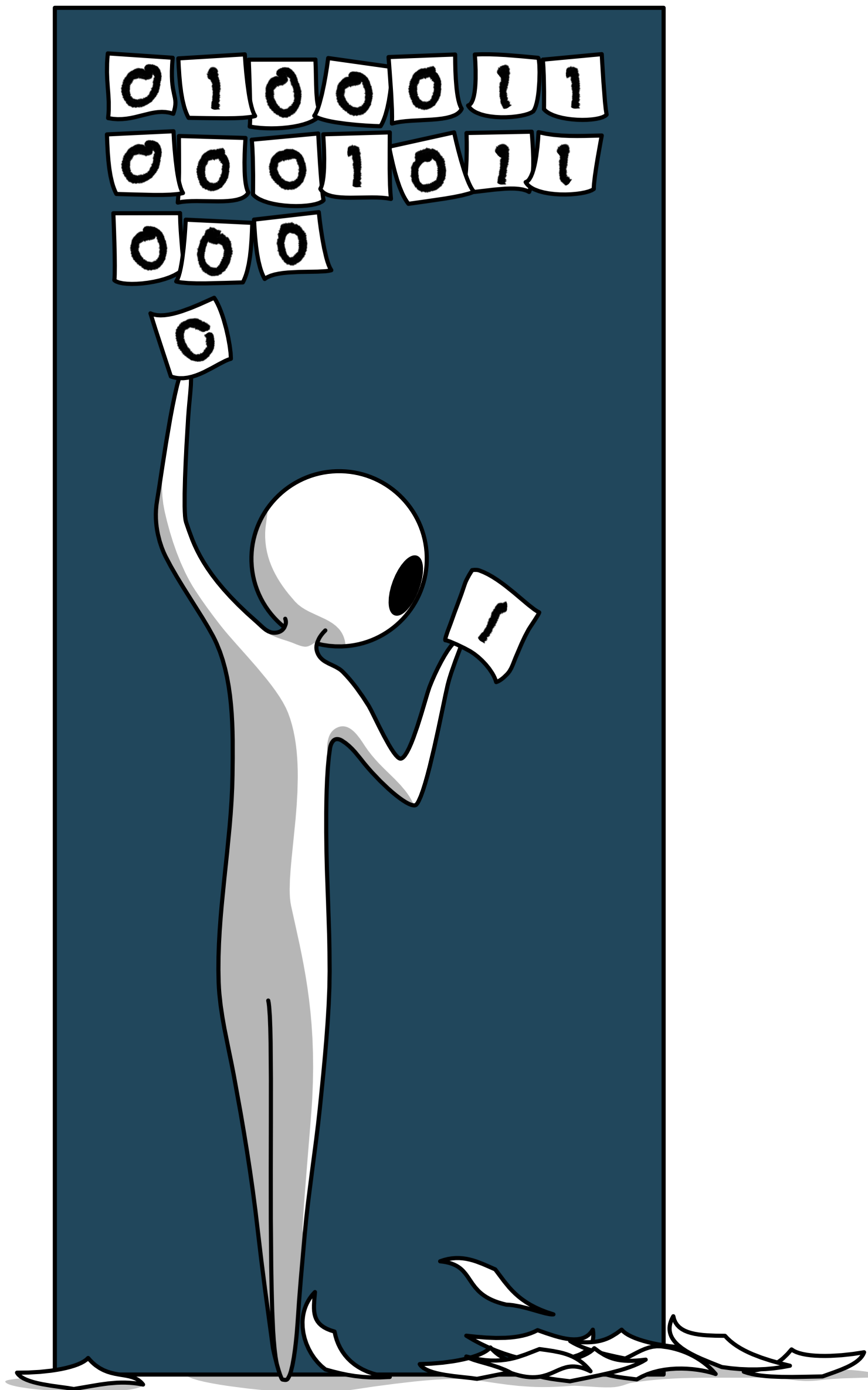




Information, Calcul et Communication

Introduction

Mirjana Stojilovic et Olivier Lévêque

Pourquoi donc vous propose-t-on 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...

Programme du semestre (théorie et programmation)

- Du vendredi 11 septembre au vendredi 30 octobre: 7 semaines de cours (+ quiz sur la sécurité informatique, comptant pour 5% de la note finale)

- Vendredi 6 novembre, 9h15-12h15:
test intermédiaire (comptant pour 40% de la note finale)
- Du vendredi 6 novembre au vendredi 11 décembre: 6 semaines de cours (inclus un projet de programmation, comptant pour 15% de la note finale)
- Vendredi 18 décembre, 9h15-12h15:
test final (comptant pour 40% de la note finale)

EPFL Points de contact

- **EPFL:** venez au cours ET aux séances d'exercices!
- **Moodle:** matériel de cours, vidéos, exercices, références, annonces
- **Ed Discussion:** forum sur lequel vous pouvez poser des questions à tout moment: profitez-en!

