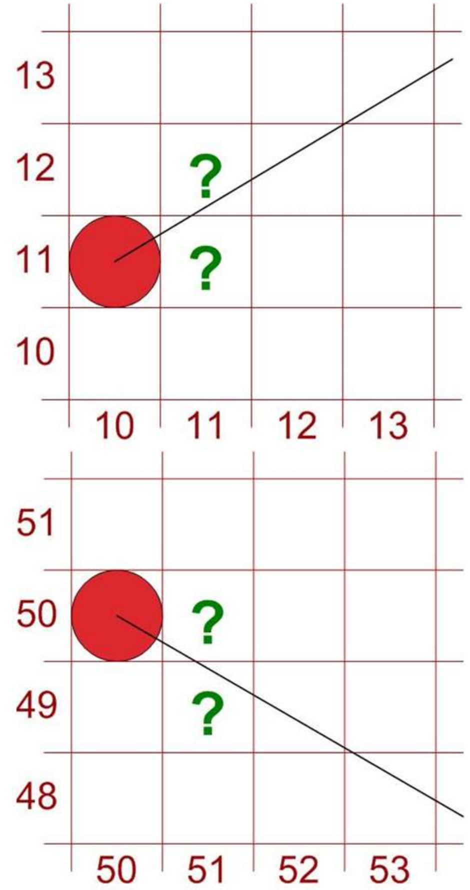
Bresenham’s Line Algorithm

Bresenham's Line Algorithm

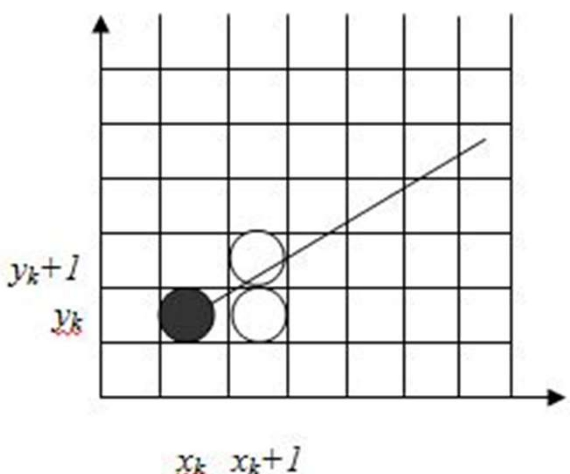
- An accurate and efficient line generating algorithm, developed by Bresenham that scan converts lines only using integer calculation to find the next *(x , y)* position to plot.
- Basic Principle
  - Increment in x or y by one unit depending upon slope of line.
  - Then, increment in other line is found on the basis of *decision variable* or *error term* i.e. the distance between the actual line location and the nearest pixel

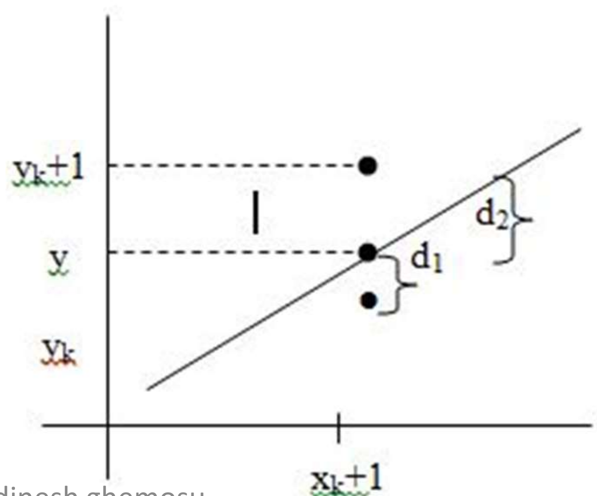
- Uses only incremental integer calculations
- Which pixel to draw ?
  - (11,11) or (11,12) ?
  - (51,50) or (51,49) ?
  - Answered by Bresenham



Line with positive slope less than 1 (0<m<1)

- Pixel position along the line path are determined by sampling at unit x intervals.
- Starting from left end point (x_0, y_0) , we step to each successive column(x) and plot the pixel closest to line path.
- Assume that (x_k, y_k) is pixel at k^{th} step then next point to plot may be either (x_k+1, y_k) or (x_k+1, y_k+1)
- At sampling position x_k+1, we label vertical pixel separation from line path as d_1& d_2 as in figure .



