UNIT1

Computer Graphics Introduction

- Computer Graphics is the creation of pictures with the help of a computer.
- Computer Graphics refers to a technology that generates images on a computer screen.
- It's used in digital photography, film and television, video games, and on electronic devices and is responsible for displaying images effectively to users.
- Computer graphics can be **dynamic** (such as an animated GIF) or **static** (such as a JPEG image).
- **Ivan Sutherland** is considered by many to be the father of computer graphics. He introduced such concepts as 3-D concepts, visual simulation, computer-aided design (CAD), and virtual reality. Sutherland's work in computer graphics began with his Ph. D thesis in 1963.
- It was called **Sketchpad** and allowed a user to sketch a mechanical part on a computer screen

Pie charts: statistical data (circle)

Bar charts/ histograms: numerical data (rectangles)

Pictograms: used images to respresent data

CG divided into following classes:

Business / Presentation Graphics:pie charts,bar charts,picto graphs

Scientific Graphics: x-y plots, curve fittings, flow charts

Scaled graphics: drawing of buildings,bridges,machines

Entertainment Graphics: Cartoons,Advertisements

GUI’s: images which appears on computer screen

Types of Computers:

1. Passive computer graphics
2. Interactive computer graphics

1. Users does not have direct control over the pictures. It involves only one-way communication between the computer and the user, User can see the produced image, and he cannot make any change in the image. Ex:advertisements,titles of movie,etc
2. In interactive Computer Graphics user have some controls over the picture, i.e., the user can make any change in the produced image. Interactive Computer Graphics require two-way communication between the computer and the user. A User can see the image and make any change by sending his command with an input device.

examples of it is the ping-pong game snake ladder,toddler…

Applications:

CAD

Electronic devices

Computer arts

Entertainment and Animation

Aerospace industry

Medical technology

Cartography(making maps)

Education and Training

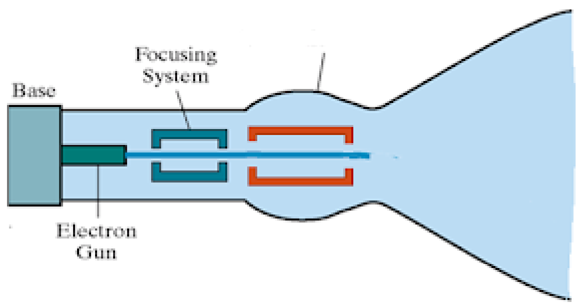
Visualization

Image processing

video-display devices

1. Cathode-Ray Tubes (CRTs)
2. Raster-Scan Displays
3. Random-Scan Displays
4. Color CRT Monitors
5. Direct View Storage Tube(DVST)

<u>Cathode-Ray Tubes (CRT)</u>

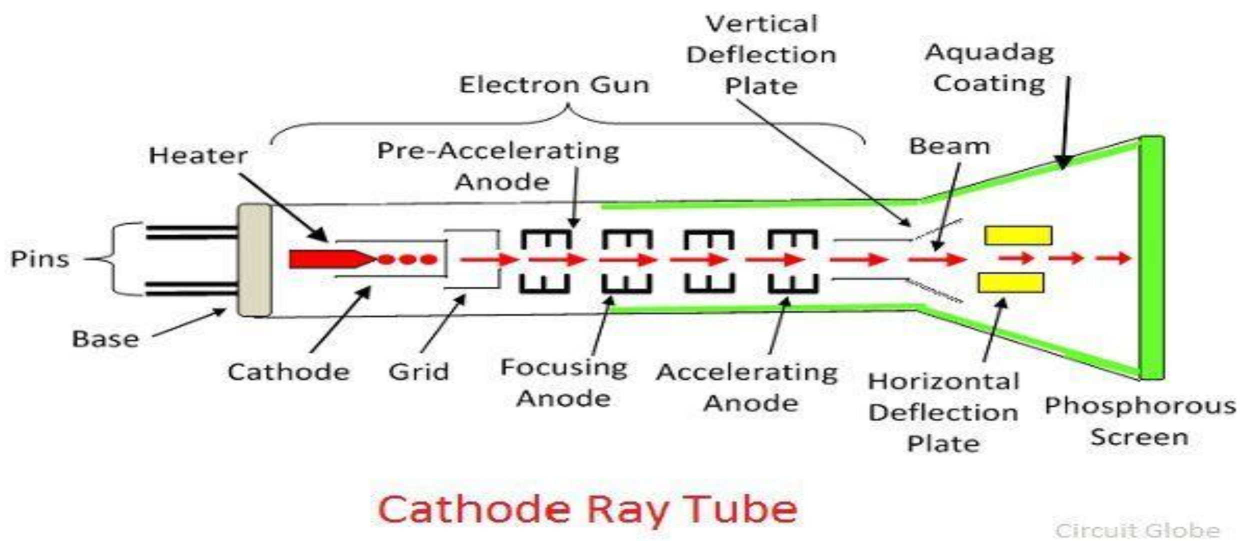


— A <u>beam of electrons</u> emitted by an <u>electron gun</u>, passes through <u>focusing and deflection systems</u> that direct the beam toward specified positions on the <u>phosphor-coated screen</u>.

— Because the light emitted by the phosphor fades very rabidly, the refresh process is needed to maintain the picture on the screen.

— Refreshing is done by redrawing the picture repeatedly by quickly directing the electron beam back over the same screen points.

— <u>Refresh rate:</u> the frequency at which a picture is redrawn on the screen.



Components of the Electron Gun :

1. The heated metal cathode
2. A control grid

— Heat is supplied to the cathode by directing a current through a coil of wire (the filament) inside the cathode.

— This causes electrons to be “boiled off” the hot cathode surface.

— Then, the free, negatively electrons are then accelerated towards the phosphor coating by a high positive voltage.

— Intensity of the electron beam is controlled by the voltage at the control grid.

— A high negative voltage applied to the control grid will shut off the beam.

— A smaller negative voltage on the control grid decreases the number of electrons passing through.

— The brightness of a display point is controlled by varying the voltage on the control grid.

— The focusing system forces the electron beam to converge to a small cross section as it strikes the phosphor.

— Deflection of the electron beam can be controlled by the deflection coils.

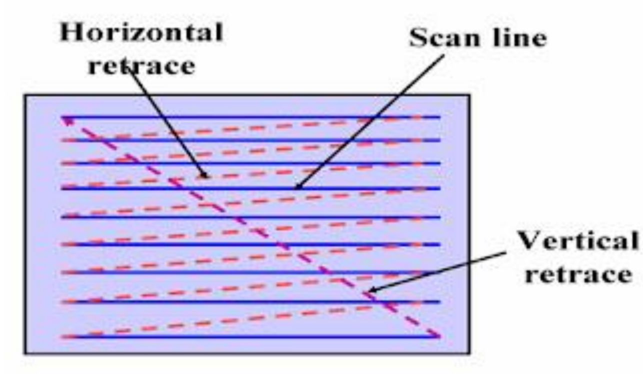
— Spots of light are produced on the screen by the transfer of the CRT beam energy to the phosphor.

— <u>Persistence:</u> how long phosphors continue to emit light after the CRT beam is removed.

— Lower-persistence phosphors require high refresh rates to maintain a picture definition on the screen without flicker and they are useful for animation.

— Higher-persistence phosphors are useful for displaying highly complex, static pictures.

<u>Raster-Scan Displays</u>



— The electron beam is swept across the screen one row at a time from top to bottom. Each row is referred to as <u>a scan line</u>.

— **Picture definition** is stored in the frame buffer. This memory area holds the set of intensity values for the screen points. These stored values are then retrieved from the refresh buffer and used to control the intensity of the electron beam as it moves from spot to spot across the screen.

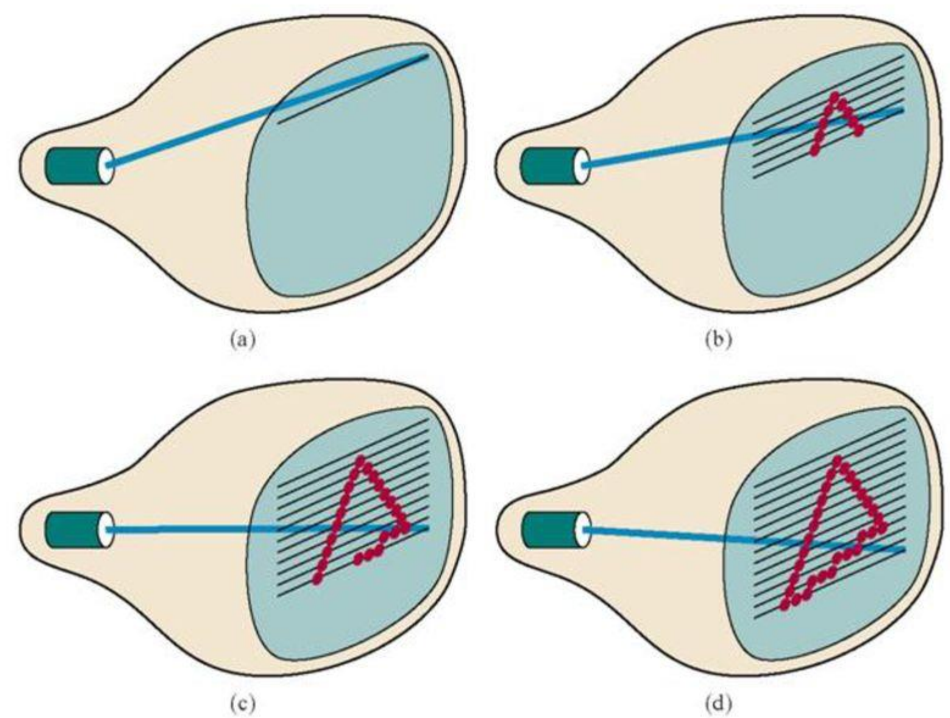


Figure 2-7
A raster-scan system displays an object as a set of discrete points across each scan line.

