

# IMA 3<sup>ème</sup> année

## Programmation Avancé

### TP10

## Un peu de généricité et fonctions à nombre variable d'arguments

### 1 Objectifs

- Savoir passer une fonction en paramètre
- Savoir utiliser les `void *`

On écrira un fichier `.c` par exercice.

### 2 Passer une fonction en paramètre

Pour rendre générique des fonctions (e.g., traitement des nœuds d'un arbre binaire lors d'un parcours), on peut passer des fonctions en paramètre d'autres fonctions. On pourra se référer par exemple au document [http://www.newty.de/ftp/zip/f\\_fpt.pdf](http://www.newty.de/ftp/zip/f_fpt.pdf) (section2, parti C uniquement)

#### 2.1 Exo1 : utilisation simple

1. Écrire une fonction `fois_deux` qui retourne un entier égal au double de l'entier passé en paramètre. Quelle est la signature de cette fonction ?
2. Écrire une fonction `appliquer_tableau(int f(int), int t[], int size)` qui applique la fonction `f` à chacun des éléments du tableau `t`.
3. Tester dans le main :

#### 2.2 Exo2 : modifier la fonction de comparaison d'un tri d'entiers

1. Récupérer dans vos codes de TP précédents un tri de tableau d'entiers.
2. Que faut-il modifier pour faire un tri décroissant ?
3. Écrire une fonction `superieur(int a, int b)` qui renvoie `1` si `a` est strictement supérieur à `b`, `0` si `a=b`, `-1` sinon.
4. Écrire une fonction `inferieur` sur le même principe.
5. Rendre votre fonction de tri générique en passant une fonction `compare` en paramètre.
6. Tester dans le main.

### 3 Faire de la généricité

La généricité d'une fonction repose sur son indépendance vis-à-vis du type, et éventuellement du nombre de ses arguments. C'est un concept important car il permet d'augmenter le niveau d'abstraction du langage <sup>1</sup>.

#### 3.1 Exo3 : utilisation de `qsort`

La `libc` fournit une fonction de tri générique sur n'importe quel tableau (entiers, de caractères, ...) pourvu qu'on lui passe la fonction de comparaison en paramètre.

1. Consulter le manuel de la fonction `qsort` (commande `man`).  
Note : `const` signifie que l'argument n'est pas modifié dans la fonction.  
A votre avis, à quoi correspondent les `void *` ?
2. Utiliser cette fonction pour trier un tableau d'entier en utilisant la fonction `compare` de l'exercice précédent. On modifiera la fonction pour qu'elle ait le bon type : `int compare (const void * pa, const void * pb)`
3. Utiliser cette fonction pour trier un tableau de chaînes de caractères en utilisant `strcmp`.

---

1. <https://fr.wikipedia.org/wiki/Généricité>