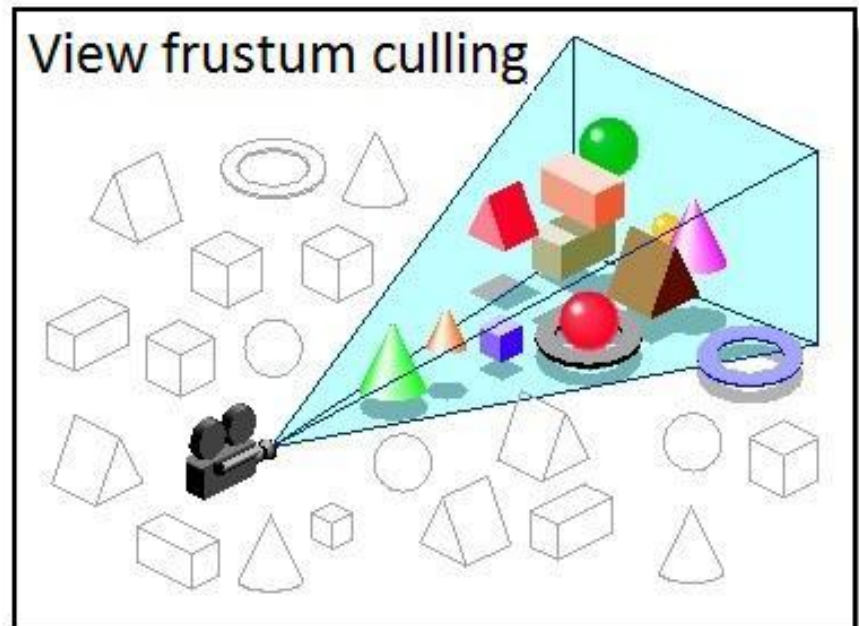
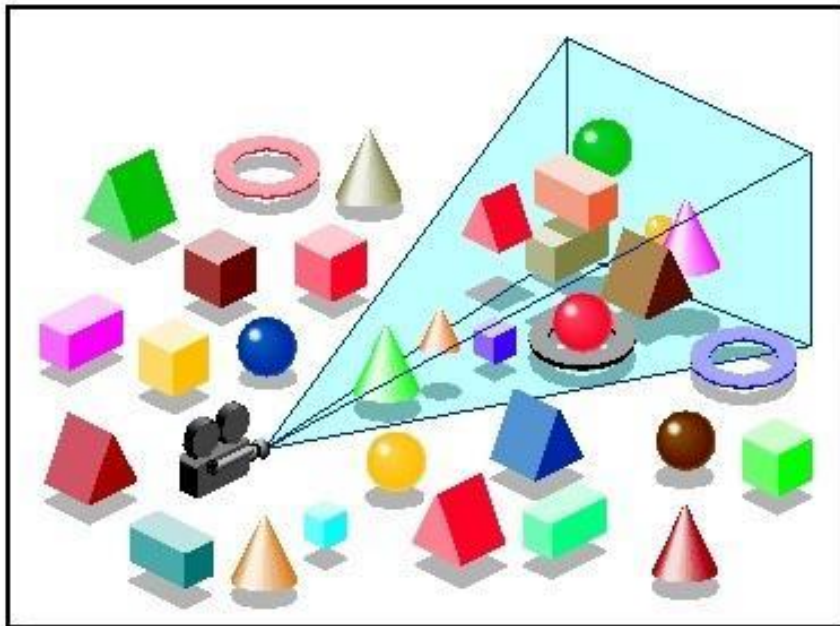




CS113 - ĐỒ HỌA MÁY TÍNH VÀ XỬ LÝ ẢNH

Clipping Polygons



Clipping Polygons

Clipping polygons :

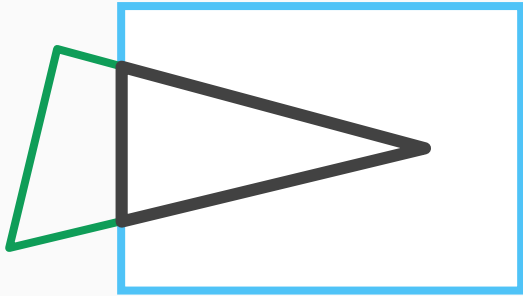
Input: polygon

Output: original polygon, new polygon, or nothing

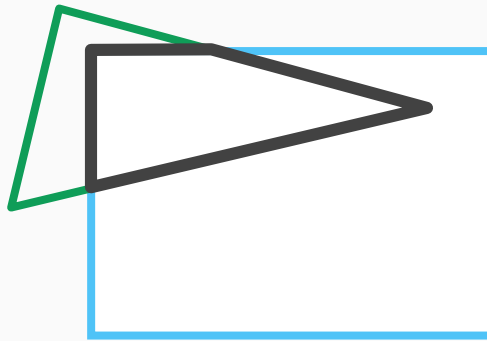
Why Is Clipping Hard?

What happens to a triangle during clipping?

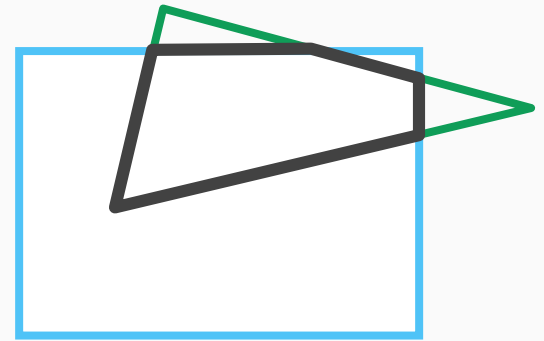
Possible outcomes:



triangle \Rightarrow triangle



triangle \Rightarrow quad

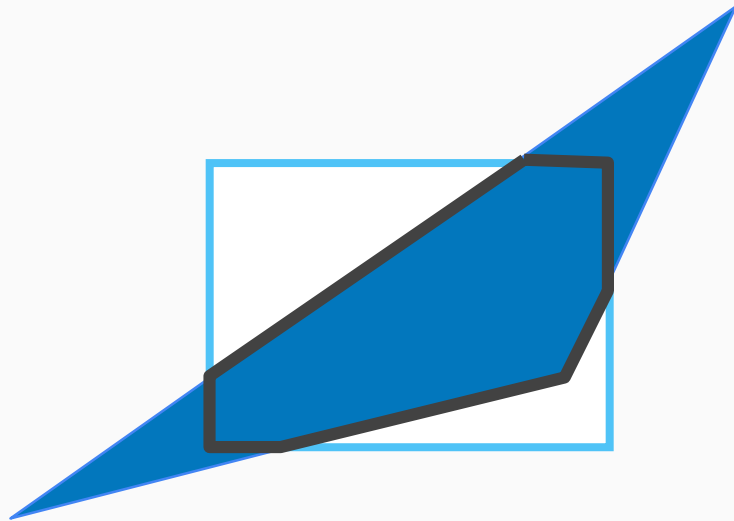


triangle \Rightarrow 5-gon

How many sides can a clipped triangle have?

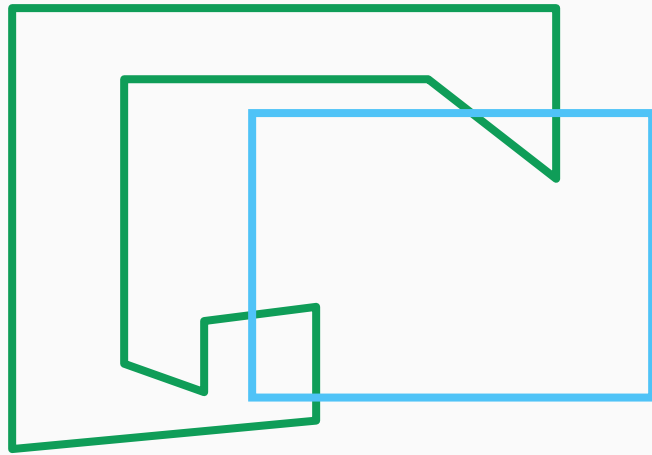
How many sides?

Seven...



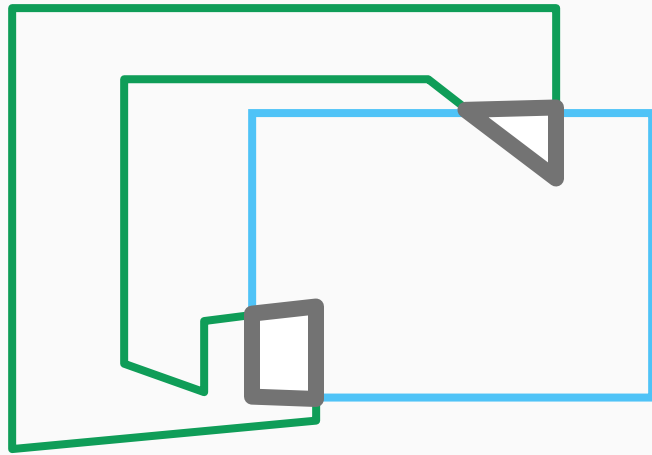
Why Is Clipping Hard?

