MODÉLISATION MATHÉMATIQUE ET INFORMATIQUE

Durée : 3 heures

L'usage d'abaques, de tables, de calculatrice et de tout instrument électronique susceptible de permettre au candidat d'accéder à des données et de les traiter par les moyens autres que ceux fournis dans le sujet est interdit.
Chaque candidat est responsable de la vérification de son sujet d'épreuve : pagination et impression de chaque page. Ce contrôle doit être fait en début d'épreuve. En cas de doute, le candidat doit alerter au plus tôt le surveillant qui vérifiera et, éventuellement, remplacera le sujet.

Ce sujet comporte 9 pages numérotées de 1 à 9.

Si, au cours de l'épreuve, un candidat repère ce qui lui semble être une erreur d'énoncé, il le signale sur sa copie et poursuit sa composition en expliquant les raisons des initiatives qu'il a été amené à prendre.

Dans tout le problème, Ω est un ensemble, $\mathcal{T}$ est une tribu sur Ω et P est une probabilité sur $(\Omega,\mathcal{T})$. Toutes les variables aléatoires considérées sont définies sur l'univers Ω. Étant donné une variable aléatoire X à valeurs réelles admettant une espérance, on note $E(X)$ son espérance et, si X admet une variance, on note $V(X)$ sa variance.
On s'intéresse dans ce problème à la notion de *fonction génératrice.*
Pour toute variable aléatoire X à valeurs dans $\mathbb{N}$, on note G_X la fonction

$$G_X\,:\,t \mapsto \sum_{n=0}^{+\infty} P(X=n)\,t^n$$

appelée *la fonction génératrice de X.*
La partie I établit les premières propriétés nécessaires pour l'ensemble du sujet et propose des exemples importants. Les parties II, III et IV sont indépendantes.
Une annexe dans laquelle certaines commandes Python sont rappelées est jointe à la fin du sujet. Pour les questions d'informatique, on considérera que les importations de modules nécessaires ont été préalablement faites.

I. Notion de fonction génératrice

Q 1. Soit X une variable aléatoire à valeurs dans $\mathbb{N}$.

Q 1.1. Montrer que G_X, la fonction génératrice de X, est au moins définie sur le segment $[0,1]$.

Q 1.2. Que valent $G_X(0)$ et $G_X(1)$?

Q 1.3. Montrer que la fonction $t \in [0,1] \mapsto G_X(t)$ est croissante.

Q 1.4. Montrer que pour tout $t \in [0,1]$, $G_X(t) \in [0,1]$.

Q 1.5. Montrer que pour tout $t \in [0,1]$, t^X admet une espérance et que $E\left(t^X\right) = G_X(t)$.

Q 1.6. Soit $t \in [0,1]$. Montrer que t^X admet une variance, que $V\left(t^X\right) = G_X\left(t^2\right) - G_X(t)^2$ et que $V\left(t^X\right) \leqslant 1$.

Q 2. Montrer que si X suit la loi binomiale $\mathscr{B}(n,p)$ de paramètre $(n,p) \in \mathbb{N}^* \times [0,1]$, alors

$$\forall t \in [0,1], \quad G_X(t) = (pt + 1 - p)^n$$

Q 3. Soit X une variable aléatoire à valeurs dans $\mathbb{N}$.

Q 3.1. Montrer que les assertions (i) et (ii) suivantes (voir page suivante) sont équivalentes.

(i) Il existe $m \in \mathbb{N}$ tel que $P(X \leqslant m) = 1$;

(ii) La fonction $t \mapsto G_X(t)$ est définie sur tout $\mathbb{R}$ et il s'agit d'un polynôme.

Q 3.2. On suppose dans cette question 3.2 que les assertions (i) et (ii) sont satisfaites. Montrer que X admet une espérance et que G_X est dérivable en 1. Exprimer $E(X)$ en fonction de $(G_X)'(1)$.

Q 4. Soit $m \in \mathbb{N}$ et X et Y deux variables aléatoires à valeurs dans $\{0, \dots, m\}$. Montrer que les deux assertions suivantes sont équivalentes.