EPFL Horaires (partie théorique)

- Cours :

les vendredis après-midis de 13h15 à 15h, en salle CO 1

- Exercices :

séances d'exercices régulières les vendredis de 15h15 à **16h15+**,
en salles AAC 1 37 (!), AAC 1 32, CO 0 15, CO 0 16 et CO 017

→ 16h15

- Note : Les exercices “pour le plaisir” des séries sont... pour le plaisir !

Equipe pédagogique (partie théorique)

Nom	Prénom	Section	Semestre
Beaufils	Timothée	MX	BA3
Asnake	Solomée	GC	BA3
Laville	Mathis	GC	BA3
Capol	Thomas	GC	BA3
Motta Pacheco	Cleo	MX	BA3
Richard	Yorick	MX	BA3
Kuder	Oscar	MX	BA3
Meng	Rachel	GC	BA3
Delcroix	Martina	GC	BA3
Taibi	Ali	GC	BA3
Linette	Adèle	SV	MA1
Verga	Loris	IN	MA1
El Harti	Sara	DS	MA1
Coste	Célia	GC	MA1
Fushnari	Margaux	GC	MA1

EPFL Références (partie théorique)

- Livre ``Découvrir le numérique'', EPFL Press, 2016
- Vidéos pré-enregistrées sur mediaspace.epfl.ch
- MOOC sur courseware.epfl.ch
- Fichiers pdf avec transparents du cours + quelques pdfs additionnels

(Tous les liens sont disponibles sur la page Moodle du cours.)

Encore quelques conseils...

- Votre participation active au cours et aux exercices est cruciale !
- Prenez des notes !
- N'hésitez pas à poser des questions !
- Retravaillez le cours et les exercices à la maison !
- Ne ratez pas le train...

EPFL Et parlons aussi un peu de l'éléphant dans le couloir...

- Depuis novembre 2022, l'IA générative est accessible à tout le monde.
- Les différents outils disponibles à l'heure actuelle sont tout simplement prodigieux... mais ils n'apprennent pas à votre place !
- 2-3 conseils:
 - essayez toujours de d'abord résoudre un problème *par vous-mêmes* ;
 - utilisez ces outils *parcimonieusement* (la planète vous en remerciera) ;
 - enfin, soyez *critiques* envers ce que vous lisez !

EPFL Contenu des six prochaines semaines de cours (partie théorique)

Introduction aux algorithmes:

- Ingrédients de base
- Complexité temporelle
- Récursivité
- Classes de complexité
- Méthodes d'approximation
- Algorithmes probabilistes

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

$$\begin{array}{r}
 10) \quad 100'213 \quad \overline{) 3} \\
 \quad \quad 33404 \\
 \quad \quad \underline{- 9} \\
 \quad \quad 10 \\
 \quad \quad \underline{- 9} \\
 \quad \quad 12 \\
 \quad \quad \underline{- 12} \\
 \quad \quad 01 \\
 \quad \quad \underline{- 00} \\
 \quad \quad 13 \\
 \quad \quad \underline{- 12} \\
 \text{Reste : } \textcircled{1}
 \end{array}$$

$$20) \quad 1 + 0 + 0 + 2 + 1 + 3 = 7$$

$$\begin{array}{r}
 7 \quad \overline{) 13} \\
 \underline{- 6} \quad 2 \\
 \textcircled{1}
 \end{array}$$

$$\begin{aligned}
 \underline{43} &= 4 \cdot 10 + 3 \cdot 1 \\
 &= 4 \cdot (9 + 1) + 3 \cdot 1 \\
 &= \cancel{4 \cdot 9} + \underline{4} + 3 \\
 &\quad \text{multiple de 3}
 \end{aligned}$$

$$30) \quad 1000 \underline{21} \cancel{3} \rightarrow \textcircled{1}$$

$$100213$$

$$= 100000 + 200 + 10 + 3$$

$$= \cancel{99999} + 1 + 2(\cancel{99} + 1) + 1 \cdot (\cancel{9} + 1) + 3$$

$$= 1 + 2 + 1 + 3$$

EPFL Exemple 2: recherche du minimum dans une liste

$L = (\underline{43}, \underline{17}, 22, \underline{8}, 41, 16, \underline{3}, 17, 22)$

