

INTRODUCTION À LA SYNTAXE DU LANGAGE JAVA

Sujet 2 - Algorithmes de tri de tableaux

L'objectif de ce TD est d'étudier diverses méthodes permettant de trier, par ordre croissant, les valeurs d'un tableau.

1 Tri par sélection (ou par recherches successives de minimum)

Le principe de cette méthode très intuitive consiste à :

- chercher le minimum dans un sous-tableau (au départ le tableau complet contenant les N valeurs non ordonnées)
- permuter ce minimum avec le premier élément du sous-tableau
- puis itérer ce traitement sur un nouveau sous-tableau de $N-1$ éléments (on ne tient plus compte du premier élément qui est maintenant à sa place)

1. Écrire la fonction `void Permut(double[] Tab, int A, int B)` qui permute les valeurs réelles contenues dans le tableau `Tab` aux indices `A` et `B`.
2. En s'inspirant de la fonction `double MinTab(double[] Tab, int N)` écrite dans le TD précédent, écrire maintenant la fonction `int RangMinTab(double[] Tab, int N, int P)` qui retourne cette fois **le rang** (et non la valeur) de la plus petite des valeurs réelles contenues dans le tableau `Tab` **à partir du rang** `P`.
3. Écrire la fonction `void TriSel(double[] Tab, int N)` qui permet de trier par ordre croissant, suivant la méthode énoncée, les N valeurs réelles contenues dans le tableau `Tab`.
4. Écrire un programme de test. Vous pouvez pour cela réutiliser les fonctions `ChargerTab` et `EditerTab` écrites lors du TD précédent.

2 Tri à bulles (ou par propagation)

Le principe de cette méthode consiste à comparer les couples de valeurs successives `Tab[i]` et `Tab[i+1]` pour i variant de 0 à $N-2$, et à les permuter si elles sont mal ordonnées. L'algorithme s'arrête lorsque l'on constate qu'aucune permutation n'a été effectuée lors du dernier "survol" du tableau.

1. Écrire la fonction `void TriBul(double[] Tab, int N)` qui permet de trier, suivant la méthode énoncée, les N valeurs réelles contenues dans le tableau `Tab`. La fonction `void Permut(double[] Tab, int A, int B)` écrite précédemment pourra bien évidemment être réutilisée.
2. Écrire un programme de test.