(i) X et Y ont même loi ;

(ii) Pour tout $t \in [0, 1]$, $G_X(t) = G_Y(t)$.

II. Sommes aléatoires : un premier modèle

II.A. Présentation du modèle

Dans cette partie, on se donne $n \in \mathbb{N}^*$ et $(p_1, p_2) \in]0, 1[^2$. On considère n bulbes de jacinthe placés dans des conditions identiques. À l'issue d'une première phase dite de *croissance* d'une durée de six semaines, on mesure la hauteur des plantes et l'on ne retient, pour les besoins de l'expérience, que celles qui atteignent un seuil de vingt centimètres. À l'issue de la septième semaine, on observe la floraison des plantes ainsi retenues. Pendant cette seconde phase dite de *floraison*, on suppose que la croissance en hauteur est négligeable. Finalement, à l'issue des sept semaines, on retient uniquement les jacinthes qui satisfont simultanément les deux critères suivants : avoir franchi le seuil de hauteur et présenter une floraison. On formule les quatre hypothèses suivantes.

(A_1) Chaque jacinthe a une probabilité p_1 d'atteindre le seuil de vingt centimètres ;

(A_2) Chaque plante a une probabilité p_2 de fleurir ;

(A_3) Les comportements des n plantes sont indépendants ;

(A_4) La capacité à fleurir est indépendante de la phase de croissance.

En notant J_1 le nombre de jacinthes ayant atteint le seuil de vingt centimètres et J_2 le nombre de jacinthes sélectionnées à l'issue de la première étape ayant fleuri, l'objectif de cette partie est de déterminer la loi de J_2.

Q 5. Donner la loi de J_1. On justifiera la réponse en citant toute hypothèse utilisée parmi $(A_1), \dots, (A_4)$.

Q 6. Déterminer la loi de J_2 lorsque $n = 2$, $p_1 = \dfrac{2}{3}$ et $p_2 = \dfrac{3}{4}$.

Q 7.1. Compléter l'algorithme suivant afin de définir une commande Python `J2(n, p1, p2)` qui prend en argument un élément `n` de $\mathbb{N}^*$ et deux éléments `p1` et `p2` de $]0, 1[$ et qui simule et renvoie le nombre de jacinthes qui ont atteint vingt centimètres et qui ont fleuri lorsque `n` bulbes ont initialement été étudiés et que la probabilité de succès à l'étape 1 (respectivement 2) est `p1` (respectivement `p2`).

```
1  def J2(n, p1, p2):
2      aux = 0
3      l = [0 for i in range(n)]
4      for i in range(n):
5          if rd.binomial(1, p1) == # Ligne à compléter
6              l[i] = # Ligne à compléter
7      for i in range(n):
8          if l[i] == # Ligne à compléter
9              if rd.binomial(1, p2) == # Ligne à compléter
10                 aux = # Ligne à compléter
11     return aux
```

Q 7.2. Comment interpréter, à l'aune du modèle, les actions réalisées par la boucle des lignes 4 à 6 puis celle des lignes 7 à 10 ?

II.B. Loi de J_2

Q 8.1. Montrer que $(J_1 = i)_{i\in\{0,1,\dots,n\}}$ est un système complet d'événements.

Q 8.2. Soit $i \in \{0, 1, \dots, n\}$. Montrer que $P(J_1 = i) > 0$. Donner la loi conditionnelle de J_2 sachant $(J_1 = i)$ en précisant dans la justification les hypothèses utilisées parmi $(A_1), \dots, (A_4)$.

Q 8.3. Déduire des questions 5, 8.1 et 8.2 que pour tout $\ell \in \{0,1,\dots,n\}$,

$$P(J_2 = \ell) = \sum_{i=\ell}^{n} \binom{i}{\ell} (p_2)^{\ell} (1-p_2)^{i-\ell} \binom{n}{i} (p_1)^{i} (1-p_1)^{n-i}$$

Q 8.4. Montrer que pour tous $\ell \in \{0,1,\dots,n\}$ et $i \in \{\ell,\dots,n\}$, $\binom{i}{\ell}\binom{n}{i} = \binom{n}{\ell}\binom{n-\ell}{i-\ell}$.

