PLANCHES d'ORAL de MATH-INFO, CENTRALE

Planche 1

1. Soit le polynôme $P = X^5 + X^4 - 10X^3 + 6X^2 + 9X - 7$, soit $M = \begin{pmatrix} 0 & 0 & 0 & 0 & 7 \\ 1 & 0 & 0 & 0 & -9 \\ 0 & 1 & 0 & 0 & -6 \\ 0 & 0 & 1 & 0 & 10 \\ 0 & 0 & 0 & 1 & -1 \end{pmatrix}$.

a. Trouver les racines entières de P. En déduire toutes les racines de P.

b. Calculer $u_n = \dfrac{\mathrm{tr}(M^{n+1})}{\mathrm{tr}(M^n)}$ pour n de 1 à 40. Que peut-on conjecturer ?

c. Montrer que P est le polynôme caractéristique de M.

2. Soit $d \in \mathbb{N}^*$, soit $P = X^d + \displaystyle\sum_{i=0}^{d-1} a_i X^i$ un polynôme unitaire de degré d dans $\mathbb{R}[X]$.

a. Montrer que P est le polynôme caractéristique de $M = \begin{pmatrix} 0 & \cdots & \cdots & 0 & -a_0 \\ 1 & \ddots & & \vdots & -a_1 \\ 0 & \ddots & \ddots & 0 & \vdots \\ \vdots & \ddots & \ddots & 0 & -a_{d-2} \\ 0 & \cdots & 0 & 1 & -a_{d-1} \end{pmatrix}$.

b. Exprimer $u_n = \dfrac{\mathrm{tr}(M^{n+1})}{\mathrm{tr}(M^n)}$ à l'aide des valeurs propres complexes $\lambda_1, \cdots, \lambda_d$ de M.

c. On suppose que λ_1 est l'unique valeur propre de module maximal de M, on a donc $\forall i \in [\![2, d]\!] \quad |\lambda_i| < |\lambda_1|$. Montrer que λ_1 est réel et que $\lim\limits_{n\to+\infty} u_n = \lambda_1$.

Planche 2

Pour $n \geq 2$, on considère la matrice $M_n \in \mathcal{M}_n(\mathbb{R})$ formée "en serpent" par les entiers de 1 à n^2,

ainsi $M_2 = \begin{pmatrix} 1 & 2 \\ 4 & 3 \end{pmatrix}$, $M_3 = \begin{pmatrix} 1 & 2 & 3 \\ 6 & 5 & 4 \\ 7 & 8 & 9 \end{pmatrix}$, $M_4 = \begin{pmatrix} 1 & 2 & 3 & 4 \\ 8 & 7 & 6 & 5 \\ 9 & 10 & 11 & 12 \\ 16 & 15 & 14 & 13 \end{pmatrix}$.

1. Écrire une fonction `f(n,i,j)` retournant le coefficient d'indices (i, j) de la matrice M_n.

2. Écrire une fonction `M(n)` retournant la matrice M_n.

3. Avec Python, afficher le rang de M_n pour $n \in [\![2, 10]\!]$. Faire une conjecture, et la démontrer.

4. Afficher les valeurs de $\dfrac{\mathrm{tr}(M_n)}{n^3}$ pour n de 2 à 100. Commenter.

5. Pour $i \in [\![1, n]\!]$, déterminer $\min\limits_{1\leq j\leq n} (M_n)_{i,j}$ et $\max\limits_{1\leq j\leq n} (M_n)_{i,j}$. En déduire un équivalent de $\mathrm{tr}(M_n)$.

Planche 3

Soit la matrice $A = \begin{pmatrix} -7 & -3 & 0 \\ 54 & 8 & -36 \\ 3 & -3 & -10 \end{pmatrix}$. Soit le vecteur $X_0 = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}$.

1. Pour tout entier naturel n, on note X_n le n-ième itéré de X_0 par A (ou par l'endomorphisme canoniquement associé), c'est-à-dire $X_n = A^n X_0$. Faire afficher les X_n pour n de 0 à 15 (*valeurs approchées*). Que remarque-t-on ?

2. Pour travailler avec des valeurs numériques "raisonnables", on norme les X_n à chaque étape, autrement dit on construit une suite de vecteurs (Y_n), avec $Y_0 = \dfrac{X_0}{\|X_0\|}$, puis $Y_{n+1} = \dfrac{AY_n}{\|AY_n\|}$.

