

Bases de la Programmation Orientée Objet - Java

TP 03

Jérôme Buisine, Rémi Cozot, Rémi Synave et Bruno Warin

Travail préliminaire

Dans votre répertoire `java`, créez un nouveau répertoire `Tp3`. Vous y placerez tous les fichiers utilisés pour la réalisation du TP. Vous avez 2 séances pour faire ce TP.

Conseil : N'oubliez pas de compiler régulièrement !

La classe `Point`

Cette classe permet de représenter un point du plan sous la forme (x, y) . Les coordonnées du point seront stockées par des entiers. Donnez le code de la classe `Point` contenant :

- Deux attributs privés entiers x et y .
- Un constructeur par défaut qui initialisera les coordonnées à 0.
- Un constructeur prenant deux entiers en paramètre permettant l'initialisation des attributs.
- Les méthodes `getX` et `getY` permettant l'accès aux attributs.
- Les méthodes `setX` et `setY` permettant de fixer une nouvelle valeur aux coordonnées.
- La méthode dont la signature est `public String toString()` retournant une chaîne de caractères sous la forme : (x, y) .

Vous pouvez utiliser la classe `PointTest` fournie pour tester le bon fonctionnement de votre classe `Point`.

La classe `Segment`

Un segment est définie par deux objets de type `Point` : `a` et `b`. Donnez le code de la classe `Segment` contenant :

- Deux attributs privés, `a` et `b` de type `Point`.
- Un constructeur par défaut qui crée un segment allant du point $(0, 0)$ au point $(1, 0)$.
- Un constructeur prenant deux objets de type `Point` en paramètre.
- Un constructeur prenant 4 entiers en paramètre `int x1, int y1, int x2, int y2`.
- Une méthode `longueur` retournant la taille du segment.

- La méthode `public String toString()` retournant une chaîne de caractères sous la forme : $(x1, y1) - (x2, y2)$.

Vous pouvez utiliser la classe `SegmentTest` fournie pour tester le bon fonctionnement de votre classe `Segment`.

La classe Carre

Un `Carre` est défini par 4 objets de type `Segment`. Donnez le code de la classe `Carre` contenant :

- Quatre attributs de type `Segment`.
- Un constructeur par défaut créant un carré ayant pour origine $(0,0)$ et de côté 1.
- Un constructeur prenant en paramètre un objet de type `Point` (qui sera le coin bas gauche du carré) et un entier donnant la taille du carré
- Une méthode `perimetre` retournant le périmètre du carré.
- Une méthode `public String toString()` retournant les caractéristiques du carré sous la forme suivante :
 $(x1, y1) - (x2, y2)$
 $(x2, y2) - (x3, y3)$
 $(x3, y3) - (x4, y4)$
 $(x4, y4) - (x1, y1)$
ou quelque chose d'approchant.

Vous pouvez utiliser la classe `CarreTest` fournie pour tester le bon fonctionnement de votre classe `Carre`.

Programme principal

Compilez l'ensemble des fichiers (y compris le fichier `Main.java` fourni). Exécutez et expliquez le résultat obtenu.

Donnez une solution à ce problème.

La classe `Carre` n'est pas optimale dans sa description. Proposez une autre implémentation basée sur seulement 2 attributs.