

Clipping (recorte)

Tomado de:

Prof. Möller et al. Universidad Simon Fraser

<http://www.cs.sfu.ca/~torsten/Teaching/Cmpt361>

Capitulo 7 – Angel Edward

Capitulo 3 – Foley et al.

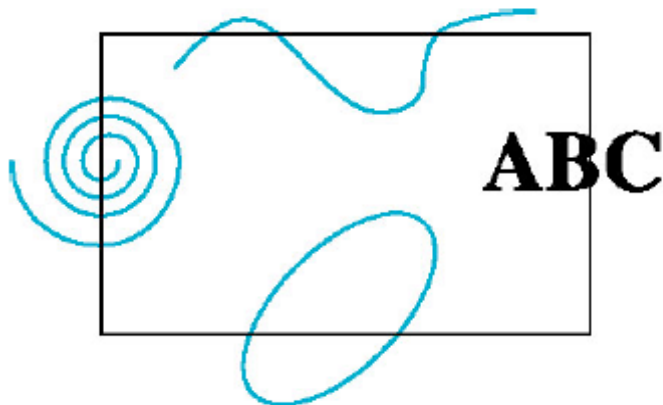


- ❑ Disminuir o por lo menos evitar costo computacional elevado
 - ❑ Iluminación
 - ❑ Determinación de visibilidad
 - ❑ Rasterización (scan conversion)
- ❑ Tambien es usado para:
 - ❑ Suavisar límites de los objetos
 - ❑ Modelado de sólidos
 - ❑ Extraer partes de una imagen (borrar, mover o copiar)

Recorte (Clipping)



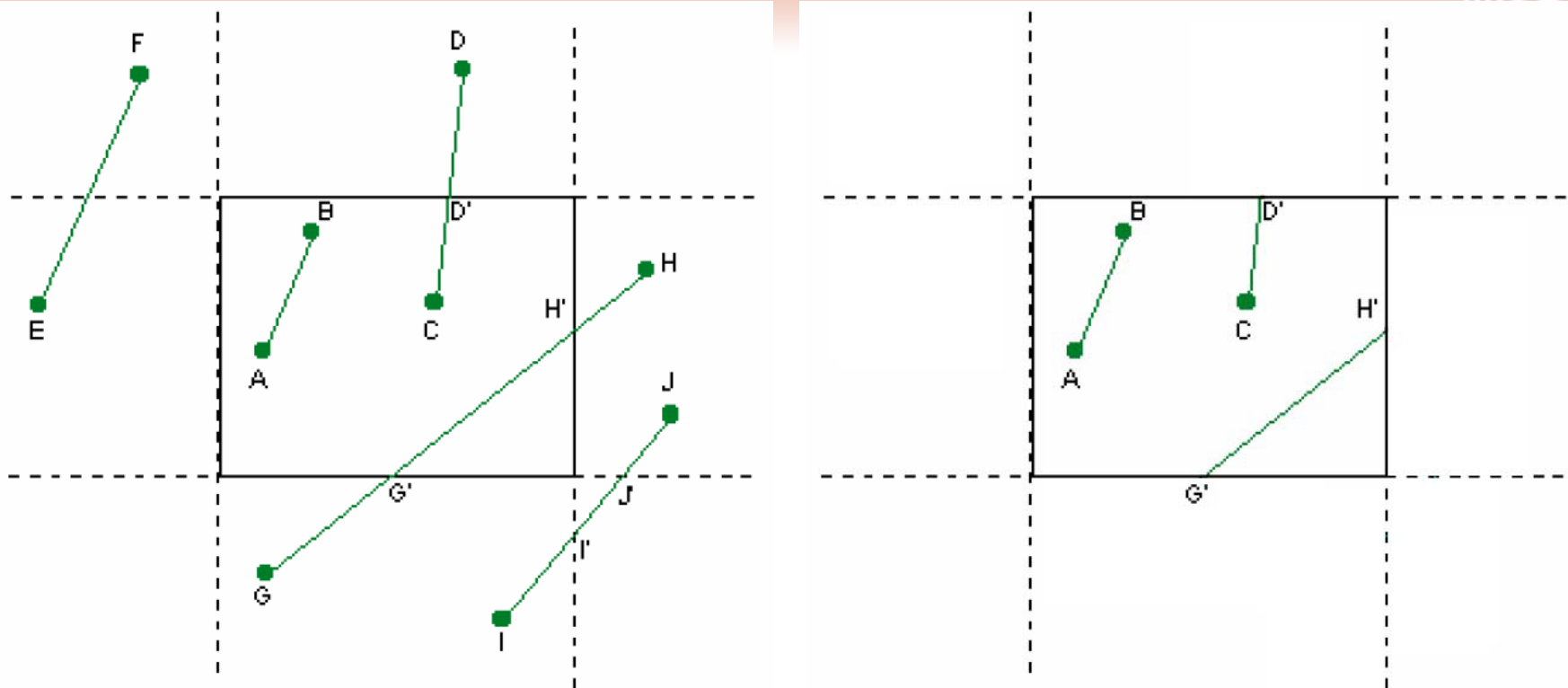
- ❑ Recorte bidimensional: una ventana
- ❑ Recorte tridimensional: recorte de volumen
- ❑ No es complicado para los casos de líneas o polígonos
- ❑ Difícil en caso de curvas y letras
 - ❑ Se convierten a polígonos