— Refreshing on raster-scan display is carried out at the rate of 60-80 frames per seconds, this can be done by using following retrace techniques

— **Horizontal retrace**

__ The process of moving the beam from right to left after completion of Row is known as horizontal retrace .

— **Vertical retrace**

__Electron beam returns to the top left corner of the screen to begin the next frame is called the vertical retrace.

__ The scan is synchronized with the access of the intensity values held in the frame buffer.

— The maximum resolution is determined by:

— The characteristics of the monitor

— Memory capacity available for storing the frame buffer

Random-Scan Displays

— The electron beam directed only to those parts of the screen where a picture is to be displayed.

— Sometimes called:store-writing or calligraphic displays.

— Picture definition is stored as a set of line-drawing commands.

— Draws all the component lines of a picture 30 to 60 times each second, with up to 100,000 “short” lines in the display list.

— Designed for line-drawing applications and they cannot display realistic shaded scenes.

— A pen plotter operate in a similar way.

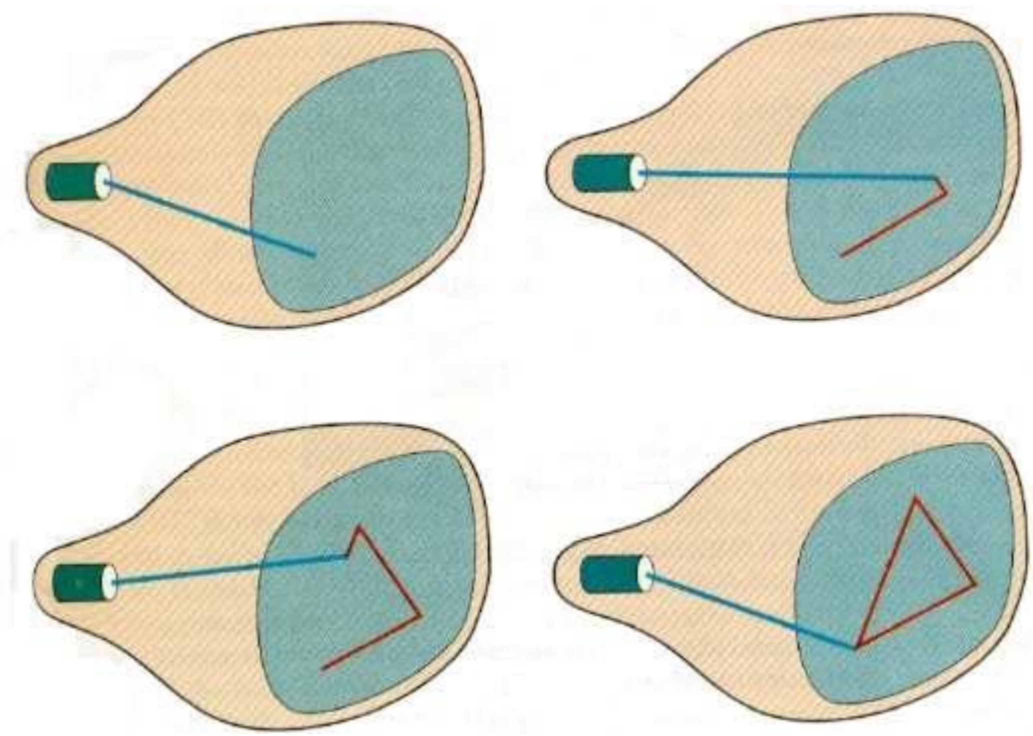
— Draws the components lines of an object in any order specified.

— Have higher resolution than raster-scan systems.

— Produce smooth line drawing.

— Refresh rate on a random scan system depends on the number of lines to be displayed.

— Picture definition is now stored as a set of line-drawing commands in an area of memory referred to as the refresh display file.



— Other names: display list, display program or refresh rate “A set of commands”.

— After all line drawing commands have been processed, the system cycles through the set of commands in the display file.

— All component lines of a picture are drawn 30 to 60 times each second

— When a small set of lines is to be displayed each refresh cycle is delayed to avoid refresh rates greater than 60 frames per second.

— **Advantages:**

— For line drawing applications

— Higher resolution than raster scan systems

— Smooth lines

— **Disadvantages:**

— Cannot display realistic shaded scenes

— Faster refreshing of the set of lines could burn out the phosphor

<u>Color CRT Monitors</u>

— Color Cathode Ray Tube(CRT) is the most common display device used to display color images.

— High resolution

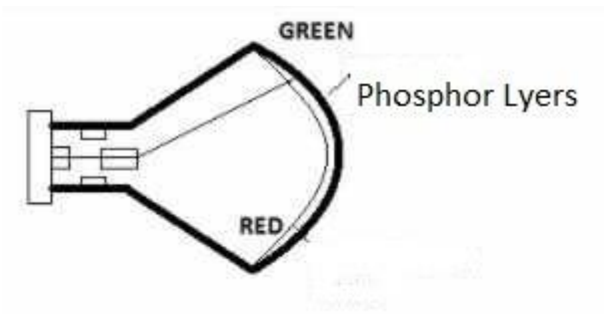
— Good color fidelity

Techniques for producing color :

— **Beam penetration method**

— **Shadow mask method**

<u>Beam Penetration Method</u>



— Random scan monitors use the beam penetration method for displaying color picture. In this, the inside of CRT screen is coated two layers of phorphor namely red and green.

— The combination of red and green light are emitted to show two addtional colors-orange and yellow.

<u>Advantages</u>

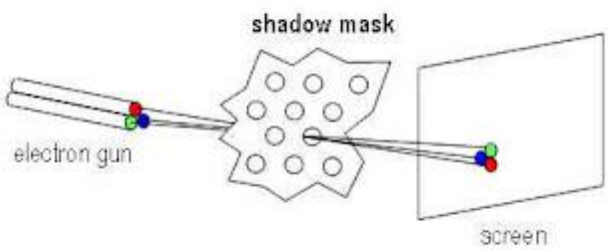
— Less expensive

<u>Disadvantages</u>

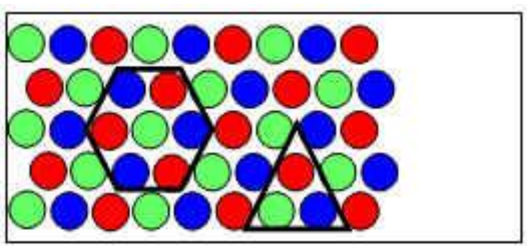
— Quality of images are not good as comparatable with other methods

— Four colors are allowed only

Shadow Mask Method



- Raster scan system are use shadow mask methods to produced a much more range of colors than beam penetration method.
- In this, CRT has three phosphor color dots. One phosphor dot emits a red light, second emits a green light and third emits a blue light.



phosphor dot pattern

- This type of CRT has three electrons guns and a shadow mask grid as shown in figure below:
- In this figure, three electrons beams are deflected and focused as a group onto the shadow mask which contains a series of holes. When three beams pass through a hole in shadow mask they activate dot triangle as shown in figure below

Advantages

- produce realistic images
- also produced different colors
- and shadows scenes.

Disadvantages

— low resolution

— expensive

— electron beam directed to whole screen

<u>Direct View Storage Tube(DVST)</u>

— A cathode-ray tube in which secondary emission of electrons from a storage grid is used to provide an intensely bright display for long and controllable periods of time. Also known as display storage tube; viewing storage tube.

- Use of no refresh buffer

— These monitors can play high resolution picture without flicker.

