2D Graphics

Computer Graphics

2D Graphics

Computer Graphics

CONTENTs

- Vẽ đối tượng đồ họa cơ sở
- Xén hình
- Tô màu
- Vẽ chữ và dựng Font
- Các phép biến đổi 2D

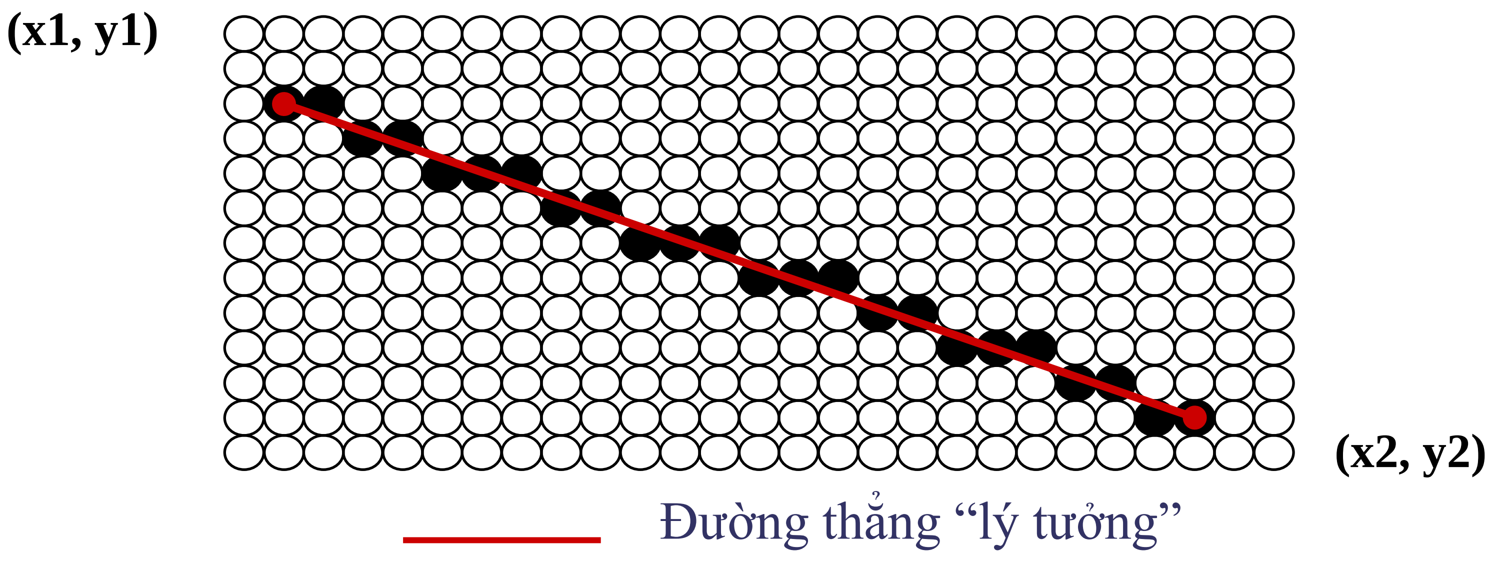
CONTENTs

- **Vẽ đối tượng đồ họa cơ sở**
- Xén hình
- Tô màu
- Vẽ chữ và dựng Font
- Các phép biến đổi 2D

- Thuật toán vẽ đoạn thẳng
- Thuật toán vẽ đường tròn
- Thuật toán vẽ ellipse
- Thuật toán vẽ các đường cong y=f(x)

Bài toán vẽ đường

- Biến đổi đường liên tục thành rời rạc (Sampling)



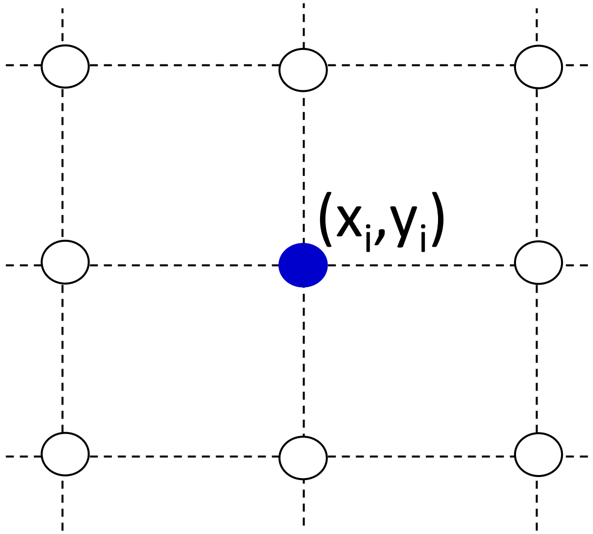
Yêu cầu

- Hình dạng liên tục, độ dày và độ sáng đều
- Các pixel gần đường "lý tưởng" được hiển thị
- Vẽ nhanh

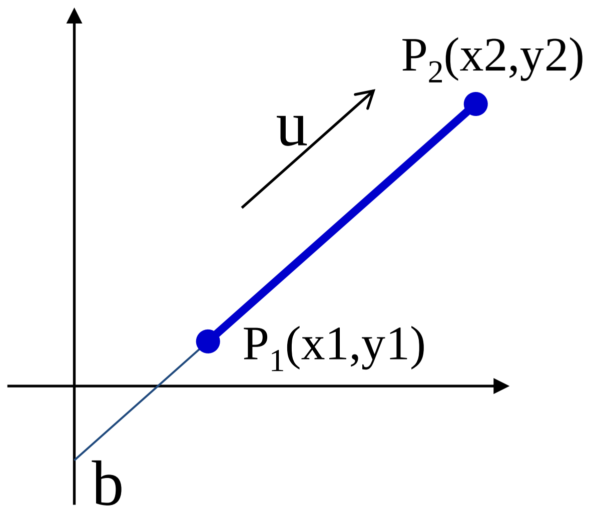
Bài toán: **Bước thứ i xác định được toạ độ nguyên (x_i,y_i).**
Tại bước i+1, điểm nguyên (x_i+1,y_i+1) xác định thế nào?

Để đối tượng hiển thị trên lưới nguyên được liền nét, các điểm (x_{i+1},y_{i+1}) có thể chọn là 1 trong 8 điểm quanh (x_i,y_i)

hay: $(x_{i+1},y_{i+1}) = (x_i \pm 1, y_i \pm 1)$.



Vẽ đoạn thẳng



- Biểu diễn dạng tường minh

  (y-y1)/(x-x1) = (y2-y1)/(x2-x1) hay y = mx + b

  - m = (y2-y1)/(x2-x1)
  - b = y1- m.x1
  - Δy = m Δx

- Biểu diễn dạng không tường minh

  (y2-y1)x - (x2-x1)y + x2y1 - x1y2 = 0 hay Ax + By + C = 0

  - A = (y2-y1)
  - B = -(x2-x1)
  - C = x2y1 - x1y2

- Biểu diễn dạng tham số

  P(u) = P1 + u(P2 - P1) với u $\in$[0,1] hay X(u) = x1 + u(x2 - x1)

  Y(u) = y1 + u(y2 - y1)

Vẽ đoạn thẳng

- **Bài toán:** Vẽ đoạn thẳng qua 2 điểm (x_1,y_1) và (x_2,y_2)

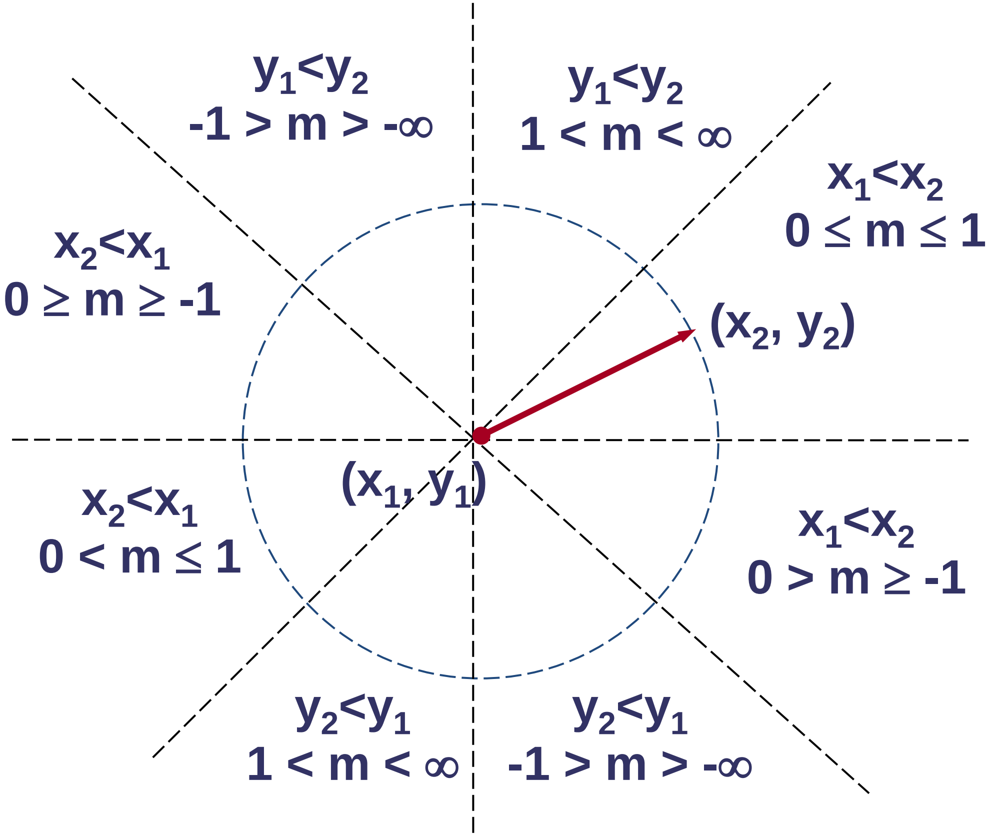
- **Giải pháp**
  - Cho tọa độ x (hoặc y) biến đổi mỗi lần 1 đơn vị:

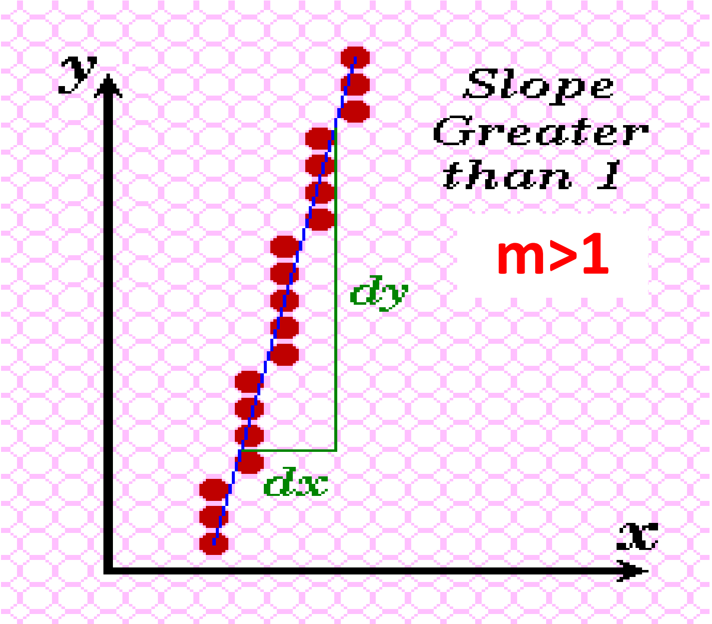
    $$x_{i+1}=x_i+1 \quad (\text{hoặc } y_{i+1}=y_i+1)$$

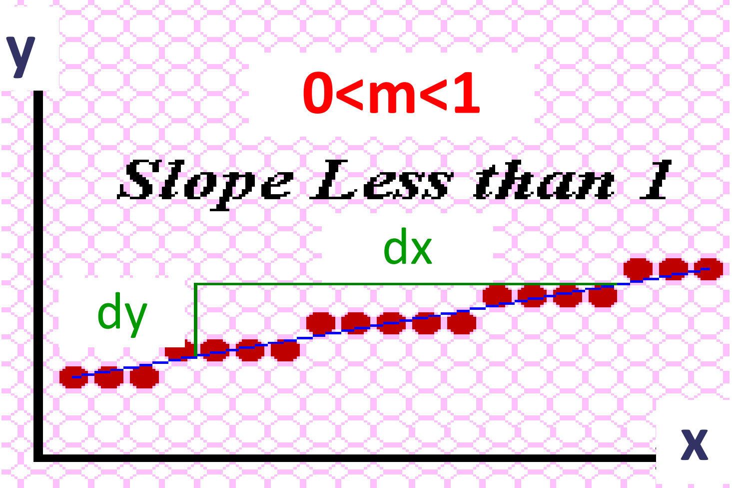
  - Tính tọa độ nguyên y (hoặc x) sao cho gần với toạ độ thực nhất.

- **Việc quyết định chọn x hay y biến đổi phụ thuộc vào độ dốc (hệ số góc *m*) của đường thẳng**

Vẽ đoạn thẳng







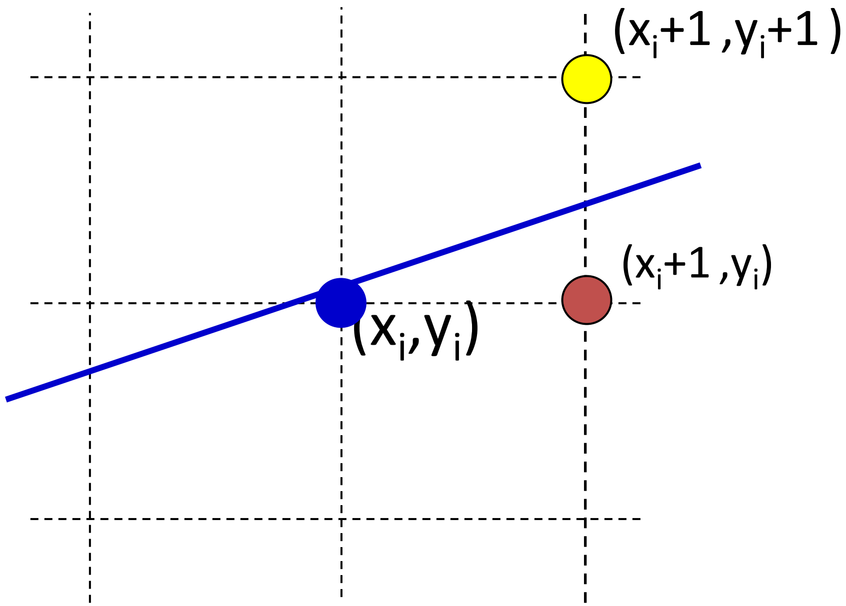
Nếu $|dx| > |dy|$: $y=f(x)$

Nếu $|dx| < |dy|$: $x=f^{-1}(y)$

Vẽ đoạn thẳng

- **Xét đoạn thẳng có hệ số góc 0<m<1.**
  - Giả sử (x_i,y_i) là điểm đã xác định được ở bước thứ i
  - Điểm cần chọn (x_{i+1},y_{i+1}) ở bước thứ i+1 sẽ là:

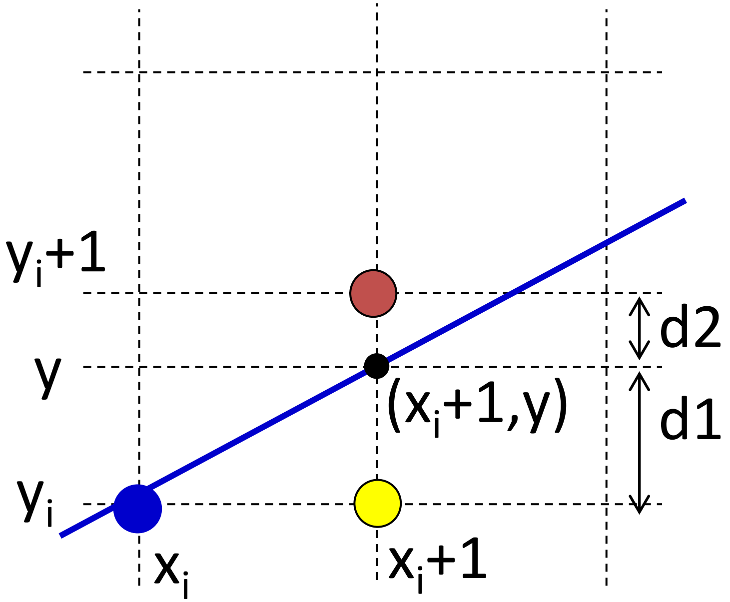
$$\begin{cases} x_{i+1} = x_i + 1 \\ y_{i+1} = \{y_i , y_i + 1\} \end{cases}$$



Vẽ đoạn thẳng - Thuật toán Bresenham

- Gọi $(x_i+1,y) = (x_i+1,m(x_i+1)+b)$ là điểm thuộc đường thẳng
- Bước i+1, chọn (x_{i+1},y_{i+1}) phụ thuộc vào d1 va d2 hay dấu d1-d2.
  - Nếu d1< d2 , chọn $y_{i+1}= y_i$
  - Nếu d1>= d2 , chọn $y_{i+1} = y_i + 1$
- Có: $d1=y - y_i \quad =m(x_i + 1) + b - y_i$

  $d2=(y_i+1)-y =y_i +1-m(x_i + 1)-b$



- Gọi D = d1-d2 = $2mx_i-2y_i+2m+2b-1$
- Trong biểu thức D, m là số thực nên việc tính toán

  với biểu thức này khá chậm. Do đó, thay vì xét dấu D ta xét dấu p_i

Vẽ đoạn thẳng - Thuật toán Bresenham

- p_i = Dx(d1-d2)   (vì Dx >0).
- Thay m=Dy/Dx, ta có:

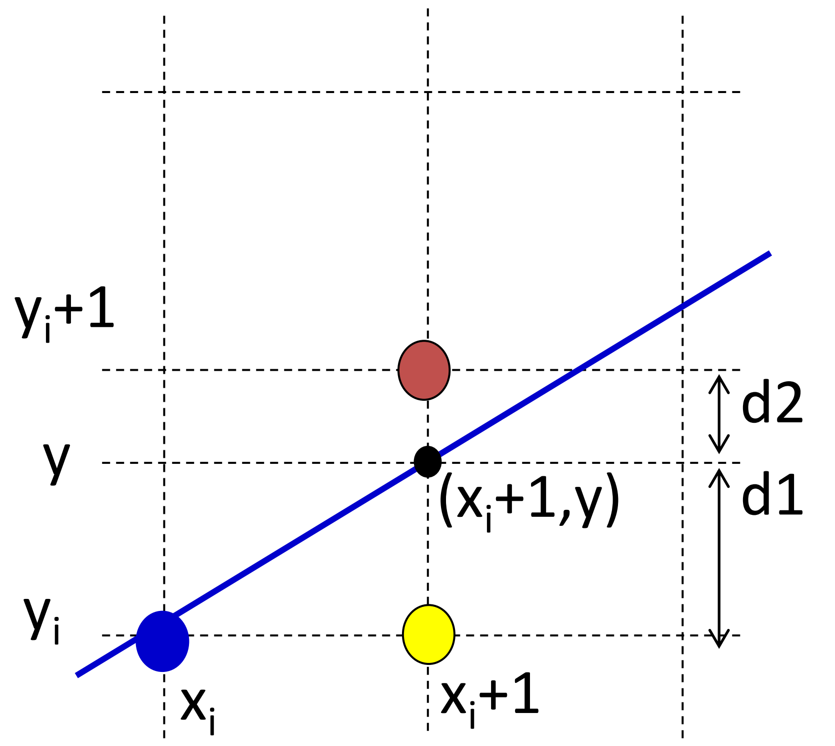
  $p_i = 2Dyx_i - 2Dxy_i + c$

  với c = 2Dy+(2b-1)Dx
- Vậy:
  - Nếu $p_i<0$ hay d1<d2,

    chọn $y_{i+1} = y_i$
  - Nếu $p_i>=0$ hay d1>=d2,

    chọn $y_{i+1} = y_i + 1$

D = d1-d2 = $2mx_i - 2y_i + 2m + 2b - 1$



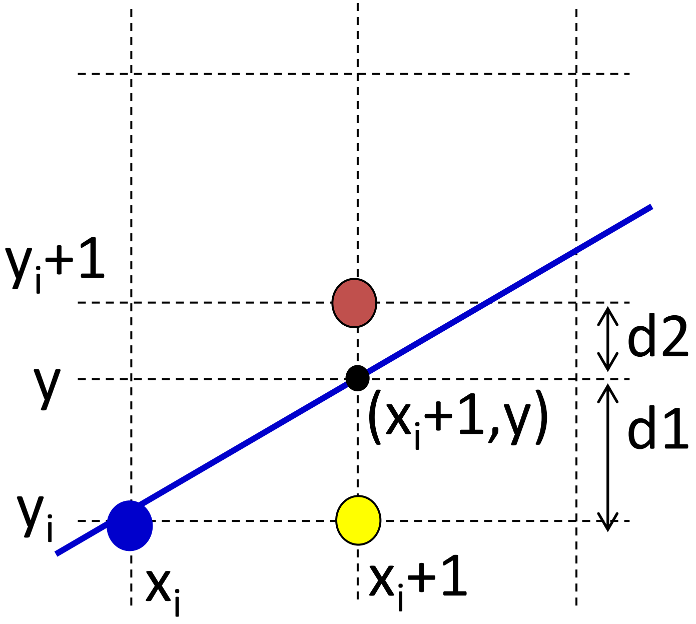
Vẽ đoạn thẳng - Thuật toán Bresenham

- Xét $p_{i+1} - p_i = (2Dyx_{i+1} - 2Dxy_{i+1} + c) - (2Dyx_i - 2Dxy_i + c)$

  $= 2Dy - 2Dx(y_{i+1} - y_i)$ (do $x_{i+1} = x_i + 1$)
- Vậy $p_{i+1} = p_i + 2Dy - 2Dx(y_{i+1} - y_i)$
- Suy ra cách tính p_{i+1} từ p_i như sau:
  - Nếu $p_i < 0$, $p_{i+1} = p_i + 2Dy$, do $y_{i+1} = y_i$
  - Nếu $p_i >= 0$, $p_{i+1} = p_i + 2(Dy - Dx)$, do $y_{i+1} = y_i + 1$
- Giá trị p_1 tại điểm vẽ đầu tiên (x_1, y_1):

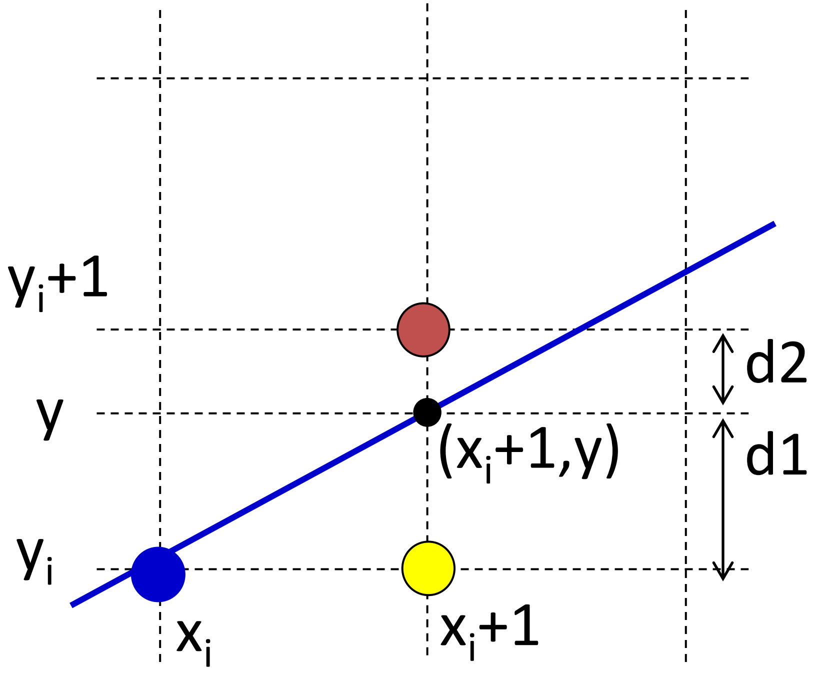
  $p_1 = 2Dyx_1 - 2Dxy_1 + c = 2(Dyx_1 - Dx(y_1 - b)) + 2Dy - Dx$

  $= 2Dy - Dx$. (Do $y_1 = mx_1 + b = (Dy/Dx)x_1 + b \Rightarrow Dyx_1 = Dx(y_1 - b)$)



Vẽ đoạn thẳng - Thuật toán Bresenham

```
void        LineBres(int x1,int y1,int x2,int y2){
            int   Dx,Dy,P,x,y,const1,const2;
            Dx=x2-x1;        Dy=y2-y1;
            const1=2*Dy;  const2=2*(Dy-Dx);
            P=2*Dy-Dx;
            x=x1;                y=y1;
            while (x<=x2){
                    putpixel(x,y,MAGENTA);
                    if (P<0)  P+=const1;
                    else{     P+=const2;
                              y++;
                    }
                    x++;
            }
}
```



Vẽ đoạn thẳng - Thuật toán MidPoint

- Phương trình tổng quát của đường thẳng:

$$Ax+By+C=0$$

Với $A=y_2-y_1=Dy$; $B=-(x_2-x_1)=-Dx$; $C=x_2y_1-x_1y_2$

- Với F(x,y)=Ax+By+C, ta có:

$$F(x,y)\begin{cases} <0 & \text{neáu (x,y) naèm phía treân ñöôøng thaúng} \\ =0 & \text{neáu (x,y) thuoäc veà ñöôøng thaúng} \\ >0 & \text{neáu (x,y) naèm phía döôùi ñöôøng thaúng} \end{cases}$$

Vẽ đoạn thẳng - Thuật toán MidPoint

- Choïn y_{i+1} laø y_i hay y_i+1 baèng caùch so saùnh ñieåm thöïc Q(x_i+1,y) vôùi ñieåm MidPoint M(x_i+1, y_i+1/2) laø trung ñieåm cuûa S(x_i+1,y_i) vaø P(x_i+1, y_i+1).

