
DS2 (version A)

La présentation, la lisibilité, l'orthographe, la qualité de la rédaction, la clarté et la précision des raisonnements entreront pour une part importante dans l'appréciation des copies.

*Les candidat·es sont invité·es à **encadrer** dans la mesure du possible les résultats de leurs calculs.*

*Aucun document n'est autorisé. **L'utilisation de toute calculatrice et de tout matériel électronique est interdite.** Seule l'utilisation d'une règle graduée est autorisée.*

Si au cours de l'épreuve, un candidat ou une candidate repère ce qui lui semble être une erreur d'énoncé, il la signalera sur sa copie et poursuivra sa composition en expliquant les raisons des initiatives qu'il sera amené à prendre.

On suppose, et c'est valable pour toute l'épreuve, que les librairies suivantes sont importées sous leurs alias habituels :

- `import numpy as np`
- `import numpy.linalg as al`
- `import numpy.random as rd`
- `import matplotlib.pyplot as plt`

Exercice 1

On considère un nombre réel a et on pose $M_a = \begin{pmatrix} 1 & 0 & 0 \\ 1-a & a & 0 \\ 0 & 1-a & a \end{pmatrix}$.

1. Déterminer une condition nécessaire et suffisante sur a pour que la matrice M_a soit inversible.

A partir de maintenant et ce, **jusqu'à la fin de l'exercice**, on suppose que a est un élément de $]0, 1[$.

2. a) Déterminer une base et la dimension de $E_1(M_a) = \{U \in \mathcal{M}_{3,1}(\mathbb{R}) \mid M_a U = U\}$.

b) Déterminer une base et la dimension de $E_a(M_a) = \{U \in \mathcal{M}_{3,1}(\mathbb{R}) \mid M_a U = aU\}$.

c) (CUBES UNIQUEMENT) En déduire que M_a n'est pas diagonalisable.

3. On pose $I = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$ et on note E l'espace vectoriel engendré par I , M_a et M_a^2 .

a) Quelle est la dimension de E ?

b) On pose $J = \begin{pmatrix} 0 & 0 & 0 \\ 1 & -1 & 0 \\ 0 & 1 & -1 \end{pmatrix}$ et $K = \begin{pmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}$.

Calculer JK^2 puis en déduire $(M_a - I)(M_a - aI)^2$.

c) En déduire que M_a^3 appartient à E .

4. a) Montrer que, pour tout entier naturel n , il existe un unique triplet de réels (u_n, v_n, w_n) tel que :

$$M_a^n = u_n M_a^2 + v_n M_a + w_n I$$

On donnera les valeurs de u_0 , v_0 et w_0 et on écrira les relations liant u_{n+1} , v_{n+1} , w_{n+1} à u_n , v_n et w_n .

b) En utilisant les relations précédentes, expliquer pourquoi le script **Python** qui suit ne permet pas de calculer et d'afficher les valeurs de u_n , v_n et w_n lorsque n et a sont entrés par l'utilisateur. On pourra examiner attentivement la boucle « for ».

```

1  n = input('entrez une valeur pour n : ')
2  a = input('entrez une valeur pour a : ')
3  u = 0
4  v = 0
5  w = 1
6  for k in range(n):
7      u = (2 * a + 1) * u + v
8      v = -a * (a + 2) * u + w
9      w = a * a * u
10 print(w, v, u)

```

c) Modifier la boucle de ce script en conséquence.

5. Démontrer : $\forall n \in \mathbb{N}, u_{n+3} = (2a + 1)u_{n+2} - a(a + 2)u_{n+1} + a^2 u_n$.

On **admet** que l'on peut en déduire u_n , pour tout entier naturel n , sous la forme :

$$u_n = \frac{(n-1)a^n - na^{n+1} + 1}{(a-1)^2}$$

6. On dit qu'une suite de matrices $(A_n)_{n \in \mathbb{N}}$ tend vers la matrice A lorsque n tend vers $+\infty$ si chaque coefficient de A_n tend vers le coefficient situé à la même place dans A . Il en résulte (et on admet ce résultat) :

$$\lim_{n \rightarrow +\infty} M_a^n = \left(\lim_{n \rightarrow +\infty} u_n \right) M_a^2 + \left(\lim_{n \rightarrow +\infty} v_n \right) M_a + \left(\lim_{n \rightarrow +\infty} w_n \right) I$$