A really tough case:



Why Is Clipping Hard?

A really tough case:



concave polygon \Rightarrow multiple polygons

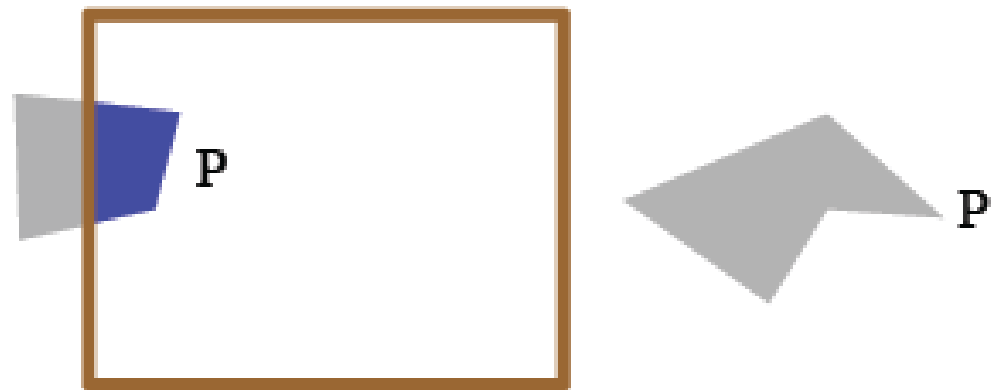
Bài toán

Input

Đa giác P

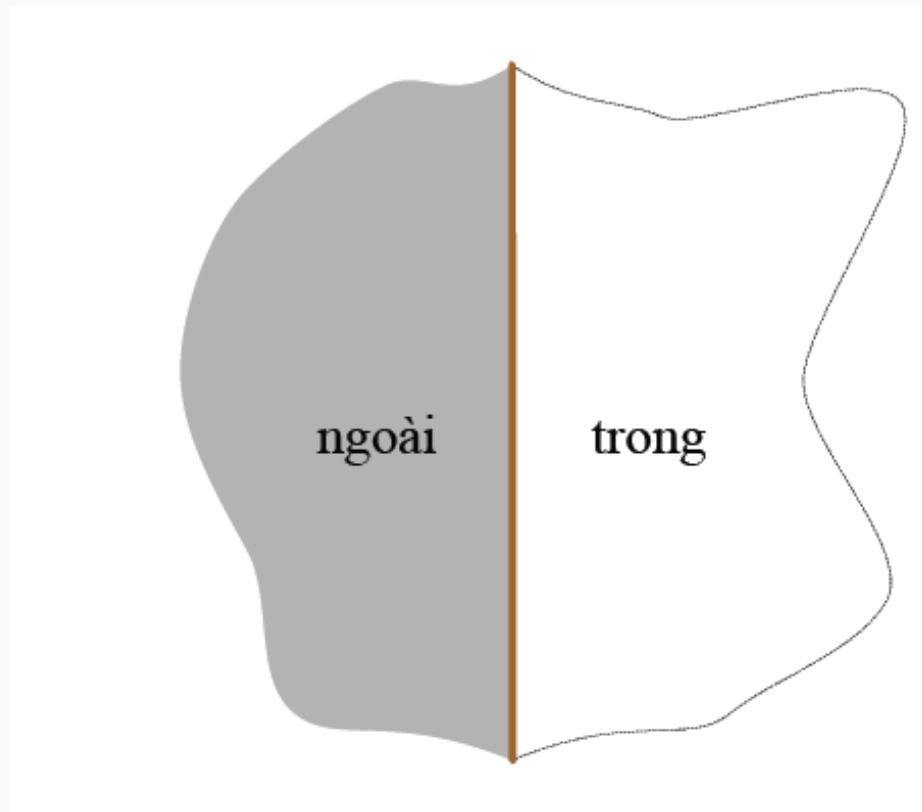
Output

$P \cap W$



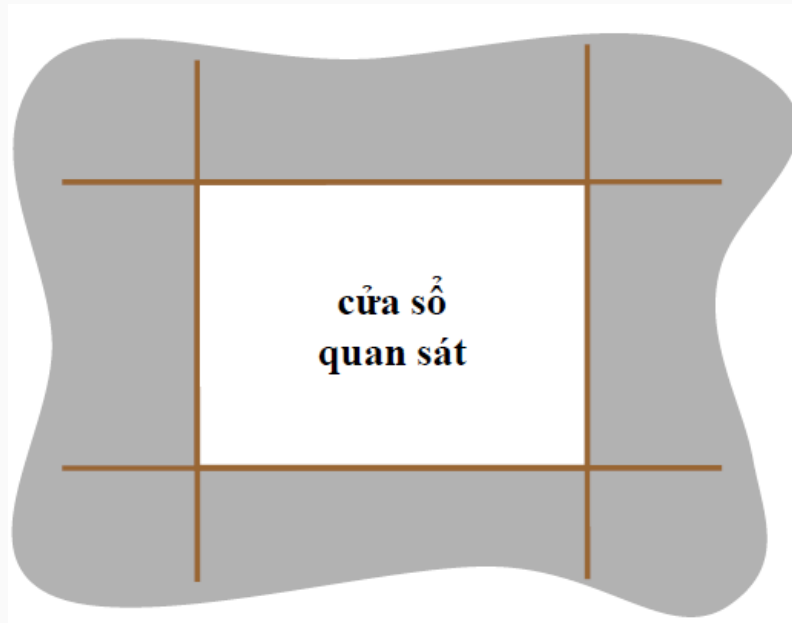
Nửa mặt phẳng trong/ngoài

Mỗi cạnh chia mặt phẳng ra làm hai phần gồm: **nửa mặt phẳng trong** và **nửa mặt phẳng ngoài**.



Nhận xét

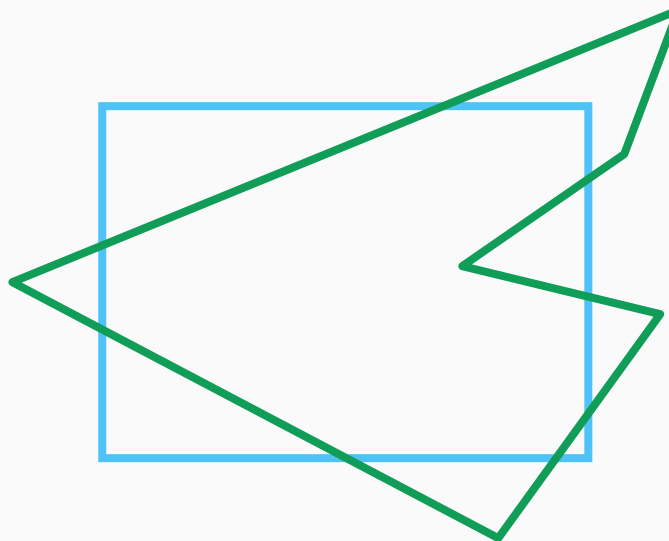
Cửa sổ quan sát là giao của các nửa mặt phẳng trong của các cạnh.