- Neáu Q naèm döôùi ñieåm M, choïn S.

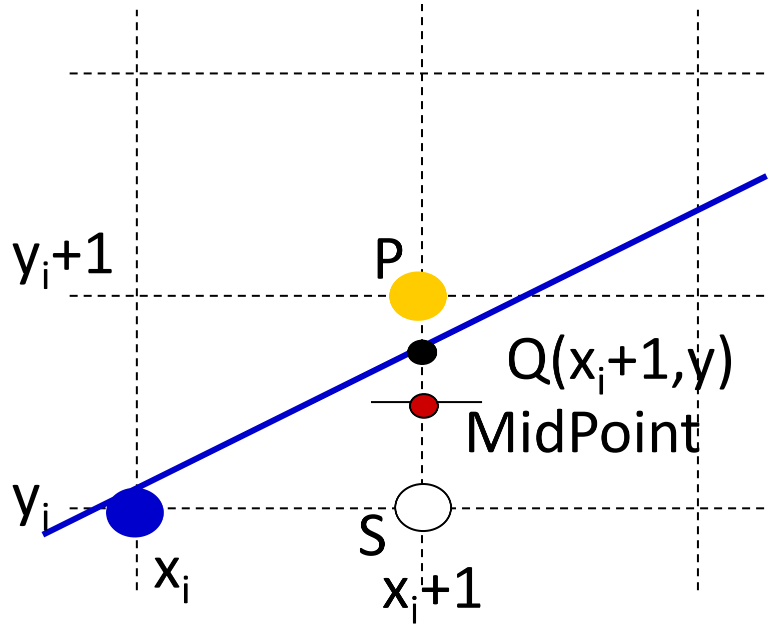
  Ngöôïc laïi, choïn P

- Choïn S hay P döïa vaøo daáu p_i:

  p_i = 2F(MidPoint) = 2F(x_i+1, y_i+1/2)

  Neáu p_i<0 ⇨Choïn S, töùc laø y_{i+1}=y_i

  Neáu p_i ≥0 ⇨Choïn P, töùc laø y_{i+1}=y_i+1



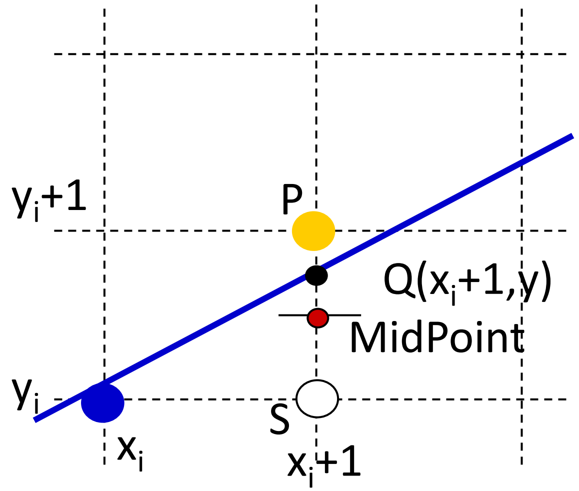
Vẽ đoạn thẳng - Thuật toán MidPoint

- Maët khaùc: $p_{i+1} - p_i = 2F(x_{i+1}+1, y_{i+1}+1/2) - 2F(x_i+1, y_i+1/2)$

$$= 2[A(x_{i+1}+1)+B(y_{i+1}+1/2)+C] - 2[A(x_i+1)+B(y_i+1/2)+C]$$

$$= 2A+2B(y_{i+1}-y_i) = 2Dy-2Dx(y_{i+1}-y_i)$$

$$\Rightarrow p_{i+1} = p_i + 2Dy-2Dx(y_{i+1}-y_i)$$



- Vaäy:
  - Neáu $p_i<0$, $y_{i+1}=y_i$ thì $p_{i+1} = p_i + 2Dy$
  - Neáu $p_i \geq 0$, $y_{i+1}=y_i+1$ thì $p_{i+1} = p_i + 2(Dy-Dx)$
- Ta tính giaù trò p_1 öùng vôùi ñieåm ban ñaàu (x_1,y_1):

$$p_1 = 2F(x_1+1, y_1+1/2) = 2(Ax_1+By_1+C)+2A+B = 2A+B \text{ (do } Ax_1+By_1+C=0)$$

$$=2Dy-Dx$$

Vẽ đường tròn

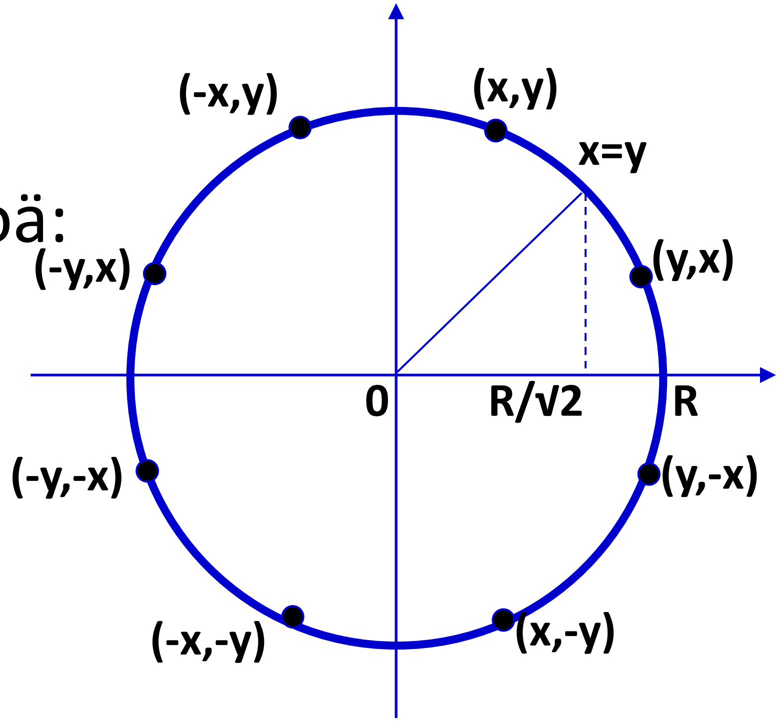
- Phöông trình ñöôøng troøn coù daïng:

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

- Pt ñöôøng troøn coù taâm ôû goác toaï ñoä:

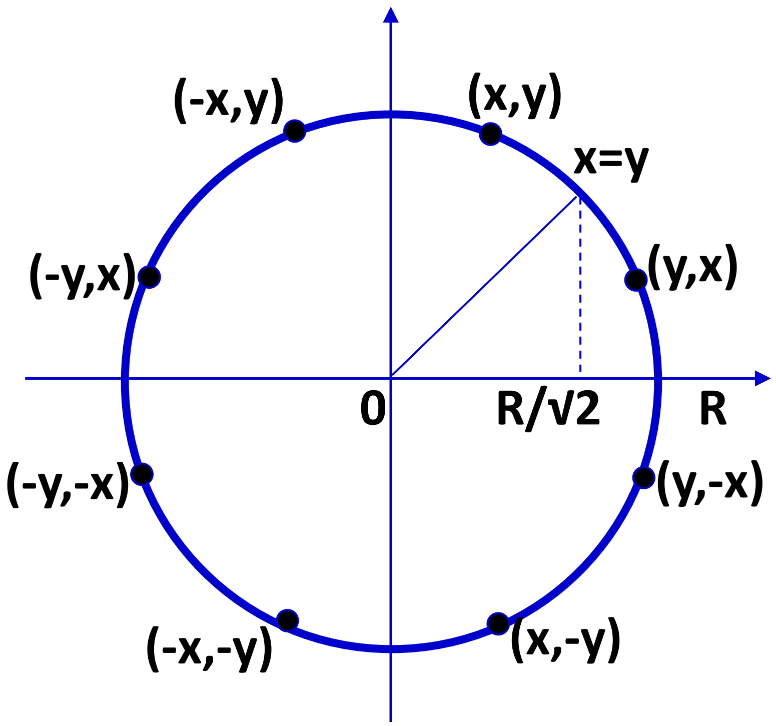
$$x^2 + y^2 = r^2$$

- Do tính ñoái xöùng cuûa ñöôøng troøn neân ta chæ caàn veõ cung 1/4 hoaëc 1/8



Vẽ đường tròn

```
void put8pixel(int xc,int yc, int x, int y){
        putpixel( x+xc, y+yc,color);
        putpixel( y+xc, x+yc,color);
        putpixel( y+xc,-x+yc,color);
        putpixel( x+xc,-y+yc,color);
        putpixel(-x+xc,-y+yc,color);
        putpixel(-y+xc,-x+yc,color);
        putpixel(-y+xc, x+yc,color);
        putpixel(-x+xc, y+yc,color);
}
```



Vẽ đường tròn -Thuật toán Bresenham

- Giaû söû taïi böôùc i ñaõ veõ ñöôïc ñieåm (x_i,y_i)
- Ñieåm caàn veõ keá tieáp (x_{i+1},y_{i+1}) laø:

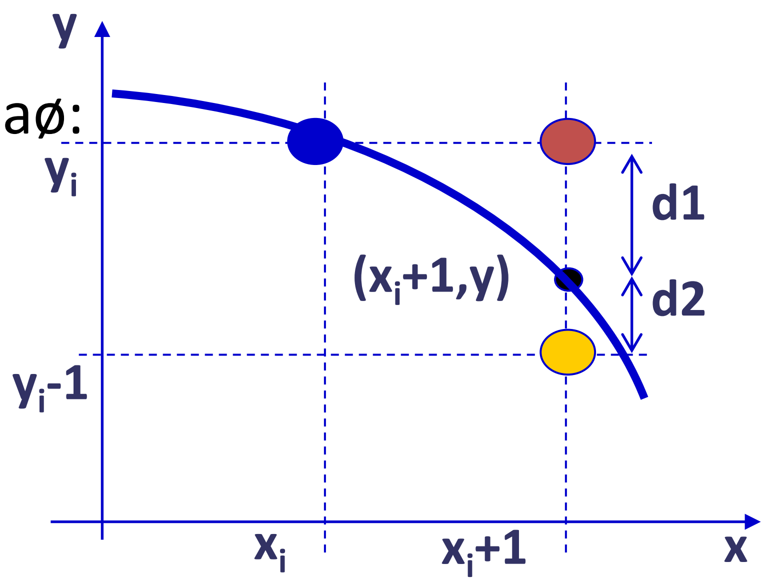
$$(x_i+1,y_i) \text{ hay } (x_i+1,y_i-1).$$

- Giaù trò y thöïc söï thuoäc ñöôøng troøn öùng vôùi x_i laø:

$$y^2 = r^2 - (x_i+1)^2$$

- Goïi $d1 = y_i^2-y^2 \qquad = y_i^2- r^2+(x_i+1)^2$

$$d2 = y^2-(y_i-1)^2 \quad = r^2-(x_i+1)^2-(y_i-1)^2$$



Vẽ đường tròn -Thuật toán Bresenham

- $p_i \quad = d1 - d2$

$$= y_i^2 - r^2 + (x_i+1)^2 - r^2 + (x_i+1)^2 + (y_i-1)^2$$

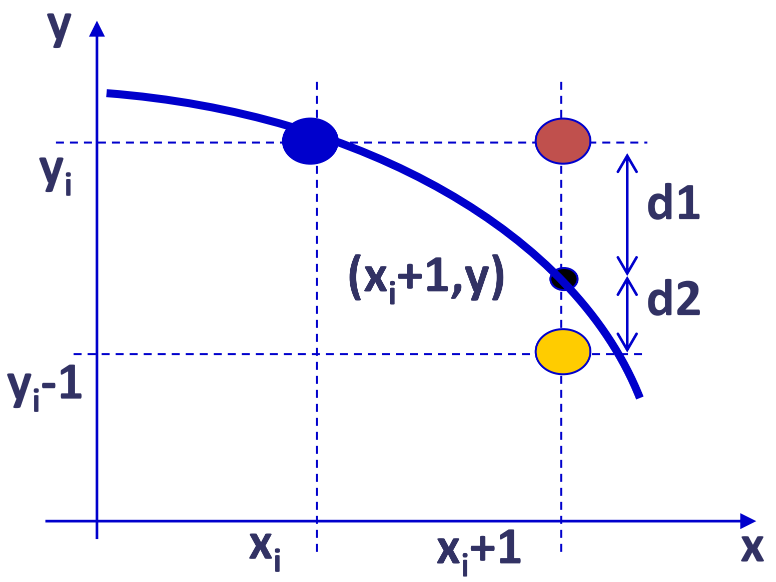
$$= 2(x_i+1)^2 + y_i^2 + (y_i-1)^2 - 2r^2$$

- $p_{i+1} - p_i = 2(x_{i+1}+1)^2 + y_{i+1}^2 + (y_{i+1}-1)^2 - 2r^2$

$$-2(x_i+1)^2 - y_i^2 - (y_i-1)^2 + 2r^2$$

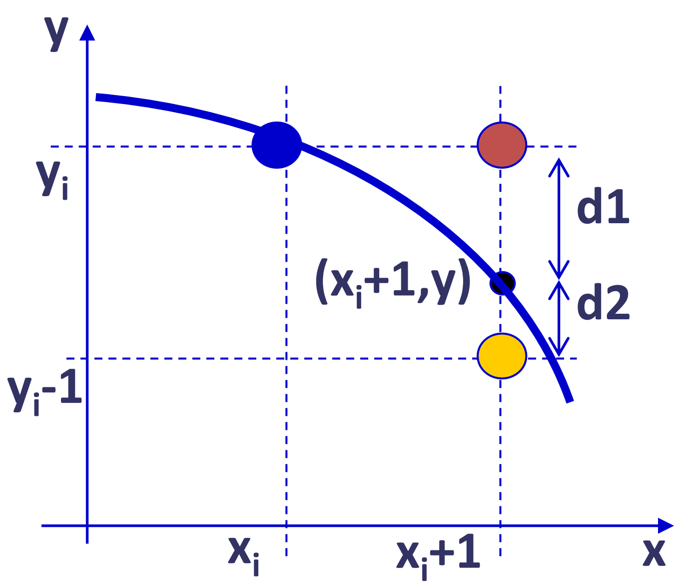
$$= 4x_i + 6 + 2(y_{i+1}^2 - y_i^2) - 2(y_{i+1} - y_i)$$

- ⇨ $p_{i+1} = p_i + 4x_i + 6 + 2(y_{i+1}^2 - y_i^2) - 2(y_{i+1} - y_i)$



Vẽ đường tròn -Thuật toán Bresenham

- $p_{i+1} = p_i + 4x_i + 6 + 2(y_{i+1}^2 - y_i^2) - 2(y_{i+1} - y_i)$



- Vaäy:
  - Neáu $p_i < 0$ thì $y_{i+1} = y_i$

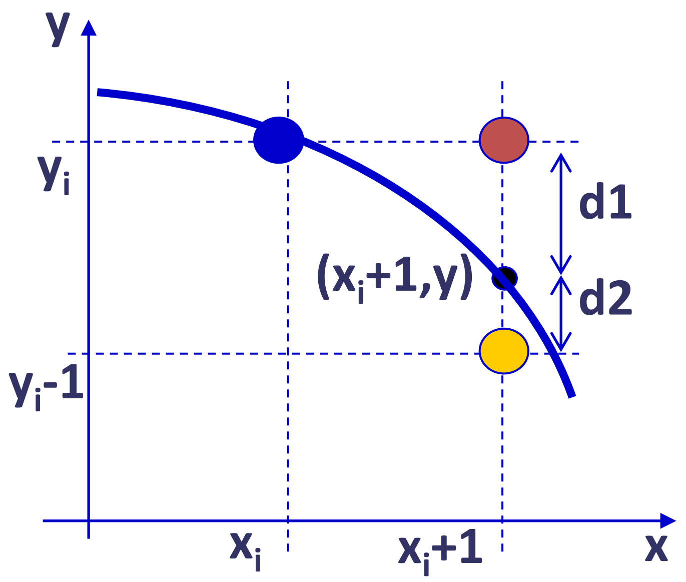
    khi ñoù $p_{i+1} = p_i + 4x_i + 6$
  - Neáu $p_i \geq 0$ thì $y_{i+1} = y_i - 1$

    khi ñoù $p_{i+1} = p_i + 4(x_i - y_i) + 10$

- Giaù trò p_i taïi ñieåm ñaàu tieân $(x_1,y_1)=(0,r)$ laø: $P_1 = 2 + r^2 + (r-1)^2 - 2r^2 = 3 - 2r$

Vẽ đường tròn -Thuật toán Bresenham

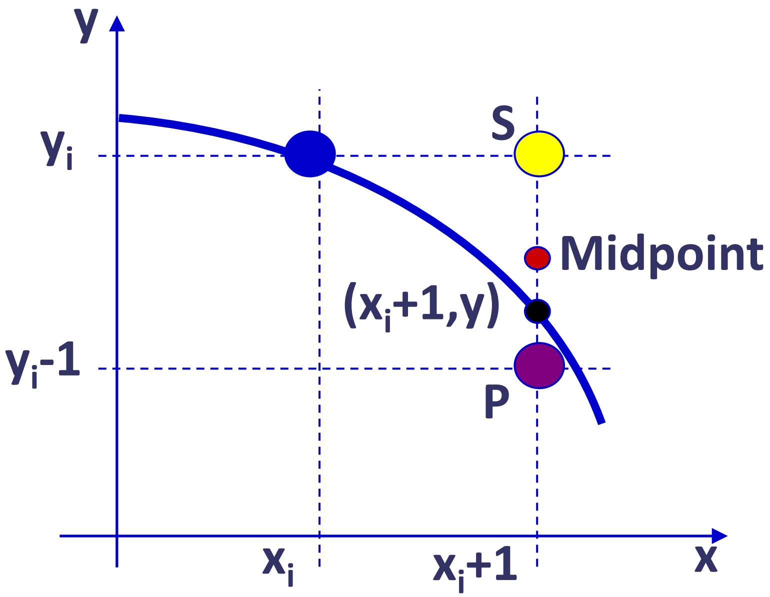
```
void CircleBres(int xc,int yc,int r){
        int x,y,p;
        x=0;   y=r;    p=3-2*r;
        while (x<=y) {
                put8pixel(xc,yc,x,y);
                if (p<0)        p+=4*x+6;
                else{   p+=4*(x-y)+10;
                        y--;
                }
                x++;
        }
}
```



Vẽ đường tròn -Thuật toán Midpoint

- Goïi F(x,y) = $x^2+y^2-r^2$, ta coù:

$$F(x,y)\begin{cases} <0 & \text{neáu (x,y) naèm trong ñöôøng troøn} \\ =0 & \text{neáu (x,y) thuoäc ñöôøng troøn} \\ >0 & \text{neáu (x,y) naèm ngoaøi ñöôøng troøn} \end{cases}$$



- Choïn ñieåm baét ñaàu veõ laø (0,r).
- Giaû söû ñaõ veõ ñöôïc ñieåm (x_i,y_i),

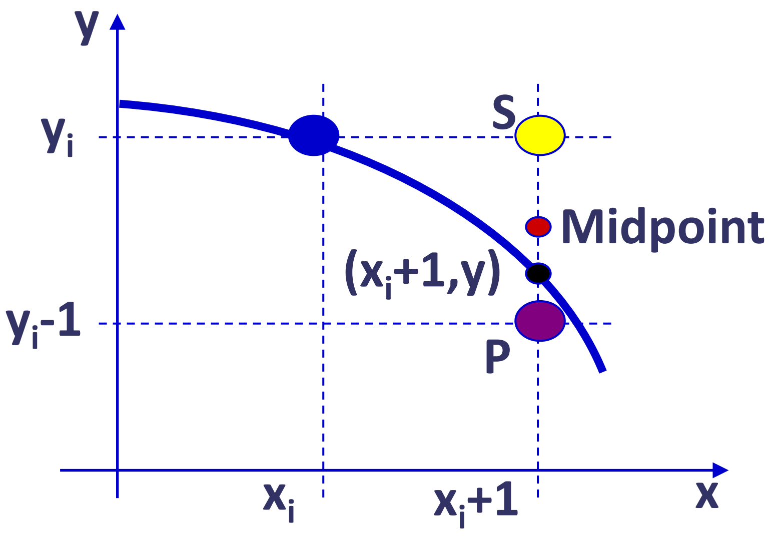
  Ñieåm caàn veõ keá tieáp (x_{i+1},y_{i+1}) laø S hay P.

Vẽ đường tròn -Thuật toán Midpoint

- Vieäc choïn ñieåm S hay P döïa treân daáu cuûa p_i:

$$p_i = F(MidPoint) = F(x_i+1, y_i-1/2)$$

- Neáu $p_i<0$ ⇨choïn S
- Neáu $p_i \geq 0$ ⇨choïn P



Vẽ đường tròn -Thuật toán Midpoint

- Maët khaùc:

$$p_{i+1} - p_i = F(x_{i+1}+1, y_{i+1}-1/2) - F(x_i+1, y_i-1/2)$$

$$= [(x_{i+1}+1)^2+(y_{i+1}-1/2)^2-r^2] - [(x_i+1)^2+(y_i-1/2)^2-r^2]$$

$$= 2x_i + 3 + (y_{i+1}^2 - y_i^2) - (y_{i+1} - y_i)$$

- Vaäy: + Neáu $p_i < 0$ thì $y_{i+1}=y_i$, khi ñoù $p_{i+1} = p_i + 2x_i+3$

  + Neáu $p_i \geq 0$ thì $y_{i+1}=y_i-1$, khi ñoù $p_{i+1} = p_i + 2(x_i-y_i)+5$

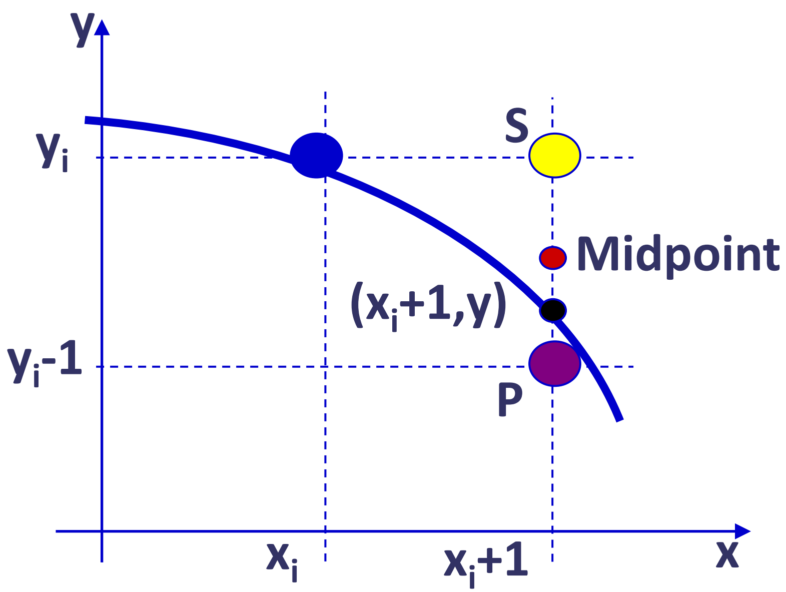
- Giaù trò p_i taïi ñieåm ñaàu tieân $(x_1,y_1)=(0,r)$ laø:

$$p_1 = F(x_1+1, y_1-1/2)= F(1, r-1/2)$$

$$= 5/4 – r.$$

Vẽ đường tròn -Thuật toán Midpoint

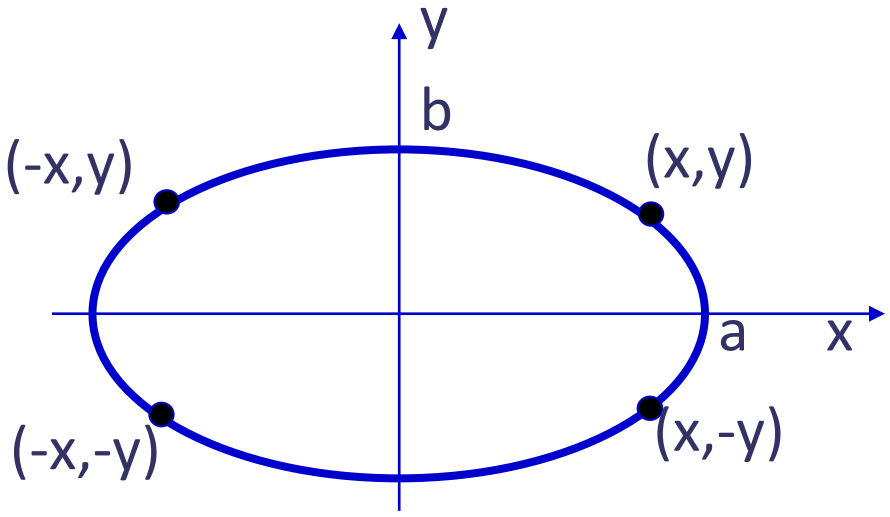
```
void CircleMidpoint(int xc,int yc,int r){
        int x,y,p;
        x=0;   y=r;    p=5/4-r;
        while (x<=y) {
                put8pixel(xc,yc,x,y);
                if (p<0)      p+=2*x+3;
                else{
                        p+=2*(x-y)+5;
                        y--;
                }
                x++;
        }
}
```



Vẽ Elip

- Phương trình của Ellipse có tâm ở gốc tọa độ có dạng:

$$\frac{x^2}{a^2} + \frac{y^2}{b^2} = 1$$



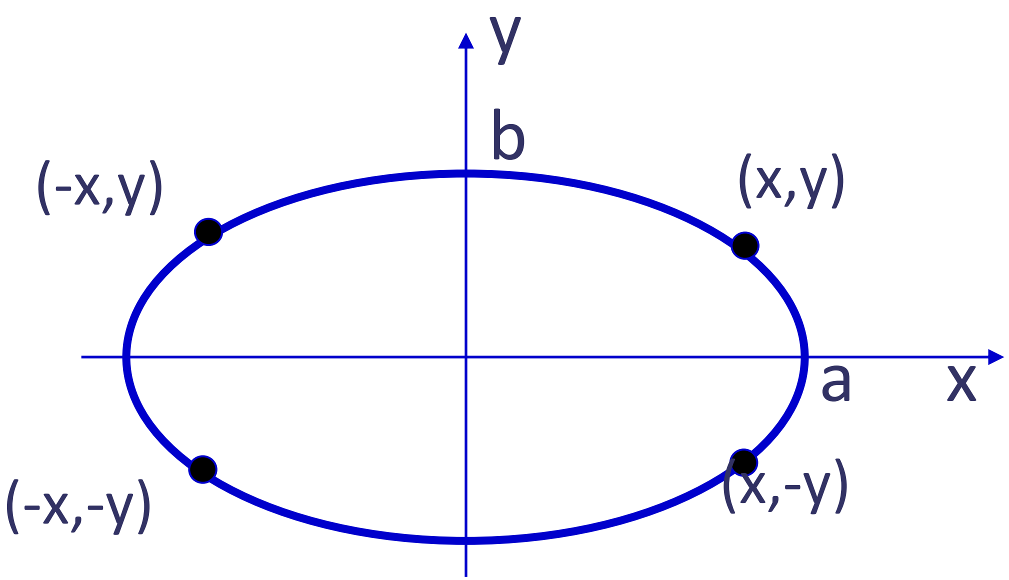
- Ta có thể viết lại:

$$y^2 = -\frac{b^2}{a^2}x^2 + b^2$$

- Do tính ñoái xöùng cuûa ellipse neân ta chæ caàn veõ ¼ ellipse.

