

# LIFAP1 – TP3 : Fonctions et Procédures

*Objectifs* : Le but de ce TP est de revoir et manipuler toutes les notions que vous avez vues jusqu'à maintenant. A la fin, vous devrez bien maîtriser :

- l'importance de la fonction main ;
- l'utilisation des procédures et des fonctions ;
- les boucles simples (tant que, faire) ;
- les boucles imbriquées ;
- la structure conditionnelle (si...alors...sinon).

## 1. Premiers programmes

- Fonction permettant de retourner le maximum de deux réels passés en paramètres.
- Fonction permettant de calculer et de retourner la factorielle d'un nombre passé en paramètre. Utilisez ce sous-programme pour afficher les 15 premières valeurs des factorielles. Comparez les résultats de factorielle (13) et factorielle (14). Ces résultats vous semblent-ils cohérents et corrects ? Pourquoi ? Modifiez le type de retour de la fonction factorielle en "**double**" au lieu de "**int**" et observez les nouvelles valeurs obtenues.
- Fonction permettant de calculer la somme des n premières puissances de 2. On devra utiliser pour cela la fonction `pow(x, y)` de la bibliothèque `math.h` qui calcule et retourne  $x^y$ .

2. L'exemple suivant permet de choisir aléatoirement une valeur comprise entre 0 et 29. La fonction `rand` retourne un entier aléatoire compris entre 0 et une constante `RAND_MAX` (32767).

```
#include <iostream>
#include <time.h>      /* pour l'initialisation avec srand */
#include <stdlib.h>    /* librairie contenant rand() */

using namespace std;

int main (void)
{
    int valea;
    srand(time(NULL)); /* une seule fois en début de programme */
    valea = rand()% 30; /* a chaque fois qu'on veut une valeur */
    cout<<"la valeur aleatoire est : "<<valea<<endl;
    return 0;
}
```

- Modifiez le code précédent afin d'obtenir une valeur aléatoire comprise entre 1 et 30 puis entre 10 et 25.
- Utilisez ce code pour écrire un sous-programme permettant de tirer aléatoirement une valeur et de la retourner au programme principal.
- Ecrivez le programme principal permettant de faire deviner au joueur la valeur choisie par l'ordinateur. Il disposera de 5 essais pour trouver la valeur.

3. La suite de Fibonacci est une suite d'entiers dans laquelle chaque terme est la somme des deux termes qui le précèdent. Elle commence par fibonacci (0) = fibonacci (1) = 1.

Affichez le nième terme de la suite de Fibonacci (n étant passé en paramètre).