Thuật toán Sutherland-Hodgman

Ý tưởng chính:

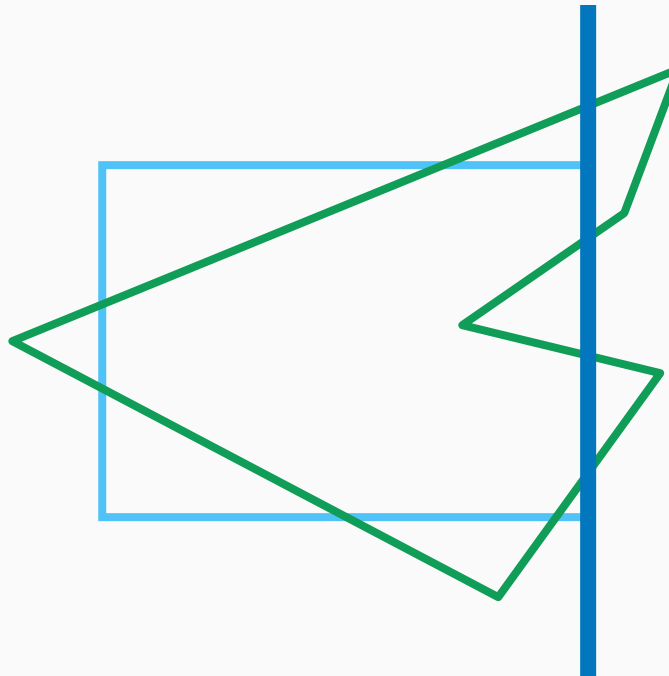
- Dùng từng cạnh của cửa sổ lần lượt xén đa giác.
- Sau khi xén xong, ta được một đa giác đã được xén



Thuật toán Sutherland-Hodgman

Ý tưởng chính:

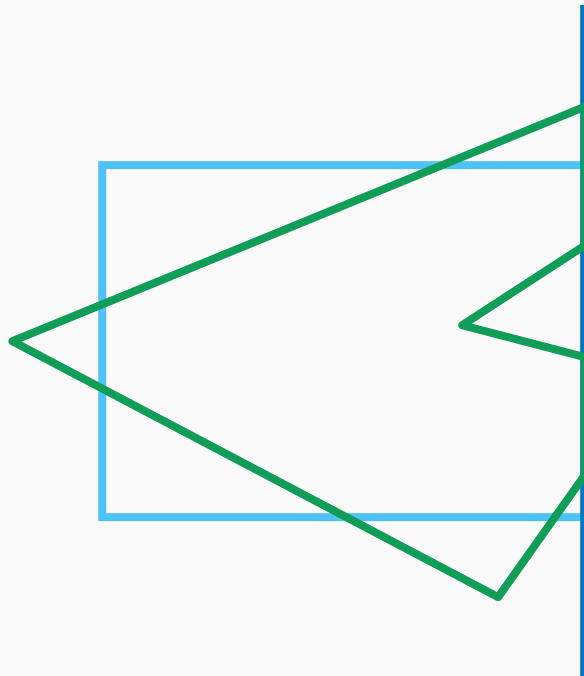
- Dùng từng cạnh của cửa sổ lần lượt xén đa giác.
- Sau khi xén xong, ta được một đa giác đã được xén



Thuật toán Sutherland-Hodgman

Ý tưởng chính:

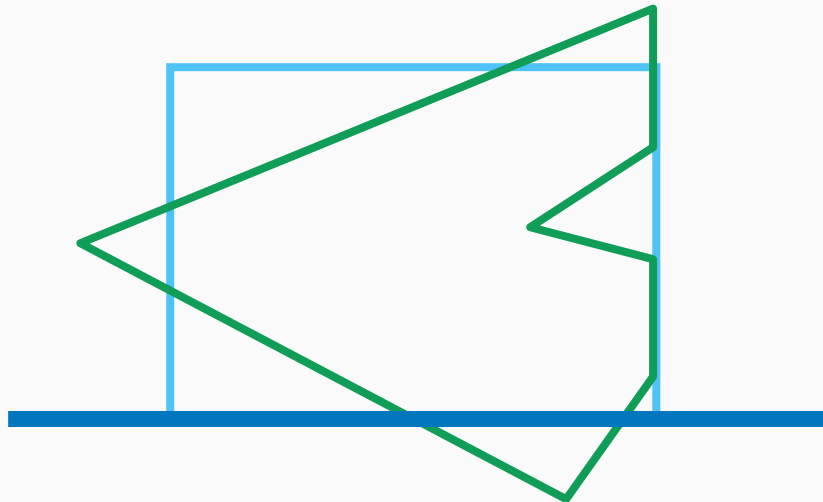
- Dùng từng cạnh của cửa sổ lần lượt xén đa giác.
- Sau khi xén xong, ta được một đa giác đã được xén



Thuật toán Sutherland-Hodgman

Ý tưởng chính:

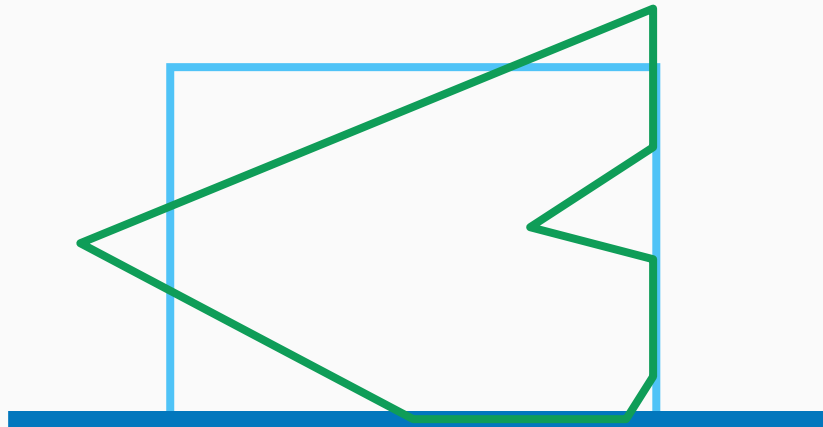
- Dùng từng cạnh của cửa sổ lần lượt xén đa giác.
- Sau khi xén xong, ta được một đa giác đã được xén



Thuật toán Sutherland-Hodgman

Ý tưởng chính:

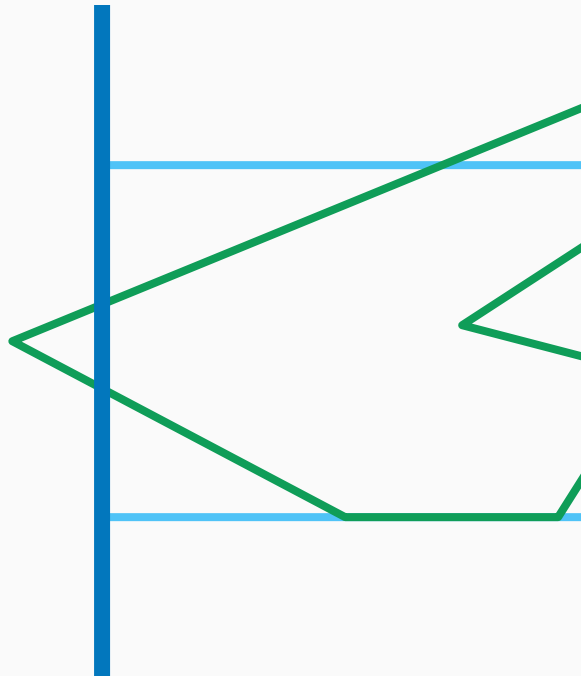
- Dùng từng cạnh của cửa sổ lần lượt xén đa giác.
- Sau khi xén xong, ta được một đa giác đã được xén



Thuật toán Sutherland-Hodgman

Ý tưởng chính:

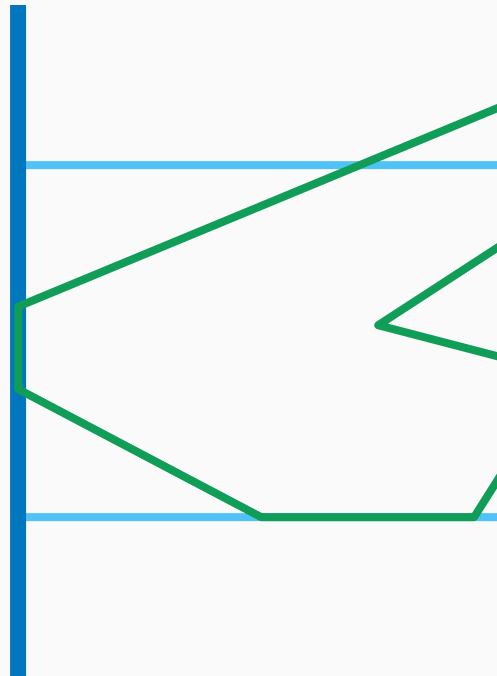
- Dùng từng cạnh của cửa sổ lần lượt xén đa giác.
- Sau khi xén xong, ta được một đa giác đã được xén



Thuật toán Sutherland-Hodgman

Ý tưởng chính:

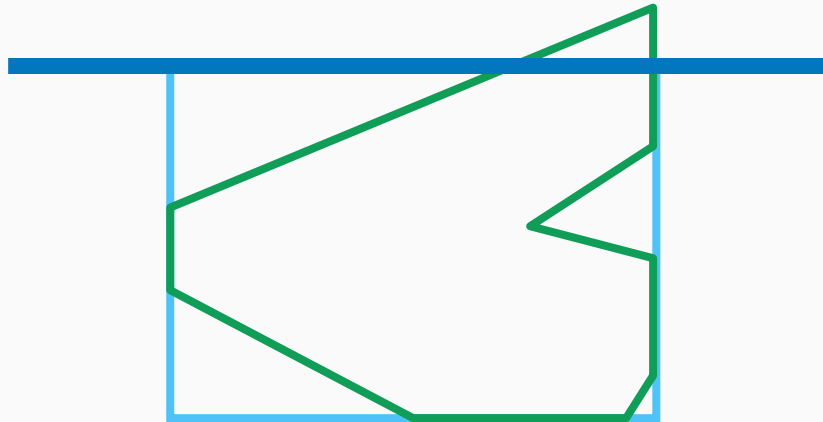
- Dùng từng cạnh của cửa sổ lần lượt xén đa giác.
- Sau khi xén xong, ta được một đa giác đã được xén



