

Exercice 1 :

On s'intéresse dans cet exercice à la manipulation de chaînes de caractères et on **s'interdit** l'utilisation des fonctions prédéfinies dans `string.h`. De plus, la longueur d'une chaîne de caractères **ne doit pas être** utilisée comme paramètre des sous-programmes et les chaînes doivent être manipulées directement via la notation pointeur (utilisation de `l*` et non des `[]`).

1. Écrire une fonction `longueur` qui calcule et retourne le nombre de caractères contenus dans une chaîne de caractères donnée.
2. Écrire une fonction `affiche` qui affiche à l'écran un à un les caractères d'une chaîne de caractère donnée, et termine par un retour à la ligne.
3. Écrire une fonction `rech` qui détermine si un caractère donné est dans une chaîne de caractères donnée.
4. Écrire une fonction `recop` qui recopie une chaîne donnée dans une autre.
5. Écrire une fonction `concat` qui recopie une chaîne donnée à la suite d'une autre chaîne donnée.
6. Écrire une fonction `compare` qui compare le contenu de deux chaînes de caractères données. Le sous-programme retournera 0 si les deux chaînes sont identiques, 1 (respectivement -1) si la première chaîne vient après (resp. avant) la seconde, selon l'ordre alphabétique.
7. Écrire une fonction `contient` qui retourne `true` si une chaîne donnée est contenue dans une autre (donnée également), `false` sinon.
8. Écrire un programme pour tester ces différents sous-programmes

BONUS Écrire une fonction (itérative) `aff_inv` qui affiche à l'écran un à un les caractères d'une chaîne de caractères donnée, mais à l'envers.

BONUS Écrire une fonction (itérative) `inv` qui inverse une chaîne de caractères (sans passer par une autre chaîne de caractères).

Exercice 2 :

1. On souhaite stocker dans un tableau un ensemble de mots. Écrire une fonction qui demande à l'utilisateur le nombre de mots à saisir puis fait la saisie mot par mot, en allouant à chaque fois l'espace **strictement nécessaire**.
2. Écrire un programme qui appelle ce sous-programme, puis affiche récursivement tous les mots, dans l'ordre inverse de la saisie, et enfin libère la mémoire allouée.