

TD4 de Programmation C : tableaux

L1 MIASHS, 2022-2023

Le but de ce TD est de réaliser différents programmes avancés mettant en œuvre des tableaux et des chaînes de caractères, et utilisant les concepts vus précédemment (conditionnelles, itérations).

Exercice 1 (Tableau de réels) *Le but de cet exercice est de manipuler un tableau de réels.*

Activité 1.1 *Écrire un programme qui remplit un tableau de N réels (type `double`) demandés à l'utilisateur. N sera déclaré comme une constante que vous pourrez fixer à 10 par exemple.*

Activité 1.2 *Compléter ce programme avec le code permettant d'afficher le contenu du tableau sur une même ligne, en séparant et formatant les valeurs pour les rendre plus lisibles.*

Activité 1.3 *Compléter ce programme avec le code permettant de calculer et d'afficher la somme de toutes les valeurs du tableau.*

Activité 1.4 *Compléter ce programme avec le code demandant une valeur à l'utilisateur, cherchant si cette valeur est dans le tableau, et affichant le résultat (et l'indice dans le tableau si trouvée). Bien sûr, arrêter de parcourir le tableau dès que la valeur a été trouvée (si trouvée).*

Activité 1.5 *Compléter ce programme avec le code comptant le nombre de cases du tableau où la valeur recherchée est présente et afficher ce nombre.*

Exercice 2 (Chaînes de caractères) *Le but de cet exercice est de manipuler des chaînes de caractères comme des objets à part entière, mais aussi comme des tableaux de caractères.*

Activité 2.1 *Écrire un programme qui fait saisir une chaîne de caractères (de longueur maximale 15) par l'utilisateur, puis l'affiche ainsi que sa longueur. Utiliser la fonction `strlen` qui calcule la longueur de la chaîne (inclure la bibliothèque `string.h`).*

Activité 2.2 *Compléter ce programme avec le code permettant de compter le nombre de lettres (minuscules ou majuscules) dans cette chaîne qui sont une de vos initiales (nom ou prénom). Afficher ce nombre. Utiliser la longueur de la chaîne afin de ne considérer que les cases utilisées dans le tableau.*

Activité 2.3 *Compléter ce programme avec le code permettant de remplacer chaque caractère de cette chaîne qui n'est pas une lettre par le caractère '*'.*

Activité 2.4 *Compléter ce programme avec le code permettant de vérifier si la chaîne est un palindrome ou non, et afficher le résultat.*

Rappel : un palindrome est un mot qui est identique, qu'on le lise de la gauche vers la droite ou de la droite vers la gauche.

Exercice 3 (Tri par sélection) *L'objectif de cet exercice est de trier par ordre décroissant un tableau d'entiers en utilisant le tri par sélection.*

Cette méthode consiste à rechercher la position de la plus grande valeur contenue dans le tableau, et à échanger l'élément en position 0 avec la plus grande valeur trouvée. Poursuivre en recommençant le même traitement à partir de l'indice 1, en recherchant la plus grande valeur située dans la portion de tableau commençant à l'indice 1, puis échanger cet élément avec la valeur située en position 1. Et ainsi de suite jusqu'à la fin du tableau.

Les activités suivantes vous permettent de construire pas à pas ce programme.

Activité 3.1 Définir la constante **Taille**, de valeur 30, qui sera utilisée comme taille du tableau d'entiers.

Activité 3.2 Écrire le code permettant de remplir le tableau d'entiers avec des valeurs aléatoires comprises entre 0 et 24. Utiliser les fonctions `srand`, `time` et `rand` vues dans le TD3.

Activité 3.3 Écrire ensuite le code permettant d'afficher toutes les valeurs du tableau, en suivant l'ordre des indices, séparées du caractère '|'.

Activité 3.4 Écrire ensuite le code recherchant l'indice **max** où se trouve la valeur maximale du tableau, sachant que la recherche se fait entre les indices **debut** et **Taille-1**. Pour cette question, la valeur de **debut** est initialisée à 0.

Activité 3.5 Compléter ce code en ajoutant l'échange des valeurs aux indices **debut** et **max**. Ainsi, la valeur maximale du tableau est placée au début, donc à sa position définitive dans le futur tableau trié.

Activité 3.6 Modifier votre code pour répéter la recherche de l'élément maximal et l'échange, en faisant progresser la valeur de **debut** à chaque itération, pour ne considérer que la partie non encore triée du tableau.

Activité 3.7 À la fin de votre programme, afficher à nouveau les valeurs du tableau, qui maintenant doivent être triées par ordre décroissant.

Activité 3.8 Compléter votre programme en demandant une valeur à l'utilisateur et en cherchant le plus efficacement possible si cette valeur est dans le tableau ou pas. Afficher le nombre de tests qui ont été nécessaires pour avoir la réponse. Pour cette recherche, vous pouvez par exemple implanter l'algorithme de recherche dichotomique vu dans le cours d'algorithmique.