DS 7 Mathématiques
Correction

Problème 1

Partie 1 : Une suite de polynômes

1. On calcule $U_2(x) = 2xU_1(x) - U_0(x) = 2x \times 2x - 1 = 4x^2 - 1$.
   On peut ensuite calculer $U_3(x) = 2xU_2(x) - U_1(x) = 2x \times (4x^2 - 1) - 2x = 8x^3 - 2x - 2x = 8x^3 - 4x$.
   Enfin on a alors $U_4(x) = 2xU_3(x) - U_2(x) = 2x \times (8x^3 - 4x) - (4x^2 - 1) = 16x^4 - 8x^2 - 4x^2 + 1 = 16x^4 - 12x^2 + 1$.
2. (a) Démontrons par récurrence double sur $n \in \mathbb{N}$ la propriété $\mathcal{P}(n)$: « U_n est de degré n » (cela est bien conforme à ce qu'on observe sur les calculs de la question précédente).
   Pour $n = 0$, U_0 est un polynôme constant non-nul, donc est bien de degré 0.
   Pour $n = 1$, U_1 est le polynôme $2x$ qui est de degré 1.
   Soit maintenant $n \in \mathbb{N}$, supposons $\mathcal{P}(n)$ et $\mathcal{P}(n+1)$. Alors dans la relation $U_{n+2} = 2xU_{n+1} - U_n$, le terme U_{n+1} est de degré $n+1$ donc $2xU_{n+1}$ est de degré $n+2$, et d'autre part U_n est de degré n. Nous sommes dans le cas d'une somme de deux polynômes de degrés différents, les coefficients dominants ne peuvent pas se compenser. Le degré est donc le maximum des deux, et c'est bien $n+2$. Ceci démontre la propriété $\mathcal{P}$ au rang $n+2$.

   (b) Reprenons la preuve précédente, notons a_n le coefficient dominant de U_n, on sait que $a_0 = 1$. Si $U_n(x) = a_n x^n + \cdots$ alors $U_{n+1}(x) = a_{n+1}x^{n+1} + \cdots$ et $2xU_{n+1}(x) = 2a_{n+1}x^{n+2} + \cdots$; quand on somme avec $-U_n(x) = -a_n x^n$ le terme dominant reste, comme on l'a dit, celui provenant de $2xU_n(x)$, donc c'est $a_{n+2}x^{n+2} = 2a_{n+1}x^{n+2}$. Cela montre que a_n vérifie la relation de récurrence $a_{n+2} = 2a_{n+1}$ pour tout $n \geqslant 0$ ou, de façon équivalente, $a_{n+1} = 2a_n$ pour tout $n \geqslant 1$.
   Comme $a_0 = 1$ et $a_1 = 2$ c'est bien une suite géométrique de raison 2, c'est $a_n = 2^n$. Cela correspond bien à ce qui a été calculé sur les premiers exemples.
3. Démontrons par récurrence double sur $n \in \mathbb{N}$ la propriété $\mathcal{P}(n)$: « $\forall x \in \mathbb{R}$, $U_n(-x) = (-1)^n U_n(x)$.
   Pour $n = 0$ il s'agit de vérifier $U_0(-x) = U_0(x)$, ce qui est bien le cas.
   Pour $n = 1$ il s'agit de $U_1(-x) = -U_1(x)$, ce qui est clair aussi.
   Soit $n \in \mathbb{N}$, supposons $\mathcal{P}(n)$ et $\mathcal{P}(n+1)$. Alors pour tout $x \in \mathbb{R}$, on a
$$\begin{aligned}
U_{n+2}(-x) &= 2 \times (-x)U_{n+1}(-x) - 2U_n(-x) && (1)\\
&= 2 \times (-x) \times (-1)^{n+1}U_{n+1}(x) - (-1)^n U_n(x) && \text{(par } \mathcal{P}(n) \text{ et } \mathcal{P}(n+1)) \quad (2)\\
&= 2 \times x \times (-1)^{n+2}U_{n+1}(x) - (-1)^{n+2}U_n(x) && \text{(en effet } (-1)^{n+2} = (-1)^n) \quad (3)\\
&= (-1)^{n+2} \times \Big(2xU_{n+1}(x) - U_n(x)\Big) && (4)\\
&= (-1)^{n+2}U_{n+2}(x) && (5)
\end{aligned}$$
   Ceci démontre la propriété $\mathcal{P}$ au rang $n+2$.
4. (a) Démontrons, encore une fois par récurrence double sur $n \in \mathbb{N}$, la propriété $\mathcal{P}(n)$: « $\forall \theta \in \mathbb{R}$, $\sin(\theta) \times U_n(\cos(\theta)) = \sin((n+1)\theta)$ ».
   Pour $n = 0$ il s'agit simplement de $\sin(\theta) \times 1 = \sin(\theta)$.
   Pour $n = 1$ il s'agit de $\sin(\theta) \times 2\cos(\theta) = \sin(2\theta)$. C'est une formule de trigonométrie classique (duplication pour le sinus), écrite le plus usuellement $\sin(2\theta) = 2\sin(\theta)\cos(\theta)$.
   Soit alors $n \in \mathbb{N}$, supposons $\mathcal{P}(n)$ et $\mathcal{P}(n+1)$, écrivons-les bien l'une au-dessus de l'autre :
$$\begin{aligned}
\sin(\theta) \times U_n(\cos(\theta)) &= \sin((n+1)\theta) && (6)\\
\sin(\theta) \times U_{n+1}(\cos(\theta)) &= \sin((n+2)\theta) && (7)
\end{aligned}$$
   On prend alors la relation de récurrence $U_{n+2}(x) = 2xU_{n+1}(x) - U_n(x)$ et on remplace x par $\cos(\theta)$: $U_{n+2}(\cos(\theta)) = 2\cos(\theta) \times U_{n+1}(\cos(\theta)) - U_n(\cos(\theta))$. Éventuellement on multiplie déjà par $\sin(\theta)$ des deux côtés :
$$\begin{aligned}
\sin(\theta) \times U_{n+2}(\cos(\theta)) &= \sin(\theta) \times 2\cos(\theta) \times U_{n+1}(\cos(\theta)) - \sin(\theta) \times U_n(\cos(\theta)) && (8)\\
&= 2\cos(\theta) \times \Big(\sin(\theta) \times U_{n+1}(\cos(\theta))\Big) - \sin(\theta) \times U_n(\cos(\theta)) && (9)
\end{aligned}$$

(On peut aussi raisonner en divisant tout par $\sin(\theta)$). On remplace alors avec les relations de récurrence :

$$\sin(\theta) \times U_{n+2}(\cos(\theta)) = 2\cos(\theta)\sin((n+2)\theta) - \sin((n+1)\theta) \tag{10}$$

Il s'agit maintenant de développer et de comparer avec $\sin((n+3)\theta)$.
Or, écrivant d'abord $(n+3)\theta = (n+2)\theta + \theta$, on trouve (formule d'addition classique du sinus)

$$\sin((n+3)\theta) = \sin((n+2)\theta)\cos(\theta) + \sin(\theta)\cos((n+2)\theta) \tag{11}$$

Poursuivant en écrivant $(n+2)\theta = (n+1)\theta + \theta$, et avec la formule d'addition classique pour le cosinus :

$$\sin((n+2)\theta) = \sin((n+1)\theta)\cos(\theta) + \cos((n+1)\theta)\sin(\theta) \tag{12}$$
$$\cos((n+2)\theta) = \cos((n+1)\theta)\cos(\theta) - \sin((n+1)\theta)\sin(\theta) \tag{13}$$

