

TD4 – développement dirigé par les tests

Exercice – retour sur le problème du triangle

Dans cet exercice, vous appliquerez la méthode du développement dirigé par les tests qui consiste à écrire les tests avant le code. Plus précisément, elle repose sur le cycle suivant :

1. Écrire un test,
2. L'exécuter et constater qu'il ne passe pas (barre rouge),
3. Écrire un code qui permet de passer le test (et rien que ce code),
4. Exécuter le test de nouveau et vérifier qu'il passe (barre verte),
5. Refactoriser le code si nécessaire,
6. Retour à l'étape 1 tant que le programme n'est pas totalement fonctionnel.

Dans un premier temps, nous proposons de représenter les longueurs des côtés d'un triangle en utilisant 3 variables d'instance réelles, comme illustré ci-dessous :

```
public class Triangle {
    public enum Type {EQUILATERAL, ISOCELE, SCALENE};
    public enum Angle {AIGU, DROIT, OBTUS};
    private double a, b, c;
    public Triangle(double a, double b, double c) throws TriangleException { }
    public Type getType() { return null; }
    public Angle getPlusGrandAngle() { return null; }
    public String toString() { return null; }
}
```

1. Écrivez le code de la classe `Triangle` en appliquant la méthode du développement dirigé par les tests.
2. Écrivez le programme principal demandant à l'utilisateur d'entrer trois nombres réels et affichant le type du triangle, ainsi que la nature de son plus grand angle, si ces trois nombres forment un triangle valide, sinon un message d'erreur approprié.
3. Refactorisez votre code de sorte que les longueurs des côtés du triangle soient désormais stockées dans un tableau trié dans l'ordre croissant.