Thuật toán Sutherland-Hodgman

Ý tưởng chính:

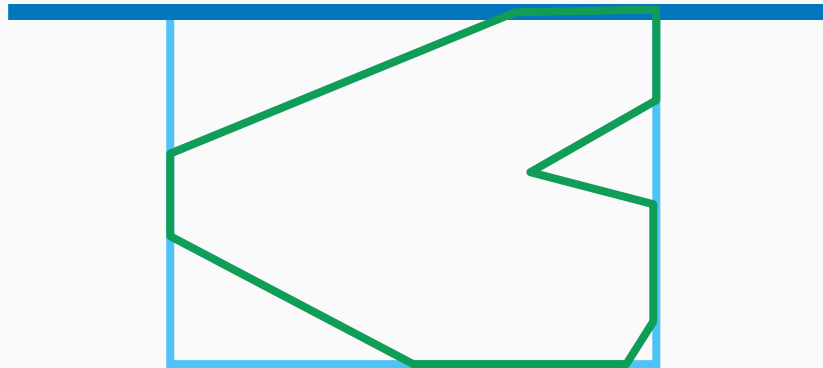
- Dùng từng cạnh của cửa sổ lần lượt xén đa giác.
- Sau khi xén xong, ta được một đa giác đã được xén



Thuật toán Sutherland-Hodgman

Ý tưởng chính:

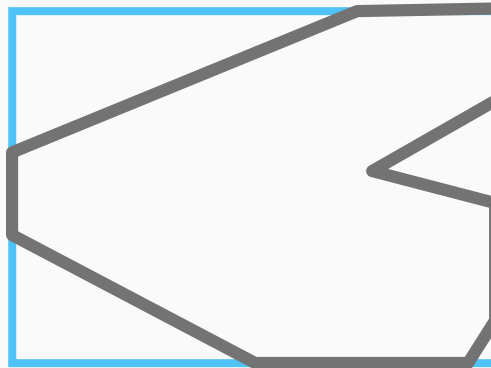
- Dùng từng cạnh của cửa sổ lần lượt xén đa giác.
- Sau khi xén xong, ta được một đa giác đã được xén



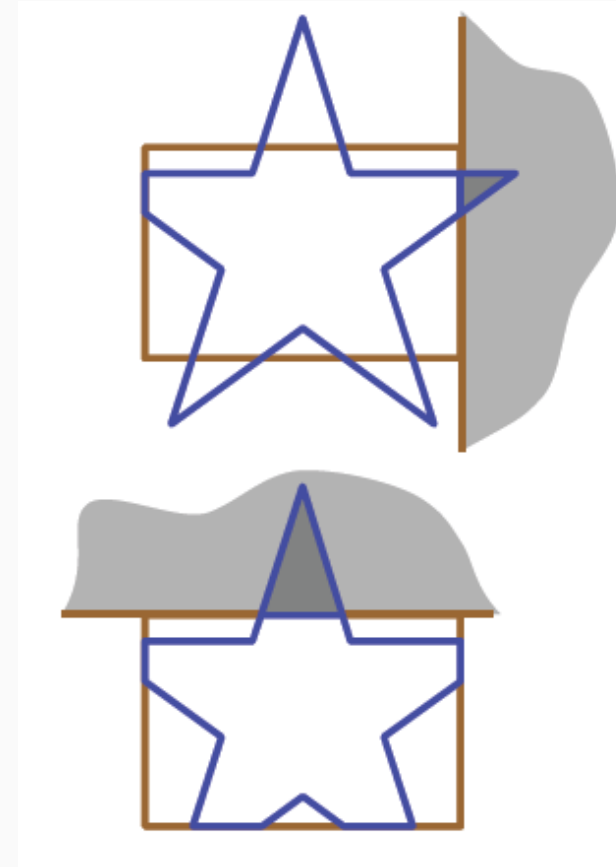
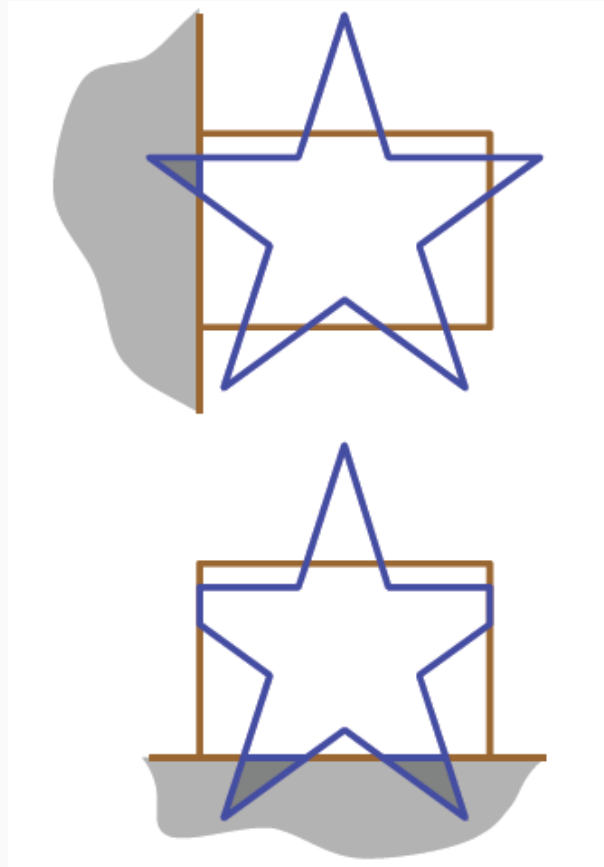
Thuật toán Sutherland-Hodgman

Ý tưởng chính:

- Dùng từng cạnh của cửa sổ lần lượt xén đa giác.
- Sau khi xén xong, ta được một đa giác đã được xén

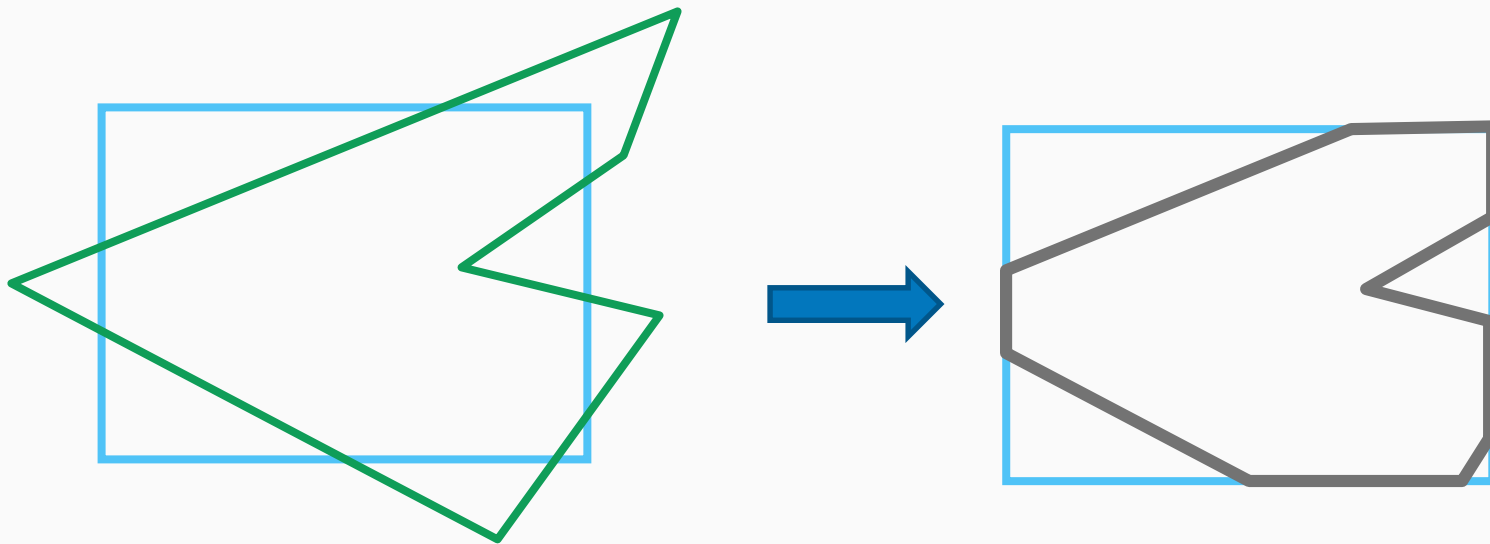


Thuật toán Sutherland-Hodgman



Thuật toán Sutherland-Hodgman

- Input: list of polygon vertices in order
- Output: list of clipped polygon vertices consisting of old vertices (maybe) and new vertices (maybe)



