# Introduction

The algorithm begins with an input list of all vertices in the subject polygon. Next, one side of the clip polygon is extended infinitely in both directions, and the path of the subject polygon is traversed. Vertices from the input list are inserted into an output list if they lie on the visible side of the extended clip polygon line, and new vertices are added to the output list where the subject polygon path crosses the extended clip polygon line.
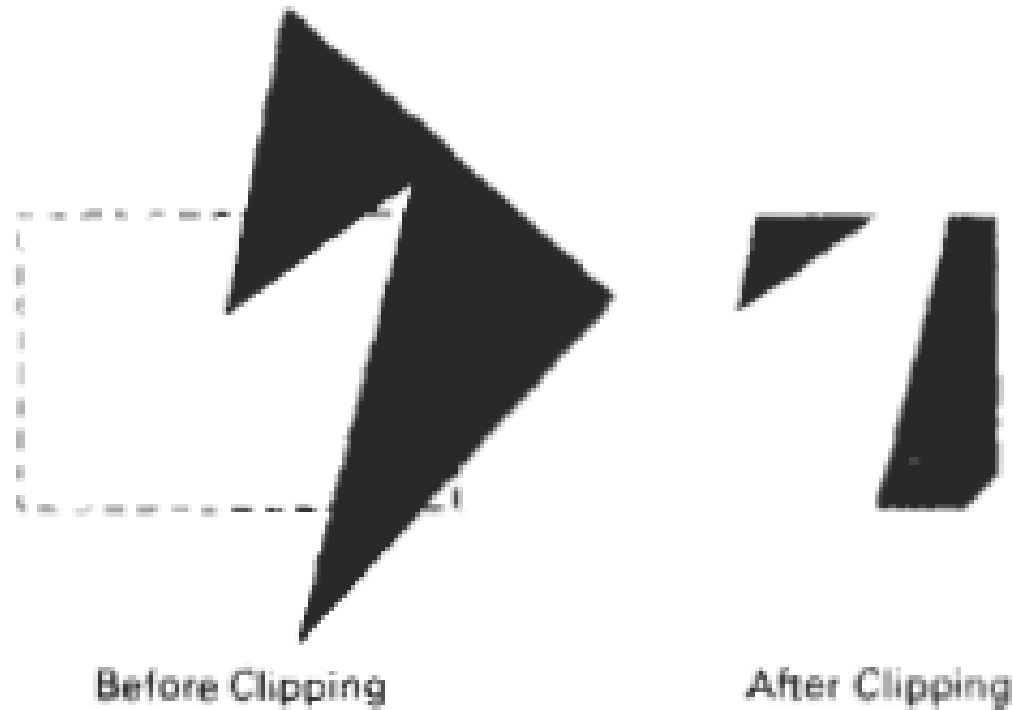
Clipping Polygon

To clip polygons, **we** need to modify the line-clipping.

A polygon boundary processed with a line clipper may be

displayed as a series of unconnected line segments (Fig. 6-17),
depending on the orientation of the polygon to the clipping
window.

Before Clipping   After Clipping
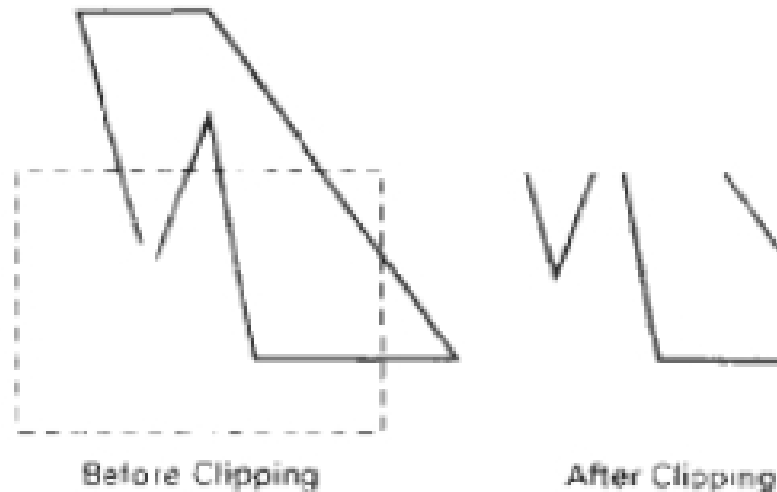
What we want to display is a bounded area after clipping.