Pour n de 1 à 15, afficher le vecteur Y_n ainsi que le produit scalaire $(AY_n|Y_n)$. Conclusion ?

3. Tester avec d'autres vecteurs "initiaux" X_0 construits aléatoirement.

4. En prenant $X_0 = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}$, qu'obtient-on avec 20 itérations ? avec 40 itérations ?

5. Utiliser les fonctions du module `numpy.linalg` pour diagonaliser la matrice A.

6. Comment interpréter les résultats des calculs précédents ?

Les notations $(\cdot|\cdot)$ et $\|\cdot\|$ représentent respectivement le produit scalaire canonique et la norme euclidienne canonique sur $\mathbb{R}^n$.

Planche 4

Pour $n \in \mathbb{N}^*$, on considère la matrice $A_n = \begin{pmatrix} 1 & \frac{1}{2} & \frac{1}{3} & \cdots & \frac{1}{n+1} \\ \frac{1}{2} & \frac{1}{3} & \frac{1}{4} & \cdots & \frac{1}{n+2} \\ \vdots & \vdots & \vdots & & \vdots \\ \frac{1}{n+1} & \frac{1}{n+2} & \frac{1}{n+3} & \cdots & \frac{1}{2n+1} \end{pmatrix} \in \mathcal{M}_{n+1}(\mathbb{R})$.

1. Écrire une fonction `matrice(n)` qui affiche A_n.

2. Avec Python, afficher les valeurs propres de A_1, A_2, A_3, A_4. Quelle conjecture peut-on faire sur le spectre de A_n ?

3. La matrice A_n est-elle diagonalisable ? Montrer que $\mathrm{Sp}(A_n) \subset \mathbb{R}_+$. *On remarquera que* $\dfrac{1}{k+1} = \displaystyle\int_0^1 t^k \,\mathrm{d}t$ *pour tout k entier naturel.*

On notera m_n la plus petite valeur propre de A_n.

4. Écrire une fonction `pluspetiteVP(n,p)`, qui prend en arguments deux entiers naturels n et p, et qui représente $k \mapsto m_k$ pour $k \in [\![n, p]\!]$. Tester avec $n = 1$ et $p = 5$, puis avec $n = 5$ et $p = 10$. Que peut-on conjecturer ?

5. Montrer que $\displaystyle\lim_{n \to +\infty} m_n = 0$.

6. On note $S^n = \big\{X \in \mathbb{R}^{n+1} \mid \|X\| = 1\big\}$, où $\|\cdot\|$ représente la norme euclidienne canonique sur $\mathbb{R}^{n+1}$, identifié à $\mathcal{M}_{n+1,1}(\mathbb{R})$. Justifier l'existence de $M_n = \displaystyle\sup_{X \in S^n} X^\top A_n X$.

Planche 5

Soit la fonction $F : x \mapsto \displaystyle\int_0^x e^{t^2} \,\mathrm{d}t$.

1. Montrer que F est de classe $\mathcal{C}^1$ et bijective de $\mathbb{R}$ vers $\mathbb{R}$.

2. Montrer que $\forall x \in \mathbb{R} \quad \exists! y(x) \in \mathbb{R} \qquad \displaystyle\int_x^{y(x)} e^{t^2} \,\mathrm{d}t = 1.$

3. Programmer sur Python une fonction prenant x comme argument et renvoyant $F(x)$.

4. Programmer sur Python une fonction prenant x comme argument et renvoyant une valeur approchée de $y(x)$ à 10^{-2} près.

Planche 6

Pour $x \in \mathbb{R} \setminus \{\ln(2)\}$, on pose $f(x) = \dfrac{1}{2 - e^x}$. Pour n entier naturel, soit $a_n = \dfrac{f^{(n)}(0)}{n!}$.

1. Pour n entier naturel non nul, montrer que $a_n = \displaystyle\sum_{k=1}^{n} \frac{a_{n-k}}{k!}$. *On pourra utiliser la relation* $(2 - e^x)\, f(x) = 1$.

2. Représenter graphiquement les dix premiers termes de la suite (a_n).

3. Sur le même graphique, faire apparaître les dix premiers termes des suites $b_n = \dfrac{1}{(\ln 2)^n}$ et $c_n = \dfrac{1}{2\,(\ln 2)^n}$. Que peut-on conjecturer ?

4. Démontrer cette conjecture.

5. Montrer que f est développable en série entière sur $]-\ln(2), \ln(2)[$.

Planche 7

Soit l'équation différentielle $(\mathbf{E}) : \quad xy'' + y' + y = 0$.

