

Computer Graphics

Subodh Kumar

Dept of Computer Sc. & Engg.
IIT Delhi



Line Clipping

- Find intersection with all boundaries
 - Upto 2 possible
- Replace one end point with the intersection point
 - Intersection closest to the vertex



Cohen-Sutherland Outcodes

- Apply trivial tests first [Cull]
 - Use outcode – Mask: 1 bit/ boundary
 - If both vertices outside some boundary
 - Discard Edge
 - If both vertices inside all boundary
 - Accept edge
- Otherwise, clip with each boundary
- Logical OPs

0011	0010	0110
0001	0000	0100
1001	1000	1100



Clip Space

- Transform to Normalized Image space
 - Clip against $-1 \leq X, Y \leq 1$
 - Implies a division (by W) per vertex
- Can we clip in 3D?
 - 3D frustum in object space
 - Camera space
- Homogeneous space clipping
 - $-w \leq X, Y \leq w$
 - What are the outcodes?



Find Intersection

- Simultaneous equations:
 - $ax+by+c=0$
 - $dx+ey+f=0$
- But (x, y) must be between end-points
- Alternative, use parametric equations
 - $x_0 + (x_1-x_0)t = x_2 + (x_3-x_2)u$
 - $y_0 + (y_1-y_0)t = y_2 + (y_3-y_2)u$
- t and u must be in $[0:1]$



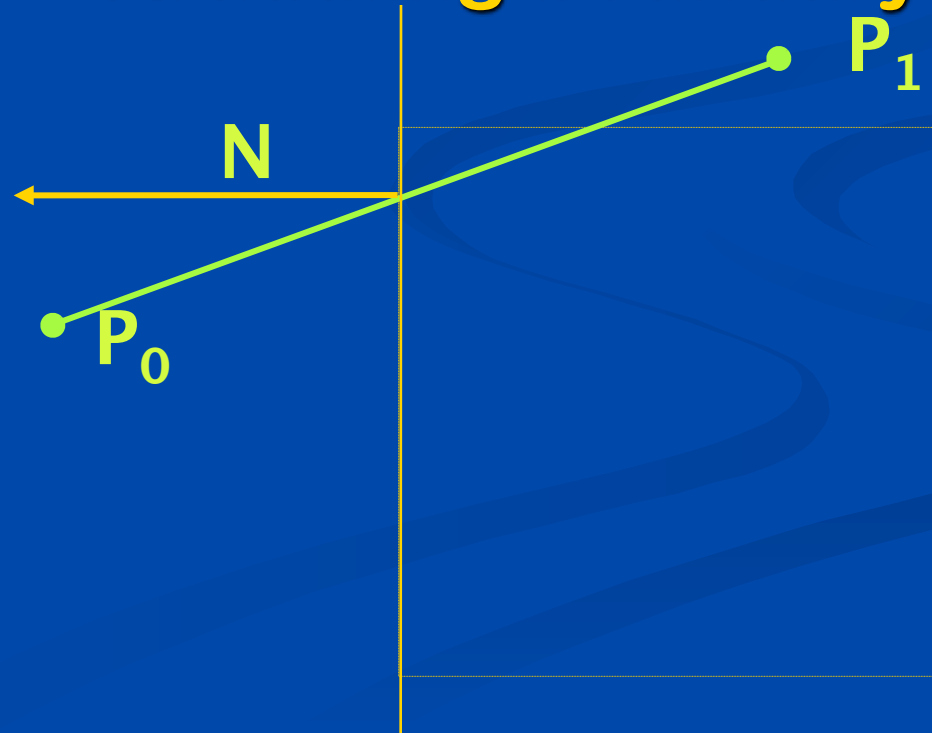
Cyrus-Beck Clipping

- Can we reduce the number of intersection computations?
- Can we quickly reject a few potential intersections?



