DS 5 Mathématiques Correction

Problème: une petite parenthèse enchantée

Remarque 1. Les nombres C_n sont très connus sous le nom **nombres de Catalan** et interviennent dans de nombreuses situations de dénombrement. Les premiers nombres de Catalan sont les suivants :

7	n	0	1	2	3	4	5	6	7	8
C	\vec{r}_n	1	1	2	5	14	42	132	429	1430

A. Dénombrement manuel

- 1. (a) Avec une seule paire : $\mathcal{M}_1 = \{ "() " \}$.
 - Avec deux paires : $\mathcal{M}_2 = \{"()()","(())"\}.$
 - Avec trois paires on trouve

$$\mathcal{M}_{3} = \left\{ "()()()","()(())","(())()","(()())","((()))" \right\}$$
 (2)

- (b) On en déduit $C_1 = 1$, $C_2 = 2$ et $C_3 = 5$.
- 2. Les 14 mots avec 4 paires de parenthèses sont :

Ils sont ici déjà listés selon l'ordre récursif de la partie C...

3. On peut former 2^q mots de longueur q sur deux symboles de parenthèse ouvrante ou fermante. Comme les mots bien parenthésés à n paires de parenthèses forment un sous-ensemble des mots de longueur 2n, on en déduit $C_n \leqslant 2^{2n}$.

B. Un test de parenthésage

4. • Pour a = "(((())()))":

i	0	1	2	3	4	5	6	7	8	9
a[i]	(((())	()))
$p_{a}(i)$	1	2	3	4	3	2	3	2	1	0

• Pour b = "(()()))(()":

	i	0	1	2	3	4	5	6	7	8	9
ŀ	b[i]	(()	()))	(()
Ţ	$\rho_{\mathtt{b}}(i)$	1	2	1	2	1	0	-1	0	1	0

- 5. C'est une reformulation de la définition adoptée dans l'introduction : la condition (ii) signifie uniquement qu'il y a *autant* de parenthèses ouvrantes que fermantes, et la condition (i) signifie qu'à tout moment en lisant le mot de gauche à droite on a toujours *au moins autant* de parenthèses ouvrantes que fermantes.
 - On voit aussi que la valeur de $p_{m}(i)$ indique à quelle « profondeur » (au sens de l'emboitement des parenthèses) on se trouve dans le mot, et qu'un indice i tel que $p_{m}(i)=0$ signifie qu'on referme la parenthèse correspondante à la toute première parenthèse (nécessairement ouvrante) du mot.
- 6. Le mot b ci-dessus vérifie (ii) mais pas (i) : il a autant de parenthèses ouvrantes que fermantes, mais n'est pas bien parenthésé. L'exemple le plus simple qu'on peut prendre est en fait ") (".

- Un mot qui vérifie (i) mais pas (ii) est un mot qui a des parenthèses ouvrantes qui ne sont pas refermées à la fin, et l'exemple le plus simple que l'on peut prendre (de longueur paire) est en fait "((".
- 7. On connait à l'avance la longueur de la liste P, il suffit donc de faire une boucle pour parcourir tous les caractères de m pour remplir la liste case par case. On compte 1 de plus si on lit une parenthèse ouvrante, un de moins si on ferme. Il faut faire attention avec le premier caractère...

```
def p(m):
    n = len(m)
    P = [0] * n
    for i in range(1, n):
        if m[i] == "(":
             # petit truc ennuyeux
            if i > 0:
                P[i] = P[i-1] + 1
            else:
                P[i] = 1
        elif m[i] == ")":
            if i > 0:
                 P[i] = P[i-1] - 1
            else:
                P[i] = -1
    return P
```

8. On peut tout à fait utiliser la fonction précédente : il faut vérifier que tous les termes de la liste sont positifs et que le dernier est nul.

```
def est_bien_parenthésé(m):
    n = len(m)
    P = p(m)
    for i in range(n):
        if P[i] < 0:
            return False
    # ici la condition (i) est vérifiée ; maintenant la (ii)
    if P[n-1] == 0:
        return True
    else:
        return False</pre>
```

On peut aussi se passer complètement de la liste, en calculant seulement les valeurs d'une variable p au fur et à mesure — qui ne doit jamais devenir négative.

```
def est_bien_parenthésé(m):
    n = len(m)
    p = 0
    for i in range(n):
        if m[i] == "(":
            p = p + 1
        elif m[i] == ")":
            p = p - 1
            if p < 0:
                 return False

if p == 0:
        return True
    else:
        return False</pre>
```