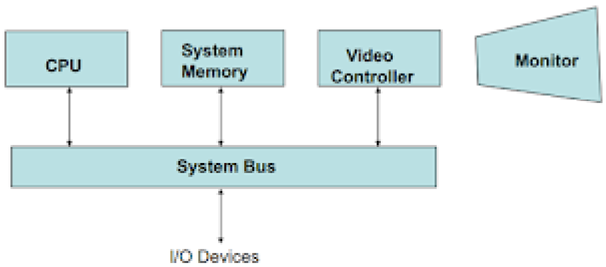
<u>Flat Panel Displays</u>

— Thin screen displays found with all portable computers and becoming the new standard with desktop computers. Instead of utilizing the cathode-ray tube technology **flat-panel displays** use Liquid-crystal display (LCD) technology or other alternative making them much lighter and thinner when compared with a traditional monitor.

Raster-scan systems

— The CPU, a special purpose processor called the video controller or display controller is used to control the operation of the display device.

— Here the frame buffer is in the system memory, the video controller access the frame buffer to refresh the screen.

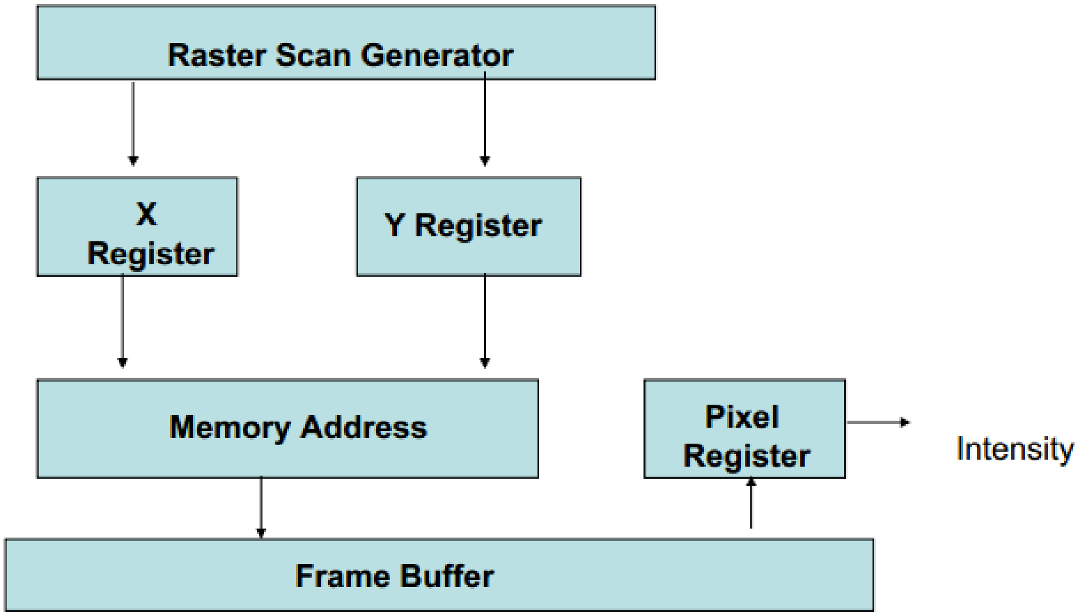


Architecture of Simple Raster graphics system

1. Video Controller

— A fixed area of the system memory is reserved for the frame buffer, and the video controller is given direct access to the frame buffer memory.

— The co-ordinates of the graphics monitor starts at the lower left screen corner. Positive x values increasing to the right and y values increasing from bottom to top.

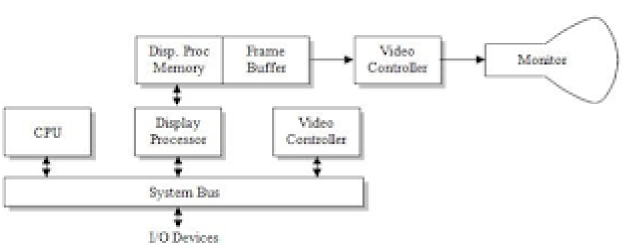


Basic Video Controller Refresh Operation

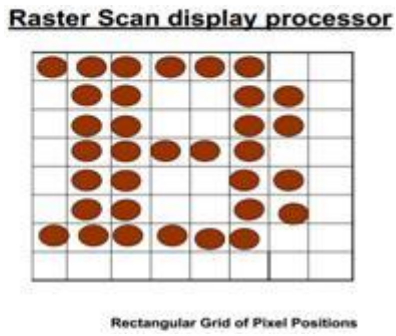
To speed up pixel processing video controllers can retrieve multiple pixel values from the refresh buffer on each pass. The multiple pixel intensities are then stored in a separate register and used to control the CRT beam intensity for a group of adjacent pixels. When this group of the pixel has been processed the next block of pixel values is retrieved from the frame buffer.

2. Display Processor

- A major task of the display processor is digitizing a picture definition given in an application program into a set of pixel-intensity values for storage in the frame buffer. This digitization process is called **scan conversion**.
- One number of each pair indicates an intensity value, and the second number specifies number of adjacent pixels the scan line that is also having same intensity. This technique is called run-length encoding.



The above diagram shows the refresh operation of video controller. Two registers are used to store the co-ordinates of the screen pixels. Initially x=0 and $y=y_{max}$

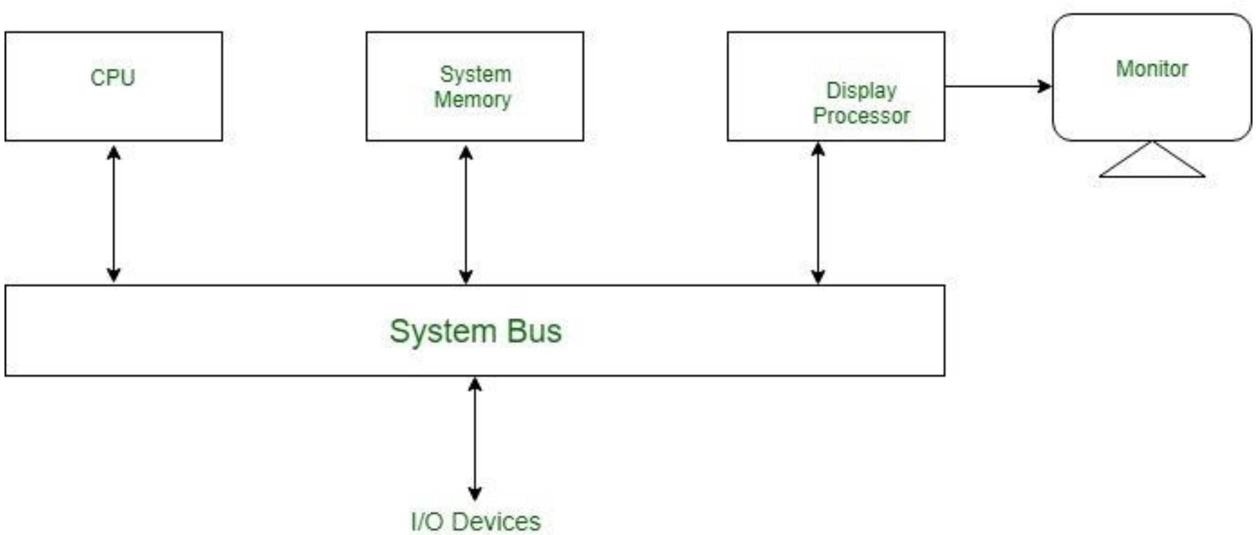


— The value stored in the frame buffer corresponding to this pixel position is retrieved.

— And the x value is incremented by 1 and the corresponding y value is retrieved, like that the pixel values are retrieved line by line.

— Once the last pixel is reached again the registers are reset to initial value to repeat the process

random scan systems



— An application program is input and stored in the system memory along with a graphics package. Graphics commands in the program are translated by the graphics package into a display file stored in the system memory.

— This display file is then accessed by the display processor to refresh the screen.

— The display processor cycles through each command in the display file program once during every refresh cycle.

— Graphic patterns are drawn on a random scan system by directing the electron beam along the component lines of the picture.

— Lines are defined by the values for their co-ordinate endpoints, and these input co-ordinate values are converted to x and y deflection voltages. A scene is then drawn one line at a time by positioning the beam to fill in the line between specified endpoints.

Computer - Input Devices

- Keyboard
- Mouse
- Joy Stick
- Light pen
- Track Ball
- Scanner
- Graphic Tablet
- Microphone
- Magnetic Ink Card Reader(MICR)
- Optical Character Reader(OCR)
- Bar Code Reader

Keyboard

Keyboard is the most common and very popular input device which helps to input data to the computer. The layout of the keyboard is like that of traditional typewriter, although there are some additional keys provided for performing additional functions.

Keyboards are of two sizes 84 keys or 101/102 keys, but now keyboards with 104 keys or 108 keys are also available for Windows and Internet.

The keys on the keyboard are as follows Typing keys which includes A-Z ,0-9, Numeric Keypad it consist 17 keys, 12 Function keys, Control keys includes Home,End,Insert,Delete,PageUp,PageDown and special purpose keys.