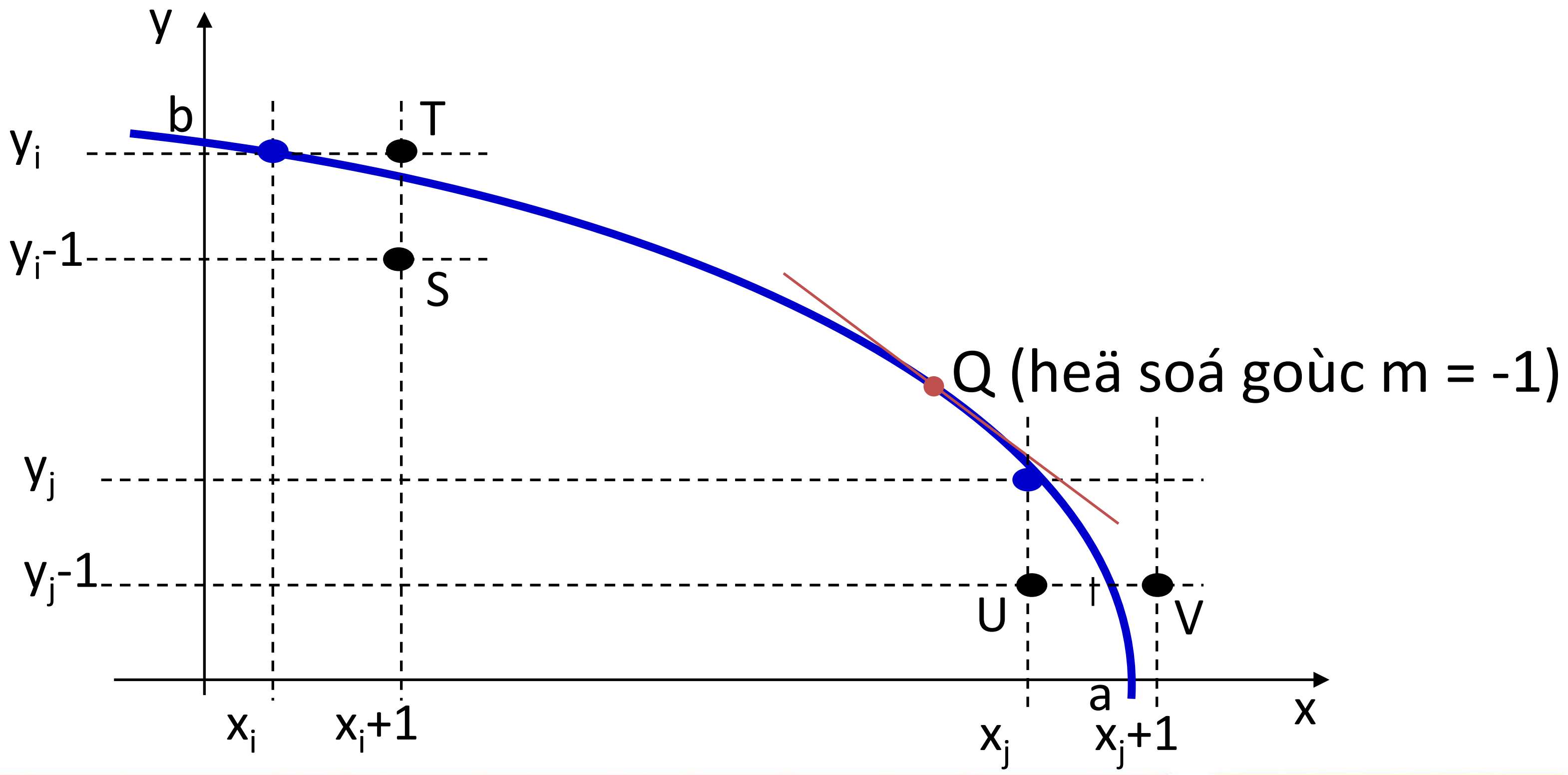
Vẽ Elip

```
void put4pixel(int xc,int yc, int x, int y, int color){
        putpixel(x+xc, y+yc,color);
        putpixel(x+xc,-y+yc,color);
        putpixel(xc-x,  yc-y,color);
        putpixel(xc-x, yc+y,color);
}
```



Vẽ Elip

- Vieäc veõ ellipse ñöôïc chia laøm 2 nhaùnh:
  - Nhaùnh 1: töø ñieåm (0,b) ñeán ñieåm Q (heä soá goùc taïi Q laø –1)
  - Nhaùnh 2: töø ñieåm Q ñeán ñieåm (a,0)



Vẽ Elip

- Ñeå xaùc ñònh heä soá goùc m cuûa ellipse taïi ñieåm (x,y) baát kyø, ta coù:

$$m = \frac{dy}{dx} = -\frac{fx}{fy} = -\frac{2b^2x}{2a^2y}$$

Trong ñoù: fx, fy laø ñaïo haøm rieâng cuûa haøm:

$$f(x,y) = b^2x^2 + a^2y^2 - a^2b^2$$

- Ñieàu kieän chuyeån töø nhaùnh 1 sang nhaùnh 2 laø:

$m<-1$ hay $2b^2x > 2a^2y$

(vì goùc ¼ thöù I cuûa heä toïa ñoä thieát bò, $a,b,x,y \geq 1$)

Vẽ Elip

- Tiếp tuyến tại $Q(x_0,y_0)$ có dạng: $$\frac{xx_0}{a^2}+\frac{yy_0}{b^2}=1$$

- Vì m = $-\frac{x_0b^2}{y_0a^2}=-1$ => $$y_0^2=\frac{b^4}{a^4}x_0^2 \quad \text{(a)}$$

- Mặt khác do Q thuộc ellipse nên: $$\frac{x_0^2}{a^2}+\frac{y_0^2}{b^2}=1 \quad \text{(b)}$$

- Từ (a) và (b) ta suy ra: $x_0=\frac{a^2}{\sqrt{a^2+b^2}}$ và $y_0=\frac{b^2}{\sqrt{a^2+b^2}}$

Vẽ Elip - Thuật toán Bresenham

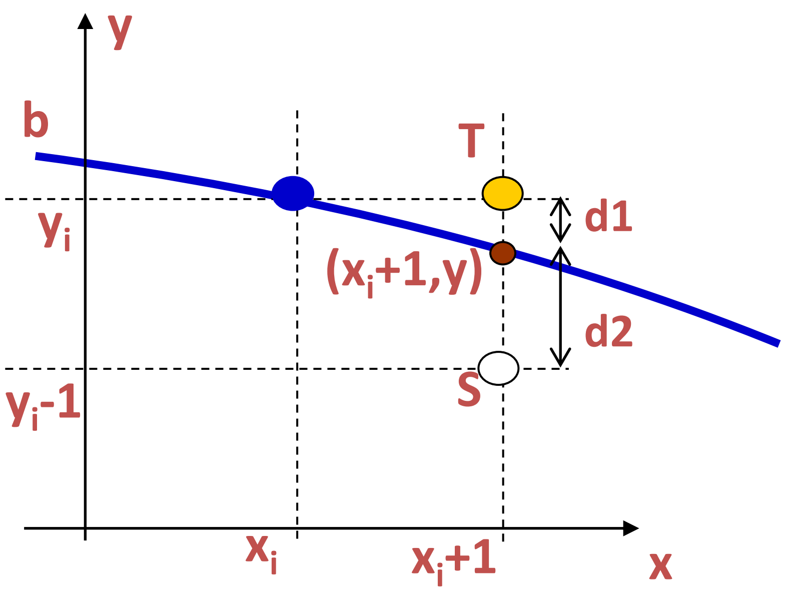
<u>Nhaùnh 1 (töø ñieåm (0,b) ñeán ñieåm Q)</u>

- Giaû söû taïi böôùc i, ñieåm (x_i, y_i) ñöôïc veõ.
- Xeùt taïi böôùc i+1, ta coù: $\begin{cases} x_{i+1} = x_i + 1 \\ y_{i+1} = \{y_i, y_i - 1\} \end{cases}$
- Đặt: $d_1 = y_i^2 - y^2 = y_i^2 + \dfrac{b^2}{a^2}.(x_i+1)^2 - b^2$

$$d_2 = y^2 - (y_i-1)^2 = -\dfrac{b^2}{a^2}.(x_i+1)^2 + b^2 - (y_i-1)^2$$

$$p_i = d_1 - d_2 = 2.[\dfrac{b^2}{a^2}.(x_i+1)^2 - b^2] + 2.(y_i^2 - y_i) + 1$$

$$p_{i+1} = 2.[\dfrac{b^2}{a^2}.(x_{i+1}+1)^2 - b^2] + 2.(y_{i+1}^2 - y_{i+1}) + 1$$



$$p_{i+1} - p_i = 2.\frac{b^2}{a^2}.[(x_{i+1}+1)^2 - (x_i+1)^2] + 2.(y_{i+1}^2 - y_i^2 - y_{i+1} + y_i)$$

$$\Rightarrow \quad p_{i+1} = p_i + 2.\frac{b^2}{a^2}.[(x_{i+1}+1)^2 - (x_i+1)^2] + 2.(y_{i+1}^2 - y_i^2 - y_{i+1} + y_i)$$

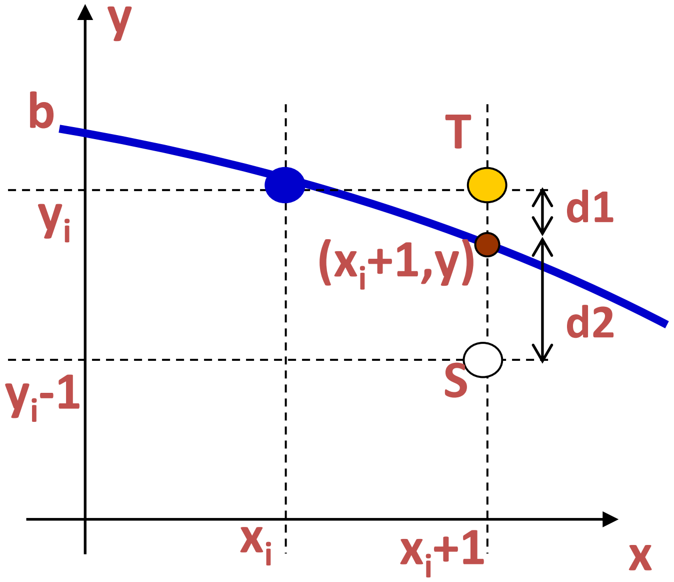
Vậy:

$p_i < 0$: Chọn $y_{i+1} = y_i$

$$\Rightarrow p_{i+1} = p_i + 2.\frac{b^2}{a^2}.(2x_i + 3)$$

$p_i \geq 0$: Chọn $y_{i+1} = y_i - 1$

$$\Rightarrow p_{i+1} = p_i + 2.\frac{b^2}{a^2}.(2x_i + 3) - 4(1-y_i)$$



Điểm đầu tiên (0,b): $p_1 = 2\frac{b^2}{a^2} - 2b + 1$

Nhaùnh 2 (töø ñieåm (a,0) ñeán ñieåm Q:

Tương tự, ta có: $p_{i+1} = p_i + 2.\frac{a^2}{b^2}.[(y_{i+1}+1)^2 - (y_i+1)^2] + 2.(x_{i+1}^2 - x_i^2 - x_{i+1} + x_i)$

Vậy:

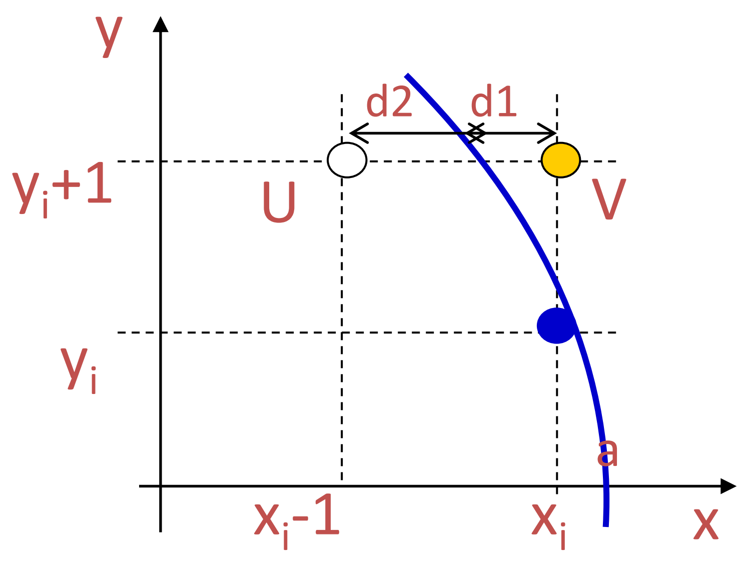
$p_i < 0$: Chọn $x_{i+1} = x_i$

$\Rightarrow p_{i+1} = p_i + 2.\frac{a^2}{b^2}.(2y + 3)$

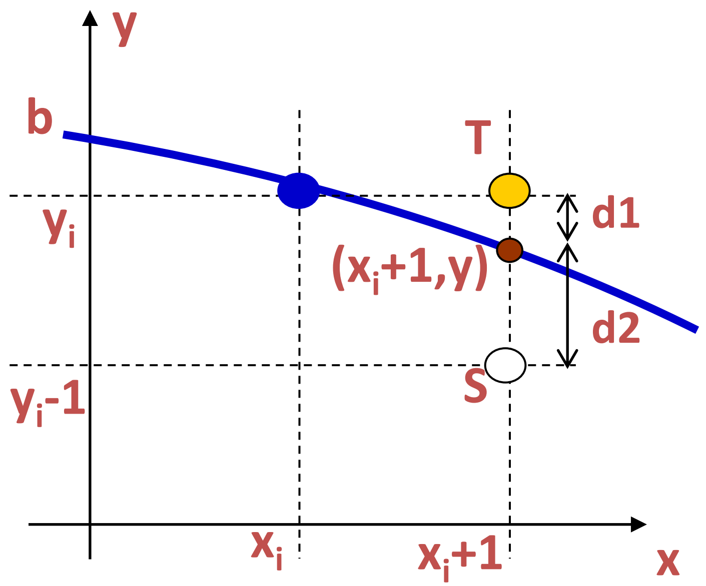
$p_i \geq 0$: Chọn $x_{i+1} = x_i - 1$

$\Rightarrow p_{i+1} = p_i + 2.\frac{a^2}{b^2}.(2y + 3) - 4(1 - x_i)$

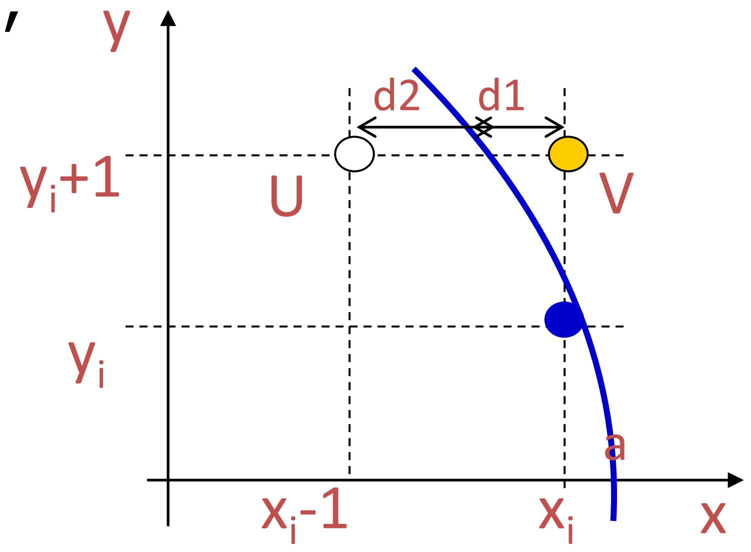
Điểm đầu tiên (a,0), ta có: $p_1 = 2\frac{a^2}{b^2} - 2a + 1$



```
void ElipBres(int xc,int yc, int a, int b){
        double x,y,p,x0, y0,a2,b2;
        a2=a*a;        b2=b*b;
        x=0;   y=b;   p=-2*b+1+2*b2/(a2);
        x0=a2/(sqrt(a2+b2));  y0=b2/(sqrt(a2+b2));
        while (x<=x0){
                put4pixel(xc,yc,YELLOW);
                if (p<0)   p+=2*b2*(2*x+3)/a2;
                else{
                        p+=4*(1-y)+ 2*b2 * (2*x+3)/a2;
                        y--;
                }
                x++;
        }
```



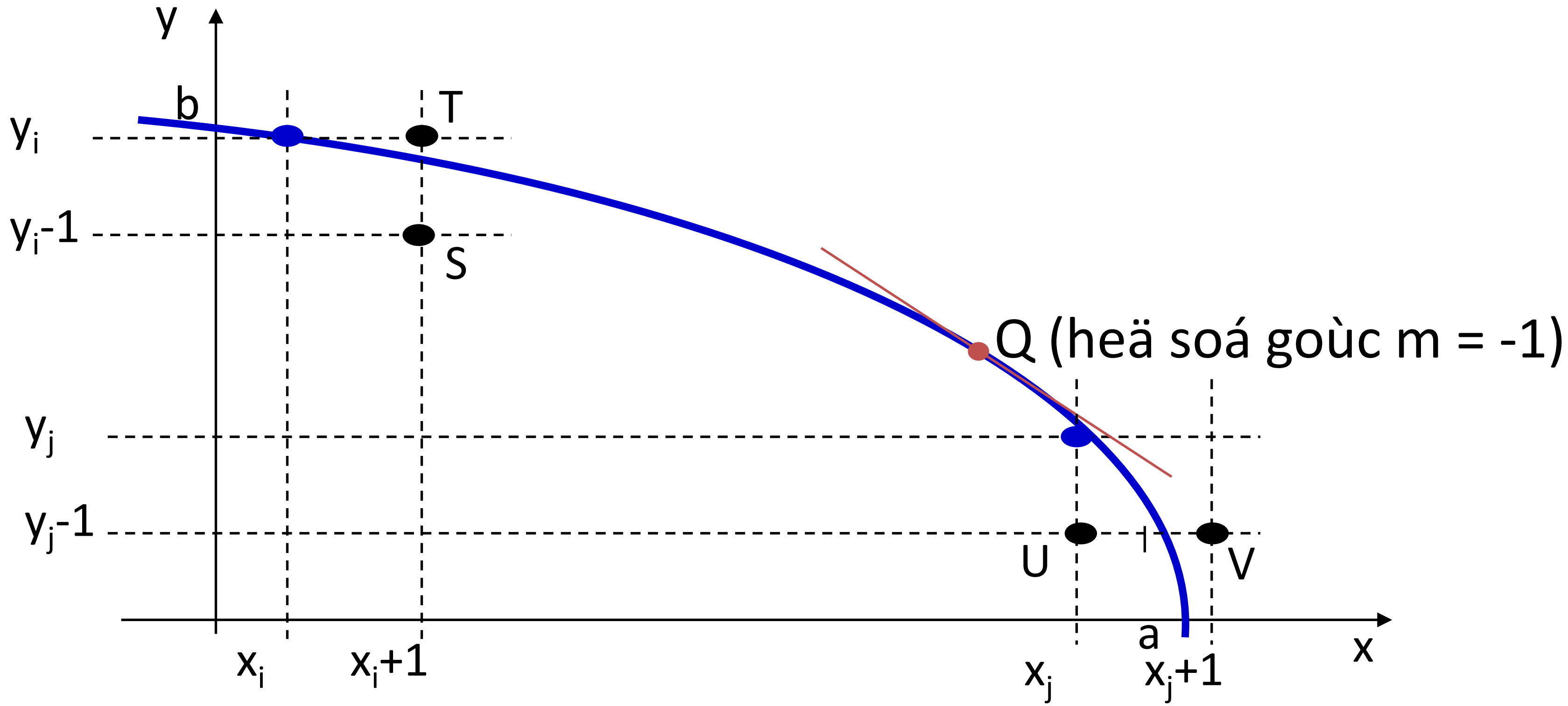
```
x=a;   y=0;   p=2*a2/b2 - 2*a+1;
while (y<=y0){
        put4pixel(xc,yc,x,y,YELLOW);
        if (p<0 )  p+=2*a2*(2*y+3)/b2;
        else{
                p+=4*(1-x) + 2*a2*(2*y+3)/b2;
                x--;
        }
        y++;
}
}
```



Vẽ Elip - Thuật toán Midpoint

Đặt $f(x,y) = b^2x^2 + a^2y^2 - a^2b^2$, ta có

$$f(x,y)\begin{cases} < 0 & \text{neáu } (x,y) \text{ naèm trong ellipse} \\ = 0 & \text{neáu } (x,y) \text{ thuoäc ellipse} \\ > 0 & \text{neáu } (x,y) \text{ naèm ngoaøi ellipse} \end{cases}$$



Vẽ Elip - Thuật toán Midpoint

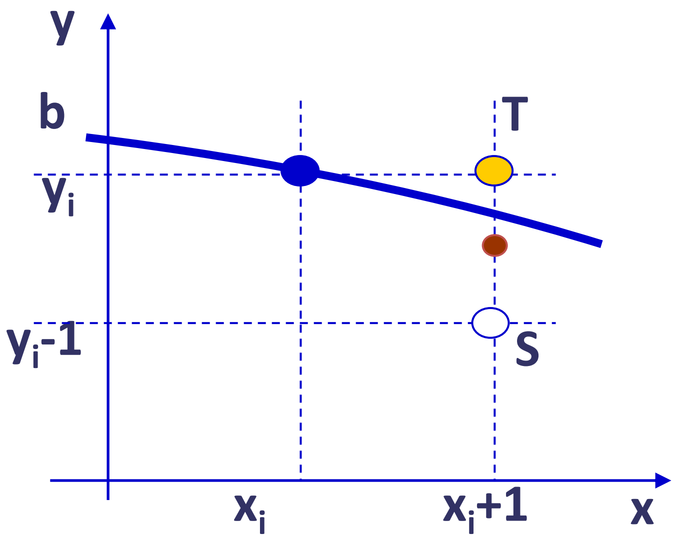
<u>Nhaùnh 1 (töø ñieåm (0,b) ñeán ñieåm Q)</u>

- Giaû söû taïi böôùc i, ñieåm (x_i, y_i) ñöôïc veõ.

