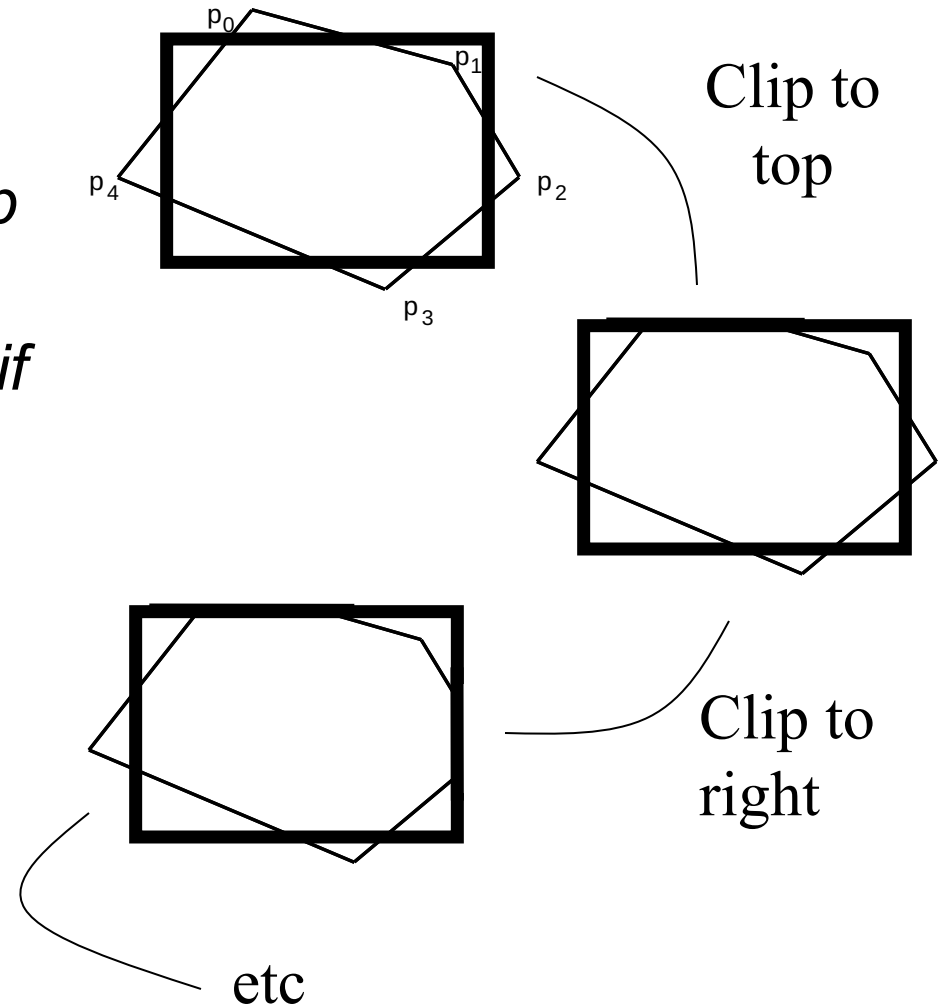# *Clipping*

©Yiorgos Chrysanthou 2001, Anthony Steed 2002-2003, Celine Loscos 2004

Clipping Summary

- *It's the process of finding the exact part of a polygon lying inside the view volume*

- *To maintain consistency, clipping of a polygon should result in a polygon, not a sequence of partially unconnected lines*

- *We will first look at 2 different 2D solutions and then extend one to 3D*

Sutherland-Hodgman Algorithm

- *Clip the polygon against each boundary of the clip region successively*
- *Result is possibly NULL if polygon is outside*
- *Can be generalised to work for any polygonal clip region, not just rectangular*

p_0
p_1
p_4
p_2
p_3

Clip to top

Clip to right

etc

Clipping To A Region

- *To find the new polygon*
  - *iterate through each of the polygon edges and construct a new sequence of points*
  - *starting with an empty sequence*
 - *for each edge there are 4 possible cases to consider*

right clip boundary

P_1 P_2

P_0

P_3

clip region
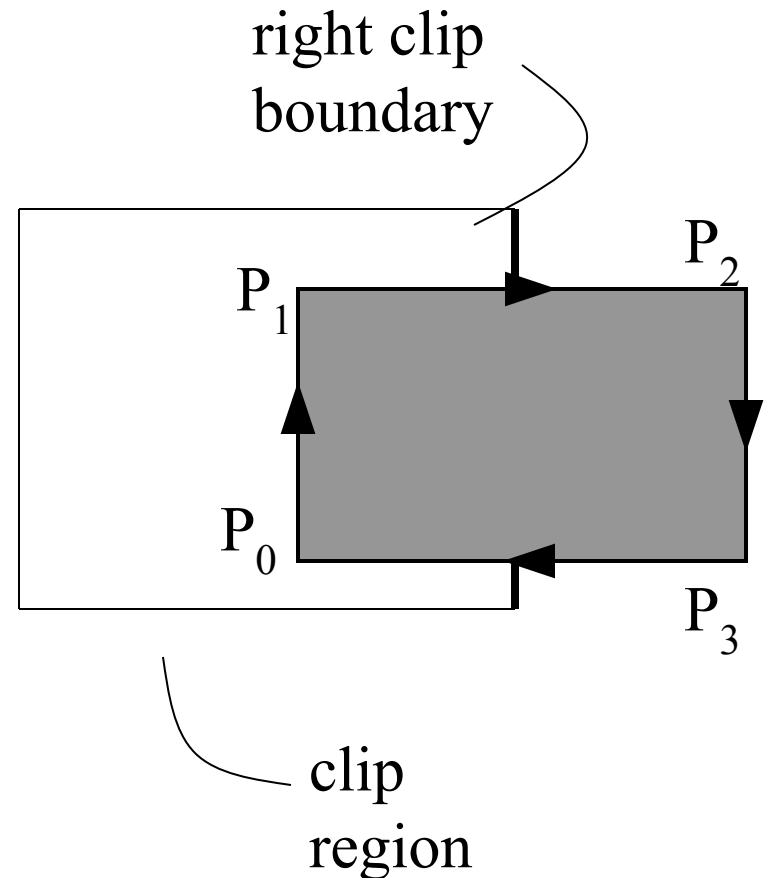
Clipping a polygon edge against the boundary

- *Given an edge P_0, P_1 we have 4 cases:*
  - *entering the clipping region*
    - *add P and P_1*
  - *leaving the region*
    - *add only P*
  - *entirely outside*
    - *do nothing*
  - *entirely inside*
    - *add only P1*
- *Where P is the point of intersection*

