SÉANCE 5
MATHÉMATIQUES 1

Mattéo Delabre, Guillaume Pérution-Kihli & Julien Rodriguez

Université de Montpellier
26 février 2022

MOTIVATIONS

- Dans certains problèmes, il suffit de trouver la bonne formule.
- Souvent autour des entiers, et souvent des formules ad-hoc.
- Il existe des recettes générales qui pourront vous être utiles.

MANIPULATION DES NOMBRES EN PYTHON

- Pas d'entier maximal en Python (contrairement à C ou C++)
  - Opérations sur les entiers en $O(1)$ seulement en dessous de `sys.maxsize` (environ 10^{18})

- **Éviter les nombres flottants à tout prix !**
  - `/` : division flottante
  - `//` : division entière

- Modules standards utiles
  - `math` : Opérations classiques (puissances, racines, arrondi, ...)
  - `fractions` : Représentation exacte des nombres rationnels (à privilégier par rapport aux flottants car pas d'arrondi)
  - `cmath` : Équivalent de `math` pour les nombres complexes
  - `decimal` : Flottants décimaux à précision ajustable

PLAN

ALGORITHME D'EUCLIDE

- $\mathrm{pgcd}(a, b)$: Plus grand diviseur commun à deux entiers a et b
- **Algorithme d'Euclide :**
  1. $\mathrm{pgcd}(a, b) = \mathrm{pgcd}(b, a \bmod b)$
  2. $\mathrm{pgcd}(a, 0) = a$ *[si $a \neq 0$]*

$$\begin{aligned} & \mathrm{pgcd}(728, 388) = \mathrm{pgcd}(388, 340) \\ = {} & \mathrm{pgcd}(340, 48) = \mathrm{pgcd}(48, 4) \\ = {} & \mathrm{pgcd}(4, 0) = 4 \end{aligned}$$

- Complexité : $O(\log(a + b))$

Statue d'Euclide au musée de l'Université d'Oxford — Photographié par Mark A. Wilson (CC-BY-SA 4.0)

PGCD EN PYTHON

- Programmation de l'algorithme *à la main*

```python
def gcd(a, b):
    while b:
        a, b = b, a % b
    return a
```

- En utilisant la bibliothèque standard

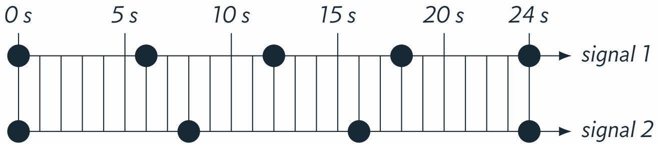
```python
>>> import math
>>> math.gcd(728, 388)
4
```

SIGNAUX CLIGNOTANTS

- *Vous apercevez deux signaux clignotants au loin. L'un s'allume toutes les 6 s et l'autre toutes les 8 s. Quel intervalle de temps faut-il pour qu'ils s'éclairent simultanément ?*

SIGNAUX CLIGNOTANTS

- *Vous apercevez deux signaux clignotants au loin. L'un s'allume toutes les 6 s et l'autre toutes les 8 s. Quel intervalle de temps faut-il pour qu'ils s'éclairent simultanément ?*



- $\text{ppcm}(6, 8) = 24$
- $\text{ppcm}(a, b)$: Plus petit multiple commun à deux entiers a et b
  - $\text{ppcm}(a, b) = ab / \text{pgcd}(a, b)$

EXERCICES ET RÉFÉRENCES

- Exercices :
  - UVa 11388, “GCD LCM”
  - UVa 10814, “Simplifying Fractions”
- Dans les livres de référence :
  - Dürr et Vie, §14.1
  - Laaksonen, §11.1.3
  - Halim, §5.5.2

PLAN

PROBLÈMES D'ARGENT

- *Vous vous trouvez dans un pays où les seules devises disponibles sont 5 ¤ et 7 ¤. Vous souhaitez acheter un objet qui coûte 1 ¤. Comment faire ?* *Indice : On peut vous rendre de la monnaie*

PROBLÈMES D'ARGENT

- *Vous vous trouvez dans un pays où les seules devises disponibles sont 5 ¤ et 7 ¤. Vous souhaitez acheter un objet qui coûte 1 ¤. Comment faire ?* *Indice : On peut vous rendre de la monnaie*