- Xeùt taïi böôùc i+1, ta coù: $\begin{cases} x_{i+1} = x_i + 1 \\ y_{i+1} = \{y_i, y_i - 1\} \end{cases}$

- Ta coù: p_i = f(Midpoint) = f(x_i+1, y_i-1/2)

  = $b^2(x_i+1)^2 + a^2(y_i-1/2)^2 - a^2b^2$



Neáu p_i<0 thì vẽ T hay $(x_{i+1},y_{i+1}) = (x_i+1,y_i)$

Neáu p_i≥0 thì veõ S hay $(x_{i+1},y_{i+1}) = (x_i+1,y_i-1)$

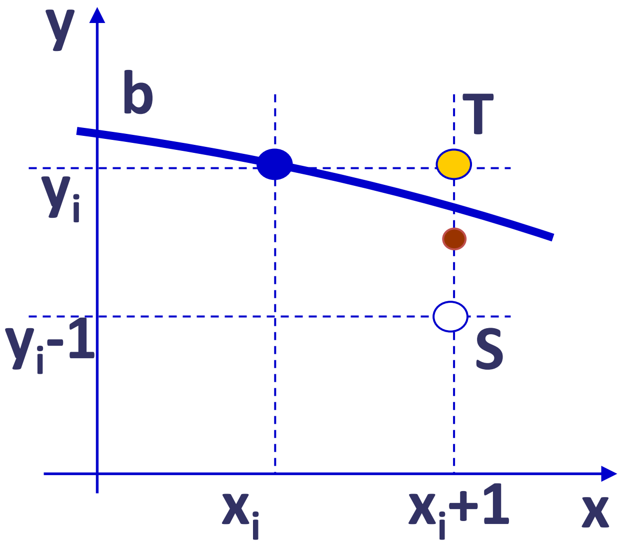
Vẽ Elip - Thuật toán Midpoint

- Tính $p_{i+1}= f(x_{i+1}+1, y_{i+1}-1/2) = b^2(x_{i+1}+1)^2+ a^2(y_{i+1}-1/2)^2- a^2b^2$

Vì $x_{i+1} =x_i+1$ neân:

$p_{i+1}- p_i = b^2 [(x_{i+1}+1)^2-(x_{i+1})^2]+a^2 [(y_{i+1}-1/2)^2-(y_i-1/2)^2]$

$\Rightarrow p_{i+1} = p_i + (2x_i +3)* b^2 + a^2 [(y_{i+1}-1/2)^2-(y_i-1/2)^2]$



- Vaäy: $p_{i+1}= p_i + (2x_i +3)* b^2$ neáu $p_i<0$

$p_{i+1}= p_i + (2x_i +3)* b^2-2 a^2(y_i-1)$ neáu $p_i\geq 0$

- Tính p_1 taïi (0,b): $P_1 = b^2 + a^2(b -1/2)^2- a^2b^2 = b^2- a^2b + a^2/4$

Vẽ vẽ Elip - Thuật toán Midpoint

Nhaùnh 2 (töø ñieåm Q ñeán ñieåm (a,0):

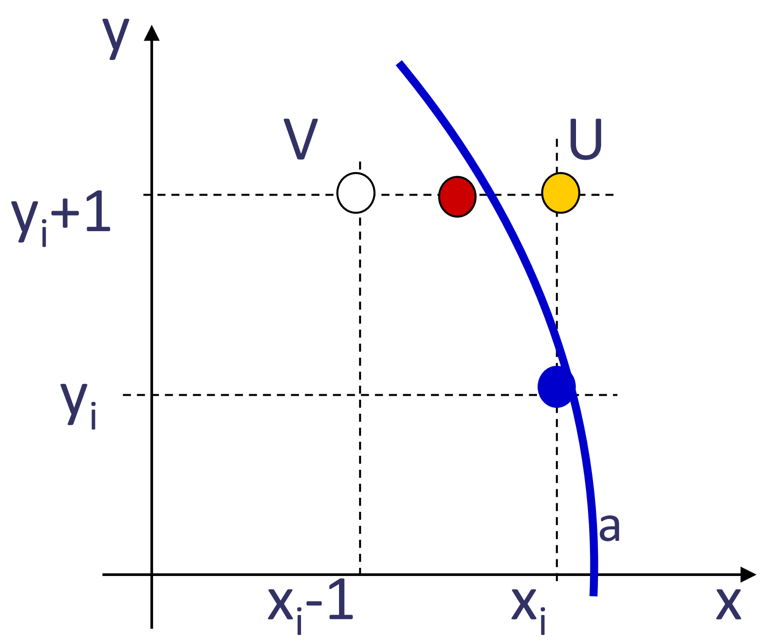
Töông töï, ta coù:

$$q_i = f(\text{Midpoint}) = f(x_i\text{-}1/2,\ y_i\text{+}1)$$

$$= b^2(x_i-1/2)^2 + a^2(y_i+1)^2 - a^2b^2$$

$q_{i+1} = q_i + (2y_i + 3) * a^2$ neáu $q_i<0$

$q_{i+1} = q_i + (2y_i + 3) * a^2 - 2b^2(x_i-1)$ neáu $q_i \geq 0$



Tính q_1 taïi (a,0): $q_1 = a^2 - ab^2 + b^2/4$

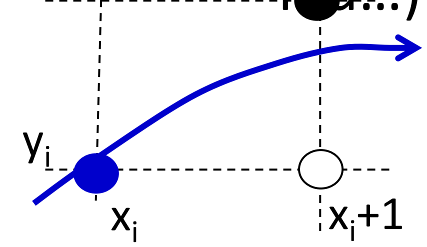
```
void  ElipMidpoint(int xc,int yc,int a,int b,int color){
        int x,y;     float x0,y0,a2,b2,p;
        a2=a*a;       b2=b*b;
        x0=(int)(a2/sqrt(a2+b2));    y0=(int)(b2/sqrt(a2+b2));
        p=b2-a2*b+(1/4)*a2;        x=0;          y=b;
        while (x<=x0){
                put4pixel(xc,yc,x,y,color);
                if (p<0)          p+=(2*x+3)*b2;
                else{      p+=(2*x+3)*b2-2*a2*(y-1);
                           y--;
                }
                x++;
        }
```

```
        x=a;   y=0;   p=a2-a*b2+(1/4)*b2;
        while (y<=y0){
                put4pixel(xc,yc,x,y,color);
                if (p<0)
                        p+=a2*(2*y+3);
                else{
                        p+=(2*y+3)*a2-2*b2*(x-1);
                        x--;
                }
                y++;
        }
}
```

Vẽ đường cong y=f(x)

Aùp duïng thuaät toaùn MidPoint/Bresenham veõ caùc ñöôøng cong theo caùc böôùc sau:

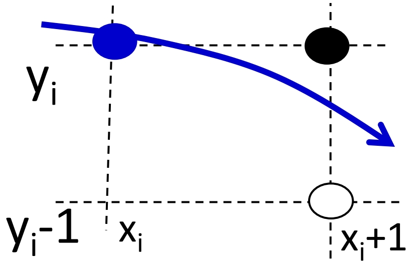
- Böôùc 1: Döïa vaøo daùng ñieäu vaø phöông trình ñöôøng cong (ñoái xuùng, chaün leû...) ñeå ruùt goïn moät phaàn ñöôøng cong caàn veõ.



- Böôùc 2: Tính ñaïo haøm ñeå töø ñoù phaân thaønh caùc vuøng veõ:

  + Neáu 0 ≤ f'(x) ≤ 1 chọn

$$\begin{cases} x_{i+1} = x_i + 1 \\ y_{i+1} \in \{y_i, y_i + 1\} \end{cases}$$

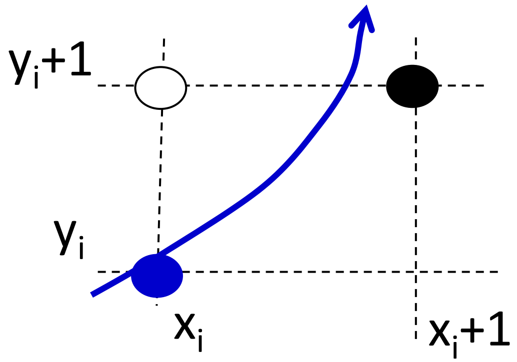


$$\begin{cases} x_{i+1} = x_i + 1 \\ y_{i+1} \in \{y_i, y_i - 1\} \end{cases}$$

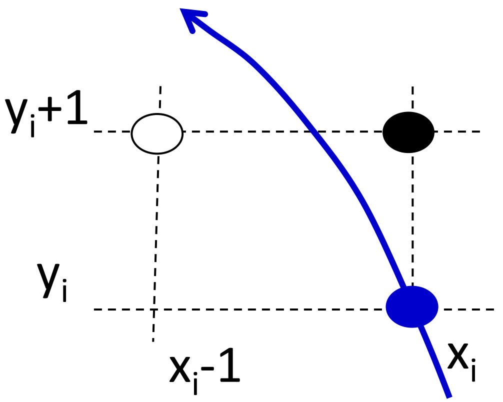
  + Neáu -1 ≤ f'(x) ≤ 0 chọn

Thuật toán vẽ đường cong y=f(x)

+ Neáu $f'(x) > 1$ chọn $\begin{cases} y_{i+1} = y_i + 1 \\ x_{i+1} \in \{x_i, x_i + 1\} \end{cases}$



+ Neáu $f'(x) < -1$ chọn $\begin{cases} y_{i+1} = y_i + 1 \\ x_{i+1} \in \{x_i, x_i - 1\} \end{cases}$



Thuật toán vẽ đường cong y=f(x)

Aùp duïng TT MidPoin/Bresenham veõ caùc ñöôøng cong theo caùc böôùc sau:

- Böôùc 3: Xaùc ñònh coâng thöùc vaø daáu cuûa $\mathbf{p_i}$ cho töøng tröôøng hôïp ñeå tìm ra toaï ñoä cuûa ñieåm caàn veõ.
  (P$_i$ thöôøng laø haøm ñöôïc xaây döïng töø phöông trình ñöôøng cong ñeå cho p_i=0 neáu (x_i,y_i) thuoäc veà ñöôøng cong. Vieäc choïn p_i caàn chuù yù sao cho thao taùc tính p_i haïn cheá caùc pheùp toaùn treân soá thöïc)
- Böôùc 4: Tìm moái lieân quan cuûa p_{i+1} vaø p_i baèng caùch xeùt hieäu p_{i+1}-p_i
- Böôùc 5: Tính p_1 vaø hoaøn chænh thuaät toaùn.

CONTENTs

- Vẽ đối tượng đồ họa cơ sở
- **Xén hình**
- Tô màu
- Vẽ chữ và dựng Font
- Các phép biến đổi 2D

- Thao taùc loaïi boû caùc phaàn hình aûnh naèm ngoaøi moät vuøng cho tröôùc ñöôïc goïi laø xeùn hình (Clipping).

- **Baøi toaùn**: Cho mieàn $D \subset R^n$ vaø $F \subset R^n$. Goïi $F \cap D$ laø hình coù ñöôïc töø hình F baèng caùch caét vaøo trong D, kyù hieäu: $Clip_D(F)$.

⇨ Tìm 1 thuaät toaùn ñeå xaùc ñònh $Clip_D(F)$.

- F là đoạn thẳng và D là hình chữ nhật
- F là đoạn thẳng và D là đường tròn
- F là đa giác và D là hình chữ nhật

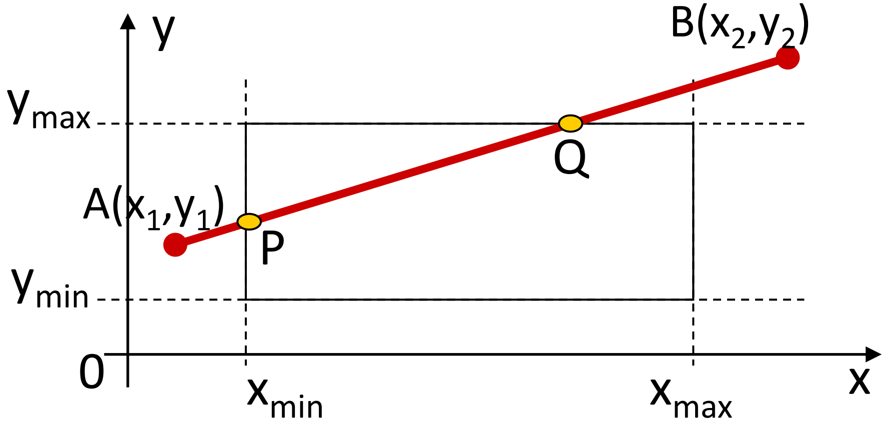
F là đoạn thẳng và D là hình chữ nhật

Giả sử caïnh cuûa hình chöõ nhaät song song vôùi truïc toaï ñoä

Ta coù:

$$D = \left\{ (x,y) \in R^2 \middle| \begin{array}{l} x_{min} \le x \le x_{max} \\ y_{min} \le y \le y_{max} \end{array} \right\}$$

$$F = \left\{ (x,y) \in R^2 \middle| \begin{array}{l} x = x_1 + (x_2 - x_1)t = x_1 + tDx \\ y = y_1 + (y_2 - y_1)t = y_1 + tDy \\ 0 \le t \le 1 \end{array} \right\}$$



Khi ñoù, giao cuûa $F \cap D$ chính laø nghieäm

cuûa baát phöông trình (theo t): $D \cap F = \begin{cases} x_{min} \le x_1 + tDx \le x_{max} \\ y_{min} \le y_1 + tDy \le y_{max} \\ 0 \le t \le 1 \end{cases}$

F là đoạn thẳng và D là hình chữ nhật

Goïi N laø taäp nghieäm cuûa heä: $D \cap F = \begin{cases} x_{min} \le x_1 + tDx \le x_{max} \\ y_{min} \le y_1 + tDy \le y_{max} \\ 0 \le t \le 1 \end{cases}$

Khi ñoù xaûy ra caùc tröôøng hôïp:

-Neáu N=∅ ⇨ Baát phöông trình voâ nghieäm

⇨ $Clip_D(F)=\varnothing$

-Neáu N≠∅ ⇨ $N=[t_1, t_2]$, vôùi $t_1 \le t_2$.

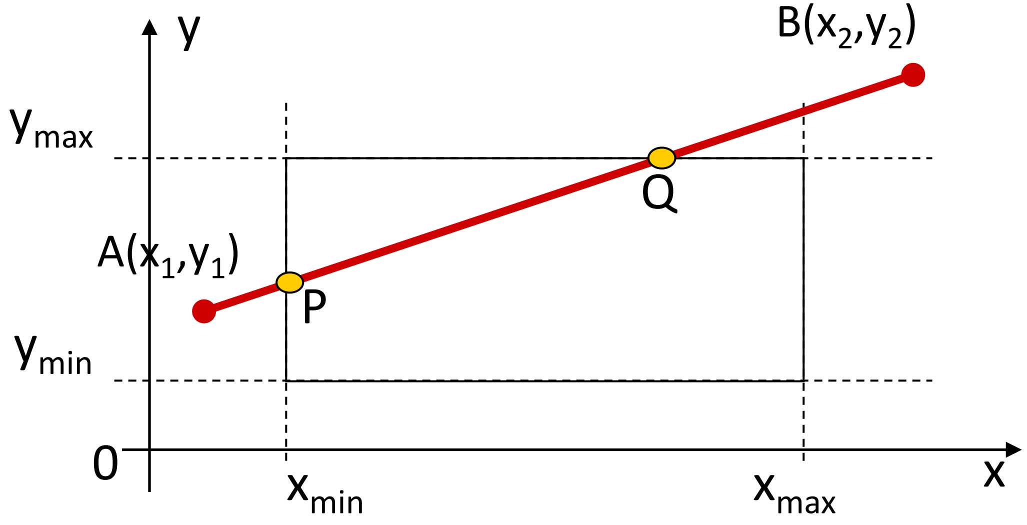
F là đoạn thẳng và D là hình chữ nhật

Nếu $N \neq \varnothing$, goïi P vaøø Q laø 2 giao ñieåm coù toaï ñoä nhö sau:

$$\begin{cases} P_x = x_1 + (x_2 - x_1)t_1 \\ P_y = y_1 + (y_2 - y_1)t_1 \end{cases} \quad \text{vaø} \quad \begin{cases} Q_x = x_1 + (x_2 - x_1)t_2 \\ Q_y = y_1 + (y_2 - y_1)t_2 \end{cases}$$

Vaäy $Clip_D(F)=PQ$

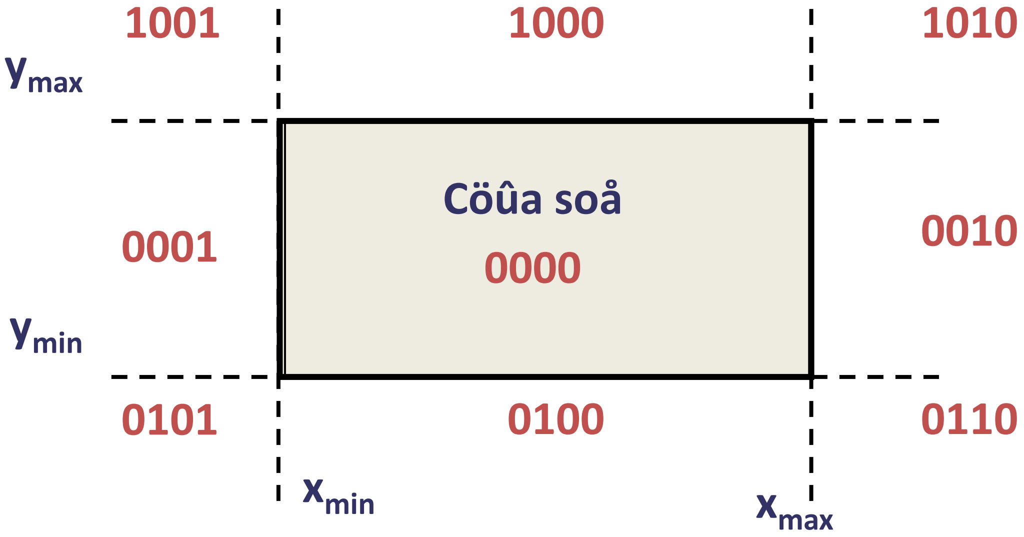
⇨ tính toaùn nhieàu treân soá thöïc

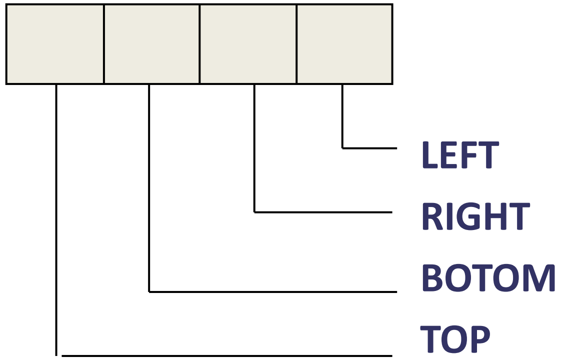


Thuật toán Cohen-Sutherland

Chia maët phaúng thaønh 9 vuøng: cöûa soå vaø 8 vuøng xung quanh noù.

Moãi vuøng ñöôïc gaùn bôûi moät maõ nhò phaân 4 bít.





Thuật toán Cohen-Sutherland

Cho ñieåm P(x,y), gaùn maõ cho ñieåm P:

$$P_{left} = \begin{cases} 1, \text{ neáu } P_x < x_{min} \\ 0, \text{ ngöôïc laïi} \end{cases} \qquad P_{right} = \begin{cases} 1, \text{ neáu } P_x > x_{max} \\ 0, \text{ ngöôïc laïi} \end{cases}$$

$$P_{bottom} = \begin{cases} 1, \text{ neáu } P_y < y_{min} \\ 0, \text{ ngöôïc laïi} \end{cases} \qquad P_{top} = \begin{cases} 1, \text{ neáu } P_y > y_{max} \\ 0, \text{ ngöôïc laïi} \end{cases}$$

Thuật toán Cohen-Sutherland

Hàm xác định mã của một điểm

```
int     ma(point M){
        int m=0;
        if (M.x<min.x)    m|= 1;
        if (M.x>max.x)    m|= 2;
        if (M.y<min.y)    m|= 4;
         if (M.y>max.y)   m|= 8;
        return m;
}
```

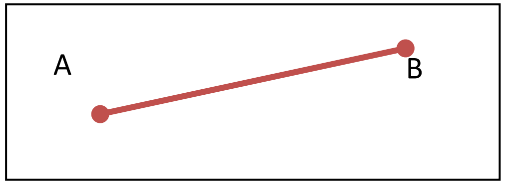
Thuật toán Cohen-Sutherland

Xeùt ñoaïn thaúng AB, ta coù caùc tröôøng hôïp sau:

1.Neáu (Ma(A)=0000) vaø (Ma(B)=0000)

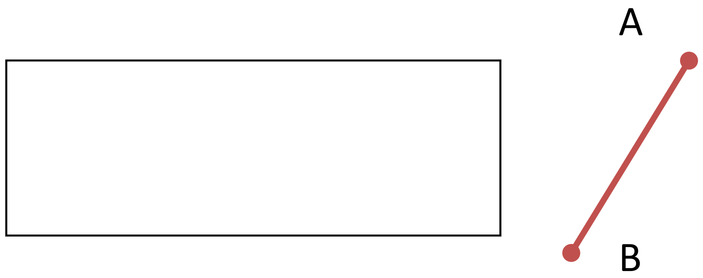
{hay (Ma(A) or Ma(B))=0000}

⇨ $Clip_D(F)=AB$



2.Neáu (Ma(A) and Ma(B))≠0000

⇨ $Clip_D(F)=\varnothing$



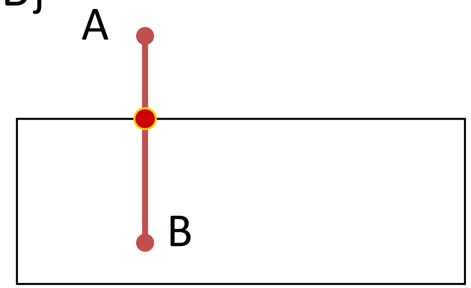
Thuật toán Cohen-Sutherland

3.Neáu ((Ma(A) and Ma(B))=0000) vaø (Ma(A)≠0000 hoaëc Ma(B)≠0000)

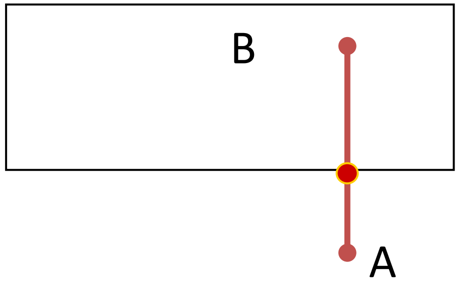
Giaû söû Ma(A)≠0000 {neáu ma(A)=0 ta ñoåi vai troø A vaø B}

-Neáu $A_x=B_x$ (AB thaúng ñöùng)

+ Neáu $A_y>y_{max}$ (A ôû treân) $A_y=y_{max}$,

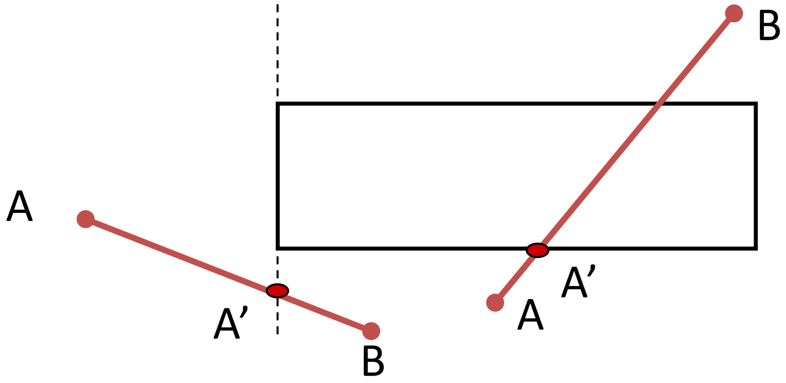


ngöôïc laïi (A ôû döôùi) $A_y=y_{min}$



⇨ $Clip_D(F)=AB$

Thuật toán Cohen-Sutherland



-Ngöôïc laïi (tröôøng hôïp $A_x \neq B_x$):

+Tính hsoá goùc m=(By-Ay)/(Bx-Ax)

{ñeå tính giao cuûa AB vôùi hcn}

Vì A naèm ngoaøi hình chöõ nhaät neân:

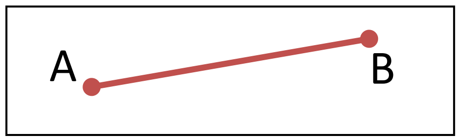
+Neáu $A_x<x_{min}$, Thay A bôûi ñieåm giao cuûa AB vôùi caïnh traùi (noái daøi) cuûa HCN.

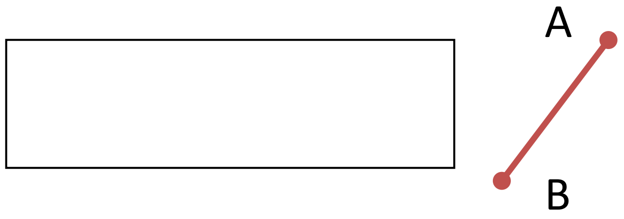
+Neáu $A_x>x_{max}$, Thay A bôûi ñieåm giao cuûa AB vôùi caïnh phaûi (noái daøi) cuûa HCN.

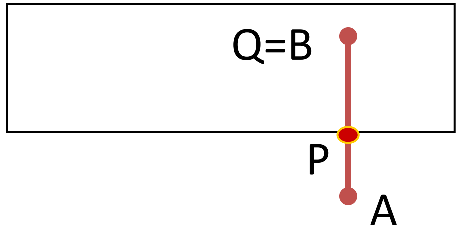
+Neáu $A_y<y_{min}$, Thay A bôûi ñieåm giao cuûa AB vôùi caïnh döôùi (noái daøi) cuûa HCN.

