sutherland Clipping

- Esquerda 0001 (1)
- Direita 0010 (2)
- Baixo 0100 (4)
- Cima 1000 (8)

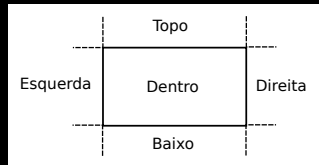
codigoDoPonto(Ponto p, Window w)

```
{
    codigo = 0;
    if (p.y > w.ymax)
        codigo = codigo + CIMA
    else
        if (p.y < w.ymin)
            codigo += BAIXO
```

O mesmo para x...

```
}
```

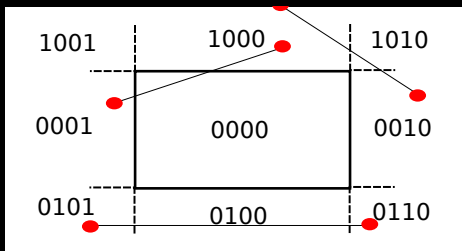
1001	1000	1010
0001	0000	0010
0101	0100	0110





Cohen-Sutherland Clipping

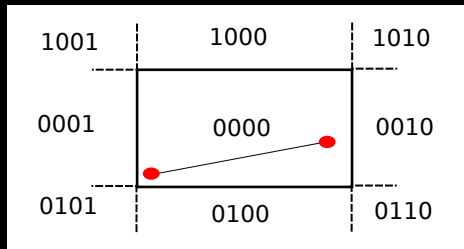
- AND de um par de vértices;
 - se diferente de zero:
descarta.
 - $0101 \text{ AND } 0110 = 0100$
 - (descarta).
 - $0001 \text{ AND } 1000 = 0000$
 - (não sei).
 - $1000 \text{ AND } 0010 = 0000$
 - (não sei).





Cohen-Sutherland Clipping

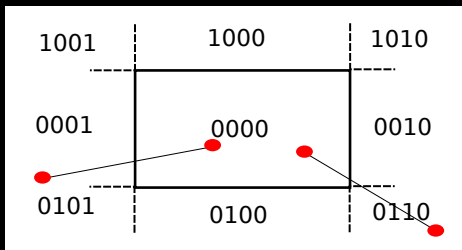
- AND de um par de vértices;
- OR de um par de vértices;
 - Se igual a zero: desenha.
 - $0000 \text{ OR } 0000 = 0000$ (desenha).





Cohen-Sutherland Clipping

- AND de um par de vértices;
- OR de um par de vértices;
- ELSE
 - calcula intersecção (AND indica a borda).
 - 0001 AND
(0001/1000/0010/0100) =
0001 (Esquerda).
 - 0110 AND
(0001/1000/0010/0100) =
0010 (Direita) ou 0100
(Baixo).





Cohen-Sutherland Clipping

- Esquerda:

- $y_e = m.(x_0 - x_1) + y_1$
- $y_e = y_0 + (y_1 - y_0).(x_{min} - x_0)/(x_1 - x_0) : x_e = x_{min}$

- Direita:

- $y_d = m.(x_0 - x_1 + y_1)$
- $y_d = y_0 + (y_1 - y_0) * (x_{max} - x_0)/(x_1 - x_0) : x_0 = x_{max}$

- Cima:

- $x_c = x_1 + \frac{1}{m}.(y_0 - y_1)$
- $x_c = x_0 + (x_1 - x_0) * (y_{max} - y_0)/(y_1 - y_0) : y_0 = y_{max}$

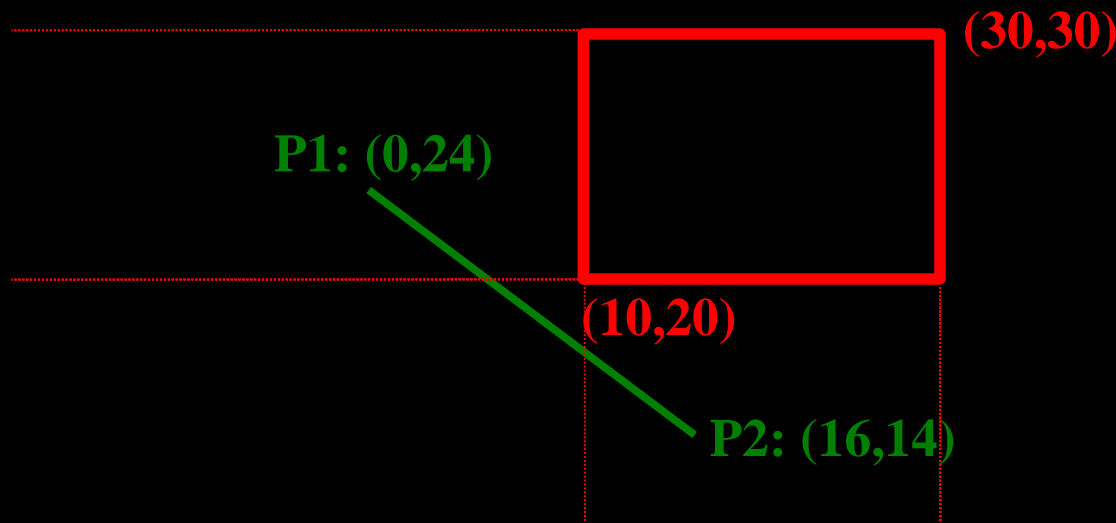
- Baixo:

- $x_b = x_1 + \frac{1}{m}.(y_0 - y_1)$
- $x_0 = x_0 + (x_1 - x_0) * (y_{min} - y_0)/(y_1 - y_0) : y_0 = y_{min}$



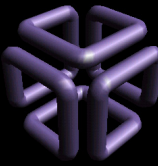
Parte I: Computação Gráfica Básica - Implementação de um Sistema Gráfico Interativo

Aula 4: Clipping # 6



Exemplo C-S #1:

- RC: $[0001, 0100]$ -> Esquerda, Abaixo
- Verificar se ocorrem intersecções!



- Coeficiente Angular:

$$m = \frac{(y_2 - y_1)}{(x_2 - x_1)} = \frac{(14 - 24)}{(16 - 0)} = -0.625$$

- Calculando para a Esquerda:

$$y = m \cdot (x_e - x_1) + y_1$$

$$y = -0.625 \cdot (10 - 0) + 24$$

$$y = 17.75$$

$$17.75 < y_{min} \quad \leftarrow \text{fora da window!}$$



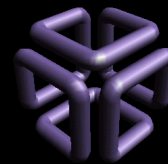
Parte I: Computação Gráfica Básica - Implementação de um Sistema Gráfico Interativo

Aula 4: Clipping # 8



Exemplo C-S #2:

- RC: [0001, 0000] -> Esquerda, Centro
- Ocorre intersecção
 - Aonde?



- Coeficiente Angular:

$$m = \frac{(y_2 - y_1)}{(x_2 - x_1)} = \frac{(26 - 26)}{(16 - 0)} = 0$$

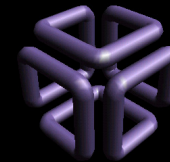
- Calculando para a Esquerda:

$$y = m \cdot (x_e - x_1) + y_1$$

$$y = 0 \cdot (10 - 0) + 26$$

$$y = 26$$

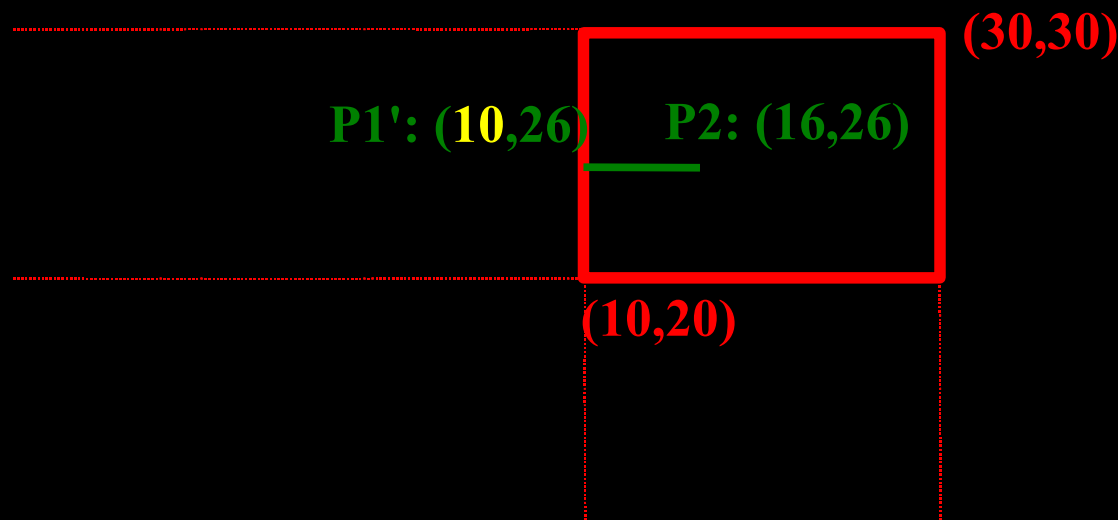
$$y_{min} < 26 < y_{max} \quad \leftarrow \text{OK!}$$



Parte I: Computação Gráfica Básica - Implementação de um Sistema Gráfico Interativo

Aula 4: Clipping # 94

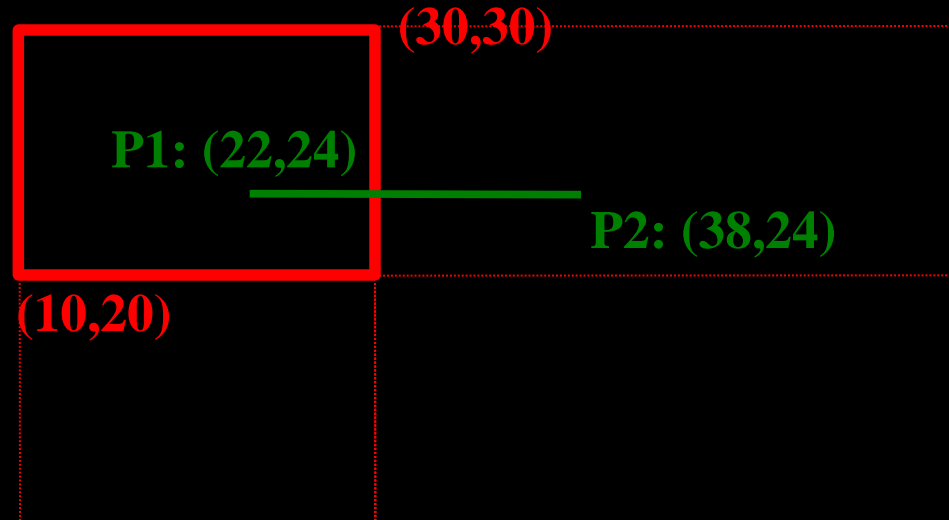
- Nova reta:





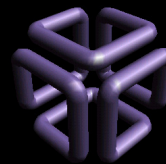
Parte I: Computação Gráfica Básica - Implementação de um Sistema Gráfico Interativo

Aula 4: Clipping # 11



Exemplo C-S #3:

- RC: [0000, 0010] -> Centro, Direita
- Ocorre intersecção
 - Aonde?



- Coeficiente Angular:

$$m = \frac{(y_2 - y_1)}{(x_2 - x_1)} = \frac{(24 - 24)}{(38 - 22)} = 0$$

- Calculando para a Esquerda:

$$y = m \cdot (x_d - x_1) + y_1$$

$$y = 0 \cdot (30 - 8) + 24$$

$$y = 24$$

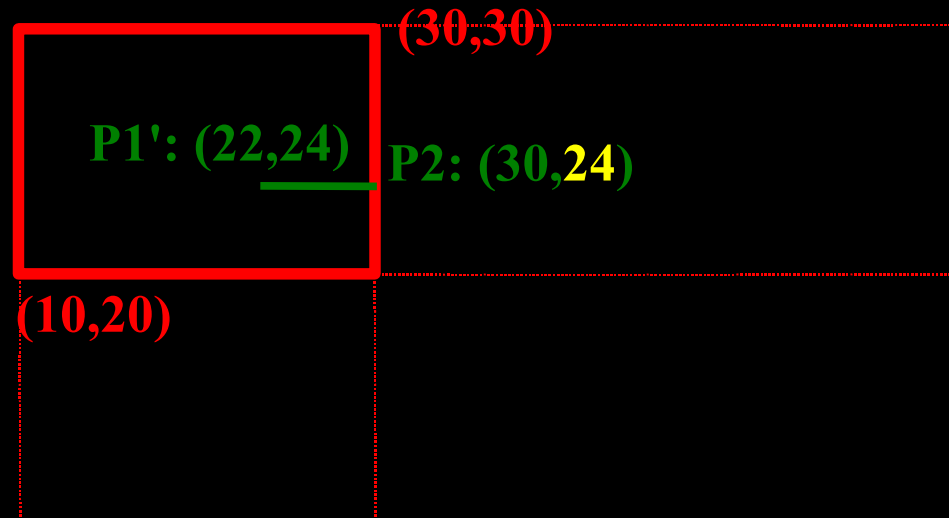
$$y_{min} < 24 < y_{max} \quad \leftarrow \text{OK!}$$



Parte I: Computação Gráfica Básica - Implementação de um Sistema Gráfico Interativo

Aula 4: Clipping # 97

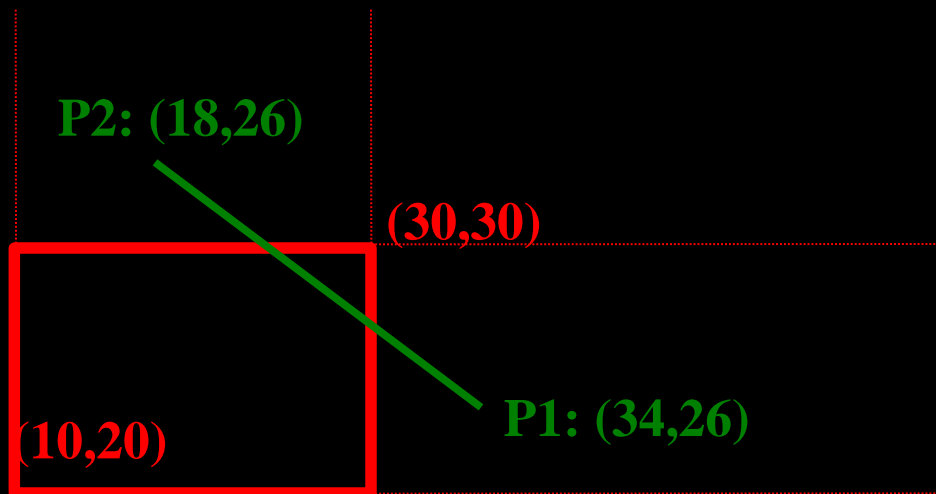
- Nova reta:





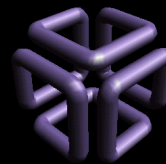
Parte I: Computação Gráfica Básica - Implementação de um Sistema Gráfico Interativo

Aula 4: Clipping # 14



Exemplo C-S #4:

- RC: [0010, 1000] -> Direita, Topo
- Verificar se ocorrem intersecções!



- Coeficiente Angular:

$$m = \frac{(y_2 - y_1)}{(x_2 - x_1)} = \frac{(36 - 26)}{(18 - 34)} = -0.625$$

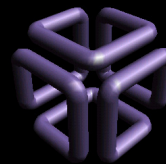
- Calculando para a Direita:

$$y = m \cdot (x_d - x_1) + y_1$$

$$y = -0.625 \cdot (30 - 34) + 26$$

$$y = 28.5$$

$$y_{min} < 28.5 < y_{max} \quad \leftarrow \text{OK!}$$



- Calculando para o Topo:

$$x = x_1 + \frac{1}{m} \cdot (y_t - y_1)$$

$$x = 34 - \frac{1}{0.625} \cdot (30.26)$$

$$x = 27.6$$

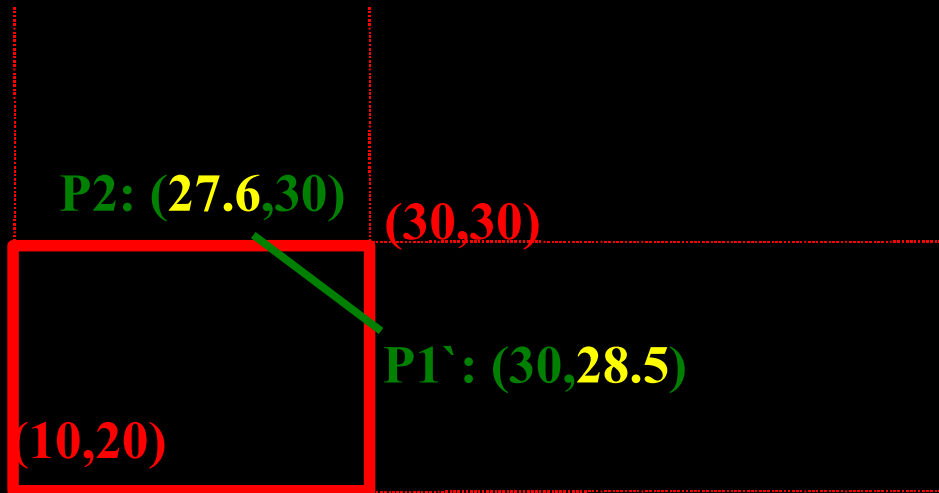
$$x_{min} < 27.6 < x_{max} \quad \leftarrow \text{OK!}$$



Parte I: Computação Gráfica Básica - Implementação de um Sistema Gráfico Interativo

Aula 4: Clipping # 101

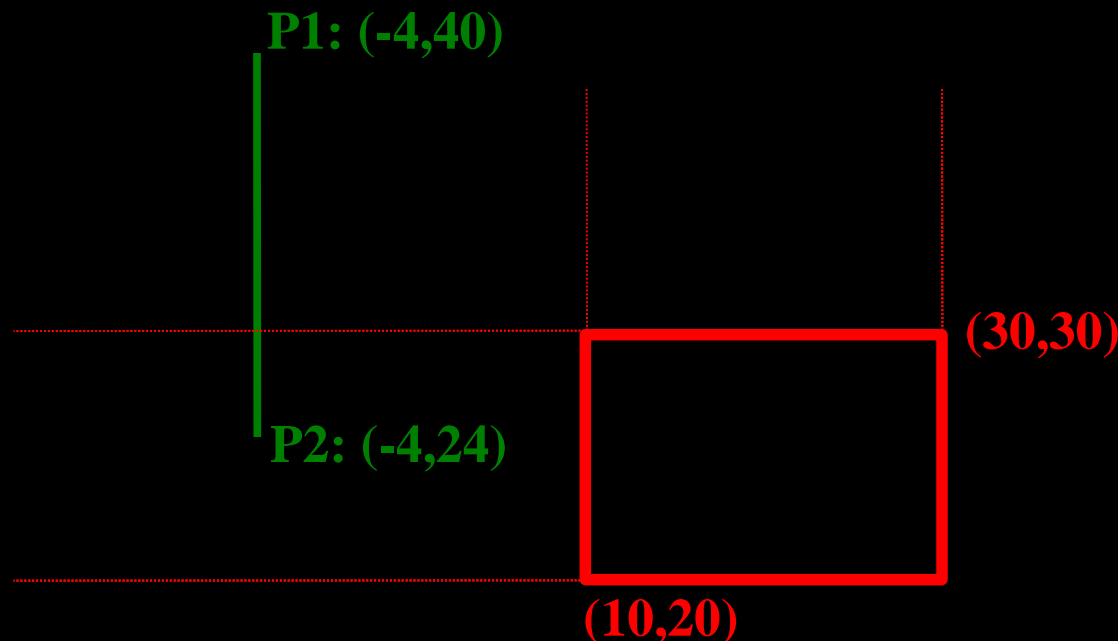
- Nova reta:





