

Mathématiques
Devoir surveillé n°3
Novembre 2025

Durée de l'épreuve : 4h

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 candidats sont invités 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 repère ce qui lui semble être une erreur d'énoncé, il le signalera sur sa copie et poursuivra sa composition en expliquant les raisons des initiatives qu'il sera amené à prendre.

Exercice 1.

Partie I : étude d'une suite

On pose, pour tout $n \in \mathbb{N}^*$:

$$H_n = \sum_{k=1}^n \frac{1}{k}.$$

1. (a) Vérifier que la suite (H_n) est croissante.
- (b) Montrer que, pour tout $n \in \mathbb{N}^*$:

$$H_{2n} - H_n > \frac{1}{2}.$$

- (c) Justifier que la suite $(H_n)_{n \in \mathbb{N}^*}$ admet une limite $\ell \in \mathbb{R} \cup \{+\infty\}$.
- (d) Montrer que :

$$\lim_{n \rightarrow +\infty} H_n = +\infty.$$

On pourra raisonner par l'absurde et utiliser la question 1.(b).

- (e) i. Écrire une fonction qui, étant donné un entier n , permet de calculer H_n .
- ii. Recopier et compléter la fonction Python suivante pour qu'elle renvoie la valeur de l'entier n pour lequel $H_n \geq 10^5$:

```
def rang():  
    H = ...  
    n = ...  
    while ... :  
        n = ...  
        H = ...  
    return ...
```

2. On pose, pour tout $n \in \mathbb{N}^*$:

$$u_n = H_n - \ln(n+1), \quad v_n = H_n - \ln(n).$$

- (a) Justifier que, pour tout $x \in]-1, +\infty[$:

$$\ln(1+x) \leq x.$$

- (b) En déduire que, pour tout $n \in \mathbb{N}^*$:

$$\ln\left(\frac{n+2}{n+1}\right) \leq \frac{1}{n+1} \leq \ln\left(\frac{n+1}{n}\right).$$

- (c) Montrer que les suites $(u_n)_{n \geq 1}$ et $(v_n)_{n \geq 1}$ sont adjacentes. En déduire qu'elles convergent vers une même limite notée γ .

3. Une application : on pose, pour tout $n \in \mathbb{N}^*$:

$$S_n = \sum_{k=1}^{2n} \frac{(-1)^{k-1}}{k}.$$

1

(a) Montrer par récurrence que, pour tout $n \in \mathbb{N}^*$:

$$S_n = H_{2n} - H_n.$$

(b) Exprimer alors, pour tout $n \in \mathbb{N}^*$, S_n en fonction de termes de la suite $(v_n)_{n \geq 1}$ et d'une constante.

(c) En déduire que la suite (S_n) converge vers $\ln(2)$.

Partie II : formule d'inversion de Pascal

1. Soient i , k et n trois entiers naturels tels que $i \leq k \leq n$. Prouver que :

$$\binom{n}{k} \binom{k}{i} = \binom{n}{i} \binom{n-i}{n-k}.$$

2. Soit $(u_n)_{n \in \mathbb{N}}$ une suite réelle et $(v_n)_{n \in \mathbb{N}}$ la suite définie par :

$$\forall n \in \mathbb{N}, \quad v_n = \sum_{k=0}^n \binom{n}{k} u_k.$$

(a) Montrer que, pour tout $n \in \mathbb{N}$:

$$\sum_{k=0}^n (-1)^{n-k} \binom{n}{k} v_k = \sum_{i=0}^n \binom{n}{i} u_i \sum_{j=0}^{n-i} (-1)^j \binom{n-i}{j}.$$

(b) On rappelle que $0^n = 0$ pour $n \in \mathbb{N}^*$ et $0^0 = 1$. En déduire que, pour tout $n \in \mathbb{N}$:

$$u_n = \sum_{k=0}^n (-1)^{n-k} \binom{n}{k} v_k.$$

Ce résultat est connu sous le nom de **formule d'inversion de Pascal**.

Partie III : une application

1. Justifier que, pour tout $x \in \mathbb{R}^*$ et pour tout $n \in \mathbb{N}^*$:

$$\frac{1 - (1-x)^n}{x} = \sum_{k=0}^{n-1} (1-x)^k.$$

2. Soit $n \in \mathbb{N}^*$ et f_n la fonction définie sur $\mathbb{R}$ par :

$$\forall x \in \mathbb{R}, \quad f_n(x) = \sum_{k=1}^n \binom{n}{k} \frac{(-1)^k}{k} x^k.$$

(a) Montrer que, pour tout $x \in \mathbb{R}$:

$$f'_n(x) = - \sum_{k=0}^{n-1} (1-x)^k.$$

(b) En déduire que, pour tout $x \in \mathbb{R}$:

$$f_n(x) = \sum_{k=0}^{n-1} \frac{(1-x)^{k+1}}{k+1} - H_n.$$

(c) Conclure que :

$$H_n = \sum_{k=1}^n \binom{n}{k} \frac{(-1)^{k+1}}{k}.$$

3. En utilisant la formule d'inversion de Pascal, montrer que, pour tout $n \in \mathbb{N}^*$:

$$\frac{1}{n} = \sum_{k=1}^n (-1)^{k-1} \binom{n}{k} H_k.$$

Exercice 2.

