

# Đồ họa máy tính

## Các thuật toán cắt xén (Clipping)

# Khung nhìn trong 2D

- Trong 2D, thế giới được định nghĩa là một mặt phẳng vô hạn, trong một hệ tọa độ nhất định.
- Chúng ta cần lấy ra một vùng trong mặt phẳng 2D này để xem, thường được gọi là ‘cửa sổ’.
- Trong thiết bị hiển thị của chúng ta, cần phải xác định một vùng để hiển thị, thường được gọi là ‘*viewport*’, và sử dụng hệ tọa độ của thiết bị.
  - Cắt bỏ tất cả những vật thể nằm ngoài cửa sổ.
  - Tịnh tiến cho khớp với viewport.
  - Co giãn theo hệ tọa độ của thiết bị.

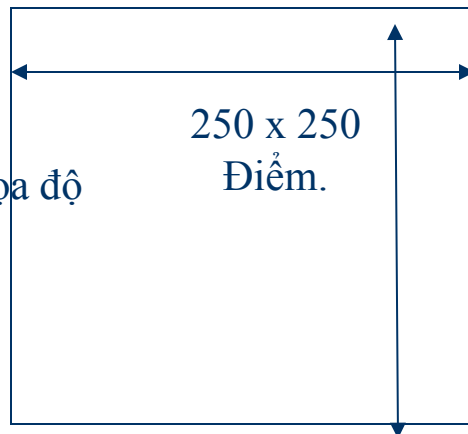
# Khung nhìn trong 2D

250°



45°

Cửa sổ trong tọa độ thế giới.

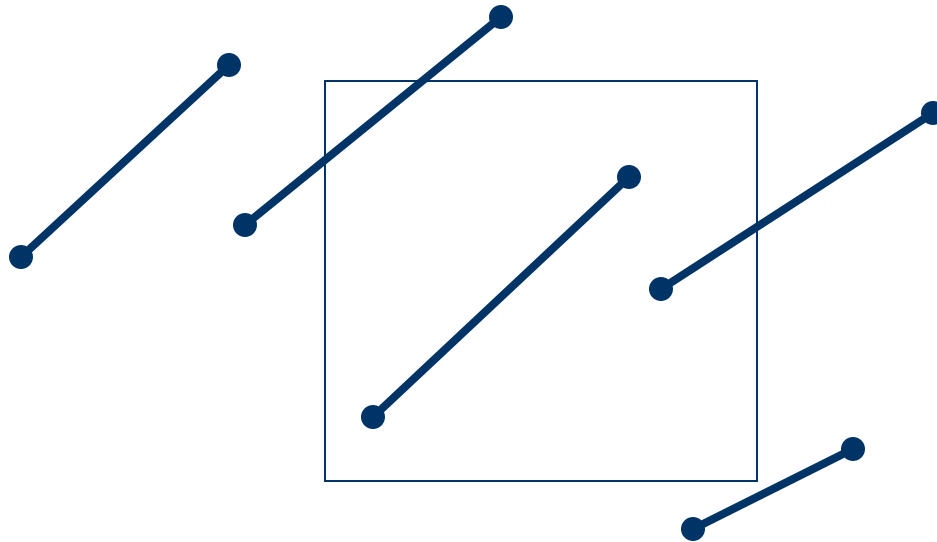


Viewport trong tọa độ  
thiết bị



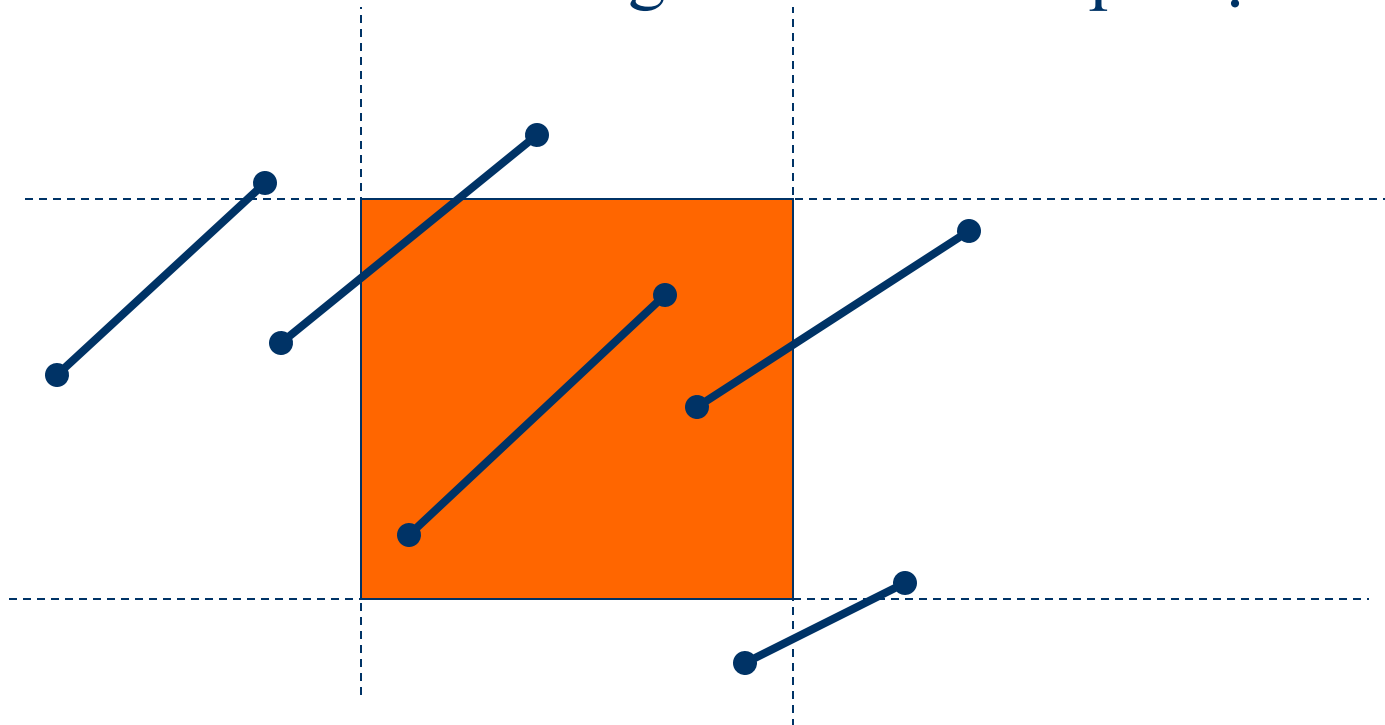
# Clipping trong 2D.

