Kathmandu University
Department of Computer Science and Engineering
Dhulikhel, Kavre

COMP 342

Lab Report 5

Submitted by

Ashish kumar khatri
Roll No: 28
CE 3rd Year/2nd Sem

Submitted To,

Mr. Dhiraj Shrestha

Department of Computer Science and Engineering

Programming Language : Javascript

Graphics Library : Opengl through webgl api

Title :

- **Implement Cohen Sutherland Line Clipping algorithm**
- **Implement Sutherland Hodgemann polygon clipping algorithm**

Cohen Sutherland Line Clipping Algorithm

The Cohen-Sutherland line clipping algorithm is a computer graphics algorithm used to efficiently clip lines against a rectangular clipping window.Here's a step-by-step explanation of the Cohen-Sutherland algorithm:

- Define the clipping window: Determine the coordinates of the rectangular clipping window that defines the region within which the lines will be clipped.
- Assign region codes: Assign binary codes to each endpoint of the line segment based on its position relative to the clipping window. The region codes divide the 2D space into nine regions, with three bits representing each region. For example, if the line is to the left of the window, the left bit will be set to 1; if it is to the right, the right bit will be set to 1, and so on.
- Perform visibility test: Perform a visibility test by checking if both endpoints are within the clipping window (i.e., the region codes for both endpoints are 0000). If they are, the line segment is entirely inside the window and can be drawn. If they are not, proceed to the next steps.

- Check trivial rejection: If the logical AND of the two region codes is not 0000 (i.e., at least one bit is set to 1), the line segment is trivially rejected and can be discarded.
- Perform clipping: If the line is not trivially rejected, find the intersection points between the line and the boundaries of the clipping window based on the region codes of the endpoints. The line can intersect with the top, bottom, left, or right boundary of the window.

- Update endpoints: Replace the outside endpoint(s) with the intersection point(s) found in the previous step. Recalculate the region codes for the updated endpoints.
- Repeat steps 3-6: Repeat steps 3 to 6 until the line is either trivially accepted (both endpoints are within the clipping window) or trivially rejected (at least one endpoint is entirely outside the clipping window).
- Draw the clipped line: If the line is trivially accepted, draw the line segment between the updated endpoints. The resulting line segment will be entirely or partially inside the clipping window.

Sutherland Hodgemann polygon clipping algorithm

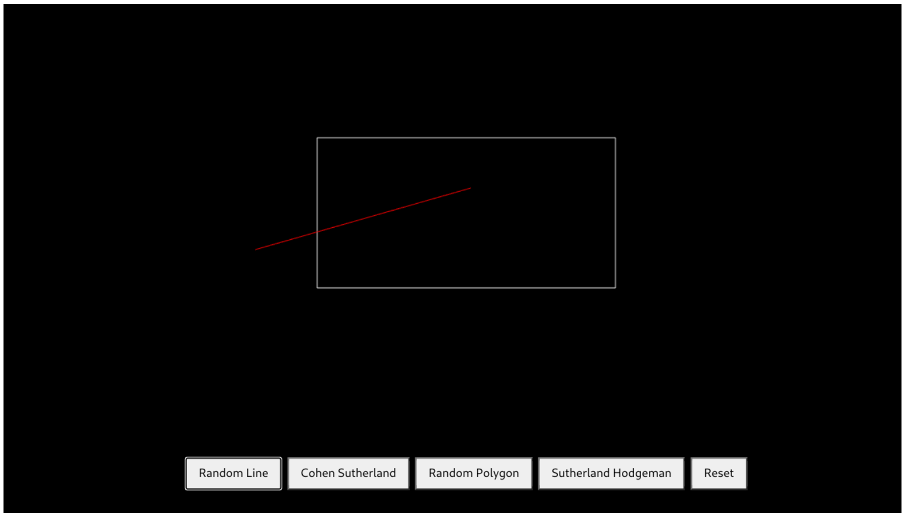
The Sutherland-Hodgman polygon clipping algorithm is a computer graphics algorithm used to clip polygons against a clipping window or another polygon.Here's a step-by-step explanation of the Sutherland-Hodgman algorithm:

- Define the clipping window or polygon: Determine the vertices of the clipping window or polygon that will be used to clip the subject polygon.
- Initialize the output polygon: Start with an empty output polygon.
- Process each edge of the clipping window or polygon: Iterate over each edge of the clipping polygon in a clockwise direction.
- Process each vertex of the subject polygon: For each vertex of the subject polygon, perform the following steps:
    - If the vertex is inside the clipping window or polygon, add it to the output polygon.
    - If the vertex is outside the clipping window or polygon, determine the intersection points between the subject polygon edge (formed by the current vertex and the previous vertex) and the clipping window or polygon edge. Add the intersection point(s) to the output polygon.
    - If the vertex is neither inside nor outside, check if the previous vertex is inside or outside the clipping window or polygon. If it is inside, add the intersection point between the subject polygon edge and the clipping window or polygon edge to the output polygon.
- Repeat step 4 for all vertices of the subject polygon.

