

Bresenham's Line

This Algorithm is used for scan converting a line. It was developed by Bresenham. It is an efficient method because it involves only integer addition subtraction, and multiplication operations. These operations can be performed very rapidly so lines can be generated quickly.

In this method, next pixel selected is that one ~~to~~ who has the latest distance from line line

steps:

- 1) Input: start and endpoints
 (x_1, y_1) and (x_2, y_2)
- 2) calculate difference:
 $dx = x_2 - x_1$
 $dy = y_2 - y_1$
- 3) Determine slope check if the line slope is gentle or steep
 $|slope| \leq 1$ or $|slope| \geq 1$

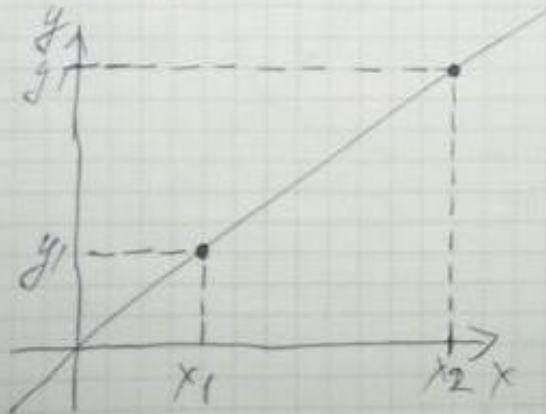
4) Based on the slope:
if gentle:

- Increment x by 1 for each.
- ~~Calculate~~ Calculate the decision parameter and update it to determine the next pixel.
- Adjust y as needed based on the decision parameter.

if steep:

- ~~if~~ Swap x and y coordinates.
- increment y by 1 for each step.
- Calculate the decision parameter and update it to determine the next pixel.
- Adjust x as needed.

5) Plot the pixels along the line's path until reaching the endpoint.



Bresenham's Circle

Algorithm.

It is not easy to display a continuous smooth arc on the computer screen as our computer screen is made of pixels organized in matrix form. So, to draw a circle on a computer screen we should always choose the nearest pixels as they could form an arc.

