COMPTE RENDU

1.a On commence par définir la fonction :

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

1.b On définit pour cela la plus petite valeur que x prend et le pas :

```
x, dx = -1.0, 0.25
```

On continue par créer un tableau point qui contiendra x et y.
On crée ensuite une boucle qui rangera dans le tableau x et y qu'il calcule grâce à la fonction f(x).
On continue par ouvrir le module maplotlib.pyplot grâce à la commande "import".

1.c la fonction g a deux racines :
x1=-0.6 x2=le nombre d'or

2.a On définit la fonction g(x).

2.b Nous faisons une boucle for permettant de calculer les termes de la suite. Cependant lorsque nous exécutions notre fichier python, nous n'arrivions pas à avoir les bons résultats. L'erreur venait du fait que nous avions mis a=1 et non a=1.0 .

2.c en exécutant notre programme on observe :

valeur courante de a = 1.61803444782
valeur courante de a = 1.6180338134
valeur courante de a = 1.61803405573
valeur courante de a = 1.61803396317
valeur courante de a = 1.61803399852
valeur courante de a = 1.61803398502
valeur courante de a = 1.61803399018
valeur courante de a = 1.61803398821
valeur courante de a = 1.61803398896
valeur courante de a = 1.61803398867
valeur courante de a = 1.61803398878

Ce qui nous montre bien la convergence de cette suite vers le nombre d'or

3.a Nous avons eu du mal mais le prof nous a aidé puis nous avons compris nos erreurs. nous avons choisis un x0=1.0 et un epsilon à 0.9 .

3.b Voici le résultat de notre premier test de la méthode du point fixe avec x0= 1.0:
test1 point_fixe :la racine de g par la méthode du point fixe est de 1.61803398875

Pour le deuxième test nous changeons la valeur de notre valeur initiale à -0.6 :
test2 point_fixe :la racine de g par la méthode du point fixe est de 1.61803398875

Comme nous pouvons le remarquer la méthode ne converge pas car les racines sont identiques.

3.c

Avec la méthode du point fixe le point qui est vers les négatifs est un point fixe répulsif, il nous envoie donc sur l'autre point fixe ce qui explique pourquoi les racines donnent le même résultat.

4.a

Comme nous le montre la courbe les points d'intersection des courbes des fonctions f et g sont les points fixes de g.

4.b

Nous créons donc la fonction newton(f, df, x0, epsi) qui repose sur une loi de récurrence où x correspond à x_{n} et x1 correspond à x_{n+1}.

4.c

Enfin, nous testons notre programme à l'aide deux différentes caractéristiques.
Nous remarquons que nous obtenons les deux racines.
test 1 newton :la racine de f a 1e-12 prés vaut r = 1.61803398875
test 2 newton :la racine de f a 1e-12 prés vaut r = -0.6180339887498948

5.a

Nous étudions cette fois ci la méthode de la sécante qui consiste à prendre une sécante de deux points de la courbe. Notre racine doit être comprise entre ses deux points. A l'aide d'une suite reposant sur la récurrence on trouvera notre racine.
On écrit donc notre fonction sécante(f,x0,x1,epsi).
Cependant une erreur apparaissait nous disant qu'on ne peut diviser par un nombre nulle. En effet le dénominateur tendait vers O. En python, tout nombre inférieur à 2/52 est considéré comme nul. On a donc du arrêter notre boucle avant que le dénominateur soit égal l'epsilon machine.

5.b

on obtient:
test 1 sécante :la racine de f a 1e-12 prés vaut r = 1.61803398875
test 2 sécante :la racine de f a 1e-12 prés vaut r = -0.6180339887498949
on retrouve bien nos deux racines.

6.a

Pour finir, utilisons la méthode de la dichotomie:
dichotomie(f, x0, x1 , epsi).
Cette fois ci nous rajoutons automatiquement un compteur pour éviter d'avoir une boucle

à l'infini.

6.b

A l'aide des tests on obtient:
test 1 dichotomie :la racine de f a 1e-12 prés vaut r = 1.61803398875
test 2 dichotomie :la racine de f a 1e-12 prés vaut r = 0.0

7.a

On rajoute cpt un compteur à chacune de nos fonctions afin de savoir laquelle est la plus rapide.
On obtient à 10^-12 prés:

test1 point_fixe :la racine de g par la méthode du point fixe est de 1.61803398875, et on a fait 30 itérations

test 1 newton :la racine de f a 1e-12 prés vaut r = 1.61803398875, et on a fait 6 itérations

test 1 sécante :la racine de f a 1e-12 prés vaut r = 1.61803398875, et on a fait 6 itérations

test 1 dichotomie :la racine de f a 1e-12 prés vaut r = 1.61803398875 , et on a fait 40 itérations

7.b

epsi=	10^-3	10^-6	10^-9	10^-12	10^-15
point fixe	9	16	23	30	38
newton	5	5	6	6	7
sécante	4	5	6	6	7
dichotomie	9	21	29	40	52

Ces résultats nous montrent que la méthode de la sécante et celle de newton sont les deux méthodes qui font le moins d’itérations.

Nehad Imane
Rahmani Abdel