Semaine 17 : 3 au 7 février 2025

A. Applications ; applications bijectives

À rajouter cette semaine :
∗ **Applications surjectives:** définition; cas des fonctions usuelles; composée d'applications surjectives.
→ **Une application est toujours surjective de E dans $f(E)$**
∗ **Applications bijectives:** définition; cas des fonctions usuelles; composée d'applications bijectives.
∗ **Application réciproque:** définition, unicité; calcul de f^{-1}; symétrie entre les courbes représentatives de f et f^{-1}
→ courbes de arcsin, arccos et arctan
→ fonctions racines n-ième, suivant la parité de n.
∗ Soit f une application de E dans F.
f est bijective ssi il existe une application $g : F \to E$ telle que $f \circ g = id_F$ et $g \circ f = id_E$. Dans ce cas, $g = f^{-1}$.
∗ Soient $f : E \to F$ et $g : F \to G$ deux applications bijectives. Alors:
(1) f^{-1} est bijective de F dans E et: $(f^{-1})^{-1} = f$. (2) $g \circ f$ est bijective de E dans G et: $(g \circ f)^{-1} = f^{-1} \circ g^{-1}$.

B. Systèmes linéaires sans paramètres

∗ **Définitions:** système échelonné (en lignes), *pivot*, réduite de Gauss, *rang* d'un système linéaire (nb de pivots de sa réduite de Gauss)
→ *A noter (programme officiel): on admet que le rang est indépendant du choix des pivots.*
∗ **Résolution d'un système linéaire par la méthode du pivot de Gauss.**
→ *A noter (programme officiel): on se limite à la mise en pratique de la méthode; l'écriture formelle d'un algorithme de réduction n'est pas un attendu du programme*
Capacité exigible (programme officiel): mettre en place une recherche de pivots sur un système linéaire; mener une démarche de résolution d'un système linéaire.

C. Langage `python`

Boucle `while` : **algoritmes de seuil**
Bibliothèque `random` : fonctions `randrange(a,b)` ; `randint(a,b)`

Déroulement de la colle :

La colle commence par une question d'informatique (langage `python`) parmi :

1. On considère la suite (S_n) définie par:
   $\forall n \in \mathbb{N}$, $S_n = \sum_{k=0}^{n} 2^k$. Écrire une fonction `premier(M)` qui prend en argument un réel M (positif strictement) et qui renvoie le premier entier n tel que $S_n \geqslant M$.

2. Écrire une fonction `premier(M)` qui prend en argument un réel M (positif strictement) et qui renvoie le premier entier n tel que $n! \geqslant M$.

3. On considère la suite (u_n) définie par:
   $u_0 = 1$ et $\forall n \geqslant 1$, $u_{n+1} = u_n + \dfrac{1}{u_n}$. Écrire une fonction `seuil(M)` qui , pour tout réel M, renvoie le premier n tel que $u_n \geqslant M$.

4. On effectue des lancers successifs d'un dé cubique non truqué.
   Écrire une fonction `jeu(valeur)` qui prend en argument une variable `valeur` et qui renvoie le nombre de lancers nécessaires pour obtenir `valeur`.

5. Un rat de laboratoire est soumis à l'expérience suivante: il est enfermé dans une cage comportant quatre portes, derrière lesquelles se trouve un morceau de gruyère. Trois des quatre portes sont munies d'un dispositif envoyant au rat une décharge électrique s'il essaye de les franchir; la quatrième laisse le passage libre.
   On suppose que le rat n'abandonne jamais. Écrire une fonction Python qui simule l'expérience aléatoire et renvoie le nombre d'essais effectués par le rat jusqu'à ce qu'il trouve la bonne porte.

Puis la résolution d'un système linéaire par la méthode du pivot de Gauss

Puis passage aux exercices