CSE423

Friday, November 3, 2023 1:48 AM

✓ Frame buffer : Bitmap. Part of video memory. Read-Write simultaneously. Dual access.

✓ Video controller : Reads

✓ **Line drawing algorithm :-**

- Digital Differential Analyzer (DDA):-

Uses Slope $= \frac{dy}{dx}$

So, steps :-

1. Input (x_0, y_0), (x_1, y_1)

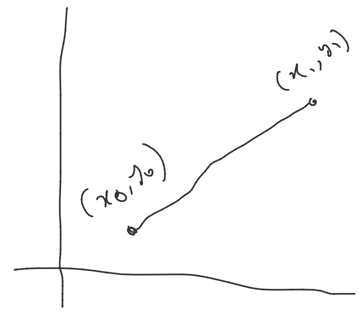
2. $dx = x_1 - x_0$
   $dy = y_1 - y_0$

3. controller $= \max(|dx|, |dy|)$

4. dx /= controller
   dy /= controller

5. $x = x_0$
   $y = y_0$

6. for $i = 0$ --- controller :
   drawpixel(round(x), round(y))
   x += dx
   y += dy




```python
x0, y0, x1, y1, = map(int, input().split())

print(f"(x0, y0) = ({x0}, {y0})\n(x1, y1) = ({x1}, {y1})")

dx = x1 - x0
dy = y1 - y0

print(f"dx = {dx}")
print(f"dy = {dy}")

controller = max(abs(dx), abs(dy))

print(controller)

dx /= controller
dy /= controller

print(f"dx = {dx}")
print(f"dy = {dy}")
print("\n\n")

x = x0
y = y0

for i in range(0, controller+1):
    print(round(x), round(y))
    x += dx
    y += dy
```

For example :-

$(-3, 5)$ to $(3, -5)$

$\Rightarrow dx = 3 - (-3) = 6$

$dy = -5 - 5 = -10$

$dy = -5-5 = -10$

$\text{controller} = \max(|6|, |-10|) = 10$

$dx \mathrel{/}= 10 \longrightarrow 6/10 = 0.6$

$dy \mathrel{/}= 10 \longrightarrow -10/10 = -1$

i	Coordinates	x	y
0	-3, 5	-3	5
1	-2, 4	-2.4	4
2	-2, 3	-1.8	3
3	-1, 2	-1.2	2
4	-1, 1	-0.6	1
5	0, 0	0	0
6	1, -1	0.6	-1
7	1, -2	1.2	-2
8	2, -3	1.8	-3
9	2, -4	2.4	-4
10	3, -5	3	-5

• Midpoint line algorithm :-

Floating point calculations are time consuming. So, we use integer values instead.

We know,

$$y = mx + c$$

$$\Rightarrow y = \frac{dy}{dx} x + c$$

$$\Rightarrow y = \frac{dy \cdot x + dx \cdot c}{dx}$$

$$\Rightarrow y\, dx = x\, dy + c \cdot dx$$

$$\Rightarrow y\, dx - x\, dy - c \cdot dx = 0$$

$$\Rightarrow -dy \cdot x + dx \cdot y - c \cdot dx = 0$$

$A = dy$

$B = -dx$

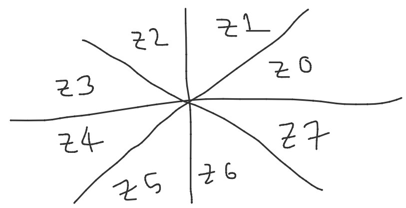
$C = c \cdot dx$

$$\Rightarrow -dy \cdot x + dx \cdot y - c \cdot dx = 0$$
$$\Rightarrow dy \cdot x - dx \, y + c \cdot dx = 0$$
$$\therefore A x + B y + C = 0$$

$C = c \cdot dx$

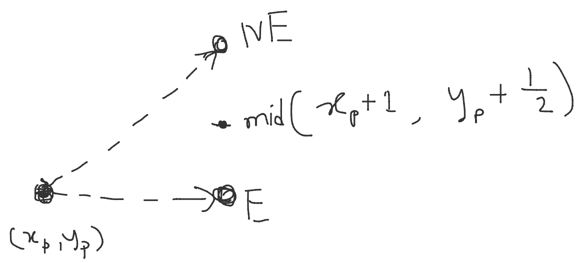
Therefore, $f(x,y) = Ax + By + c$

If a point is exactly on the line, the equation will return 0.