- Definidas las fronteras del rectángulo de recorte
 - $X_{\min}, X_{\max}, Y_{\min}, Y_{\max}$
 - $X_{\min} \leq x \leq X_{\max}$
 - $Y_{\min} \leq y \leq Y_{\max}$

Recorte de líneas



AB se aceptan trivialmente

CD un punto extremo dentro otro fuera

EF, GH e IJ Los puntos extremos caen fuera



- Analítico (Ecuaciones simultáneas)
- Cohen-Sutherland
- Liang-Barsky



❑ Fuerza bruta

❑ Intersectar la línea con cada uno de los cuatro bordes de recorte
(Xmin, Xmax, Ymin, Ymax)

❑ Chequear los puntos de intersección y verificar si caen dentro o fuera del área de recorte

$$p = p_0 + u(p_1 - p_0)$$

❑ Si u está entre $[0, 1]$ existe una intersección

❑ Este método es ineficiente

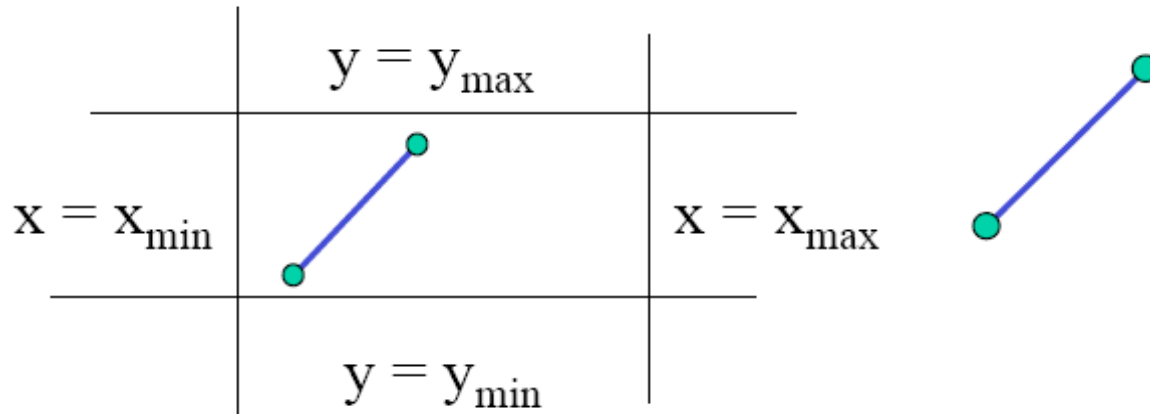


- ❑ Ejecuta el chequeo de los puntos extremos
- ❑ Si la línea no es rechazada o aceptada trivialmente, se subdivide
 - ❑ Trivialmente aceptada: ambos puntos extremos caen en el área de recorte
 - ❑ Trivialmente rechazada: ambos puntos extremos caen fuera del plano medio de los bordes del área de recorte

Cohen-Sutherland



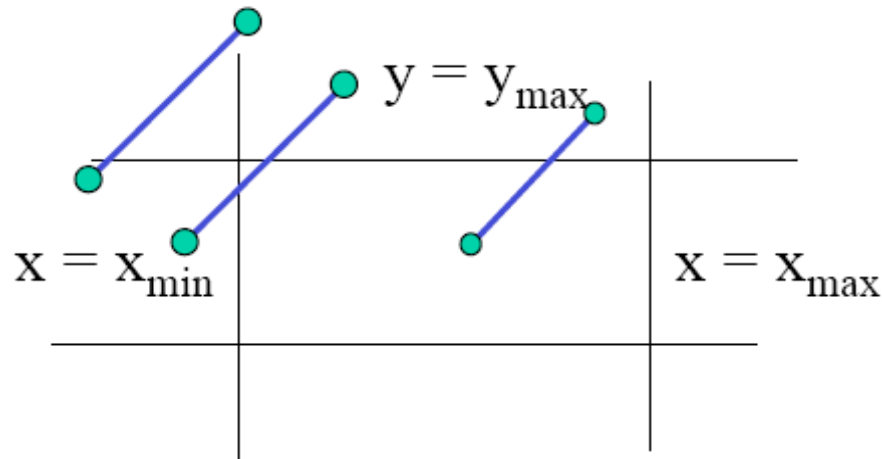
- ❑ Caso 1: ambos puntos extremos caen dentro del área de recorte
 - ❑ Trivialmente aceptada



