Questions de cours :

1. DL 5 de sin
2. définition du produit scalaire dans $\mathbb{R}^n$.
3. Loi faible des grands nombres
4. Définir ce qu'est une racine d'un polynôme puis une racine multiple.
5. A quelle(s) condition(s) sur sa fonction de répartition une variable aléatoire X admet-elle une densité de probabilité ? Comment détermine-t-on alors une densité de X ?
6. représentation graphique de la fonction sinus.
7. Représenter une densité d'une loi normale centrée réduite.
8. DL 4 en 0 de $\ln(1+x)$.
9. Définition d'une application surjective.
10. Définition d'une valeur propre et d'un sous espace propre pour une matrice.
11. Covariance de deux variables aléatoires discrètes.
12. Définition de deux suites adjacentes. Théorème les concernant.

Exercice 1

Soient $(X_k)_{k\in\mathbb{N}^*}$ des variables aléatoires indépendantes définies sur un même espace probabilisé $(\Omega, \mathcal{A}, P)$ et de même loi telle que $\forall k \in \mathbb{N}^* \quad P(X_k = -1) = P(X_k = 1) = \frac{1}{2}$.

On pose $\forall n \in \mathbb{N}^* \quad S_n = \displaystyle\sum_{k=1}^{n} X_k$.

1. (a) Calculer l'espérance et la variance de $\frac{S_n}{n}$.

   (b) Montrer que $\forall \varepsilon > 0 \quad P\left(\left|\frac{S_n}{n}\right| > \varepsilon\right) \leq \frac{1}{n\varepsilon^2}$.

   (c) Ecrire un programme Python de paramètres n et ε qui simule S_n et renvoie 1 si $\left|\frac{S_n}{n}\right| \geq \varepsilon$ et 0 sinon.

   (d) Ecrire un programme Python renvoyant une valeur approchée de $P\left(\left|\frac{S_n}{n}\right| \geq \varepsilon\right)$. le tester pour $n = 100$, $n = 1000$, $n = 10000$.
   Apporter un regard critique par rapport à la question 1)b).

2. (a) Montrer que $\forall n \in \mathbb{N}^* \quad \forall t \geq 0 \quad E\left(\mathrm{e}^{tS_n}\right) = \left(\frac{e^t+e^{-t}}{2}\right)^n$.

   (b) Montrer que $\forall t \geq 0 \quad \frac{e^t+e^{-t}}{2} = \displaystyle\sum_{k=0}^{+\infty} \frac{t^{2k}}{(2k)!}$, puis montrer que $\frac{e^t+e^{-t}}{2} \leq e^{\frac{t^2}{2}}$.

   (c) Montrer que $P\left(\left|\frac{S_n}{n}\right| > \varepsilon\right) \leq 2\exp\left(\frac{nt^2}{2} - nt\varepsilon\right) \leq 2\exp\left(\frac{-n\varepsilon^2}{2}\right)$.
   (On étudiera les variations sur $\mathbb{R}^+$ de $t \mapsto \frac{nt^2}{2} - nt\varepsilon$).
   Apporter un regard critique par rapport à la question 1)b).

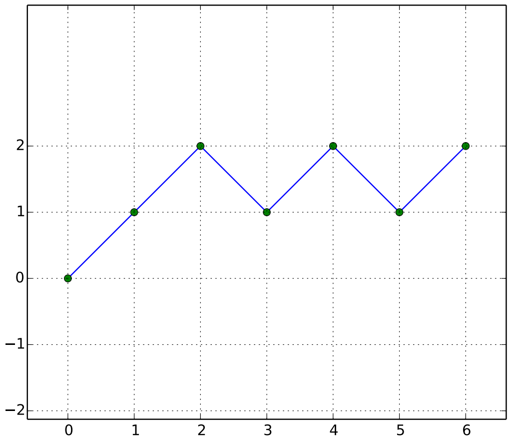
Exercice 2
On admet que si $\forall n \in \mathbb{N} \quad u_n = \sum_{k=0}^{n} \binom{n}{k} v_k$, alors $v_n = \sum_{k=0}^{n} \binom{n}{k} (-1)^{n-k} u_k$.
On rappelle qu'une permutation de $[\![1, n]\!]$ est une bijection de $[\![1, n]\!]$ dans lui-même .
On définit un dérangement de $[\![1, n]\!]$ comme étant une permutation de $[\![1, n]\!]$ sans point fixe . (rappel : on dit que k est un point fixe de la permutation p ssi $p(k) = k$.
On rappelle qu'on représente usuellement une permutation p de $[\![1, n]\!]$ par la n-liste des images $[p(1), p(2), \dots, p(n)]$.
Dans une telle liste, tout entier de $[\![1, n]\!]$ apparait une fois et une seule.

