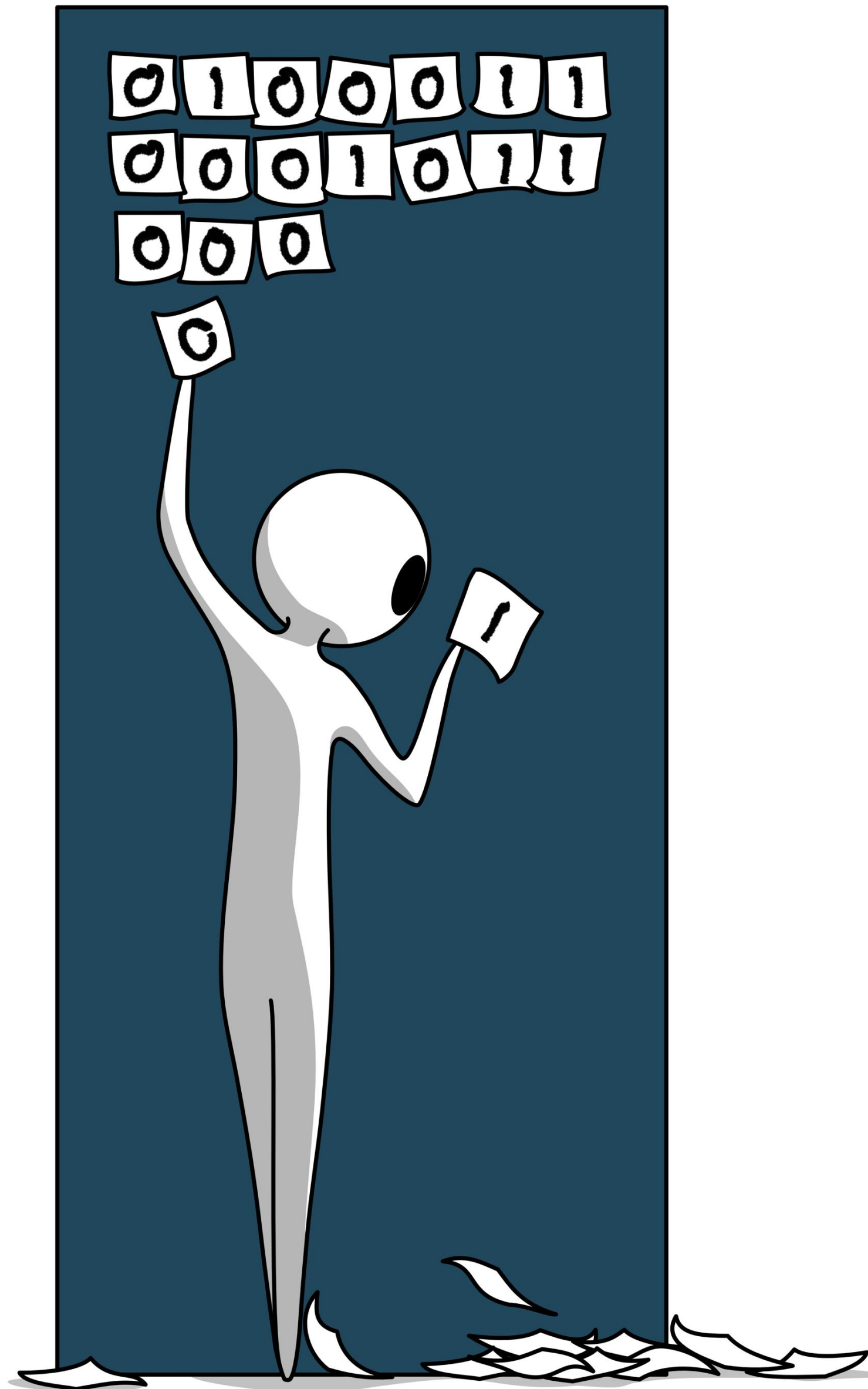




Information, Calcul et Communication

Introduction

Olivier Lévêque
et
Sébastien Doeraene

EPFL Pourquoi un cours d'introduction à l'informatique ?

