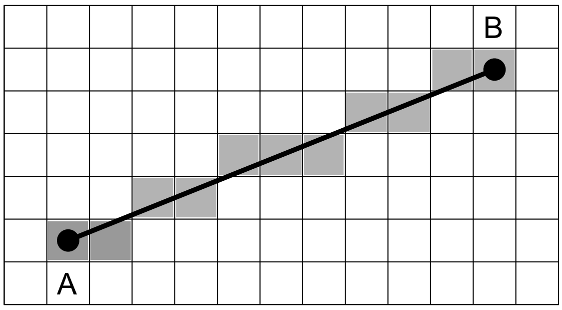
I. Algorithmes de tracé

Le traçage de primitives 2D (segment, cercle, polynômes, ...) est une des fonctionnalité de base du graphisme 2D. Ces primitives sont naturellement définies par des équations sur $\mathbb{R}$. Une image constituée de primitives données par leurs équations est une image *vectorielle*. Un écran d'ordinateur est quand à lui constitué de pixels : c'est une matrice de pixels. Un écran affiche donc des images *discrète* ou *matricielle*.

La *rastérisation* est un procédé qui consiste à convertir une image vectorielle en une image matricielle. Il répond donc à la question : ≪ Quels sont les pixels à allumer pour dessiner le segment passant par deux points A et B ? ≫



Dans ce chapitre on considère le plan muni d'un repère orthonormé $(O, \vec{i}, \vec{j})$. Ceci signifie notamment que les vecteurs $\vec{i}$ est $\vec{j}$ sont de norme 1 et sont orthogonaux entre eux. Chaque point M du plan est ainsi muni de deux coordonnées (X_M, Y_M) qui déterminent sa position : $\vec{OM} = x_M\vec{i} + y_M\vec{j}$. Les nombres x_M et y_M sont respectivement l'abscisse et l'ordonnée du point M. En écrivant $M = (x_M, y_M)$, on identifie le point M à ses coordonnées.

1 Cas d'un segment

On considère deux point distincts $A = (x_A, y_A)$ et $B = (x_B, y_B)$ du plan. Par ces deux points passent une seule droite, que l'on nommera D. Notons $dx = x_B - x_A$ et $dy = y_B - y_A$ les différences entre les coordonnées de A et B. On suppose que dx est non nul, ce qui revient à demander que la droite D ne soit pas verticale. On note alors $m = \dfrac{dy}{dx}$ le nombre que l'on appelle *pente* de la droite D.

Définition 1.1. On appelle équation cartésienne d'une droite Δ une fonction bilinéaire F_Δ de $\mathbb{R} \times \mathbb{R}$ dans $\mathbb{R}$ vérifiant

$$F_\Delta(x, y) = 0 \Leftrightarrow (x, y) \in \Delta$$

Evidement, si l'application F_Δ de $\mathbb{R} \times \mathbb{R}$ dans $\mathbb{R}$ est une equation cartésienne de la droite Δ alors l'application de $\mathbb{R} \times \mathbb{R}$ dans $\mathbb{R}$ qui à (x, y) associe $\lambda \cdot F_\Delta(x, y)$ en est une autre quelque siot le réel $\lambda \in \mathbb{R} \setminus \{0\}$.

Exercice 1.2. Donner en fonction de x_A, y_A, dx et dy une équation cartésienne de la droite $D = (AB)$.

Solution. On pose $F_D(x, y) = dy(x - x_A) - dx(y - y_A)$. La droite (AB) à pour équation $y = mx + p$ avec $m = \dfrac{dy}{dx}$ et $p = y_A - m\,x_A$. On a alors

$$\begin{aligned} F(x, y) = 0a &\Leftrightarrow dy(x - x_A) - dx(y - y_A) = 0 \Leftrightarrow dy(x - x_A) = dx(y - y_A) \\ &\Leftrightarrow m(x - x_A) = y - y_A \Leftrightarrow y = mx - mx_A + y_A = mx + p \end{aligned}$$

Exercice 1.3. Soit $P = (x_P, y_P)$ un point de la droite $D = (AB)$ ayant pour abscisse

$$x_P = x_A + \Delta x$$

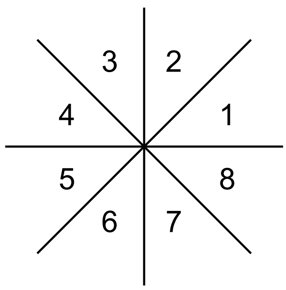
Donner une expression de y_P en fonction de y_A, m et Δx.

Solution. Le point P est sur D, on a donc $F_D(x_P, y_P) = 0$ et donc

$$\begin{aligned} F_D(x_P, y_P) = 0 &\Leftrightarrow dy(x_P - x_A) - dx(y_P - y_A) = 0 \\ &\Leftrightarrow dy(x_A + \Delta x - x_A) - dx(y_P - y_A) = 0 \\ &\Leftrightarrow dy \cdot \Delta x = dx(y_P - y_A) \\ &\Leftrightarrow m \cdot \Delta x = y_P - y_A \\ &\Leftrightarrow y_P = y_A + m \cdot \Delta x \end{aligned}$$

L'augmentation de y le long de la droite D est proportionnel à l'augmentation de x, le coefficient de proportionnalité étant égal à la pente m de D.

