Partie 1 - Nombres complexes

LSINC1113 - Compléments de mathématiques

TP1 - Nombres complexes

Rappels théoriques

1. Nombres Complexes

Un nombre complexe est une extension des nombres réels, défini comme :

$$z = a + ib,$$

où a et b sont des nombres réels, et i est l'unité imaginaire, définie par $i^2 = -1$. On appelle $a = \Re(z)$ la **partie réelle** et $b = \Im(z)$ la **partie imaginaire**.

Chaque nombre complexe peut être représenté comme un point ou un vecteur dans le **plan d'Argand** (plan complexe), avec l'axe horizontal pour la partie réelle et l'axe vertical pour la partie imaginaire. Par exemple, $z = 3 + 4i$ correspond au point (3,4).

Module et argument. Le module d'un nombre complexe $z = a + ib$ est la distance à l'origine :

$$|z| = \sqrt{a^2 + b^2}.$$

L'argument de z, noté $\arg(z)$, est l'angle (en radians) entre le vecteur (a, b) et l'axe réel positif :

$$\arg(z) = \theta = \arctan\left(\frac{b}{a}\right).$$

Il est défini à $2k\pi$ près. L'argument principal est noté $\mathrm{Arg}(z) \in (-\pi, \pi]$.

Différentes formes. Un nombre complexe peut être écrit sous trois formes équivalentes :

$$\begin{aligned} \text{Forme cartésienne} &: z = a + ib \\ \text{Forme trigonométrique} &: z = r(\cos\theta + i\sin\theta) = r\,\mathrm{cis}(\theta) \\ \text{Forme exponentielle (Euler)} &: z = re^{i\theta} \end{aligned}$$

avec $a = r\cos\theta$ et $b = r\sin\theta$.

Puissances et racines. La formule de Moivre donne :

$$(re^{i\theta})^n = r^n e^{in\theta}.$$

Pour les racines n-ièmes :

$$\sqrt[n]{z} = \sqrt[n]{r}\; e^{i(\theta+2k\pi)/n}, \quad k = 0, 1, \ldots, n-1.$$

Exemple : les racines carrées de $i = e^{i\pi/2}$ sont $\sqrt{i} = e^{i\pi/4}$ et $e^{i5\pi/4}$.

Conjugaison. Le conjugué de $z = a + ib$ est

$$\overline{z} = a - ib,$$

et on a

$$z \cdot \overline{z} = |z|^2, \quad \frac{1}{z} = \frac{\overline{z}}{|z|^2} \quad (z \neq 0).$$

Géométriquement, $\overline{z}$ est le symétrique de z par rapport à l'axe réel.

Polynômes et racines complexes. Un polynôme de degré n à coefficients complexes s'écrit sous la forme :

$$P(z) = z^n + a_{n-1}z^{n-1} + \cdots + a_1 z + a_0, \quad a_i \in \mathbb{C}.$$

- **Théorème fondamental de l'algèbre :** tout polynôme non constant de degré n admet exactement n racines dans $\mathbb{C}$, comptées avec leur multiplicité.
- **Multiplicité :** si $P(z) = (z - \alpha)^k Q(z)$, alors α est une racine de multiplicité k.
- **Conjugaison des racines :** si les coefficients du polynôme sont réels, les racines non réelles apparaissent par paires conjuguées.
- **Factorisation :** tout polynôme de degré n à coefficients complexes peut se factoriser en produit de facteurs linéaires :

$$P(z) = (z - \alpha_1)(z - \alpha_2) \cdots (z - \alpha_n).$$

Exemple. Le polynôme $P(z) = z^2 + 1$ n'a pas de racine réelle. Dans $\mathbb{C}$, il se factorise en :

$$z^2 + 1 = (z - i)(z + i).$$

2. La matrice compagnon

La **matrice compagnon** est une représentation matricielle d'un polynôme, permettant d'étudier ses racines comme valeurs propres d'une matrice.

Pour un polynôme univarié de degré n :

$$P(z) = z^n + a_{n-1}z^{n-1} + a_{n-2}z^{n-2} + \cdots + a_1 z + a_0,$$

la matrice compagnon associée est :

$$C = \begin{pmatrix} 0 & 0 & 0 & \dots & 0 & -a_0 \\ 1 & 0 & 0 & \dots & 0 & -a_1 \\ 0 & 1 & 0 & \dots & 0 & -a_2 \\ 0 & 0 & 1 & \dots & 0 & -a_3 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & 1 & -a_{n-1} \end{pmatrix}.$$

Les valeurs propres de C sont exactement les racines de $P(z)$.

Exemple. Pour $P(z) = z^3 - 2z^2 + z - 5$, on obtient

$$C = \begin{pmatrix} 0 & 0 & 5 \\ 1 & 0 & -1 \\ 0 & 1 & 2 \end{pmatrix},$$

et les valeurs propres de C correspondent aux racines de P.

1. Conversion de formes

Exercice 1. Déterminer la partie réelle et la partie imaginaire des complexes suivants.

(a) $z = 2 + i6$
(b) $z = 45 - i\pi$
(c) $z = \dfrac{4i+2}{3}$
(d) $z = \dfrac{-12-7i}{7}$
(e) $z = 2\,\mathrm{cis}\,(\dfrac{\pi}{4})$
(f) $z = \dfrac{1}{2}\,\mathrm{cis}\,(\dfrac{\pi}{3})$
(g) $z = 5e^{-i\frac{\pi}{3}}$
(h) $z = e^{i\frac{\pi}{2}}$

2. Opérations sur les complexes

Exercice 2. Effectuer les opérations suivantes pour simplifier le nombre complexe au maximum. La réponse finale doit être dans la forme initiale de l'exercice : si vous effectuez une conversion durant l'exercice, revenez à la forme du début, une fois l'opération terminée.

(a) $z = (2-i) + (1+2i)$
(b) $z = (\sqrt{2}\,\mathrm{cis}\,(45°))(\sqrt{2}\,\mathrm{cis}\,(300°))$
(c) $z = \dfrac{12\,\mathrm{cis}\,(330°)}{4\,\mathrm{cis}\,(210°)}$
(d) $z = (3\,\mathrm{cis}\,(60°))^4$
(e) $z = (2-i)(1+2i)$
(f) $z = (2+i)^3$
(g) $z = 2\,\mathrm{cis}\,(\dfrac{\pi}{3}) + 4\,\mathrm{cis}\,(\dfrac{5\pi}{3})$
(h) $z = (\sqrt{3}+i)^3$

Exercice 3. Calculer le complexe conjugué des nombres de l'exercice 2.

3. Équations dans les complexes

Exercice 4. Résoudre les équations dans $\mathbb{C}$. Vous pouvez utiliser votre ordinateur mais pas le package `Polynomials.jl` (sauf les trois derniers).
Astuce : pour les degrés élevés, construisez la matrice compagnon et calculez ses valeurs propres.

(a) $(1+i)z = -2 + 5i$
(b) $\dfrac{1}{z} = \dfrac{i}{1+i}$
(c) $4 + z^2 = 0$
(d) $3z^2 - 4z + 2 = 0$
(e) $z^2 - 5iz = 6$
(f) $z^4 + 5z^2 + 4 = 0$
(g) $2z^2 + 5iz - 3 = 0$
(h) $z^4 - 6z^3 + 5iz^2 - 8z + 2i = 0$
(i) $2z^5 + 3iz^4 - 4z^3 + 5iz^2 - 6z + 7 = 0$
(j) $z^7 - iz^6 + 2z^5 - 5iz^4 + 6z^3 - 4iz^2 + z - i = 0$

4. Exercices théoriques

Exercice 5. Montrer que la multiplication complexe est *(i)* associative, *(ii)* commutative, *(iii)* distributive et *(iv)* absorbée par l'élément nul.

Exercice 6. Montrer que le conjugué du produit de deux complexes est le produit des conjugués.

5. Exercices supplémentaires

Exercice 7. Convertir les expressions suivantes dans les deux autres formes possibles.

(a) $z = 1 - i$
(b) $z = 1 + i$
(c) $z = \sqrt{3} + i$
(d) $z = 1 - i\sqrt{3}$
(e) $z = i$
(f) $z = \dfrac{-1}{2} + \dfrac{\sqrt{3}}{2}i$
(g) $z = 5\,\mathrm{cis}\,(\dfrac{\pi}{3})$
(h) $z = 2\,\mathrm{cis}\,(\dfrac{\pi}{6})$
(i) $z = e^{i\frac{\pi}{4}}$
(j) $z = 4e^{i\frac{5\pi}{3}}$

Exercice 8. Représenter les complexes de l'exercice 7 dans le plan complexe.

Exercice 9. Simplifier les expressions suivantes.

(a) $\dfrac{1}{1+i}$
(b) $\dfrac{i}{i-1}$
(c) $\dfrac{i+1}{2j}$
(d) $\dfrac{1+i}{1-i} + \dfrac{1-i}{1+i}$

Exercice 10. Montrer que $i^3 = \dfrac{1}{i}$.

Exercice 11. Pour chacun des polynômes suivants :

(a) $P_1(z) = z^3 - 1$
(b) $P_2(z) = z^4 + 1$
(c) $P_3(z) = z^3 - 2z + 2$
(d) $P_4(z) = 2z^5 + 3iz^4 - 4z^3 + 5iz^2 - 6z + 7$

Écrire explicitement la matrice compagnon C correspondante. Calculer numériquement les valeurs propres de C (sans `Polynomials.jl`) et les donner en forme exponentielle $re^{i\theta}$ si possible.

LSINC1113 - Compléments de mathématiques

Correction TP1 - Nombres complexes

---

1. Conversion de formes

Solution 1.

(a) $\Re(z) = 2 \quad \text{et} \quad \Im(z) = 6$

(b) $\Re(z) = 45 \quad \text{et} \quad \Im(z) = -\pi$

(c) $\Re(z) = \frac{2}{3} \quad \text{et} \quad \Im(z) = \frac{4}{3}$

(d) $\Re(z) = \frac{-12}{7} \quad \text{et} \quad \Im(z) = -1$

(e) $\Re(z) = \sqrt{2} \quad \text{et} \quad \Im(z) = \sqrt{2}$

(f) $\Re(z) = \frac{1}{4} \quad \text{et} \quad \Im(z) = \frac{\sqrt{3}}{4}$

(g) $\Re(z) = \frac{5}{2} \quad \text{et} \quad \Im(z) = -\frac{5\sqrt{3}}{2}$

(h) $\Re(z) = 0 \quad \text{et} \quad \Im(z) = 1$

2. Opérations sur les complexes

Solution 2.

(a) $z = 3 + i$

(b) $z = (\sqrt{2}\sqrt{2})(\text{cis}\,(45° + 300°))$
$= 2(\text{cis}\,(345°))$

(c) $z = 3\,\text{cis}\,(330° - 210°) = 3\,\text{cis}\,(120°)$

(d) $z = (3\,\text{cis}\,60°)^4$
$= (3)^4(\text{cis}\,(60°))^4$
$= 81\,\text{cis}\,(240°)$

(e) $z = 2 + 4i - i + 2 = 4 + 3i$

(f) $z = (4 + 4i - 1)(2 + i)$
$= (3+4i)(2+i) = 6+3i+8i-4 = 2+11i$

(g) $z = 2(\cos\frac{\pi}{3} + i\sin\frac{\pi}{3}) + 4(\cos\frac{5\pi}{3} + i\sin\frac{5\pi}{3})$
$= 2(\frac{1}{2} + i\frac{\sqrt{3}}{2}) + 4(\frac{1}{2} - i\frac{\sqrt{3}}{2}) = 3 - \sqrt{3}i$
$= 2\sqrt{3}\,\text{cis}\,\frac{-\pi}{6}$

(h) $z = (2 + 2\sqrt{3}i)(\sqrt{3} + i) = 8i$

Solution 3.

(a) $\overline{z} = 3 - i$

(b) $\overline{z} = 2\,\text{cis}\,(-345°)$

(c) $\overline{z} = 3(\cos 120° - i\sin 120°)$
$= 3(\cos -120° + i\sin -120°)$
$= 3\,\text{cis}\,(-120°)$

(d) $\overline{z} = 81\,\text{cis}\,(-240°)$

(e) $\overline{z} = 4 - 3i$

(f) $\overline{z} = 2 - 11i$

(g) $\overline{z} = 2\sqrt{3}\,\text{cis}\,\frac{\pi}{6}$

(h) $\overline{z} = -8i$

3. Équations dans les complexes

Solution 4.

(a) $(1 + i)z = -2 + 5i$
$z = \frac{(-2 + 5i)}{(1 + i)}\frac{(1 - i)}{(1 - i)}$
$z = \frac{7i + 3}{2}$

(b) $\frac{1}{z} = \frac{i}{1+i}$
$z = 1 - i$

(c) $4 + z^2 = 0$
$z = \pm\sqrt{-4} = \pm 2i$

(d) $3z^2 - 4z + 2 = 0$

$$z = \frac{-(-4) \pm \sqrt{-8}}{2(3)} = \frac{4 \pm \sqrt{8}i}{6} = \frac{2 \pm \sqrt{2}i}{3}$$

(e) $z^2 - 5iz - 6 = 0$

$$z = \frac{-(-5i) \pm \sqrt{-1}}{2(1)} = \frac{5i \pm i}{2}$$

$z = 3i \quad \text{ou} \quad z = 2i$

(f) $z^4 + 5z^2 + 4 = 0$

$u^2 + 5u + 4 = 0 \quad \text{avec} \quad u = z^2$

$$u = \frac{-5 \pm \sqrt{9}}{2} = \frac{-5 \pm 3}{2}$$

$u = -1 \quad \text{et} \quad u = -4$

$z^2 = -1 \quad \text{et} \quad z^2 = -4$

$z = \pm i \quad \text{et} \quad z = \pm 2i$

(g) $2z^2 + 5iz - 3 = 0$

$$z = \frac{-5i \pm i}{4}$$

$$z = -i \quad \text{et} \quad z = -\frac{3i}{2}$$

Pour les trois derniers, on utilise package Polynomials.jl :

(h) $z^4 - 6z^3 + 5iz^2 - 8z + 2i = 0$

```
roots_h, C = ComplexF64[-0.24516834438926072 + 1.4589624661717018im,
-0.04460154641931789 - 0.9408025929978315im, 0.00028237067868848144 +
0.22662796784914033im, 6.289487520129891 - 0.7447878410230104im]
```

(i) $2z^5 + 3iz^4 - 4z^3 + 5iz^2 - 6z + 7 = 0$

```
roots_h, C = ComplexF64[-1.6617801577835343 - 0.7552099765505914im,
-0.4226881817780331 - 1.1238772595174344im, -0.1996043866118091 +
1.1714751194360913im, 0.6661203004414142 + 0.22500303678199204im,
1.6179524257319624 - 1.0173909201500568im]
```

(j) $z^7 - iz^6 + 2z^5 - 5iz^4 + 6z^3 - 4iz^2 + z - i = 0$

```
roots_h, C = ComplexF64[-1.2122843374555718 + 1.5515061479247336im,
-0.5442538074761938 - 1.2510385596893394im, -0.23252932993398978 +
0.46018338432092526im, -5.936445475461463e-16 - 0.5213019451126377im,
0.23252932993398925 + 0.46018338432092526im, 0.5442538074761942 -
1.251038559689337im, 1.2122843374555687 + 1.5515061479247292im]
```

4. Exercices théoriques

Solution 5.

Associativité

Soient trois nombres complexes $z_1 = a_1 + b_1 i$, $z_2 = a_2 + b_2 i$, et $z_3 = a_3 + b_3 i$. Montrons que :

$$(z_1 \cdot z_2) \cdot z_3 = z_1 \cdot (z_2 \cdot z_3).$$

Calculons d'abord $(z_1 \cdot z_2) \cdot z_3$:

$$(z_1 \cdot z_2) \cdot z_3 = [(a_1 a_2 - b_1 b_2)a_3 - (a_1 b_2 + a_2 b_1)b_3 + ((a_1 b_2 + a_2 b_1)a_3 + (a_1 a_2 - b_1 b_2)b_3)i]$$

De même, calculons $z_1 \cdot (z_2 \cdot z_3)$:

$$z_1 \cdot (z_2 \cdot z_3) = [(a_2 a_3 - b_2 b_3)a_1 - (a_2 b_3 + a_3 b_2)b_1 + ((a_2 b_3 + a_3 b_2)a_1 + (a_2 a_3 - b_2 b_3)b_1)i]$$

Les deux résultats sont identiques, donc la multiplication complexe est associative.

Commutativité
Soient deux nombres complexes $z_1 = a_1 + b_1 i$ et $z_2 = a_2 + b_2 i$. Montrons que :

$$z_1 \cdot z_2 = z_2 \cdot z_1.$$

En développant les produits :

$$z_1 \cdot z_2 = (a_1 + b_1 i)(a_2 + b_2 i) = (a_1 a_2 - b_1 b_2) + (a_1 b_2 + a_2 b_1)i,$$

$$z_2 \cdot z_1 = (a_2 + b_2 i)(a_1 + b_1 i) = (a_2 a_1 - b_2 b_1) + (a_2 b_1 + a_1 b_2)i.$$

Comme les deux expressions sont égales, la multiplication complexe est commutative.

Distributivité
Soient trois nombres complexes $z_1 = a_1 + b_1 i$, $z_2 = a_2 + b_2 i$, et $z_3 = a_3 + b_3 i$. Montrons que :

$$z_1 \cdot (z_2 + z_3) = z_1 \cdot z_2 + z_1 \cdot z_3.$$

Calculons les deux côtés :

$$z_1 \cdot (z_2 + z_3) = (a_1 + b_1 i) \cdot [(a_2 + a_3) + (b_2 + b_3)i],$$

$$z_1 \cdot z_2 + z_1 \cdot z_3 = [(a_1 + b_1 i) \cdot (a_2 + b_2 i)] + [(a_1 + b_1 i) \cdot (a_3 + b_3 i)].$$

En développant et en simplifiant, on trouve que les deux côtés sont égaux, donc la multiplication complexe est distributive.

Absorption par l'élément nul
Montrons que pour tout nombre complexe $z = a + bi$, on a :

$$z \cdot 0 = 0.$$

Calculons :

$$z \cdot 0 = (a + bi) \cdot 0 = 0.$$

Donc, la multiplication complexe est absorbée par l'élément nul.

Solution 6.
On veut montrer que le conjugué du produit de deux complexes est le produit des conjugués.

$$\overline{z_1 \cdot z_2} = \overline{z_1} \cdot \overline{z_2}.$$

$$z_1 \cdot z_2 = a_1 a_2 - b_1 b_2 + (a_1 b_2 + a_2 b_1)i$$

Puis :

$$\overline{z_1 \cdot z_2} = a_1 a_2 - b_1 b_2 - (a_1 b_2 + a_2 b_1)i$$

Le produit des conjugués est donc :

$$\overline{z_1} \cdot \overline{z_2} = (a_1 - b_1 i)(a_2 - b_2 i) = a_1 a_2 - b_1 b_2 - (a_1 b_2 + a_2 b_1)i$$

5. Exercices Supplémentaires

Solution 7.

(a) $z = 1 - i$
$z = \sqrt{2}\,\mathrm{cis}\,(\frac{-\pi}{4}) = \sqrt{2}e^{-\frac{\pi i}{4}}$

(b) $z = 1 + i$
$z = \sqrt{2}\,\mathrm{cis}\,(\frac{\pi}{4}) = \sqrt{2}e^{\frac{\pi i}{4}}$

(c) $z = \sqrt{3} + i$
$z = 2\,\mathrm{cis}\,(\frac{\pi}{6}) = 2e^{\frac{\pi i}{6}}$

(d) $z = 1 - i\sqrt{3}$
$z = 2\,\mathrm{cis}\,(-\frac{\pi}{3}) = 2e^{-\frac{\pi i}{3}}$

(e) $z = i$
$z = \mathrm{cis}\,(\frac{\pi}{2}) = e^{\frac{\pi i}{2}}$

(f) $z = \frac{-1}{2} + \frac{\sqrt{3}}{2}i$
$z = \mathrm{cis}\,(\frac{-\pi}{3}) = e^{\frac{-\pi i}{3}}$

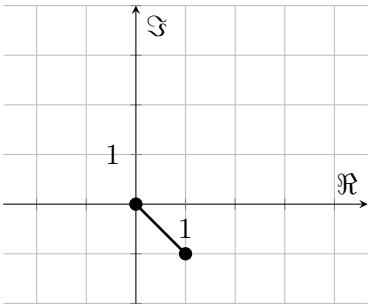
(g) $z = 5\,\mathrm{cis}\,(\frac{\pi}{3})$
$z = 5e^{\frac{\pi i}{3}} = \frac{5}{2} + \frac{5\sqrt{3}i}{2}$

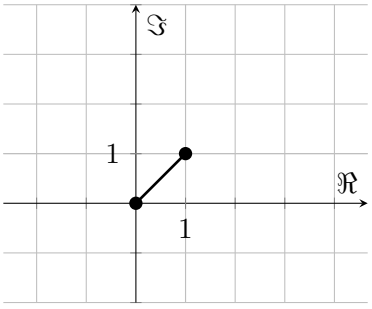
(h) $z = 2\,\mathrm{cis}\,(\frac{\pi}{6})$
$z = 2e^{\frac{\pi i}{6}} = \sqrt{3} + i$

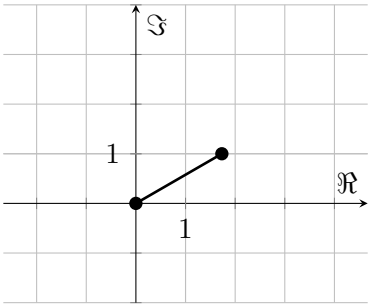
(i) $z = e^{i\frac{\pi}{4}}$
$z = \mathrm{cis}\,(\frac{\pi}{4}) = \frac{\sqrt{2}}{2}(1 + i)$

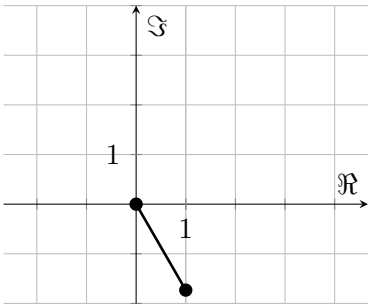
(j) $z = 4e^{i\frac{5\pi}{3}}$
$z = 4\,\mathrm{cis}\,(\frac{5\pi}{3}) = 2 - 2\sqrt{3}i$

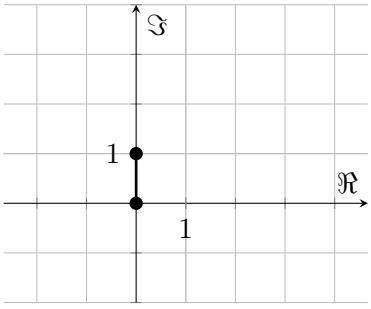
Solution 8.
Les représentations graphiques des différents nombres complexes sont :

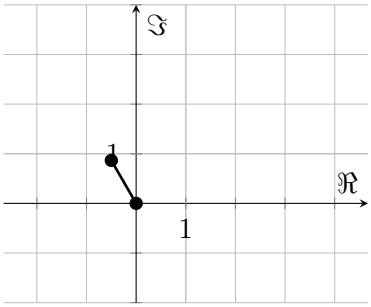
(a)


(b)


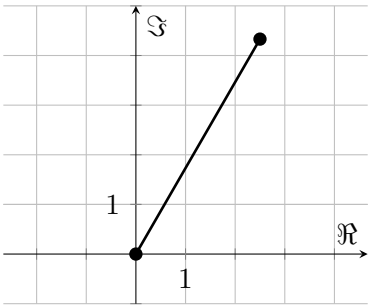
(c)


(d)


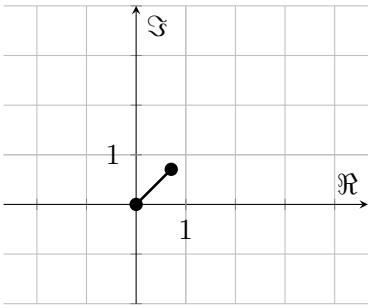
(e)


(f)


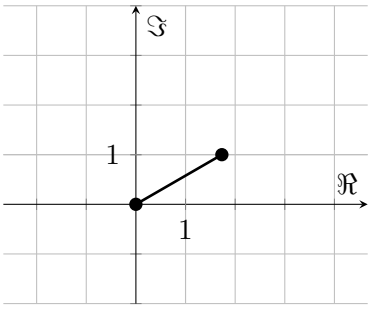
(g)



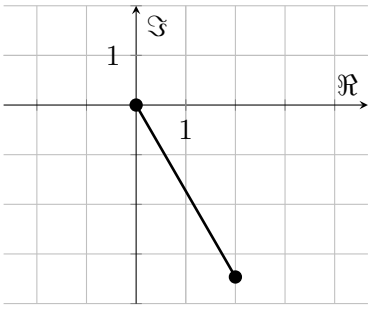
(i)



(h)



(j)



Solution 9.

(a) $\dfrac{1}{1+i} = \dfrac{1}{2}(1-i)$

(b) $\dfrac{i}{i-1} = -\dfrac{1}{2}(1-i)$

(c) $\dfrac{i+1}{2i} = \dfrac{1}{2}(1-i)$

(d) $\dfrac{1+i}{1-i} + \dfrac{1-i}{1+i} = 0$

Solution 10.
On veut montrer que $i^3 = \dfrac{1}{i}$.
$i^3 = i^2 i = -i$
$\frac{1}{i} = \frac{1}{i}\frac{-i}{-i} = -i$

Solution 11.