- Cần phải cắt những đối tượng cơ bản theo các cạnh của cửa sổ.
  - v.d. các đoạn thẳng



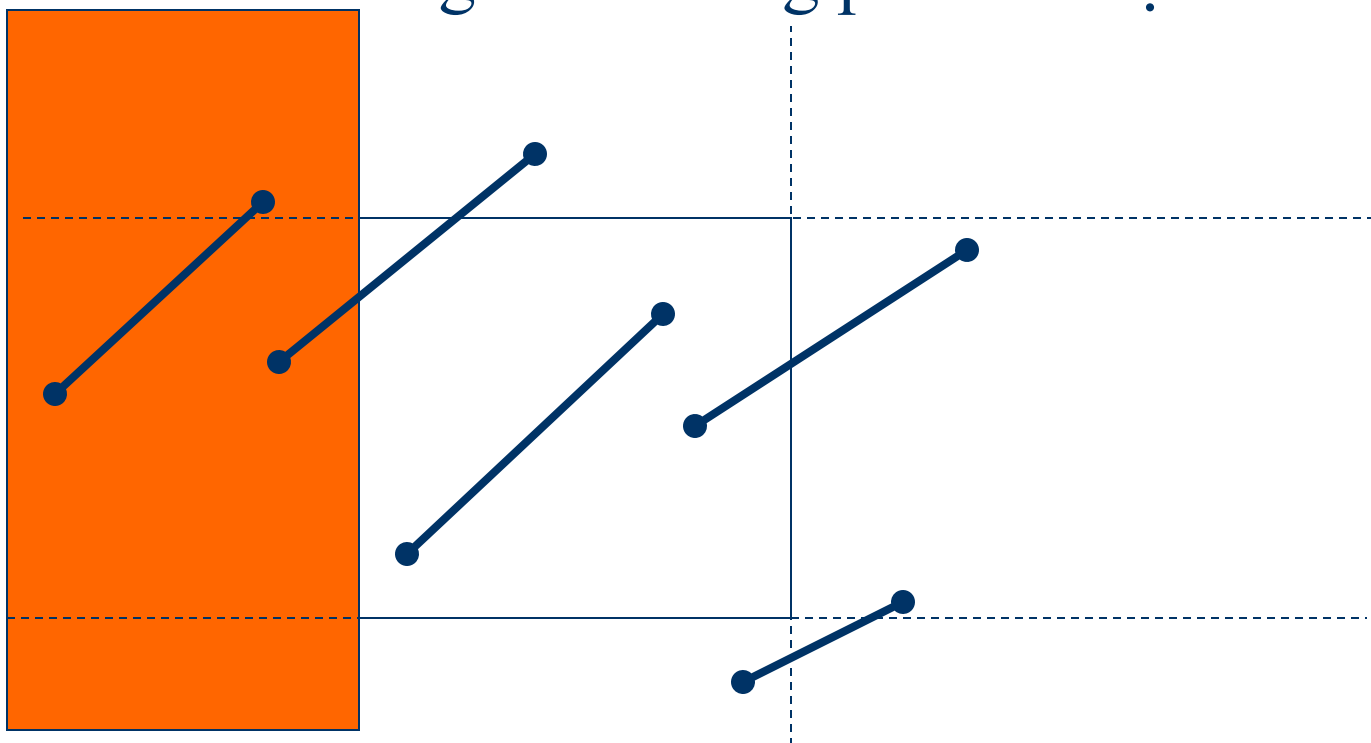
# Chấp nhận đơn giản

Hai đầu mút nằm trong cửa số  $\rightarrow$  chấp nhận.



# Loại bỏ đơn giản

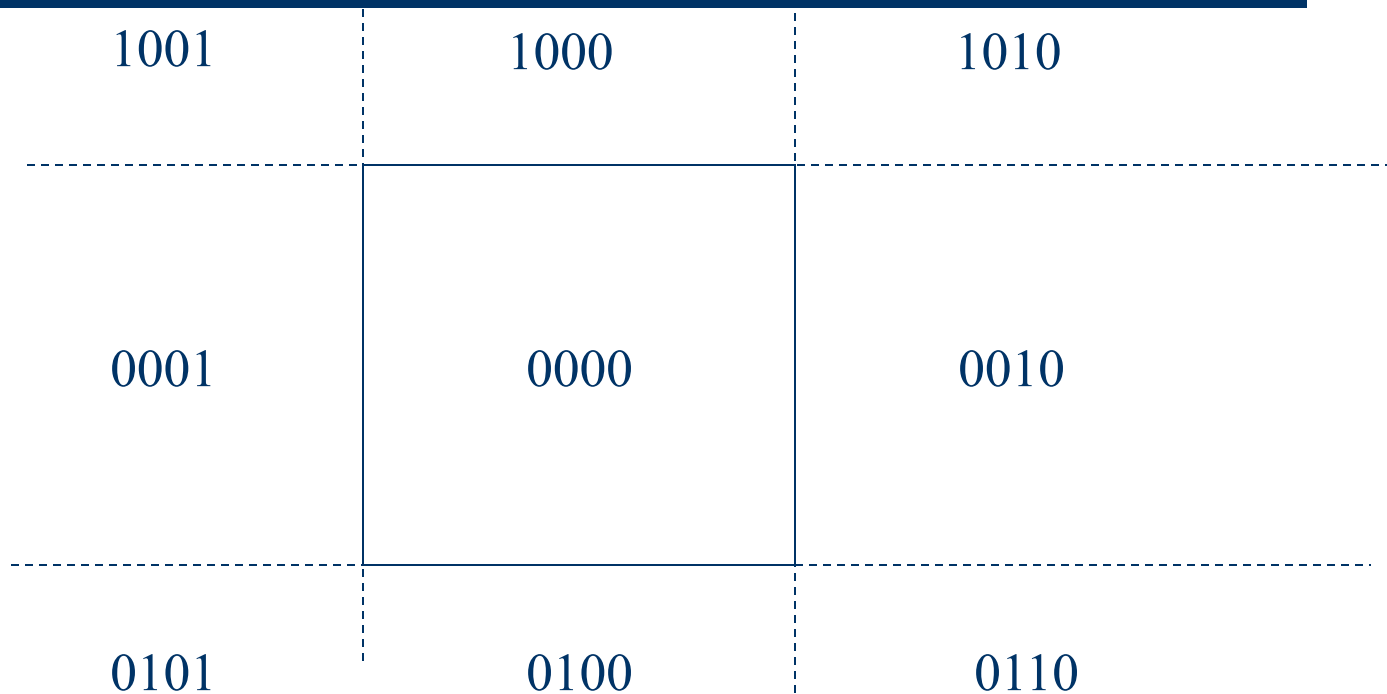
Hai đầu mút nằm ngoài và cùng phía  $\rightarrow$  loại bỏ.



