

Algorithmique et programmation C++ (DEV11)

TP N°4

Exercice 1.

1. Écrire une fonction `estPremier()` qui prend comme paramètre un entier et qui retourne `true` si `n` est premier et `false` sinon.
2. Tester la fonction `estPremier()` dans `main()` en l'appelant avec des arguments saisis par l'utilisateur.
3. Modifier votre programme en mettant le prototype de la fonction `estPremier()` dans un fichier header (`.hpp`).
4. Utiliser la fonction `estPremier()` pour afficher les nombres premiers de l'intervalle `[2..1000]`

Exercice 2.

1. Écrire une fonction qui prend comme paramètres une chaîne de caractères `s` et un caractère `c` et qui affiche le nombre d'occurrence de `c` dans `s`.
2. Tester la fonction avec des arguments saisis par l'utilisateur. Traiter le cas d'une chaîne `s` contenant des espaces.

Exercice 3.

1. Écrire une fonction `permutation()` qui prend deux nombres réels `a` et `b` et qui permute leurs valeurs. La fonction ne retourne rien.
2. Tester la fonction avec `a=20` et `b=30.75`. Afficher `a` et `b` avant et après la permutation.
3. Proposer deux solutions différentes pour que les variables `a` et `b` soient modifiés après l'appel de la fonction.

Exercice 4.

On appelle récursive toute fonction qui s'appelle elle-même.

Exemple :

```
int factorielle(int n){
    if ((n == 0)or(n==1))
        return 1;
    else
        return n*factorielle(n-1);
}
```

1. Écrire une fonction récursive qui calcule la somme $n+(n-1) + \dots + 0$.
2. Écrire une fonction récursive qui calcule X^n .

Exercice 5.

Écrire fonction pour trouver le PGCD de deux nombres en utilisant la récursivité.

Exercice 6.

1. Écrire une fonction qui prend comme paramètre un tableau `t` d'entiers et son nombre d'éléments `n`, et la fonction affiche le minimum et le maximum du tableau. Voici le prototype de la fonction

```
void affichage(int t[ ], int n)
```

2. Tester la fonction avec :
 - a. Un tableau initialisé au moment de la déclaration (`t={....}`).
 - b. Un tableau dont les éléments sont saisis par l'utilisateur
 - c. Un tableau rempli par des valeurs aléatoires de l'intervalle `[-100, 100]`

Exercice 7.

1. Écrire une fonction `affichage()` qui prend comme paramètres une matrice carrée `M` d'ordre `n` et affiche les éléments de `M` sous forme matricielle. Voici le prototype de la fonction.

```
void affichage(int t[MAX][MAX], int n)
```

`MAX` est une constante déclarée avec le mot clé `#define` et initialisé à 100.

2. Écrire une fonction `min(int t[MAX][MAX], int n, int i)` qui affiche le minimum de la ligne `i` de `M`.