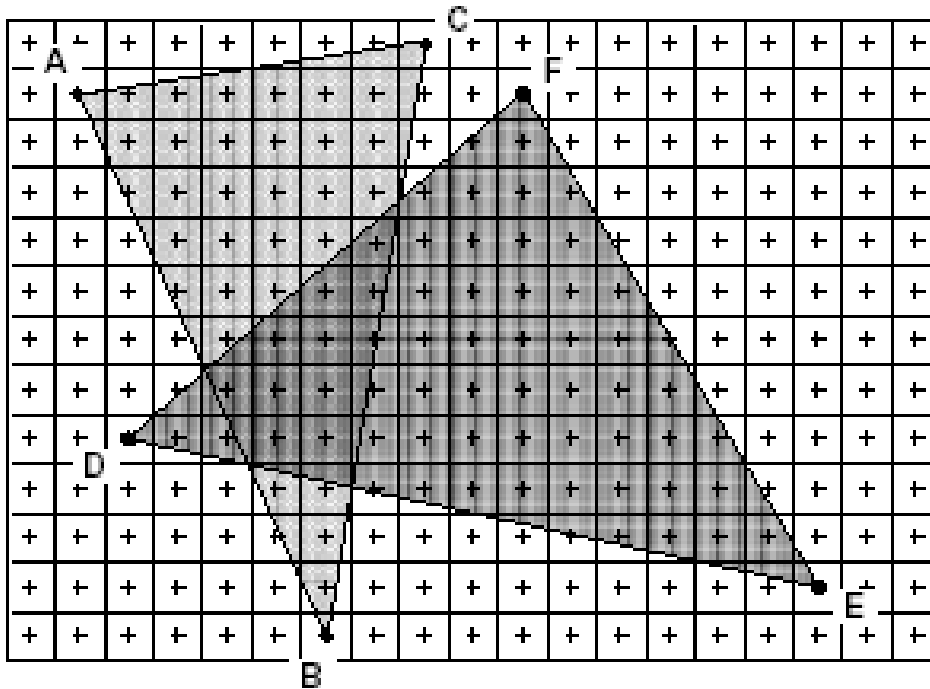


Rastérisation

Rastérisation

- ▶ Découpe la primitive 2D en pixels
- ▶ Interpole les valeurs connues aux sommets : couleur, profondeur,...



Modeling
Transformations

Illumination
(Shading)

Viewing Transformation
(Perspective / Orthographic)

Clipping

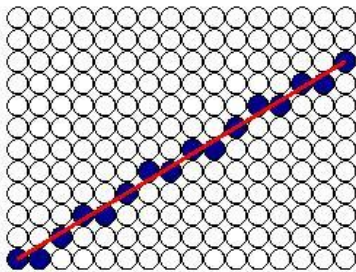
Projection
(to Screen Space)

Scan Conversion
(Rasterization)

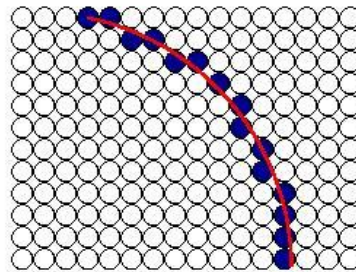
Visibility / Display

Primitives 2D

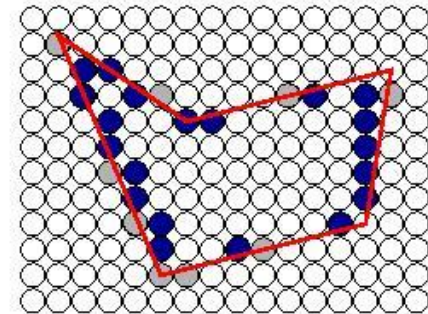
- ▶ Dans tous les cas il faut savoir afficher du 2D, ligne ou polygone
 - ⇒ il faut savoir quels pixels allumer pour un objet mathématique
 - ⇒ il faut savoir remplir un polygone



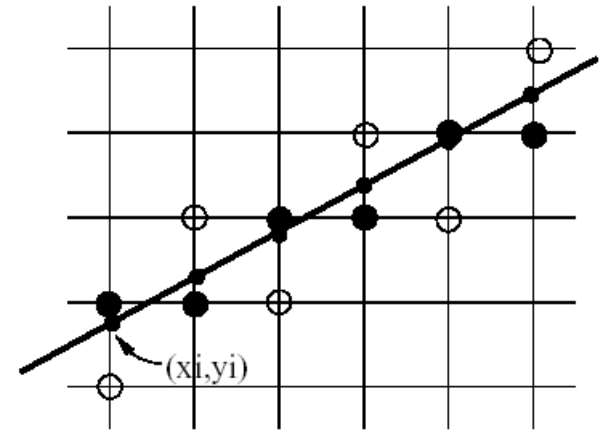
Droite



Cercle



Tracé de droites



▶ Méthode brute (naïve)