(a) $P_1(z) = z^3 - 1$
Matrice compagnon :

$$C = \begin{pmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}.$$

Les valeurs propres (racines) vérifient $z^3 - 1 = 0$:

$$z \in \{\, 1,\ e^{2i\pi/3},\ e^{4i\pi/3} \,\}.$$

(b) $P_2(z) = z^4 + 1$

$$C = \begin{pmatrix} 0 & 0 & 0 & -1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}.$$

On peut écrire $z^4 = -1$. Si on passe dans la forme trigonométrique on a, $-1 = \exp(i\pi)$. Mais comme les angles sont définis à 2π près, on peut aussi écrire : $-1 = \exp(i(\pi + 2k\pi)), \quad k \in \mathbb{Z}$. On a donc $z^4 = \exp(i(\pi + 2k\pi)) \Rightarrow z = \exp(i\frac{\pi+2k\pi}{4})$ et comme on a exactement 4 solutions distinctes, $k = \{0, 1, 2, 3\}$ Racines : $e^{i\pi/4},\ e^{3i\pi/4},\ e^{5i\pi/4},\ e^{7i\pi/4}$.

(c) $P_3(z) = z^3 - 2z + 2$
Matrice compagnon :

$$C = \begin{pmatrix} 0 & 0 & -2 \\ 1 & 0 & 2 \\ 0 & 1 & 0 \end{pmatrix}.$$

On doit repasser par du numérique pour trouver les racines :

$$z \approx -1.7693, \quad 0.8846 \pm 0.5897\, i.$$

(d) $P_4(z) = 2z^5 + 3iz^4 - 4z^3 + 5iz^2 - 6z + 7$
Astuce simple : on divise l'équation par 2 (ça ne change pas les racines) :

$$z^5 + \tfrac{3i}{2} z^4 - 2z^3 + \tfrac{5i}{2} z^2 - 3z + \tfrac{7}{2} = 0.$$

Matrice compagnon correspondante :

$$C = \begin{pmatrix} 0 & 0 & 0 & 0 & -\frac{7}{2} \\ 1 & 0 & 0 & 0 & 3 \\ 0 & 1 & 0 & 0 & -\frac{5i}{2} \\ 0 & 0 & 1 & 0 & 2 \\ 0 & 0 & 0 & 1 & -\frac{3i}{2} \end{pmatrix}.$$

On doit repasser par du numérique pour trouver les racines :

$$-1.662 - 0.755i, \quad -0.423 - 1.124i, \quad -0.200 + 1.171i,$$
$$0.666 + 0.225i, \quad 1.618 - 1.017i.$$

Partie 2 - Principe de récurrence et ordre de grandeur

LSINC1113 - Compléments de mathématiques

TP 2 - Principe de récurrence et ordre de grandeur

Rappels théoriques

1. Récurrences linéaires à coefficients constants

Homogène d'ordre r :

$$u_{n+r} + a_{r-1}u_{n+r-1} + \cdots + a_1 u_{n+1} + a_0 u_n = 0.$$

Polynôme caractéristique

$$p(\lambda) = \lambda^r + a_{r-1}\lambda^{r-1} + \cdots + a_1\lambda + a_0.$$

- Racines simples $\lambda_1, \ldots, \lambda_s$ (distinctes) :

$$u_n = \sum_{j=1}^{s} c_j\, \lambda_j^n.$$

- Racine λ de multiplicité m :

$$u_n = (c_0 + c_1 n + \cdots + c_{m-1} n^{m-1})\, \lambda^n.$$

Cas d'ordre 2 (utile en pratique). Pour $u_{n+2} + bu_{n+1} + cu_n = 0$:

$$\lambda^2 + b\lambda + c = 0, \quad \Delta = b^2 - 4c.$$

- $\Delta > 0$ (deux racines réelles $\lambda_1 \neq \lambda_2$) : $u_n = A\lambda_1^n + B\lambda_2^n$.
- $\Delta = 0$ (double racine λ) : $u_n = (A + Bn)\lambda^n$.
- $\Delta < 0$ (paires complexes $\lambda = \rho e^{\pm i\theta}$) :

$$u_n = \rho^n\big(A\cos(n\theta) + B\sin(n\theta)\big).$$

Non homogène : $u_{n+r} + a_{r-1}u_{n+r-1} + \cdots + a_0 u_n = f(n)$. La solution générale s'écrit $u_n = u_n^{(h)} + u_n^{(p)}$ (superposition). *Essais usuels pour* $u_n^{(p)}$:

Second membre $f(n)$	Essai pour $u_n^{(p)}$
$c\,\alpha^n$	$A\,\alpha^n$ (si α racine de p de mult. s, essayer $A\,n^s\alpha^n$)
polynôme en n de degré d	polynôme en n de degré d
$\alpha^n \times$ polynôme en n	idem, multiplié par α^n

(*Règle de résonance* : multiplier par n^s si l'essai "entre en résonance" avec une racine de multiplicité s de p.)

2. Formulation matricielle et valeurs propres

Poser $X_n = \begin{pmatrix} u_{n+r-1} \\ u_{n+r-2} \\ \vdots \\ u_n \end{pmatrix}$. Alors

$$X_{n+1} = A\,X_n + B(n),$$

où la *matrice compagnon* de la partie homogène est

$$A = \begin{pmatrix} -a_{r-1} & -a_{r-2} & \cdots & -a_1 & -a_0 \\ 1 & 0 & \cdots & 0 & 0 \\ 0 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & 0 \end{pmatrix}.$$

Si $B \equiv 0$, on a $X_n = A^n X_0$. Les valeurs propres de A sont exactement les racines de p. Si A est diagonalisable ($A = SDS^{-1}$), alors $A^n = SD^nS^{-1}$. En cas non diagonalisable (blocs de Jordan), les facteurs $n^k\lambda^n$ réapparaissent, cohérents avec la multiplicité dans la partie "homogène".

3. Exponentiation rapide (fast powering)

Pour calculer a^n ou A^n en $O(\log n)$:

$$\text{si } n \text{ pair} : a^n = (a^2)^{n/2}, \qquad \text{si } n \text{ impair} : a^n = a \cdot a^{n-1}.$$

Même principe pour les matrices (remplacer 1 par l'identité I). *Intérêt* : accéder à u_n via $X_n = A^n X_0$ même pour des n très grands.

4. Exemple clé : suite de Fibonacci

La suite de Fibonacci (F_n) est définie par $F_{n+1} = F_n + F_{n-1}$, $F_0 = 0$, $F_1 = 1$. Forme matricielle :

$$\begin{pmatrix} F_{n+1} \\ F_n \end{pmatrix} = \underbrace{\begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}^n}_{M} \begin{pmatrix} 1 \\ 0 \end{pmatrix}.$$

Le polynôme caractéristique de M est $\lambda^2 - \lambda - 1$, de racines $\varphi = \frac{1+\sqrt{5}}{2}$, $\psi = \frac{1-\sqrt{5}}{2}$.

1. Equation caractéristique et superposition

Exercice 12. En utilisant la méthode de l'équation caractéristique et le principe de superposition, résoudre les quatre récurrences suivantes :

1. $x_{n+2} + 3 \cdot x_{n+1} + 2 \cdot x_n = 5 \cdot 3^n$, avec $x_0 = 0, x_1 = 1$
2. $y_{n+2} - 4 \cdot y_{n+1} + 4 \cdot y_n = 1$, avec $y_0 = 0, y_1 = 3$
3. $z_n + 6 \cdot z_{n-1} + 9 \cdot z_{n-2} = 16 \cdot n$, avec $z_0 = 2, z_1 = 2$
4. $v_{n+2} = 2(v_{n+1} - v_n)$, avec $v_0 = 1, v_1 = 2$

Exercice 13. Pour quelles valeurs de x_0 la récurrence du premier ordre $x_n \cdot x_{n+1} + 15 = 0$, avec $n \in \mathbb{N}$, possède-t-elle une solution en nombres entiers ? Quelle est cette solution ?

Exercice 14. On s'intéresse aux mots de longueur n sur l'alphabet $\{a, b, c, d\}$. Un mot M de ce type sera dit $d-pair$ si la lettre d apparaît un nombre pair de fois dans M. Voici trois exemples de mots $d-pairs$ de longueur quatre : $abca, bdad, dddd$. Combien y a-t-il de mot $d-pairs$ de longueur n sur l'alphabet considéré ?
Suggestion : D'abord, établir une relation de récurrence entre les nombres de mots $d-pairs$ de deux longueurs successives, n et $n+1$. Ensuite, résoudre la récurrence obtenue.

Exercice 15. Démontrer par récurrence que pour tout entier $n \geq 1$, on a

$$S_n = \sum_{k=1}^{n} k^2 = 1^2 + 2^2 + \cdots + n^2 = \frac{n(n+1)(2n+1)}{6}$$

2. Méthode matricielle & exponentiation rapide

Exercice 16. On considère la suite

$$v_{n+2} = 2\,(v_{n+1} - v_n), \qquad v_0 = 1,\ v_1 = 2.$$

1. Écrire une fonction `pow_fast(a,n)` qui calcule a^n en $O(\log n)$.
2. Écrire `mat_pow_fast(A,n)` qui calcule A^n en $O(\log n)$ (avec I_2 comme neutre).
3. Poser $X_n = \binom{v_{n+1}}{v_n}$ et
$$M = \begin{pmatrix} 2 & -2 \\ 1 & 0 \end{pmatrix}.$$
Montrer que $X_{n+1} = MX_n$, donc $X_n = M^n X_0$ avec $X_0 = \binom{2}{1}$, et en déduire
$$v_n = \left(M^n \tbinom{2}{1}\right)_2.$$
4. Coder `v_of(n)` qui renvoie v_n via `mat_pow_fast`. Tester $v_0, \ldots, v_8$.
5. Calculer v_{10^6} mod $(10^9 + 7)$ et comparer le temps avec une méthode naïve (itération de la récurrence n fois).

LSINC1113 - Compléments de mathématiques

Correction TP2 - Principe de récurrence et ordre de grandeur

---

1. Equation caractéristique et superposition

Solution 12.
Méthode : on résout d'abord la partie homogène (polynôme caractéristique), puis on cherche une solution particulière adaptée au second membre, et on détermine les constantes par les conditions initiales.

1. $x_{n+2} + 3 \cdot x_{n+1} + 2 \cdot x_n = 5 \cdot 3^n, \quad x_0 = 0,\ x_1 = 1.$

   Homogène : $r^2 + 3r + 2 = 0 \ \Rightarrow \ (r+1)(r+2) = 0$
   $\Rightarrow$ les racines sont donc $r = -1, -2$.
   Cela donne comme solution homogène : $x_n^{(h)} = A(-1)^n + B(-2)^n$.

   Particulière : $5 \cdot 3^n$.
   Il n'y a pas de résonance car 3 n'est pas racine de la solution homogène.
   $\Rightarrow$ On essaye $x_n^{(p)} = A\,3^n$.
   On remplace dans l'équation de récurrence : $A\,3^{n+2} + 3 \cdot A\,3^{n+1} + 2 \cdot A\,3^n = 5 \cdot 3^n$
   On peut factoriser 3^n :
   $3^n(9A + 9A + 2A) = 20A \cdot 3^n = 5 \cdot 3^n \Rightarrow A = \frac{1}{4}$.

   Générale : $x_n = A(-1)^n + B(-2)^n + \dfrac{1}{4}\,3^n$.
   CI :
   $x_0 = 0 \Rightarrow A + B + \frac{1}{4} = 0$;
   $x_1 = 1 \Rightarrow -A - 2B + \frac{3}{4} = 1$
   $\Rightarrow -A - 2B = \frac{1}{4}$. On en déduit $B = 0$, $A = -\frac{1}{4}$.

   $$\boxed{x_n = \frac{1}{4}(3^n - (-1)^n)}$$

2. $y_{n+2} - 4y_{n+1} + 4y_n = 1, \quad y_0 = 0,\ y_1 = 3.$

   Homogène : $r^2 - 4r + 4 = (r-2)^2 = 0$.
   On a donc une racine double, $r = 2$.
   Cela donne comme solution homogène $y_n^{(h)} = (A + Bn)\,2^n$.

   Particulière : Le second membre est constant 1, on essaye $y_n^{(p)} = K$.
   On remplace dans l'équation de récurrence : $K - 4K + 4K = K = 1 \Rightarrow K = 1$. Il n'y a pas de résonance car 1 n'est pas solution de l'équation homogène).

   Générale : $y_n = (A + Bn)\,2^n + 1$.

CI :
$y_0 = 0 \Rightarrow A + 1 = 0 \Rightarrow A = -1.$
$y_1 = 3 \Rightarrow 2(A + B) + 1 = 3 \Rightarrow A + B = 1 \Rightarrow B = 2.$

$$\boxed{y_n = (2n - 1)\, 2^n + 1}$$

3. $z_n + 6z_{n-1} + 9z_{n-2} = 16\, n, \quad z_0 = 2,\ z_1 = 2.$

Homogène : On pose $z_n = r^n$
$r^n + 6r^{n-1} + 9r^{n-2} = 0 \Rightarrow r^{n-2}(r^2 + 6r + 9) = 0 \Rightarrow (r + 3)^2 = 0$
On a donc une racine double -3.
$z_n^{(h)} = (A + Bn)\,(-3)^n$

Particulière : $16n$
Le second membre est un polynôme de degré 1, on essaye $z_n^{(p)} = an + b$.
Alors,

$$z_{n-1}^{(p)} = a(n - 1) + b = an + (b - a)$$
$$z_{n-2}^{(p)} = a(n - 2) + b = an + (b - 2a)$$

Substitution dans l'équation :

$$(an + b) + 6(an + b - a) + 9(an + b - 2a) = 16n$$
$$16an + 16b - 24a = 16n$$

$\Rightarrow a = 1$ et $16b - 24 = 0 \Rightarrow b = \frac{3}{2}$.

Générale : $z_n = (A + Bn)\,(-3)^n + (n + \frac{3}{2})$.
CI :
$z_0 = 2 \Rightarrow A + \frac{3}{2} = 2 \Rightarrow A = \frac{1}{2}$.
$z_1 = 2 \Rightarrow -3(A + B) + \frac{5}{2} = 2 \Rightarrow A + B = \frac{1}{6} \Rightarrow B = -\frac{1}{3}$.

$$\boxed{z_n = \left(\tfrac{1}{2} - \tfrac{n}{3}\right)(-3)^n + n + \tfrac{3}{2}}$$

4. $v_{n+2} = 2\,(v_{n+1} - v_n), \quad v_0 = 1,\ v_1 = 2.$

Homogène : $v_{n+2} - 2v_{n+1} + 2v_n = 0 \Rightarrow r^2 - 2r + 2 = 0.$

$$\Delta = -4 = i^2 4$$
$$r = \frac{2 \pm 2i}{2} = 1 \pm i$$

Pour exploiter ces racines, on écrit :

$$1 + i = \sqrt{2} \cdot \exp(\frac{i\pi}{4}), \quad 1 - i = \sqrt{2} \cdot \exp(\frac{-i\pi}{4})$$

Cela signifie que les solutions générales sont des combinaisons linéaires de

$$(1 + i)^n = (\sqrt{2})^n \cdot \exp(\frac{in\pi}{4}), \quad (1 - i)^n = (\sqrt{2})^n \cdot \exp(\frac{-in\pi}{4})$$

En développant Euler,

$$\alpha(\sqrt{2})^n(\cos(n\pi/4) + i\sin(n\pi/4)) + \beta(\sqrt{2})^n(\cos(n\pi/4) - i\sin(n\pi/4))$$

$$\Rightarrow (\sqrt{2})^n(\alpha+\beta)(\cos(n\pi/4) + (\sqrt{2})^n(\alpha-\beta)i\sin(n\pi/4))$$

Comme la suite doit être réelle,

$$(\alpha+\beta) \text{ doit être réel et } i(\alpha-\beta) \text{ doit être réelle aussi}$$

Soit $\alpha = a + ib$ et $\beta = u + iv$: Donc la partie imaginaire de $(\alpha+\beta)$ vaut 0 donc $b+v=0$ et la partie imaginaire de $i(\alpha-\beta)$ vaut 0 donc $a-u=0$. On tombe donc sur α est le conjugué de β ce qui nous donne finalement, comme forme réelle :

$$v_n = (\sqrt{2})^n(C\cos(n\pi/4) + D\sin(n\pi/4)).$$

CI : $v_0 = 1 \Rightarrow C = 1$.
$v_1 = 2 \Rightarrow (\sqrt{2})(\frac{\sqrt{2}}{2}C + \frac{\sqrt{2}}{2}D) = C + D = 2 \Rightarrow D = 1$.

$$\boxed{v_n = (\sqrt{2})^n(\cos(n\pi/4) + \sin(n\pi/4))}$$

Solution 13. Pour que $x_{n+1} = -\dfrac{15}{x_n}$ soit un **entier**, il faut et il suffit que $\dfrac{15}{x_n} \in \mathbb{Z}$, c'est-à-dire

$$x_n \mid 15 \quad (\text{“}x_n \text{ divise } 15\text{”}).$$

Donc chaque terme x_n doit être un diviseur entier non nul de 15 :

$$\mathcal{D} = \{\pm 1,\ \pm 3,\ \pm 5,\ \pm 15\}.$$

En particulier, c'est nécessaire que $x_0 \in \mathcal{D}$ (et bien sûr $x_0 \neq 0$ pour éviter la division par zéro). Réciproquement, si on choisit $x_0 = d \in \mathcal{D}$, alors

$$x_1 = -\frac{15}{d} \in \mathcal{D}, \qquad x_2 = -\frac{15}{x_1} = -\frac{15}{-15/d} = d = x_0.$$

Donc

$$\boxed{x_{n+2} = x_n \ \text{ pour tout } n,}$$

la suite est périodique de période 2 (un 2-cycle) et reste entière.
Forme explicite (pour $d = x_0 \in \mathcal{D}$) :

$$\boxed{x_{2k} = d, \qquad x_{2k+1} = -\frac{15}{d} \quad (k \geq 0).}$$

Conclusion :
La récurrence admet une solution entière si et seulement si

$$\boxed{x_0 \in \{\pm 1,\ \pm 3,\ \pm 5,\ \pm 15\}\,.}$$

Dans ce cas, la suite est périodique de période 2 :

$$x_0 = d,\ x_1 = -\frac{15}{d},\ x_2 = d,\ x_3 = -\frac{15}{d},\ \ldots$$

Exemples :

1. $x_0 = 3 \Rightarrow (3, -5, 3, -5, \ldots)$
2. $x_0 = -15 \Rightarrow (-15, 1, -15, 1, \ldots)$

3. $x_0 = 5 \Rightarrow (5, -3, 5, -3, \dots)$

(Cas interdit : $x_0 = 0$, car on ne peut pas définir x_1).

Solution 14.
Soit :

- E_n = nombre de mots de longueur n avec un **nombre pair** de d,
- O_n = nombre de mots de longueur n avec un **nombre impair** de d.

Bases : $E_0 = 1$ (le mot vide a 0 d, pair) et $O_0 = 0$.

Construisons un mot de longueur $n+1$ en ajoutant une lettre à droite d'un mot de longueur n. Si on ajoute a, b ou c (3 choix), on ne change pas la parité du nombre de d. Si on ajoute d (1 choix), on bascule la parité, un mot avec un nombre impair de d devient pair et vice versa.

Donc :

- Pour être pair à l'étape $n+1$, on vient soit d'un mot pair + $a/b/c$ ($3 \cdot E_n$), soit d'un mot impair + d ($1 \cdot O_n$) :
$$E_{n+1} = 3E_n + O_n.$$
- Pour être impair à l'étape $n+1$, on vient soit d'un mot impair + $a/b/c$ ($3 \cdot O_n$), soit d'un mot pair + d ($1 \cdot E_n$) :
$$O_{n+1} = E_n + 3O_n.$$

Sous forme matricielle :

$$\begin{pmatrix} E_{n+1} \\ O_{n+1} \end{pmatrix} = \underbrace{\begin{pmatrix} 3 & 1 \\ 1 & 3 \end{pmatrix}}_{T} \begin{pmatrix} E_n \\ O_n \end{pmatrix}, \qquad \begin{pmatrix} E_0 \\ O_0 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix}.$$

D'où $\binom{E_n}{O_n} = T^n \binom{1}{0}$.

On va résoudre la récurrence, via diagonalisation. La matrice T a pour valeurs propres ($\det(T - \lambda I) = 0$) $\lambda_1 = 4$ et $\lambda_2 = 2$, avec vecteurs propres $(T - \lambda I)v = 0$ $v_1 = \binom{1}{1}$ et $v_2 = \binom{1}{-1}$. On décompose le vecteur initial. Pour arriver à ça : on cherche α, β tels que

$$\begin{pmatrix} 1 \\ 0 \end{pmatrix} = \alpha \begin{pmatrix} 1 \\ 1 \end{pmatrix} + \beta \begin{pmatrix} 1 \\ -1 \end{pmatrix}.$$

Cela donne le système

$$\begin{cases} \alpha + \beta = 1, \\ \alpha - \beta = 0. \end{cases} \qquad \Longrightarrow \quad \alpha = \beta = \tfrac{1}{2}.$$

Donc

$$\begin{pmatrix} 1 \\ 0 \end{pmatrix} = \tfrac{1}{2} \begin{pmatrix} 1 \\ 1 \end{pmatrix} + \tfrac{1}{2} \begin{pmatrix} 1 \\ -1 \end{pmatrix}.$$

Alors

$$\begin{pmatrix} E_n \\ O_n \end{pmatrix} = \tfrac{1}{2}\, 4^n \begin{pmatrix} 1 \\ 1 \end{pmatrix} + \tfrac{1}{2}\, 2^n \begin{pmatrix} 1 \\ -1 \end{pmatrix} = \begin{pmatrix} \frac{4^n + 2^n}{2} \\ \frac{4^n - 2^n}{2} \end{pmatrix}.$$

Finalement,

$$\boxed{E_n = \frac{4^n + 2^n}{2}} \quad \text{et} \quad O_n = \frac{4^n - 2^n}{2}.$$

Solution 15.
On veut montrer, pour tout entier $n \geq 1$,

$$S_n = \sum_{k=1}^{n} k^2 = \frac{n(n+1)(2n+1)}{6}.$$

1) Initialisation
Pour $n = 1$,

$$S_1 = 1^2 = 1 \quad \text{et} \quad \frac{1 \cdot (1+1) \cdot (2 \cdot 1 + 1)}{6} = \frac{1 \cdot 2 \cdot 3}{6} = 1.$$

Donc la propriété est vraie au rang 1.

2) Hérédité
Supposons la formule vraie pour un certain $n \geq 1$ (hypothèse de récurrence) :

$$S_n = \sum_{k=1}^{n} k^2 = \frac{n(n+1)(2n+1)}{6}.$$

Montrons-la au rang $n+1$. On part de

$$S_{n+1} = \sum_{k=1}^{n+1} k^2 = \underbrace{\sum_{k=1}^{n} k^2}_{S_n} + (n+1)^2.$$

En remplaçant S_n par l'hypothèse de récurrence :

$$S_{n+1} = \frac{n(n+1)(2n+1)}{6} + (n+1)^2.$$

Factorisons $(n+1)$:

$$S_{n+1} = (n+1)\left(\frac{n(2n+1)}{6} + (n+1)\right).$$

Mettons tout au même dénominateur :

$$S_{n+1} = (n+1)\left(\frac{2n^2+n}{6} + \frac{6n+6}{6}\right) = (n+1) \cdot \frac{2n^2+7n+6}{6}.$$

Factorisons le polynôme :

$$2n^2 + 7n + 6 = (2n+3)(n+2).$$

Donc

$$S_{n+1} = (n+1) \cdot \frac{(2n+3)(n+2)}{6} = \frac{(n+1)(n+2)(2n+3)}{6}.$$

Or

$$\frac{(n+1)(n+2)(2(n+1)+1)}{6} = \frac{(n+1)(n+2)(2n+3)}{6},$$

ce qui est exactement la formule au rang $n+1$.

3) Conclusion
La propriété est vraie au rang 1 et héréditaire de n à $n+1$. Par principe de récurrence, pour tout $n \geq 1$,

$$\boxed{\sum_{k=1}^{n} k^2 = \frac{n(n+1)(2n+1)}{6}}$$

est démontré.

Solution 16.

1. Pour calculer, a^n, on va vouloir faire $a \cdot a \cdot \cdot a$ ($n-1$ multiplications en $O(n)$) mais plus n devient grand plus cela va devenir long. Un algorithme d'exponentiation rapide (aussi appelé exponentiation binaire) sert à calculer a^n beaucoup plus vite que la méthode naïve. L'idée c'est d'exploiter l'écriture binaire de n pour ne plus faire qu'$O(\log n)$ multiplications (au lieu de $O(n)$).
   De plus, le fait que diviser l'exposant par 2 est moins coûteux :
   - si n est pair : $a^n = (a^2)^{n/2}$;
   - si n est impair : $a^n = a \cdot (a^2)^{(n-1)/2}$.

   n étant binaire, on peut l'écrire comme $n = \sum_{k=0}^{m} b_k 2^k$ avec $b_k \in \{0, 1\}$. Alors,

   $$a^n = \prod_k a^{b_k 2^k} = \prod_{k|b_k=1} a^{2^k}$$

   Or cela revient juste à mettre k fois la base au carré seulement quand le bit vaut 1.

   Ecrivons :

   $$\texttt{res} \cdot \texttt{base}^{\texttt{exp}} = a^n \quad \text{(invariant)}.$$

   Au départ $\texttt{res} = 1$, $\texttt{base} = a$, $\texttt{exp} = n$.
   À chaque tour :
   - si exp est impair : $\texttt{res} \leftarrow \texttt{res} \cdot \texttt{base}$; si exp est pair : $\texttt{res} \leftarrow \texttt{res}$
   - on met *toujours* la base au carré : $\texttt{base} \leftarrow \texttt{base}^2$;
   - on décale exp d'un bit : $\texttt{exp} \leftarrow \lfloor \texttt{exp}/2 \rfloor$ (division entière).