Q 8.5. En déduire, pour tout $\ell \in \{0,1,\dots,n\}$, la valeur $P(J_2 = \ell)$ puis la loi de J_2.

Q 9. Expliquer ce que simule et renvoie la commande Python `mystere1(n, p1, p2)` suivante, qui prend en argument un élément `n` de $\mathbb{N}^*$ et deux éléments `p1` et `p2` de $]0, 1[$. On justifiera avec soin la réponse.

```
def mystere1(n, p1, p2):
    i = rd.binomial(n, p1)
    return rd.binomial(i, p2)
```

Q 10. On propose de déterminer la loi de J_2 d'une nouvelle manière. L'idée est de réindexer les n jacinthes pour voir apparaître en premier celles qui franchissent l'étape 1.

Q 10.1. Montrer qu'on peut écrire J_2 sous la forme

$$J_2 = \sum_{k=1}^{J_1} X_k \qquad \text{c'est-à-dire} \qquad \forall \omega \in \Omega, \quad J_2(\omega) = \sum_{k=1}^{J_1(\omega)} X_k(\omega)$$

(avec la convention que si $J_1(\omega) = 0$, alors la somme « vide » $\sum_{k=1}^{J_1(\omega)} X_k(\omega)$ allant de $k = 1$ à $k = 0$ est nulle, c'est-à-dire $J_2(\omega) = 0$), où :

- $X_1, \dots, X_n$ sont des variables aléatoires suivant toutes la loi de Bernoulli $\mathscr{B}(p_2)$ de paramètre p_2 ;
- les $(n + 1)$ variables aléatoires $X_1, \dots, X_n$ et J_1 sont indépendantes.

Q 10.2. En admettant que $G_{J_2} = G_{J_1} \circ G_{X_1}$ et en utilisant les questions 2 et 5, calculer G_{J_2}.

Q 10.3. Déduire des questions 2, 4 et 10.2 la loi de J_2.

Q 11. On propose de déterminer la loi de J_2 d'une troisième manière. On imagine que les n jacinthes sont étudiées pendant les sept semaines et que sont finalement retenues uniquement celles qui ont atteint le seuil de hauteur et qui ont fleuri.

Q 11.1. Calculer la probabilité qu'une plante donnée soit finalement retenue.

Q 11.2. En déduire la loi de J_2.

Q 12. Expliquer ce que simule et renvoie la commande Python `mystere2(n, p1, p2)` suivante, qui prend en argument un élément `n` de $\mathbb{N}^*$ et deux éléments `p1` et `p2` de $]0, 1[$. On justifiera avec soin la réponse.

```
def mystere2(n, p1, p2):
    aux = 0
    for i in range(n):
        hauteur_i = rd.binomial(1, p1)
        floraison_i = rd.binomial(1, p2)
        if (hauteur_i == 1) and (floraison_i == 1):
            aux = aux + 1
    return aux
```

III. Sommes aléatoires : un second modèle

III.A. Présentation du modèle

Dans cette partie, on s'intéresse à une population d'euglènes, organismes unicellulaires strictement asexués dotés d'un flagelle leur permettant de se déplacer, capables de réaliser la photosynthèse et d'absorber des matières organiques. On suppose qu'au jour $j = 0$, une seule euglène est présente. Chaque jour j, à la même heure, on compte les euglènes présentes. On formule les hypothèses suivantes.

(H_1) Chaque euglène engendre un nombre aléatoire de descendants ;