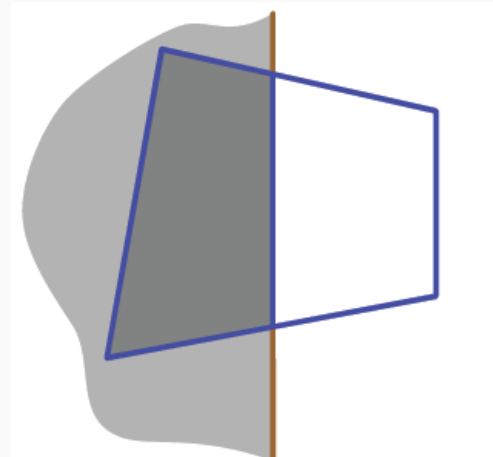
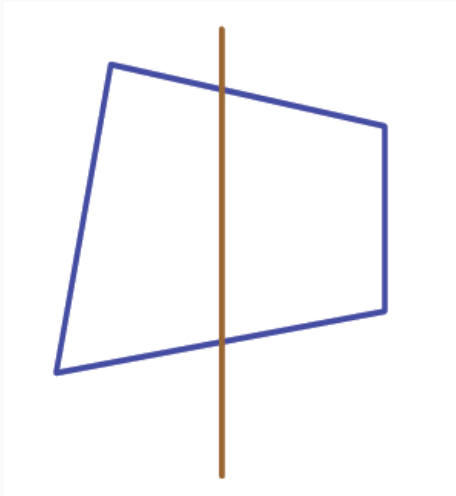
Xén đa giác bằng cạnh trái

