

## INTRODUCTION À LA SYNTAXE DU LANGAGE JAVA

### Sujet 1 - Fonctions et tableaux

On souhaite écrire en langage Java une bibliothèque de fonctions permettant d'assurer un certain nombre de traitements sur un ensemble de valeurs réelles représenté par une structure de données de type tableau. On suppose que le nombre de valeurs à traiter ne dépassera jamais 100, quel que soit le contexte d'utilisation<sup>1</sup>. Cet exercice a deux objectifs :

1. Tout d'abord de vous remémorer les principes de base de l'algorithmique (boucles, tests, fonctions,...) en programmant quelques algorithmes élémentaires sur des tableaux.
2. Ensuite, il s'agit bien évidemment de s'habituer à la syntaxe du langage Java et de s'appropriier les outils de votre environnement de programmation (système Linux, ligne de commandes, éditeur de texte, compilateur Java, JVM,...).

★ ★ ★

La définition de ces fonctions a déjà été réalisée (phase de conception). Pour chaque fonction il est spécifié :

- son nom (et vous devez respecter les majuscules/minuscules),
- l'identificateur de chaque paramètre formel,
- la fonctionnalité globale et la sémantique des paramètres

Il vous est formellement interdit de modifier la signature de ces fonctions (ex : ajouter des paramètres) ou d'utiliser des variables globales à vos programmes. Toujours dans le même ordre d'idée, vos fonctions ne doivent faire aucune lecture ou écriture de données, sauf si ceci est explicitement mentionné dans la description de la fonction.

Bien évidemment, il vous faudra également écrire un programme de test pour vérifier le bon fonctionnement de ces algorithmes. Ce n'est pas parce qu'un programme compile qu'il n'y a plus aucune erreur...

#### Description des fonctions :

1. Écrire la fonction `int ChargerTab(double[] Tab)` qui permet de stocker en mémoire `N` valeurs réelles dans le tableau `Tab`. Le nombre `N` doit être lu par cette fonction qui le renverra au programme appelant comme valeur de retour (résultat).
2. Écrire la fonction `void EditerTab(double[] Tab, int N)` qui permet d'afficher (proprement) les `N` valeurs réelles stockées dans le tableau `Tab`.
3. Écrire la fonction `double SommeTab(double[] Tab, int N)` qui retourne la somme des `N` valeurs réelles stockées dans le tableau `Tab`.
4. Écrire la fonction `double MoyenneTab(double[] Tab, int N)` qui retourne la moyenne des `N` valeurs réelles stockées dans le tableau `Tab`. Cette fonction devra impérativement utiliser la fonction `SommeTab` écrite précédemment.
5. Écrire les fonctions `double MaxTab(double[] Tab, int N)` et `double MinTab(double[] Tab, int N)` qui retournent respectivement la valeur maximale et la valeur minimale des `N` valeurs réelles stockées dans le tableau `Tab`.
6. Écrire la fonction `int SupValTab(double[] Tab, int N, double Val)` qui retourne le nombre de valeurs du tableau `Tab` qui sont supérieures à la valeur `Val` passée en paramètre.

---

1. Rappel : la taille d'un tableau est fixée lors de son initialisation, et cette taille ne peut plus varier par la suite, même si le tableau est plein  $\Rightarrow$  nous sommes donc obligés de **sur-dimensionner** nos tableaux.