Université de Nantes

BUT1 Info

Outils mathématiques fondamentaux

2023-2024

1 Arithmétique

Cours Equations diophantiennes linéaires

Théorème 1.1 *Considérons l'équation $ax + by = c$ où $a,\ b,\ c\ \in \mathbb{Z}$*

- *L'équation possède des solutions $(x, y) \in \mathbb{Z}^2$ si et seulement si $pgcd(a, b)|c$*
- *Si $pgcd(a, b)|c$ alors il existe une infinité de solutions paramétrables avec un k parcourant $\mathbb{Z}$*

Le premier point est une conséquence de l'identité de Bézout. Nous allons voir le second point via un exemple en donnant explicitement les solutions :

$$(E) : 210x + 55y = 15$$

Première étape : Y a-t-il des solutions ? Via l'algorithme d'Euclide on obtient que le PGCD de $a = 210$ et $b = 55$ est 5 et celui-ci divise $c = 15$ car $15 = 5 \times 3$). Il existe donc bien des solutions entières.

Deuxième étape : Trouver une solution particulière L'algorithme d'Euclide étendu nous donne l'identité de Bézout suivante :

$$210 \times 5\ +\ 55 \times (-19) = 5$$

et on a $15 = 5 \times 3$, une solution évidente s'impose donc en multipliant l'identité de Bézout par 3 :

$$210 \times 15\ +\ 55 \times (-57) = 15$$

Ainsi $(x_0, y_0) = (15, -57)$ est une solution particulière de (E).

Troisième étape : Expression d'une solution quelconque Soit $(x, y) \in \mathbb{Z}^2$ une solution de (E). On a alors :

$$210 \times x\ +\ 55 \times y = 15 \text{ et } 210 \times x_0\ +\ 55 \times y_0 = 15$$

Soit par soustraction : $210 \times (x - x_0)\ +\ 55 \times (y - y_0) = 0$.

Ici on peut (on doit même) simplifier l'équation en divisant chaque membre par le pgcd obtenu soit par 5 ici pour obtenir :

$42 \times (x - x_0)\ +\ 11 \times (y - y_0) = 0$ soit $42 \times (x - x_0)\ =\ -11 \times (y - y_0)$ Ainsi $42| - 11 \times (y - y_0)$, or $pgcd(42, 11) = 1$ (du fait de la simplification par division par le pgcd dont on comprend la nécessité ici) donc d'après le théorème de Gauss $42|(y - y_0)$. Il existe donc $k \in \mathbb{Z}$ tel que $y - y_0 = 42k$ ce qui va nous permettre d'exprimer x en repartant de $42 \times (x - x_0)\ =\ -11 \times 42k$ soit $x = -11k + x_0$.

Nous avons donc le fait que tout couple solution est de la forme $(-11k+x_0, 42k+y_0)$ soit $(-11k + 15, 42k - 57)$ avec $k \in \mathbb{Z}$.

Dernière étape : Conclusion Il convient enfin de vérifier que réciproquement tout couple ainsi formé avec un k quelconque est bien solution :

$$\begin{aligned} 210 \times (-11k + 15)\ +\ 55 \times (42k - 57) &= 210 \times (-11k)\ +\ 55 \times (42k) + 15 \\ &= 2310k - 2310k + 15 \\ &= 15 \end{aligned}$$

On a alors $S = \{(x, y) \in \mathbb{Z}^2 | \exists k \in \mathbb{Z}, (x, y) = (-11k + 15, 42k - 57)$

Exercice 1

Résoudre dans $\mathbb{Z}^2$ les équations suivantes :

1.1. $37x + 23y = 1$

1.2. $161x + 368y = 115$

1.3. $1980x + 455y = 39$

Exercice 2

Déterminer les pgcd et ppcm des couples de nombres suivants en exprimant systématiquement leurs décompositions en nombres premiers :

2.1. $(a, b) = (4\ 235, 156)$

2.2. $(c, d) = (213\ 444, 84)$

2.3. $(e, f) = (15\ 730, 2\ 535)$

Exercice 3

3.1. Montrer que $\forall n \in \mathbb{N}$:

$$n(n+1)(n+2)(n+3)\text{est divisible par } 24$$

3.2. Dites en justifiant si les affirmations suivantes sont vraies pour tous $a, m, n \in \mathbb{Z}$:

a. $a|mn \Rightarrow (a|m \vee a|n)$

b. $(a|m \vee a|n) \Rightarrow a|mn$

c. $(a|m \wedge a|n) \Rightarrow a|(m+n)$

Exercice 4

Démontrer que le nombre $7^n + 1$ est divisible par 8 si n est impair et donner le reste de sa division par 8 dans le cas où n est pair.

Exercice 5

Trouver le reste de la division par 13 du nombre 100^{1000}.

Exercice 6

6.1. Les entiers 19 et 193 sont-ils inversibles dans $Z_{/2014\mathbb{Z}}$? Calculer le cas échéant leur(s) inverse(s).

6.2. Quel est le plus petit entier naturel n tel que $n \times 193 \equiv 1024[2014]$? Décrivez l'obtention de la solution en 2 opérations seulement. Même question en remplaçant 193 par 19.

Exercice 7

Combien 15! admet-il de diviseurs ?

Exercice 8

Démontrer que, si a et b sont premiers entre eux, il en est de même des entiers $a + b$ et ab.

Exercice 9 Les nombres de Mersenne

On utilisera, pour $n \geq 1$ ce qu'on considèrera dans la suite de l'exercice, la relation $x^n - 1 = (x-1) \times (x^{n-1} + x^{n-2} + ... + x + 1)$ et $1 + x + x^2 + ... + x^n = \frac{1-x^{n+1}}{1-x}$ (avec $x \neq 1$).

9.1. Montrer que si $a^n - 1$ est premier alors $a = 2$.

9.2. Montrer que $2^n - 1$ ne peut être premier que si n est premier

9.3. Vérifier que $2^{11} - 1$ n'est pas premier

2 Opérations sur les vecteurs et matrices

Exercice 10

10.1. Calculer :

a. $\begin{pmatrix}3\\-4\\5\end{pmatrix}+\begin{pmatrix}1\\1\\-2\end{pmatrix}$ **b.** $\begin{pmatrix}1\\2\\-3\end{pmatrix}+\begin{pmatrix}4\\-5\end{pmatrix}$ **c.** $-3\begin{pmatrix}4\\-5\\-6\end{pmatrix}$ **d.** $-\begin{pmatrix}-6\\7\\-8\end{pmatrix}$

10.2. Soient $u=\begin{pmatrix}2\\-7\\1\end{pmatrix}$, $v=\begin{pmatrix}-3\\0\\4\end{pmatrix}$ et $w=\begin{pmatrix}0\\5\\-8\end{pmatrix}$.