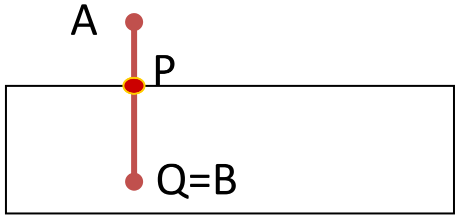
+Neáu $A_y>y_{max}$, Thay A bôûi ñieåm giao cuûa AB vôùi caïnh treân (noái daøi) cuûa HCN

```
void    CohenClip (Point  A, Point B, Point  wmin, Point  wmax) {
    int           escape, draw;  double m;
    escape=0;   draw =1;
    while  (escape==0) {
      if((ma(A) | ma(B))==0)              escape=1
      elseif  ((ma(A) & ma(B)) !=0)     {escape=1; draw=0}
      else {
              if (m(A)==0)     swap(&A,&B);
              if(A.x==B.x) {
                  if (A.y>wmax.y)  A.y=wmax.y;
                   else        A.y=wmin.y;
      }
      else{
```

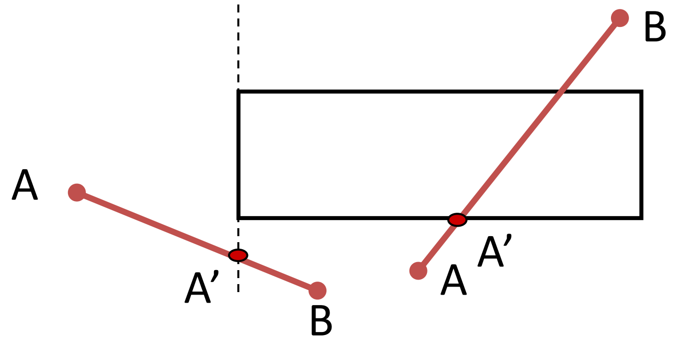






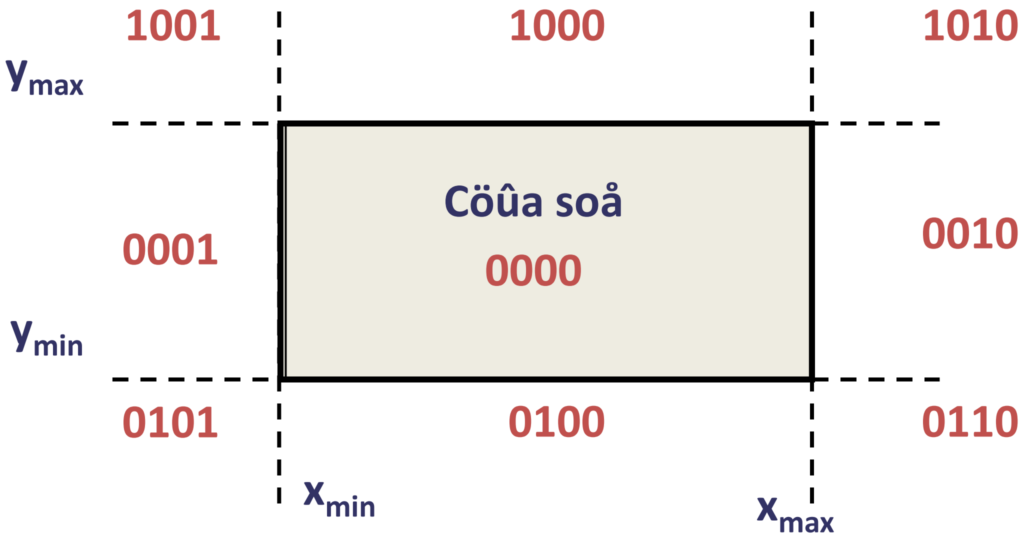


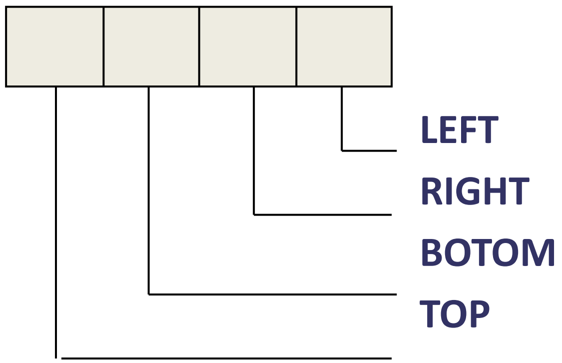
```
                m=(double)(B.y-A.y)/(B.x-A.x);
                if      (A.x<wmin.x){   A.y=A.y+ (wmin.x-A.x)*m;    A.x=wmin.x; }
                elseif  (A.x>wmax.x){   A.y=A.y+ (wmax.x-A.x)*m;    A.x=wmax.x; }
                elseif  (A.y<wmin.y) {  A.x=A.x+ (wmin.y-A.y)/m;        A.y=wmin.y;
          }
                elseif (A.y>wmax.y){   A.x=A.x+ (wmax.y-A.y)/m;    A.y=wmax.y; }
             }
         }
     }//end while
      if   (draw)  drawLine(A,B);
}
```



Thuật toán chia nhị phân

Chia maët phaúng thaønh 9 vuøng, moãi vuøng ñöôïc gaùn bôûi moät maõ nhò phaân 4 bít.

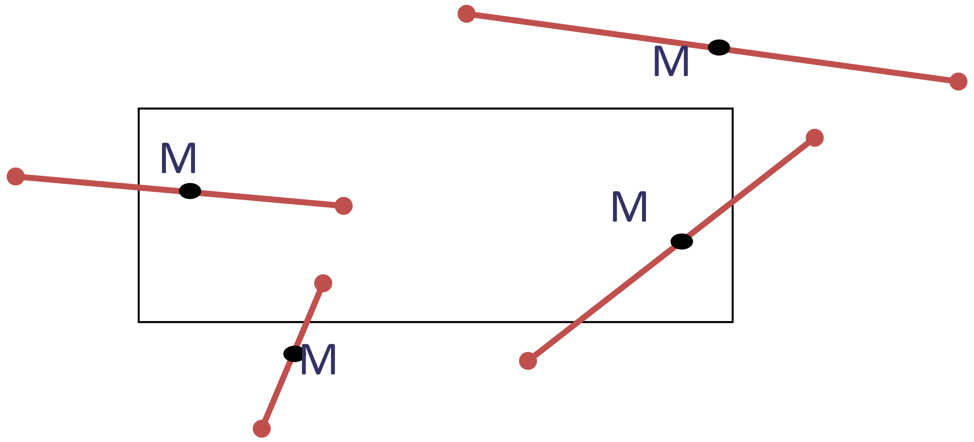




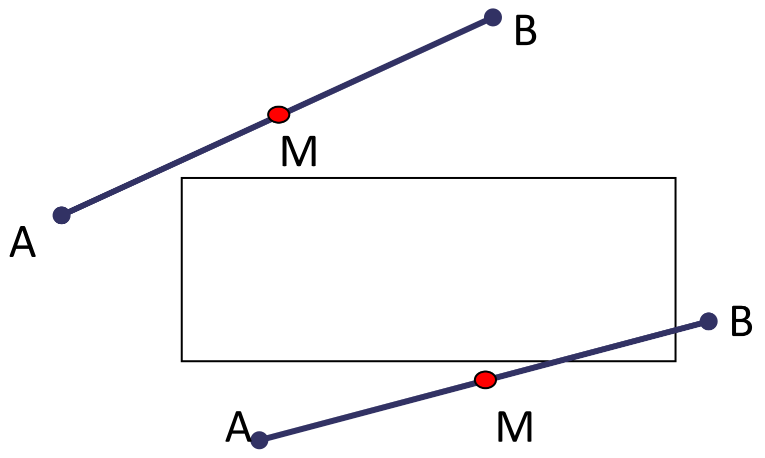
Thuật toán chia nhị phân

Tư tưởng của thuật toán

- Kiểm tra mã của trung điểm đoạn thẳng để loại dần các đoạn con không chứa giao điểm.
- Cuối cùng, trung điểm hội tụ về giao điểm của đoạn thẳng với hình chữ nhật.
- Kết quả, thu được đoạn con nằm trong hình chữ nhật (nếu có)



Thuật toán chia nhị phân



☞**Mệnh đề**:

Cho M trung điểm của đoạn AB,

Mã(A) ≠ 0000,

Mã(B) ≠ 0000,

Mã(M) ≠ 0000

ta có: [Mã(A) AND Mã(M)] ≠ 0000 hoặc [Mã(M) AND Mã(B)] ≠ 0000.

☞ **Ý nghĩa hình học của mệnh đề**:

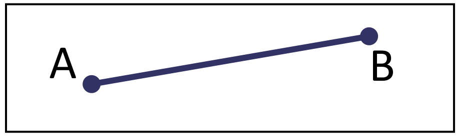
Nếu cả ba điểm A, B, M đều ở ngoài hình chữ nhật thì có ít nhất một đoạn AM (hoặc BM) hoàn toàn nằm ngoài hình chữ nhật.

Thuật toán chia nhị phân

Thuật toán:

1.Nếu (Mã(A) = 0000) và (Mã(B) = 0000)

⇨ $Clip_D(F) = AB$

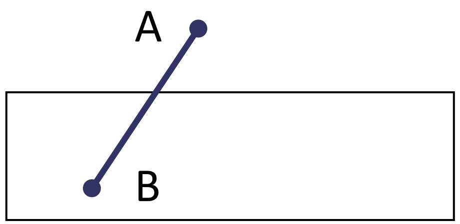


2.Nếu (Mã(A) AND Mã(B)) ≠ 0000

⇨ $Clip_D(F) = \varnothing$

3.Nếu (Mã(A) ≠ 0000) và (Mã(B) = 0000) thì:

Đổi vai trò của A, B và áp dụng 4



Thuật toán chia nhị phân

Thuật toán:

4.Nếu (Mã(A) = 0000) và (Mã(B) $\neq$ 0000):

P:=A; Q:=B;

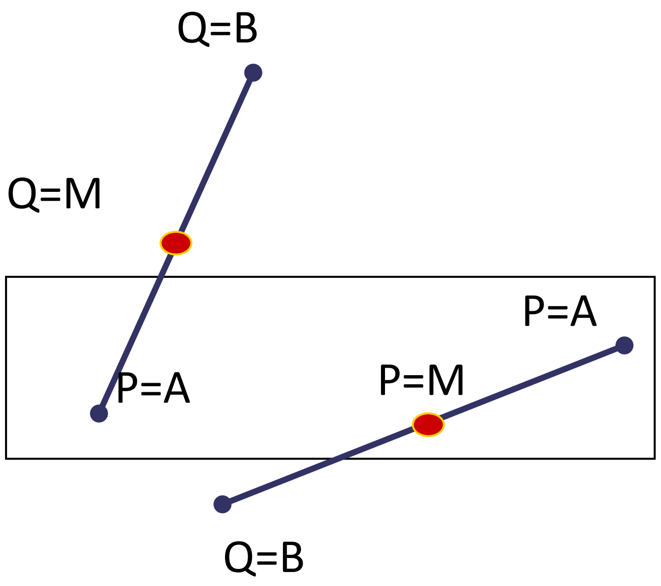
Khi $|x_P - x_Q| + |y_P - y_Q| \geq 2$:

Lấy trung điểm M của PQ

Nếu Mã(M) $\neq$ 0000, Q:= M.

Ngược lại, P:= M.

⇨ $Clip_D(F) = AP$



Thuật toán chia nhị phân

Thuật toán:

5.Nếu (Mã(A) $\neq$ 0000 $\neq$ Mã(B) và [Mã(A) AND Mã(B)]= 0000:

P:=A; Q:=B;

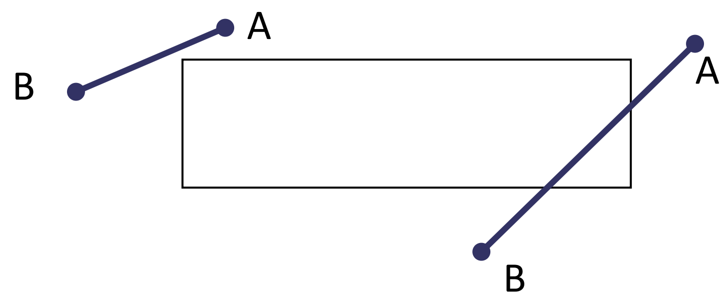
Lấy M là trung điểm PQ;

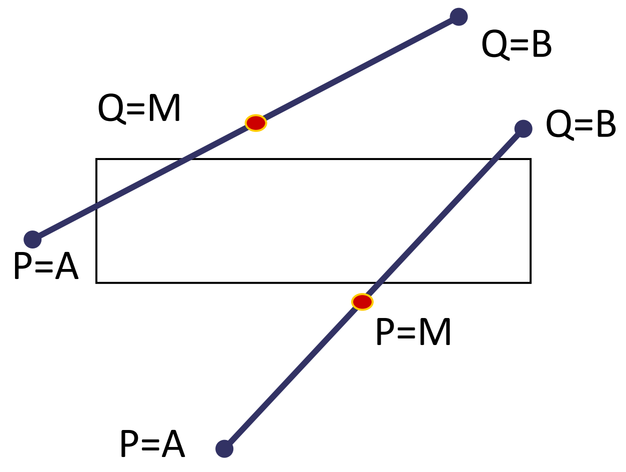
Khi (Mã(M) $\neq$ 0000) và $(|x_P - x_Q| + |y_P - y_Q| \geq 2)$:

Nếu (Mã(M) AND Mã(Q)) $\neq$ 0000, Q:=M.

Ngược lại, P:=M.

Lấy M là trung điểm PQ.





Thuật toán chia nhị phân

Thuật toán:

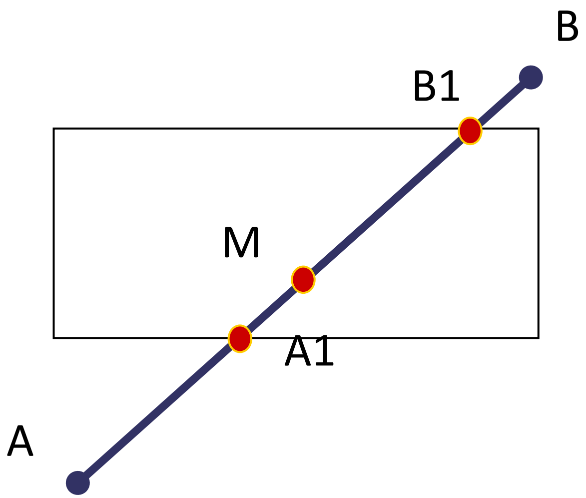
Nếu Mã(M) $\neq$ 0000, $Clip_D(F) = \varnothing$

Ngược lại Ma(M)==0), áp dụng 4 ta có:

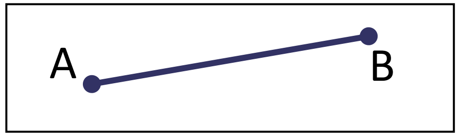
$$Clip_D(MP) = MA_1$$

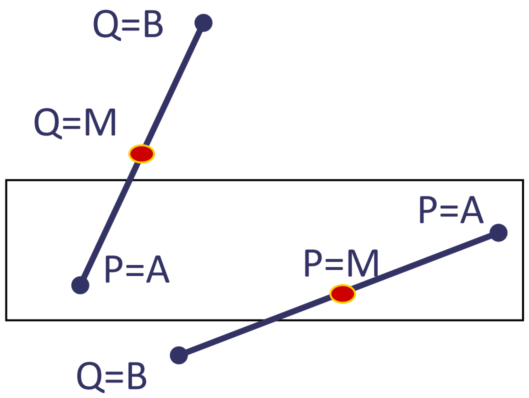
$$Clip_D(MQ) = MB_1$$

⇨ $Clip_D(F) = A_1B_1$

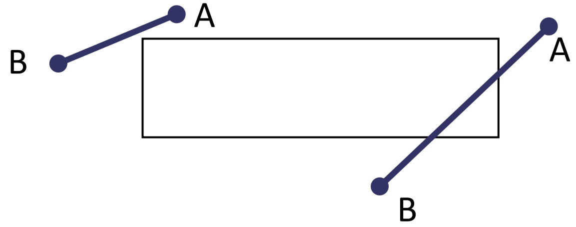


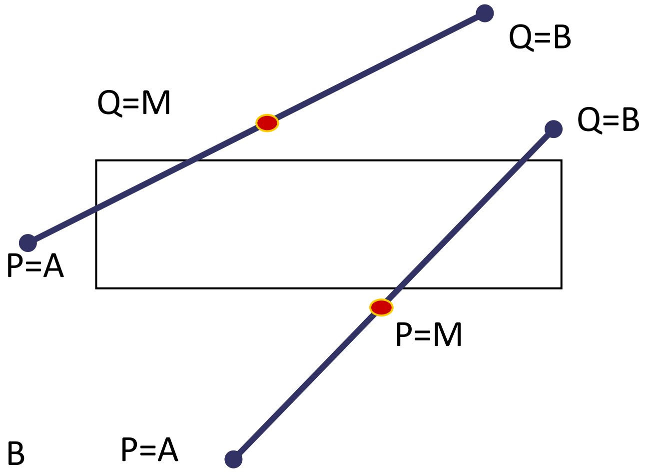
```
void BinaryClip(Point A,Point B){
        Point P,Q,M;
        if ((Ma(A) | Ma(B))==0) drawLine(A,B);
        if ((Ma(A) & Ma(B)) != 0) return;
        if ((Ma(A) != 0) && (Ma(B) == 0)) swap(&A,&B);
        if ((Ma(A) == 0) && (Ma(B) != 0)){
                P=A; Q=B;
                while ((abs(P.x-Q.x)+abs(P.y-Q.y)) > 2){
                        M.x=(P.x+Q.x)/2;        M.y=(P.y+Q.y)/2;
                        if (Ma(M)==0)           P=M;
                        else                    Q=M;
                }
                drawLine(A,P);
}
```

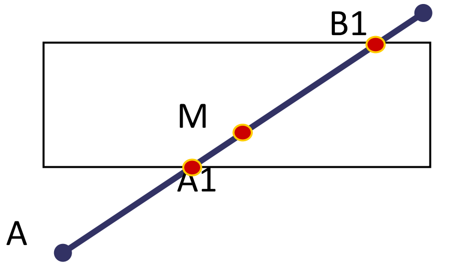




```
if (((Ma(A) != 0) && (Ma(B) != 0)) && ((Ma(A) & Ma(B))==0)){
        P=A; Q=B;
        M.x=(P.x+Q.x)/2;   M.y=(P.y+Q.y)/2;
        while ((Ma(M)!=0) && ((abs(P.x-Q.x)+abs(P.y-Q.y)) > 2)){
                if ((Ma(P) & Ma(M))!=0)   P=M;
                else                      Q=M;
                M.x=(P.x+Q.x)/2;  M.y=(P.y+Q.y)/2;
        }
        if (Ma(M)==0){
                BinaryClip (P,M);
                BinaryClip (M,Q);
        }
}
```





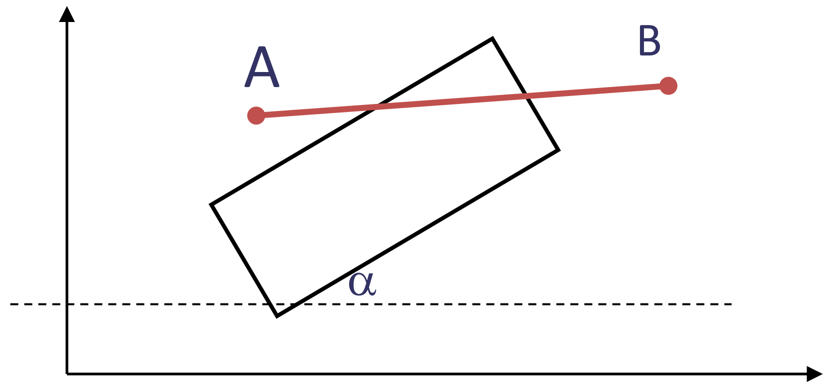


F là đoạn thẳng và D là hình chữ nhật

Caïnh cuûa hình chöõ nhaät taïo vôùi truïc hoaønh moät goùc α

1, Goïi R laø ma traän cuûa pheùp quay ñoåi truïc,

2,Ta tính: $$(X_{max}, Y_{max}) = (x_{max}, y_{max}).R$$
$$(X_{min}, Y_{min}) = (x_{min}, y_{min}).R$$



⇨ A_1=A.R; B_1=B.R

Ñaët F = ñoaïn A_1B_1

D = hình chöõ nhaät ñöôïc taïo bôûi 2 ñieåm (X_{min}, Y_{min}), (X_{max}, Y_{max})

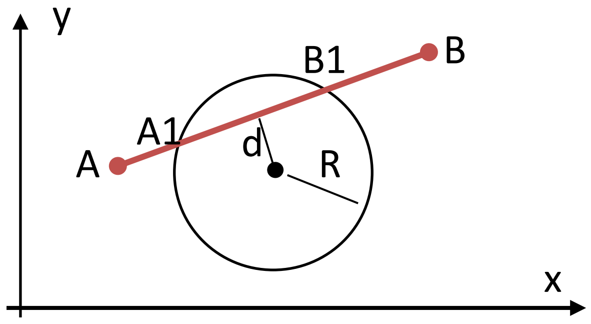
3, Xaùc ñònh $Clip_D(F)$ baèng moät trong caùc thuaät toaùn treân.

4, Neáu $Clip_D(F)=\emptyset$ thì keát quaû cuûa pheùp xeùn hình laø $\emptyset$

Neáu $Clip_D(F)= A_2B_2$, keát quaû laø A_3B_3 vôùi $A_3=A_2.R^{-1}$, $B_3=B_2.R^{-1}$

F là đoạn thẳng và D là đường tròn

Giaûi quyeát baèng caùch xeùt vò trí tuông ñoái giöõa ñöờng thaúng vaø ñöôøng troøn.



1. Tính khoaûng caùch d töø O ñeán AB

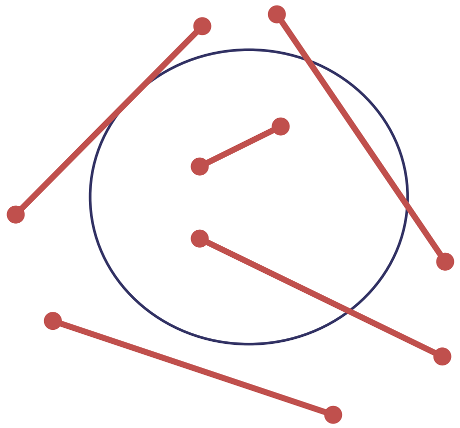
2. Neáu d>R, ⇨ $Clip_D(AB)=\varnothing$

3. Neáu d=R, ⇨ $Clip_D(AB)=\{A_0\}$

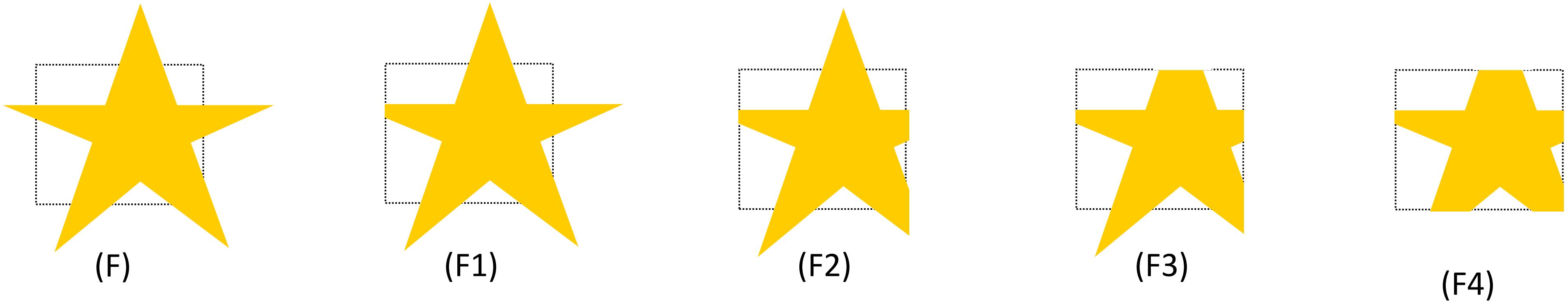
{A_0 : tieáp ñieåm cuûa ñthaúng vôùi ñtroøn}.

4. Neáu d<R

....



F là đa giác và D là hình chữ nhật



1. Với đa giác F, cắt bỏ phần bên trái HCN (nghĩa là bên trái của cạnh trái nối dài) ta thu được đa giác mới F_1

2. Với đa giác F_1 cắt bỏ phần bên phải HCN ta thu được đa giác mới F_2

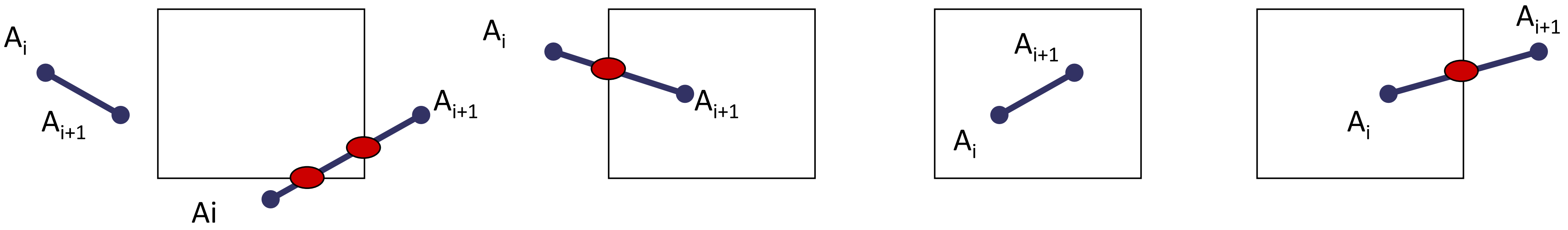
3. Với đa giác F_2 cắt bỏ phần bên trên HCN ta thu được đa giác mới F_3

4. Với đa giác F_3 cắt bỏ phần bên dưới HCN ta thu được đa giác mới F_4

Kết quả: Nếu $F_4 = \varnothing$ thì $Clip_D(F)=\varnothing$.

Ngược lại kết quả xén là đa giác F_4, hay $Clip_D(F)=F_4$

Giải thuật Sutherland - Hodgeman



i.Nếu tất cả các đỉnh đa giác đều nằm trong HCN, hình cần xén chính là đa giác.

ii.Ngược lại: Từ một đỉnh nằm ngoài HCN, chạy theo dọc biên của đa giác. Với mỗi cạnh của đa giác, ta có các trường hợp sau:

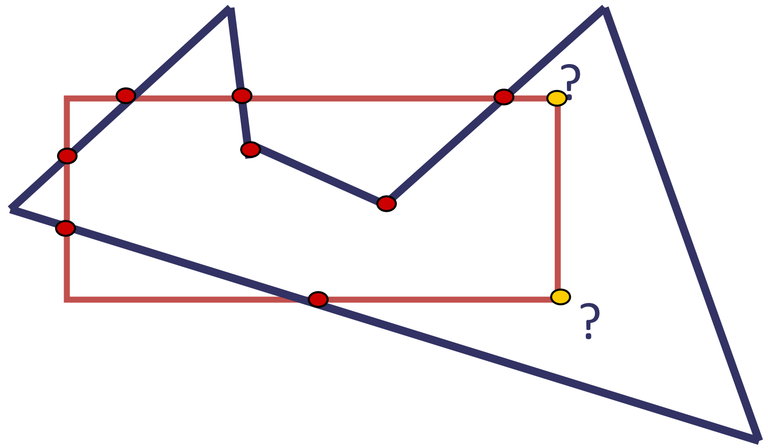
- Nếu cả hai đỉnh đều nằm ngoài hình chữ nhật thì:

  Nếu Ma(A_i) and Ma(A_{i+1}) $\neq$ 0000 thì không lưu đỉnh

  Ngược lại thì lưu hai giao điểm.

