

# Java avancé - TP 4 - Généricité et itérateurs

Romain Péchoux

## Exercice 1 :

L'objectif est d'implémenter une séquence générique, c'est-à-dire une séquence ordonnée d'éléments. Définir une classe générique **Seq** représentant une collection d'éléments (le choix le plus simple est de la faire dériver d'une classe liste de Java). Les éléments de la séquence pourront être n'importe quel type implémentant l'interface **Comparable**. Ajouter à cette classe, les méthodes suivantes :

- `public boolean occursAt(E s, int pos)` : qui retourne vrai si `s` se trouve à la position `pos`
- `public int find(E s)` : retourne la position de la première occurrence de `s` dans la séquence ou `-1` si `s` n'est pas dans la séquence
- `public int countOccurrences(E s)` : retourne le nombre d'occurrences de `s` dans la séquence
- `public void insert(int pos, Seq<E> sseq)` : insère la sous-séquence `sseq` à la position `pos` dans la séquence

Tester l'utilisation de ces méthodes :

```
Seq<String> s = new Seq<String>();
s.add("le");
s.add("petit");
s.add("chat");
s.add("est");
s.add("sur");
s.add("le");
s.add("fauteuil");
//s[le, petit, chat, est, dans, le, fauteuil]
//s.occursAt("chat",2) retourne vrai
//s.find("le") retourne 0
//s.countOccurrences("le") retourne 2
Seq<String> sous = new Seq<String>();
s.add("bien");
```

```
s.add("assis");  
s.insert(4,sous); //s[le, petit, chat, est, bien, assis, dans, le,  
fauteuil]
```

## Exercice 2 :

- Créer une classe **Entreprise**. Une entreprise possède un nom et plusieurs employés (utiliser la collection adéquate). Un employé se distingue par son nom. Il devra être possible d'ajouter et de supprimer des employés dans une entreprise.
- Utiliser une classe d'exception au cas où on veuille enlever un employé dans une entreprise où il ne travaille pas.
- Ecrire une méthode **toString** qui décrit une entreprise en affichant son nom et les noms de tous ses employés. A cet effet, modifier la classe de sorte à ce que l'on puisse faire un **foreach** sur une entreprise.
- Ajouter une méthode **iterator** retournant un itérateur d'employés.
- Pour tester, ajoutez quelques employés dans 2 entreprises et faites afficher les noms des entreprises et de leurs employés. Utiliser la méthode **iterator** pour calculer le salaire total de chacune des 2 entreprises.