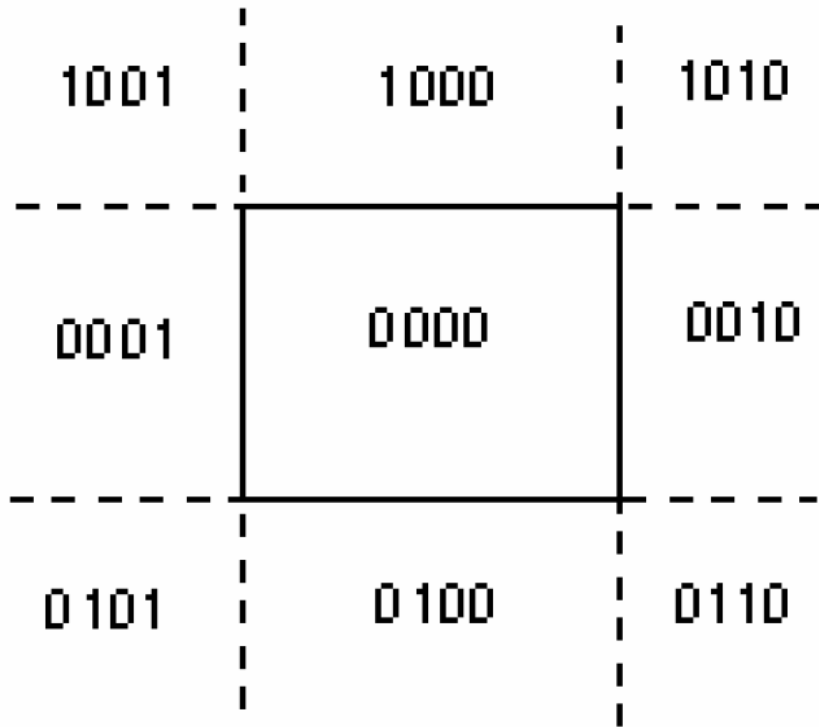
- ❑ Caso 2: ambos puntos extremos caen fuera de las líneas del área de recorte y en un mismo lado de una de las líneas del área de recorte
 - ❑ Trivialmente rechazada



- ❑ Caso 3: uno de los puntos extremos cae dentro y otro fuera del área de recorte
 - ❑ Debe haber al menos una intersección



- ❑ Caso 4: ambos puntos extremos caen fuera de las del área de recorte
 - ❑ Pueden estar dentro o fuera