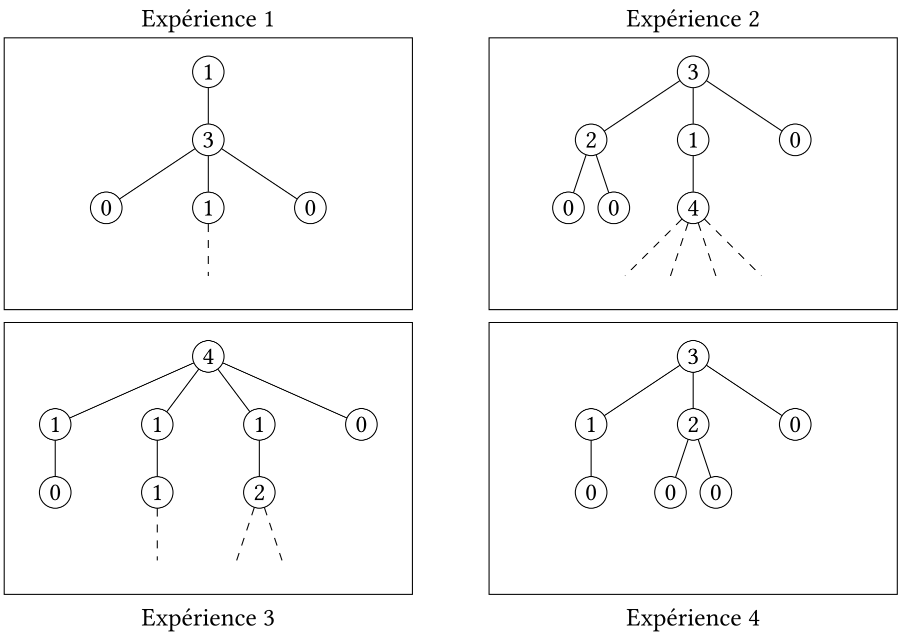
(H_2) Aucune euglène ne vit plus d'une journée.

Pour tout $j \in \mathbb{N}$, on note N_j le nombre d'euglènes présentes au jour j; pour tout $k \in \{1, \dots, N_j\}$, on note $D_{j,k}$ le nombre de descendants vivants au jour $j+1$ issus de l'euglène numéro k présente au jour j. Ainsi,

$$N_0 = 1 \qquad \text{et} \qquad \forall j \in \mathbb{N}, \quad N_{j+1} = \sum_{k=1}^{N_j} D_{j,k} \quad \text{c'est-à-dire} \quad \forall \omega \in \Omega, \quad N_{j+1}(\omega) = \sum_{k=1}^{N_j(\omega)} D_{j,k}(\omega)$$

(on adopte la convention que si $N_j(\omega) = 0$, alors $\displaystyle\sum_{k=1}^{N_j(\omega)} D_{j,k}(\omega) = 0$, c'est-à-dire $N_{j+1}(\omega) = 0$).

On a représenté sous forme d'arbres les premiers résultats issus de quatre expériences différentes, notées respectivement ω_1, ω_2, ω_3 et ω_4. Chaque nœud des arbres correspond à une euglène et indique le nombre (entouré pour faciliter la lecture) de descendants qu'elle engendre en une journée. Par exemple, pour l'expérience 1, les résultats sont les suivants : $D_{0,1}(\omega_1) = 1$, $D_{1,1}(\omega_1) = 3$, $D_{2,1}(\omega_1) = 0$, $D_{2,2}(\omega_1) = 1$, $D_{2,3}(\omega_1) = 0$, etc. En particulier $N_1(\omega_1) = 1$, $N_2(\omega_1) = 3$ et $N_3(\omega_1) = 1$. Par ailleurs, pour l'expérience 4, aucune euglène n'a survécu au troisième jour : $N_3(\omega_4) = 0$.



On fait les quatre autres hypothèses suivantes :

(H_3) Il existe une variable aléatoire D telle que pour tout $(j,k) \in \mathbb{N} \times \mathbb{N}^*$, $D_{j,k}$ et D suivent la même loi ;

(H_4) Il existe un entier $m \geqslant 2$ tel que D est à valeurs dans $\{0,1,\dots,m\}$ et $P(D = m) > 0$;

(H_5) Pour tout $(j,k) \in \mathbb{N} \times \mathbb{N}^*$, $D_{j,k}$ est à valeurs dans $\{0,1,\dots,m\}$;

(H_6) $\left(D_{j,k}\right)_{(j,k)\in\mathbb{N}\times\mathbb{N}^*}$ est une famille de variables aléatoires indépendantes.

Q 13. Que traduit chacune des hypothèses (H_3), (H_5) et (H_6) quant aux comportements des euglènes ?