Calculer :

a. $3u-4v$

b. $2u+3v-5w$

10.3. Calculer x et y si $\begin{pmatrix}x\\3\end{pmatrix}=\begin{pmatrix}2\\x+y\end{pmatrix}$

10.4. Calculer x, y et z si $\begin{pmatrix}2\\3\\4\end{pmatrix}=x\begin{pmatrix}1\\1\\1\end{pmatrix}+y\begin{pmatrix}1\\1\\0\end{pmatrix}+z\begin{pmatrix}1\\0\\0\end{pmatrix}$

Exercice 11

11.1. Calculer lorsque c'est possible :

a. $\begin{pmatrix}1&2&-3&4\\0&-5&1&-1\end{pmatrix}+\begin{pmatrix}3&-5&6&-1\\2&0&-2&-3\end{pmatrix}$

b. $\begin{pmatrix}1&2&-3\\0&-4&1\end{pmatrix}+\begin{pmatrix}3&5\\1&-2\end{pmatrix}$

c. $-3\begin{pmatrix}1&2&-3\\4&-5&6\end{pmatrix}$

11.2. Soient $A=\begin{pmatrix}2&-5&1\\3&0&-4\end{pmatrix}$, $B=\begin{pmatrix}1&-2&-3\\0&-1&5\end{pmatrix}$ et $C=\begin{pmatrix}0&1&-2\\1&-1&-1\end{pmatrix}$.

Calculer $3A+4B-2C$.

11.3. Trouver x, y, z et w si $3\begin{pmatrix}x&y\\z&w\end{pmatrix}=\begin{pmatrix}x&6\\-1&2w\end{pmatrix}+\begin{pmatrix}4&x+y\\z+w&3\end{pmatrix}$

Exercice 12

12.1. Calculer la transposée de $A=\begin{pmatrix}1&0&1&0\\2&3&4&5\\4&4&4&4\end{pmatrix}$

Exercice 13

On note $(r\times s)$ une matrice de dimensions $r\times s$.

13.1. Trouver les dimensions des produits suivants (quand ils existent) :

a. $(2\times3)(3\times4)$ **c.** $(1\times2)(3\times1)$ **e.** $(3\times4)(3\times4)$

b. $(4\times1)(1\times2)$ **d.** $(5\times2)(2\times3)$ **f.** $(2\times2)(2\times4)$

13.2. Soit $A = \begin{pmatrix} 2 & -1 & 0 \\ 1 & 0 & -3 \end{pmatrix}$ et $B = \begin{pmatrix} 1 & -4 & 0 & 1 \\ 2 & -1 & 3 & -1 \\ 4 & 0 & -2 & 0 \end{pmatrix}$.

a. Déterminer les dimensions de AB.

b. On note $A \times B = c_{ij}$. Calculer c_{23}, c_{14} et c_{21}.

13.3. Soient $A = \begin{pmatrix} 1 & 3 \\ 2 & -1 \end{pmatrix}$ et $B = \begin{pmatrix} 2 & 0 & -4 \\ 3 & -2 & 6 \end{pmatrix}$. Calculer AB et BA.

13.4. Soient $A = \begin{pmatrix} 2 & 1 \end{pmatrix}$ et $B = \begin{pmatrix} 1 & -2 & 0 \\ 4 & 5 & -3 \end{pmatrix}$. Calculer AB et BA.

13.5. Soit $A = \begin{pmatrix} 1 & 2 & 0 \\ 3 & -1 & 4 \end{pmatrix}$. Calculer ${}^tAA$ et $A\,{}^tA$.

Exercice 14

14.1. Écrire les matrices suivantes définies par leur terme général
- $A = (a_{ij}) \in \mathscr{M}_{2,3}(\mathbb{R})$, $a_{ij} = (-1)^{i+j}$
- $B = (b_{ij}) \in \mathscr{M}_{3,4}(\mathbb{R})$, $b_{ij} = j^{i-1}$
- $C = (c_{ij}) \in \mathscr{M}_{4,2}(\mathbb{R})$, $c_{ij} = i - j$

14.2. Parmi les produits suivants dire lesquels ont un sens et donner la taille de la matrice correspondante

$$A \times A, A \times B, A \times C, B \times A, B \times C, C \times A, C \times B, \ {}^tC \times C.$$

14.3. Quelle est la taille de la matrice $D = A \times B \times C$? Donner deux manières différentes de calculer $A \times B \times C$. En effectuer une.

14.4. Quelle est la taille de la matrice $E = {}^t(C \times A)$? Donner deux manières différentes de calculer E. En effectuer une.

14.5. Quelle est la taille de la matrice $F = A \times (B + E)$? Calculer de deux manières différentes F.

Exercice 15

Un fabriquant de meubles propose 4 canapés à la vente à la vente. Leurs confections nécessitent le passage éventuel par différents ateliers (les autres opérations sont sous-traitées) :
- Atelier soudure : 10h sont nécessaires pour la confection de 2 canapés $C1$ ou de 2 canapés C_2 ou encore d'un canapé C_4
- Atelier menuiserie : 20h sont nécessaires pour la confection de deux canapés C_1 ou d'un canapé C_3
- Atelier peinture : 8h sont nécessaires pour la confection de 4 canapés C_1 ou de 2 canapés C_2 ou de 4 canapés C_3 ou encore d'un canapé C_4

Une enseigne commerciale a passé la commande suivante :
- 8 canapés C_1
- 13 canapés C_2
- 20 canapés C_3
- 3 canapés C_4

15.1. Déterminez quelles quantités de temps dans les différents ateliers vont devoir être utilisées pour la fabrication pour répondre à cette commande. *(Exprimez le résultat comme produit d'une matrice et d'un vecteur)*

3 Systèmes linéaires

Exercice 16

16.1. Résoudre :

a. $\begin{cases} 2x+3y = 1 \\ 5x+7y = 3 \end{cases}$ **b.** $\begin{cases} 2x+4y = 10 \\ 3x+6y = 15 \end{cases}$ **c.** $\begin{cases} 4x-2y = 5 \\ -6x+3y = 1 \end{cases}$

16.2.

a. Déterminer une solution de :

$$\mathscr{E} \Leftrightarrow \begin{cases} 2x - 3y + 6z + 2v - 5w = 3 \\ \quad y - 4z + v \quad = 1 \\ \quad\quad\quad v - 3w = 2 \end{cases}$$

b. N'y a-t-il qu'une solution ?

c. Déterminer l'ensemble des solutions de $\mathscr{E}$.