En remplaçant pour exprimer tout en fonction de l'angle $(n+1)\theta$, on trouve alors d'une part

$$\begin{aligned}
\sin((n+3)\theta) &= \cos(\theta)\Big(\sin((n+1)\theta)\cos(\theta) + \cos((n+1)\theta)\sin(\theta)\Big) && (14)\\
&\quad + \sin(\theta)\Big(\cos((n+1)\theta)\cos(\theta) - \sin((n+1)\theta)\sin(\theta)\Big) && (15)\\
&= \Big(\cos^2(\theta) - \sin^2(\theta)\Big)\sin((n+1)\theta) + \Big(2\sin\theta\cos(\theta)\Big)\cos((n+1)\theta) && (16)
\end{aligned}$$

et d'autre part

$$\begin{aligned}
\sin(\theta)U_{n+2}(\cos(\theta)) &= 2\cos(\theta)\Big(\sin((n+1)\theta)\cos(\theta) + \cos((n+1)\theta)\sin(\theta)\Big) - \sin((n+1)\theta) && (17)\\
&= \Big(2\cos^2(\theta) - 1\Big)\sin((n+1)\theta) + \Big(2\sin(\theta)\cos(\theta)\Big)\cos((n+1)\theta) && (18)
\end{aligned}$$

Ces deux quantités sont bien égales car $2\cos^2(\theta) - 1 = \cos^2(\theta) - \sin^2(\theta)$, qui est aussi $\cos(2\theta)$.
Ceci démontre notre propriété au rang $n+2$.

(b) Connaissant la formule précédente, alors il suffit de remplacer : on fixe $n \in \mathbb{N}^*$, puis $1 \leqslant k \leqslant n$ et alors on remplace θ par $\frac{k\pi}{n+1}$ et on obtient

$$\sin\left(\tfrac{k\pi}{n+1}\right) \times U_n(y_k) = \sin\left((n+1) \times \tfrac{k\pi}{n+1}\right) \tag{19}$$

À droite, le terme se simplifie en $\sin(k\pi)$, qui est toujours nul. Ainsi $\sin\left(\frac{k\pi}{n+1}\right) \times U_n(y_k) = 0$.
Pour conclure que $U_n(y_k) = 0$, il faut justifier que le terme $\sin\left(\frac{k\pi}{n+1}\right)$ qui est devant est non-nul. Mais par définition $1 \leqslant k \leqslant n$ donc

$$\frac{\pi}{n+1} \leqslant \frac{k\pi}{n+1} \leqslant \frac{n\pi}{n+1} \tag{20}$$

Ces quantités sont donc dans l'intervalle $]0, \pi[$ (à droite $\frac{n}{n+1} < 1$), où la fonction sinus ne s'annule pas. On en déduit donc $U_n(y_k) = 0$.
Il faut aussi vérifier que ces quantités sont distinctes. Mais là encore, les angles ci-dessus sont dans $]0, \pi[$, intervalle sur lequel la fonction cosinus est strictement décroissante et donc prend une unique fois chaque valeur (injective). On conclut donc qu'on a bien trouvé n racines distinctes de U_n.

(c) Le polynôme U_n est de degré n et on en a trouvé n racines distinctes. Ce sont donc toutes les racines de U_n, c'est un polynôme scindé. On en déduit qu'il existe une constante $a \in \mathbb{R}$ non nulle (un polynôme de degré 0) telle que $U_n(x) = a\prod_{k=1}^{n}(x - y_k)$. De plus a est nécessairement le coefficient dominant de U_n (on peut le voir en commençant à développer ce produit), donc c'est 2^n. En conclusion

$$\forall x \in \mathbb{R}, \quad U_n(x) = 2^n \prod_{k=1}^{n}\left(x - \cos\left(\tfrac{k\pi}{n+1}\right)\right) \tag{21}$$

Partie 2 : Modélisation en Python

5. Dans ce cas, le degré du polynôme est un de moins que la longueur de la liste, ce qui s'écrira en Python `len(P) - 1`. Remarquons qu'il est alors assez naturel de vouloir dire que le polynôme nul est de degré -1.
6. Au départ la fonction est correcte : si `n = len(P) - 1` alors c'est bien le dernier indice de la liste `P` donc `P[n]` ne doit pas provoquer d'erreur de type indice de la liste hors des bords (c'est le sens de `list index out of range`). Sauf si `P` est une liste vide : la boucle `while` testant toujours sa condition en début de boucle, elle va effectivement provoquer une erreur.
On pourrait penser la corriger en excluant dès le début la liste vide. Mais ce n'est pas suffisant. En effet si on appelle la fonction avec une liste de zéros telle que `[0, 0, 0]` alors un problème se produit aussi : au dernier passage dans la boucle $n = 0$, le passage dans la boucle est exécuté car `P[0] == 0` est vrai, ce qui diminue n de 1. Puis on revient au test de la boucle c'est-à-dire à tester `P[-1] == 0`. Erreur possible.
Ce qu'il faut, c'est réellement empêcher d'écrire `P[n]` si n devient strictement négatif. Une possibilité est de tester si n est positif dans la condition du `while` et **avant** d'écrire `P[n]` :

```python
def degré(P):
    n = len(P) - 1
    while n >= 0 and P[n] == 0:
        n = n - 1
    return n
```

Pour une liste de zéros, à la fin n devient -1 puis la condition de la boucle ne bloque plus mais est fausse, ainsi c'est bien -1 qui est renvoyé.
Une autre possibilité est d'utiliser `return` dans la boucle avant l'instruction `n = n - 1` pour que n ne devienne jamais négatif ; il faut alors traiter dès le début le cas de la liste vide :

```python
def degré(P):
    if P == []:
        return -1
    n = len(P) - 1
    while P[n] == 0:
        if n > 0:
            n = n - 1
        else:
            return -1
    return n
```

7. La fonction crée une liste de zéros de la même taille que `P` puis met dedans les nombres `-P[i]`. Autrement dit elle renvoie un polynôme Q dont les coefficients sont les opposés de ceux de P, c'est-à-dire précisément qu'elle renvoie le polynôme $-P$.
8. On peut s'inspirer de la fonction donnée en modèle, il faudra créer une liste `R` (résultat) et mettre dans `R[i]` la somme `P[i] + Q[i]`. Si les deux listes ont la même longueur, ce n'est pas spécialement difficile :

```python
def somme_essai(P, Q):
    n = len(P)
    # on suppose que Q est aussi de longueur n
    R = [0] * n
    for i in range(n):
        R[i] = P[i] + Q[i]
    return R
```

On peut aussi créer les listes de zéros avec `[0 for _ in range(n)]`, comme nous avons vu pour les matrices, ici cela ne change rien.
Si les deux listes ne sont pas de la même longueur, mais que `P` est plus grande que `Q`, il faut faire attention : on ne peut sommer les coefficients que sur la longueur de la plus petite des listes, ensuite il faut copier les coefficients de l'autre.

```
def somme(P, Q):
    n = len(P)
    m = len(Q)
    # on suppose n >= m
    R = [0] * n
    for i in range(m):
        R[i] = P[i] + Q[i]
    for i in range(m, n):
        R[i] = P[i]
    return R
```

Une autre méthode à laquelle on peut penser est d'agrandir la liste `Q` par des zéros pour qu'elle atteigne la même taille que `P`. Cela ressemble aux preuves mathématiques dans lesquels nous avons parfois eu besoin de compléter les coefficients pour ramener par exemple deux sommes $\sum_{k=0}^{n} a_k x^k$ et $\sum_{k=0}^{m} b_k x^k$ sous un même ensemble d'indices pour k. On utilise alors au début de la fonction précédente

```
    for i in range(n-m):
        Q.append(0)
```

et on est ramené à deux listes de même longueur. Cette méthode n'est pas excellente car `Q.append` va modifier la liste `Q` passée en argument...