Q 14. Montrer que N_1 et D suivent la même loi.

Q 15. Montrer que pour tout $j \in \mathbb{N}$, N_j est à valeurs dans $\{0, 1, \dots, m^j\}$.

On note, pour tout $n \in \mathbb{N}$, $p_n = P(D = n)$ et $S = \{n \in \mathbb{N}\,;\ p_n > 0\}$.

Q 16. Montrer que $S \subset \{0,1, \dots, m\}$ et que $m \in S$.

Q 17. On étudie dans cette question 17 quelques exemples. On suppose que $\{0, 1, 2, 3, 4\} \subset S$ et on considère les événements A_1 et A_2 suivants :

$$A_1 = (D_{0,1} = 4, D_{1,1} = 0, D_{1,2} = 2, D_{1,3} = 1, D_{1,4} = 0) \quad \text{et} \quad A_2 = (D_{0,1} = 3, D_{1,1} = 0, D_{1,2} = 0, D_{1,3} = 0).$$

Q 17.1. Donner les représentations arborescentes des événements A_1 et A_2.

Q 17.2. Exprimer $P(A_1)$ et $P(A_2)$ en fonction de la suite $(p_n)_{n\in\mathbb{N}}$.

Q 17.3. Montrer que $P(A_1) > 0$ et que $P(A_2) > 0$.

Q 17.4. Montrer qu'il existe un unique entier $n \in \mathbb{N}$, dont on donnera la valeur, tel que $A_1 \subset (N_2 = n)$.

Q 17.5. Montrer que $P(A_1) < P(N_2 = n)$.

III.B. Étude de la population moyenne

Q 18.1. Montrer que pour tout $j \in \mathbb{N}$, les $(m^j + 1)$ variables aléatoires $D_{j,1}, D_{j,2}, \dots, D_{j,m^j}$ et N_j sont indépendantes.

Q 18.2. Montrer que pour tous $j \in \mathbb{N}$ et $n \in \{0, 1, \dots, m^{j+1}\}$, $P(N_{j+1} = n) = \displaystyle\sum_{\ell=0}^{m^j} P(N_j = \ell) P\left(\sum_{k=1}^{\ell} D_{j,k} = n\right)$.

Q 18.3. Soit $j \in \mathbb{N}$. Montrer que les variables aléatoires D, N_j et N_{j+1} admettent des espérances. Déduire de la question 18.2 que $E(N_{j+1}) = E(D)E(N_j)$.

Q 18.4. Montrer que pour tout $j \in \mathbb{N}$, $E(N_j) = E(D)^j$.

Q 19. Lorsque $E(D) > 1$, que peut-on déduire de la question 18.4 sur la pertinence des hypothèses retenues — en particulier sur l'hypothèse (H_6) — pour modéliser l'évolution de la population d'euglènes ?

III.C. Calcul de la probabilité d'extinction : étude d'exemples

On s'intéresse désormais à l'*extinction* de la population d'euglènes. Plus précisément, en notant E_0 l'événement « toute la population d'euglènes finit par disparaître », on cherche à calculer la probabilité $P(E_0)$ de cet événement, notée α et appelée *la probabilité d'extinction* :

$$E_0 = \bigcup_{j=0}^{+\infty} (N_j = 0) \quad \text{et} \quad \alpha = P(E_0).$$

On pourra utiliser dans toute cette sous-partie III.C le résultat suivant, que l'on établira dans la sous-partie suivante III.D :

α est la plus petite solution de l'équation $G_D(t) - t = 0$ d'inconnue $t \in [0, 1]$.

Q 20. On reprend les exemples de la question 17. Exprimer $P(E_0|A_1)$ et $P(E_0|A_2)$ en fonction de α.

Q 21. Soit $p \in\,]0, 1]$. On suppose dans cette question 21 que $p_0 = 1 - p$ et que $p_2 = p$.

Q 21.1. Vérifier que l'hypothèse (H_4) est satisfaite.

Q 21.2. Exprimer en fonction de p la probabilité d'extinction (que l'on note exceptionnellement α_p dans cette question 21 pour souligner sa dépendance par rapport au réel p).

