Feuille Oraux_6 : Des difficultés observées aux oraux blancs.

EX 1 : *Tirage sans remise utilisation de* `L.pop()`.

Exemple 1 : On fait un tirage sans remise de k boules dans une urne en contenant n_1 rouges et n_2 vertes.

1) Ecrire une fonction qui simule cette expérience.
2) On tire 5 boules sans remise d'une urne contenant 10 boules rouges et 20 boules vertes.
   On note X le nombre de boules rouges tirées.
   a. Estimer la loi de X.
   b. Estimer l'espérance de X.

Exemple 2 : Que fait la fonction suivante ?

```
def mystere(L, S):
    n = len(L)
    for k in range(n):
        S.append(L.pop())
```

Quelle affichage obtient-on avec les instructions suivantes ?

```
A = [2, 3, 4]
B = [1, 3, 5, 6]
mystere(A, B)
print(A)
print(B)
```

EX 2 : Reprenons un à un les éléments du formulaire Python :

PYTHON
AGRO-VETO
2023

Listes

`[]`	Créer une liste vide	`L.pop(k)`	Renvoie le $k^{\text{ème}}$ élément de la liste `L` et l'enlève de `L`
`[a]*n`	Créer une liste avec n fois l'élément `a`	`L.remove(a)`	Enlève une fois la valeur `a` de la liste `L`
`L.append(a)`	Ajoute l'élément `a` à la fin de la liste `L`	`max(L)`	Renvoie le plus grand élément de la liste `L`
`L1 + L2`	Concatène les deux listes `L1` et `L2`	`min(L)`	Renvoie le plus petit élément de la liste `L`
`len(L)`	Renvoie le nombre d'éléments de la liste `L`	`sum(L)`	Renvoie la somme de tous les éléments de la liste `L`

Numpy.linalg

`import numpy.linalg as la`

`la.inv(M)`	Renvoie l'inverse de la matrice M si elle est inversible
`la.eigvals(M)`	Renvoie la liste des valeurs propres de M
`la.eig(M)`	Renvoie un couple `L,P` où `L` est la liste des valeurs propres de M et `P` la matrice de passage associée
`la.matrix_rank(M)`	Renvoie le rang de M

Numpy

`import numpy as np`

`np.array()`	Transforme une liste en matrice `numpy`
`np.linspace(a,b,n)`	Crée une matrice ligne de n valeurs uniformément réparties entre a et b (inclus)
`np.zeros([n,m])`	Crée la matrice nulle de taille $n \times m$
`np.eye(n)`	Crée la matrice identité de taille n
`np.diag(L)`	Crée la matrice diagonale dont les termes diagonaux sont les éléments de la liste `L`
`np.transpose(M)`	Renvoie la transposée de M
`np.dot(M,P)`	Renvoie le produit matriciel MP
`np.sum(M)`	Renvoie la somme de tous les éléments de M
`np.prod(M)`	Renvoie le produit de tous les éléments de M
`np.max(M)`	Renvoie le plus grand élément de M
`np.min(M)`	Renvoie le plus petit élément de M
`np.shape(M)`	Renvoie dans un couple le format de la matrice M
`np.size(M)`	Renvoie le nombre d'éléments de M

Random

`import random as rd`

`rd.random()`	Simule une réalisation d'une variable $X \hookrightarrow \mathcal{U}([0,1])$
`rd.randint(a,b)`	Simule une réalisation d'une variable $X \hookrightarrow \mathcal{U}([\![a,b]\!])$
`rd.gauss(0,1)`	Simule une réalisation d'une variable $X \hookrightarrow \mathcal{N}(0,1)$
`rd.choice(L)`	Choisit aléatoirement un élément de la liste `L`

Math

`import math as m`

`m.atan(x)`	Renvoie $\arctan(x)$	`m.sqrt(x)`	Renvoie $\sqrt{x}$ si $x \geq 0$
`m.floor(x)`	Renvoie $\lfloor x \rfloor$	`m.log(x)`	Renvoie $\ln(x)$ si $x > 0$
`m.factorial(n)`	Renvoie $n!$ si $n \in \mathbb{N}$	`m.exp(x)`	Renvoie e^x

Logique

`a == b`	Teste l'égalité « $a = b$ »
`a != b`	Teste « $a \neq b$ »
`a < b`	Teste « $a < b$ »
`a <= b`	Teste « $a \leq b$ »
`a > b`	Teste « $a > b$ »
`a => b`	Teste « $a \geq b$ »
`not A`	Renvoie la négation de A
`A and B`	Renvoie « A et B »
`A or B`	Renvoie « A ou B »
`True`	Constante booléenne « Vrai »
`False`	Constante booléenne « Faux »