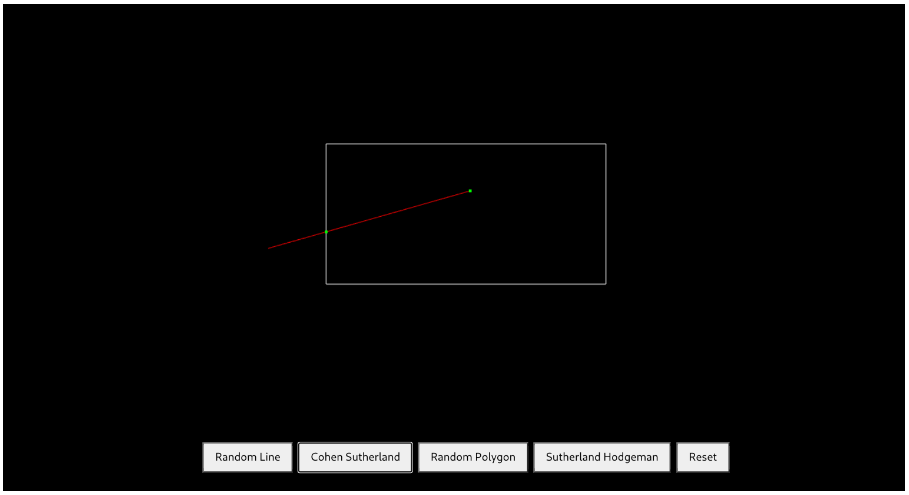
- Repeat steps 3-5 for all edges of the clipping window or polygon.
- Return the output polygon: The resulting output polygon is the clipped portion of the subject polygon.

Screenshots showing the result of each algorithm:

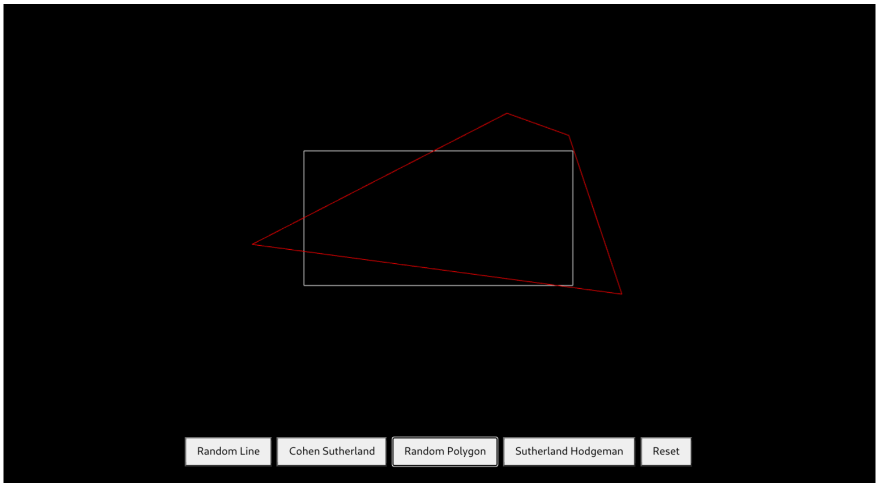
1. Generate random line



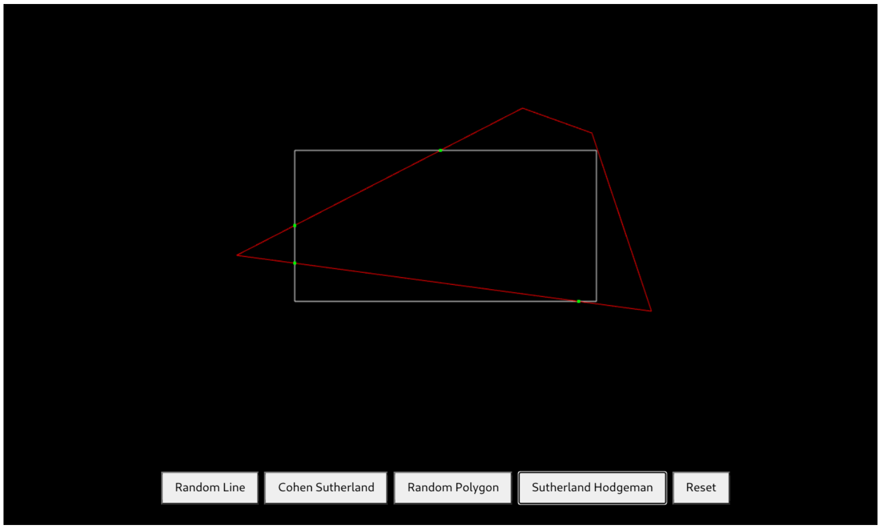
2. Applying cohen sutherland algorithm



3. Generate random polygon



4. Applying sutherland hodgeman algorithm



Conclusion:

Hence, in this lab cohen sutherland line clipping algorithm and sutherland hodgeman polygon clipping algorithm was implemented using webgl in JS. Instead of clipping the polygon, we found at what poin of the the line and polygon will the clipping occur and we highlighted it with green point as in the picture.