- 4e pilier de la culture (après la lecture, l'écriture et l'arithmétique)
- Elle constitue désormais une discipline scientifique à part entière: la science du traitement automatique de l'information.
- L'informatique a non seulement changé notre société, mais aussi notre façon de faire de la science.
- De nos jours, tout.e ingénieur.e qui maîtrise les sciences du numérique a clairement un avantage sur les autres...

EPFL Plan du cours (partie théorique)

Première partie : Calcul (introduction aux algorithmes)

- Ingrédients de base
- Complexité temporelle
- Récursivité
- Programmation dynamique
- Calculabilité
- Classes de complexité
- Méthodes d'approximation

Seconde partie : Information et communication

- Représentation de l'information
- Architecture des ordinateurs
- Échantillonnage
et reconstruction de signaux
- Entropie
et compression de données
- Correction d'erreurs
- Cryptographie et sécurité

Horaires (partie théorique)

- Cours :

les vendredis après-midi de 13h15 à 15h
en salle SG 1 (enregistrements disponibles après coup sur Mediaspace)

- Exercices:

les vendredis après-midi de 15h15 à 16h15+ en salles:

CO 015, CO 016, CO 017, CO 120, CO 121, CO 122, CO 123, CO 124

A-F

G-Z

17 enseignant·e·s sont là pour vous : ***profitez-en !***

- Mini-projet de programmation, valant pour 15% de la note finale
- Quiz sur la sécurité informatique, valant pour 5% de la note finale
- Examen final en juin-juillet, portant sur les deux parties du cours (théorie et programmation), valant pour 80% de la note finale.

EPFL Points de contact

- Ici à l'EPFL !
- **Moodle** : matériel de cours, vidéos, exercices, corrigés, références, ...
(de manière générale, vous trouverez là *toutes* les informations sur le cours)
- **Chaîne Mediaspace** avec vidéos pré-enregistrées
et enregistrements des cours
- **Forum EdDiscussion** : vous pouvez poser des questions à tout moment,
de manière anonyme si vous le désirez ; encore une fois, ***profitez-en !***
Des êtres humains vous répondent...

- Livre ``Découvrir le numérique'', EPFL Press, 2016
- Fichiers pdf avec transparents du cours, plus quelques compléments

Encore quelques conseils...

(que vous connaissez déjà sans doute)

- Votre participation active au cours et aux exercices est cruciale !
- Prenez des notes !
- N'hésitez pas à poser des questions ! pendant le cours aussi !
- Retravaillez le cours et les exercices après les séances...

Qu'est-ce qu'un algorithme ?

- Un algorithme n'est **pas** un programme.
- Un algorithme est la description des étapes **élémentaires** menant à la résolution d'un problème; c'est donc la description conceptuelle d'un programme.
- Un **programme** est l'implémentation d'un algorithme dans un langage donné et dans un système particulier.

Exemple 1: calcul du modulo 3 d'un grand nombre

Algo 1

$$\begin{array}{r} 243722 \quad | 3 \\ - 24 \\ \hline 03 \\ - 3 \\ \hline 07 \\ - 6 \\ \hline 12 \\ - 12 \\ \hline \text{reste} \rightarrow 02 \end{array} \quad \begin{array}{r} 8124 \\ \hline \text{quotient} \end{array}$$

reste $\rightarrow 02$

$$= 243722 \pmod{3}$$

Algo 2

$$2 + 4 + 3 + 7 + 2 + 2 = 20$$

$$\begin{array}{r} 20 \quad | 3 \\ - 18 \quad 6 \\ \hline \text{reste } 2 \end{array}$$

$$2 + 0 = 2$$

Explication : $43 = 4 \cdot 10 + 3$

$$\begin{aligned} 437 &= 4 \cdot 100 + 3 \cdot 10 + 7 &= 4(9+1) + 3 \\ &= 4(1+9) + 3(1+9) + 7 &= 4 \cdot 9 + 4 + 3 \end{aligned}$$

