Semaine n°21 du 17 au 21 mars

Informatique(Python) : cf exemples en annexe

- boucle `while`, boucle `for`,.
- listes en Python : création d'une liste, extraction d'un élément, parcours d'une liste, concaténation, `len`, `append`...etc
- chaîne de caractère.
- Tri par selection, tri par comptage

Probabilités

- Définitions : univers, évènements, évènements élémentaires, évènement certain, évènement impossible, évènements incompatibles, système complet d'évènements, espace probabilisable $(\Omega, \mathcal{P}(\Omega))$.
- Probabilité : définition, propriétés $\Big(P(\bar{A}) = 1 - P(A)$, $P(\varnothing) = 0$, $0 \leq P(A) \leq 1$, $A \subset B \Rightarrow P(A) \leq P(B)$, $P(A \cup B) = P(A) + P(B) - P(A \cap B)$, $P(\bigcup\limits_{i=1}^{n} A_i) = \sum\limits_{i=1}^{n} P(A_i)$ pour des évènements deux à deux incompatibles $\Big)$, espace probabilisé $(\Omega, \mathcal{P}(\Omega), P)$.
- Une probabilité est entièrement définie par sa valeur sur les évènements élémentaires.
- Probabilité uniforme.
- Probabilité conditionnelle : définition, propriété : si $P(A) \neq 0$ alors P_A est une probabilité (démonstration exigible).
- Formules des probabilités composées (simple et généralisée).
- Formule des probabilités totales (démonstration exigible pour 2 événements).
- Formule de Bayes (démonstration exigible).
- Indépendance et indépendance mutuelle.

Suites réelles

- Suite majorée, minorée, bornée, (strictement) croissante/décroissante, (strictement) monotone, constante, stationnaire.
- Définition d'une suite $(u_n)_{n\in N}$ convergente de limite ℓ :
  $\forall \epsilon > 0, \exists n_\epsilon \in \mathbb{N}, \forall n \in \mathbb{N}, n \geq n_\epsilon \Rightarrow |u_n - \ell| \leq \epsilon$
- Unicité de la limite d'une suite convergente, toute suite convergente est bornée (réciproque fausse)
- Suite divergente : définition.
- Limites des suites usuelles, opérations sur les limites, croissances comparées.
- Théorème de composition des limites : si $(u_n)_{n\in N}$ est convergente de limite ℓ et si f admet une limite L en ℓ, alors $(f(u_n))_{n\in N}$ est convergente de limite L.
- Si une suite est convergente vers une limite $l > 0$, alors à partir d'un certain rang les termes de la suites sont tous strictement positifs.
- Théorème de passage à la limite dans une inégalité, théorèmes de comparaison, théorème des gendarmes.
- Suites extraites $(u_{2n})_{n\in\mathbb{N}}$ et $(u_{2n+1})_{n\in\mathbb{N}}$. Utilisation pour montrer la convergence ou la divergence de $(u_n)_{n\in\mathbb{N}}$.
- Suites adjacentes : définition, propriété.
- suites équivalentes : définition, caractérisation en pratique, transitivité, produit, quotient, puissance, multiplication par un scalaire non nul. Deux suites équivalentes admettent les mêmes limites.
- Equivalents usuels : polynômes, et si (u_n) converge vers 0 :
  - $\sin(u_n) \underset{n\to+\infty}{\sim} u_n$

- $\tan(u_n) \underset{n\to+\infty}{\sim} u_n$
- $\ln(1+u_n) \underset{n\to+\infty}{\sim} u_n$
- $e^{u_n} - 1 \underset{n\to+\infty}{\sim} u_n$
- pour tout $\alpha \in \mathbb{R}^*$, $(1+u_n)^\alpha - 1 \underset{n\to+\infty}{\sim} \alpha u_n$ et en particulier (pour $\alpha = \frac{1}{2}$), $\sqrt{1+u_n} - 1 \underset{n\to+\infty}{\sim} \frac{u_n}{2}$
- $1 - \cos(u_n) \underset{n\to+\infty}{\sim} \frac{u_n^2}{2}$

➟ Exemples d'études guidées de suites de type $u_{n+1} = f(u_n)$

Matrices

➟ Définitions : matrice nulle, carrée, identité, ligne, colonne, diagonale, triangulaire supérieure ou inférieure.

➟ Opérations : additions de deux matrices, multilpication par un scalaire, produit de matrices.

➟ Propriétés du produit : produit avec la matrice identité ou la matrice nulle, associativité, distributivité, non commutativité, non intégrité, $AB = AC \not\Rightarrow B = C$.

Remarques aux colleurs

— Merci aussi de poser une petite question d'informatique (cf Annexe).

— Veuillez à bien vérifier que les élèves connaissent la définition d'un système complet d'événements et le théorème de probabilités totales.

— Aucun exercice n'a été fait sur les suites.

Exemples de programmes informatiques

Exercice 1

Ecrire en Python une fonction `existence` qui prend en entrée une liste L et un nombre *element* et renvoie `True` si *element* se trouve dans la liste L, `False` sinon.

```
def existence(L,element):
    n=len(L)   # taille de la liste
    for i in range(n):
        if L[i]==element:
            return True
    return False  # si on n' a pas trouvé element après avoir parcouru toute la liste
```

Exercice 2

Ecrire en Python une fonction `MaximumListe` qui prend en entrée une liste L et renvoie la plus grande valeur de cette liste

```
def MaximumListe(L):
    n=len(L)  #taille de la liste
    maxi=L[0]  #on considère temporairement que le max est le premier élément
    for i in range(n):
        if L[i]>maxi:
            maxi=L[i]  #on a trouvé une plus grande valeur
    return maxi
```

Exercice 3

Ecrire en Python une fonction `Somme` qui prend en entrée une liste L et renvoie la somme de ses éléments :

```
def Somme(L):
    n=len(L)   #taille de la liste
    S=0  #initialisation de la somme
    for i in range(n):
        S=S+L[i]
    return S
```

Exercice 4

Ecrire une fonction `experience` qui prend en paramètre un entier n et simule n lancers successifs d'une pièce de monnaie équilibrée en renvoyant une liste aléatoire composée de n valeurs égales à 0 ou 1. On considérera que 0 correspond à Face et 1 à Pile.

```
from random import *   # bibliothèque nécessaire pour créer des nombres aléatoires
def experience(n):
    L=[] #liste vide initialement
    for i in range(n):
        L.append(randint(0,1)) # 0 ou 1 choisi de manière aléatoire
    return L
```