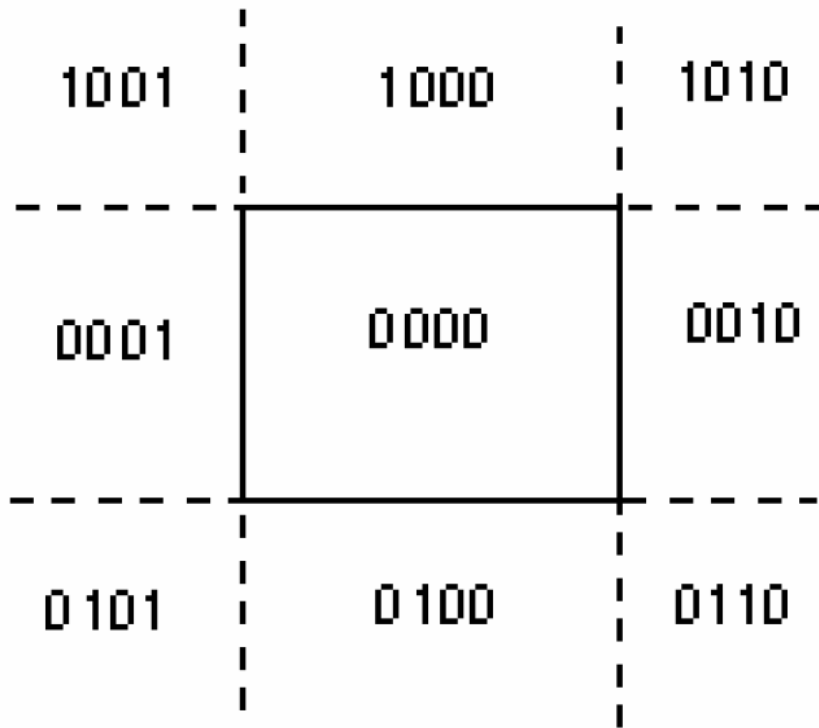
□ ABRL

□ $A := y > Y_{max}$

□ $B := y < Y_{min}$

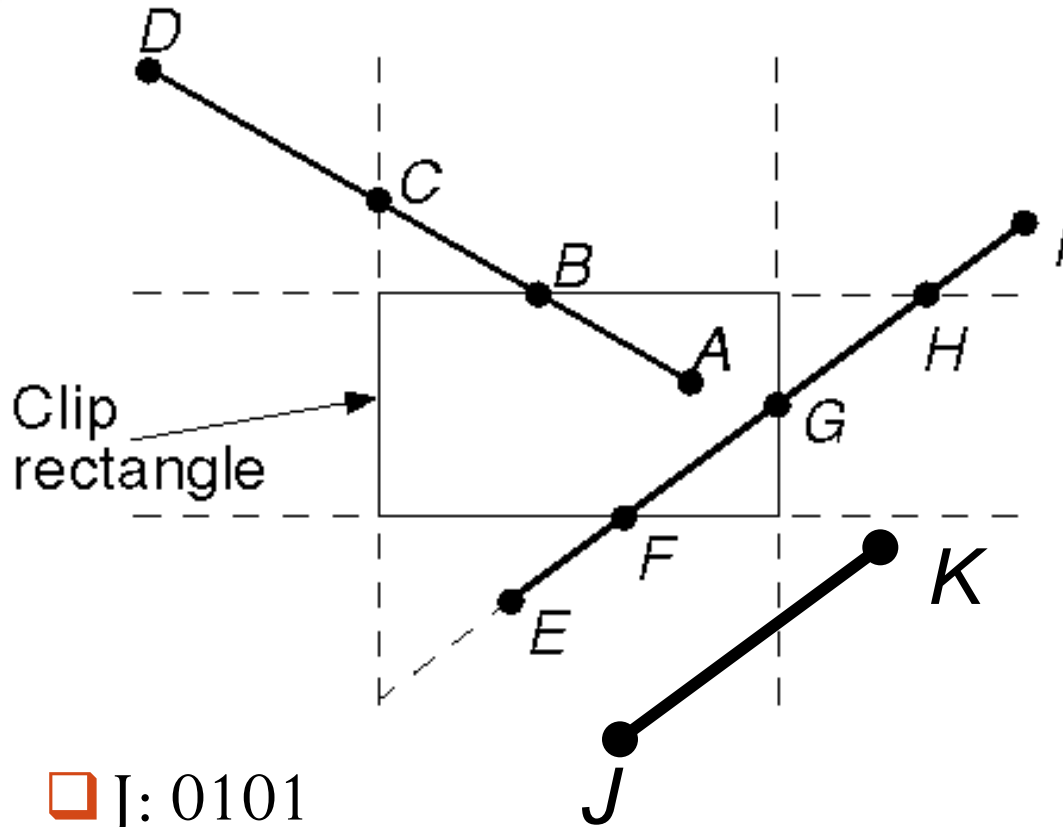
□ $R := x > X_{max}$

□ $L := x < X_{min}$



- ❑ Si los dos extremos de una línea tienen código cero, la línea está totalmente dentro del rectángulo.
- ❑ Si los puntos extremos están fuera del rectángulo de recorte, entonces, tendrán códigos con 1's intercalados y al aplicar la operación lógica and debe ser cero, sino se rechaza trivialmente.