- a) Déterminer $\lim_{n \rightarrow +\infty} u_n$, puis $\lim_{n \rightarrow +\infty} v_n$ et $\lim_{n \rightarrow +\infty} w_n$.
- b) En déduire la limite L_a lorsque n tend vers $+\infty$, de la suite $(M_a^n)_{n \in \mathbb{N}}$.
- c) Vérifier : $L_a^2 = L_a$.
7. (CUBES UNIQUEMENT) On note φ_a l'endomorphisme de $\mathbb{R}^3$ dont la matrice dans la base canonique de $\mathbb{R}^3$ est L_a et f_a l'endomorphisme de $\mathbb{R}^3$ dont la matrice dans la base canonique de $\mathbb{R}^3$ est M_a .
- a) Démontrer : $\forall x \in \text{Ker}(f_a - \text{id}), \varphi_a(x) = x$.
- b) Démontrer : $\forall x \in \text{Im}(f_a - \text{id}), \varphi_a(x) = 0$.

Exercice 2

On considère la fonction $f : x \mapsto \sqrt{\frac{1+x}{2}}$ et la suite (u_n) définie par

$$\begin{cases} u_0 \geq 0 \\ \forall n \in \mathbb{N}, u_{n+1} = f(u_n) \end{cases}$$

Partie A : Étude de la fonction f (on note $\mathcal{C}_f$ la courbe représentative de f)

- Déterminer le domaine de définition de f puis les limites de f aux bords de ce domaine.
- (CUBES UNIQUEMENT) Déterminer le développement limité de f en 0 à l'ordre 2.
- Montrer que la fonction f est dérivable sur $] -1, +\infty[$, calculer sa dérivée puis expliciter l'équation de la tangente à $\mathcal{C}_f$ en 0.
- Étudier les variations de f et dresser son tableau de variations.
- Résoudre l'équation $f(x) = x$ puis l'inéquation $f(x) > x$.
- Tracer la courbe $\mathcal{C}_f$.

Partie B : Étude de la suite (u_n) dans le cas où $u_0 = 0$

- Montrer que la suite (u_n) est bien définie et que, pour tout $n \in \mathbb{N}$, $0 \leq u_n \leq 1$.
- Montrer que la suite (u_n) est croissante.
- Montrer que la suite (u_n) est convergente et déterminer sa limite.
- Écrire une fonction **Python**, nommée **SuiteU**, qui prend en paramètre un entier n et renvoie u_n .
- Écrire une fonction **Python**, nommée **PremierEntier**, qui prend en paramètre un réel **eps** strictement positif et renvoie le premier entier n vérifiant $1 - \text{eps} < u_n \leq 1$.

Partie C : Étude de la suite (u_n) dans le cas où $u_0 > 1$

- Montrer que la suite (u_n) est bien définie.
- Montrer que la suite (u_n) est monotone.
- Étudier la convergence de la suite (u_n) et déterminer sa limite.

Partie D : Étude de fonctions auxiliaires

Dans **toute la suite de l'exercice** on suppose que $u_0 > 1$.

On définit les fonctions ch et sh sur $\mathbb{R}$ par

$$\text{ch}(x) = \frac{e^x + e^{-x}}{2}, \quad \text{et} \quad \text{sh}(x) = \frac{e^x - e^{-x}}{2}.$$

15. Exprimer les dérivées des fonctions ch et sh en fonction de ch et sh .

Montrer que, pour tout $x \in \mathbb{R}$, $\text{ch}(x) > 0$.

Calculer $\text{sh}(0)$ et déterminer le signe de $\text{sh}(x)$ en fonction de x .

16. En déduire qu'il existe un unique réel $\alpha \geq 0$ tel que $\text{ch}(\alpha) = u_0$.

17. On considère le programme **Python** suivant

```
1  def ch(x):
2      return (np.exp(x)+np.exp(-x))/2
3
4  u0=3/2
5  a=0
6  b=2
7  c=(a+b)/2
8  while b-a > 10**(-3):
9      if (ch(a)-u0)*(ch(c)-u0) < 0:
10         b=c
11     else:
12         a=c
13     c=(a+b)/2
14 print(c)
```

a. Que fait ce programme ? Comment s'appelle ce type de programme ?

b. Pourquoi a-t-on pris $b = 2$?

