COMPUTER GRAPHICS
UNIT-III

R.MANIMEGALAI

DEPARTMENT OF COMPUTER SCIENCE

PERIYAR GOVT ARTS COLLEGE

CUDDALORE.

Two Dimensional Viewing
and Clipping.

Two-Dimensional Viewing and Clipping

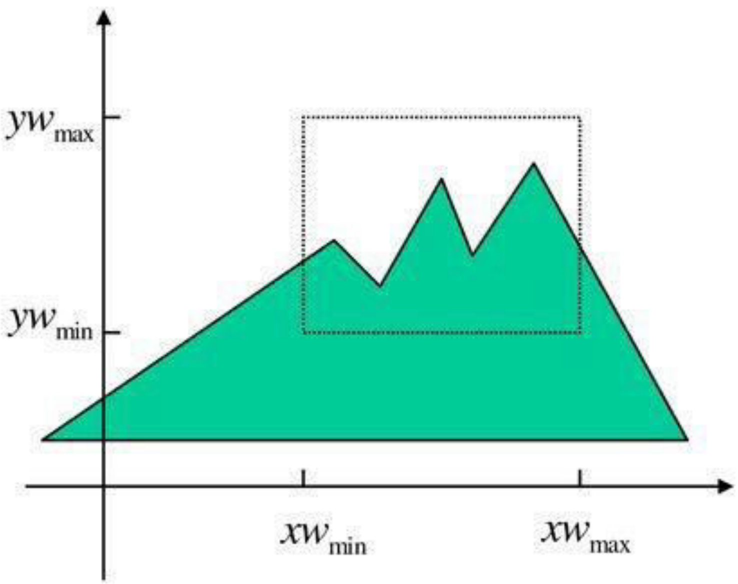
1) Much like what we see in real life through a small window on the wall or the viewfinder of a camera, a computer generated image often depicts a partial view of a large scene.

2) Objects are placed in to scenes by modelling transformations to a master co-ordinate system, commonly referred to as world co-ordinate system.

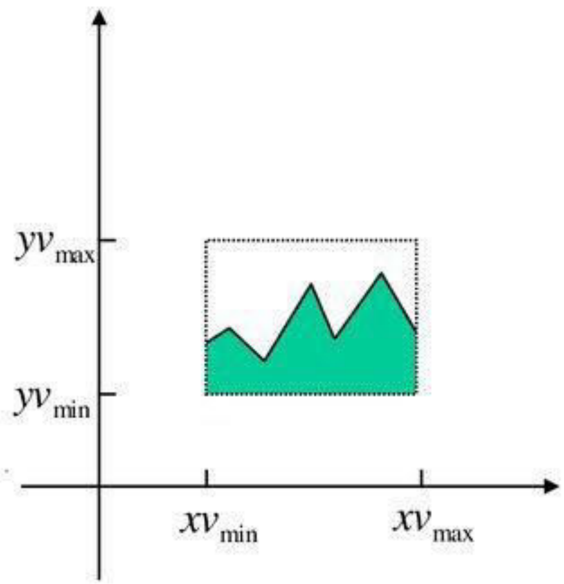
3) An image Representing a view often becomes part of a larger image, like a photo on an album page, which models a computer monitors display area. Since the monitor sizes differ from one system to another, we want to introduce a device independent tool to describe a display area . This tool is called normalized device co-ordinate system.

4) The process that converts object co-ordinate in WCS to normalized device co-ordinates is called window to viewport mapping

5) A world co-ordinate area selected for display is called window. An area on a diplay deivice to which a window is mapped is called a viewport.



