

La classe Polygone

Objectifs :

- Écrire complètement une classe Java ;
- Revoir (et appliquer !) les concepts objets ;
- Savoir écrire la documentation d'une classe ;
- Savoir tester une classe avec JUnit.

Exercice 1 : Écrire une classe Polygone

Un polygone est défini par une suite finie de points appelés sommets. On appelle ordre d'un polygone, son nombre de sommets. Si on note A_1, A_2, \dots, A_n les points constituant les sommets d'un polygone d'ordre n , le polygone est dessiné en traçant les segments $[A_1A_2]$, $[A_2A_3]$, ..., $[A_nA_1]$. La figure 1 présente trois polygones. Le polygone $ABCDEF$ est *concave*, RST est *convexe* et $IJKL$ est dit *croisé*.

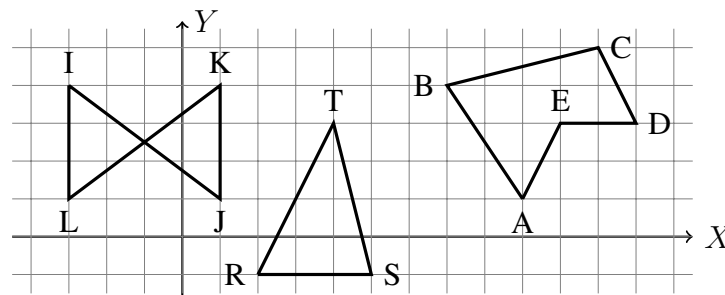


FIGURE 1 – Quelques exemples de polygones

Spécification

On demande d'écrire une classe Polygone respectant les exigences suivantes et passant avec succès les tests de la classe SujetPolygoneTest.

- E₁ On peut translater un polygone en précisant un déplacement suivant l'axe des X et un déplacement suivant l'axe des Y.
- E₂ On peut obtenir l'ordre d'un polygone.
- E₃ On peut obtenir un sommet d'un polygone en précisant son numéro. On suppose que le premier sommet a le numéro 0 et le dernier le numéro ordre - 1.
- E₄ On peut obtenir le périmètre d'un polygone. Le périmètre d'un polygone est la somme des longueurs de ses côtés.

- E₅ Le polygone possède une couleur. Cette couleur peut bien sûr être changée.
- E₆ Lorsqu'un polygone est affiché sur le terminal, ses sommets sont affichés du premier au dernier, entre chevrons. Par exemple, le polygone *RST* sera affiché comme suit :

```
<< (2.0, -1.0) (5.0, -1.0) (4.0, 3.0) >>
```
- E₇ On peut construire un polygone à partir de sa couleur et d'une suite de points qui deviendront ses sommets.
- E₈ On peut construire un polygone à partir de la seule suite de points. La couleur du polygone est alors le bleu.
- E₉ On ne doit pouvoir modifier l'état d'un polygone qu'au moyen d'une méthode de Polygone.
- E₁₀ Une méthode de classe `creerRectangle(Point, Point)` permet de créer un rectangle (un polygone d'ordre 4) dont les côtés sont parallèles aux axes X et Y. Les deux points fournis en paramètres sont deux sommets non adjacents du rectangle. La suite de ses sommets sera donnée dans le sens direct (sens inverse des aiguilles d'une montre) en commençant par le sommet de plus petites abscisse et ordonnée. Par exemple, les points *L* et *K* donneront le polygone *LJKI* (figure 1).
- E₁₁ On peut ajouter un nouveau sommet à un polygone en précisant son numéro (par exemple, 0 pour l'ajouter comme premier sommet, ordre pour l'ajouter comme dernier sommet). Si on ne précise pas le numéro, le nouveau sommet sera ajouté comme dernier sommet.

Contraintes

Les contraintes de réalisation à respecter *impérativement* sont les suivantes :

- C₁ Ce travail est un travail individuel. *Toute tricherie sera sanctionnée par la note minimale.*
- C₂ Les programmes doivent fonctionner sans modification sur les machines linux de l'N7.
- C₃ Les principes vus en cours, TD et TP doivent être appliqués lors de la réalisation de ce travail.
- C₄ Il est nécessaire de partir des classes fournies sur le dépôt SVN.
- C₅ Le rendu se fera via le dépôt SVN.
- C₆ Les classes fournies ne doivent pas être modifiées.
- C₇ Les lettres accentuées ne doivent pas être utilisées dans les identifiants.
- C₈ La classe Polygone doit être documentée avec javadoc.
- C₉ La classe Polygone doit réussir les tests de `SujetPolygoneTest`.
- C₁₀ `CreerRectangleTest`, test en JUnit 4, doit tester complètement la méthode `creerRectangle`.
- C₁₁ Tous les tests supplémentaires seront écrits dans `PolygoneTest`.
- C₁₂ On utilisera le mot-clé `assert` de Java pour exprimer toutes les préconditions des méthodes.
- C₁₃ Les sommets du polygone seront stockés dans un tableau (`Point []`).
- C₁₄ On autorise les polygones d'ordre 1 ou 2.
- C₁₅ Deux sommets adjacents pourront être égaux.

Livrables

Les livrables (documents à rendre) et les modalités de leur remise sont détaillés sur le site Web du module. Les livrables sont :

L₁ le fichier Polygone.java,

L₂ le fichier CreerRectangleTest.java,

L₃ le fichier PolygoneTest.java qui contient les tests supplémentaires faits sur la classe Polygone (en plus de ceux fournis dans SujetPolygoneTest et ceux écrits dans CreerRectangleTest).

L₄ le fichier LISEZ-MOI.txt complété (réponse aux questions et explications utiles à la compréhension des livrables rendus).

Attention : Les consignes et les dates à respecter sont disponibles sur le site Web du module Langage Objet.