Exemple 2: recherche du minimum dans une liste

$$L = (11, 17, 12, 22, 8, 7, 77, 42, 15, 9)$$

$$\underline{\underline{n=10}}$$

$$L(1) \quad L(2) \quad L(3) \quad \dots \quad L(n)$$

$L(i)$ = valeur du nombre en position i dans la liste L

$$\text{min} \leftarrow L(1)$$

~~$$\text{Si } \text{min} > L(2), \text{ alors } \text{min} \leftarrow L(2)$$~~

~~$$\text{Si } \text{min} > L(3), \text{ alors } \text{min} \leftarrow L(3)$$~~

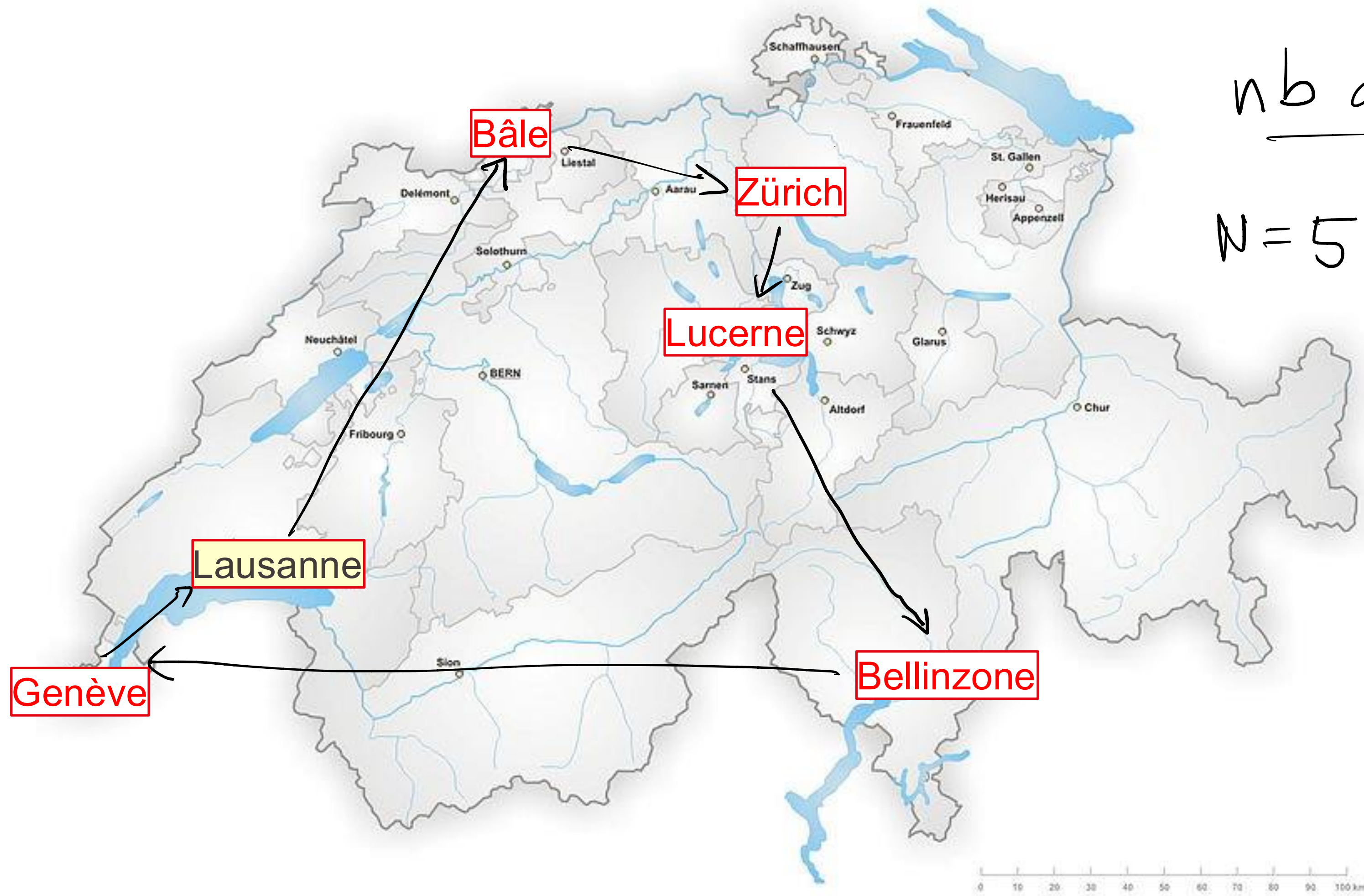
~~$$\dots$$~~

~~$$\text{Si } \text{min} > L(n), \text{ alors } \text{min} \leftarrow L(n)$$~~

$\boxed{\text{Sortir min}}$

Pour i allant de 2 à n :
 Si $\text{min} > L(i)$, alors $\text{min} \leftarrow L(i)$

Exemple 3: problème du voyageur de commerce



nb de possibilités:

$$N = 5 \cdot 4 \cdot 3 \cdot 2 \cdot 1$$

$$= 5! = \underline{\underline{120}}$$

Un journaliste américain a réalisé un reportage vidéo sur les transports publics en Suisse. Il s'est lancé le défi de traverser les 26 cantons en moins de 24 heures.