16.3. Résoudre :

$$\begin{cases} x + 2y - 3z = -1 \\ 3x - y + 2z = 7 \\ 5x + 3y - 4z = 2 \end{cases}$$

16.4. Résoudre :

$$\begin{cases} 2x + y - 2z = 10 \\ 3x + 2y + 2z = 1 \\ 5x + 4y + 3z = 4 \end{cases}$$

Exercice 17

17.1. Déterminer si chacun des systèmes suivants a une solution non nulle :

a. $\begin{cases} x - 2y + 3z - 2w = 0 \\ 3x - 7y - 2z + 4w = 0 \\ 4x + 3y + 5z + 2w = 0 \end{cases}$ **b.** $\begin{cases} x + 2y - 3z = 0 \\ 2x + 5y + 2z = 0 \\ 3x - y - 4z = 0 \end{cases}$

c. $\begin{cases} x + 2y - z = 0 \\ 2x + 5y + 2z = 0 \\ x + 4y + 7z = 0 \\ x + 3y + 3z = 0 \end{cases}$

Exercice 18

18.1. Résoudre :

a. $\begin{cases} 2x + y - 3z = 5 \\ 3x - 2y + 2z = 5 \\ 5x - 3y - z = 16 \end{cases}$ **b.** $\begin{cases} 2x + 3y - 2z = 5 \\ x - 2y + 3z = 2 \\ 4x - y + 4z = 1 \end{cases}$

c. $\begin{cases} x + 2y + 3z = 3 \\ 2x + 3y + 8z = 4 \\ 3x + 2y + 17z = 1 \end{cases}$

Exercice 19

Joshua se déplace sur son réseau de transport uniquement selon 4 trajets :

- Le premier trajet lui fait emprunter 5 tronçons de la ligne 1, 10 tronçons de la ligne 2 et 2 tronçons de la ligne 3
- Le deuxième trajet lui fait emprunter 5 tronçons de la ligne 1 et 4 tronçons de la ligne 3
- Le troisième trajet lui fait emprunter 20 tronçons de la ligne 2 (-ah ouais comême chavais pas qu'elle était aussi longue la ligne 2 ! -Bah si tu vois) et 2 tronçons de la ligne 3
- Le quatrième trajet lui fait emprunter 10 tronçons de la ligne 1 et 8 tronçons de la ligne 3.

Lors d'un mois, il a emprunté :

- 45 fois un tronçon de la ligne 1
- 240 fois un tronçon de la ligne 2 (mais faut dire qu'elle est longue comême)
- 48 fois un tronçon de la ligne 3

19.1. Déterminez combien de trajets de chaque type il a effectué.

Exercice 20

Une entreprise assure la production de trois types de machines M_1, M_2 et M_3 en quantités hebdomadaires respectives x, y et z.

Le coût des éléments installés est de 6 € pour une machine M_1, 8 € pour une machine M_2 et 9 € pour une machine M_3.

Une machine M_1 nécessite 1 heure de travail, une machine M_2 nécessite 1.5 heures et une machine M_3, 2 heures.

Un programme de production hebdomadaire occasionne un coût c et un nombre t d'heures de travail.

20.1. Déterminez une matrice A telle que :

$$\begin{pmatrix} c \\ t \end{pmatrix} = A \times \begin{pmatrix} x \\ y \\ z \end{pmatrix}$$

20.2. Durant une semaine, l'entreprise constate un coût total de 3050 € et 550 heures de travail.

a. Déterminez le nombre de machines de chaque type fabriquées cette semaine. Y a-t-il une seule solution possible ?

b. Sachant que durant cette semaine, 50 machines M_3 ont été produites, déterminez combien de machines M_1 et M_2 ont été produites. Y a-t-il une seule solution possible ?

Exercice 21

Le bronze est un alliage de cuivre et d'étain et le laiton, un alliage de cuivre et de zinc, avec des proportions variables données dans le tableau ci-dessous :

	Bronze	Laiton
Cuivre	85%	90%
Étain	15%	0%
Zinc	0%	10%

21.1. Un objet, fabriqué à partir de bronze et de laiton, contient :
- 890g de cuivre
- 30g d'étain
- 80g de zinc

De combien de bronze et de laiton est-il constitué ?

21.2. Un second objet contient :
- 800g de cuivre
- 60g d'étain
- 40g de zinc

Est-il possible qu'il ne soit constitué que de bronze et de laiton ?

4 Inversion de matrices

Exercice 22

Si $B = \begin{pmatrix} 5 & -9 & 6 \\ 0 & 2 & 3 \\ 0 & 0 & 7 \end{pmatrix}$ alors avec des opérations élémentaires on peut avoir :

$$B \sim \begin{pmatrix} 5 & -9 & 6 \\ 0 & 2 & 3 \\ 0 & 0 & 1 \end{pmatrix} \sim \begin{pmatrix} 5 & -9 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 1 \end{pmatrix} \sim \begin{pmatrix} 5 & -9 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \sim \begin{pmatrix} 5 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \sim \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

Le signe $\sim$ étant « est équivalent à ».

22.1. Quelles sont les opérations élémentaires de lignes pour aller de la gauche vers la droite ? Donner systématiquement la matrice élémentaire qui, par multiplication à gauche, effectue cette opération.

Exercice 23

Soient les 3 matrices suivantes :

$$E_1 = \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad E_2 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 7 \end{pmatrix} \quad E_3 = \begin{pmatrix} 1 & 0 & 0 \\ -3 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

23.1. Quelles sont les opérations élémentaires de lignes pour arriver à trois matrices identité ?

Exercice 24

24.1. Calculer l'inverse des matrices suivantes :

$$M_1 = \begin{pmatrix} 4 & -3 & -2 \\ 2 & 2 & 0 \\ 2 & 1 & 0 \end{pmatrix}; \quad M_2 = \begin{pmatrix} 0 & -2 & 0 \\ -1 & 1 & 2 \\ -2 & 3 & 2 \end{pmatrix}; \quad M_3 = \begin{pmatrix} 2 & -1 & 3 & 3 \\ 2 & -2 & -3 & 0 \\ 2 & -2 & 3 & 3 \\ -3 & 0 & -1 & -3 \end{pmatrix}$$

Exercice 25

Soit $A = \begin{pmatrix} -1 & 1 & 1 \\ 1 & -1 & 1 \\ 1 & 1 & -1 \end{pmatrix}$

25.1.

a. Montrer que $A^2 = 2I_3 - A$.

b. En déduire que A est inversible et calculer A^{-1}.

Soit $B = \begin{pmatrix} 1 & 0 & 2 \\ 0 & -1 & 1 \\ 1 & -2 & 0 \end{pmatrix}$

25.2.

a. Calculer $B^3 - B$.

b. En déduire que B est inversible et calculer B^{-1}.