Parte I: Computação Gráfica Básica - Implementação de um Sistema Gráfico Interativo

Aula 4: Clipping # 18



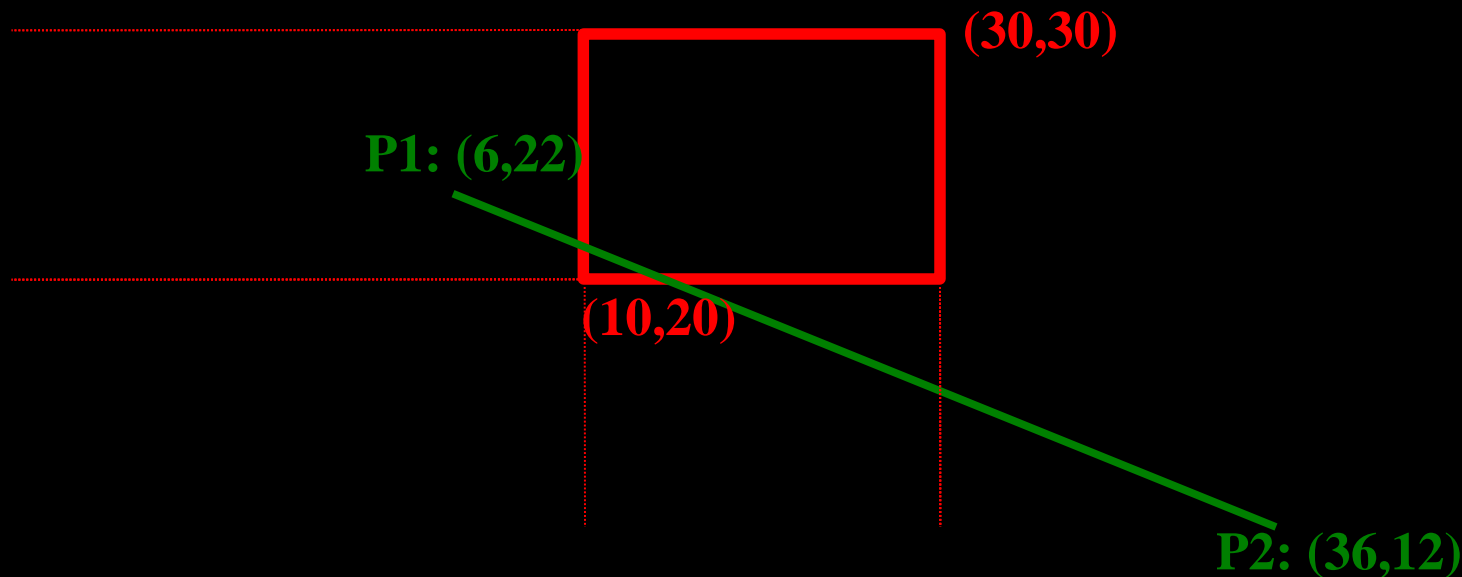
Exemplo C-S #5:

- RC: [1001, 0001] -> "E" Lógico dá 0001
- Totalmente fora.



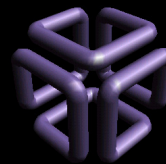
Parte I: Computação Gráfica Básica - Implementação de um Sistema Gráfico Interativo

Aula 4: Clipping # 19



Exemplo C-S #6:

- RC: [0001, 0110] -> Esquerda, Abaixo-Direita
- Verificar se ocorrem intersecções!



- Coeficiente Angular:

$$m = \frac{(y_2 - y_1)}{(x_2 - x_1)} = \frac{(12 - 22)}{(36 - 6)} = -0.3334$$

- Calculando para a Esquerda:

$$y = m \cdot (x_e - x_1) + y_1$$

$$y = -0.3334 \cdot (10 - 6) + 22$$

$$y = 20.6$$

$$y_{min} < 20.6 < y_{max} \quad \leftarrow \text{OK!}$$



- Calculando para a Direita:

$$y = m \cdot (x_d - x_1) + y_1$$

$$y = -0.3334 \cdot (30 - 6) + 22$$

$$y = 14$$

$$14 < y_{min}$$



fora da Window!



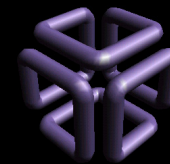
- Finalmente, calculando para o Fundo:

$$x = x_1 + \frac{1}{m} \cdot (y_f - y_1)$$

$$x = 6 - \frac{1}{0.3334} \cdot (20 - 22)$$

$$x = 12$$

$$x_{min} < 12 < x_{max} \quad \leftarrow \text{OK!}$$



Parte I: Computação Gráfica Básica - Implementação de um Sistema Gráfico Interativo

Aula 4: Clipping # 107

- Nova reta:





Liang-Barsky Line Clipping

Zeta1/Zeta2: Define as partes (Ponto inicial/final) dentro da área.

$x = x_1 + n\Delta x$		$y = y_1 + n\Delta y$		
$p_1 = -\Delta x$		$q_1 = x_1 - x_{w_{min}}$		$k = \{1,2,3,4\}$
$p_2 = \Delta x$		$q_2 = x_{w_{max}} - x_1$		
$p_3 = -\Delta y$		$q_3 = y_1 - y_{w_{min}}$		$p_k < 0$: fora -> dentro
$p_4 = \Delta y$		$q_4 = y_{w_{max}} - y_1$		$p_k > 0$: dentro -> fora
$Z_1 = \max(0, r_k)$		$r_k = q_k / p_k$		if ($p_k < 0$)
$Z_2 = \min(1, r_k)$		$r_k = q_k / p_k$		if ($p_k > 0$)
if ($Z_1 = 0$) rejeita				if ($Z_2 = 1$) rejeita

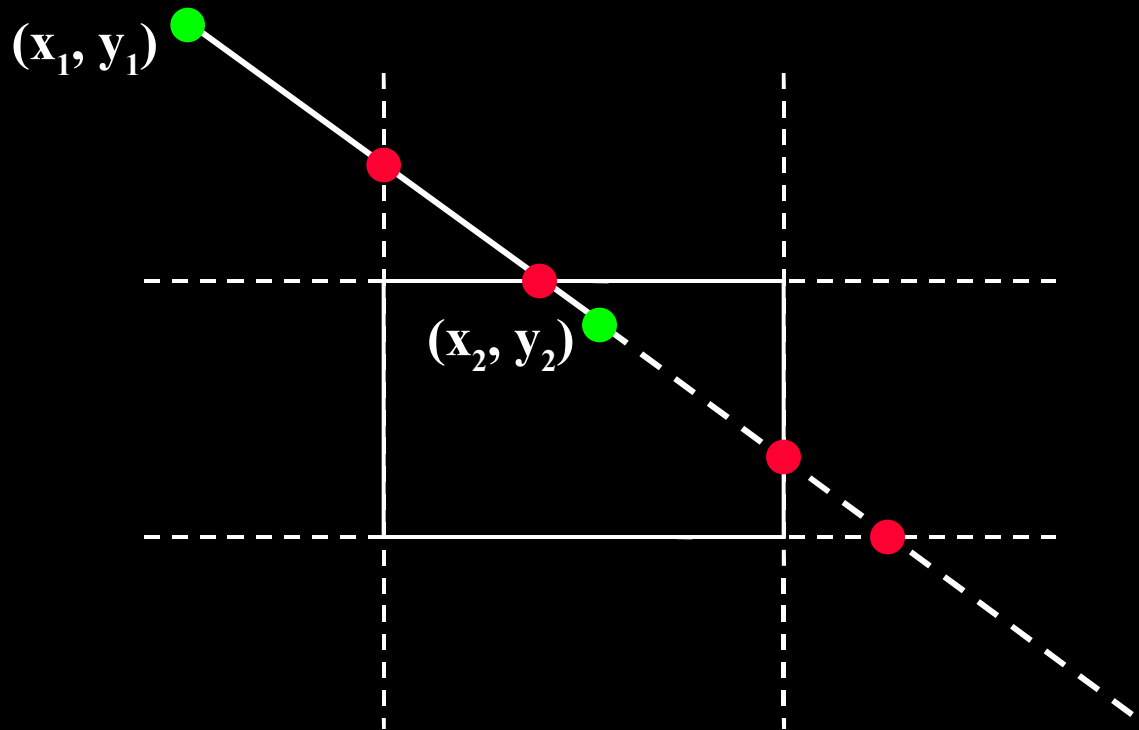


3 exemplos numéricos resolvidos por Liang-Barsky

(vamos repetir exemplos de C-S
resolvendo-os agora por Liang-Barsky)



Parte 2: Liang-Barsky

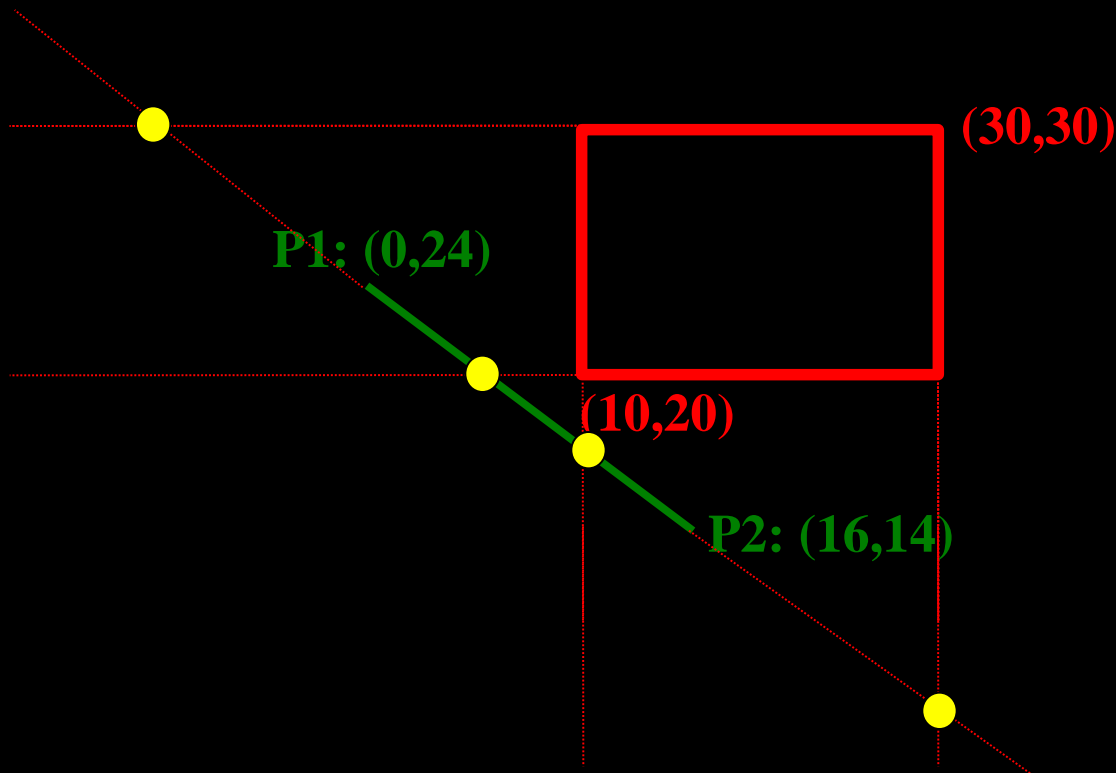


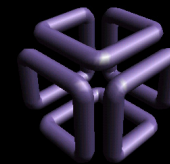


Parte I: Computação Gráfica Básica - Implementação de um Sistema Gráfico Interativo

Aula 4: Clipping # 26

Exemplo L-B #1:





Parte I: Computação Gráfica Básica - Implementação de um Sistema Gráfico Interativo

Aula 4: Clipping # 110

- $p_1, p_2 \dots q_1, q_2 \dots$

$$p_1 = -\Delta x = -16$$

$$p_2 = \Delta x = 16$$

$$p_3 = -\Delta y = -(14 - 24) = 10$$

$$p_4 = \Delta y = -10$$

$$q_1 = x_1 - x_{min} = 0 - 10 = -10$$

$$q_2 = x_{max} - x_1 = 30 - 0 = 30$$

$$q_3 = y_1 - y_{min} = 24 - 20 = 4$$

$$q_4 = y_{max} - y_1 = 30 - 24 = 6$$

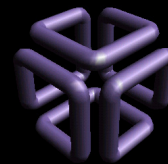


- u_1 – valores menores do que zero – p_1 e p_4

$$r_1 = \frac{q_1}{p_1} = \frac{-10}{-16} = 0.625$$

$$r_4 = \frac{q_4}{p_4} = \frac{6}{-10} = -0.6$$

$$u_1 = \max(0, r_1, r_4) = 0.625$$



- u_2 – valores menores do que zero – p_2 e p_3

$$r_2 = \frac{q_2}{p_2} = \frac{30}{16} = 1,875$$

$$r_4 = \frac{q_4}{p_4} = \frac{4}{10} = 0,4$$

$$u_2 = \min(1, r_2, r_3) = 0,4$$

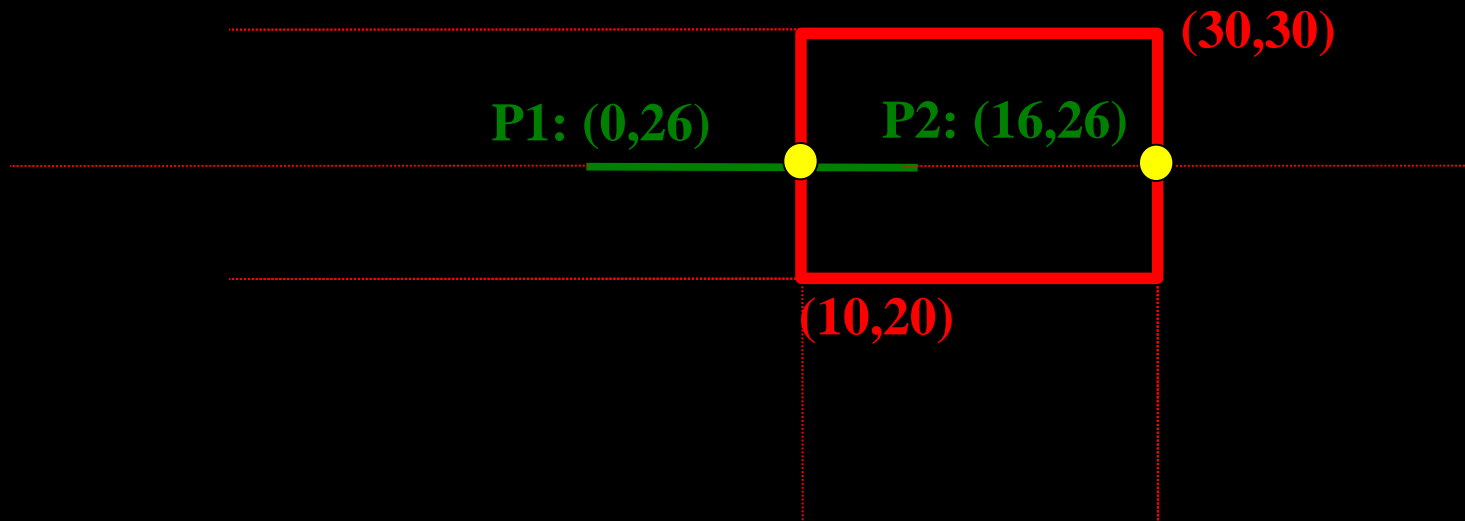
- Como $u_1 > u_2$, a reta está fora!

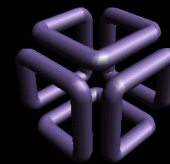


Parte I: Computação Gráfica Básica - Implementação de um Sistema Gráfico Interativo

Aula 4: Clipping # 30

Exemplo L-B #2:





Parte I: Computação Gráfica Básica - Implementação de um Sistema Gráfico Interativo

Aula 4: Clipping # 114

- $p_1, p_2 \dots q_1, q_2 \dots$

$$p_1 = -\Delta x = -16$$

$$p_2 = \Delta x = 16$$

$$p_3 = -\Delta y = 0$$

$$p_4 = \Delta y = 0$$

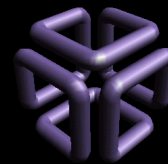
$$q_1 = x_1 - x_{min} = 0 - 10 = -10$$

$$q_2 = x_{max} - x_1 = 30 - 0 = 30$$

$$q_3 = y_1 - y_{min} = 26 - 20 = 6$$

$$q_4 = y_{max} - y_1 = 30 - 26 = 4$$

- p_3 e p_4 valem 0!
 - Mas q_3 e $q_4 \geq 0$
 - Dentro dos limites



Parte I: Computação Gráfica Básica - Implementação de um Sistema Gráfico Interativo

Aula 4: Clipping # 115

- u_1 – valores menores do que zero – p_1

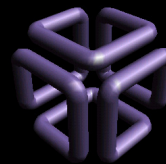
$$r_1 = \frac{q_1}{p_1} = \frac{-10}{-16} = 0,625$$

$$u_1 = \max(0, r_1) = 0,625$$

- u_2 – valores maiores do que zero – p_2

$$r_2 = \frac{q_2}{p_2} = \frac{30}{16} = 1,875$$

$$u_2 = \min(1, r_2) = 1$$

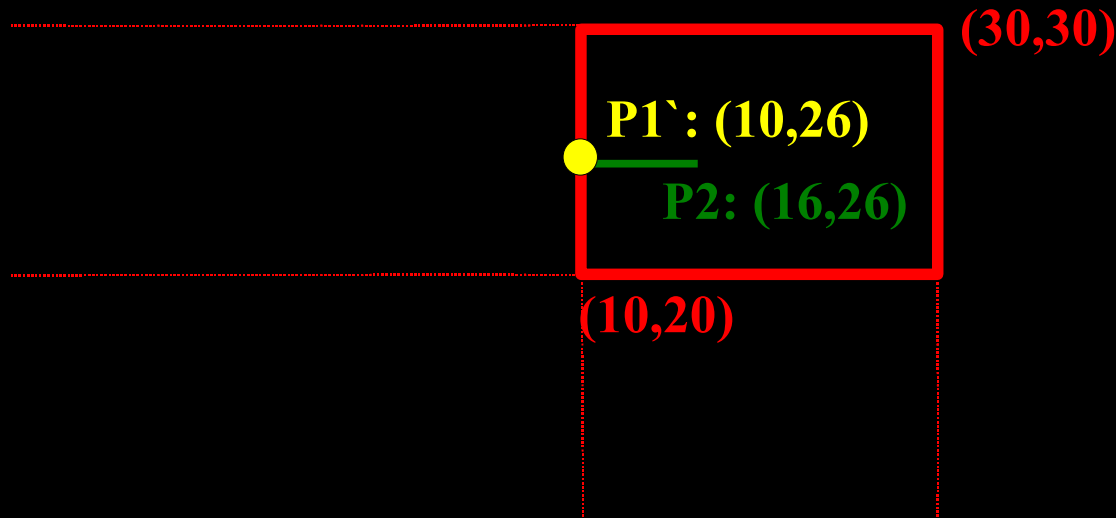


- u_2 é maior do que 1
 - Logo, de dentro para fora o segmento não toca a window.
- Colocando u_1 na equação paramétrica:
$$x = x_1 + u \Delta x$$
$$x = 0 + 0,625 \cdot 16$$
$$x = 10$$
$$y = y_1 + u \Delta y$$
$$y = 26 + 0,625 \cdot 0$$
$$y = 26$$
 - Novo ponto, de fora para dentro: (10, 26)



