Exercice 1. *Matrices, probabililités et simulation informatique*

Les parties ne sont pas indépendantes.

Partie I. *Calculs matriciels*

On considère les matrices $B = \begin{pmatrix} 3 & 2 & 0 \\ 3 & 2 & 3 \\ 0 & 2 & 3 \end{pmatrix}$, $P = \begin{pmatrix} 1 & -1 & 2 \\ -2 & 0 & 3 \\ 1 & 1 & 2 \end{pmatrix}$, $Q = 5I_3 + 3P - P^2$

et $M = \dfrac{1}{6}B$.

1. (a) Expliciter les coefficients de Q. On rappelle que I_3 est la matrice identité de $\mathscr{M}_3(\mathbb{R})$.
   (b) Effectuer le produit matriciel PQ et montrer que celui-ci peut s'écrire $\lambda_0 I_3$ pour un certain $\lambda_0 \in \mathbb{R}^*$.
   (c) Montrer que P est inversible et exprimer P^{-1} en fonction de Q. Expliciter les coefficients de P^{-1}.
2. On considère la matrice $C = P^{-1}BP$.
   (a) Expliciter les coefficients de C et vérifier qu'il s'agit d'une matrice diagonale.
   (b) Justifier que C est inversible et expliciter C^{-1}. Exprimer B en fonction de P, P^{-1} et C. Justifier que B est inversible et exprimer B^{-1} en fonction de P, P^{-1} et C^{-1}. Expliciter les coefficients de B^{-1}.
   (c) Expliciter les coefficients de C^n pour tout $n \in \mathbb{N}$. Exprimer B^2 en fonction de P, P^{-1}, C^2 et conjecturer une expression de B^n en fonction de P, P^{-1}, C^n puis démontrer cette relation pour tout $n \in \mathbb{N}$.
   (d) Justifier que M est inversible puis expliciter les coefficients de M^{-1}. Exprimer M^n en fonction de P, P^{-1}, C^n et n pour tout $n \in \mathbb{N}$.
   (e) Vérifier que $M^n = \dfrac{1}{14}\begin{pmatrix} 3\alpha_n + 7\beta_n + 4 & -4\alpha_n + 4 & 3\alpha_n - 7\beta_n + 4 \\ -6\alpha_n + 6 & 8\alpha_n + 6 & -6\alpha_n + 6 \\ 3\alpha_n - 7\beta_n + 4 & -4\alpha_n + 4 & 3\alpha_n + 7\beta_n + 4 \end{pmatrix}$ pour tout $n \in \mathbb{N}$,
   avec $\alpha_n = \left(-\frac{1}{6}\right)^n$ et $\beta_n = \left(\frac{1}{2}\right)^n$. Vérifier que cette relation est également vraie pour $n = -1$.

Partie II. *Un jeu à trois*

Astrid, Bilal et Candice jouent à un jeu qui consiste à lancer un dé équilibré un nombre indéfini de fois en respectant les règles suivantes :

- Si Astrid lance le dé et obtient :
  — 1, 2 ou 3 alors elle garde le dé pour le lancer suivant.
  — 4, 5 ou 6 alors elle donne le dé à Bilal pour le lancer suivant.
- Si Bilal lance le dé et obtient :
  — 1 ou 2 alors il donne le dé à Astrid pour le lancer suivant.
  — 3 ou 4 alors il garde le dé pour le lancer suivant.
  — 5 ou 6 alors il donne le dé à Candice pour le lancer suivant.
- Si Candice lance le dé et obtient :
  — 1, 2 ou 3 alors elle garde le dé pour le lancer suivant.
  — 4, 5 ou 6 alors elle donne le dé à Bilal pour le lancer suivant.