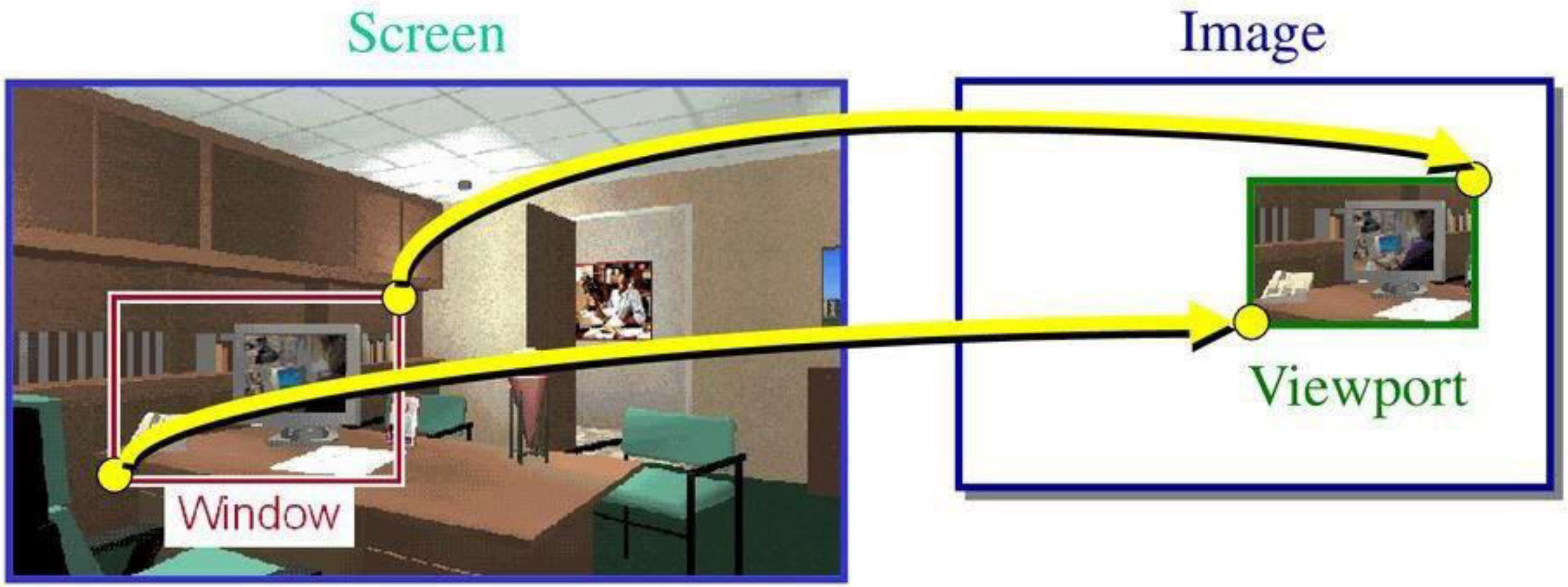
world coordinates



device coordinates

Viewport Transformation

- Transform 2D Geometric Primitives from Screen Coordinate System (Projection Coordinates) to Image Coordinate System (Device Coordinates)



Window vs. Viewport

- Window
  - World-coordinate area selected for display
  - What is to be viewed
- Viewport
  - Area on the display device to which a window is mapped
  - Where it is to be displayed

Window to viewport Mapping

1) A window is specified by four world co-ordinates : Wxmin, Wxmax, Wymin, Wy max.

2) Similarly a viewport is specified by four normalized device co-ordinate : Vxmin, Vxmax, Vymin, Vymax.

3) The objective of window to viewport mapping is to convert the world co-ordinates (Wx,Wy) of an arbitrary point to its corresponding normalized device co-ordinate(Vx,Vy).

$$Wx-Wxmin/Wxmax-Wxmin = Vx-Vxmin/Vxmax-Vxmin$$

$$Wy-Wymin/Wymax-Wymin= Vy- Vymin/Vymax-Vymin$$

Thus $Vx=Vxmax-Vxmin/Wxmax-Wxmin(Wx-Wxmin)+Vxmin$

$$Vy=Vymax-Vymin/Wymax-Wymin(Wy-Wymin) + Vymin.$$

- The above Equation can also be derived with a set of transformation theta converts the window area in to the viewport area.

1) Perform a scaling transformation using fixed point position that scales the window area to the size of viewport.

2) Translate the scaled window area to the position of the viewport.

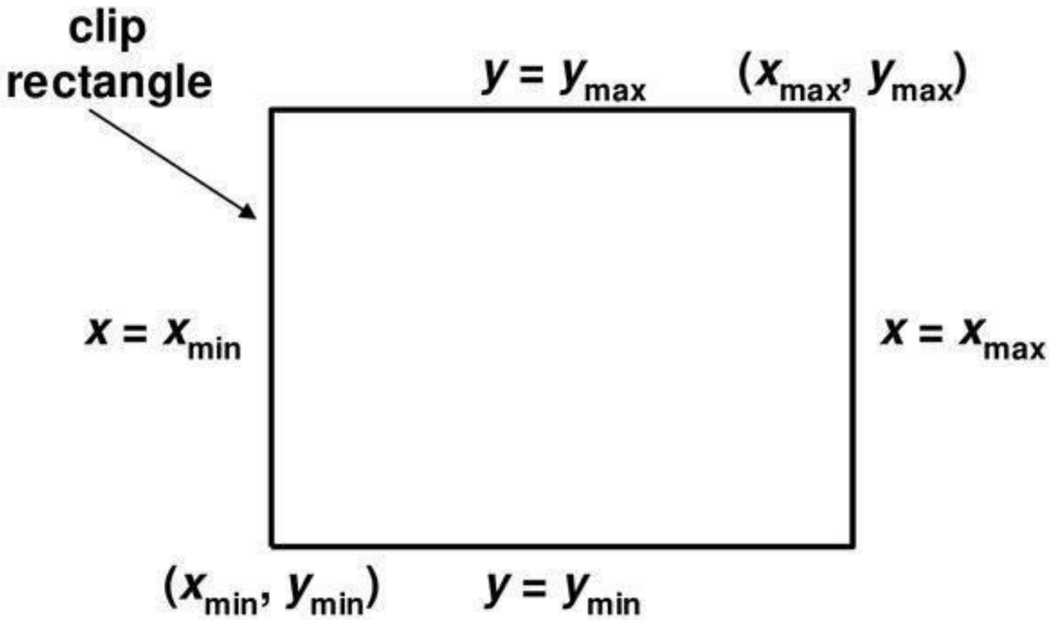
3) Sx=vxmax-vxmin/wxmax-wxmin

4) Sy=vymax-vymin/Wymax-Wymin

Clipping: The Clipping operation eliminates objects or portions of objects that are not visible through .the window