18. Montrer que, pour tout $x \in \mathbb{R}$,

$$2 \left(\text{ch} \left(\frac{x}{2} \right) \right)^2 - 1 = \text{ch}(x)$$

19. En déduire que, pour tout entier n ,

$$u_n = \text{ch} \left(\frac{\alpha}{2^n} \right)$$

20. Montrer que, pour tout $x \in \mathbb{R}$,

$$\text{ch}(x) - 1 = 2 \left(\text{sh} \left(\frac{x}{2} \right) \right)^2$$

21. Calculer $\text{sh}'(0)$.

22. En déduire les équivalences suivantes

$$\text{sh}(x) \underset{x \rightarrow 0}{\sim} x, \quad \text{et} \quad \text{ch}(x) - 1 \underset{x \rightarrow 0}{\sim} \frac{x^2}{2}$$

23. En déduire un équivalent de $(u_n - 1)$ quand n tend vers $+\infty$.

Exercice 3

Une urne contient initialement une boule blanche et une boule noire. On effectue une succession infinie de tirages d'une boule dans cette urne. Après chaque tirage, on remet la boule tirée dans l'urne, et on rajoute dans l'urne une boule de couleur opposée à celle qui vient d'être tirée.

Pour tout $k \in \mathbb{N}$, on note X_k la variable aléatoire égale au nombre de boules blanches présentes dans l'urne juste après le k^{e} tirage. En particulier, on a $X_0 = 1$.

Pour tout $i \in \mathbb{N}^*$, on note :

B_i : « on obtient une boule blanche au i^{e} tirage »

N_i : « on obtient une boule noire au i^{e} tirage »

1. Compléter la fonction **Python** ci-dessous afin qu'elle simule la variable aléatoire X_n .

```

1  def Simu_X(n):
2      nB, nN = 1, 1
3      for k in range(n) :
4          if _____:
5              nN = nN + 1
6          else :
7              _____
8      return _____

```

2. Déterminer la loi de X_1 . Donner son espérance et sa variance. (*On pourra utiliser les événements B_1 et N_1 pour rédiger la réponse.*)

3. a. Justifier soigneusement que la loi de X_2 est donnée par :

$$\mathbb{P}([X_2 = 1]) = \frac{1}{6}, \quad \mathbb{P}([X_2 = 2]) = \frac{2}{3}, \quad \mathbb{P}([X_2 = 3]) = \frac{1}{6}$$

- b. En déduire la valeur de $\mathbb{E}(X_2)$.

- c. On rappelle qu'après avoir chargé la bibliothèque `matplotlib.pyplot` sous l'alias `plt`, on a accès à la fonction `plt.hist` qui trace l'histogramme d'une liste ou d'un tableau donné en argument. Représenter précisément, en justifiant la réponse, la figure que l'on peut s'attendre à ce que **Python** affiche à l'exécution des instructions suivantes

```

1  L = []
2  for k in range(10000):
3      L.append(Simu_X(2))
4  plt.hist(L, range(1,5), density = True)
5  plt.show()

```

4. Soit $k \in \mathbb{N}$. Préciser l'ensemble $X_k(\Omega)$ des valeurs que peut prendre X_k .

5. Soient $i \in \mathbb{N}^*$ et $j \in X_k(\Omega)$. En distinguant trois cas, déterminer

$$\mathbb{P}_{[X_k=j]}([X_{k+1} = i]).$$

6. Dédurre de ce qui précède que :

$$\forall k \in \mathbb{N}, \forall i \in \mathbb{N}^*, \mathbb{P}([X_{k+1} = i]) = \frac{i}{k+2} \mathbb{P}([X_k = i]) + \frac{3+k-i}{k+2} \mathbb{P}([X_k = i-1]) \quad (*)$$

7. À l'aide de la formule (*) déterminer la loi de X_3 .

8. a. Montrer par récurrence que, pour tout $k \in \mathbb{N}$,

$$\mathbb{P}([X_k = 1]) = \frac{1}{(k+1)!}.$$

b. Montrer de même que, pour tout $k \in \mathbb{N}$,

$$\mathbb{P}([X_k = k+1]) = \frac{1}{(k+1)!}.$$

c. Pour tout $k \in \mathbb{N}$, on pose : $a_k = (k+1)! \times \mathbb{P}([X_k = 2])$.

Exprimer a_{k+1} en fonction de a_k et de k .

d. Pour tout $k \in \mathbb{N}$, on pose : $b_k = a_k + k + 2$. Montrer que la suite (b_k) est géométrique. En déduire alors que :

$$\forall k \in \mathbb{N}, \mathbb{P}([X_k = 2]) = \frac{2^{k+1} - k - 2}{(k+1)!}$$

9. a. À l'aide de la formule (*), montrer que :

$$\forall k \in \mathbb{N}, \mathbb{E}(X_{k+1}) = \frac{k+1}{k+2} \mathbb{E}(X_k) + 1$$

b. Déduire de ce qui précède que :

$$\forall k \in \mathbb{N}, \mathbb{E}(X_k) = \frac{k+2}{2}$$