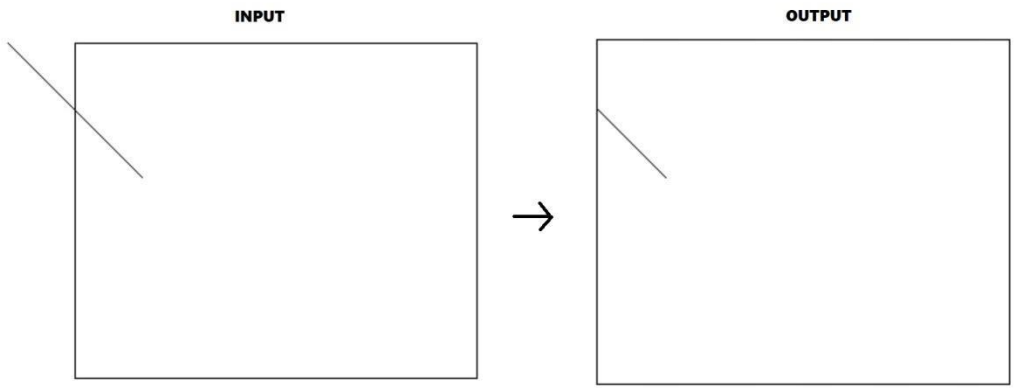
Assignment No. 2

Title	Polygon clipping using Cohen Southerland line clipping algorithm
Aim/Problem Statement	Write C++ program to implement Cohen Southerland line clipping algorithm.
CO Mapped	
Pre-requisite	1. Basic programming skills of C++ 2. 64-bit Open source Linux 3. Open Source C++ Programming tool like G++/GCC
Learning Objective	To learn Cohen Southerland line clipping algorithm.

Theory:

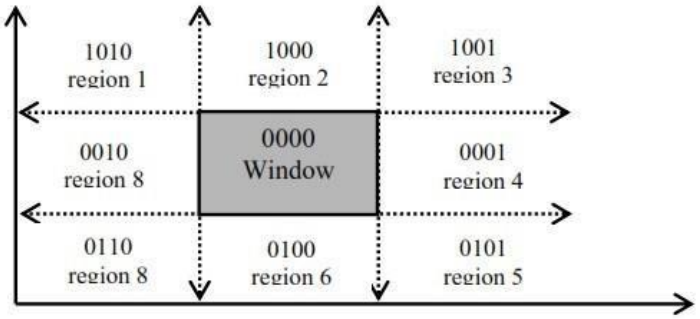
Cohen Sutherland Algorithm is a **line clipping algorithm** that cuts lines to portions which are within a rectangular area. It eliminates the lines from a given set of lines and rectangle area of interest (view port) which belongs outside the area of interest and clip those lines which are partially inside the area of interest.
Example:



Cohen Sutherland Line Clipping Algorithm

Algorithm

The algorithm divides **a two-dimensional space** into **9 regions** (eight outside regions and one inside region) and then efficiently determines the lines and portions of lines that are visible in the central region of interest (the viewport).
Following image illustrates the 9 regions:

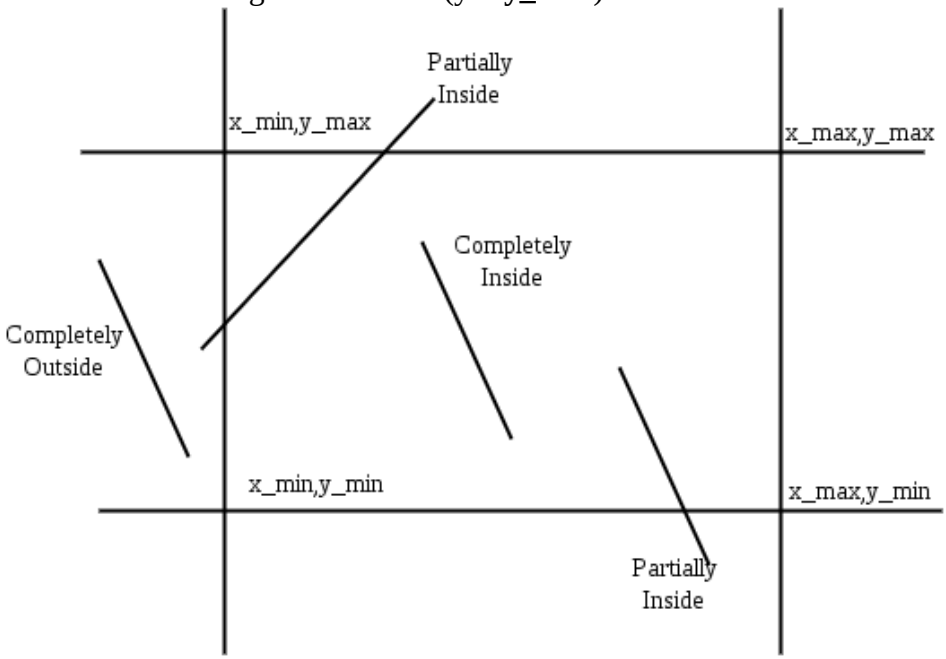


As you seen each region is denoted by a 4 bit code like 0101 for the bottom right region

Four Bit code is calculated by comparing extreme end point of given line (x,y) by four co-ordinates x_min, x_max, y_max, y_min which are the coordinates of the area of interest (0000)

Calculate the four bit code as follows:

- Set First Bit if 1 Points lies to left of window (**x < x_min**)
- Set Second Bit if 1 Points lies to right of window (**x > x_max**)
- Set Third Bit if 1 Points lies to left of window (**y < y_min**)
- Set Forth Bit if 1 Points lies to right of window (y > **y_max**)



The more efficient Cohen-Sutherland Algorithm performs initial tests on a line to determine whether intersection calculations can be avoided.

Pseudocode

- **Step 1** : Assign a region code for two endpoints of given line
- **Step 2** : If both endpoints have a region code 0000 then given line is completely inside and we will keep this line.
- **Step 3:** If step 2 fails, perform the logical AND operation for both region codes.
    - **Step 3.1:** If the result is not 0000, then given line is completely outside.
    - **Step 3.2 :** Else line is partially inside.
        - **Step 3.2.a :** Choose an endpoint of the line that is outside the given rectangle.
        - **Step 3.2.b :** Find the intersection point of the rectangular boundary(based on region code)
        - **Step 3.2.c :** Replace endpoint with the intersection point and upgrade the region code.

- **Step 3.2.d :** Repeat step 2 until we find a clipped line either trivially accepted or rejected.

- **Step 4:** Repeat step 1 for all lines.

Conclusion:

Questions:

1. What is the limitation of Cohen Sutherland Line Clipping algorithm?
2. What are the Advantage of Cohen Sutherland Line Clipping?