Cohen-Sutherland



D: 1001

A: 0000

A (AND) D : 0000

E: 1010

I: 0100

E (AND) I: 0000

J: 0101

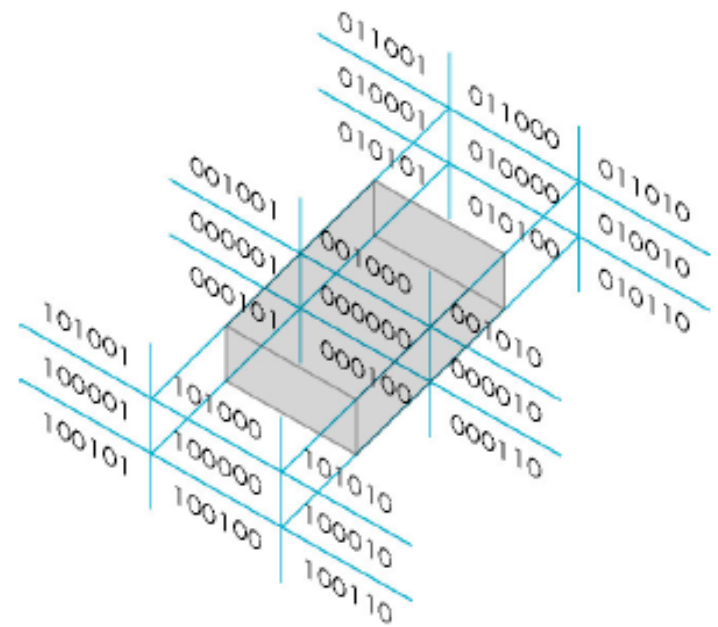
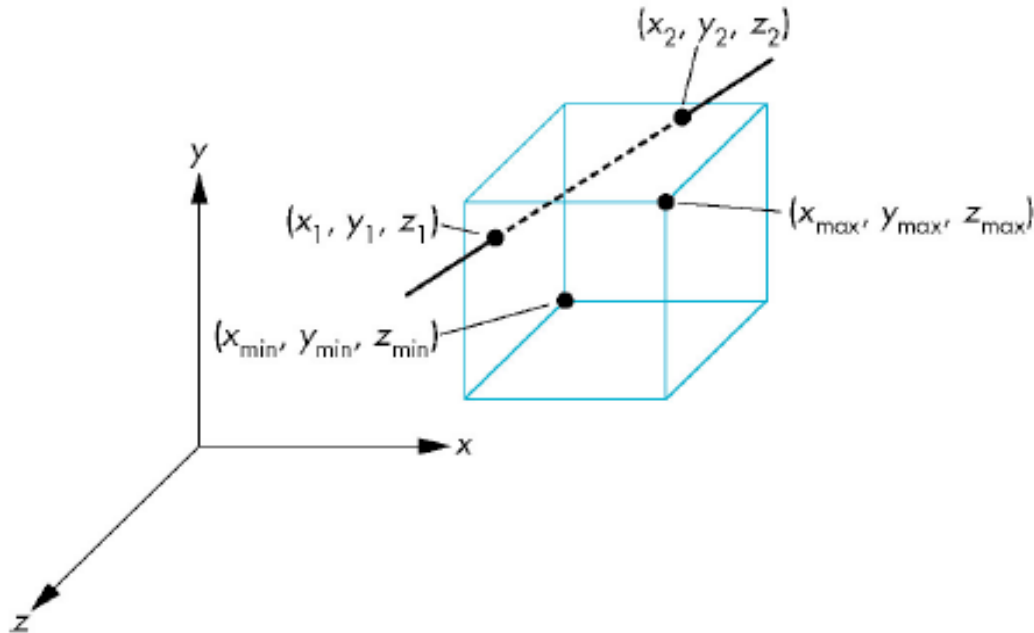
K: 0110

J (AND) K : 0100

Cohen-Sutherland en 3D



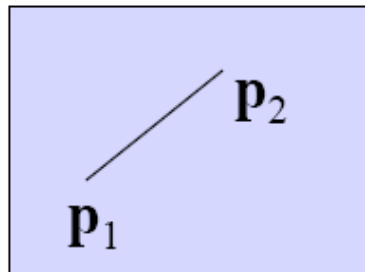
- Se usan codigos de 6 bits





- Sea la forma paramétrica de un segmento de recta

$$\mathbf{p}(\alpha) = (1-\alpha)\mathbf{p}_1 + \alpha\mathbf{p}_2 \quad 1 \geq \alpha \geq 0$$