9. Il s'agit d'écrire que le produit entre le coefficient de x^i dans P et le coefficient x^j dans Q *contribue* au coefficient de x^{i+j} dans $P \times Q$. On a donc très certainement besoin d'une double boucle qui calcule `P[i] * Q[j]` et le met en degré $i+j$ dans un nouveau polynôme `R`. Mais attention, ce n'est pas directement `R[i+j] = P[i] * Q[j]` car les termes de degré $i+j$ peuvent apparaitre pour différentes valeurs du couple (i,j), donc ce sera plutôt `R[i+j] = R[i+j] + P[i] * Q[j]` (c'est ainsi que ce produit « contribue à » `R[i+j]`). Enfin, attention à la longueur des listes : si n est la longueur de `P`, correspondant à un polynôme éventuellement de degré $n-1$, et si m est la longueur de `Q`, correspondant à un polynôme éventuellement de degré $m-1$, alors le résultat peut être de degré $(n-1)+(m-1) = n+m-2$ et donc la liste pour le contenir sera de longueur $n+m-1$ (une autre convention pratique est de donner non pas par n la longueur des listes, mais leur longueur moins un, aligné avec les degrés). Bref, en résumé :

```
def produit(P, Q):
    n = len(P)
    m = len(Q)
    R = [0] * (n+m-1)
    for i in range(n):
        for j in range(m):
            R[i+j] = R[i+j] + P[i] * Q[j]
    return R
```

10. On s'inspire de la méthode pour le calcul des suites d'ordre 2, ayant sous les yeux les opérations nécessaires : somme, opposé (donc différence), produit, et le polynôme $2x$ est la liste `[0, 2]`.

```
def suite(n):
    # U représente U_(k)
    U = [1]
    # V représente U_(k+1)
    V = [0, 2]
    for k in range(n):
        # polynôme suivant U_(k+2)
        W = somme(produit([0, 2], V), fonction_mysterieuse_mais_bien_utile(U))
        U = V
        V = W
    return V
```

On peut aussi vouloir écrire une fonction récursive mais, attention, celle-ci sera fort peu efficace en pratique (phénomène d'explosion de la pile d'appels) :

```python
def suite(n):
    if n == 0:
        return [1]
    elif n == 1:
        return [0, 2]
    else:
        return somme(produit([0, 2], suite(n-1)),
            fonction_mysterieuse_mais_bien_utile(suite(n-2)))
```

Partie 3 : Une autre expression

11. La formule s'appelle formule de De Moivre, mais on la retrouve directement avec l'exponentielle complexe :

$$\begin{aligned} &\cos((2n+1)\theta) + i\sin((2n+1)\theta) && (22)\\ &= \exp(i(2n+1)\theta) && (23)\\ &= \exp(i\theta)^{2n+1} && (24)\\ &= (\cos(\theta) + i\sin(\theta))^{2n+1} && (25)\end{aligned}$$

On a alors envie d'appliquer le binôme de Newton à cette quantité :

$$\begin{aligned}(\cos(\theta) + i\sin(\theta))^{2n+1} &= \sum_{r=0}^{2n+1} \binom{2n+1}{r} \Big(i\sin(\theta)\Big)^r \cos^{2n+1-r}(\theta) && (26)\\ &= \sum_{r=0}^{2n+1} \binom{2n+1}{r} i^r \times \sin^r(\theta)\cos^{2n+1-r}(\theta) && (27)\end{aligned}$$

12. On rappelle ou on retrouve au brouillon les puissances successives de i, partant de i^0 : 1, i, -1, $-i$. Les termes i^r sont réels si r est pair et sont imaginaires purs sinon ; de plus si $r = 2k+1$ alors $i^r = (-1)^k$. La partie imaginaire de la quantité ci-dessus est d'une part la partie imaginaire de $\cos((2n+1)\theta) + i\sin((2n+1)\theta)$, c'est-à-dire $\sin((2n+1)\theta)$, et d'autre part la somme où on garde seulement les indices impairs. Posant $r = 2k+1$, les indices sont déjà bons, si k varie de 0 à n alors r varie de 1 à $2n+1$ en étant impair. On trouve donc bien

$$\sin((2n+1)\theta) = \sum_{k=0}^{n} (-1)^k \binom{2n+1}{2k+1} \sin^{2k+1}(\theta)\cos^{2n-2k}(\theta) \tag{28}$$

13. Il suffit de diviser tout cette somme par $\sin^{2n+1}(\theta)$, dans le cas où celui-ci est non-nul, ce qui est équivalent à $\sin(\theta) \neq 0$:

$$\begin{aligned}\frac{\sin((2n+1)\theta)}{\sin^{2n+1}(\theta)} &= \frac{1}{\sin^{2n+1}(\theta)} \sum_{k=0}^{n} (-1)^k \binom{2n+1}{2k+1} \sin^{2k+1}(\theta)\cos^{2n-2k}(\theta) && (29)\\ &= \sum_{k=0}^{n} (-1)^k \binom{2n+1}{2k+1} \frac{\sin^{2k+1}(\theta)\cos^{2n-2k}(\theta)}{\sin^{2n+1}(\theta)} && (30)\\ &= \sum_{k=0}^{n} (-1)^k \binom{2n+1}{2k+1} \sin^{(2k+1)-(2n+1)}(\theta)\cos^{2n-2k}(\theta) && (31)\\ &= \sum_{k=0}^{n} (-1)^k \binom{2n+1}{2k+1} \sin^{2k-2n}(\theta)\cos^{2n-2k}(\theta) && (32)\\ &= \sum_{k=0}^{n} (-1)^k \binom{2n+1}{2k+1} \frac{\cos^{2n-2k}(\theta)}{\sin^{2n-2k}(\theta)} && (33)\\ &= \sum_{k=0}^{n} (-1)^k \binom{2n+1}{2k+1} \left(\frac{\cos(\theta)}{\sin(\theta)}\right)^{2n-2k} && (34)\\ &= \sum_{k=0}^{n} (-1)^k \binom{2n+1}{2k+1} \left(\left(\frac{\cos(\theta)}{\sin(\theta)}\right)^2\right)^{n-k} && (35)\end{aligned}$$

En fait, après avoir divisé, il s'agit seulement de jeux sur les puissances.

14. Sachant qu'on a démontré dans la première partie $U_n(\cos(\theta)) = \frac{\sin((n+1)\theta)}{\sin(\theta)}$ (toujours dans le cas $\sin(\theta) \neq 0$), on a par la même méthode que précédemment

$$\begin{aligned}
\frac{\sin((2n+1)\theta)}{\sin(\theta)} &= \frac{1}{\sin(\theta)} \sum_{k=0}^{n} (-1)^k \binom{2n+1}{2k+1} \sin^{2k+1}(\theta) \cos^{2n-2k}(\theta) \quad (36)\\
&= \sum_{k=0}^{n} (-1)^k \binom{2n+1}{2k+1} \frac{\sin^{2k+1}(\theta)}{\sin(\theta)} \cos^{2n-2k}(\theta) \quad (37)\\
&= \sum_{k=0}^{n} (-1)^k \binom{2n+1}{2k+1} \sin^{2k}(\theta) \cos^{2n-2k}(\theta) \quad (38)
\end{aligned}$$