Mouse

Mouse is a hand held pointing device. It consist a small palm size box with a round ball at its base, which senses the movement of the mouse and sends corresponding signals to the CPU when the mouse buttons are pressed.

Generally, it has two buttons called the left and the right button and a wheel is present between the buttons. A mouse can be used to control the position of the cursor on the screen, but it cannot be used to enter text into the computer.

Joystick

Joystick is also a pointing device, which is used to move the cursor position on a monitor screen. It is a stick having a spherical ball at its both lower and upper ends. The lower spherical ball moves in a socket. The joystick can be moved in all four directions.

Light Pen

Light pen is a pointing device similar to a pen. It is used to select a displayed menu item or draw pictures on the monitor screen. It consists of a photocell and an optical system placed in a small tube.

Track Ball

Track ball is an input device that is mostly used in notebook or laptop computer, instead of a mouse. This is a ball which is half inserted and by moving fingers on the ball, the pointer can be moved.

Scanner

Scanner is an input device, which works more like a photocopy machine. It is used when some information is available on paper and it is to be transferred to the hard disk of the computer for further manipulation.

Scanner captures images from the source which are then converted into a digital form that can be stored on the disk. These images can be edited before they are printed.

Digitizer

Digitizer is an input device which converts analog information into digital form. Digitizer can convert a signal from the television or camera into a series of numbers that could be stored in a computer. They can be used by the computer to create a picture of whatever the camera had been pointed at.

Microphone

Microphone is an input device to input sound that is then stored in a digital form. The microphone is used for various applications such as adding sound to a multimedia presentation or for mixing music.

Magnetic Ink Card Reader (MICR)

MICR input device is generally used in banks as there are large number of cheques to be processed every day. The bank's code number and cheque number are printed on the cheques with a special type of ink that contains particles of magnetic material that are machine readable.

Bar Code Readers

Bar Code Reader is a device used for reading bar coded data (data in the form of light and dark lines). Bar coded data is generally used in labelling goods, numbering the books, etc. It may be a handheld scanner or may be embedded in a stationary scanner.

M<1	m>1	m=1
Case1: x – moves unit intervals $x_{k+1}=x_k+1$ Y -- ? $= \frac{y_{k+1} - y_k}{xk+1 - xk}$ $y_{k+1} = m+y_k$	Case2: y – moves unit intervals $y_{k+1}=y_k+1$ x-- ? $= \frac{y_k+1- y_k}{x_{k+1} - x_k}$ $x_{k+1} =x_k+1/m$	Case3: x & y– moves unit intervals $x_{k+1}=x_k+1$ $Y_{k+1}=y_k+1$ $= \frac{Yk+1-yk}{xk+1 - xk}$ M=1

<u>DDA – Digital Differential Analyser Algorithm</u>

Step1: Let take the starting points (x1,y1) and ending points (x2,y2)

Step2 : m = (y2-y1)/(x2-x1)

Step3 : If slope m<1 then increment x as x1+1 and calculate y1=y1+m

Step4 : If slope m>1 then increment y as y1+1 and calculate x1=x1+1/m

Step5 : If slope m=1 then increment x as x1+1 and calculate y1=y1+1

Step6 : Repeat the steps till end of line (x2,y2) is reached

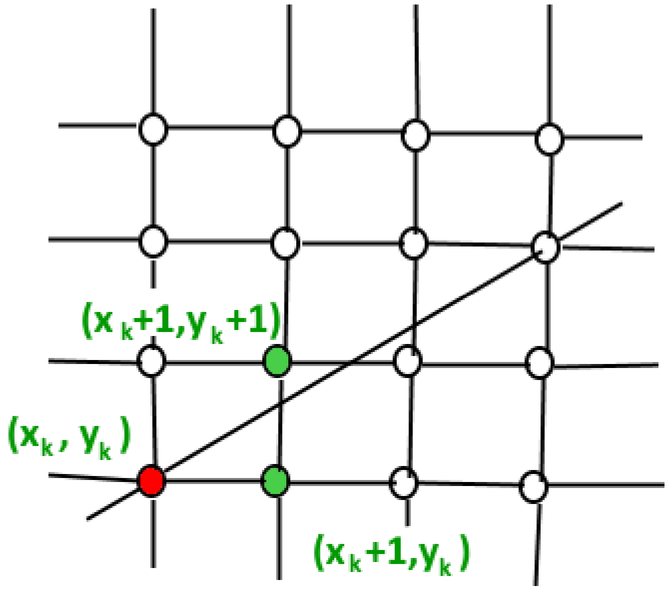
In DDA

1. It involves floating point additions rounding off is done. Rounding off operations and floating point operations consumes a lot of time. So we can move to Bresenham's Line Drawing Algorithm.

<u>Bresenham's Line Drawing Algorithm</u>

The idea of Bresenham's algorithm is to avoid floating point multiplication and addition to compute mx + c, and then compute the round value of (mx + c) in every step. In Bresenham's algorithm, we move across the x-axis in unit intervals.

We always increase x by 1, and we choose about next y, whether we need to go to y+1 or remain on y. In other words, from any position (X_k, Y_k) we need to choose between (X_k + 1, Y_k) and (X_k + 1, Y_k + 1).



We would like to pick the y value (among $Y_k + 1$ and Y_k) corresponding to a point that is closer to the original line.

We need a decision parameter to decide whether to pick $Y_k + 1$ or Y_k as the next point.

Given-

- Starting coordinates = (X_0, Y_0)
- Ending coordinates = (X_n, Y_n)

The points generation using Bresenham Line Drawing Algorithm involves the following steps-

Step-01:

Calculate ΔX and ΔY from the given input.

These parameters are calculated as-

- $\Delta X = X_n - X_0$
- $\Delta Y = Y_n - Y_0$

Step-02:

Calculate the decision parameter P_k.

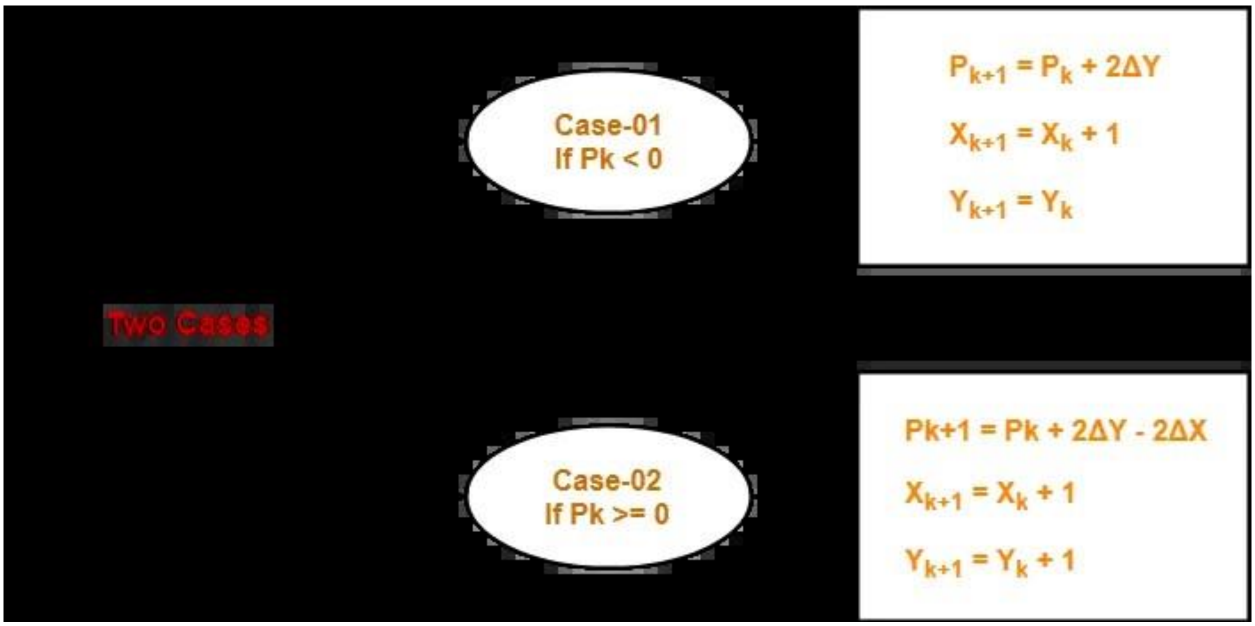
It is calculated as-

$$P_k = 2\Delta Y - \Delta X$$

Step-03:

Suppose the current point is (X_k, Y_k) and the next point is (X_{k+1}, Y_{k+1}).
Find the next point depending on the value of decision parameter P_k.

