

Prénom _____
NOM _____

20

Interrogation de Java

(Date)

Remarques :

- ↪ vous travaillerez dans une répertoire interro2¹,
- ↪ vous pouvez utiliser **toutes les notes** que vous voulez ainsi que les programmes qui se trouvent dans votre répertoire,
- ↪ l'interro se fait sur linux1,
- ↪ vous disposez de 2 heures

LE PROBLÈME. Nous allons écrire une classe **Droite** permettant de *gérer* les droites. Nous nous contenterons d'afficher l'équation d'une droite, de calculer l'intersection de deux droites, de rechercher l'équation d'une droite perpendiculaire passant par un point.

L'intersection de deux droites peut être un point, une droite ou encore l'ensemble vide. Nous choisirons donc **Object** comme type de retour afin de pouvoir retourner au choix un **String**, un **Point** ou encore une **Droite**.

Afin de gérer les *points*, vous écrirez une classe **Point** permettant de créer et d'afficher un point. Une classe **Test** nous permettra de tester tout ça.

La classe Point

Un point appartenant au plan ($\in \mathbb{R}^2$) se compose d'une abscisse et d'une ordonnée. Un objet **Point** contiendra donc deux attributs de type **double**

```
private double x;    // Abscisse
private double y;    // Ordonnee
```

Cette classe proposera les services suivants :

- ↪ Un constructeur, recevant l'abscisse et l'ordonnée en paramètre,
- ↪ Une méthode **public String toString ()**, retournant une représentation d'un point, du style (x,y) .
- ↪ Des assesseurs si vous le jugez opportun.

La classe Droite

L'équation générale d'une droite D est la suivante :

$$D \equiv ax + by + c = 0 \text{ où } (x, y) \in \mathbb{R}^2$$

Un objet **Droite** comprendra 3 attributs de type **double** représentant les coefficients a , b et c . La classe proposera les méthodes suivantes :

- ↪ Un constructeur, recevant les trois coefficients en paramètres
- ↪ Une méthode **public String toString()**, retournant une représentation d'une droite, du style $ax+by+c=0$.
- ↪ Des assesseurs si vous le jugez opportun

¹Remarquez l'absence de majuscule au nom de répertoire

↪ Une méthode public `Object inter(Droite droite)` calculant l'intersection de deux droites. Cette méthode retourne un `Object`. Ceci permet de gérer les différents cas d'intersection, vous pouvez retourner un `String`, un `Point` ou encore une `Droite` suivant le cas.

Pour rappel, l'intersection de deux droites se calcule comme suit :

$$\begin{cases} D_1 \equiv a_1x + b_1y + c_1 = 0 \\ D_2 \equiv a_2x + b_2y + c_2 = 0 \end{cases}$$

On calcule

$$\Delta = \begin{vmatrix} a_1 & a_2 \\ b_1 & b_2 \end{vmatrix} = a_1b_2 - b_1a_2$$

L'ensemble des solutions S est donné par

$$S = \begin{cases} D_1 \equiv D_2 & \text{si } \Delta = 0 \text{ et } \begin{pmatrix} a_1 \\ b_1 \\ c_1 \end{pmatrix} = k \begin{pmatrix} a_2 \\ b_2 \\ c_2 \end{pmatrix} \text{ où } k \in \mathfrak{R} \\ \emptyset & \text{si } \Delta = 0 \text{ et sinon} \\ (x, y) = \left(\frac{\begin{vmatrix} -c_1 & b_1 \\ -c_2 & b_2 \end{vmatrix}}{\Delta}, \frac{\begin{vmatrix} a_1 & -c_1 \\ a_2 & -c_2 \end{vmatrix}}{\Delta} \right) & \text{si } \Delta \neq 0 \\ = \left(\frac{-c_1b_2 + c_2b_1}{\Delta}, \frac{-a_1c_2 + a_2c_1}{\Delta} \right) & \end{cases}$$

↪ Une méthode public `Droite perpendicular(Point p)` qui retourne l'équation de P tq

$$\begin{cases} P \perp D \\ \ni p(p_1, p_2) \end{cases}$$

Pour rappel, l'équation de la droite P perpendiculaire à $D \equiv ax + by + c = 0$ et $\ni p(p_1, p_2)$ se trouve comme suit, il faut chercher la valeur des coefficients a_\perp , b_\perp et c_\perp de P

$$\begin{aligned} a_\perp &= -b \\ b_\perp &= a \\ c_\perp &= -a_\perp p_1 - b_\perp p_2 \end{aligned}$$

La classe Test

Ecrire une classe **Test** permettant de tester les classes **Droite** et **Point**.

Remarque La répartition des points est la suivante ; 3 points pour la classe **Point**, 2 points pour la classe **Test**, les 15 points restant sont attribués à la classe **Droite** dont 5 pour la méthode `inter` et 5 pour la méthode `perpendicular`.