892



69



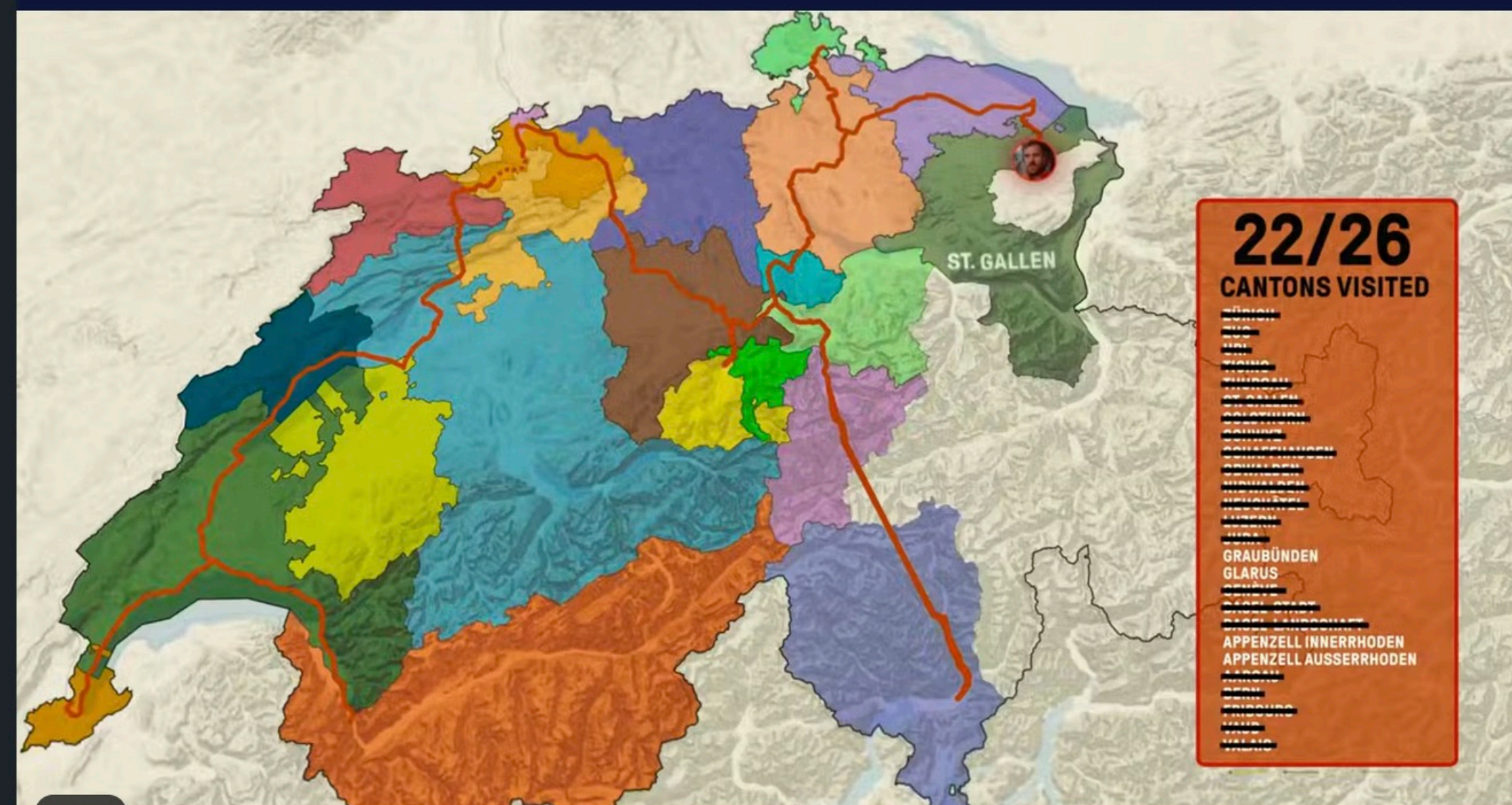
75



Marquer

$$\sim \underline{\underline{4 \cdot 10^{26}}}$$

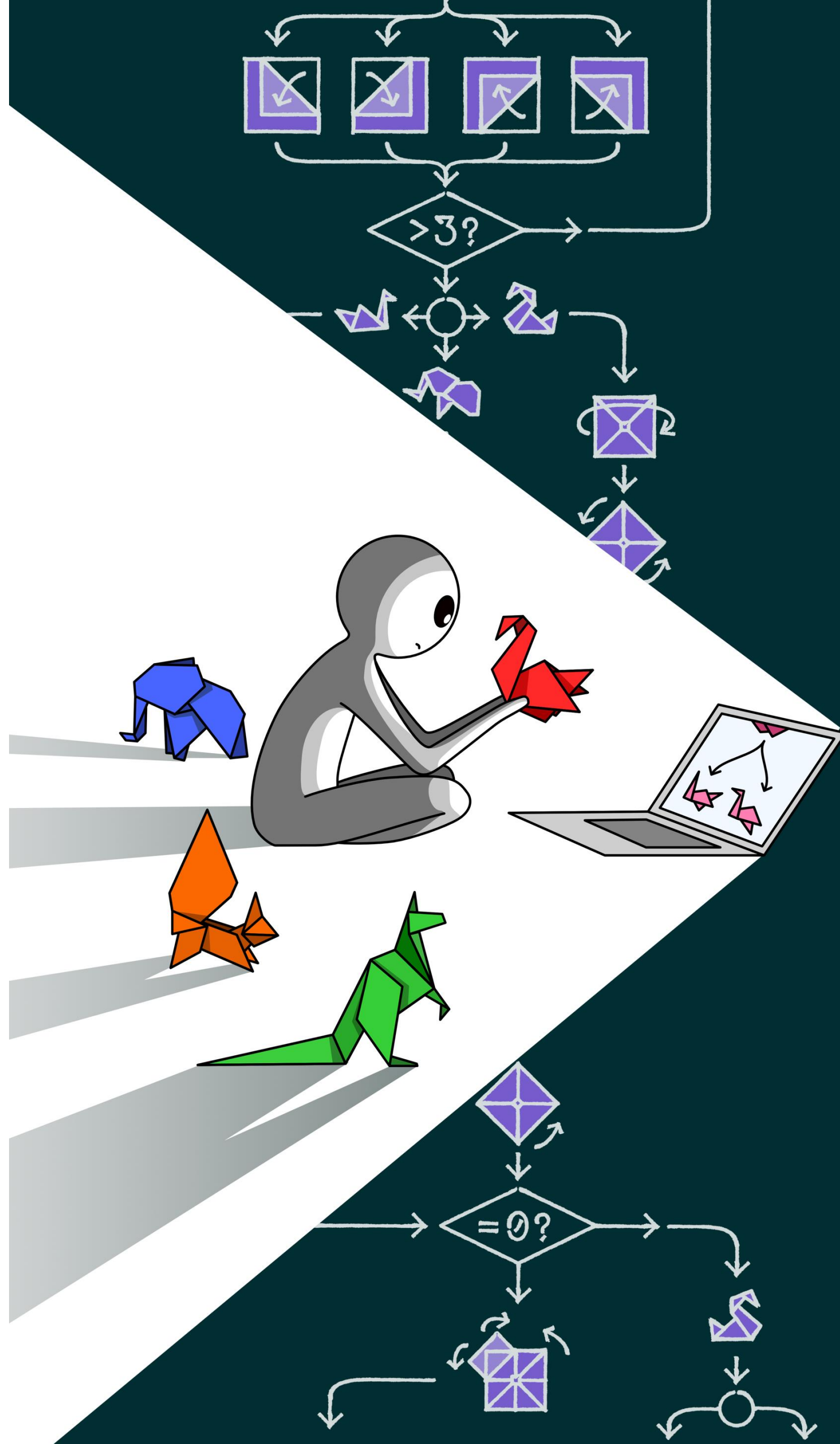
(extrait du journal 20 minutes, 25.12.25)