Considering zone 0

$\therefore 0 \leq m \leq 1$



$\therefore$ For mid, $f(x_p+1,\ y_p+\frac{1}{2}) = A(x_p+1) + B(y_p+\frac{1}{2}) + C$

$$= A x_p + A + B y_p + \frac{B}{2} + C$$
$$= f(x_p, y_p) + A + \frac{B}{2}$$

$\therefore d_{mid} = A + \frac{B}{2}$

$$= dy - \frac{dx}{2}$$

$\therefore$ for E, $f(x_p+1,\ y_p) = Ax_p + A + By_p + C = f(x_p, y_p) + A$

$\therefore d_E = A$

$$= dy$$

For NE, $f(x_p+1,\ y_p+1) = Ax_p + A + By_p + B + C = f(x_p, y_p) + A + B$

$\therefore d_{NE} = A + B$

$$= dy - dx$$

For NE, $f(x_p+1, y_p+1) = Ax_p + A + By_p + B \ldots$

$\therefore d_{NE} = A + B$
$= dy - dx$ ✓

Then, $d_{mid} < 0 \longrightarrow dE$
else $\longrightarrow dNE$

Therefore, according to Bresenham,

$d_{init} = d_{mid}$

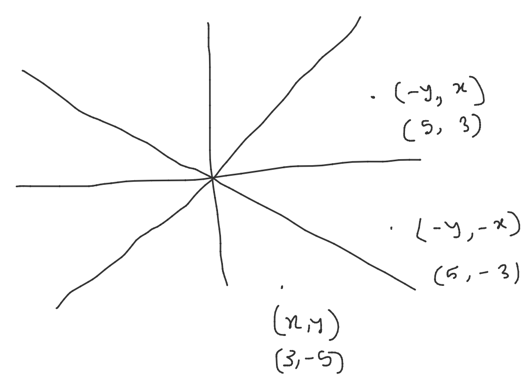
Then, if $d_{mid} < 0$:- $d_{new} = d_{mid} + d_E$
else :- $d_{new} = d_{mid} + d_{NE}$

$\left.\begin{array}{l} \\ \end{array}\right\} d_{mid} = d_{new}$

Ex :-

$(-3, 5)$ to $(3, -5)$

$\Rightarrow \left.\begin{array}{l} dx = 6 \\ dy = -10 \end{array}\right\}$ Zone - 6



After converting to Zone 0,

$(-3, 5) \longrightarrow (-5, -3)$
$(3, -5) \longrightarrow (5, 3)$

$dx = 5 - (-5) = 10$
$dy = 3 - (-3) = 6$

$\therefore d_{init} = 2dy - dx$ $\quad \left[\because d_{mid} = dy - \frac{dx}{2}\right]$
$= 2 \times 6 - 10$
$= 2$

$\therefore d_E = 2\,dy$

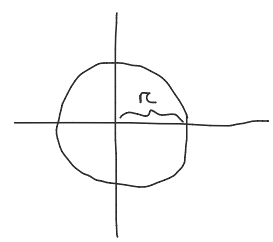
$= 12$

$\therefore d_{NE} = 2(d_y - d_x)$

$= -8$

i	Converted (x,y)	Real (x,y)	d	dE / dNE
0	−5, −3	−3, 5	2	d_{NE}
1	−4, −2	−2, 4	2−8 = −6	d_E
2	−3, −2	−2, 3	−6+12 = 6	d_{NE}
3	−2, −1	−1, 2	6−8 = −2	d_E
4	−1, −1	−1, 1	−2+12 = 10	d_{NE}
5	0, 0	0, 0	10−8 = 2	d_{NE}
6	1, 1	1, −1	2−8 = −6	d_E
7	2, 1	1, −2	−6+12 = 6	d_{NE}
8	3, 2	2, −3	6−8 = −2	d_E
9	4, 2	2, −4	−2+12 = 10	d_{NE}
10	5, 3	3, −5	10−8 = 2	d_{NE}