C. Une approche récursive

- 9. (a) $\mathcal{N}_{3,0} = \{ "()()()","()(())" \}$: on doit démarrer par "()" puis on ne peut que lui concaténer les deux mots possibles de \mathcal{M}_2 qui sont "()()" et "(())".
 - $\mathcal{N}_{3,1} = \{ "(())()" \} :$ on doit démarrer par "(" et laisser deux caractères puis placer un ")", on ne peut donc que placer "()" dans ce trou et aussi après le mot.
 - $\mathcal{N}_{3,2} = \left\{ \text{"(()())","((()))"} \right\}$: le mot doit être encadré entièrement par des parenthèses; entre, on peut seulement caser les deux mots de \mathcal{M}_2 .
 - (b) $\mathcal{N}_{4,1} = \left\{ \text{"(())()()","(())(())"} \right\}$: le mot ne peut commencer que par "(())", auquel on peut concaténer deux mots possibles de \mathcal{M}_2 .
 - $\mathcal{N}_{4,2} = \{ "(()())()", "((()))()" \}$: on doit démarrer par une parenthèses ouvrante qui se referme en laissant la place de caser un mot de \mathcal{M}_2 , et à la fin on ne peut plus mettre que "()".
 - $\mathcal{N}_{5,2} = \left\{ \text{"(()())()()","((()))()","(()())","((()))(())"} \right\}$: on place la première parenthèse ouvrante et la fermante qui lui correspond; entre, on peut placer chacun des deux mots de \mathcal{M}_2 , et à la fin, on peut aussi placer chacun des deux mots de \mathcal{M}_2 .
- 10. (a) D'abord par définition $\mathcal{N}_{n,0}$ est l'ensemble des mots à n paires de parenthèses ouvrantes-fermantes qui commencent par "()". Plus généralement pour un mot de \mathcal{M}_n la parenthèse qui referme la première parenthèse (nécessairement ouvrante) doit être à un indice $impair\ 2k+1$, en effet le mot qui est alors pris entre parenthèses doit être encore bien parenthésé, donc de longueur paire! C'est aussi le premier indice où la fonction p de la partie précédente revient à 0. De plus on doit avoir $1 \le 2k+1 \le n-1$ c'est-à-dire $0 \le k \le n-1$.
 - c'est-à-dire $0 \le k \le n-1$. En conclusion $extbf{les} (\mathcal{N}_{n,k})_{k \in \llbracket 0,n-1 \rrbracket}$ forment une partition de \mathcal{M}_n .
 - (b) D'abord l'application Φ est bien définie : si x est bien parenthésé avec k paires de parenthèses ouvrantesfermantes, alors le mot "("+x+")" est encore bien parenthésé (c'est x « emboité entre parenthèses ») et avec k+1 paires ; on lui concatène y qui est bien parenthésé à n-1-k paires et on obtient un mot bien parenthésé à n paires, qui est par définition dans $\mathcal{N}_{n,k}$.
 - L'application Φ est bijective car on peut donner directement son inverse : partant d'un mot de $\mathcal{N}_{n,k}$ alors la première parenthèse (le mot démarre nécessairement par une parenthèse ouvrante) se referme à l'indice 2k+1, entre il y a un mot bien parenthésé à k paires (c'est x), et après, il y a le mot y qui est encore bien parenthésé et à n-1-k paires.
 - (c) On fixe $n \ge 1$. Comme les $(\mathcal{N}_k)_{0 \le n-1}$ forment une partition de \mathcal{M}_n on a d'abord

$$\operatorname{Card}(\mathcal{M}_n) = \sum_{k=0}^{n-1} \operatorname{Card}(\mathcal{N}_{n,k})$$
(6)

Puis à cause de la bijection ci-dessus alors $\forall 0\leqslant k\leqslant n-1$

$$\operatorname{Card}(\mathcal{N}_{n,k}) = \operatorname{Card}(\mathcal{M}_k \times \mathcal{M}_{n-1-k}) = \operatorname{Card}(\mathcal{M}_k) \times \operatorname{Card}(\mathcal{M}_{n-1-k}) \tag{7}$$

et donc puisque $C_n = \operatorname{Card}(\mathcal{M}_n)$ ce la donne bien la relation de récurrence

$$C_n = \sum_{k=0}^{n-1} C_k \times C_{n-1-k}$$
 (8)

valable pour tout $n \ge 1$.

Remarque 2. La conclusion de tout cela, c'est que les mots bien parenthésés sont tous obtenus en ouvrant d'abord une paire de parenthèses, à l'intérieure de laquelle on colle un mot bien parenthésé plus petit, et on concatène à la fin un autre mot bien parenthésé. Si le mot final est formé de n paires, alors on met un mot à k paires entre parenthèses, et après, on peut encore concaténer un mot à n-1-k paires. Ainsi on explique simultanément la formule de récurrence précédente, mais aussi comment obtenir tous les mots bien parenthésés par une fonction récursive, et comment les lister intelligemment dans d'ordre : on place d'abord la première paire de parenthèses celle ouvrante en position 0, et on colle dans la place restante tous les mots bien parenthésés possibles plus petits.

