

## TP 2 : MÉTHODES DE POINT FIXE

On importera le module `math` en tapant `from math import *` en début de TP et on utilisera  $|x| < \text{prec}$  pour tester  $x = 0$ , avec  $\text{prec} = 10^{-15}$  (précision machine).

Toute fonction mathématique sera définie comme une procédure Python.

**Exercice 1.** On veut calculer l'unique racine réelle de la fonction  $f(x) = x^3 - x^2 + 8x - 8$  en appliquant la méthode du point fixe avec les 3 fonctions suivantes :

$$g_1(x) = -x^3 + x^2 - 7x + 8, \quad g_2(x) = \frac{x^3 - 8}{x - 8}, \quad g_3(x) = \frac{2x^3 - x^2 + 8}{3x^2 - 2x + 8}.$$

- Justifier que  $f$  admet  $\ell = 1$  pour seule racine réelle et vérifier que  $f(x) = 0$  équivaut à  $g_i(x) = x$  pour  $i = 1, \dots, 3$ .
- Pour chaque fonction  $g_i$ , calculer les 6 premières itérations de la méthode du point fixe en prenant  $x_0 = 0.5$  comme condition initiale. Commenter.

### Exercice 2.

- Écrire une procédure `PointFixe(g, x0, p, N)` qui calcule par la méthode du point fixe avec condition initiale  $x_0$  le couple  $(x_n, n)$  où  $n$  est le plus petit nombre d'itérations tel que  $|g(x_n) - x_n| < p$ . La procédure retournera `False` si  $n$  dépasse  $N$ .
- On veut calculer le point fixe de la fonction cosinus.
  - Vérifier que  $\cos(]0, 1[) \subset ]0, 1[$ . Justifier que la fonction  $g(x) = \cos(x)$  a un unique point fixe dans  $]0, 1[$ .  
*On admet que la méthode du point fixe converge pour tout  $x_0 \in ]0, 1[$ .*
  - Appliquer l'algorithme `PointFixe` à la fonction  $g$  avec  $x_0 = 0.5$ ,  $p = 10^{-10}$  et  $N = 100$ .
  - Regarder comment varie le nombre d'itérations pour d'autres choix de la condition initiale  $x_0 \in \mathbb{R}$  et de la précision  $p$ .

### Exercice 3.

- Écrire une procédure `Newton(f, g, x0, p, N)` basée sur la méthode de Newton avec condition initiale  $x_0$  qui retourne le couple  $(x_n, n)$  où  $n$  est le plus petit nombre d'itérations tel que  $|f(x_n)| < p$ . On retournera `False` si  $n$  dépasse  $N$ . On utilise l'argument `g` pour fournir la fonction  $g = f'$  à la procédure.
- Utiliser votre procédure `Newton` pour calculer  $\sqrt{3}$  avec précision  $10^{-10}$  en partant de  $x_0 = 0, 1, 2, 3, 20$ . Comparer le nombre d'itérations requises avec la méthode dichotomique (TP 1). Combien d'itérations faut-il en partant d'une approximation  $x_0$  de  $\sqrt{3}$  à 0.1 près ?
- Procéder de même pour calculer  $\ln(2)$ . Qu'observe-t-on pour  $x_0 = 20$  ? Comment adopter la stratégie ?
- À l'aide de la librairie `matplotlib` et de votre procédure `Newton`, tracer le graphe de la fonction racine carrée sur  $[0, 4]$ . On pourra calculer par exemple 100 valeurs de la fonction. Comparer avec le TP 1.

### Exercice 4.

- Montrer que la méthode de Newton pour rechercher une racine de  $f$  est un cas particulier de méthode de point fixe. Pour quelle fonction  $g$  ?
- Commenter à nouveau l'exercice 1.
- Écrire la relation de récurrence correspondant au calcul de  $\sqrt{3}$  dans l'exercice 2. Connaissez-vous cette suite récurrente ?

### Exercice 5.

- Modifier la procédure `Newton` en une procédure `Sécante(f, x0, x1, p, N)` basée sur la méthode de la sécante.
- Tester la procédure `Sécante(f, x0, x1, p, N)` sur les exemples précédents et comparer avec `Newton`.
- Quel est l'avantage de la méthode de la sécante ?  
Proposer une variante de `Newton` qui n'utilise pas l'argument `g`.