Q 21.3. Étudier les variations de la fonction $p \in]0, 1] \mapsto \alpha_p$. Interpréter ce résultat.

Q 22. On suppose disposer d'une commande Python `simule_D()` qui simule la variable aléatoire D. On propose les deux commandes Python `extinction_iter(j)` et `extinction_rec(j)` suivantes qui prennent en argument un élément `j` de $\mathbb{N}$ et qui renvoient `True` si aucune euglène ne vit au jour `j`, `False` sinon.

```
1  def extinction_iter(j):
2      N = 1
3      for i in range(j):
4          aux = 0
5          for k in range(N):
6              aux = aux + simule_D()
7          N = aux
8          if N == 0:
9              return True
10     return False
11
12 def extinction_rec(j):
13     if j == 0:
14         return False
15     else:
16         N1 = simule_D()
17         for k in range(N1):
18             if extinction_rec(j-1) == False:
19                 return False
20         return True
```

Q 22.1. Expliquer les lignes 4 à 6.

Q 22.2. Que modélise la variable `N1` présente à la ligne 16 ?

Q 22.3. La commande `extinction_iter(j)` réalise-t-elle un parcours en largeur ou un parcours en profondeur de l'arbre associé à l'expérience ? Qu'en est-il de la commande `extinction_rec(j)` ? On justifiera avec soin les réponses et on comparera l'efficacité des deux commandes.

Q 23. On suppose dans cette question 23 que $p_0 = p_2 = \dfrac{1}{2}$ et on considère l'ensemble (aléatoire) $\mathscr{E}$ défini, pour tout $\omega \in \Omega$, par : $\mathscr{E}(\omega) = \{j \in \mathbb{N}^* \ ; \ N_j(\omega) = 0\}$. Pour tout $\omega \in \Omega$, on note $J(\omega)$ l'entier naturel suivant :

$$J(\omega) = \begin{cases} \min(\mathscr{E}(\omega)) & \text{si l'ensemble } \mathscr{E}(\omega) \text{ est non vide} \\ 0 & \text{si } \mathscr{E}(\omega) = \varnothing \end{cases}$$

Q 23.1. Vérifier que l'hypothèse (H_4) est satisfaite.

Q 23.2. Montrer que $\alpha = 1$ et en déduire la valeur de $P(J = 0)$.

Q 23.3. On admet que $P(J = j) \underset{j \to +\infty}{\sim} \dfrac{2}{j^2}$. La variable aléatoire J admet-elle une espérance ?

Q 23.4. Compléter l'algorithme suivant afin de définir une commande Python `temps_extinction()` récursive qui simule et renvoie la valeur du premier jour où toute la population d'euglènes a disparu.

```
def temps_extinction():
    if # Ligne à compléter
        return 1
    else:
        t1 = temps_extinction()
        t2 = temps_extinction()
        return # Ligne à compléter
```

III.D. Calcul de la probabilité d'extinction : étude générale

Pour tout $j \in \mathbb{N}$, on note $a_j = P(N_j = 0)$ la probabilité de l'événement « aucune euglène ne vit au jour j ».

Q 24.1. Montrer que $E_0 = \displaystyle\bigcup_{j=0}^{+\infty} \big(N_{j+1} = 0, N_j \neq 0\big)$.

Q 24.2. Montrer que pour tout $j \in \mathbb{N}$, $\big(N_{j+1} = 0\big) = \big(N_{j+1} = 0, N_j \neq 0\big) \cup \big(N_j = 0\big)$.

Q 24.3. Déduire des questions 24.1 et 24.2 que $\alpha = \displaystyle\sum_{j=0}^{+\infty} \big(a_{j+1} - a_j\big)$.

Q 24.4. En déduire que $\displaystyle\lim_{j \to +\infty} a_j = \alpha$.

Q 25. L'objectif de cette question est d'exprimer α en fonction de G_D.

Q 25.1. Montrer que pour tout $j \in \mathbb{N}$, $a_{j+1} = \displaystyle\sum_{n \in S} p_n P(N_{j+1} = 0 | N_1 = n)$.