   Par exemple 3^{13} :
   (a) $\texttt{res} = 1$; $\texttt{base} = 3$; $\texttt{exp} = 13$
   (b) exp est impair ; $\texttt{res} = 1 * 3 = 3$;$\texttt{base} = 3^2 = 9$; $\texttt{exp} = 6$
   (c) exp est pair ; $\texttt{res} = 3$;$\texttt{base} = 9^2 = 81$; $\texttt{exp} = 3$
   (d) exp est impair ; $\texttt{res} = 3 * 81 = 243$;$\texttt{base} = 81^2 = 6561$; $\texttt{exp} = 1$
   (e) exp est impair ; $\texttt{res} = 6561 * 243 = 1594323$;$\texttt{base} = 6561^2 = 43046721$; $\texttt{exp} = 0$
   On a donc $3^{13} = 1594323$.

```
function pow_fast(a, n::Integer)
    n < 0 && throw(ArgumentError("n => 0 requis"))
    res, base, exp = one(a), a, n
    while exp > 0
        if isodd(exp)
            res *= base
        end
        base *= base
        exp >>= 1
    end
    return res
end

```

2. C'est exactement la même idée mais on remplace 1 par l'identité I_2 et les multiplications deviennent par le produit matriciel. De plus, on calcule le modulo m pour éviter l'explosion de la taille des nombres (qui croissent exponentiellement), garder tous les calculs bornés dans $[0, m-1]$ donc faisables, et comme la réduction est un homomorphisme $((AB) \bmod m = ((A \bmod m)(B \bmod m)) \bmod m)$, on obtient le **même résidu** que si l'on réduisait à la fin.

   Le code devient :

```
function mul2(A, B; m::Integer=0)
    C11 = A[1,1]*B[1,1] + A[1,2]*B[2,1]
    C12 = A[1,1]*B[1,2] + A[1,2]*B[2,2]
    C21 = A[2,1]*B[1,1] + A[2,2]*B[2,1]
    C22 = A[2,1]*B[1,2] + A[2,2]*B[2,2]
    if m != 0
        C11 %= m; C12 %= m; C21 %= m; C22 %= m
    end
    return [C11 C12; C21 C22]
end

function mat_pow_fast(A, n::Integer; m::Integer=0)
    n < 0 && throw(ArgumentError("n => 0 requis"))
    R = [1 0; 0 1]
    B, e = copy(A), n
    while e > 0
        if isodd(e)
            R = mul2(R, B; m=m)
        end
        B = mul2(B, B; m=m)
        e >>= 1
    end
    return R
end

```

3. On l'applique maintenant à la récurrence $v_{n+2} = 2(v_{n+1} - v_n)$. Traduisons cela en produit de matrices :

$$\begin{pmatrix} v_{n+2} \\ v_{n+1} \end{pmatrix} = \underbrace{\begin{pmatrix} 2 & -2 \\ 1 & 0 \end{pmatrix}}_{M} \begin{pmatrix} v_{n+1} \\ v_n \end{pmatrix}$$

   Par itération :

$$X_n = M^n X_0, \quad X_n = \begin{pmatrix} v_{n+1} \\ v_n \end{pmatrix} \quad X_0 = \begin{pmatrix} v_1 \\ v_0 \end{pmatrix} = \begin{pmatrix} 2 \\ 1 \end{pmatrix}.$$

   Donc,

$$v_n = (M^n \begin{pmatrix} 2 \\ 1 \end{pmatrix})_2$$

   Le polynôme caractéristique de la partie homogène est $\lambda^2 - 2\lambda + 2 = 0$ a comme racines $1 \pm i = \sqrt{2}\exp(\pm i\pi/4)$ (cf. exercice 12).
   Donc,

$$v_n = (\sqrt{2})^n(\cos(\frac{n\pi}{4}) + \sin(\frac{n\pi}{4})).$$

```
        function v_of(n::Integer; m::Integer=0)
            n < 0 && throw(ArgumentError("n => 0 requis"))
            if n == 0
                return m == 0 ? 1 : 1 % m)
            end
            M = [2 -2; 1 0]
            V = mat_pow_fast(M, n; m=m) * [2; 1]
            return m == 0 ? V[2] : Int(V[2] % m)
        end

```

Partie 3 - Théorie des nombres

LSINC1113 - Compléments de mathématiques

TP3 - Théorie des nombres

Rappels Théoriques

L'**arithmétique modulaire** est la manière de raisonner sur les entiers "à reste près". On y utilise les notions de divisibilité, de reste et d'inverse modulaire.

Ces outils sont fondamentaux pour construire des algorithmes efficaces en informatique et en cryptographie (par exemple pour le chiffrement ou les calculs rapides sur les ordinateurs). Autrement dit, on ne s'intéresse pas aux valeurs exactes, mais au reste obtenu après division.

Par exemple : $17 \equiv 2 \pmod 5$ car $17 = 5 * 3 + 2 = 15 + 2$ et 15 est divisible par 5 donc on sait le "retirer" du calcul et il reste 2.

0. Le modulo

Le modulo, c'est une façon de dire : "je ne m'intéresse qu'au reste de la division euclidienne."

Quand on écrit

$$a \equiv b \pmod m$$

on lit : a et b laissent le même reste quand on les divise par m. On peut aussi dire m divise $(a-b)$.

Exemple : Si m = 5. $7 \equiv 12 \equiv 17 \equiv 22 \pmod 5$ car leur division par 5 donne le même reste, 2.

1. Division Euclidienne

Pour deux entiers a (dividende) et d (diviseur, $d \neq 0$), il existe des entiers q (quotient) et r (reste) tels que :

$$a = dq + r \quad \text{avec } 0 \leq r < |d|.$$

- Le quotient q est donné par $q = \lfloor a/d \rfloor$
- Le reste r est obtenu par $r = a - dq$

En notation modulaire $a \equiv r \pmod d$.

2. Théorème de Bézout

Bézout dit qu'on peut toujours écrire le PGCD (le plus grand commun diviseur) de deux nombres a et b sous la forme d'une combinaison linéaire :

$$ax + by = \text{pgcd}(a, b).$$

Cette relation est essentielle car elle relie arithmétique et algorithme :

- Elle montre que le PGCD n'est pas qu'un nombre, mais qu'il est "fabriqué" à partir de a et b.
- Elle sert surtout à démontrer l'existence d'inverses modulaires lorsque le PGCD vaut 1 (condition à l'existence de l'inverse modulaire).

En notation modulaire,

$$ax \equiv \text{pgcd}(a,b) \pmod{b} \quad \text{et} \quad by \equiv \text{pgcd}(a,b) \pmod{a}$$

Lemme :

$$\text{Si } a \equiv r \pmod{b} \text{ alors } \text{pgcd}(a,b) = \text{pgcd}(b,r)$$

Arithmétique modulaire : somme.

$$a \equiv \alpha \pmod{n} \quad b \equiv \beta \pmod{n} \quad \Rightarrow \quad a+b \equiv \alpha+\beta \pmod{n}$$

$$a \equiv b \pmod{n} \quad a+k \equiv b+k \pmod{n}$$

Arithmétique modulaire : produit.

$$a \equiv \alpha \pmod{n} \quad \text{et} \quad b \equiv \beta \pmod{n} \quad \Rightarrow \quad ab \equiv \alpha\beta \pmod{n}$$

3. Algorithme d'Euclide

L'algorithme d'Euclide sert à trouver le PGCD entre deux nombres efficacement. On peut également le dire comme : quel est le plus grand nombre qui divise à la fois a et d.

On va appliquer la division euclidienne à répétition :

1. Diviser a par d, et noter le reste r.
2. Remplacer a par d, et d par r.
3. Répéter jusqu'à ce que $r = 0$.

Le dernier reste non nul est le $\text{pgcd}(a,d)$.

Par exemple :

$$\begin{aligned}
&pgcd(252,105) \quad ? \\
&252 = 2*105+42 \quad \text{(le reste)} \\
&pgcd(252,105) = pgcd(105,42) \\
&105 = 2*42+21 \quad \text{(le reste)} \\
&pgcd(252,105) = pgcd(105,42) = pgcd(42,21) \\
&42 = 2*21+0
\end{aligned}$$

Le PGCD est le dernier reste non nul : ici 21.

Au lieu de tester tous les diviseurs possibles, on réduit le problème à des restes de plus en plus petits. Le PGCD est utile pour savoir si deux nombres sont premiers entre eux, ce qui conditionne la possibilité d'inverser un nombre modulo un autre.

4. Euclide étendu

L'algorithme d'Euclide étendu ne se contente pas de trouver le PGCD. Il permet aussi de trouver les coefficients de Bézout, c'est-à-dire les deux nombres u et v de l'équation de Bezout $au + bv = pgcd(a,b)$.

Autrement dit, elle calcule explicitement u et v tels que :

$$au + bv = pgcd(a,b)$$

Quand le PGCD vaut 1, on a :

$$au + bv = 1$$

Et si on regarde cette équation modulo b, le terme en bv s'annule (car divisible par b), donc :

$$au \equiv 1(modb)$$

Cela signifie que u **est l'inverse de** a **modulo** b et c'est grâce à cette propriété qu'on peut calculer les inverses nécessaires dans de nombreux algorithmes cryptographiques.

5. Inverses Modulo

L'inverse modulo a par rapport à m (a^{-1}) est un entier x tel que :

$$a \cdot x \equiv 1 \pmod{m}.$$

Il existe si et seulement si $\text{pgcd}(a, m) = 1$, et peut être trouvé avec l'algorithme d'Euclide étendu.

Si $\text{pgcd}(a, b) = 1$, on peut trouver un inverse modulo alors il existe x tel que $ax \equiv 1 \pmod{b}$.

6. Congruences

Une congruence est une relation de la forme :

$$ax \equiv b \pmod{m},$$

Tu peux la réécrire comme :

$$ax - b = km \quad \Rightarrow \quad ax + m(-k) = b$$

En utilisant l'algorithme d'Euclide étendu, tu peux déterminer si elle a une solution (si $\text{pgcd}(a, m)$ divise b) et la trouver explicitement.

7. Théorème des Restes Chinois

Il permet de résoudre des systèmes de congruences simultanées, c'est-à-dire plusieurs équations modulaires à la fois. Il s'applique lorsque les moduli sont premiers entre eux (c'est-à-dire que leur PGCD vaut 1 deux à deux).

Autrement dit, on cherche un nombre x qui satisfait plusieurs conditions du type :

$$x \equiv a_1 \pmod{m_1} \quad \text{et} \quad x \equiv a_2 \pmod{m_2},$$

Le théorème des restes Chinois garantit alors (si les moduli sont premiers entre eux) qu'il existe une solution unique modulo $m_1 \cdot m_2$.

Ces outils sont la base de nombreux algorithmes modernes (RSA, Diffie-Hellman, calculs cryptographiques). Ils permettent de manipuler efficacement des entiers très grands, tout en gardant des opérations rapides et sûres.

8. La crypto

L'algorithme RSA à clé publique repose entièrement sur de l'arithmétique modulaire et donc sur les théorèmes / algorithmes cités ci-dessus.

1. Divisions et congruences

Exercice 17. En utilisant une calculatrice, déterminer le quotient et le reste de :

(a) 34787 divisé par 353

Exercice 18. Utiliser l'algorithme d'Euclide pour déterminer le plus grand commun diviseur des nombres suivants :

(a) pgcd(291,252)

(b) pgcd(16261,85652)

(c) pgcd(139024789,93278890)

(d) pgcd(16534528044,8332745927)

Exercice 19. En utilisant l'algorithme d'Euclide, trouver les entiers p et q tel que

$$3066p + 713q = 1$$

Exercice 20. Trouver toutes les valeurs de x comprises entre 0 et $m-1$ qui sont solutions des congruences suivantes :

(a) $x + 17 \equiv 23 \pmod{37}$

(b) $x + 42 \equiv 19 \pmod{51}$

2. Inverses, unités et générateurs

Exercice 21. Trouver une unique valeur x qui résoud simultanément les deux congruences suivantes :

$$x \equiv 4 \pmod{7} \qquad \text{et} \qquad x \equiv 3 \pmod{9}.$$

Exercice 22. Trouver une unique valeur x qui résoud simultanément les deux congruences suivantes :

$$x \equiv 13 \pmod{71} \qquad \text{et} \qquad x \equiv 41 \pmod{97}.$$

Exercice 23. Trouver une unique valeur x qui résoud simultanément les trois congruences suivantes :

$$x \equiv 4 \pmod{7} \qquad \text{et} \qquad x \equiv 5 \pmod{8} \qquad \text{et} \qquad x \equiv 11 \pmod{15}.$$

LSINC1113 - Compléments de mathématiques

Correction TP3 - Théorie des nombres

---

1. Divisions et congruences

Solution 17. $34787 \div 353$, si on fait $34787 - 353$ dans une calculette et que l'on compte le nombre de fois qu'on peut faire cette soustraction, on trouve le quotient et le reste. Le quotient est donc 98 et le reste est $r = 34787 - (353 \cdot q) = 193$.

Solution 18.

(a) $\text{pgcd}(291, 252) = 3$
1. $291 \div 252 = 1$, reste $291 - 252 = 39$
2. $252 \div 39 = 6$, reste $252 - 39 \times 6 = 18$
3. $39 \div 18 = 2$, reste $39 - 18 \times 2 = 3$
4. $18 \div 3 = 6$, reste $18 - 3 \times 6 = 0$

(b) $\text{pgcd}(16261, 85652) = 161$
1. $85652 \div 16261 = 5$, reste $85652 - 16261 \times 5 = 4347$
2. $16261 \div 4347 = 3$, reste $16261 - 4347 \times 3 = 3220$
3. $4347 \div 3220 = 1$, reste $4347 - 3220 \times 1 = 1127$
4. $3220 \div 1127 = 2$, reste $3220 - 1127 \times 2 = 966$
5. $1127 \div 966 = 1$, reste $1127 - 966 \times 1 = 161$
6. $966 \div 161 = 6$, reste $966 - 161 \times 6 = 0$

(c) $\text{pgcd}(139024789, 93278890) = 1$
1. $139024789 \div 93278890 = 1$, reste $139024789 - 93278890 \times 1 = 45745899$
2. $93278890 \div 45745899 = 2$, reste $93278890 - 45745899 \times 2 = 1787092$
3. $45745899 \div 1787092 = 25$, reste $45745899 - 1787092 \times 25 = 1068599$
4. $1787092 \div 1068599 = 1$, reste $1787092 - 1068599 \times 1 = 718493$
5. $1068599 \div 718493 = 1$, reste $1068599 - 718493 \times 1 = 350106$
6. $718493 \div 350106 = 2$, reste $718493 - 350106 \times 2 = 18281$
7. $350106 \div 18281 = 19$, reste $350106 - 18281 \times 19 = 2767$
8. $18281 \div 2767 = 6$, reste $18281 - 2767 \times 6 = 1679$
9. $2767 \div 1679 = 1$, reste $2767 - 1679 \times 1 = 1088$
10. $1679 \div 1088 = 1$, reste $1679 - 1088 \times 1 = 591$
11. $1088 \div 591 = 1$, reste $1088 - 591 \times 1 = 497$
12. $591 \div 497 = 1$, reste $591 - 497 \times 1 = 94$
13. $497 \div 94 = 5$, reste $497 - 94 \times 5 = 27$
14. $94 \div 27 = 3$, reste $94 - 27 \times 3 = 13$
15. $27 \div 13 = 2$, reste $27 - 13 \times 2 = 1$
16. $13 \div 1 = 13$, reste $13 - 1 \times 13 = 0$

(d) $\text{pgcd}(16534528044, 8332745927) = 43$

Solution 19. Pour $3066p + 713q = 1$, utiliser l'algorithme d'Euclide étendu.
On remonte les étapes pour prouver que l'on peut écrire 1 comme combinaison linéaire de 3066 et 713.

$$3066 \div 713 = 4, \text{, reste } 214$$
$$713 \div 214 = 3, \text{, reste } 71$$
$$214 \div 71 = 3, \text{, reste } 1$$
$$71 \div 1 = 71, \text{, reste } 0$$

Le pgcd$(3066, 713) = 1$. Maintenant, on peut donc remonter et appliquer le théorème d'Euclide étendu pour exprimer 1 en fonction des deux termes de départ :

$$\begin{aligned} 1 &= 214 - 71 \cdot 3 \\ 1 &= 214 - (713 - 214 \cdot 3) \cdot 3 \\ 1 &= 214 \cdot 10 - 713 \cdot 3 \\ 1 &= (3066 - 713 \cdot 4) \cdot 10 - 713 \cdot 3 \\ 1 &= 3066 \cdot 10 - 713 \cdot 43 \end{aligned}$$

On trouve donc $q = -43$ et $p = 10$.

Solution 20.

(a) $x + 17 \equiv 23 \pmod{37}$

$$x \equiv 23 - 17 \pmod{37} \quad \Rightarrow \quad x \equiv 6 \pmod{37}.$$

(b) $x + 42 \equiv 19 \pmod{51}$

$$x \equiv 19 - 42 \pmod{51} \quad \Rightarrow \quad x \equiv -23 \pmod{51}.$$

Ajouter 51 pour obtenir une solution positive :

$$x \equiv 28 \pmod{51}.$$

2. Inverses, unités et générateurs

Solution 21. On va utiliser le théorème des Restes Chinois car c'est un système de congruences :

$$x \equiv 4 \pmod{7} \text{ et } x \equiv 3 \pmod{9}$$

1. $x = a_1 \cdot N_1 \cdot N_1^{-1} + a_2 \cdot N_2 \cdot N_2^{-1} \pmod{M}$
2. *Les conditions du théorème des restes Chinois*
   $\rightarrow$ 7 et 9 sont premiers entre eux (pgcd$(9, 7) = 1$). Une solution unique existe donc $7 \cdot 9 = 63 = M$
3. *Calculs des inverses*
   $N_1 = \frac{M}{m_1} = \frac{63}{7} = 9$
   On cherche l'inverse donc de 9 (mod 7) $\equiv$ 2 (mod 7). Mais pour qu'un inverse existe, il faut que $2t \equiv 1 \pmod{7}$ ait une solution. Pour se faire, on utilise Euclide étendu (lien entre les deux équations du point 4) :
   $2x + 7k = 1 \rightarrow 7 = 2 \cdot 3 + 1$
   On trouve finalement $x = -3$ et $k = 1$. Et donc $N_1^{-1} = -3 \pmod{7} \equiv 4 \pmod{7}$

   $N_2 = \frac{M}{m_2} = \frac{63}{9} = 7$
   Dans ce cas ci, on ne sait pas réduire plus. On cherche donc à inverser 7 (mod 9). Mais pour qu'un inverse existe, il faut que $7t \equiv 1 \pmod{9}$ ait une solution. Pour se faire, on utilise Euclide étendu :

$$7z + 9k = 1 \to 9 = 7 \cdot 1 + 2$$
$$\to 7 = 2 \cdot 3 + 1$$
$$\to 2 = 1 \cdot 2 + 0$$

Ensuite,

$$7 - 2 \cdot 3 = 1$$
$$7 - (9 - 7 \cdot 1) \cdot 3 = 1$$
$$7 \cdot 4 - 9 \cdot 3 = 1$$
$$z = 4 \text{ et } k = -3$$

Et donc $N_2^{-1} = 4 \pmod 9$

4. *Finalement,*
$x = 4 \cdot 9 \cdot 4 + 3 \cdot 7 \cdot 4 = 144 + 84 \pmod{63}$
$= 18 + 21 \pmod{63}$
$= 39 \pmod{63}$

Solution 22.

$$x \equiv 13 \pmod{71} \text{ et } x \equiv 41 \pmod{97}$$

1. $x = a_1 \cdot N_1 \cdot N_1^{-1} + a_2 \cdot N_2 \cdot N_2^{-1} \pmod M$
2. *Les conditions du théorème des restes Chinois*
$\to$ 71 et 97 sont premiers entre eux ($\text{pgcd}(97, 71) = 1$). Une solution unique existe donc $71 \cdot 97 = 6887 = M$
3. *Calculs des inverses*
$N_1 = \frac{6887}{71} = 97$
On cherche l'inverse de $97 \equiv 26 \pmod{71}$. Pour vérifier que l'inverse existe, on résout $26t \equiv 1 \pmod{71}$ par Euclide étendu :
$26x + 71k = 1 \to 71 = 2 \cdot 26 + 19$
$26 = 1 \cdot 19 + 7$
$19 = 2 \cdot 7 + 5$
$7 = 1 \cdot 5 + 2$
$5 = 2 \cdot 2 + 1$
$2 = 2 \cdot 1 + 0$

On remonte ensuite,

$$5 - 2 \cdot 2 = 1$$
$$5 - (7 - 5 \cdot 1) \cdot 2 = 1$$
$$5 \cdot 3 - 7 \cdot 2 = 1$$
$$(19 - 7 \cdot 2) \cdot 3 - 7 \cdot 2 = 1$$
$$19 \cdot 3 - 7 \cdot 8 = 1$$
$$19 \cdot 3 - (26 - 19 \cdot 1) \cdot 8 = 1$$
$$19 \cdot 11 - 26 \cdot 8 = 1$$
$$(71 - 26 \cdot 2) \cdot 11 - 26 \cdot 8 = 1$$
$$71 \cdot 11 - 26 \cdot 30 = 1$$

On trouve $x = -30$ mais qu'on rapporte à $x = 41 \pmod{71}$.
$N_1^{-1} = 41 \pmod{71}$

$N_2 = \frac{6887}{97} = 71$
On cherche l'inverse de $71 \pmod{97}$. Pour se faire, on utilise Euclide étendu :

$71z + 97k = 1 \rightarrow$ on reprend la dernière ligne d'après

$$71 \cdot 11 - 26 \cdot 30 = 1$$
$$71 \cdot 11 - (97 - 71) \cdot 30 = 1$$
$$71 \cdot 41 - 97 \cdot 30 = 1$$

On trouve donc finalement $z = 41$ mais qu'on rapporte à $z = 41 \pmod{97}$.

4. *Finalement*,
$$\begin{aligned} x &= 13 \cdot 97 \cdot 41 + 41 \cdot 71 \cdot 41 = 51701 + 119351 \pmod{6887} \\ &= 3492 + 2272 \pmod{6887} \\ &= 5764 \pmod{6887} \end{aligned}$$

Solution 23.

$$x \equiv 4 \pmod{7} \quad \text{et} \quad x \equiv 5 \pmod{8} \quad \text{et} \quad x \equiv 11 \pmod{15}$$

1. $x = a_1 \cdot N_1 \cdot N_1^{-1} + a_2 \cdot N_2 \cdot N_2^{-1} + a_3 \cdot N_3 \cdot N_3^{-1} \pmod{M}$
2. *Les conditions du théorème des restes Chinois*
$\rightarrow$ 7, 8 et 15 sont premiers entre eux ($\text{pgcd}(7, 8) = 1$, $\text{pgcd}(7, 15) = 1$ and $\text{pgcd}(8, 15) = 1$).
Une solution unique existe donc $7 \cdot 8 \cdot 15 = 840 = M$
3. *Calculs des inverses*
$N_1 = \frac{840}{7} = 120$
On cherche l'inverse de $120 \equiv 1 \pmod{7}$. Pour se faire, on utilise Euclide étendu :
$1x + 7k = 1 \rightarrow 7 - 7 \cdot 1 = 0$
On trouve donc finalement $x = 1 \pmod{7}$.

$N_2 = \frac{840}{8} = 105$
On cherche l'inverse de $105 \equiv 1 \pmod{8}$. Pour se faire, on utilise Euclide étendu :
$1z + 8k = 1 \rightarrow 8 - 8 \cdot 1 = 0$
On trouve donc finalement $z = 1 \pmod{8}$.

$N_3 = \frac{840}{15} = 56$
On cherche l'inverse de $56 \equiv 11 \pmod{15}$. Pour se faire, on utilise Euclide étendu :
$11y + 15k = 1 \rightarrow 15 - 11 \cdot 1 = 4$

$$\begin{aligned} 11 - 4 \cdot 2 &= 3 \\ 4 - 3 \cdot 1 &= 1 \\ 3 - 3 \cdot 1 &= 1 \\ 4 - 3 \cdot 1 &= 1 \\ 4 - (11 - 4 \cdot 2) \cdot 1 &= 1 \\ 4 \cdot 3 - 11 \cdot 1 &= 1 \\ (15 - 11 \cdot 1) \cdot 3 - 11 \cdot 1 &= 1 \\ 15 \cdot 3 - 11 \cdot 4 &= 1 \end{aligned}$$

On trouve donc finalement $y = -4 \pmod{15}$ $y = 11 \pmod{15}$.

4. *Finalement*,
$$\begin{aligned} x &= (4 \cdot 120 \cdot 1) + (5 \cdot 105 \cdot 1) + (11 \cdot 56 \cdot 11) \pmod{840} \\ &= 480 + 525 + 6784 \pmod{840} \\ &= 7789 \pmod{840} \\ &= 469 \pmod{840} \end{aligned}$$

LSINC1113 - Compléments de mathématiques

TP4 - Théorie des nombres

Rappels Théoriques

On note

$$a \equiv b \pmod{n}$$

pour dire que a et b laissent le **même reste** quand on les divise par n.
Exemples :

$$17 \equiv 2 \pmod{5} \quad \text{car } 17 = 3 \cdot 5 + 2,$$

$$-3 \equiv 4 \pmod{7} \quad \text{car } -3 = -1 \cdot 7 + 4.$$

En arithmétique modulaire (et en informatique en général), on calcule souvent des puissances très grandes du type : an (mod m)

Par exemple : 7^{560} (mod 561) Cependant, si on essaye de calculer 7^{560} directement, c'est impossible à la main (et même pour un ordinateur, c'est beaucoup trop grand). Donc, on a besoin d'une méthode efficace pour calculer ce résultat sans exploser les nombres. C'est exactement ce que fait le fast powering (ou exponentiation rapide).

1. Exponentiation rapide (Fast Power)

On veut souvent calculer des choses du type

$$a^n \pmod{m}$$

avec n très grand. Calculer a^n "normalement" est impossible à la main (et très lourd pour un ordi).

Idée clé : au lieu de multiplier a par lui-même n fois, on utilise le fait que

$$a^8 = (a^4)^2 = ((a^2)^2)^2,$$

donc on peut atteindre de grands exposants en **faisant surtout des carrés**. C'est ce qui rend possible les calculs cryptographiques comme RSA, où les exposants ont parfois des centaines de chiffres !