Sous cette somme, le sinus apparait seulement avec des exposants pairs. On utilise alors la formule $\sin^2(\theta) + \cos^2(\theta) = 1$ soit $\sin^2(\theta) = 1 - \cos^2(\theta)$, et donc

$$\begin{aligned}
\frac{\sin((2n+1)\theta)}{\sin(\theta)} &= \sum_{k=0}^{n} (-1)^k \binom{2n+1}{2k+1} \left(\sin^2(\theta)\right)^k \cos^{2n-2k}(\theta) \quad (39)\\
&= \sum_{k=0}^{n} (-1)^k \binom{2n+1}{2k+1} \left(1 - \cos^2(\theta)\right)^k \cos^{2n-2k}(\theta) \quad (40)
\end{aligned}$$

mais ceci est aussi égal à $U_{2n}(\cos(\theta))$ par la première partie ! Posant $x = \cos(\theta)$ ceci signifie bien $U_n(x) = \sum_{k=0}^{n} (-1)^k \binom{2n+1}{2k+1} (1 - x^2)^k x^{2n-2k}$. Il reste à justifier que cette égalité est vérifiée pour tout $x \in \mathbb{R}$, ce qui n'est pas si évident car avec cette démonstration elle est vraie seulement pour les x de la forme $\cos(\theta)$ avec $\theta \in \mathbb{R}$ et $\sin(\theta) \neq 0$. Mais il suffit de voir qu'il y a bien une infinité de tels x (tous les $x \in]0, \pi[$) : deux polynômes sont égaux si et seulement si ils sont égaux pour une infinité de valeurs réelles.

Partie 4 : Analyse

Remarque : on a donc en résume $P_n(\text{cotan}^2(\theta)) = \frac{\sin((2n+1)\theta)}{\sin(\theta)}$, avec $P_n(x) = \sum_{k=0}^{n} (-1)^k \binom{2n+1}{2k+1} x^{n-k}$ et $\text{cotan}(\theta) = \frac{\cos(\theta)}{\sin(\theta)} = \frac{1}{\tan(\theta)}$.

15. Dans cette somme, le plus grand exposant de x qui apparait est celui pour $k = 0$ et c'est n, donc le polynôme est de degré *au plus* n. Pour $k = 0$ le terme sous la somme est $(+1) \times \binom{2n+1}{1} = 2n + 1$. Ceci est non-nul, donc P_n est bien de degré n et son coefficient dominant est $2n + 1$.
16. $\text{cotan}(\theta)$ est défini si et seulement si $\sin(\theta) \neq 0$. Or $\sin(\theta) = 0$ si et seulement si $\exists k \in \mathbb{Z}$, $\theta = k\pi$. En résumé $\mathcal{D}_{\text{cotan}} = \mathbb{R} \setminus \{\, k\pi \mid k \in \mathbb{Z} \,\}$.
17. On calcule la dérivée de cotan comme un quotient, sur son domaine de définition, exactement comme pour la fonction tangente. On trouve naturellement

$$\text{cotan}' = \frac{\cos' \sin - \cos \sin'}{\sin^2} = \frac{-\sin^2 - \cos^2}{\sin^2} = -\frac{\sin^2 + \cos^2}{\sin^2} \tag{41}$$

Alors : ou bien on utilise $\sin^2 + \cos^2 = 1$, et dans ce cas $\text{cotan}' = -\frac{1}{\sin^2}$. Ou bien on sépare la fraction et alors $\text{cotan}' = -\left(1 + \frac{\cos^2}{\sin^2}\right) = -\left(1 + (\frac{\cos}{\sin})^2\right) = -(1 + \text{cotan}^2)$.

On en déduit donc que $\text{cotan}' < 0$ sur $]0, \pi[\subset \mathcal{D}_{\text{cotan}}$, de plus on en déduit facilement les limites en 0 et π en connaissant celles de sin :

$$\begin{aligned}
&\lim_{x \to 0^+} \sin(x) = 0^+ \qquad \lim_{x \to 0^+} \cos(x) = 1 \qquad \lim_{x \to 0^+} \text{cotan}(x) = +\infty \quad (42)\\
&\lim_{x \to \pi^-} \sin(x) = 0^+ \qquad \lim_{x \to \pi^-} \cos(x) = -1 \qquad \lim_{x \to \pi^-} \text{cotan}(x) = -\infty \quad (43)\\
&\quad (44)
\end{aligned}$$

(ce qui est bien cohérent avec une fonction décroissante) et donc le tableau de variations :

x	0		π
$f(x)$	$+\infty$	$\searrow$	$-\infty$

(45)

Remarque : au milieu c'est $\text{cotan}(\pi/2) = 0$.

18. On note pour $1 \leqslant k \leqslant n$, $x_k = \text{cotan}^2\left(\frac{k\pi}{2n+1}\right)$.

(a) On a par construction $P_n(\text{cotan}^2(\theta)) = \frac{\sin((2n+1)\theta)}{\sin\theta}$, donc pour ces valeurs x_k, correspondant à $\theta = \frac{k\pi}{2n+1}$, on trouve directement

$$P_n(x_k) = \frac{\sin\left((2n+1) \times \frac{k\pi}{2n+1}\right)}{\sin\left(\frac{k\pi}{2n+1}\right)} \tag{46}$$

$$= \frac{\sin(k\pi)}{\sin\left(\frac{k\pi}{2n+1}\right)} \tag{47}$$

$$= 0 \tag{48}$$

car $\sin(k\pi) = 0$. Il reste à vérifier que tout ceci est bien défini et que ces racines sont distinctes. Mais c'est parce que $1 \leqslant k \leqslant n$ donc $0 < \frac{1}{2n+1} \leqslant \frac{k}{2n+1} \leqslant \frac{n}{2n+1} \leqslant \frac{1}{2} < 1$ et donc $0 < \frac{k\pi}{2n+1} \leqslant \frac{\pi}{2} < \pi$: la fonction cotangente est bien définie et en plus est strictement décroissante (en particulier injective), donc ces valeurs sont différentes.

(b) *Erreur dans le sujet : il faut montrer qu'il existe un $\lambda \in \mathbb{R}$ tel que $\forall x \in \mathbb{R}$, $P_n(x) = \lambda \prod_{k=1}^{n}(x - x_k)$.*

P_n étant de degré n, et ayant trouvé n racines distinctes, on en déduit qu'il existe un $\lambda \in \mathbb{R}$ tel que $\forall x \in \mathbb{R}$, $P_n(x) = \lambda \prod_{k=1}^{n}(x - x_k)$ (polynôme scindé). Ce λ est le coefficient dominant, ce qu'on peut voir en commençant à développer le produit, c'est donc $\lambda = 2n + 1$.

(c) *Erreur dans le sujet, le terme de droite est $\frac{n(2n-1)}{3}$.*
D'une part avec la formule de somme, le coefficient de x^{n-1} dans P_n est obtenu pour $k = 1$, donc c'est

$$-\binom{2n+1}{3} = -\frac{(2n+1) \times (2n) \times (2n-1)}{3 \times 2 \times 1} = -\frac{n(2n-1)(2n+1)}{3} \tag{49}$$

D'autre part, en commençant à développer sous forme de produit, on trouve

$$(x - x_1)\dots(x - x_n) = x^n - (x_1 + \cdots + x_n)x^{n-1} + \cdots \tag{50}$$

et donc dans $\lambda(x - x_1)\dots(x - x_n)$ le coefficient de x^{n-1} est $-\lambda(x_1 + \cdots + x_n)$, c'est la relation entre coefficients et somme des racines pour les polynômes de degré n. Comme les coefficients du polynôme sont uniques, on trouve alors

$$-\lambda(x_1 + \cdots + x_n) = -\frac{n(2n-1)(2n+1)}{3} \tag{51}$$

ce qui, avec $\lambda = 2n + 1$, donne

$$x_1 + \cdots + x_n = \frac{n(2n-1)}{3} \tag{52}$$

et c'est bien la formule demandée.