Visible Side

p_1 ◀ p p_0

p p_1
p_0
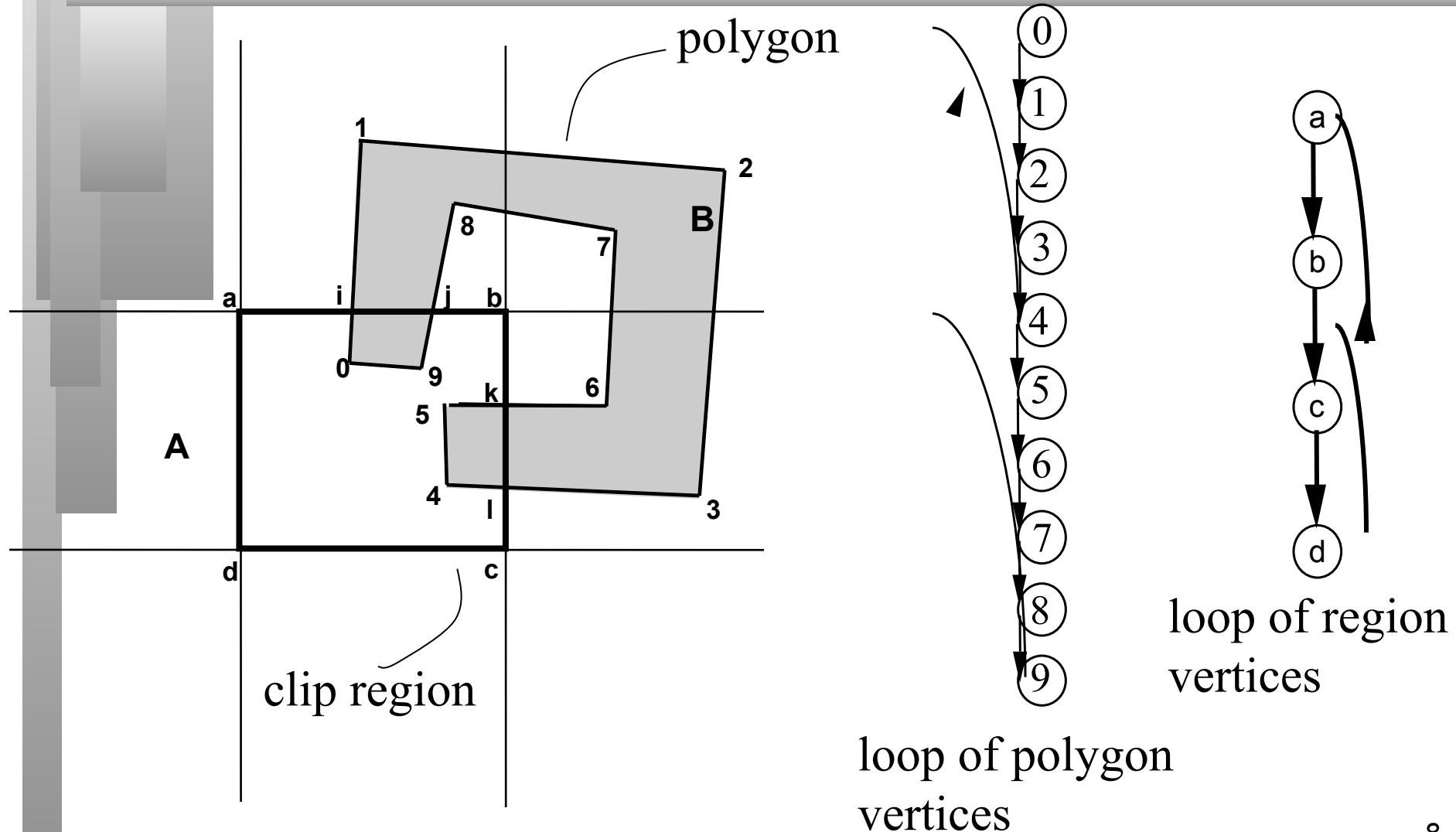
p_1
p_0

p_1 ◀ p_0

IN   OUT
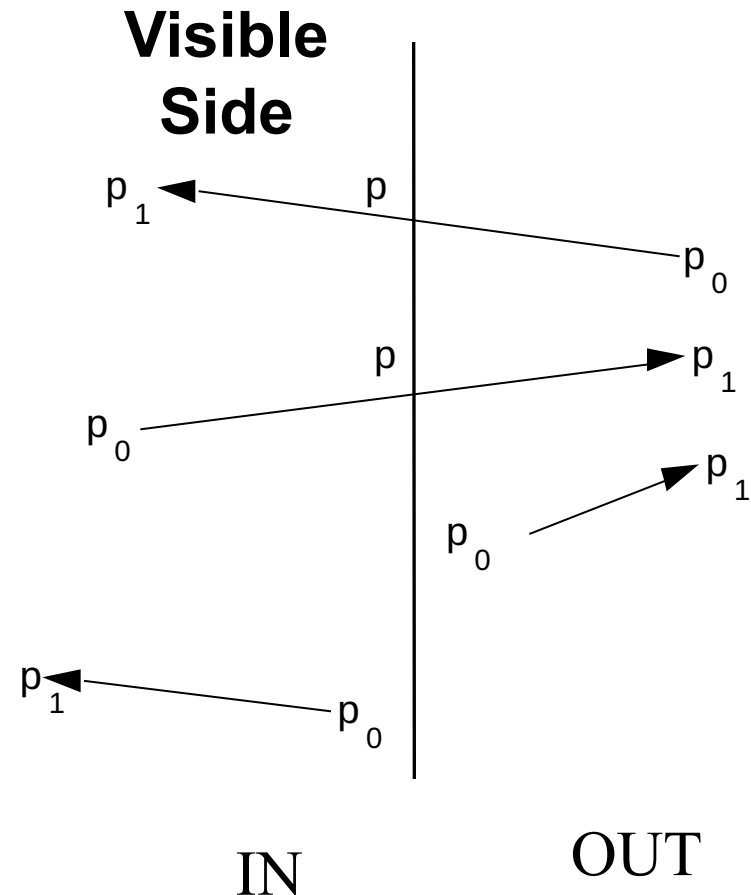
Still the Sutherland-Hodgman