Matplotlib.pyplot

`import matplotlib.pyplot as plt`

`plt.plot(X,Y,'+-r')`	Génère la courbe des points définis par les listes `X` et `Y` (abscisses et ordonnées) avec les options : • symbole : `'.'` point, `'o'` rond, `'h'` hexagone, `'+'` plus, `'x'` croix, `'*'` étoile, … • ligne : `'-'` trait plein, `'--'` pointillé, `'-.'` alterné, … • couleur : `'b'` **bleu** , `'r'` **rouge** , `'g'` **vert** , `'c'` **cyan** , `'m'` **magenta** , `'k'` **noir** , …
`plt.bar(X,Y)`	Génère l'histogramme des points définis par les listes `X` et `Y` (abscisses et ordonnées)
`plt.axis('equal')`	Rend le repère orthonormé
`plt.xlim(xmin,xmax)`	Fixe les bornes de l'axe des abscisses
`plt.ylim(ymin,ymax)`	Fixe les bornes de l'axe des ordonnées
`plt.show()`	Affiche le graphique

Cette liste est non exhaustive. Les candidats sont libres d'utiliser les commandes de leur choix.

D'autres opérations ou fonctions peuvent être utiles :

`a in L` `a%b` `a//b` `m.sin(x)` `m.cos(x)` `m.tan(x)` `rd.shuffle(L)` `np.ones([n, m])`

`sorted(L)` `L.count(x)` `m.asin(x)` `m.acos(x)` `m.comb(n, k)` `rd.sample(L, k)`

`all(L)` `any(L)` `M.all()` `M.any()`

EX 3 : Suites.

On sait que : $\forall n \in \mathbb{N}, \quad r_{n+1} = \dfrac{1}{2} r_n + b_n \quad$ et $\quad r_n + b_n = 1$

Déterminer une expression de r_n en fonction de n.

EX 4 : Trigonométrie.

1) Rappeler les formules de trigonométrie au programme.
2) Calculer l'intégrale $\displaystyle\int_0^{\frac{\pi}{2}} \cos^2(t)\,\mathrm{d}t$.
3) Démontrer que : $\quad \sin(a)\cos(b) = \dfrac{1}{2}\Big(\sin(a+b) + \sin(a-b)\Big)$

EX 5 : Dichotomie.

1) Ecrire une fonction Python qui pour un entier n renvoie l'unique solution de $\cos(x) = x^n$ sur $[0,1]$.
   on notera α_n cette solution.
2) Représenter graphiquement avec un programme Python cette suite (α_n) pour n entre 1 et 20.

EX 6 : Calcul de limite, continuité.

1) Soit f définie sur $\mathbb{R}$ par $\quad f(x) = \dfrac{\sin(x)}{x}$ si $x \neq 0 \quad$ et $\quad f(0) = 1$.

   Montrer que f est continue sur $\mathbb{R}$.

2) Dresser le tableau de variations de $f : x \longmapsto xe^{-x}$ sur $\mathbb{R}$.
   Quel est, suivant la valeur de $k \in \mathbb{R}$, le nombre de solutions de l'équation $\ f(x) = k$?

3) Soit $0 < x < \delta$ et (u_n) une suite réelle tels que : $u_n \sim n\left(\dfrac{x}{\delta}\right)^{2^n}$
   Quelle est la limite de (u_n) ?

EX 7 : Algèbre linéaire.

1) Soit a un réel et f_a l'endomorphisme de $\mathbb{R}^3$ dont la matrice dans la base canonique est $\begin{pmatrix} a & 0 & a \\ 0 & 1 & 0 \\ a & 0 & a \end{pmatrix}$,

   Déterminer une base du noyau et de l'image de f_a. *(On discutera suivant les valeurs de a)*

2) On note $E = \mathbb{R}_{n-1}[X]$ muni de sa base canonique $(1, \dots, X^{n-1})$ et $\mathbb{R}^n$ muni de sa base canonique $(e_1, \dots, e_n)$.
   On considère $a_1, a_1, \dots, a_n$ des réels deux à deux distincts et l'application T de E dans $\mathbb{R}^n$ définie par :

$$T(P) = (P(a_1), P(a_2), \dots, P(a_n))$$

   Montrer que T est un isomorphisme de E sur $\mathbb{R}^n$.

EX 8 : Intégration par parties.

1) Montrer que la fonction $g : x \longmapsto x\sqrt{x}$ est de classe C^1 sur $\big[0\,,1\big]$.
2) Montrer que pour tout $k \in \mathbb{N}$,

$$U_{k+1} = \frac{2(2k+1)}{k+2} U_k$$

   Indication : Calculer $4U_k - U_{k+1}$ en utilisant une intégration par parties.

3) Exprimer U_k en fonction de k avec des factoriels.
4) En utilisant la formule de Stirling : $n! \underset{n\to+\infty}{\sim} \dfrac{n^n}{e^n}\sqrt{2\pi n}$ montrer que :

$$U_k \underset{k\to+\infty}{\sim} \frac{4^k}{k\sqrt{\pi k}}$$