(d) Il faut faire le lien avec la formule de la dérivée de cotan, qu'on peut lire comme $\frac{1}{\sin^2} = 1 + \text{cotan}^2$. On trouve alors

$$\sum_{k=1}^{n} \frac{1}{\sin^2\left(\frac{k\pi}{2n+1}\right)} = \sum_{k=1}^{n} \left(1 + \operatorname{cotan}^2\left(\frac{k\pi}{2n+1}\right)\right) \tag{53}$$

$$= \sum_{k=1}^{n} 1 + \sum_{k=1}^{n} \operatorname{cotan}^2\left(\frac{k\pi}{2n+1}\right) \tag{54}$$

$$= n + \frac{n(2n-1)}{3} \tag{55}$$

$$= \frac{3n + n(2n-1)}{3} \tag{56}$$

$$= \frac{n(3+2n-1)}{3} \tag{57}$$

$$= \frac{n(2n+2)}{3} = \frac{2n(n+1)}{3} \tag{58}$$

(59)

19. Ces inégalités sont vraiment très classiques et se démontrent avec un petit tableau de variations. Pour sin, on pose $f : y \mapsto y - \sin(y)$, donc $f'(y) = 1 - \cos(y)$.

y	0 $\qquad\qquad$ $\frac{\pi}{2}$
$f'(y)$	$+$
$f(y)$	$0 \nearrow \frac{\pi}{2}$

(60)

Pour tan, on pose $g : y \mapsto \tan(y) - y$, $g'(y) = (1 + \tan^2(y)) - 1 = \tan^2(y)$ et donc

y	0 $\qquad\qquad$ $\frac{\pi}{2}$
$g'(y)$	$+$ $\Vert$
$g(y)$	$0 \nearrow +\infty$ $\Vert$

(61)

Si $y \neq 0$ alors de plus $\sin(y) \neq 0$, $\tan(y) \neq 0$ et donc $0 < \sin^2(y) \leqslant y^2 \leqslant \frac{1}{\tan^2(y)}$. Passant à l'inverse, on trouve l'inégalité demandée $0 < \frac{1}{\tan^2(y)} \leqslant \frac{1}{y^2} \leqslant \frac{1}{\sin^2(y)}$, et le terme de gauche est $\operatorname{cotan}^2(y)$.

20. On applique l'inégalité précédente avec les valeurs $\frac{k\pi}{2n+1}$, ce qui est bien possible puisqu'on a bien montré qu'ils étaient tous dans $]0, \pi/2[$:

$$\forall 1 \leqslant k \leqslant n, \quad \operatorname{cotan}^2\left(\frac{k\pi}{2n+1}\right) \leqslant \underbrace{\frac{1}{\left(\frac{k\pi}{2n+1}\right)^2}}_{\frac{(2n+1)^2}{\pi^2 k^2}} \leqslant \frac{1}{\sin^2\left(\frac{k\pi}{2n+1}\right)} \tag{62}$$

Sommons :

$$\sum_{k=1}^{n} \operatorname{cotan}^2\left(\frac{k\pi}{2n+1}\right) \leqslant \sum_{k=1}^{n} \frac{(2n+1)^2}{\pi^2 k^2} \leqslant \sum_{k=1}^{n} \frac{1}{\sin^2\left(\frac{k\pi}{2n+1}\right)} \tag{63}$$

Sortons la constante dans la somme du milieu et on trouve bien

$$\underbrace{\sum_{k=1}^{n} \operatorname{cotan}^2\left(\frac{k\pi}{2n+1}\right)}_{\frac{n(2n-1)}{3}} \leqslant \frac{(2n+1)^2}{\pi^2} \sum_{k=1}^{n} \frac{1}{k^2} \leqslant \underbrace{\sum_{k=1}^{n} \frac{1}{\sin^2\left(\frac{k\pi}{2n+1}\right)}}_{\frac{2n(n+1)}{3}} \tag{64}$$

Les termes à gauche et à droite ont déjà été calculés dans les questions précédentes.

21. On obtient en divisant l'inégalité par n^2 :

$$\frac{n(2n-1)}{3n^2} \leqslant \frac{(2n+1)^2}{n^2\pi^2}\sum_{k=1}^{n}\frac{1}{k^2} \leqslant \frac{2n(n+1)}{3n^2} \tag{65}$$

Ceci est vrai (depuis le début) quelque soit $n \in \mathbb{N}^*$. Faisons alors tendre n vers $+\infty$. Les deux morceaux qui encadrent la somme tendent vers la même limite finie, ce sont des quotients de polynômes donc la limite est facile à calculer, le numérateur $n(2n-1)$ est équivalent à $2n^2$ et $2n(n+1)$ aussi :

$$\lim_{n\to+\infty}\frac{n(2n-1)}{3n^2} = \frac{2}{3} \tag{66}$$

$$\lim_{n\to+\infty}\frac{2n(n+1)}{3n^2} = \frac{2}{3} \tag{67}$$

Par le théorème des gendarmes, on en déduit donc

$$\lim_{n\to+\infty}\frac{(2n+1)^2}{n^2\pi^2}\sum_{k=1}^{n}\frac{1}{k^2} = \frac{2}{3} \tag{68}$$

Mais le terme devant la somme

$$\frac{(2n+1)^2}{n^2\pi^2} \sim \frac{(2n)^2}{n^2\pi^2} \sim \frac{4^n 2}{n^2\pi^2} \longrightarrow \frac{4}{\pi^2} \tag{69}$$

Si on note $S_n = \sum_{k=1}^{n}\frac{1}{k^2}$ cela signifie que

$$S_n = \underbrace{\frac{n^2\pi^2}{(2n+1)^2}}_{\to\frac{\pi^2}{4}} \times \underbrace{\frac{(2n+1)^2}{n^2\pi^2}\sum_{k=1}^{n}\frac{1}{k^2}}_{\to\frac{2}{3}} \tag{70}$$

D'où on déduit

$$\lim_{n\to+\infty}\sum_{k=1}^{n}\frac{1}{k^2} = \frac{\pi^2}{4}\times\frac{2}{3} = \frac{\pi^2}{6} \tag{71}$$

Exercice

1. Pour F : posons $\overrightarrow{u_1} = (1,2,-2,1)$, $\overrightarrow{u_2} = (1,2,0,3)$, $\overrightarrow{u_3} = (0,0,1,1)$ qui sont trois vecteurs de $\mathbb{R}^4$. Alors on reconnait $\mathrm{Vect}(\overrightarrow{u_1},\overrightarrow{u_2},\overrightarrow{u_3})$, qui est bien un sous-espace vectoriel de $\mathbb{R}^4$.
   Pour G : on reconnait trois équations d'hyperplans, qui sont chacun des sous-espaces vectoriels, et c'est l'intersection des trois, qui est encore un sous-espace vectoriel.
   On peut aussi le démontrer à la main en prenant des vecteurs $\overrightarrow{u}$, $\overrightarrow{v}$, une constante $\lambda \in \mathbb{R}$; mais cela prend beaucoup de temps, et vous vous pénalisez tout seul à ne pas utiliser les théorèmes du cours...
2. Par définition la famille $(\overrightarrow{u_1},\overrightarrow{u_2},\overrightarrow{u_3})$ qu'on a exhibée est génératrice de F. Il suffit donc de tester si elle est libre (auquel cas c'est une base) ou si certains vecteurs sont combinaisons linéaires des autres, auquel cas on peut les retirer...

