
Mathématiques - mercredi 12 novembre 2025
Devoir n°3 Durée : 2 h 30 min

- **Aucun document autorisé. Calculatrice interdite.**
- **Les qualités de rédaction (clarté des raisonnements, lisibilité, orthographe...) seront sensiblement prises en considération dans l'évaluation des copies.**
- **Le sujet comporte 3 pages et 4 exercices.**
- $\boxed{\textbf{L'exercice 4 est à rendre sur une copie séparée.}}$

Exercice 1.
Les 3 questions sont indépendantes.

1. Énoncer et démontrer l'inégalité triangulaire dans $\mathbb{C}$. *On pourra utiliser sans démonstration que :* $\forall z \in \mathbb{C},\ Re(z) \leq |z|$.
2. Linéariser $\sin^2(\theta)\cos(\theta)$ pour $\theta \in \mathbb{R}$.
3. Résoudre sur $\mathbb{R}$ les équations suivantes :
   (a) $\dfrac{2e^x - e^{-x}}{e^x + 3e^{-x}} = \dfrac{1}{4}$
   (b) $\ln(2x+1) + \ln(1-x) = 0$

Exercice 2.
Les 2 questions sont indépendantes.

1. (a) Déterminer deux nombres réels a, b tels que :
$$\forall k \geq 1,\ \frac{1}{k(k+1)} = \frac{a}{k} + \frac{b}{k+1}.$$
   (b) En déduire la valeur de $\displaystyle\sum_{k=1}^{n} \frac{1}{k(k+1)}$ pour $n \geq 1$.
2. On considère les nombres complexes
$$z_1 = \frac{\sqrt{6} - i\sqrt{2}}{2}, \quad z_2 = 1 - i \quad \text{et} \quad z_3 = \frac{z_1}{z_2}.$$
   (a) Déterminer la forme exponentielle de z_3.
   (b) Déterminer la forme algébrique de z_3.
   (c) Déduire de ce qui précède les valeurs de $\cos\left(\dfrac{\pi}{12}\right)$ et $\sin\left(\dfrac{\pi}{12}\right)$.

Exercice 3.
Dans cet exercice on s'intéresse aux valeurs des sommes suivantes pour $n \in \mathbb{N}$:

$$S_0(n) = \sum_{k=0}^{n} 1 \quad ; \quad S_1(n) = \sum_{k=0}^{n} k \quad ; \quad S_2(n) = \sum_{k=0}^{n} k^2 \quad \text{et} \quad S_3(n) = \sum_{k=0}^{n} k^3.$$

1. Donner sans démonstration les valeurs de $S_0(n)$, $S_1(n)$ et $S_2(n)$ pour $n \in \mathbb{N}$.

Dans la suite de l'exercice, on fixe $n \in \mathbb{N}$ et on démontre selon trois méthodes indépendantes le résultat $(*)$ suivant :

$$(*) \ : \ \forall n \in \mathbb{N}, \ S_3(n) = \frac{n^2(n+1)^2}{4}.$$

2. *Via un retournement.*
   (a) Démontrer que $S_3(n) = \sum_{k=0}^{n}(n-k)^3$.
   (b) Pour $k \in [\![0, n]\!]$ développer la quantité $(n-k)^3$.
   (c) En déduire que $2S_3(n) = n^3 S_0(n) - 3n^2 S_1(n) + 3n S_2(n)$ puis utiliser la question 1. pour obtenir le résultat $(*)$.

3. *Via un téléscopage.*
   On introduit la somme $T(n) = \sum_{k=0}^{n} \left((k+1)^4 - k^4\right)$.
   (a) Pour $k \in [\![0, n]\!]$ développer et simplifier l'expression $(k+1)^4 - k^4$.
   (b) En déduire l'expression de $T(n)$ en fonction des sommes $S_i(n)$ pour $i \in [\![0, 3]\!]$.
   (c) Simplifier la somme $T(n)$ puis retrouver le résultat $(*)$.

4. *Via une somme de nombres impairs.*

   En 1854, le physicien Charles Wheatstone donne une preuve du résultat $(*)$ s'appuyant sur le tableau triangulaire ci-contre. Dans ce tableau, seuls des nombres impairs apparaissent, et on remarque que la somme des nombres placés sur la k-ème ligne vaut k^3.