1) Point Clipping:
2) Line Clipping
   - Cohen Sutherland Algorithm (Line)
   - Mid-Point Sub Division Algorithm
3) Polygon Clipping
   - Sutherland-Hodgeman Algorithm (Polygon)
   - Weiler Atherton Algorithm
4) Text Clipping

Point Clipping

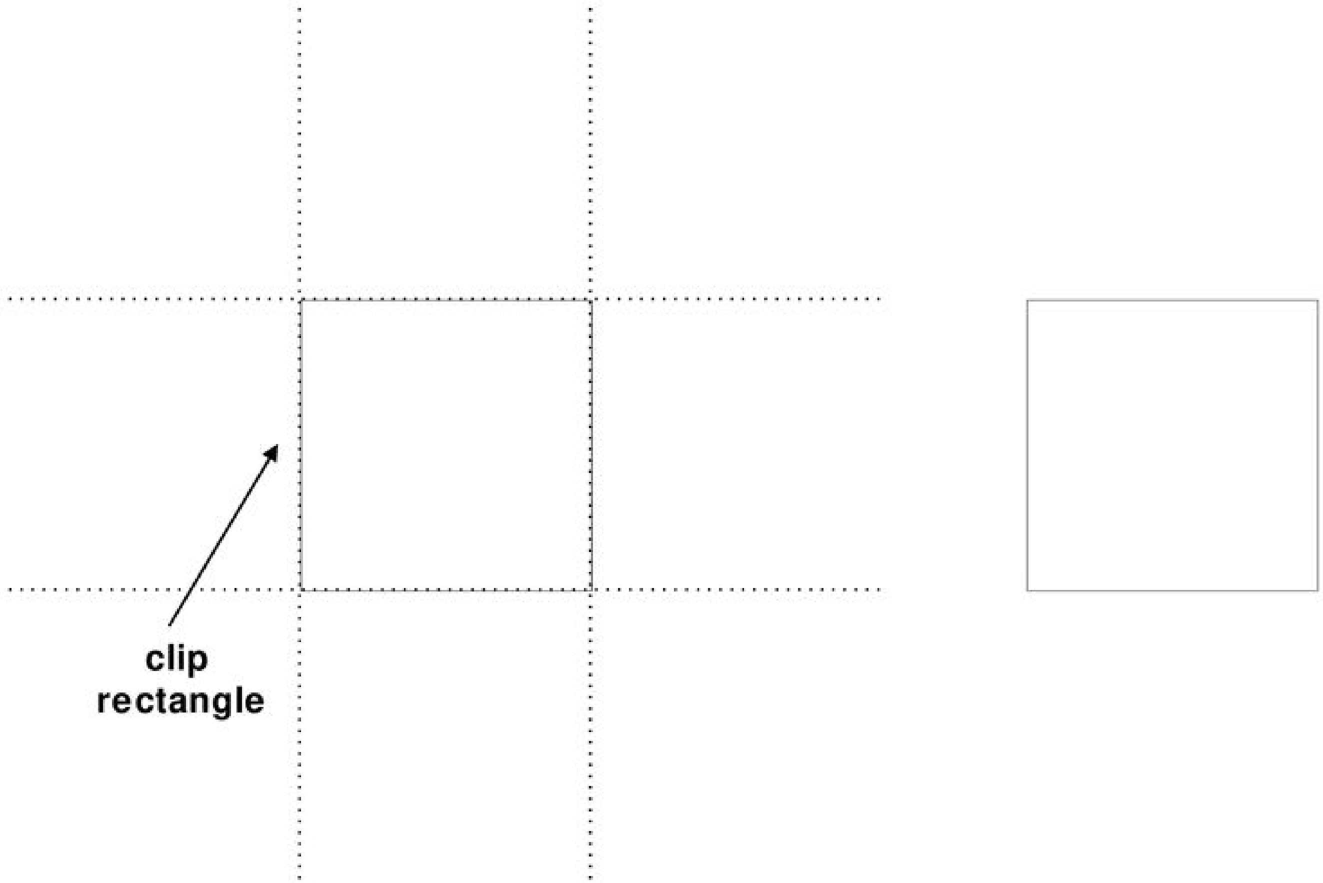


For a point (x,y) to be inside the clip rectangle:

$$x_{min} \leq x \leq x_{max}$$

$$y_{min} \leq y \leq y_{max}$$

Line Clipping



Cases for clipping lines

Cohen-Sutherland Algorithm

- In this algorithm we divide the line Clipping Process in to Two Phases.