Principe de l'algorithme (intuition) :

- On écrit l'exposant n en **binaire** (somme de puissances de 2).
- On parcourt les bits de n et on :
  - **carré** le résultat à chaque étape (on double l'exposant),
  - **multiplie par** a seulement quand le bit courant vaut 1.
- À chaque multiplication, on **réduit modulo** m pour que les nombres restent petits.

Exemple : calculer $3^{13} \mod 7$

- 13 en binaire : $13 = 8 + 4 + 1 = 1101_2$.

- On construit le résultat pas à pas (en réduisant mod 7 à chaque fois) :
  - $3^1 \equiv 3 \pmod 7$
  - $3^2 \equiv 3^2 = 9 \equiv 2 \pmod 7$
  - $3^4 \equiv 2^2 = 4 \pmod 7$
  - $3^8 \equiv 4^2 = 16 \equiv 2 \pmod 7$
- Puis $3^{13} = 3^{8+4+1} = 3^8 \cdot 3^4 \cdot 3^1$:

$$3^{13} \equiv 2 \cdot 4 \cdot 3 = 24 \equiv 3 \pmod 7.$$

2. Petit théorème de Fermat et inverse modulaire

Si p est un nombre premier et a n'est pas multiple de p, alors :

$$a^{p-1} \equiv 1 \pmod p.$$

On peut réécrire

$$a^{p-1} = a^{p-2} \cdot a \equiv 1 \pmod p.$$

Donc a^{p-2} joue le rôle de **l'inverse de a modulo p** :

$$a^{-1} \equiv a^{p-2} \pmod p.$$

Intuition : multiplier par a^{p-2} "annule" a modulo p.

Pour les exos : pour trouver l'inverse de a modulo p (avec p premier) :

$$a^{-1} \equiv a^{p-2} \pmod p$$

en utilisant l'**exponentiation rapide**.

3. Déterminer $(\mathbb{Z}/m\mathbb{Z})^*$

On note

$$(\mathbb{Z}/m\mathbb{Z})^* = \{a \in \{1, \dots, m-1\} \mid \text{pgcd}(a, m) = 1\}.$$

Intuition : ce sont tous les nombres entre 1 et $m-1$ qui n'ont **aucun facteur commun** avec m (à part 1). Ce sont exactement ceux pour lesquels un **inverse mod** m existe.

Exemple : $m = 10$.

$$(\mathbb{Z}/10\mathbb{Z})^* = \{1, 3, 7, 9\}$$

car $\text{pgcd}(1, 10) = \text{pgcd}(3, 10) = \text{pgcd}(7, 10) = \text{pgcd}(9, 10) = 1$.

La taille de cet ensemble est la **fonction d'Euler** $\phi(m)$.

4. Racines primitives modulo p

On travaille ici modulo un **nombre premier** p.

L'ensemble $(\mathbb{Z}/p\mathbb{Z})^*$ contient $p-1$ éléments. Un entier g est une **racine primitive modulo** p si ses puissances :

$$g^1, g^2, \dots, g^{p-1} \pmod p$$

donnent **tous les éléments** de $(\mathbb{Z}/p\mathbb{Z})^*$ (chacun apparaît une fois).

Intuition : g est comme un "générateur" : en multipliant toujours par g, on fait le tour de tous les nombres inversibles mod p.

Exemple : modulo 7, on vérifie que 3 est racine primitive :

$$3^1 = 3, \quad 3^2 = 9 \equiv 2, \quad 3^3 \equiv 6, \quad 3^4 \equiv 4, \quad 3^5 \equiv 5, \quad 3^6 \equiv 1 \pmod 7.$$

On a bien obtenu $1, 2, 3, 4, 5, 6$ avant d'avoir $\equiv 1$.

5. Valeur de $2^{(p-1)/2} \mod p$

Un entier a est un **résidu quadratique modulo** p (avec p premier) s'il existe un $x \in \{1, \ldots, p-1\}$ tel que

$$x^2 \equiv a \pmod p.$$

Sinon, on dit que a est un **non-résidu quadratique**.

Critère d'Euler : pour un entier a et un nombre premier p avec $\text{pgcd}(a, p) = 1$:

$$a^{(p-1)/2} \equiv \begin{cases} 1 \pmod p, & \text{si } a \text{ est un résidu quadratique modulo } p, \\ -1 \equiv p-1 \pmod p, & \text{si } a \text{ est un non-résidu quadratique.} \end{cases}$$

Intuition : la puissance $(p-1)/2$ "teste" si a est un carré modulo p :

- Si le résultat est 1 : "oui, a est un carré mod p" ;
- Si le résultat est -1 : "non, ce n'est pas un carré".

Exemple : modulo 7, avec $a = 2$

On calcule les carrés modulo 7 :

$$1^2 = 1, \ 2^2 = 4, \ 3^2 = 9 \equiv 2, \ 4^2 = 16 \equiv 2, \ 5^2 = 25 \equiv 4, \ 6^2 = 36 \equiv 1.$$

On obtient les valeurs possibles : $1, 2, 4$. Donc 2 **est** un résidu quadratique modulo 7.

On vérifie le critère d'Euler :

$$2^{(7-1)/2} = 2^3 = 8 \equiv 1 \pmod 7 \quad \Rightarrow \quad 2 \text{ est bien un résidu quadratique.}$$

1. Inverses, unités et générateurs

Exercice 24. Pour chacun des nombres premiers p et nombre a, calculer $a^{-1} \bmod p$ en utilisant (i) l'algorithme d'Euclide étendu et (ii) le fast power algorithm et le petit théorème de Fermat.

(a) $p = 47$ et $a = 11$.

(b) $p = 587$ et $a = 345$.

(c) $p = 104801$ et $a = 78467$.

Exercice 25. Déterminer l'espace $(\mathbb{Z}/m\mathbb{Z})^*$ pour $m = \{7, 10, 13, 24\}$.

Exercice 26. Rappelons que g est appelé une racine primitive module p, si la puissance de g donne tous éléments non-nuls de $\mathbb{F}_p \coloneqq (\mathbb{Z}/p\mathbb{Z})^*$:

(a) Pour lequel des nombres premiers suivants 2 est-il une racine primitive modulo p ?
(i) $p = 7$ (ii) $p = 13$ (iii) $p = 19$ (iv) $p = 23$

(b) Pour lequel des nombres premiers suivants 3 est-il une racine primitive modulo p ?
(i) $p = 5$ (ii) $p = 7$ (iii) $p = 11$ (iv) $p = 17$

(c) Trouvez une racine primitive pour chacun des nombres premiers suivants.
(i) $p = 23$ (ii) $p = 29$ (iii) $p = 41$ (iv) $p = 43$

Exercice 27. Déterminer la valeur de

$$2^{(p-1)/2} \pmod{p}$$

pour tous les nombres premiers $3 \leq p < 20$. Faites une conjecture sur les valeurs possibles de $2^{(p-1)/2} (\bmod\ p)$ lorsque p est premier et prouvez que votre conjecture est correcte.

LSINC1113 - Compléments de mathématiques

Correction TP4 - Théorie des nombres

---

1. Inverses, unités et générateurs

Solution 24.

(a) $a = 11$ and $p = 47$, on cherche l'inverse de $11 \pmod{47}$.

1. Pour qu'il soit inversible, on doit vérifier que $11t \equiv 1 \pmod{47}$ autrement dit que Euclide étendu : $11t + 47k = 1$

Pour ce faire on calcule le $pgcd(a, p)$:

$47 - 11 \cdot 4 = 3$

$11 - 3 \cdot 3 = 2$

$3 - 2 \cdot 1 = 1$

$2 - 1 \cdot 2 = 0$

11 est inversible modulo 47, le *pgcd* vaut 1.

$$\begin{aligned} 3 - 2 \cdot 1 &= 1 \\ 3 - (11 - 3 \cdot 3) \cdot 1 &= 1 \\ 3 \cdot 4 - 11 \cdot 1 &= 1 \\ (47 - 11 \cdot 4) \cdot 4 - 11 \cdot 1 &= 1 \\ 47 \cdot 4 - 11 \cdot 17 &= 1 \end{aligned}$$

On a $t = -17$ qu'on transforme en $t = 30 \pmod{47}$.

2. Fermat. On va tirer parti de la décomposition binaire.

Théorème : $11^{47-2} \pmod{47} = 11^{45} \pmod{47}$

On écrit sous forme binaire : $45 = 32 + 8 + 4 + 1 = 2^5 + 2^3 + 2^2 + 2^0$

$$\begin{aligned} 11^2 &\equiv 121 \equiv 27 \pmod{47} \\ 11^4 &\equiv 27^2 \equiv 729 \equiv 24 \pmod{47} \\ 11^8 &\equiv 24^2 \equiv 576 \equiv 12 \pmod{47} \\ 11^{16} &\equiv 12^2 \equiv 144 \equiv 3 \pmod{47} \\ 11^{32} &\equiv 3^2 \equiv 9 \pmod{47} \end{aligned}$$

$11^{45} = 11^{32} \cdot 11^8 \cdot 11^4 \cdot 11 = 9 \cdot 12 \cdot 24 \cdot 11 = 9504 = 30 \pmod{47}$

(b) $a = 345$ and $p = 587$, on cherche l'inverse de $345 \pmod{587}$.

1. Pour qu'il soit inversible, on doit vérifier que $345t \equiv 1 \pmod{587}$ autrement dit que Euclide étendu : $345x + 587k = 1$

$587 - 345 \cdot 1 = 242$

$345 - 242 \cdot 1 = 103$

$242 - 103 \cdot 2 = 36$

$103 - 36 \cdot 2 = 31$

$36 - 31 \cdot 1 = 5$

$31 - 5 \cdot 6 = 1$

$5 - 1 \cdot 5 = 0$

345 est inversible modulo 587, le *pgcd* vaut 1.

$$31 - 5 \cdot 6 = 1$$
$$31 - (36 - 31) \cdot 6 = 1 \rightarrow 31 \cdot 7 - 36 \cdot 6 = 1$$
$$(103 - 36 \cdot 2) \cdot 7 - 36 \cdot 6 = 1 \rightarrow 103 \cdot 7 - 36 \cdot 20 = 1$$
$$103 \cdot 7 - (242 - 103 \cdot 2) \cdot 20 = 1 \rightarrow 103 \cdot 47 - 242 \cdot 20 = 1$$
$$(345 - 242) \cdot 47 - 242 \cdot 20 = 1 \rightarrow 345 \cdot 47 - 242 \cdot 67 = 1$$
$$345 \cdot 47 - (587 - 345) \cdot 67 = 1 \rightarrow 345 \cdot 114 - 587 \cdot 67 = 1$$

On a $t = 114 \pmod{587}$.

2. Fermat

$345^{587-2} \pmod{587} = 345^{585} \pmod{587}$

$$585 = 512 + 64 + 8 + 1$$
$$345^2 \equiv 119025 \equiv 451 \pmod{587}$$
$$345^4 \equiv 451^2 \equiv 299 \pmod{587}$$
$$345^8 \equiv 299^2 \equiv 177 \pmod{587}$$
$$345^{16} \equiv 177^2 \equiv 218 \pmod{587}$$
$$345^{32} \equiv 218^2 \equiv 564 \pmod{587}$$
$$345^{64} \equiv 564^2 \equiv 529 \pmod{587}$$
$$345^{128} \equiv 529^2 \equiv 429 \pmod{587}$$
$$345^{256} \equiv 429^2 \equiv 310 \pmod{587}$$
$$345^{512} \equiv 310^2 \equiv 419 \pmod{587}$$

$345^{585} = 345^{512} \cdot 345^{64} \cdot 345^8 \cdot 345 = 419 \cdot 529 \cdot 177 \cdot 345 = 114 \pmod{587}$

(c) $a = 78467$ et $p = 104801$ on cherche l'inverse de $78467 \pmod{104801}$.

1. Euclide étendu : On cherche à calculer $78467x \equiv 1 \pmod{104801}$ ou encore $78467 \cdot x + 104801 \cdot y = 1$. On commence par s'assurer que le *pgcd* $= 1$.

$$104801 = 1 \cdot 78467 + 26334$$
$$78467 = 2 \cdot 26334 + 25799$$
$$26334 = 1 \cdot 25799 + 535$$
$$25799 = 48 \cdot 535 + 119$$
$$535 = 4 \cdot 119 + 59$$
$$119 = 2 \cdot 59 + 1$$
$$59 = 1 \cdot 59 + 0$$

78467 est inversible modulo 104801.

$$1 = 119 - 2 \cdot 59$$
$$1 = 119 - 2 \cdot (535 - 4 \cdot 119)$$
$$1 = 9 \cdot 119 - 2 \cdot 535$$
$$1 = 9 \cdot (25799 - 48 \cdot 535) - 2 \cdot 535$$
$$1 = 9 \cdot 25799 - 434 \cdot 535$$
$$1 = 9 \cdot 25799 - 434 \cdot (26334 - 1 \cdot 25799)$$
$$1 = 443 \cdot 25799 - 434 \cdot 263341 \qquad = 443 \cdot (78467 - 2 \cdot 26334) - 434 \cdot 263341 = 443 \cdot 78467 - 1320$$

On a $x = 1763 \pmod{104801}$.

2. Fermat :

$$78467^{104801-2} \pmod{104801} = 78467^{104799} \pmod{104801}$$

$$104799 = 65536 + 32768 + 4096 + 2048 + 256 + 64 + 32 + 16 + 8 + 4 + 2 + 1$$

$$104799 = 2^{16} + 2^{15} + 2^{12} + 2^{11} + 2^8 + 2^6 + 2^5 + 2^4 + 2^3 + 2^2 + 2^1 + 2^0$$

$$78467^2 \equiv 6157070089 \equiv 11339 \pmod{104801}$$
$$78467^4 \equiv 11339^2 \equiv 86895 \pmod{104801}$$
$$78467^8 \equiv 86895^2 \equiv 38577 \pmod{104801}$$
$$78467^{16} \equiv 38577^2 \equiv 10729 \pmod{104801}$$
$$78467^{32} \equiv 10729^2 \equiv 39943 \pmod{104801}$$
$$78467^{64} \equiv 39943^2 \equiv 57626 \pmod{104801}$$
$$78467^{128} \equiv 57626^2 \equiv 31390 \pmod{104801}$$
$$78467^{256} \equiv 31390^2 \equiv 97899 \pmod{104801}$$
$$78467^{512} \equiv 97899^2 \equiv 57950 \pmod{104801}$$
$$78467^{1024} \equiv 57950^2 \equiv 64057 \pmod{104801}$$
$$78467^{2048} \equiv 64057^2 \equiv 25696 \pmod{104801}$$
$$78467^{4096} \equiv 25696^2 \equiv 38116 \pmod{104801}$$
$$78467^{8192} \equiv 38116^2 \equiv 77994 \pmod{104801}$$
$$78467^{16384} \equiv 77994^2 \equiv 99593 \pmod{104801}$$
$$78467^{32768} \equiv 99593^2 \equiv 84606 \pmod{104801}$$
$$78467^{65536} \equiv 84606^2 \equiv 57334 \pmod{104801}$$

$78467^{104799} = (78467^{65536})(78467^{32768})(78467^{4096})(78467^{2048})(78467^{256})(78467^{64})(78467^{16})(78467^{8})(78467$
$= 57334 \cdot 84606 \cdot 38116 \cdot 25696 \cdot 97899 \cdot 57626 \cdot 10729 \cdot 38577 \cdot 86895 \cdot 11339 \cdot 78467 \pmod{104801}$
on fait comme précédemment et on tombe sur $x = 1763 \pmod{104801}$.

Solution 25.

(a) $m = 7 \to \mathbb{Z}/7\mathbb{Z} = 1, 2, 3, 4, 5, 6 \quad \Phi(7) = 6$

(b) $m = 10 \to \mathbb{Z}/10\mathbb{Z} = 1, 3, 7, 9 \quad \Phi(10) = 4$

(c) $m = 13 \to \mathbb{Z}/13\mathbb{Z} = 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12 \quad \Phi(13) = 12$

(d) $m = 24 \to \mathbb{Z}/24\mathbb{Z} = 1, 5, 7, 11, 13, 17, 19, 23 \quad \Phi(24) = 8$

Solution 26.

(a)

(i) $p = 7$

Pour vérifier que 2 est une racine primitive modulo p, il faut vérifier que l'ordre de 2 modulo p est p-1. Cela revient à $2^k \equiv 1 \pmod{p}$ pour k allant de 1 à $p - 1, k \neq 0$.

$2^1 \equiv 2 \pmod 7$

$2^2 \equiv 4 \pmod 7$

$2^3 \equiv 8 \equiv 1 \pmod 7$

2 n'est donc pas une racine primitive modulo 7.

(ii) $p = 13$

$2^1 \equiv 2 \pmod{13}$

$2^2 \equiv 4 \pmod{13}$

$2^3 \equiv 8 \pmod{13}$
$2^4 \equiv 16 \equiv 3 \pmod{13}$
$2^5 \equiv 32 \equiv 6 \pmod{13}$
$2^6 \equiv 64 \equiv 12 \pmod{13}$
$2^7 \equiv 128 \equiv 11 \pmod{13}$
$2^8 \equiv 256 \equiv 9 \pmod{13}$
$2^9 \equiv 512 \equiv 5 \pmod{13}$
$2^{10} \equiv 1024 \equiv 10 \pmod{13}$
$2^{11} \equiv 2048 \equiv 7 \pmod{13}$
$2^{12} \equiv 4096 \equiv 1 \pmod{13}$

2 est une racine primitive modulo 13.

(iii) $p = 19$
$2^1 \equiv 2 \pmod{19}$
$2^2 \equiv 4 \pmod{19}$
$2^3 \equiv 8 \pmod{19}$
$2^4 \equiv 16 \pmod{19}$
$2^5 \equiv 32 \equiv 13 \pmod{19}$
$2^6 \equiv 64 \equiv 7 \pmod{19}$
$2^7 \equiv 128 \equiv 14 \pmod{19}$
$2^8 \equiv 256 \equiv 9 \pmod{19}$
$2^9 \equiv 512 \equiv 18 \pmod{19}$
$2^{10} \equiv 1024 \equiv 17 \pmod{19}$
$2^{11} \equiv 2048 \equiv 15 \pmod{19}$
$2^{12} \equiv 4096 \equiv 11 \pmod{19}$
$2^{13} \equiv 8192 \equiv 3 \pmod{19}$
$2^{14} \equiv 16384 \equiv 6 \pmod{19}$
$2^{15} \equiv 12 \pmod{19}$
$2^{16} \equiv 5 \pmod{19}$
$2^{17} \equiv 10 \pmod{19}$
$2^{18} \equiv 1 \pmod{19}$

2 est une racine primitive modulo 19.

(iv) $p = 23$
$2^1 \equiv 2 \pmod{23}$
$2^2 \equiv 4 \pmod{23}$
$2^3 \equiv 8 \pmod{23}$
$2^4 \equiv 16 \pmod{23}$
$2^5 \equiv 32 \equiv 9 \pmod{23}$
$2^6 \equiv 64 \equiv 18 \pmod{23}$
$2^7 \equiv 128 \equiv 13 \pmod{23}$
$2^8 \equiv 256 \equiv 3 \pmod{23}$
$2^9 \equiv 512 \equiv 6 \pmod{23}$
$2^{10} \equiv 1024 \equiv 12 \pmod{23}$
$2^{11} \equiv 2048 \equiv 1 \pmod{23}$
2 n'est pas une racine primitive modulo 23.

(b)

(i) $p = 5$
$3^4 \equiv 1 \pmod{5}$
3 est une racine primitive modulo 5.

(ii) $p = 7$
$3^6 \equiv 1 \pmod 7$
3 est une racine primitive modulo 7.

(iii) $p = 11$
$3^5 \equiv 1 \pmod{11}$
3 n'est une racine primitive modulo 11.

(iv) $p = 17$
$3^{16} \equiv 1 \pmod{17}$
3 est une racine primitive modulo 17.

(c)

(i) $p = 23$
$5^{22} \equiv 1 \pmod{23}$
5 est une racine primitive modulo 23.

(ii) $p = 29$
$2^{28} \equiv 1 \pmod{29}$
2 est une racine primitive modulo 29.

(iii) $p = 41$
$6^{16} \equiv 1 \pmod{41}$
6 n'est une racine primitive modulo 41.

(iv) $p = 43$
$3^{42} \equiv 1 \pmod{43}$
3 est une racine primitive modulo 43.

Solution 27.
On rappelle le critère d'Euler :
Pour un nombre premier impair p,

$$2^{(p-1)/2} \equiv \begin{cases} 1 \pmod p, & \text{si 2 est un } \textbf{résidu quadratique} \text{ modulo p,} \\ -1 \equiv p-1 \pmod p, & \text{si 2 est un } \textbf{résidu non quadratique} \text{ modulo p.} \end{cases}$$

$2^{(p-1)/2} \pmod p 3 \leq p \leq 20 \to p \in [3, 5, 7, 11, 13, 17, 19]$

- Si $p = 3$, $2^1 \equiv 2 \pmod 3 \equiv p - 1$ donc 2 est un résultat non quadratique modulo 3.
  On peut vérifier en testant si un carré donne 2 modulo 3 :
  $1^2 \equiv 1 \pmod 3$
  $2^2 \equiv 4 \equiv 1 \pmod 3$
- Si $p = 5$, $2^2 \equiv 4 \pmod 5 \to p - 1$ donc 2 est un résultat non quadratique modulo 5.
  On peut vérifier en testant si un carré donne 2 modulo 5 :
  $1^2 \equiv 1 \pmod 5$
  $2^2 \equiv 4 \pmod 5$
  $3^2 \equiv 9 \equiv 4 \pmod 5$
  $4^2 \equiv 16 \equiv 1 \pmod 5$
- Si $p = 7$, $2^3 \equiv 8 \equiv 1 \pmod 7 \to 1$ donc 2 est un résidu quadratique modulo 7.
  On peut vérifier en testant si un carré donne 2 modulo 5 :
  $1^2 \equiv 1 \pmod 7$
  $2^2 \equiv 4 \pmod 7$
  $3^2 \equiv 9 \equiv 2 \pmod 7 \Rightarrow$ TOP
- Si $p = 11$, $2^5 \equiv 32 \equiv 10 \pmod{11} \to p - 1$ donc 2 est un résultat non quadratique modulo 11.

- Si $p = 13$, $2^6 \equiv 64 \equiv 12 \pmod{13} \rightarrow p - 1$ donc 2 est un résultat non quadratique modulo 13.
- Si $p = 17$, $2^8 \equiv 256 \equiv 1 \pmod{17} \rightarrow 1$ donc 2 est un résultat quadratique modulo 17.
- Si $p = 19$, $2^9 \equiv 512 \equiv 18 \pmod{19} \rightarrow p - 1$ donc 2 est un résultat non quadratique modulo 19.

$\rightarrow$ Si 2 est un résidu quadratique modulo p alors $2^{(p-1)/2} \equiv 1$ sinon $\equiv p - 1$

Partie 4 - Graphes

LSINC1113 - Compléments de mathématiques

TP5 - Graphes (spanning)

Rappels Théoriques

1. Un graphe pondéré

Un **graphe non orienté pondéré** est défini par un triplet $G = (V, E, w)$ où :

- V : ensemble des **sommets** (ou nœuds),
- $E \subseteq \{\{u, v\} : u, v \in V, u \neq v\}$: ensemble des **arêtes**,
- $w : E \to \mathbb{R}^+$: **poids** (ou coût) associé à chaque arête.

Intuition : un graphe, c'est un **réseau** :

- les sommets = des *villes*, des *machines*, des *personnes*...,
- les arêtes = des *routes*, des *câbles*, des *liens*,
- les poids = un *coût*, une *distance*, un *temps*, etc.

Un graphe est dit **connexe** si, pour *toute* paire de sommets, il existe un **chemin** qui les relie.
Intuition : le graphe est connexe si *tout est accessible* : à partir de n'importe quel sommet, on peut atteindre tous les autres en suivant des arêtes.

2. Arbres : des graphes simples sans boucles

Un **arbre** est un graphe connexe **sans cycle**.

Intuition :

- pas de "boucle" : on ne peut pas tourner en rond,
- il n'y a qu'**un seul chemin** entre deux sommets.

Un arbre à n sommets contient toujours $n - 1$ arêtes.
En effet, si on a moins de $n - 1$ arêtes, le graphe n'est pas assez "lié" pour être connexe. Si on a plus de $n - 1$ arêtes, on est obligé de créer au moins un cycle.

3. Arbre couvrant (Spanning Tree)

Un **arbre couvrant** d'un graphe connexe $G = (V, E)$ est un sous-graphe $T = (V, E_T)$ qui :

- contient **tous** les sommets de G,
- est **connexe**,
- ne contient **aucun cycle**.

Intuition : c'est une façon de **relier tous les sommets avec le minimum de "câbles"** : on garde toutes les villes, mais on ne garde que certaines routes, juste assez pour que tout reste connecté, sans boucle inutile.

Un même graphe connexe peut avoir **plusieurs** arbres couvrants différents.

4. Arbre couvrant minimal (MST)

Quand les arêtes sont pondérées, on s'intéresse au **coût total** :

$$w(T) = \sum_{e \in E_T} w(e).$$

Un **arbre couvrant minimal (ACM)** ou *Minimum Spanning Tree (MST)* est un arbre couvrant dont ce coût total est **le plus petit possible**.

Intuition : on veut relier tous les sommets en payant *le moins cher possible*.
Exemples typiques :

- construire un réseau de câbles entre des villes au coût minimal,
- connecter des ordinateurs avec un minimum de longueur de câble,
- simplifier un réseau en gardant l'essentiel des connexions.

5. Propriétés fondamentales

- Un MST d'un graphe connexe à n sommets comporte toujours $n - 1$ arêtes.
- Si tous les poids d'arêtes sont distincts, le MST est **unique**.
- **Propriété de coupe.** On considère une **coupe**, c.-à-d. une séparation des sommets en deux groupes A et B. Parmi toutes les arêtes qui traversent la coupe (qui relient un sommet de A à un sommet de B), l'arête de **plus faible poids** appartient forcément à *un* MST.
  Intuition : si tu dois connecter deux "blocs" de sommets, tu as tout intérêt à choisir la route la moins chère entre ces deux blocs.
- **Propriété de cycle.** Dans **n'importe quel cycle**, l'arête de **plus grand poids** ne peut *jamais* appartenir à un MST.
  Intuition : si tu as une boucle, tu peux toujours supprimer la route la plus chère de cette boucle : tout reste connecté, mais ça coûte moins cher. Donc un MST n'a aucune raison de garder cette arête.