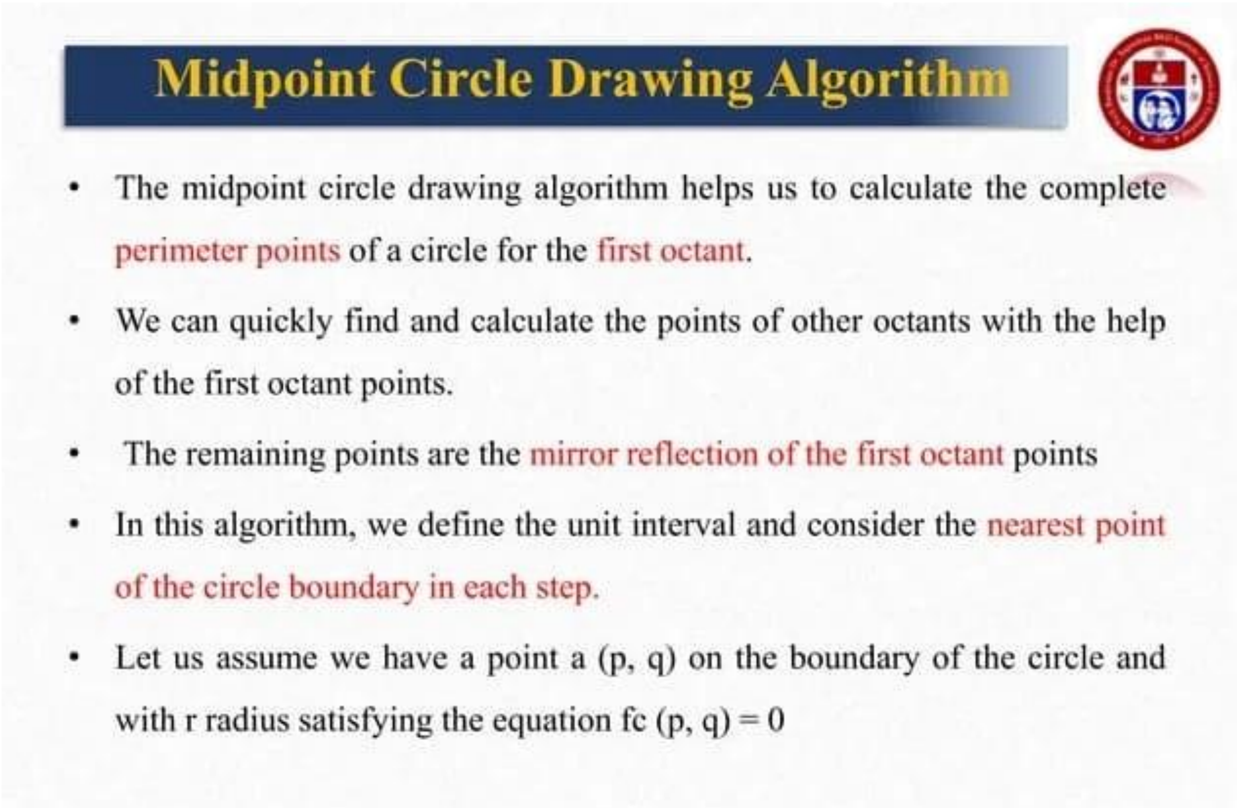
Follow the below two cases-



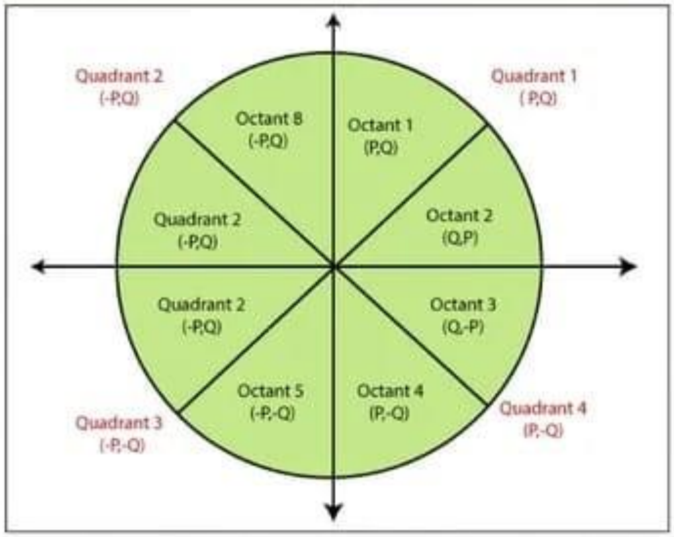
Step-04:

Keep repeating Step-03 until the end point is reached or number of iterations equals to $(\Delta X-1)$ times.

Mid Point Circle Drawing Algorithm

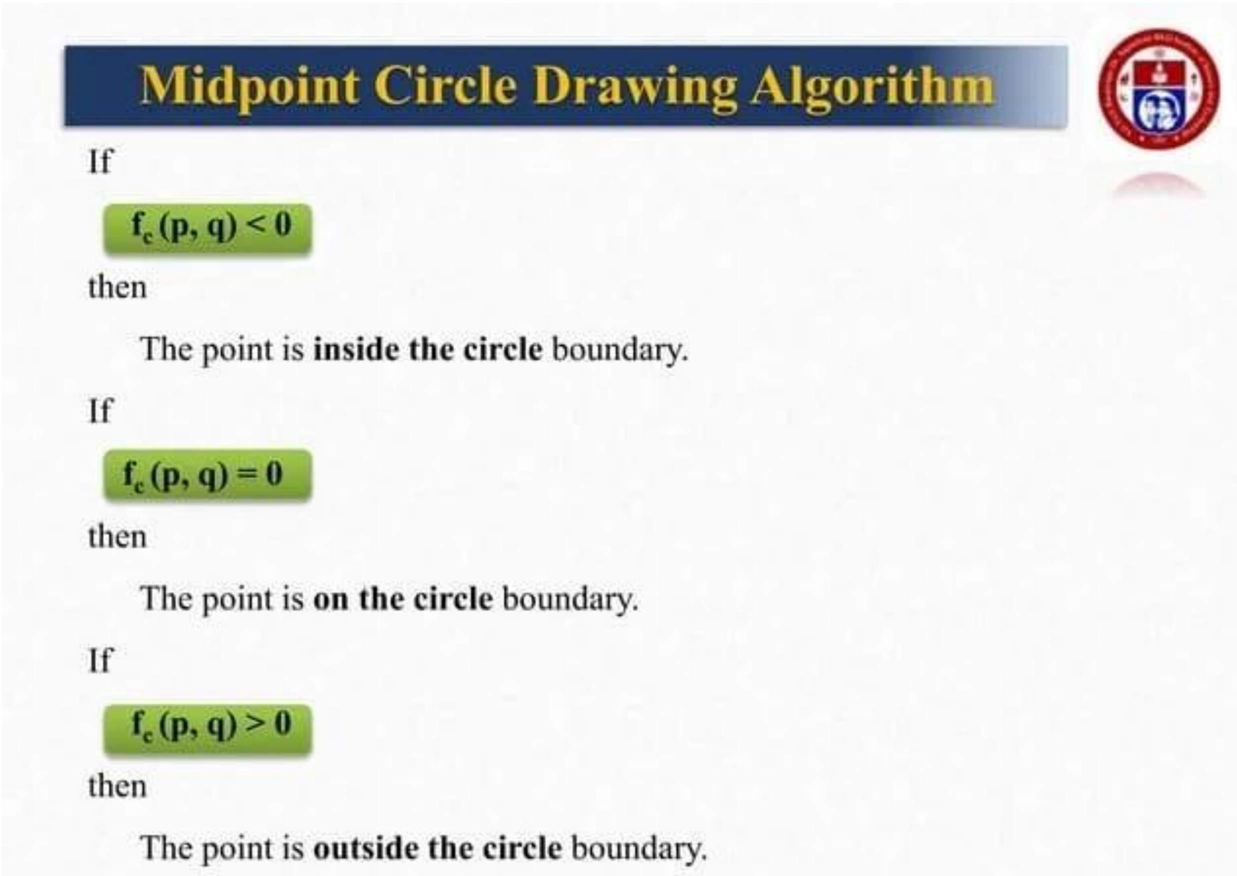


Midpoint Circle Drawing Algorithm



The equation of the circle is

$f_c(p, q) = p^2 + q^2 = r^2$ .... (1)



The points for other octacts are generated using the eight symmetry property.