Liang-Barsky

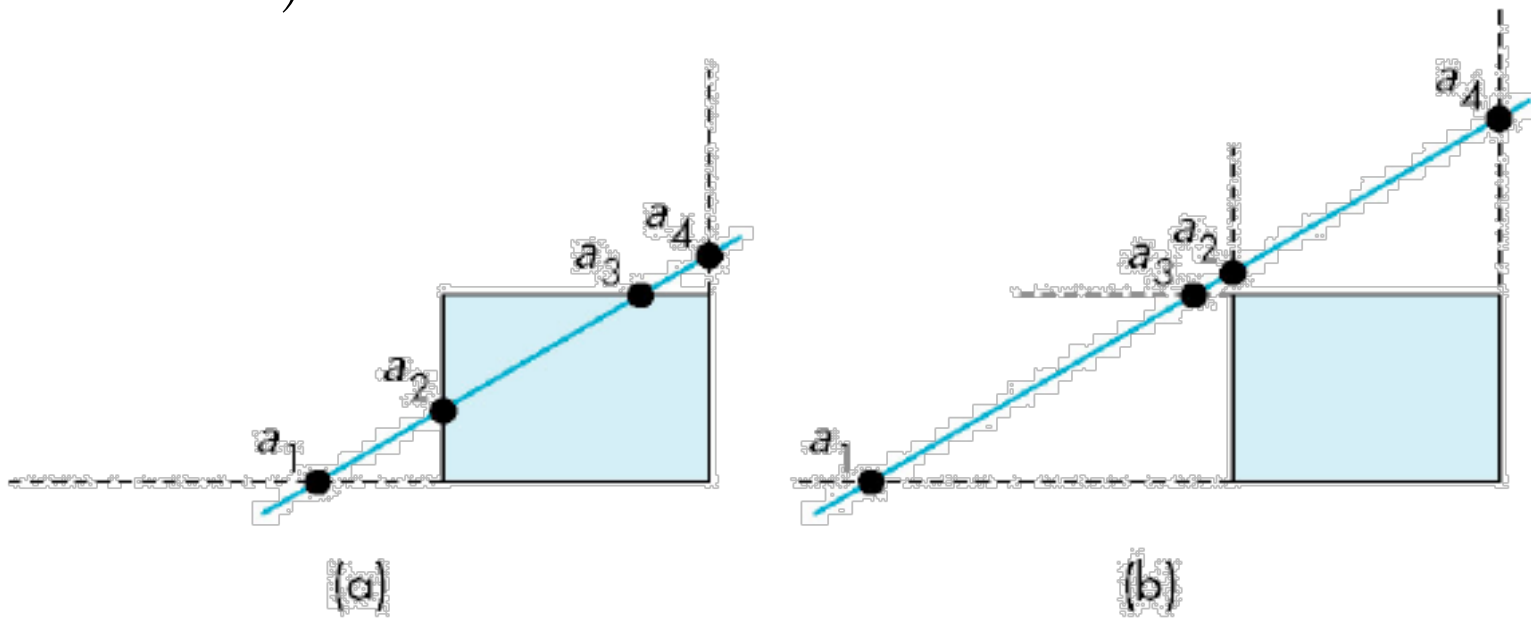


- En (a): $\alpha_4 > \alpha_3 > \alpha_2 > \alpha_1$

Intersecta el lado inferior, el derecho, el tope, el izquierdo (no se puede rechazar)

- En (b): $\alpha_4 > \alpha_2 > \alpha_3 > \alpha_1$

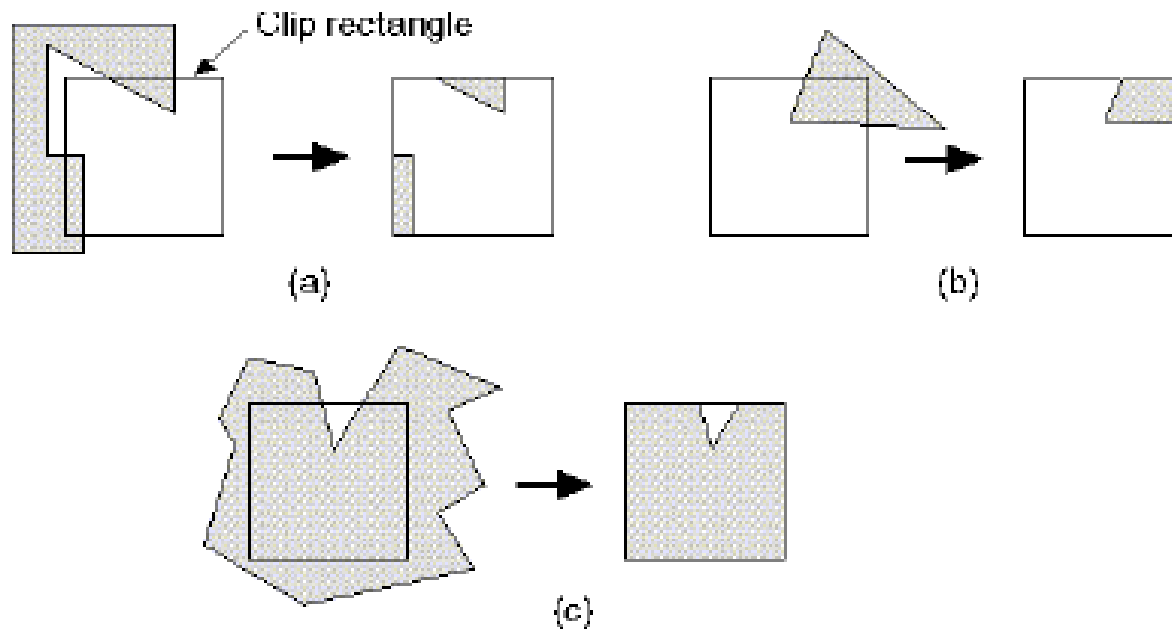
Intersecta el lado inferior, el tope, el derecho y el izquierdo (se rechaza)



Recorte de polígonos



- ❑ Puede ser bastante complejo
 - ❑ Múltiples componentes
 - ❑ concavo con muchos bordes exteriores
 - ❑ Convexo simple