✓ Circle Drawing algorithm :-



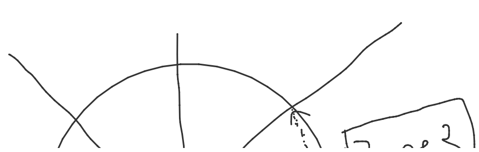
Equation of a circle : $x^2 + y^2 = r^2$

$\Rightarrow x^2 + y^2 - r^2 = 0$

$\therefore f(x,y) = x^2 + y^2 - r^2$ $\begin{cases} = 0 \rightarrow \text{on the circle} \\ > 0 \rightarrow \text{out of the circle} \\ < 0 \rightarrow \text{inside of the circle} \end{cases}$

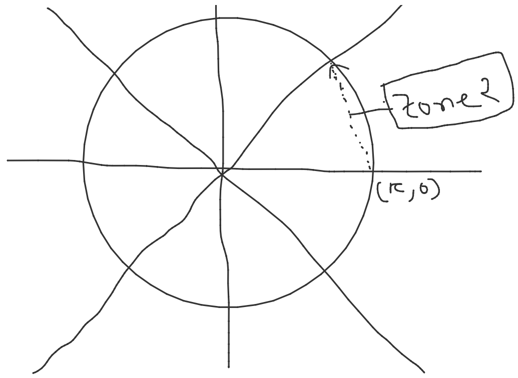
We will now try to draw the circle only in 1 zone.

Later, transform the points using 8 way symmetry.



$m \rightarrow f(x_p - \frac{1}{2}, y_p + 1) = (x_p - \frac{1}{2})^2 + (y_p + 1)^2 - r^2$

$= x_p^2 - x_p + \frac{1}{4} + y_p^2 + 2y_p + 1 - r^2$



$$= x_p^2 - x_p + \frac{1}{4} + y_p^2 + 2y_p + 1 - r^2$$

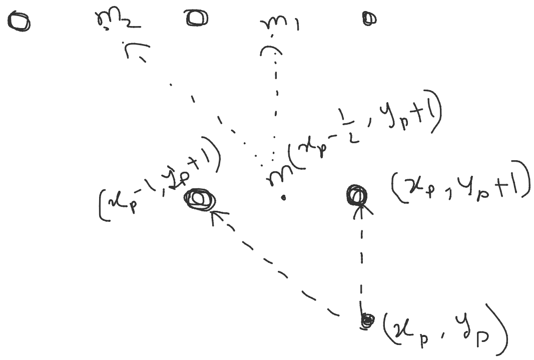
$$= (x_p^2 + y_p^2 - r^2) + \left(-x_p + 2y_p + \frac{5}{4}\right)$$

$$\therefore d_m = -x_p + 2y_p + \frac{5}{4}$$

For, $(r, 0)$, $d_m = -r + 2\times 0 + \frac{5}{4}$

$$\therefore d_m = \boxed{-r + \frac{5}{4}}$$ ✓

dinit



$d_N \Rightarrow f(m_1) - f(m)$

$$\Rightarrow \left(x_p - \frac{1}{2}\right)^2 + (y_p + 2)^2 - r^2$$

$$(-)\ \left(x_p - \frac{1}{2}\right)^2 + (y_p + 1)^2 - r^2$$

$$y_p^2 + 4y_p + 4 - y_p^2 - 2y_p - 1$$

$$= 2y_p + 3$$ ✓

$d < 0 \rightarrow d_N$

else $\rightarrow d_{NW}$

$d_{NW} \Rightarrow f(m_2) - f(m)$

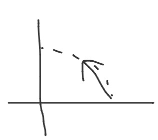
$$\left(x_p - \frac{3}{2}\right)^2 + (y_p + 2)^2 - r^2$$

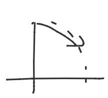
$$(-)\ \left(x_p - \frac{1}{2}\right)^2 + (y_p + 1)^2 - r^2$$

$$x_p^2 - 3x_p + \frac{9}{4} + y_p^2 + 4y_p + 4 - x_p^2 + x_p - \frac{1}{4} - y_p^2 - 2y_p - 1$$

$$= -2x_p + 2y_p + 5$$

Point to focus:- Loop controller will be according to zone (Resident)

 $\rightarrow$ Z0 $\rightarrow$ while $x > y$

 $\rightarrow$ Z1 $\rightarrow$ while $y > x$

For example:-

... (2,3) . points in region 1 ?

