DS 4 – Maths & Info

La calculatrice est interdite.
On pourra admettre une question (en le précisant) pour poursuivre la résolution des exercices.
La qualité de la rédaction, la clarté et la précision des raisonnements seront pris en compte dans l'évaluation, ***ainsi que le soin.***

Exercice 1. Système à paramètre

On considère le système suivant, de paramètre m et d'inconnues $x, y, z \in \mathbb{R}$:

$$(S_m) \left\{ \begin{array}{rcrcrcr} mx & - & m^2y & + & mz & = & 1 \\ x & - & my & + & m^2z & = & m \\ mx & + & y & - & m^3z & = & -1 \end{array} \right.$$

1. Résoudre soigneusement le système dans le cas $m = -1$. Quel est son rang ?
2. Discuter en fonction des valeurs de $m \in \mathbb{R}$ du nombre de solutions du système (S_m).
3. **Facultatif** : On autorise désormais le paramètre m à prendre des valeurs complexes (et on cherche les solutions dans $\mathbb{C}^3$). Pour quelles valeurs de $m \in \mathbb{C}$ le système (S_m) est-il de Cramer ?

Exercice 2.

Dans cet exercice, toutes les questions sont indépendantes.

1. Développer $(1-z)^6$ pour $z \in \mathbb{C}$. On explicitera la formule utilisée.
2. Soit $n \in \mathbb{N}$. Calculer les sommes et produits suivants. Si possible, il est demandé de donner les résultats sous forme factorisée.

   a) $\displaystyle\prod_{i=2}^{n} \exp\left(2^{-i}\right)$ b) $\displaystyle\sum_{k=1}^{n-1} \binom{n}{k} \exp(k)$ c) $\displaystyle\sum_{1 \leq i < j \leq n} \frac{i^2}{2j-1}$

3. Soit $x \in \mathbb{R}$ et $n \in \mathbb{N}$. On souhaite calculer le produit suivant :

$$P_n = \prod_{k=0}^{n} \left(1 + x^{(2^k)}\right)$$

   a) Écrire une fonction Python `calcule_produit(x,n)` qui prend en argument un réel x et un entier naturel n, et qui renvoie la valeur de P_n.
   b) Déterminer P_0 pour $x \in \mathbb{R}$ quelconque, puis calculer P_n lorsque $x = 1$.
   c) On suppose maintenant $x \neq 1$. Montrer par récurrence que pour tout $n \in \mathbb{N}$:

$$P_n = \frac{1 - x^{(2^{n+1})}}{1 - x}$$

Exercice 3. Des systèmes linéaires en Python

Soit $p \in \mathbb{N}^*$. Dans cet exercice, on représente des équations linéaires à p inconnues par des listes de taille $p+1$ contenant les coefficients de l'équation dans l'ordre, en finissant par le coefficient du second membre. Ainsi :
`[a_1, a_2, …, a_n, b]` représente l'équation $a_1x_1 + a_2x_2 + \ldots + a_px_p = b$.
Par exemple, l'équation $3x - 2y - z = 5$ est représentée par la liste `[3, -2, -1, 5]`.
De même, une solution éventuelle $(x_1, \ldots, x_p) \in \mathbb{R}^p$ est représentée par une liste de longueur p.

1. a) Écrire une fonction **`somme_combinée(A,B)`** qui prend en argument deux listes de même longueur $A = [a_0, \ldots, a_{n-1}]$ et $B = [b_0, \ldots, b_{n-1}]$ et qui renvoie la valeur de : $\sum\limits_{i=0}^{n-1} a_ib_i$.
   Par exemple : `somme_combinée([3, 1, 6], [0, -1, 2])` devra renvoyer `11`.

   b) En déduire une fonction Python **`est_solution(E, X)`**, qui prend en paramètre une liste E représentant une équation et une liste X représentant une solution éventuelle, et qui renvoie `True` si X est bien solution de l'équation E, et `False` sinon.
   Par exemple : `est_solution([3, -2, -1, 5], [1, 0, -2])` devra renvoyer `True` puisque $3 \times 1 - 2 \times 0 - 1 \times (-2) = 5$.