Parte I: Computação Gráfica Básica - Implementação de um Sistema Gráfico Interativo

Aula 4: Clipping # 117

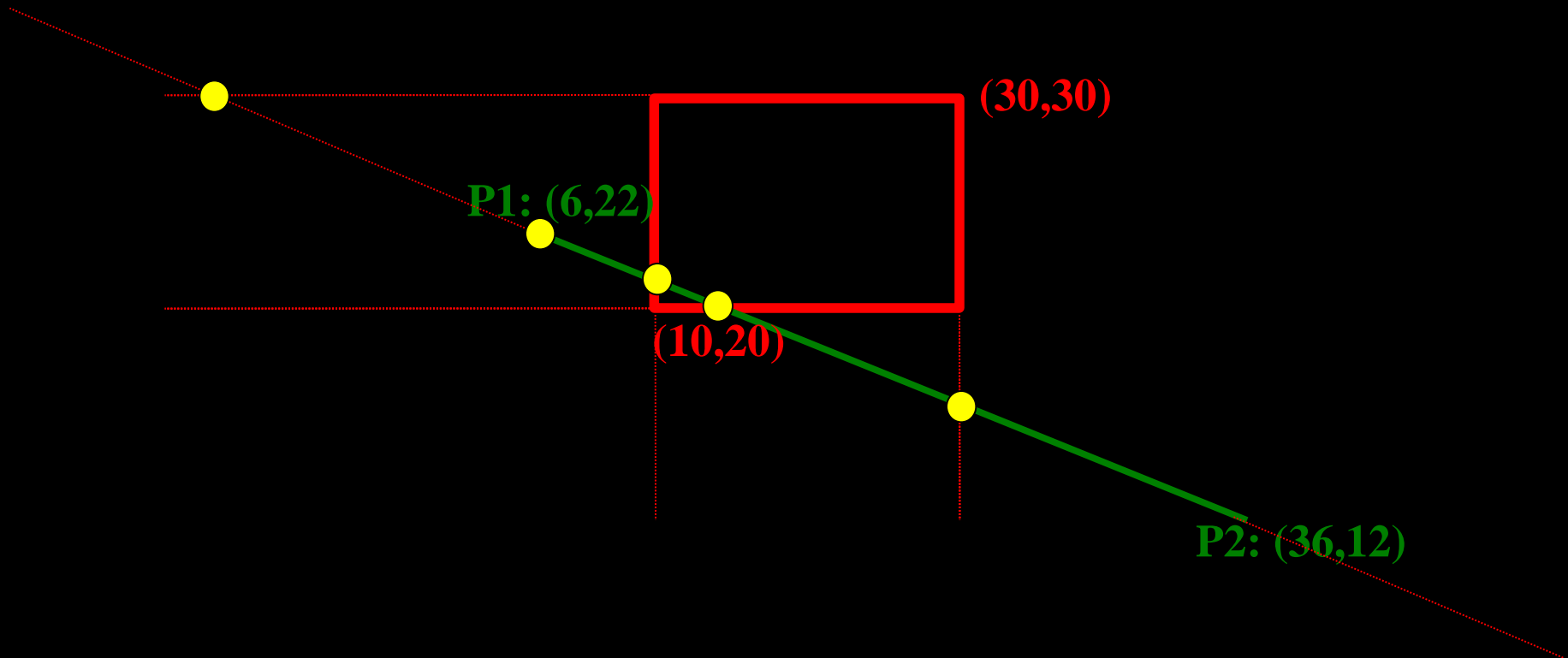


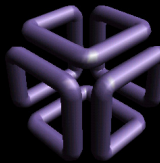


Parte I: Computação Gráfica Básica - Implementação de um Sistema Gráfico Interativo

Aula 4: Clipping # 35

Exemplo L-B #3:





Parte I: Computação Gráfica Básica - Implementação de um Sistema Gráfico Interativo

Aula 4: Clipping # 119

- $p_1, p_2 \dots q_1, q_2 \dots$

$$p_1 = -\Delta x = -(36 - 6) = -30$$

$$p_2 = \Delta x = 30$$

$$p_3 = -\Delta y = -(12 - 22) = 10$$

$$p_4 = \Delta y = -10$$

$$q_1 = x_1 - x_{\min} = 6 - 10 = -4$$

$$q_2 = x_{\max} - x_1 = 30 - 6 = 24$$

$$q_3 = y_1 - y_{\min} = 22 - 20 = 2$$

$$q_4 = y_{\max} - y_1 = 30 - 22 = 8$$



Parte I: Computação Gráfica Básica - Implementação de um Sistema Gráfico Interativo

Aula 4: Clipping # 120

- u_1 – valores menores do que zero – p_1 e p_4

$$r_1 = \frac{q_1}{p_1} = \frac{-4}{-30} = 0,1334$$

$$r_4 = \frac{q_4}{p_4} = \frac{6}{-10} = -0,6$$

$$u_1 = \max(0, r_1, r_4) = 0,1334$$



- u_2 – valores maiores do que zero – p_2 e p_3

$$r_2 = \frac{q_2}{p_2} = \frac{30}{16} = 1,875$$

$$r_3 = \frac{q_3}{p_3} = \frac{2}{10} = 0,2$$

$$u_2 = \min(1, r_2, r_3) = 0,2$$



- Colocando u_1 na equação paramétrica:

$$x = x_1 + u \Delta x$$

$$x = 6 + 0,1334.30$$

$$x = 10$$

$$y = y_1 + u \Delta y$$

$$y = 22 + 0,1334.(-10)$$

$$y = 20.6667$$

- Novo ponto, de fora para dentro: (10, 20.6667)



- Colocando u_2 na equação paramétrica:

$$x = x_1 + u \Delta x$$

$$x = 6 + 0,2 \cdot 30$$

$$x = 12$$

$$y = y_1 + u \Delta y$$

$$y = 22 + 0,2 \cdot (-10)$$

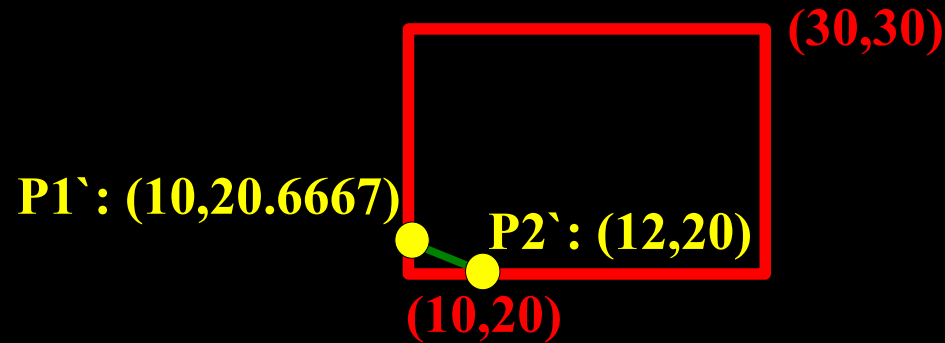
$$y = 20$$

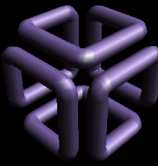
- Novo ponto, de dentro para fora: (12, 20)



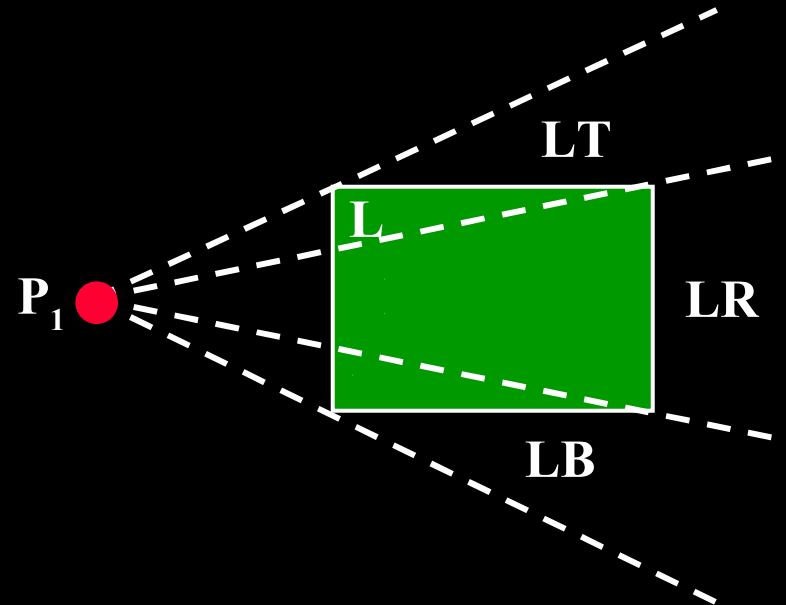
Parte I: Computação Gráfica Básica - Implementação de um Sistema Gráfico Interativo

Aula 4: Clipping # 124





Parte 3: Nicholl-Lee-Nicholl



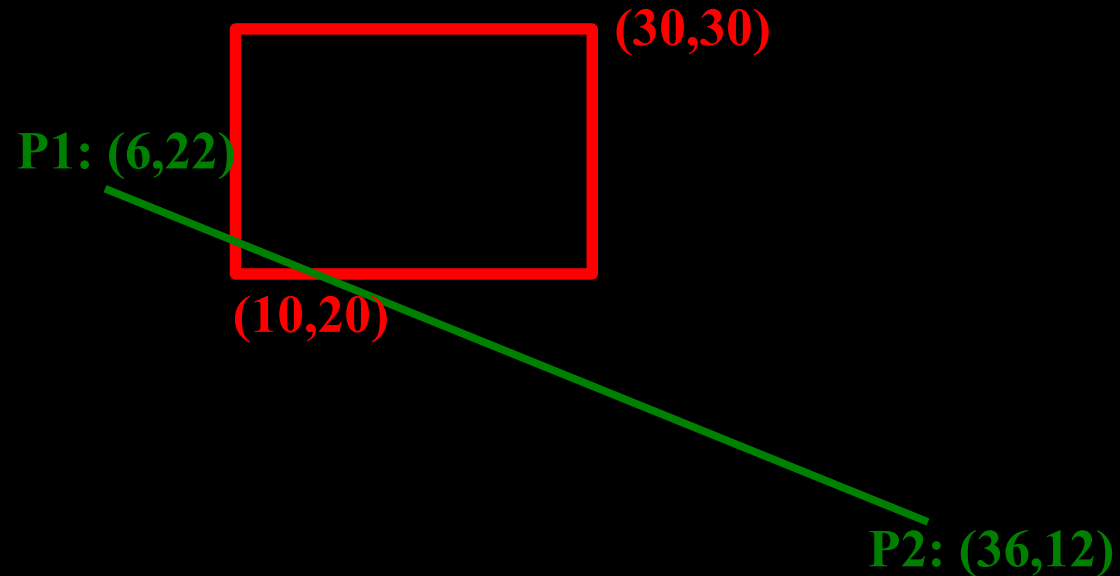


Um exemplo numérico resolvido por Nicholl-Lee-Nicholl



Parte I: Computação Gráfica Básica - Implementação de um Sistema Gráfico Interativo

Aula 4: Clipping # 126

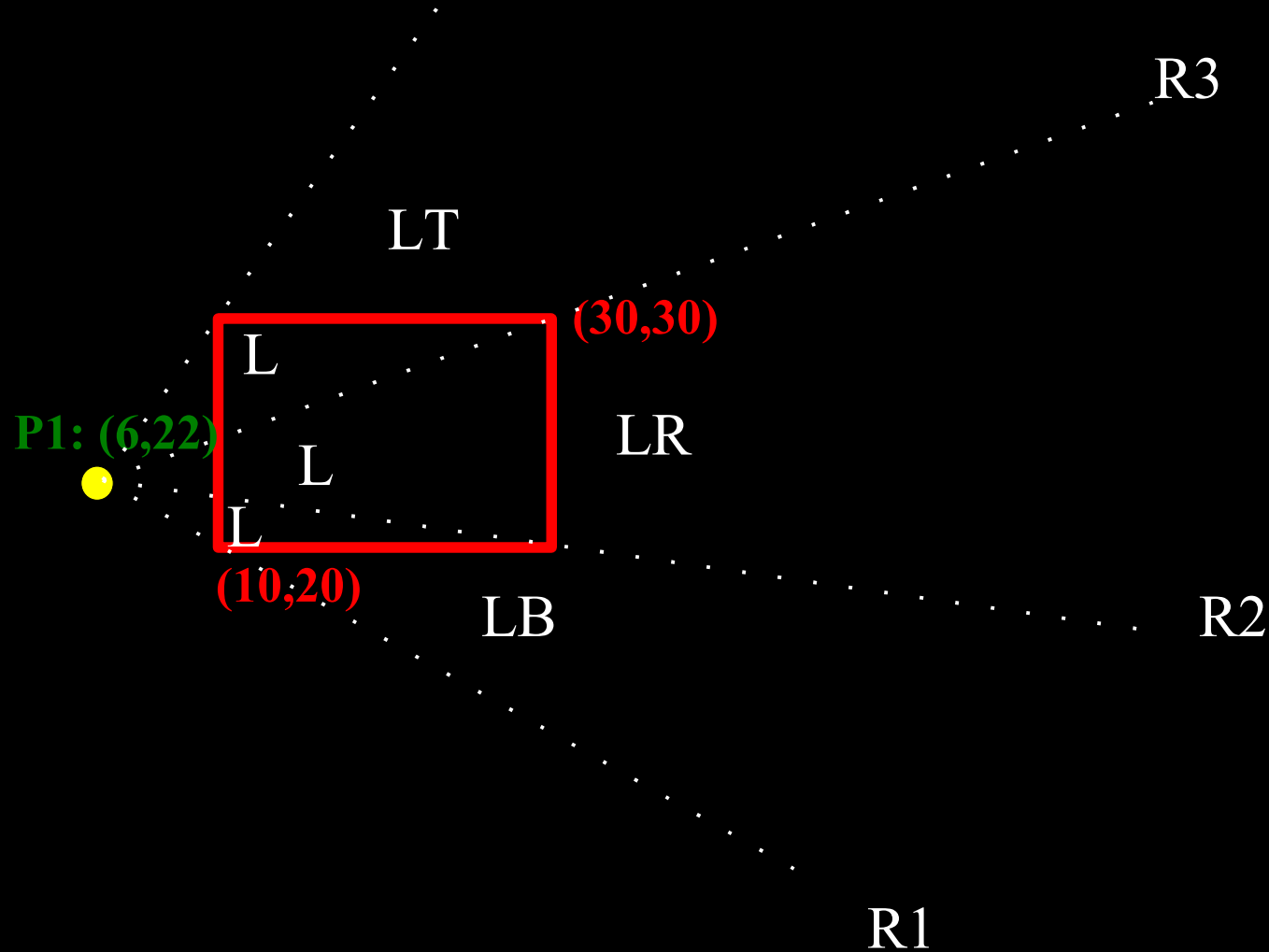


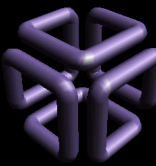


R4

Parte I: Computação Gráfica Básica - Implementação de um Sistema Gráfico Interativo

Aula 4: Clipping # 127





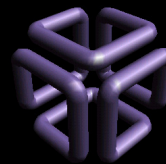
- Coeficiente angular das novas retas:

$$m_1 = \frac{(20 - 22)}{(10 - 6)} = \frac{-2}{4} = -0,5$$

$$m_2 = \frac{(20 - 22)}{(30 - 6)} = \frac{-2}{24} = -0.0833$$

$$m_3 = \frac{(30 - 22)}{(30 - 6)} = \frac{8}{24} = 0.3334$$

$$m_4 = \frac{(30 - 22)}{(10 - 6)} = \frac{8}{4} = 2$$



- Coeficiente angular da reta de teste:

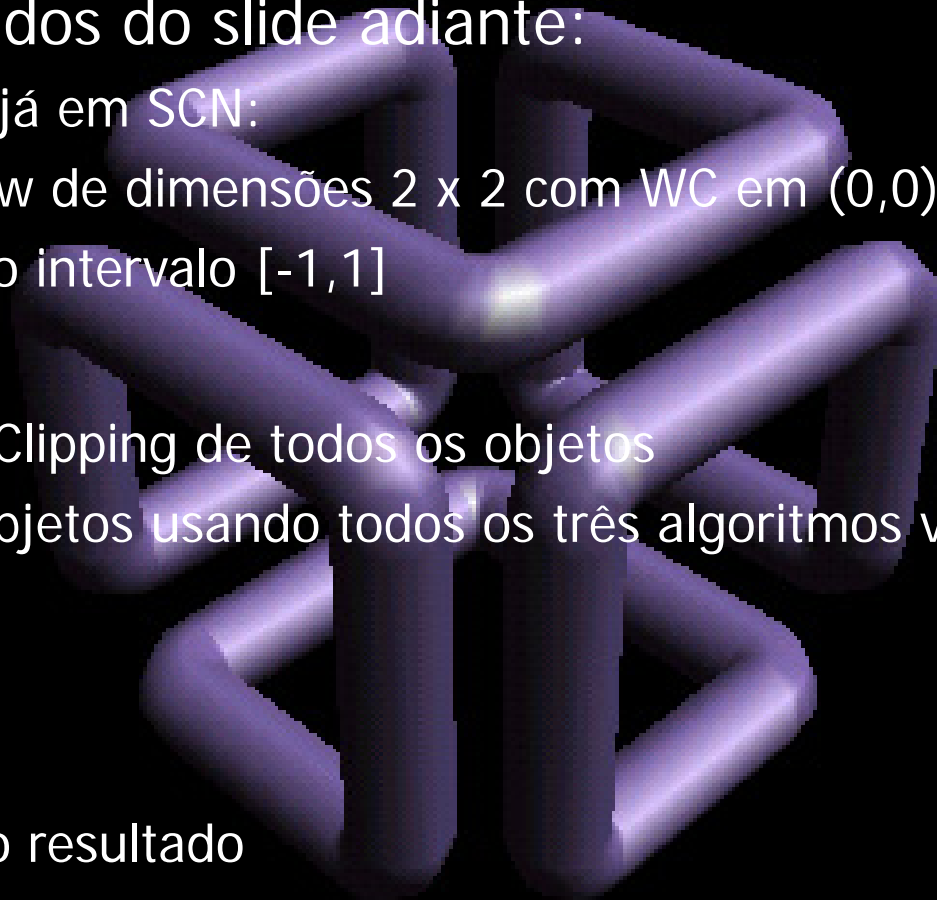
$$m_r = \frac{(12 - 22)}{(36 - 6)} = \frac{-10}{30} = -0.3334$$