For example:

$R = 10$, $center = (2,3)$, points in region 1 ?

$\Rightarrow d_{init} = -4r + 5 = -4 \times 10 + 5 = -35$

$d_N = 4(2y_p + 3) = 8y_p + 12$

$d_{NW} = 4(-2x_p + 2y_p + 5) = -8x_p + 8y_p + 20$

$x = 10$, $y = 0$

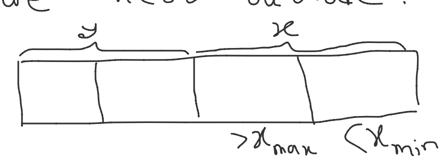
Pixel	(x,y)	Translated $(x+2, y+3)$	d	d_N / d_{NW}
1	10, 0	12, 3	-35	d_N
2	10, 1	12, 4	$-35 + (8 \times 0 + 12) = -23$	d_N
3	10, 2	12, 5	$-23 + (8 \times 1 + 12) = -3$	d_N
4	10, 3	12, 6	$-3 + (8 \times 2 + 12) = 25$	d_{NW}
5	9, 4	11, 7	$25 + (-8 \cdot 10 + 8 \cdot 3 + 20) = -11$	d_N
6	9, 5	11, 8	$-11 + (8 \times 4 + 12) = 33$	d_{NW}
7	8, 6	10, 9	$33 + (-8 \times 9 + 8 \cdot 5 + 20) = 21$	d_{NW}
8	7, 7	9, 10	$21 + (-8 \cdot 8 + 8 \cdot 6 + 20) = 25$	d_{NW}

✓ Line clipping algo :-

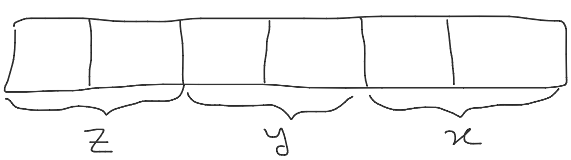
• Cohen-Sutherland :-

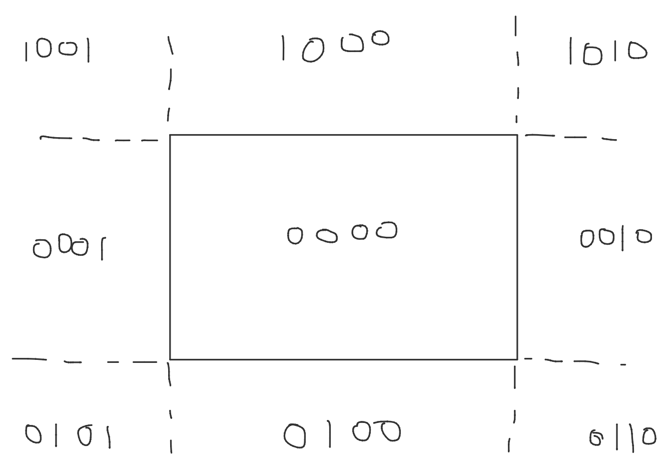
Firstly we need outcode.