Input

Đa giác **IN** = $\{p_0, p_1, \dots, p_{n-1}\}$

Output

Đa giác **OUT** = $\mathbf{IN} \cap W$



Thuật toán xén đa giác bằng cạnh trái

Bước 1

OUT = {}

Bước 2

Lặp

p : $p_0 \dots p_{n-1}$

s là đỉnh kề trước của **p**

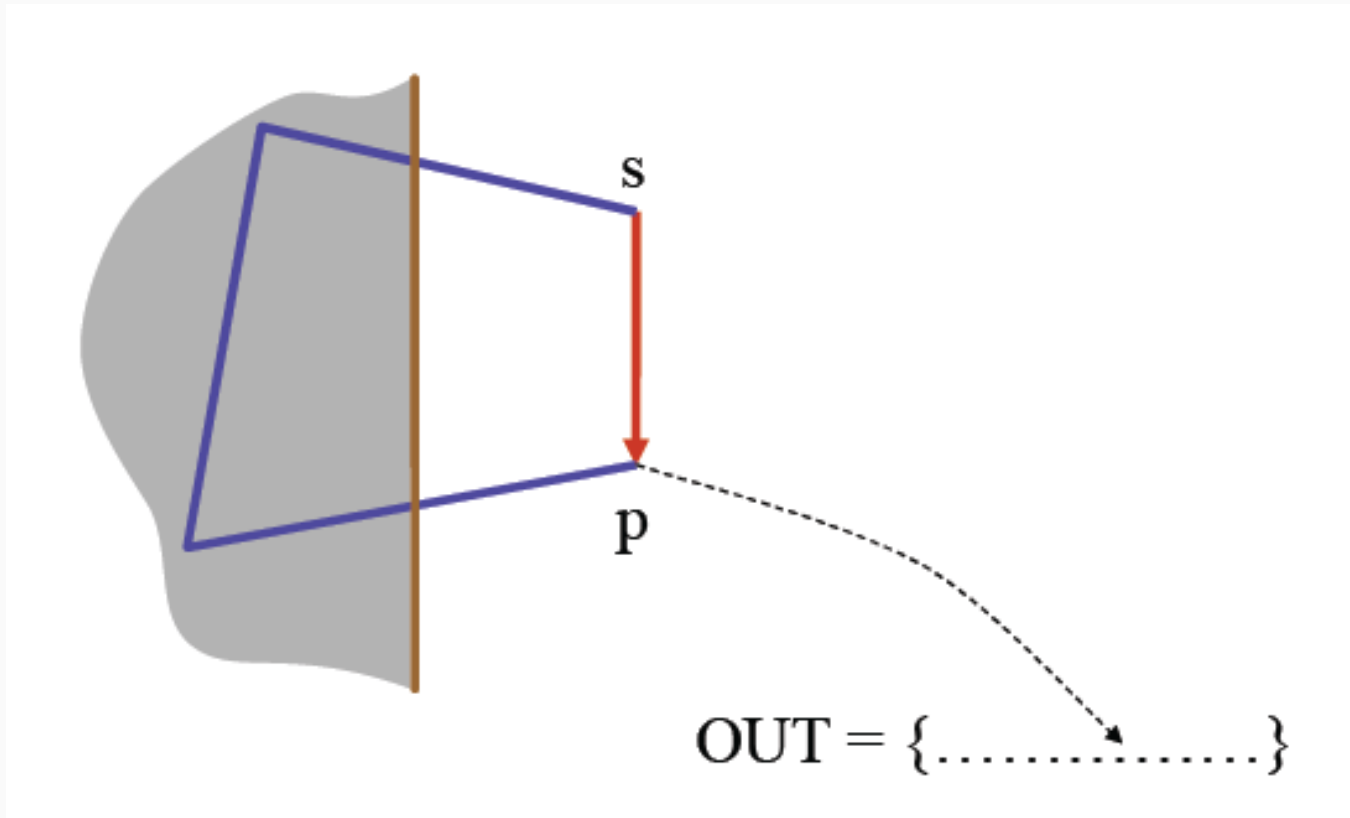
th1 : **p** bên trong, **s** bên trong

th2 : **p** bên ngoài, **s** bên trong

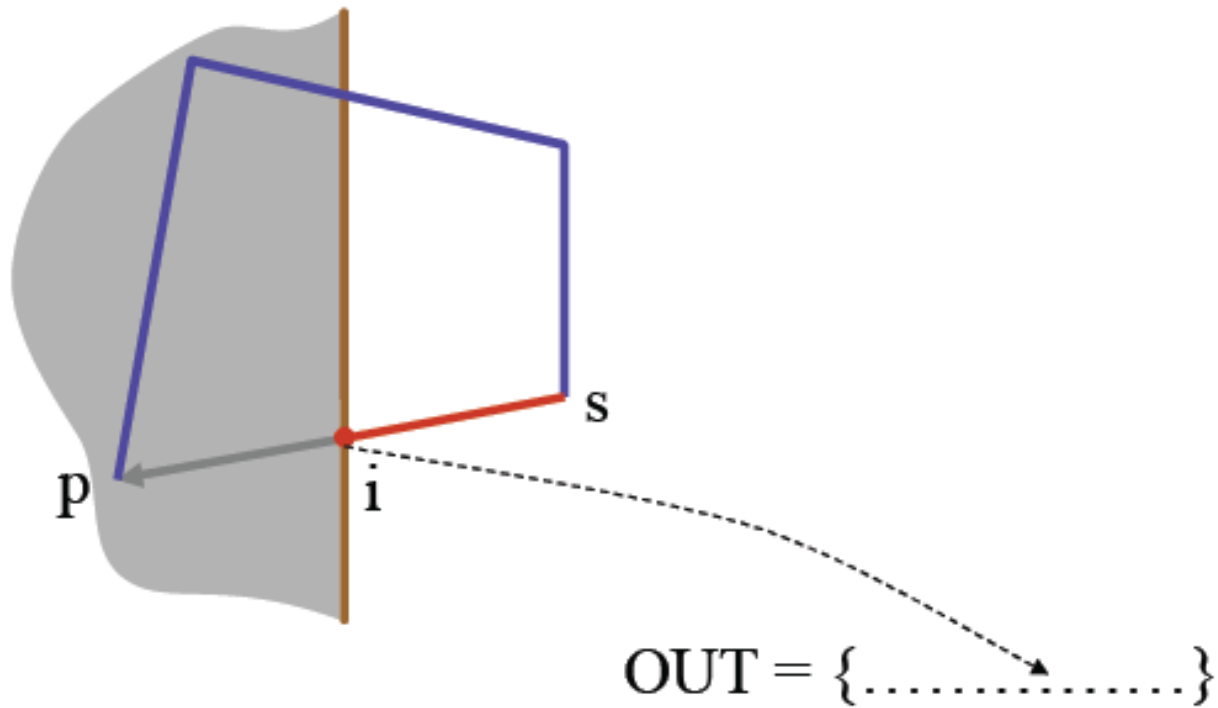
th3 : **p** bên ngoài, **s** bên ngoài

th4 : **p** bên trong, **s** bên ngoài

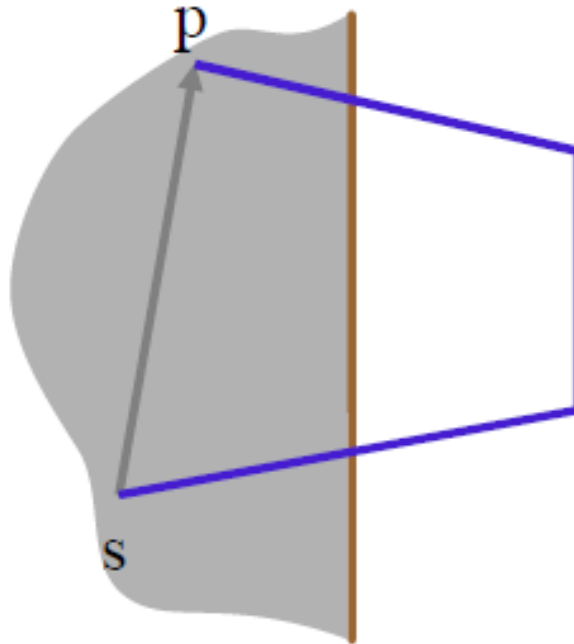
Trường hợp 1



Trường hợp 2

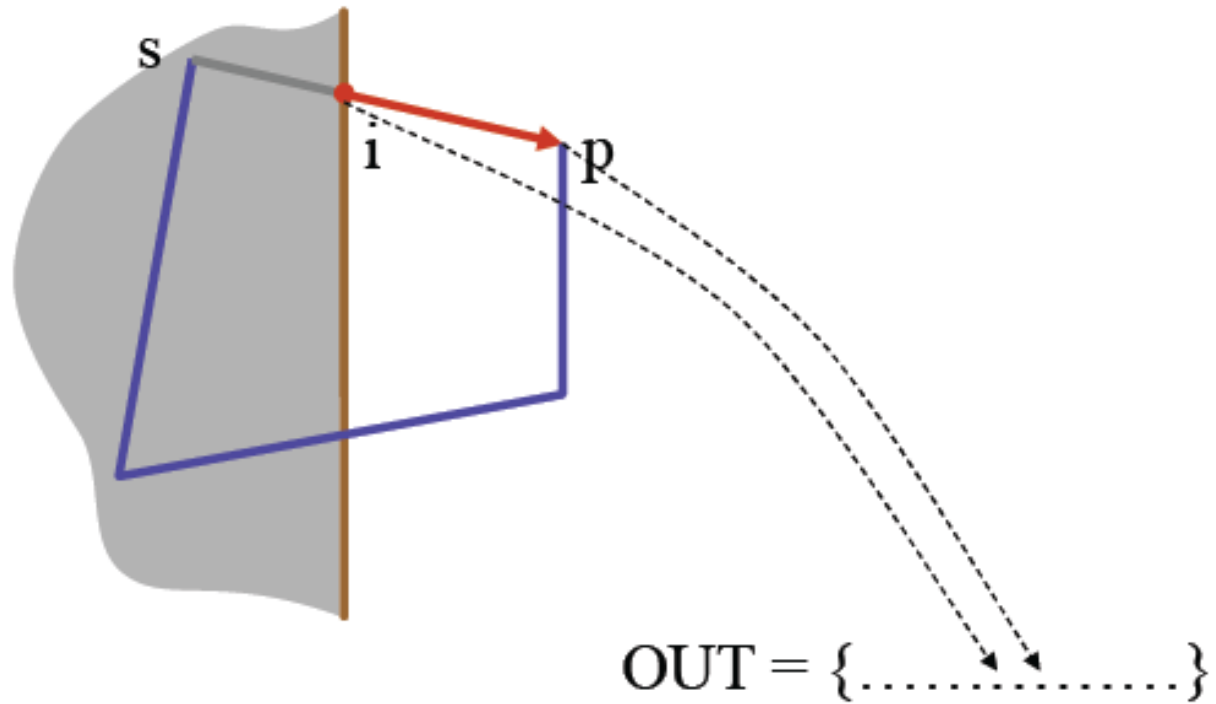


Trường hợp 3

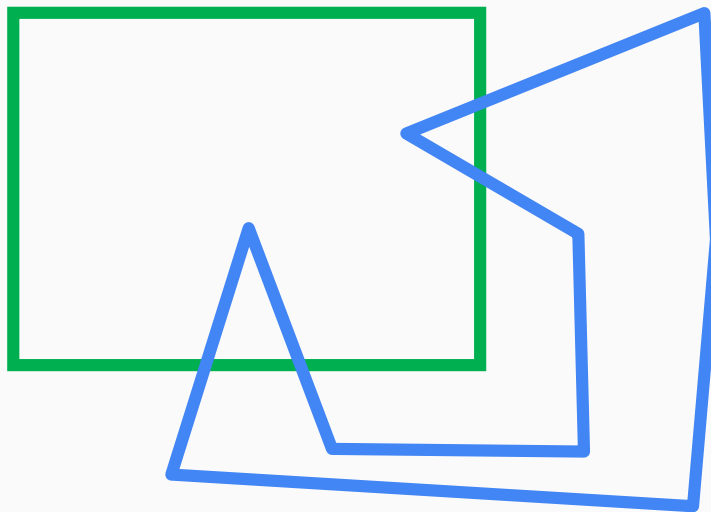
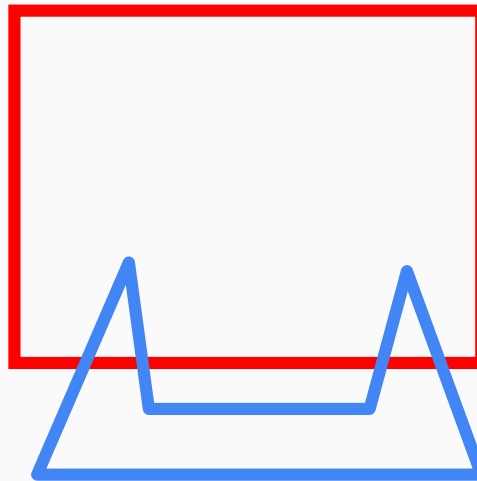
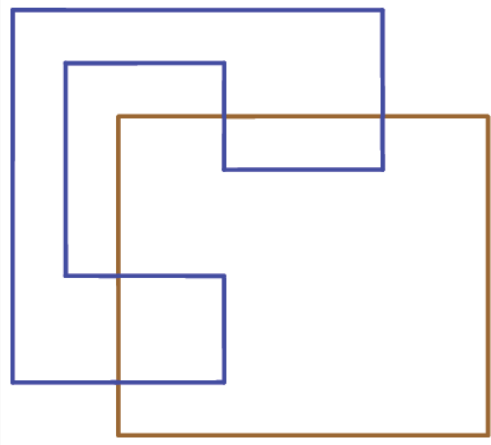


