

## Atelier 2 : Fonctions, Variables locales et globales, Tableaux, Allocation dynamique, Pointeurs en C++

### Exercice 1

Écrire une fonction, sans argument ni valeur de retour, qui se contente d'afficher, à chaque appel, le nombre total de fois où elle a été appelée sous la forme : *appel numéro 3*

### Exercice 2

Écrire 2 fonctions à un argument **entier** et une valeur de retour **entière** permettant de préciser si l'argument reçu est multiple de 2 (pour la première fonction) ou multiple de 3 (pour la seconde fonction).

Utiliser ces deux fonctions dans un petit programme qui lit un nombre entier et qui précise s'il est pair, multiple de 3 et/ou multiple de 6, comme dans cet exemple (il y a deux exécutions) :

```
donnez un entier : 9
il est multiple de 3
-----
donnez un entier : 12
il est pair
il est multiple de 3
il est divisible par 6
```

### Exercice 3

Écrire, de deux façons différentes, un programme qui lit **10 nombres entiers** dans un tableau avant d'en rechercher le plus grand et le plus petit :

- en utilisant uniquement le « *formalisme tableau* » ;
- en utilisant le « *formalisme pointeur* », à chaque fois que cela est possible.

### Exercice 4

Écrire un programme **allouant dynamiquement** un emplacement pour un **tableau d'entiers**, dont la taille est fournie en donnée.

- Utiliser ce tableau pour y placer des nombres entiers lus également en donnée.
- Créer ensuite dynamiquement un nouveau tableau destiné à recevoir les carrés des nombres contenus dans le premier.
- Supprimer le premier tableau, afficher les valeurs du second et supprimer le tout.

### Exercice 5

Écrire un programme C++ qui :

- déclare un entier a;
- déclare une référence vers cet entier ref\_a;
- déclare un pointeur vers cet entier p\_a;
- affiche les variables, leurs adresses, la valeur pointée.

### Exercice 6

Écrire une fonction nommée **incrémenter()** permettant d'incrémenter la valeur d'une variable passée en paramètre et une fonction nommée **permuter()** permettant d'échanger les contenus de 2 variables de type int fournies en argument :

- en transmettant l'adresse des variables concernées (seule méthode utilisable en C) ;
- en utilisant la transmission par référence.

Dans les deux cas, écrire un programme (**main**) qui teste les deux fonctions.

### Exercice 7

Écrire un programme qui demande à l'utilisateur de taper 10 entiers qui seront stockés dans un tableau. Le programme doit trier le tableau par ordre croissant et doit afficher le tableau.

Algorithme suggéré (tri bulle) :

On parcourt le tableau en comparant t[0] et t[1] et en échangeant ces éléments s'ils ne sont pas dans le bon ordre.

- On recommence le processus en comparant t[1] et t[2],... et ainsi de suite jusqu'à t[8] et t[9].
- On compte lors de ce parcours le nombre d'échanges effectués.
- On fait autant de parcours que nécessaire jusqu'à ce que le nombre d'échanges soit nul : le tableau sera alors trié.