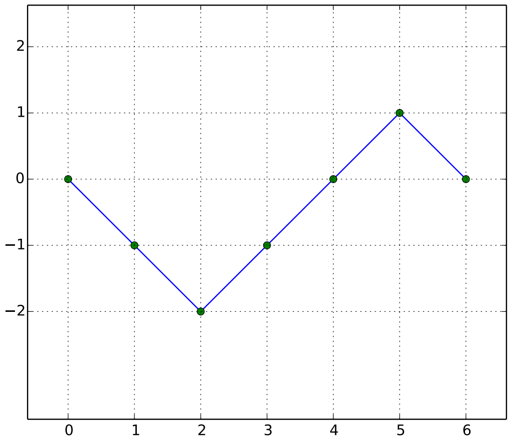
1. En utilisant la fonction randint du module random, écrire une fonction Python permettant de simuler une permutation de $[\![1, n]\!]$.
2. Ecrire une fonction Python prenant en argument une permutation P de $[\![1, n]\!]$, et renvoyant True si c'est un dérangement, false sinon.
3. Déterminer la probabilité qu'une permutation de $[\![1, 50]\!]$ soit un dérangement.
4. On note D_n le nombre de dérangements de $[\![1, n]\!]$.
   (a) Justifier que $n! = \sum_{k=0}^{n} \binom{n}{k} D_{n-k}$.
   (b) En déduire la probabilité d_n qu'une permutation de $[\![1, n]\!]$ prise au hasard soit un dérangement.
   (c) Donner $\lim_{n \to +\infty} d_n$. Ce résultat est-il en accord avec celui de la question 3) ?
5. On appelle F_n le nombre de points fixes d'une permutation de $[\![1, n]\!]$.
   (a) Exprimer F_n à l'aide de variables aléatoires de Bernoulli.
   En déduire l'espérance de F_n.
   (b) Donner la loi de F_n.

Exercice 3
On effectue $2n$ tirages d'une pièce de monnaie équilibrée . On appelle X_{2n} la variable aléatoire égale au nombre de "pile" obtenus. On note A_{2n} l'événement « au cours des $2n$ lancers, le nombre de "pile" obtenus a toujours été strictement supérieur au nombre de "face" obtenus ».

1. Si $n = 2$, quelle est la probabilité de A_{2n} ?
2. (a) Ecrire une fonction Python simulant $2n$ lancers de la pièce et renvoyant 1 si A_{2n} est réalisé, 0 sinon.
   (b) A l'aide de la fonction précédente, estimer la probabilité de A_{2n} pour $n = 2$ et $n = 5$.

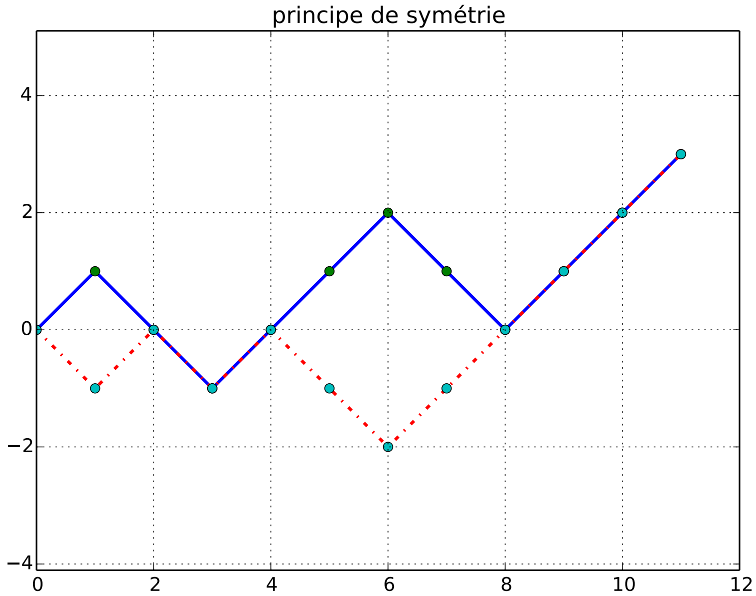
   Dans la suite, on représente chaque suite des $2n$ lancers de la pièce par un chemin partant du point $O(0, 0)$, tel que, pour chaque lancer de la pièce, on progresse d'une unité vers la droite et d'une unité vers le haut si l'on obtient pile, et on progresse d'une unité vers la droite et d'une unité vers le bas si l'on obtient face. Voici par exemple, pour $n = 3$, les chemins





3. Combien y a-t-il de chemins possibles ?
4. Pour $i \in [\![0, 2n]\!]$, combien de chemins réalisent $(X_{2n} = i)$?
5. Quelle est la loi de X_{2n} ?
6. On admet dans un premier temps que, pour $i > n$, $P_{(X_{2n}=i)}(A_{2n}) = \frac{i-n}{n}$.
   (a) Justifier que $P(A_{2n}) = \dfrac{1}{2^{2n}}\left(\dfrac{1}{n}\underbrace{\sum_{i=n+1}^{2n} i\binom{2n}{i}}_{S_1} - \underbrace{\sum_{i=n+1}^{2n}\binom{2n}{i}}_{S_2}\right)$.
   (b) En calculant séparément les deux sommes intervenant dans l'expression ci-dessus, donner $P(A_{2n})$.
7. Le but de cette question est de démontrer la propriété admise précédemment, à savoir que pour $i > n$, $P_{(X_{2n}=i)}(A_{2n}) = \frac{i-n}{n}$.
   On suppose donc dans toute cette question $(X_{2n} = i)$ réalisé. On s'intéresse en conséquence uniquement aux chemins partant de O et pour lesquels, parmi les $2n$ pas effectués vers la droite, il y en a eu i qui montaient . On note $Z(2n, 2i - 2n)$ le point d'arrivée commun à ces chemins.
   (a) Combien y a-t-il de ces chemins au total ?
   (b) Combien y a-t-il de ces chemins passant par $A(1, 1)$?
   (c) Montrer que parmi ces chemins il y a en a exactement $\binom{2n-1}{i}$ passant par $B(1, -1)$.
   (d) On remarque que l'événement A_{2n} est réalisé ssi le chemin obtenu ne coupe pas l'axe Ox. On cherche donc à dénombrer les chemins partant de O et arrivant en Z et ne coupant pas l'axe Ox. Pour celà, on dénombre tout d'abord ceux qui coupent l'axe Ox.

i. En utilisant une symétrie axiale d'axe Ox, comme dans le dessin suivant, justifier qu'il y a autant de chemins passant par A et coupant l'axe Ox, que de chemins passant par B .



ii. En déduire le nombre de chemins passant par A et ne coupant pas l'axe Ox.

iii. Retrouver ainsi que pour $i > n$, $P_{(X_{2n}=i)}(A_{2n}) = \frac{i-n}{n}$.