OUT = {.....}

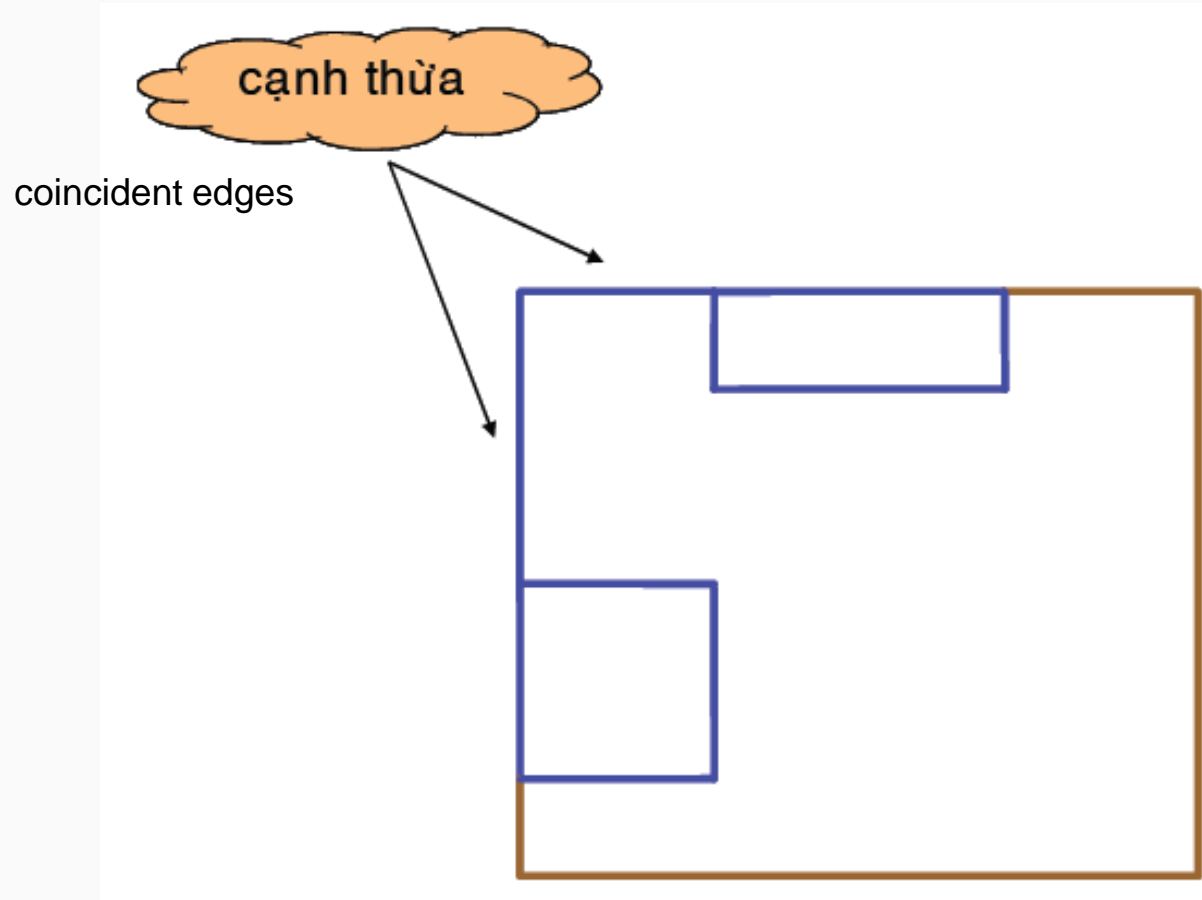
Trường hợp 4



Vấn đề với đa giác lõm



Vấn đề với đa giác lõm



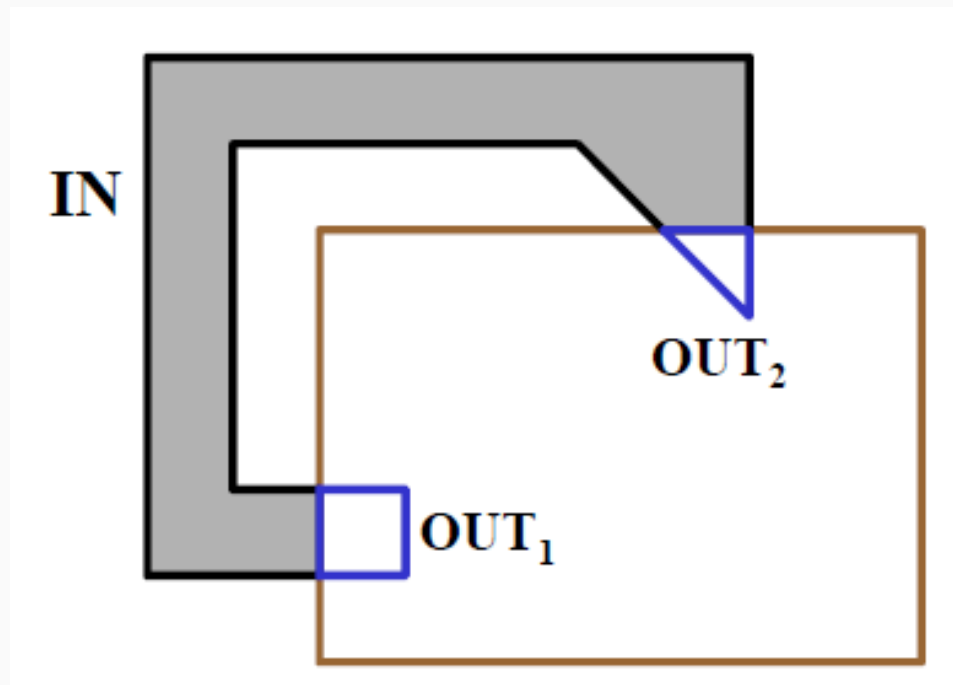
Đặt bài toán

Input

Đa giác IN

Output

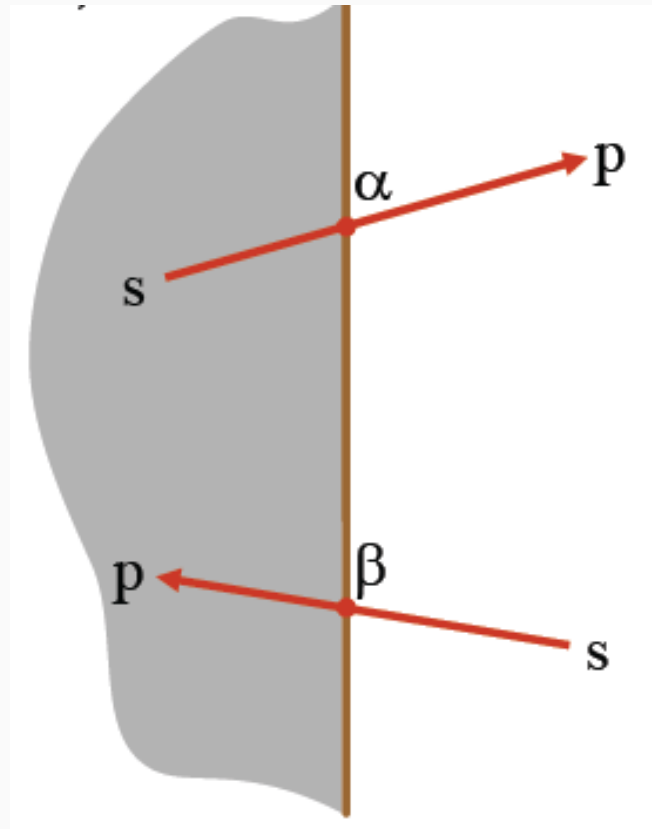
Tập hợp các đa giác $\{OUT_i\}$



Phân loại giao điểm

Giao điểm được chia làm 2 loại

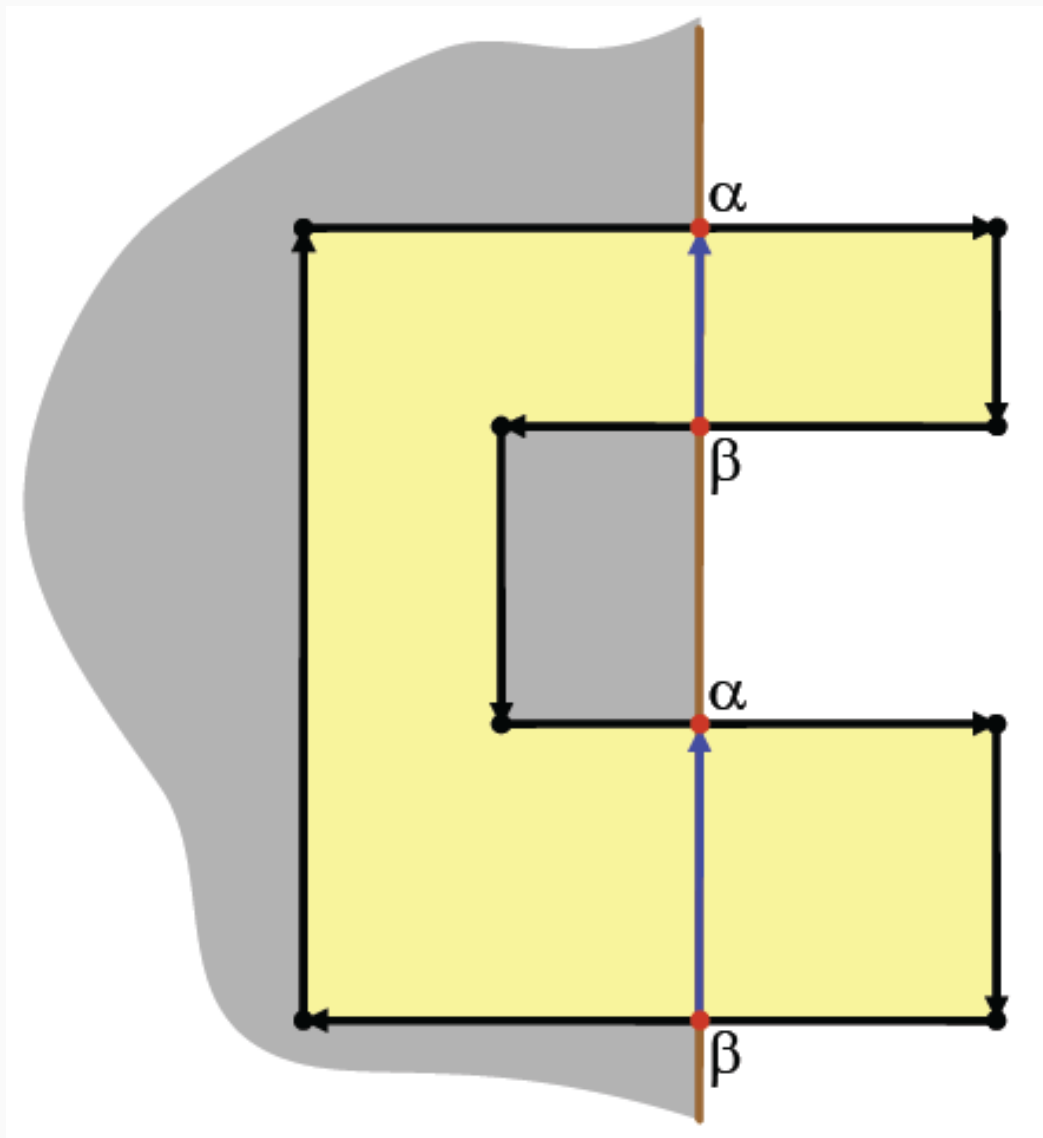
- Loại α (ngoài – trong)
- Loại β (trong – ngoài)



