



TD n°01

Exercice n°01 : « Élément dominant »

1. Proposez une méthode Java qui prend en entrée un tableau d'entiers T et qui retourne l'élément le plus dominant en T .
2. Utilisez la structure Java ***HashMap*** pour résoudre ce problème en effectuant un coût linéaire.

Exercice n°02 : « Sous-tableau maximal »

1. Proposez une méthode Java qui prend en entrée un tableau T contenant des entiers (positifs et/ou négatifs) et qui permet de trouver le sous-tableau de T dont la somme des valeurs est maximale.
2. La solution proposée est-elle optimale ? Autrement dit, peut-on passer d'un coût cubique à un coût quadratique ?
3. Proposez une version ***récursive*** de votre méthode.
4. En fonction de la taille du tableau T , notée $|T|$, trouvez le nombre maximal d'appels récurrents de votre méthode.

Exercice n°03 :

1. Proposez une méthode Java qui prend en entrée un tableau T composé d'entiers distincts et qui retourne le $K^{\text{ème}}$ plus petit élément de T .
2. Utilisez la structure Java ***TreeMap*** pour résoudre ce problème.
3. Laquelle des deux méthodes est plus efficace ?

Exercice n°04 :

On dispose d'un tableau T composé d'entiers appartenant à l'intervalle $[1, N]$.

1. Proposez une méthode Java qui permet de trier le tableau T avec un ***coût linéaire*** (ceci dit qu'aucune boucle imbriquée ne devra être utilisée). La structure ***HashMap*** peut être utilisée.
2. En déduire l'inconvénient de ce tri.