- Ai ngoài, A_{i+1} trong: lưu giao điểm P và A_{i+1}.
- Cả hai đỉnh đều nằm trong hình chữ nhật: lưu A_i và A_{i+1}.
- A_i trong, A_{i+1} ngoài: lưu A_i và giao điểm P.

Giải thuật Sutherland - Hodgeman

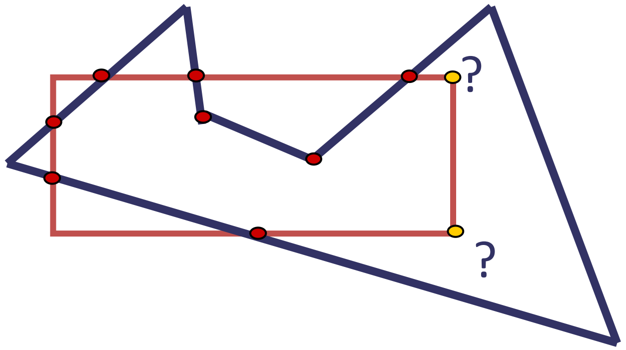


-Sau khi duyệt qua tất cả các cạnh của đa giác,
ta có được một dãy các đỉnh mới phát sinh: B_1, B_2, ..., B_n

Nếu trong dãy các đỉnh mới này có hai đỉnh liên tiếp không nằm trên cùng một cạnh của hình chữ nhật , giả sử hai đỉnh đó là B_i và B_{i+1} thì ta đi dọc các cạnh của hình chữ nhật từ B_i đến B_{i+1} để tìm tất cả các đỉnh của hình chữ nhật nằm trong đa giác rồi bổ sung chúng vào giữa B_i và B_{i+1}.

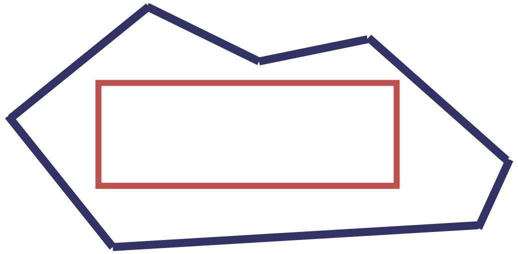
Giải thuật Sutherland - Hodgeman

Tập đỉnh mới tìm được chính là đa giác xén được.

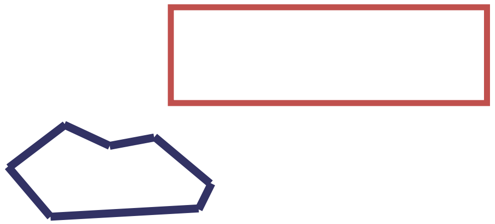


Nếu tập đỉnh mới này rỗng:

+Nếu có một đỉnh của hình chữ nhật nằm trong đa giác thì hình xén được chính là toàn bộ hình chữ nhật.



+Ngược lại, hình xén được là rỗng.

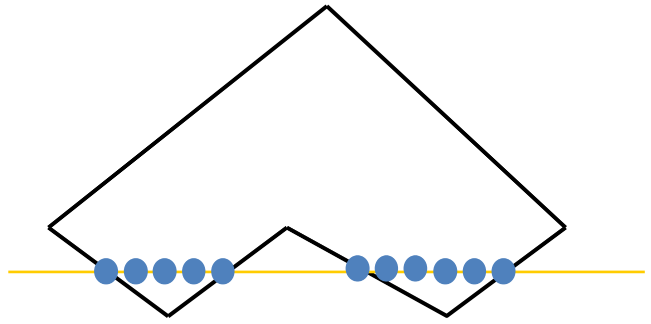


CONTENTs

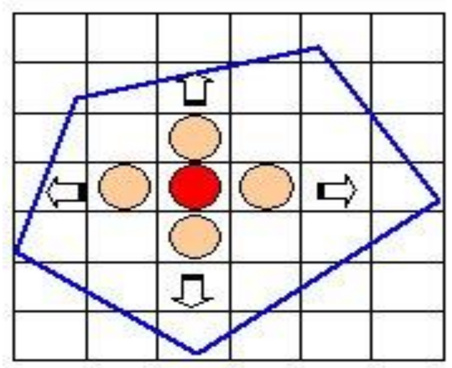
- Vẽ đối tượng đồ họa cơ sở
- Xén hình
- **Tô màu**
- Vẽ chữ và dựng Font
- Các phép biến đổi 2D

- Thuật toán tô màu theo dòng quét
- Thuật toán tô màu theo đường biên

Có 2 cách tiếp cận chính để tô màu một vùng

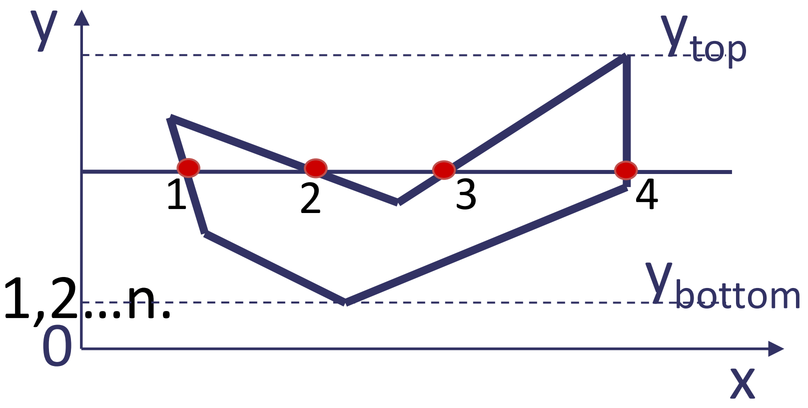


- **Toâ theo doøng queùt** (toâ loaït –scan line fill): xaùc ñònh caùc phaàn giao cuûa caùc doøng queùt keá tieáp nhau vôùi ñöôøng bieân cuûa vuøng toâ, sau ñoù seõ toâ maøu caùc ñieåm thuoäc veà phaàn giao naøy.



- **Toâ theo ñöôøng bieân** (toâ loang – boundary fill): baét ñaàu töø moät ñieåm trong vuøng toâ vaø töø ñoù loang daàn ra cho tôùi khi gaëp caùc ñieåm bieân.

Thuật toán tô màu theo dòng quét



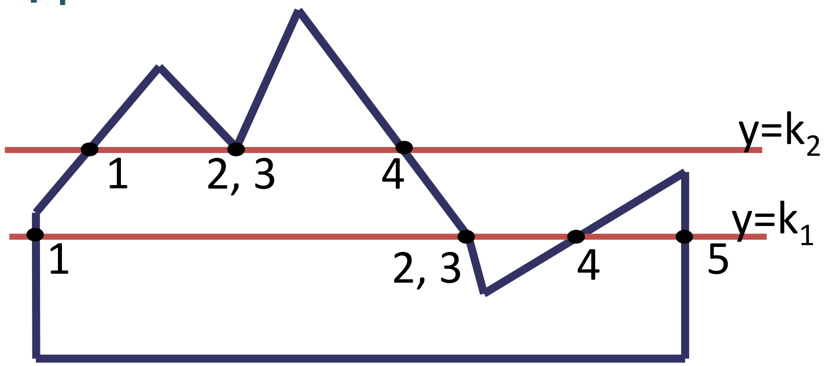
Thuaät toaùn chung:

- Giaû söû vuøng toâ laø moä ña giaùc n ñænh: $P_i(x_i,y_i)$, i=1,2…n.
- Caùc böôùc chính cuûa thuaät toaùn nhö sau:
  - Tìm y_{top} vaø y_{bottom} laø gía trò lôùn nhaát,ø nhoû nhaát cuûa taäp caùc tung ñoä caùc ñænh da giaùc ñaõ cho.
  - ÖÙng vôùi moãi doøng queùt y=k ($y_{bottom} \leq k \leq y_{top}$), laëp:
    - Tìm taát caû hoaønh ñoä giao ñieåm cuûa doøng queùt y=k vôùi caùc caïnh cuûa ña giaùc.
    - Saép xeáp caùc hoaønh ñoä giao ñieåm theo thöù töï taêng daàn: x_1, x_2,…
    - Toâ maøu caùc ñoaïn thẳng treân ñöôøng thaúng y=k lần löôït ñöôïc giôùi haïn bôûi caùc caëp hoøanh ñoä giao ñieåm (x_1,x_2), (x_3,x_4), (x_5,x_6)… (x_{2m-1},x_{2m})

Thuật toán tô màu theo dòng quét

Một số trường hợp:

- Trường hợp 1



- Giải quyết

Quy tắc hình 1 điểm giao (a) và 2 điểm giao (b)

Thuật toán tô màu theo dòng quét

- Tröôøng hôïp 2*:* Neáu giaûi heä phöông trình tìm giao ñieåm cuûa caïnh ña giaùc vôùi moãi doøng queùt seõ gaëp caùc pheùp toaùn nhaân, chia… treân soá thöïc.

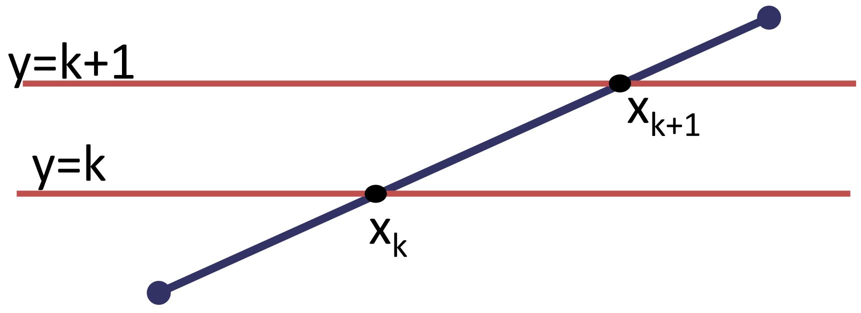
⇨giaûm toác ñoä thuaät toaùn khi laëp laïi nhieàu laàn vôùi moãi doøng queùt.

- Giaûi quyeát: goïi x_k vaø x_{k+1} laàn löôït laø hoaønh ñoä giao ñieåm cuûa moät caïnh naøo ñoù vôùi caùc doøng queùt y=k vaø y=k+1. Ta coù:

$$x_{k+1} - x_k = \frac{y_{k+1} - b}{m} - \frac{y_k - b}{m} = \frac{y_{k+1} - y_k}{m} = \frac{k+1-k}{m} = \frac{1}{m}$$

hay

$$x_{k+1} = x_k + \frac{1}{m}$$

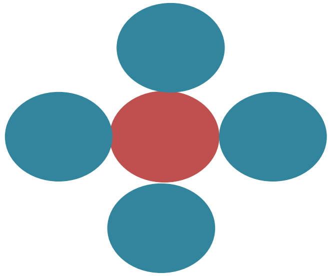


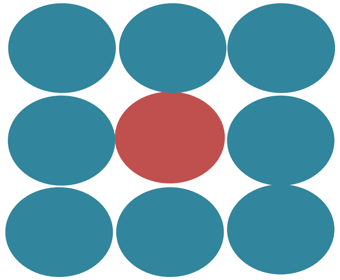
Thuật toán tô màu theo đường biên

YÙ töôûng:

Baét ñaàu töø ñieåm P(x,y) naèm beân trong vuøng toâ, kieåm tra caùc ñieåm laân caän cuûa P ñaõ ñöôïc toâ maøu hay coù phaûi laø ñieåm bieân hay khoâng, neáu khoâng phaûi laø ñieåm ñaõ toâ vaø vaø khoâng phaûi laø ñieåm bieân thì ta seõ toâ maøu noù. Quaù trình naøy ñöôïc laëp laïi cho ñeán khi khoâng coøn toâ ñöôïc ñieåm naøo nöõa thì döøng.

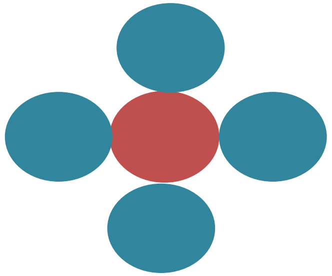
⇨ Choïn ñieåm laân caän: choïn 4 hay 8 laân caän ñoái vôùi ñieåm ñang xeùt.

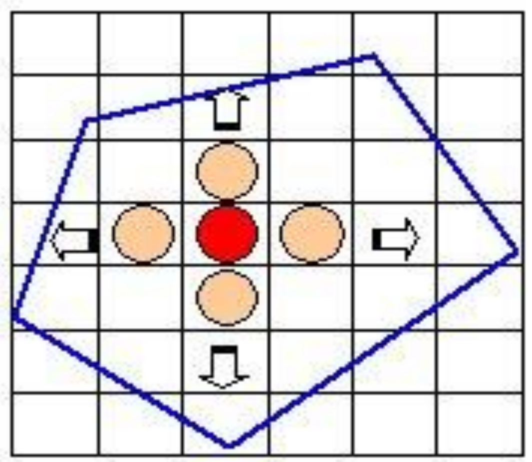




Thuật toán tô màu theo đường biên

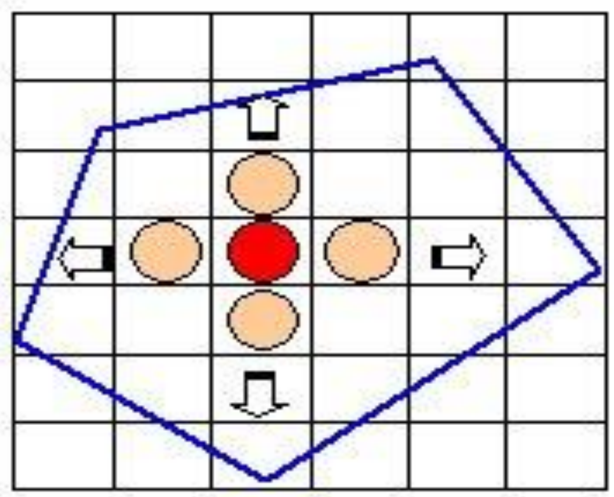
```
void    Boundary Fill (int x,int  y,int  fillColour, int  boundaryColour){
        int    colour=getpixel(x,y);
        if (colour != boundaryColour) & (colour != fillColour){
            Putpixel(x, y, fillColour);
            Boundary Fill(x-1, y,  fillColour, boundaryColour);
            Boundary Fill(x, y+1, fillColour, boundaryColour);
            Boundary Fill(x+1, y, fillColour, boundaryColour);
            Boundary Fill(x, y-1,  fillColour, boundaryColour);
        }
}
```





⇨ Nhược điểm của pp đệ quy là không thực hiện được khi vùng loang có diện tích lớn (tràn Stack).

Thuật toán tô màu theo đường biên



Phương pháp không đệ quy:

Bước 1:Khởi tạo hàng đợi (hoặc stack) với phần tử đầu tiên là P(x,y) đã được tô.

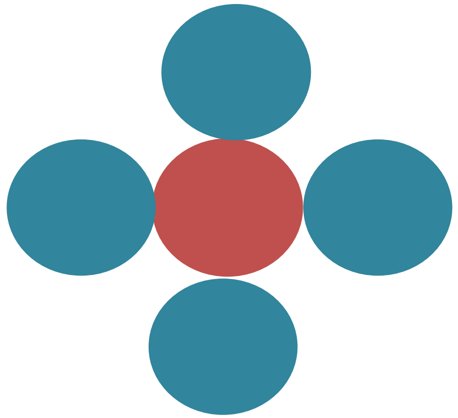
Bước 2: Khi hàng đợi (hoặc stack) không rỗng thì:

+ Lấy ra từ hàng đợi (hoặc stack) một điểm Q.
+ Tìm các điểm lân cận của Q chưa tô thì tô chúng và đưa chúng vào hàng đợi (hoặc stack) .

Bước 2 được lặp đi lặp lại cho đến khi hàng đợi (hoặc stack) rỗng

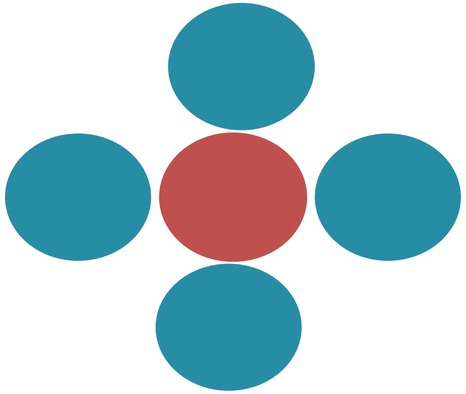
Thuật toán tô màu theo đường biên

```
struct   LIST  {  int x,y;
                  struct   LIST *next;};
struct   LIST *top;
void push(int x,int y){
        struct LIST *P;
        P=(LIST *) calloc(1,sizeof(LIST));
        P->x=x;     P->y=y;     P->next=NULL;
        if   (top!=NULL)        P->next=top;
        top=P;
}
void pop(int *x,int *y){
        struct   LIST *P;
        P=top;     top=top->next;     *x=P->x;     *y=P->y;
        free(P);
}
```



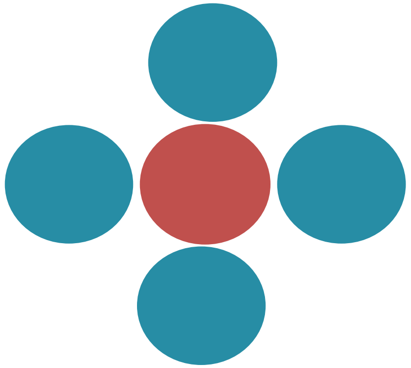
Thuật toán tô màu theo đường biên

```
void BoundaryFill(int x,int y, int fillColour, int boundaryColour){
    int colour = getpixel(x,y);
    if ((colour!= fillColour)&&(colour!=boundaryColour)){
        putpixel(x,y, fillColour);
        push(x,y);
    }
}
```



Thuật toán tô màu theo đường biên

```
void BoundaryFill_Stack(int x0,int y0,int fillColour,int boundaryColour){
	int x,y;
	putpixel(x0,y0, fillColour);
	top=NULL;
	push(x0,y0);
	while(top!=NULL){
		pop(&x,&y);
		BoundaryFill (x-1,y, fillColour, boundaryColour);
		BoundaryFill (x+1,y,fillColour, boundaryColour);
		BoundaryFill (x,y+1,fillColour, boundaryColour);
		BoundaryFill (x,y-1, fillColour, boundaryColour);
	}
}
```



Flood Fill (Boundary fill)	Scanline Fill (Scan Conversion)
Đơn giản	Phức tạp hơn
Thuật toán rời rạc hóa trong không gian màn hình	Thuật toán rời rạc hóa trong đối tượng hoặc/và không gian màn hình
Yêu cầu gọi hệ thống GetPixel/Val	Độc lập với thiết bị
Đòi hỏi điểm seed	Không đòi hỏi điểm seed
Yêu cầu stack rất lớn	Yêu cầu stack nhỏ

CONTENTs

Các khái niệm cơ sở về Font

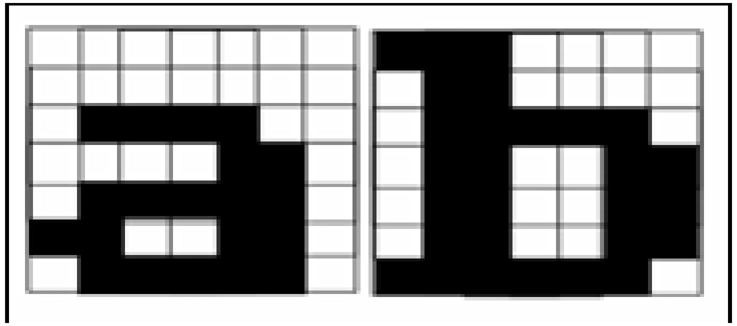
- Font được *Guttenberg* thiết kế, sử dụng từ nhiều thế kỷ, ngày nay rất phong phú.
- Font là tập đầy đủ các ký tự có kiểu dáng (style)
  - Weight: light, normal, bold
  - Shape: round, oval, straight
  - Posture: Oblique, *Italic*
  - Serif, sans-serif

Các loại Font

- font bitmap (raster)
- font vector
- font TrueType

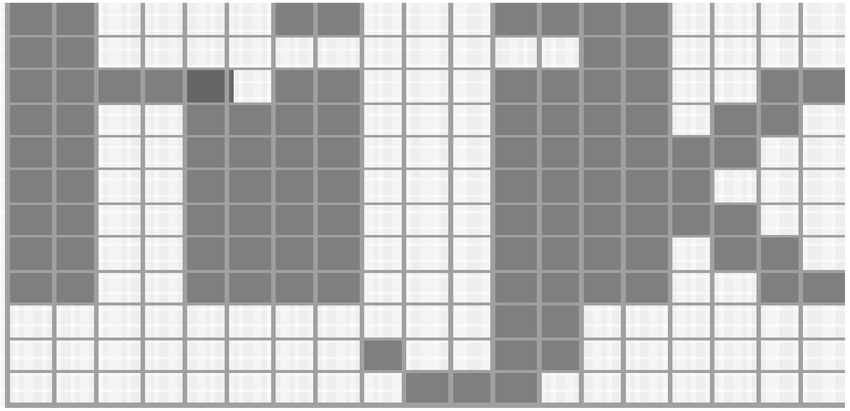
Font raster (bitmap)

- Là loại phông đầu tiên, ngày nay vẫn đang sử dụng
- Ban đầu font bitmap được nhúng trong các vỉ điều khiển màn hình, máy in
- Mỗi ký tự là một bitmap chữ nhật nhỏ
  - Font/typeface: set of character shapes
  - Fontcache: các ký tự theo chuỗi liên tiếp nhau trong bộ nhớ
  - Dạng cơ bản: (thường N, nghiêng I, đậm B, nghiêng đậmB+I)
  - Thuộc tính: colour, size, spacing and orientation



Font raster (bitmap)

```
Typedef struct {
          int leftx,
          int width;        //độ rộng chữ
  } Charlocation;           //Vị trí của chữ
Typedef struct {
          CacheId;
          Heiglit;          // Độ cao chữ
          CharSpace;        // Khoảng cách giữa các ký tự
          Charlocation Table [128];
  } fontcache
```



Nhận xét về font bitmap

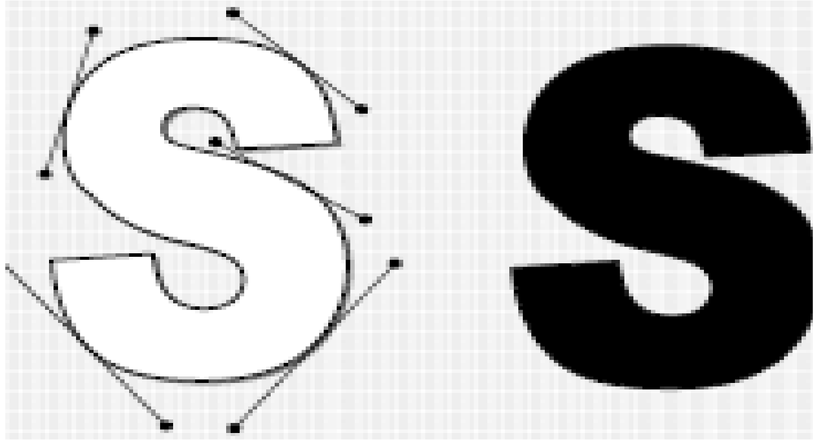
- Có độ rộng và độ cao cố định
- Lưu trữu: tách biệt các ảnh phông. Vị trí byte thứ nhất của khối bitmap trong bộ font:

  Offset = (ASCII code) * (Bytes per character)

- Ưu điểm
  - Hiển thị nhanh, đơn gian trong việc sinh ký tự
  - Dễ tạo lập và dễ sửa đổi
- Nhược điểm
  - Kích thước không đổi
  - Co dãn, các phép biến đổi (I, B, scale) đòi hỏi phải lưu trữ thêm
  - Dung lượng lưu trữ lớn

Font vector

- Sử dụng ngôn ngữ mô tả nào đó
  - Ngôn ngữ mô tả bao gồm các lệnh như Line, Curve, Polygon...
  - Tọa độ: tương đối trong chữ nhật chứa ký tự
  - Chương trình con xử lý các lệnh để hiển thị



Font vector

- **Ưu điểm**
  - Chất lượng cao
  - Dễ co giãn, trơn tru, dễ tạo lập hiệu ứng đặc biệt: xoay, gập, cong...
  - Lưu trữ gọn nhẹ

- **Nhược điểm**
  - Phức tạp (tính toán phương trình)
  - Khi hiển thị font nhỏ: chậm hơn bitmap font.
  - Vấn đề hiển thị font nét chữ dày.

Font True Type

- True Type là công nghệ font của Apple Computer Inc. (1987); Tác giả: Kathryn Weisberg, Sampo Kaasila...
- MS bắt đầu sử dụng True Type vào đầu 1992 trên Windows 3.1
- Công nghệ True Type bao gồm:
  - Các tệp chứa True Type Fonts (TTF)
  - Bộ raster hóa True Type của hệ điều hành (MacOS, Windows...) trước khi hiển thị, in trên giấy.
- OpenType: 5-1996 MS và Adobe System kết hợp công nghệ True Type với PostScript.

Font True Type

- **File TTF chứa**
  - Mô tả hình dạng ký tự
  - Các thông tin khác: tên font, bản quyền, hãng sản xuất..
- **Mô tả ký tự trong TTF:**
  - Mô tả đường viền font -> TTF là font outline
  - Mô tả toán học của ký tự từ dãy các điểm
  - Viền ký tự được tạo bởi các line và B-splines toàn phương
    - Mỗi B-Spline ứng với dãy các Bezier bậc 2 được xác định bởi ba điểm điều khiển.

