Assignment 5

Problem Statement:
Implement Cohen Sutherland polygon clipping method to clip the polygon with respect the view-port and window. Use mouse click, keyboard interface

Objective:
To understand the basic concepts of polygon clipping

Outcome:
To implement basic concept of polygon clipping using Cohen Sutherland clipping algorithm

CO Relevance: CO4

PO/PSOs Relevance: PO1, PO2, PO5, PO6

Theory Concepts:

Line Clipping – Cohen Sutherland

In computer graphics, 'line clipping' is the process of removing lines or portions of lines outside of an area of interest. Typically, any line or part thereof which is outside of the viewing area is removed.

The Cohen–Sutherland algorithm is a computer graphics algorithm used for line clipping. The algorithm divides a two-dimensional space into 9 regions (or a three-dimensional space into 27 regions), and then efficiently determines the lines and portions of lines that are visible in the center region of interest (the viewport).

The algorithm was developed in 1967 during flight simulator work by Danny Cohen and Ivan Sutherland The design stage includes, excludes or partially includes the line based on
where:

- Both endpoints are in the viewport region (bitwise OR of endpoints == 0): trivial accept.
- Both endpoints share at least one non-visible region which implies that the line does not cross the visible region. (bitwise AND of endpoints! = 0): trivial reject.
- Both endpoints are in different regions: In case of this nontrivial situation the algorithm finds one of the two points that is outside the viewport region (there will be at least one point outside). The intersection of the outpoint and extended viewport border is then calculated (i.e. with the parametric equation for the line) and this new point replaces the outpoint. The algorithm repeats until a trivial accept or reject occurs.

The numbers in the figure below are called outcodes. The outcode is computed for each of the two points in the line. The outcode will have four bits for two-dimensional clipping, or six bits in the three-dimensional case.

- The first bit is set to 1 if the point is above the viewport. The bits in the 2D outcode represent: Top, Bottom, Right, Left. For example, the outcode 1010 represents a point that is top-right of the viewport.

Note that the outcodes for endpoints must be recalculated on each iteration after the clipping occurs.

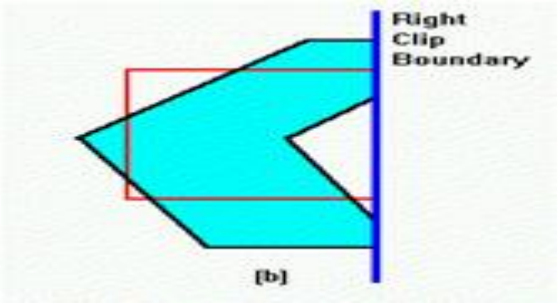
1001	1000	1010
0001	0000	0010
0101	0100	0110

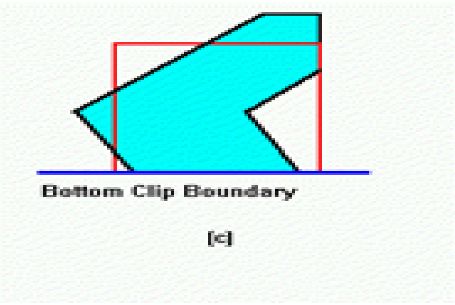
Sutherland Hodgman Polygon Clipping:

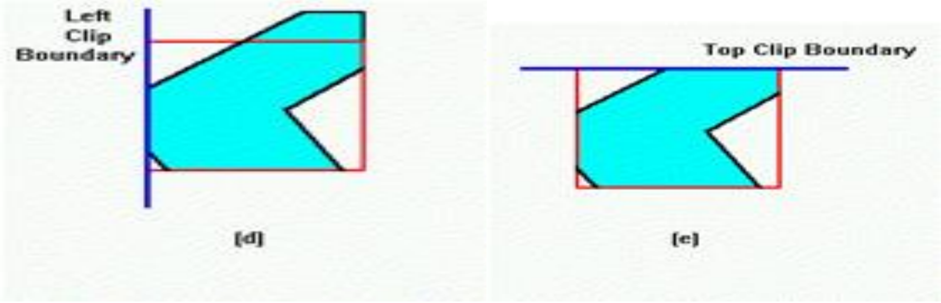
It is used for clipping polygons. It works by extending each line of the convex clip polygon in turn and selecting only vertices from the subject polygon those are on the visible side.

An algorithm that clips a polygon must deal with many different cases. The case is particularly noteworthy in that the concave polygon is clipped into two separate polygons. All in all, task of clipping seems rather complex. Each edge of the polygon must be tested against each edge of the clip rectangle; new edges must be added, and existing edges must be discarded, retained, or divided. 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.

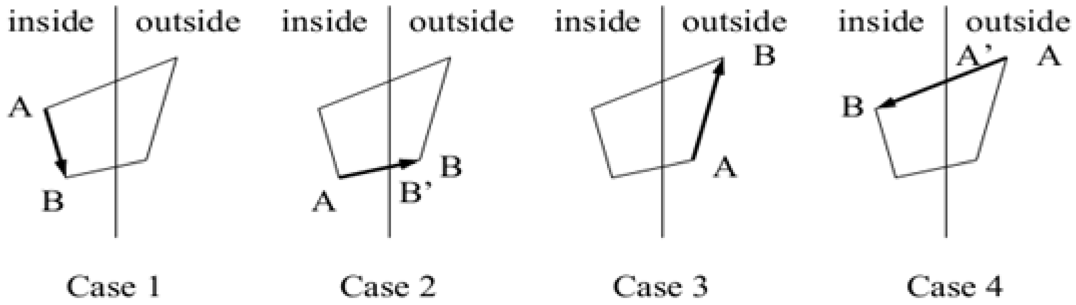
This process is repeated iteratively for each clip polygon side, using the output list from one stage as the input list for the next. Once all sides of the clip polygon have been processed, the final generated list of vertices defines a new single polygon that is entirely visible. Note that if the subject polygon was concave at vertices outside the clipping polygon, the new polygon may have coincident (i.e. overlapping) edges – this is acceptable for rendering, but not for other applications such as computing shadows.







Clipping polygons would seem to be quite complex. A single polygon can actually be split into multiple polygons. The Sutherland-Hodgman algorithm clips a polygon against all edges of the clipping region in turn. The algorithm steps from vertex to vertex, adding 0, 1, or 2 vertices to the output list at each step.



The Sutherland-Hodgeman Polygon-Clipping Algorithms clips a given polygon successively against the edges of the given clip-rectangle. These clip edges are denoted with e1, e2, e3, and e4, here. The closed polygon is represented by a list of its verteces (v1 to vn; Since we got 15 verteces in the example shown above, vn = v15.).

Output:

(Execute the program and attach the printout here)

Conclusion:

In This way we have studied that how to clip polygon using cohen Sutherland clipping algorithm

Viva Questions:

1. What is clipping
2. Explain Cohen Sutherland line clipping algorithm
3. Explain Sutherland hodgman polygon clipping algorithm
4. Exlpain region code

Date:	
Marks obtained:	
Sign of course coordinator:	
Name of course Coordinator :	