Exercice 4
On dispose de N boules numérotées de 1 à N, indiscernables au toucher, et de 2 urnes A et B.

On répartit initialement les boules entre les deux urnes, puis on effectue une série illimitée d'étapes selon le protocole suivant : à chaque étape, on tire au hasard un nombre entre 1 et N et on change d'urne la boule portant le numéro correspondant.
On note X_0 la variable aléatoire égale au nombre de boules présentes initialement dans l'urne A et pour tout entier naturel n non nul, on note X_n la variable aléatoire égale au nombre de boules présentes dans l'urne A après n étapes.

1. Soit $k \in [\![1, N-1]\!]$. Déterminer $P(X_{n+1} = k)$ en fonction de $P(X_n = k-1)$ et $P(X_n = k+1)$.
2. (a) Écrire une fonction `etape` prenant en arguments `k` (nombre de boules dans A à un instant donné) et `N` et renvoyant la valeur nombre de boules dans A à l'instant suivant .
   (b) Écrire une fonction renvoyant sous forme d'une liste les valeurs successivement prises par $X_0, ..., X_n$.

   A partir de maintenant et dans toute la suite, on suppose $N = 3$.

3. Pour tout $n \in \mathbb{N}$ on note $U_n = \begin{pmatrix} P(X_n = 0) \\ P(X_n = 1) \\ P(X_n = 2) \\ P(X_n = 3) \end{pmatrix}$; on note aussi $M = \begin{pmatrix} 0 & \frac{1}{3} & 0 & 0 \\ 1 & 0 & \frac{2}{3} & 0 \\ 0 & \frac{2}{3} & 0 & 1 \\ 0 & 0 & \frac{1}{3} & 0 \end{pmatrix}$,
   (a) Montrer que pour tout entier naturel n, $U_{n+1} = MU_n$.
   (b) Montrer que 1 est valeur propre de M et donner le sous espace propre associé .
   (c) On suppose que X_0 suit une loi binomiale de paramètres 3 et 1/2. Quelle loi suit alors X_n ?
   (d) Quel est l'ensemble des lois que pourrait suivre X_0 pour que X_n ait la même loi que X_0 ?
4. On suppose que l'urne A est initialement vide. On appelle D la variable aléatoire égale au nombre de tirages nécessaires pour que l'urne A soit à nouveau vide.
   (a) Ecrire une fonction python simulant D.
   (b) i. Calculer $P(D = 2)$ et $P(D = 4)$.
   ii. Pourquoi D ne peut-il prendre que des valeurs paires ?
   iii. Montrer que $P(X_{2k+2} = 0) = \frac{1}{9}P(X_{2k} = 0) + \frac{2}{9}$.
   (Question non posée et non indispensable pour la suite, mais que je pose pour les révisions : calculer $P(X_{2k} = 0)$ en fonction de k.)
   iv. On note désormais $u_k = P(X_{2k} = 0)$ et $d_k = P(D = 2k)$.
   Montrer que $(X_{2k} = 0) = \bigcup_{j=1}^{k} ((X_{2k} = 0) \cap (D = 2j))$.
   v. En déduire que $d_k = u_k - \sum_{j=1}^{k-1} d_j u_{k-j}$.

   (c) A l'aide des relations $u_{k+1} = \frac{1}{9}u_k + \frac{2}{9}$ et $d_k = u_k - \sum_{j=1}^{k-1} d_j u_{k-j}$, écrire une fonction Python renvoyant la liste $[d_1, \ldots, d_n]$.
   S'il reste du temps, pour $n = 5$, comparer les résultats obtenus par simulation de D avec les résultats de la fonction précédente.

Exercice 5

Deux personnes, Brigitte et Antoine, décident de vider une urne de n beignets contenant 1 beignet au chocolat et $n-1$ beignets aux pommes.

Quand Antoine pioche un beignet, il le mange, quand Brigitte pioche un beignet, elle le remet dans l'urne.

Tous deux tirent à tour de rôle et c'est Antoine qui commence.

1. (a) Combien de tirages sont nécessaires pour vider l'urne ?

   (b) Soit i appartenant à $\{1, 2, \dots, n-1\}$. Combien reste-t-il de beignets dans l'urne après le $2i$ ème tirage ? après le $2i+1$ ème ?

2. Ecrire une fonction Python prenant n en argument, simulant l'expérience et renvoyant le numéro du tirage où le beignet au chocolat a été mangé. (On pourra utiliser au choix la fonction random ou la fonction randint

3. On note X_i la variable aléatoire égale à 1 si le beignet au chocolat est tiré au i ème tirage (que ce soit ou non pour la première fois) . On note X le nombre total d'apparitions du beignet au chocolat. Justifier que X_i suit une loi de Bernoulli de paramètre $1/n$.

4. Trouver l'espérance de X après avoir exprimé X en fonction des X_i.

5. + 1 question finale que je n'ai même pas eu le temps de lire

Exercice 6

On dispose d'un dé équilibré à n faces avec lequel 2 joueurs (nommés A et B) prennent part à un jeu . A lance le dé et verse 3 euros à B

B lance le dé et tant qu'il n'obtient pas un nombre supérieur ou égal à celui obtenu par A il relance le dé. À chaque lancer il verse 1 euro à A.

On note : X : la variable aléatoire égale au nombre obtenu par A.

M : la variable aléatoire égale à la somme versée à A.