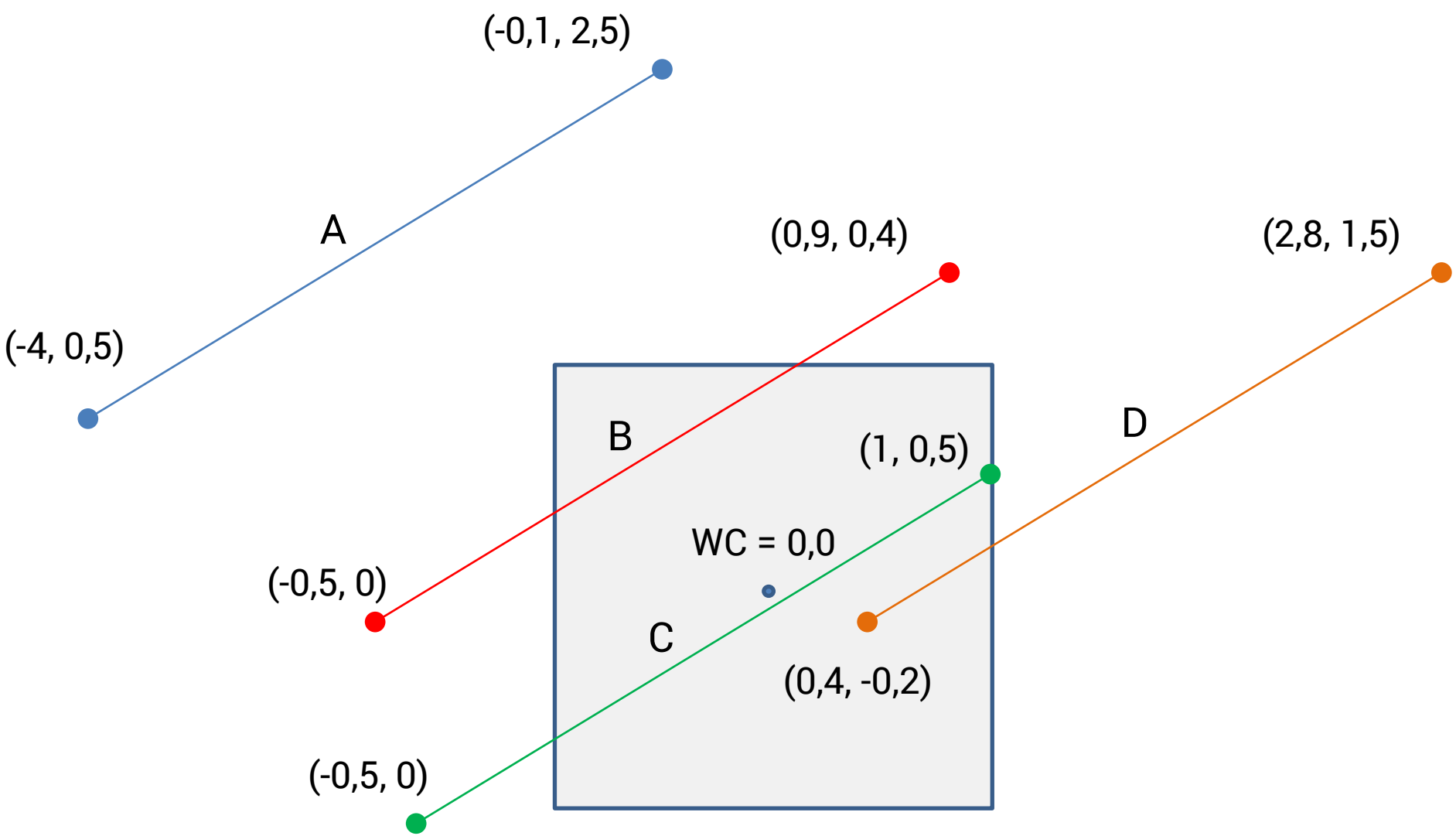
- Se encontra entre m_1 e m_2
 - Logo, basta testar a intersecção com os limites esquerdo e inferior da Window.
 - Se m_r fosse $<$ que m_1 ou $>$ m_4 , a reta estaria descartada automaticamente.



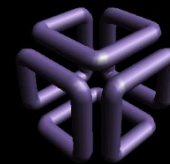
Exercício em Papel:

- Tome os dados do slide adiante:
 - Considere já em SCN:
 - Window de dimensões 2 x 2 com WC em (0,0)
 - SCN no intervalo $[-1,1]$
- Faça:
 - Calcule o Clipping de todos os objetos
 - Clipe os objetos usando todos os três algoritmos vistos:
 - C-S
 - L-B
 - N-L-N
 - Desenhe o resultado









Método #1: Sutherland-Hodgeman

- Clipe contra cada borda da Window
 - Percorra a lista de vértices adicionando ou removendo pontos conforme o caso.

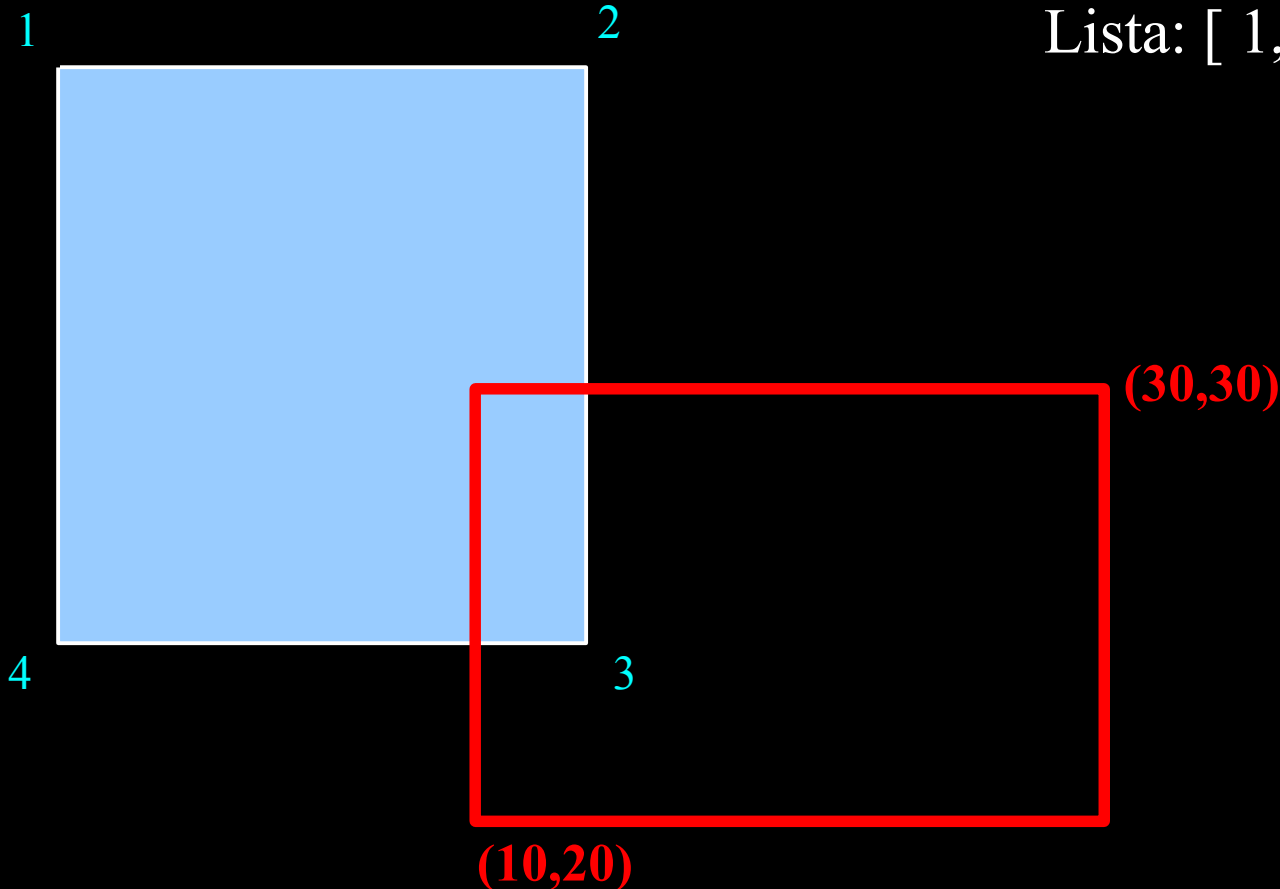


- Por Sutherland-Hodgeman:
- Clipe contra cada borda da Window
 - Percorra a lista de vértices adicionando ou removendo pontos conforme o caso.



Parte I: Computação Gráfica Básica - Implementação de um Sistema Gráfico Interativo

Aula 4: Clipping # 132

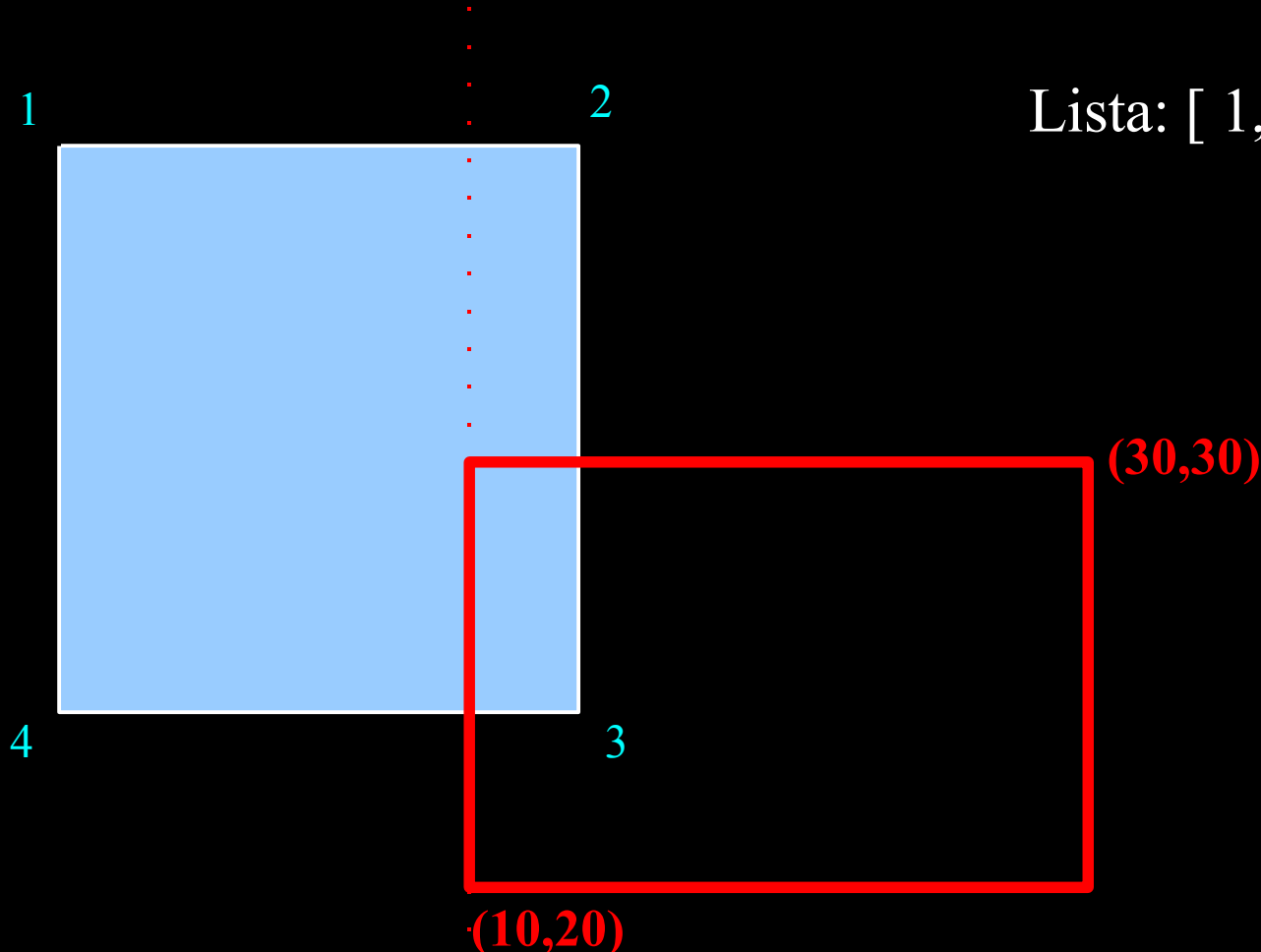


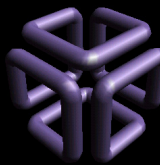


Parte I: Computação Gráfica Básica - Implementação de um Sistema Gráfico Interativo

Aula 4: Clipping # 133

- Borda Esquerda:

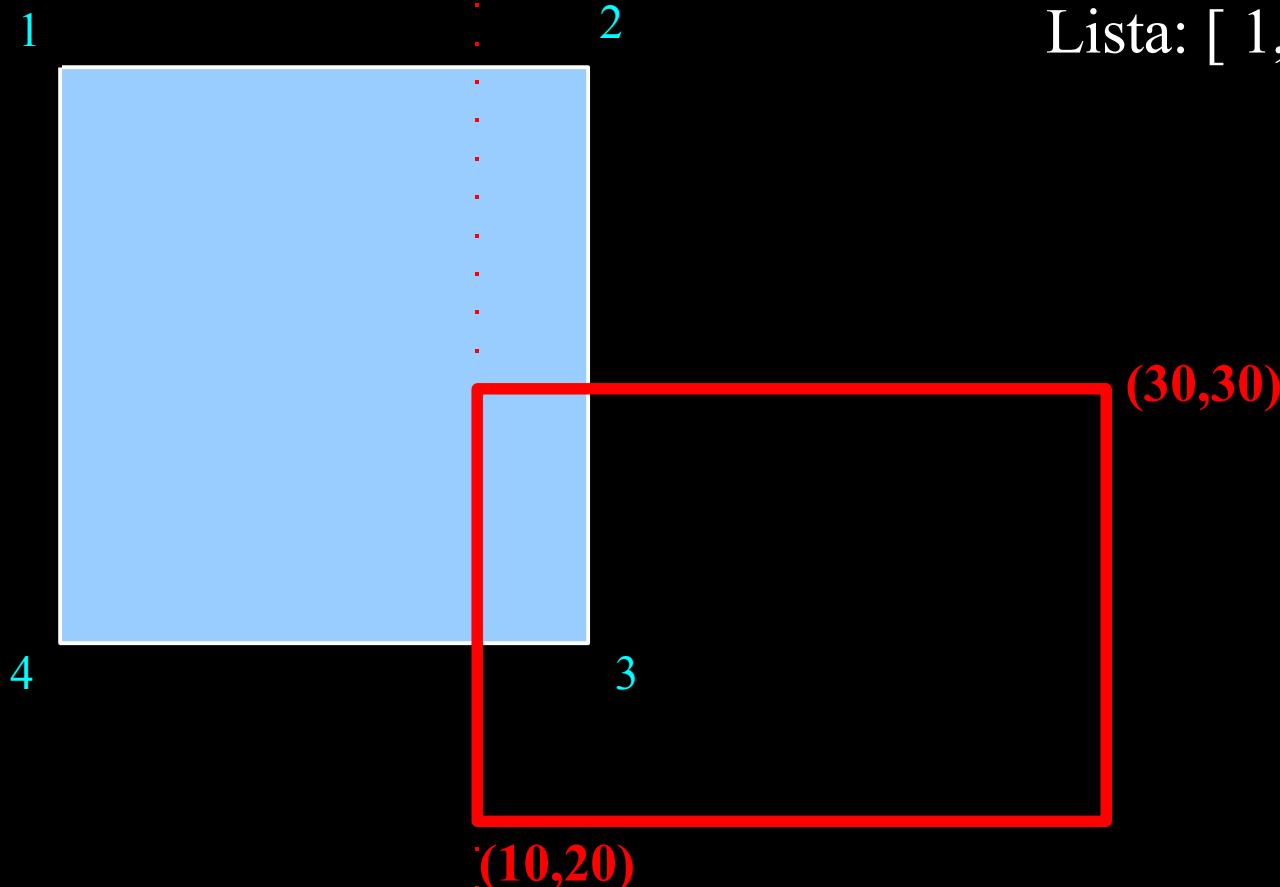


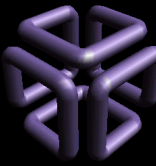


Parte I: Computação Gráfica Básica - Implementação de um Sistema Gráfico Interativo

Aula 4: Clipping # 134

- Vértices 1 e 2: Fora, Dentro
- Remove 1, Mantém 2, Adiciona 1' entre 1 e 2

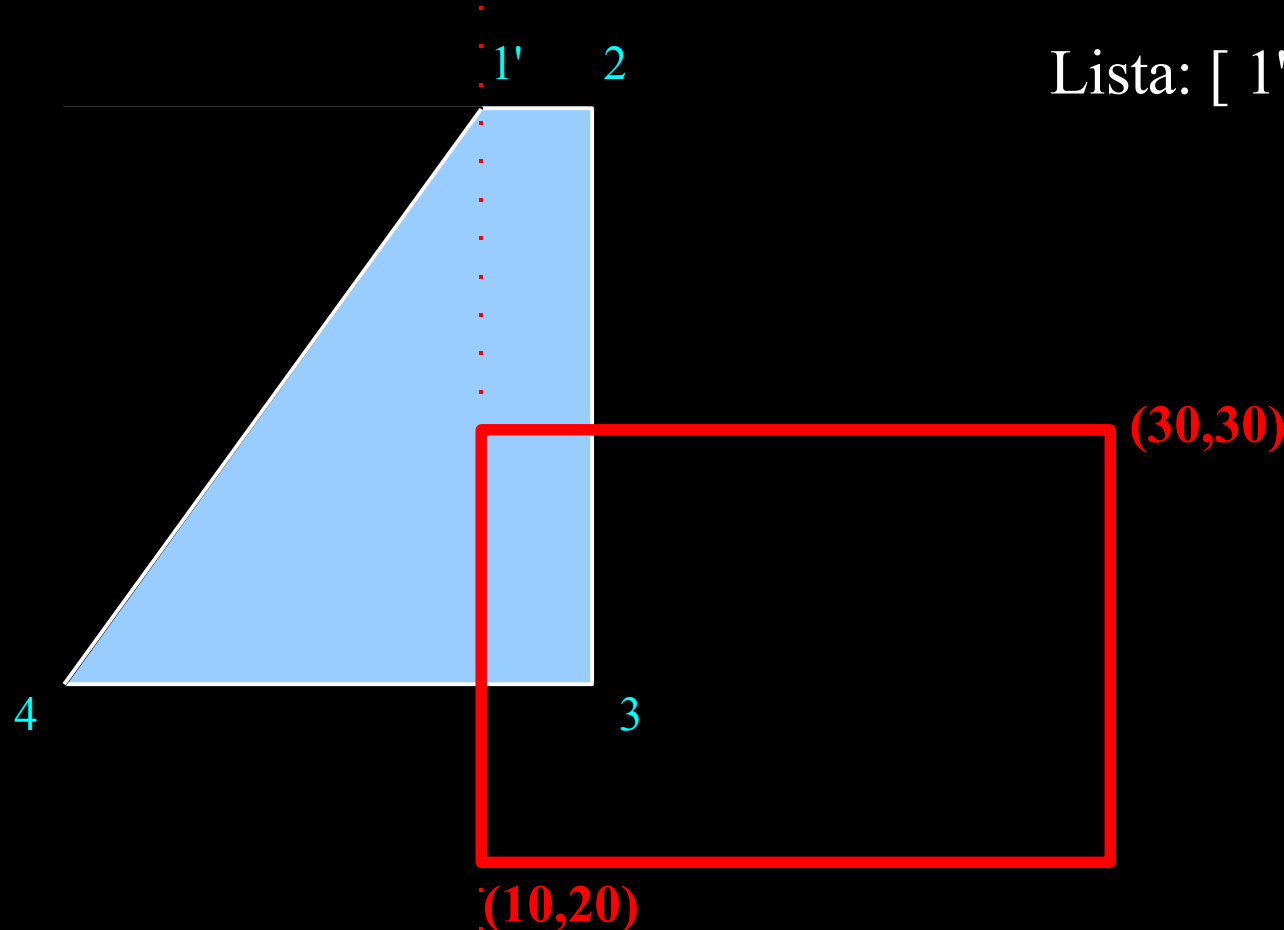




Parte I: Computação Gráfica Básica - Implementação de um Sistema Gráfico Interativo