Thí dụ: có ba điểm điều khiển (A_x, A_y), (B_x B_y), (C_x, C_y)

Các điểm P trên đường cong được tính với t trong đoạn [0,1]

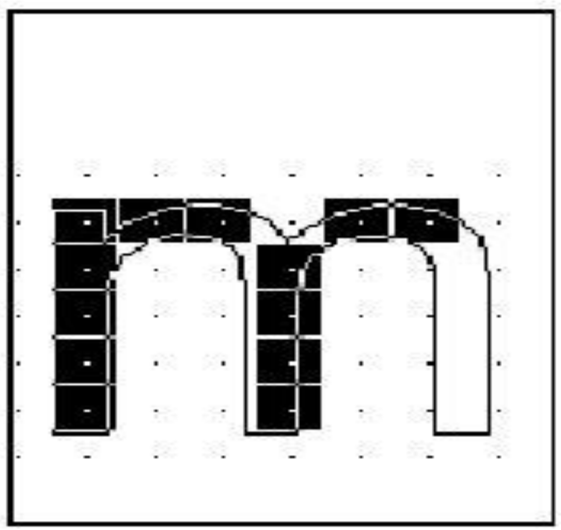
$$p_x = (1-t)^2 .A_x + 2t(1-t).B_x + t^2.C_x$$
$$p_y = (1-t)^2 .A_y + 2t(1-t).B_y + t^2.C_y$$

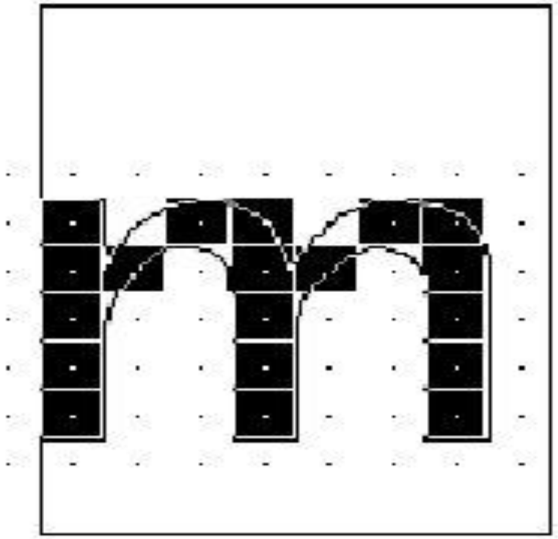
Font True Type

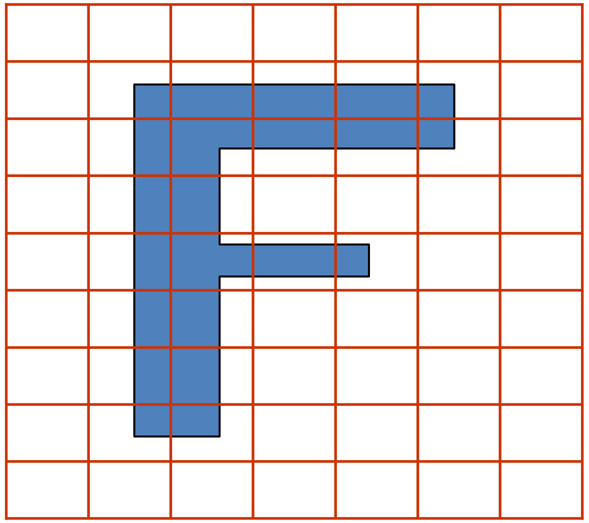
- Ví dụ mô tả outline của chữ b Arial:
  - các điểm 4, 5, 6 xác định Bézier
  - các điểm từ 6 đến 11 xác định B-Spline, chứa 4 Béziers
- Đánh số điểm điều khiển
  - Theo thứ tự: tô màu phía phải
  - Đánh số kế tiếp giữa các contour
- Chuyển outline sang bitmap

Font True Type

- **Hinting**
  - Sau khi chuyển outline sang bitmap thường phải hiệu chỉnh nét vẽ để có chất lượng cao
  - Mỗi nét trong font có chương trình con bằng ngôn ngữ thông dịch (tựa asm) kèm theo để thực hiện hiệu chỉnh (Ngôn ngữ này có khoảng 150 lệnh, kích thước lệnh 1 byte)







Font True Type

- **MS Windows sử dụng TTF?**
  - Nạp font từ tệp
  - Trình Scaler:
    - Ánh xạ tọa độ điểm điều khiển của glyph từ FUnit sang tọa độ thiết bị (pixel/dot). (Co dãn đường viền đến kích thước yêu cầu theo mật độ cho trước trên thiết bị ra).
  - Trình Interpreter:
    - Áp dụng 'hint' cho đường viền: biến đổi đường cong để hình thành đường viền đã hiệu chỉnh (grid-fitting).
  - Trình Scan-converter:
    - Tô trong đường viền đã hiệu chỉnh bằng các pixel để tạo bitmap cho chữ đặc.
  - Hiển thị/in các bitmap ký tự.

CONTENTs

- Vẽ đối tượng đồ họa cơ sở
- Xén hình
- Tô màu
- Vẽ chữ và dựng Font
- **Các phép biến đổi 2D**

- Các phép biến đổi hình học cơ sở
- Phép biến đổi ngược
- Hệ toạ độ thuần nhất
- Kết hợp các phép biến đổi

Các phép toán cơ sở với ma trận

- Cộng, trừ ma trận

$$[A(m, n)] + [B(m, n)] = [C(m, n)]$$

$$\left\lfloor c_{ij} \right\rfloor = \left\lfloor a_{ij} \right\rfloor + \left\lfloor b_{ij} \right\rfloor$$

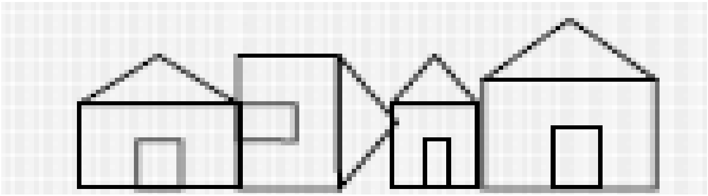
- Nhân hai ma trận

$$[A(m, n)].[B(n, p)] = [C(m, p)]$$

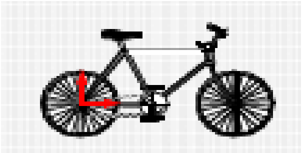
$$c_{jk} = \sum_{i=1}^{n} a_{ji} b_{ik}$$ j=1,...,m và k=1,...,p

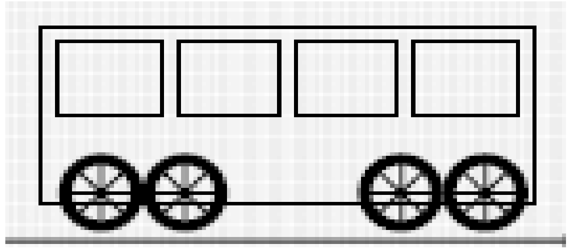
Ứng dụng biến đổi 2D

- **Mô hình hóa (modeling)**
  - Định vị và thay đổi kích thước các phần của đối tượng phức tạp
- **Quan sát (viewing)**
  - Định vị và quan sát camera ảo



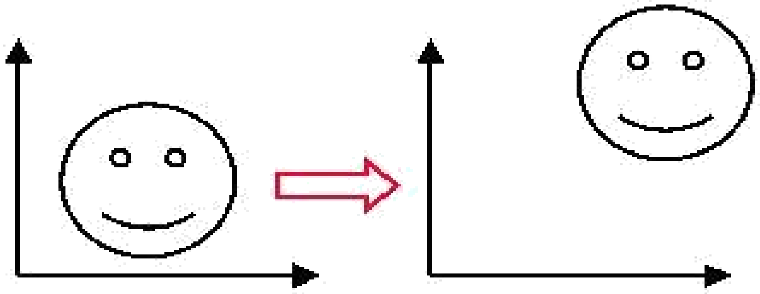
- **Animation**
  - Xác định đối tượng chuyển động thay đổi theo thời gian.



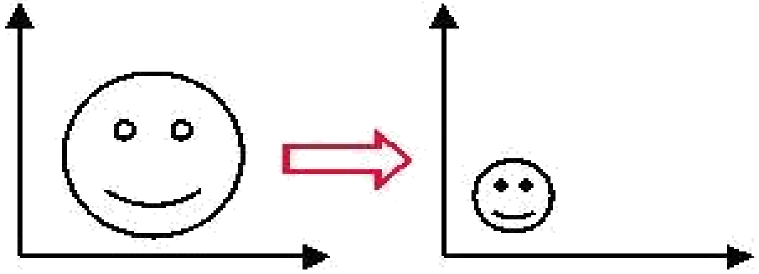


Các phép biến đổi hình học cơ sở:

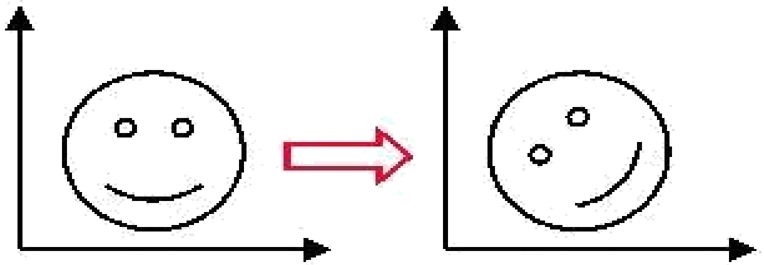
- Tịnh tiến (translation)
- Quay (rotation)
- Tỉ lệ (scaling).
- Đối xứng (reflection)
- Biến dạng (shearing).



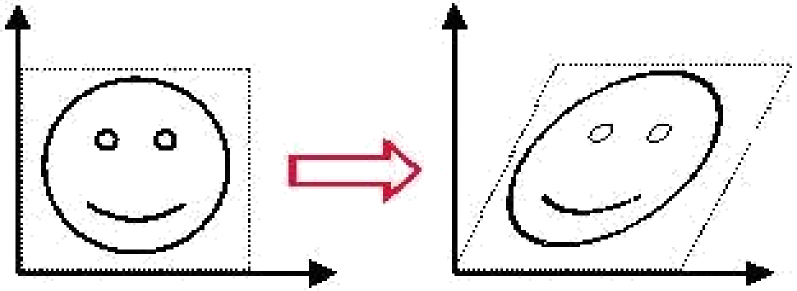
translation



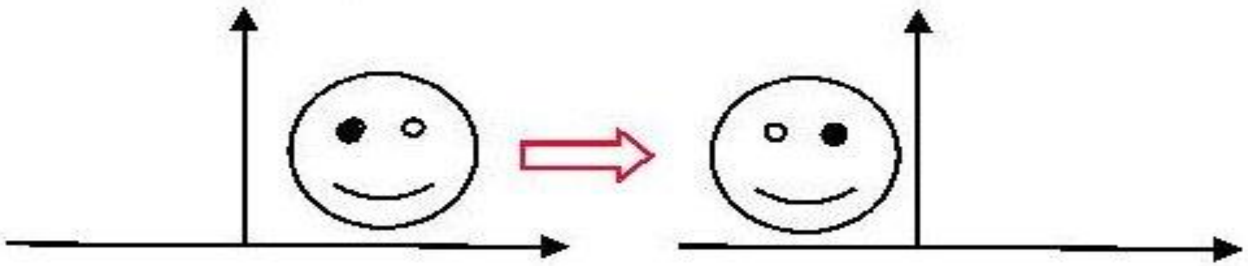
scaling



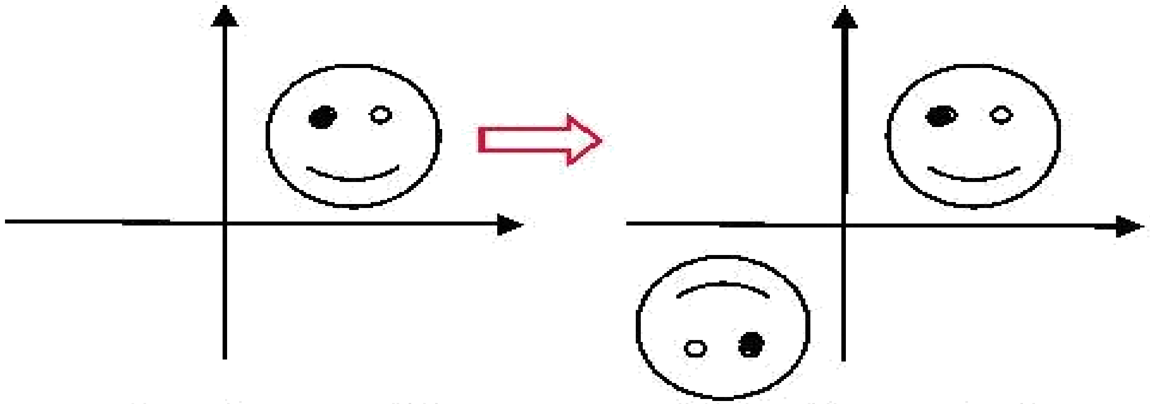
rotation



shear



reflection with respect to the y-axis



reflection with respect to the origin

Pheùp bieán ñoåi ñieåm laø moät aùnh xaï T ñöôïc ñònh nghóa:

$$T: \quad R^2 \quad \rightarrow \quad R^2$$

$$P(x,y) \rightarrow Q(x',y')$$

hay T laø haøm soá T(x,y) theo hai bieán x,y

Pheùp bieán ñoåi affine laø pheùp bieán ñoåi vôùi f(x,y) vaø g(x,y) laø caùc haøm tuyeán tính coù daïng:

$$\begin{cases} x' = f(x,y) = ax + cy + e \\ y' = g(x,y) = bx + dy + f \end{cases}$$

vôùi a, b, c, d, e, f $\in R$ vaø ad-bc$\neq 0$

$$\begin{cases} x' = f(x,y) = ax + cy + e \\ y' = g(x,y) = bx + dy + f \end{cases}$$ vôùi a, b, c, d, e, f $\in$R vaø ad-bc$\neq$0

Viết dưới dạng daïng ma traän:

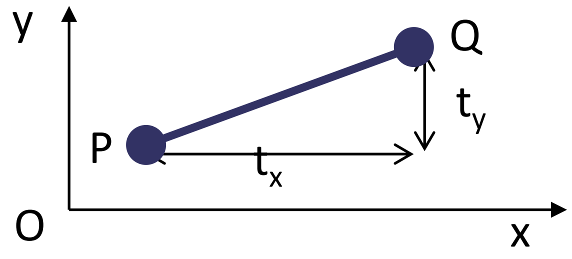
$$[x' \quad y'] = [x \quad y] \times \begin{bmatrix} a & b \\ c & d \end{bmatrix} + [e \quad f]$$

Trong ñoù: $M = \begin{bmatrix} a & b \\ c & d \end{bmatrix}$ laø ma traän bieán ñoåi.

$T = [e \quad f]$ laø vector offset hay vector tònh tieán.

Phép tịnh tiến

Goïi t_x vaø t_y laø ñoä dôøi theo truïc x vaø y

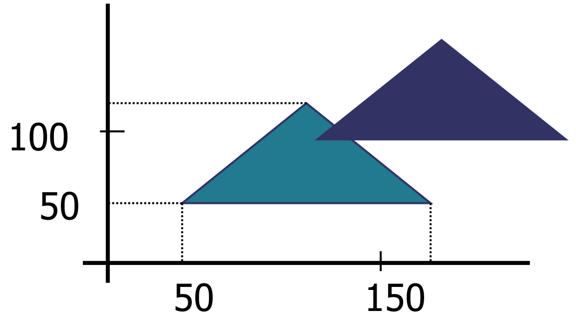


Toaï ñoä cuûa ñieåm môùi Q(x', y') seõ laø:

$$\begin{cases} x' = x + t_x \\ y' = y + t_y \end{cases}$$

hay $\begin{bmatrix} x' & y' \end{bmatrix} = \begin{bmatrix} x & y \end{bmatrix} \times \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} + \begin{bmatrix} t_x & t_y \end{bmatrix}$ hay Q=P+T

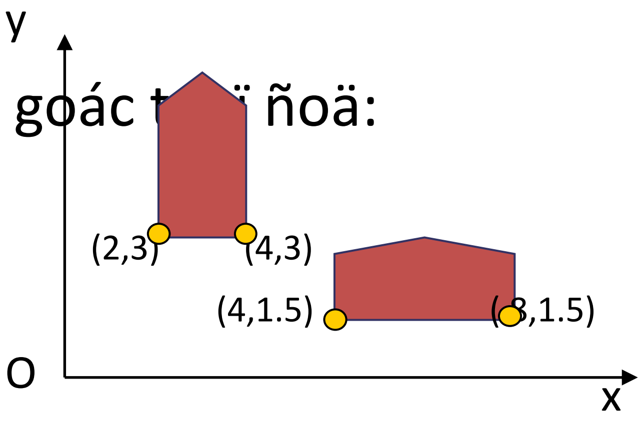
Vôùi: $T = \begin{bmatrix} t_x & t_y \end{bmatrix}$ laø vector tònh tieán/vector ñoä dôøi.



$M = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$ laø ma traän ñôn vò.

Phép tỉ lệ

Goïi heä soá tæ leä: s_x vaø s_y, pheùp tæ leä với taâm tæ leä laø goác t ï ñoä:



Bieán ñoåi tæ leä vôùi sx=2 vaø sy=0

$$\begin{cases} x' = x.s_x \\ y' = y.s_y \end{cases} \quad \text{hay} \quad [x' \quad y'] = [x \quad y] \times \begin{bmatrix} s_x & 0 \\ 0 & s_y \end{bmatrix}$$

hay Q=P.M vôùi $M = \begin{bmatrix} s_x & 0 \\ 0 & s_y \end{bmatrix}$

Khi: $(s_x,s_y)=(-1,1) \rightarrow$ pheùp laät (flipping), coù aûnh ñoái xöùng qua truïc y

$(s_x,s_y)=(1,-1) \rightarrow$ pheùp laät (flipping), coù aûnh ñoái xöùng qua truïc x

$|s_x|$, $|s_y| < 1 \rightarrow$ thu nhoû ñoái töôïng.

$|s_x|$, $|s_y| > 1 \rightarrow$ pheùp phoùng to ñoái töôïng.

$s_x=s_y=s \rightarrow$ pheùp ñoàng daïng (uniform scaling)

Phép tỉ lệ

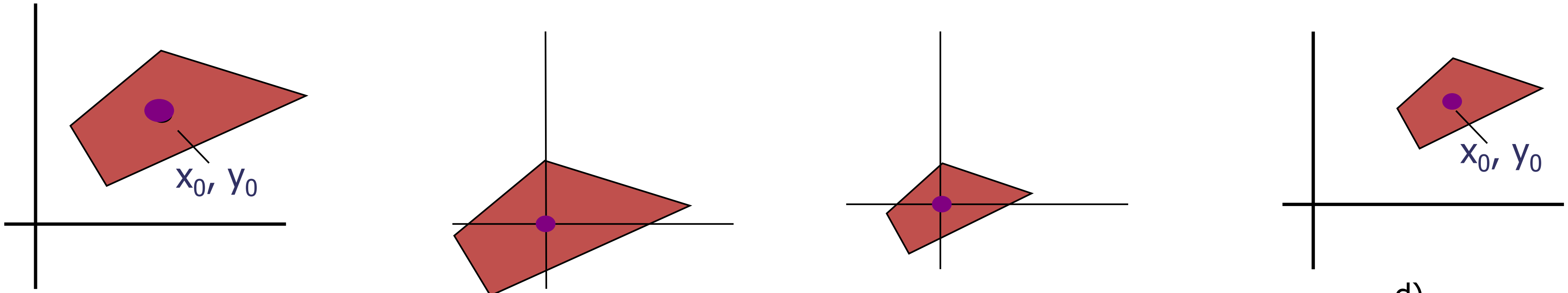
Pheùp bieán ñoåi tæ leä vôùi taâm tæ leä laø ñieåm $T(t_x,t_y)$:

Xaây döïng töø nhöõng pheùp bieán ñoåi cô baûn sau:

+ Tònh tieán theo vector $(-t_x,-t_y)$ ñeå ñöa veà goác toaï ñoä O

+ Bieán ñoåi tæ leä quanh goác toaï ñoä O theo heä soá tæ leä s_x vaø s_y

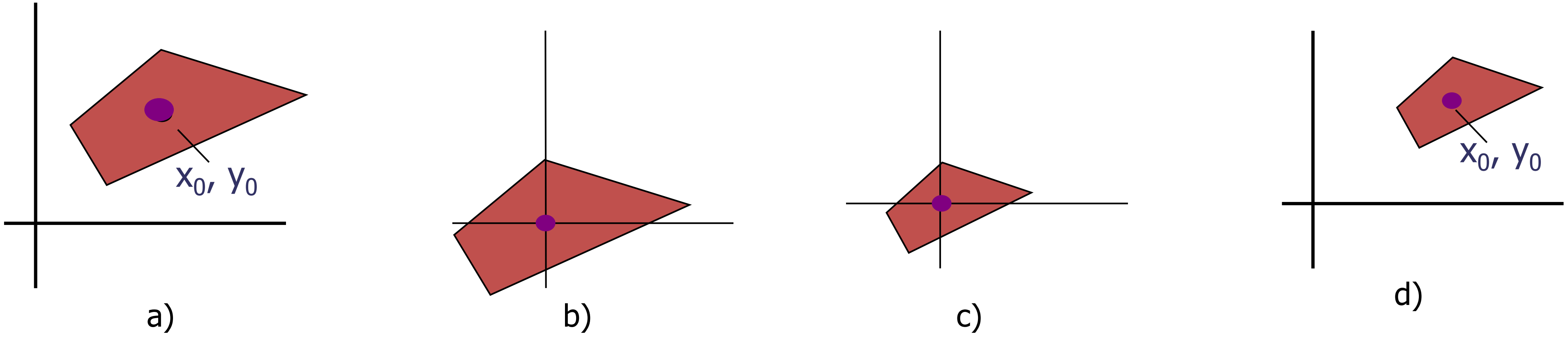
+ Tònh tieán theo vector (t_x, t_y) ñeå ñöa veà vò trí ban ñaàu.



a) b) c) d)

Phép tỉ lệ

Pheùp bieán ñoåi tæ leä vôùi taâm tæ leä laø ñieåm T(t_x,t_y):



Coâng thöùc bieán ñoåi nhö sau: Q=(P-T).M+T vôùi $T = \begin{bmatrix} t_x & t_y \end{bmatrix}$

hay $$\begin{bmatrix} x' & y' \end{bmatrix} = \left(\begin{bmatrix} x & y \end{bmatrix} - \begin{bmatrix} t_x & t_y \end{bmatrix} \right) \times \begin{bmatrix} Sx & 0 \\ 0 & Sy \end{bmatrix} + \begin{bmatrix} t_x & t_y \end{bmatrix}$$

Phép quay

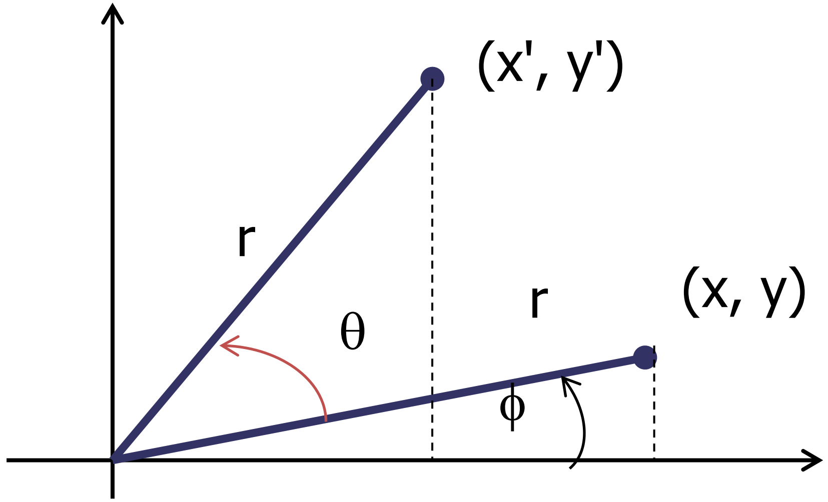
Pheùp quay ñieåm P(x,y) quanh goác toaï ñoä moät goùc θ :

$$x' = r\cos(\phi + \theta) = r\cos\phi\cos\theta - r\sin\phi\sin\theta$$
$$y' = r\sin(\phi + \theta) = r\sin\phi\cos\theta + r\cos\phi\sin\theta$$

$$x = r\cos\phi,\ y = r\sin\phi$$

$$x' = x\cos\theta - y\sin\theta$$
$$y' = x\sin\theta + y\cos\theta$$

$$\begin{bmatrix} x' & y' \end{bmatrix} = \begin{bmatrix} x & y \end{bmatrix} \times \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix}$$

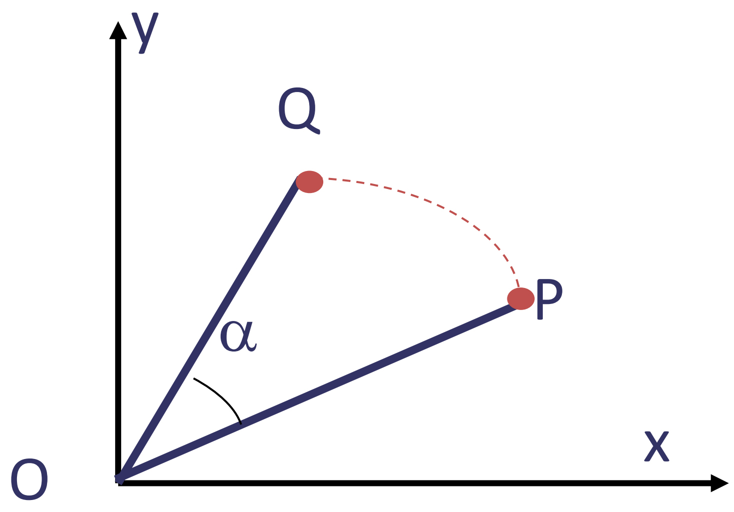


Phép quay

Pheùp quay ñieåm P(x,y) quanh goác toaï ñoä moät goùc α:

Ta coù coâng thöùc bieán ñoåi: $\begin{cases} x' = x.\cos\alpha - y.\sin\alpha \\ y' = x.\sin\alpha + y.\cos\alpha \end{cases}$

hay $\begin{bmatrix} x' & y' \end{bmatrix} = \begin{bmatrix} x & y \end{bmatrix} \times \begin{bmatrix} \cos\alpha & \sin\alpha \\ -\sin\alpha & \cos\alpha \end{bmatrix}$



hay Q=P.M

vôùi ma traän M: $M = \begin{bmatrix} \cos\alpha & \sin\alpha \\ -\sin\alpha & \cos\alpha \end{bmatrix}$

Phép quay

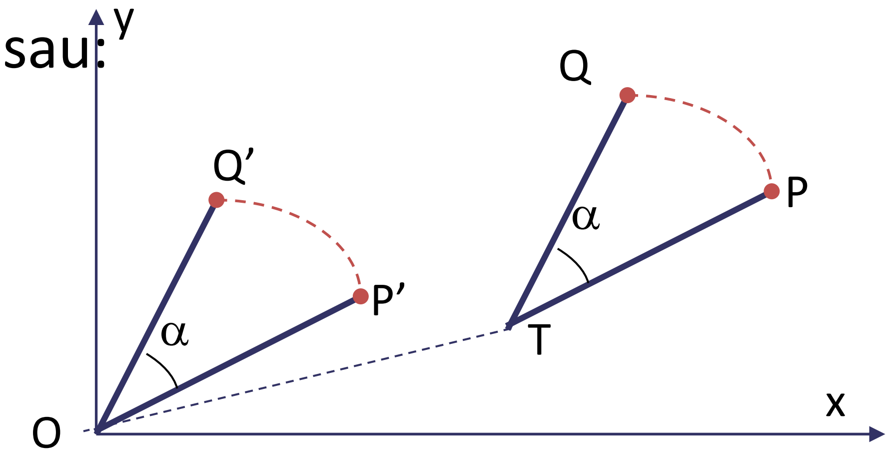
- **Pheùp quay quanh ñieåm T(t_x,t_y) moät goùc α:**

  Xaây döïng töø nhöõng pheùp bieán ñoåi cô baûn sau:

  +Tònh tieán theo vector (-t_x,-t_y)

  +Quay quanh goác toaï ñoä O moät goùc α

  +Tònh tieán theo vector (t_x,t_y)



Coâng thöùc bieán ñoåi nhö sau:

Q=(P-T).M+T vôùi $T = \begin{bmatrix} t_x & t_y \end{bmatrix}$

hay: $\begin{bmatrix} x' & y' \end{bmatrix} = \left(\begin{bmatrix} x & y \end{bmatrix} - \begin{bmatrix} t_x & t_y \end{bmatrix} \right) \times \begin{bmatrix} \cos\alpha & \sin\alpha \\ -\sin\alpha & \cos\alpha \end{bmatrix} + \begin{bmatrix} t_x & t_y \end{bmatrix}$

Phép biến đổi ngược

Vôùi pheùp bieán ñoåi affine ta coù: Q=P.M+T ⇨ P=(Q-T).M^{-1}

Vôùi $M = \begin{bmatrix} a & b \\ c & d \end{bmatrix}$ (ad-bc≠0) thì $M^{-1} = \frac{1}{ad-bc}\begin{bmatrix} d & -b \\ -c & a \end{bmatrix}$

☞ Pheùp tònh tieán: ñôn giaûn laø tröø x vaø y ñi moät löôïng t_x vaø t_y.