1. Quelle est la loi de X ?
2. Ecrire une fonction Python permettant de simuler M. (L'emploi de la fonction `randint` est conseillé.)
3. Calculer la probabilité que $(M = j)$ sachant que $(X = 1)$.
4. Soit $k \geq 2$. Calculer la probabilité que $(M = j)$ sachant que $(X = k)$, ainsi que $(M \geq j)$ sachant que $(X = k)$.
5. Calculer $(M \geq j)$.
6. On admet que $E(M) = \sum_{j=1}^{+\infty} P(M \geq j)$. Calculer $E(M)$.
   Donner sa limite lorsque n tend vers $+\infty$.
7. Déterminer l'espérance de G, gain algébrique de A, ainsi que sa limite lorsque n tend vers $+\infty$.

Exercice 7
Un fabricant de poudre chocolatée met dans chacune de ses boîtes une image à collectionner. Il y a en tout n images différentes , et une seule par boîte.
On note T la variable aléatoire correspondant au nombre de boîtes nécessaires pour avoir toute la collection d'images.
Pour tout i appartenant à $[\![2, n]\!]$, on note T_i la variable aléatoire égale au nombre de boîtes à acheter pour obtenir une $i$ème image différente des $i-1$ déjà obtenues.
On définit également une suite $(u_n)_{n\in\mathbb{N}^*}$ par $u_n = -\ln(n) + 1 + 1/2 + ... + 1/n$. On admet qu'elle est convergente.

1. Faire un programme prenant en argument epsilon (supposé réel strictement positif) et qui renvoie u_n tel que l'écart $u_{n+1} - u_n$entre deux termes consécutifs soit inférieur à epsilon.
2. (a) Pour $i \in [\![2, n]\!]$,donner la loi de T_i
   (b) Exprimer T en fonction de $T_2, T_3, ..., T_n$.
   (c) Montrer que $E(T) = n * (1 + 1/2 + ... + 1/n)$.
   (d) c étant un réel strictement positif, montrer que $P(T > n * c * ln(n)) \leq 1/c + u_n/(c * ln(n))$
3. (a) Pour i, k dans $\mathbb{N}^*$, on pose $A_{i,k}$:" on n'a pas obtenu l'image i lors des k premiers tirages". Calculer $P(A_{i,k})$.
   (b) c étant un réel strictement positif, montrer que $P\left(\bigcup_{i=1}^{n} A_{i,k}\right) \leq n * \exp(-k/n)$.
   (indication : $1 + t \leq exp(t)$).
   (c) Montrer que $P(T > n * c * ln(n)) \leq 1/(n^{c-1})$.
4. Comparer les deux inégalités des questions 2.4 et 3.3 (Question un peu vague......)

Exercice 8

Soient a et b deux réels tels que $a < b$.

1. Ecrire une fonction Python qui renvoie une liste de n valeurs comprises entre a et b pour une variable aléatoire centrée réduite.
   On utilisera l'instruction random.randn() du module numpy.
2. Donner la valeur moyenne obtenue pour $a = \frac{1}{2}$, $b = 10^4$ et $n = 10$.

   Soit X suivant une loi normale d'espérance μ et de variance θ^2, et $U = \frac{X-\mu}{\theta}$. On note Φ la fonction de répartition de la loi normale centrée réduite et pour tout réel x on pose $F_{\mu,\theta}(x) = P_{(a<X\leq b)}(X \leq x)$.
3. (a) Donner la loi de U. En déduire que $\forall x \in \mathbb{R} \quad P(X \leq x) = \Phi\left(\frac{x-\mu}{\theta}\right)$.
   (b) Montrer que $F_{\mu,\theta}$ est une fonction de répartition d'une variable aléatoire à densité.
4. Soit Y admettant $F_{0,1}$ comme fonction de répartition.
   (a) Donner une densité de Y.
   (b) Donner, en fonction de a, b et $\displaystyle\int_a^b \frac{1}{\sqrt{2\pi}} e^{-\frac{t^2}{2}} dt$, l'espérance de Y.
   (c) Calculer $E(Y)$. Que devient-elle si a tend vers $-\infty$ et b tend vers $+\infty$?
5. Il y avait une nouvelle question avec une nouvelle var dont il fallait trouver densité ou fonction de répartition

Exercice 9
La formule du produit de convolution est rappelée.

On fixe une origine des temps et on s'intéresse aux appels reçus par un vétérinaire de campagne à partir de cet instant.
On considère que le temps écoulé entre cet instant initial et l'appel du vétérinaire par le $k$iéme propriétaire de chevaux suit une variable aléatoire X_k dont une densité est f_k telle que $f_k(t) = \begin{cases} \lambda^k \frac{t^{k-1}}{(k-1)!} e^{-\lambda t} & \text{si } t \geq 0 \\ 0 & \text{sinon} \end{cases}$
et que le temps écoulé entre cet instant initial et l'appel du vétérinaire par le $k$iéme propriétaire de vaches suit une variable aléatoire Y_k dont une densité est g_k telle que $g_k(t) = \begin{cases} \mu^k \frac{t^{k-1}}{(k-1)!} e^{-\mu t} & \text{si } t \geq 0 \\ 0 & \text{sinon} \end{cases}$,
λ et μ étant des réels strictement positifs fixés.
On suppose également que les variables aléatoires X_i sont indépendantes des variables aléatoires Y_j et on admet qu'on ne reçoit jamais deux appels au même instant.

On appelle également U_n la variable aléatoire égale au nombre de propriétaires de chevaux parmi les n premiers appels reçus par ce vétérinaire.