Procedure-

Given-

- Centre point of Circle = (X_0, Y_0)
- Radius of Circle = R

The points generation using Mid Point Circle Drawing Algorithm involves the following steps-

Step-01:

Assign the starting point coordinates (X_0, Y_0) as-

- $X_0 = 0$
- $Y_0 = R$

Step-02:

Calculate the value of initial decision parameter P_0 as-

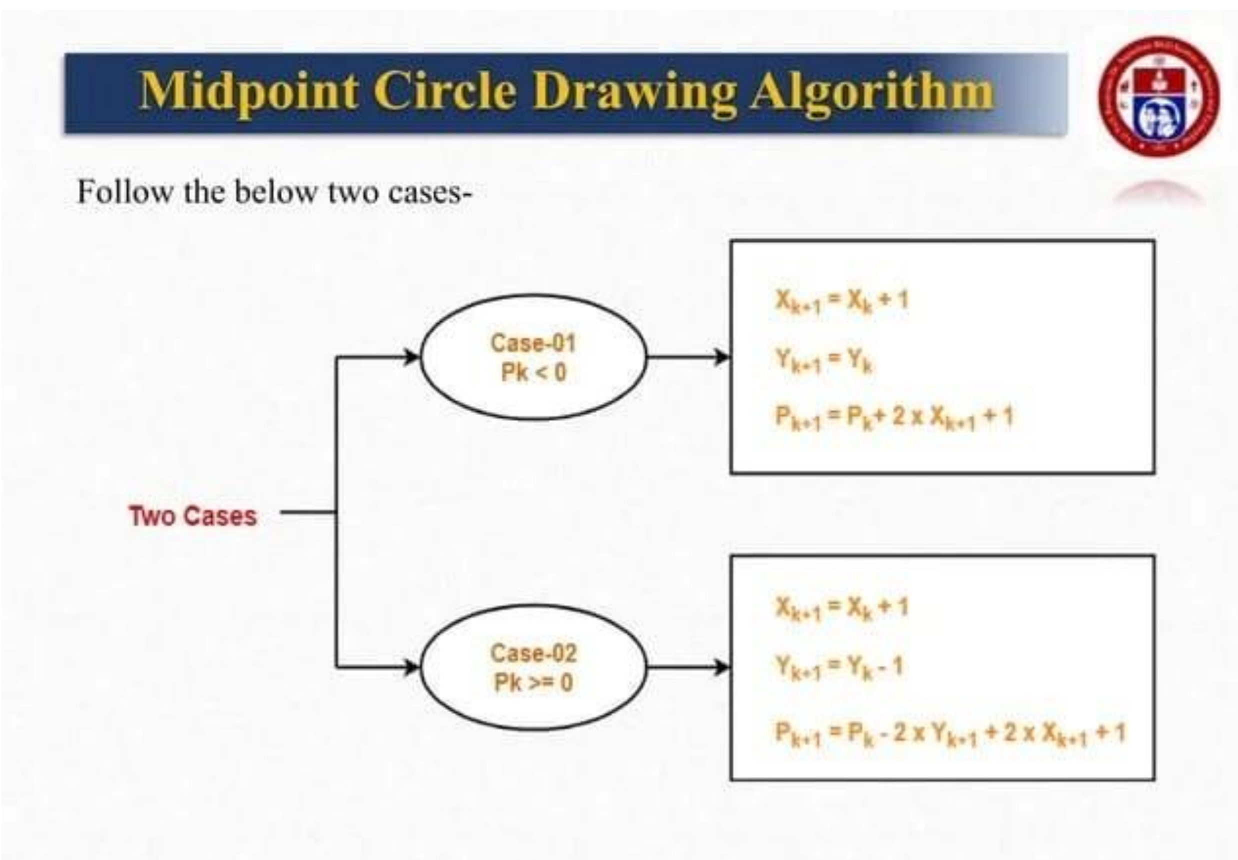
$P_0 = 1 – R$

Step-03:

Suppose the current point is (X_k, Y_k) and the next point is (X_{k+1}, Y_{k+1}).

Find the next point of the first octant depending on the value of decision parameter P_k.

Follow the below two cases-



Step-04:

If the given centre point (X0, Y0) is not (0, 0), then do the following and plot the point-

- Xplot = Xc + X0
- Yplot = Yc + Y0

Here, (Xc, Yc) denotes the current value of X and Y coordinates.

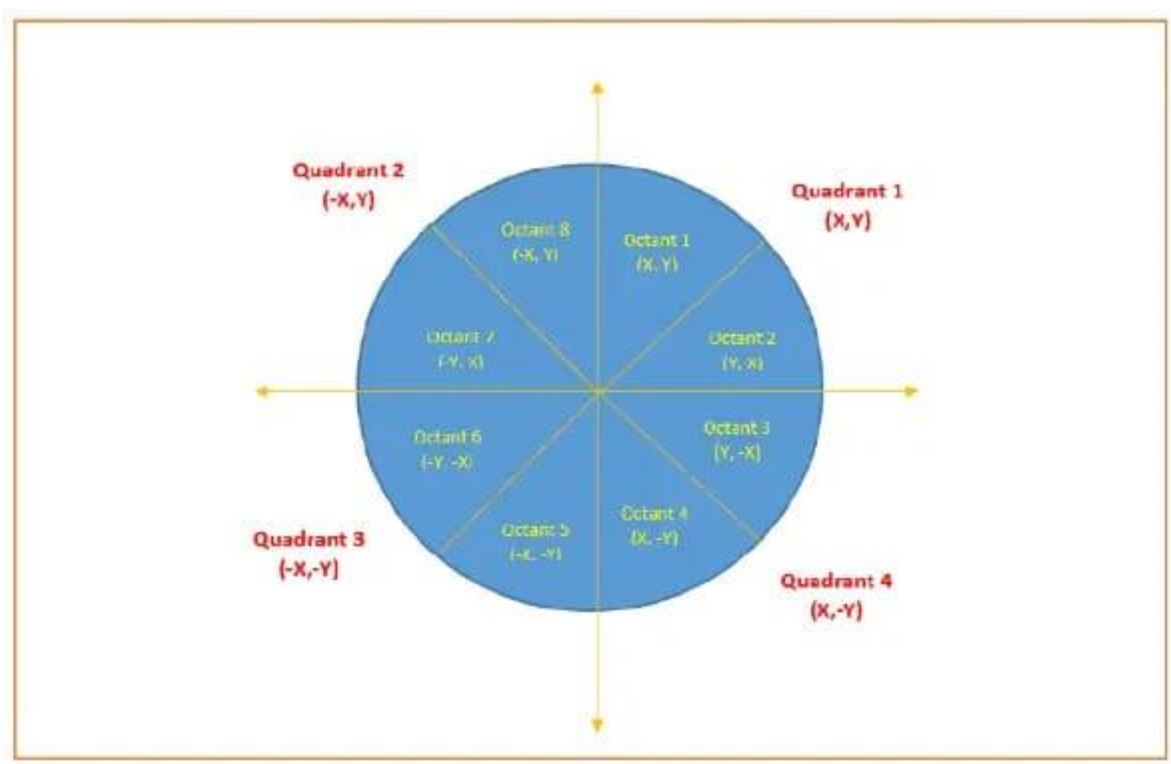
Step-05:

Keep repeating Step-03 and Step-04 until Xplot >= Yplot.

Step-06:

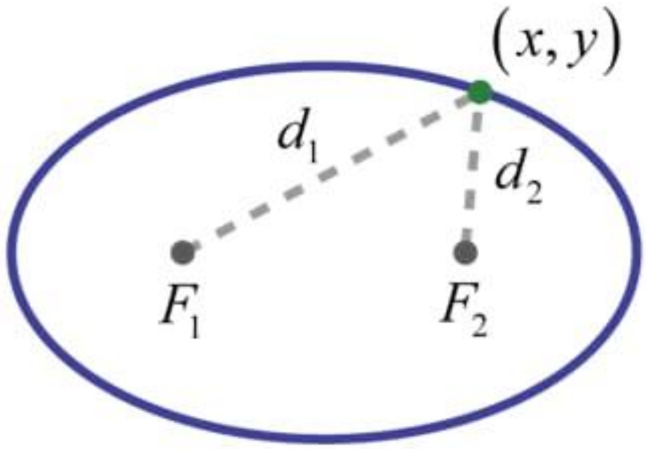
Step-05 generates all the points for one octant.

