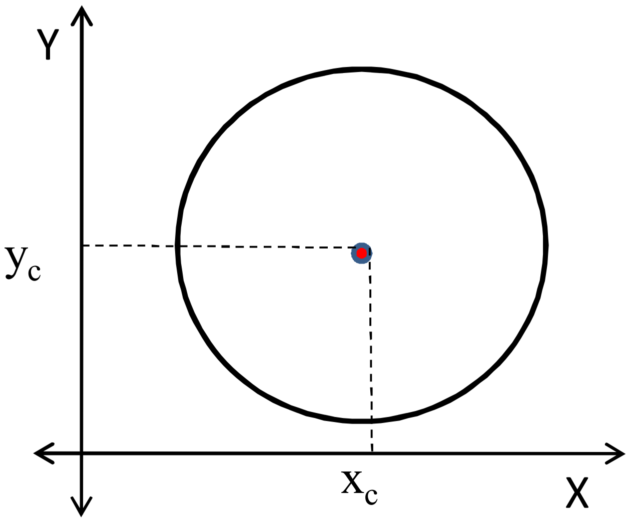
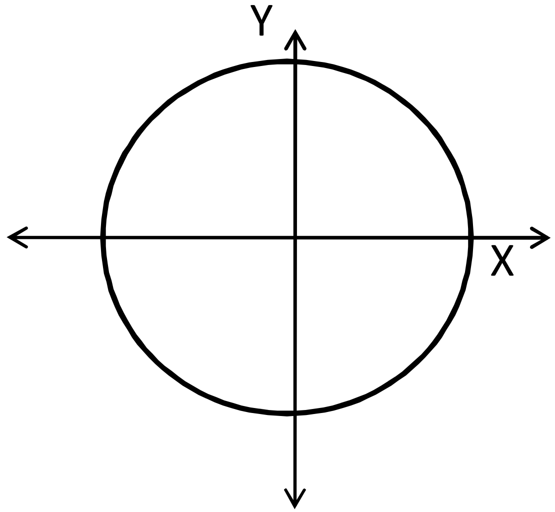
Midpoint circle drawing algorithm

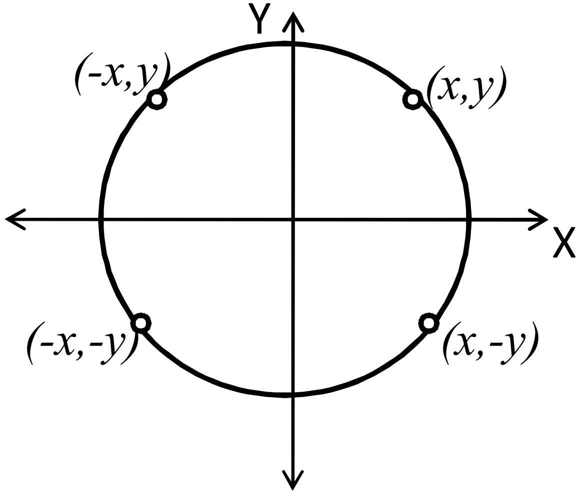
Midpoint circle drawing algorithm



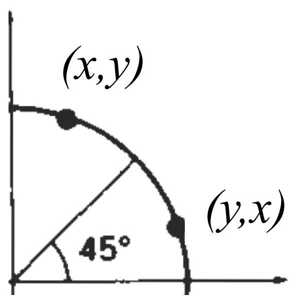
$$(x - x_c)^2 + (y - y_c)^2 = r^2$$

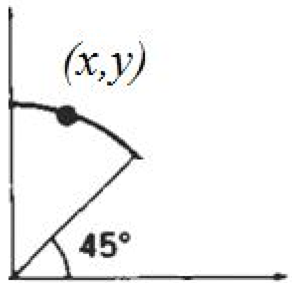


$$x^2 + y^2 = r^2$$



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





Hearn and Baker, "Computer Graphics" C version, second edition

$$f_{circle}(x, y) = x^2 + y^2 - r^2$$

$f_{circle}(x, y) = 0$ If (x,y) is on the boundary

$f_{circle}(x, y) < 0$ If (x,y) is inside the boundary

$f_{circle}(x, y) > 0$ If (x,y) is outside the boundary

We have plotted (x_k,y_k) and next pixel is (x_{k+1},y_{k+1}) where $x_{k+1} = x_k+1$