1) Identify Those Lines which intersect the clipping window and so need to be clipped.

2) Perform the Clipping.

All Lines Fall in to one of the following Categories:

1) Visible: Both the end points of the line Lie with in the window.

2) Not Visible: The Line Definitely Lies outside the window. This will occur if the line from (x1,y1) to (x2,y2) satisfies any one of the following four inequalities.

   x1,x2>Xmax , x1,x2 <Xmin, y1,y2> Ymax , y1,y2<Ymin.

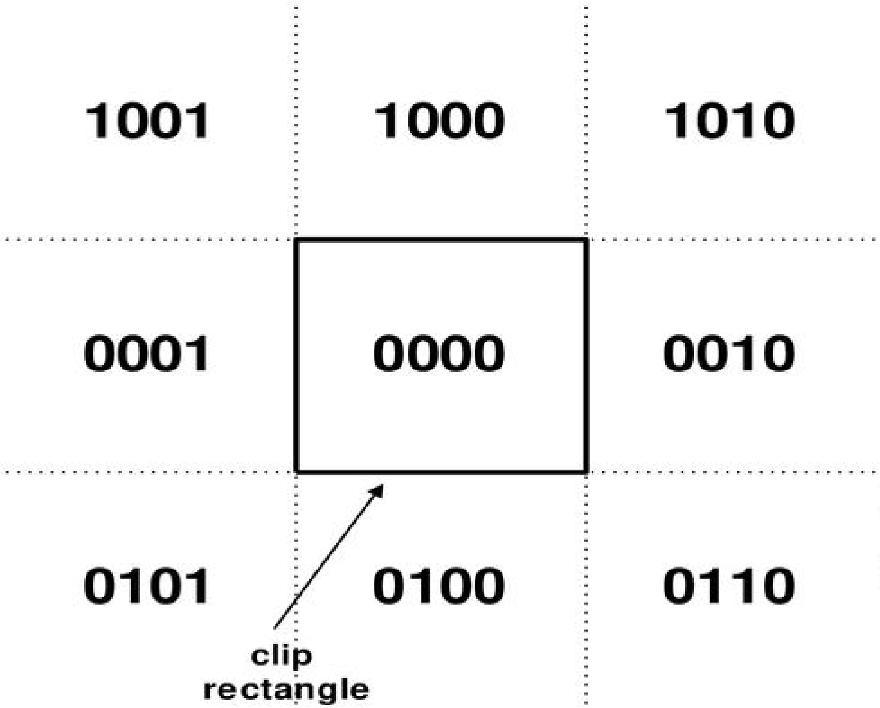
3) Clipping Candidate: The Line is in neither category 1 nor 2.

- The algorithm employs an efficient procedure for finding the category of a line. It proceeds in Two Steps.

1) Assign a 4-bit Code to each endpoint of the line. The code is determined according to which of the following nine regions of the plane the endpoint lies in.

bit 0: $y > y_{\max}$

bit 1: $y < y_{\min}$



bit 2: $x > x_{ma}$

bit 3: $x < x_{mi}$

Region outcodes

Starting From the left most bit each bit of the code is set to true(1) or false(0) according to scheme.

- Bit1=endpoint is above the window=sign(y-ymax)
- Bit2=endpoint is below the window=sign(ymin-y)
- Bit3=endpoint is to the right of the window=sign(x-xmax)
- Bit4=endpoint is to the left of the window=sign(xmin-x)

We use the convention that sign(a)=1 if a is positive , 0 otherwise.

2) **The Line is Visible if both the region codes are 0000**

Not Visible if the bitwise logical AND of the codes is Not 0000.

Candidate for Clipping if the bitwise Logical AND of the region Codes is 0000.

- For the line in Category 3 we proceed to find the intersection point of the line with one of the boundries of the clipping window.

- If bit 1and 2 is 1 then intersect with Line y=ymax and y= ymin.

- If bit 3and 4 is 1 then intersect with Line x=xmax and x= xmin.

- The co-ordinates of the intersection points are:

  – xi= xmin or xmax (if the boundary line is vertical:

  – Yi=y1+m(xi-x1);

