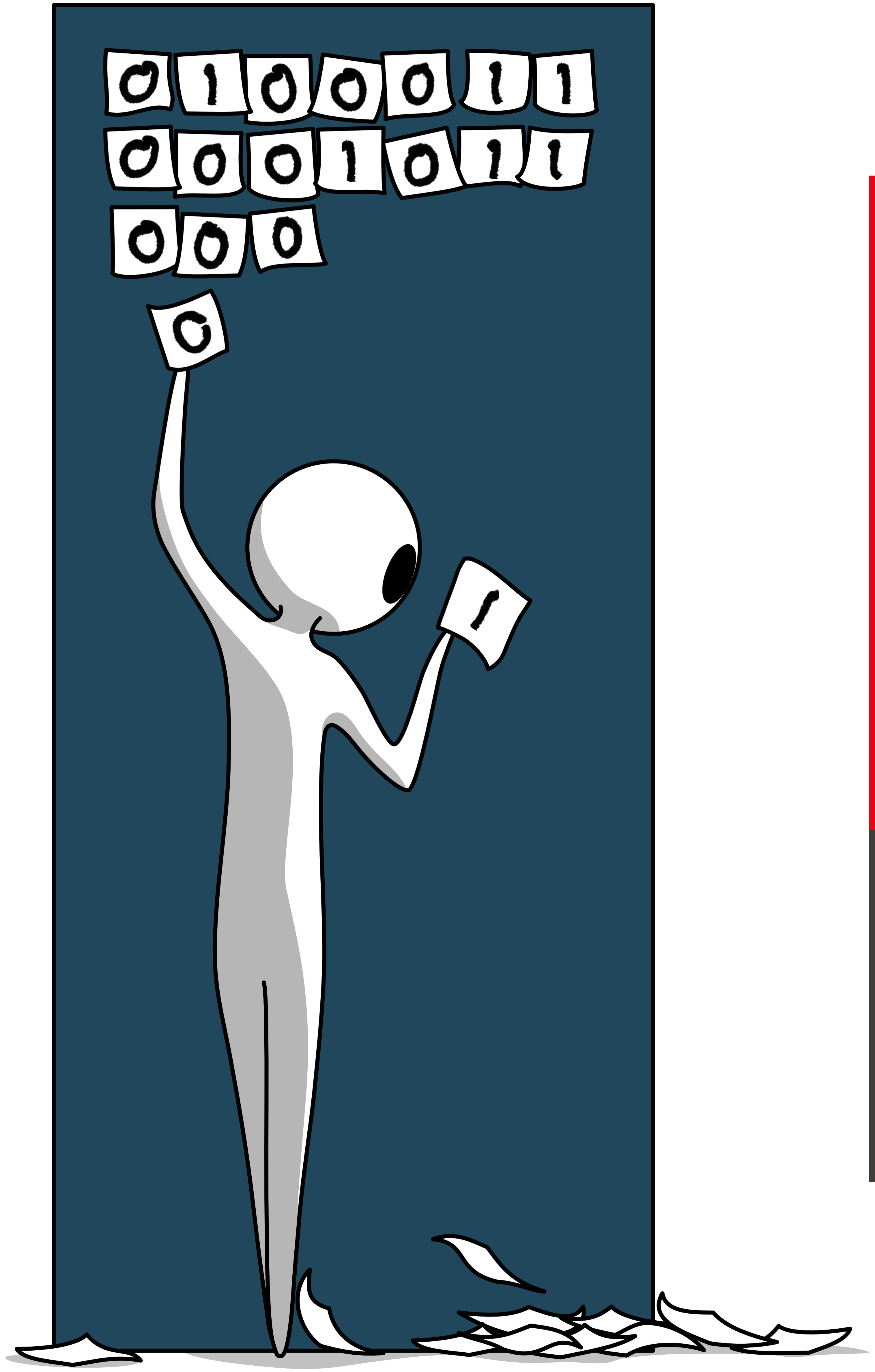
EPFL



Information, Calcul et Communication

Introduction

Olivier Lévêque, Dan-Cristian Tomozei
et Jean-Philippe Pellet

École
polytechnique
fédérale
de Lausanne

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…

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
- Echantillonnage
  et reconstruction de signaux
- Entropie
  et compression de données
- Correction d’erreurs
- Réseaux
- Cryptographie et sécurité

