

Programme de colles : semaine 15, du 20/1 au 24/1

Les nouveautés par rapport à la semaine précédente sont en bleu.

1 Systèmes linéaires

Aucune interprétation géométrique n'a été abordée en classe.

- définition d'un système linéaire, système homogène, échelonné, de Cramer, compatible
- résolution d'un système échelonné, notion de variables principales et auxiliaires
- algorithme du pivot de Gauss.
- notion d'équation de compatibilité
- structure de l'ensemble des solutions : l'ensemble des solutions d'un système homogène est stable par addition et par multiplication par un scalaire ; les solutions d'un système quelconque s'écrivent comme solution particulière + solutions du système homogène
- exemples de systèmes linéaires dépendant d'un paramètre dans le second membre et/ou dans les coefficients
- notion de rang d'un système linéaire, un système carré de taille p est de Cramer si et seulement s'il est de rang p
- exemples de calcul du rang d'un système linéaire en fonction d'un paramètre dans les coefficients

2 Suites réelles

Attention : début de chapitre uniquement. Nous n'avons pas encore vu les limites infinies, ni l'étude des suites du type $u_{n+1} = f(u_n)$. Pour les calculs de limites, je rappelle que ni les croissances comparées ni les équivalents n'ont encore été abordés en classe.

Les définitions des limites avec des quantificateurs doivent être connues, mais leur utilisation ne doit pas constituer le cœur des exercices demandés.

- monotonie (méthodes $u_{n+1} - u_n$, $\frac{u_{n+1}}{u_n}$)
- suites majorées, minorées, bornées
- suites convergentes :
 - définition avec des quantificateurs
 - unicité de la limite, toute suite convergente est bornée
 - (u_n) converge vers ℓ si et seulement si (u_{2n}) et (u_{2n+1}) convergent vers ℓ (la notion générale de suite extraite n'a pas été étudiée en classe)
 - théorème d'encadrement (ou des "gendarmes")
 - Si $|u_n| \leq \varepsilon_n$ et $\varepsilon_n \rightarrow 0$ alors $u_n \rightarrow 0$
 - théorème de la limite monotone
 - suites adjacentes

3 Informatique en langage Python

Listes :

- parcours d'une liste par indices
- sous-listes $L[p:q]$, $L[p:]$, $L[:q]$, $L[p:q:r]$ d'une liste L
- recherche d'un "facteur" dans une liste (exemple : recherche d'un mot dans un texte)
- modification d'un élément d'une liste
- suppression d'un élément d'une liste avec la commande `pop`

4 Questions de cours

Cette semaine, les colles suivront le schéma suivant :

- A) résolution d'un système linéaire à n équations et p inconnues (avec $n, p \in \llbracket 1, 4 \rrbracket$) sans paramètres
- B) calcul d'une intégrale du type $\int_a^b f(ct + d) dt$ où $c, d \in \mathbb{R}$ et f est une fonction simple, voir remédiation 9, exo 5 : <https://cahier-de-prepa.fr/bcpst1b-berthelot/download?id=5609>
- C) une ou plusieurs question de cours issue de la liste ci-dessous
- D) exercices au choix du colleur

1. Donner la définition avec des quantificateurs de : (u_n) converge vers $\ell \in \mathbb{R}$ et savoir l'expliquer sur un dessin.
2. Énoncer un des théorèmes suivants : théorème de la limite monotone, théorème d'encadrement, théorème des suites adjacentes.
3. Pour $n \geq 2$ on pose $u_n = \sum_{k=2}^n \frac{1}{k^2}$. Montrer que (u_n) converge.
On utilisera que $\forall k \geq 2, \frac{1}{k(k-1)} = \frac{1}{k-1} - \frac{1}{k}$.
4. Pour $n \in \mathbb{N}^*$ on note $a_n = \sum_{k=1}^n \frac{(-1)^k}{k}$. Montrer que la suite (a_n) converge en montrant que les suites $(u_n) = (a_{2n})$ et $(v_n) = (a_{2n+1})$ sont adjacentes.
5. Écrire une fonction Python prenant en argument une liste $L = [x_0, x_1, \dots, x_{n-1}]$ et renvoyant $\sum_{k=0}^{n-1} x_k 2^k$.
6. Écrire une fonction Python `recherche` prenant en argument une liste L et un élément de type quelconque a et renvoyant l'indice de la première occurrence de a dans la liste L . La fonction renverra "introuvable" si a n'est pas un élément de L .
7. On décrit un brin d'ADN par une liste dont les éléments sont toujours un des quatre caractères "A", "T", "C" et "G". Le début d'un message génétique est généralement signalé par le codon "A", "T", "G". Écrire une fonction `depart` qui prend en argument un brin d'ADN et renvoie l'indice du début du message génétique.
Par exemple, `depart(["C", "G", "A", "A", "T", "C", "A", "T", "G", "G", "T", "A"])` doit renvoyer 6. La fonction renverra un message d'erreur si le codon de départ n'est pas présent dans le message génétique considéré.
8. Écrire une fonction Python `transforme` prenant en argument une liste et la renvoyant après avoir transformé tous les 4 qu'elle contient en des 5.
9. Écrire une fonction Python `miroir` prenant en argument une liste et renvoyant la liste inversée. Par exemple `miroir([1,2,3])` doit renvoyer `[3,2,1]`. *Le colleur pourra imposer, ou non, que la liste prise en argument ne soit pas modifiée par l'exécution de la fonction.*

La question de cours est notée sur 10 points, le reste des exercices sur 10 autres points.