- *We can determine which of the 4 cases and also the point of intersection with just if statements*
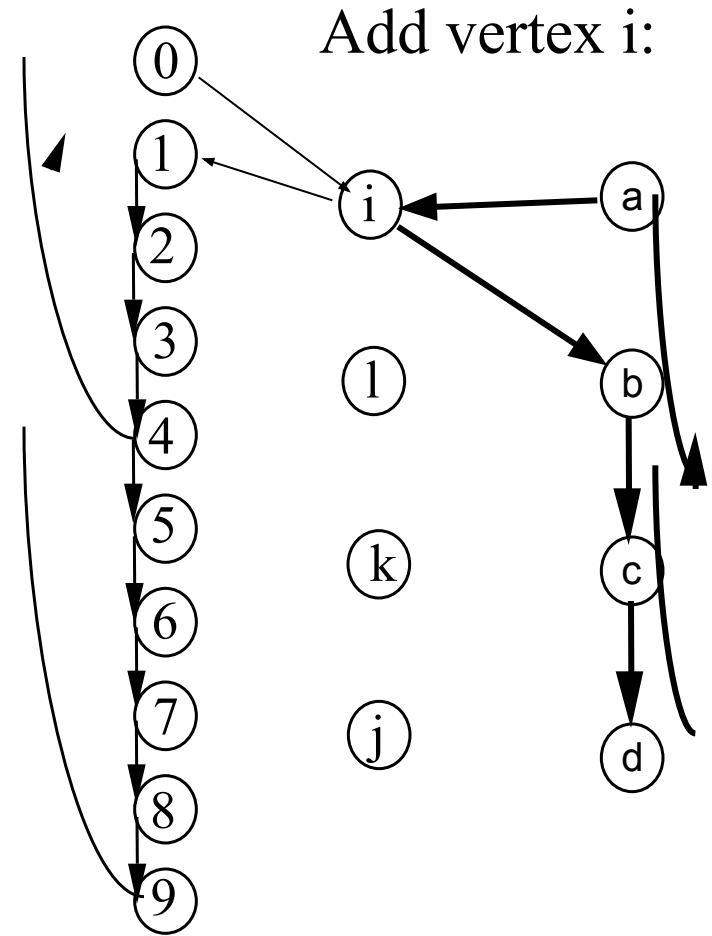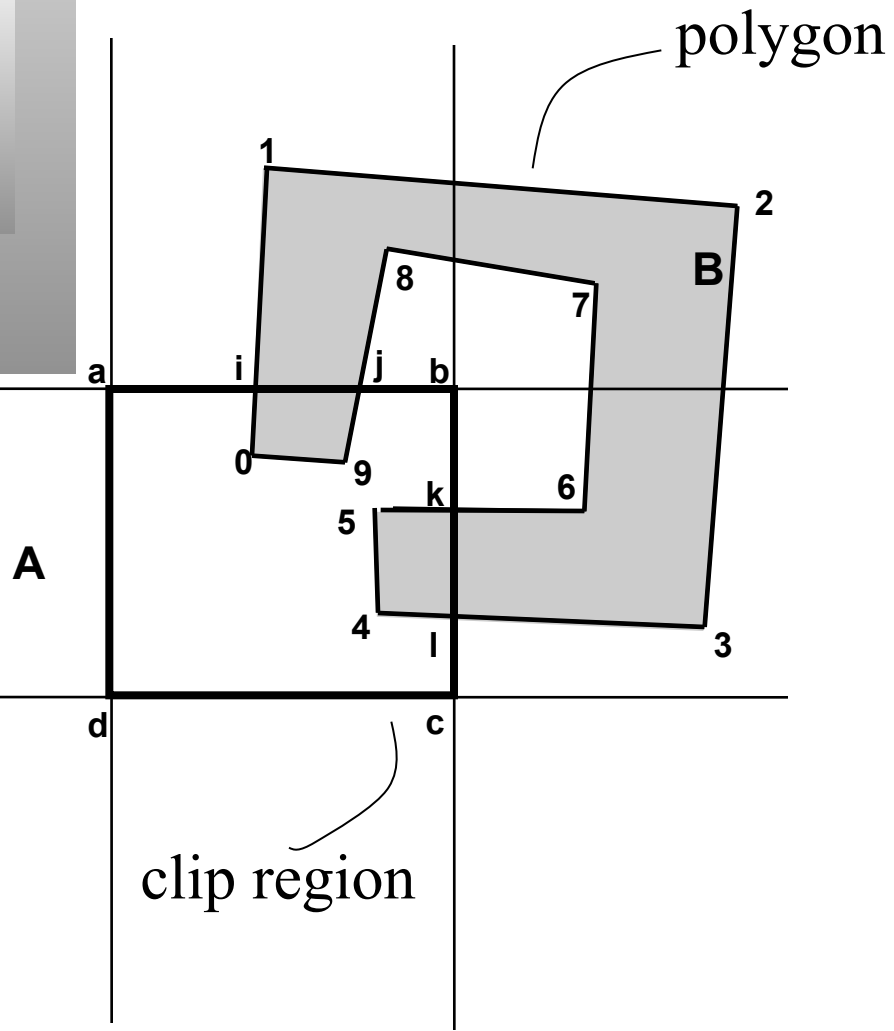- *To sum it up, an example:*