Aula 4: Clipping # 135

- Vértices 1 e 2: Fora, Dentro
- Remove 1, Mantém 2, Adiciona 1' entre 1 e 2

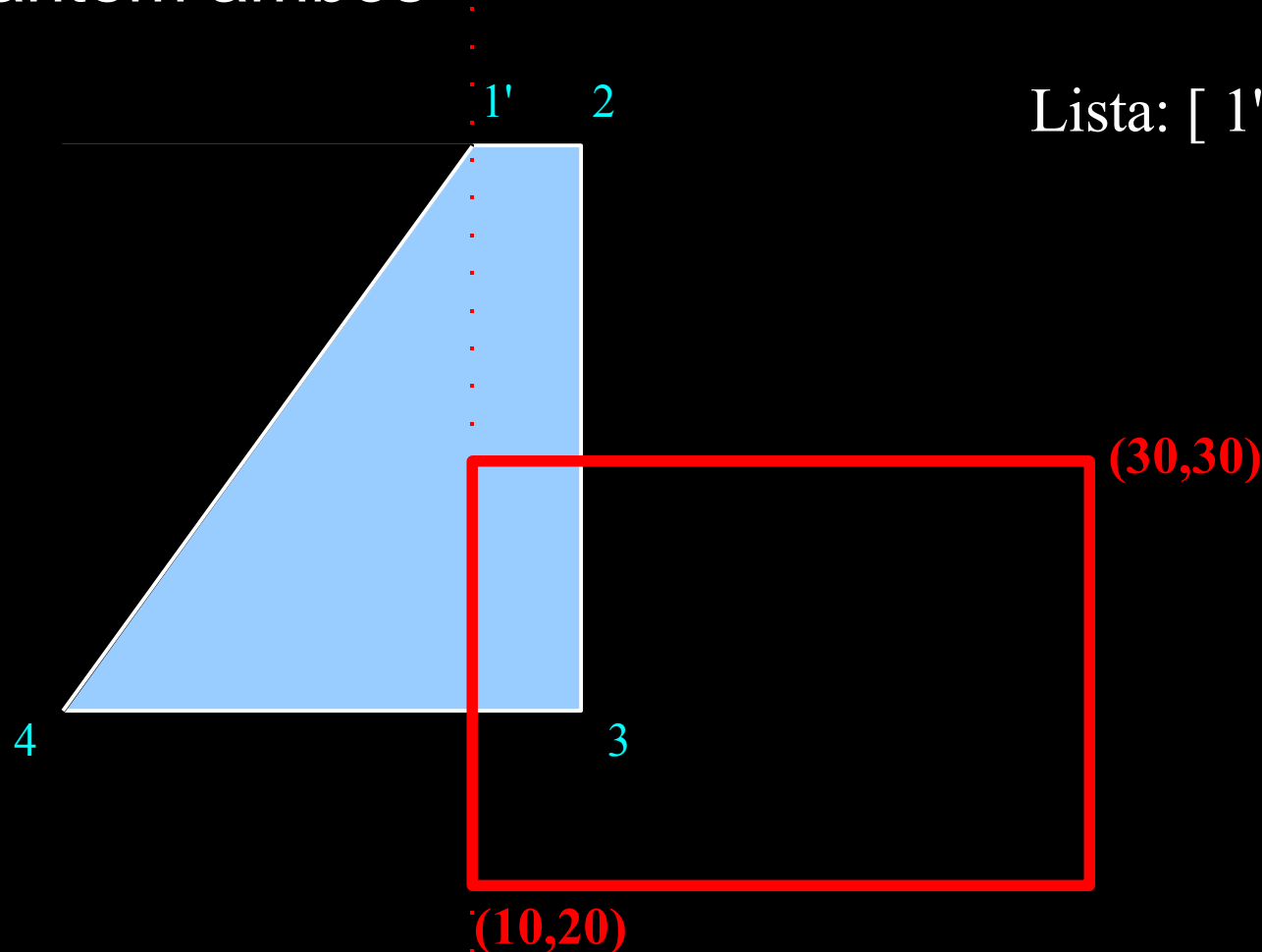


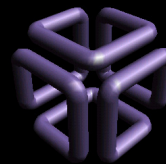


Parte I: Computação Gráfica Básica - Implementação de um Sistema Gráfico Interativo

Aula 4: Clipping # 136

- Vértices 2 e 3: Dentro, Dentro
- Mantém ambos

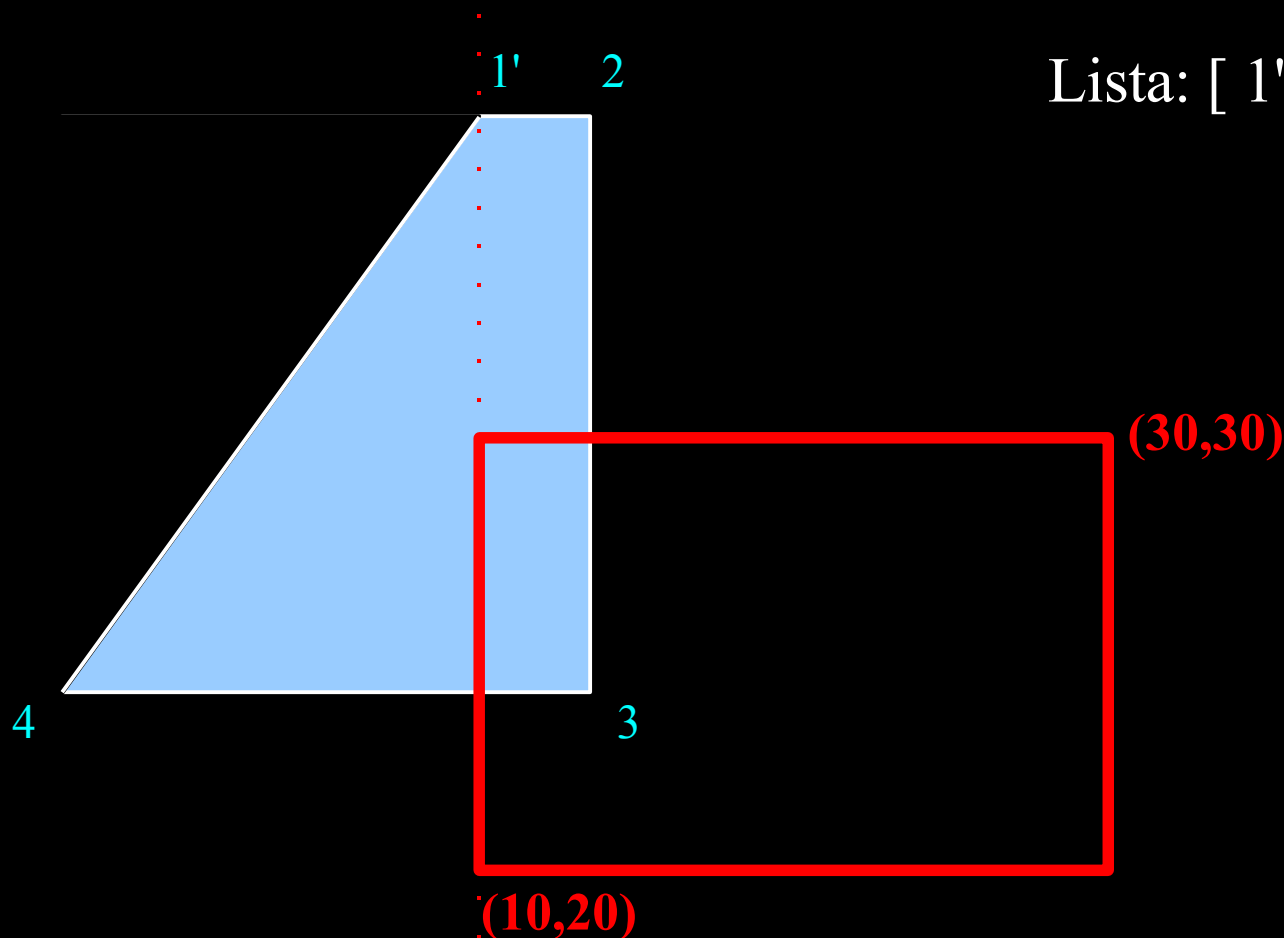




Parte I: Computação Gráfica Básica - Implementação de um Sistema Gráfico Interativo

Aula 4: Clipping # 137

- Vértices 3 e 4: Dentro, Fora
- Mantém 3, Remove 4, Adiciona 4' entre 3 e 4





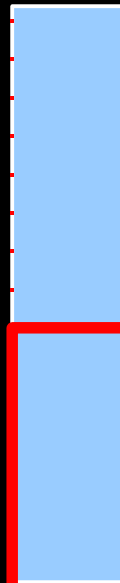
Parte I: Computação Gráfica Básica - Implementação de um Sistema Gráfico Interativo

Aula 4: Clipping # 138

- Vértices 3 e 4: Dentro, Fora
- Mantém 3, Remove 4, Adiciona 4' entre 3 e 4

1' 2

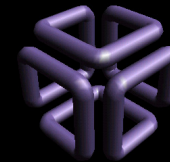
Lista: [1', 2, 3, 4']



(30,30)

4' 3

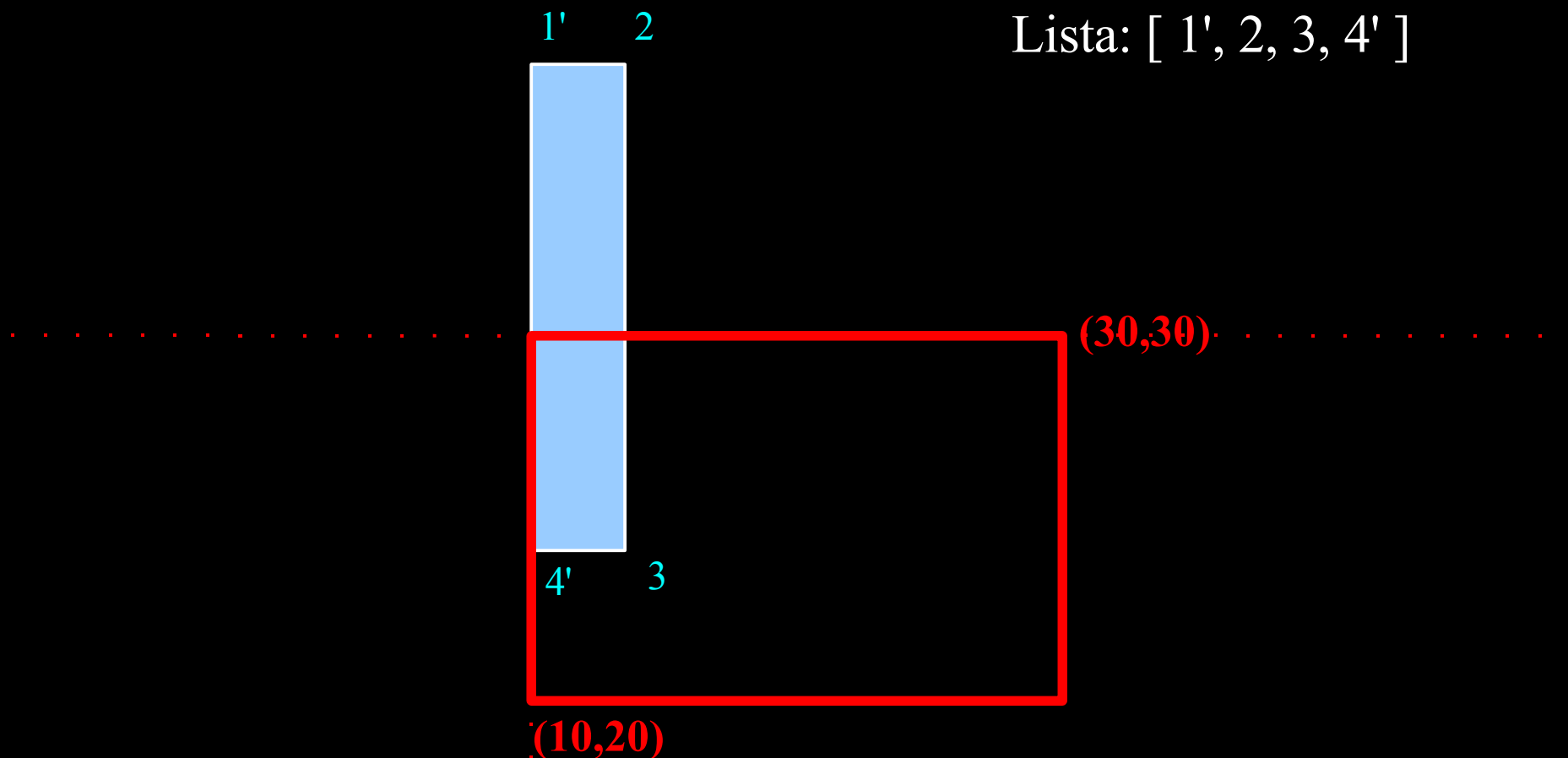
(10,20)

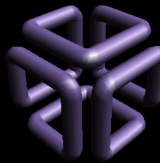


Parte I: Computação Gráfica Básica - Implementação de um Sistema Gráfico Interativo

Aula 4: Clipping # 139

- Borda Superior:

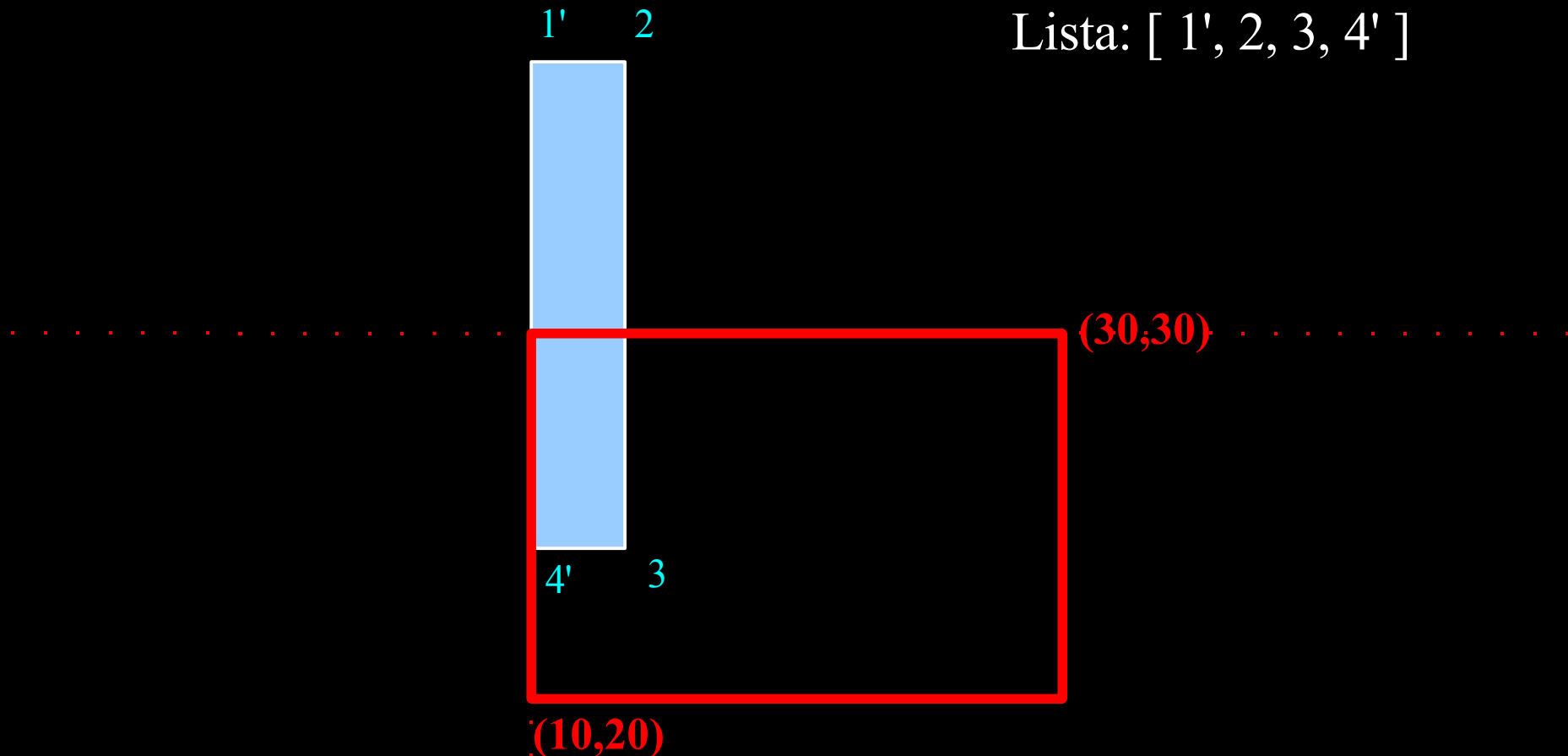




Parte I: Computação Gráfica Básica - Implementação de um Sistema Gráfico Interativo