# Thuật toán Cohen-Sutherland

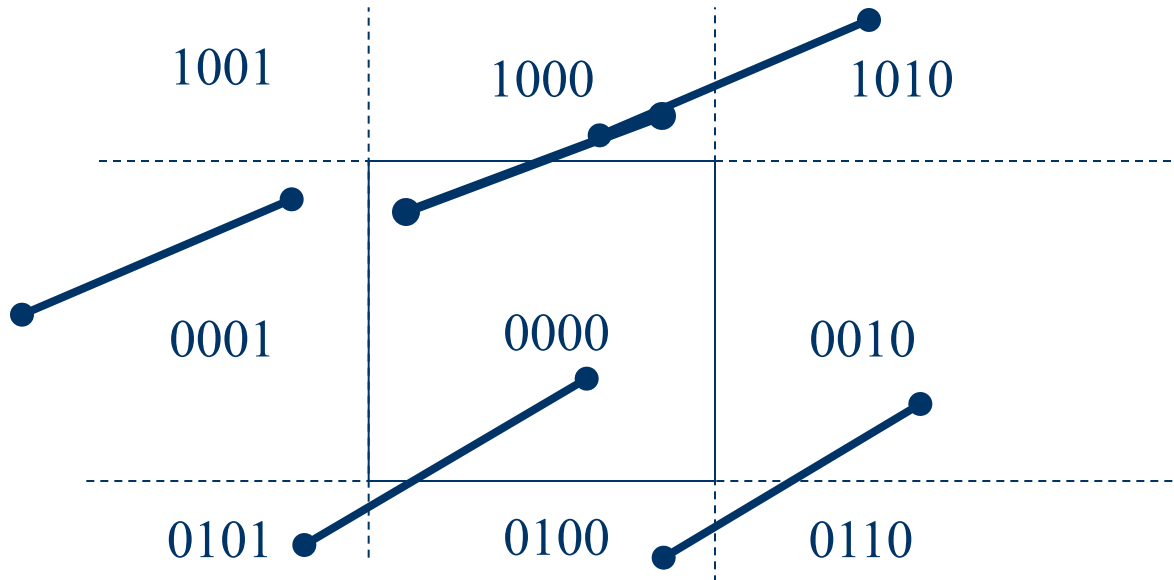
- Phương pháp hiệu quả để chấp nhận hoặc loại bỏ những đoạn thẳng không cắt các cạnh của cửa sổ.
- Gán mã 4 bit cho mỗi đầu mút:  $c(P) = x_3x_2x_1x_0$ 
  - Bit 1: ở trên đỉnh của cửa sổ,  $y > y_{max}$
  - Bit 2: ở phía dưới đáy,  $y < y_{min}$
  - Bit 3 : bên phải của cạnh phải,  $x > x_{max}$
  - Bit 4 : bên trái của cạnh trái,  $x < x_{min}$
  - Mã 4-bit được gọi là: *Outcode*

# Mã Cohen-Sutherland 2D



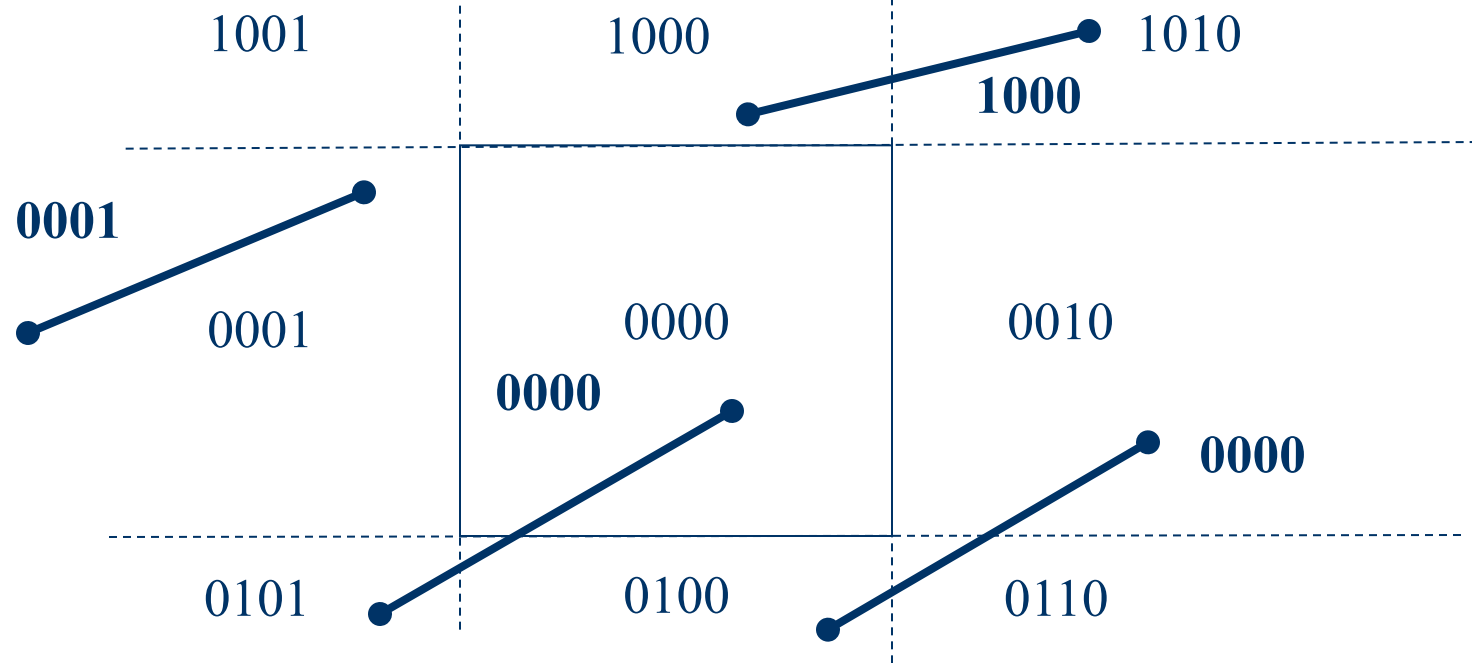


# Thuật toán Cohen-Sutherland



Nếu cả hai đầu có mã là 0000, chấp nhận, nếu không:  
Thực hiện phép AND logic 2 mã