Or pour $(a,b,c)\in\mathbb{R}^3$ alors l'équation $a\overrightarrow{u_1}+b\overrightarrow{u_2}+c\overrightarrow{u_3}=\overrightarrow{0}$ est équivalente à

$$\begin{cases} a+b=0 \\ 2a+2b=0 \\ -2a+c=0 \\ a+3b+c=0 \end{cases} \tag{72}$$

$$\iff \begin{cases} a+b=0 & \\ 0=0 & L_2-2L_1 \\ 2b+c=0 & L_3+2L_1 \\ 2b+c=0 & L_3-L_1 \end{cases} \tag{73}$$

$$\iff \begin{cases} a+b=0 & \\ 2b+c=0 & L_3 \\ 0=0 & \\ 0=0 & \end{cases} \tag{74}$$

On voit que la famille n'est pas libre, c est une variable libre, $\overrightarrow{u_3}$ est combinaison linéaire de $\overrightarrow{u_1}$ et $\overrightarrow{u_2}$ et le retirer ne change pas l'espace engendré par la famille.
En résumé une base est formée de $\overrightarrow{u_1}$ et $\overrightarrow{u_2}$: on pose $\overrightarrow{f_1}=\overrightarrow{u_1}=(1,2,-2,1)$, $\overrightarrow{f_2}=\overrightarrow{u_2}=(1,2,0,3)$, et $(\overrightarrow{f_1},\overrightarrow{f_2})$ est une base de F.

3. Par définition un vecteur $(x,y,z,t)\in\mathbb{R}^4$ est dans G si et seulement si

$$\begin{cases} x-y-z+t=0 \\ 2y+z-t=0 \\ 3x+y-z+t=0 \end{cases} \tag{75}$$

$$\iff \begin{cases} x-y-z+t=0 & \\ 2y+z-t=0 & \\ 4y+2z-2t=0 & L_3-3L_1 \end{cases} \tag{76}$$

$$\iff \begin{cases} x-y-z+t=0 & \\ 2y+z-t=0 & \\ 0=0 & L_3-2L_2 \end{cases} \tag{77}$$

Dans ce système, z et t sont toutes les deux des variables libres. Posons $z=\lambda\in\mathbb{R}$ et $t=\mu\in\mathbb{R}$. Alors L_3 donne $2y=-z+t=-\lambda+\mu$ donc $y=\frac{-1}{2}\lambda+\frac{1}{2}\mu$. Et L_1 donne $x=y+z-t=\frac{-1}{2}\lambda+\frac{1}{2}\mu+\lambda-\mu=\frac{1}{2}\lambda-\frac{1}{2}\mu$. On reconnait alors que $G=\operatorname{Vect}\Big((1/2,-1/2,1,0),(-1/2,1/2,0,1)\Big)$. Pour simplifier les calculs dans la suite, il est fort intéressant de chasser les fractions : on pose $\overrightarrow{w_1}=(1,-1,2,0)$ et $\overrightarrow{w_2}=(-1,1,0,2)$. C'est une famille génératrice de G et quand on applique cette méthode elle est automatiquement libre : si on a $\lambda\overrightarrow{w_1}+\mu\overrightarrow{w_2}$ alors en regardant la troisième composante on trouve $2\lambda=0$ donc $\lambda=0$, et idem dans la quatrième composante on trouve $\mu=0$. En résumé, $(\overrightarrow{w_1},\overrightarrow{w_2})$ est une base de G.

4. On peut former la matrice pour étudier le rang : on pose la matrice dans la base canonique de $\mathbb{R}^4$

$$A=\operatorname{Mat}(\overrightarrow{v_1},\overrightarrow{v_2},\overrightarrow{w_1},\overrightarrow{w_2})=\begin{pmatrix} 1 & 1 & 1 & -1 \\ 2 & 2 & -1 & 1 \\ -2 & 0 & 2 & 0 \\ 1 & 3 & 0 & 2 \end{pmatrix} \tag{78}$$

Alors par une suite d'opérations élémentaires

$$A \longrightarrow \begin{pmatrix} 1 & 1 & 1 & -1 \\ 0 & 0 & -3 & 3 \\ 0 & 2 & 4 & 2 \\ 0 & 2 & -1 & 3 \end{pmatrix} \tag{79}$$

$$\longrightarrow \begin{pmatrix} 1 & 1 & 1 & -1 \\ 0 & 2 & -1 & 3 \\ 0 & 0 & 5 & -5 \\ 0 & 0 & -3 & 3 \end{pmatrix} \tag{80}$$

$$\longrightarrow \begin{pmatrix} 1 & 1 & 1 & -1 \\ 0 & 2 & -1 & 3 \\ 0 & 0 & 1 & -1 \\ 0 & 0 & 1 & -1 \end{pmatrix} \tag{81}$$

$$\longrightarrow \begin{pmatrix} 1 & 1 & 1 & -1 \\ 0 & 2 & -1 & 3 \\ 0 & 0 & 1 & -1 \\ 0 & 0 & 0 & 0 \end{pmatrix} \tag{82}$$

La conclusion est que la matrice A est de rang 3, donc la famille aussi.
Ceci démontre que la famille n'est pas génératrice (le sous-espace engendré devrait être de dimension 4, or il est seulement de dimension 3), et n'est pas non plus libre (il y a 4 vecteurs, donc s'ils formaient une famille libre ils engendreraient un sous-espace de dimension 4).

5. Soit $H = \mathrm{Vect}(\overrightarrow{v_1}, \overrightarrow{v_2}, \overrightarrow{w_1}, \overrightarrow{w_2})$. Soit un vecteur quelconque $\overrightarrow{u} = (x, y, z, t) \in \mathbb{R}^4$. La condition $\overrightarrow{u} \in H$ est la condition qu'il existe $(a, b, c, d) \in \mathbb{R}^4$ tel que $\overrightarrow{u} = a\overrightarrow{v_1} + b\overrightarrow{v_2} + c\overrightarrow{w_1} + e\overrightarrow{w_2}$, ce qui conduit au système

$$(S) : \begin{cases} a + b + c - d = x \\ 2a + 2b - c + d = y \\ -2a + 2c = z \\ a + 3b + 2d = t \end{cases} \tag{83}$$

On applique alors les mêmes opérations que précédemment :

$$(S) \iff \begin{cases} a + b + c - d = x & \\ -3c + 3d = y - 2x & L_2 - 2L_1 \\ 2b + 4c - 2d = z + 2x & L_3 + 2L_1 \\ 2b - c + 3d = t - x & L_4 - L_1 \end{cases} \tag{84}$$

$$\iff \begin{cases} a + b + c - d = x & \\ 2b - c + 3d = t - x & L_4 \\ 5c - 5d = (z + 2x) - (t - x) & L_3 - L_4 \\ -3c + 3d = y - 2x & L_2 \end{cases} \tag{85}$$

$$\iff \begin{cases} a + b + c - d = x & \\ 2b - c + 3d = t - x & \\ 5c - 5d = (z + 2x) - (t - x) & \\ 0 = 3\Big((z + 2x) - (t - x)\Big) + 5\Big(y - 2x\Big) & 3L_3 + 5L_4 \end{cases} \tag{86}$$