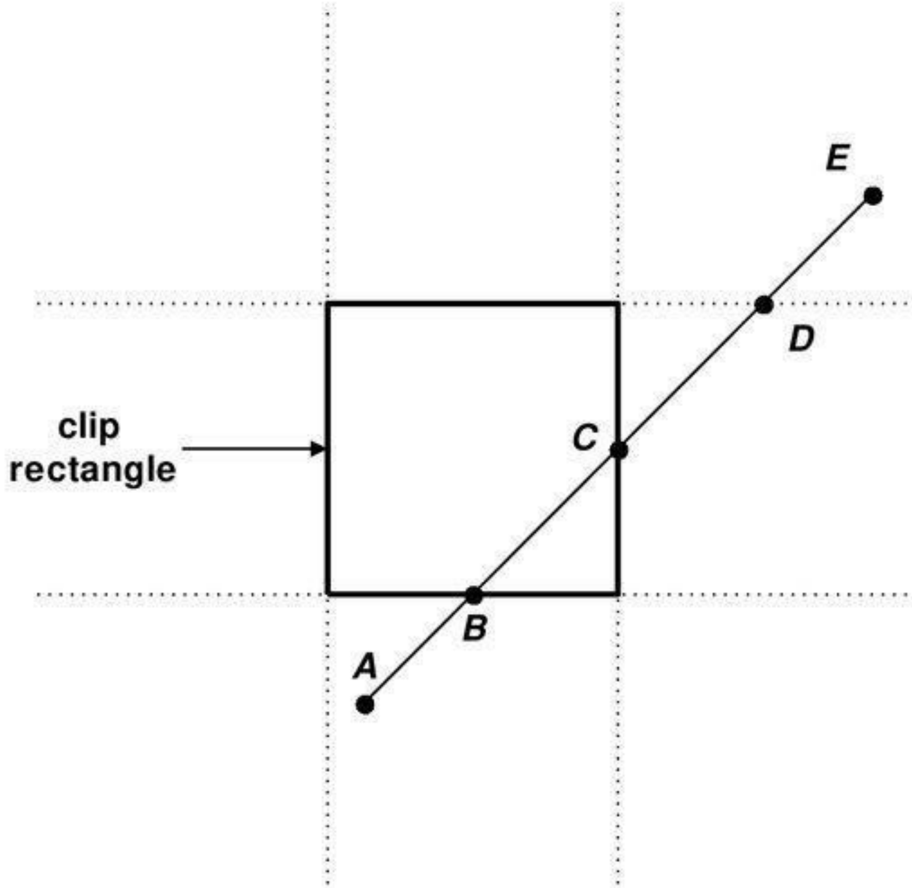
  Or

  xi= x1+(yi-y1)/m (if the boundary line is horizontal)

  Yi=ymin or ymax;

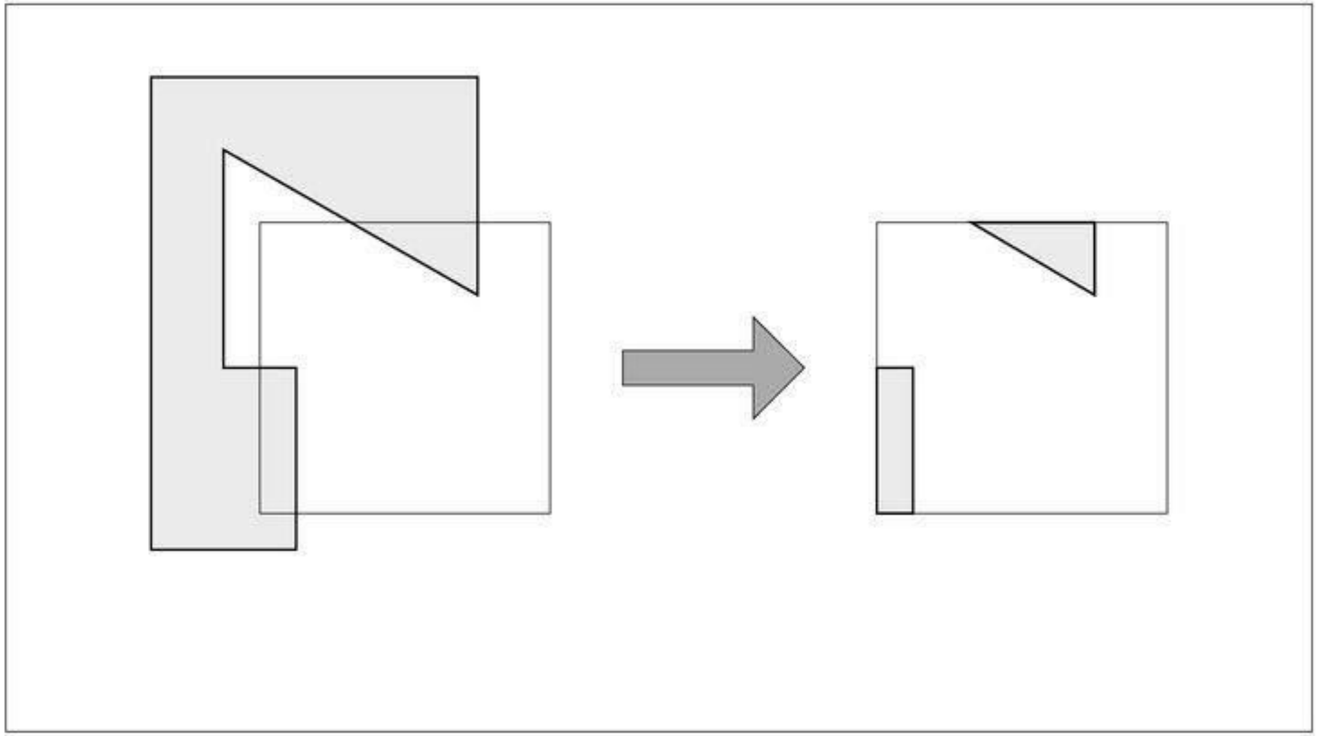
  Where m is slope = (y2-y1)/(x2-x1)