Aula 4: Clipping # 140

- Vértices 1' e 2: Fora, Fora
- Remove 1', Adiciona 1'' entre 1' e 4

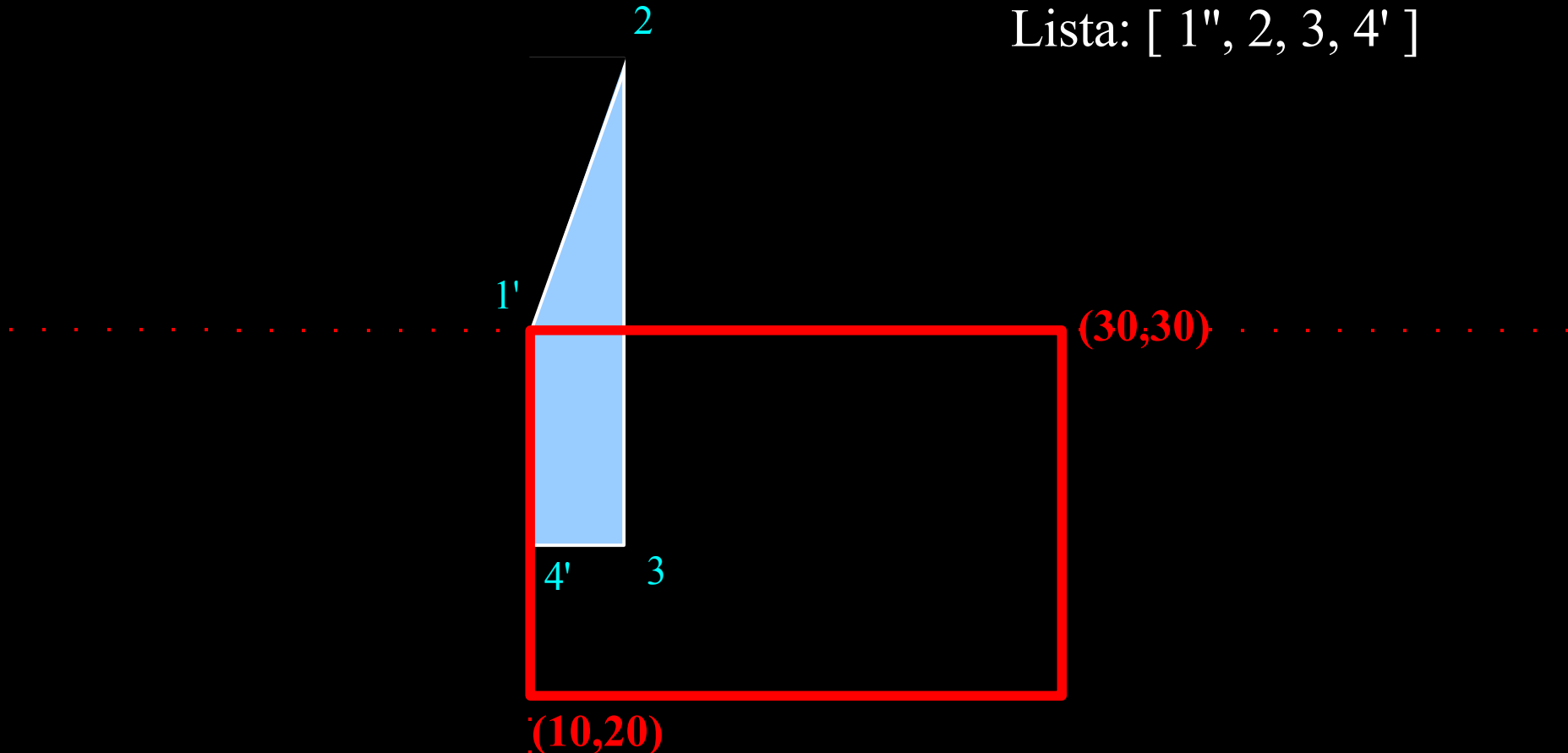




Parte I: Computação Gráfica Básica - Implementação de um Sistema Gráfico Interativo

Aula 4: Clipping # 141

- Vértices 1' e 2: Fora, Fora
- Remove 1', Adiciona 1'' entre 1' e 4

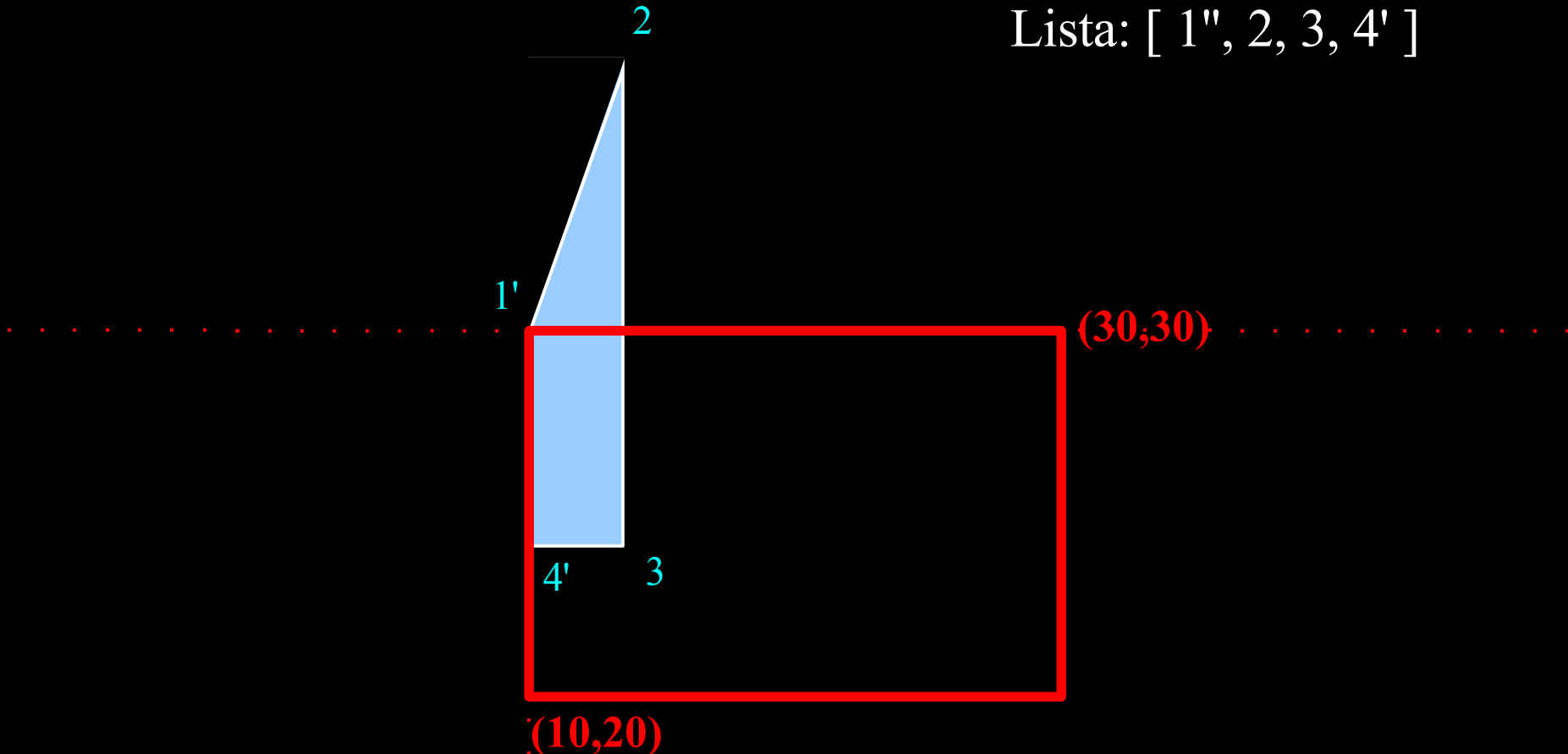




Parte I: Computação Gráfica Básica - Implementação de um Sistema Gráfico Interativo

Aula 4: Clipping # 142

- Vértices 1' e 2: Fora, for a (cont)
- Remove 2, Adiciona 2' entre 2 e 3



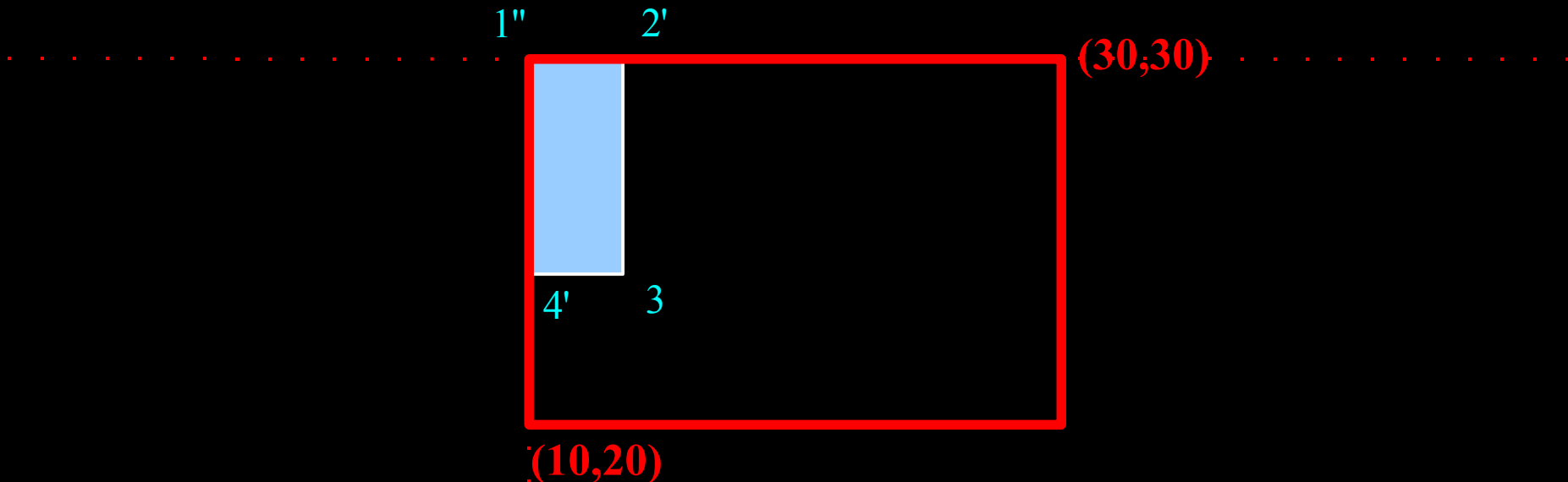


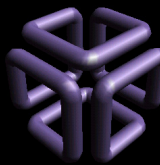
Parte I: Computação Gráfica Básica - Implementação de um Sistema Gráfico Interativo

Aula 4: Clipping # 143

- Vértices 1' e 2: Fora, for a (cont)
- Remove 2, Adiciona 2' entre 2 e 3

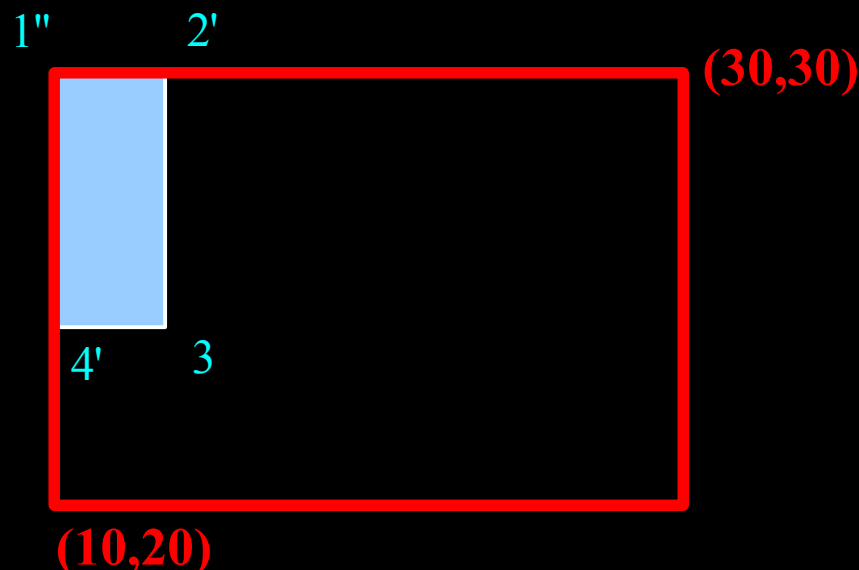
Lista: [1'', 2', 3, 4']

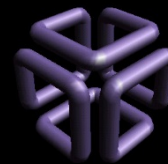




- Polígono final clippado:
 - O algoritmo continua, mas todos os vértices já estão dentro, neste caso...

Lista: [1'', 2', 3, 4']



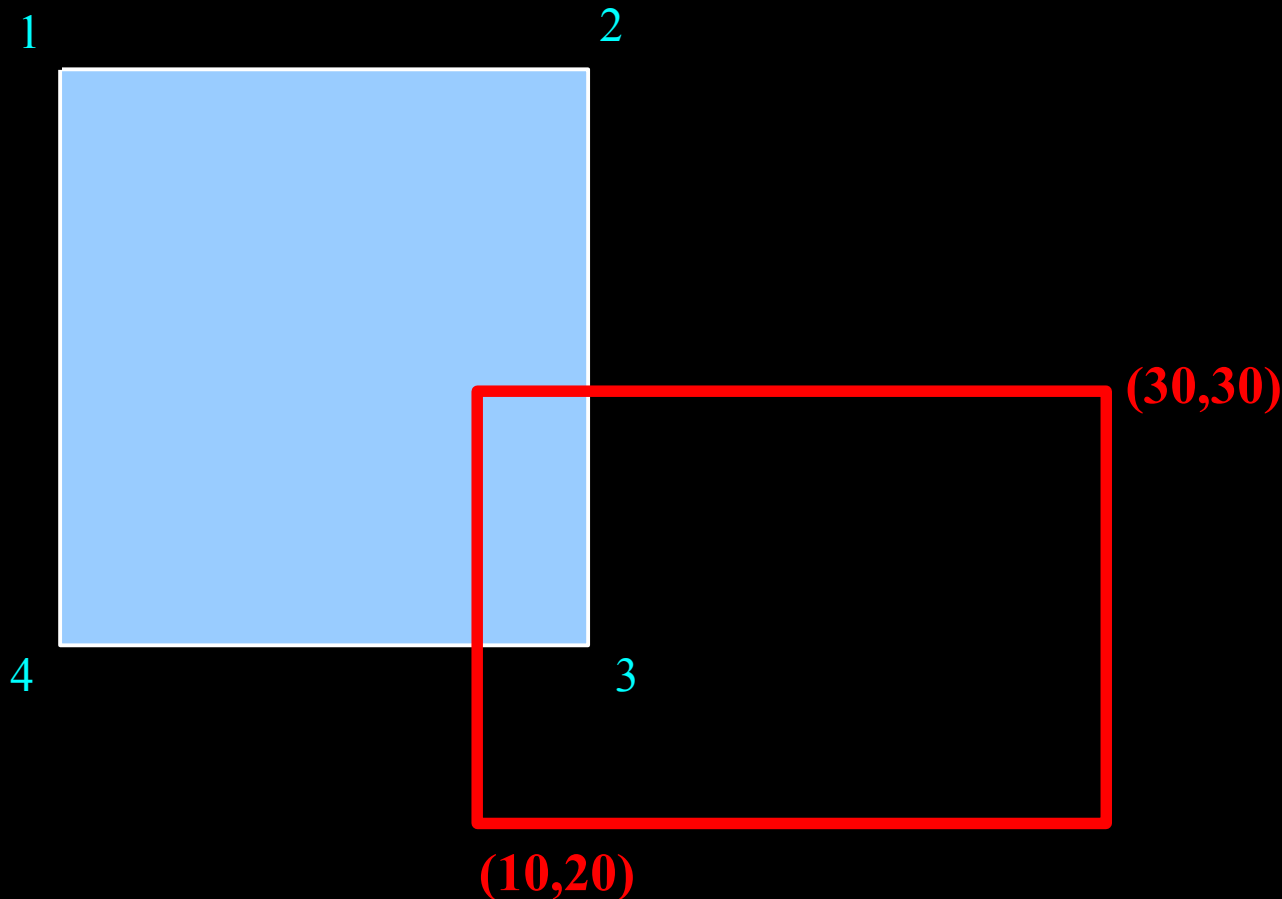


Método #2: Weiler-Atherton

- Percorra a lista de vértices
 - Percorra a lista de vértices adicionando ou removendo pontos conforme o caso.
 - Se processamos em *sentido horário* procedemos assim:
 - De fora para dentro: siga a borda do polígono.
 - De dentro para fora: siga a borda da window em *sentido horário*.



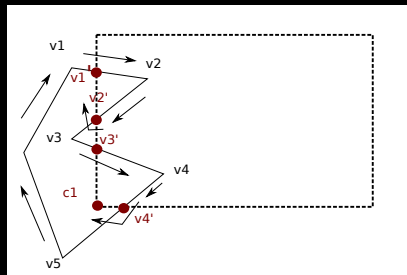
Clipping de Polígonos





Weiler-Atherton Clipping

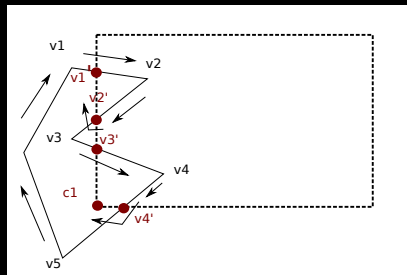
- Percorrer a lista em sentido horário





Weiler-Atherton Clipping

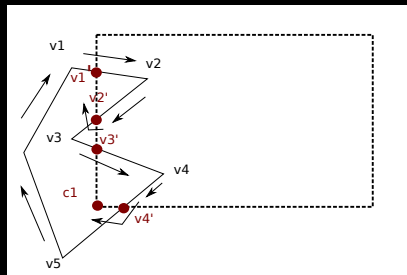
- Percorrer a lista em sentido horário
 - $v1'$ (entra), $v2$, $v2'$ (sai), $v1'$.





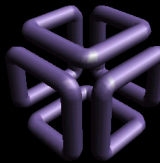
Weiler-Atherton Clipping

- Percorrer a lista em sentido horário
 - $v1'$ (entra), $v2$, $v2'$ (sai), $v1'$.
 - $v3'$ (entra), $v4$, $v4'$ (sai), $c1$.