Vous — TOTAL : 0 — Vendeur

PROBLÈMES D’ARGENT

- *Vous vous trouvez dans un pays où les seules devises disponibles sont 5 ¤ et 7 ¤. Vous souhaitez acheter un objet qui coûte 1 ¤. Comment faire ?* *Indice : On peut vous rendre de la monnaie*

PROBLÈMES D'ARGENT

- *Vous vous trouvez dans un pays où les seules devises disponibles sont 5 ¤ et 7 ¤. Vous souhaitez acheter un objet qui coûte 1 ¤. Comment faire ?* *Indice : On peut vous rendre de la monnaie*

PROBLÈMES D'ARGENT

- *Vous vous trouvez dans un pays où les seules devises disponibles sont 5 ¤ et 7 ¤. Vous souhaitez acheter un objet qui coûte 1 ¤. Comment faire ?* *Indice : On peut vous rendre de la monnaie*

$$
\begin{tikzcd}[column sep=huge]
\text{Vous} \arrow[r, bend left, "7 + 7"] \arrow[r, phantom, "\text{TOTAL : } 9"] & \text{Vendeur} \arrow[l, bend left, "5"]
\end{tikzcd}
$$

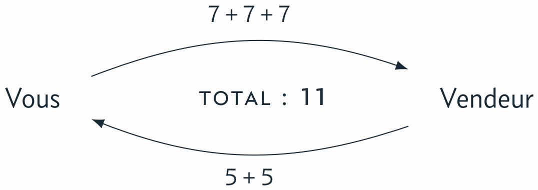
PROBLÈMES D'ARGENT

- *Vous vous trouvez dans un pays où les seules devises disponibles sont 5 ¤ et 7 ¤. Vous souhaitez acheter un objet qui coûte 1 ¤. Comment faire ?* *Indice : On peut vous rendre de la monnaie*

$$
\begin{tikzcd}
\text{Vous} \arrow[rr, bend left, "7+7"] & \text{TOTAL : } 4 & \text{Vendeur} \arrow[ll, bend left, "5+5"]
\end{tikzcd}
$$

PROBLÈMES D'ARGENT

- *Vous vous trouvez dans un pays où les seules devises disponibles sont 5 ¤ et 7 ¤. Vous souhaitez acheter un objet qui coûte 1 ¤. Comment faire ?* *Indice : On peut vous rendre de la monnaie*



- *Vous vous trouvez dans un pays où les seules devises disponibles sont 5 ¤ et 7 ¤. Vous souhaitez acheter un objet qui coûte 1 ¤. Comment faire ?* *Indice : On peut vous rendre de la monnaie*

$$\begin{tikzcd} \text{Vous} \arrow[r, bend left, "7+7+7"] \arrow[r, phantom, "\textbf{TOTAL : 1}"] & \text{Vendeur} \arrow[l, bend left, "5+5+5+5"] \end{tikzcd}$$

- Puisqu'on peut payer 1 ¤, on peut payer n'importe quel montant !
- Est-ce vrai pour des devises de valeur 4 ¤ et 6 ¤ ?

IDENTITÉ DE BÉZOUT

- Nous cherchions en fait des solutions entières à l'équation suivante, où x et y sont des entiers relatifs (le nombre de devises échangées)

$$5x + 7y = 1$$

IDENTITÉ DE BÉZOUT

- Nous cherchions en fait des solutions entières à l'équation suivante, où x et y sont des entiers relatifs (le nombre de devises échangées)

$$ax + by = c$$

- Cette forme générale est appelée *identité de Bézout*
  - x et y sont des *coefficients de Bézout*

- **Théorème :** Il existe des solutions *si et seulement si* c est un multiple de $\text{pgcd}(a, b)$
  - 5x + 7y = 1 $\rightsquigarrow \exists$ solutions car $\text{pgcd}(5, 7) = 1$
  - 4x + 6y = 1 $\rightsquigarrow$ pas de sol. car $\text{pgcd}(4, 6) = 2$

ALGORITHME D'EUCLIDE ÉTENDU

- Résoudre $899x + 493y = 29$
- Notons $a = 899, b = 493$ et $c = 29$

$$899 = 493 \times 1 + 406$$

Statue d'Euclide au musée de l'Université d'Oxford — Photographié par Mark A. Wilson (CC-BY-SA 4.0)

