Scan conversion Algorithm

Bresenham algorithm

Circle Symmetry

Mid point Circle

The Bresenham Line Algorithm

(for $|m| < 1.0$)

1. Input the two line end-points, storing the left end-point in (x_0, y_0)
2. Plot the point (x_0, y_0)
3. Calculate the constants Δx, Δy, $2\Delta y$, and $(2\Delta y - 2\Delta x)$ and get the first value for the decision parameter as:

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

4. At each x_k along the line, starting at $k = 0$, perform the following test. If $p_k < 0$, the next point to 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 $(\Delta x - 1)$ times

Circle Scan Conversion

- The equation for a circle is: Pythagoras

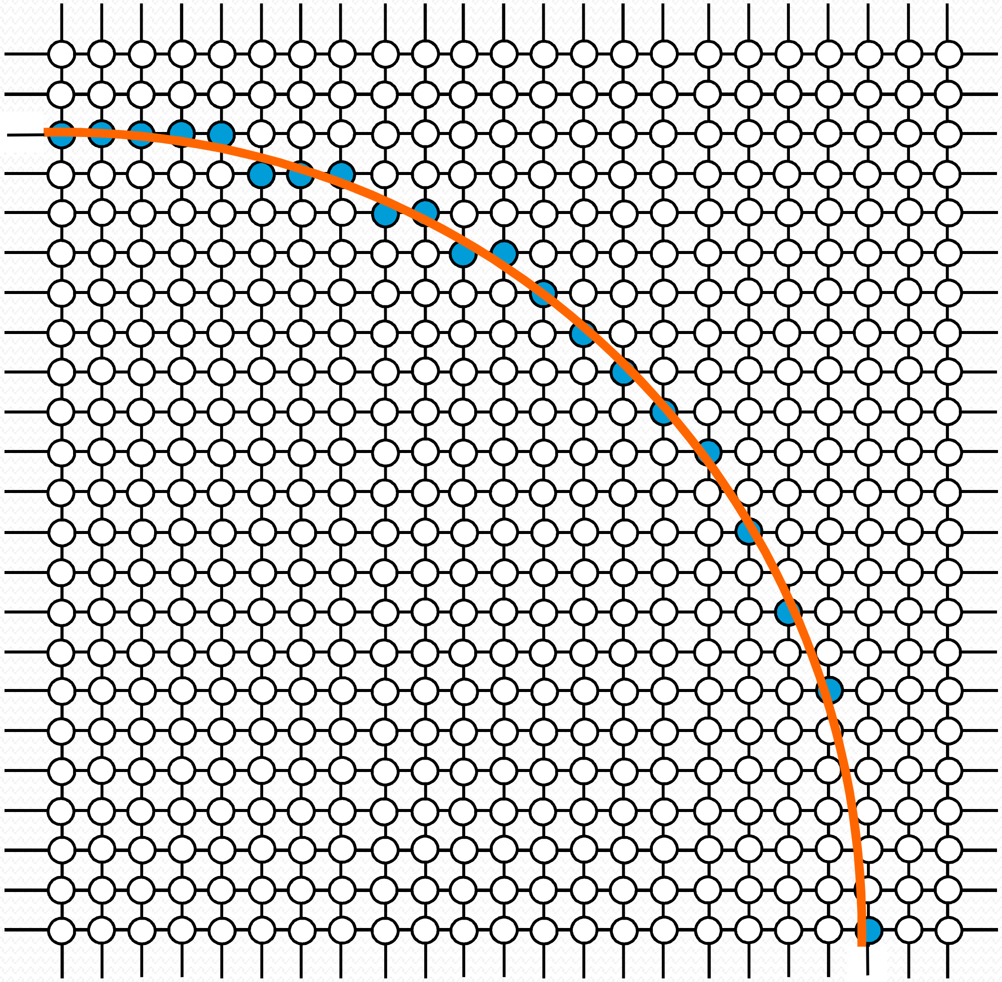
$$x^2 + y^2 = r^2$$

- where r is the radius of the circle

- So, we can write a simple circle drawing algorithm by solving the equation for y at unit x intervals using:

$$y = \pm\sqrt{r^2 - x^2}$$

A Simple Circle Drawing Algorithm (cont…)



$$y_0 = \sqrt{20^2 - 0^2} \approx 20$$

$$y_1 = \sqrt{20^2 - 1^2} \approx 20$$

$$y_2 = \sqrt{20^2 - 2^2} \approx 20$$

$$\vdots$$

$$y_{19} = \sqrt{20^2 - 19^2} \approx 6$$

$$y_{20} = \sqrt{20^2 - 20^2} \approx 0$$

Continue..

