Solutions approchées d'équations du type $f(x)=0$

Soit f la fonction définie sur $[1;3]$ par $f(x)=x^3-2x-5$.

Le but de la séances est d'étudier différentes méthodes pour trouver des valeurs approchées des solutions de $f(x)=0$.

Vous pouvez utiliser géogébra ou la calculatrice pour faire les constructions, Python pour les programmes.

Tout d'abord, étudier les variations de f , expliquer pourquoi l'équation $f(x)=0$ admet une unique solution sur [1;3].

1) Méthode par balayage :

La première idée peut être de faire un tableau de valeurs bien choisi pour obtenir une valeur approchée à 10^{-2} près. Le faire à l'aide de la calculatrice ou du tableur de géogébra.

Réponse :

On considère le programme python ci-contre :

```
def f(x) :
        return x**3-2x-5
def resolbalay(p) :
        x=1
        while f(x)<0 :
                x=x+p
        return x
```

Faire tourner ce programme et vérifier à l'aide des résultats précédents.

Comment peut-on faire pour connaître le nombre d'étapes que l'ordinateur a effectuées ? L'ajouter sur le programme et relancer l'exemple précédent.

Réponses :

2) Méthode par dichotomie :

La première idée demande de calculer toutes les valeurs une à une. Si l'intervalle est grand ou la précision très petite, cela demande beaucoup de calculs, donc de temps de travail pour l'ordinateur.

L'idée suivante est de couper à chaque étape l'intervalle en deux parties égales. Comment peut-on tester dans quelle moitié d'intervalle se trouve alors la solution ?

Programmer l'algorithme ci contre et comparer avec les résultats précédents. Comment peut-on faire pour connaître le nombre d'étapes que l'ordinateur a effectuées ? L'ajouter sur le programme et relancer l'exemple précédent.

```
a=1
b=3
Tant que b-a>p faire m=(a+b) /2
        Si f(m)>0 Alors b=m
        Sinon a=m
        fin si
Fin tant que
Afficher a et b
```

Réponses :

3) Méthode de Newton :

Toujours à la recherche d'une méthode encore plus rapide, Newton (vers 1670) a l'idée de partir d'une première valeur a et de suivre la tangente à la courbe représentative de f au point d'abscisse a jusqu'au point d'intersection avec l'axe des abscisse.

Ce point est logiquement assez proche de l'intersection de la courbe de la fonction et de l'axe des abscisses. (Faire une représentation graphique avec a=3 pour bien comprendre le principe)

On pose $a_0=3$ et A_1 le point de la courbe d'abscisse a_0 , T_1 la tangente en A_1 et a_1 le point d'intersection de T_1 avec l'axe des abscisses.

En utilisant l'équation de T_1 , montrer que : $a_1=a_0-\dfrac{f(a_0)}{f'(a_0)}$

On reproduit de proche en proche le procédé et on défini la suite (a_n) par $a_0=3$

et pour tout $n\in\mathbb{N}$, $a_{n+1}=a_n-\dfrac{f(a_n)}{f'(a_n)}$

Construire une fonction python permettant de calculer le terme a_n en fonction du rang n .

Soit $p>0$. Modifier la fonction précédente de façon à afficher le premier rang n

tel que $|a_{n-1}-a_n|<p$.

Comme pour les autres exemples, trouver le nombre d'étapes effectuées par l'ordinateur.

Réponses :