ALGORITHME D'EUCLIDE ÉTENDU

- Résoudre $899x + 493y = 29$
- Notons $a = 899$, $b = 493$ et $c = 29$

$$899 = 493 \times 1 + 406$$
$$493 = 406 \times 1 + 87$$

Statue d'Euclide au musée de l'Université d'Oxford — Photographié par Mark A. Wilson (CC-BY-SA 4.0)

ALGORITHME D'EUCLIDE ÉTENDU

- Résoudre $899x + 493y = 29$
- Notons $a = 899, b = 493$ et $c = 29$

$$899 = 493 \times 1 + 406$$
$$493 = 406 \times 1 + 87$$
$$406 = 87 \times 4 + 58$$

Statue d'Euclide au musée de l'Université d'Oxford — Photographié par Mark A. Wilson (CC-BY-SA 4.0)

ALGORITHME D'EUCLIDE ÉTENDU

- Résoudre $899x + 493y = 29$
- Notons $a = 899, b = 493$ et $c = 29$

$$
\begin{aligned}
899 &= 493 \times 1 + 406 \\
493 &= 406 \times 1 + 87 \\
406 &= 87 \times 4 + 58 \\
87 &= 58 \times 1 + 29
\end{aligned}
$$

Statue d'Euclide au musée de l'Université d'Oxford — Photographié par Mark A. Wilson (CC-BY-SA 4.0)

ALGORITHME D'EUCLIDE ÉTENDU

- Résoudre $899x + 493y = 29$
- Notons $a = 899, b = 493$ et $c = 29$

$$
\begin{aligned}
899 &= 493 \times 1 + 406 \\
493 &= 406 \times 1 + 87 \\
406 &= 87 \times 4 + 58 \\
87 &= 58 \times 1 + 29 \\
58 &= 29 \times 2 + 0
\end{aligned}
$$

Statue d'Euclide au musée de l'Université d'Oxford — Photographié par Mark A. Wilson (CC-BY-SA 4.0)

ALGORITHME D'EUCLIDE ÉTENDU

- Résoudre $899x + 493y = 29$
- Notons $a = 899, b = 493$ et $c = 29$

$$
\begin{aligned}
a &= b \times 1 + 406 \\
b &= 406 \times 1 + 87 \\
406 &= 87 \times 4 + 58 \\
87 &= 58 \times 1 + c
\end{aligned}
$$

Statue d'Euclide au musée de l'Université d'Oxford — Photographié par Mark A. Wilson (CC-BY-SA 4.0)

ALGORITHME D'EUCLIDE ÉTENDU

- Résoudre $899x + 493y = 29$
- Notons $a = 899, b = 493$ et $c = 29$

$$\mathbf{406} = a - b \times 1$$
$$87 = b - \mathbf{406} \times 1$$
$$58 = 406 - 87 \times 4$$
$$c = 87 - 58 \times 1$$

Statue d'Euclide au musée de l'Université d'Oxford — Photographié par Mark A. Wilson (CC-BY-SA 4.0)

ALGORITHME D'EUCLIDE ÉTENDU

- Résoudre $899x + 493y = 29$
- Notons $a = 899$, $b = 493$ et $c = 29$

$$
\begin{aligned}
406 &= a - b \times 1 \\
87 &= b - (a - b \times 1) \times 1 \\
58 &= 406 - 87 \times 4 \\
c &= 87 - 58 \times 1
\end{aligned}
$$

Statue d'Euclide au musée de l'Université d'Oxford — Photographié par Mark A. Wilson (CC-BY-SA 4.0)

ALGORITHME D'EUCLIDE ÉTENDU

- Résoudre $899x + 493y = 29$
- Notons $a = 899, b = 493$ et $c = 29$

$$\mathbf{406} = a - b \times 1$$
$$\mathbf{87} = -a + b \times 2$$
$$58 = \mathbf{406} - \mathbf{87} \times 4$$
$$c = 87 - 58 \times 1$$

Statue d'Euclide au musée de l'Université d'Oxford — Photographié par Mark A. Wilson (CC-BY-SA 4.0)

ALGORITHME D'EUCLIDE ÉTENDU

- Résoudre $899x + 493y = 29$
- Notons $a = 899, b = 493$ et $c = 29$

$$
\begin{aligned}
406 &= a - b \times 1 \\
87 &= -a + b \times 2 \\
58 &= (a - b \times 1) - (-a + b \times 2) \times 4 \\
c &= 87 - 58 \times 1
\end{aligned}
$$