Soit n un entier naturel non nul. Dans une fête foraine, un stand propose le jeu suivant : le joueur lance n fois une pièce et compte le nombre de *Pile* obtenus.

Si ce nombre est pair, le joueur est déclaré vainqueur, et s'il est impair, il est déclaré perdant. Si le joueur est déclaré vainqueur, il gagne 10 euros pour chaque *Pile* obtenu, mais s'il a perdu, il doit payer 10 euros pour chaque *Pile* obtenu.

En particulier, s'il n'obtient aucun *Pile*, il est déclaré vainqueur, mais ne remporte rien.

La pièce est truquée : à chaque lancer, la probabilité d'obtenir *Pile* est égale à p avec $p \in]0, 1[$, et celle d'obtenir *Face* est $1 - p$.

On note X la variable aléatoire égale au nombre de *Pile* obtenus, et G la variable aléatoire égale au gain algébrique du joueur.

On note A l'événement « le joueur est déclaré vainqueur » et on dit que le jeu est **favorable au joueur** si l'espérance mathématique $\mathbb{E}(G)$ est positive.

Partie I

Dans cette partie, on suppose que $n = 3$ et $p = \frac{2}{3}$.

1. Reconnaître la loi de X et vérifier que

$$\mathbb{P}(A) = \frac{13}{27}.$$

2. Montrer que

$$G(\Omega) = \{-30, -10, 0, 20\},$$

puis expliciter la loi de G .

3. Calculer l'espérance de G . Le jeu est-il favorable au joueur ?

4. Python

- (a) Recopier et compléter la fonction suivante pour qu'elle simule la loi de X . La commande `rd.rand()` retourne un nombre réel dans l'intervalle $[0, 1[$.

```
def loiX():
    x = ...
    for k in range(...):
        if rd.rand() ... :
            x = ...
    return ...
```

- (b) Recopier et compléter la fonction suivante pour qu'elle simule la loi de G .

```
def loiG():
    x = ...
    if x == 0:
        G = ...
    elif ...:
        ...
    elif ...:
        ...
    else:
        ...
    return ...
```

Partie II

On revient au cas général où $n \in \mathbb{N}^*$ et $p \in]0, 1[$.

Le tenancier souhaite afficher : « À ce jeu, il y a plus de gagnants que de perdants ! ». Il cherche donc les conditions nécessaires sur p et n pour que cet affichage ne soit pas mensonger.

On définit la variable aléatoire Y par

$$Y = (-1)^X.$$

Ainsi $Y = 1$ lorsque X est pair, et $Y = -1$ lorsque X est impair.

1. (a) On pose

$$Z = \frac{Y + 1}{2}.$$

Déterminer $Z(\Omega)$, puis montrer que Z suit une loi de Bernoulli de paramètre $\mathbb{P}(A)$.

- (b) Démontrer que

$$\mathbb{E}(Y) = 2\mathbb{P}(A) - 1.$$

2. (a) Donner la loi de X .
(b) En déduire que

$$\mathbb{E}(Y) = \sum_{k=0}^n (-1)^k \binom{n}{k} p^k (1-p)^{n-k},$$

puis que

$$\mathbb{E}(Y) = (1-2p)^n.$$

3. Exprimer alors $\mathbb{P}(A)$ en fonction de n et p .
4. Démontrer que

$$\mathbb{P}(A) \geq \frac{1}{2} \iff \left(p \leq \frac{1}{2} \text{ OU } n \text{ pair} \right).$$

Partie III

Le concepteur du jeu souhaite vérifier que, tout en rendant le jeu attractif $\left(i.e. \mathbb{P}(A) \geq \frac{1}{2} \right)$, son activité reste rentable (i.e. $\mathbb{E}(G) \leq 0$).

1. Exprimer G en fonction de X et Y . En déduire que

$$\mathbb{E}(G) = 10 \sum_{k=0}^n (-1)^k k \mathbb{P}(X = k).$$

2. Démontrer que, pour tout $k \in \llbracket 1, n \rrbracket$:

$$k \binom{n}{k} = n \binom{n-1}{k-1}.$$

3. Montrer que

$$\mathbb{E}(G) = -10np(1-2p)^{n-1}.$$

4. Démontrer que :

$$\begin{cases} \mathbb{P}(A) \geq \frac{1}{2} \\ \mathbb{E}(G) \leq 0 \end{cases} \iff p \leq \frac{1}{2}.$$

5. (a) Étudier la fonction $f : \left[0, \frac{1}{2} \right] \rightarrow \mathbb{R}$ définie par

$$f(x) = x(1-2x)^{n-1}.$$

- (b) Pour une valeur de n fixée, déterminer la valeur optimale de $p \in \left[0, \frac{1}{2} \right]$ afin de maximiser la rentabilité pour le concepteur du jeu.