



ASD – 2023/2024

TP2 : Itération et Récursivité

Exercice 1

Avec un tableau **T** de **N** entiers, comme Représentation Interne (R.I) d'une suite de nombre, translatez votre algorithme (discuté en TD) en un programme Java qui permet de supprimer une valeur **V** de **T** en utilisant les fonctions et procédures suivantes :

1. une méthode `int[] Lecture(int[] T)` pour lire les valeurs du tableau, à partir du clavier.
2. une méthode récursive `int Recherche(?, ?, int V)` qui retourne la position de **V** si elle existe dans **T** sinon retourne -1.
3. une méthode `int[] Suppression(int[] T, ?, ?)` qui appelle la méthode `Recherche(?, ?, V)`, puis supprime **V** de **T** (si elle existe).
4. une méthode `void Affichage(int[] T)` à appeler avant et après la suppression de **V**.

Exercice 2

1. Proposez un algorithme, puis le programme Java correspondant, qui détermine les **M** premiers nombres premiers, utilisez une fonction *itérative* `boolean Premier (int N)`.
2. Pouvez-vous optimiser (au maximum) la fonction *itérative* `boolean Premier (int N)` ?
3. Remplacez votre fonction *itérative* `boolean Premier (int N)` par la fonction *récursive* `int NbreDiviseurs(int N, int ?)` (traitée à l'exo 3, TD1).

Questions Bonus

1. Proposez une fonction *récursive* `float Maximum (float[] T, int N)` qui détermine le maximum d'un tableau **T** de **N** réels.
2. Ecrire une fonction **Main en Java** qui appelle une méthode `int[] Lecture (int[] T)` pour lire les valeurs de **T** puis Affiche son maximum en faisant appel à la fonction *récursive* `float Maximum (float[] T, int N)`.

Bon travail de développement modulaire et de raisonnement récursif.