Horaires (partie théorique)

- **Cours :**

  les vendredis après-midis de 14h15 à 16h
  en salles SG 1, ~~SG 0123~~ (retransmission) et sur Zoom
  CM 1 4

- **Exercices:**

  ICC-C

  les vendredis après-midis de 16h15 à **17h15+** en salles DIA 004 / DIA 005 / INF 1 / INF 119 / INJ 218 / INM 201 / INM 202 / INM 203 (!)

  ICC-P

  Une vingtaine de personnes sont là pour vous : ***profitez-en !***

Evaluation

- Pour la partie théorique:

  un examen final seulement, valant pour 50% de la finale

- Pour la partie programmation:

  (un midterm,) un devoir noté et un examen final,
  valant pour 50% de la note finale

Points de contact

- **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 du cours
- **Zoom :** cours retransmis en direct, enregistrements du cours
- **Forum EdDiscussion :** vous pouvez poser des questions à tout moment, de manière anonyme si vous le désirez; encore une fois, ***profitez-en !***

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

$$
\begin{array}{r|l}
76'321 & 3 \\ \hline
-6 & 2'544 \\
16 & \\
-15 & \\
13 & \\
-12 & \\
12 & \\
-12 & \\
01 &
\end{array}
$$

$$76'321 \pmod 3 = 1$$

$$7 + \cancel{6} + \cancel{3} + 2 + \cancel{1} = 19$$

→ Reste 1

$$1+9=10$$

$$1+0=1$$

$$
\begin{array}{r|l}
47 & 3 \\ \hline
 & 15
\end{array}
$$

Reste: 2

$$4+7=11$$

$$1+1=2$$

$$47 = 4\cdot 10 + 7 = \cancel{4\cdot 9} + 4\cdot 1 + 7 = 4+7$$

Exemple 2: recherche du minimum dans une liste

$$\begin{cases} L = (13, 47, 18, 15, 11, 19, 46, 18, 15) \\ n = 9 \end{cases}$$

- on considère le premier nombre de la liste "début"
- on le compare au suivant "suivant" :

$$\begin{cases} \text{si suivant} < \text{début, alors début} \leftarrow \text{suivant} \\ \text{sinon on continue} \end{cases}$$