On note :
A_n : “Astrid effectue le $n^{\text{ème}}$ lancer” et $a_n = \mathbb{P}(A_n)$.
B_n : “Bilal effectue le $n^{\text{ème}}$ lancer” et $b_n = \mathbb{P}(B_n)$.
C_n : “Candice effectue le $n^{\text{ème}}$ lancer” et $c_n = \mathbb{P}(C_n)$.

1. Que peut-on dire de la famille (A_n, B_n, C_n) ? Qu'en déduit-on pour a_n, b_n et c_n ?
2. Établir la relation $a_{n+1} = \dfrac{1}{2}a_n + \dfrac{1}{3}b_n$ pour tout $n \in \mathbb{N}^*$. Déterminer de la même façon une relation entre b_{n+1} et a_n, b_n, c_n puis une relation entre c_{n+1} et a_n, b_n, c_n.
3. On pose $X_n = \begin{pmatrix} a_n \\ b_n \\ c_n \end{pmatrix}$. Déterminer une relation matricielle entre X_{n+1}, X_n et M où M est la matrice introduite dans la première partie.
4. Démontrer que $\forall n \in \mathbb{N}^*,\ X_n = M^{n-1}X_1$.
5. Déterminer les coefficients de M^{n-1} que l'on doit utiliser pour le calcul de b_n puis exprimer b_n en fonction de $a_1, b_1, c_1, \alpha_{n-1}$. On rappelle que $\alpha_n = \left(-\frac{1}{6}\right)^n$.
   Vérifier que l'une des expressions de b_n est $\frac{1}{7}\Big((7b_1 - 3)\alpha_{n-1} + 3\Big)$.
6. Calculer $\lim\limits_{n\to+\infty} b_n$. Cette limite dépend-elle de a_1, b_1, c_1 ? Donner une interprétation de ces résultats.
7. Montrer que $\lim\limits_{n\to+\infty} \mathbb{P}(B_n) - \mathbb{P}(B_n | B_{n-1}) = \dfrac{2}{21}$.
   Qu'en déduit-on à partir d'un certain rang pour l'indépendance des événements B_n et B_{n-1} ?
8. Dans cette question on suppose que $a_1 = c_1 = \frac{2}{7}$. Calculer b_1, déterminer X_2 puis X_3 et conjecturer l'expression de X_n pour tout $n \in \mathbb{N}^*$. Démontrer cette conjecture. Que peut-on dire des suites $(a_n)_{n\in\mathbb{N}^*}, (b_n)_{n\in\mathbb{N}^*}$ et $(c_n)_{n\in\mathbb{N}^*}$? Interpréter ce résultat.
9. Dans cette question on suppose que $b_2 = \frac{1}{2}$ et $a_2 = c_2$.
   Calculer a_2 et c_2, en déduire X_2 puis déterminer a_1, b_1 et c_1.
10. Dans cette question on suppose que le premier lanceur a été choisi au hasard.
    (a) Établir que $b_2 = \frac{4}{9}$.
    (b) Si Bilal a effectué le deuxième lancer, quelle est la probabilité qu'il ait aussi effectué le premier ?
    (c) B_1 et B_2 sont-ils indépendants ?

(d) i. Calculer la probabilité que le résultat du premier lancer soit 6 et que Bilal ait effectué le deuxième lancer.

ii. En déduire la probabilité que le résultat du premier lancer soit 6 si l'on sait que Bilal a effectué le deuxième lancer.

11. Calculer la probabilité que Bilal ait effectué les n premiers lancers. Déterminer à partir de quel entier cette probabilité est strictement inférieure à 10^{-9}. On pourra utiliser les fonctions $x \mapsto \lfloor x \rfloor$ et ln.

Partie III. *Simulation*

Dans cette partie, on examine deux protocoles qui respectent les règles du jeu définies à la deuxième partie.
On conviendra qu'Astrid, Bilal et Candice sont respectivement caractérisés par les numéros 0, 1 et 2.
On dispose des fonctions des bibliothèques *random, matplotlib.pyplot, numpy.*

1. Compléter la fonction Python *premier* de paramètres $a1, b1$ pour qu'elle renvoie le numéro du premier lanceur sachant que les probabilités initiales correspondent à $a1$, $b1$ et $1 - a1 - b1$.

