

TP2 : Chaînes de caractères

★ Exercice 1. Déclaration des chaînes de caractères

Dans la fonction main, reprendre les déclarations suivantes :

```
char s1[] = {'S', 'a', 'l', 'u', 't', '\0'};
char s2[] = "Salut";
char *s3; s3="Salut";
char *s4="Salut";
char s5[20];
```

▷ **Question 1.** Initialiser s5 par la chaîne de caractères “Salut” et afficher l’ensemble des chaînes déclarées à l’aide de printf.

▷ **Question 2.** Remplacer la première lettre (S) des différentes chaînes par un autre caractère au choix et les afficher à l’écran. Conclure !

★ Exercice 2. Majuscules et minuscules

▷ **Question 1.** Écrire une fonction `char min2maj(char c)` qui convertit un caractère minuscule en majuscule.

▷ **Question 2.** Écrire une fonction `char maj2min(char c)` qui convertit un caractère majuscule en minuscule.

▷ **Question 3.** Écrire dans le main un programme qui lit à partir du clavier une chaîne de caractères et qui l’affiche en transformant chaque caractère minuscule en majuscule et inversement.

★ Exercice 3. Occurrences d’un caractère dans un mot

▷ **Question 1.** Écrire la fonction `first_occurrence` qui renvoie le rang de la première occurrence d’un caractère dans une chaîne de caractères si trouvé sinon elle retourne 0.

▷ **Question 2.** Écrire la fonction `last_occurrence` qui renvoie le rang de la dernière occurrence d’un caractère dans une chaîne de caractères si trouvé sinon elle retourne 0.

▷ **Question 3.** Écrire un programme qui trouve la première et la dernière occurrence d’un caractère dans un mot. Le mot et le caractère étant lu à partir du clavier.

▷ **Question 4.** Généraliser à une suite de mots tapée au clavier se terminant par le mot “exit” qui provoque la terminaison du programme.

★ Exercice 4. Tableaux de caractères : Cadavre exquis L’objectif de cet exercice est de constituer un écrivain automatique. Pour cela, il faut initialiser plusieurs tableaux de chaînes de caractères : un tableau de sujets accompagnés d’un article, un tableau de verbes conjugués à la troisième personne, un tableau de compléments d’objets directs au masculin et un tableau d’adjectifs au masculin.

<i>Sujet</i>	<i>Adjectif</i>	<i>Verbe</i>	<i>Complément</i>	<i>Adjectif</i>
Le cadavre	exquis	boira	le vin	nouveau
Le chat	noir	mange	un rat	gris
Le chien	pouilleux	aime	un os	moelleux
La grenouille	gluante	capture	un moustique	agressif

Ce jeu fut inventé par les surréalistes vers 1954. La première phrase construite sur ce modèle fut «Le cadavre exquis boira le vin nouveau», ce qui donna le nom du jeu.

▷ **Question 1.** Le premier problème à résoudre est de savoir comment stocker les morceaux de phrases qui seront combinés ensuite. Le plus simple est d'utiliser des tableaux statiques de chaînes statiques. Vous avez deux possibilités, présentées dans les deux blocs suivants. Choisissez l'une d'entre elles, et écrivez le début de votre programme.

```
char *nom[]={"le cadavre","le chat","le chien","la grenouille"};
char *adj1[]={"exquis","noir","pouilleux","gluant"};
...

char *elements[][4]={
    {"le cadavre","le chat","le chien","la grenouille"},
    {"exquis","noir","pouilleux","gluant"},
    ...
};
```

▷ **Question 2.** Complétez votre programme.

Obtenir un nombre aléatoire en C n'est pas très compliqué. Il faut utiliser la fonction `rand()` qui retourne un entier entre 0 et `RAND_MAX`. Le problème est que la suite pseudo-aléatoire est toujours la même par défaut. Initialiser cette suite à l'heure actuelle est une bonne solution à ce manque de variété.

```
#include <stdlib.h>
#include <time.h>

int main() {
    // Initialise le pseudo aléatoire
    srand(time(NULL));
    // tire un nombre entre 0 et 3
    int nb = rand() % 4;
}
```