11. Sachant $C_0=1$, la formule (*) donne automatiquement $C_1=C_0\times C_0$ donc $\boxed{C_1=1}$, puis $C_2=C_1\times C_0+C_0\times C_1$ soit $\boxed{C_2=2}$, ensuite $C_3=C_0\times C_2+C_1\times C_1+C_2\times C_0=1+2\times 2+1$ soit $\boxed{C_3=5}$. On connaissait bien sûr déjà ces valeurs par le dénombrement, ou alors on se rend compte à cette étape s'il en manque dans les premières questions...

```
 \begin{array}{l} \text{On trouve ensuite } C_4 = C_0 \times C_3 + C_1 \times C_2 + C_2 \times C_1 + C_3 \times C_0 \text{ soit } C_4 = 1 \times 5 + 1 \times 2 + 2 \times 1 + 5 \times 1 = 5 + 2 + 2 + 5 \\ \text{d'où } \boxed{C_4 = 14}. \\ \text{Enfin } C_5 = C_0 \times C_4 + C_1 \times C_3 + C_2 \times C_2 + C_3 \times C_1 + C_4 \times C_0 \text{ soit } C_5 = 1 \times 14 + 1 \times 5 + 2 \times 2 + 5 \times 1 + 14 \times 1 = 14 + 5 + 4 + 5 + 14 \text{ d'où } \boxed{C_5 = 42}. \end{array}
```

12. (a) On calcule une somme en appelant récursivement la fonction, ce n'est pas difficile.

```
def C(n):
    if n == 0:
        return 1
    else:
        S = 0
        for k in range(n):
            S = S + C(k) * C(n-1-k)
        return S
```

(b) On écrit une fonction qui calcule les termes d'une suite, en partant d'une liste de zéros (on sait faire) qu'on prend de taille n+1 (pour avoir les C_k avec $0 \le k \le n$. Pour chaque terme suivant de la suite, on calcule une somme.

```
def C_liste(n):
    L = [0] * (n+1)
    L[0] = 1
    for i in range(1, n+1):
        # jusqu'ici comme d'habitude, puis on calcule une somme
        S = 0
        for k in range(i):
            S = S + L[k] * L[i-1-k]
        L[i] = S
    return L
```

(c) La première fonction s'appelle récursivement elle-même de nombreuses fois avec tous les entiers possibles plus petits. Il y a un « arbre des appels » (c.f. l'exemple de Fibonacci dans le TP sur la récursivité) compliqué, encore plus compliqué que pour Fibonacci. La deuxième fonction calcule en quelque sorte plus de choses (la liste de tous les termes), mais c'est direct, elle ne revient pas de nombreuses fois en arrière. Elle est en fait bien plus efficace. Pour calculer seulement C_n , il suffirait de lui faire renvoyer seulement le dernier terme de la liste.

Tout cela est en fait bien normal : puisque chaque terme de la suite dépend de *tous* les précédents, c'est la méthode la plus normale possible de vouloir calculer une *liste* de tous les termes de la suite, pour les garder en mémoire, sinon on serait obligé de les recalculer plusieurs fois.

13. Les questions précédentes nous disent clairement comment procéder pour lister intelligemment les mots de \mathcal{M}_n : il faut choisir tous les mots possibles x de \mathcal{M}_k et tous les mots possibles y de \mathcal{M}_{n-1-k} , et pour tout k, qu'on récupère grâce à la fonction récursive; et former le mot "("+x+")"+y. Le programme est donc fortement lié aux raisonnements précédents.

En pratique, on écrirait cette boucle plutôt avec l'itération sur les éléments for x in X pour le rendre plus clair :

```
X = mots_bien_parenthésés(k)
Y = mots_bien_parenthésés(n-1-k)
for x in X:
    for y in Y:
        P.append("(" + x + ")" + y)
```

D. Une formule pour C_n

14. Par définition

$$a_n = \frac{1}{n+1} \times \frac{(2n)!}{n! \times n!} \tag{9}$$

Écrivons alors

$$a_{n+1} = \frac{1}{n+2} \times \frac{(2n+2)!}{(n+1)! \times (n+1)!} = \frac{1}{n+2} \times \frac{(2n+2) \times (2n+1) \times (2n)!}{(n+1) \times n! \times (n+1) \times n!}$$
(10)

mais 2n+2=2(n+1) qui se simplifie une fois en haut et en bas, autrement dit

$$a_{n+1} = \frac{1}{n+2} \times \frac{2(2n+1) \times (2n)!}{(n+1) \times n! \times n!} = \frac{2(2n+1)}{n+2} \times \frac{1}{n+1} \times \frac{(2n)!}{n! \times n!}$$
(11)

et on reconnait bien $a_{n+1} = \frac{2(2n+1)}{n+2}a_n$.