Ce système (de rang 3 comme on l'a déjà déterminé) a une condition de compatibilité, c'est la condition à laquelle un vecteur de $\mathbb{R}^4$ est combinaison linéaire des $(\overrightarrow{v_1}, \overrightarrow{v_2}, \overrightarrow{w_1}, \overrightarrow{w_2}$, et c'est une équation de H. C'est

$$3\Big((z + 2x) - (t - x)\Big) + 5\Big(y - 2x\Big) = 0 \tag{87}$$

$$\iff 3(3x + z - t) + 5(-2x + y) \tag{88}$$

$$\iff -x + 5y + 3z - 3t = 0 \tag{89}$$

6. D'abord il faut vérifier que ces vecteurs sont dans H, c'est-à-dire vérifient l'équation de H qu'on vient de trouver. Ceci est bien le cas. Le sous-espace qu'ils engendrent est donc inclus dans H.
   On peut ensuite écrire les équations pour trouver le sous-espace qu'ils engendrent et voir que c'est bien la même équation que H. Mais il y a aussi plus simple : sachant que H est de dimension 3 (il a une base formée de 3 vecteurs) et qu'ici nous avons aussi 3 vecteurs, il suffit de vérifier qu'ils forment une famille libre. Le sous-espaces qu'ils engendrent est alors de dimension 3 inclus dans H, donc est H tout entier.
7. On peut faire beaucoup de calculs, ou le minimum pour montrer que la famille est de rang 4 (échelonner la matrice de cette famille de 4 vecteurs). Ou ne pas faire de calculs du tout : on vérifie que $\overrightarrow{e_4}$ n'est pas dans H (ne vérifie pas l'équation de H qu'on a trouvée), mais les trois premiers sont dans H, donc $\overrightarrow{e_4}$ n'est pas combinaison linéaire de ceux-là. C'est donc une famille de 4 vecteurs, qui est libre dans $\mathbb{R}^4$, donc une base de $\mathbb{R}^4$.
8. Soit $\overrightarrow{u} \in \mathbb{R}^4$, par définition puisque $(\overrightarrow{f_1}, \overrightarrow{f_2}, \overrightarrow{f_3}, \overrightarrow{e_4})$ est une base de $\mathbb{R}^4$ il existe un unique (a, b, c, d) tels que

$$\overrightarrow{u} = \underbrace{a\overrightarrow{f_1} + b\overrightarrow{f_2} + c\overrightarrow{f_3}}_{\in H} + d\overrightarrow{e_4} \tag{90}$$

   Mais dans cette combinaison linéaire un morceau est dans H, et l'autre est proportionnel à $\overrightarrow{e_4}$, il suffit donc de poser $\overrightarrow{f} = a\overrightarrow{f_1} + b\overrightarrow{f_2} + c\overrightarrow{f_3}$ et $\lambda = d$.
   On vérifie que ceci est bien l'unique choix possible : si on a un autre $(\overrightarrow{f'}, \lambda') \in H \times \mathbb{R}$ tel que $\overrightarrow{u} = \overrightarrow{f'} + \lambda'\overrightarrow{e_4}$ alors écrivant $\overrightarrow{f'}$ dans la (même) base de $\mathbb{R}^4$, avec les coefficients $(\alpha, \beta, \gamma, \delta)$

$$\overrightarrow{f'} = \alpha\overrightarrow{f_1} + \beta\overrightarrow{f_2} + \gamma\overrightarrow{f_3} + \delta\overrightarrow{e_4} \tag{91}$$

   En fait, puisque $(\overrightarrow{f_1}, \overrightarrow{f_2}, \overrightarrow{f_3})$ est déjà une base de H, la composante δ est nulle, car le vecteur s'écrit de façon *unique* comme combinaison linéaire de $(\overrightarrow{f_1}, \overrightarrow{f_2}, \overrightarrow{f_3}, \overrightarrow{f_4})$ mais il s'écrit aussi de façon unique comme combinaison linéaire de ceux-ci sans utiliser $\overrightarrow{e_4}$ (donc avec $\delta = 0$).
   On déduit alors

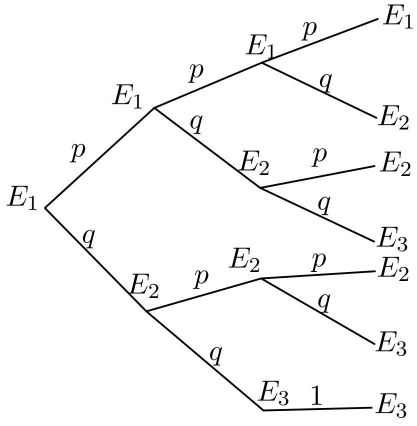
$$\begin{aligned} \overrightarrow{u} &= \left(\alpha\overrightarrow{f_1} + \beta\overrightarrow{f_2} + \gamma\overrightarrow{f_3}\right) + \lambda'\overrightarrow{e_4} \qquad (92) \\ &= \alpha\overrightarrow{f_1} + \beta\overrightarrow{f_2} + \gamma\overrightarrow{f_3} + \lambda'\overrightarrow{e_4} \qquad (93) \end{aligned}$$

   Mais comme on a aussi $\overrightarrow{u} = a\overrightarrow{f_1} + b\overrightarrow{f_2} + c\overrightarrow{f_3} + d\overrightarrow{e_4}$ et que les composantes dans une base sont uniques, on a alors $\alpha = a$, $\beta = b$, $\gamma = c$ (donc $\overrightarrow{f'} = \overrightarrow{f}$), et $\lambda' = \lambda$. Ceci démontre $(\overrightarrow{f'}, \lambda') = (\overrightarrow{f}, \lambda)$.

Problème 2

Partie 1 : Première approche

1. L'arbre qu'on trace ressemble à ceci :



   (a) Au bout d'une heure, le système est encore dans E_1 avec probabilité p, il est passé dans E_2 avec probabilité $q = 1 - p$, et il n'est pas dans E_3 (probabilité 0).

(b) La probabilité est p^3 qu'il soit toujours dans E_1 au bout de 3 heures (une seule possibilité : il reste chaque heure en E_1)

(c) Pour que le système soit à l'état E_3 au bout de 3 heures, on voit 3 chemins possibles : $E_1E_1E_2E_3$, $E_1E_2E_2E_3$ et $E_1E_2E_3E_3$. La probabilité est donc $pqq + qpq + qq = 2pq^2 + q^2$.

(d) Dans ce cas il n'y a que les chemins possibles, $E_1E_1E_2E_3$ et $E_1E_2E_2E_3$. La probabilité est $pqq + qpq = 2pq^2$.

2. Ces évènements forment un système complet d'évènements : à tout instant, le réseau est dans l'un de ces états, et dans un seul.

3. Avec la formule des probabilités totales, appliquée à l'évènement A_{k+1} par rapport au système d'évènements (A_k, B_k, C_k), on déduit pour tout $k \geqslant 0$

$$\mathbb{P}(A_{k+1}) = \mathbb{P}(A_k)\mathbb{P}_{A_k}(A_{k+1}) + \mathbb{P}(B_k)\mathbb{P}_{B_k}(A_{k+1}) + \mathbb{P}(C_k)\mathbb{P}_{C_k}(A_{k+1}) \tag{94}$$

Dans cette somme, le terme tout à gauche est a_{k+1}. Par définition $\mathbb{P}_{A_k}(A_{k+1}) = p$, $\mathbb{P}_{B_k}(A_{k+1}) = 0$ (probabilité de passer à E_1 sachant qu'on est en E_2 : c'est impossible, le système ne peut que se dégrader), et $\mathbb{P}_{C_k}(A_{k+1}) = 0$ (idem). Il n'y a qu'à remplacer, on trouve bien $a_{k+1} = pa_k$.
De même pour B_{k+1}, on trouve

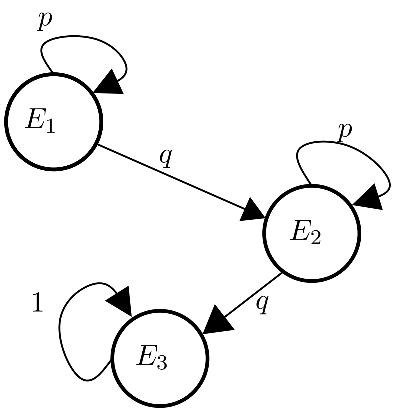
$$\mathbb{P}(B_{k+1}) = \mathbb{P}(A_k)\mathbb{P}_{A_k}(B_{k+1}) + \mathbb{P}(B_k)\mathbb{P}_{B_k}(B_{k+1}) + \mathbb{P}(C_k)\mathbb{P}_{C_k}(B_{k+1}) \tag{95}$$

