

# TD 10 Java – Programmation Générique

## Cours

- 1 – Comparer les méthodes **addAll** de la classe **java.util.Collections** et de l'interface **Collection**.
- 2 – Application : déclarer une liste de figures et ajouter en une fois plusieurs points (on réutilisera les classes **Figure** et **Point** des premiers TDs).

## Exercice : graphes

Le but de cette séance est d'écrire en Java une structure générique de graphe avec des arêtes valuées. Les sommets du graphe sont des entiers de 1 à **n** et les valeurs des arêtes peuvent être quelconques à condition d'être comparables.

- 1 – Écrire la classe générique **AreteValuee** qui contient deux sommets (numéro entier) et une valeur d'un type générique qui implémente **java.lang.Comparable**
- 2 – Écrire la classe générique **GrapheValue** qui contient des sommets de 1 à **n** et un ensemble d'arêtes valuées. Le constructeur de la classe initialise le nombre de sommets à une valeur passée en paramètre et l'ensemble des arêtes à l'ensemble vide. Ajouter une méthode pour ajouter une arête valuée en vérifiant la cohérence sur les sommets.
- 3 – Ajouter un constructeur au graphe qui permet d'avoir un ensemble initial d'arêtes non vide à partir d'une collection quelconque. Écrire également une méthode d'ajout par lot.
- 4 – Ajouter un itérateur au graphe permettant de parcourir les valeurs des arêtes de la plus petite à la plus grande.
- 5 – Bloquer les modifications sur le graphe pendant l'utilisation d'un itérateur. Un itérateur n'est plus utilisé quand on a accédé au dernier élément.
- 6 – Tester vos classes avec une application créant un graphe de quelques sommets avec des arêtes valuées par des objets comparables (*Integer*, *String*, ...)
- 7 – Refaire ce projet en remplaçant le caractère **Comparable** des valeurs par un comparateur fourni à la construction des objets.

NB : regarder la documentation des classes **Comparable** et **Collections**.