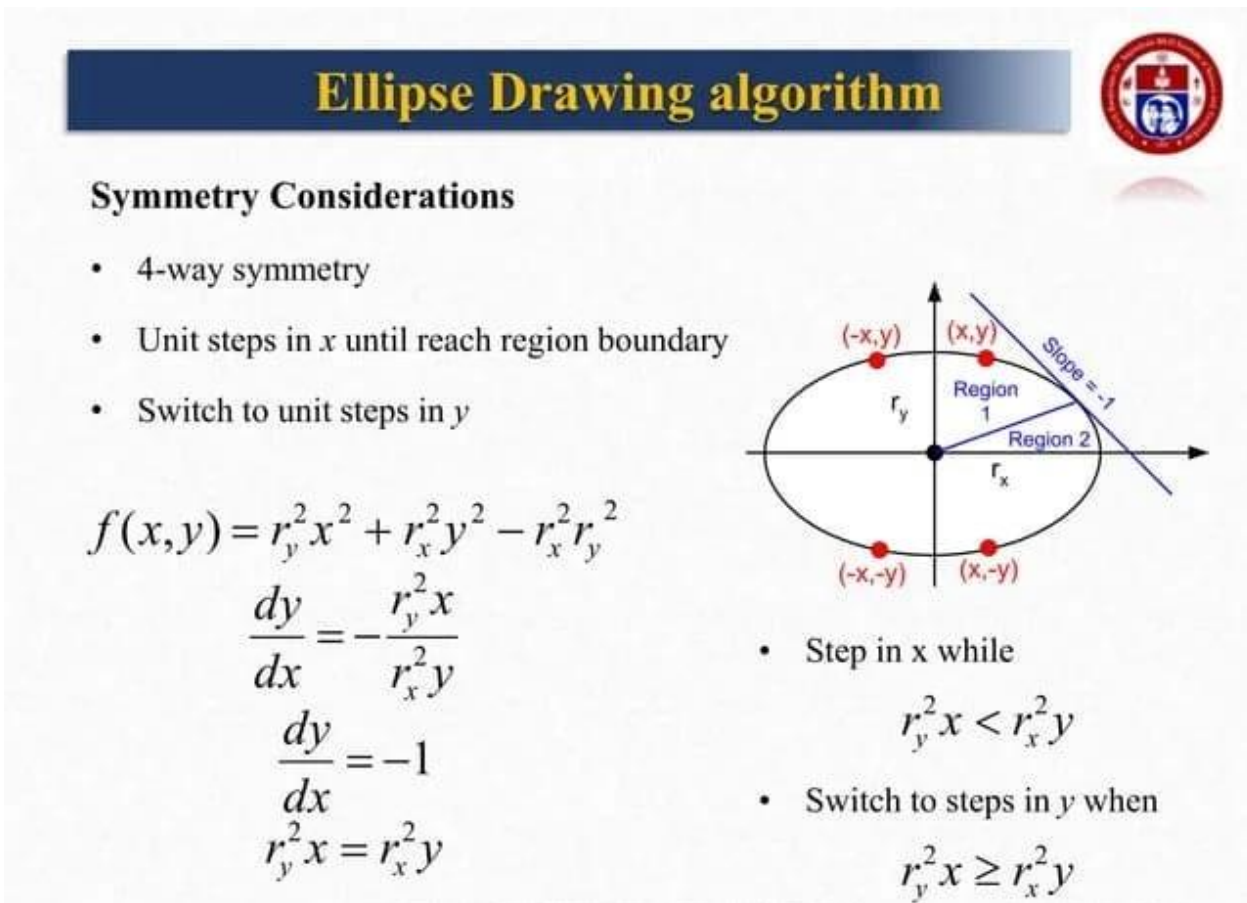
To find the points for other seven octants, follow the eight symmetry property of circle



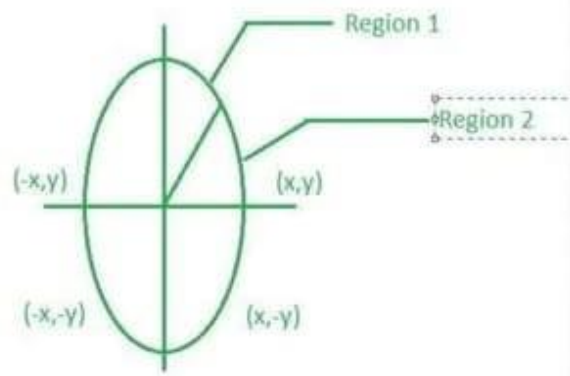
Midpoint ellipse drawing algorithm

An ellipse is defined as the set of points such that the sum of the distances from two fixed positions is the same for all points.





Mid-Point Ellipse Algorithm



•Midpoint ellipse algorithm plots(finds) points of an ellipse on the first quadrant by dividing the quadrant into two regions.

•Each point(x, y) is then projected into other three quadrants (-x, y), (x, -y), (-x, -y) i.e. it uses 4-way symmetry.

Mid-Point Ellipse Algorithm

Function of ellipse:

$f_{ellipse}(x, y) = x^2b^2 + y^2a^2 - a^2b^2$

$f_{ellipse}(x, y) < 0$ *then (x, y) is inside the ellipse.*

$f_{ellipse}(x, y) > 0$ *then (x, y) is outside the ellipse.*

$f_{ellipse}(x, y) = 0$ *then (x, y) is on the ellipse.*

Decision parameter:

Initially, we have two decision parameters $p1_k$ in region 1 and $p2_k$ in region 2.

These parameters are defined as : $p1_k$ in region 1 is given as

$$p1_k = r_y^2 + 1/4r_x^2 - r_x^2r_y$$

<u>Mid-Point Ellipse Algorithm :</u>

1. Take input radius along x axis and y axis and obtain center of ellipse.
2. Initially, we assume ellipse to be centered at origin and the first point as : $(x, y_0) = (0, r_y)$.
3. Obtain the initial decision parameter for region 1 as: $p1_0 = r_y^2 + 1/4r_x^2 - r_x{}^2r_y$
4. For every x_k position in region 1 :

   If $p1_k < 0$ then the next point along the is (x_{k+1}, y_k) and $p1_{k+1} = p1_k + 2r_y^2x_{k+1} + r_y^2$

   Else, the next point is (x_{k+1}, y_{k-1})

   And $p1_{k+1} = p1_k + 2r_y^2x_{k+1} - 2r_x^2y_{k+1} + r_y^2$
5. Obtain the initial value in region 2 using the last point (x_0, y_0) of region 1 as:

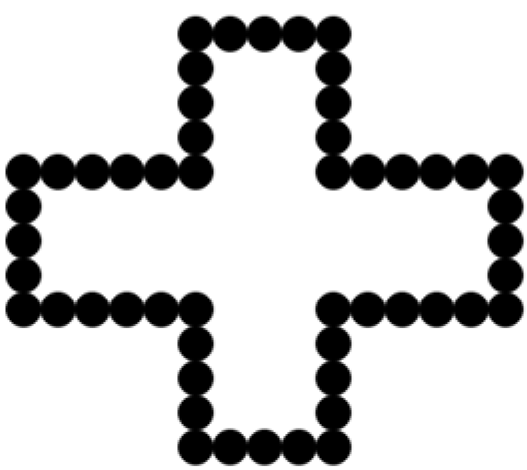
   $p2_0 = r_y^2(x_0 + 1/2)^2 + r_x^2(y_0 - 1)^2 - r_x^2r_y^2$
6. At each y_k in region 2 starting at k =0 perform the following task.

   If $p2_k > 0$ the next point is (x_k, y_{k-1}) and $p2_{k+1} = p2_k - 2r_x^2y_{k+1} + r_x^2$

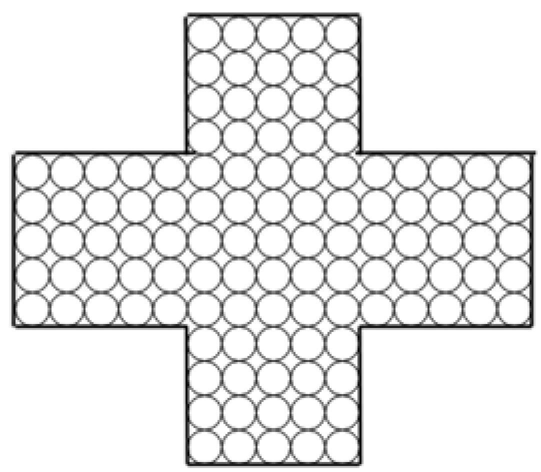
7. Else, the next point is (x_{k+1}, y_{k-1}) and $p2_{k+1}=p2_k+2r_y^2x_{k+1} -2r_x^2y_{k+1}+r_x^2$
8. Now obtain the symmetric points in the three quadrants and plot the coordinate value as: x=x+xc, y=y+yc
9. Repeat the steps for region 1 until $2r_y^2x\&gt=2r_x^2y$

Filled Area Primitives:

Region filling is the process of filling image or region. Filling can be of boundary or interior region as shown in fig. Boundary Fill algorithms are used to fill the boundary and flood-fill algorithm are used to fill the interior.