3 / 6

Il a traversé tous les cantons suisses en transports publics en 24 heures. Youtube / Johnny Harris





Information, Calcul et Communication

Algorithmes :
ingrédients de base

Olivier Lévêque

EPFL Algorithmes : ingrédients de base

Données

- Entrées : a, b, c
- Sorties : $\{x \in \mathbb{R} : ax^2 + bx + c = 0\}$
- Variables internes Δ

Instructions

- Affectations $\Delta \leftarrow b^2 - 4ac$
- Structures de contrôle
 - Branchements conditionnels (tests)
 - Itérations (boucles)
 - Boucles conditionnelles

Al-khwārizmī (~ 800)

Pb: trouver $\{x \in \mathbb{R} : ax^2 + bx + c = 0\}$

étant donné $a, b, c \in \mathbb{R}$ tq $a \neq 0$

Algo

entrée : a, b, c tq $a \neq 0$

sortie : $\{x \in \mathbb{R} : ax^2 + bx + c = 0\}$

$\Delta \leftarrow b^2 - 4ac$

si $\Delta < 0$, sortir $\emptyset$

sinon, si $\Delta = 0$, sortir $\{-\frac{b}{2a}\}$

sinon, sortir $\{-\frac{b \pm \sqrt{\Delta}}{2a}\}$

Boucles: Pour i allant de 1 à n :
(répéter ...)

Boucles conditionnelles: Tant que (condition vérifiée)
(répéter ...)

Sont équivalents:

Pour i allant de 1 à n

| Si $\text{min} > L(i)$
| alors $\text{min} \leftarrow L(i)$

$i \leftarrow 1$

Tant que $i \leq n$:

| Si $\text{min} > L(i)$
| alors $\text{min} \leftarrow L(i)$
| $i \leftarrow i + 1$

Illustration

■ Question :

Est-ce que tous les objets visibles sur cette photo sont différents les uns des autres ?



Illustration

■ **Question :**

Est-ce que tous les objets visibles sur cette photo sont différents les uns des autres ?

- **Question réciproque :**

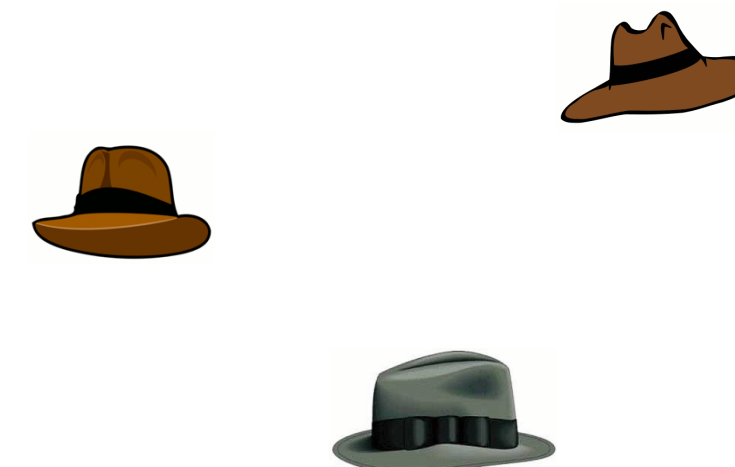
Y a-t-il au moins deux objets identiques sur cette photo ?



Tous différents ?

Problème à résoudre:

Parmi une liste de 3 objets, identifier si ceux-ci sont tous différents les uns des autres.



Algorithme

entrée : $L = (L(1), L(2), L(3))$ liste de 3 objets
sortie : valeur binaire oui/non