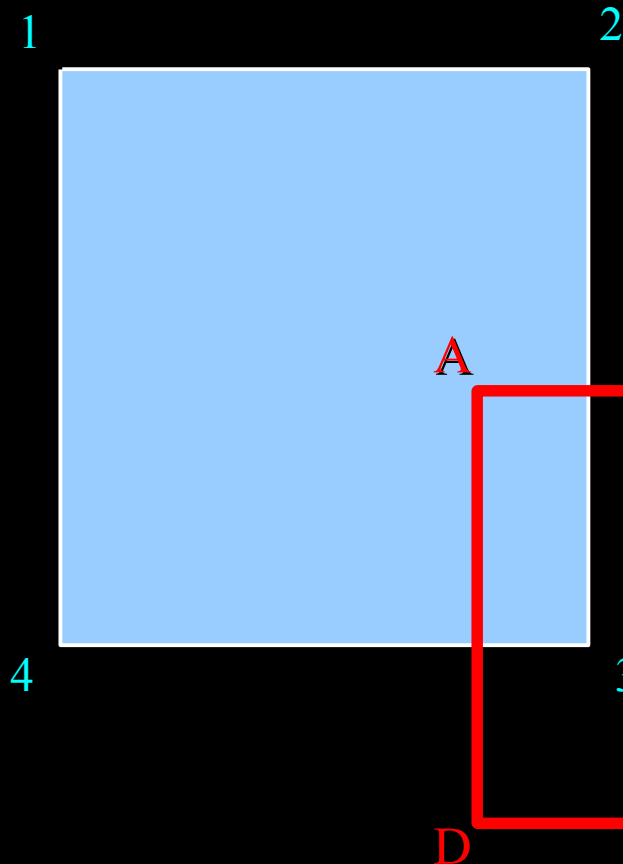




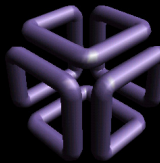
- Listas Ligadas:
 - Uma com os vértices do polígono
 - Outra com os vértices da Window
 - Uma terceira temporária para vértices entrantes



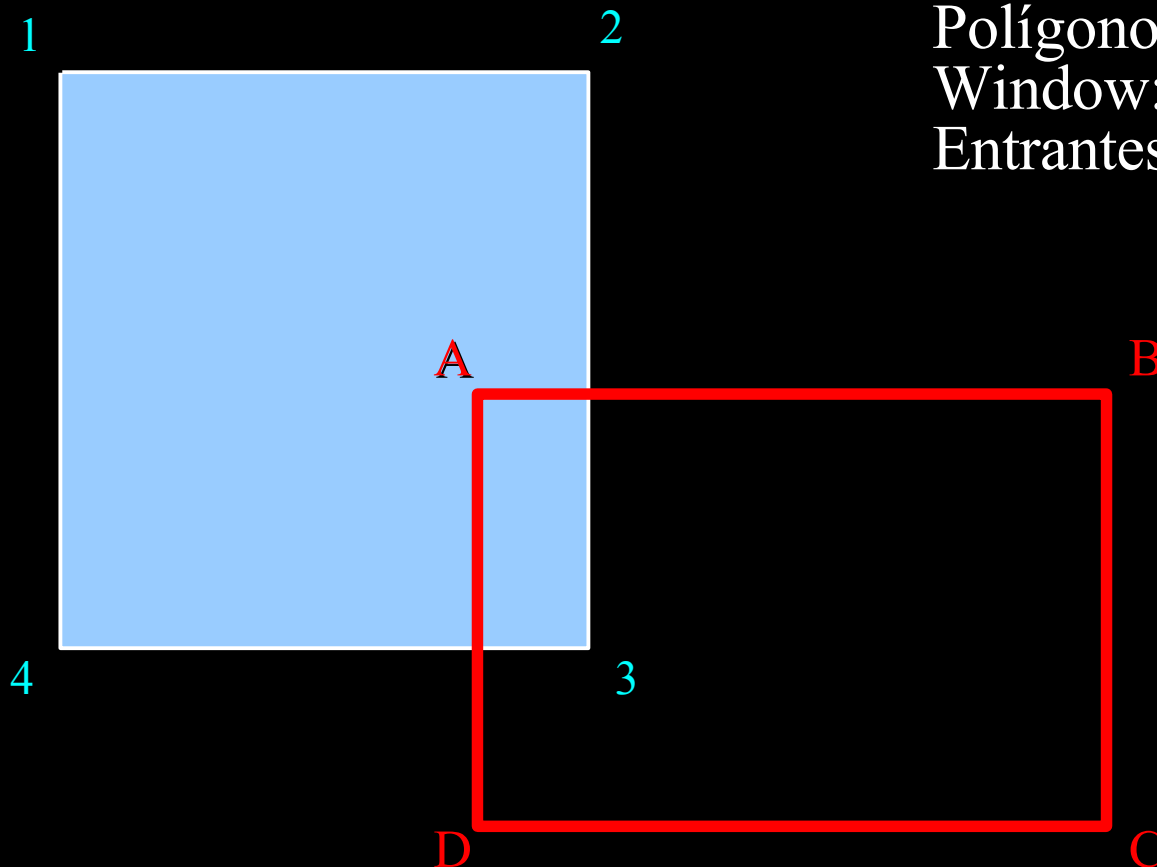
- Vértices 1 e 2: Não corta Window



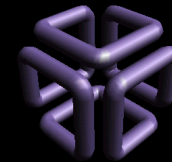
Polígono: [1, 2, 3, 4]
Window: [A, B, C, D]
Entrantes: []



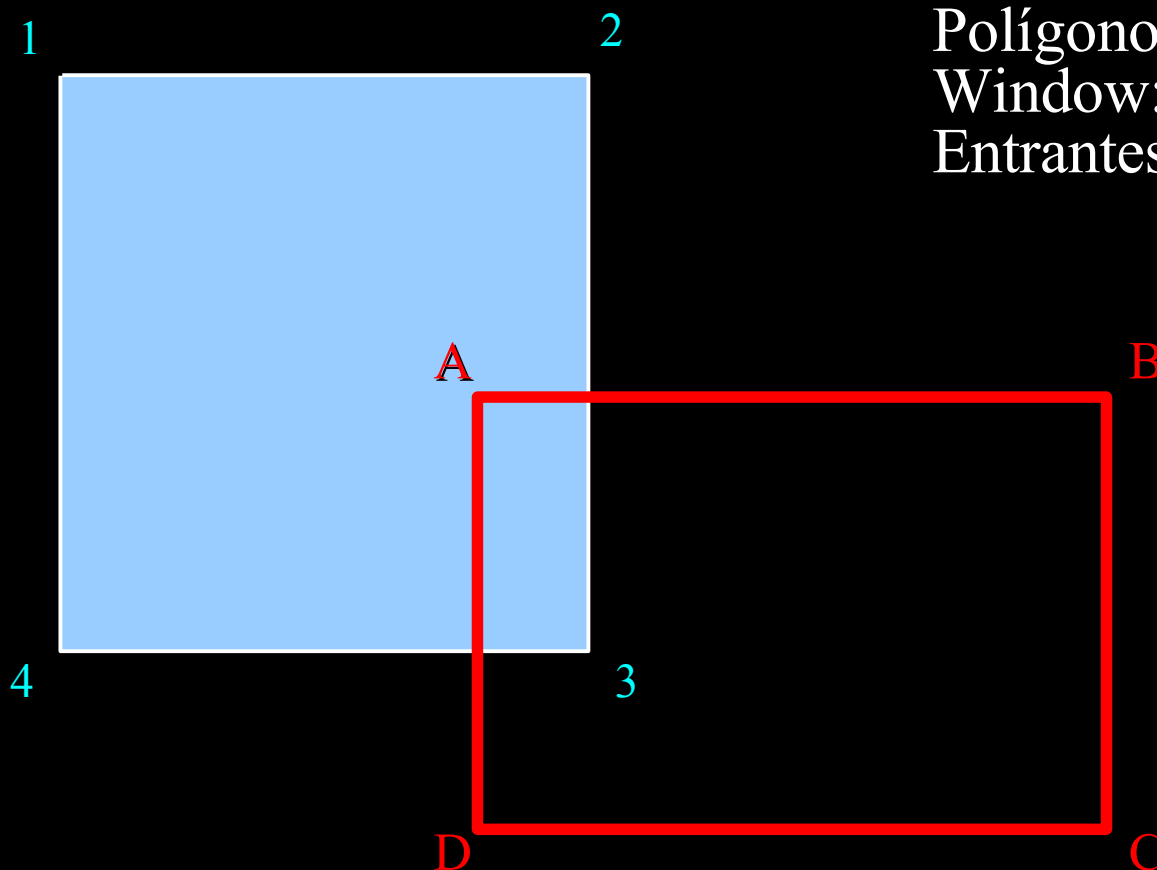
- Vértices 2 e 3: Corta a Window de fora pra dentro:



Polígono: [1, 2, 3, 4]
Window: [A, B, C, D]
Entrantes: []



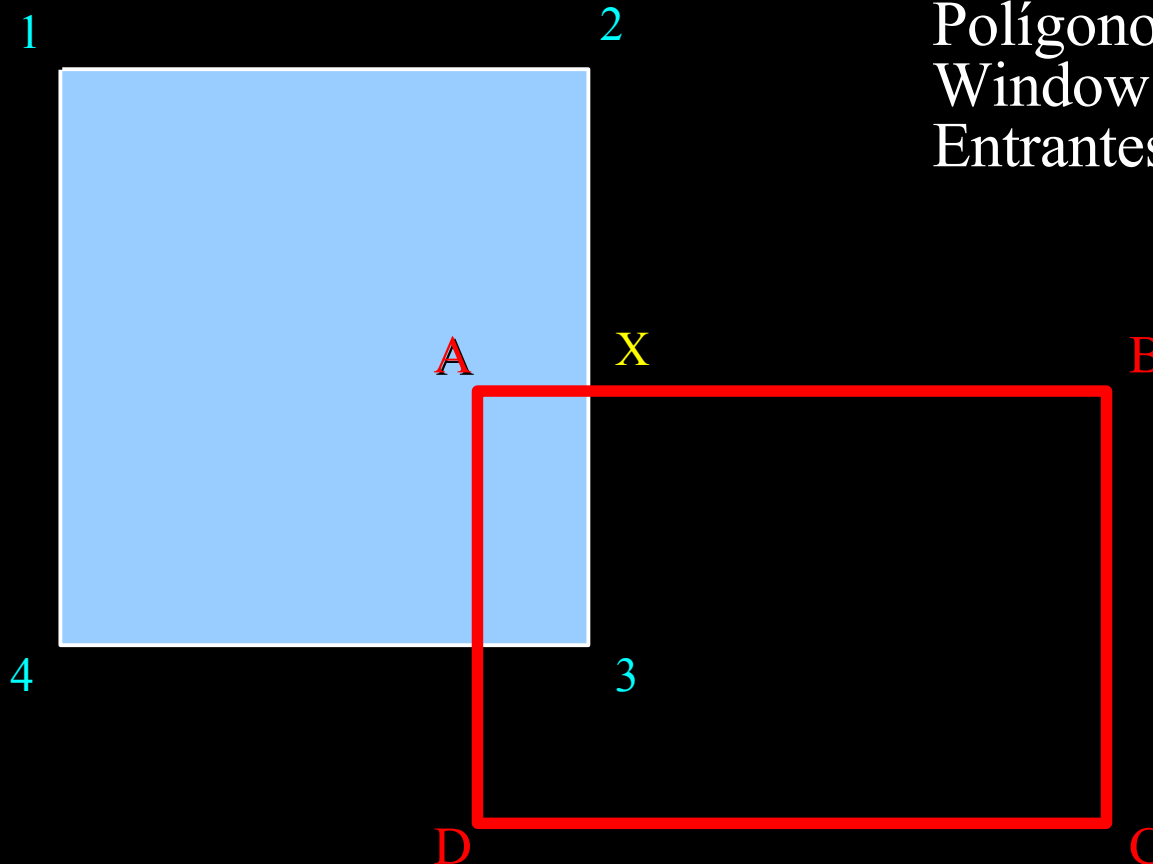
- Vértices 2 e 3: Corta a Window de fora pra dentro:
- Criamos vértice X, adicionado às 3 listas



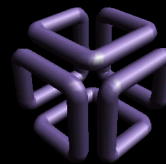
Polígono: [1, 2, 3, 4]
Window: [A, B, C, D]
Entrantes: []



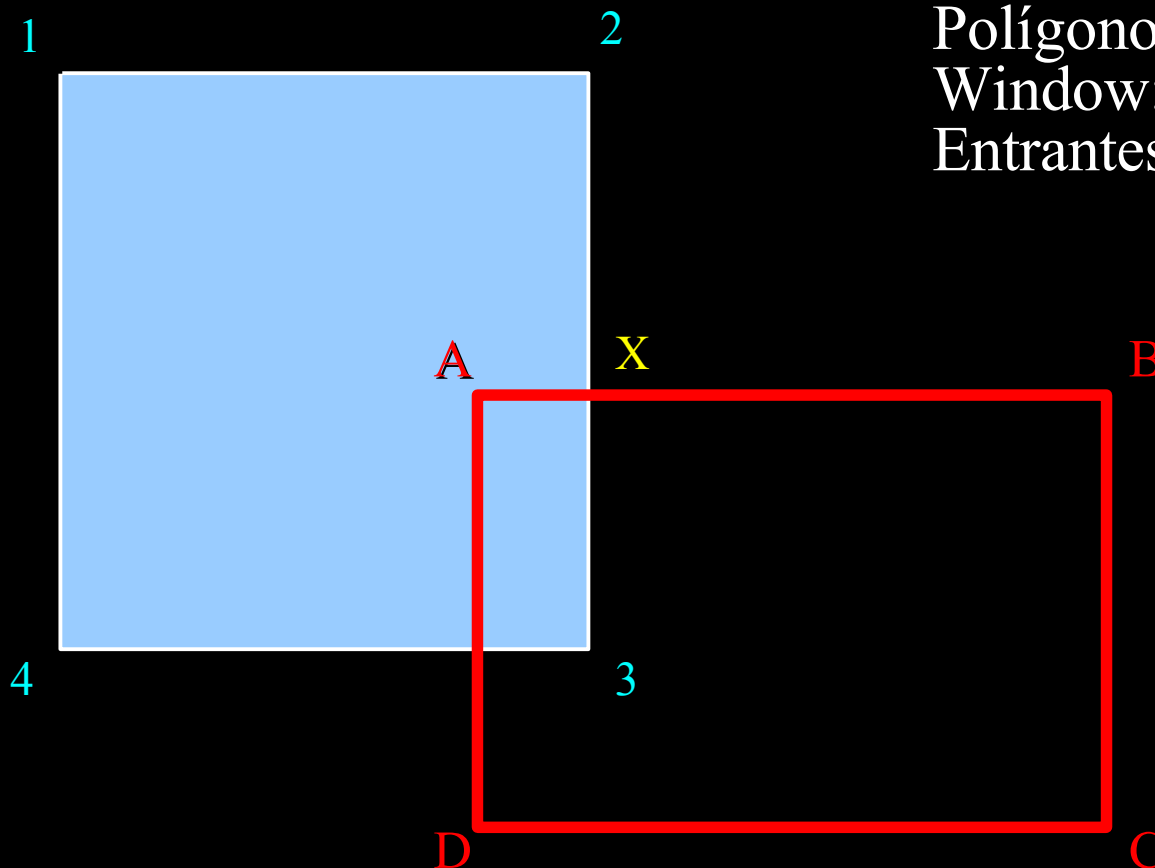
- Vértices 2 e 3: Corta a Window de fora pra dentro:
- Criamos vértice X, adicionado às 3 listas



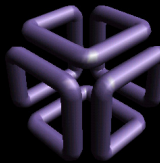
Polígono: [1, 2, X, 3, 4]
Window: [A, X, B, C, D]
Entrantes: [X]



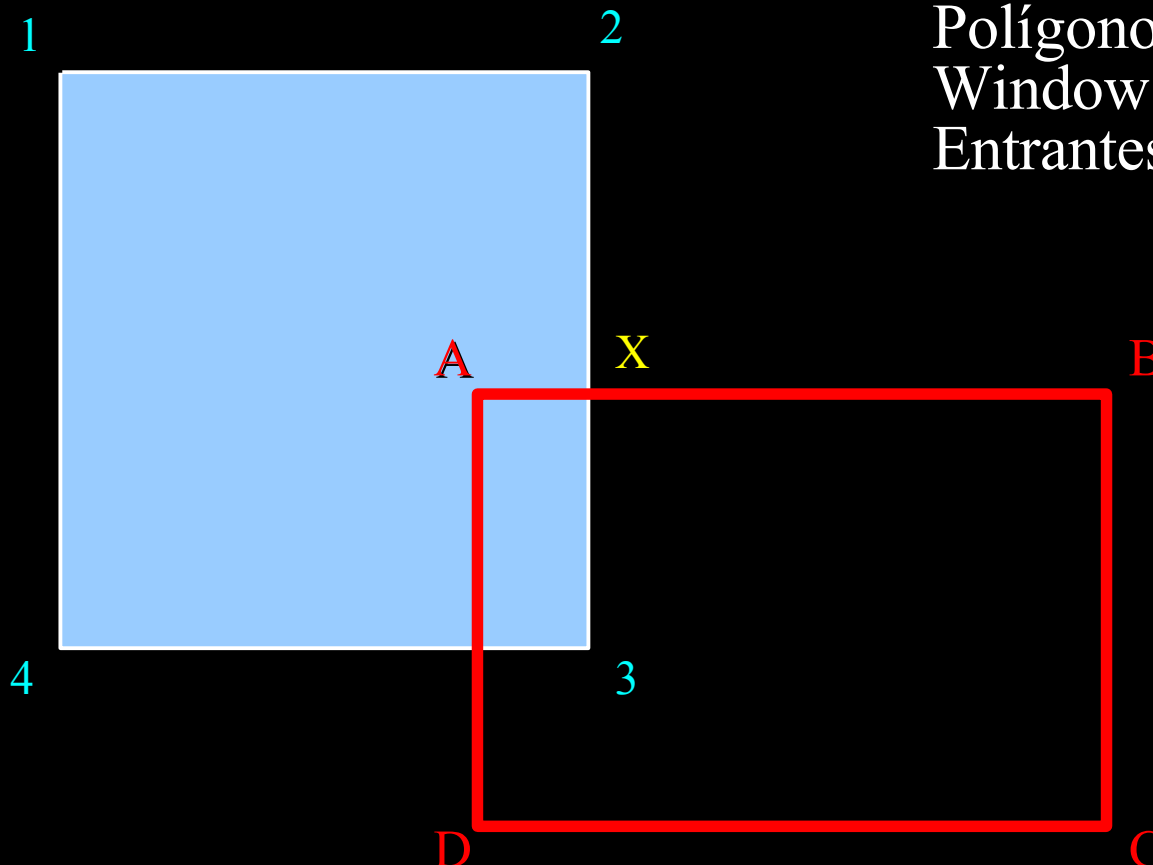
- Vértices 3 e 4: Corta a Window de dentro pra fora:



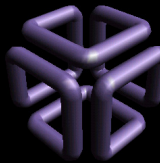
Polígono: [1, 2, X, 3, 4]
Window: [A, X, B, C, D]
Entrantes: [X]