1. Que dire de l'ensemble des solutions de **(E)** sur $\mathbb{R}_+^*$? Que dire de l'ensemble des solutions y de **(E)** sur $\mathbb{R}_+^*$ si l'on impose de plus $y(1) = 1$ et $y'(1) = 0$?

2. Tracer sur un intervalle raisonnable la solution de **(E)** telle que $y(1) = 1$ et $y'(1) = 0$. Représenter des solutions de **(E)** avec d'autres valeurs de $y(1)$ et $y'(1)$.

3. Trouver les solutions de **(E)** développables en série entière sur $\mathbb{R}$. Montrer qu'il en existe une et une seule prenant la valeur 1 en 0, on la notera f par la suite.

4. Montrer que f est décroissante sur le segment $[0, 2]$.

5. Donner une valeur approchée de $f(2)$ à 10^{-4} près.

6. Représenter f avec Python.

Planche 8

Pour $n \in \mathbb{N}^*$, soit la fonction $f_n : x \mapsto x^n + x^{n-1} + \cdots + x - 1 = \displaystyle\sum_{k=1}^{n} x^k - 1$.

1. Écrire une fonction `courbe(n)` permettant de représenter f_n sur $[0, 1]$.

2. Montrer que la fonction f_n admet un unique zéro, que l'on notera u_n, dans l'intervalle $[0, 1]$.

3. Écrire une fonction qui prend en argument un entier n et qui retourne une valeur approchée de u_n à 10^{-5} près.

4. Conjecturer le comportement de la suite (u_n).

5. Démontrer votre conjecture.

Planche 9

$$\text{Soient } f : x \mapsto \sum_{n=1}^{+\infty} \frac{x^n}{n^2} \quad \text{et} \quad g : x \mapsto \int_0^x \frac{\ln(1-t)}{t}\, \mathrm{d}t.$$

1. Donner le domaine de définition de f, et tracer sa courbe représentative à la précision 10^{-5}.

2. Montrer que g est définie sur $[-1,1]$, de classe $\mathcal{C}^1$ sur $]-1,1[$ et tracer g à l'aide de Python. Que peut-on conjecturer ?

3. Calculer f' sur $]-1,1[$. En déduire que $f+g=0$ sur $[-1,1]$.

4. À l'aide d'une intégration par parties et d'un changement de variable, établir une relation entre $g(x)$, $g(1-x)$ et $g(1)$.

5. En déduire la valeur de $S = \displaystyle\sum_{n=1}^{+\infty} \frac{1}{2^n\, n^2}$.

Planche 10

1. Écrire une fonction Python qui, pour n entier naturel passé comme argument, crée la liste des entiers de 0 à $n-1$, puis qui sélectionne un élément au hasard, le retire de cette liste et le place seul dans une autre liste.

2. Écrire une fonction Python qui réalise l'opération précédente jusqu'à ce que la liste initiale soit vide, et retourne la liste des éléments qui en ont été retirés, dans leur ordre de tirage. Que simule-t-on de cette façon?

3. On note $\mathcal{S}_n$ l'ensemble des permutations de l'ensemble $[\![0, n-1]\!]$, c'est-à-dire des bijections de $[\![0, n-1]\!]$ vers lui-même.

a. Quel est le cardinal de $\mathcal{S}_n$?

b. Écrire une fonction qui prend comme argument une permutation $f \in \mathcal{S}_n$ (codée sous une forme de votre choix) et qui retourne le nombre de ses points fixes.

c. On note F_n la variable aléatoire qui compte le nombre de points fixes d'une permutation de $\mathcal{S}_n$. Faire une simulation de $N = 500$ tirages (avec remise) de permutations de $\mathcal{S}_n$ avec $n = 10$. Calculer pour ces tirages les valeurs de F_n, la moyenne observée des valeurs de F_n. Répéter l'expérience avec $N = 1000$ et $n = 1000$. En déduire une conjecture sur l'espérance de F_n.

d. Démontrer cette conjecture. *On pourra introduire les variables aléatoires X_k, $0 \leq k \leq n-1$, telles que $X_k(f) = 1$ si k est un point fixe de f, et $X_k(f) = 0$ sinon.*

4.a. Vérifier expérimentalement $\mathrm{E}(F_n^2) = 2$, puis le démontrer mathématiquement.

b. Quelle est la variance de F_n ?