Statue d'Euclide au musée de l'Université d'Oxford — Photographié par Mark A. Wilson (CC-BY-SA 4.0)

ALGORITHME D'EUCLIDE ÉTENDU

- Résoudre $899x + 493y = 29$
- Notons $a = 899, b = 493$ et $c = 29$

$$
\begin{aligned}
406 &= a - b \times 1 \\
\mathbf{87} &= -a + b \times 2 \\
\mathbf{58} &= a \times 5 - b \times 9 \\
c &= \mathbf{87} - \mathbf{58} \times 1
\end{aligned}
$$

Statue d'Euclide au musée de l'Université d'Oxford — Photographié par Mark A. Wilson (CC-BY-SA 4.0)

ALGORITHME D'EUCLIDE ÉTENDU

- Résoudre $899x + 493y = 29$
- Notons $a = 899, b = 493$ et $c = 29$

$$
\begin{aligned}
406 &= a - b \times 1 \\
87 &= -a + b \times 2 \\
58 &= a \times 5 - b \times 9 \\
c &= (-a + b \times 2) - (a \times 5 - b \times 9) \times 1
\end{aligned}
$$

Statue d'Euclide au musée de l'Université d'Oxford — Photographié par Mark A. Wilson (CC-BY-SA 4.0)

ALGORITHME D'EUCLIDE ÉTENDU

- Résoudre $899x + 493y = 29$
- Notons $a = 899, b = 493$ et $c = 29$

$$\begin{aligned} 406 &= a - b \times 1 \\ 87 &= -a + b \times 2 \\ 58 &= a \times 5 - b \times 9 \\ c &= -a \times 6 + b \times 11 \end{aligned}$$

- Solution : $x = -6, y = 11$
- Complexité : $O(\log(a + b))$

Statue d'Euclide au musée de l'Université d'Oxford — Photographié par Mark A. Wilson (CC-BY-SA 4.0)

COEFFICIENTS DE BÉZOUT EN PYTHON

```python
def bezout(a, b):
    px, py = 1, 0
    x, y = 0, 1
    while b:
        a, b, q = b, a % b, a // b
        px, py, x, y = x, y, px - q * x, py - q * y
    return a, px, py
```

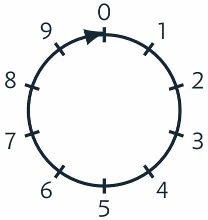
EXERCICES ET RÉFÉRENCES

- Exercices :
  - Codeforces 633A, "Ebony and Ivory"
  - UVa 10090, "Marbles"
- Dans les livres de référence :
  - Dürr et Vie, §14.2
  - Laaksonen, §11.1.6
  - Halim, §5.5.9

PLAN

ARITHMÉTIQUE MODULAIRE

- Système dans lequel on revient à 0 dès qu'on atteint un *n* fixé.



- Permet de faire des calculs sur des nombres de taille bornée.
- Souvent utilisée dans les problèmes mathématiques des concours.

QUELQUES RÈGLES DU MODULO

- L'opérateur modulo est linéaire et se comporte intuitivement la plupart du temps.
- *Addition*
  - $a + b = c \Rightarrow (a \bmod n) + (b \bmod n) \equiv c \pmod{n}$
  - $a \equiv b \pmod{n} \Rightarrow a + k \equiv b + k \pmod{n}$
  - $a \equiv b \pmod{n} \land c \equiv d \pmod{n} \Rightarrow a + c \equiv b + d \pmod{n}$
- *Multiplication*
  - $a \times b = c \Rightarrow (a \bmod n) \times (b \bmod n) \equiv c \pmod{n}$
  - $a \equiv b \pmod{n} \Rightarrow ka \equiv kb \pmod{n}$
  - $a \equiv b \pmod{n} \land c \equiv d \pmod{n} \Rightarrow ac \equiv bd \pmod{n}$
- *Exponentiation*
  - $a \equiv b \pmod{n} \Rightarrow a^k \equiv b^k \pmod{n} \quad [k \geq 0]$

INVERSE MODULAIRE

- Qu’est-ce que l’inverse d’un nombre a ?