Pour le reste de ce chapitre, nous divisons le plan en huit octant selons la figure suivante :



Il faut imaginer le point A au centre de cette « étoile » et le point B dans l'une des section numéroté. On dit que le segment $[A, B]$ est dans l'octant numéro i si le point B se trouve dans la section correspondante.

Pour simplifier le problème, nous traitons tout d'abord le tracer d'un segment appartenant au premier octant. La droite portant un tel segment à une pente m comprise entre 0 et 1 et sa rastérisation nécessite d'allumer exactement un pixel par colonne.

Pour la suite, on suppose que nous avons à notre disposition :

- une structure `Point` possédant deux attributs `x` et `y` de type `Entier`.
- une fonction `void draw(Point P)` qui affiche le `Point P` à l'écran.

1.1 Algorithme naif

On note m la pente réelle de la droite D passant par les points A et B. L'algorithme naïf consiste à calculer, pour chaque valeur entière de l'abscisse x comprise entre x et $x_0 + dx$, la valeur de y correspondante par une opération d'arrondi.

Exercice 1.4. Ecrire en pseudo-langage une fonction

```
void draw_segment_v1(Point A, Point B)
```

tracant le segment reliant les points A et B.

Cette méthode comporte de gros défauts : pour chaque pixel à dessiner, on effectue une multiplication en virgule flottante, une opération d'arrondi et une addition. L'opérations la plus couteuse est la multiplication.

Exercice 1.5. Ecrire en pseudo-langage une fonction

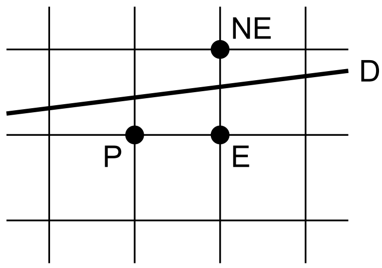
```
void draw_segment_v2(Point A, Point B)
```

inspirée de la fonction `void draw_segment_v1` mais ne nécessitant pas de multiplication en virgule flottante pour chaque pixel dessiner.

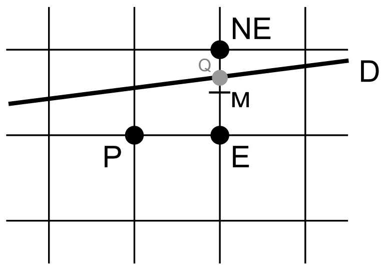
Avec cet algorithme, on a supprimé le plus couteux : la multiplication en virgule flottante à chaque étape. Cependant, il reste une addition en virgule flottante et un calcul d'arrondi. L'objet de la prochaine approche est de supprimer le calcul d'arrondi et l'utilisation de variables en virgule flottante.

1.2 Algortihme du point milieu

Cet algorithme a été inventé par Bresenham en mai 1962 travaillant alors pour IBM. L'idée de l'algorithme est la suivante. Etant donné que l'on travaille dans le premier octant et que la pente de la droite est comprise entre 0 et 1, le pixel à allumer après un pixel P (en allant de la gauche vers la droite) sera forcément l'un des deux suivants (parmi les huis pixels adjacents) : soit il s'agira de celui situé à droite (que l'on appellera point Est, ou E), soit il s'agira de celui situé en haut à droite (que l'on appellera Nord-Est, ou NE).



Notons $P_0, ..., P_n$ les différents points (ou pixels) à dessiner (ou allumer) pour tracer le segment $[AB]$. Notons (x_i, y_i) les coordonnées du point P_i. Supposons que nous ayons déjà décidé des points $P_0, ..., P_i$. Voyons comment déssider si P_{i+1} sera le point Est (que l'on note E) ou le point Nord-Est (que l'on note NE) de P_i. Les coordonnées de E et NE sont respectivement $(x_i + 1, y_i)$ et $(x_i + 1, y_i + 1)$. Soit Q le point d'intersection entre la droite réel $D = (AB)$ et la droite verticale d'équation $x = x_i + 1$. Le point Q se trouve sur le segment vertical reliant les points E et NE. Soit M le milieu de ce segment vertical. Les coordonnées de M sont $(x_i + 1, y_i + \frac{1}{2})$. Si Q se trouve sur le segment $[E, M]$, le point P_{i+1} sera le point E sinon ce sera le point NE.



Exercice 1.6. Soit N un point de coordonnées (x_N, y_N) qui se trouve au dessus de la droite D. Que pouvez-vous dire de $F_D(x_N, y_N)$. Et s'il est en dessous ?

Solution. Soit N' le point de coordonnées (x_N, y'_N) qui se trouve sur D. On a $y_N > y'_N$, d'où

$$\begin{aligned} F_D(x_N, y_N) &= dy(x_N - x_A) - dx(y_N - y_A) \\ &= dy(x_N - x_A) - dx(y'_N + (y_N - y'_N) - y_A) \\ &= dy(x_N - x_A) - dx(y'_N - y_A) - dx(y_N - y'_N) \\ &= -dx(y_N - y'_N) < 0 \end{aligned}$$

De même si N est en dessous de D, on a $F_D(x_N, y_N) > 0$.

