

TD1 : Récursivité

Exercice 1

Ecrire une fonction récursive qui calcule la puissance de deux nombres: a^n . Le prototype de la fonction est fourni ci-dessous:

```
double Power (double a, unsigned int n);
```

Exercice 2

Ecrire sous forme récursives les fonctions $u(n)$ et $v(n)$ conformément à la description suivante:

Valeurs initiales:

$$u_0=1$$

$$v_0=2$$

Définition:

$$u_n=3.u_{n-1} + 2.v_{n-1}$$

$$v_n=2.u_{n-1} + 3.v_{n-1}$$

Exercice 3

Ecrire le code d'une fonction qui affiche un nombre n en binaire. Inspirez-vous du principe des divisions successives pour faire apparaître la récursivité. Le prototype est donné :

```
void Base2 (unsigned int n);
```

Par exemple le programme principal suivant:

```
void main()  
{  
    Base2(85);  
}
```

affiche :

> 1010101

Exercice 4

Ecrire une fonction récursive qui retourne vrai si le mot passé en paramètre est un palindrome¹ et faux sinon. On supposera ici que la chaîne de caractères ne contient pas d'espaces. Le prototype est donné ci-dessous :

```
int Palindrome (char *phrase, int NbCaract)
```

phrase pointe vers la chaîne de caractères à tester.

NbCaract est le nombre de caractères de la chaîne à tester.

1 Un palindrome est un texte ou un mot dont l'ordre des symboles (lettres, chiffres, etc) reste le même qu'on le lise de gauche à droite ou de droite à gauche (par ex: "radar", "Engage le jeu que je le gagne" ou "Tu l'as trop écrasé, César, ce Port-Salut !").