1. (a) On suppose qu'un jour donné, les valeurs prises par les variables aléatoires X_1, X_2, X_3 ont été respectivement 2, 4, 4.1 et les valeurs prises par les variables aléatoires Y_1, Y_2, Y_3 ont été 2.5, 3.2, 4.2. Quelle est la valeur prise par U_4 ?

   (b) Ecrire une fonction Python U(n,X,Y) , où X représente la liste des valeurs de $X_1, \ldots, X_n$ et Y représente la liste des valeurs de $Y_1, \ldots, Y_n$ et qui renvoie U_n. (On rappelle que tous les instants des appels sont distincts.)

2. (a) Justifier que $(U_n \geq k) = (X_k \leq Y_{n-k+1})$. (On pourra s'aider d'un schéma.)

   (b) On note $p_k = P(X_k \leq Y_{n-k+1})$.
   Pour k compris entre 1 et $n-1$, donner $P(U_n = k)$ en fonction de p_k et p_{k+1}, puis $P(U_n = n)$.

3. On suppose que $\lambda + \mu = 1$.

   (a) En utilisant f_k comme une densité, montrer que $\displaystyle\int_0^{+\infty} t^{k-1} e^{-\lambda t} dt = \frac{(k-1)!}{\lambda^k}$.

   (b) Montrer qu'une densité de $-X_k$ est $t \mapsto f_k(-t)$.

   (c) Soit h_k une densité de $Y_{n+1-k} - X_k$; à l'aide de la formule du produit de convolution, montrer que, si $x \geq 0$,
   $$h_k(x) = \frac{\lambda^k \mu^{n+1-k}}{(k-1)!(n-k)!} e^{\lambda x} \int_x^{+\infty} (t-x)^{k-1} t^{n-k} e^{-t} dt$$
   $$= \frac{\lambda^k \mu^{n+1-k}}{(k-1)!(n-k)!} e^{-\mu x} \int_0^{+\infty} u^{k-1} (u+x)^{n-k} e^{-u} du.$$

   (d) Montrer que, si $x \geq 0$, $h_k(x) = \dfrac{\lambda^k \mu^{n+1-k}}{(k-1)!(n-k)!} \displaystyle\sum_{i=0}^{n-k} \binom{n-k}{i} (n-i-1)!\, x^i e^{-\mu x}$.

   (e) En déduire une expression de p_k.

   (f) Montrer que $P(U_n = k) = \dbinom{n}{k} \lambda^k \mu^{n-k}$. Quelle est la loi de U_n ?

Exercice 10

Pour tout entier $n \geq 1$, on pose $c_n = \displaystyle\int_0^1 \frac{x^{n-1}}{1+x} dx$.

1. (a) Montrer que pour tout entier $n \geq 2 \quad c_{n+1} \leq c_n \leq c_{n-1}$.
   (b) Montrer que pour tout entier $n \geq 1 \quad c_{n+1} + c_n = \frac{1}{n}$.
   (c) En déduire que $c_n \sim \frac{1}{2n}$ lorsque n tend vers $+\infty$.
2. Déterminer c_1.

   Montrer que $\forall n \in \mathbb{N} \quad c_n = (-1)^n \left(\displaystyle\sum_{k=1}^{n-1} \frac{(-1)^{k+1}}{k} - \ln 2 \right)$.
3. Ecrire une fonction Python de paramètre n renvoyant la valeur de c_n.
4. Pour tout entier $n \geq 1$, on définit la fonction f_n par $f_n(x) = \begin{cases} \frac{x^{n-1}}{c_n(1+x)} & si \quad x \in [0,1] \\ 0 & sinon \end{cases}$.
   (a) Montrer que f_n est une densité . On note désormais X_n une variable à densité f_n.
   (b) Montrer que X_n a une espérance et donner la limite de cette espérance lorsque n tend vers $+\infty$.
   (c) On note F_n la fonction de répartition de X_n et $F(x) = \begin{cases} 1 & si \quad x \geq 1 \\ 0 & sinon \end{cases}$.

   Montrer que pour tout réel x, la suite $(F_n(x))$ a pour limite $F(x)$ quand n tend vers $+\infty$.

Exercice 11

Soient $a_0, a_1, \cdots, a_{n-1}$ des nombres complexes (pas forcément distincts). on considère la matrice

$$A = \begin{pmatrix} a_0 & a_1 & a_2 & \dots & a_{n-2} & a_{n-1} \\ a_{n-1} & a_0 & a_1 & \dots & \dots & a_{n-2} \\ a_{n-2} & a_{n-1} & a_0 & a_1 & \dots & a_{n-3} \\ \vdots & & \ddots & \ddots & & \vdots \\ a_2 & & & \ddots & \ddots & a_1 \\ a_1 & a_2 & a_3 & \dots & a_{n-1} & a_0 \end{pmatrix}$$

1. Etude du cas $n = 2$. On a $A = \begin{pmatrix} a_0 & a_1 \\ a_1 & a_0 \end{pmatrix}$.
   (a) Quelles sont les valeurs propres de A ?
   (b) A est-elle diagonalisable ?
2. Ecrire une fonction Python prenant en entrée une liste $L = [a_0, \dots, a_{n-1}]$ et renvoyant la matrice A.
3. On note P le polynôme $a_0 + a_1X + \cdots + a_{n-1}X^{n-1}$, et ω un nombre complexe vérifiant $\omega^n = 1$.

   Soit $X = \begin{pmatrix} 1 \\ \omega \\ \omega^2 \\ \vdots \\ \omega^{(n-1)} \end{pmatrix}$.

