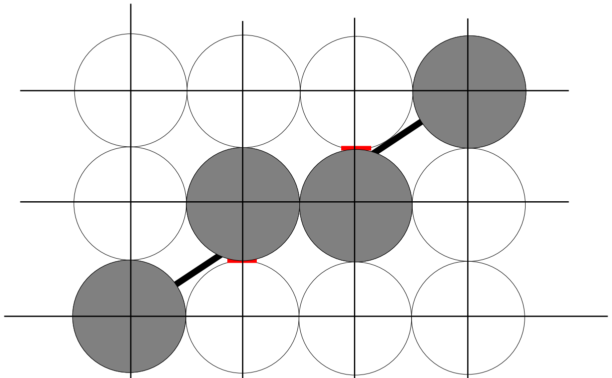
Bresenhamのアルゴリズムについて

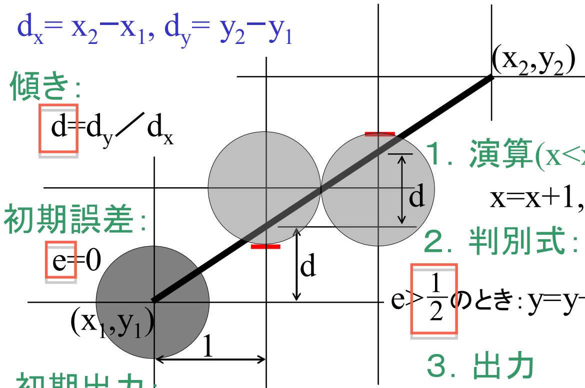


画素中間の上下どちらを真の直線が通過するか

1

Bresenhamアルゴリズムの考え方

$d_x = x_2 - x_1,\ d_y = y_2 - y_1$



傾き：

$\boxed{d} = d_y / d_x$

初期誤差：

$\boxed{e} = 0$

初期出力：

(x_1, y_1)をプロット, $x=x1, y=y1$

1．演算($x<x_2$の間)

$x=x+1,\ e=e+d$

2．判別式：

$e>\boxed{\frac{1}{2}}$のとき：$y=y+1,\ e=e-1$ とする

3．出力

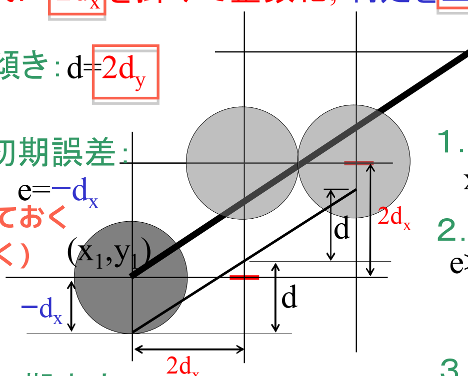
(x,y)をプロット

2

アルゴリズムの高速演算化

式に$\boxed{2d_x}$を掛けて整数化, 判定を$\boxed{正負で}$行う

符号ビット1bitだけを見ればいいので



傾き：$d=\boxed{2d_y}$

初期誤差：

$e=-d_x$

最初に下駄を履かせておく
(1/2だけずらしておく)

初期出力：

(x_1, y_1)をプロット, $x=x1, y=y1$

($x<x_2$の間)

1．繰り返し演算：

$x=x+1,\ e=e+d$

2．判別式：

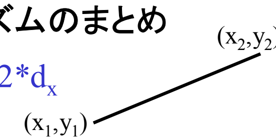
$e>\boxed{0}$のとき：$y=y+1$,
$e=e-\boxed{2d_x}$
とする

3．出力

(x,y)をプロット

3

Bresenhamのアルゴリズムのまとめ



$d_x = x_2 - x_1,\ d_y = y_2 - y_1,\ d_2 = 2*d_x$

1．初期化

(x_1, y_1)をプロット, $\boxed{d=2d_y,\ e=-d_x}$, **x=x1, y=y1**

2．繰り返し演算（$x < x_2$のあいだ中）

2.1 $x=x+1,\ e=e+d$

2.2 もし $e>0$ ならば、$y=y+1,\ e=e-d_2$

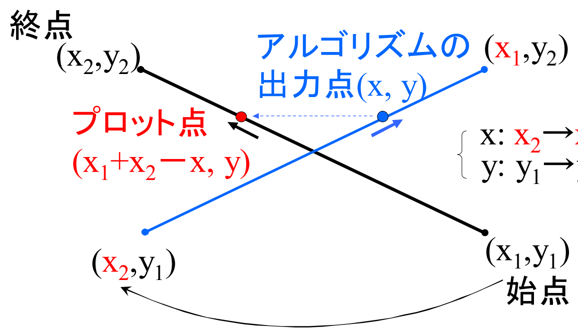
2.3 (x,y)をプロット

但し、$0 \leqq y_2 - y_1 \leqq x_2 - x_1$ のときのみ有効

4

適用領域の拡張のための変数交換(1/4)