For polygon clipping, we require an algorithm that wiIl generate one or more closed areas that are then scan converted.

Before Clipping

After Clipping

Sutherland-Hodgenial1 polygon Clipping

clip a polygon by processing the polygon boundary as a whole against each window edge.

Process all polygon vertices against each clip rectangle boundary.

Beginning with the initial set of polygon vertices, first clip the polygon against the left rectangle boundary to produce a new sequence of vertices. The new set of vertices could then be successively passed to a right boundary clipper, a bottom boundary clipper, and a top boundary clipper.

Original
Polygon

Clip
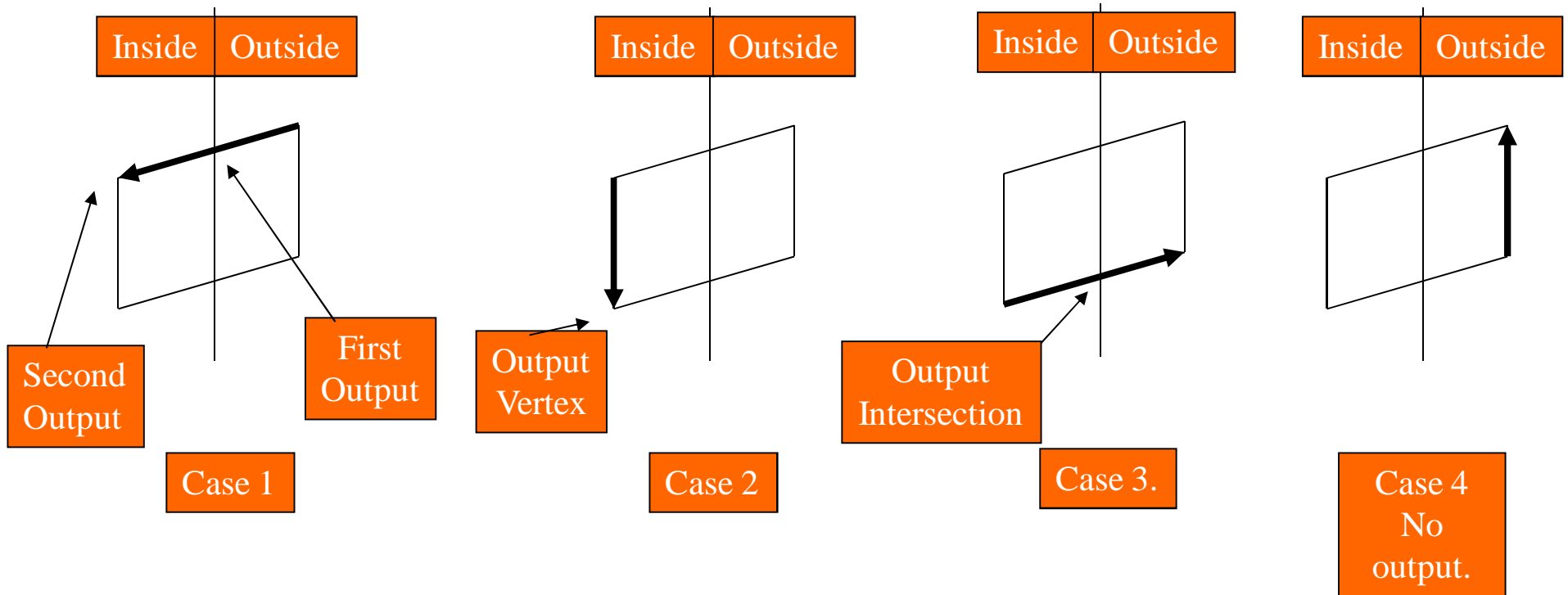Left

Clip
Right

Clip
Bottom

Clip
Top

There are four possible cases.