Classify Intersections

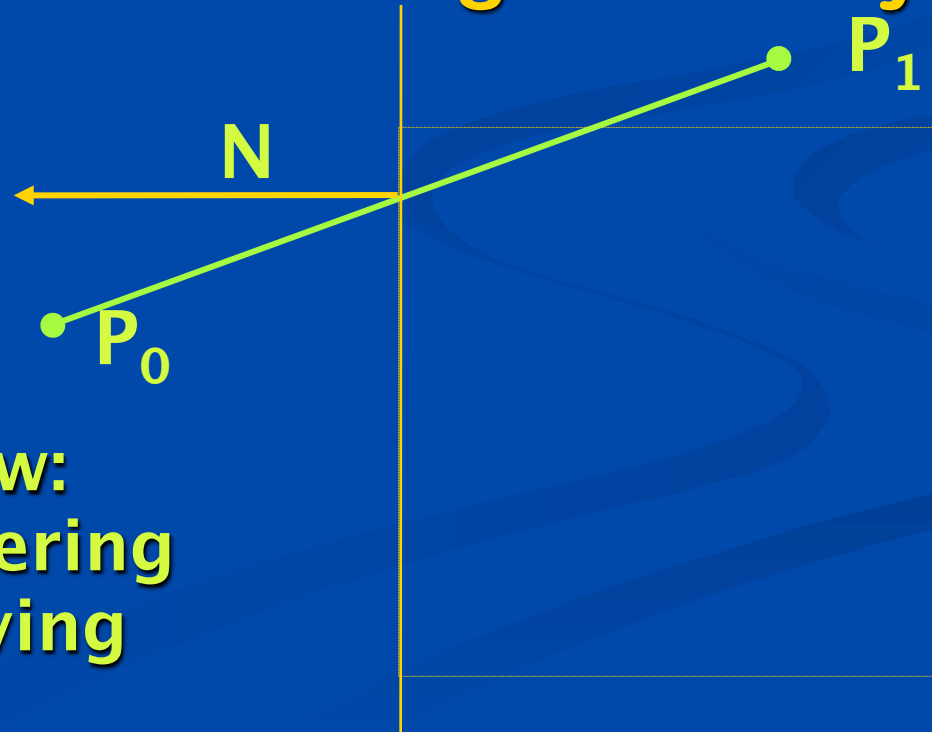
- $\overrightarrow{P_0 P_1} \cdot \vec{N} < 0 \Rightarrow$ entering boundary
- $\overrightarrow{P_0 P_1} \cdot \vec{N} > 0 \Rightarrow$ leaving boundary
- $\overrightarrow{P_0 P_1} \cdot \vec{N} = 0 \Rightarrow$ straddling boundary





Classify Intersections

- $\overrightarrow{P_0 P_1} \cdot \vec{N} < 0 \Rightarrow$ entering boundary
- $\overrightarrow{P_0 P_1} \cdot \vec{N} > 0 \Rightarrow$ leaving boundary
- $\overrightarrow{P_0 P_1} \cdot \vec{N} = 0 \Rightarrow$ straddling boundary