On va ainsi créer une variable de décision δ_i qui sera du même signe que la valeur de F_D au point M. La valeur de δ_i indiquera lequel des deux points E ou NE il faut choisir.

Exercice 1.7. Donner une expression de δ_i qui soit du même signe que la valeur de F_D au point M mais appartenant à $\mathbb{Z}$.

Solution. Par définition de F_D et de M, on a

$$\begin{aligned} F_D(x_M, y_M) &= F_D\left(x_i + 1, y_i + \frac{1}{2}\right) \\ &= dy(x_i + 1 - x_A) - dx\left(y_i + \frac{1}{2} - y_A\right) \end{aligned}$$

Seule le signe nous intéresse, on peut tout multiplier par 2 et obtenir

$$\begin{aligned} \delta_i &= 2F_D(x_M, y_M) \\ &= 2dy(x_i + 1 - x_A) - 2dx\left(y_i + \frac{1}{2} - y_A\right) \\ &= 2dy(x_i + 1 - x_A) - dx\left(2y_i + 1 - 2y_A\right) \end{aligned}$$

Le calcul de δ_i est lui aussi couteux. Il est intéressant de le calculer de manière incrémentale.

Exercice 1.8. En fonction du choix (E ou NE) pour P_{i+1} donner une formule liant δ_i et δ_{i+1}.

Solution. Pour $P_{i+1} = E$, on a $x_{i+1} = x_i + 1$ et $y_{i+1} = y_i$ d'où

$$\begin{aligned} \delta_{i+1} &= 2F_D\left(x_{i+1} + 1, y_{i+1} + \frac{1}{2}\right) \\ &= 2F_D\left(x_i + 2, y_i + \frac{1}{2}\right) \\ &= 2\left(dy(x_i + 2 - x_A) - dx\left(y_i + \frac{1}{2}\right)\right) \\ &= 2dy + 2\left(dy(x_i + 1 - x_A) - dx\left(y_i + \frac{1}{2}\right)\right) \\ &= 2dy + \delta_i \end{aligned}$$

Pour $P_{i+1} = NE$, on a $x_{i+1} = x_i + 1$ et $y_{i+1} = y_i + 1$ d'où

$$\begin{aligned}
\delta_{i+1} &= 2F_D(x_{i+1} + 1, y_{i+1} + \frac{1}{2}) \\
&= 2F_D(x_i + 2, y_i + 1 + \frac{1}{2}) \\
&= 2\left(dy(x_i + 2 - x_A) - dx\left(y_i + 1 + \frac{1}{2}\right)\right) \\
&= 2dy - 2dx + 2\left(dy(x_i + 1 - x_A) - dx\left(y_i + \frac{1}{2}\right)\right) \\
&= 2(dy - dx) + \delta_i
\end{aligned}$$

Exercice 1.9. Ecrire en pseudo-langage une fonction

```
void draw_segment_v3(Point A, Point B)
```

tracent le segment allant de A à B à l'aide de l'algorithme du point milieu, si ce segment est dans le premier octant.

Exercice 1.10.

1. Généraliser l'algorithme du point milieu à n'importe quel segment du plan

2. Ecrire en pseudo-langage une fonction

```
void draw_segment(Point A, Point B)
```

tracant le segment allant de A à B et utilisant la méthode du point milieu.

2 Cercles

La méthode de tracer d'un cercle s'effectue suivant les mêmes principes décrit précédemment, à savoir :

- procédure incrémentale,
- découpage de l'espace en ocants.

Exercice 2.1.

1. Donner l'équation cartésienne F_C du cercle C de centre $A = (x_A, y_A)$ et de rayon R.

2. Quel est le signe de F_C pour un point à l'intérieur du cercle ?

Cette fois-ci, nous traiterons dans un premier temps les pixels du deuxième octant. De même que pour le segment , on note P_i les différents pixels à allumer dans le sens trigonométrique et (x_i, y_i) les coordonnées de P_i.

Exercice 2.2. Parmis les pixels adjacent à P_i que sont les choix possibles pour P_{i+1}.

Comme pour le segment nous allons utiliser une variable de décision δ_i qui sera du même signe que la valeur de F_C évalué au point M milieu des points possibles pour P_{i+1}.

Exercice 2.3. Donner une expression entière pour δ_i.

Exercice 2.4. En fonction du choix pour P_{i+1} donner une formule liant δ_i et δ_{i+1}.

Exercice 2.5. Ecrire en pseudo-langage la fonction

```
void draw_circle_v1(Point A, Entier R)
```

traçant la portion du cercle de centre A et de rayon R se trouvant dans le deuxième octant.

Exercice 2.6.

1. Généraliser l'algorithme au traçage du cercle tout entier.
2. Ecrire en pseudo-lanage la fonction

   ```
   void draw_circle(Point A, Entier R)
   ```

   traçant la portion du cercle de centre A et de rayon R.