15. (a) Dans la somme pour T_n , on effectue le changement d'indice j=n-1-k. Cela est équivalent à k=n-1-j, et k=0 correspond alors à j=n-1, et k=n-1 correspond bien à j=0. Bref, on trouve directement

$$T_n = \sum_{j=0}^{n-1} (n-1-j)a_{n-1-j}a_j \tag{12}$$

Quitte à renommer j en k, et inverser les deux terme a, cela est bien

$$T_n = \sum_{k=0}^{n-1} (n-1-k)a_k a_{n-1-k}$$
(13)

(b) On calcule alors $T_n + T_n$, l'un des deux avec cette nouvelle formule et l'autre sans :

$$T_n + T_n = \sum_{k=0}^{n-1} k a_k a_{n-1-k} + \sum_{k=0}^{n-1} (n-1-k) a_k a_{n-1-k}$$
 (14)

$$=\sum_{k=0}^{n-1} \left(ka_k a_{n-1-k} + (n-1-k)a_k a_{n-1-k}\right) \tag{15}$$

$$=\sum_{k=0}^{n-1}(k+(n-1-k))a_ka_{n-1-k} \tag{16}$$

$$=\sum_{k=0}^{n-1} (n-1)a_k a_{n-1-k} \tag{17}$$

$$= \left[(n-1) \sum_{k=0}^{n-1} a_k a_{n-1-k} = (n-1) S_n = T_n + T_n \right]$$
 (18)

(c) On calcule cette fois (avec la première expression de T_n)

$$T_{n+1} + S_{n+1} = \sum_{k=0}^{n} k a_k a_{n-k} + \sum_{k=0}^{n} a_k a_{n-k} = \sum_{k=0}^{n} (k+1) a_k a_{n-k}$$
(19)

Sous la somme, on utilise la relation de récurrence de la première question qui dit $(k+1)a_k = 2(2k-1)a_k$ $1)a_{k-1}$ pour tout $k \ge 1$, il faut donc sortir le terme k=0 de la somme (avec $a_0=1$):

$$T_{n+1} + S_{n+1} = a_n + \sum_{k=1}^{n} (k+1)a_k a_{n-k} = a_n + \sum_{k=1}^{n} 2(2k-1)a_{k-1}a_{n-k}$$
 (20)

On voit qu'on peut alors poser le changement d'indice j = k - 1, c'est-à-dire k = j + 1, et on trouve

$$T_{n+1} + S_{n+1} = a_n + \sum_{j=0}^{n-1} 2(2j+1)a_j a_{n-1-j}$$
(21)

Il n'y a alors plus qu'à séparer cette somme par linéarité :

$$T_{n+1} + S_{n+1} = a_n + \sum_{j=0}^{n-1} \left(4ja_j a_{n-1-j} + 2a_j a_{n-1-j} \right)$$
 (22)

$$=a_{n}+4\sum_{j=0}^{n-1}ja_{j}a_{n-1-j}+2\sum_{j=0}^{n-1}a_{j}a_{n-1-j} \tag{23}$$

$$= a_n + 4T_n + 2S_n = T_{n+1} + S_{n+1}$$
 (24)

16. Des relations précédentes on peut éliminer les T_n : on tire $T_n=\frac{(n-1)S_n}{2}$ donc $4T_n+2S_n=2(n-1)S_n+2S_n=2nS_n$, et aussi donc $T_{n+1}=\frac{nS_{n+1}}{2}$ soit $T_{n+1}+S_{n+1}=\frac{(n+2)S_{n+1}}{2}$. La formule encadrée précédemment se lit donc

$$\frac{(n+2)S_{n+1}}{2} = a_n + 2nS_n \tag{25}$$

Partant de là il est assez rapide de démontrer par récurrence sur $n \in \mathbb{N}$ la propriété $\mathcal{P}(n)$: « $S_n = a_n$ ».

- Initialisation n=0 : directement $S_0=a_0,$ et d'ailleurs les deux termes sont égaux à 1.
- Soit $n \in \mathbb{N}$, supposons $\mathcal{P}(n)$. Alors on sait

$$\frac{(n+2)S_{n+1}}{2} = a_n + 2nS_n = a_n + 2na_n = (2n+1)a_n \tag{26}$$

et donc

$$S_{n+1} = \frac{2(2n+1)}{n+2}a_n \tag{27}$$

mais précisément on reconnait $S_{n+1} = a_{n+1}$

Conclusion : $\forall n \in \mathbb{N}, S_n = a_n$

- 17. C'est clair si on pose l'hypothèse de récurrence $\mathcal{P}(n)$: « $\forall 0 \leqslant k \leqslant n, C_k = a_k$ », pour tout $n \in \mathbb{N}$.
 - Initialisation n = 0: on sait $C_0 = 1$, et d'autre part $a_0 = 1$.
 - Soit $n \in \mathbb{N}$, supposons $\mathcal{P}(n)$. On a alors

$$C_{n+1} = \sum_{k=0}^{n-1} C_k C_{n-1-k} = \sum_{k=0}^{n-1} a_k a_{n-1-k} = a_{n+1}$$
(28)

Conclusion : $\forall n \in \mathbb{N}, C_n = a_n$

(En d'autres termes, on démontre que les nombres $(a_n)_{n\in\mathbb{N}}$ vérifient la même relation de récurrence que les $(C_n)_{n\in\mathbb{N}}$, et ont la même valeur en 0, et donc ils sont automatiquement égaux pour tout $n\in\mathbb{N}$.)