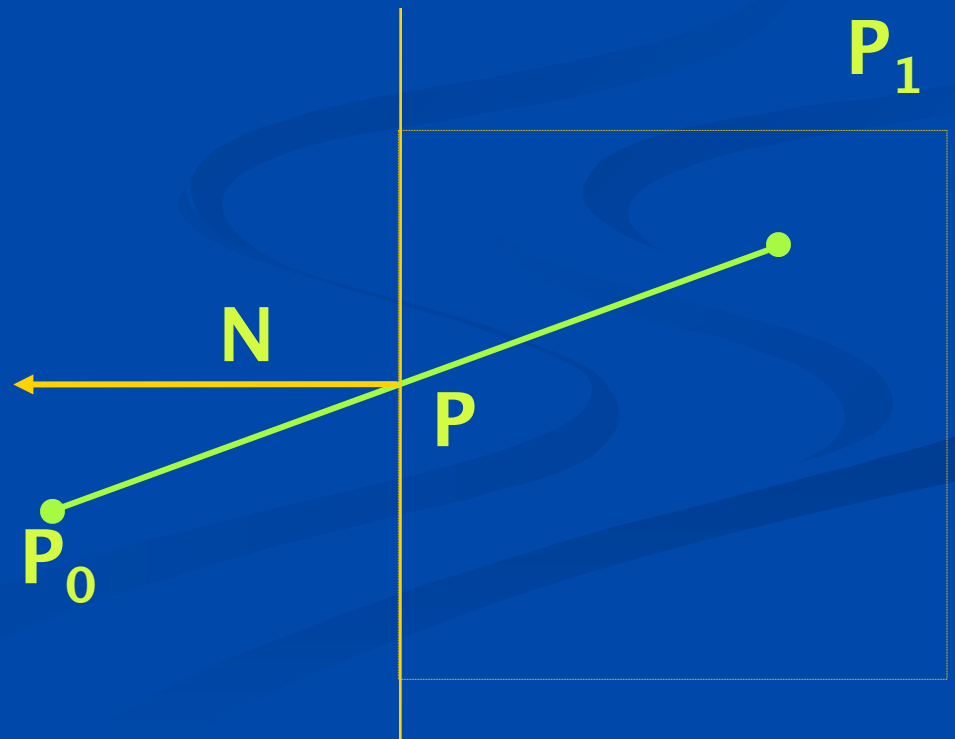


For convex window:
Find last entering
and first leaving

Intersection Computation



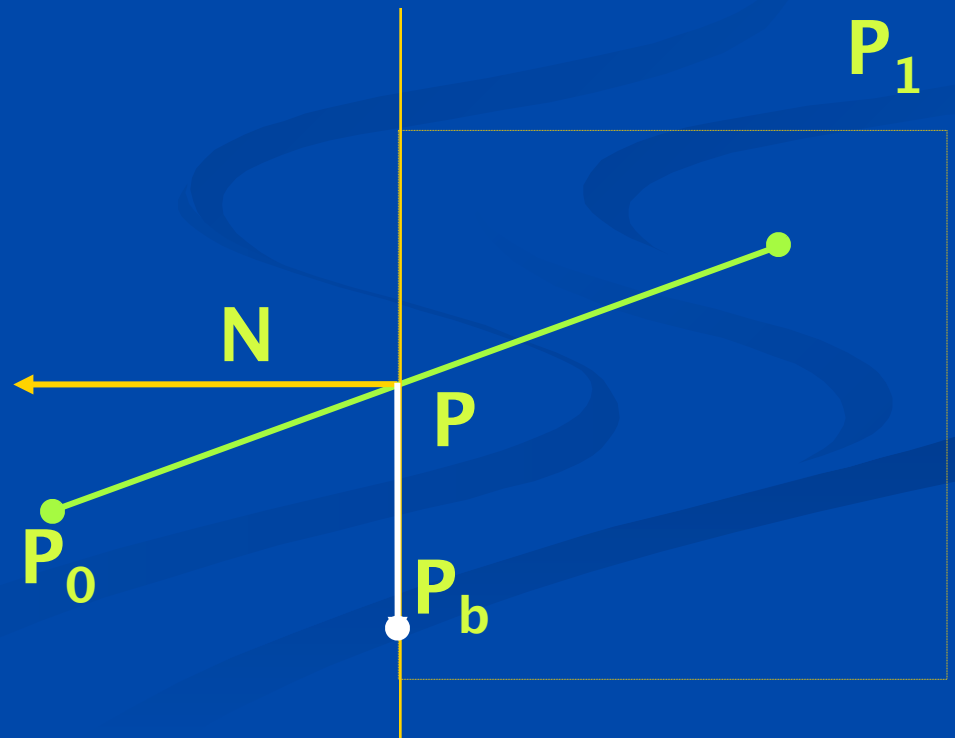
[Liang-Barsky]



Intersection Computation



[Liang–Barsky]