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

Exercice 26

Une société produit et vend 3 types de vêtements (V_1, V_2 et V_3) en 3 taille (T_1, T_2 et T_3) :

	T_1	T_2	T_3
V_1	1 €	3 €	5 €
V_2	2 €	4 €	7 €
V_3	5 €	5 €	5 €

Des équipes sportives souhaitent acheter le même type de vêtement pour tous leurs membres. Ils reçoivent les devis suivants :

V_1	290 €
V_2	410 €
V_3	450 €

équipe 1

V_1	205 €
V_2	290 €
V_3	325 €

équipe 2

V_1	210 €
V_2	305 €
V_3	350 €

équipe 3

V_1	290 €
V_2	410 €
V_3	350 €

équipe 4

26.1. Déterminer le nombre de membres de chaque équipe pour chaque taille de vêtement.

Travaux Pratiques

Exercice 1 Algorithmes abordés dans le cours

1.1. Réaliser les programmes suivants sans faire appel aux opérateurs ou fonctions déjà intégrés dans Python permettant le renvoi de ces résultats.

a. Programmer l'algorithme prenant en arguments deux entiers naturels a et b et renvoyant le couple formé du quotient et du reste de la division euclidienne de a par b (sans faire appel aux opérateurs déjà intégrés dans Python pour le renvoi de ces résultats)

b. Programmer l'algorithme d'Euclide étendu prenant en arguments deux entiers naturels a et b et renvoyant le triplet formé du pgcd d de ces deux entiers et de deux entiers u et v vérifiant $au + bv = d$.

Exercice 2 Test de primalité

2.1. Justifier qu'un nombre p est premier si et seulement si il n'est divisible par aucun nombre premier inférieur ou égal à $\sqrt{p}$

2.2. Ecrire une fonction renvoyant le $n^{\text{ème}}$ nombre premier en exploitant au mieux l'idée précédente.

Exercice 3 Conversion bases système numération

3.1. Ecrire une fonction permettant la conversion d'un nombre n en écriture décimale en nombre en base $b < 10$ soit un n-uplet de valeurs dans $\{0, \ldots, b-1\}$.

3.2. Ecrire une fonction inverse de la première

Exercice 4 Calcul de $a^k(\bmod n)$

Dans l'exemple ci-dessous on exploite le développement de k en base 2 pour le calcul de $5^{11}(\bmod 14)$. Ecrire un programme vous inspirant de ceci. Pouvez-vous remarquer un gain en temps d'exécution par rapport à un calcul direct ?

- Conversion de l'exposant en base 2 :
  $11 = 8 + 2 + 1 = 1 \times 2^3 + 1 \times 2^1 + 1 \times 2^0$
- Calcul des $5^{2^i}(\bmod 14)$:
  $5^{2^0} \equiv 5(\bmod 14)$
  $5^{2^1} \equiv 5 \times 5 \equiv 25 \equiv 11(mod 14)$
  $5^{2^2} \equiv 5^{2^1} \times 5^{2^1} \equiv 11 \times 11 \equiv 121 \equiv 9(\bmod 14)$
  $5^{2^3} \equiv 5^{2^2} \times 5^{2^2} \equiv 9 \times 9 \equiv 81 \equiv 11(\bmod 14)$
- Conclusion :

$$\begin{aligned} 5^{11}(\bmod 14) &\equiv 5^{2^3} \times 5^{2^1} \times 5^{2^0} \\ &\equiv 11 \times 11 \times 5(\bmod 14) \\ &\equiv 605(\bmod 14) \\ &\equiv 3(\bmod 14) \end{aligned}$$

5 Découverte de numpy

Exercice 5 `numpy`

`numpy` est une bibliothèque python utilisée notamment pour le calcul matriciel.
Elle s'utilise en l'important au début de votre fichier par la commande :

```
import numpy
```

ou plus généralement :

```
import numpy as np
```

5.1. Déterminer ce qu'effectuent les commandes suivantes :

- `np.zeros(7)`
- `np.ones(6)`
- `np.identity(3)`
- `np.array([3,7,-1,2])`
- `np.array([[3,7],[-1,2]])`
- `np.arange(10,30,5)`
- `np.linspace(0,2,9)`
- `np.sin(np.linspace(0,2*np.pi,20))`

On définit deux matrices :

- `a = np.array([[1,3],[0,4]])`
- `b = np.array([[4,0],[-1,1]])`

5.2. Déterminer ce qu'effectuent les commandes suivantes :

- `a+b`
- `a+4`
- `a*b`
- `3*a`
- `a*3`
- `np.add(a,b)`
- `a.dot(b)`
- `a @ b`
- `a.transpose()`
- `np.linalg.matrix_power(a,2)`
- `np.linalg.det(a)`
- `np.linalg.inv(a)`
- `a.shape`

5.3. Parmi les commandes précédentes, lesquelles ont un sens en algèbre linéaire ?

5.4. Toujours avec les matrices `a` et `b` :

- `a.sum()`
- `a.sum(axis=0)`
- `a.sum(axis=1)`
- `a.min()`
- `a.max()`
- `a[1]`
- `a[0,1]`
- `a[0][1]`

Notez que `numpy` est une librairie très utilisée et très efficace en calcul numérique. Elle permet de faire beaucoup d'autres choses, mais dont nous ne parlerons pas ici.

6 Opérations élémentaires sur les matrices

Exercice 6

Soit M la matrice carrée d'ordre 3 ci-dessous :

$$M = \begin{pmatrix} -5 & -5 & 2 \\ 6 & 8 & -4 \\ 6 & 5 & -1 \end{pmatrix}$$

6.1. Calculer $M^3 - 2M^2 - 5M$ et exprimer le résultat en fonction de Id_3.

6.2. Mettre en facteur la matrice M dans l'expression $M^3 - 2M^2 - 5M$ (*i.e.* écrire $M^3 - 2M^2 - 5M = M \times$?).

6.3. L'inverse de la matrice M est une matrice N telle que $M \times N = N \times M = Id_3$. Déduire de la question précédente l'inverse de M en fonction de Id_3, M et M^2.

