

---

Licence E.E.A.  
TD N°3 de Langage C/C++

---

Ce TD a pour objectif de vous faire découvrir et utiliser le langage C. Il s'agira de transcrire sous la forme d'un programme C, des algorithmes afin de réaliser les fonctions demandées.

---

## 1 Les pointeurs

`adr1` et `adr2` sont des pointeurs pointant sur des réels. Le contenu de `adr1` vaut -45,78 ; le contenu de `adr2` vaut 678,89. Écrire un programme qui affiche les valeurs de `adr1`, `adr2` et de leur contenu.

## 2 Les chaînes

Saisir un texte. Ranger les caractères en mémoire. Lire le contenu de la mémoire et y compter le nombre d'espaces et de lettres `e`.

## 3 Jouons avec les adresses

Saisir 6 entiers et les ranger à partir de l'adresse `adr_deb`. Rechercher le maximum, l'afficher ainsi que son adresse.

## 4 Les tableaux

Saisir 10 réels, les ranger dans un tableau. Calculer et afficher la moyenne et l'écart-type.

## 5 Déterminant de matrice

Saisir une matrice d'entiers 2x2, calculer et afficher son déterminant.

## 6 Les fonctions sans paramètres

Écrire un programme qui affiche `bonjour` et `coucou` par l'intermédiaire de deux fonctions dont l'une appelle l'autre. On écrira le prototype de ces fonctions dans le fichier `fcn.h`.

## 7 Les fonctions qui renvoient des paramètres

Un programme contient la déclaration suivante :

```
double tableau liste[8] = 1.6,-6,9.67,5.90,345,-23.6,78,34.6 ; /* variable globale */  
Écrire une fonction de prototype double min(void) qui renvoie le minimum de la liste. Écrire une  
fonction de prototype double max(void) qui renvoie le maximum de la liste. Les mettre en oeuvre dans  
main().
```

## 8 Les fonctions avec passage d'un argument

Écrire une fonction `carre` qui prend un entier en argument et qui renvoie le carré de l'argument. Mettre en œuvre cette fonction dans un programme principal.

## 9 Les fonctions avec passages de paramètre par adresse

`tab1` et `tab2` sont des variables locales à main :

```
int tab1[10] = 0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19 ;
```

```
int tab2[10] = -19,18,-17,16,-15,14,-13,12,-11,10,-9,8,-7,6,-5,4,-3,2,-1,0 ;
```

Écrire une fonction de prototype `void affiche(int *tab)` qui permet d'afficher les 20 nombres suivant un tableau 4x5. La mettre en œuvre dans `main()` pour afficher `tab1` et `tab2`.

## 10 Définir ses types

Définir le type `typedef char ligne[80]` ; Déclarer dans `main()` un pointeur de ligne. Puis lui attribuer de la place en mémoire (pour 5 lignes). Écrire une fonction qui effectue la saisie de 5 lignes puis une autre qui les affiche. Les mettre en œuvre dans `main()`.

## 11 Les structures

Créer la structure d'une fiche signalétique qui contiendra le nom, le prénom, l'âge et une note et les fonctions qui permettent de saisir et afficher les fiches.

## 12 les structures : le retour

Créer une structure `pointint num;float x;float y;`.

Saisir 4 points, les ranger dans un tableau puis les afficher.