2. Écrire une fonction Python **`compte_zéros(E)`** qui prend en argument une liste E, et qui compte le nombre de zéros consécutifs **au début** de la liste. *Par exemple :*
   - `compte_zéros([0, 0, 1, 0, 3, 0])` devra renvoyer `2`
   - `compte_zéros([1, 0, 0, -2])` devra renvoyer `0`

3. On représente un système linéaire de n équations à p inconnues par une liste S dont les éléments sont les listes représentant les équations du système. Il s'agit donc d'une liste de listes !
   L'instruction `len(S)` renvoie alors le *nombre d'équations* du système.
   On supposera dans la suite que tous les systèmes représentés en Python ont bien un sens mathématiquement et ont au moins une équation.

   a) Donner la liste Python qui représente le système suivant : $(S)\begin{cases} x + 2y = 0 \\ 2x - y = 3 \end{cases}$

   b) Écrire une fonction **`nb_inconnues(S)`** qui prend en argument un système S et qui renvoie le nombre d'inconnues du système.

   c) Écrire une fonction **`est_solution_système(S, X)`** qui prend en argument un système S représenté de cette manière, et une liste X correspondant à une solution éventuelle, et qui renvoie un **booléen** indiquant si X est une solution du système S.

4. Compléter la fonction suivante afin qu'elle permette de déterminer si un système (S) est échelonné ou non.

```python
def est_échelonné(S):
    n = len(S)
    p = nb_inconnues(S)
    for i in range(n-1):
        if compte_zeros(S[i]) >= p:   # si la ligne i est triviale
            if compte_zeros(S[i+1]) .................. :
                return False
        else:
            if compte_zeros(S[i+1]) .................. :
                return False
    return True
```

Exercice 4. Problème : Formule de Pascal et sommes classiques

Les parties 1 et 2 du problème sont indépendantes.

1. a) Donner la définition du coefficient binomial $\binom{b}{a}$, et donner son expression factorielle pour $0 \leq a \leq b$.
   On rappelle que pour $a, b \in \mathbb{N}$ avec $a > b$, on pose par convention $\binom{b}{a} = 0$.

   b) Rappeler sans la démontrer la formule de Pascal.
2. Montrer que pour toute suite $(u_k)_{k\in\mathbb{N}}$ et pour tout $n \in \mathbb{N}$, on a : $\sum_{k=0}^{n}(u_{k+1} - u_k) = u_{n+1} - u_0$
3. Soient $p, n \in \mathbb{N}$. Montrer que :
$$\sum_{k=0}^{n} \binom{k}{p} = \binom{n+1}{p+1}$$
   On écrira $\binom{k}{p}$ sous la forme $u_{k+1} - u_k$ à l'aide de la formule de Pascal.

Partie 1 - Sommes des entiers, des carrés et des cubes.

4. Déduire de la question précédente la valeur de $\displaystyle\sum_{k=0}^{n} \binom{k}{1}$. Quel résultat retrouve-t-on ?
5. a) Déterminer des entiers $a, b \in \mathbb{Z}$ tels que pour tout $k \in \mathbb{N}$, $k^2 = a\binom{k}{2} + b\binom{k}{1}$. On procèdera par *identification*.

   b) En déduire la valeur de la somme des n premiers carrés parfaits non nuls.
6. De la même manière, retrouver la valeur de la somme $\displaystyle\sum_{k=0}^{n} k^3$.

Partie 2 - Suite harmonique

Pour tout $n \in \mathbb{N}$, on note $H_n = \displaystyle\sum_{k=1}^{n} \frac{1}{k}$.

7. Soit $n \in \mathbb{N}^*$. Montrer que $\displaystyle\sum_{k=1}^{n} H_k = (n+1)H_n - n$
8. Soient $m, n \in \mathbb{N}$. Montrer que $\displaystyle\sum_{k=1}^{n} \binom{k}{m} H_k = \sum_{i=1}^{n} \frac{1}{i} \sum_{k=i}^{n} \binom{k}{m}$.
9. En utilisant l'égalité de la question 3, en déduire que
$$\sum_{k=1}^{n} \binom{k}{m} H_k = \binom{n+1}{m+1} H_n - \sum_{i=1}^{n} \frac{1}{i} \binom{i}{m+1}$$
10. a) Montrer que pour tout $a, b \in \mathbb{N}^*$, avec $a \leq b$: $a\binom{b}{a} = b\binom{b-1}{a-1}$. Cette formule est-elle encore vraie pour $a > b$?

    b) En déduire que
$$\sum_{k=1}^{n} \binom{k}{m} H_k = \binom{n+1}{m+1} H_n - \frac{1}{m+1} \binom{n}{m+1}$$
    c) Démontrer que
$$\sum_{k=1}^{n} \binom{k}{m} H_k = \binom{n+1}{m+1} \left(H_{n+1} - \frac{1}{m+1}\right)$$
11. À l'aide du résultat de la question précédente, retrouver la formule obtenue à la question 7.