6.4. Vérifier avec `python` que la matrice obtenue est bien l'inverse de M.

Exercice 7

Soient $A, B, C, D \in \mathscr{M}_{4,4}(\mathbb{R})$

$$A = \begin{pmatrix} 2 & -1 & -1 & -2 \\ 4 & 4 & 2 & 0 \\ 0 & 4 & -2 & 0 \\ -1 & -1 & 0 & -3 \end{pmatrix}, B = \begin{pmatrix} -2 & 2 & 1 & -3 \\ 1 & -3 & 0 & -2 \\ 2 & -1 & -2 & 3 \\ -4 & 3 & 3 & 3 \end{pmatrix},$$

$$C = \begin{pmatrix} 0 & -4 & 3 & 1 \\ 4 & 2 & -4 & 1 \\ 1 & -1 & 3 & 3 \\ 4 & 1 & 4 & -4 \end{pmatrix}, D = \begin{pmatrix} -8 & -6 & 26 & 0 \\ 4 & -8 & 0 & -11 \\ 10 & -5 & 26 & -6 \\ -12 & -23 & 0 & 38 \end{pmatrix}$$

7.1. Calculer $(A + B)^2$

7.2. Calculer $A^2 + 2AB + B^2$

7.3. Qu'en déduisez vous ?

7.4. De même calculer $(C + D)^2, C^2 + 2CD + D^2$, quel est votre conclusion cette fois ?

Exercice 8

On considère le système d'équations $(\mathscr{E}) \Leftrightarrow \begin{cases} 5x & +7y & = & 3 \\ 2x & +3y & = & 1 \end{cases}$

8.1. Montrer que $(\mathscr{E}) \Longleftrightarrow \begin{pmatrix} 5 & 7 \\ 2 & 3 \end{pmatrix} \times \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 3 \\ 1 \end{pmatrix}$

8.2. En déduire que $\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 5 & 7 \\ 2 & 3 \end{pmatrix}^{-1} \times \begin{pmatrix} 3 \\ 1 \end{pmatrix}$ si la matrice $\begin{pmatrix} 5 & 7 \\ 2 & 3 \end{pmatrix}$ est inversible.

8.3. En déduire la solution (x, y) du système d'équation à l'aide de `python`.

8.4. De même résoudre le système d'équations :

$$\begin{cases} x_1 & + & 2x_2 & & & = & 5 \\ x_1 & + & x_2 & - & x_3 & = & 1 \\ 2x_1 & + & x_2 & - & 2x_3 & = & 2 \end{cases}$$

Exercice 9

Dans chacun cas suivant, écrire une fonction `python` qui résout le problème posé.

9.1. Écrire une fonction `diagonale` qui à partir de A une matrice de taille $p \times n$, quelconque, calcule une matrice B de même taille :

$$\begin{array}{rccl} \texttt{diagonale}: & \mathscr{M}_{p,n}(\mathbb{R}) & \longrightarrow & \mathscr{M}_{p,n}(\mathbb{R}) \\ & A & \longmapsto & B \ \ avec \ \ B_{ij} = \left\{ \begin{array}{cc} A_{ij} & si \quad i=j \\ 0 & sinon \end{array} \right. \end{array}$$

9.2. Écrire une fonction `transpose` qui à partir de A une matrice de taille $p \times n$, quelconque, calcule sa transposée ${}^tA$, **sans utiliser la fonction existante** :

$$\begin{array}{rccl} \texttt{transpose}: & \mathscr{M}_{p,n}(\mathbb{R}) & \longrightarrow & \mathscr{M}_{n,p}(\mathbb{R}) \\ & A & \longmapsto & B = {}^tA \end{array}$$

9.3. Écrire une fonction `produit` qui à partir de deux matrices A et B, de tailles $p \times l$ et $l \times n$, calcule leur produit matriciel (ou renvoie [ ] si $A \times B$ n'existe pas) **sans utiliser `@` comme produit matriciel** :

$$\begin{array}{rccl} \texttt{produit}: & \mathscr{M}_{p,l}(\mathbb{R}) \times \mathscr{M}_{l,n}(\mathbb{R}) & \longrightarrow & \mathscr{M}_{p,n}(\mathbb{R}) \\ & A, B & \longmapsto & C = A \times B \end{array}$$

9.4. Écrire une fonction `Vandermonde` qui à partir d'un vecteur v quelconque, de longueur n, calcule la matrice A, de taille $n \times n$, d'élément courant $v[i]^j$ ($v[i]$ puissance j) :

$$\begin{array}{rccl} \texttt{Vandermonde}: & \mathscr{M}_{1,n}(\mathbb{R}) & \longrightarrow & \mathscr{M}_n(\mathbb{R}) \\ & v & \longmapsto & A = (v[i]^j) \quad pour \quad i,j = 0, \ldots, n-1 \end{array}$$

9.5. Écrire une fonction `permute_colonnes` qui à partir d'une matrice A, de taille $p \times n$, calcule une matrice B de même taille telle que :

- la première colonne de B soit la $n^{\text{ième}}$ colonne de A
- la deuxième colonne de B soit l'avant dernière colonne de A (la « $n-1$ »)
- ...
- la dernière colonne de B soit la $1^{\text{ière}}$ colonne de A

Indication : *si la colonne j de A devient la colonne j' de B calculer $j + j'$ en fonction de n, en déduire j' en fonction de j et n.*

9.6. Écrire une fonction `permute_lignes` identique à la fonction précédente mais qui permute les lignes d'une matrice.

9.7. Écrire une fonction `minimum` qui trouve la valeur minimale des coefficients d'une matrice A et sa position `(i_min,j_min)` dans la matrice A.

7 Méthode de Gauss

Un système de p équations linéaires à n inconnues se présente sous la forme suivante :

$$(\mathscr{E}) \Leftrightarrow \begin{cases} a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n &= y_1 \\ a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n &= y_2 \\ \vdots \qquad\qquad\qquad\qquad\qquad\quad \vdots & \vdots \quad \vdots \\ a_{p1}x_1 + a_{p2}x_2 + \cdots + a_{pn}x_n &= y_p \end{cases}$$

les différentes variables qui apparaissent ci-dessus ont des rôles bien différents :

- x_j sont les inconnues du système,
- a_{ij} sont les coefficients du système,
- y_i sont les seconds membres du système,

quand on veut résoudre un tel système seuls les coefficients et les seconds membres sont connus. En utilisant le produit matriciel le système d'équations linéaires $(\mathscr{E})$ peut être récrit sous forme matricielle :

$$\begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \vdots & \vdots & & \vdots \\ a_{p1} & a_{p2} & \dots & a_{pn} \end{pmatrix} \times \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} = \begin{pmatrix} a_{11}x_1 & +a_{12}x_2 & +\dots & +a_{1n}x_n \\ a_{21}x_1 & +a_{22}x_2 & +\dots & +a_{2n}x_n \\ \vdots & & & \vdots \\ a_{p1}x_1 & +a_{p2}x_2 & +\dots & +a_{pn}x_n \end{pmatrix} = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_p \end{pmatrix} \Leftrightarrow A\mathbf{x} = \mathbf{y}$$

avec des matrices $A \in \mathscr{M}_{p,n}(\mathbb{R})$, $\mathbf{x} \in \mathscr{M}_{n,1}(\mathbb{R})$ et $\mathbf{y} \in \mathscr{M}_{p,1}(\mathbb{R})$. Dans `python` un système d'équations linéaires sera donc représenté par deux matrices A et $\mathbf{y}$, par exemple :

système d'équations	version matricielle
$\begin{cases} x_1 + x_2 + x_3 = 2 & \mathscr{L}_1 \\ x_1 - x_2 + 2x_3 = 9 & \mathscr{L}_2 \\ -x_1 + 2x_2 + x_3 = -2 & \mathscr{L}_3 \end{cases}$	$A = \begin{pmatrix} 1 & 1 & 1 \\ 1 & -1 & 2 \\ -1 & 2 & 1 \end{pmatrix}$ $\mathbf{y} = \begin{pmatrix} 2 \\ 9 \\ -2 \end{pmatrix}$

