Informatique appliqué aux mathématiques
Mr. Cardinal

Compte Rendu TP n°1 : Résolution numérique

SRIVASTAVA Shakul-Raman
MARADEI Clément

Année 2017/2018

Introduction :

Ce TP a été pour mon binôme et moi notre premier exercice en Python. L'occasion pour nous de nous familiariser avec le langage une première fois. L'objectif de cet exercice était la résolution numérique d'une équation et plus particulièrement trouver les racines d'une fonction relativement simple. Plusieurs méthodes étaient à notre disposition et nous avons dû expérimenter avec chacune d'entre elle.
Dans ce compte rendu, nous allons voir dans un premier temps en quoi consistait chacune des méthodes pour ensuite faire part de nos difficultés et réussites durant ce travail.

Méthodes :

1) Point Fixe

On transforme notre équation de départ f(x)=0 sous la forme g(x)=x. On choisit un réel a tel que *g(a)=a.*
On choisit ensuite un xo proche de a et on calcule par récurrence les termes de la suite x(n+1) = g(xn).
Si la valeur absolue de g(a) est inférieure à 1 et si xo est suffisamment proche de a alors la suite converge vers a.

2) Newton

La méthode de Newton suit un principe itératif qui a pour but de construire une suite d'approximations.
On choisit xo proche de la racine recherchée puis on calcule par récurrence les termes de la suite x(n+1) = x(n) = -f(x(n))/f'(x(n)).
La suite converge vers la racine recherchée.

3) Sécante

Le principe est similaire à la méthode de Newton.
On chosit x0 et x1 proches de la racine recherchée. On calcule ensuite par récurrence les termes de la suite

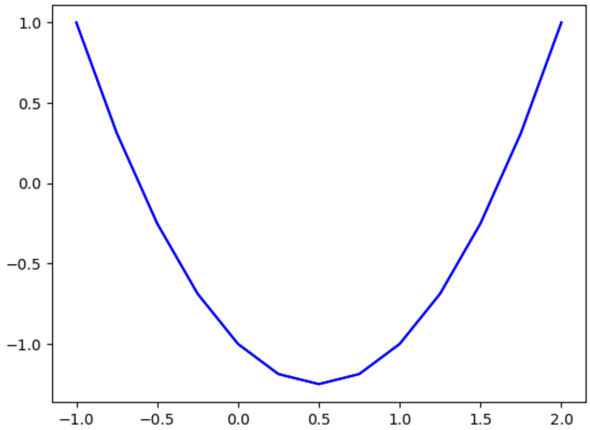
4) Dichotomie

La méthode de dichotomie suppose que f est continue sur l'intervalle (a,b) et change de signe sur cet intervalle ; on est donc assuré que f possède un zéro sur cet intervalle. Ensuite on coupe (a,b) en deux et on garde celui des deux intervalles où f change de signe ; On répète l'opération jusqu'à obtenir la précision souhaitée.

Commentaires sur l'exercice :

Question 1.

Nous n'avons fait face à aucun problème et avons pu répondre à cette question très rapidement. Voici le graphe attendu dans la partie b) :



On estime à 1,6 et -0,6 les deux zéros de f. Mr. Cardinal nous a montré comment enregistré un graphe à l'aide de la commande plt.savefig.

Question 2.

Même chose pour la question 2. Toutes les réponses attendues sont dans le fichier code. (ligne 13)

Question 3.

La fonction a pu être définie sans soucis grâce à l'aide de Mr. Cardinal qui nous a orienté durant la rédaction du code. Les tests ont été effectués et ont fonctionné sans soucis. (ligne 43)

Question 4.

Pour la fonction demandée, une difficulté s'est faite ressentir. Au lieu d'initialiser la variable i à 1, nous l'avons initialisé à 0. De cette façon le programme ne parcourait pas la boucle. Nous n'avons pas pu testé correctement le programme ; nous avons perdu du temps de cette façon. (ligne 64)

Question 5.

La méthode étant similaire à celle de Newton, la rédaction du code ne nous a pas posé davantage de problèmes. Quelques discussions avec nos camarades nous ont permis d'y voir plus clair par rapport à la rédaction de notre boucle principale qui fonctionne le même principe que celui utilisé précédemment. Les tests on également pu être réalisés sans soucis. (ligne 88)

Question 6.

Nous avons réfléchi un moment sur une manière de définir la fonction. Nous avons d'abord pris la peine d'élaborer un algorithme sur papier en le faisant « tourner » à la main. Nous avons ensuite effectué quelques tests sur ordinateur. A l'aide d'une recherche internet nous avons pu trouver la fonction 'abs' (valeur absolue) indispensable à l'élaboration de notre fonction. Nous avons gardé le même principe que les questions précédentes à savoir une boucle while et une variable nous permettant de sortir de celle ci. Lors des tests nous avons été confrontés à un problème. Effectivement une erreur d'indentation empêchait le programme de fonctionner correctement. Après avoir demandé de l'aide à Mr. Cardinal le problème a été résolu. (ligne 110)

Question 7.

Pour cette question nous avons crée différentes variables qui prennent la valeur d'une fonction avec des paramètres epsilon différents. De cette façon nous avons pu organiser en tableau faisant apparaître le nombre d'itération de chaque fonction suivant le epsilon choisi. Évidemment plus le epsilon est petit plus le nombre d'itération est grand et ce pour chaque fonction. Nous avons affiché le tableau dans le terminal. Le code se trouve à la fin du fichier.

Conclusion

Ce TP nous a permis d'apprivoiser le langage python et de nous familiariser avec ses quelques spécificités. Auparavant nous étions habitués à l'utilisation du langage C. Pour cette raison nous avons eu quelques problèmes avec l'indentation et la syntaxe en général. Nous considérons, mon binôme et moi la difficulté de ce TP comme moyenne. Les questions posées attendaient un niveau de réflexion inférieur à ce qui était demandé l'année dernière. La méthode de résolution était sensiblement la même pour chaque fonction demandée. Aucun nouveau concept informatique ne nous a été enseigné durant ces travaux. Les compétences demandées étaient dans une mesure conséquente mathématiques. Finalement nous sommes satisfait des connaissances acquises au cours de cet exercice.