Ces deux propriétés expliquent pourquoi les algorithmes comme Kruskal fonctionnent.

6. Algorithme classique : Kruskal

Idée générale : on part de **zéro arête** et on ajoute progressivement les arêtes **les moins chères** possibles *sans créer de cycle*.

1. Trier toutes les arêtes par **poids croissant**.
2. Parcourir la liste :
   - si l'ajout d'une arête **ne crée pas de cycle**, on l'ajoute à l'arbre,
   - sinon, on la **saute**.
3. Arrêter dès que l'on a $n - 1$ arêtes : on a alors un MST.

Intuition :

- on choisit toujours la prochaine arête la *moins chère* possible (stratégie "gourmande"),
- on évite les cycles pour garder une structure d'arbre,
- grâce à la propriété de coupe, on sait que cette stratégie donne bien un MST.

La complexité est $O(E \log E)$ à cause du tri des arêtes.

7. Applications

- Réseaux électriques, télécoms ou de transport (minimisation du coût total).
- Distribution logistique (réseaux de livraison, hôpitaux).
- Clustering hiérarchique et génération de labyrinthes.

1. Réseau hospitalier optimal (Kruskal)

Contexte : Un hôpital comprend plusieurs unités (urgences, chirurgie, pédiatrie, oncologie, réanimation) et un centre pharmaceutique. Chaque unité doit être reliée au centre pharmaceutique via un réseau de tubes pneumatiques, au coût minimal.

Objectif : Trouver un **arbre couvrant minimal** représentant le réseau optimal à l'aide de l'algorithme de **Kruskal**.

Données du graphe :

Connexion (arête)	**Coût (distance / travaux)**
(Pharmacie, Urgences)	4
(Pharmacie, Chirurgie)	2
(Urgences, Pédiatrie)	1
(Chirurgie, Oncologie)	3
(Pédiatrie, Oncologie)	5
(Chirurgie, Réanimation)	6
(Oncologie, Réanimation)	2

Travail à faire :

1. Implémentez l'algorithme de Kruskal en Julia.
2. Déterminez les arêtes de l'arbre couvrant minimal.
3. Calculez le coût total du réseau.

LSINC1113 - Compléments de mathématiques

Correction TP5 - Graphes (spanning)

---

1. Réseau hospitalier optimal (Kruskal)

L'hôpital veut relier toutes ses unités (urgences, chirurgie, pédiatrie, oncologie, réanimation) à la pharmacie centrale avec un réseau de tubes pneumatiques minimal. Chaque lien possible entre deux unités a un coût (distance, temps, ou difficulté d'installation).

Notre objectif : Relier toutes les unités avec un coût total minimal, sans cycle (boucle inutile).
→ C'est exactement ce que fait un arbre couvrant minimal (Minimum Spanning Tree — MST).

Rappel intuitif :

- **Couvrant** = relie tous les sommets du graphe.
- **Arbre** = pas de cycle.
- **Minimal** = la somme des poids des arêtes est la plus faible possible.

L'algorithme de **Kruskal** est basé sur un principe très simple et intuitif : "Commence par les connexions les moins chères, et ajoute-les tant qu'elles ne forment pas de boucle."

C'est une stratégie gloutonne (greedy algorithm) : à chaque étape, on fait le meilleur choix local (l'arête la moins chère), et à la fin, on obtient la meilleure solution globale.

1. Données de départ

Sommets : Pharmacie (Ph), Urgences (Ur), Chirurgie (Ch), Pédiatrie (Pe), Oncologie (On), Réanimation (Re)

Arêtes pondérées :

Arête	Poids (coût)
(Ph, Ur)	4
(Ph, Ch)	2
(Ur, Pe)	1
(Ch, On)	3
(Pe, On)	5
(Ch, Re)	6
(On, Re)	2

L'objectif est de trouver un **arbre couvrant minimal (MST)** minimisant la somme des coûts.

2. Étape 1 — Tri des arêtes par poids croissant

$$(Ur, Pe) : 1, \quad (Ph, Ch) : 2, \quad (On, Re) : 2, \quad (Ch, On) : 3, \quad (Ph, Ur) : 4, \quad (Pe, On) : 5, \quad (Ch, Re) : 6$$

3. Étape 2 — Initialisation

Chaque sommet est dans sa propre composante :

$$\{Ph\}, \{Ur\}, \{Ch\}, \{Pe\}, \{On\}, \{Re\}$$

Le MST est vide et le coût total est 0.

4. Étape 3 — Parcours des arêtes et décisions

On va maintenant parcourir cette liste et construire le réseau petit à petit :

- Ur - Pe (1) : ce sont deux unités isolées → pas de cycle. Donc on ajoute l'arête. (Intuition : le lien le moins cher, c'est toujours un bon début.)
  $\{Ur, Pe\}, \{Ph\}, \{Ch\}, \{On\}, \{Re\}$
- Ph – Ch (2) : ce sont deux unités isolées → pas de cycle. Donc on ajoute l'arête.
  $\{Ur, Pe\}, \{Ph, Ch\}, \{On\}, \{Re\}$
- On – Re (2) : ce sont deux unités isolées → pas de cycle. Donc on ajoute l'arête.
  $\{Ur, Pe\}, \{Ph, Ch\}, \{On, Re\}$
- Ch – On (3) : Ch appartient à $\{Ph, Ch\}$, On à $\{On, Re\}$ → relier ces ensembles ne crée pas de boucle. On ajoute. On vient de connecter deux sous-réseaux séparés.
  $\{Ur, Pe\}, \{Ph, Ch, On, Re\}$
- Ph – Ur (4) : Ph est dans Ph, Ch, On, Re, Ur dans Ur, Pe → pas encore connectés. On peut donc ajouter et on relie les deux grands sous ensemble. Tout l'hôpital est connecté.
  $\{Ur, Pe, Ph, Ch, On, Re\}$
- Pe – On (5), Ch – Re (6) : ces arêtes relient déjà des sommets du même ensemble. Elles créeraient des cycles, donc on les rejette.

Arrêt : après la 5^{e} arête ajoutée, le graphe est connexe (6 sommets ⇒ 5 arêtes).

5. Étape 4 — Résultat final

Arbre couvrant minimal (MST) :

$$\{(Ur, Pe),\ (Ph, Ch),\ (On, Re),\ (Ch, On),\ (Ph, Ur)\}$$

Coût total :

$$1 + 2 + 2 + 3 + 4 = \boxed{12}$$

6. Vérifications

- **Cardinalité :** 6 sommets ⇒ MST avec 5 arêtes.
- **Connexité :** tous les sommets sont reliés après la 5^{e} arête.
- **Pas de cycle :** assuré par Union-Find.
- **Optimalité :** respecte la propriété de coupe : l'arête (Ph, Ur) de coût 4 est la plus légère reliant les deux sous-ensembles ⇒ nécessairement dans le MST.

Conclusion : le coût minimal pour connecter toutes les unités hospitalières au réseau est de **12 unités de distance ou de coût**.

7. Code Julia correspondant

```julia
using Graphs
using GraphPlot
using Colors

# ---------- Données ----------
nodes = ["Pharmacie","Urgences","Chirurgie","Pédiatrie","Oncologie","Réanimation"]
idx = Dict(n => i for (i,n) in enumerate(nodes))

edge_list = [
    ("Pharmacie", "Urgences", 4),
    ("Pharmacie", "Chirurgie", 2),
    ("Urgences", "Pédiatrie", 1),
    ("Chirurgie", "Oncologie", 3),
    ("Pédiatrie", "Oncologie", 5),
    ("Chirurgie", "Réanimation", 6),
    ("Oncologie", "Réanimation", 2)
]

# ---------- Graphe ----------
g = SimpleGraph(length(nodes))
wmap = Dict{Tuple{Int,Int}, Int}()
for (u,v,w) in edge_list
    a, b = idx[u], idx[v]
    add_edge!(g, a, b)
    wmap[(min(a,b), max(a,b))] = w
end

# ---------- MST rapide ----------
function mst_edges(edge_list, idx)
    sorted = sort(edge_list, by = e -> e[3])
    parent = Dict(v => v for v in values(idx))
    rank   = Dict(v => 0 for v in values(idx))
    mst = Set{Tuple{Int,Int}}()

    find(x) = parent[x] == x ? x : (parent[x] = find(parent[x]))
    function union!(x,y)
        rx, ry = find(x), find(y)
        rx == ry && return false
        if rank[rx] < rank[ry]
            parent[rx] = ry
        elseif rank[rx] > rank[ry]
            parent[ry] = rx
        else
            parent[ry] = rx
            rank[rx] += 1
        end
        return true
    end

    for (u,v,_) in sorted
        a, b = idx[u], idx[v]
        if union!(a,b)
            push!(mst, (min(a,b), max(a,b)))
        end
    end
    return mst
end

mst = mst_edges(edge_list, idx)

# ---------- Couleurs et labels ----------
```

```
edge_colors = Colorant[]
edge_labels = String[]
for e in Graphs.edges(g)
    a, b = src(e), dst(e)
    key = (min(a,b), max(a,b))
    push!(edge_labels, string(wmap[key]))
    if key in mst
        push!(edge_colors, colorant"red")
    else
        push!(edge_colors, colorant"gray")
    end
end

# ---------- Affichage ----------
gplot(g;
    nodelabel = nodes,
    edgelabel = edge_labels,
    edgestrokec = edge_colors,
    layout = spring_layout
)
```

LSINC1113 - Compléments de mathématiques

TP6 - Graphes (DAG)

Rappels Théoriques

1. Définitions

Un **graphe orienté acyclique** (Directed Acyclic Graph, ou **DAG**) est un graphe $G = (V, E)$:

- dont les arêtes $(u, v) \in E$ sont **orientées** (elles vont de u vers v),
- et qui ne contient **aucun cycle**, c'est-à-dire qu'il n'existe aucune suite de sommets $v_1, v_2, \ldots, v_k$ telle que :

$$(v_i, v_{i+1}) \in E \quad \text{pour tout } i \text{ et } v_k = v_1.$$

Intuition :
Un DAG, c'est un graphe où l'on peut **avancer** en suivant les flèches, mais **jamais revenir au point de départ**. Il y a donc une idée de *temps* ou de *priorité* : on ne peut jamais faire un tour complet et revenir à une tâche déjà vue.

Exemples typiques :

- tâches avec des dépendances (on doit faire A avant B, B avant C, etc.) ;
- étapes de calcul (on a besoin du résultat précédent pour calculer le suivant).

2. Représentation mathématique

Un DAG est souvent représenté par sa **matrice d'adjacence** A, où :

$$A_{ij} = \begin{cases} 1 & \text{si une arête va de } i \text{ vers } j, \\ 0 & \text{sinon.} \end{cases}$$

Intuition : la ligne i indique « vers qui part i », et la colonne j indique « qui vient vers j ».

L'absence de cycle implique qu'il existe un **ordre topologique** des sommets : un ordre $v_1, v_2, \ldots, v_n$ tel que :

$$(u, v) \in E \;\Rightarrow\; \text{indice}(u) < \text{indice}(v).$$

Autrement dit, chaque arête va d'un sommet **plus tôt** vers un sommet **plus tard** dans cet ordre. On peut aligner les sommets sur une ligne (du plus « ancien » au plus « récent »), et *toutes* les flèches vont de la gauche vers la droite.

3. Programmation dynamique et graphes

Beaucoup de problèmes combinatoires peuvent être vus comme un **parcours dans un DAG implicite** (c'est-à-dire que le graphe existe conceptuellement, même si on ne le dessine pas explicitement).

- Chaque **nœud** représente un **état** du système (par exemple : quels patients sont déjà affectés, quelle place il reste dans le sac à dos, etc.).
- Chaque **arête** représente une **décision** ou une **transition** (par exemple : ajouter un objet, affecter un patient à un infirmier, ouvrir un nouveau sac).

- Comme on avance étape par étape, sans revenir en arrière sur les décisions, le graphe n'a **pas de cycle** ⇒ c'est un **DAG**.

Programmation dynamique (PD) :
La PD revient à **explorer ce DAG intelligemment** :

- on résout d'abord les **petits sous-problèmes** (les nœuds « proches du début »),
- puis on réutilise ces résultats pour les états plus complexes,
- et on **mémorise** ce qu'on a déjà calculé pour ne pas le refaire.

Intuition : chaque état dépend seulement de quelques états précédents : on peut donc remplir un **tableau de DP** en suivant un **ordre topologique** sur les états.

4. Opérations fondamentales

Le **tri topologique** produit un ordre linéaire des sommets qui respecte les dépendances : si $(u, v) \in E$, alors u apparaît avant v dans cet ordre.
Deux façons classiques de le calculer :

- **Par DFS (parcours en profondeur)** : on explore en profondeur, et on ajoute les sommets à la fin de l'ordre lorsqu'ils sont « terminés » ; puis on inverse cet ordre.
- **Algorithme de Kahn** : on répète
  - — prendre un sommet sans prédécesseur (degré entrant nul),
  - — l'ajouter à l'ordre,
  - — supprimer ses arêtes sortantes.

Intuition : on enlève progressivement les tâches qui n'attendent plus rien (pas de prérequis), comme quand on fait la liste des cours qu'on peut suivre en respectant les prérequis.

Dans un DAG, on peut trouver le **chemin le plus long** *efficacement* (ce qui est en général impossible dans un graphe avec cycles, car ce problème est NP-difficile).

On suppose que chaque sommet v a une durée $d(v)$ (temps nécessaire pour accomplir la tâche v), et les arêtes indiquent les **dépendances**.

On définit, pour chaque sommet v :

$$L(v) = d(v) + \max_{u \in \text{pred}(v)} L(u),$$

où $\text{pred}(v)$ est l'ensemble des prédécesseurs de v (les sommets ayant une arête vers v).

Si v n'a pas de prédécesseur, on prend simplement $L(v) = d(v)$.

Intuition :

- $L(v)$ = durée minimale pour **terminer** toutes les tâches nécessaires avant v, *plus* la tâche v elle-même.
- On calcule $L(v)$ en suivant un **ordre topologique** : on est sûr que, quand on traite v, tous ses prédécesseurs u ont déjà un $L(u)$ calculé.

La durée totale minimale du projet (aussi appelée **chemin critique** ou **chaîne critique**) est :

$$\max_{v \in V} L(v).$$

C'est la durée de la séquence de tâches la plus « contraignante » : le chemin qui détermine la durée totale du projet.

5. Applications

Les DAGs apparaissent très souvent en pratique :

- **Planification de tâches** : certaines tâches doivent être finies avant d'en commencer d'autres (prérequis).
- **Ordonnancement** : organiser un projet, un pipeline de production, ou des jobs dans un système.
- **Circuits logiques et flux de données** : un signal suit des portes logiques dans un ordre acyclique.
- **Programmation dynamique** : chaque sous-problème dépend d'autres plus petits, ce qui forme un DAG de dépendances.

Dans beaucoup d'énoncés, le DAG n'est pas dessiné, mais il est « caché » dans la structure du problème.

Intuition : si tu peux raconter le problème comme « on part d'un état initial, on fait une suite de choix, et on avance toujours *vers la droite* sans jamais revenir en arrière », alors il y a souvent un DAG de dépendances derrière.

1. Planification des soins infirmiers

Contexte : Un hôpital dispose d'un certain nombre de patients ayant chacun besoin d'un temps de soins (en heures). Chaque infirmier peut travailler au maximum 8 heures par jour.

Objectif : Affecter les patients aux infirmiers de manière à minimiser le nombre d'infirmiers nécessaires, tout en respectant la contrainte de 8 heures par infirmier.

Données :

Patient	**Temps de soins (h)**
P_1	4
P_2	3
P_3	6
P_4	2
P_5	5

Travail à faire :

1. Représentez le problème sous forme d'un DAG.
2. Justifiez pourquoi ce graphe est un DAG.
3. Implémenter votre solution en Julia.

LSINC1113 - Compléments de mathématiques

Correction TP6 - Graphes (DAG)

---

1. Planification des soins infirmiers

Formulation : On cherche le nombre minimal d'infirmiers n tel que :

$$\forall i \in \{1, ...n\}, \quad \sum_{p \in S_i} t_p \leq C$$

où :

- S_i est l'ensemble des patients pris en charge par l'infirmier i,
- t_p est le temps nécessaire pour le patient p,
- $C = 8$ est la capacité maximale journalière d'un infirmier.

Chaque infirmier est donc un sac, chaque patient un objet, et la durée des soins le poids de l'objet.

Modélisation : Le problème peut être vu comme un *graphe d'états* :

- chaque nœud représente la répartition actuelle (patients déjà affectés et capacités restantes),
- chaque arête correspond à une décision (ajouter un patient dans un infirmier ou en ouvrir un nouveau).

Ce graphe est un **DAG** : à chaque étape, le nombre de patients restants diminue, il est donc impossible de former un cycle. Le but est de trouver le chemin le plus court menant à un état où tous les patients sont affectés.

Programmation dynamique : La PD explore ce DAG implicitement : chaque appel récursif correspond à un nœud (état), et les résultats intermédiaires sont mémorisés pour éviter les recalculs.
On définit un sous-problème :

$$f(S) = \min_{p \in S} (1 + f(S \setminus \{p\}))$$

sous contrainte que la somme des durées ne dépasse pas C. Comme cet espace d'états est trop vaste pour un calcul direct, on adopte une approche plus constructive.

Stratégie algorithmique :

- On commence par la borne inférieure :

$$k_{\min} = \left\lceil \frac{\sum_p t_p}{C} \right\rceil$$

- On teste successivement $k_{\min}, k_{\min} + 1, ...$ jusqu'à trouver une configuration faisable.
- Pour chaque k, une recherche récursive tente d'ajouter les patients dans les bacs existants sans dépasser C.
- Les états déjà explorés (mêmes capacités restantes) sont mémorisés.

Intuition : On explore un DAG implicite où :

- la racine représente l'état initial (aucun patient affecté),
- les arêtes sont les choix d'affectation,
- les feuilles sont les répartitions complètes valides.

Exemple : Pour trois patients [6, 2, 3] et $C = 8$:

$$[8, 8, 8] \to [2, 8, 8] \to [0, 8, 8] \to [0, 5, 8]$$

Succès : toutes les tâches sont affectées sans dépasser la capacité.

```
# Données
patients = ["P1","P2","P3","P4","P5"]
times = [4,3,6,2,5]
const C = 8

# Test de faisabilité : peut-on placer 'times' dans k bacs de capacité C ?
function feasible_with_k(times::Vector{Int}, k::Int, C::Int)
    items = sort(times; rev=true)
    caps0 = fill(C, k)
    cache = Dict{Tuple{Int,NTuple{Int,Int}}, Bool}()

    function dfs(i::Int, caps::Vector{Int})
        i > length(items) && return true
        key = (i, tuple(sort(caps; rev=true)...))
        if haskey(cache, key); return cache[key]; end
        used = Set{Int}()
        w = items[i]
        for b in eachindex(caps)
            cap = caps[b]
            if cap >= w && !(cap in used)
                used |= Set([cap])
                caps[b] -= w
                if dfs(i+1, caps)
                    cache[key] = true
                    caps[b] += w
                    return true
                end
                caps[b] += w
            end
        end
        cache[key] = false
        return false
    end

    return dfs(1, copy(caps0))
end

# Recherche du nombre minimal d'infirmiers
lower = cld(sum(times), C)
best_k = nothing
for k in lower:length(times)
    if feasible_with_k(times, k, C)
        best_k = k
        break
    end
end

println("Nombre minimal d'infirmiers nécessaires : ", best_k)
```

Résultat attendu :

```
1 Nombre minimal d'infirmiers nécessaires : 3
```

Répartition possible :

- Infirmier 1 : 6h + 2h
- Infirmier 2 : 4h + 3h
- Infirmier 3 : 5h

LSINC1113 - Compléments de mathématiques

TP7 - Graphes (shortest path)

---

1. Idée générale

On a un patient avec plusieurs **pathologies** (maladies) et une liste de **médicaments**. Chaque médicament :

- **guérit** certaines pathologies,
- peut créer des **effets secondaires** (de nouvelles pathologies),
- a un **prix** (coût ≥ 0).

On cherche une **suite de médicaments** qui :

- guérit **toutes** les pathologies,
- pour un **coût total minimal**.

Pour raisonner proprement, on voit le problème comme un **graphe d'états** :

- un **sommets** = un **état du patient** (le "set" de pathologies encore actives),
- une **arête** = l'application d'un médicament qui transforme un état en un autre,
- le **poids de l'arête** = le **prix** de ce médicament (coût ≥ 0).

L'état **initial** = les pathologies de départ, l'état **final** désiré = l'état **0** = « plus aucune pathologie ».

$\Rightarrow$ On cherche le **plus court chemin** (au sens du coût total) entre **l'état initial** et l'état **0**.

C'est exactement un **problème de plus court chemin dans un graphe pondéré à coûts positifs** $\Rightarrow$ algorithme de **Dijkstra**.

2. Modélisation

Pour pouvoir le coder, on a besoin d'une représentation compacte des états.

▷ **États :** On code les pathologies par des bits dans un entier :

$$s \in \{0,1\}^n$$

(un **bitset**), où chaque bit vaut 1 si la pathologie correspondante est présente, 0 sinon. Exemple : $s = 1010_2$ signifie que les pathologies 1 et 3 sont actives.

▷ **Transition $s \to s'$ par un médicament m :**

— m.`cures` = bitset des pathologies que le médicament guérit,

— m.`side_effects` = bitset des pathologies qu'il ajoute.

Alors le nouvel état est :

$$s' \;=\; (s \;\&\; \sim m.\texttt{cures}) \;\mid\; m.\texttt{side\_effects}.$$

(On **retire** les pathologies guéries, puis on **ajoute** les effets secondaires.)

▷ **Coût de la transition :**

$$c(s, m) = m.\texttt{price} \geq 0.$$

▷ **Objectif :** partir de l'état initial s_0 et atteindre l'état 0 (aucune pathologie) avec un **coût total minimal**.

3. Pourquoi Dijkstra marche ici ?

L'algorithme de Dijkstra suppose que **tous les coûts d'arêtes sont** ≥ 0. C'est le cas ici : **un médicament ne "rend jamais de l'argent"**, il coûte toujours au moins 0.

1. **Les coûts augmentent toujours.** Chaque fois qu'on prend un médicament, on **ajoute** son prix au coût total. Faire des détours ou tourner en rond ne pourra donc **qu'augmenter** le coût.
2. **Premier sorti = coût optimal.** Dijkstra maintient une file de priorité avec les états atteints, triés par **coût cumulé**. Quand un état s sort de la file (c'est le « moins cher » restant), on sait qu'il n'existe **aucun chemin moins cher** pour y arriver. On peut donc **geler** son coût : il ne changera plus.
3. **On peut revisiter un état, mais seulement si c'est moins cher.** Il est possible qu'un autre chemin mène plus tard au **même état** s (mêmes pathologies restantes). On ne le garde que si le coût est **plus petit** que celui qu'on connaissait déjà. Sinon, on l'ignore : **un cycle avec des coûts positifs ne pourra jamais améliorer le prix**.

En particulier : dès que Dijkstra atteint l'état 0 (dans la file de priorité), le coût associé est **garanti minimal**.

4. Recette pour coder Dijkstra (3 grandes étapes)

1. **Initialisation**
   - Pour chaque état possible s, on stocke un **meilleur coût connu** $dist[s]$ (au début, ∞ partout).
   - On met $dist[s_0] = 0$ pour l'état initial.
   - On crée une **file de priorité** (min-heap) contenant $(0, s_0)$.
   - (Optionnel mais utile) On garde aussi un tableau $parent[s]$ pour mémoriser *d'où* vient s et *quel médicament* a été utilisé.
2. **Boucle principale**
   - Tant que la file n'est pas vide :
     (a) extraire (coût, s), l'état au **coût minimal** restant ;
     (b) si $s = 0$, on peut s'arrêter : on a trouvé le **coût minimal** pour guérir toutes les pathologies ;
     (c) sinon, pour chaque médicament m :
     — calculer le nouvel état s' après application de m,
     — coût candidat : $new\_cost = dist[s] + m.\texttt{price}$,
     — si $new\_cost < dist[s']$, alors on a trouvé un chemin **plus intéressant** vers s' :
     — mettre à jour $dist[s']$,
     — mettre $parent[s']$,
     — insérer (new_cost, s') dans la file.
3. **Reconstruction de la solution**
   - Si $dist[0]$ est fini, on peut retrouver la séquence de médicaments en partant de $s = 0$ et en remontant les $parent[s]$ jusqu'à s_0.
   - On obtient la suite **dans le bon ordre** en renversant la liste.

5. Quelques astuces

- **No-op :** Si un médicament ne change pas l'état (on a $s' = s$), il ne sert à rien : on peut l'ignorer dans cet état.
- **Effets secondaires :** Un médicament peut **réintroduire** des pathologies. On peut donc revisiter des états plus « malades » qu'avant, mais ça coûtera toujours **plus cher** (prix supplémentaires). Donc, au pire, on tourne en rond avec un coût qui augmente, mais ça ne peut pas améliorer une solution.
- **Ce qu'il faut retenir vraiment :**
  - **état = pathologies restantes**, codées en bits ;
  - **transition = médicament**, qui modifie l'état et coûte un certain prix ;
  - comme tous les coûts sont ≥ 0, **Dijkstra** assure que la première fois qu'on sort un état d'un certain coût, c'est le **meilleur** possible.

En une phrase :
Dijkstra = "je prends toujours l'état atteignable le moins cher, je le fige, et j'essaie d'améliorer ses voisins" jusqu'à ce que l'état 0 *soit atteint.*

LSINC1113 - Compléments de mathématiques

TP8 - Graphes (max-flow)

---

1. L'intuition

Imagine un **réseau de tuyaux** :

- une **source** S où le flux entre,
- un **puits** T où le flux sort,
- des **arêtes orientées** (tuyaux) avec une **capacité** maximale (débit possible).