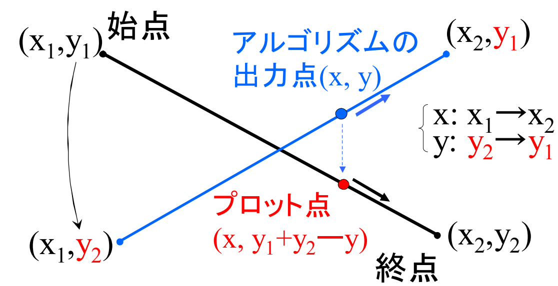
$\begin{cases} x_1 > x_2 \\ y_1 < y_2 \end{cases}$ の場合 ➡ $x_1 \longleftrightarrow x_2$ (交換)



5

適用領域の拡張のための変数交換(2/4)

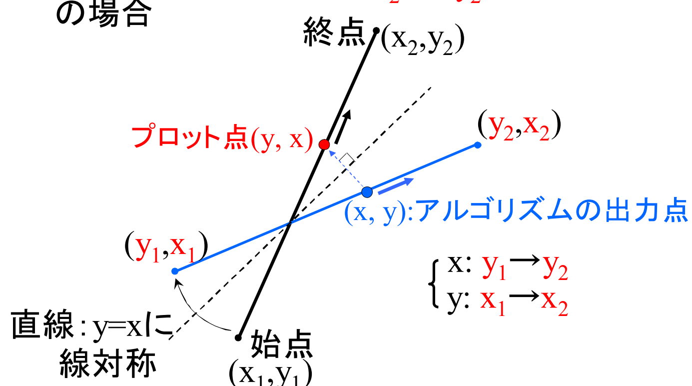
$\begin{cases} x_1 < x_2 \\ y_1 > y_2 \end{cases}$ の場合 ➡ $y_1 \longleftrightarrow y_2$ (交換)



6

適用領域の拡張のための変数交換(3/4)

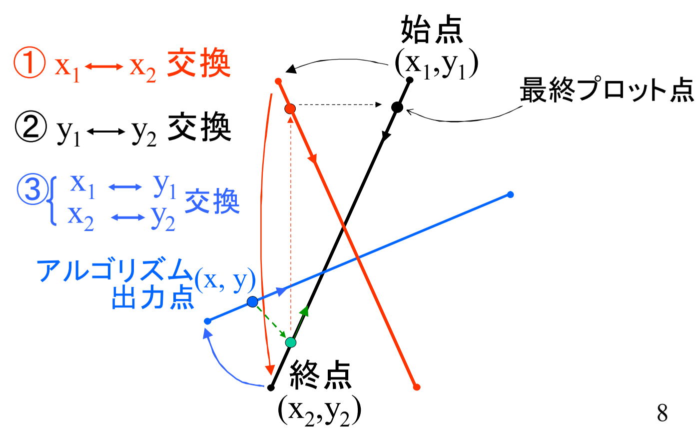
$y_2 - y_1 > x_2 - x_1 > 0$ の場合 ➡ $\begin{cases} x_1 \longleftrightarrow y_1 \\ x_2 \longleftrightarrow y_2 \end{cases}$ xとyを交換



7

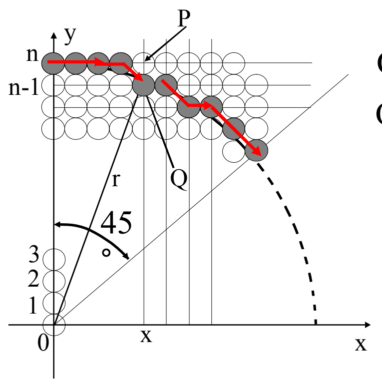
適用領域の拡張のための変数交換(4/4)