La fonction `affichage(A,y)`, disponible sur Madoc dans le fichier *TP02.py*, permet de passer de la représentation matricielle à un système d'équation (en mode texte) :

```
A=np.array([[1,1,1],[1,-1,2],[-1,2,1]])
y=np.array([2,9,-2])

>>> affichage(A,y)
|   x_1 +   x_2 +   x_3 = 2
|   x_1 -   x_2 +  2x_3 = 9
|  -x_1 +  2x_2 +   x_3 = -2
```

La méthode la plus employée pour résoudre des systèmes d'équations est la méthode de Gauss. Elle consiste à transformer le système $(\mathscr{E})$, par une série de manipulations élémentaires sur les équations du système (permutation, addition d'équations, multiplication par un réel non-nul), en un système équivalent mais « triangulaire

supérieur », c'est à dire qu'on a un triangle de coefficients nuls en bas à gauche :

$$(\mathscr{E})\begin{cases} a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n &= y_1 \\ a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n &= y_2 \\ \vdots \qquad\qquad\qquad\qquad\quad \vdots\ \vdots\ \vdots & \\ a_{p1}x_1 + a_{p2}x_2 + \cdots + a_{pn}x_n &= y_p \end{cases} \xrightarrow{Gauss} \begin{cases} \tilde{a}_{11}x_1 + \tilde{a}_{12}x_2 + \ldots + \tilde{a}_{1n}x_n &= \tilde{y}_1 \\ \tilde{a}_{22}x_2 + \ldots + \tilde{a}_{2n}x_n &= \tilde{y}_2 \\ \ddots \qquad\quad \vdots & \\ \tilde{a}_{pn}x_n &= \tilde{y}_p \end{cases}$$

système qui est alors facile à résoudre en partant de la dernière équation.
Voici un exemple très simple de mise en œuvre de la méthode de Gauss :

système d'équations	version matricielle
$\begin{cases} x_1 + x_2 + x_3 = 2 & \mathscr{L}_1 \\ x_1 - x_2 + 2x_3 = 9 & \mathscr{L}_2 \\ -x_1 + 2x_2 + x_3 = -2 & \mathscr{L}_3 \end{cases}$	$A = \begin{pmatrix} 1 & 1 & 1 \\ 1 & -1 & 2 \\ -1 & 2 & 1 \end{pmatrix} \quad \mathbf{y} = \begin{pmatrix} 2 \\ 9 \\ -2 \end{pmatrix}$
$\begin{cases} x_1 + x_2 + x_3 = 2 & \mathscr{L}_1 \\ -2x_2 + x_3 = 7 & \tilde{\mathscr{L}}_2 = \mathscr{L}_2 - \mathscr{L}_1 \\ 3x_2 + 2x_3 = 0 & \tilde{\mathscr{L}}_3 = \mathscr{L}_3 + \mathscr{L}_1 \end{cases}$	$A = \begin{pmatrix} 1 & 1 & 1 \\ 0 & -2 & 1 \\ 0 & 3 & 2 \end{pmatrix} \quad \mathbf{y} = \begin{pmatrix} 2 \\ 7 \\ 0 \end{pmatrix}$
$\begin{cases} x_1 + x_2 + x_3 = 2 & \mathscr{L}_1 \\ -2x_2 + x_3 = 7 & \mathscr{L}_2 - \mathscr{L}_1 \\ 7x_3 = 21 & 2\tilde{\mathscr{L}}_3 + 3\tilde{\mathscr{L}}_2 \end{cases}$	$A = \begin{pmatrix} 1 & 1 & 1 \\ 0 & -2 & 1 \\ 0 & 0 & 7 \end{pmatrix} \quad \mathbf{y} = \begin{pmatrix} 2 \\ 7 \\ 21 \end{pmatrix}$

D'un point de vue matriciel, à une étape donnée de la méthode de Gauss, une partie de la matrice a déjà été traitée (le haut et la gauche de la matrice) et l'on se trouve en position (i, j) dans la matrice A (et en position i dans le vecteur $\mathbf{y}$) :

$$A = \left(\begin{array}{c|c} \ddots & \text{partie déjà traitée} \\ \hline 0 & \begin{array}{cc} a_{ij} & \cdots \\ \vdots & \text{partie restante} \end{array} \end{array}\right) \qquad \mathbf{y} = \begin{pmatrix} y_1 \\ \vdots \\ y_i \\ \vdots \\ y_p \end{pmatrix}$$

Cette méthode se décompose en 3 parties principales :
- rechercher un « pivot » (un $a_{ij} \neq 0$) dans la colonne j ($1^{\text{ière}}$ variable non éliminée)
- permuter deux lignes de A pour faire remonter le pivot sur la ligne i,
- éliminer les a_{lj} pour $l = i + 1, \ldots, p$ par combinaison des lignes i et l de A

Une fois le système mis sous forme triangulaire on retrouve facilement les valeurs des variables en partant de la dernière équation :

$$\begin{cases} x_1 + x_2 + x_3 = 2 \\ -2x_2 + x_3 = 7 \\ x_3 = 3 \end{cases} \Rightarrow \begin{cases} x_1 + x_2 = -1 \\ -2x_2 = 4 \\ x_3 = 3 \end{cases} \Rightarrow \begin{cases} x_1 + x_2 = -1 \\ x_2 = -2 \\ x_3 = 3 \end{cases} \Rightarrow \begin{cases} x_1 = 1 \\ x_2 = -2 \\ x_3 = 3 \end{cases}$$

et l'on peut vérifier que les valeurs trouvées sont bien solutions du système de départ :

$$A \times \mathbf{x} = \begin{pmatrix} 1 & 1 & 1 \\ 1 & -1 & 2 \\ -1 & 2 & 1 \end{pmatrix} \times \begin{pmatrix} 1 \\ -2 \\ 3 \end{pmatrix} = \begin{pmatrix} 2 \\ 9 \\ -2 \end{pmatrix} = \mathbf{y}$$

Remarque importante par défaut, les tableaux `numpy` conservent leur type. Aussi, si un tableau n'est initialisé qu'avec des entiers, une modification du tableau par une opération sur des entiers retournera par défaut un entier. Exemple :

```
A=np.array([[2,3,4],[5,6,7]])
print(A[1]/2)

>>> array([2.5,  3.  ,  3.5])

A[1] = A[1]/2
print(A[1])

>>> array([ 2,  3,  3])
```