Q 25.2. Montrer que pour tous $j \in \mathbb{N}$ et $n \in S$, $P(N_{j+1} = 0 | N_1 = n) = (a_j)^n$.

Q 25.3. Déduire des questions 25.1 et 25.2 que pour tout $j \in \mathbb{N}$, $a_{j+1} = G_D(a_j)$.

Q 25.4. Montrer que $\alpha = G_D(\alpha)$.

On note $\mathscr{F} = \{t \in [0,1]\,;\, G_D(t) - t = 0\}$.

Q 25.5. Montrer que pour tous $t \in \mathscr{F}$ et $j \in \mathbb{N}$, $a_j \leqslant t$.

Q 25.6. En déduire que α est le plus petit élément de $\mathscr{F}$.

Q 26. Montrer que $\alpha = 0$ si et seulement si $p_0 = 0$.

Q 27. On note f la fonction

$$f : \begin{array}{rcl} [0,1] & \longrightarrow & \mathbb{R} \\ t & \longmapsto & G_D(t) - t \end{array}$$

Q 27.1. Montrer que f est deux fois dérivable.

Q 27.2. Montrer que f' est strictement croissante.

Q 27.3. Montrer que $f'(0) < 0$.

Q 28. On suppose dans cette question 28 que $E(D) \leqslant 1$.

Q 28.1. Montrer que $f'(1) \leqslant 0$.

Q 28.2. Étudier les variations de f.

Q 28.3. En déduire que $\mathscr{F} = \{1\}$ et que $\alpha = 1$. Interpréter ce résultat.

Q 29. On suppose dans cette question 29 que $E(D) > 1$.

Q 29.1. Montrer que $f'(1) > 0$.

Q 29.2. Montrer qu'il existe un unique réel $\beta \in \,]0,1[$ tel que $f'(\beta) = 0$.

Q 29.3. Étudier les variations de f.

Q 29.4. En déduire que $\mathscr{F} = \{\alpha, 1\}$ et que $\alpha < 1$. Interpréter ce résultat.

III.E. Une application : calcul numérique de la probabilité d'extinction

Q 30. On suppose disposer d'une commande Python `G_D(t)` qui prend en argument un élément `t` de $[0,1]$ et qui renvoie la valeur de $G_D(\texttt{t})$. Compléter l'algorithme suivant afin de définir une commande Python `proba_extinction(epsilon)` qui prend en argument un élément `epsilon` de $]0,1]$ et qui renvoie une valeur approchée de α par excès à `epsilon` près via une recherche dichotomique.

```python
def proba_extinction(epsilon):
    a = 0
    c = 1
    while # Ligne à compléter
        b = (a+c)/2
        if G_D(b) - b > 0:
            # Ligne à compléter
        else:
            # Ligne à compléter
    return # Ligne à compléter
```

IV. Tracés de graphes

Q 31. Soit n et p deux entiers naturels non nuls, ε et η deux réels strictement positifs, X une variable aléatoire à valeurs dans $\mathbb{N}$ et $X_0, \dots, X_{n-1}$ n variables aléatoires indépendantes suivant toutes la loi de X.

Q 31.1. Soit $t \in [0, 1]$. À l'aide des questions 1.5 et 1.6, montrer que

$$P\left(\left|\frac{1}{n}\sum_{j=0}^{n-1} t^{X_j} - G_X(t)\right| \geqslant \varepsilon\right) \leqslant \frac{1}{n\varepsilon^2}.$$

Q 31.2. On note, pour tout $i \in \{0,1,\dots,p\}$, A_i l'événement suivant :

$$A_i = \left(\frac{1}{n}\sum_{j=0}^{n-1}\left(\frac{i}{p}\right)^{X_j} - \varepsilon < G_X\left(\frac{i}{p}\right) < \frac{1}{n}\sum_{j=0}^{n-1}\left(\frac{i}{p}\right)^{X_j} + \varepsilon\right).$$

Montrer que $P\left(\bigcap_{i=0}^{p} A_i\right) \geqslant 1 - \dfrac{p+1}{n\varepsilon^2}$.