On veut savoir :

> « Combien de liquide au maximum peut-on faire passer de S à T sans casser de tuyau ? »

C'est ça, le Max Flow.

C'est exactement le **problème de flot maximum (Max Flow)**. Le **flot** modélise :

- de l'eau dans des tuyaux,
- de la marchandise dans un réseau de routes,
- des tâches qui passent d'une étape à l'autre,
- des appariements (matching) entre deux ensembles.

2. Modèle et contraintes

On a un graphe orienté $G = (V, E)$, une source S, un puits T et une capacité $c(u, v) \geq 0$ sur chaque arête (u, v).

Un **flot** $f(u, v)$ doit respecter deux règles :

1. **Capacité** : on ne peut pas dépasser la capacité :

$$0 \leq f(u, v) \leq c(u, v) \quad \text{pour toute arête } (u, v).$$

2. **Conservation du flot** : pour tous les sommets sauf S et T, ce qui *rentre* doit *sortir* :

$$\sum_u f(u, v) = \sum_w f(v, w) \quad \text{pour tout } v \neq S, T.$$

La **valeur du flot** est la quantité qui sort de la source (ou qui arrive au puits) :

$$|f| = \sum_v f(S, v).$$

Le problème Max Flow est :

> Trouver un flot de valeur maximale $|f|$ respectant ces contraintes.

3. Idée de l'algorithme (Ford–Fulkerson / Edmonds–Karp)

Intuition : on commence avec un flot nul, puis on « pousse » du flux petit à petit dans le réseau, tant qu'on trouve encore un chemin avec de la place.

1. Chercher un **chemin de S à T** où il reste de la capacité disponible (on l'appelle **chemin augmentant**).
2. On envoie sur ce chemin la **quantité maximale possible** (le minimum des capacités restantes le long du chemin).
3. On **met à jour** le **graphe résiduel** :
   - sur les arêtes utilisées, la capacité restante diminue,
   - on ajoute des arêtes « en sens inverse » qui permettent éventuellement de **retirer du flot** si on change d'avis plus tard.
4. On recommence tant qu'il existe un chemin augmentant de S à T.

Quand il n'y a plus de chemin augmentant, le flot obtenu est **maximal**.

Graphe résiduel (idée simple) :
Pour chaque arête (u, v) avec capacité $c(u, v)$ et flot $f(u, v)$:

- capacité résiduelle **forward** : $c(u, v) - f(u, v)$ (ce qu'on peut encore **ajouter**),
- capacité résiduelle **backward** : $f(u, v)$ (ce qu'on peut **retirer** en rembobinant).

Le graphe résiduel te dit : « à partir du flot actuel, où puis-je encore augmenter ou diminuer le flot ? »

4. Lien fondamental : Max Flow = Min Cut

Une **coupe** (S, T) est une façon de séparer les sommets en deux groupes :

- un côté contenant S,
- l'autre côté contenant T.

La **capacité de la coupe** est la somme des capacités des arêtes qui vont du côté de S vers le côté de T.

Théorème Max Flow = Min Cut :

- la valeur du **flot maximal** de S à T est égale
- à la **capacité minimale** d'une coupe séparant S et T.

Intuition :

- le Min Cut = le **goulot d'étranglement** du réseau,
- le Max Flow = combien tu peux vraiment faire passer,
- ces deux nombres coïncident.

5. Comment reconnaître un exercice de flot ?

Tu penses à un problème de Max Flow quand :

- on veut **maximiser une quantité** qui « circule » dans un réseau (eau, trafic, marchandises, tâches, appariements...),
- il y a une **source** naturelle (départ) et un **puits** (arrivée),
- chaque lien a une **capacité** (max par lien, ou max par personne / machine),
- on ne veut pas **doubler** une ressource (un employé ne peut traiter qu'un scanner, etc.).

Recette dans un exo :

1. Construire le graphe : choisir S, T, les sommets intermédiaires, les capacités.
2. Appliquer un algorithme de Max Flow (Ford–Fulkerson / Edmonds–Karp).
3. Lire la valeur du flot maximal, puis **traduire** dans le langage de l'énoncé.

6. Exemple : scanners et employés

On veut utiliser un maximum de scanners en parallèle, en assignant chaque scanner à un seul employé, et chaque employé à un seul scanner compatible.

Exemple :

$$\begin{aligned} S_1[\text{Scanner 1}] &\to E_1[\text{Employé A}] \\ S_1 &\to E_2[\text{Employé B}] \\ S_2[\text{Scanner 2}] &\to E_1 \\ S_2 &\to E_2 \\ S_3[\text{Scanner 3}] &\to E_2 \\ S_4[\text{Scanner 4}] &\to E_2 \\ S_4 &\to E_4[\text{Employé D}] \\ S_4 &\to E_5[\text{Employé E}] \\ S_5[\text{Scanner 5}] &\to E_1 \\ S_5 &\to E_3[\text{Employé C}] \\ S_5 &\to E_5 \end{aligned}$$

Étape 1 : construire le graphe de flot

- Source S.
- Un sommet pour chaque **scanner** S_i.
- Un sommet pour chaque **employé** E_j.
- Puits T.

Arêtes et capacités :

- $S \to S_i$ de capacité 1 (chaque scanner au plus une fois),
- $S_i \to E_j$ de capacité 1 si l'employé E_j sait utiliser le scanner S_i,
- $E_j \to T$ de capacité 1 (un employé ne fait qu'un scan à la fois).

Max Flow = nombre maximal de paires (scanner, employé) utilisables en parallèle.

Dans l'exemple donné, on arrive à un flot de valeur 4 : on peut utiliser **4 scanners en même temps**, pas 5.

Pourquoi pas 5 ? Intuition par la coupe minimale :
Les scanners S_1, S_2, S_3 ne peuvent être utilisés que par E_1 ou E_2. Or il n'y a que **2** employés E_1 et E_2 pour ces trois scanners. Donc au plus **2** scanners parmi $\{S_1, S_2, S_3\}$ peuvent fonctionner en même temps.
Même si S_4 et S_5 peuvent être affectés à d'autres employés (E_3, E_4, E_5), on obtient au total :

$$2 \text{ (bloqués par } E_1, E_2) + 2 \text{ (autres)} = 4 \text{ scanners maximum.}$$

On peut formaliser cela comme une **coupe** de capacité 4 (par exemple en séparant $\{S, S_1, S_2, S_3, E_1, E_2\}$ du reste du graphe). Par le théorème Max Flow = Min Cut, le flot maximum vaut donc aussi 4.

On recherche les sommets atteignables depuis la source dans le graphe résiduel. On démarre de S et on suit les arêtes résiduelles de capacité > 0. Dans notre cas, les arêtes S $\to$ S_1,S_3,S_4,S_5

sont saturées (flux=1) → forward residuel = 0 (donc non utilisables). L'arête S → S_2 n'est pas saturée (flux=0) → forward residuel = 1 → S_2 est atteignable depuis S.

Ensuite, depuis E_1 on peut revenir (backward) vers S_1 (car $S_1 \rightarrow E_1$ a flux 1), et depuis E_2 on peut revenir vers S_3 (car $S_3 \rightarrow E_2$ a flux 1). En suivant toutes ces arêtes résiduelles on obtient l'ensemble atteignable : R = S, S_2, E_1, E_2, S_1, S_3. Les sommets non atteignables (complément) sont : S_4, S_5, E_3, E_4, E_5, T.

La coupe est l'ensemble des arêtes orientées du côté atteignable vers le côté non-atteignable dans le graphe original. Ce sont :

- S → S_4 (S atteignable, S_4 non-atteignable)
- S → S_5 (S atteignable, S_5 non-atteignable)
- E_1 → T (E_1 atteignable, T non-atteignable)
- E_2 → T (E_2 atteignable, T non-atteignable)

Chaque arête a une capacité de 1, donc la capacité totale de cette coupe = 1+1+1+1 = 4.

La coupe montre les scanners du côté source non utilisables (S_4,S_5) et les employés du côté puits non utilisés (E_3,E_4,E_5). Pour augmenter la capacité (faire passer la valeur du min-cut / max-flow à 5), il faut ajouter au moins une arête qui connecte un scanner du côté source (ici S_4 ou S_5) à un employé du côté puits (E_3, E_4 ou E_5) qui n'était pas capable de l'utiliser avant, c'est-à-dire former un employé à un scanner de façon à « traverser la coupe ». Concrètement, former par exemple E_3 à S_4 ou E_5 à $S_1/S_2/S_3$ selon la topologie — mais note le raisonnement : relier un nœud dans le côté S à un nœud dans le côté T supprime une arête de la coupe et peut augmenter le max flow.

À retenir :

- on transforme un problème d'affectation en **réseau de flot**,
- le **Max Flow** donne le nombre maximal de couples (scanner, employé),
- la **Min Cut** montre où se trouve le « goulot d'étranglement » du système.

Partie 5 - Transformée de Fourier

LSINC1113 - Compléments de mathématiques

TP9 - La transformée de Fourier

Rappels théoriques

Un signal $x(t)$ peut se voir de deux manières complémentaires :

- **Domaine temporel** : on regarde comment $x(t)$ évolue au cours du temps (forme de la courbe, impulsions, décroissance, etc.).
- **Domaine fréquentiel** : on regarde de quelles **fréquences** il est composé (quelles sinusoïdes, avec quelle amplitude, etc.).

Exemples :

- En audio, $x(t)$ est le son dans le temps, et $X(\omega)$ montre les graves / médiums / aigus (comme un égaliseur).
- En traitement d'images, la Transformée de Fourier (TF) permet de voir quelles "textures" (fréquences spatiales) sont présentes.

La transformée de Fourier sert à **passer du temps à la fréquence**, et la transformée inverse permet de **reconstruire le signal temporel à partir du spectre de fréquences**.

1. Signaux Périodiques

Un signal $x(t)$ est **périodique** si :

$$x(t+T) = x(t)$$

pour une période $T > 0$. On définit :

$$f = \frac{1}{T} \quad \text{et} \quad \omega = 2\pi f = \frac{2\pi}{T}.$$

Exemples :

$$\cos(\omega t) \text{ et } \sin(\omega t)$$

sont périodiques avec $T = \frac{2\pi}{\omega}$.

2. Transformée de Fourier (TF) et TF inverse

La transformée de Fourier d'un signal $x(t)$ est :

$$X(\omega) = \int_{-\infty}^{\infty} x(t) e^{-i\omega t}\, dt$$

- $x(t)$: signal dans le temps,
- $X(\omega)$: même information vue dans le domaine fréquentiel.

On peut revenir dans le temps grâce à la transformée inverse :

$$x(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} X(\omega) e^{i\omega t} \, d\omega.$$

Intuition :

- $X(\omega)$ dit "combien de fréquence ω il y a dans le signal".
- La TF inverse "remixe" toutes ces sinusoïdes pour reconstruire $x(t)$.

3. Signaux de base : $u(t)$, Dirac, sinc

Fonction échelon $u(t)$ (Heaviside)

$$u(t) = \begin{cases} 0 & t < 0, \\ 1 & t \geq 0. \end{cases}$$

C'est un "interrupteur" qui allume le signal à partir de $t = 0$. Par exemple $e^{-at}u(t)$ est une exponentielle décroissante **causale** (qui commence à $t = 0$).

Dirac $\delta(t)$ (impulsion)
La fonction $\delta(t)$ est nulle partout sauf en $t = 0$, où elle est "infiniment piquée", avec :

$$\int_{-\infty}^{\infty} \delta(t) \, dt = 1.$$

Elle sert à modéliser une impulsion idéale (un "coup" instantané). Propriété clé :

$$f(t) \, \delta(t - a) = f(a) \, \delta(t - a).$$

Fonction sinc
En traitement du signal, on utilise souvent :

$$\mathrm{sinc}(x) = \frac{\sin x}{x}, \quad \mathrm{sinc}(0) = 1$$

(par continuité).

Elle vaut 1 au centre, oscille et décroît pour $|x|$ grand, avec des zéros en $x = \pm\pi, \pm 2\pi, \ldots$

Elle apparaît tout le temps en Fourier :

- une **porte en temps** (signal non nul sur un intervalle fini) a une TF en forme de sinc,
- une **sinc en temps** donne une TF en forme de **porte** (signal bande limitée).

Idée importante : couper un signal dans le temps (limiter sa durée) permet de tout faire rentrer, mais cela "étale" son spectre en fréquence avec des oscillations (sinc).

4. Transformées de Fourier usuelles

Domaine temporel	Domaine fréquentiel
$x(t) = u(t)$	$X(\omega) = \dfrac{1}{i\omega} + \pi\delta(\omega)$
$x(t) = \delta(t)$	$X(\omega) = 1$
$x(t) = 1$	$X(\omega) = 2\pi\delta(\omega)$
$x(t) = \dfrac{1}{\pi t}\sin(Wt)$	$X(\omega) = \begin{cases} 1 & \text{si } \lvert\omega\rvert \leq W \\ 0 & \text{sinon.} \end{cases}$
$x(t) = e^{-at}u(t), a > 0$	$X(\omega) = \dfrac{1}{a + i\omega}$

5. Propriétés de la Transformée de Fourier

Linéarité

$$\mathcal{F}\{ax_1(t) + bx_2(t)\} = aX_1(\omega) + bX_2(\omega).$$

Modulation (translation en fréquence)
Si on multiplie le signal par une exponentielle :

$$\mathcal{F}\{x(t)e^{i\omega_0 t}\} = X(\omega - \omega_0),$$

on **décale le spectre** autour de ω_0. C'est le principe de la **modulation** (radio, transmission, etc.).

Convolution

$$\mathcal{F}\{x_1 * x_2\} = X_1(\omega)X_2(\omega).$$

Convolution dans le temps $\Rightarrow$ produit en fréquence (filtrage).

Dualité

$$\mathcal{F}\{x(t)\} = X(\omega) \implies \mathcal{F}\{X(t)\} = 2\pi x(-\omega).$$

6. Formules d'Euler

Les formules d'Euler relient les fonctions trigonométriques aux exponentielles complexes :

$$\cos(\omega t) = \frac{1}{2}(e^{i\omega t} + e^{-i\omega t}), \qquad \sin(\omega t) = \frac{1}{2i}(e^{i\omega t} - e^{-i\omega t}).$$

Elles permettent de réécrire les cos/sin en exponentielles, ce qui simplifie énormément les calculs de TF.

1. Périodicité

Exercice 28. Déterminer si les signaux suivants sont périodiques et donner, le cas échéant, la période T, la fréquence f, ainsi que la vitesse angulaire ω :

(a) $x(t) = \cos(2t) + \sin(3t)$

(b) $x(t) = \cos(t)u(t)$

(c) $x(t) = v(t) + v(-t)$ avec $v(t) = \sin(t)u(t)$

2. Transformée de Fourier

Exercice 29. Calculer la transformée de Fourier des signaux suivants :

(a) $x(t) = e^{-at}u(t), \ \ a > 0$

(b) $x(t) = e^{-t}\cos(2\pi t)u(t)$

(c) $x(t) = \begin{cases} 1 & |t| < T_1 \\ 0 & \text{sinon} \end{cases}$

(d) $x(t) = \frac{1}{\pi t}\sin(Wt)$

(e) (*) $x(t) = e^{-t+2}u(t-2)$

(f) (*) $x(t) = e^{-a|t|}$

Exercice 30. Calculer les signaux qui ont comme transformée de Fourier les fonctions suivantes :

(a) $X(i\omega) = e^{-2\omega}u(\omega)$

(b) $X(i\omega) = \begin{cases} \cos(2\omega) & \text{si } |\omega| < \pi/4 \\ 0 & \text{sinon.} \end{cases}$

(c) $X(i\omega) = \begin{cases} 1 & \text{pour } |\omega| < W \\ 0 & \text{pour } |\omega| > W \end{cases}$

3. Lecture et tracé de spectres

Exercice 31. Tracer à la main (amplitude uniquement) l'allure de $|X(\omega)|$ pour les signaux suivants.

(a) $x(t) = \mathbf{1}_{|t|<T_0}$

(b) $x(t) = \dfrac{1}{\pi t}\sin(Wt)$

(c) $x(t) = \delta(t)$

(d) $x(t) = 1$

4. Exercices supplémentaires

Exercice 32. Déterminer si les signaux suivants sont périodiques et donner, le cas échéant, la période fondamentale, la fréquence, ainsi que la vitesse angulaire :

(a) $x(t) = \sum_{k=-\infty}^{\infty}(-1)^k\delta(t-2k)$

(b) $x(t) = v(t) + v(-t)$ avec $v(t) = \cos(t)u(t)$

Exercice 33. Calculer la transformée de Fourier des signaux suivants :

(a) $x(t) = e^{-2t}\sin(3\pi t)u(t)$
(b) $x(t) = e^{3t}\cos(5\pi t)u(-t)$
(c) $x(t) = e^{-t^2}$
(d) $x(t) = te^{-t}u(t)$
(e) $x(t) = \delta(t-3)$
(f) $x(t) = \frac{1}{t^2+1}$
(g) $x(t) = e^{-bt}u(t)$, $b > 0$
(h) $x(t) = \sin(\pi t)u(t)$
(i) $x(t) = e^{-t}\cos(2\pi t)u(t-1)$
(j) $x(t) = \delta'(t)$

Exercice 34. Calculer les signaux qui ont comme transformée de Fourier les fonctions suivantes :

(a) $X(i\omega) = \frac{1}{\omega^2+9}$
(b) $X(i\omega) = e^{-\omega^2}$
(c) $X(i\omega) = \frac{1}{i\omega+1}$
(d) $X(i\omega) = \sin(2\omega)u(\omega)$
(e) $X(i\omega) = \delta(\omega-5)$
(f) $X(i\omega) = \frac{1}{\omega^2+a^2}$, $a > 0$
(g) $X(i\omega) = e^{-\omega}u(\omega)$
(h) $X(i\omega) = \frac{1}{1+\omega^2}$

LSINC1113 - Compléments de mathématiques

Correction TP9 - Transformée de Fourier

---

1. Périodicité

Solution 28.

(a) $x(t) = \cos(2t) + \sin(3t)$

$\cos(2t)$ a pour période $T_1 = \pi$ (car $2(t + T_1) = 2t + 2\pi$).
$\sin(3t)$ a pour période $T_2 = \dfrac{2\pi}{3}$.

La somme est périodique si les deux périodes ont un PPCM :

$$\mathrm{PPCM}\Big(\pi, \frac{2\pi}{3}\Big) = 2\pi.$$

Donc $x(t)$ est **périodique** de période fondamentale $T = 2\pi$.

$$f = \frac{1}{T} = \frac{1}{2\pi}, \qquad \omega = 2\pi f = 1.$$

(b) $x(t) = \cos(t)u(t)$

$\cos(t)$ seul a une période 2π, mais $u(t)$ vaut 0 pour $t < 0$ et 1 pour $t \geq 0$.
Le produit tronque la sinusoïde pour $t < 0$, donc la forme globale n'est pas répétée périodiquement :

$$x(t + T) \neq x(t) \quad \text{pour tout } T > 0.$$

$x(t)$ **n'est pas périodique**.

(c) $x(t) = v(t) + v(-t)$ avec $v(t) = \sin(t)u(t)$

On a

$$v(t) = \sin(t)u(t), \quad v(-t) = \sin(-t)u(-t) = -\sin(t)u(-t).$$

Donc

$$x(t) = \sin(t)u(t) - \sin(t)u(-t).$$

Pour $t > 0$, $u(t) = 1$ et $u(-t) = 0$, donc $x(t) = \sin(t)$.
Pour $t < 0$, $u(t) = 0$ et $u(-t) = 1$, donc $x(t) = -\sin(t)$.

On obtient un signal **impair** :

$$x(t) = \begin{cases} -\sin(t) & t < 0, \\ 0 & t = 0, \\ \sin(t) & t > 0. \end{cases}$$

Ce n'est pas une sinusoïde complète répétée : la définition change autour de 0.

En particulier,

$$x(-\tfrac{\pi}{2}) = -\sin(-\tfrac{\pi}{2}) = 1, \quad x(-\tfrac{\pi}{2} + 2\pi) = x(\tfrac{3\pi}{2}) = \sin(\tfrac{3\pi}{2}) = -1.$$

Donc $x(t + 2\pi) \neq x(t)$ pour tout t, le signal **n'est pas périodique**.

2. Transformée de Fourier

Solution 29.

(a) $x(t) = e^{-at}u(t),\ \ a > 0$

$$\begin{aligned} X(\omega) &= \int_{-\infty}^{\infty} e^{-at}u(t)e^{-i\omega t}\,dt \\ &= \int_{0}^{\infty} e^{-(a+i\omega)t}\,dt \\ &= \left[\frac{-1}{a+i\omega}e^{-(a+i\omega)t}\right]_0^{\infty} \\ &= \frac{1}{a+i\omega}. \end{aligned}$$

(b) $x(t) = e^{-t}\cos(2\pi t)u(t)$

On utilise les exponentielles complexes :

$$\cos(2\pi t) = \frac{1}{2}\left(e^{i2\pi t} + e^{-i2\pi t}\right),$$

donc

$$x(t) = \frac{1}{2}e^{-t}e^{i2\pi t}u(t) + \frac{1}{2}e^{-t}e^{-i2\pi t}u(t).$$

La TF de $e^{-t}u(t)$ est $\dfrac{1}{1+i\omega}$ (cas $a = 1$).

La multiplication par $e^{\pm i2\pi t}$ décale le spectre :

$$\mathcal{F}\{x(t)e^{i\omega_0 t}\} = X(\omega - \omega_0).$$

On obtient :

$$X(\omega) = \frac{1}{2}\left[\frac{1}{1+i(\omega-2\pi)} + \frac{1}{1+i(\omega+2\pi)}\right].$$

On peut également le calculer comme suit :

$$\begin{aligned} X(\omega) &= \int_{-\infty}^{\infty} (\frac{1}{2}e^{-t}e^{i2\pi t}u(t) + \frac{1}{2}e^{-t}e^{-i2\pi t}u(t))e^{-i\omega t}\,dt \\ &= \int_{-\infty}^{\infty} \frac{1}{2}e^{-t}e^{i2\pi t}u(t)e^{-i\omega t}\,dt + \int_{-\infty}^{\infty} \frac{1}{2}e^{-t}e^{-i2\pi t}u(t)e^{-i\omega t}\,dt \\ &= \int_{-\infty}^{\infty} \frac{1}{2}e^{-t+i2\pi t-i\omega t}u(t)\,dt + \int_{-\infty}^{\infty} \frac{1}{2}e^{-t-i2\pi t-i\omega t}u(t)\,dt \\ &= \int_{0}^{\infty} \frac{1}{2}e^{-t+i2\pi t-i\omega t}\,dt + \int_{0}^{\infty} \frac{1}{2}e^{-t-i2\pi t-i\omega t}\,dt \\ &= \frac{1}{2}\left[\frac{e^{-t(1-i2\pi+i\omega)}}{1-i2\pi+i\omega}\right]_0^{\infty} + \frac{1}{2}\left[\frac{e^{-t(1+i2\pi+i\omega)}}{1+i2\pi+i\omega}\right]_0^{\infty} \\ &= \frac{1}{2}\left[\frac{1}{1+i(\omega-2\pi)} + \frac{1}{1+i(\omega+2\pi)}\right] \end{aligned}$$

(c) $x(t) = \begin{cases} 1 & |t| < T_1 \\ 0 & \text{sinon} \end{cases}$

$$\begin{aligned} X(\omega) &= \int_{-T_1}^{T_1} e^{-i\omega t}\,dt = \left[\frac{e^{-i\omega t}}{-i\omega}\right]_{-T_1}^{T_1} \\ &= \frac{e^{-i\omega T_1} - e^{i\omega T_1}}{-i\omega} = \frac{-2i\sin(\omega T_1)}{-i\omega} \\ &= 2T_1\,\frac{\sin(\omega T_1)}{\omega T_1} = 2T_1\,\mathrm{sinc}(\omega T_1). \end{aligned}$$

(d) $x(t) = \dfrac{1}{\pi t}\sin(Wt)$

C'est exactement le cas dual de (c). On admet (ou on retrouve par intégrale) que :

$$X(\omega) = \begin{cases} 1 & |\omega| < W, \\ 0 & \text{sinon.} \end{cases}$$

On va donc en fait résoudre cette intégrale :

$$X(\omega) = \int_{-\infty}^{\infty} \frac{1}{\pi t}\sin(Wt)e^{-i\omega t}\,dt$$

(*) $x(t) = e^{-t+2}u(t-2)$

On reconnaît une version **décalée** de $e^{-t}u(t)$:

$$x(t) = e^2\, e^{-t}u(t-2).$$

On peut écrire $x(t) = e^2 y(t-2)$ avec $y(t) = e^{-t}u(t)$. La translation dans le temps donne :

$$\mathcal{F}\{y(t-t_0)\} = e^{-i\omega t_0}Y(\omega).$$

Donc, avec $Y(\omega) = \dfrac{1}{1+i\omega}$,

$$X(\omega) = e^2 e^{-i\omega 2}\frac{1}{1+i\omega}.$$

(*) $x(t) = e^{-a|t|}$

On sépare $t > 0$ et $t < 0$:

$$e^{-a|t|} = \begin{cases} e^{-at} & t > 0, \\ e^{at} & t < 0. \end{cases}$$

On a un signal **pair**. En posant l'intégrale et en l'évaluant (ou en utilisant une table), on obtient :

$$X(\omega) = \int_{-\infty}^{\infty} e^{-a|t|}e^{-i\omega t}dt = \frac{2a}{a^2+\omega^2}.$$

Solution 30.