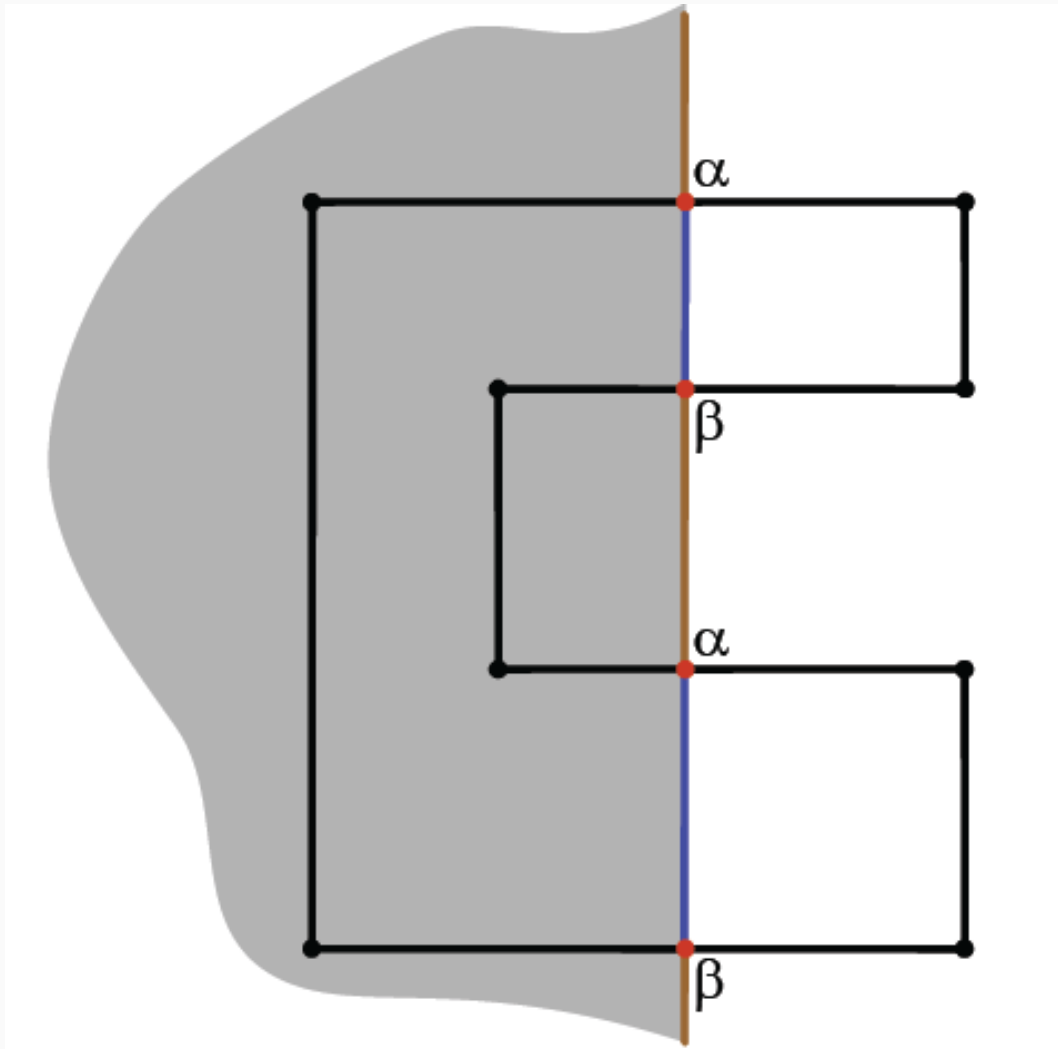
Thuật toán Weiler-Atherton

- Bắt đầu từ bên ngoài
- Gặp giao điểm α thì khởi động đa giác OUT = {}
- Gặp giao điểm β thì kết thúc đa giác OUT

Minh họa



Minh họa



Point-to-Plane test

A very general test to determine if a point p is “inside” a plane P , defined by q and n :

$$(p - q) \cdot n < 0:$$

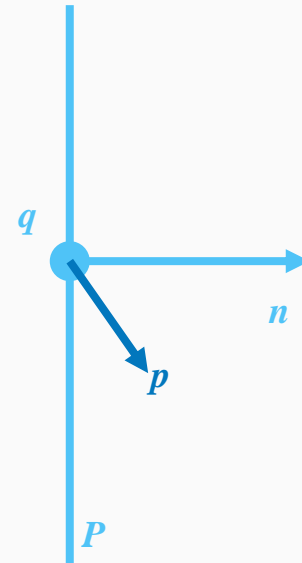
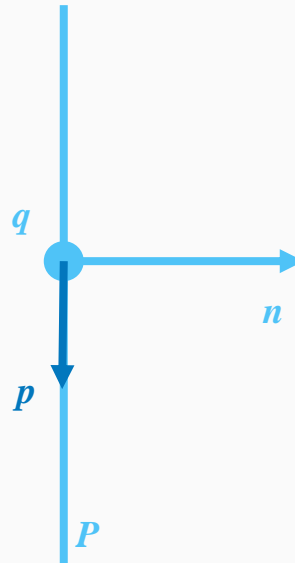
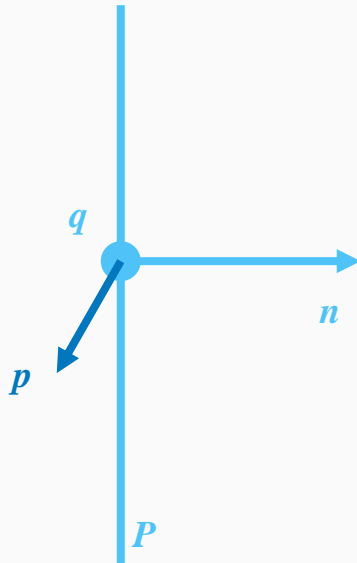
p inside P

$$(p - q) \cdot n = 0:$$

p on P

$$(p - q) \cdot n > 0:$$

p outside P



Finding Line-Plane Intersections

Use parametric definition of edge:

$$E(t) = s + t(p - s)$$

If $t = 0$ then $E(t) = s$

If $t = 1$ then $E(t) = p$

Otherwise, $E(t)$ is part way from s to p

Finding Line-Plane Intersections

Edge intersects plane P where $E(t)$ is on P

q is a point on P

n is normal to P

$$(E(t) - q) \cdot n = 0$$

$$(s + t(p - s) - q) \cdot n = 0$$

$$t = [(q - s) \cdot n] / [(p - s) \cdot n]$$

The intersection point $i = E(t)$ for this value of t

Line-Plane Intersections

Note that the length of \mathbf{n} doesn't affect result:

$$t = [(\mathbf{q} - \mathbf{s}) \cdot \mathbf{n}] / [(\mathbf{p} - \mathbf{s}) \cdot \mathbf{n}]$$

Again, lots of opportunity for optimization

Tài liệu tham khảo

- Slide này được biên soạn được tham khảo từ một số tài liệu sau: