

Série TD - 01 -

Exercice 01 Écrire une fonction en algorithmique (itérative puis récursive) permettant de calculer un terme de la suite de *Fibonacci* définie comme suit:

$$U_n = \begin{cases} n & \text{si } n = 0 \text{ ou } n = 1 \\ U_{n-1} + U_{n-2} & \text{sinon} \end{cases}$$

Exercice 02

1. Écrire une fonction récursive qui calcule le produit de deux entiers naturels a et b ;
2. Écrire une fonction récursive qui permet de calculer le résultat de la division entière de a sur b ($a \text{ div } b$);
3. Écrire une fonction récursive qui permet de calculer le reste de la division entière de a sur b ($a \text{ mod } b$).

Exercice 03 La fonction récursive *convertirEnBase10*(n : entier, b : entier) convertit un nombre $n \geq 0$ écrit en base b en un nombre en base 10.

Exemple: *convertirEnBase10*(100, 2) donne 4; *convertirEnBase10*(137, 11) donne 161; *convertirEnBase10*(100, 16) donne 256.

1. Écrire la fonction *convertirEnBase10*;
2. Dérouler cet algorithme pour $n = 137$ et $b = 11$.

Exercice 04 Transformer la boucle suivante en algorithme récursif avec deux paramètres.

```

pour n = 1 à 4 faire
  pour m = 1 à 10 faire
    écrire(n*10+m)
  finpour
finpour

```

Exercice 05 Écrire un algorithme itératif puis récursif qui fait l'approximation par *dichotomie* de la racine unique d'une fonction f dans un intervalle $]A, B[$.