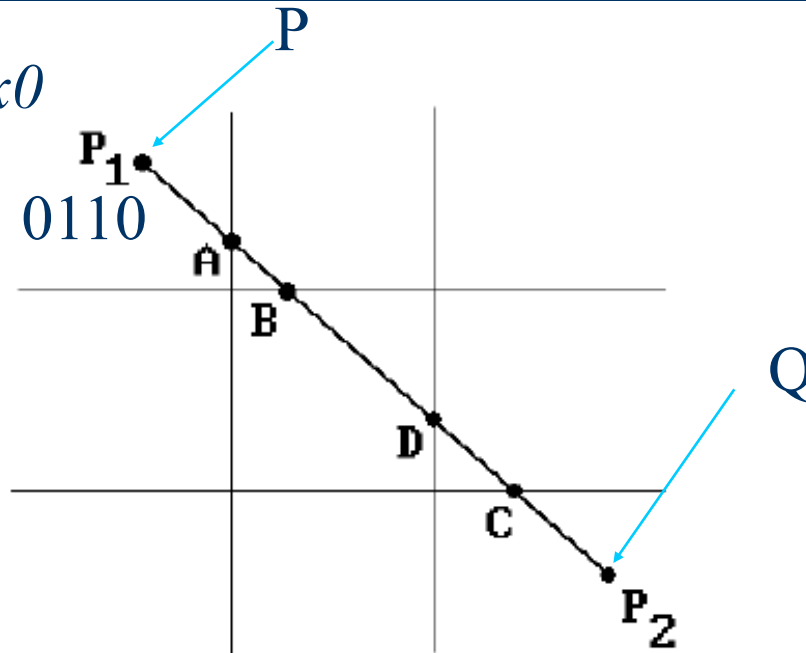
# Thuật toán Cohen-Sutherland



Thực hiện AND logic mã của 2 đầu mút,  
Loại bỏ đoạn thẳng nếu khác không.

# Thuật toán Cohen-Sutherland

$$c(P) = x_3x_2x_1x_0$$

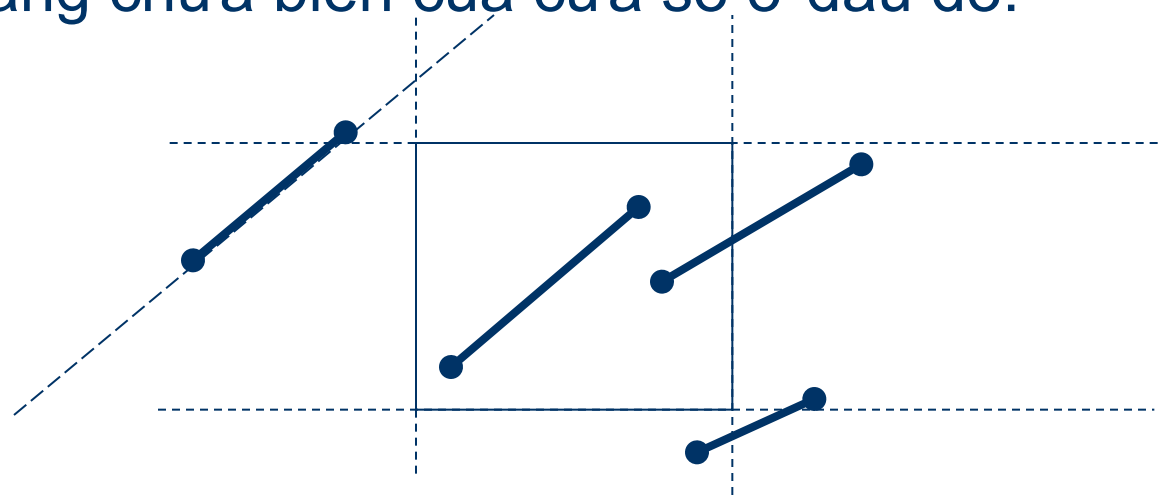


# Giao đoạn thẳng

- Cần xác định giao điểm của các đoạn thẳng với các cạnh của cửa sổ để tiến hành cắt các đoạn thẳng.
- Chọn một cạnh cửa sổ bất kỳ, cắt các đoạn thẳng, thực hiện lại thuật toán Cohen-Sutherland

# Thuật toán Cyrus & Beck

- Sử dụng phương trình tham số
- Đường thẳng chứa đoạn cần xử lý sẽ cắt các đường thẳng chứa biên của cửa sổ ở đâu đó:

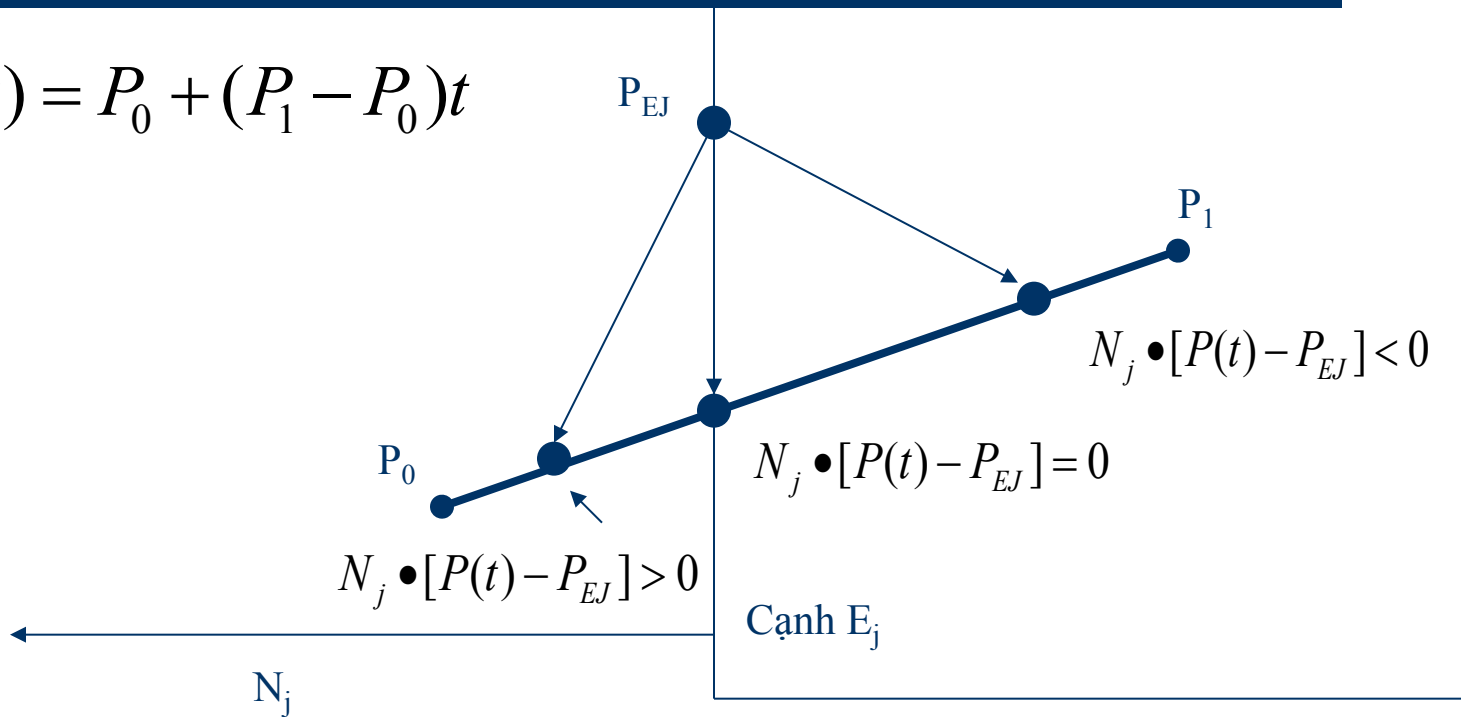


# Thuật toán Cyrus & Beck

- Sử dụng phương trình tham số
- Đường thẳng chứa đoạn cần xử lý sẽ cắt các đường thẳng chứa biên của cửa sổ ở đâu đó:
  - Tìm tất cả các giao điểm, kiểm tra xem nó có nằm trên cửa sổ hay không.
  - Xem xét vectơ pháp tuyến tại một điểm.