Parametric:

$$x = R\cos\theta$$
$$y = R\sin\theta$$

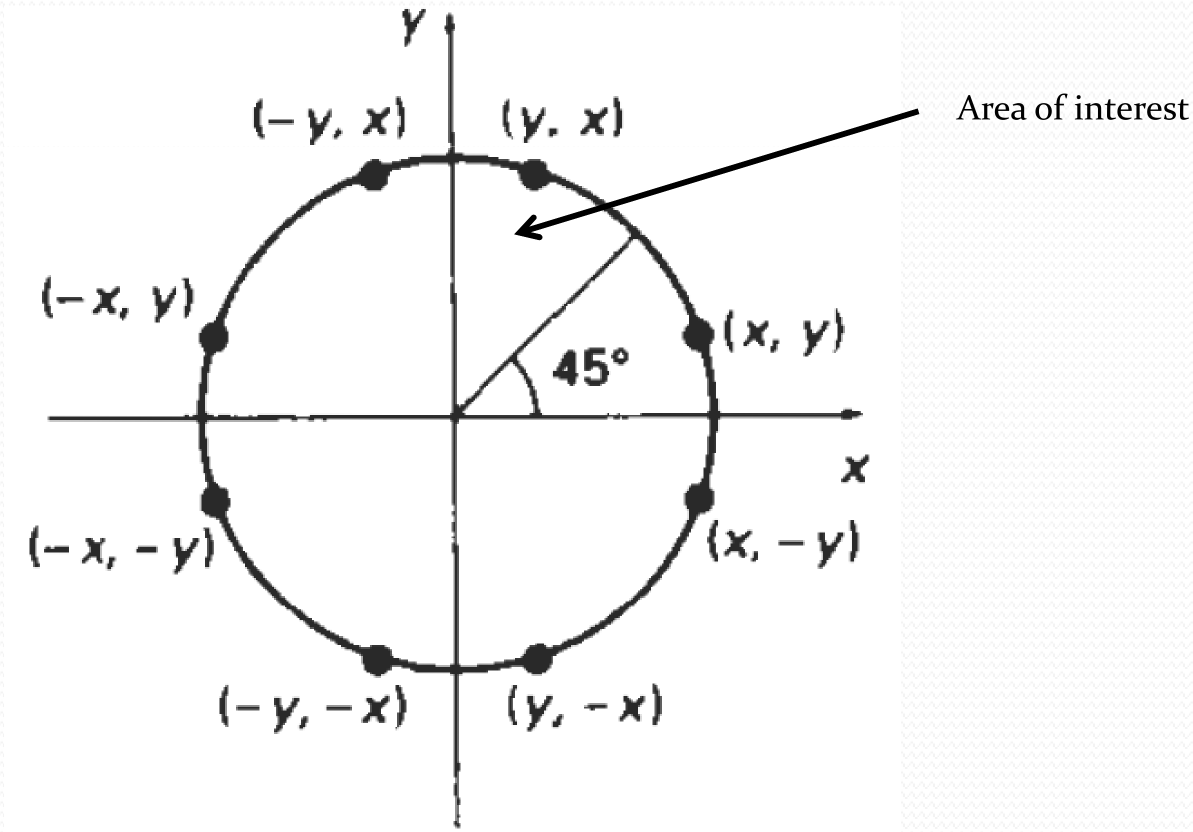
- by stepping the angle from 0 to 90

- avoids large gaps but still insufficient.

- However, unsurprisingly this is not a brilliant solution!
- Firstly, the resulting circle has large gaps where the slope approaches the vertical
- Secondly, the calculations are not very efficient
  - The square (multiply) operations
  - The square root operation – try really hard to avoid these!
- We need a more efficient, more accurate solution

Circle Symmetry (cont…)

- The first thing we can notice to make our circle drawing algorithm more efficient is that circles centred at (*0, 0*) have *eight-way symmetry*



Mid-Point Circle Algorithm

MID-POINT CIRCLE ALGORITHM

1. Input radius r and circle centre (x_c, y_c), then set the coordinates for the first point on the circumference of a circle centred on the origin as:

$$(x_0, y_0) = (0, r)$$

2. Calculate the initial value of the decision parameter as:

$$p_0 = \frac{5}{4} - r$$

3. Starting with $k = 0$ at each position x_k, perform the following test. If $p_k < 0$, the next point along the circle centred on (0, 0) is (x_{k+1}, y_k) and:

$$p_{k+1} = p_k + 2x_{k+1} + 1$$

Otherwise the next point along the circle is (x_k+1, y_k-1) and:

$$p_{k+1} = p_k + 2x_{k+1} + 1 - 2y_{k+1}$$

Continue..

4. Determine symmetry points in the other seven octants
5. Move each calculated pixel position *(x, y)* onto the circular path centred at (x_c, y_c) to plot the coordinate values:

$$x = x + x_c \qquad\qquad y = y + y_c$$

6. Repeat steps 3 to 5 until x >= y