   Montrer que X est vecteur propre de A et donner la valeur propre associée.
4. Etude du cas $n = 4$.
   (a) Déterminer tous les complexes z tels que $z^4 = 1$.
   (b) Déduire de la question précédente 4 vecteurs propres de A .
   (c) On pose $P = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & i & -1 & -i \\ 1 & -1 & 1 & -1 \\ 1 & -i & -1 & i \end{pmatrix}$ et on note $\overline{P}$ la matrice conjuguée de P (c'est à dire la matrice carrée d'ordre 4 dont chaque terme de la ligne i colonne j est le conjugué du terme de la ligne i colonne j de P.

      i. Calculer $P\overline{P}$.
      ii. En déduire que les 4 vecteurs propres donnés précédement forment une famille libre.
      iii. Montrer que A est diagonalisable et donner ses valeurs propres en fonction de P.

   (d) Donner les valeurs propres et la dimension des sous-espaces vectoriels propres de

   $A = \begin{pmatrix} 2 & 1 & -2 & 1 \\ 1 & 2 & 1 & -2 \\ -2 & 1 & 2 & 1 \\ 1 & -2 & 1 & 2 \end{pmatrix}$.

Exercice 12

On considère 2 applications linéaires f, allant de $\mathbb{R}^p$ dans $\mathbb{R}^n$, et g allant de $\mathbb{R}^n$ dans R^p, où $p < n$. On appelle A matrice de f et B matrice de g, relativement aux bases canoniques de leurs espaces de départ et d'arrivée respectifs.

1. Donner les tailles de A et B.
2. Montrer que $g \circ f$ est un endomorphisme.
3. (a) Montrer que le rang de g vérifie $\mathrm{Rg}(g) \leq p$.
   (b) Montrer que $g \circ f$ n'est pas surjective.
4. On prend $n = 3$, $p = 2$ et $AB = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$.
   (a) En utilisant l'instruction `eig` (qui était rappelée) montrer que AB est diagonalisable et la diagonaliser.
   (b) Soit $X = \begin{pmatrix} x \\ y \end{pmatrix}$ non nul.
   Montrer que BX est différent de 0.
   (c) Montrer que si λ est valeur propre de AB alors λ est aussi valeur propre de BA.
   (d) BA est-elle diagonalisable ?
5. On suppose maintenant $p = 3$ et $n > 3$. On note C la matrice carrée d'ordre 3 telle que $C = AB$.
   (a) Ecrire une fonction python prenant A et B 2 matrices en argument et renvoyant les valeurs propres de AB et de BA.
   (b) Montrer que si λ est une valeur propre non nulle de C, alors λ est une valeur propre non nulle de BA.
   (c) On suppose $n = 4$ et que C a trois valeurs propres distinctes non nulles. Montrer que BA est diagonalisable

Exercice 13

Dans cet exercice , on considère une population de tortues.

1. Le nombre X d'oeufs pondus par une tortue chaque année suit une loi de Poisson de paramètre λ. Chacun d'eux a une probabilité p d'éclore . On appelle Y le nombre d'oeufs éclos .
   (a) Ecrire une fonction Python simulant une ponte et donnant le nombre d'oeufs éclos.
   (b) Déterminer la loi conditionnelle de Y sachant $(X = k)$.
   (c) En déduire la loi de Y. Donner l'espérance de Y.
2. Des études sur ce type de tortue ont permis de déterminer que :
   — les tortues deviennent adultes à 2 ans, et que seules 20% parviennent à cet âge
   — 40% des tortues adultes de l'année n meurent avant la fin de l'année
   — les femelles composent la moitié de la population et donnent naissance à 4 bébés chaque année, de l'âge de 2 ans jusqu'à la fin de leur vie .

   On définit a_n comme le nombre d'adultes vivant l'année n, et b_n le nombre de bébés de cette même année.
   (a) Déterminer la valeur de a_{n+2} en fonction de a_{n+1} et b_n ainsi que celle de b_{n+1} en fonction de a_n .
   (b) On note $E = \{(u_n)\mathbb{C}^{\mathbb{N}} \quad tq \quad \forall n \in \mathbb{N} \quad u_{n+3} - 0.6u_{n+2} - 0.4u_n = 0\}$. Montrer que E est un $\mathbb{R}$-espace vectoriel et que la suite $(a_n)_{n\in\mathbb{N}}$ est élément de E.
   (c) On considère que l'on a comme conditions initiales $a_0 = 8000$, $a_1 = 7700$ et $a_2 = 7400$.
   Ecrire une fonction Python de paramètre n qui renvoie a_n.
   La suite (a_n) semble-t-elle converger ?
   (d) i. Donner les racines réelles et complexes du polynôme $P = X^3 - 0.6X^2 - 0.4$.
   ii. Prouver que si r est racine de P, la suite des puissances de r appartient à E.
   iii. Montrer que l'application ϕ, qui à toute suite u appartenant à E associe (u_0, u_1, u_2) est un isomorphisme de E sur $\mathbb{C}^3$.
   En déduire que E est de dimension 3.
   iv. Notons r_1, r_2, r_3 les trois racines de P. On admet que la famille $((r_1^n), (r_2^n), (r_3^n))$ est libre dans E. Montrer que c'est une base de E.
   v. En déduire la convergence de la suite $(a_n)_{n\in\mathbb{N}}$.(On ne demande pas la valeur explicite de la limite.)

Exercice 14
Soient deux matrices carrées A et B appartenant à $M_n(\mathbb{R})$ telles que :
$A \neq 0$, $B \neq 0$, $AB = BA = 0$, $A^2 = A$ et $B^2 = B$.