Ici $\mathbb{P}_{A_k}(B_{k+1}) = q$, $\mathbb{P}_{B_k}(B_{k+1}) = p$ et $\mathbb{P}_{C_k}(B_{k+1}) = 0$. On trouve bien $b_{k+1} = qa_k + pb_k$.
Prolongeant la même méthode pour C_k on a :

$$\mathbb{P}(C_{k+1}) = \mathbb{P}(A_k)\mathbb{P}_{A_k}(C_{k+1}) + \mathbb{P}(B_k)\mathbb{P}_{B_k}(C_{k+1}) + \mathbb{P}(C_k)\mathbb{P}_{C_k}(C_{k+1}) \tag{96}$$

Ici $\mathbb{P}_{A_k}(C_{k+1}) = 0$ (on ne peut pas passer directement de E_1 à E_3), $\mathbb{P}_{B_k}(C_{k+1}) = q$, $\mathbb{P}_{C_k}(C_{k+1}) = 1$ (une fois en E_3, on y reste). On trouve donc $c_{k+1} = qb_k + c_k$

4. Il s'agit directement d'une question inspirée par le TP d'aléatoire, la section processus aléatoire (le chat). Le graphe ressemble à ceci :



Remarquez que sur chaque sommet, la somme des probabilités sur les arêtes *sortantes* est de 1.

Partie 2 : Modélisation en Python

5. Comme nous avons vu dans le TP.

```python
def heure_suivante(x, p):
    if x == 1:
        t = rd.random()
        if t < p:
            return 1
        else:
            return 2
    elif x == 2:
        t = rd.random()
        if t < p:
            return 2
        else:
            return 3
    elif x == 3:
        return 3
```

6. C'est une boucle qui fait `x = heure_suivante(x, p)`, ce qui fait avancer l'état d'une heure.

```python
def état_futur(p, k):
    # état inital
    x = 1
    for i in range(k):
        x = heure_suivante(x, p)
    return x
```

7. Attention ici car il faut une boucle `while`, en espérant qu'on atteindra bien l'état E_3 à un moment (sinon, c'est une boucle infinie).

```python
def temps_survie(p):
    # état inital
    x = 1
    # heure initiale
    k = 0
    while x != 3:
        x = heure_suivante(x, p)
        k = k + 1
    # si on arrive là, la boucle s'est arrêtée, on est en E3
    return k
```

8. Si la fonction précédente est correcte, il suffit de l'appeler N fois et de calculer la moyenne.

```python
def temps_survie_moyen(p, N):
    # somme, pour calculer la moyenne
    S = 0
    for i in range(N):
        k = temps_survie(p)
        S = S + k
    return S / N
```

Partie 3 : Étude matricielle

9. Les trois équations données par $X_{k+1} = MX_k$ sont précisément

$$\begin{cases} a_{k+1} = pa_k \\ b_{k+1} = qa_k + pb_k \\ c_{k+1} = qb_k + c_k \end{cases} \tag{97}$$

et ce sont bien celles démontrées dans la partie précédente.

10. Par récurrence, on en déduit $\forall k \in \mathbb{N}$, $X_k = M^k X_0$: c'est vrai pour $k = 0$ car $M^0 = I_3$ et $I_3 X_0 = X_0$, et si c'est vrai au rang k alors $X_{k+1} = MX_k = M\left(M^k X_0\right) = M^{k+1}X_0$.
Il s'agit en fait de l'analogue d'une suite géométrique, mais dans les matrices !
11. Par les relations ci-dessus, le coefficient (i,j) de M donne précisément la probabilité de passer, en une heure, de l'état j vers l'état i. Mais alors, comme $X_{k+2} = M^2 X_k$, le coefficient (i,j) de M^2 donne, par ce même principe, la probabilité de passer de j vers i en *deux* heures exactement ; plus généralement, le coefficient (i,j) de M^k est la probabilité de passer de l'état j à l'état k au bout d'exactement k heures.
12. Montrons par récurrence qu'il existe un coefficient u_k tel que

$$\forall k \in \mathbb{N}, \quad M^k = \begin{pmatrix} p^k & 0 & 0 \\ kqp^{k-1} & p^k & 0 \\ u_k & 1-p^k & 1 \end{pmatrix} \tag{98}$$

Pour $k = 0$ on doit avoir $M^k = I_3$, ce qui est bien le cas ici, avec $u_0 = 0$.
Soit $k \in \mathbb{N}$, supposons que l'égalité soit vérifiée et qu'on ait construit u_k, on trouve alors

$$M^{k+1} = M \cdot M^k = \begin{pmatrix} p & 0 & 0 \\ q & p & 0 \\ 0 & q & 1 \end{pmatrix} \begin{pmatrix} p^k & 0 & 0 \\ kqp^{k-1} & p^k & 0 \\ u_k & 1-p^k & 1 \end{pmatrix} \tag{99}$$

$$= \begin{pmatrix} p \times p^k & 0 & 0 \\ q \times p^k + p \times kqp^{k-1} & p \times p^k & 0 \\ q \times kqp^{k-1} + u_k & qp^k + (1-p^k) & 1 \end{pmatrix} \tag{100}$$

$$= \begin{pmatrix} p^{k+1} & 0 & 0 \\ (q+kq)p^k & p^{k+1} & 0 \\ kq^2p^{k-1} + u_k & 1 + (q-1)p^k & 1 \end{pmatrix} \tag{101}$$

$$= \begin{pmatrix} p^{k+1} & 0 & 0 \\ (1+k)qp^k & p^{k+1} & 0 \\ kq^2p^{k-1} + u_k & 1 - p^{k+1} & 1 \end{pmatrix} \tag{102}$$

(à la fin on utilise $q = 1-p$, donc $(q-1)p^k = -p \times p^k = -p^{k+1}$).
On voit que la matrice est bien sous forme voulue si on pose $u_{k+1} = kq^2p^{k-1} + u_k$. Pas besoin d'avoir de valeur plus explicite !
13. Sachant que $\left(\begin{smallmatrix} a_k \\ b_k \\ c_k \end{smallmatrix}\right) = M^k X_0 = M^k \left(\begin{smallmatrix} 1 \\ 0 \\ 0 \end{smallmatrix}\right)$, on en déduit

$$\forall k \in \mathbb{N}, \quad \begin{cases} a_k = p^k \\ b_k = kqp^{k-1} \\ c_k = u_k \end{cases} \tag{103}$$

Puisque $p \in \left]0,1\right[$ on a alors $\lim_{n\to+\infty} a_k = 0$: la probabilité tend vers 0 que le système reste toujours dans l'état E_1.
De même pour b_k, mais avec nécessité d'invoquer la comparaison entre suite polynomiales et géométriques, par exemple $k = o((1/q)^k)$ pour $k \to +\infty$ et $kqp^{k-1} = k \times q^k \times \frac{q}{p}$ (le dernier facteur est une constante). On en déduit $\lim_{n\to+\infty} b_k = 0$: plus le temps passe, moins il devient probable que le système soit en E_2.
Enfin puisqu'on a toujours $a_k + b_k + c_k = 1$ on en déduit que $\lim_{n\to+\infty} c_k = 1$: plus le temps passe, plus il devient probable que le système soit à l'arrêt.