# Thuật toán Cyrus & Beck

$$P(t) = P_0 + (P_1 - P_0)t$$



# Thuật toán Cyrus & Beck

$$P(t) = P_0 + (P_1 - P_0)t$$

$$N_j \cdot [P(t) - P_{EJ}] = 0$$

$$N_j \cdot [P_0 + (P_1 - P_0)t - P_{EJ}] = 0$$

$$N_j \cdot [P_0 - P_{EJ}] + N_j \cdot [P_1 - P_0]t = 0$$

$$D = (P_1 - P_0), \text{ tính } t$$

$$t = \frac{N_j \cdot [P_0 - P_{EJ}]}{-N_j \cdot D}$$

Dấu của mẫu số là quan trọng.

Phải  $\neq 0$  (bỏ qua những đoạn song song).

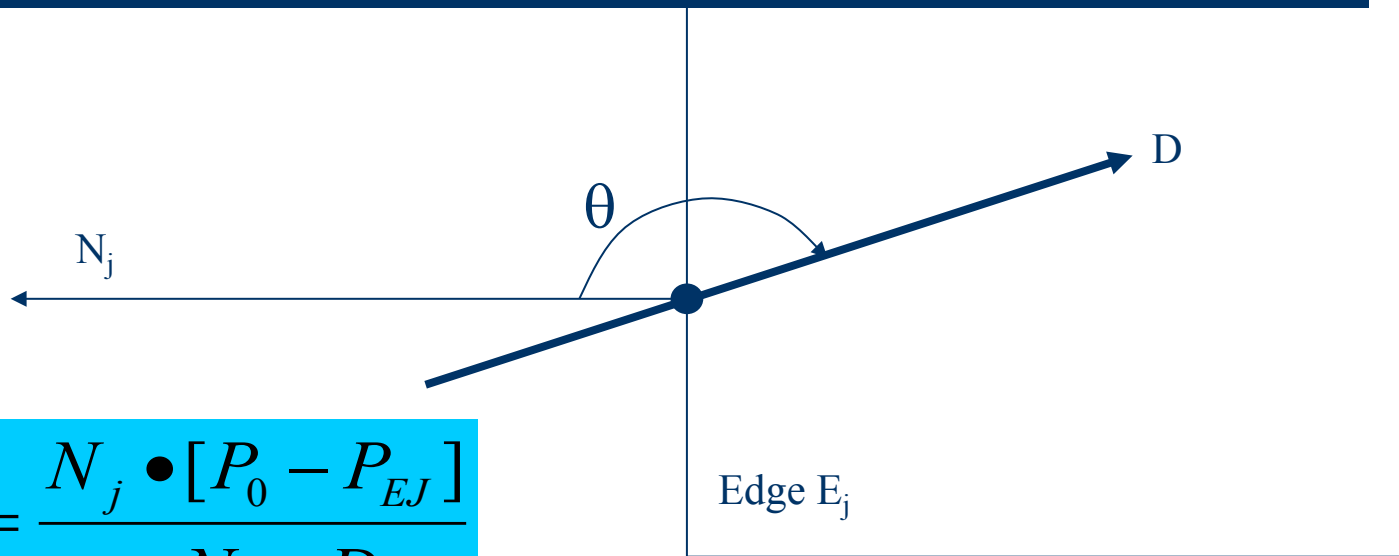
Phương trình tham số giúp thể hiện hướng.

Mẫu số  $< 0 \rightarrow$  điếm vào khu vực cửa sổ.

Mẫu số  $> 0 \rightarrow$  điếm ra khỏi khu vực cửa sổ.



# Thuật toán Cyrus & Beck



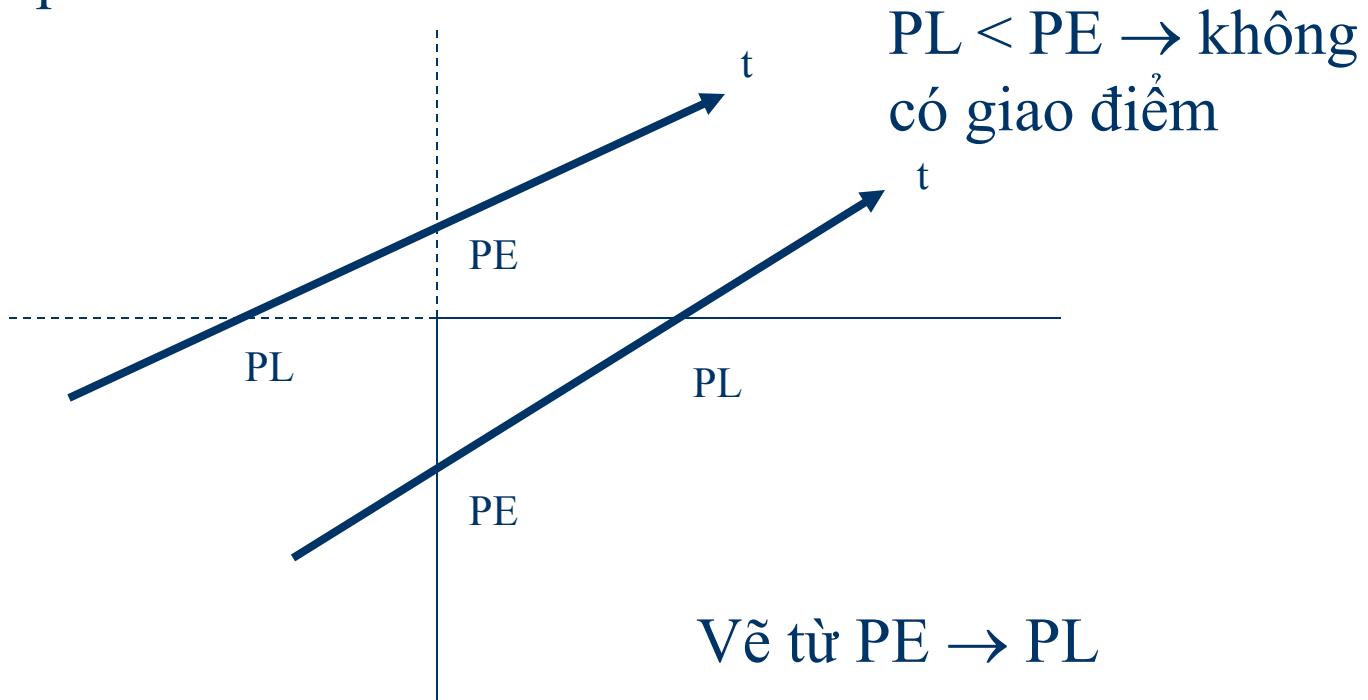
$$t = \frac{N_j \cdot [P_0 - P_{EJ}]}{-N_j \cdot D}$$