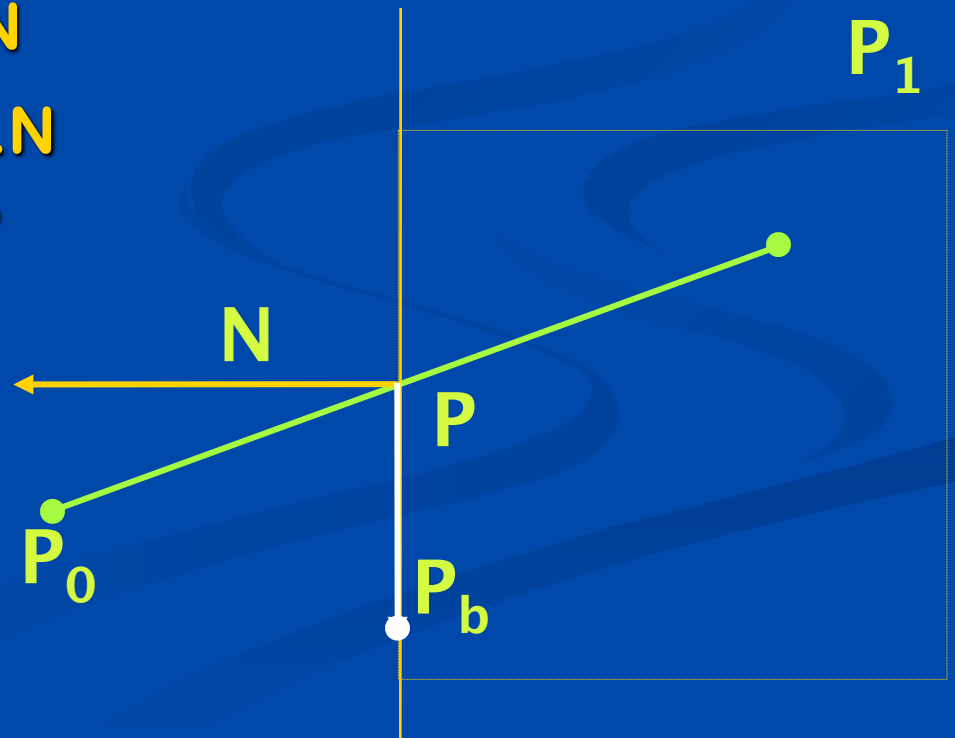




Intersection Computation

- $\overrightarrow{P P_b} \cdot \overrightarrow{N} = 0$
 - $(P - P_b) \cdot N = 0$
 - $[P_0 + (P_1 - P_0)t - P_b] \cdot N = 0$
 - $(P_1 - P_0) \cdot N t = (P_b - P_0) \cdot N$
 - $t = (P_b - P_0) \cdot N / (P_1 - P_0) \cdot N$
 - If $N = \pm [0, 1]$ or $[1, 0]$?
 - If P_b is a corner?

[Liang-Barsky]



