DS3

Devoir surveillé du 07/11/2023

Durée : 0h45

La qualité de la rédaction, la clarté et la précision des raisonnements interviendront pour une part importante dans l'appréciation des copies. Les résultats doivent être encadrés.
La calculatrice n'est pas autorisée.

Exercice 1

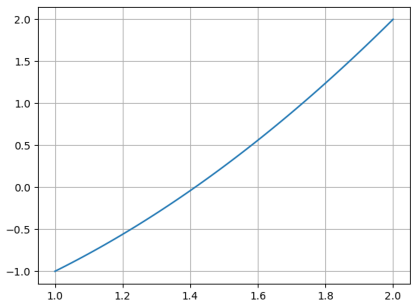
Dans cet exercice, on propose deux méthodes algorithmiques pour le calcul de la racine carrée de 2. On propose dans une dernière partie un algorithme de calcul d'une racine carrée d'une matrice.

Méthode par balayage

On considère la fonction $f : x \in [1,2] \mapsto x^2 - 2$.

1. Rappeler la commande permettant d'importer la librairie `numpy`.

   On suppose dans la suite avoir importé les librairies `numpy` et `matplotlib.pyplot` à l'aide des préfixes `np` et `plt` respectivement.



2. Écrice un script, incluant la définition de la fonction f par l'en-tête `def f(x)`, qui renvoie la courbe représentative de la fonction f sur l'intervalle $[1,2]$.

   Après exécution, ce script génère le graphe ci-contre.

   On constate que f est strictement croissante sur $[1,2]$, et s'annule en changeant de signe en $\sqrt{2}$.

3. On considère le script suivant :

```python
def balayage(p):
    v = np.linspace(1,2,10**p+1)
    i = 0
    while f(v[i])<0 :
        i=i+1
    return v[i]
```

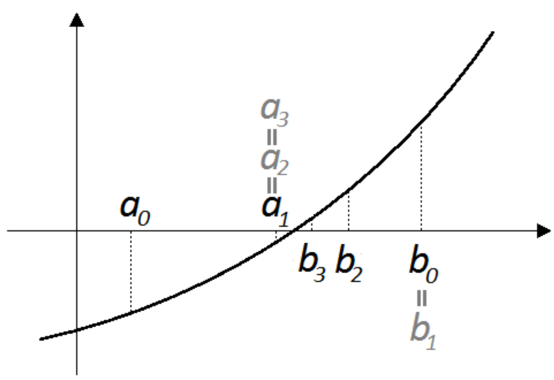
   Que renvoie la commande balayage(3) ? On justifiera sa réponse en expliquant le fonctionnement du script ci-dessus.

Méthode de dichotomie

Rappelons la méthode de dichotomie.

On construit une suite d'intervalles $[a_n, b_n]$ contenant $\sqrt{2}$. Pour cela, on définit trois suites $(a_n)_{n\in\mathbb{N}}$, $(b_n)_{n\in\mathbb{N}}$, $(m_n)_{n\in\mathbb{N}^*}$ par $a_0 = 1$, $b_0 = 2$ et pour tout $n \in \mathbb{N}$, $m_{n+1} = \dfrac{a_n + b_n}{2}$ et

- si $f(a_n)f(m_{n+1}) \leq 0$, alors $\begin{cases} a_{n+1} = a_n \\ b_{n+1} = m_{n+1} \end{cases}$,
- si $f(m_{n+1})f(b_n) \leq 0$, alors $\begin{cases} a_{n+1} = m_{n+1} \\ b_{n+1} = b_n \end{cases}$.



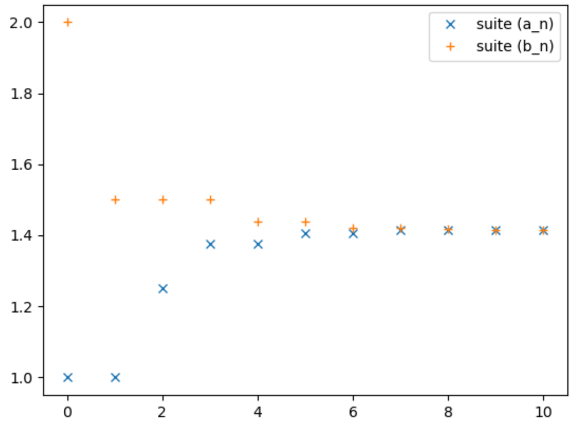
4. Compléter la fonction suivante, qui prend comme paramètre d'entrée un entier n, et qui renvoie les valeurs de a_n et b_n.

```python
def suites(n):
        a = 1
        b = 2
        for ...... :
                m = ......
                if ...... :
                        b = m
                else :
                        a = ......
        return a, b
```

5. On souhaite étudier le comportement des suites (a_n) et (b_n), ou tout du moins de leurs premiers termes.

   Écrire pour cela un script renvoyant un graphique sur lequel sont représentés les points (n, a_n) et (n, b_n) pour $n \in [\![0, 10]\!]$.

   On obtient le graphe ci-contre.

   Les suites (a_n) et (b_n) semblent adjacentes, et converger vers la même limite ℓ. On admettra dans la suite que $\ell = \sqrt{2}$, et que pour tout $n \in \mathbb{N}$:

   $$a_n \leq \sqrt{2} \leq b_n \quad \text{et} \quad |\sqrt{2} - a_n| \leq |b_n - a_n|.$$

6. Écrire une fonction d'en-tête `def dichotomie(eps)`, qui prend comme paramètre d'entrée un réel $\varepsilon > 0$, et qui renvoie une valeur approchée de $\sqrt{2}$ à ε près.

7. En modifiant la fonction précédente, proposer une fonction d'en-tête `def etapes(p)` permettant de calculer le nombre d'étapes nécessaires afin d'obtenir p décimales de $\sqrt{2}$ par l'algorithme de dichotomie.

Algorithme de calcul d'une racine carrée d'une matrice

On considère la matrice :

$$A = \begin{pmatrix} 2 & 1 & \dots & 1 \\ 1 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & 1 \\ 1 & \dots & 1 & 2 \end{pmatrix} \in \mathscr{M}_{10}(\mathbb{R}).$$

8. Sans rentrer les coefficients un à un, déclarer la matrice A.

9. On suppose avoir importé la librairie `numpy.linalg` à l'aide du préfixe `al`. Proposer une commande `Python` permettant de s'assurer que la matrice A est inversible.

10. Rappelons qu'une matrice $R \in \mathscr{M}_{10}(\mathbb{R})$ est appelée *racine carrée* de A si $R^2 = A$.

    On considère les suites matricielles (Y_n) et (Z_n) définies par $Y_0 = A$ et $Z_0 = I_{10}$, et pour tout $n \geq 1$:

    $$Y_{n+1} = \frac{1}{2}(Y_n + Z_n^{-1}) \quad \text{et} \quad Z_{n+1} = \frac{1}{2}(Z_n + Y_n^{-1}).$$

    On admet que la suite (Y_n) converge vers une racine carré de A.

    Écrire une fonction d'en-tête `def suite(n)` qui prend comme paramètre d'entrée un entier n et renvoie la matrice Y_n.

11. Proposer une commande permettant de calculer $(Y_{2000})^2$, et ainsi de vérifier la pertinence du résultat renvoyé par la fonction `suite`.