   La somme $S_3(n)$ sera donc égale à la somme de tous les nombres présents dans ce tableau jusqu'à la n-ème ligne, ce qui permet de la calculer.

1	$\ldots$	$1 = 1^3$
$3 + 5$	$\ldots$	$8 = 2^3$
$7 + 9 + 11$	$\ldots$	$27 = 3^3$
$13 + 15 + 17 + 19$	$\ldots$	$64 = 4^3$
$21 + 23 + 25 + 27 + 29$	$\ldots$	$125 = 5^3$
$31 + 33 + 35 + 37 + 39 + 41$	$\ldots$	$216 = 6^3$

   (a) Démontrer que pour $p, q \in \mathbb{N}$ tels que $p \leq q$, on a $\sum_{j=p}^{q}(2j-1) = q^2 - (p-1)^2$.
   (b) Expliquer pourquoi la somme des nombres placés sur la k-ème ligne du tableau présenté ci-dessus est donnée par $L_k = \sum_{j=S_1(k-1)+1}^{S_1(k)} (2j-1)$ où S_1 désigne toujours la somme introduite au début de l'exercice.
   (c) En déduire que pour tout $k \in [\![1, n]\!]$ on a $L_k = k^3$.
   (d) En utilisant que $S_3(n) = \sum_{k=1}^{n} L_k$, retrouver le résultat $(*)$.

Exercice 4. Exercice à rédiger sur une copie séparée

Dans cet exercice, on souhaite manipuler des nombres complexes en Python. Pour cela, on représente le nombre complexe $z = x + iy$ (avec $x, y \in \mathbb{R}$) par le couple (x, y).

1. Écrire une fonction `module` prenant en argument un nombre complexe $z = x + iy$ représenté par le couple (x, y) et renvoyant $|z|$.

On s'intéresse à la suite de nombres complexes (z_n) définie par la donnée de $z_0 \in \mathbb{C}$ et :

$$\forall n \in \mathbb{N},\ z_{n+1} = z_n^2 + \frac{-1+i}{2}.$$

Pour tout $n \in \mathbb{N}$ on note $z_n = x_n + iy_n$ avec $x_n, y_n \in \mathbb{R}$.

2. Démontrer que pour tout $n \in \mathbb{N}$, on a $x_{n+1} = x_n^2 - y_n^2 - \frac{1}{2}$ et $y_{n+1} = 2x_n y_n + \frac{1}{2}$.

3. Écrire une fonction `suite` prenant en argument $n \in \mathbb{N}$ et $z_0 \in \mathbb{C}$ (représenté par le couple (x_0, y_0)) et renvoyant le nombre complexe z_n (représenté par le couple (x_n, y_n)).

On s'intéresse maintenant à un sous-ensemble de $\mathbb{C}$ associé à la suite (z_n) et appelé ensemble de Julia ; c'est l'ensemble suivant :

$$J = \{z_0 \in \mathbb{C} \ \text{ tel que } \ \forall n \in \mathbb{N},\ |z_n| \leq 2\}.$$

Informatiquement, on ne peut bien sûr pas tester si la condition $|z_n| \leq 2$ est satisfaite pour une infinité de valeurs de n. On choisit que pour décider si un nombre complexe z_0 appartient ou non à l'ensemble J, on se contentera de tester si : $\forall n \in [\![0, 100]\!],\ |z_n| \leq 2$.

4. Compléter la fonction `dansJ` ci-contre prenant en argument $z_0 \in \mathbb{C}$ et renvoyant `True` si z_0 appartient à l'ensemble de Julia et `False` sinon.

   On recopiera entièrement la fonction `dansJ` sur la copie.

```
def dansJ(x0,y0):
    n = ...
    z = ...
    while ........ and n < ...
        n = ...
        z = suite(n,x0,y0)
    if n == 101 :
        return True
    else :
        return False
```

5. La fin du code proposé est particulièrement maladroite. Par quoi devrait-on remplacer les lignes ci-contre ?

```
if n == 101 :
    return True
else :
    return False
```

6. Expliquer pourquoi il n'est en fait pas judicieux d'utiliser la fonction `suite` au sein de la fonction `dansJ`. Écrire ensuite une fonction `dansJ_bis` effectuant les mêmes calculs que `dansJ` mais sans ce défaut.