RAPPORT TP1

Résolution numérique f(x) = 0

NOMS Prénoms :
SAVADOGO Hamed Kouka
DIACK Aliou

Année : 2017/2018
L2 Mathématiques

1) Création d'une fonction

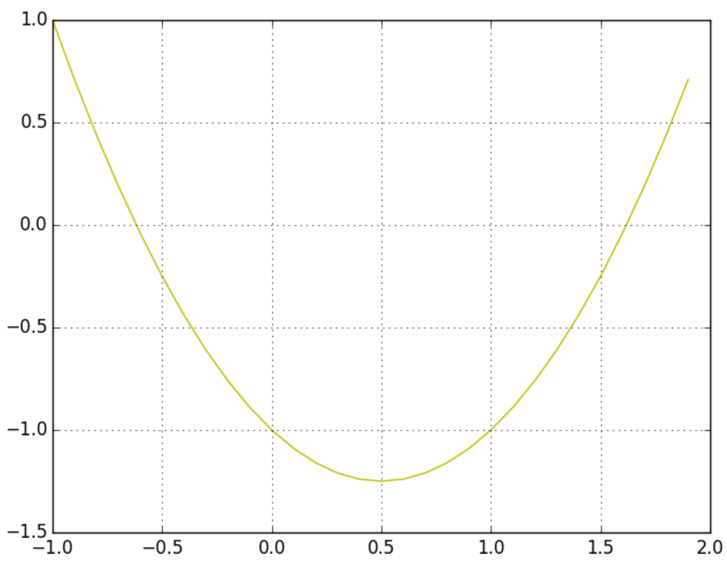
Fonction f(x) et représentation graphique

Nous définissons une fonction $f(x) = x^2 - x - 2$ en python. Cette fonction s'écrit comme suit :

```
>>>  def f(x):
>>>      return x**2 - x – 1
```

Représentation graphique

A l'aide du module matplotlib.pyplot représentons la fonction f(x)



2) Méthode du point fixe

Vérifions que les points fixes de g sont les zéros de f :

$$f(x) = 0$$

$\Leftrightarrow$ $$x^2 - x - 1 = 0$$

$\Leftrightarrow$ $$x - 1 - \frac{1}{x} = 0$$

$\Leftrightarrow$ $$x = 1 + \frac{1}{x}$$

donc $g(x) = x + \frac{1}{x}$

Les 25 premiers termes de la suite définies par $x_{n+1} = g(x_n)$ et $x_0 = 1.0$ sont :

1.0
2.0
1.5
1.66666666667
1.6
1.625
1.61538461538

1.61904761905
1.61764705882
1.61818181818
1.61797752809
1.61805555556
1.61802575107
1.61803713528
1.61803278689
1.61803444782
1.6180338134
1.61803405573
1.61803396317
1.61803399852
1.61803398502
1.61803399018
1.61803398821
1.61803398896
1.61803398867
1.61803398878

3) Méthode de point fixe

La fonction point fixe (voir code)

<u>Tests :</u>

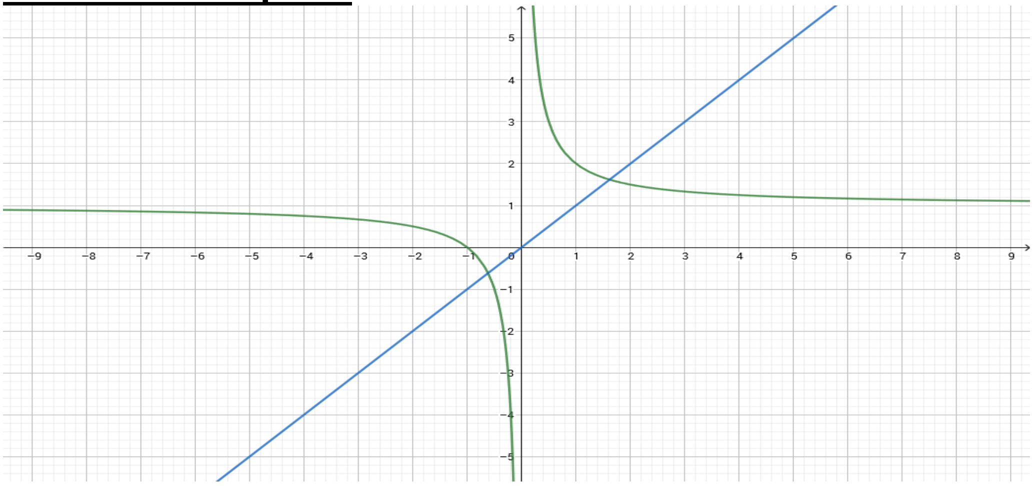
- Test 1 avec $g(x) = 1 + \frac{1}{x}$, $x_0 = 1.0$, $\mathcal{E} = 10^{-12}$

On obtient comme résultats : r = 1.61803398875 et nbiter = 38
g(r)=1.61803398875

- Test 2 avec $g(x) = 1 + \frac{1}{x}$, $x_0 = -0.6$, $\mathcal{E} = 10^{-12}$

On obtient comme résultats   r2 = 1.61803398875 et nbiter = 44

<u>Dessins sur le repère :</u>



- Point fixe attractif, les points se rapprochent et dans ce cas la suite converge.

- Point fixe répulsif, les points du colimaçon s'éloignent et dans ce cas la suite diverge.

4) La méthode de newton

Appliquons la méthode du point fixe à la fonction $g(x) = x - \frac{f(x)}{f'(x)}$

On écrit une fonction python newton qui prend en arguments une fonction *f*, sa dérivée *df*, une valeur initiale x_0, un réel positif ε, et qui renvoie une approximation *r* d'une racine de la fonction. (voir fichier .py)

Tests :

- Test 1 : *r1* = 1.61803398875 et nbiter = 6

- Test 2 : *r2* = -0.61803398875 et nbiter = 5

5) La méthode de dichotomie

On écrit une fonction python dichotomie qui prend en arguments une fonction *f*, deux valeurs initiales *a*, *b*, un réel positif ε , et qui renvoie un encadrement *a, b* d'une racine de la fonction *f*. Voir fichier .py

Tests :

- Test 1 : Le test 1 renvoie t1 = 1.61803398875

- Test 2 : Le test 2 renvoie t2 = -0.61803398875

Nous faisons varier la valeur d'epsilon pour chaque méthode en vue de compter à chaque fois le nombre d'itérations. Les résultats sont dans le tableau ci-dessous.

	10^{-3}	10^{-6}	10^{-9}	10^{-12}	10^{-15}
Point fixe	9 et 15	16 et 22	23 et 29	30 et 36	38 et 44
Newton	5 et 4	5 et 4	6 et 5	6 et 5	7 et 6
Dichotomie	1	1	1	1	1

La fonction point fixe est celle qui a le plus d'itérations, newton et sécante ont un nombre d'itération à peu près égale et beaucoup plus petit comparé à point fixe. Dichotomie nous semble la plus efficace.