Cohen-Sutherland Algorithm



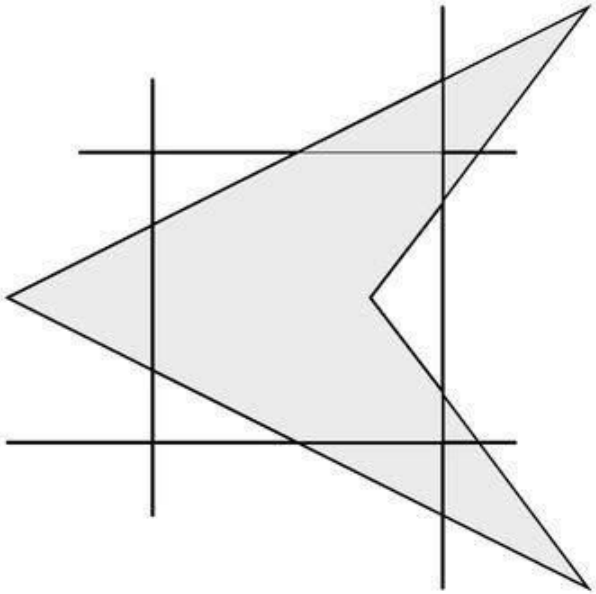
An Example

Polygon Clipping



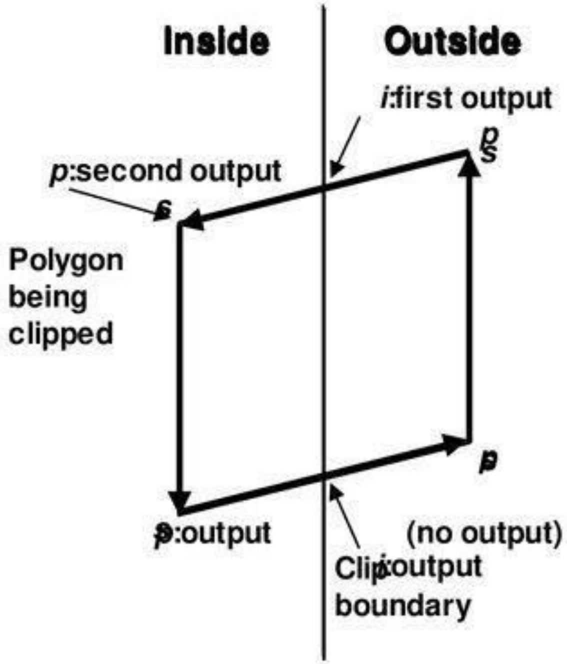
Example

Sutherland-Hodgeman Algo.