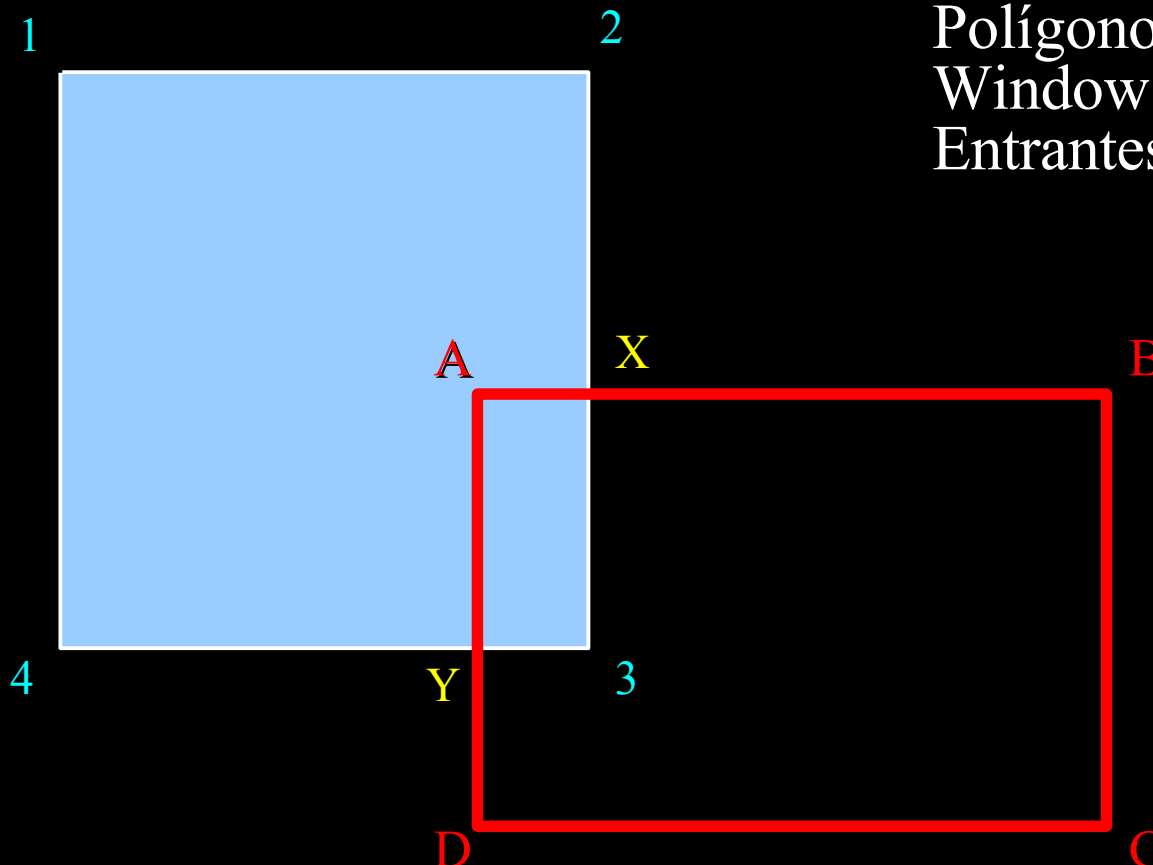
- Vértices 3 e 4: Corta a Window de dentro pra fora:
- Criamos vértice Y, adicionado às 2 primeiras listas



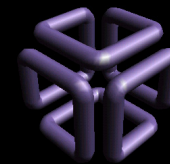
Polígono: [1, 2, X, 3, 4]
Window: [A, X, B, C, D]
Entrantes: [X]



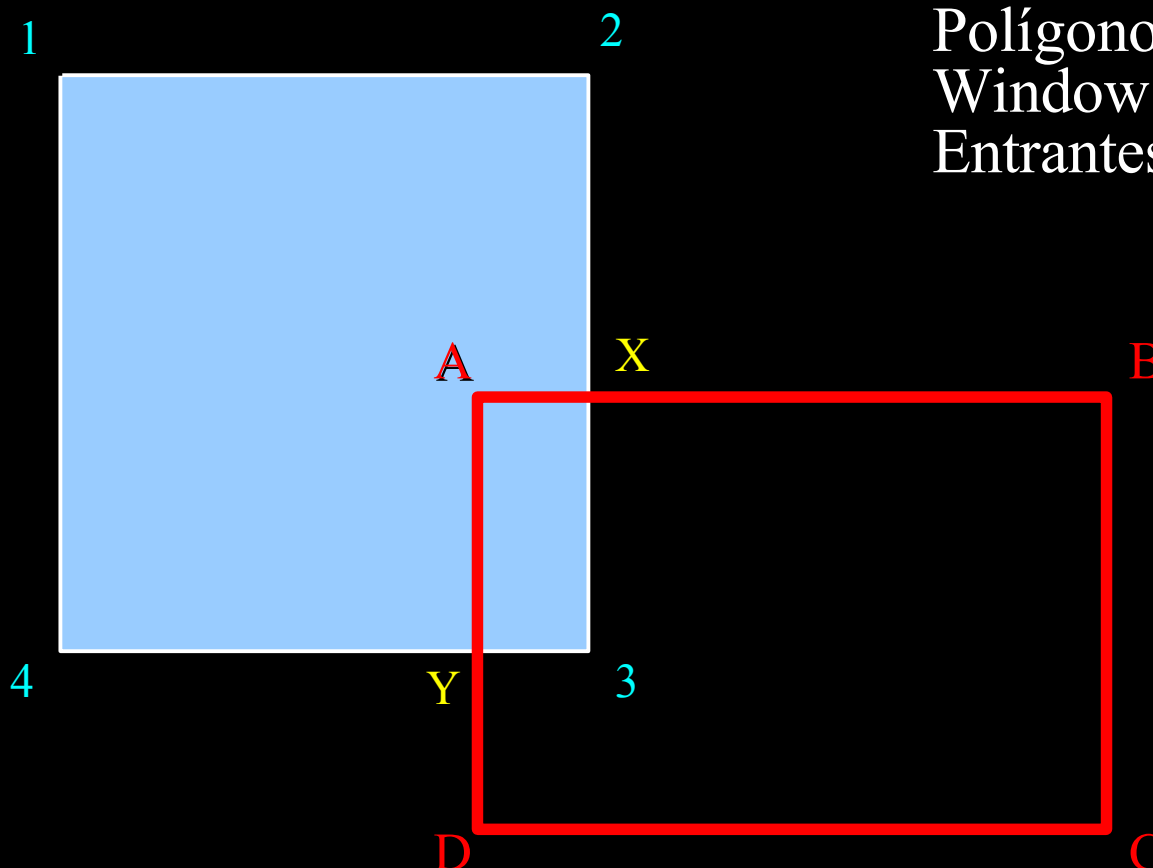
- Vértices 3 e 4: Corta a Window de dentro pra fora:
- Criamos vértice Y, adicionado às 2 primeiras listas



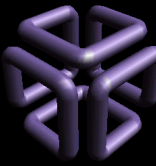
Polígono: [1, 2, X, 3, Y, 4]
Window: [A, X, B, C, D, Y]
Entrantes: [X]



- Os próximos vértices não cortam a Window



Polígono: [1, 2, X, 3, Y, 4]
Window: [A, X, B, C, D, Y]
Entrantes: [X]



- Próximo passo:
- Para cada Ponto em “Entrantes”...:
 - Percorra a lista “Polígono” partindo deste ponto em diante – vá guardando os vértices percorridos;
 - Ao encontrar outro vértice adicionado artificialmente, troque para a lista “Window”;



- Para cada Ponto em “Entrantes”... (cont):
 - Percorra a lista “Window” até encontrar outro vértice artificial – continue guardando os vértices;
 - Ao encontrar outro vértice artificial, pare;
 - Os vértices percorridos formam o novo polígono;
 - Remova o Ponto de “Entrantes”.



Parte I: Computação Gráfica Básica - Implementação de um Sistema Gráfico Interativo

Aula 4: Clipping # 158

Polígono: [1, 2, X, 3, Y, 4]
Window: [A, X, B, C, D, Y]
Entrantes: [X]
Temporária: []

- Ponto em Entrantes: “X”;



Parte I: Computação Gráfica Básica - Implementação de um Sistema Gráfico Interativo

Aula 4: Clipping # 159

Polígono: [1, 2, **X**, 3, **Y**, 4]
Window: [A, **X**, B, C, D, **Y**]
Entrantes: [**X**]
Temporária: []

- Ponto em Entrantes: “X”;
- Percorrendo “Polígono” de X até Y ...



Parte I: Computação Gráfica Básica - Implementação de um Sistema Gráfico Interativo

Aula 4: Clipping # 160

Polígono: [1, 2, X, 3, Y, 4]
Window: [A, X, B, C, D, Y]
Entrantes: [X]
Temporária: [X]

- Ponto em Entrantes: “X”;
- Percorrendo “Polígono” de X até Y ...



Parte I: Computação Gráfica Básica - Implementação de um Sistema Gráfico Interativo

Aula 4: Clipping # 161

Polígono: [1, 2, X, 3, Y, 4]
Window: [A, X, B, C, D, Y]
Entrantes: [X]
Temporária: [X, 3]

- Ponto em Entrantes: “X”;
- Percorrendo “Polígono” de X até Y ...



Parte I: Computação Gráfica Básica - Implementação de um Sistema Gráfico Interativo

Aula 4: Clipping # 162

Polígono: [1, 2, X, 3, Y, 4]
Window: [A, X, B, C, D, Y]
Entrantes: [X]
Temporária: [X, 3, Y]

- Ponto em Entrantes: “X”;
- Percorrendo “Polígono” de X até Y ...



Parte I: Computação Gráfica Básica - Implementação de um Sistema Gráfico Interativo

Aula 4: Clipping # 163

Polígono: [1, 2, X, 3, Y, 4]
Window: [A, X, B, C, D, Y]
Entrantes: [X]
Temporária: [X, 3, Y]

- Ponto em Entrantes: “X”;
- Percorrendo “Polígono” de X até Y ...
- Percorrendo “Window” de Y até X ...



Parte I: Computação Gráfica Básica - Implementação de um Sistema Gráfico Interativo

Aula 4: Clipping # 164

Polígono: [1, 2, X, 3, Y, 4]
Window: [A, X, B, C, D, Y]
Entrantes: [X]
Temporária: [X, 3, Y, A]

- Ponto em Entrantes: “X”;
- Percorrendo “Polígono” de X até Y ...
- Percorrendo “Window” de Y até X ...



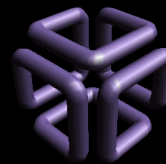
Polígono: [1, 2, X, 3, Y, 4]
Window: [A, X, B, C, D, Y]
Entrantes: [X]
Temporária: [X, 3, Y, A, X]

- Ponto em Entrantes: “X”;
- Percorrendo “Polígono” de X até Y ...
- Percorrendo “Window” de Y até X ...



Polígono: [1, 2, X, 3, Y, 4]
Window: [A, X, B, C, D, Y]
Entrantes: [X]
Temporária: [X, 3, Y, A, X]

- Ponto em Entrantes: “X”;
- Percorrendo “Polígono” de X até Y ...
- Percorrendo “Window” de Y até X ...
- Alcançamos outro “artificial”
 - Não precisava ser X, necessariamente!



Parte I: Computação Gráfica Básica - Implementação de um Sistema Gráfico Interativo

Aula 4: Clipping # 167

Polígono: [1, 2, X, 3, Y, 4]
Window: [A, X, B, C, D, Y]
Entrantes: [X]
Temporária: [X, 3, Y, A, X]

- Polígono final: [X, 3, Y, A, X]

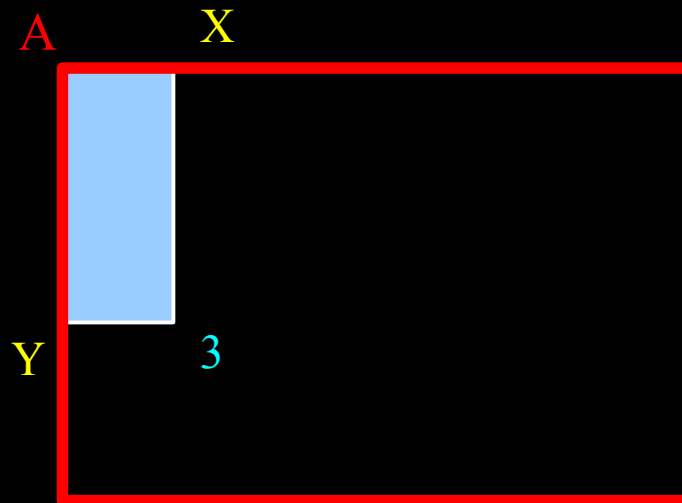


Parte I: Computação Gráfica Básica - Implementação de um Sistema Gráfico Interativo

Aula 4: Clipping # 168

Polígono: [1, 2, X, 3, Y, 4]
Window: [A, X, B, C, D, Y]
Entrantes: [X]
Temporária: [X, 3, Y, A]

- Polígono final: [X, 3, Y, A, X]





The Cyclops Project

German-Brazilian Cooperation Programme on IT
CNPq GMD DLR

Disciplina Computação Gráfica

Curso de Ciência da Computação
INE/CTC/UFSC



Parte I: Computação Gráfica Básica - Implementação de um Sistema Gráfico Interativo

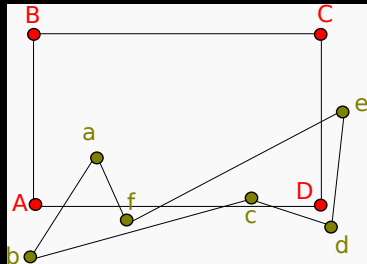
Aula 4: Clipping # 87

Exemplo #2: Weiler-Atherton



Weiler-Atherton Clipping

A B C D
a f e d c b

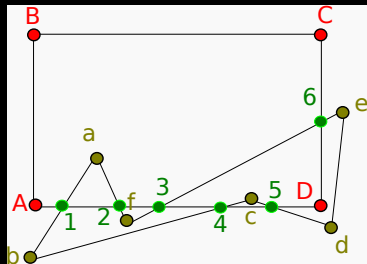


- Listar pontos da window e do polígono (em sentido horário).



Weiler-Atherton Clipping

A	B	C	6	D	5	4	3	2	1		
a	2	f	3	6	e	d	5	c	4	b	1

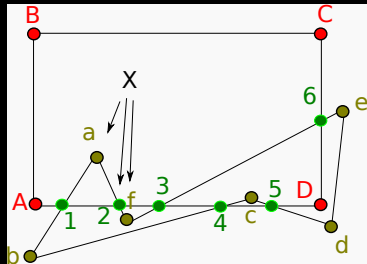


- Encontrar o ponto de intersecção de cada segmento de linha.
- Adicionar cada ponto a sua respectiva lista.



Weiler-Atherton Clipping

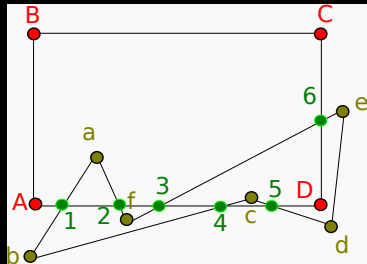
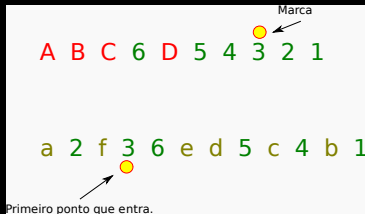
A	B	C	6	D	5	4	3	2	1		
a	2	f	3	6	e	d	5	c	4	b	1



- Percorrer a lista do polígono até encontrar um ponto de intersecção que "entre" na window.

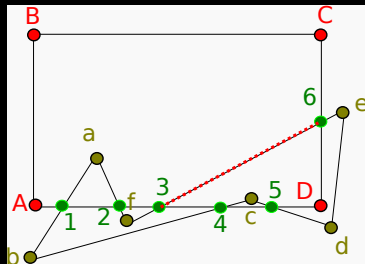
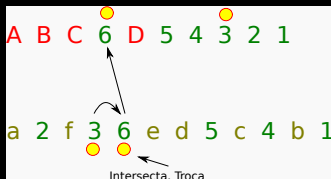


Weiler-Atherton Clipping



- Marcar o ponto de entrada (3) em ambas as listas e continuar percorrendo a lista de pontos do polígono.

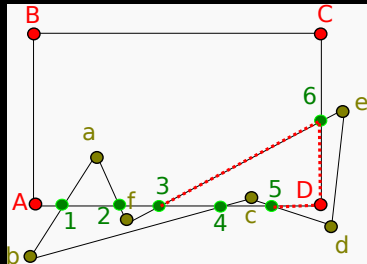
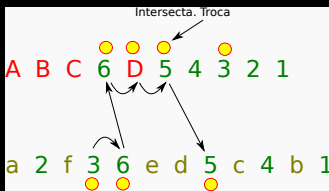
Weiler-Atherton Clipping



- Percorrer a lista (marcando os pontos visitados) até encontrar uma intersecção. Ao encontrar uma intersecção troque de lista.



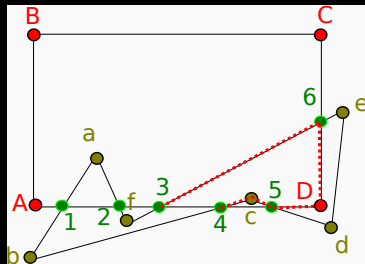
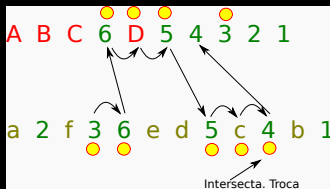
Weiler-Atherton Clipping



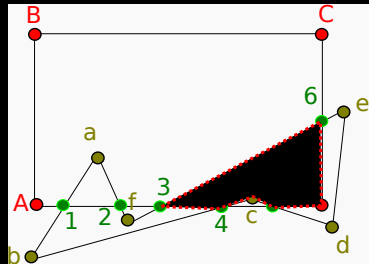
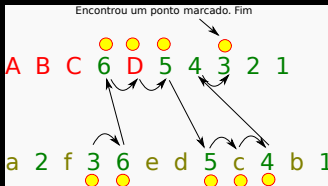
- Continuar percorrendo a lista (procurando um ponto visitado) e sempre que encontrar uma intersecção trocar de lista.



Weiler-Atherton Clipping



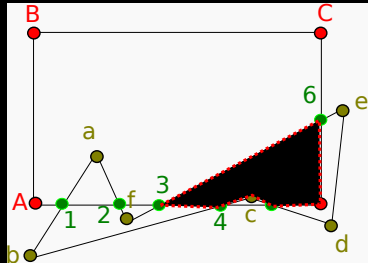
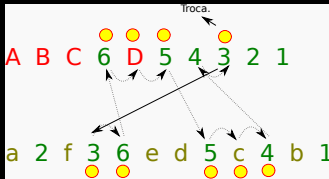
Weiler-Atherton Clipping



- Encontrou um ponto visitado fecha o polígono.



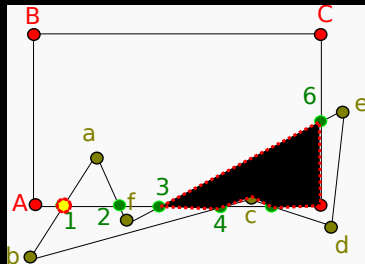
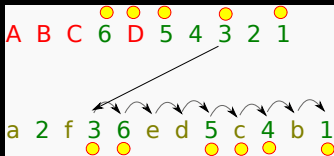
Weiler-Atherton Clipping



- Retorna para lista de pontos do polígono e busca um ponto que "entre" e que não tenha sido marcado.

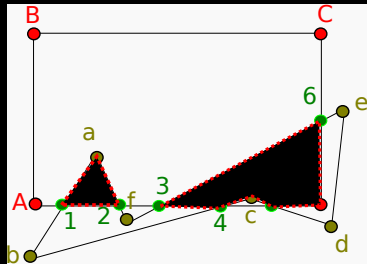
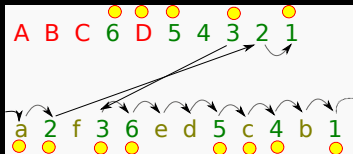


Weiler-Atherton Clipping



- Repete os passos anteriores.

Weiler-Atherton Clipping





Exercício em Papel #2:

- Tome os dados do slide adiante:
 - Considere já em SCN:
 - Window de dimensões 2 x 2 com WC em (0,0)
 - SCN no intervalo [-1,1]
- Faça:
 - Calcule o Clipping de todos os polígonos fechados
 - Clipe os objetos usando todos os dois algoritmos vistos:
 - Sutherland-Hodgeman
 - Weiler-Atherton
 - Desenhe o resultado