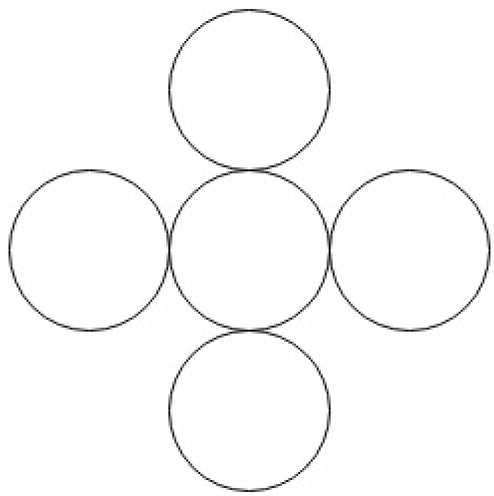
Boundary Filled Region



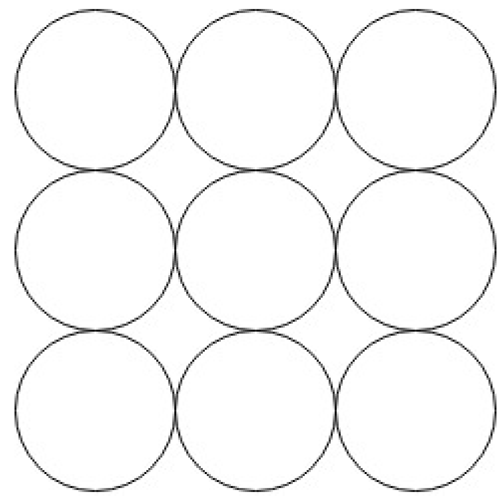
Interior or Flood Filled Region

Boundary Filled Algorithm:

This algorithm uses the recursive method. First of all, a starting pixel called as the seed is considered. The algorithm checks boundary pixel or adjacent pixels are colored or not. If the adjacent pixel is already filled or colored then leave it, otherwise fill it. The filling is done using four connected or eight connected approaches.



Four Connected



Eight Connected

Four connected approaches is more suitable than the eight connected approaches.

1. Four connected approaches: In this approach, left, right, above, below pixels are tested.

2. Eight connected approaches: In this approach, left, right, above, below and four diagonals are selected.

Boundary can be checked by seeing pixels from left and right first. Then pixels are checked by seeing pixels from top to bottom. The algorithm takes time and memory because some recursive calls are needed.

Problem with recursive boundary fill algorithm:

It may not fill regions sometimes correctly when some interior pixel is already filled with color. The algorithm will check this boundary pixel for filling and will found already filled so recursive process will terminate. This may vary because of another interior pixel unfilled.

So check all pixels color before applying the algorithm.

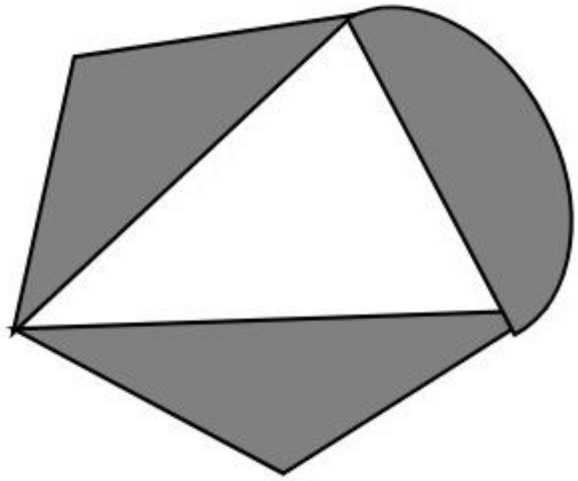
Algorithm:

1. Procedure fill (x, y, color, color1: integer)
2. **int** c;
3. c=getpixel (x, y);
4. **if** (c!=color) (c!=color1)
5. {
6. setpixel (x, y, color)
7. fill (x+1, y, color, color 1);
8. fill (x-1, y, color, color 1);
9. fill (x, y+1, color, color 1);
10. fill (x, y-1, color, color 1);
11. }

Flood Fill Algorithm:

In this method, a point or seed which is inside region is selected. This point is called a seed point. Then four connected approaches or eight connected approaches is used to fill with specified color.

The flood fill algorithm has many characters similar to boundary fill. But this method is more suitable for filling multiple colors boundary. When boundary is of many colors and interior is to be filled with one color we use this algorithm.



In fill algorithm, we start from a specified interior point (x, y) and reassign all pixel values are currently set to a given interior color with the desired color. Using either a 4-connected or 8-connected approaches, we then step through pixel positions until all interior points have been repainted.

Disadvantage:

1. Very slow algorithm
2. May be fail for large polygons
3. Initial pixel required more knowledge about surrounding pixels.

Algorithm:

```
Procedure floodfill (x, y,fill_ color, old_color: integer)
   If (getpixel (x, y)=old_color)
  {
   setpixel (x, y, fill_color);
   fill (x+1, y, fill_color, old_color);
    fill (x-1, y, fill_color, old_color);
   fill (x, y+1, fill_color, old_color);
   fill (x, y-1, fill_color, old_color);
    }
      }
```

Scan Line Polygon Fill Algorithm:

The basic scan-line algorithm is as follows:

- Find the intersections of the scan line with all edges of the polygon
- Sort the intersections by increasing x coordinate
- Fill in all pixels between pairs of intersections that lie interior to the polygon

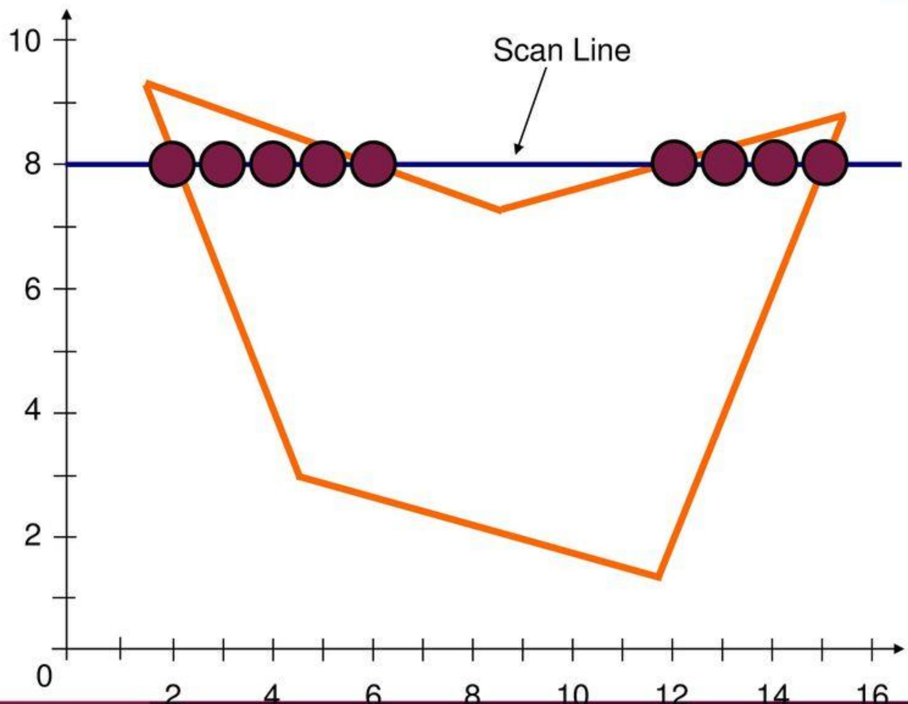
The scan-line polygon-filling algorithm involves

• the horizontal scanning of the polygon from its lowermost to its topmost vertex,

• identifying which edges intersect the scan-line,

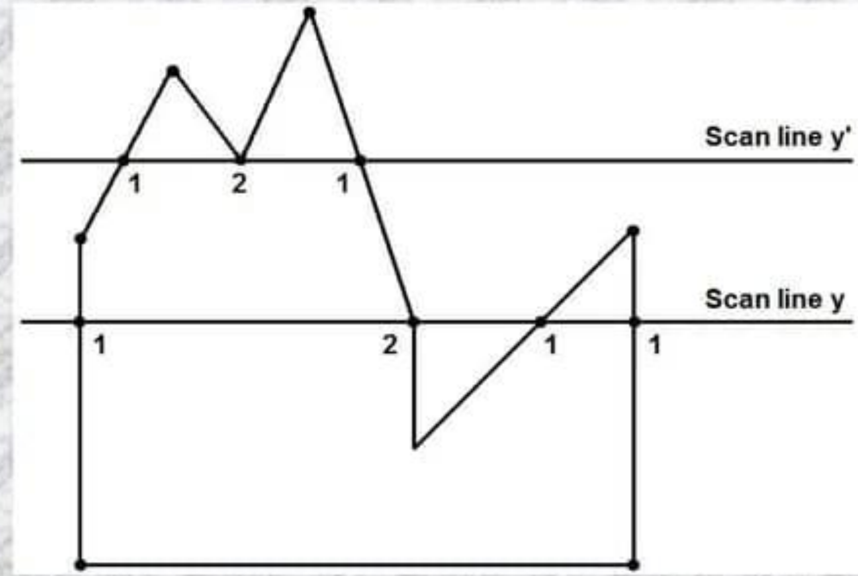
• and finally drawing the interior horizontal lines with the specified fill color. process.

Scan-Line Polygon Fill Algorithm



The Scan-Line Polygon Fill Algorithm

Dealing with vertices



The Scan-Line Polygon Fill Algorithm

Dealing with vertices

- When the endpoint **y** coordinates of the two edges are **increasing**, the **y** value of the upper endpoint for the **current edge** is decreased by one (a)
- When the endpoint **y** values are **decreasing**, the **y** value of the **next edge** is decreased by one (b)