Devoir surveillé 1 : Suites numériques

Le sujet se compose de cinq exercices
(estimation : 30 mn pour les exercices 1 et 2, les exercices 3 et 4 et enfin pour l'ex. 5).
Il sera tenu compte de la présentation et en particulier de l'encadrement des résultats.
L'usage de la calculatrice n'est pas autorisé au cours de l'épreuve.

Exercice 1 : Manipulations élémentaires avec Python

① On considère la liste `L1 = [1, 3, 7, 'cinq', 7, 'neuf', 7, 11]`.
Comment, avec une commande Python et le seul recours à `L1`, pouvez-vous :
a) Obtenir le nombre d'éléments dans `L1` :
b) afficher l'entier `11`, dernier élément de la liste :
c) afficher le `f` de la chaîne de caractère `'neuf'`
d) compléter la liste pour qu'elle devienne : `[1, 3, 7, 'cinq', 7, 'neuf', 7, 11, 13]`.
e) Compter le nombre de `7` dans la liste :

② Créer la liste `L2 = [1,4,9,16,25, ...,100]`

③ On suppose avoir importé la bibliothèque `numpy` grâce à la commande `import numpy as np`.
Que retourne :
a) `np.arange(10)` ?
b) `np.arange(1,10)` ?
c) `np.arange(1,10,2)` ?
d) `np.linspace(1,10,10)` ?

Exercice 2 : Calcul des sommes usuelles

1. Donner la valeur des sommes suivantes :
$$S_1 = \sum_{k=1}^{n} \frac{1}{3^k};\ S_2 = \sum_{k=0}^{n-1} \binom{n}{k} \frac{1}{3^k};\ S_3 = \sum_{k=0}^{n} k \binom{n}{k} 3^k \text{ et } S_4 = \sum_{k=0}^{n} k^2 \binom{n}{k} 3^k$$
2. Rappeler la formule de Pascal : $\binom{k}{p} + \binom{k}{p+1} = \cdots$
Application : Montrer que $\sum_{k=p}^{n} \binom{k}{p} = \binom{n+1}{p+1}, \forall 0 \leq p \leq n$
3. Calculer $\sum_{k=2}^{n-1} \frac{1}{k(k+1)}$

Exercice 3 : Suites adjacentes

On pose, pour $n \in \mathbb{N}^*$, $u_n = \sum_{k=1}^{n} \frac{1}{\sqrt{k}} - 2\sqrt{n}$ et $v_n = \sum_{k=1}^{n} \frac{1}{\sqrt{k}} - 2\sqrt{n+1}$

① Montrer que les suites (u_n) et (v_n) sont adjacents.

② En déduire que $S_n = \sum_{k=1}^{n} \frac{1}{\sqrt{k}} \underset{n\to+\infty}{\sim} 2\sqrt{n}$

Exercice 4 : Connaissances de suites récurrentes usuelles

Soient (u_n) et (v_n) les suites définies par $u_0 = 1$ et $v_0 = 2$ et pour tout $n \in \mathbb{N}$:

$$u_{n+1} = 3u_n + 2v_n \text{ et } v_{n+1} = 2u_n + 3v_n$$

① Écrire une fonction Python `suitesRecurrentes(n)` d'argument n un entier naturel non nul qui renvoie deux listes formées respectivement des $(n+1)$ premiers termes des suites (u_n) et (v_n).
✎ à savoir : `Lu = [u0, u1, ..., un]` et `Lv = [v0, v1, ..., vn]`.

② Montrer que la suite $(u_n - v_n)_{(n\geq 0}$ est constante.

③ Prouver que (u_n) est une suite arithmético-géométrique.

④ Exprimer les termes généraux des suites (u_n) et (v_n).

⑤ Écrire une fonction `suiteExplicite(n)` de même argument que la fonction écrite à la question 1. et qui renvoie les mêmes listes mais en utilisant cette fois les formes explicites de u_n et v_n.

Exercice 5 : suite numérique et série

On considère la suite définie par : $\begin{cases} u_0 & = 1 \\ u_{n+1} & = 2u_n + n - 1 \end{cases}$, $\forall n \in \mathbb{N}$.

① Étude de (u_n) :

a) Calculer u_1, u_2 et u_3.

b) Montrer par récurrence que, pour tout entier naturel n, on a : $u_n \geq 1$.

c) Sans récurrence, en déduire que : $\forall n \in \mathbb{N}$, $u_n \geq n$, puis la limite de la suite.

d) Montrer que (u_n) est croissante.

e) Écrire une fonction Python qui permet de vérifier graphiquement le résultat précédent en affichant les termes u_0 à u_n.

② Expression de u_n : On pose $v_n = u_n + n$ pour tout n entier naturel.

a) Exprimer v_{n+1} en fonction de u_n et n.

b) En déduire que (v_n) est une suite géométrique et donner l'expression de (v_n) en fonction de n.

c) Déterminer l'expression de u_n en fonction de n.

d) Donner un équivalent de u_n en $+\infty$.

③ Calcul d'une somme :

a) En reprenant la définition de (u_n), montrer que : $\forall n \in \mathbb{N}$, $\dfrac{u_{n+1} - 1}{2^n} = \dfrac{u_n - 1}{2^{n-1}} + \dfrac{n}{2^n}$.

b) Par récurrence, en déduire que : $\forall n \in \mathbb{N}^*$, $\dfrac{u_n - 1}{2^{n-1}} = \displaystyle\sum_{k=0}^{n-1} \frac{k}{2^k}$.

c) En déduire l'expression de $\displaystyle\sum_{k=0}^{n} \frac{k}{2^k}$ en fonction de n puis sa limite lorsque n tend vers $+\infty$.

d) Donner le moyen de valider graphiquement votre réponse avec Python.

- FIN -