P_1 P_2

P_0 P_3

P_a P_b

P_0 P_3
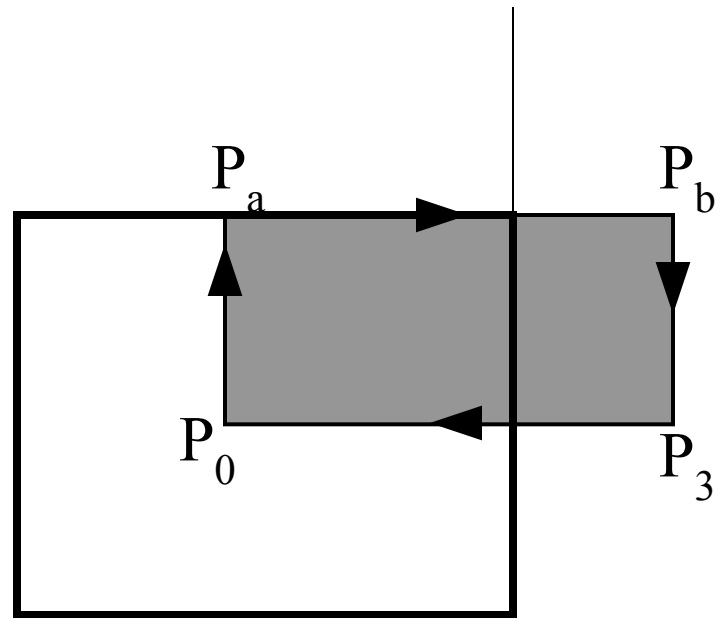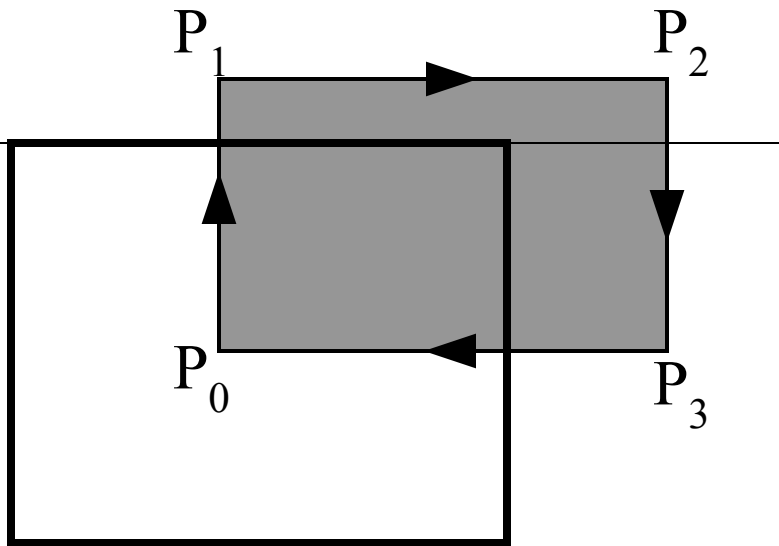
Weiler-Atherton Algorithm

- *When we have non-convex polygons then the algorithm above might produce polygons with coincident edges*