INVERSE MODULAIRE

- Qu'est-ce que l'inverse d'un nombre a ?
  - Sur les rationnels et les réels, l'inverse de a est $1/a$

INVERSE MODULAIRE

- Qu'est-ce que l'inverse d'un nombre a ?
  - Sur les rationnels et les réels, l'inverse de a est $1/a$
  - Un nombre x qui « annule » l'effet multiplicatif de a

INVERSE MODULAIRE

- Qu'est-ce que l'inverse d'un nombre a ?
  - Sur les rationnels et les réels, l'inverse de a est $1/a$
  - Un nombre x qui « annule » l'effet multiplicatif de a
  - Une solution à l'équation $ax = 1$

- En arithmétique modulaire, les nombres ont des inverses entiers !
  - Ex : Pour un module $n = 20$, 7 est l'inverse de 3
    $$7 \times 3 = 21 \equiv 1 \pmod{20}$$
  - Comment trouver l'inverse de a modulo n ?

INVERSE MODULAIRE

- Qu'est-ce que l'inverse d'un nombre a ?
  - Sur les rationnels et les réels, l'inverse de a est $1/a$
  - Un nombre x qui « annule » l'effet multiplicatif de a
  - Une solution à l'équation $ax = 1$

- En arithmétique modulaire, les nombres ont des inverses entiers !
  - Ex : Pour un module $n = 20$, 7 est l'inverse de 3
    $7 \times 3 = 21 \equiv 1 \pmod{20}$
  - Comment trouver l'inverse de a modulo n ?

$$ax \equiv 1 \pmod{n}$$
$$ax + qn = 1 \quad \rightsquigarrow \text{identité de Bézout !}$$

INVERSE MODULAIRE EN PYTHON

```python
def inv(a, n):
    gcd, inv, _ = bezout(a, n)
    if gcd != 1:
        return None
    return inv % n
```

BIO-RYTHMES

Trois cycles courent en parallèle chez une personne : *physique*, *émotionnel* et *intellectuel*. Ils atteignent un pic respectivement tous les 23, 28 et 33 jours. Sachant le numéro de trois jours où se sont produits des pics pour chacun des trois cycles, ainsi que le numéro du jour actuel, dans combien de jours se produira le prochain pic simultané ?

Entrée Quatre entiers p, e, i et d, les numéros des jours où se sont produits les pics des trois types et le numéro du jour actuel.

Limites $0 \leq p, e, i, d \leq 365$. Temps : 3 s.

Sortie « the next triple peak occurs in x days. », où x est votre réponse.

Tiré de UVa 756, "Biorhythms"

PENSEZ MODULO

- Pouvez-vous reformuler le problème en arithmétique modulaire ?

- Pouvez-vous reformuler le problème en arithmétique modulaire ?
- Si k est le numéro d'un jour de triple-pic, il vérifie les équations

$$\begin{cases} k \equiv p \pmod{23} \\ k \equiv e \pmod{28} \\ k \equiv i \pmod{33} \end{cases}$$

- Le **théorème des restes chinois** nous permet de résoudre les systèmes d'équations de cette forme

THÉORÈME DES RESTES CHINOIS

- Soit un système d'équations modulaires de la forme

$$\begin{cases} x \equiv a_1 \pmod{n_1} \\ \vdots \\ x \equiv a_k \pmod{n_k} \end{cases}$$

- Si $n_1, ..., n_k$ sont premiers entre eux, une solution est

$$x = a_1 X_1 \ \mathrm{inv}_{n_1}(X_1) + \cdots + a_k X_k \ \mathrm{inv}_{n_k}(X_k)$$
$$\text{où } \ X_i = (n_1 \times \cdots \times n_k)/n_i$$

- D'autres solutions obtenues en ajoutant ou retranchant $n_1 \times \cdots \times n_k$

THÉORÈME DES RESTES CHINOIS EN PYTHON

```python
from math import prod
def crt(a, n):
    x = 0
    p = prod(n)
    for ai, ni in zip(a, n):
        xi = p // ni
        x += ai * xi * inv(xi, ni)
    return x
```

EXERCICES ET RÉFÉRENCES

- Exercices :
  - AoC 2020-13, “Shuttle Search”
  - **UVa 756, “Biorhythms”**
- Dans les livres de référence :
  - Laaksonen, §11.1.6

PLAN

NOMBRES DE CATALAN

$$C_n = \frac{1}{n+1}\binom{2n}{n} = \frac{(2n)!}{(n+1)!\,n!} = \prod_{k=2}^{n} \frac{n+k}{k} \qquad \text{pour } n \geq 0.$$

Liste non-exhaustive :

- Nombre de mots de Dyck de longueur $2n$
- Nombre d'arbres binaires entiers à $n+1$ feuilles
- Nombre d'arbres planaires enracinés à n arêtes
- ...