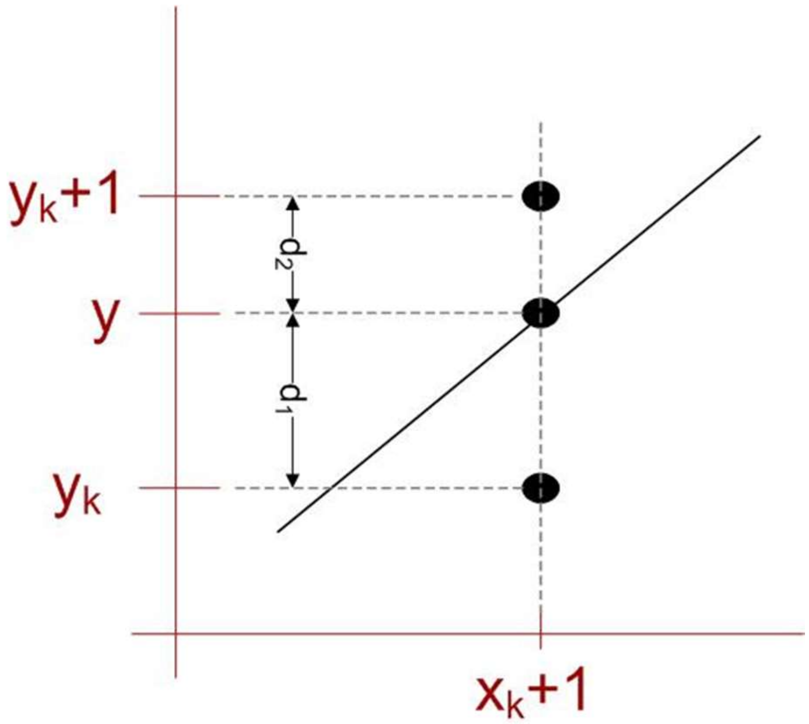


- The y-coordinate on the mathematical line path at pixel column x_k+1 is:

$$y = m(x_k + 1) + b$$

Then $$d_1 = y - y_k$$

And
$$\begin{aligned} d_2 &= (y_k + 1) - y \\ &= y_k + 1 - m(x_k + 1) - b \end{aligned}$$



Difference between separations

$$d_1 - d_2 = 2m(x_k + 1) - 2y_k + 2b - 1$$

Constant=2Δy + Δx(2b-1) Which is independent of pixel position

Defining decision parameter

$$p_k = \Delta x(d_1 - d_2) \quad [1]$$
$$= 2\Delta y.x_k - 2\Delta x.y_k + c$$

Sign of p_k is same as that of d_1-d_2 for Δx>0 (left to right sampling)

$$p_{k+1} = 2\Delta y.x_{k+1} - 2\Delta x.y_{k+1} + c$$

c eliminated here

$$p_{k+1} - p_k = 2\Delta y(x_{k+1} - x_k) - 2\Delta x(y_{k+1} - y_k)$$

because $x_{k+1} = x_k + 1$

$$p_{k+1} = p_k + 2\Delta y - 2\Delta x(y_{k+1} - y_k)$$

y_{k+1}-y_k = 0 if $p_k < 0$
y_{k+1}-y_k = 1 if $p_k \geq 0$

For Recursive calculation, initially

$$p_0 = 2\Delta y - \Delta x$$

Substitute $b = y_0 - m.x_0$ and $m = \Delta y/\Delta x$ *in [1]*

Bresenham's Line Drawing Algorithm for $|m| < 1$

1. Input the two line endpoints and store the left endpoint in (x_0,y_0)
2. Plot first point (x_0,y_0)
3. Calculate constants Δx, Δy, $2\Delta y$ and $2\Delta y - 2\Delta x$, and obtain $p_0 = 2\Delta y - \Delta x$
4. At each x_k along the line, starting at k=0, perform the following test:

   If $p_k<0$, the next point plot is (x_k+1,y_k) and

   $$P_{k+1} = p_k + 2\Delta y$$

   Otherwise, the next point to plot is $(x_k + 1, y_k+1)$ and

   $$P_{k+1} = p_k + 2\Delta y - 2\Delta x$$
5. Repeat step 4 Δx times.

Example

Endpoints (20,10) and (30,18)

Slope m = 0.8

$\Delta x = 10$, $\Delta y = 8$

$P_0 = 2\Delta y - \Delta x = 6$

$2\Delta y = 16$, $2\Delta y - 2\Delta x = -4$

Plot $(x_0, y_0) = (20,10)$

k	p_k	(x_{k+1}, y_{k+1})	k	p_k	(x_{k+1}, y_{k+1})
0	6	(21,11)	5	6	(26,15)
1	2	(22,12)	6	2	(27,16)
2	-2	(23,12)	7	-2	(28,16)
3	14	(24,13)	8	14	(29,17)
4	10	(25,14)	9	10	(30,18)

Advantages

- Involves simple integer calculations, so does not need to perform round off operation.
- It can be used to generate circles and other curves.
- Additional parameter i.e., decision parameter must be calculated at each step.

For |m| >1 case, we can set following changes:

- Calculate m, if |m| >1.
  - This case occurs when $\Delta x < \Delta y$, so $m = (\Delta y / \Delta x) > 1$.
    - Simply we can interchange the role of x and y, that is we can step along the y-direction in unit steps and calculate successive x-values.
      - Calculate initial decision parameter $p_0 = 2\Delta x - \Delta y$.
      - If $p_k < 0$, select point (x_k, y_k+1), and set $p_{k+1} = p_k + 2\Delta x$.
      - Otherwise if $p_k > 0$, select point $(x_k + 1, y_k+1)$, and set $p_{k+1} = p_k + 2\Delta x - 2\Delta y$.

Questions

- ***Digitize the line with endpoints (1, -6) and (4, 4) using BSA algorithm.***
- ***Digitize the line with endpoints (1, 6), (6, 10) using BSA algorithm.***
- ***Trace BSA for line with endpoints (15, 15), (10, 11).***
- ***Trace BSA for line with endpoints (20, 10), (30, 18).***
- ***Trace BSA for endpoints (5, 10), (10, 7).***