Initial Condition

4 Cases of Polygon Clipping

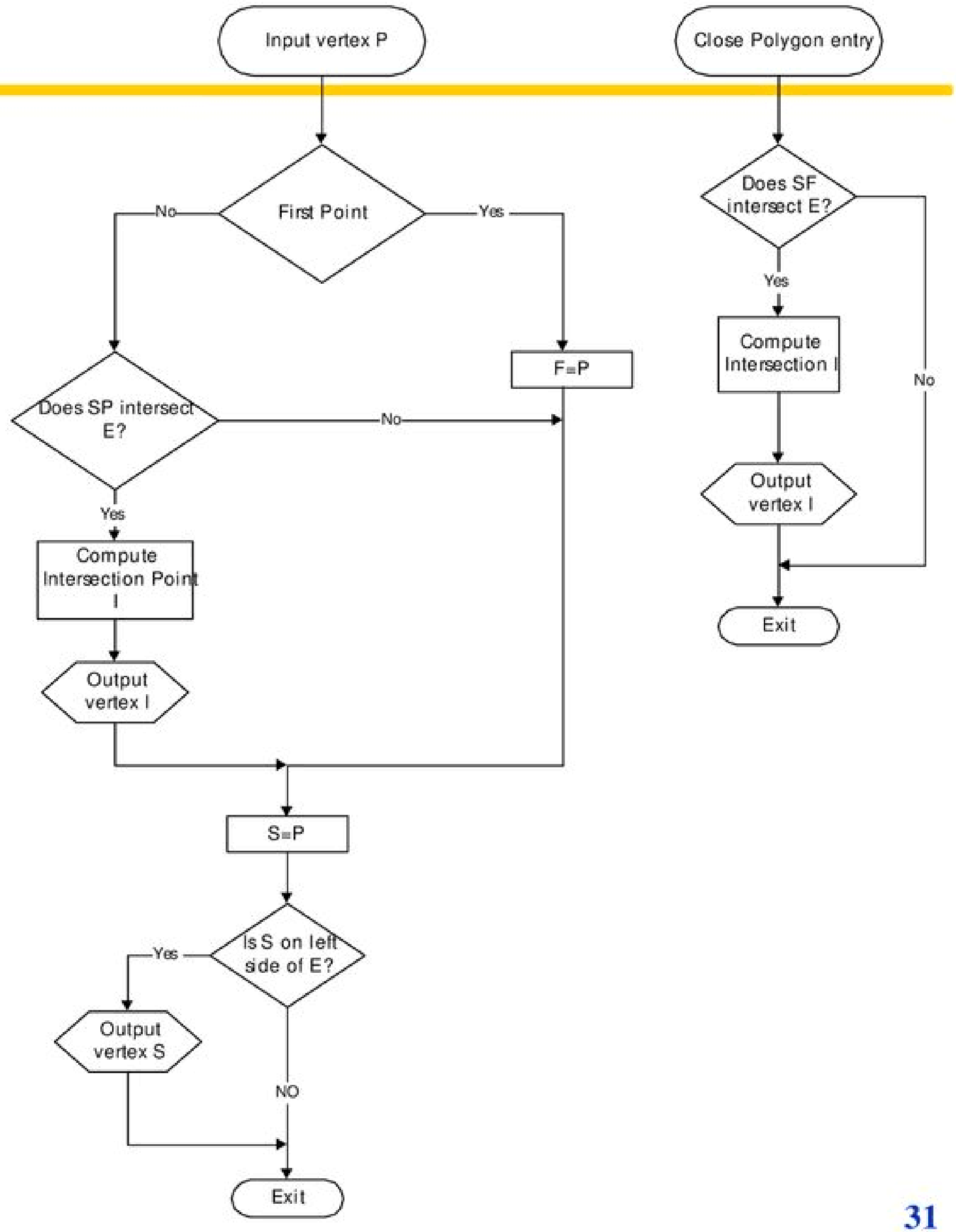


Case 2

Algorithm

1) Begin with the initial set of verteces
2) Define the category of the line as per the diagrams.
3) Clip the polygon against left rectangle boundary to produce new set of verteces.
4) Pass this new set of verteces to right, bottom, and top Boundary for clipping and generating new verteces during each clipping process.

Algorithm


Input vertex P
First Point
No
Yes
Does SP intersect E?
No
F=P
Yes
Compute Intersection Point I
Output vertex I
S=P
Is S on left side of E?
Yes
Output vertex S
NO
Exit
Close Polygon entry
Does SF intersect E?
Yes
No
Compute Intersection I
Output vertex I
Exit

LOGICAL CLASSIFICATION OF IN PUT DIVICES

WHAT IS AN INPUT DEVICE?

- An **<u>input device</u>** is any peripheral used to provide data and control signals to an information processing system.

LIST OF SOME INPUT DEVICES:

- Mouse
- Keyboard
- Trackball
- Space ball
- Joystick
- Digitizer
- Dials
- Button boxes

APPLICATION BASED INPUT DEVICES:

- Data gloves
- Touch panels
- Image scanners
- Voice systems

KEYBOARD

KEYBOARD

- IT is a device primarily used to enter TEXT STRINGS.

APLLICATIONS:

- Used to enter Text Strings
- Short cuts to many Functions

<u>In Graphics</u>:

- Used to provide screen coordinates
- Menu selection
- Gaming controls
- And FOR entering many graphics function

LATEST TYPES:

MOUSE

MOUSE

- Hand-held BOX used to position the screen cursor
- Wheels or Rollers(now-a-days Laser lights) on the bottom are used to record the position of the screen
- Generally there are two or three buttons, used for operations like recording of the cursor positions or invoking of a function
- In order to increase the number of INPUT parameters, additional devices can be included
- The Z-MOUSE is an example of this

Z-MOUSE

KEY FEATURES:

- **Has three buttons, a thumbwheel on the side, a trackball on the top and a standard mouse ball underneath**
- **This provides SIX degrees of freedom to select the positions, rotations ETC**
- **Allows 3D viewing**

Applications:

10. **Animation**
11. **Auto CAD**

And many more areas

SOME FACTS ABOUT THE MOUSE(COMPUTING)

- **Invented By** Douglas C Engelbart
  An AMERICAN inventor
- The first prototype computer mouse was made to use with a graphical user interface, in 1964
- It was patented on 17 November 1970, under the name "X-Y Po

Display System“

TRACKBALL AND SPACEBALL

TRACKBALL

- It is a 2D positioning device
- It consists a ball held by a socket containing sensors to detect the rotation of ball about TWO axis
- User rolls the ball to move the cursor
- They are often mounted on devices such as keyboards, Z-mouse ETC