Q 31.3. En déduire une condition suffisante sur l'entier n pour que $P\left(\bigcap_{i=0}^{p} A_i\right) \geqslant 1 - \eta$.

Q 32. Soit X une variable aléatoire à valeurs dans $\mathbb{N}$. On suppose disposer d'une commande Python `simule_X()` qui simule X. On définit la commande Python `trace1(p, epsilon, eta)` suivante qui prend en argument un élément `p` de $\mathbb{N}^*$ et deux éléments `epsilon` et `eta` de $\mathbb{R}_+^*$ et qui trace deux courbes, chacune d'entre elles contenant $\mathtt{p}+1$ points d'abscisses

$$0, \frac{1}{\mathtt{p}}, \frac{2}{\mathtt{p}}, \dots, 1$$

```python
def trace1(p, epsilon, eta):
    x = [i/p for i in range(p+1)]
    y_empir = [0 for i in range(p+1)]
    n = int((p+1)/(eta*epsilon**2))+1
    echantillon = [simule_X() for j in range(n)]
    for i in range(p+1):
        for j in range(n):
            y_empir[i] = y_empir[i] +  x[i]**echantillon[j] / n
    ymax = [y_empir[i] + epsilon  for i in range(p+1)]
    ymin = [y_empir[i] - epsilon  for i in range(p+1)]
    plt.plot(x, ymax)
    plt.plot(x, ymin)
    plt.show()
```

Q 32.1. Expliquer la ligne 4.

Q 32.2. Expliquer les lignes 7 et 8.

Q 32.3. Expliquer les lignes 9 et 10.

Q 32.4. Pour tout $i \in \{0, \dots, \texttt{p}\}$, on note M_i le point de coordonnées (`x[i]`, `ymax[i]`) et N_i le point de coordonnées (`x[i]`, `ymin[i]`). Montrer que les `p+1` segments $[M_i, N_i]$ (lorsque i décrit $\{0, \dots, \texttt{p}\}$) sont de longueur (2 `epsilon`) et que, avec une probabilité au moins égale à (1 − `eta`), la courbe représentative de la fonction G_X intersecte ces (`p+1`) segments.

Q 33. Soit $m \in \mathbb{N}$ et X une variable aléatoire à valeurs dans $\mathbb{N}$ tels que $P(X \leqslant m) = 1$. Compléter l'algorithme suivant afin de définir une commande Python `trace2(p, liste)` qui prend en argument un élément `p` de $\mathbb{N}^*$ et une liste de réels positifs de somme 1, disons `[p0, p1, ..., pm]`, et qui trace $\texttt{p} + 1$ points de coordonnées

$$(0, G_X(0)), \left(\frac{1}{\texttt{p}}, G_X\left(\frac{1}{\texttt{p}}\right)\right), \left(\frac{2}{\texttt{p}}, G_X\left(\frac{2}{\texttt{p}}\right)\right), \dots, (1, G_X(1))$$

où, pour tout $n \in \{0, \dots, m\}$, $P(X = n) = \texttt{pn}$. Par exemple, si X suit la loi de Bernoulli $\mathscr{B}(3/5)$, la commande `trace2(10, [0.4, 0.6])` trace onze points du graphe de la fonction G_X.

```python
def trace2(p, liste):
    x = # Ligne à compléter
    y = # Ligne à compléter
    m = len(liste)-1
    for i in # Ligne à compléter
        for n in # Ligne à compléter
            y[i] = y[i] + # Ligne à compléter
    # Ligne à compléter
    # Ligne à compléter
```

Annexe Python

Dans le module `numpy.random` importé sous l'alias `rd` :
`rd.binomial(n, p)` simule une variable aléatoire suivant une loi binomiale $\mathscr{B}$(n,p) de paramètre (n,p).

Dans le module `matplotlib.pyplot` importé sous l'alias `plt` :
`plt.plot(x, y)` prend en entrée deux vecteurs ou deux listes de même taille et réalise le tracé des points d'abscisses prises dans `x` et d'ordonnées prises dans `y` tandis que `plt.show()` affiche le tracé.

FIN DU SUJET