For 2D :



For 3D:





Algo :-

1. Calculate 2 endpoints :- e_1 , e_2

2. Calculate out code for both :- O_1 , O_2

3. Check :- if , $O_1 = O_2 = 0000$:

Trivially accept ✓

else if, (O_1 Bitwise AND O_2) != 0 :

Trivially reject ✗

else :

Partially accepted

choose one endpoint

if y_{max} crossed:

$y = y_{max}$

$x = x_1 + \frac{x_2 - x_1}{y_2 - y_1} \cdot (y - y_1)$

elif y_{min} crossed:

$y = y_{min}$

$x = x_1 + \frac{x_2 - x_1}{y_2 - y_1} \cdot (y - y_1)$

elif x_{max} crossed :-

$x = x_{max}$

$y = y_1 + \frac{y_2 - y_1}{x_2 - x_1} \cdot (x - x_1)$

elif x_{min} crossed :-

$x = x_{min}$

$y = y_1 + \frac{y_2 - y_1}{x_2 - x_1} \cdot (x - x_1)$

For example

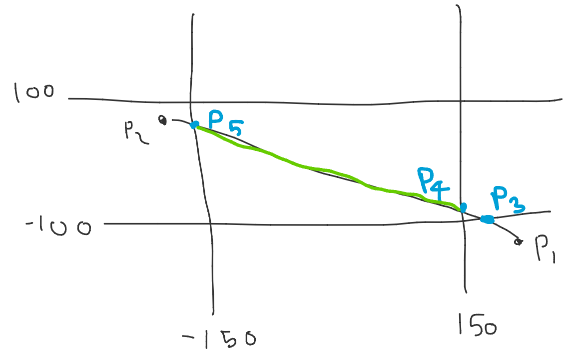
$x_{max} = 150$, $x_{min} = -150$, $y_{max} = 100$, $y_{min} = -100$

$P_1 (250, -120)$, $P_2 (-180, 88)$

$\Rightarrow$ Outcodes :-

$C_1 = 0110$

$C_2 = 0001$



$C_1 \,\&\, C_2 = 0000$

$\therefore$ So, they are partially accepted

Lets choose C_1, which crosses y_{min}

So, new $y = y_{min} = -100$

$$x = x_1 + \frac{x_2 - x_1}{y_2 - y_1} \cdot (y - y_1) = 250 + \frac{-180 - 250}{88 - (-120)} \cdot (-100 - (-120)) = 208.7$$

$\therefore P_3 = (208.7, -100)$

$O_3 = 0010$

Lets choose P_3, which crosses x_{max}

So, new $x = x_{max} = 150$

$$y = y_1 + \frac{y_2 - y_1}{x_2 - x_1} \cdot (x - x_1) = -100 + \frac{88 - (-100)}{-180 - 208.7} (150 - 208.7) = -71.6$$

$P_4 (150, -71.6)$

$O_4 = 0000$

Now choose, P_2, which crosses x_{min}

So, new $x = x_{min} = -150$

$$y = -71.6 + \frac{88 - (-71.6)}{-180 - 150} \cdot (-150 - 150) = 73.5$$

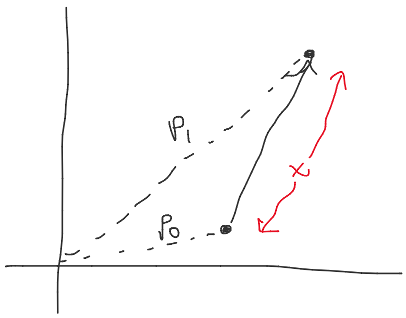
$P_5 = (-150, 73.5)$

$O_5 = 0000$

P_4 to P_5 should be the selected line.