Recorte de polígonos



- ❑ Se deben probar cada arista contra cada borde del área de recorte
 - ❑ Se pueden agregar nuevas aristas
 - ❑ Aristas existentes pueden ser eliminadas, retenidas o divididas
- ❑ Pueden resultar en uno o más polígonos



- ❑ Se usa la estrategia de divide y conquistarás
- ❑ Recortar el polígono contra una línea borde del área de recorte
- ❑ Sucesivamente se va recortando el polígono contra cada línea borde del área de recorte
- ❑ El algoritmo es más general y puede ser aplicado para recortar cualquier polígono contra cualquier área convexa de recorte

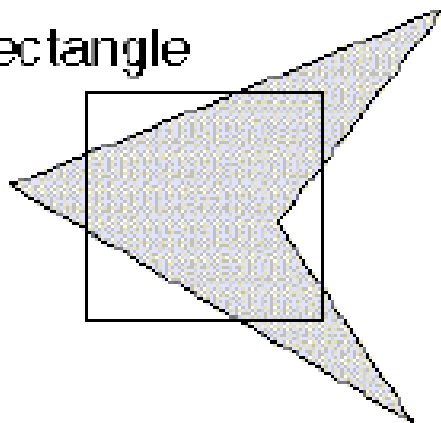


- ❑ Entrada: una lista de vertices que definen las líneas de de las aristas del polígono
- ❑ Salida: Despues de cada pasada por cada arista del poígono, retorna una lista de vertices definiendo un polígono parcialmente recortado

Sutherland-Hodgman

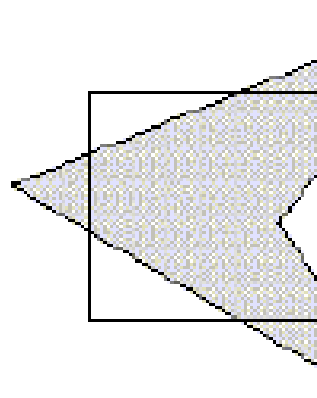


Clip rectangle



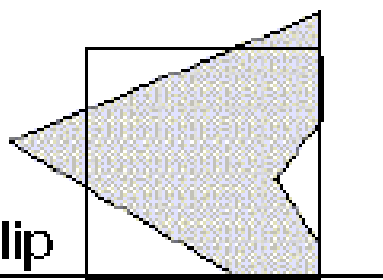
(a)

Right clip boundary



(b)

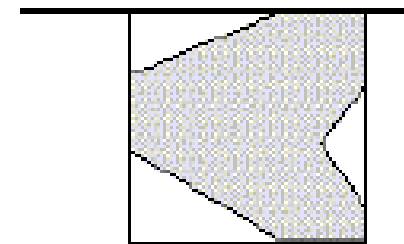
Bottom clip boundary



(c)



(d)

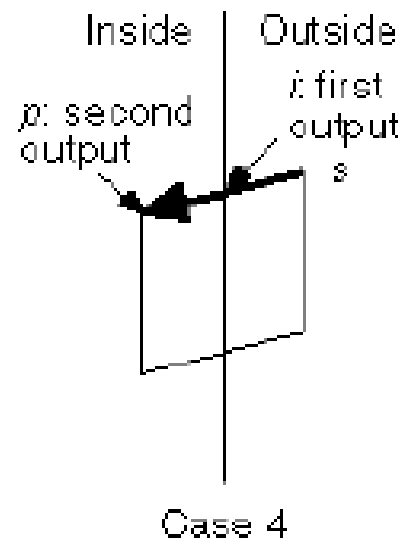
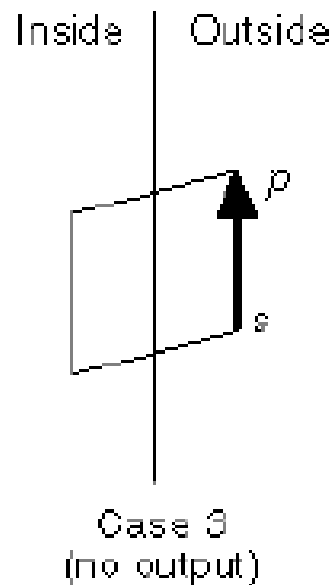
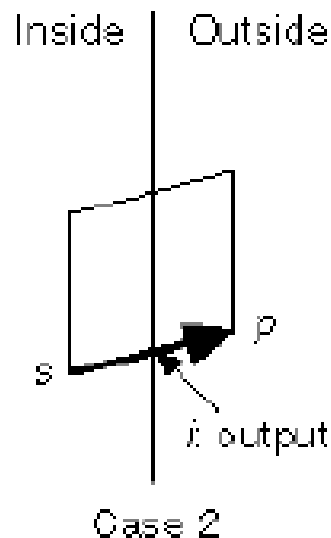
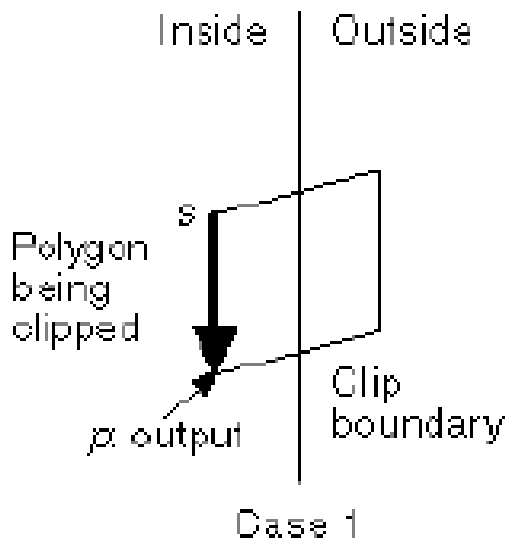