- Pas possible de tout retenir, il faut utiliser votre plaquette SWERC
- Exercice : **Bracket Sequences I**

NOMBRES DE CATALAN

$$C_n = \frac{1}{n+1}\binom{2n}{n} = \frac{(2n)!}{(n+1)!\,n!} = \prod_{k=2}^{n} \frac{n+k}{k} \qquad \text{pour } n \geq 0.$$

Liste non-exhaustive :

- Nombre de mots de Dyck de longueur $2n$
- Nombre d'arbres binaires entiers à $n+1$ feuilles
- Nombre d'arbres planaires enracinés à n arêtes
- ...

- Pas possible de tout retenir, il faut utiliser votre plaquette SWERC
- Exercice : **Bracket Sequences I**
- Plus de détails et de problèmes : [Tutorial] Catalan Numbers and Catalan Convolution

PLAN

PIVOT DE GAUSS

On veut déterminer les solutions d'un système d'équations linéaires à n variables :

$$a_{1,1}x_1 + a_{1,2}x_2 + \ldots + a_{1,n}x_n = b_1 \tag{1}$$
$$a_{2,1}x_1 + a_{2,2}x_2 + \ldots + a_{2,n}x_n = b_2 \tag{2}$$
$$\ldots \tag{3}$$
$$a_{n,1}x_1 + a_{n,2}x_2 + \ldots + a_{n,n}x_n = b_n \tag{4}$$

Représentation en matrice :

$$\left[\begin{array}{cccc|c} a_{1,1} & a_{1,2} & \ldots & a_{1,n} & b_1 \\ a_{2,1} & a_{2,2} & \ldots & a_{1,n} & b_2 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ a_{n,1} & a_{n,2} & \ldots & a_{n,n} & b_n \end{array}\right]$$

PIVOT DE GAUSS

Pour résoudre le système, on veut transformer la matrice :

$$\left[\begin{array}{cccc|c} a_{1,1} & a_{1,2} & \dots & a_{1,n} & b_1 \\ a_{2,1} & a_{2,2} & \dots & a_{1,n} & b_2 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ a_{n,1} & a_{n,2} & \dots & a_{n,n} & b_n \end{array}\right]$$

En :

$$\left[\begin{array}{cccc|c} 1 & 0 & \dots & 0 & c_1 \\ 0 & 1 & \dots & 0 & c_2 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & 1 & c_n \end{array}\right]$$

PIVOT DE GAUSS

Pour l'échelonnement, trois types d'opérations :

- Échange de deux lignes
- Multiplication d'une ligne par un scalaire non nul
- Ajout du multiple d'une ligne à une autre ligne

Exercice : résoudre le système d'équations

$$2x_1 + 4x_2 + x_3 = 16 \tag{5}$$
$$x_1 + 2x_2 + 5x_3 = 17 \tag{6}$$
$$3x_1 + x_2 + x_3 = 8 \tag{7}$$

PIVOT DE GAUSS

Pour l'échelonnement, trois types d'opérations :

- Échange de deux lignes
- Multiplication d'une ligne par un scalaire non nul
- Ajout du multiple d'une ligne à une autre ligne

Exercice : résoudre le système d'équations

$$\left[\begin{array}{ccc|c} 2 & 4 & 1 & 16 \\ 1 & 2 & 5 & 17 \\ 3 & 1 & 1 & 8 \end{array}\right] \quad \Bigg| \quad \left[\begin{array}{ccc|c} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 3 \\ 0 & 0 & 1 & 2 \end{array}\right]$$

Matrice en entrée / Matrice échelonnée

PIVOT DE GAUSS

Pour l'échelonnement, trois types d'opérations :

- Échange de deux lignes
- Multiplication d'une ligne par un scalaire non nul
- Ajout du multiple d'une ligne à une autre ligne

Ces 3 opérations sont utilisées pour réaliser l'algorithme de Gauss-Jordan (complexité $O(n^3)$) :
`https://github.com/jilljenn/tryalgo/blob/master/tryalgo/gauss_jordan.py`

Exercice : Por Costel and Bujor