Mẫu số  $< 0 \rightarrow$  điểm vào khu vực cửa sổ, xếp vào loại PE.

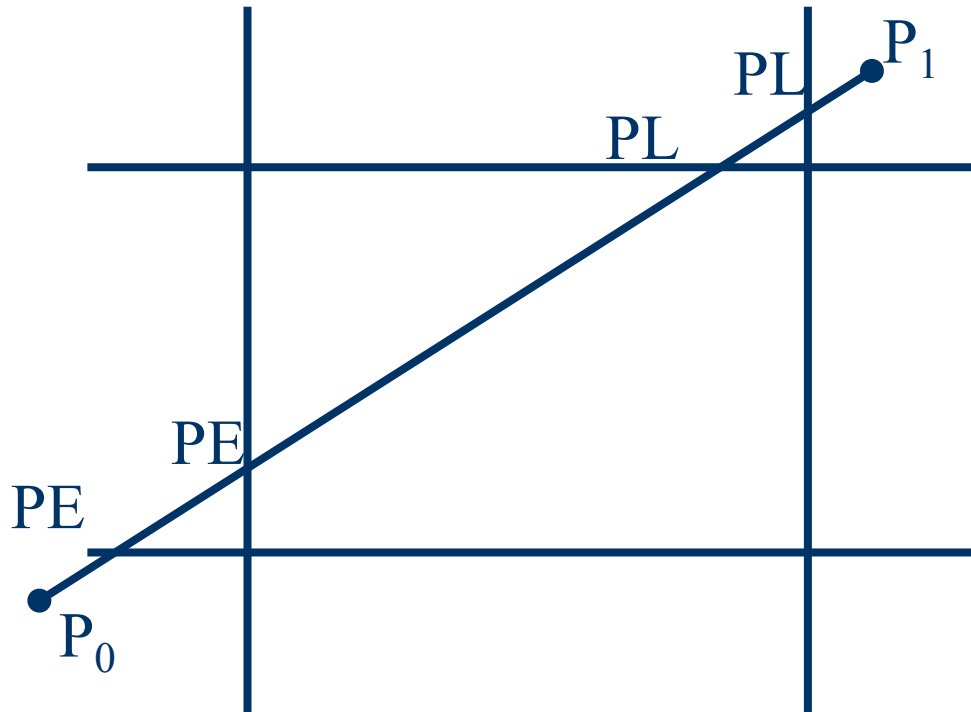
Mẫu số  $> 0 \rightarrow$  điểm ra khỏi khu vực cửa sổ, xếp vào loại PL.

# Thuật toán Cyrus & Beck

Sắp xếp các điểm PE và PL theo t.

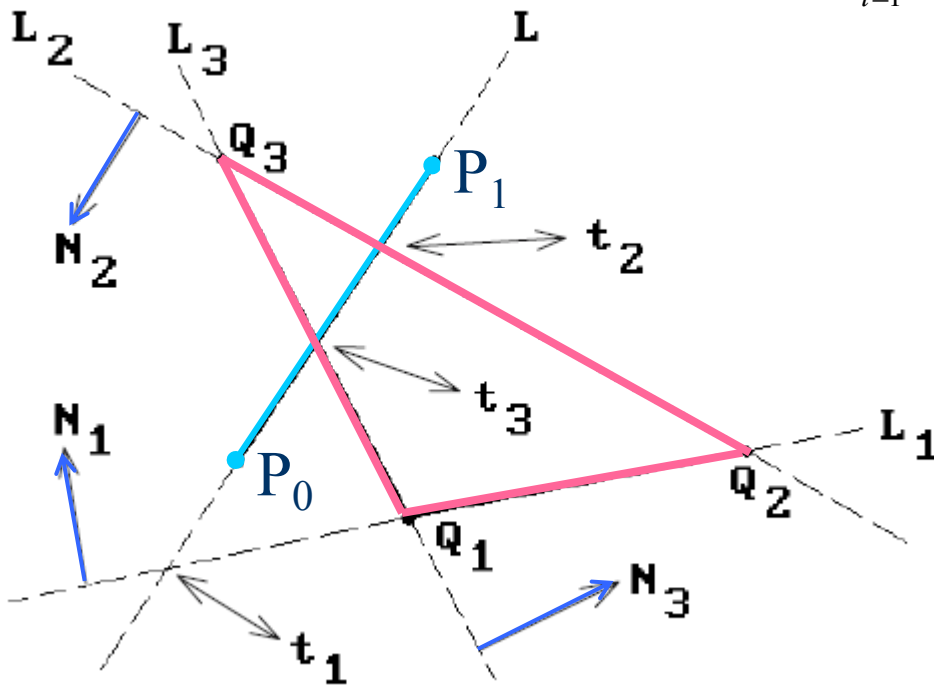


# Thuật toán Cyrus & Beck



# Thuật toán Cyrus & Beck

$$H_i = \{Q \mid N_i \cdot (Q - Q_i) \geq 0\} \longrightarrow X = \bigcap_{i=1}^k H_i$$