☞ Pheùp bieán ñoåi tæ leä: $M^{-1} = \frac{1}{s_x s_y}\begin{bmatrix} s_y & 0 \\ 0 & s_x \end{bmatrix} = \begin{bmatrix} 1/s_x & 0 \\ 0 & 1/s_y \end{bmatrix}$

☞ Pheùp quay: $M^{-1} = \begin{bmatrix} \cos\alpha & -\sin\alpha \\ \sin\alpha & \cos\alpha \end{bmatrix}$

Hệ tọa độ thuần nhất (hormogeneous coordinates)

Pheùp bieán ñoåi affine 2Dø: **Q=P.M+T** (vôùi M laø ma traän 2x2 vaø T laø vector tònh tieán)

Daïng ma traän cuûa pheùp bieán ñoåi affine 2D laø:

$$[x' \quad y'] = [x \quad y] \times \begin{bmatrix} a & b \\ c & d \end{bmatrix} + [e \quad f]$$

Ñeå chæ coøn pheùp nhaân ma traän, ta ñöa noù veà daïng:

$$[x' \quad y' \quad 1] = [x \quad y \quad 1] \times \begin{bmatrix} a & b & 0 \\ c & d & 0 \\ e & f & 1 \end{bmatrix}$$

P(x,y,1) vaø Q(x’,y’,1) ñöôïc goïi laø toaï ñoä thuaàn nhaát.

Coâng thöùc cuûa pheùp bieán ñoåi theo heä toaï ñoä thuaàn nhaát: **Q=P.M**

(vôùi M laø ma traän 3x3).

Ta coù : $M = \begin{pmatrix} a & b & 0 \\ c & d & 0 \\ e & f & 1 \end{pmatrix}$ Khi ñoù: $M^{-1} = \dfrac{1}{ad - bc}\begin{pmatrix} d & -b & 0 \\ -c & a & 0 \\ cf - de & be - af & 1 \end{pmatrix}$

Ma traän cuûa caùc pheùp bieán ñoåi cô sôû ôû heä toaï ñoä thuaàn nhaát ñöôïc vieát laïi:

-Pheùp tònh tieán: $M = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ tr_x & tr_y & 1 \end{pmatrix} \Rightarrow M^{-1} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -tr_x & -tr_y & 1 \end{pmatrix}$

-Pheùp bieán ñoåi tæ leä: $M = \begin{pmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & 1 \end{pmatrix} \Rightarrow M^{-1} = \dfrac{1}{s_x s_y}\begin{pmatrix} s_y & 0 & 0 \\ 0 & s_x & 0 \\ 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 1/s_x & 0 & 0 \\ 0 & 1/s_y & 0 \\ 0 & 0 & 1 \end{pmatrix}$

-Pheùp quay: $M = \begin{pmatrix} \cos\alpha & \sin\alpha & 0 \\ -\sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{pmatrix} \Rightarrow M^{-1} = \begin{pmatrix} \cos\alpha & -\sin\alpha & 0 \\ \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{pmatrix}$

Pheùp Ñoái xöùng (reflection)

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

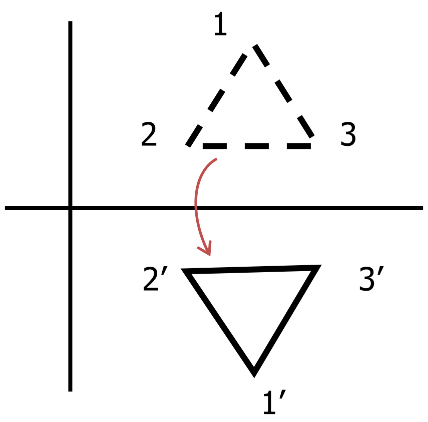
Đối xứng
qua trục x

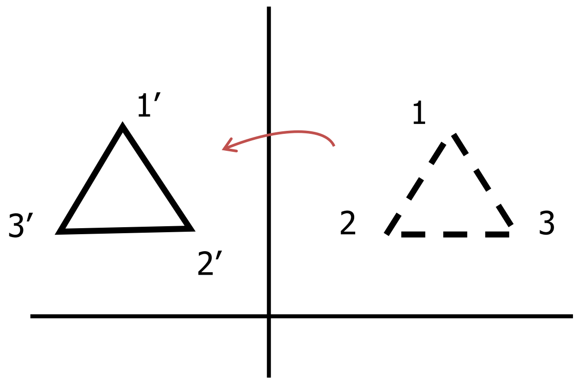
$$\begin{bmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

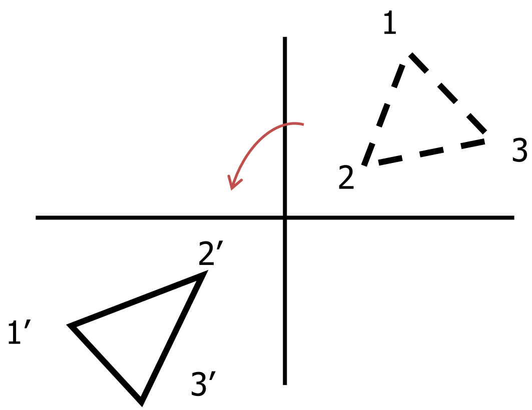
Đối xứng
qua trục y

$$\begin{bmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Đối xứng
qua gốc tọa độ







Kết hợp các phép biến đổi

Quaù trình aùp duïng caùc pheùp bieán ñoåi lieân tieáp ñeå taïo neân moät pheùp bieán ñoåi toång theå goïi laø söï keát hôïp caùc pheùp bieán ñoåi.

Giaû söû coù 2 pheùp bieán ñoåi: T_1: $P \rightarrow Q$

T_2: $Q \rightarrow W$

Ta tìm pheùp bieán ñoåi keát hôïp: T: $P \rightarrow W$

ÔÛ heä toaï ñoä thuaàn nhaát:

+T_1 coù ma traän bieán ñoåi M_1 neân: $Q = P.M_1$

+T_2 coù ma traän bieán ñoåi M_2 neân: $W = Q.M_2 = (P.M_1).M_2 = P.(M_1.M_2)$

$= P.M$

Vaäy keát hôïp hai pheùp bieán ñoåi cuõng laø moät pheùp bieán ñoåi coù ma traän bieán ñoåi : $\mathbf{M = M_1.M_2}$

Kết hợp các phép tịnh tiến

Goïi T_1 vaø T_2 laø 2 pheùp tònh tieán sao cho: T_1: $P(x,y) \rightarrow P'$

T_2: $P' \rightarrow Q(x',y')$

Ma traän bieán ñoåi töø P sang Q laø:

$$M = M_1 M_2 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ tr_{x1} & tr_{y1} & 1 \end{pmatrix} \times \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ tr_{x2} & tr_{y2} & 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ tr_{x1} + tr_{x2} & tr_{y1} + tr_{y2} & 1 \end{pmatrix}$$

Vaäy Q=P.M hay $[x' \quad y'] = [x \quad y] \times \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ tr_{x1} + tr_{x2} & tr_{y1} + tr_{y2} & 1 \end{pmatrix}$

Vaäy, keát hôïp hai hay nhieàu pheùp tònh tieán laø moät pheùp tònh tieán.

Kết hợp các phép tỉ lệ

Pheùp bieán ñoåi tæ leä với taâm tæ leä laø goác toaï ñoä:

Goïi T_1 vaø T_2 laø 2 pheùp tæ leä sao cho: T_1: $P(x,y) \rightarrow P'$

T_2: $P' \rightarrow Q(x',y')$

Ma traän bieán ñoåi töø P sang Q laø:

$$M = M_1 M_2 = \begin{pmatrix} s_{x1} & 0 & 0 \\ 0 & s_{y1} & 0 \\ 0 & 0 & 1 \end{pmatrix} \times \begin{pmatrix} s_{x2} & 0 & 0 \\ 0 & s_{y2} & 0 \\ 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} s_{x1}s_{x2} & 0 & 0 \\ 0 & s_{y1}s_{y2} & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

Vaäy Q=P.M hay $[x' \quad y'] = [x \quad y] \times \begin{pmatrix} s_{x1}s_{x2} & 0 & 0 \\ 0 & s_{y1}s_{y2} & 0 \\ 0 & 0 & 1 \end{pmatrix}$

Vaäy, keát hôïp hai hay nhieàu pheùp tæ leä laø moät pheùp tæ leä.

Kết hợp các phép tỉ lệ

Pheùp bieán ñoåi tæ leä vôùi taâm tæ leä laø ñieåm I:

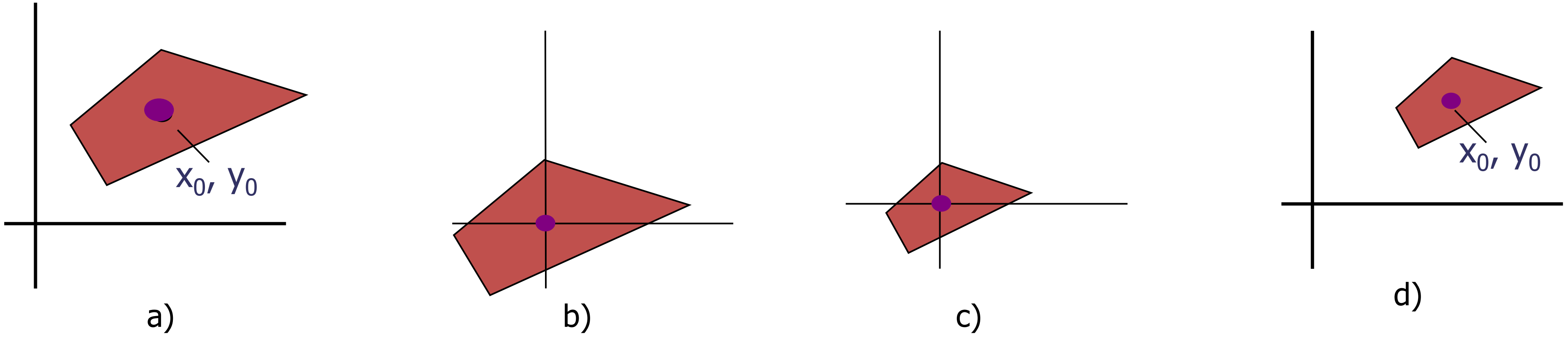
Giaû söû taâm tỉ lệ coù goác toaï ñoä I(x0 , y0). Pheùp tæ leä taâm I ñöôïc keát hôïp töø

caùc pheùp bieán ñoåi cô sôû sau:

-Tònh tieán theo vector (-x0 ,-y0) ñeå ñöa taâm tỉ lệ veà goác toaï ñoä .(M1)

-Bieán ñoåi tæ leä quanh goác toaï ñoä O theo tæ leä sx vaø sy.(M2)

-Tònh tieán theo vector (x0 ,y0) ñeå ñöa taâm tỉ lệ veà vò trí ban ñaàu.(M3)



Kết hợp các phép tỉ lệ

Pheùp bieán ñoåi tæ leä vôùi taâm tæ leä laø ñieåm I:

Ta coù ma traän cuûa pheùp bieán ñoåi keát hôïp laø:

M = M1.M2.M3

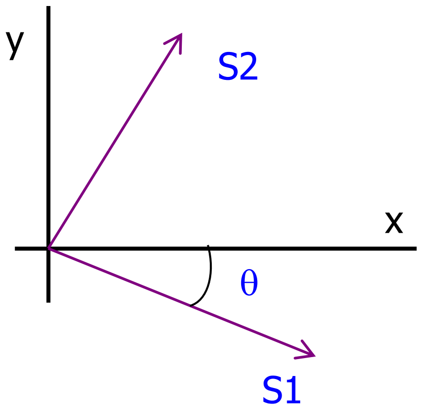
$$= \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -x_0 & -y_0 & 1 \end{pmatrix} \times \begin{pmatrix} Sx & 0 & 0 \\ 0 & Sy & 0 \\ 0 & 0 & 1 \end{pmatrix} \times \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ x_0 & y_0 & 1 \end{pmatrix}$$

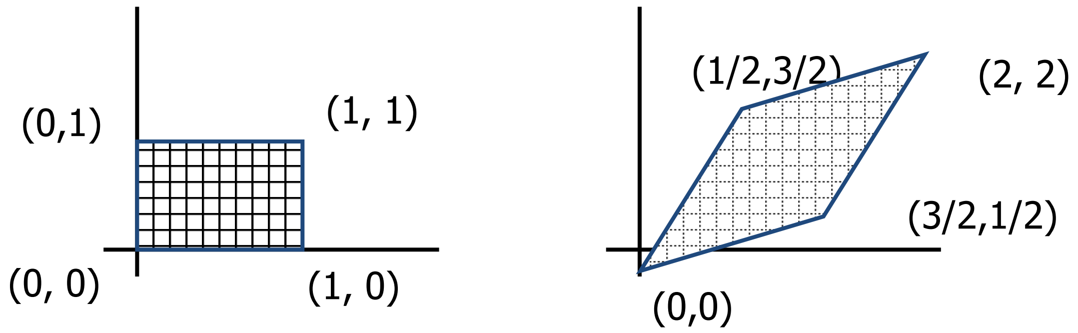
$$= \begin{pmatrix} Sx & 0 & 0 \\ 0 & Sy & 0 \\ (1 - Sx).x_0 & (1 - Sy).y_0 & 1 \end{pmatrix}$$

Kết hợp các phép tỉ lệ

Pheùp bieán ñoåi tæ leä theo höôùng tuøy yù:

- Biến đổi tỉ lệ cơ bản: tỷ lệ Sx và Sy áp dụng theo chiều trục x và y
- Tỉ lệ theo hướng tùy ý: thực hiện chuyển đổi gộp: xoay và tỷ lệ
- Vấn đề: cho hình vuông ABCD, biến đổi tỉ lệ nó theo hướng như biểu diễn trên hình a) và theo tỷ lệ S1, S2.





Kết hợp các phép tỉ lệ

Pheùp bieán ñoåi tæ leä theo höôùng tuøy yù:

- Giải pháp
  - Xoay hướng S1, S2 sao cho trùng với trục x và y (góc θ)
  - Áp dụng biến đổi theo tỷ lệ S1, S2
  - Xoay trả lại hướng ban đầu
- Ma trận biến đổi

$$\begin{bmatrix} S1.\cos^2\theta + S2.\sin^2\theta & (-S1+S2)\sin\theta\cos\theta & 0 \\ (-S1+S2)\sin\theta\cos\theta & S1.\sin^2\theta + S2.\cos^2\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Kết hợp các phép quay

Pheùp quay quanh goác toaï ñoä

Goïi T_1 vaø T_2 laø 2 pheùp quay quanh goác toaï ñoä sao cho: T_1: $P(x,y) \rightarrow P'$

T_2: $P' \rightarrow Q(x',y')$

Ma traän bieán ñoåi töø P sang Q laø:

$$M = M_1 M_2 = \begin{pmatrix} \cos\alpha_1 & \sin\alpha_1 & 0 \\ -\sin\alpha_1 & \cos\alpha_1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \times \begin{pmatrix} \cos\alpha_2 & \sin\alpha_2 & 0 \\ -\sin\alpha_2 & \cos\alpha_2 & 0 \\ 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} \cos(\alpha_1+\alpha_2) & \sin(\alpha_1+\alpha_2) & 0 \\ -\sin(\alpha_1+\alpha_2) & \cos(\alpha_1+\alpha_2) & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

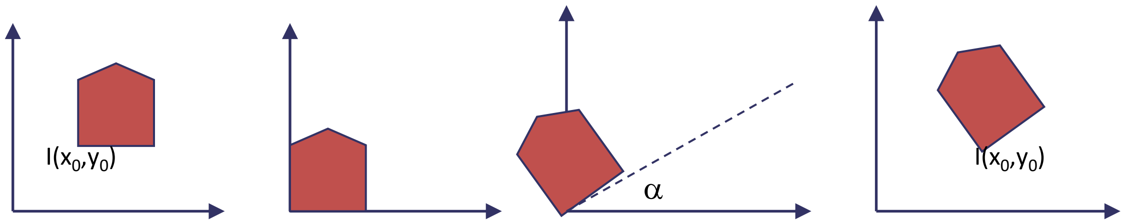
Vaäy Q=P.M hay

$$\begin{bmatrix} x' & y' \end{bmatrix} = \begin{bmatrix} x & y \end{bmatrix} \times \begin{pmatrix} \cos(\alpha_1+\alpha_2) & \sin(\alpha_1+\alpha_2) & 0 \\ -\sin(\alpha_1+\alpha_2) & \cos(\alpha_1+\alpha_2) & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

Vaäy, keát hôïp hai/nhieàu pheùp quay quanh goác toaï ñoä laø 1 pheùp quay quanh goác toaï ñoä.

Kết hợp các phép quay

Pheùp quay coù taâm baát kyø

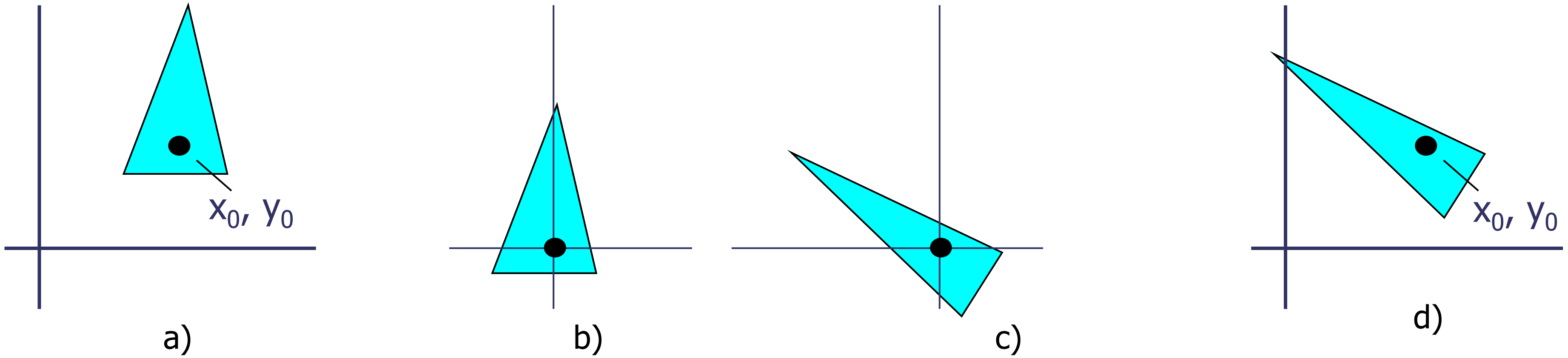


Pheùp quay quanh taâm I(x0 , y0) goùc α ñöôïc keát hôïp töø caùc pheùp bieán ñoåi cô sôû sau:

-Tònh tieán theo vector ($-x_0$,$-y_0$) ñeå ñöa taâm quay veà goác toaï ñoä (M_1)

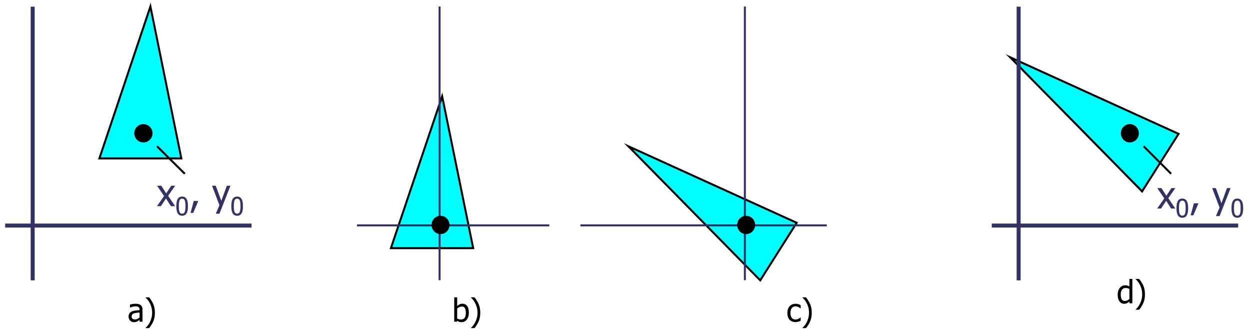
-Quay quanh goác toaï ñoä moät goùc α. (M_2)

-Tònh tieán theo vector (x_0 ,y_0) ñeå ñöa taâm quay veà vò trí ban ñaàu.(M_3)



Kết hợp các phép quay

Pheùp quay coù taâm baát kyø



Ta coù ma traän cuûa pheùp bieán ñoåi keát hôïp laø: M=M1.M2.M3

$$= \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -x_0 & -y_0 & 1 \end{pmatrix} \times \begin{pmatrix} \cos\alpha & \sin\alpha & 0 \\ -\sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{pmatrix} \times \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ x_0 & y_0 & 1 \end{pmatrix}$$

$$= \begin{pmatrix} \cos\alpha & \sin\alpha & 0 \\ -\sin\alpha & \cos\alpha & 0 \\ (1-\cos\alpha).x_0 + \sin\alpha.y_0 & -\sin\alpha.x_0 + (1-\cos\alpha).y_0 & 1 \end{pmatrix}$$

Ví duï1: Tìm ma traän bieán ñoåi trong heä toaï ñoä thuaàn nhaát cuûa pheùp tæ leä
vôùi Sx=Sy=2

a.Taâm tæ leä laø goác toaï ñoä

b.Taâm tæ leä laø ñieåm (5,2)

Aâp duïng ñeå tìm aûnh cuûa tam giaùc ABC vôùi A(0,0), B(1,1) C(5,2)

Ví du 2: Tìm ma traän bieán ñoåi trong heä toaï ñoä thuaàn nhaát cuûa pheùp laáy ñoái xöùng:

a.Qua ñöôøng thaúng y=x-1.

b.Qua ñieåm (1,2)

AÙp duïng tìm aûnh cuûa tam giaùc ABC vôùi A(0,0), B(0,2), C(-2,0)

Bài tập

1. Tìm ma trận biến đổi để đối tượng đối xứng qua y=x và y=-x.

2. Cho tam giác A(3, 1), B(1, 3), C(3,3). Xác định tọa độ mới của các đỉnh tam giác sau khi:

- Quay một góc 90^0 ngược chiều kim đồng hồ xung quanh điểm P(2, 2).
- Phóng to tam giác lên hai lần, giữ nguyên vị trí điểm C.

3. Tìm ma trận biến đổi trong phép đối xứng qua đường thẳng nằm nghiêng có độ nghiêng m và đi qua điểm (0, c).

SUMMARY

- Vẽ đối tượng đồ họa cơ sở
- Xén hình
- Tô màu
- Vẽ chữ và dựng Font
- Các phép biến đổi 2D

Nhân bản – Phụng sự – Khai phóng

Enjoy the Course…!