$\hookrightarrow$ on passe à celui d'après entrée : liste L, de n nombres

Sortie : le minimum de la liste

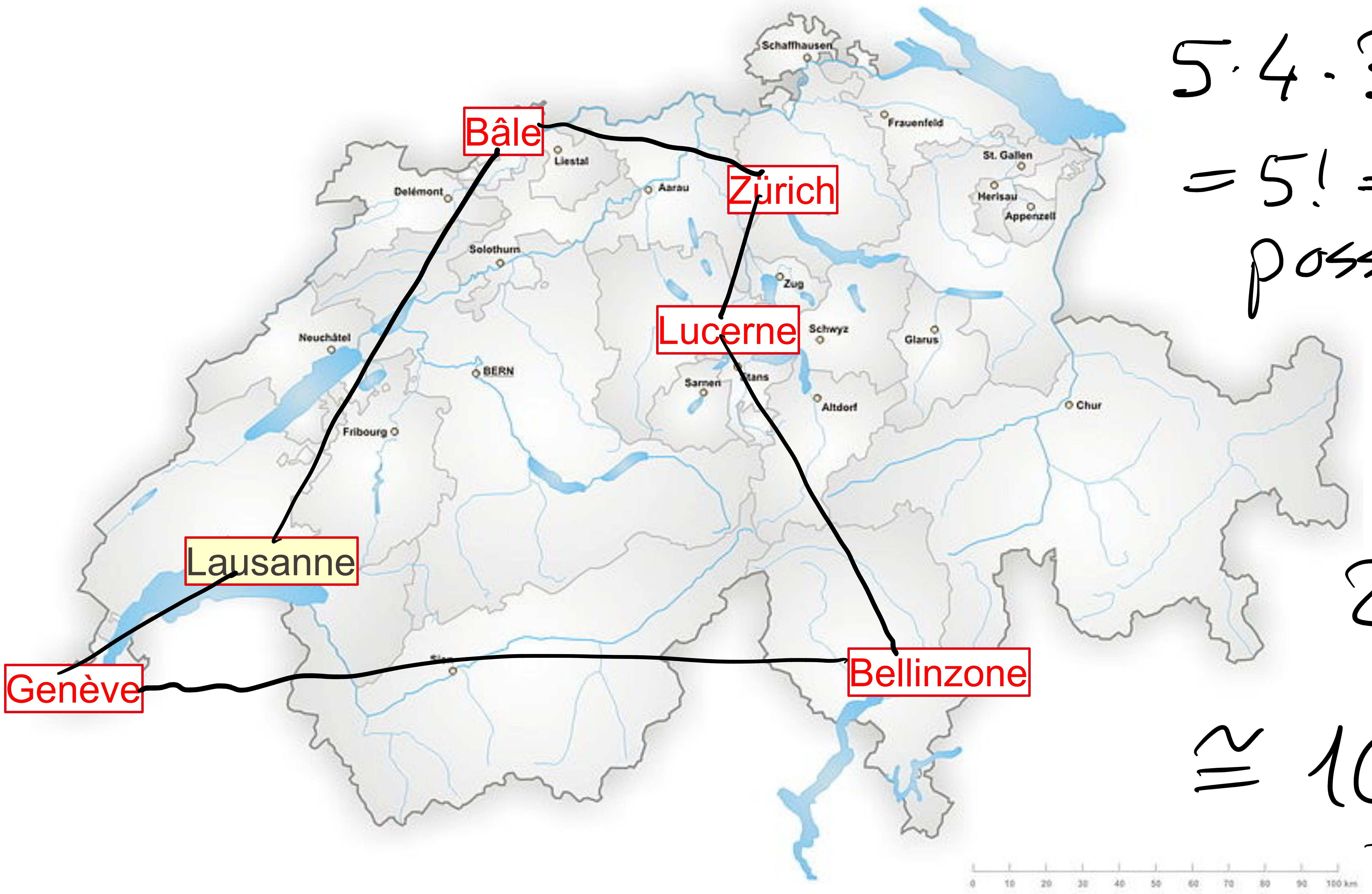
$x \leftarrow L(1)$

Pour i allant de 2 à n

    si $L(i) < x$, alors $x \leftarrow L(i)$

Sortir x

Exemple 3: problème du voyageur de commerce



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

$= 5! = 120$

possibilités

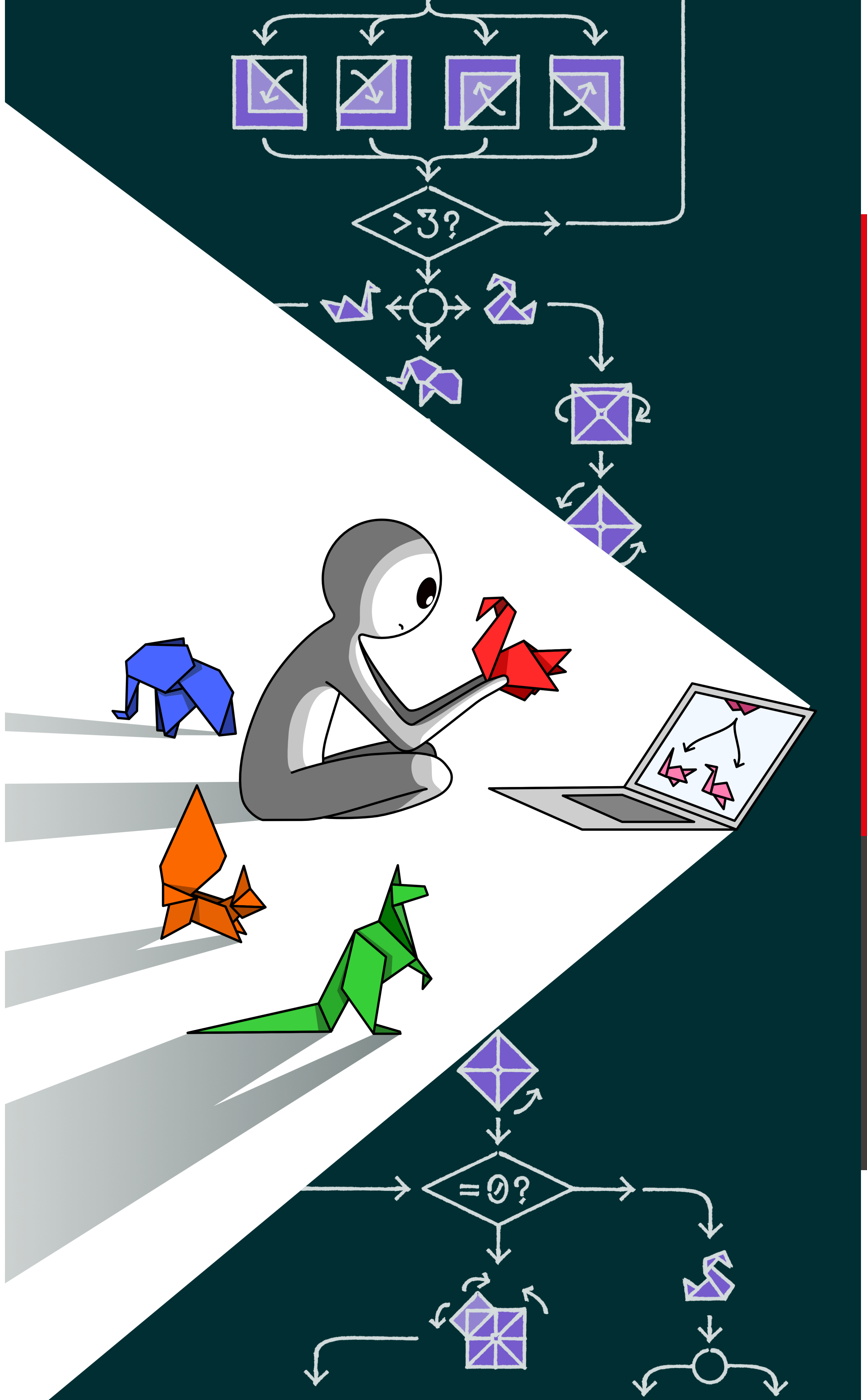
26 villes

$26!$

$\simeq 10^{26}$ possibilités

Information, Calcul et Communication

EPFL



Information, Calcul et Communication

Algorithmes : ingrédients de base

Olivier Lévêque

École
polytechnique
fédérale
de Lausanne

Algorithmes : ingrédients de base

Données

- Entrées
- Sorties
- Variables internes

Instructions

- Affectations
- Structures de contrôle
  - Branchements conditionnels (tests)
  - Itérations (boucles)
  - Boucles conditionnelles

entrée : $a \neq 0, b, c$ (nbs réels)

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 $\left\{-\frac{b}{2a}\right\}$