Pour éviter ces problèmes, on peut :

- ne créer que des tableaux de réels :

  ```
  A = np.array([[1.,2.],[3.,4.]])
  ```

- ajouter l'option `dtype='f'` à la création de tableau

  ```
  A = np.array([...],dtype='f')
  ```

Exercice 10 Méthode de Gauss

10.1. Écrire la fonction `python permutation(A,y,i,k)` qui retourne une matrice A' et un vecteur y' dans lesquels les équations $n°i$ et k du système ont été permutées. Par exemple permuter les lignes $i = 0$ et $k = 2$ de $(\mathscr{E})$ renvoie :

$$A = \begin{pmatrix} 1 & 1 & 1 \\ 1 & -1 & 2 \\ -1 & 2 & 1 \end{pmatrix} \quad \mathbf{y} = \begin{pmatrix} 2 \\ 9 \\ -2 \end{pmatrix} \longrightarrow A = \begin{pmatrix} -1 & 2 & 1 \\ 1 & -1 & 2 \\ 1 & 1 & 1 \end{pmatrix} \quad \mathbf{y} = \begin{pmatrix} -2 \\ 9 \\ 2 \end{pmatrix}$$

Indication : *on pourra utiliser la commande* `.copy()`.

10.2. Écrire la fonction `python elimination(A,y,i,j)` qui élimine la variable x_j des équations $n°i + 1$ à p, par exemple `elimination(A,y,0,0)` élimine x_1 à partir de l'équation 1 dans $(\mathscr{E})$:

$$A = \begin{pmatrix} 1 & 1 & 1 \\ 1 & -1 & 2 \\ -1 & 2 & 1 \end{pmatrix} \quad \mathbf{y} = \begin{pmatrix} 2 \\ 9 \\ -2 \end{pmatrix} \longrightarrow A = \begin{pmatrix} 1 & 1 & 1 \\ 0 & -2 & 1 \\ 0 & 3 & 2 \end{pmatrix} \quad \mathbf{y} = \begin{pmatrix} 2 \\ 7 \\ 0 \end{pmatrix}$$

10.3. Premier algorithme

a. Déterminez une suite de commande (utilisant `elimination`) permettant de rendre le système précédent triangulaire.

b. Programmez une fonction `Gauss(A,y)` retournant un couple `(A',y')` représentant un système triangulaire équivalent à `(A,y)`, en effectuant la suite de commandes précédentes.

c. Que se passe-t-il lorsque vous appliquez votre algorithme au système suivant :

$$B = \begin{pmatrix} 2 & 2 & 1 \\ 1 & 1 & 0 \\ 2 & 3 & 4 \end{pmatrix} \quad \mathbf{y}_1 = \begin{pmatrix} 4 \\ 0 \\ 3 \end{pmatrix}$$

d. Pourquoi ?

10.4. Recherche du pivot On considère la situation suivante :

$$C = \begin{pmatrix} 1 & 2 & -2 & 3 & 1 \\ 0 & 0 & 0 & 5 & -1 \\ 0 & 1 & -2 & 1 & 1 \\ 0 & 0 & 0 & 1 & -1 \end{pmatrix} \quad \mathbf{y} = \begin{pmatrix} 1 \\ 3 \\ -1 \\ 2 \end{pmatrix}$$

- Le premier pivot se trouve dans la case $(0, 0)$. Il faudra donc effectuer la commande :
  — `C,y = elimination(C,y,0,0)` (qui ne change rien ici)
- Dans le cas précédent, on s'attend à ce que le pivot soit dans la case $(1, 1)$, or il se trouve dans la case $(2, 1)$. Il faudra donc permuter la seconde et la troisième ligne avant d'appliquer l'élimination :
  — `C,y = permutation(C,y,1,2)`
  — `C,y = elimination(C,y,1,1)` (qui ne change rien ici)
  on obtient alors :

$$C = \begin{pmatrix} 1 & 2 & -2 & 3 & 1 \\ 0 & 1 & -2 & 1 & 1 \\ 0 & 0 & 0 & 5 & -1 \\ 0 & 0 & 0 & 1 & -1 \end{pmatrix} \quad \mathbf{y} = \begin{pmatrix} 1 \\ -1 \\ 3 \\ 2 \end{pmatrix}$$

- le troisième pivot se trouve dans la case $(3, 4)$. Il n'y a pas besoin d'effectuer de permutation ici, il faudra seulement effectuer la commande :
  — `C,y = elimination(C,y,3,4)`

Remarque importante Étant donné que nous travaillons avec des nombres réels, il y a des approximations dans leurs représentations. Aussi il n'est pas prudent d'utiliser l'opérateur `x==y` pour tester l'égalité de deux variables, il est préférable d'utiliser la commande `np.isclose(x,y)` pour déterminer s'ils sont suffisamment proches pour être considérés comme égaux.

a. Élaborer un algorithme prenant en paramètre une case (i, j) et permettant de déterminer dans quelle case sera le pivot (il est possible qu'il se trouve dans la case (i, j)).

b. Programmez une fonction `get_next_pivot(A,i,j)` prenant en paramètre une matrice `A` et les coordonnées `(i,j)` du dernier pivot et retournant la case du prochain pivot.

c. Modifiez votre fonction `Gauss` en y intégrant `get_next_pivot` de manière à gérer les cas de recherche du prochain pivot.

10.5. Ajouter la commande `affichage(A,y)` à l'intérieur de votre programme pour afficher les différentes étapes de la méthode de Gauss. Tester avec le système de 3 équations à 3 inconnues donné en exemple.

Exercice 11 Résolution de systèmes d'équations

11.1. Pour chacun des systèmes d'équations suivants :
- Appliquer la méthode de Gauss pour rendre le système d'équations triangulaire,
- En déduire si le système a des solutions, une seule solution ou aucune solution,
- si le système a au moins une solution, en calculer une à la main.