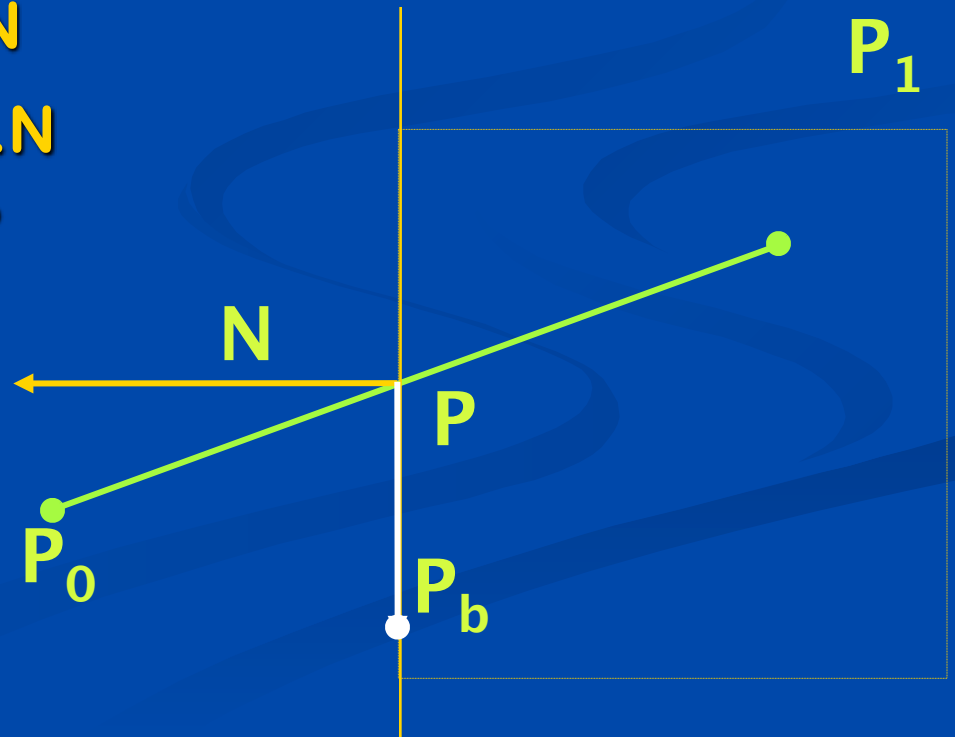


Intersection Computation

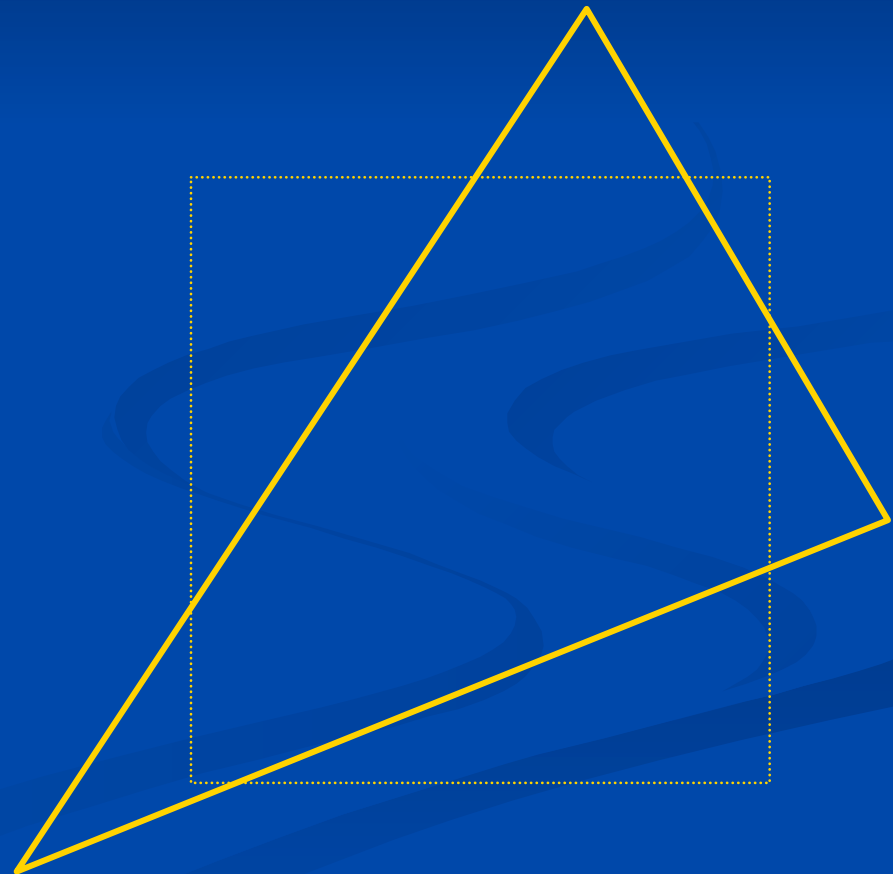
- $\overrightarrow{P P_b} \cdot \overrightarrow{N} = 0$
 - $(P - P_b) \cdot N = 0$
 - $[P_0 + (P_1 - P_0)t - P_b] \cdot N = 0$
 - $(P_1 - P_0) \cdot N t = (P_b - P_0) \cdot N$
 - $t = (P_b - P_0) \cdot N / (P_1 - P_0) \cdot N$
 - If $N = \pm [0, 1]$ or $[1, 0]$?
 - If P_b is a corner?

[Liang-Barsky]

What happens in 3D?