- Discrétisation de la primitive en n points
- Approximation au pixel le plus proche pour les n points
- Peu efficace et peu précis

▶ Méthode incrémentale

- La position du point suivant est choisie de façon à minimiser l'erreur d'approximation
- Méthode optimale pour le tracé de segments de droites

Algorithme de Bresenham (1965)

Premier octant :

Une droite est définie par l'équation

$$y = mx + B$$

On cherche le prochain pixel comme celui qui minimise l'erreur

$$e = (d2 - d1)/2$$

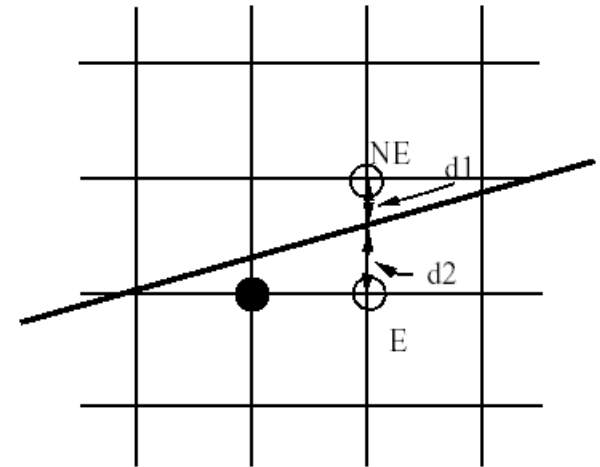
Au premier point (x_0+1, y_0+m)

$$e1 = (d2 - d1)/2 = -0.5;$$

Ensuite l'erreur se propage

$$\text{NE : } e = e + m - 1;$$

$$\text{E : } e = e + m.$$



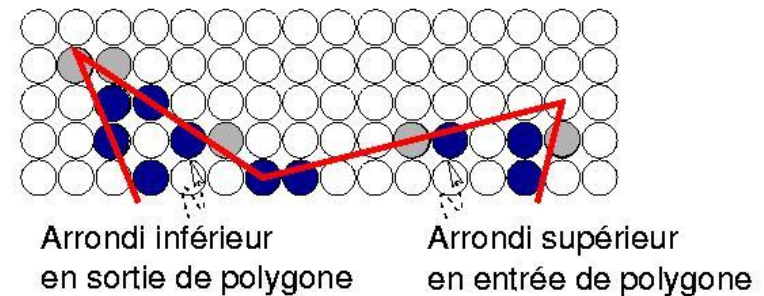
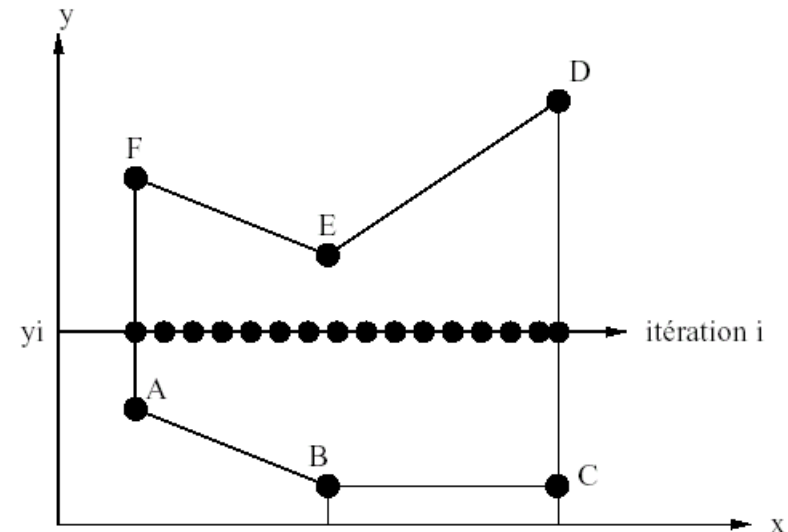
```
void Bresenham(x0, y0, x1, y1, valeur)
```

```
{ int x;  
  double m = y1 - y0 / x1 - x0 ;  
  double e = -0.5;  
  int y = y0;  
  for(x = x0; x <= x1 ; x++)  
  { e = e + m ;  
    AfficherPixel(x, y, valeur);  
    if (e >= 0)  
    { y = y + 1;  
      e = e - 1;  
    }  
  }  
}}
```

