

TD 4 - Fonctions, tableaux et pointeurs

Techniques de Programmation

Exercice 1

Implémenter une fonction permutant deux variables entières.

Exercice 2

Implémenter un programme demandant à un utilisateur de saisir un tableau d'entier. Le nombre d'éléments du tableau sera inconnu et donné par l'utilisateur.

Implémenter, en langage C, l'algorithme du tri à bulle sur un tableau statique. Modifier le programme afin de placer la partie tri de l'algorithme dans une fonction prenant en entrée le tableau non trié et donnant en sortie le tableau trié.

Modifier le programme afin de travailler sur un tableau dynamique.

Exercice 3

Déclarer et initialiser 2 tableaux de caractères (`cTab1` et `cTab2`). Puis :

1. Écrire une fonction `longueur_chaine1` qui admette en paramètre un tableau de caractères se terminant par `NULL`, et qui retourne le nombre de caractères du tableau (`NULL` exclu).
2. Écrire une fonction `longueur_chaine2` qui implante la même interface que `longueur_chaine1`, mais en donnant à son paramètre le type pointeur vers un `char`.
3. Le programme imprimera le nombre d'éléments de `cTab1` et `cTab2` par un appel de `longueur_chaine1` et `longueur_chaine2`.

Exercice 4

Écrire un programme permettant d'effectuer un ensemble d'opérations sur une chaîne de caractères quelconque saisie à partir du clavier. Ce programme est constitué d'un menu comportant le choix de l'opération à effectuer. Les opérations sur cette chaîne sont les suivantes :

- Fonction `saisir` : elle lit une chaîne de caractères à partir du clavier, elle retourne cette chaîne.
- Fonction `afficher` : elle affiche la chaîne argument.
- Fonction `inverser` : elle inverse la chaîne argument (elle la modifie donc).
- Fonction `mots` : elle compte le nombre de mots de la chaîne. On considère le caractère ' ' (blanc) comme le caractère séparateur de mots. Il peut y avoir plusieurs blancs consécutifs dans la chaîne.

Après chaque opération, le retour au menu s'effectue après l'affichage du message "Frapper une touche pour revenir au menu".