Si $L(1) = L(2)$, alors sortir non

(sinon) Si $L(1) = L(3)$, alors sortir non

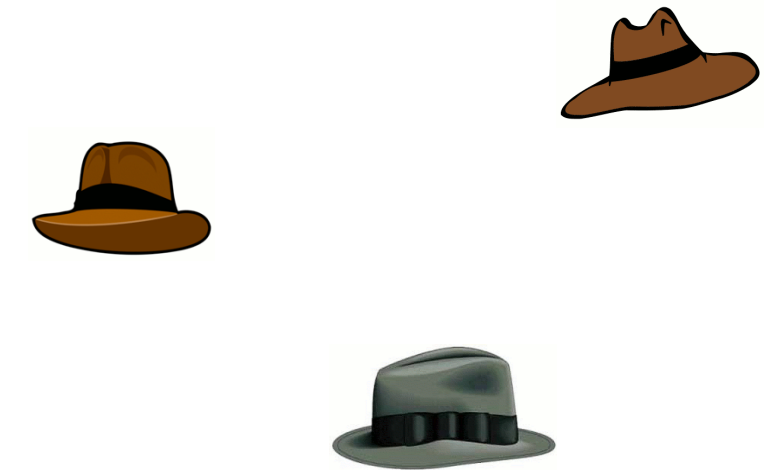
(sinon) Si $L(2) = L(3)$, alors sortir non

(sinon) sortir oui

Tous différents ?

Problème à résoudre:

Parmi une liste de 3 objets, identifier si ceux-ci sont tous différents les uns des autres.



Algorithme

entrée : $L = (L(1), L(2), L(3))$ liste de 3 objets
sortie : valeur binaire oui/non

```
 $s \leftarrow \text{oui}$   
Si  $L(1) = L(2)$ , alors :  $s \leftarrow \text{non}$   
Si  $L(1) = L(3)$ , alors :  $s \leftarrow \text{non}$   
Si  $L(2) = L(3)$ , alors :  $s \leftarrow \text{non}$   
Sortir :  $s$ 
```

Parmi une liste de n objets, identifier si ceux-ci sont tous différents les uns des autres.

Algorithme

entrée : L liste de n objets, n taille de la liste
sortie : valeur binaire oui/non

$$x \leftarrow l(i)$$

Pour j allant de $i+1$ à n ,

↳ $L(j) = x$, sortir nœud

Scatter and


$$\begin{array}{ll} i=1: & j=2 \dots n \\ i=2: & j=3 \dots n \end{array}$$
$$i = n-1 : \quad \gamma_i = n$$

↑ ↑
η-γ ✓

Tous différents ? (bis)

Problème à résoudre:

Parmi une liste de n objets, identifier si ceux-ci sont tous différents les uns des autres.

Algorithme

entrée : L liste de n objets, n taille de la liste
 sortie : valeur binaire oui/non

~~$s \leftarrow \text{oui}$~~

Pour i allant de 1 à $n - 1$:

|| Pour j allant de $i + 1$ à n :

|| || Si $L(i) = L(j)$, alors : ~~$s \leftarrow \text{non}$~~

Sortir : ~~s~~ oui ~~Sortir non~~



EPFL Algorithme d'Euclide

L'algorithme d'Euclide utilise une boucle conditionnelle pour trouver le plus grand diviseur commun (pgdc) de deux nombres entiers.

Rem : $\text{pgdc}(72, 60)$

$$\begin{array}{r|l} 72 & (2) \\ 36 & (2) \\ 18 & 2 \\ 9 & (3) \\ 3 & 3 \\ 1 & \end{array} \quad \begin{array}{r|l} 60 & (2) \\ 30 & (2) \\ 15 & (3) \\ 5 & 5 \\ 1 & \end{array}$$

$$\text{pgdc} = 2 \cdot 2 \cdot 3 = \underline{\underline{12}}$$

EPFL Algorithme d'Euclide

L'algorithme d'Euclide utilise une boucle conditionnelle pour trouver le plus grand diviseur commun (pgdc) de deux nombres entiers.

Algorithme

entrée : a, b deux nombres entiers positifs

sortie : $\text{pgdc}(a, b)$

Tant que $b \neq 0$:

$\text{temp} \leftarrow b$

$b \leftarrow a \bmod b = \text{reste de la division de } a \text{ par } b$

$a \leftarrow \text{temp}$

Sortir : a

$\text{pgdc}(72, 60)$

$b \neq 0$? oui

$\text{temp} \leftarrow 60$

$b \leftarrow 12$

$a \leftarrow 60$

$b \neq 0$? oui

$\text{temp} \leftarrow 12$

$b \leftarrow 0$

$a \leftarrow 12$

$b \neq 0$? non

Sortir 12