

J1MI2013: Algorithmes et Programmes: feuille 5**Tableaux(2)****Travaux dirigés**

Dans les exercices qui suivent on suppose que les tableaux ont une taille maximale fixée par une constante “assez grande” `TAILLE_MAX`. Les fonctions ont systématiquement parmi leurs paramètres, en plus d’un tableau `t`, un entier `n` représentant le nombre d’éléments du tableau `t`. On supposera toujours que $n \in [0, \text{TAILLE_MAX}]$.

Exercice 1. Écrire une fonction booléenne `bool estConstant(int t[], int n)` qui renvoie `true` si le tableau `t` passé en paramètre contient la même valeur dans toutes les cases entre les indices 0 et `n-1` inclus, et renvoie `false` sinon.

Exercice 2. Deux nombres sont opposés si leur somme est égale à 0.

1. Écrire une fonction `bool sontOpposes(int a, int b)` qui renvoie `true` si `a` et `b` sont opposés et renvoie `false` sinon.
2. Écrire une fonction `existeOppConsecutifs(int t[], int n)` qui renvoie `true` si le tableau `t` passé en paramètre contient deux nombres consécutifs opposés et renvoie `false` sinon.

Exercice complémentaire 1.

1. Écrire une fonction `existeOpp(int t[], int n)` qui renvoie `true` si `t` contient deux nombres, ayant des indices différents, opposés et renvoie `false` sinon.
2. Écrire une fonction `nombreOpp(int t[], int n)` qui renvoie le nombre de paires d’indices (i, j) telles que, d’une part $i < j$, d’autre part `t[i]` et `t[j]` soient des nombres opposés.