18. (a) Sachant $a_0 = 1$, c'est très direct :

```
def a(n):
    if n == 0:
        return 1
    else:
        return 2*(2*n+1) / (n+2) * a(n-1)
```

(b) Cette fonction est beaucoup plus efficace : elle est toujours récursive mais ne s'appelle elle-même qu'une seule fois à chaque fois, l'arbre des appels est une « ligne droite ».

Problème 2 : Intégrales

1. D'abord

$$I_0 = \int_0^1 1 \, \mathrm{d}t = \left[t\right]_0^1 = \boxed{1 = I_0}.$$
 (29)

Pour I_1 on a aussi une primitive directement :

$$I_1 = \int_0^1 (1 - t^2) \, \mathrm{d}t = \left[t - \frac{t^3}{3} \right]_0^1 = 1 - \frac{1}{3} = \left[\frac{2}{3} = I_1 \right]$$
 (30)

Pour I_3 il suffit de développer : $\forall t \in [0,1], (1-t^2)^2 = 1-2t^2+t^4$, puis on trouve là encore une primitive directement (ce sont des polynômes).

$$I_4 = \int_0^1 (1 - 2t^2 + t^4) \, dt = \left[t - \frac{2t^3}{3} + \frac{t^5}{5} \right]_0^1 = 1 - \frac{2}{3} + \frac{1}{5} = \boxed{\frac{8}{15} = I_3}$$
 (31)

2. (a) Soit $n \in \mathbb{N}$, alors par linéarité

$$I_{n+1} - I_n = \int_0^1 \left((1 - t^2)^n - (1 - t^2)^{n+1} \right) \mathrm{d}t = \int_0^1 (1 - t^2)^n \left(1 - (1 - t^2) \right) \mathrm{d}t$$

Soit

$$I_{n+1} - I_n = \int_0^1 (1-t^2)^n \times (-t^2) \, \mathrm{d}t = \boxed{-\int_0^1 t^2 (1-t^2)^n \, \mathrm{d}t = I_{n+1} - I_n}$$

(b) Pour tout $n \in \mathbb{N}$ et pour tout $t \in [0,1]$ alors d'une part $t^2 \ge 0$, d'autre part $t^2 \le 1$ donc $1-t^2 \ge 0$ et ainsi $(1-t^2)^n \ge 0$. Le terme sous l'intégral $t^2(1-t^2)^n$ est donc bien positif pour tout $t \in [0,1]$. Par positivité de l'intégrale on déduit

$$\int_{0}^{1} t^{2} (1 - t^{2})^{n} \, \mathrm{d}t \geqslant 0 \tag{32}$$

Donc avec le signe moins devant, $I_{n+1} - I_n \leq 0$. Cela est vrai pour tout $n \in \mathbb{N}$ et signifie que la suite $(I_n)_{n \in \mathbb{N}}$ est décroissante.

3. On peut essayer diverses stratégies, se tromper et ré-essayer... Ce qui marche est d'intégrer par parties en partant de I_{n+1} et en écrivant $(1-t^2)^{n+1}=1\times (1-t^2)^{n+1}$. On pose alors u'(t)=1, de primitive u(t)=t, et $v(t)=(1-t^2)^{n+1}$ de dérivée $v'(t)=-2t(n+1)(1-t^2)^n$. Ce sont bien quatre fonctions continues sur [0,1]. L'intégration par parties donne alors

$$I_{n+1} = \left[t(1-t^2)^{n+1} \right]_0^1 - \int_0^1 t \times (-2(n+1))t(1-t^2)^n \, \mathrm{d}t \tag{33}$$

soit, les termes entre crochet étant nuls en t=0 et en t=1, et en sortant les constantes et les signes de l'intégrale :

$$I_{n+1} = 2(n+1) \int_0^1 t^2 (1-t^2)^n \, \mathrm{d}t$$
 (34)

On reconnait alors $-(I_{n+1}-I_n)$ de la question précédente, donc $I_{n+1}=(2n+2)(I_n-I_{n+1})$. De cette relation il est facile de tirer $(2n+3)I_{n+1}=(2n+2)I_n$ soit $\boxed{I_{n+1}=\frac{2n+2}{2n+3}I_n}$.

