National University of Sciences and Technology (NUST)
School of Electrical Engineering and Computer Science

Department of Computing

CS361: Computer Graphics

Class: BSCS-2AB & BESE-3AB

Lab 11: Line clipping Algorithm

Date: 21 December, 2015

Time: 2:00pm- 5:00pm

Instructor: Dr. Muhammad Muddassir Malik

Lab 11: Line clipping Algorithm

Introduction

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.

There are two common algorithms for line clipping: Cohen–Sutherland and Liang–Barsky.

A line clipping method consists of various parts first we can tests a giving line segment to find out whether it lies entirely outside the window finally of we cannot recognizes a line as entirely inside or outside the window . We must does intersection calculations with one or more clipping boundaries.

We process line the insides ,outside test by checking the line endpoints. A line with both endpoint inside all clipping boundaries such as the line from inside the box is saved a line with both endpoints outside one of the clipping boundaries is outside the windows all other line cross alone or more clipping boundaries and may need calculations of multiple intersection point

Objective

Objective of this lab is to understand and implement line clipping algorithm

Tools/Software Requirement

Description

Concept

It is desirable to restrict the effect of graphics primitives to a subregion of the canvas, to protect other portions of the canvas. All primitives are clipped to the boundaries of this **clipping rectangle**; that is, primitives lying outside the clip rectangle are not drawn.

The default clipping rectangle is the full canvas (the screen), and it is obvious that we cannot see any graphics primitives outside the screen.

Line Clipping

This section treats clipping of lines against rectangles. Although there are specialized algorithms for rectangle and polygon clipping, it is important to note that other graphic primitives can be clipped by repeated application of the line clipper.

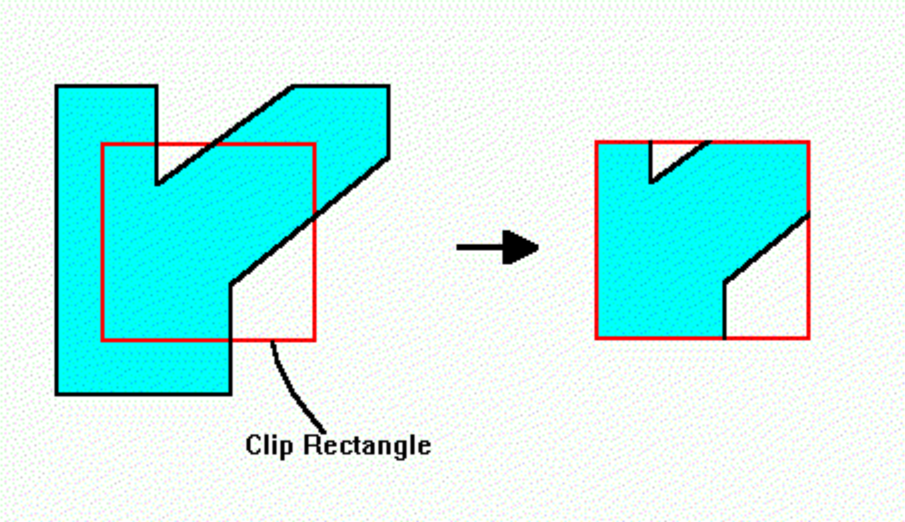
1. Clipping Individual Points

Before we discuss clipping lines, let's look at the simpler problem of clipping individual points.

If the x coordinate boundaries of the clipping rectangle are Xmin and Xmax, and the y coordinate boundaries are Ymin and Ymax, then the following inequalities must be satisfied for a point at (X,Y) to be inside the clipping rectangle:

$$\text{Xmin} < X < \text{Xmax}$$
$$\text{and Ymin} < Y < \text{Ymax}$$

If any of the four inequalities does not hold, the point is outside the clipping rectangle.



Lab Task

Create a program that takes a clip window and a line as arguments and using Line Clipping Algorithm, outputs a clipped line.

Deliverable

Upload your code with snap shots of the output.