

Chaînes de caractères

Florian Bourse

Mémo

Caractères :

char c

'a' caractères

'\\' caractères particuliers

'\n' retour à la ligne

'\0' caractère nul qui termine une chaîne de caractères

Chaînes :

char str[]

""

"Hello_World!\n"

str[i]

ou **char * str**

chaîne vide

chaîne de caractères

(i + 1)-ième char de **str**

1 En-tête <string.h>

0 Écrire une suite d'instruction qui permet de calculer la longueur d'une chaîne de caractères.
Il s'agit de la fonction **size_t strlen(const char *)**;

1 Écrire une suite d'instruction qui permet de copier une chaîne de caractères à la place d'une autre. On supposera que la destination est suffisamment grande pour contenir la source.
Il s'agit de la fonction **char *strcpy(char *toHere, const char *fromHere)**;

2 Écrire une suite d'instruction qui permet de copier une chaîne de caractères à la suite d'une autre. On supposera que la destination est suffisamment grande pour contenir les deux chaînes de caractères mises bout à bout.

Il s'agit de la fonction **char *strcat(char *dest, const char *src)**;

3 Écrire une suite d'instruction qui permet de comparer deux chaînes de caractères en utilisant l'ordre lexicographique. Un nombre négatif représente <, 0 représente =, et un nombre positif représente >.

Il s'agit de la fonction **int strcmp(const char *, const char *)**;

4 Écrire une suite d'instruction qui permet de chercher la première occurrence d'une chaîne de caractères à l'intérieur d'une autre.

Il s'agit de la fonction **char *strstr(const char *haystack, const char *needle)**;

2 Écriture d'un nombre

5 Écrire une suite d'instruction qui permet de déterminer le nombre contenu dans une chaîne de caractère.

Il s'agit de la fonction **int atoi(const char *str)**;

6 Écrire une suite d'instruction qui permet de remplir une chaîne de caractères avec l'écriture décimale d'un nombre donné. On supposera que la chaîne de caractère donnée est de taille suffisante.

Il s'agit de la fonction **int sprintf (char * str, const char * format, ...)**;
avec la chaîne de formattage "%d".

3 Analyse de chaînes de caractères

- [7] Écrire une suite d'instruction qui permet de savoir si une chaîne de caractère donnée est un palindrome, c'est-à-dire qu'elle est la même qu'on la lise de gauche à droite ou de droite à gauche.
- [8] Écrire une suite d'instruction qui permet de savoir si une chaîne de caractère donnée est un carré, c'est à dire qu'elle est formée de deux copies successives d'une même chaîne, comme par exemple `bonjourbonjour` ou `abbabb`.
- [9] Écrire une suite d'instruction qui permet de vérifier si une chaîne de caractères est bien parenthésée. C'est-à-dire que toute parenthèse fermante rencontrée correspond à une parenthèse qui a précédemment été ouverte, et aucune parenthèse ne reste ouverte sans être fermée.

4 Anagrammes

- [10] Écrire une suite d'instructions qui permet de compter dans un tableau le nombre d'occurrence de chaque caractère au sein d'une chaîne de caractères.
- [11] En déduire une suite d'instructions qui permet de tester si deux chaînes de caractères sont des anagrammes, c'est-à-dire qu'elles possèdent les mêmes caractères mais pas forcément dans le même ordre.