(a) $X(\omega) = e^{-2\omega}u(\omega)$

$$\begin{aligned} x(t) &= \frac{1}{2\pi}\int_{-\infty}^{\infty} e^{-2\omega}u(\omega)e^{i\omega t}\,\mathrm{d}\omega \\ &= \frac{1}{2\pi}\int_{0}^{\infty} e^{\omega(-2+it)}\,\mathrm{d}\omega \\ &= \frac{1}{2\pi}\cdot\frac{-1}{-2+it} \\ &= \frac{1}{2\pi}\cdot\frac{-1}{-2+it}\cdot\frac{-2-it}{-2-it} \\ &= \frac{1}{2\pi}\cdot\frac{2+it}{4+t^2} \end{aligned}$$

(b) $X(\omega) = \begin{cases} \cos(2\omega) & \text{si } |\omega| < \pi/4 \\ 0 & \text{sinon.} \end{cases}$

$$\begin{aligned} x(t) &= \frac{1}{2\pi} \int_{-\pi/4}^{\pi/4} \cos(2\omega) e^{i\omega t} \, \mathrm{d}\omega \\ &= \frac{1}{4\pi} \int_{-\pi/4}^{\pi/4} (e^{2i\omega} + e^{-2i\omega}) e^{i\omega t} \, \mathrm{d}\omega \\ &= \frac{1}{4\pi} \int_{-\pi/4}^{\pi/4} \left(e^{(2i+it)\omega} + e^{(-2i+it)\omega} \right) \mathrm{d}\omega \\ &= \frac{1}{4\pi} \left[\frac{1}{(2+t)i} \left(e^{i(2+t)\frac{\pi}{4}} - e^{-i(2+t)\frac{\pi}{4}} \right) + \frac{1}{(-2+t)i} \left(e^{i(-2+t)\frac{\pi}{4}} - e^{-i(-2+t)\frac{\pi}{4}} \right) \right] \\ &= \frac{1}{2\pi} \left(\frac{1}{2+t} \sin\left((2+t)\frac{\pi}{4} \right) + \frac{1}{-2+t} \sin\left((-2+t)\frac{\pi}{4} \right) \right) \end{aligned}$$

(c) $X(\omega) = \begin{cases} 1 & \text{pour } |\omega| < W \\ 0 & \text{pour } |\omega| > W \end{cases}$

$$\begin{aligned} x(t) &= \frac{1}{2\pi} \int_{-W}^{W} e^{i\omega t} \, \mathrm{d}\omega \\ &= \frac{1}{2it\pi} \left(e^{itW} - e^{-itW} \right) \\ &= \frac{1}{t\pi} \sin(tW) \end{aligned}$$

3. Lecture et tracé de spectres

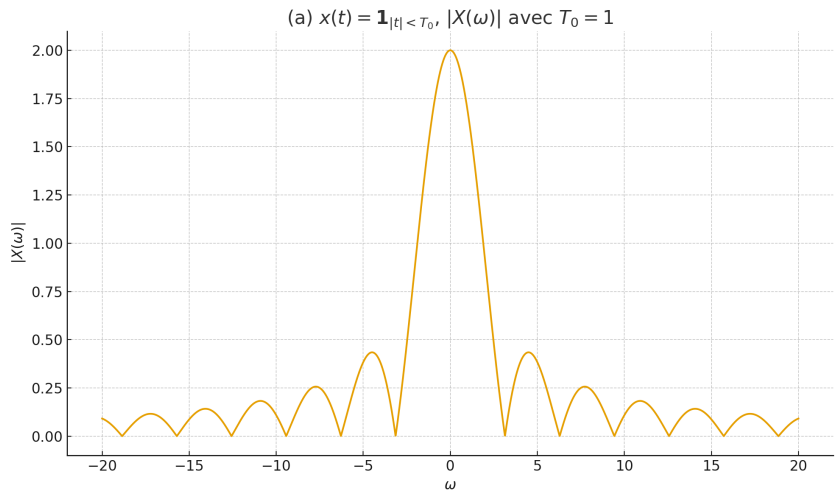
Solution 31. Tracer l'allure de $|X(\omega)|$.

(a) $x(t) = \mathbf{1}_{|t|<T_0}$

On a vu que :

$$X(\omega) = 2T_0 \operatorname{sinc}(\omega T_0).$$

Allure : lobe principal centré en 0, décroissant, avec des zéros à $\omega = \pm k \frac{\pi}{T_0}$.

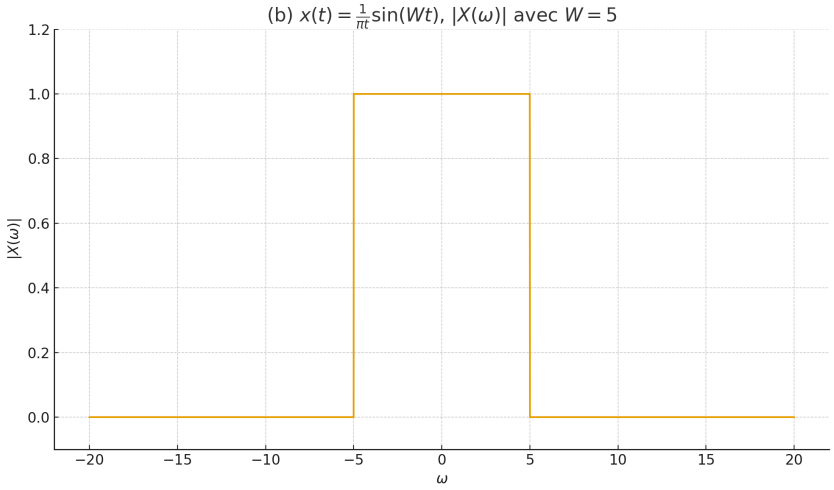


(b) $x(t) = \frac{1}{\pi t} \sin(Wt)$

TF rectangulaire :

$$X(\omega) = 1 \text{ pour } |\omega| < W, \quad 0 \text{ sinon.}$$

Allure : un rectangle de hauteur 1 entre $-W$ et $+W$.

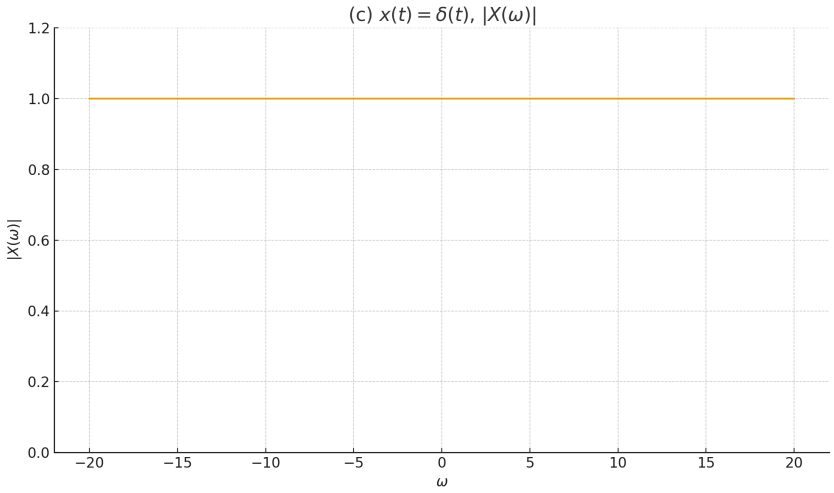


(c) $x(t) = \delta(t)$

TF constante :

$$X(\omega) = 1.$$

Allure : ligne horizontale constante (spectre “plat”).



(d) $x(t) = 1$

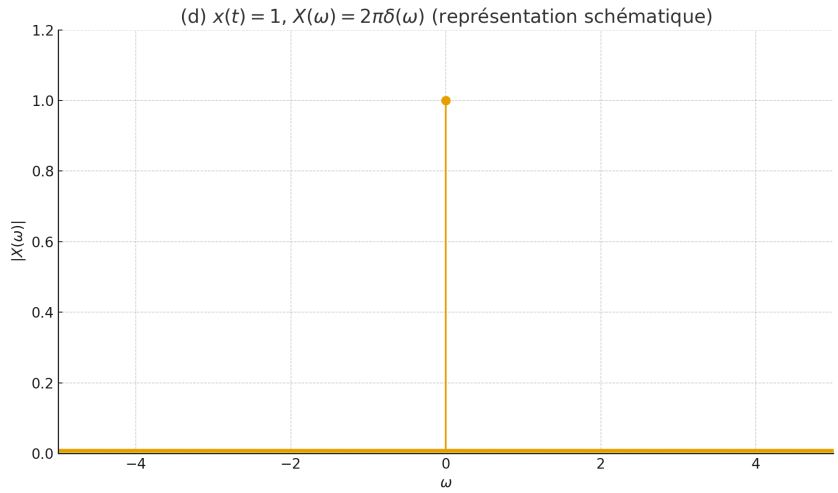
TF :

$$X(\omega) = 2\pi\delta(\omega).$$

Allure : pic de Dirac au niveau $\omega = 0$ (toute l’énergie concentrée en fréquence 0).

4. Exercices Supplémentaires

Solution 32.



(a) Chaque impulsion est un Dirac situé en (t = 2k), donc les impulsions sont espacées régulièrement de 2.
De plus, le facteur $(-1)^k$ fait simplement **alterner le signe** : une impulsion positive, puis une négative, etc.

$$x(t) = \delta(t) - \delta(t-2) + \delta(t-4) - \delta(t-6) + \dots$$

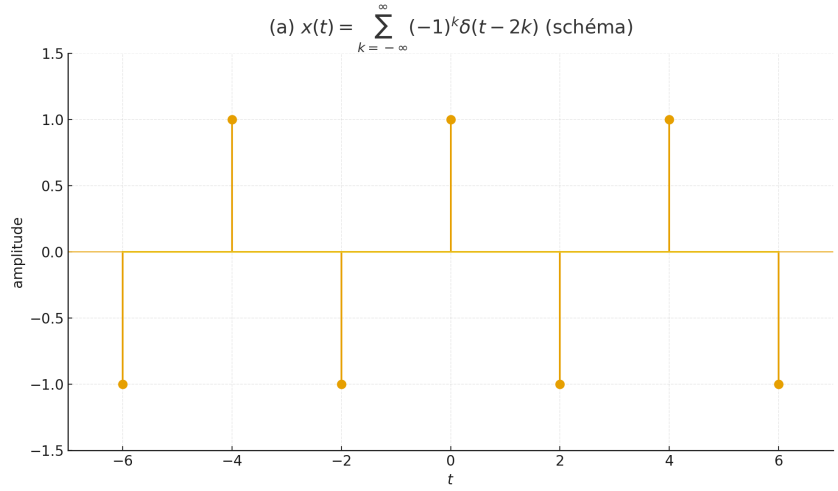
Le motif ((+,-)) se répète identiquement tous les (2) secondes :

$$x(t+2) = (-1)^{k+1}\delta(t+2-2(k+1)) = (-1)^k\delta(t-2k) = x(t)$$

Le signal est donc périodique avec :

$$T = 2, \quad f = \frac{1}{2}\ \text{Hz}, \quad \omega = 2\pi f = \pi\ \text{rad/s}.$$

Intuition : c'est un "peigne de Dirac" (train d'impulsions) alterné en signe, répété toutes les 2 unités de temps.



(b) On a :

$$v(t) = \cos(t)u(t) = \begin{cases} \cos(t) & t > 0, \\ 0 & t < 0. \end{cases}$$

Et :

$$v(-t) = \cos(-t)u(-t) = \cos(t)u(-t) = \begin{cases} 0 & t > 0, \\ \cos(t) & t < 0. \end{cases}$$

En les additionnant :

$$x(t) = v(t) + v(-t) = \begin{cases} \cos(t) & t > 0, \\ \cos(t) & t < 0, \\ \cos(0) = 1 & t = 0. \end{cases}$$

Autrement dit :

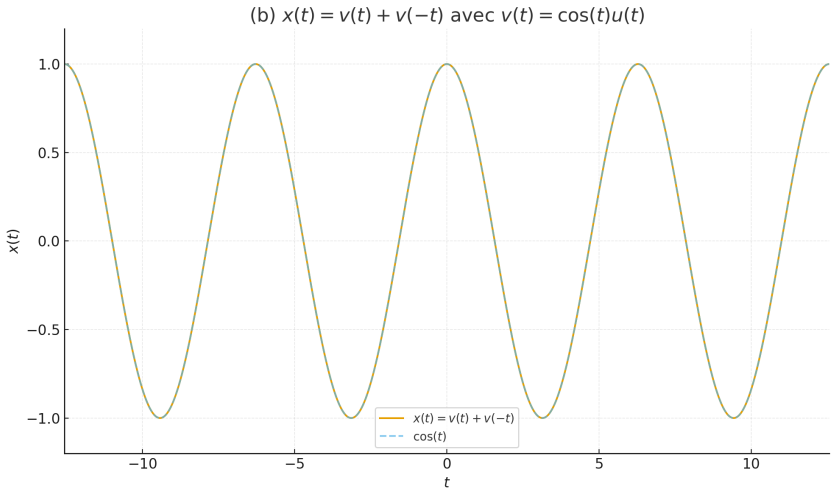
$$x(t) = \cos(t) \text{ pour tout } t.$$

La fonction obtenue est bien continue et périodique, de période $T = 2\pi$. Mais attention : cela n'est vrai que parce que les morceaux se complètent parfaitement.

Si on gardait les échelons déphasés (par ex. (u(t)) et (u(-t)) sans symétrie exacte), le signal aurait une discontinuité et ne serait pas périodique.

$$T = 2\pi, \quad f = \frac{1}{2\pi}, \quad \omega = 1.$$

Intuition : la combinaison $v(t) + v(-t)$ "recolle" les deux moitiés du cosinus, donc on récupère le signal cosinus complet et périodique.



Solution 33.

(a) $X(\omega) = \frac{1}{2i}\left(\frac{1}{2-i(3\pi-\omega)} - \frac{1}{2-i(3\pi+\omega)}\right)$

(b) $X(\omega) = \frac{1}{2}\left(\frac{1}{i\omega-(3+i5\pi)} + \frac{1}{i\omega-(3-i5\pi)}\right)$

(c) $X(\omega) = \sqrt{\pi}e^{-\omega^2/4}$

(d) $X(\omega) = \frac{1}{(1+i\omega)^2}$

(e) $X(\omega) = e^{-i3\omega}$

(f) $X(\omega) = \pi e^{-|\omega|}$

(g) $X(\omega) = \frac{1}{b+i\omega}$

(h) $X(\omega) = \frac{\pi}{\pi^2+\omega^2}$

(i) $X(\omega) = \frac{e^{-i\omega}}{2}\left(\frac{1}{1+i(\omega-2\pi)} + \frac{1}{1+i(\omega+2\pi)}\right)$

(j) $X(\omega) = i\omega$

Solution 34.

(a) $x(t) = \frac{1}{6}e^{-3|t|}$

(b) $x(t) = \sqrt{\pi}e^{-t^2/4}$

(c) $x(t) = e^{-t}u(t)$

(d) $x(t) = \frac{-1}{\pi(t^2-4)}$

(e) $x(t) = \frac{1}{2\pi}e^{i5t}$

(f) $x(t) = \frac{1}{2a}e^{-a|t|}$

(g) $x(t) = \frac{1}{2\pi} \cdot \frac{1+it}{1+t^2}$

(h) $x(t) = \frac{1}{2} \cdot e^{-|t|}$

LSINC1113 - Compléments de mathématiques

TP10 - La transformée de Fourier

Rappels théoriques

1. Ce que veut dire « échantillonner » un signal

On part d'un signal analogique $x(t)$ (continu dans le temps).
On le mesure toutes les T_e secondes : on obtient un signal discret, une suite de valeurs $x[k] = x(kT_e)$.

Mathématiquement, on peut écrire le signal échantillonné comme :

$$x_e(t) = \sum_{k=-\infty}^{\infty} x(kT_e)\,\delta(t - kT_e),$$

où

$$T_e = \text{période d'échantillonnage}, \qquad f_e = \frac{1}{T_e} = \text{fréquence d'échantillonnage (en Hz)}.$$

À retenir :
Échantillonner = ne garder que des valeurs ponctuelles du signal, à intervalles réguliers T_e.

2. Ce qui se passe dans le domaine fréquentiel

Soit $X(f)$ le spectre (Transformée de Fourier) du signal continu $x(t)$.

L'échantillonnage à la fréquence f_e a deux effets importants dans le domaine fréquentiel (de façon schématique) :

- le spectre $X(f)$ est **recopié périodiquement** autour de toutes les fréquences kf_e ($k \in \mathbb{Z}$) ;
- si ces copies se chevauchent, on ne peut plus les distinguer : c'est le **repliement spectral** (aliasing).

Cela signifie :
Tant que le spectre original est petit (par ex. limité à $[-1, 1]$ kHz) et que f_e est grand (8 kHz), ces copies :

- ne se chevauchent pas ;
- on voit clairement quelle partie est "l'original" (autour de 0) ;
- et on peut "filtrer" pour récupérer le bon spectre.

Mais si le spectre original est plus large, par exemple jusqu'à 5 kHz, et qu'on l'échantillonne à 8 kHz : la copie centrée en 0 va de -5 à +5 kHz, alors que la copie centrée en 8 kHz va de 3 à 13 kHz. Entre 3 et 5 kHz, les deux se chevauchent. On parle d'aliasing : des morceaux de la copie se replient dans la bande utile.

A retenir :
Quand on échantillonne, on ne garde pas qu'un seul spectre : on crée une infinité de copies

du spectre continu, décalées de $\pm f_e, \pm 2f_e$. Si la fréquence d'échantillonnage est assez grande (Shannon respecté), ces copies ne se touchent pas. Si elle est trop petite, elles se chevauchent. Les hautes fréquences se "replient" et on ne peut plus distinguer ce qui vient d'où : c'est le repliement spectral.

Théorème de Shannon–Nyquist (version complète)

Si un signal **ne contient pas de fréquences au-dessus de $f_{\max}$ en valeur absolue**, c'est-à-dire que son spectre est nul en dehors de :

$$X(f) = 0 \quad \text{pour } |f| > f_{\max} \quad (\text{autrement dit } f \in [-f_{\max}, f_{\max}]),$$

alors il suffit de l'échantillonner à une fréquence :

$$f_e \geq 2f_{\max}$$

pour pouvoir le reconstruire parfaitement à partir de ses échantillons. Et garantir donc qu'il n'y a **pas d'aliasing**.

Remarque importante :
Le facteur 2 vient du fait que le spectre d'un signal réel est **symétrique** : il contient à la fois $+f$ et $-f$. Par exemple, un simple cosinus

$$x(t) = \cos(2\pi f_0 t)$$

possède deux pics dans son spectre :

$$X(f) = \tfrac{1}{2}\delta(f - f_0) + \tfrac{1}{2}\delta(f + f_0),$$

c'est-à-dire un à $+f_0$ et un à $-f_0$. Ainsi, même si on parle souvent uniquement de la fréquence positive f_0, il faut se rappeler que le spectre réel occupe en réalité toute la bande $[-f_{\max}, +f_{\max}]$. C'est précisément pour cela que la condition de Shannon fait intervenir $2f_{\max}$: il faut éviter que les copies du spectre (autour de $+f_e$ et $-f_e$) se chevauchent.

3. Fréquence réduite et aliasing

Lorsqu'on échantillonne un signal à la fréquence f_e, on passe d'un signal continu $x(t)$ à une suite discrète $x[k] = x(kT_e)$, avec $T_e = 1/f_e$.

En temps discret, ce qui compte n'est plus la fréquence « réelle » f (en Hz), mais sa position par rapport à la fréquence d'échantillonnage.

Fréquence réduite (ou normalisée)
On définit la fréquence réduite :

$$\tilde{f} = \frac{f}{f_e}.$$

C'est une fréquence *sans unité*, qui indique « combien de fois par période d'échantillonnage » le signal oscille.

Dans beaucoup de formules en temps discret, la fréquence n'apparaît plus comme f, mais comme ce rapport f/f_e.

Périodicité en fréquence et aliasing

Après échantillonnage, le spectre discret est **périodique** en fréquence avec une période f_e :

$$\text{Les fréquences } f,\ f \pm f_e,\ f \pm 2f_e, \ldots \text{ deviennent indistinguables.}$$

Autrement dit, toutes les fréquences de la forme

$$f' = f + nf_e, \qquad n \in \mathbb{Z},$$

sont des **alias** les unes des autres : elles donnent le même comportement du signal après échantillonnage.

Fréquence alias observée
En pratique, on ne peut observer (sans ambiguïté) que les fréquences dans la **bande de Nyquist**

$$[0, f_e/2].$$

Toute fréquence réelle f (même très grande) se « replie » dans cette bande. On appelle *fréquence aliasée* (ou *fréquence observée*) une version repliée de f dans $[0, f_e/2]$.

On peut l'obtenir en retranchant ou ajoutant des multiples de f_e :

$$f_{\text{alias}} = |\, f - mf_e \,| \quad \text{pour un certain entier } m \text{ choisi de sorte que } f_{\text{alias}} \in [0, f_e/2].$$

À retenir :

- Après échantillonnage, les fréquences sont vues modulo f_e (d'où la fréquence réduite $\tilde{f} = f/f_e$).
- Les fréquences qui diffèrent d'un multiple de f_e sont des *alias* : le système discret ne peut pas les distinguer.
- On ne voit effectivement que les fréquences dans $[0, f_e/2]$; les autres se replient dans cette bande sous forme de f_{alias}.
- Échantillonner trop lentement (f_e trop petit) fait se replier les composantes de haute fréquence dans la bande $[0, f_e/2]$.
- Plusieurs fréquences analogiques différentes peuvent donner le *même signal échantillonné* : on ne sait plus retrouver l'original — c'est l'aliasing.

4. Harmoniques et fondamentale

En traitement du son, on parle souvent de **fréquence fondamentale** et d'**harmoniques**.

- La **fréquence fondamentale** f_0 est la fréquence de base d'un son périodique (sa hauteur principale, par ex. 440 Hz pour le “La”).
- Les **harmoniques** sont les fréquences multiples de la fondamentale :

  $$f_n = nf_0, \quad n = 2, 3, 4, \ldots$$

  Par exemple, si $f_0 = 440$ Hz :

  $$2f_0 = 880 \text{ Hz}, \quad 3f_0 = 1320 \text{ Hz}, \quad 4f_0 = 1760 \text{ Hz}, \ldots$$

Un son peut donc être vu comme la somme :

$$x(t) = A_1 \cos(2\pi f_0 t) + A_2 \cos(2\pi 2 f_0 t) + A_3 \cos(2\pi 3 f_0 t) + \ldots$$

- La fondamentale f_0 donne la **hauteur perçue** (la note).

- Les harmoniques ($2f_0$, $3f_0$, ...) donnent le **timbre** (son “rond”, “métallique”, etc.).

Dans le spectre $|X(f)|$:

- On voit un pic à f_0 (fondamentale),
- D'autres pics à $2f_0, 3f_0, \dots$ (harmoniques),
- Leurs amplitudes relatives $(A_1, A_2, A_3, \dots)$ déterminent la couleur du son.

Remarque : dans certains cas, la fondamentale peut être absente du spectre (par ex. seulement 880 Hz et 1320 Hz présents), mais le cerveau peut tout de même percevoir la hauteur $f_0 = 440$ Hz : on parle de *fondamentale manquante*.

1. Pourquoi échantillonner un signal ?

Un signal réel (son, tension, image, etc.) est **continu dans le temps** : il existe pour tout t, comme $x(t)$. Mais un ordinateur ou une carte son ne peuvent **pas traiter un signal continu** : ils ne peuvent manipuler que des **valeurs discrètes**, prises à intervalles réguliers.

Échantillonner un signal, c'est donc :

- mesurer $x(t)$ toutes les T_e secondes,
- et ne garder que la suite de valeurs $x[k] = x(kT_e)$.

C'est le **passage du monde analogique au monde numérique**.

2. Ce que fait l'échantillonnage au spectre

Quand on échantillonne dans le temps, on ne **regarde** plus tout le signal, on ne prend que des points réguliers. Mathématiquement, c'est comme si on multipliait $x(t)$ par un *peigne de Dirac* :

$$x_e(t) = x(t) \sum_{k=-\infty}^{+\infty} \delta(t - kT_e).$$

En fréquence, **une multiplication dans le temps devient une convolution en fréquence**. Résultat :

Le spectre $X(f)$ du signal est **recopié périodiquement** autour de tous les multiples de $f_e = 1/T_e$.

avec f_e la fréquence d'échantillonnage.

C'est l'**origine** du repliement spectral (aliasing).

3. Le repliement spectral (aliasing)

Imaginons que le signal de base a un spectre $X(f)$ contenu entre -5 kHz et $+5$ kHz. Si on l'échantillonne à $f_e = 8$ kHz, on crée des copies de $X(f)$ autour de 0, ± 8 kHz, ± 16 kHz, etc.

Ces copies peuvent **se chevaucher** :

- la bande $[-5, +5]$ kHz chevauche la copie centrée à $+8$ kHz,
- les hautes fréquences se "mélangent" avec les basses.

Quand on regarde le signal discret, **on ne peut plus savoir** quelle fréquence vient d'où : par exemple, une composante à 7 kHz sera vue comme une fréquence

$$f_{\text{alias}} = |7 - 8| = 1 \text{ kHz}.$$

Le repliement spectral (aliasing), c'est le fait que les hautes fréquences du signal original se replient dans la bande $[0, f_e/2]$ et imitent des fréquences plus basses.

4. À quoi ça sert de le comprendre ?

Le repliement spectral, c'est un phénomène qu'on cherche en général à **éviter** quand on veut :

- numériser un son (pour ne pas fausser les fréquences),
- traiter un signal en numérique (FFT, filtrage, etc.),

- ou reconstruire le signal analogique plus tard.

Mais c'est aussi une notion **utile** :

- pour comprendre le spectre discret : on sait qu'il est périodique, donc il suffit d'observer $[0, f_e/2]$;
- dans certaines techniques (modulation, sur-échantillonnage, synthèse numérique), on exploite ce repliement volontairement.

Exercice 1 – Repliement d'un cosinus simple

On considère le signal analogique

$$x(t) = \cos(2\pi f_0 t).$$

(a) On échantillonne $x(t)$ à $f_e = 8000$ Hz. Pour chacun des cas suivants, calculer la fréquence réduite $\tilde{f} = f_0/f_e$ (modulo 1) et donc la fréquence *perçue* après échantillonnage :

$$f_0 \in \{1000,\, 3000,\, 5000,\, 9000\}\,\text{Hz}.$$

Donner pour chaque cas la fréquence finale entre 0 et $f_e/2$.

