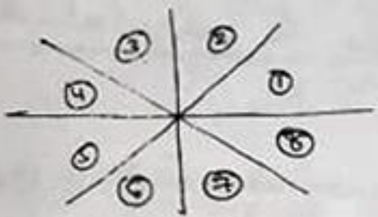
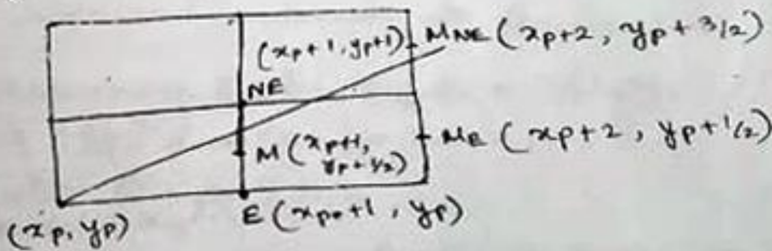
Bresenham's mid-point line drawing algorithm:

The serious drawback of DDA algorithm is that it is very time consuming as it deals with rounding off operation and floating point arithmetic. Successive addition of floating point increment causes accumulation of round off error and eventually drift away of the plotted pixels from the true line path in case of long line segments.

Bresenham's Mid-point line algo is more efficient and accurate as it avoids the round function and scan converts lines using only incremental integer calculation



Considering the slope and endpoints, a straight line can belong to one of the eight octants. We are developing the algorithm for the 1st octant



Let us consider two forms of straight line equation

$$y = mx + B$$
$$= (dy/dx) \cdot x + B \quad \text{———— ①}$$

$$f(x,y) = ax + by + c = 0 \quad \text{———— ②}$$

Comparing ① and ②:

$a = dy$

$b = -dx$

$c = B.dx.$

Abhishek Dey
Assistant Professor
Department of Computer Science
Bethune College, Kolkata

If $F(M) > 0$, then the straight line is above mid-point M else the straight line is below M.

The value of $F(M)$ can be calculated in an incremental way

Considering a decision variable/parameter

$$d = F(M) = F(x_p+1, y_p+1/2)$$
$$= a(x_p+1) + b(y_p+1/2) + c.$$

If $d \le 0$, choose E as next point

else if $d > 0$ choose NE as next point.

The next decision variable is calculated as follows:

Set $d_{old} = d$

i) Case E is chosen (current point is (x_p+1, y_p))

$$d_{new} = F(M_E)$$
$$= F(x_p+2, y_p+1/2)$$
$$= a(x_p+2) + b(y_p+1/2) + c.$$

$\therefore$ Increment of decision variable,

$$(\Delta d)_E = d_{new} - d_{old}$$
$$= a(\cancel{x_p}+2) + b(\cancel{y_p}+1/2) + \cancel{c}$$
$$- a(\cancel{x_p}+1) - b(\cancel{y_p}+1/2) - \cancel{c}$$
$$= a$$
$$= dy.$$

ii) Case NE is chosen (current point is (x_p+1, y_p+1))

$$d_{new} = F(M_{NE})$$
$$= F(x_p+2, y_p+3/2)$$
$$= a(x_p+2) + b(y_p+3/2) + c.$$

Abhishek Dey
Assistant Professor
Department of Computer Science
Bethune College, Kolkata

∴ Increment of decision variable,

$$(\Delta d)_E = d_{new} - d_{old}$$

$$= a(x_p+2) + b(y_p+3/2) + c - a(x_p+1) - b(y_p+1/2) - c$$

$$= a + b = (dy - dx)$$

The initial decision variable is.

$$d_{start} = f(x_0+1, y_0+1/2)$$

$$= a(x_0+1) + b(y_0+1/2) + c.$$

$$= ax_0 + a + by_0 + b/2 + c.$$

$$= dy - \tfrac{1}{2}dx$$

$$= \tfrac{1}{2}(2dy - dx)$$

To avoid fractional computation we can take

$$d_{start} = 2dy - dx.$$

$$(\Delta d)_E = 2dy.$$

$$(\Delta d)_{NE} = 2(dy - dx)$$

Abhishek Dey
Assistant Professor
Department of Computer Science
Bethune College, Kolkata

Pseudocode for only Octant 1:

```
Procedure Bresenham_line (xS, yS, xE, yE) //Starting and ending points of straight line are
                                            Passed as parameters
Begin
        dx = (xE – xS)
        dy = (yE – yS)
        d = 2dy – dx // Initialization of decision variable d
        (Δd)E = 2dy // Increment of d due to selection of E
        (Δd)NE = 2(dy-dx) // Increment of d due to selection of NE
        x = xS, y = yS // Initialization of starting point
        SetPixel(x ,y)
        While (x < xE)
        Begin
                If (d ≤ 0) // E is chosen
                Then
                        d = d + (Δd)E
                Else // NE is chosen
                        d = d + (Δd)NE
                        y = y + 1
                EndIF
                x = x +1
                SetPixel(x ,y)
        EndWhile
End
```

Functions used:

SetPixel(x, y) is used to plot the corresponding pixel defined by co-ordinate (x, y) on the screen.

Abhishek Dey
Assistant Professor
Department of Computer Science
Bethune College, Kolkata

Q. Find plotted pixels or plotted points of straight line A (3, 2), B (11, 4) using Bresenham's mid-point line drawing algorithm.

Answer:

dx = 8, dy = 2

d = -4

$(\Delta d)_E = 4$, $(\Delta d)_{NE} = -12$

Initialize, x =3, y = 2

Now 1st point (3, 2) is plotted

Next points will be:

d	x	y	Plotted point
0	4	2	(4,2)
4	5	2	(5, 2)
-8	6	3	(6, 3)
-4	7	3	(7,3)
0	8	3	(8, 3)
4	9	3	(9, 3)
-8	10	3	(10, 3)
-4	11	4	(11, 4)

Hence, the plotted points / pixels are: **(3, 2), (4, 2), (5, 2), (6, 3), (7, 3), (8, 3), (9, 3), (10, 3), (11, 4)**

Abhishek Dey
Assistant Professor
Department of Computer Science
Bethune College, Kolkata