4. On peut démontrer le résultat par récurrence sur $n \in \mathbb{N}$, en fixant $t \in [0,1]$. La propriété est donc

$$\mathcal{P}(n): \left(\sum_{k=0}^{n} \frac{(1-t^2)^k}{2^{k+1}} = \frac{1}{1+t^2} - \frac{(1-t^2)^{n+1}}{2^{n+1}(1+t^2)} \right)$$
 (35)

Pour n = 0: tout cela se réduit à

$$\frac{1}{2} = \frac{1}{1+t^2} - \frac{1-t^2}{2(1+t^2)} \tag{36}$$

ce qui est bien vrai : mettant au même dénominateur à droite on trouve $\frac{2(1+t^2)-(1-t^2)}{2(1+t^2)}$ qui se simplifie en $\frac{1+t^2}{2(1+t^2)}$ soit $\frac{1}{2}$. Donc $\boxed{\mathcal{P}(0)}$ est vraie.

Soit maintenant $n \in \mathbb{N}$, supposons $\mathcal{P}(n)$ vérifiée. Alors on écrit

$$\sum_{k=0}^{n+1} \frac{(1-t^2)^k}{2^{k+1}} = \sum_{k=0}^{n} \frac{(1-t^2)^k}{2^{k+1}} + \frac{(1-t^2)^{n+1}}{2^{n+2}}$$
(37)

ce qui par l'hypothèse de récurrence $\mathcal{P}(n)$ donne

$$\sum_{k=0}^{n+1} \frac{(1-t^2)^k}{2^{k+1}} = \frac{1}{1+t^2} - \frac{(1-t^2)^{n+1}}{2^{n+1}(1+t^2)} + \frac{(1-t^2)^{n+1}}{2^{n+2}}$$
(38)

Il ne reste plus qu'à mettre les deux dernier termes au même dénominateur, ce qui doit forcément marcher... Or

$$-\frac{(1-t^2)^{n+1}}{2^{n+1}(1+t^2)} + \frac{(1-t^2)^{n+1}}{2^{n+2}} = -\frac{2(1-t^2)^{n+1}}{2^{n+2}(1+t^2)} + \frac{(1-t^2)^{n+1}(1+t^2)}{2^{n+2}(1+t^2)}$$
(39)

$$=\frac{(-2+(1+t^2))(1-t^2)^{n+1}}{2^{n+2}(1+t^2)}\tag{40}$$

$$= -\frac{(1-t^2)(1-t^2)^{n+1}}{2^{n+2}(1+t^2)} \tag{41}$$

$$= -\frac{(1-t^2)^{n+2}}{2^{n+2}(1+t^2)} \tag{42}$$

et donc on trouve bien

$$\sum_{k=0}^{n+1} \frac{(1-t^2)^k}{2^{k+1}} = \frac{1}{1+t^2} = -\frac{(1-t^2)^{n+2}}{2^{n+2}(1+t^2)}$$
(43)

et $|\mathcal{P}(n+1)|$ est vraie

En conclusion par récurrence, $\mathcal{P}(n)$ est vraie pour tout $n \in \mathbb{N}$

Remarque : une autre approche consiste à utiliser la formule pour la somme des termes successifs d'une suite géométrique, quitte à écrire $\frac{(1-t^2)^k}{2^{k+1}}=\frac{1}{2}\times\left(\frac{1-t^2}{2}\right)^k$ qui donne directement

$$\sum_{k=0}^{n} \frac{(1-t^2)^k}{2^{k+1}} = \frac{1}{2} \times \frac{1 - \left(\frac{1-t^2}{2}\right)^{n+1}}{1 - \left(\frac{1-t^2}{2}\right)}$$
(44)

Multiplier par 2 en haut et en bas donne assez vite le résultat :

$$\frac{1}{2} \times \frac{1 - \left(\frac{1 - t^2}{2}\right)^{n+1}}{1 - \left(\frac{1 - t^2}{2}\right)} = \frac{1}{2} \times \frac{2 - \left(\frac{1 - t^2}{2}\right)^{n+1}}{2 - (1 - t^2)} = \frac{1}{2} \times \frac{2 - \frac{(1 - t^2)^{n+1}}{2^{n+1}}}{1 + t^2} = \frac{1}{1 + t^2} - \frac{(1 - t^2)^{n+1}}{2^{n+1}(1 + t^2)} \tag{45}$$

5. Par définition, grâce à la linéarité pour la somme et pour l'intégrale, pour tout $n \in \mathbb{N}$,

$$S_n = 4\sum_{k=0}^n \frac{1}{2^{k+1}} \left(\int_0^1 (1-t^2)^k \, \mathrm{d}t \right)$$
 (46)

$$=4\sum_{k=0}^{n}\int_{0}^{1}\frac{(1-t^{2})^{k}}{2^{k+1}}\,\mathrm{d}t\tag{47}$$

$$=4\int_{0}^{1} \sum_{k=0}^{n} \frac{(1-t^{2})^{k}}{2^{k+1}} dt$$
 (48)