- *This is fine for rendering but maybe not for other applications (eg shadows)*

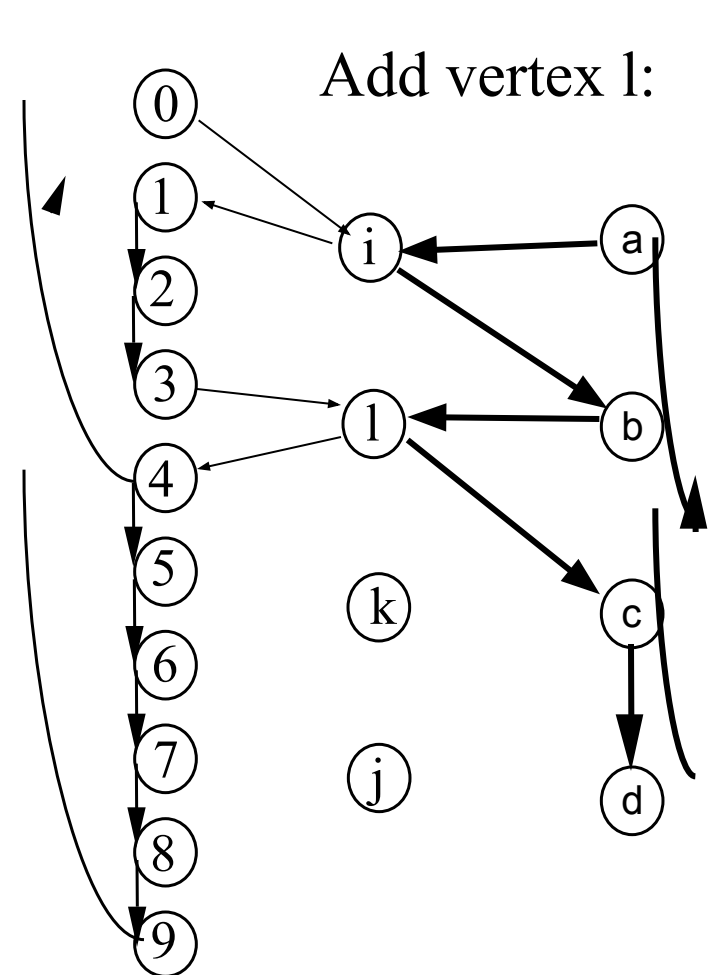- *The Weiler-Atherton algorithm produces separate polygons for each visible fragment*
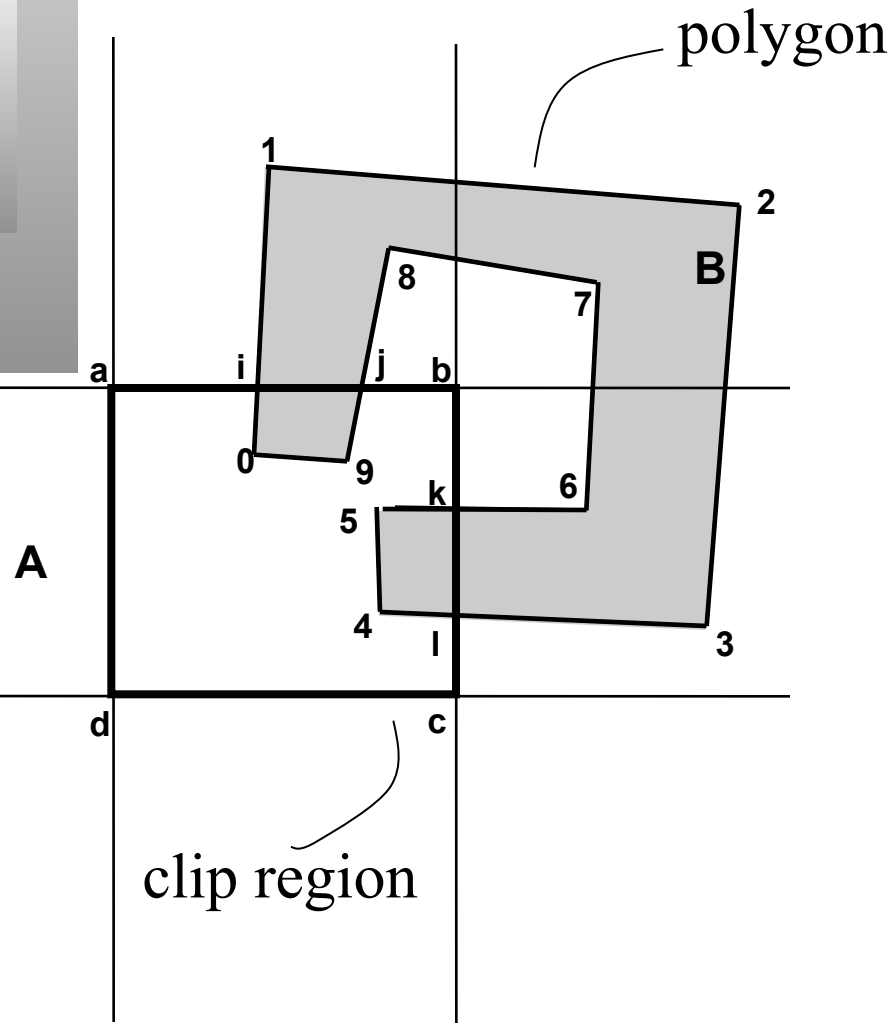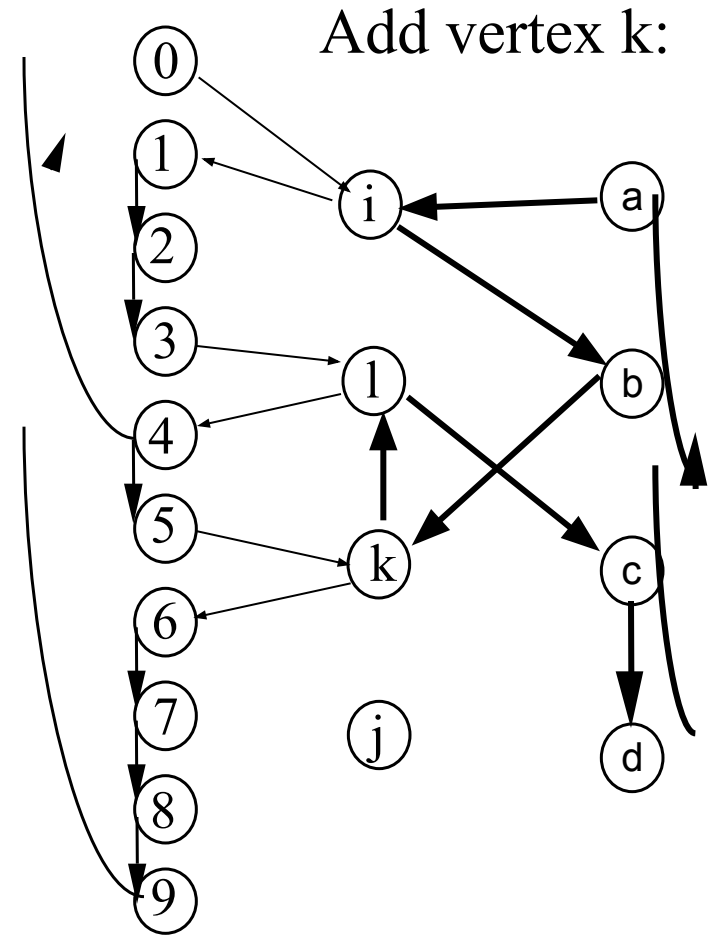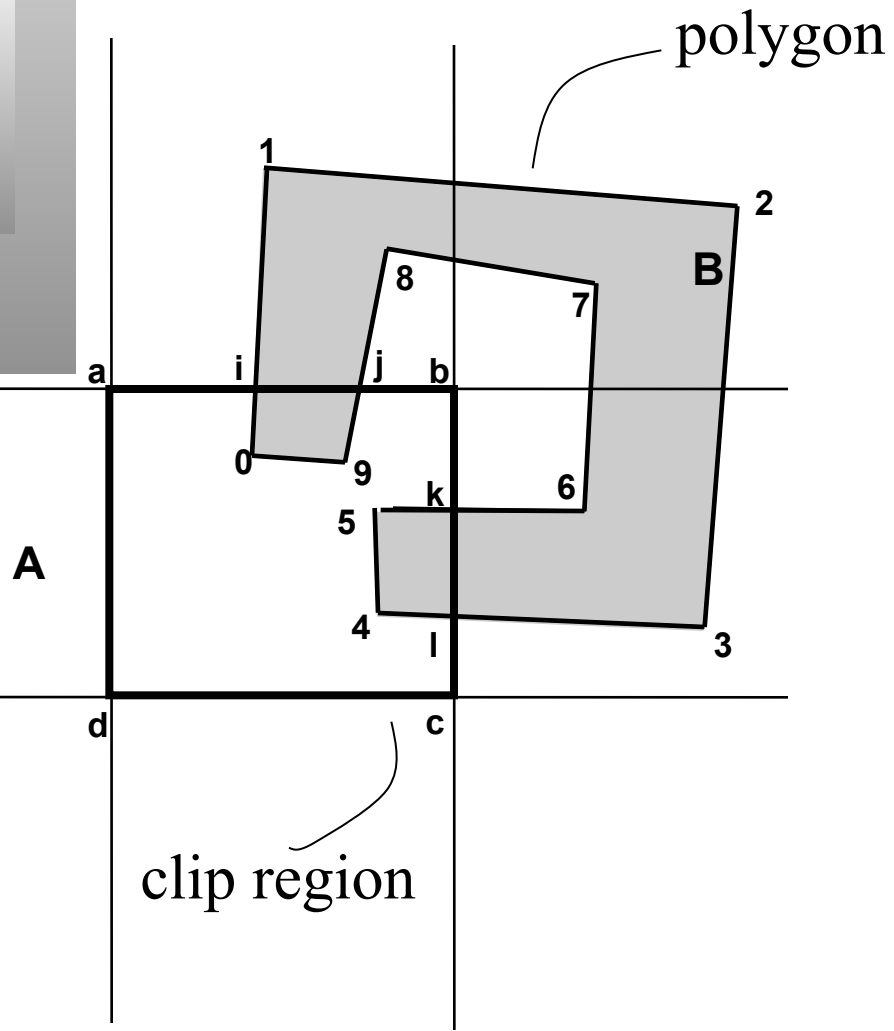
Weiler-Atherton Algorithm

polygon

clip region

loop of polygon vertices

loop of region vertices

Find the intersection vertices and connect them in the two lists

polygon

clip region

Add vertex i:

Find the intersection vertices and connect them in the two lists

polygon

B

A

i   j   b

a

8   7

0   9

k   6

5

4   l   3

d   c

clip region
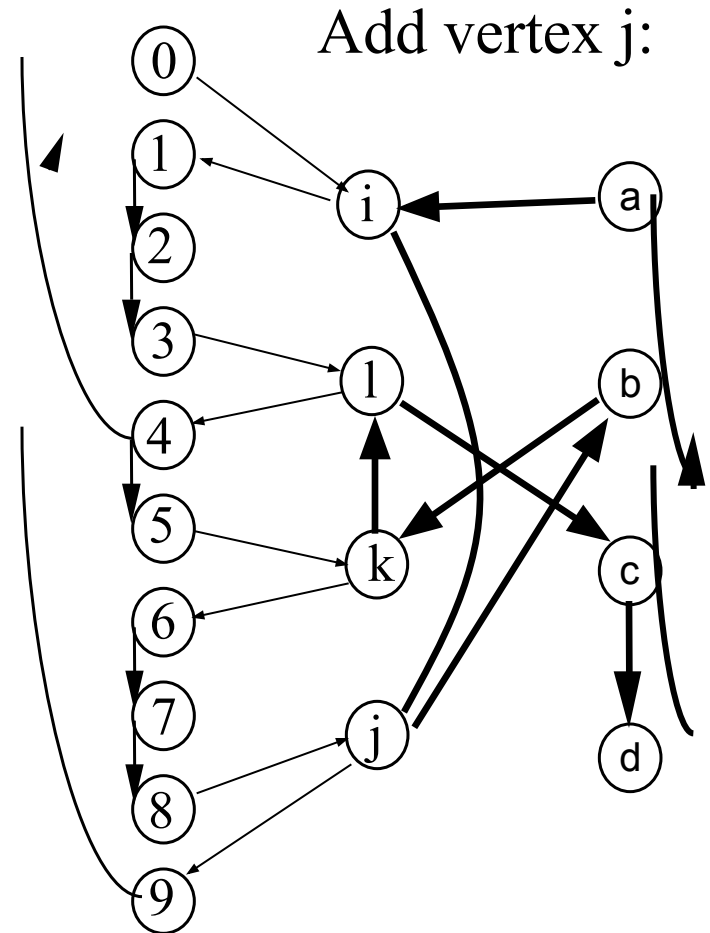
Add vertex l:

0
1
2
3
4
5
6
7
8
9

i   a

l   b

k   c

j   d

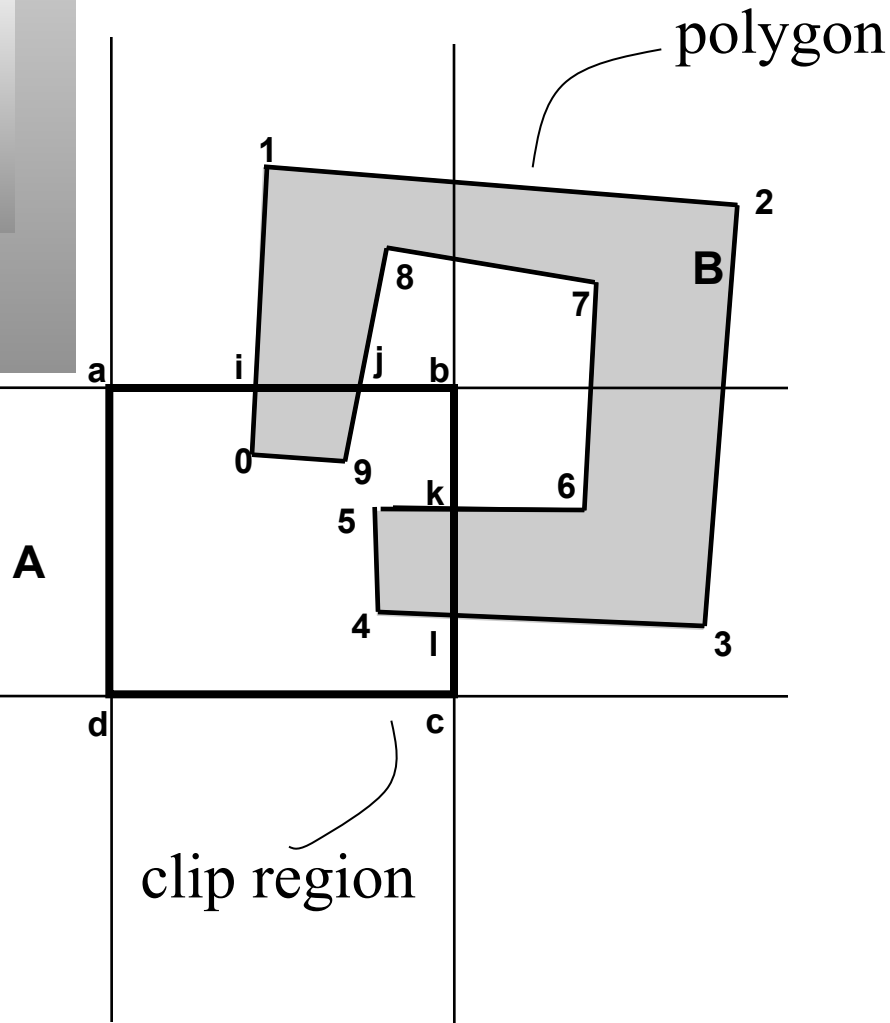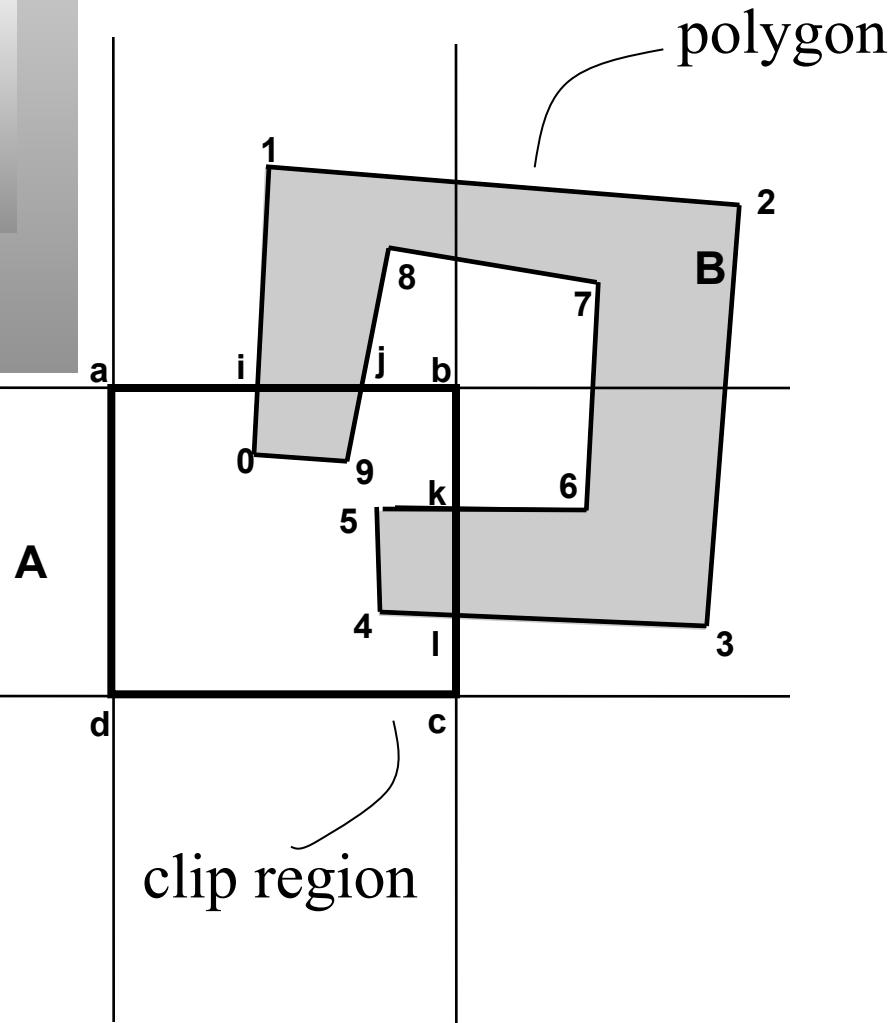# *Find the intersection vertices and connect them in the two lists*