$n = 9$ nombres

$\left\{ \begin{array}{l} \text{plus_petit} \leftarrow L(1) \\ \text{Parcourir la liste } L; \text{ à l'étape } i \quad (1 \leq i \leq n-1): \\ \quad \text{Si } L(i+1) < \text{plus_petit} \\ \quad \quad \text{alors } \text{plus_petit} \leftarrow L(i+1) \\ \text{Sortir plus_petit} \end{array} \right.$

Minimum d'une liste L de taille n

$min \leftarrow L(1)$

Pour i allant de 2 à n :

si $L(i) < min$,

alors $min \leftarrow L(i)$

Sortir min

Exemple 3: problème du voyageur de commerce

$n = 5$ villes

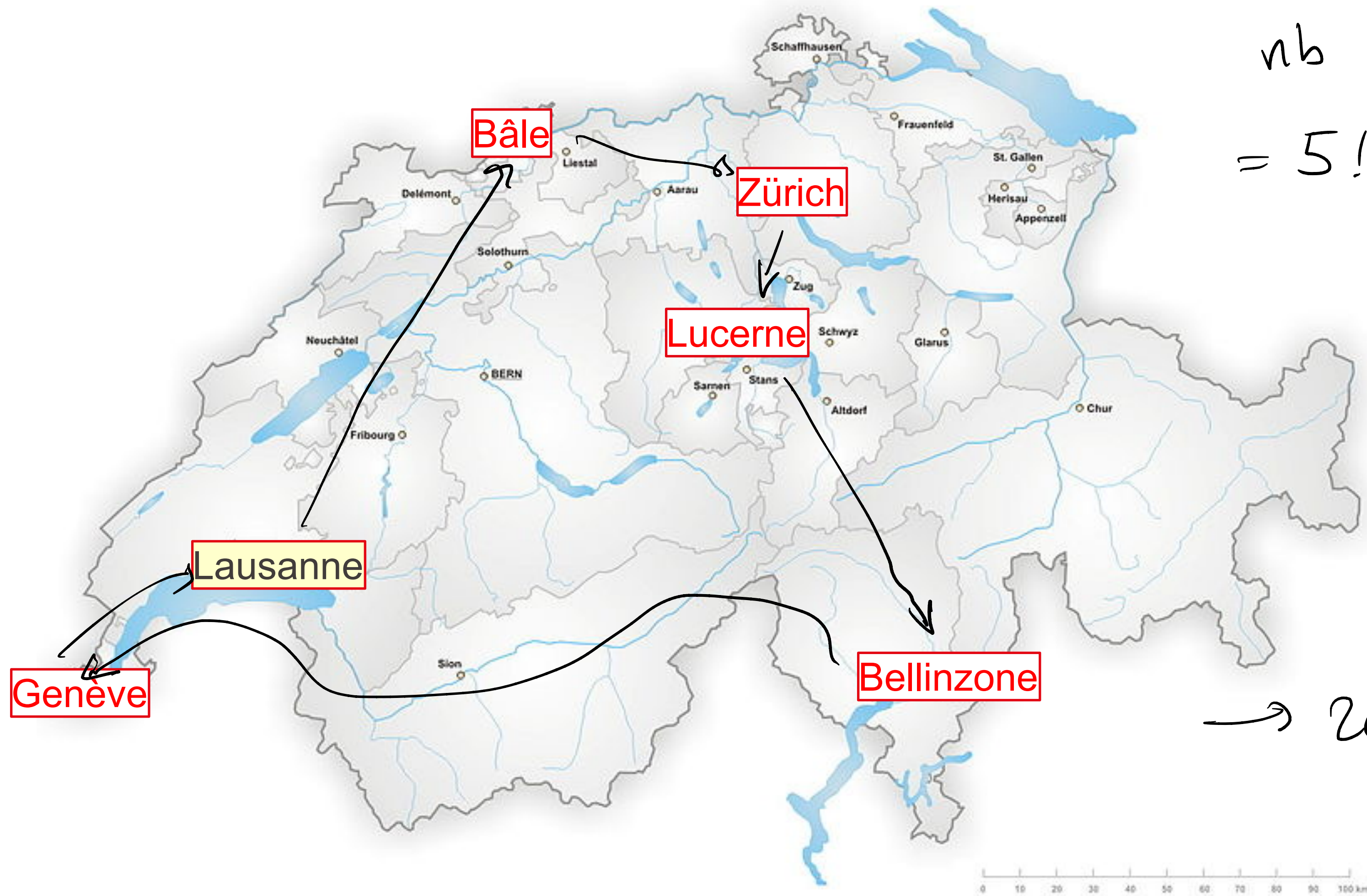
nb possibilités

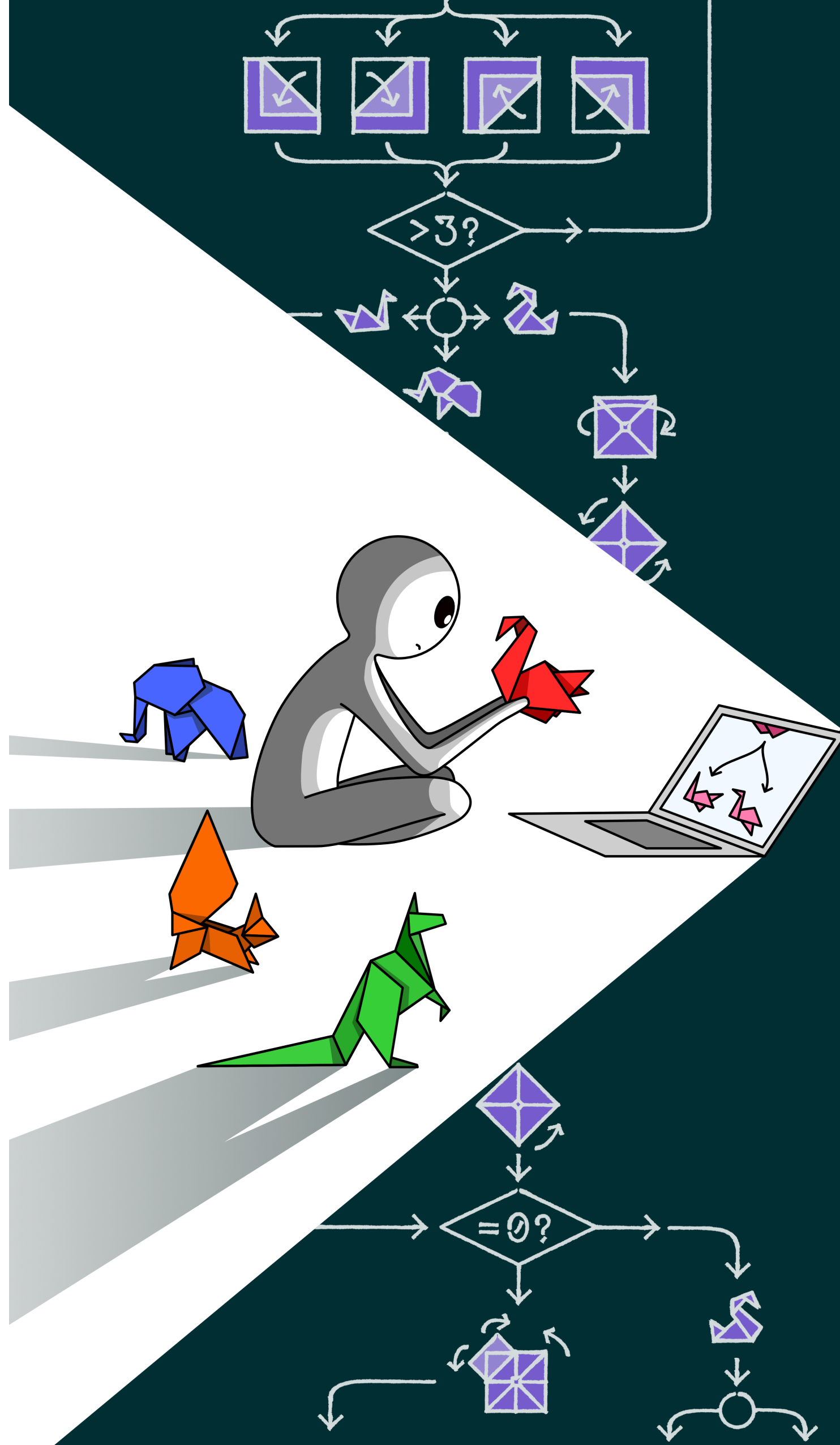
$$= 5! = 5 \cdot 4 \cdot 3 \cdot 2 \cdot 1$$

$$= \underline{\underline{120}}$$

$$\underline{n = 26}$$

$$\rightarrow 26! \approx 4 \cdot 10^{26}$$





Information, Calcul et Communication

Algorithmes :
ingrédients de base

Olivier Lévêque

EPFL Algorithmes : ingrédients de base (pseudo-code)

Données

- Entrées a, b, c
- Sorties $\{x : \dots\}$
- Variables internes Δ

Instructions

- Affectations $\Delta \leftarrow b^2 - 4ac$
- Structures de contrôle
 - Branchements conditionnels (tests) *si...*
 - Itérations (boucles) *Pour i allant de 1 à n*
 - Boucles conditionnelles
Tant que (...), répéter