(b) Représenter qualitativement (à la main) le spectre $|X(f)|$ avant échantillonnage (un pic à $\pm f_0$).

(c) Représenter qualitativement le spectre après échantillonnage (copies autour de kf_e) et montrer où se trouve le pic "replié" pour $f_0 = 9000$ Hz.

Exercice 2 – Quel son j'entends après sous-échantillonnage ?

On considère un signal audio constitué de deux sinusoïdes :

$$x(t) = \cos(2\pi \cdot 1000\, t) + \cos(2\pi \cdot 7000\, t).$$

(a) On échantillonne ce signal à $f_e = 16$ kHz. Y a-t-il du repliement spectral ? Justifier à partir de Shannon.

(b) Même question pour $f_e = 10$ kHz. Calculer la ou les fréquences **après repliement**, c'est-à-dire les fréquences qui apparaîtront effectivement dans le signal échantillonné (comprises entre 0 et $f_e/2$).

(c) Interpréter en termes de "ce que j'entends" : quelles hauteurs (grave/aigu) sont présentes dans chaque cas ?

Exercice 3 – Gamme musicale et aliasing

On prend une gamme "idéale" composée des notes suivantes (fréquences approximatives) :

$$\{220,\ 440,\ 880,\ 1760\}\ \text{Hz}.$$

(a) On échantillonne à $f_e = 8$ kHz. Vérifier que toutes ces fréquences respectent le critère de Shannon. Que se passe-t-il pour la reconstruction ?

(b) On échantillonne maintenant à $f_e = 3$ kHz (échantillonnage trop lent). Pour chaque note, calculer la(les) fréquence(s) repliée(s) observée(s) dans le signal échantillonné (dans l'intervalle $[0, f_e/2]$).

(c) Discuter : pourquoi une note "aiguë" peut-elle se transformer en une note "plus grave" après échantillonnage trop lent ? Relier cela au repliement spectral.

Exercice 4 – Application pratique : spectre et modulation

Un signal audio $x(t)$ est la somme de deux composantes :

$$x(t) = \cos(2\pi \cdot 440\, t) + 0.5\cos(2\pi \cdot 880\, t).$$

(a) Tracer qualitativement son spectre $|X(f)|$ (axe en Hz). Indiquer clairement les fréquences présentes et leurs amplitudes relatives.

(b) Que représente chaque pic fréquentiel physiquement ? (Amplitudes et hauteurs : fondamentale 440 Hz, harmonique 880 Hz, etc.)

(c) On multiplie $x(t)$ par une porteuse :

$$y(t) = x(t)\cos(2\pi \cdot 2000\, t).$$

En utilisant la relation

$$\cos(2\pi f_1 t)\cos(2\pi f_2 t) = \frac{1}{2}\cos(2\pi(f_1+f_2)t) + \frac{1}{2}\cos(2\pi(f_1-f_2)t),$$

déterminer les nouvelles fréquences présentes dans $Y(f)$ et tracer qualitativement $|Y(f)|$.

(d) Expliquer en une phrase le lien avec les transmissions radio : rôle de la porteuse, déplacement du spectre vers des fréquences plus élevées, etc.

LSINC1113 - Compléments de mathématiques

Correction TP10 - Transformée de Fourier

Exercice 1 – Repliement d'un cosinus simple

On considère le signal

$$x(t) = \cos(2\pi f_0 t), \qquad f_e = 8000 \text{ Hz}.$$

(a) **Fréquence réduite et fréquence perçue**
On définit la fréquence réduite :

$$\tilde{f} = \frac{f_0}{f_e},$$

et on cherche ensuite la fréquence *perçue* (aliasée) dans la bande de Nyquist $[0, f_e/2] = [0, 4000]$ Hz. On peut utiliser

$$f_{\text{alias}} = \left| f_0 - m f_e \right| \quad \text{avec } m \in \mathbb{Z} \text{ choisi pour que } f_{\text{alias}} \in [0, 4000].$$

$\underline{f_0 = 1000 \text{ Hz}}$

$$\tilde{f} = \frac{1000}{8000} = 0{,}125, \qquad 1000 < 4000 \Rightarrow f_{\text{alias}} = 1000 \text{ Hz}.$$

$\underline{f_0 = 3000 \text{ Hz}}$

$$\tilde{f} = \frac{3000}{8000} = 0{,}375, \qquad 3000 < 4000 \Rightarrow f_{\text{alias}} = 3000 \text{ Hz}.$$

$\underline{f_0 = 5000 \text{ Hz}}$

$$\tilde{f} = \frac{5000}{8000} = 0{,}625.$$

On replie autour de f_e :

$$f_{\text{alias}} = |5000 - 8000| = 3000 \text{ Hz}.$$

$\underline{f_0 = 9000 \text{ Hz}}$

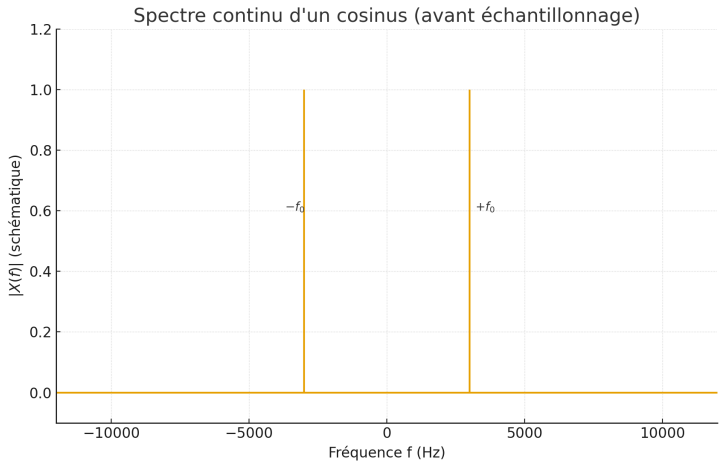
$$\tilde{f} = \frac{9000}{8000} = 1{,}125 \equiv 0{,}125 \pmod 1,$$

et

$$f_{\text{alias}} = |9000 - 8000| = 1000 \text{ Hz}.$$

Résumé :

f_0 (Hz)	$\tilde{f} = f_0/f_e$	f_{alias} (Hz)
1000	0,125	1000
3000	0,375	3000
5000	0,625	3000
9000	$1{,}125 \equiv 0{,}125$	1000



(b) **Spectre $|X(f)|$ avant échantillonnage**
Le spectre continu d'un cosinus est

$$X(f) = \frac{1}{2}[\delta(f - f_0) + \delta(f + f_0)].$$

Donc $|X(f)|$ se représente par deux pics en $f = +f_0$ et $f = -f_0$.

(c) **Spectre après échantillonnage et repliement pour $f_0 = 9000$ Hz**
Après échantillonnage à $f_e = 8000$ Hz, le spectre est périodisé :

$$X_e(f) = \frac{1}{T_e} \sum_{k=-\infty}^{\infty} X(f - kf_e),$$

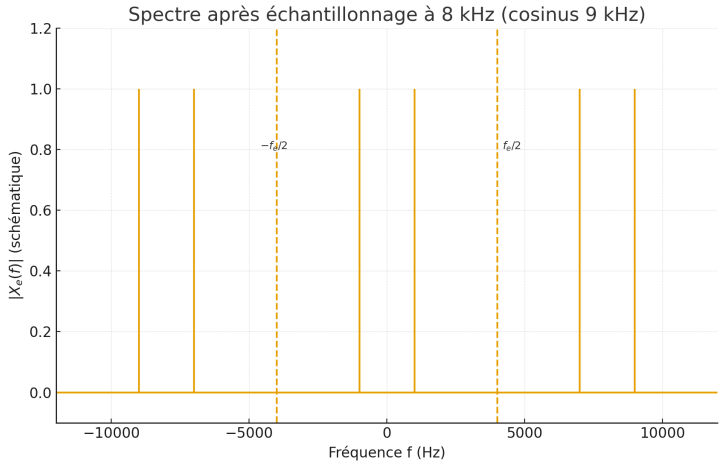
c'est-à-dire que les Dirac en $\pm f_0$ sont recopiés en

$$f = \pm f_0 + kf_e, \qquad k \in \mathbb{Z}.$$

Pour $f_0 = 9000$ Hz, on obtient notamment des pics en

$$\pm 9000 \text{ Hz}, \quad 9000 - 8000 = 1000 \text{ Hz}, \quad -9000 + 8000 = -1000 \text{ Hz}, \ldots$$

Dans la bande de Nyquist $[-4000, 4000]$ Hz, les pics visibles sont donc en ± 1000 Hz : c'est la fréquence alias $f_{\text{alias}} = 1000$ Hz.



Exercice 2 – Quel son j'entends après sous-échantillonnage ?

On considère

$$x(t) = \cos(2\pi \cdot 1000\, t) + \cos(2\pi \cdot 7000\, t),$$

donc le signal contient deux fréquences : $f_1 = 1000$ Hz et $f_2 = 7000$ Hz.

(a) **Cas $f_e = 16$ kHz**
La fréquence maximale du signal est

$$f_{\max} = 7000 \text{ Hz}.$$

Le critère de Shannon–Nyquist demande :

$$f_e \geq 2f_{\max} \quad \Rightarrow \quad f_e \geq 14000 \text{ Hz}.$$

Ici, $f_e = 16000$ Hz ≥ 14000 Hz, donc **le critère est respecté** : il n'y a **pas de repliement spectral**.
Les deux composantes sont donc correctement représentées :

fréquences observées : 1000 Hz et 7000 Hz.

(b) **Cas $f_e = 10$ kHz**
On a toujours $f_{\max} = 7000$ Hz, mais maintenant

$$2f_{\max} = 14000 \text{ Hz} > f_e = 10000 \text{ Hz}.$$

⇒ Le critère de Shannon–Nyquist **n'est pas respecté** : il y a **aliasing**.

La bande de Nyquist est :

$$[0, f_e/2] = [0, 5000] \text{ Hz}.$$

- La composante à 1000 Hz est dans la bande de Nyquist ⇒ **pas de repliement** :

$$f_{1,\text{alias}} = 1000 \text{ Hz}.$$

- La composante à 7000 Hz est au-dessus de 5000 Hz. On la replie en utilisant, par exemple :

$$f_{\text{alias}} = |f_0 - mf_e|.$$

Avec $f_0 = 7000$ Hz, $f_e = 10000$ Hz, et $m = 1$:

$$f_{2,\text{alias}} = |7000 - 10000| = 3000 \text{ Hz}.$$

3000 Hz est bien dans $[0, 5000]$.
Au final, après échantillonnage à 10 kHz, les fréquences effectivement présentes dans le signal échantillonné sont :

fréquences observées : 1000 Hz et 3000 Hz.

(c) **Interprétation « ce que j'entends »**

- **Avec $f_e = 16$ kHz :**
  — une composante à 1 kHz (plutôt « grave / médium »),
  — une composante à 7 kHz (son nettement plus aigu).
  On entend bien un son composé de deux hauteurs distinctes : un grave + un aigu.
- **Avec $f_e = 10$ kHz :**
  — la composante à 1 kHz reste à 1 kHz,
  — la composante à 7 kHz est mal échantillonnée et se replie en 3 kHz.
  Le signal échantillonné contient donc des fréquences à 1 kHz et 3 kHz : on entend un son à deux hauteurs plus proches (1 kHz et 3 kHz), et **plus du tout le vrai 7 kHz d'origine**. C'est une distorsion due à l'aliasing.

Exercice 3 – Gamme musicale et aliasing

On considère une gamme idéale

$$\{220,\ 440,\ 880,\ 1760\}\ \text{Hz}.$$

On note ces fréquences $f_1 = 220$, $f_2 = 440$, $f_3 = 880$, $f_4 = 1760$ Hz.

(a) **Cas $f_e = 8$ kHz**
La fréquence d'échantillonnage vaut

$$f_e = 8000\ \text{Hz}, \qquad \frac{f_e}{2} = 4000\ \text{Hz}.$$

La fréquence maximale dans la gamme est

$$f_{\max} = 1760\ \text{Hz}.$$

Le critère de Shannon–Nyquist demande

$$f_e \geq 2f_{\max} \quad \Longleftrightarrow \quad f_e \geq 3520\ \text{Hz}.$$

On a bien $8000 \geq 3520$, donc **le critère est respecté**.

Toutes les fréquences de la gamme sont dans la bande de Nyquist $[0, 4000]$ Hz et **aucune ne subit de repliement spectral**.

Conséquence : avec un filtre de reconstruction idéal passe-bas (coupure à 4 kHz), on peut théoriquement **reconstruire exactement** le signal analogique original : la gamme entendue après conversion analogique vers numérique puis inversément est la même.

(b) **Cas $f_e = 3$ kHz**
Ici :

$$f_e = 3000\ \text{Hz}, \qquad \frac{f_e}{2} = 1500\ \text{Hz}.$$

La bande de Nyquist est donc $[0, 1500]$ Hz.

On traite chaque note :
$\underline{f_1 = 220\ \text{Hz}}$
Cette fréquence est déjà dans $[0, 1500]$ Hz, donc pas de repliement :

$$f_{1,\text{alias}} = 220\ \text{Hz}.$$

$\underline{f_2 = 440\ \text{Hz}}$
Même chose : $440 \in [0, 1500]$:

$$f_{2,\text{alias}} = 440\ \text{Hz}.$$

$\underline{f_3 = 880\ \text{Hz}}$
Toujours dans la bande de Nyquist :

$$f_{3,\text{alias}} = 880\ \text{Hz}.$$

$\underline{f_4 = 1760\ \text{Hz}}$
Ici, $1760 > 1500$, donc il y aura repliement.

$$f_{\text{alias}} = |f_0 - m f_e| \quad \text{avec } m \in \mathbb{Z} \text{ choisi de sorte que } f_{\text{alias}} \in [0, 1500].$$

On prend $m = 1$:

$$f_{4,\text{alias}} = |1760 - 3000| = 1240 \text{ Hz},$$

qui appartient bien à $[0, 1500]$.

Résumé : après échantillonnage à 3 kHz, les fréquences effectivement observées dans le signal échantillonné sont :

$$220,\ 440,\ 880,\ \text{et } 1240 \text{ Hz}$$

au lieu de

$$220,\ 440,\ 880,\ 1760 \text{ Hz}.$$

(c) **Discussion : aigu qui devient plus grave**

L'idée clé est la suivante :

- En continu, 1760 Hz est la note la plus aiguë de la gamme.
- Mais avec un échantillonnage à $f_e = 3000$ Hz, on ne peut distinguer que les fréquences dans $[0, 1500]$ Hz.
- Toute fréquence au-dessus de 1500 Hz est repliée (aliasée) dans cette bande, en "rebondissant" autour des multiples de f_e.

Ici, la note à 1760 Hz se replie en

$$f_{\text{alias}} = 1240 \text{ Hz},$$

qui est plus *grave* que 1500 Hz, et même plus proche de 880 Hz que de 1760 Hz.

Donc : une note très aiguë (1760 Hz) peut être perçue, après échantillonnage trop lent, comme une note *plus grave* (1240 Hz).

C'est exactement le **repliement spectral** :

- le spectre continu est copié autour de kf_e,
- les copies se recouvrent,
- des composantes de haute fréquence se retrouvent déplacées dans la bande basse $[0, f_e/2]$,
- d'où une confusion entre hauteurs différentes.

En audio, cela se traduit par des sons déformés, des notes fausses ou carrément méconnaissables quand la fréquence d'échantillonnage est trop basse.

Exercice 4 - Application pratique : spectre et modulation

On considère

$$x(t) = \cos(2\pi \cdot 440\,t) + 0.5\cos(2\pi \cdot 880\,t).$$

(a) **Spectre** $|X(f)|$

Rappel : la transformée de Fourier de $\cos(2\pi f_0 t)$ est

$$\mathcal{F}\{\cos(2\pi f_0 t)\} = \frac{1}{2}[\delta(f - f_0) + \delta(f + f_0)].$$

Donc :

$$\mathcal{F}\{\cos(2\pi \cdot 440\,t)\} = \frac{1}{2}[\delta(f - 440) + \delta(f + 440)],$$

$$\mathcal{F}\{0.5\cos(2\pi \cdot 880\,t)\} = 0.5 \cdot \frac{1}{2}[\delta(f - 880) + \delta(f + 880)] = \frac{1}{4}[\delta(f - 880) + \delta(f + 880)].$$

Donc le spectre de $x(t)$ est

$$X(f) = \frac{1}{2}[\delta(f - 440) + \delta(f + 440)] + \frac{1}{4}[\delta(f - 880) + \delta(f + 880)].$$

À tracer qualitativement :

- Deux pics en $f = \pm 440$ Hz, d'amplitude $\frac{1}{2}$;
- Deux pics en $f = \pm 880$ Hz, d'amplitude $\frac{1}{4}$;
- Les pics à 880 Hz ont donc une amplitude **deux fois plus petite** que ceux à 440 Hz.

(b) **Interprétation des pics fréquentiels**

Physiquement :

- Le pic à 440 Hz correspond à une sinusoïde pure de fréquence 440 Hz,
- Le pic à 880 Hz est l'harmonique supérieure (2e harmonique), à une octave au-dessus.

L'amplitude temporelle vaut :

$$\text{pour 440 Hz : amplitude 1,} \qquad \text{pour 880 Hz : amplitude 0.5.}$$

Cela signifie que :

- La composante à 440 Hz domine (fondamentale),
- La composante à 880 Hz est présente mais deux fois moins forte (timbre plus riche, "son harmonique").

Chaque pic fréquentiel représente donc une sinusoïde présente dans le son, avec :

- Sa **fréquence** (hauteur),
- Son **amplitude** (intensité / poids dans le timbre).

(c) **Multiplication par une porteuse :**
$y(t) = x(t)\cos(2\pi \cdot 2000t)$

On applique la relation produit cos cos :

$$\cos(2\pi f_1 t)\cos(2\pi f_2 t) = \frac{1}{2}\cos(2\pi(f_1 + f_2)t) + \frac{1}{2}\cos(2\pi(f_1 - f_2)t).$$

On traite les deux termes de $x(t)$.

Premier terme : $\cos(2\pi \cdot 440t)\cos(2\pi \cdot 2000t)$

$$\cos(2\pi \cdot 440t)\cos(2\pi \cdot 2000t) = \frac{1}{2}\cos(2\pi(2000+440)t) + \frac{1}{2}\cos(2\pi(2000-440)t)$$
$$= \frac{1}{2}\cos(2\pi \cdot 2440t) + \frac{1}{2}\cos(2\pi \cdot 1560t).$$

Deuxième terme : $0.5\cos(2\pi \cdot 880t)\cos(2\pi \cdot 2000t)$

D'abord sans le 0.5 :

$$\cos(2\pi \cdot 880t)\cos(2\pi \cdot 2000t) = \frac{1}{2}\cos(2\pi(2000+880)t) + \frac{1}{2}\cos(2\pi(2000-880)t)$$
$$= \frac{1}{2}\cos(2\pi \cdot 2880t) + \frac{1}{2}\cos(2\pi \cdot 1120t).$$

En re-multipliant par 0.5 :

$$0.5\cos(2\pi \cdot 880t)\cos(2\pi \cdot 2000t) = 0.25\cos(2\pi \cdot 2880t) + 0.25\cos(2\pi \cdot 1120t).$$

Donc

$$y(t) = x(t)\cos(2\pi \cdot 2000t)$$

contient les fréquences :

$$1560,\ 2440,\ 1120,\ 2880 \text{ Hz}$$

avec amplitudes temporelles :

- 0.5 pour les composantes à 1560 Hz et 2440 Hz,
- 0.25 pour les composantes à 1120 Hz et 2880 Hz.

Spectre $|Y(f)|$ (qualitatif) :

- Des pics en ± 1560 Hz et ± 2440 Hz (plus grands),
- Des pics en ± 1120 Hz et ± 2880 Hz (plus petits).

On voit que le spectre initial (pics à 440 et 880 Hz) s'est transformé en **paires de bandes latérales** autour de 2000 Hz :

$$2000 \pm 440, \quad 2000 \pm 880.$$

(d) **Lien avec les transmissions radio**

La multiplication par $\cos(2\pi f_c t)$ (la porteuse) **déplace le spectre** de $x(t)$ autour de la fréquence f_c (ici 2000 Hz), en créant des bandes latérales $f_c \pm f$; c'est exactement le principe de la modulation en radio, où on "colle" un signal audio (basses fréquences) sur une porteuse haute fréquence pour pouvoir l'envoyer dans l'air puis le démoduler à la réception.

LSINC1113 - Compléments de mathématiques

TP4 - Fonctions à deux variables

Rappels théoriques

Limites de Fonctions de Deux Variables
Pour une fonction $f(x, y)$ de deux variables, la limite en un point (a, b) est définie comme suit :

$$\lim_{(x,y)\to(a,b)} f(x, y) = L$$

si, pour toute suite de points (x, y) s'approchant de (a, b), les valeurs de $f(x, y)$ tendent vers L. Il est important de vérifier la limite selon différents chemins pour prouver son existence.

Dérivées Partielles
La dérivée partielle d'une fonction $f(x, y)$ par rapport à x au point (a, b) mesure le taux de variation de f dans la direction x, en gardant y constant. Elle est définie par :

$$f_x(a, b) = \lim_{h\to 0} \frac{f(a + h, b) - f(a, b)}{h}.$$

De même, la dérivée partielle par rapport à y est donnée par :

$$f_y(a, b) = \lim_{k\to 0} \frac{f(a, b + k) - f(a, b)}{k}.$$

Les dérivées partielles permettent de déterminer les taux de changement de la fonction dans chaque direction indépendante.

Le Gradient d'une Fonction
Le gradient d'une fonction $f(x, y)$, noté ∇f, est un vecteur qui regroupe les dérivées partielles de f par rapport à x et y :

$$\nabla f(x, y) = \begin{pmatrix} f_x(x, y) \\ f_y(x, y) \end{pmatrix}.$$

Le gradient indique la direction de la variation maximale de f et sa norme donne l'intensité de cette variation.

La Matrice Hessienne
La matrice Hessienne d'une fonction $f(x, y)$, notée $H(f)$, est la matrice des dérivées secondes de f par rapport à x et y. Elle est définie comme suit :

$$H(f) = \begin{pmatrix} f_{xx} & f_{xy} \\ f_{yx} & f_{yy} \end{pmatrix},$$

où : $f_{xx} = \frac{\partial^2 f}{\partial x^2}$ est la dérivée seconde de f par rapport à x, $f_{yy} = \frac{\partial^2 f}{\partial y^2}$ est la dérivée seconde de f par rapport à y et $f_{xy} = f_{yx} = \frac{\partial^2 f}{\partial x \partial y}$ est la dérivée croisée.
La Hessienne est utile pour étudier la courbure de f et pour déterminer la nature des points critiques.

Points Critiques et Nature des Extrémums
Un point critique de $f(x,y)$ est un point où les dérivées partielles de f sont toutes nulles : $f_x = 0$ et $f_y = 0$. Pour déterminer la nature du point critique, on utilise la matrice Hessienne $H(f)$ au point critique (a,b) :

- Si $\det(H(f)(a,b)) > 0$ et $f_{xx}(a,b) > 0$, alors (a,b) est un minimum local.
- Si $\det(H(f)(a,b)) > 0$ et $f_{xx}(a,b) < 0$, alors (a,b) est un maximum local.
- Si $\det(H(f)(a,b)) < 0$, alors (a,b) est un point-selle.
- Si $\det(H(f)(a,b)) = 0$, le test est indéterminé.

1. Limites

Exercice 35. Calculer la limite suivante :

(a) $\lim\limits_{(x,y)\to(0,0)} \dfrac{x^3 y}{x^6 + y^2}$

Exercice 36. Soit $f(x,y) = \frac{x^2y^2}{x^2+y^2}$

(a) Montrer que $f(x,y) \leq x^2 + y^2$ autour de $(0,0)$

2. Dérivation de fonctions à deux variables

Exercice 37. Calculer les dérivées partielles de fonctions ci-dessous aux différents points de leur domaine naturel :

(a) $f(x,y) = \sin(3xy) + e^{-2x^2y} + 2x^3$

(b) $f(x,y) = (x+y)^{-\frac{1}{2}}$

(c) $f(x,y) = \ln(x^2 + y^2)$

(d) $g(x,y) = x\cos(y) + y$

(e) $g(x,y) = \cos^3(5x - y^3) + \ln(3\ln(xy))$

(f) $h(x,y) = \arctan(y\sqrt{x}) + sin^2(3x^2 + xy - 5y^3)$

Exercice 38. Soient $g(x,y) = f(x,y)$ et

$$f(x,y) = \begin{cases} \dfrac{xy^2}{x^4+y^2} & \text{si } (x,y) \neq (0,0) \\ 0 & \text{si } (x,y) = (0,0) \end{cases}$$

(a) Monter que f est continue à l'origine.

(b) Calculer les dérivées partielles de f à l'origine.

Exercice 39. Calculer la hessienne des fonctions suivantes :

(a) $f(x,y) = x^2 + 5y^2 + 4xy - 2y$

(b) $f(x,y) = 3x^2y + 4x^3y^4 - 7x^9y^4$

(c) $f(x,y) = e^x \sin(y)$

Exercice 40. Calculer le gradient des fonctions suivantes :

(a) $f(x,y) = x + 3y^2$

(b) $f(x,y) = \sqrt{x^2 + y^2}$

(c) $f(x,y) = \dfrac{4y}{(x^2+1)}$

(d) $f(x,y) = 3x^2\sqrt{y}$

Exercice 41. Etudier les points critiques :

(a) $f(x,y) = 4xy - 2x^2 - y^4$

(b) $f(x,y) = 3xy - x^2 - y^2$

(c) $f(x,y) = 2x^4 + y^4 - x^2 - 2y^2$

(d) $f(x,y) = 4x^2 - 12xy + 9y^2$