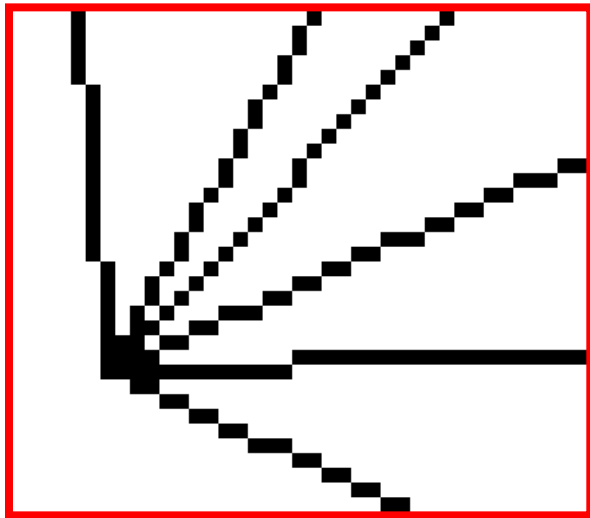
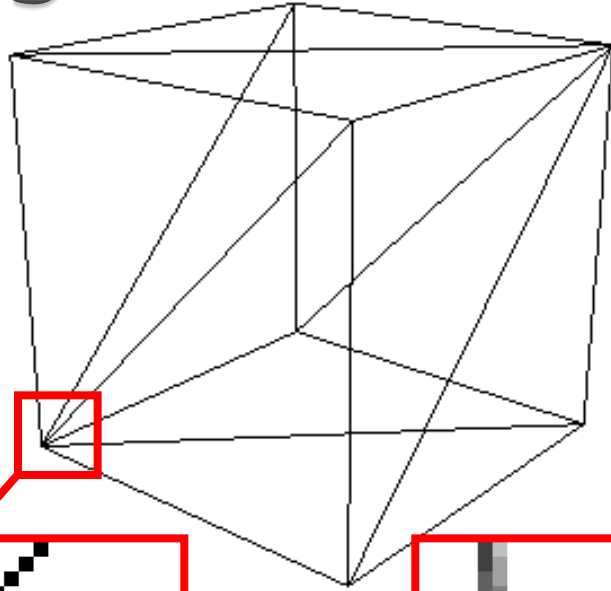
Remplissage de polygone

► Principe (scan-line)

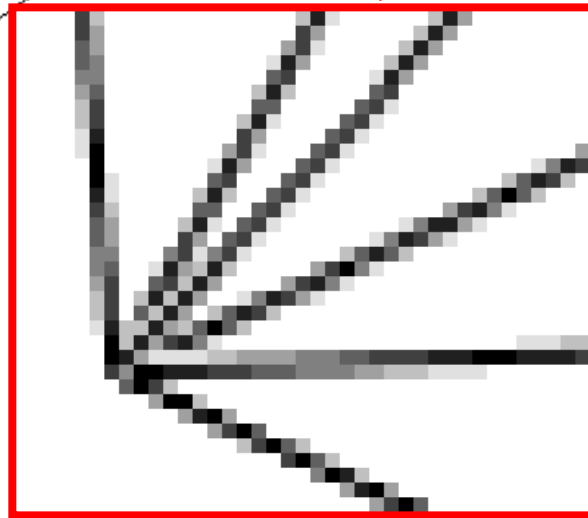
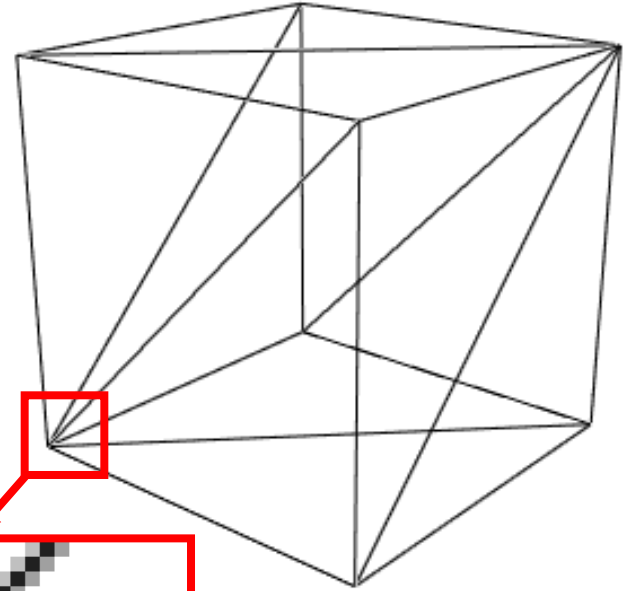
1. Remplissage par recherche des points d'intersection d'une ligne horizontale avec des contours (un nombre pair).
2. Une fois les points d'intersection obtenus, remplissage selon une règle de parité : incrémentation de la parité à chaque traversée de frontière et tracé si impair.
3. Gestion des conflits frontaliers : on prend les points intérieurs au polygone pour éviter les chevauchements.



Aliasing



aliased



antialiased