## •L song song $L_i$ :

✓L nằm trong  $H_i$ :  $I_i = (-\infty, +\infty)$

✓L nằm ngoài  $H_i$ :  $I_i = \emptyset$

## •L không song song $L_i$ :

✓Đi vào:  $li = [ti, +\infty)$

✓Đi ra:  $li = (-\infty, ti]$

## •Đặt $I_0 = [0, 1]$

$$I = \bigcap_{i=0}^k I_i$$

# Thuật toán Liang - Barsky

Phương trình tham số của đường thẳng nối  $(x_1, y_1)$  và  $(x_2, y_2)$

$$\begin{cases} x = x_1 + t * \Delta x \\ y = y_1 + t * \Delta y \end{cases}$$

Với

$$\Delta x = x_2 - x_1$$

$$\Delta y = y_2 - y_1$$

# Thuật toán Liang - Barsky

Điểm  $P$  thuộc về cửa sổ  $W$  khi và chỉ khi:

$$x_{\min} \leq x_1 + \Delta x t \leq x_{\max}$$

$$y_{\min} \leq y_1 + \Delta y t \leq y_{\max}$$

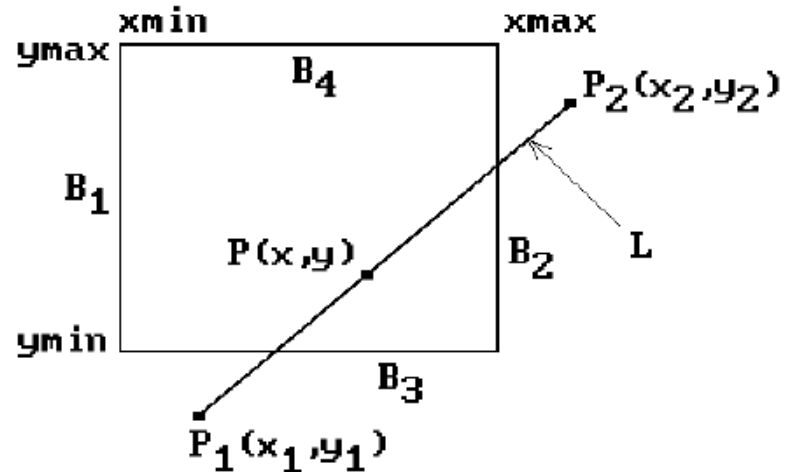
hay

$$-\Delta x t \leq x_1 - x_{\min}$$

$$\Delta x t \leq x_{\max} - x_1$$

$$-\Delta y t \leq y_1 - y_{\min}$$

$$\Delta y t \leq y_{\max} - y_1$$



# Thuật toán Liang - Barsky

$$-\Delta x t \leq x_1 - x_{\min}$$

$$\Delta x t \leq x_{\max} - x_1$$

$$-\Delta y t \leq y_1 - y_{\min}$$

$$\Delta y t \leq y_{\max} - y_1$$

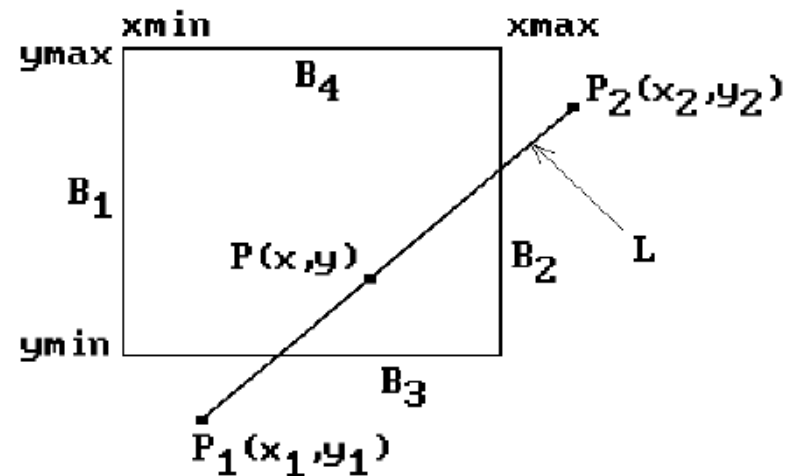
Đặt các biến phụ ci, qi

$$c_1 t \leq q_1 \quad c_3 t \leq q_3$$

$$c_2 t \leq q_2 \quad c_4 t \leq q_4$$

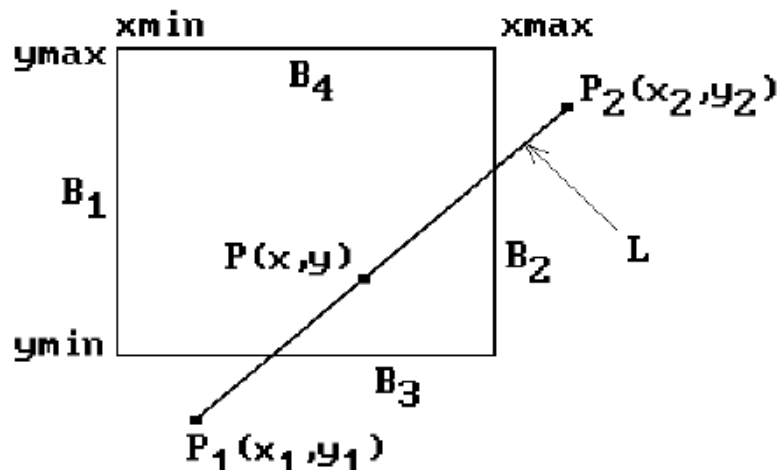


$$t_k = q_k / c_k$$



# Thuật toán Liang - Barsky

$$t_k = q_k / c_k$$



- (1)  $ck > 0$ , đt  $L$  đi từ phía trong ra phía ngoài của đường biên  $B_k$  khi  $t$  tăng, và chúng ta gọi  $tk$  là điểm ra.
- (2)  $ck < 0$ , đt  $L$  đi từ phía ngoài vào phía trong của đường biên  $B_k$  khi  $t$  tăng và ta gọi  $tk$  là điểm vào.
- (3)  $ck = 0$ , đt  $L$  song song với  $B_k$ , và ngoài cửa sổ nếu  $qk < 0$



# Thuật toán Liang - Barsky

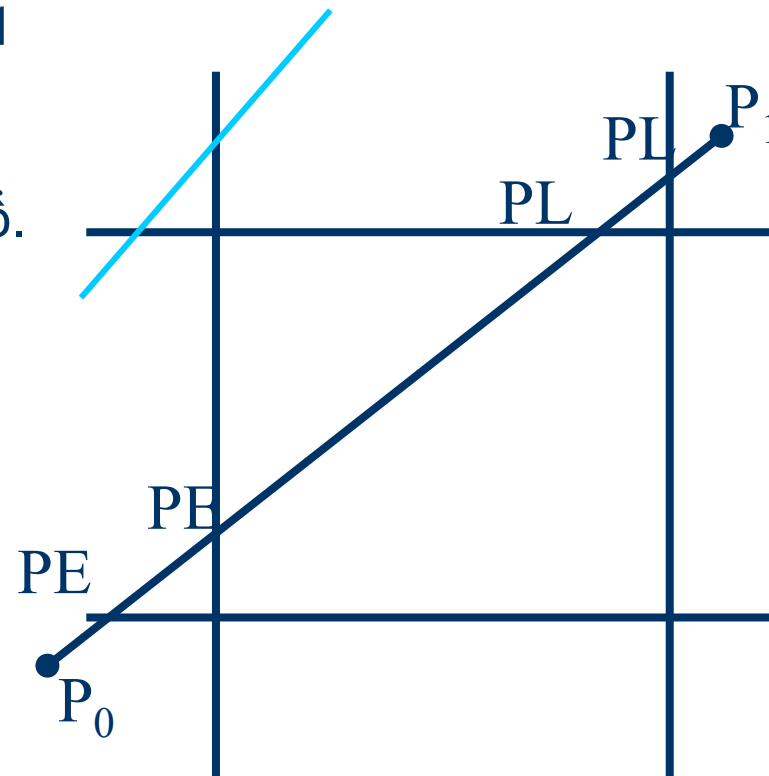
Loại bỏ đoạn thẳng nếu:

- Một giá trị vào ( $t$  ứng với điểm vào)  $> 1$
- Hoặc giá trị ra ( $t$  ứng với điểm ra)  $< 0$
- Hoặc một giá trị vào  $>$  hơn giá trị ra

Nếu không đoạn thẳng sẽ giao với cửa sổ.