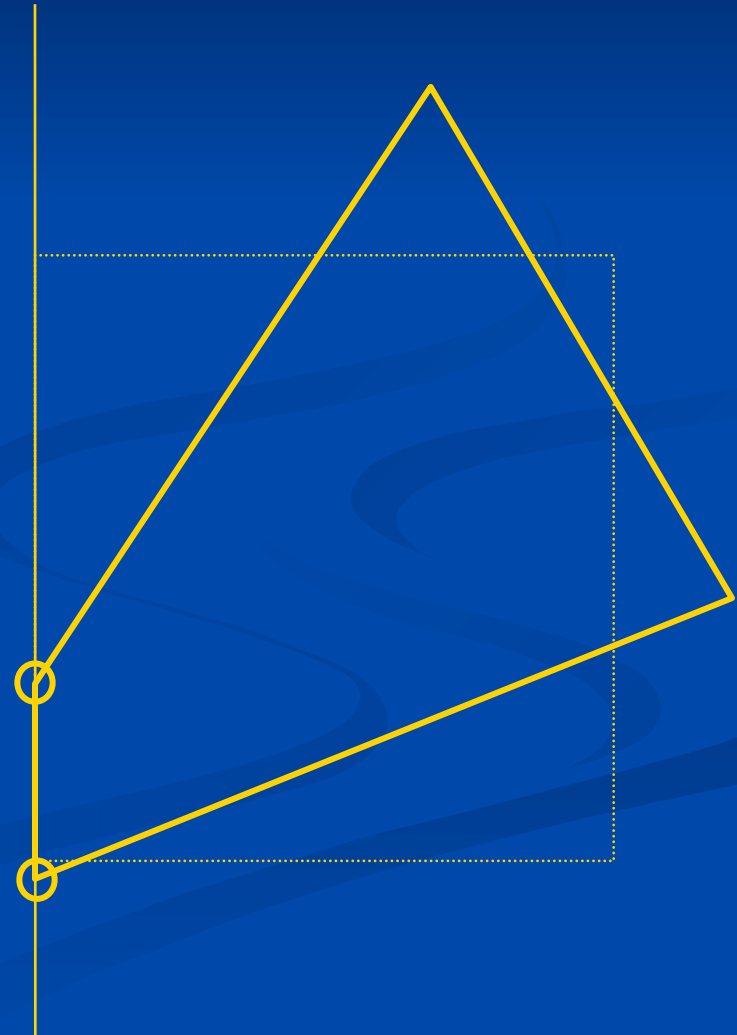
Triangle Clipping





Clip Against Each Boundary

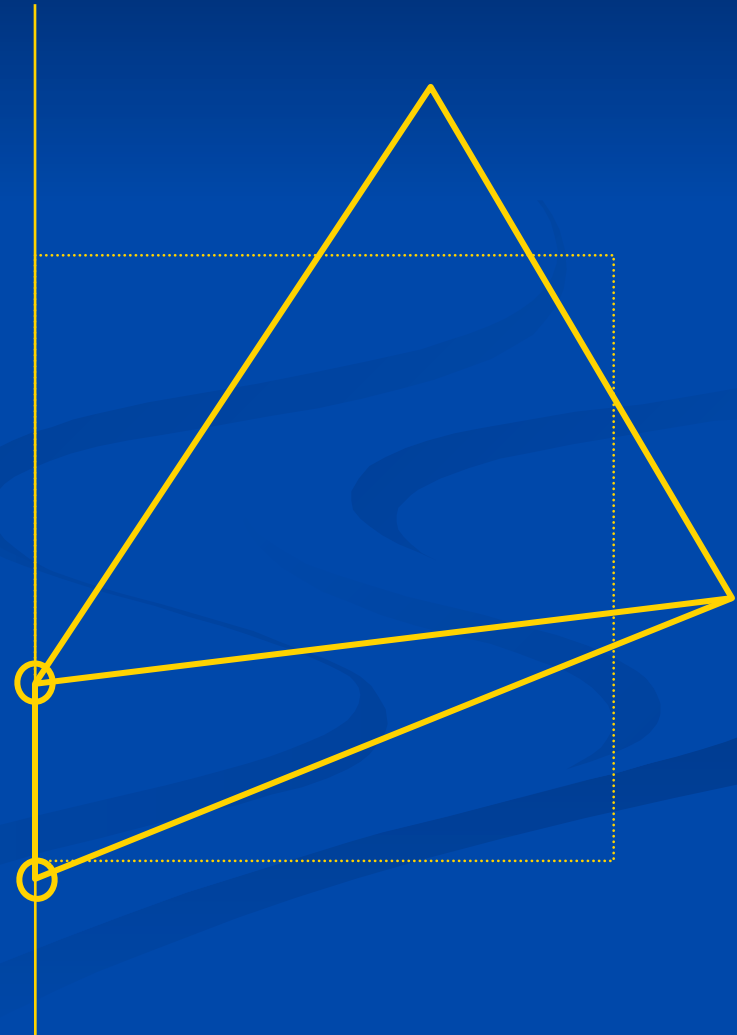
- Could triangulate now
 - Any other case?