Autrement dit il s'agit surtout d'intégrer l'égalité précédente sur $t \in [0, 1]$. Par ces calculs (et la linéarité de la somme), on a alors

$$S_n = 4 \int_0^1 \frac{1}{1+t^2} dt - 4 \int_0^1 \frac{(1-t^2)^{n+1}}{2^{n+1}(1+t^2)} dt$$
 (49)

Pour la première de ces intégrales : on reconnait une primitive en arctan

$$4\int_{0}^{1} \frac{1}{1+t^{2}} dt = 4 \times \left[\arctan(t)\right]_{0}^{1} = 4 \times \left(\arctan(1) - \arctan(0)\right) = 4 \times \left(\frac{\pi}{4} - 0\right) = \pi$$
 (50)

Pour la deuxième on ne peut pas la calculer immédiatement mais

$$4\int_{0}^{1} \frac{(1-t^{2})^{n+1}}{2^{n+1}(1+t^{2})} dt = \frac{4}{2^{n+1}} \int_{0}^{1} \frac{(1-t^{2})^{n+1}}{1+t^{2}} dt = \frac{1}{2^{n-1}} \int_{0}^{1} \frac{(1-t^{2})^{n+1}}{1+t^{2}} dt$$
 (51)

On a alors démontré

$$S_n = \pi - \frac{1}{2^{n-1}} \int_0^1 \frac{(1-t^2)^{n+1}}{1+t^2} dt$$
 (52)

ce qui est bien équivalent à

$$\pi - S_n = \frac{1}{2^{n-1}} \int_0^1 \frac{(1-t^2)^{n+1}}{1+t^2} \, \mathrm{d}t$$
 (53)

valable pour tout $n \in \mathbb{N}$ (même avec le 2^{n-1}).

6. (a) D'une part l'intégrale ci-dessus est positive : pour $0 \le t \le 1$ alors $0 \le 1 - t^2 \le 1$ et donc $0 \le (1 - t^2)^n \le 1$, ainsi la fonction sous l'intégrale est positive et donc $0 \le \pi - S_n$.

D'autre part on a pour tout $n \in \mathbb{N}$ et pour tout $t \in [0,1]$ les inégalités $0 \leqslant 1-t^2 \leqslant 1$ donc $0 \leqslant (1-t^2)^n \leqslant 1$, et $1+t^2 \geqslant 1$ donc $\frac{1}{1+t^2} \geqslant 1$. En multipliant ces inégalités positives on obtient

$$0 \leqslant \frac{(1-t^2)^n}{1+t^2} \leqslant 1\tag{54}$$

Intégrant cette inégalité alors

$$\int_{0}^{1} 0 \, \mathrm{d}t \leqslant \int_{0}^{1} \frac{(1 - t^{2})^{n}}{1 + t^{2}} \, \mathrm{d}t \leqslant \int_{0}^{1} 1 \, \mathrm{d}t \tag{55}$$

soit

$$0 \leqslant \int_0^1 \frac{(1-t^2)^n}{1+t^2} \, \mathrm{d}t \leqslant 1 \tag{56}$$

d'où en multipliant par $\frac{1}{2^{n-1}}, \boxed{0 \leqslant \pi - S_n \leqslant \frac{1}{2^{n-1}}}$

(b) D'une part le 0 est une constante, d'autre part $\lim_{n\to+\infty}\frac{1}{2^{n-1}}=0$. Par le théorème des gendarmes, on déduit $\lim_{n\to+\infty}\pi-S_n=0$ ce qui est la même chose que

$$\lim_{n \to +\infty} S_n = \pi$$
(57)

7. Poser $t = \varphi(\theta) = \cos(\theta)$. On prend pour bornes $\theta = \frac{\pi}{2}$, où $\cos(\theta) = 0$, et $\theta = 0$, où $\cos(\theta) = 1$. Alors il s'agit le la fonction φ qui est dérivable et à dérivée continue sur $[0, \frac{\pi}{2}]$ et $dt = -\sin(\theta) d\theta$. La formule de changement de variable donne alors

$$I_n = \int_{\frac{\pi}{2}}^0 (1 - \cos^2(\theta))^n \times (-\sin(\theta)) \, \mathrm{d}\theta = \int_0^{\frac{\pi}{2}} (1 - \cos^2(\theta))^n \times \sin(\theta) \, \mathrm{d}\theta \tag{58}$$

Attention à la rigueur : les bornes sont d'abord exactement dans le sens donné par le changement de variable, mais le signe moins sous l'intégrale permet de les intervertir et de donner un signe plus. Dans tous les cas, notre intégrale qui était positive au départ reste toujours positive à la fin.