- Cyrus Beak algo :-

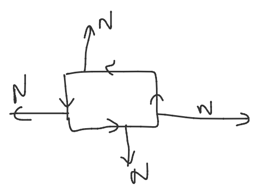
Parametric equation :-

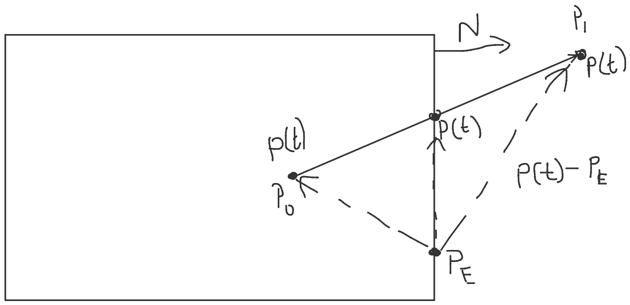


$$P(t) = P_0 + t\,(P_1 - P_0)$$

$$0 \leq t \leq 1$$

Normal :-





Position

$(P(t) - P_E) \cdot N = 0 \longrightarrow$ On edge

" $> 0 \longrightarrow$ outside

" $< 0 \longrightarrow$ inside

Movement

$(P_1 - P_0) \cdot N = 0 \longrightarrow$ Parallel

$> 0 \longrightarrow$ Leaving

$< 0 \longrightarrow$ Entering

Intersecting t :-

$$(P(t) - P_E) \cdot N = 0$$

$$\Rightarrow (P_0 + t(P_1 - P_0) - P_E) \cdot N = 0$$

$$\Rightarrow (P_0 - P_E) \cdot N + t\,(P_1 - P_0) \cdot N = 0$$

$$\Rightarrow \boxed{t = -\frac{(P_0 - P_E) \cdot N}{(P_1 - P_0) \cdot N}}$$

Algorithm :-

1. $t_E = 0$ # $t_{entering}$

   $t_L = 1$ # $t_{leaving}$

1. $t_E = 0$ # $t_{entering}$

$t_L = 1$ # $t_{leaving}$

2. For each edge:

if $(P_1 - P_0) \cdot N = 0$:

Parallel, so break

else:

calculate $t = -\dfrac{(P_0 - P_E) \cdot N}{(P_1 - P_0) \cdot N}$

if $(P_1 - P_0) \cdot N > 0$: # Leaving t_L

if $t < t_L$:

$t_L = t$

else: # entering t_E

if $t > t_E$:

$t_E = t$

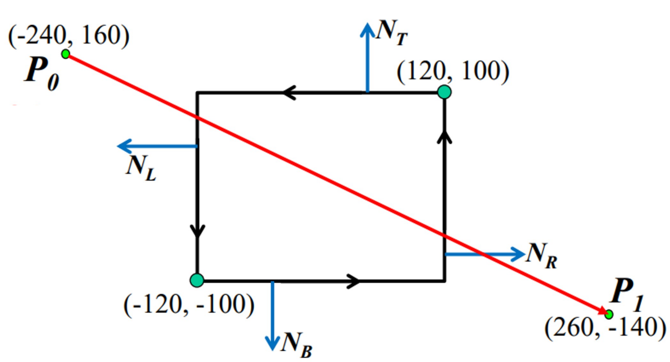
3. if $t_E > t_L$:

trivial reject

else:

return $P(t_E)$ and $P(t_L)$

For example:-



what are the coordinates after clipping?

$\Rightarrow \quad t_E = 0$

$t_L = 1$

For left edge :

$N = -\hat{i}$

$P_1 - P_0 = 500\hat{i} - 300\hat{j}$

Not parallel, so,

$P_0 - P_E = -120\hat{i} + 260\hat{j}$

$$\therefore t = -\frac{(-120\hat{i} + 260\hat{j})\cdot(-\hat{i})}{(500\hat{i} - 300\hat{j})\cdot(-\hat{i})}$$

$$= -\frac{120}{-500} = 0.24$$

As $(P_1 - P_0)\cdot N = -500 < 0$

Its an entering edge.

So, $t > t_E$, $t_E = 0.24$

For right edge :-

$N = \hat{i}$

$P_1 - P_0 = 500\hat{i} - 300\hat{j}$

Not parallel, so,

$P_0 - P_E = -360\hat{i} + 60\hat{j}$

$$\therefore t = -\frac{(-360\hat{i} + 60\hat{j})\cdot\hat{i}}{(500\hat{i} - 300\hat{j})\cdot\hat{i}}$$

$$= -\frac{-360}{500} = 0.72$$

As, $(P_1 - P_0)\cdot N = 500 > 0$

Its a leaving edge.

So, $t < t_L$, $t_L = 0.72$

Top and bottom will produce unnecessary values.

$\therefore P(t_E = 0.24) = (-240\hat{i} + 160\hat{j}) + 0.24(500\hat{i} - 300\hat{j})$

$= -120\hat{i} + 88\hat{j} = (-120, 88)$

$\therefore P(t_L = 0.72) = 120\hat{i} - 56\hat{j} = (120, -56)$