Clip Against Each Boundary

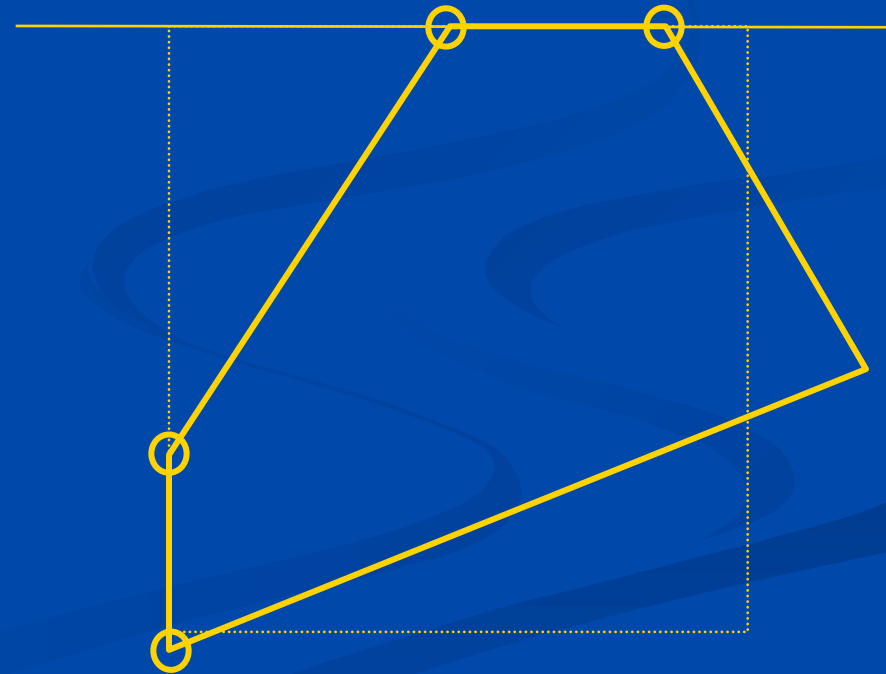
- Could triangulate now
 - Any other case?