- 複数の置き換え



8

Michenerの円弧アルゴリズムの考え方



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

$OP^2=x^2+n^2>r^2$

$OQ^2=x^2+(n-1)^2<r^2$

判別式：

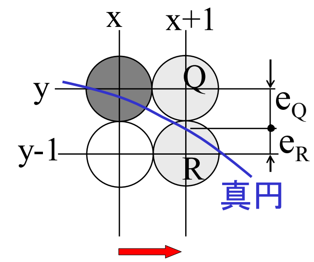
$D=(OP^2-r^2)+(OQ^2-r^2)$

正ならばQを、

負ならばPを選択

9

x位置の画素から(x+1)の画素の描画へ



$e_Q=(x+1)^2+y^2-r^2$

$e_R=(x+1)^2+(y-1)^2-r^2$

$d_1=e_Q+e_R \geqq 0$ならばR

<0ならばQ

xから(x+1)に移るときの判定式 d_1 は

$\boxed{d_1}=2(x+1)^2+y^2+(y-1)^2-2r^2$

２次式であり、計算時間がかかる

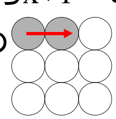
10

判定式を高速に計算するために

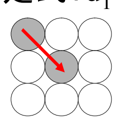
xからx+1、x+1からx+2の２段階の移動を考える

１．xからx+1への移動時の判定式：d_1

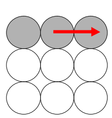
$d_1<0$のとき



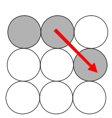
$d_1 \geqq 0$のとき



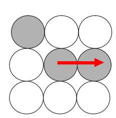
２．x+1からx+2への移動時の判定式：d_2



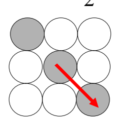
$d_1<0$, $d_2<0$



$d_1<0$, $d_2 \geqq 0$



$d_1 \geqq 0$, $d_2<0$

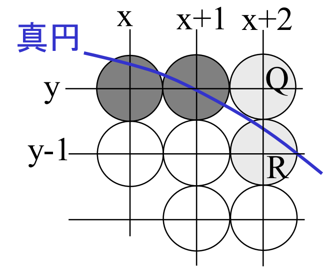


$d_1 \geqq 0$, $d_2 \geqq 0$

11

最適なy座標値の判定式(1/2)

$d_1<0$ のとき： x+1からx+2への移動時の判定式：d_2



$d_2=(x+2)^2+y^2-r^2$

$+(x+2)^2+(y-1)^2-r^2$

$=d_1+4x+6$

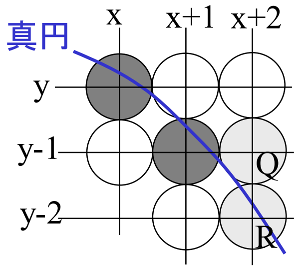
となる

なぜなら $d_1=2(x+1)^2+y^2+(y-1)^2-2r^2$

12

最適なy座標値の判定式(2/2)

$d_1 \geqq 0$ のとき：x+1からx+2への移動時の判定式：d_2



$$d_2=(x+2)^2+(y-1)^2-r^2$$
$$+(x+2)^2+(y-2)^2-r^2$$
$$=d_1+4(x-y)+10$$

となる

なぜなら　$d_1=2(x+1)^2+y^2+(y-1)^2-2r^2$

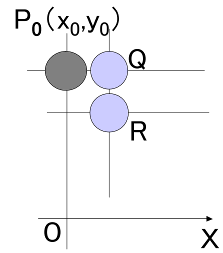
13

---

x＝0から$r/\sqrt{2}$まで、半径rの1/8円弧をプロットする

円の方程式：$x^2+y^2=r^2$

$x_0=0$；　$y_0=r$；



まず最初の判別式d_1を求める

$$d_1=e_{Q1}+e_{R1}$$
$$=(x_0+1)^2+y_0^2-r^2$$
$$+(x_0+1)^2+(y_0-1)^2-r^2$$
$$=1+r^2-r^2+1+(r-1)^2-r^2$$
$$(\because\ x_0=0,\ y_0=r)$$
$$=3-2r$$

14

Michenerアルゴリズムのまとめ
（中心が原点の場合に1／8円をプロット）

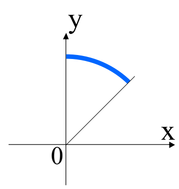
1．初期化　　x=0, y=r, d=3−2r, (x,y)をプロット

2．繰り返し計算(x<yが成り立つ間)

2.1 ｛もし(d<0)ならば、d=d+4x+6
　　　そうでなければ、d=d+4(x−y)+10, y=y−1

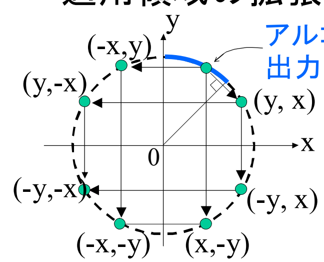
2.2　x=x+1

2.3　(x,y)をプロット



15

適用領域の拡張のための変数追加



(x, y)が円周上の点ならば
(±x, ±y),(±x, ±y)も
同様に円周上の点

中心が(x_0,y_0)の円を描くには(x,y)の代わりに、

$(x_0\pm x,\ y_0\pm y)$, $(x_0\pm y, y_0\pm x)$にプロットする

複号任意 (合計8通りある)

16