Đoạn giao chỉ khi  $t_0 > 0$  và  $t_1 < 1$

- $t_0 = \max(0, \max(\text{các giá trị vào } t_k))$
- $t_1 = \min(1, \min(\text{các giá trị ra } t_k))$



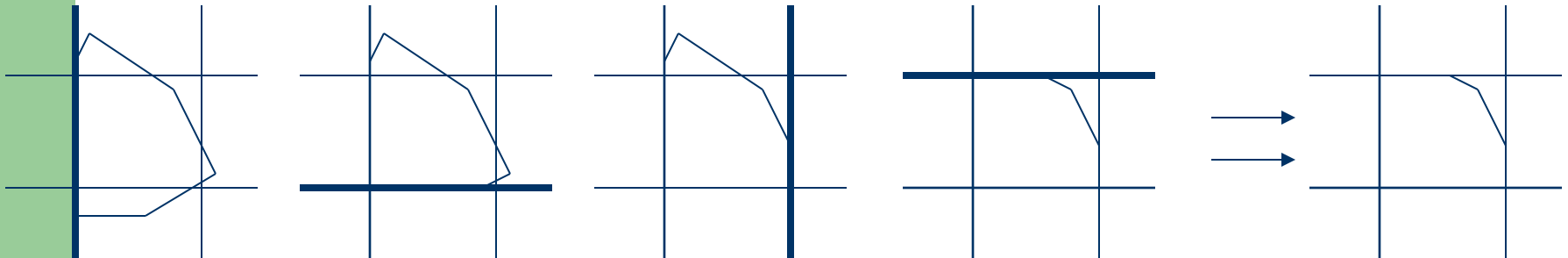
# Thuật toán Liang - Barsky

Cài đặt các thuật toán cắt xén đoạn thẳng

- Cohen Sutherland
- Cyrus - Beck
- Liang – Barsky

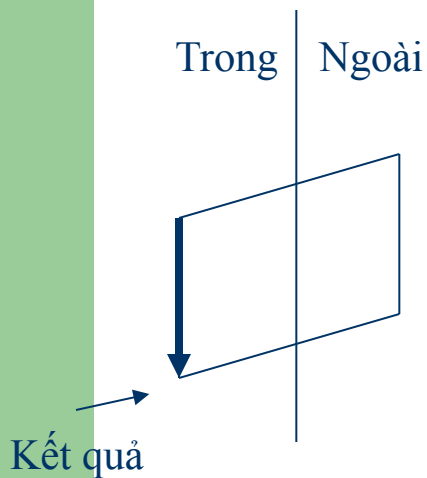
# Clipping đa giác

- Cắt đa giác bằng cách lần lượt sử dụng các cạnh của cửa sổ để cắt đa giác.

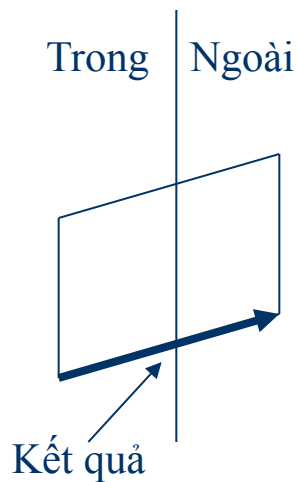


# Thuật toán Sutherland-Hodgman

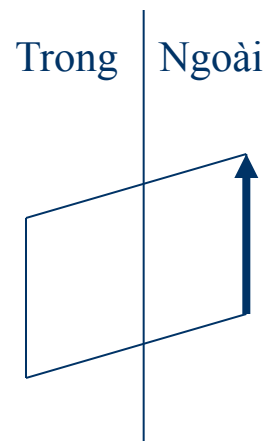
Bốn trường hợp cắt đa giác:



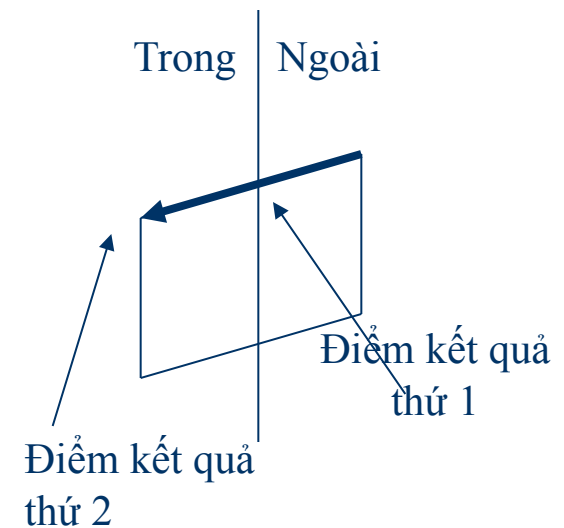
Trường hợp 1



Trường hợp 2.



Trường  
hợp 3  
Không  
có điểm  
ra nào.



Trường hợp 4

# Thuật toán Sutherland-Hodgman

- Đi vòng quanh các điểm của đa giác, kiểm tra với cạnh đang dùng để cắt của cửa sổ.
- Chạy thuật toán lại với đa giác mới vừa được tạo ra với cạnh tiếp theo của cửa sổ
- Không cần lưu trữ nhiều
  - Dễ dàng cài đặt.

# Clipping 3D

- Sử dụng thuật toán Cohen-Sutherland
  - Mã 6-bit.
  - Chấp nhận đơn giản khi cả mã của cả hai đầu mút là 0.
  - Thực hiện phép AND logic, loại bỏ nếu khác 0.
  - Tìm phần giao với một mặt phẳng của khối nhìn và thêm hai đoạn thẳng mới vào để xử lý lại.

# Clipping đa giác 3D

- Mở rộng thuật toán Sutherland-Hodgman cho 3 chiều.
- Cắt 6 lần thay vì 4 lần