polygon

Add vertex k:

clip region

Completed Loop

polygon

B

A

clip region

Add vertex j:

Classify each intersection vertex as Entering *or* Leaving
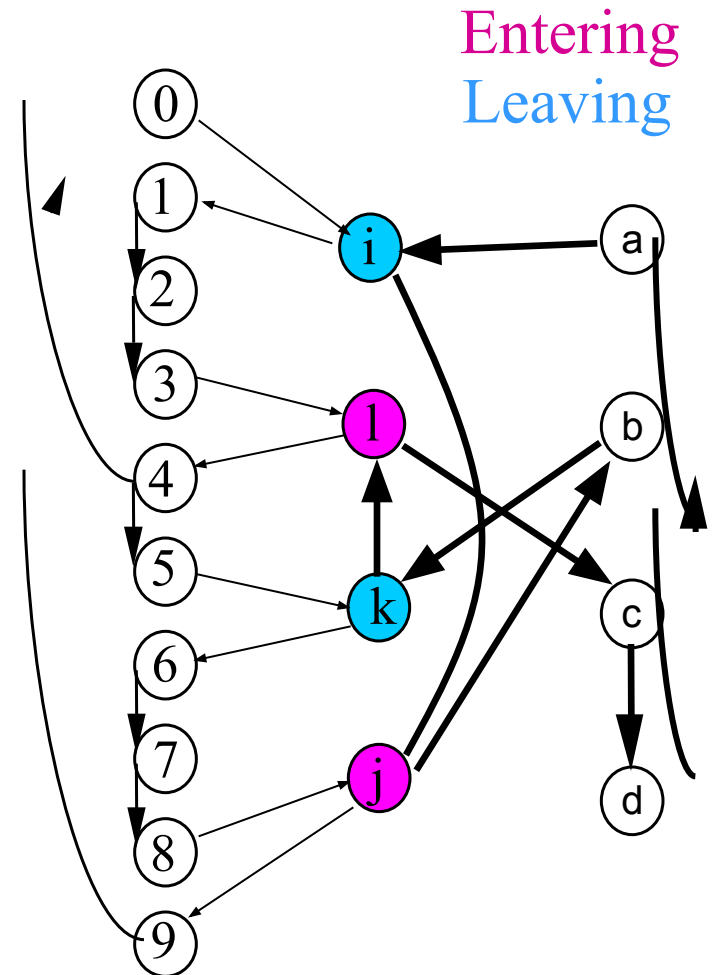
Capture clipped polygons
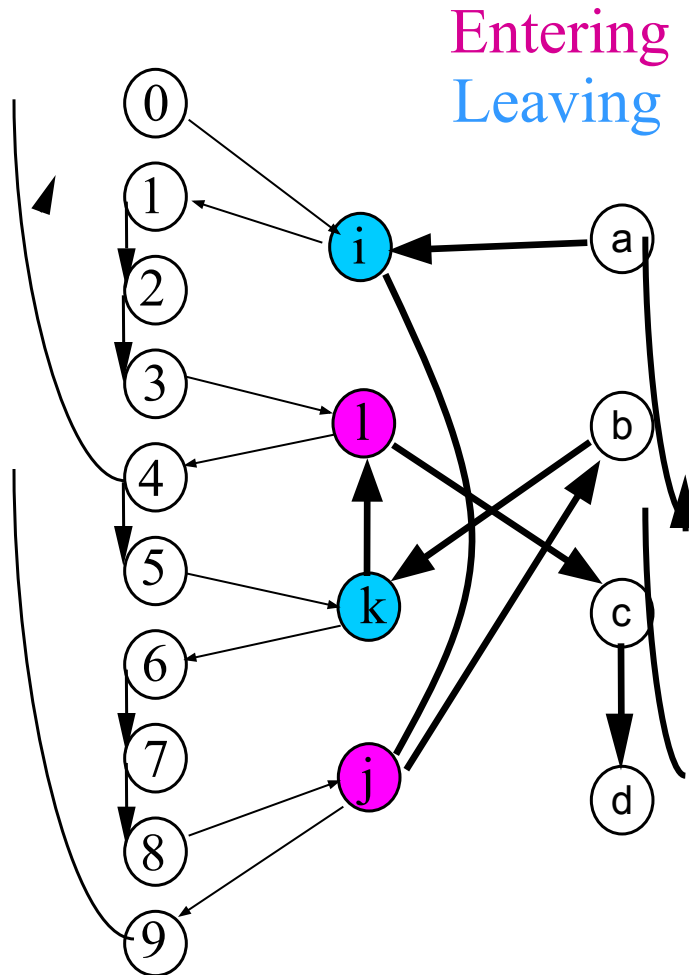
Entering
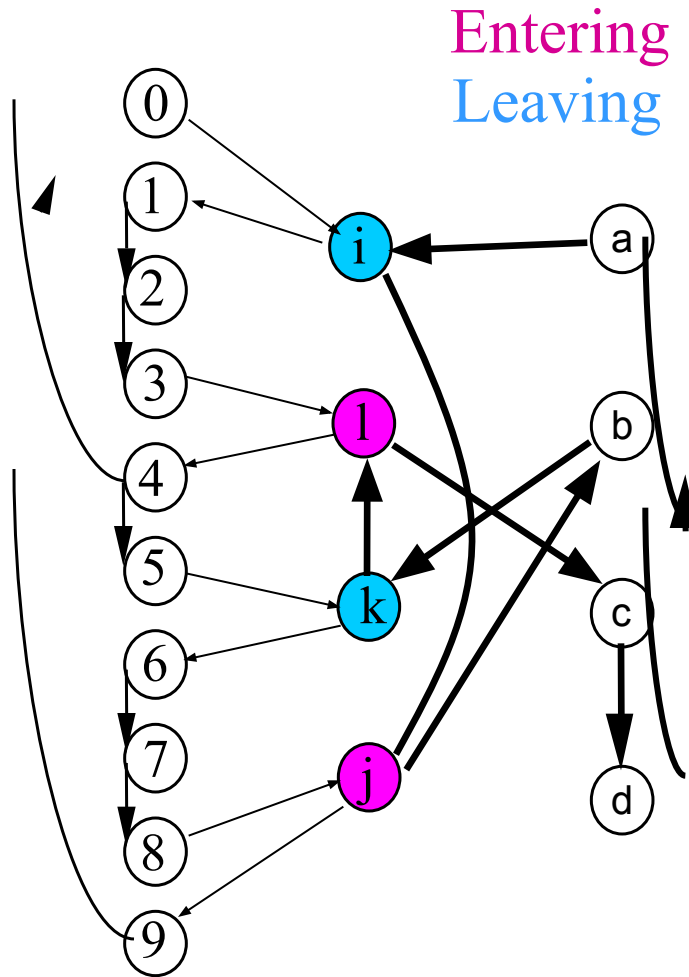Leaving

