

## TRAVAUX PRATIQUES Structures de données

### TP 9 : Mise en œuvre d'une liste et d'itérateurs

#### Objectif pédagogique :

Séparation de la notion de parcours et de la notion de conteneur de données. Réalisation d'une première mise en œuvre.

#### Prérequis :

Travaux pratiques et cours d'initiation à la POO, TDs de SDD

#### Sujet :

Nous allons réaliser la mise en œuvre de la liste dont l'interface (ou du moins une interface similaire) a été vue en TD. Pour le TP, on se référera exclusivement aux interfaces fournies.

Le conteneur (`SimpleListImpl`) qui implémente l'interface `SimpleList` et l'itérateur

`SimpleListIteratorImpl` qui implémente l'interface `SimpleListIterator` sont des classes

différentes. La création d'un itérateur ne peut se faire que grâce à une méthode de la liste comme vu en TD et il est d'autre part nécessaire que les itérateurs aient seul accès aux structures internes de la liste.

Pour ce faire, la classe `SimpleListIteratorImpl` sera une classe interne (InnerClass).

`SimpleListImpl`. Pour déclarer une telle classe, il suffit d'écrire sa définition à l'intérieur de la classe `SimpleListImpl`.

#### **Q0) Réaliser la mise en œuvre de la liste et de l'itérateur**

Pour cette mise en œuvre, et afin de varier les plaisirs, vous utiliserez un tableau pour gérer les éléments. On utilisera une gestion circulaire du tableau pour plus de facilité.

Attention, il a été choisi de mettre cette structure de données dans un package Java nommé `sdd`.

#### **Q1) Vérifiez son fonctionnement**

Vous testerez cette liste en écrivant quelques tests de fonctionnement.

#### **Q2) Gestion du débordement (facultative)**

Une question plus difficile, pour ceux qui ont du temps.

Imaginez une solution permettant de gérer une liste plus grande que la taille maximale d'une liste (`MAX_SIZE`) utilisant la mise en œuvre actuelle.

L'idée est de gérer une grande liste par une liste de listes.

On appellera `ExtendedListImpl` cette mise en œuvre qui devra implémenter l'interface `Liste` et qui utilisera en interne une liste `SimpleListImpl` qui gèrera des sous-listes.

Commencez par la mise en œuvre du parcours dans un nouvel itérateur (`ExtendedListIteratorImpl`) qui implémente `SimpleListIterator`, l'accès à l'élément courant et la modification de l'élément courant. Les fonctions avancées comme ôter et ajouter sont les plus difficiles à réaliser.

#### **Remarques**

Ce document et les interfaces Java se trouvent sous `/home/clyde/dl/insa3/commun/java-sdd/tp9`. Les interfaces sont dans le sous-répertoire `sdd`.