

TP N° 2 : DIVISER POUR RÉGNER

Compléments Python :

Rendre globale une variable locale :

```
global nb_mult # lors de la déclaration dans la fonction
nb_mult=0      # remise à zéro avant l'appel de la fonction
```

Exercice 1 : Recherche dichotomique du zéro d'une fonction

1. En utilisant l'algorithme vu en cours, écrire une fonction `dicho` qui, à partir d'un intervalle $[a, b]$, permet de trouver une valeur approchée à `epsilon` près de la solution de l'équation $x^2 - r = 0$, où `a`, `b`, `r` et `epsilon` sont des flottants choisis par l'utilisateur.
2. Écrire une fonction `dicho2` qui prend en paramètre :
 - la fonction dont on recherche le 0
 - `a` et `b` les bornes de l'intervalle de recherche supposées telles que $f(a) * f(b) < 0$
 - `epsilon` la précision souhaitée

et renvoie une valeur approchée d'une solution de l'équation $f(x)=0$ sur l'intervalle $[a, b]$.

Tester avec les fonctions :

- `cos` sur l'intervalle $[0, 2]$
- `log` sur l'intervalle $[0.5, 2]$

Exercice 2 : Calcul de la puissance d'un réel x

Étant donné un entier `n` et un réel `x`, on veut calculer x^n . Écrire une fonction récursive `puis` qui implémente la spécification récursive suivante :

si `n=0` `resultat=1`

sinon si `n` est pair `resultat=puissance(x,n//2)*puissance(x,n//2)`

sinon `resultat=x * puissance(x,n-1)`

Soit $m(n)$ le nombre de multiplications nécessaires à l'exécution de `puissance(x,n)`. Calculer $m(n)/\ln(n)$ pour `n` grand. Quelle est la complexité de la fonction récursive `puis` ?

Exercice 3 (Contrôle Intermédiaire 2007-2008)

On dispose d'un tableau d'entiers **relatifs** `tab` de taille $n > 1$. On cherche à déterminer le maximum de $\text{tab}[i] + \text{tab}[i+1]$ lorsque i parcourt les indices du tableau `tab`. Par exemple, pour le tableau `tab = [-1, 9, -3, 12, -5, 4]`, la solution est 9 ($= \text{tab}[2] + \text{tab}[3]$). On écrira deux versions :

1. Une fonction `meilleurMax0` à l'aide d'un algorithme itératif élémentaire.
2. Une fonction `meilleurMax1` à l'aide d'un algorithme de type « diviser pour régner ».
3. Tester en générant des listes aléatoires et en vérifiant l'égalité des résultats renvoyés par les 2 fonctions.