Soit $F = \{M \in M_n(\mathbb{R}) \text{ telles que } M = xA + yB, (x, y) \in R^2\}$

1. Montrer que F est un espace vectoriel de dimension 2.
2. (a) Trouver 2 matrices A et B appartenant à $M_n(\mathbb{R})$, où $n \geq 3$ telles que la matrice identité I_n n'appartienne pas à F.
   (b) Trouver 2 matrices A et B appartenant à $M_n(\mathbb{R})$, où $n \geq 3$ telles que la matrice identité I_n appartienne à F.
3. Faire un programme permettant de vérifier que 2 matrices A et B respectent les 3 conditions du début. (On rappelle la fonction produit matriciel `dot` et celle permettant de créer des matrices nulles `zeros` .)
4. On considère dans cette question seulement les deux matrices $A = \begin{pmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ -1 & 1 & 1 \end{pmatrix}$ et $B = \begin{pmatrix} 0 & 0 & 0 \\ -1 & 1 & 0 \\ 1 & -1 & 0 \end{pmatrix}$.
   Montrer qu'elles respectent les conditions $A \neq 0$, $B \neq 0$, $AB = BA = 0$, $A^2 = A$ et $B^2 = B$.
   Donner les coordonnées de la matrice identité dans la base (A, B) de F
5. On revient au cas général où A et B sont deux matrices carrées d'ordre n telles que $A \neq 0$, $B \neq 0$, $AB = BA = 0$, $A^2 = A$ et $B^2 = B$.

   (a) On suppose que la matrice identité appartient à F . Donner ses coordonnées dans la base (A, B) de F.
   (b) En déduire une CNS sur A et B pour que la matrice identité appartienne à F.
6. On suppose que la matrice identité appartient à F. Donner les conditions pour qu'une matrice de F soit inversible.
7. Déterminer toutes les matrices M de F qui vérifient $M^2 = M$.
8. Soit $M \in F$. A l'aide de la formule du binôme, calculer M^n pour $n \in \mathbb{N}$.
9. On souhaite trouver deux matrices A et B non triviales vérifiant $A \neq 0$, $B \neq 0$, $AB = BA = 0$, $A^2 = A$ et $B^2 = B$.
   (a) Si A est diagonalisable , que peut-on dire de ses valeurs propres ? Donner les formes possibles de A.
   (b) A l'aide de Python, proposer deux matrices A et B non triviales vérifiant les conditions.

Exercice 15
Soit f un endomorphisme de $\mathbb{R}^3$ de matrice $A = \begin{pmatrix} 2 & 1 & 1 \\ 1 & 2 & 1 \\ 0 & 0 & 3 \end{pmatrix}$.

1. Montrer que $Spec(f) = \{1, 3\}$;
2. On donne les trois vecteurs de $\mathbb{R}^3$ suivants : $a_1 = (1, 1, 0), a_2 = (0, 0, 1), a_3 = (1, -1, 0)$.
   (a) Montrer que (a_1, a_2, a_3) est une base de $\mathbb{R}^3$.
   (b) Déterminer la matrice M de f dans cette base.
   (c) Déterminer P tel que $A = PMP^{-1}$ et vérifier votre calcul avec Python.
   On rappelle ici comment calculer l'inverse d'une matrice en Python.
3. On considère le système d'équations différentielles linéaires :

$$\begin{cases} f'(t) & = & 2f(t) + g(t) + h(t) \\ g'(t) & = & f(t) + 2g(t) + h(t) \\ h'(t) & = & 3h(t) \end{cases}$$

   (a) Déterminer $h(t)$ en fonction de t.
   (b) On pose $X(t) = \begin{pmatrix} f(t) \\ g(t) \\ h(t) \end{pmatrix}$,$Y(t) = \begin{pmatrix} u(t) \\ v(t) \\ w(t) \end{pmatrix}$, $X'(t) = \begin{pmatrix} f'(t) \\ g'(t) \\ h'(t) \end{pmatrix}$, $Y'(t) = \begin{pmatrix} u'(t) \\ v'(t) \\ w'(t) \end{pmatrix}$ de sorte que $Y(t) = P^{-1}X(t)$. On admet qu'on a alors $Y'(t) = P^{-1}X'(t)$.
   Vérifier qu'il existe λ réel tel que $u'(t) = u(t) + \lambda e^{3t}$.
   (c) En déduire les formes de $f(t), g(t), h(t)$.
   (d) Représenter graphiquement sur $[0, 1]$ l'application $x \mapsto \exp(3x)$.

Exercice 16

$\forall n \in \mathbb{N} \quad u_n = \int_0^{\frac{\pi}{4}} \tan^{2n+2}(t)\, dt.$

1. Ecrire en Python une fonction permettant l'affichage graphique de $\tan^2, \tan^4, \tan^6, \tan^8$.
2. Faire une conjecture sur la monotonie et l'existence d'une limite de la suite (u_n) et la démontrer.
3. Montrer que $\forall n \in \mathbb{N} \quad u_{n+1} + u_n = \frac{1}{2n+3}$ et $u_n \geq \frac{1}{2(2n+3)}$.
   Donner la limite de (u_n) lorsque n tend vers $+\infty$.
4. De même qu'à la question précédente, en utilisant la valeur de $u_{n+1} + u_n$, donner un majorant de u_n.
5. En déduire un équivalent de u_n lorsque n tend vers $+\infty$.
6. La série de terme général u_n est-elle convergente ?
7. Ecrire une fonction prenant n en argument et renvoyant la valeur de u_n.
8. Soit $S_n = \sum\limits_{k=0}^{n} \frac{(-1)^k}{2k+1}$.
   (a) Montrer que $\forall t \in \left[0, \frac{\pi}{4}\right] \quad (1 + \tan^2 t) \sum\limits_{k=0}^{n} (-\tan^2)^k(t) = 1 + (-1)^n \tan^{2n+2}(t)$.
   (b) Montrer que $S_n = \frac{\pi}{4} + (-1)^n u_n$.
   (c) En déduire un algorithme permettant d'avoir une valeur approchée de π à ϵ près.