1. $\begin{cases} 0 = 0 \\ 0 = 0 \end{cases}$

2. $\begin{cases} -x_1 - 3x_2 = 3 \\ x_1 + 2x_2 + 3x_3 = 4 \\ 2x_1 - 2x_2 + x_3 = 4 \end{cases}$

3. $\begin{cases} -2x_1 + 10x_2 + 3x_3 = -18 \\ -5x_1 + 34x_2 + 12x_3 = -54 \\ -x_1 + 8x_2 + 3x_3 = -12 \end{cases}$

4. $\begin{cases} -3x_1 - x_2 + 4x_3 + 3x_4 = 8 \\ + 4x_3 + 2x_4 = 10 \\ -2x_1 - 2x_2 - 2x_3 + x_4 = -1 \end{cases}$

5. $\begin{cases} 2x_1 + 15x_2 + 3x_3 = 25 \\ + 21x_2 + 7x_3 = 35 \\ 4x_1 + 9x_2 - x_3 = 10 \end{cases}$

6. $\begin{cases} 4x_1 + 3x_2 + 4x_3 = 1 \\ - x_3 = 0 \\ 3x_1 - 3x_2 + x_3 = 6 \\ - x_2 + 3x_3 = 1 \end{cases}$

7. $\begin{cases} - 3x_2 - 2x_3 = 3 \\ -2x_1 + 2x_2 - 2x_3 = -4 \\ -4x_1 + 4x_2 - 4x_3 = -8 \\ -2x_1 + -1x_2 + -4x_3 = -1 \end{cases}$

Exercice 12 Résolution d'un système triangulaire

Un système d'équations triangulé par la méthode de Gauss se présente sous la forme :

$$(\mathscr{E}) \Leftrightarrow \begin{cases} a_{11}x_1 + a_{12}x_2 + \dots \quad \dots + a_{1n}x_n = y_1 \\ a_{22}x_2 + \dots \quad \dots + a_{2n}x_n = y_2 \\ \ddots \quad \vdots \\ a_{pp}x_p + \dots \quad a_{pn}x_n = y_p \end{cases} \Leftrightarrow \begin{array}{c}\text{sous forme matricielle} \\ A\mathbf{x} = \mathbf{y} \\ \text{avec}\end{array}$$

$$A = \begin{pmatrix} a_{11} & a_{12} & \dots & \dots & \dots & a_{1n} \\ 0 & a_{22} & \dots & \dots & \dots & a_{2n} \\ \vdots & \ddots & \ddots & & \vdots & \vdots \\ 0 & \dots & 0 & a_{pp} & \dots & a_{pn} \end{pmatrix}, \quad \mathbf{y} = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_p \end{pmatrix} \quad et \quad \mathbf{x} = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix}.$$

Quand le système a des solutions, on les trouve facilement en calculant une des variables x_i pour chaque équation **en partant de la dernière équation** et en laissant les variables indéterminées à une valeur par défaut (0 ou autre). Exemple :

$$\begin{cases} -2x_1 + 2x_2 - 3x_3 = -9 \\ -8x_2 + 4x_3 = 12 \\ 0x_3 = 0 \\ \Rightarrow \text{on fixe } x_3 = 0 \end{cases} \Rightarrow \begin{cases} -2x_1 + 2x_2 - 3x_3 = -9 \\ -8x_2 = 12 - 4x_3 = 12 \\ \Rightarrow \text{on obtient } x_2 = -\frac{3}{2} \\ x_3 = 0 \end{cases} \Rightarrow \begin{cases} -2x_1 = -9 - (2x_2 - 3x_3) \\ \Rightarrow -2x_1 = -6 \\ \Rightarrow \text{on obtient } x_1 = 3 \\ x_2 = -\frac{3}{2};\ x_3 = 0 \end{cases}$$

12.1. Résoudre l'équation $a_1x_1 + a_2x_2 + \dots + a_px_p = y$ dans laquelle les variables x_2, …, x_p sont instanciée et où la seule inconnue est x_1.

Pour résoudre un système triangulaire, on peut utiliser l'algorithme suivant :

Algorithme 1 : solveTriSup(A, y)

p =nombre d'inconnues, n =nombre d'équations;
x = vecteur à p lignes, initialisé à 0 ; `// valeur par défaut des` x_i
pour chaque *ligne i de A (de $n-1$ jusqu'à 0)* **faire**
  k = indice du premier coeff. non nul de la ligne i de A;
  `// l'équation courante est alors`
    $a_{i,k}x_k + a_{i,k+1}x_{k+1} + \cdots + a_{i,p}x_p = y_i$`, les` $x_{k+1}, \ldots, x_p$ `sont instanciés, on cherche la valeur de` x_k`.`
  si *k existe* **alors**
    Instancier la valeur de x_k à partir de l'équation
    $a_{i,k}x_k + a_{i,k+1}x_{k+1} + \cdots + a_{i,p}x_p = y_i$
  sinon
    `// l'équation est du type` $0 = y_i$
    Déterminer dans quel cas il n'y a pas de solution et dans quel cas l'algorithme continue
  fin
fin
retourner x

12.2. Écrire une fonction `first_non_zero(L)` prenant en paramètre une ligne de matrice et retournant l'indice de la première coordonnée non nulle ou -1 si toutes les coordonnées de la ligne sont nulles.

12.3. Compléter la fonction `solveTriSup(A,y)` qui prend en entrée deux matrices A et $\mathbf{y}$, représentant un système d'équations **triangulaire supérieur**, et qui calcule **UNE** solution $\mathbf{x}$ du système $A\mathbf{x} = \mathbf{y}$, s'il en existe, ou renvoie la matrice vide sinon en affichant « pas de solution ».

Dans la résolution de système triangulaire, il y a deux cas possibles :

- le système contient une ligne $0 = y$ avec $y \neq 0$: le système n'a alors pas de solution
- le système n'en contient pas, il a alors au moins une solution. En supposant qu'on supprime du système les lignes $0 = 0$, il y a deux autres possibilités :
  - le système comporte autant d'équations que d'inconnues, le système a alors une unique solution :
$$\begin{cases} x_1 + 3x_2 = -5 \\ \quad - 4x_2 = 8 \end{cases}$$
  - le système comporte strictement plus d'inconnues que d'équations, le système a une infinité de solutions :
$$\begin{cases} -2x_1 + 2x_2 - 2x_3 = -4 \\ \quad\quad\quad - 2x_3 = 3 \end{cases}$$
  - *remarque : un système triangulaire ne peut pas comporter strictement plus d'équations que d'inconnues.*

12.4.

a. Dans votre fonction `solveTriSup`, définissez deux compteurs `nb_eq` et `nb_inc` représentant le nombre d'équations et le nombre d'inconnues.

b. Modifiez votre programme pour actualiser la valeur de ces compteurs lorsque vous rencontrer une ligne $0 = 0$.

c. Avant le `return` final, effectuer un test afin d'afficher s'il y a une unique ou une infinité de solutions.

Exercice 13

13.1. Pour chacun des systèmes d'équations de l'exercice 11 :

- saisir le système sous forme matricielle dans `python` (matrices `A` et `y`),
- vérifier le résultat avec la fonction `affichage`,
- appliquer la méthode de Gauss au système, récrire le nouveau système obtenu,
- dire si le système possède une, aucune ou une infinité de solutions,
- si le système possède au moins une solution `x` en trouver une (« à la main »), et vérifier le résultat obtenu en calculant `A * x`
- retrouve-ton le résultat de $A\mathbf{x} = \mathbf{y}$ en calculant `x=np.linalg.inv(A)@y` avec `python` ?