
SÉANCE 5



Objectif

Le but de cette cinquième séance est de manipuler les fonctions, le passage en paramètres des tableaux et des chaînes de caractères, ainsi que l'utilisation des pointeurs.



Exercices

✎ Exercice 1 (Fonctions de manipulation de chaînes de caractères)

L'objectif de cet exercice consiste à écrire des fonctions de manipulation de chaînes de caractères **sans** utiliser les fonctions de la bibliothèque standard (donc sans inclure `string.h`).

Testez les unes après les autres les fonctions en utilisant la fonction principale fournie dans le fichier `tp5ex1main.c` et en enlevant les commentaires au fur et à mesure de votre progression.

Dans la suite, une chaîne de caractères est manipulée au travers de l'adresse de son premier caractère (adresse de début de la chaîne).

1. Écrivez la fonction d'en-tête :

```
int Lecture(char *ch, int NbCarMax)
```

qui lit une ligne de texte tapée au clavier et la stocke, en ajoutant le caractère `'\0'` mais sans stocker le caractère `'\n'`, dans la chaîne désignée par le paramètre `ch`. Le paramètre `NbCarMax` contient le nombre maximal de caractères lus (pour éviter les débordements en mémoire). La fonction doit retourner le nombre de caractères lus (sans compter le caractère `'\n'`).

Vous pouvez vous inspirer de ce que vous avez écrit pour l'exercice 3 de la séance 3.

2. Écrivez la fonction d'en-tête :

```
int Longueur(char *ch)
```

qui retourne la longueur (nombre de caractères sans compter le caractère `'\0'`) de la chaîne dont l'adresse du premier caractère est passée en paramètre.

3. Écrivez la fonction d'en-tête :

```
void Copie(char *Destination, char *Source)
```

qui copie la chaîne de caractères désignée par le paramètre `Source` dans la chaîne désignée par le paramètre `Destination`. Attention, la copie doit être une « vraie » chaîne de caractères, c'est-à-dire se terminer par `'\0'`.

4. Écrivez la fonction d'en-tête :

```
int Differentes(char *ch1, char *ch2)
```

qui retourne 1 si les chaînes désignées par `ch1` et `ch2` sont différentes (contiennent au moins un caractère différent), 0 sinon.

5. Écrivez la fonction d'en-tête :

```
void Ajout(char *Destination, char *Source)
```

qui ajoute la chaîne de caractères désignée par `Source` à la fin de la chaîne de caractères désignée par `Destination`.

☞ Exercice 2 (Plus petit et plus grand éléments d'un tableau d'entiers)

Pour les deux fonctions ci-dessous, écrivez une fonction principale permettant de les tester.

On suppose que le tableau contient au moins un élément.

1. Écrivez la fonction d'en-tête :

```
void MinMax(int Tab[], int NbElt, int *pMin, int *pMax)
```

qui délivre en sortie le plus petit et le plus grand éléments d'un tableau de `int`.

Paramètres :

- `Tab` : tableau de `int` ;
- `NbElt` : nombre d'éléments du tableau `Tab` ;
- `pMin` : adresse où doit être stocké le plus petit élément ;
- `pMax` : adresse où doit être stocké le plus grand élément.

2. Écrivez une variante de la fonction précédente qui, au lieu de délivrer en sortie le plus petit et le plus grand éléments, délivre en sortie les **adresses** du plus petit et plus grand éléments.