SPACEBALLS

- It provides SIX degrees of freedom
- It is a fix device
- Movement detection is done using strain gauges
- Cursor can move in any direction
- It is more efficient then trackball

APPLICATIONS:

- **Used in CAD workstations**
- **In animation**
- **Sometimes on special Workstations such as the radar consoles in air-traffic control room**
- **In Gaming consoles**
- **People with a mobility impairment use trackballs as an assistive technology input**

JOYSTICK

JOYSTICK

- Consists of a stick pivoted on a base
- Used to steer the screen cursor
- It also has one or two PUSH buttons as input switches to perform certain actions
- Most joystick are 2D, but 3D do exist
- Distance moved from the CENTER position corresponds to the screen cursor movement in that direction

APPLICATIONS:

- In Gaming consoles
- In 3D animation
- Used to drive machines like cranes, mining trucks, hydraulics ETC
- Used as assistive technology pointing device such as in Electronic wheelchairs

DATA GLOVE

DATA GLOVE

- Used to grasp a "virtual" object
- Uses sensors to detect the Hand and finger motion
- Electromagnetic coupling between signals provides information about the position and orientation of the <u>HAND</u>

APPLICATIONS:

- In 3D animation movies
- Visual effects
- Gestures can be categorized into useful information, such as to recognize Sign Language or other symbolic functions
- 3D Virtual environment Games

DIGITIZERS

DIGITIZERS

- **Common device for drawing, painting, or interactively selecting coordinate positions on an object**

- **Typically, it is used to scan an Object and to input discrete coordinate positions**

ONE TYPE of Digitizer is the Graphics Tablet

GRAPHICS TABLET

- A **graphics tablet** is a computer input device that allows one to hand-draw images and graphics, similar to the way one draws images with a pencil and paper. These tablets may also be used to capture data or handwritten signatures

- The common drawing TOOLS used to draw are **HAND CURSOR** and **STYLUS**

- A **STYLUS** is a pencil-shaped pointing device

GRAPHICS TABLET

- This Digitizing system uses electromagnetic resonance to detect the 3D positions of the STYLUS
- This allows one to produce different shades of brush strokes with different pressure on its surface
- Acoustic Tablets use sound waves to detect the position of the STYLUS

GRAPHICS TABLET

- **3D Digitizers use sonic or electromagnetic transmissions to record positions**

- **It uses the same mechanism used in the Data Gloves**

APPLICATIONS:

- Used in generating Computer generated graphic images
- Used in creating characters for Animation
- In Technical drawings and CAD
- Used for Handwriting recognition

IMAGE SCANNERS

IMAGE SCANNERS

- In computing, a **scanner** is a device that optically scans images, printed text, handwriting, or an object, and converts it to a digital image
- When the scanning is performed, the gradation of gray scale or colors are recorded and stored in an array
- Once scanned, any kind of transformations can be applied to the object image

LIGHT PENS

LIGHT PENS

- A **light pen** is a computer input device in the form of a light-sensitive wand used in conjunction with a computer's CRT TV set or monitor
- Allows the users to point to displayed objects and to draw objects on screen
- The position points are highly accurate and sensitive
- It generates electrical pulse which records the position of the electron beam
- Not very much popular

Interactive picture construction techniques, Computer Graphics

Assignment Help:

Explain the interactive picture construction techniques. interactive picture- construction methods are commonly used in variety of applications, including design and painting packages. These methods provide user with the capability to position objects, to constrain fig. to predefined orientations or alignments, to sketch fig., and to drag objects around the screen. Grids, gravity fields, and rubber band methods are used to aid in positioning and other picture construction operations. The several techniques used for interactive picture construction that are incorporated into graphics packages are:

(1) Basic positioning methods:- coordinate values supplied by locator input are often used with positioning methods to specify a location for displaying an object or a character string. Coordinate positions are selected interactively with a pointing device, usually by positioning the screen cursor.

(2) constraints:-A constraint is a rule for altering input coordinates values to produce a specified orientation or alignment of the displayed coordinates. the most common constraint is a horizontal or vertical alignment of straight lines.

(3) Grids:- Another kind of constraint is a grid of rectangular lines displayed in some part of the screen area. When a grid is used, any input coordinate position is rounded to the nearest intersection of two grid lines.

(4) Gravity field:- When it is needed to connect lines at positions between endpoints, the graphics packages convert any input position near a line to a position on the line. The conversion is accomplished by creating a gravity area around the line. Any related position within the gravity field of line is moved to the nearest position on the line. It illustrated with a shaded boundary around the line.

(5) Rubber Band Methods:- Straight lines can be constructed and positioned using rubber band methods which stretch out a line from a starting position as the screen cursor.

(6) Dragging:- This methods move object into position by dragging them with the screen cursor.

(7) Painting and Drawing:- Cursor drawing options can be provided using standard curve shapes such as circular arcs and splices, or with freehand sketching procedures. Line widths, line styles and other attribute options are also commonly found in painting and drawing packages.