(1) If the first vertex is outside the window boundary and the second vertex is inside, both the intersection point of the polygon edge with the window boundary and the second vertex are added to the output vertex list.

(2) If both input vertices are inside the window boundary, only the second vertex is added to the output vertex list.

(3) It the first vertex is inside the inside the window boundary and the second vertex is outside, only the edge intersection with the window boundary is added to the output vertex list.

(4) if both input vertices are outside the window boundary, nothing is added to the output list.

Note: Once all vertices have been processed for one clip window boundary, the output list of vertices is clipped against the next window boundary

Sutherland-Hodgman algorithm.
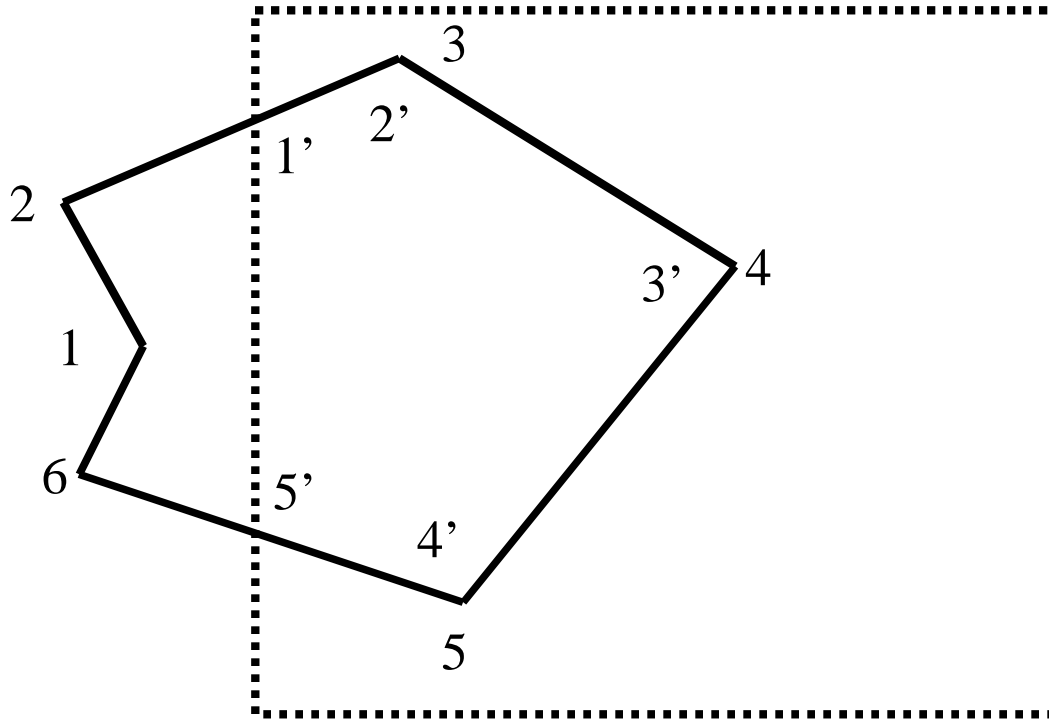
Four cases of polygon clipping:

Inside	Outside

Second Output

First Output

Case 1

Inside	Outside

Output Vertex

Case 2

Inside	Outside

Output Intersection

Case 3.

Inside	Outside

Case 4
No output.

Fig. Clip the polygon against the left boundary of a Window

Application

The **General Polygon Clipper** (**GPC**) is a software library providing for computing the results of clipping operations on sets of polygons. It generalizes the computer graphics clipping problem of intersecting polygons with polygons.

Scope of Research

The Weiler–Atherton clipping algorithm is used in computer graphics. It allows clipping of a *subject polygon* by an arbitrarily shaped *clip polygon* window. It is generally applicable only in 2D. However, it can be used in 3D through visible surface determination and with improved efficiency through Z-ordering.