   On rappelle que si $0 \leqslant u \leqslant v \leqslant 1$ alors :
   - "L'événement" $rd.random() < v$ se réalise avec la probabilité v.
   - "L'événement" $u <= rd.random() < v$ se réalise avec la probabilité $v - u$.
   - "L'événement" $u <= rd.random()$ se réalise avec la probabilité $1 - u$.

```
def premier(a1, b1):
    g = rd.random()
    if ________:
        return __
    elif __________:
        return __
    else:
        return __
```

2. Dans cette question on suppose que le dé est lancé exactement 10 fois et on s'intéresse au nombre de lancers effectués par Astrid.

   (a) Compléter la fonction *simulation1* de paramètres $a1, b1$ de sorte qu'elle renvoie le nombre de lancers effectués par Astrid.

```
def simulation1(a1, b1):
    compteurA = __ # à la fin, compteurA contiendra le nombre de lancers d'Astrid
    joueur = ________________ # joueur contient le numéro du premier lanceur
    for k in range(____):
        de = rd.randint(__, __) # on simule un lancer de dé
        if joueur == __:
            compteurA ________
            if de >= __:
                joueur = __
        elif joueur == __:
            if de <= __:
                joueur = __
            elif de >= __:
                joueur = __
        else:
            if de >= __:
                joueur = __
    return __________
```

   (b) Compléter la fonction *courbe* de paramètres $q, a1, b1$ qui affiche la courbe (plot) de la liste des probabilités des événements : "Astrid lance exactement i fois le dé " pour i variant de 0 à 10. Ces 11 probabilités étant approchées à l'aide de q simulations.

   On rappelle que la fonction *zeros* de paramètre n renvoie un tableau *numpy* (type similaire à une liste) avec n zéros. Les opérations algébriques $(+, -, *, /)$ entre un nombre et un tableau se répercutent sur chaque terme du tableau. Par exemple, si a est le tableau $[2, 1, 3]$ alors $a/6$ est le tableau $[1/3, 1/6, 1/2]$.

   On rappelle également que si a est une liste ou un tableau à 11 éléments alors l'instruction $plt.plot(a)$ affiche la courbe qui relie par des segments de droites les points de coordonnées :
   $(0, a[0])$, $(1, a[1]), \dots, (10, a[10])$.

```
def courbe(q, a1, b1):
    tableau = np.zeros(___) # on initialise le tableau de fréquences des 11 événements
    for k in range(___):
        i = simulation1(___, ___)
        tableau[__] ________
    tableau = __________ / __ #on passe des fréquences absolues aux fréqu. relatives
    plt.plot(_________)
```

3. Dans cette question on suppose que le dé est lancé tant que le nombre de lancers d'Astrid est supérieur ou égal à celui de Bilal.

   (a) Écrire une fonction *simulation2* de paramètres $a1, b1$ qui renvoie le nombre de lancers total.

   On pourra s'inspirer de la fonction *simulation*1 de la question 2(a) et introduire deux autres compteurs : un pour le nombre de lancers de Bilal et un autre pour le nombre de lancers total.

   (b) Compléter la fonction *approx* qui renvoie une approximation de la probabilité que le nombre de lancers soit au plus égal à 5 sachant que ce nombre est au moins égal à 3 à l'aide de q simulations.

```
def approx(q, a1, b1):
    aumoins3 = __ # compteur des issues comportant au moins 3
lancers
    entre3et5 = __ # compteur des issues avec un nombre de lancers
dans ⟦3, 5⟧
    for k in range(__):
        nblancers = ______________________
        if nblancers _________:
            aumoins3 ________
            if nblancers _________:
                entre3et5 ________
    return __________ / __________
```