- *Start at an entering vertex*
- *If you encounter a leaving vertex swap to right hand (clip polygon) loop*
- *If you encounter an entering vertex swap to left hand (polygon) loop*
- *A loop is finished when you arrive back at start*
- *Repeat whilst there are entering vertices*

Capture clipped polygons

Entering
Leaving

- *Loop 1:*
  - *L, 4, 5, K*
- *Loop 2:*
 - *J, 9, 0, i*

Clipping Polygons in 3D

- *The Sutherland-Hodgman can easily be extended to 3D*
  - *the clipping boundaries are 6 planes instead of 4 lines*
  - *intersection calculation is done by comparing an edge to a plane instead of edge to edge*
- *It can either be done in Projection Space or in Canonical Perspective*

Clipping in Projection Space

- *The view volume is defined by:*

$$-1 \leq x \leq 1$$

$$-1 \leq y \leq 1$$

$$-1 \leq z \leq 1$$

- *Testing for the 4 cases is fast, for example for the x = 1 (right) clip plane:*
  - *$x_0 \leq 1$ and $x_1 \leq 1$ entirely inside*
  - *$x_0 \leq 1$ and $x_1 > 1$ leaving*
  - *$x_0 > 1$ and $x_1 \leq 1$ entering*
  - *$x_0 > 1$ and $x_1 > 1$ entirely outside*

Clipping in Canonical Perspective

- *When we have an edge that extends from the front to behind the COP, then if we clip after projection (which in effect is what the PS does) we might get wrong results*
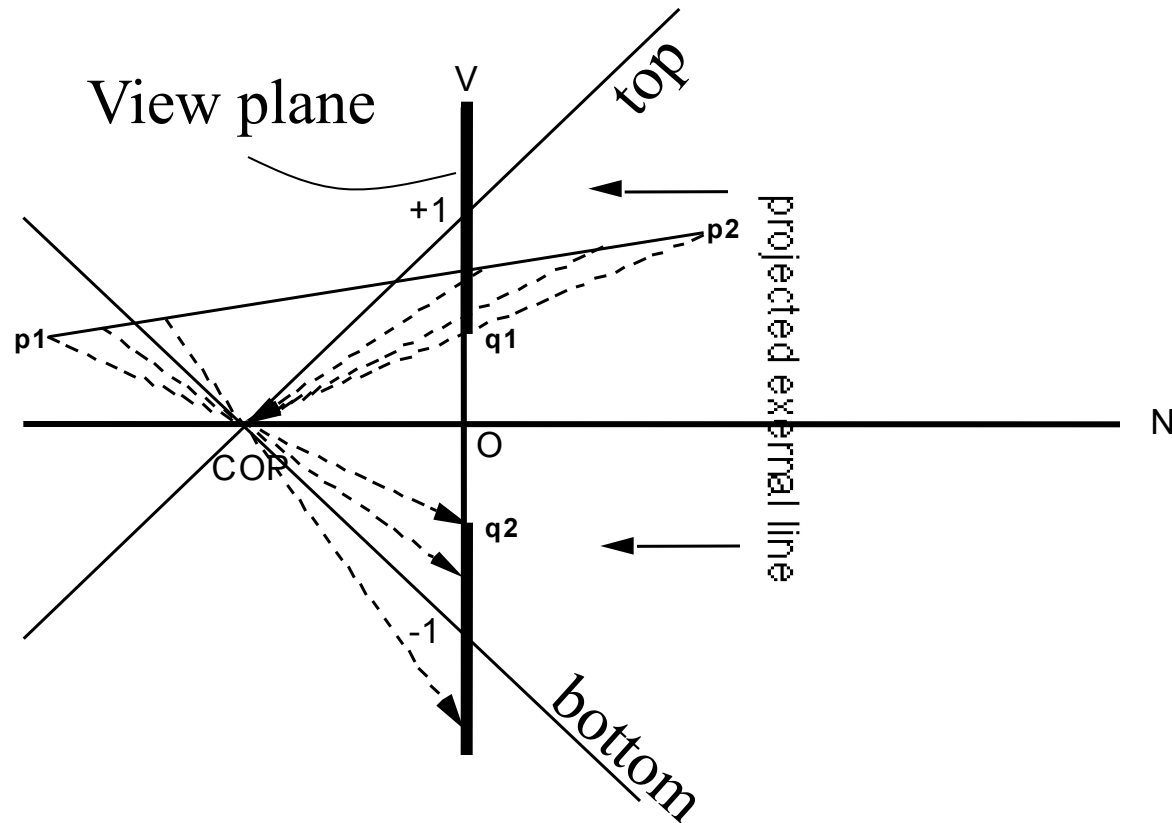
Clipping in Homogeneous Coord.

- *The Sutherland-Hodgman can also be used for clipping in 4D before dividing the points by the* w

- *This can have the advantage that is even more general, it even allows for the front clip plane to be behind the COP*

Clipping Recap

- *Sutherland-Hodgman is simple to describe but fails in certain cases*
- *Weiler-Atherton clipping is more robust but considerably harder*
- *Both extend to 3D but we need to consider projection and end up clipping in 4D*