1-5 : General Bresenham's algorithm

A full implementation of Bresenham algorithm requires modification for lying in the other octanats. These can easily be developed by considering quadrant in which the line lies and its slope. When the absolute magnitude of the slop of the line is > 1 , Y is incremented by one end Bresenham error is used to determine when to increment X.

Whether X or Y incremented by +(-) 1 depends on the quadrant.

General Bresenham's algorithm for all quadrants

```
X=X1
Y=Y1
dX=Abs (X2-X1)
dY=Abs(Y2-Y1)
S1=Sign (X2-X1)
S2=Sign (Y2-Y1)

If dY > dX Then
Begin
      T=dX : dX=dY : dY=T  : Interchange=1
End
Else
    Interchange =0
End If
E= 2 dy - dx
For I=1 to dX
 Plot (X,Y)
 While ( E ≥ 0)
 Begin
      If Interchange=1 Then X=X + S1
                       Else Y= Y + S2
       End If
      E= E - 2 dx
 End While
If Interchange=1 Then Y=Y + S2
                 Else X=X + S1
End If
E = E+ 2 dy
Next I
Finish
```

Example 4 : Draw the line from (0,0) to (-8,-4) using General Bresenham algorithm

Sol 4 : X=0 ; Y=0 ; dX=8 ; dY=4 ; S1=-1 ; S2=-1

Because dX>dY then Interchange=0 ; E=0

I	Plot	E	X	Y
		0	0	0
1	(0,0)			
		-16	0	-1
		-8	-1	-1
2	(-1, -1)			
		0	-2	-1
3	(-2, -1)			
		-16	-2	-2
		-8	-3	-2
4	(-3, -2)			
		0	-4	-2
5	(-4, -2)			
		-16	-4	-3
		-8	-5	-3
6	(-5, -3)			
		0	-6	-3
7	(-6, -3)			
		-16	-6	-4
		8	-7	-4
8	(-7, -4)			
		0	-8	-4

2- Circle Drawing algorithms

Bresenham algorithm for circle drawing

In addition to drawing a straight line we need to drae a circle , ellipse, etc.

To begin with circle drawing, note that only one octant of the circle need to be generated. The other parts can be obtained by successive reflections. To drive Bresenham circle generation algorithm , consider the first quadrant of the origin centered circle.

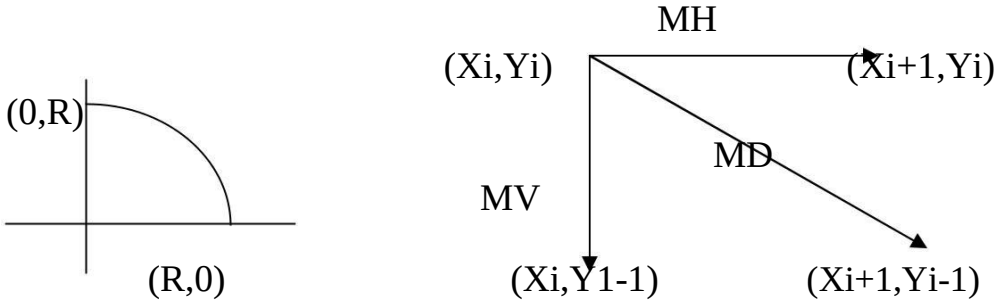
Notice that, if the algorithm begins at X=0 and Y=R (R is the radius) , then for clockwise generation of the circle Y is a monotonically decreasing function of X in the first quadrant.

Similarly if the algorithm begins at Y=0 and X=R then for counterclockwise generation of the circle, X is a decreasing function of Y.

For clockwise generation of the circle there are only three possible selections for the next pixel which best represent the circle:

1- Horizontal to the right
2- Diagonally downward to the right
3- Vertically downward

These are labeled MH , MD , MV



The algorithm chose the pixel (the movement)which minimize the square of the distance between one of these pixels and the true circle

The distance in the three cases are measured by :

Case 1 : MH = | (Xi+1)^2 + (Yi)^2 - R^2 |
Case 2 : MD = | (Xi+1)^2 + (Yi-1)^2 - R^2 |
Case 3 : MV = | (Xi)^2 + (Yi-1)^2 - R^2 |

The difference between the square of the distance from the center of the circle to the diagonal pixel at (Xi+1 , Yi-1) and the distance to a point on the circle R^2 is

$$Di = (Xi+1)^2 + (Yi-1)^2 - R^2$$

1- If Di<0 then the diagonal point (Xi+1,Yi-1) is *inside* the actual circle i.e. we use case 1 or case 2 .

It is clear that either the pixel at (Xi+1,Yi) == MH or that the pixel at (Xi+1,Yi-1) == MD must be chosen.

$$ə = | (Xi+1)^2 + (Yi)^2 - R^2| - | (Xi+1)^2 + (Yi-1)^2 - R^2|$$

if ə <0 then the difference from the actual circle to the diagonal pixel (MD) is greater than that to the horizontal pixel (MH).

If ə <0 choose MH (Xi+1,Yi)
If ə >0 choose MD (Xi+1, Yi-1)
The horizontal move has been selected when ə=0 ; i.e. when the distance are equal.

2- if Di>0 then the diagonal point (Xi+1,Yi-1) is *outside* the actual circle i.e. we use case 2 or case 3 .

$$β = | (Xi+1)^2 + (Yi-1)^2 - R^2 | - | (Xi)^2 + (Yi-1)^2 + R^2 |$$

if β <=0 choose MD (Xi+1, Yi-1)
if β >0 choose MV (Xi , Yi-1)

3- if Di=0 then we choose the pixel at (Xi+1 , Yi-1) i.e. MD

Summery

1- Di < 0

ə ≤ 0 , then choose pixel at (Xi+1 , Yi) i.e. MH

ə > 0 , then choose pixel at (Xi+1 , Yi-1) i.e. MD

2- Di > 0

β ≤ 0 , then choose pixel at (Xi+1 , Yi-1) i.e. MD

β > 0 , then choose pixel at (Xi , Yi-1) i.e. MV

3- Di = 0

choose pixel at (Xi+1 , Yi-1) i.e. MD

Bresenham circle algorithm (for the first quadrant)

Xi=0
Yi=R
Di= 2 (1-R)
Limit = 0

1: Plot (Xi , Yi)
If Yi ≤ Limit then goto 4
If Di < 0 then goto 2
If Di > 0 then goto 3
If Di = 0 then goto 20

2: ə = 2Di + 2Yi -1
If ə ≤ 0 then goto 10
If ə > 0 then goto 20

3: β= 2Di – 2X1 -1
If β ≤ 0 then goto 20
If β > 0 then goto 30

10: Xi = Xi + 1 { MH }
Di = Di + 2Xi + 1
Goto 1

20: Xi = Xi +1 { MD }
Yi = Yi - 1
Di = Di + 2Xi – 2Yi + 2
Goto 1

30: Yi = Yi -1 { MV }
Di = Di – 2Yi + 1
Goto 1

4: Finish

Example 5 :

Draw a circle with R=4

Plot (X,Y)	Di	ə	β	Xi	Yi
	-6	-	-	0	4
(0,4)	-3	-5	-	1	4
(1,4)	-3	1	-	2	3
(2,3)	4	-1	-	3	3
(3,3)	1	-	1	3	2
(3,2)	9	-	-5	4	1
(4,1)	10	-	9	4	0
(4,0)	-3				