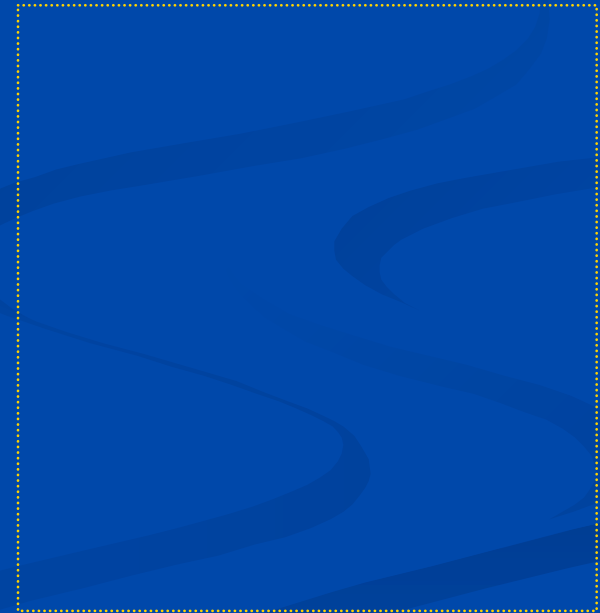


Clip Against Each Boundary

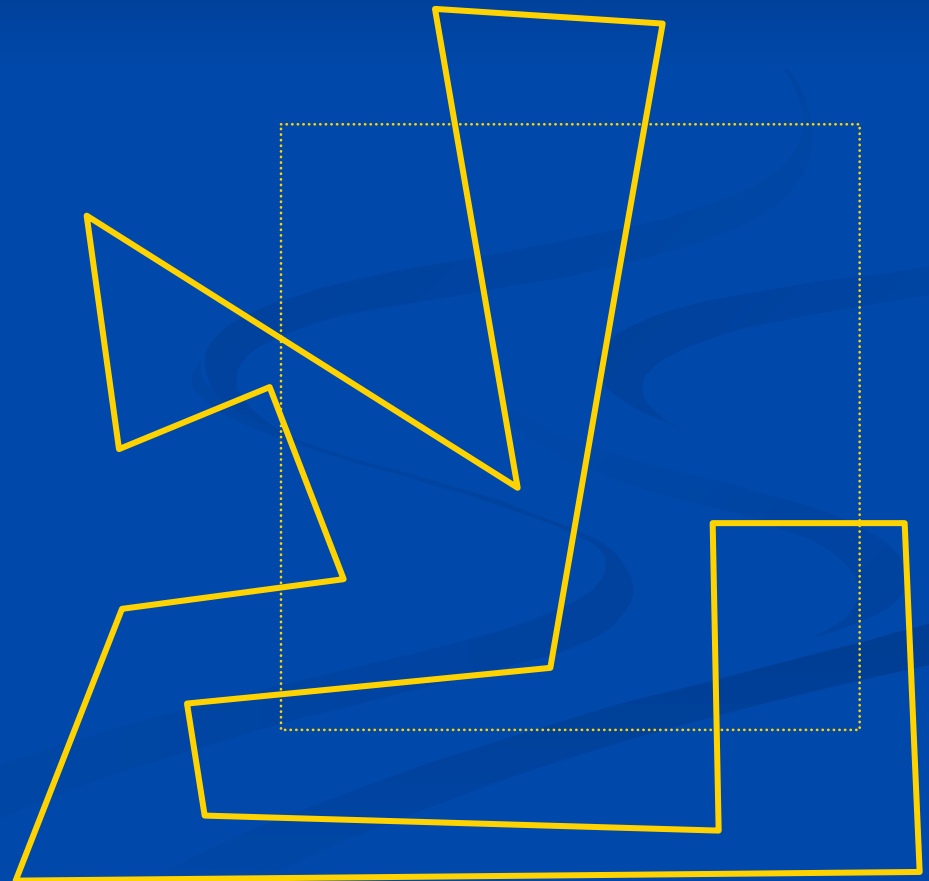
- Clip against the next boundary
 - And so on..



Clipping General Polygons



Clipping General Polygons





Clipping General Polygon

- Clip against each boundary
- Traverse Polygon for each boundary
 - If In \rightarrow In
 - Output new vertex
 - If In \rightarrow Out
 - Output intersection point
 - Output new vertex
 - If Out \rightarrow In
 - Output Intersection point

[Sutherland–Hodgeman]

Clipping General Polygon



- Clip against each boundary
- Traverse Polygon for each boundary
 - If In \rightarrow In
 - Output new vertex
 - If In \rightarrow Out
 - Output intersection point
 - Output new vertex
 - If Out \rightarrow In
 - Output Intersection point

Test Trivial Reject/Accept First!

[Sutherland–Hodgeman]

Trivial Acceptance Rejection



- Object space partitioning
 - Object bounding volume hierarchy
- Occlusion culling
- Hierarchical Z-buffer
 - Early-Z test

Clipping Vertex Attributes



$$a_t = (1-t)a_0 + t a_1$$