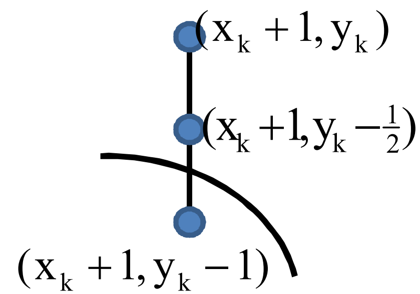
$y_{k+1} = y_k$ or y_k-1

$$p_K = f_{circle}(x_k + 1, y_k - \frac{1}{2})$$

$$= (x_k + 1)^2 + (y_k - \frac{1}{2})^2 - r^2$$

$$p_{K+1} = f_{circle}(x_{k+1} + 1, y_{k+1} - \frac{1}{2})$$

$$= (x_{k+1} + 1)^2 + (y_{k+1} - \frac{1}{2})^2 - r^2$$



$$p_{K+1} - p_K = ((x_{k+1} + 1)^2 + (y_{k+1} - \frac{1}{2})^2 - r^2) - ((x_k + 1)^2 + (y_k - \frac{1}{2})^2 - r^2)$$

$$\boxed{p_{K+1} = p_K + 2(x_k + 1) + (y^2{}_{k+1} - y^2{}_k) - (y_{k+1} - y_k) + 1}$$

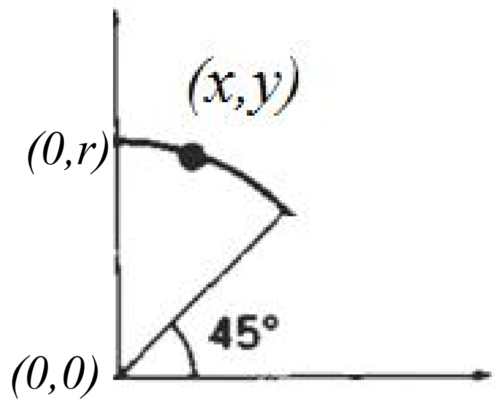
$$\text{If } p_K > 0 \qquad y_{k+1} = y_k - 1$$

$$\begin{aligned} p_{K+1} &= p_K + 2(x_k + 1) + (y^2{}_{k+1} - y^2{}_k) - (y_{k+1} - y_k) + 1 \\ &= p_K + 2(x_k + 1) + (1 - 2y_k) \\ &= p_K + 2x_{k+1} + (1 - 2y_k) \end{aligned}$$

$$\text{If } p_K < 0 \qquad y_{k+1} = y_k$$

$$\begin{aligned} p_{K+1} &= p_K + 2(x_k + 1) + (y^2{}_{k+1} - y^2{}_k) - (y_{k+1} - y_k) + 1 \\ &= p_K + 2(x_k + 1) + 1 \\ &= p_K + 2x_{k+1} + 1 \end{aligned}$$

$$\begin{aligned} p_0 &= f_{circle}(0 + 1, r - \frac{1}{2}) \\ &= (0 + 1)^2 + (r - \frac{1}{2})^2 - r^2 \\ &= \frac{5}{4} - r \end{aligned}$$



$$p_0 \approx (1 - r)$$

Hearn and Baker, "Computer Graphics" C version, Second edition

Midpoint Circle Algorithm

1. Input radius r and circle center (x_c, y_c), and obtain the first point on the circumference of a circle centered 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. At each x_k position, starting at $k = 0$, perform the following test: If $p_k < 0$, the next point along the circle centered 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

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

   where $2x_{k+1} = 2x_k + 2$ and $2y_{k+1} = 2y_k - 2$.
4. Determine symmetry points in the other seven octants.
5. Move each calculated pixel position (x, y) onto the circular path centered on (x_c, y_c) and plot the coordinate values:

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

6. Repeat steps 3 through 5 until $x \geq y$.

Hearn and Baker, "Computer Graphics" C version, Second edition