Exercice 17

Soient p et q deux réels compris entre 0 et 1 tels que $p+q \leq 1$.
On considère la fonction f définie sur $[0,1]$ par $f(x) = q+(1-p-q)\,x+p\,x^2$,_ ainsi que la suite (v_n) définie par la donnée de $v_0 = 0$ et la relation $\forall n \in \mathbb{N} \quad v_{n+1} = q + (1-p-q)\,v_n + p\,v_n^2$.

1. Écrire une fonction Python qui affiche sur le même graphique une dizaine de courbes représentatives de f pour p égal à 0.1 , 0.2 , ..., 0.9 et q choisi aléatoirement dans $[0,1]$.
   (Rappel : on peut utiliser la fonction `random` du module `random` pour choisir q.)
   Tracer sur le même graphique la droite d'équation $y = x$.
2. Démontrer que $\forall n \in \mathbb{N} \quad v_n \in [0,1]$.
3. Montrer que $f(x) = x$ ssi $x = 1$ ou $x = q/p$.
4. Si $q < p$, montrer la convergence de (v_n) et donner sa limite .(On pourra s'aider du graphique et de la droite d'équation $y = x$)
5. Que peut on dire dans le cas où q est supérieur ou égal à p ?
6. On s'intéresse à l'extinction d'une population de bactéries procaryotes dans un écosystème donné répondant au modèle suivant.
   L'évolution est supposée réalisée par étapes successives, suivant chacune le même fonctionnement ; à chaque étape donnée, chaque bactérie, indépendamment des autres peut :
   * soit donner lieu à une fission binaire, et se diviser en deux bactéries identiques indépendantes, ceci avec une probabilité p ;
   * soit engendrer une seule bactérie avec une probabilité $1-p-q$;
   * soit mourir et se désintégrer avec une probabilité q.

   On appelle X_n la variable aléatoire égale au nombre de bactéries présentes après la n-ième étape.
   Au départ, il n'y a qu'une seule bactérie dans l'écosystème, et on note $X_0 = 1$.
   Pour tout entier naturel n, on note $U_n = P(X_n = 0)$.
   En considérant un système complet d'événements lié à X_1, montrer que la suite (U_n) correspond à la suite (v_n) étudiée précédemment. Quelle interprétation faire des résultats précédemment démontrés ?

Exercice 18

Pour $x > 0$, on pose $f(x) = \frac{e^{-x}}{x}$. On définit ensuite la suite (u_n) par $u_0 = 1$ et pour $\forall n \geq 0 \quad u_{n+1} = f(u_n)$.

1. (a) Etudier les variations de f.
   (b) Montrer que la suite (u_n) est bien définie.
2. (a) Montrer que l'équation $f(x) = x$ a une unique solution . On l'appelle a.
   (b) Monter que $\frac{1}{e} < a < 1$.
   (c) Ecrire une fonction donnant une valeur approchée de a à 10^{-3} près.
   (La méthode de dichotomie était rappelée .)
3. (a) Montrer que $a < u_0 < u_2$ et $u_3 < u_1 < a$.
   (b) Montrer que les suites (u_{2n}) et (u_{2n+1}) sont monotones .
   Peuvent-elles converger vers une même limite ?
   Qu'en déduit-on concernant la convergence de (u_n) ?
4. On pose, pour $x \geq 0$, $h(x) = \begin{cases} (f \circ f)(x) & si \quad x > 0 \\ 0 & si \quad x = 0 \end{cases}$.
   (a) Expliciter $h(x)$ pour $x > 0$. Montrer que h est continue en 0.
   (b) Résoudre $h(x) = x$.
   (c) Montrer que la suite (u_{2n+1}) est convergente et préciser sa limite.
   (d) Montrer que la suite (u_{2n}) diverge vers $+\infty$.

Exercice 19
Pour tout entier naturel n on définit la fonction f_n par : $\forall x \geq n \quad f_n(x) = \displaystyle\int_n^x e^{\sqrt{t}} dt$.

1. (a) Montrer que f_n est dérivable sur son domaine de définition et donner f_n'.
   (b) A l'aide d'une minoration de f_n, montrer que $\lim\limits_{x \to +\infty} f_n(x) = +\infty$
   (c) En déduire que pour tout entier naturel n, l'équation $f_n(x) = 1$ a une unique solution.
   On note dans la suite u_n cette solution.
   (a) Montrer que (u_n) diverge vers $+\infty$.
   (b) Montrer que $\forall n \in \mathbb{N} \quad e^{-\sqrt{u_n}} \leq u_n - n \leq \mathrm{e}^{-\sqrt{n}}$.
   (c) A l'aide de la question précédente, écrire un programme Python affichant une valeur de n telle que $u_n - n < 10^{-4}$.
2. Pour tout entier naturel n, on pose $v_n = u_n - n$.
   (a) Montrer que (v_n) converge vers 0.
   (b) Montrer que pour tout réel x tel que $x \geq -1$ on a $\sqrt{1+x} \leq 1 + \frac{x}{2}$.
   (c) Montrer que pour tout n de $\mathbb{N}^*$ on a $e^{-\sqrt{u_n}} \geq e^{-\sqrt{n}} e^{-\frac{v_n}{2\sqrt{n}}}$.
   (d) A l'aide de la question 2)b), montrer que $u_n - n$ équivaut à $e^{-\sqrt{n}}$ lorsque n tend vers $+\infty$.