Résolution de $ax^2 + bx + c = 0$

entrées : $a, b, c \in \mathbb{R}, a \neq 0$

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

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

Si $\Delta < 0$, alors sortir $\emptyset$

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

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

■ Question :

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

■ Question réciproque :

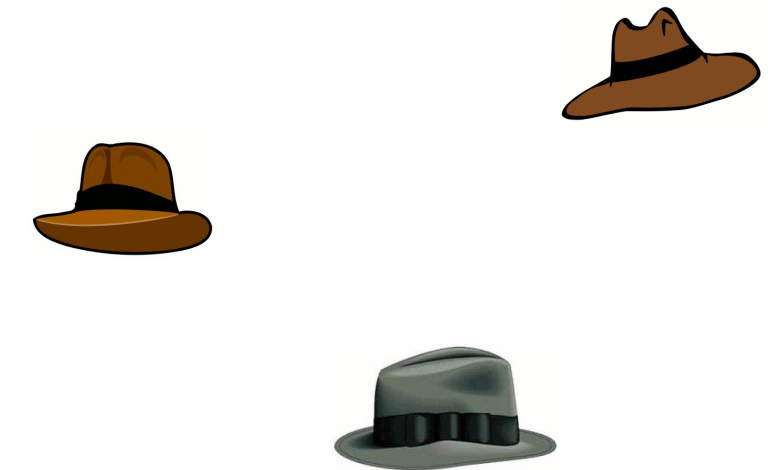
Y a-t-il au moins deux objets identiques sur cette photo ?



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

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

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

Sortir oui

~~Si $L(1) \neq L(2)$, sortir oui~~

~~Si $L(1) \neq L(3)$, sortir oui~~

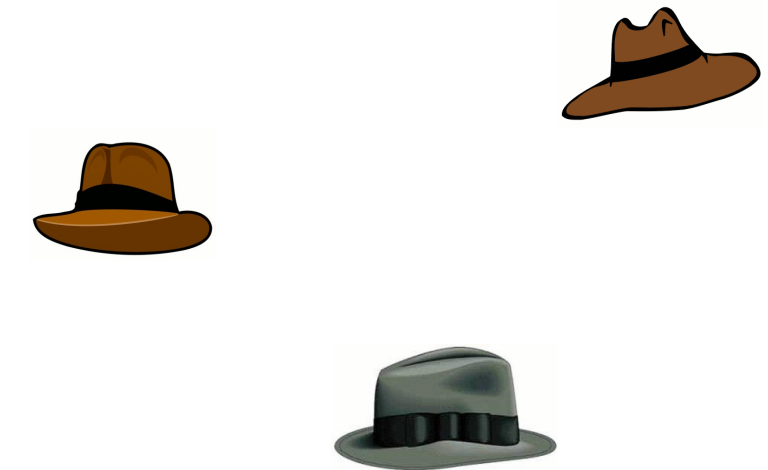
~~Si $L(2) \neq L(3)$, sortir oui~~

~~Sortir non~~

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

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



EPFL Algorithme d'Euclide

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

$$\begin{array}{r|l} 72 & 2 \\ 36 & 2 \\ 18 & 2 \\ 9 & 3 \\ 3 & 3 \\ 1 & \end{array}$$

$$\begin{array}{r|l} 50 & 2 \\ 25 & 5 \\ 5 & 5 \\ 1 & \end{array}$$

$$\Rightarrow \text{pgcd}(72, 50) = 2$$

NB: Ceci n'est pas l'algorithme d'Euclide !

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 $b \neq 0$:

$\text{temp} \leftarrow b$

$b \leftarrow a \bmod b$

$a \leftarrow \text{temp}$

Sortir : a

$\text{pgdc}(70, 52)$

$b \neq 0$? oui

$\text{temp} \leftarrow 52$

$b \leftarrow 70 \bmod 52 = 18$

$a \leftarrow 52$

$b \neq 0$? oui

$\text{temp} \leftarrow 18$

$b \leftarrow 52 \bmod 18 = 16$

$a \leftarrow 18$

$b \neq 0$? oui

$\text{temp} \leftarrow 16$

$b \leftarrow 18 \bmod 16 = 2$

$a \leftarrow 16$

$b \neq 0$? oui

$\text{temp} \leftarrow 2$

$b \leftarrow 16 \bmod 2 = 0$

$a \leftarrow 2$

$b \neq 0$? non

↓
Sortir 2