(e)

Sutherland-Hodgman



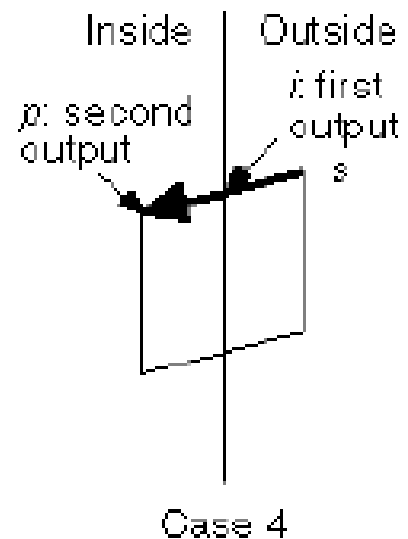
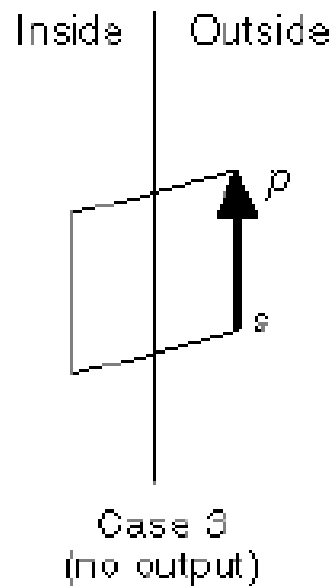
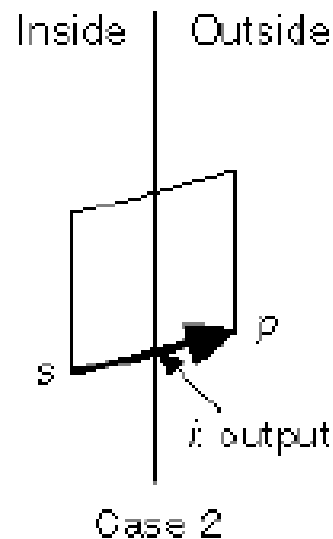
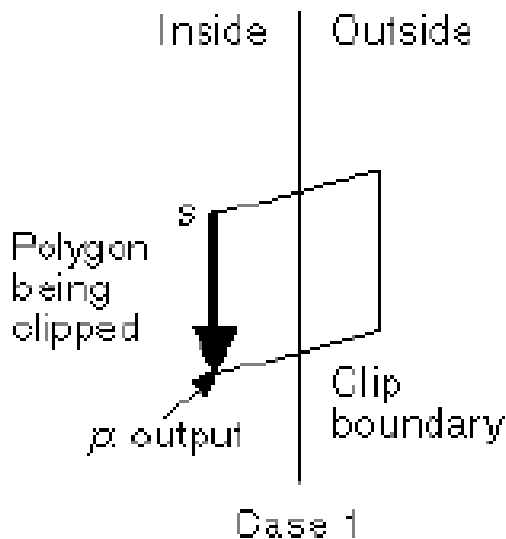
- Examina sucesivos pares de vertices etiquetados como s y p (por ejemplo)
- 0, 1, o 2 vertices se suman a la salida de cada paso, de acuerdo a 4 casos



Sutherland-Hodgman



- El algoritmo puede agregar vertices que luego pueden ser eliminados en la fase de post-procesamiento



Recorte de Caracteres



- Se define una curva de orden alto (cubico): se recorta la curva
- Se define como un bitmap: se van eliminando punto por punto o pixel por pixel
- Más rápido: se dibuja el caracter completo o no