Remarque 3. C'est toujours ce qui se passe pour un changement de variable donné par une fonction φ décroissante : les bornes risquent d'arriver avec la plus grande en bas et la plus petite en haut... Mais puisque φ est décroissante alors φ' est négative, et son signe moins permet de remettre les bornes dans le bon sens.

Enfin il suffit d'écrire $1 + \cos^2(\theta) = \sin(\theta)$, valable quelque soit $\theta \in \mathbb{R}$:

$$I_n = \int_0^{\frac{\pi}{2}} (\sin^2(\theta))^n \times \sin(\theta) d\theta = \left| \int_0^{\frac{\pi}{2}} \sin^{2n+1}(\theta) d\theta = I_n \right|$$
 (59)

Exercice

D'abord ré-écrivons bien aligné

$$\mathcal{S}_m: \left\{ \begin{array}{cccccc} (3-m)x & -2y & -6z & = & 2 & L_1 \\ 3x & -(2+m)y & -9z & = & 3 & L_2 \\ -x & +y & +(4-m)z & = & -1 & L_3 \end{array} \right. \tag{60}$$

Plaçons une ligne sympathique en haut :

$$\mathcal{S}_{m} \iff \begin{cases} x & -y & -(4-m)z = 1 & -L_{3} \\ 3x & -(2+m)y & -9z = 3 & L_{2} \\ (3-m)x & -2y & -6z = 2 & L_{1} \end{cases}$$
 (61)

On peut faire une étape pour éliminer les x:

$$\mathcal{S}_{m} \iff \begin{cases} x & -y & -(4-m)z = 1 & L_{1} \\ & (1-m)y & +(3-3m)z = 0 & L_{2}-3L_{1} \\ & (1-m)y & +\Big(-6+(3-m)(4-m)\Big)z = -1+m & L_{3}-(3-m)L_{1} \end{cases} \tag{62}$$

Calculons à part : $-6 + (3 - m)(4 - m) = -6 + 12 - 4m - 3m + m^2 = 6 - 7m + m^2$, ce qui se factorise aussi en (m - 6)(m - 1) :

$$\mathcal{S}_{m} \iff \begin{cases} x & -y & -(4-m)z = 1 & L_{1} \\ & (1-m)y & +(3-3m)z = 0 & L_{2} \\ & (1-m)y & +(m-6)(m-1)z = -1+m & L_{3} \end{cases}$$
 (63)

On voit apparaître une première séparation de cas intéressante car on veut simplifier par 1-m:

• Cas m=1: alors on ré-écrit tout

$$S_1 \iff \begin{cases} x & -y & -3z = 1 & L_1 \\ 0 & = 0 & L_2 \\ 0 & = 0 & L_3 \end{cases}$$
 (64)

Ce système est compatible, de rang 2, les variables y et z sont libres (on les renomme α et β). La première ligne est équivalente à x=1+y+3z. L'ensemble des solutions est

$$S_1 = \left\{ (1 + \alpha + 3\beta, \alpha, \beta) \mid (\alpha, \beta) \in \mathbb{R}^2 \right\}$$
 (65)

• Cas $m \neq 1$: on peut simplifier par $1 - m \neq 0$ et

$$\mathcal{S}_{m} \iff \begin{cases} x & -y & -(4-m)z = 1 & L_{1} \\ y & +3z = 0 & L_{2}/(1-m) \\ y & +(6-m)z = -1 & L_{3}/(1-m) \end{cases}$$
(66)

On peut alors poursuivre une étape d'échelonnement :

$$\mathcal{S}_{m} \iff \begin{cases} x & -y & -(4-m)z = 1 & L_{1} \\ y & +3z = 0 & L_{2} \\ & (3-m)z = -1 & L_{3} - L_{2} \end{cases}$$
 (67)

Ici le système est complètement échelonné, on doit alors encore distinguer des cas :

• Cas m=3: alors on ré-écrit

$$S_{3} \iff \begin{cases} x - y - z = 1 & L_{1} \\ y + 3z = 0 & L_{2} \\ 0 = -1 & L_{3} \end{cases}$$
 (68)

Le système est incompatible de rang 2 . L'ensemble des solutions est vide :

$$\boxed{S_3 = \emptyset} \tag{69}$$

• Cas $m \neq 3$: alors le système est de Cramer, il y a une unique solution. On résout en divisant dans L_3 par 3-m puis en remontant : L_3 donne $z=\frac{1}{m-3}$, puis L_2 donne $y=-3z=-\frac{3}{m-3}$, et enfin L_1 donne x=1+y+(4-m)z soit $x=1-\frac{3}{m-3}+\frac{4-m}{m-3}=\frac{(m-3)-3+(4-m)}{m-3}=-\frac{2}{m-3}$. Conclusion : si $m\notin\{1,3\}$ alors

$$S_m = \left\{ \left(-\frac{2}{m-3}, -\frac{3}{m-3}, \frac{1}{m-3} \right) \right\} \tag{70}$$