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

Illustration

- **Question :**

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

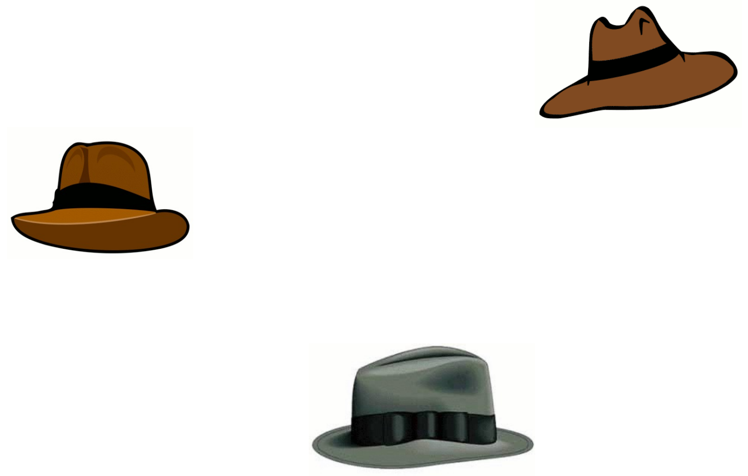
- **Question réciproque :**

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

Pour i allant de 1 à 3
  Pour k allant de 1 à 3
    Si $L(i)=L(k)$ et $i \neq k$, sortir non
Sortir oui

~~Pour i allant de 1 à 2~~
~~Pour k allant de 2 à 3~~
~~Si $L(i)=L(k)$, sortir non~~
~~Sortir oui~~

~~Pour i allant de 1 à 2~~

~~Si $L(i) = L(3)$, sortir non~~

~~Sortir oui~~

~~Pour i allant de 1 à 3~~

~~Pour k allant de 1 à 2~~

~~Si $L(i) = L(i+k)$~~

Pour i allant de 1 à 2

Pour k allant de $i+1$ à 3

Si $L(i) = L(k)$, sortir non

Sortir oui

Si $L(1) = L(2)$, sortir non ✓ $i=1$, $k=2$

si $L(1) = L(3)$, sortir non $i=1$, $k=3$

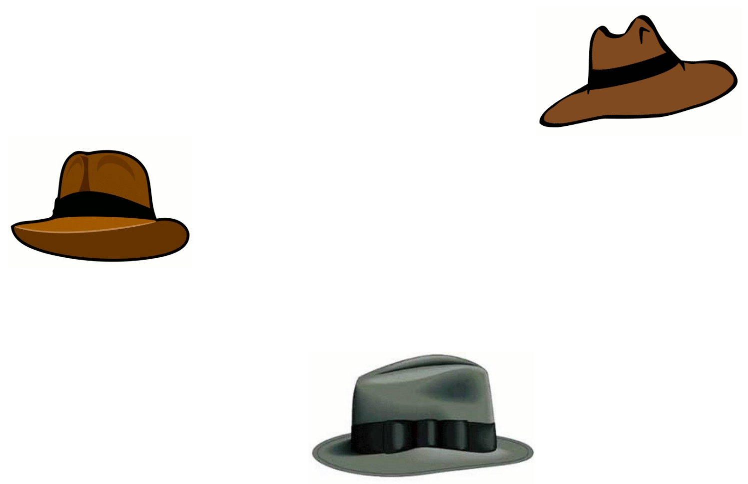
si $L(2) = L(3)$, sortir non $i=2$, $k=3$

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

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

EPFL

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

$x \leftarrow L(1)$

Pour i allant de 2 à n ?

Si $x = L(i)$, sortir non

Pour i allant de 1 à $n-1$

Pour k allant de $i+1$ à n ✓

Si $L(i) = L(k)$, sortir non

Sortir oui

$i=1$: $k=2 \ldots n$ $n-1$ comp.

$i=2$: $k=3 \ldots n$ $n-2$ comp.

$i=3$: $k=4 \ldots n$ $n-3$ comp.

$\vdots$ $\vdots$

$i=n-2$: $k=n-1, n$ 2 comp.

$i=n-1$: $k=n$ 1 comp.

$$1+2+3+\ldots+(n-2)+(n-1)=\frac{n(n-1)}{2}$$

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

$s \leftarrow$ oui
Pour i allant de 1 à $n-1$:
    Pour j allant de $i+1$ à n :
        Si $L(i) = L(j)$, alors : $s \leftarrow$ non
Sortir : s

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 : $pgdc(a, b)$

Tant que $b \neq 0$:
- $temp \leftarrow b$
- $b \leftarrow a \bmod b$
- $a \leftarrow temp$

Sortir : a

$a = 30$, $b = 12$

- $12 \neq 0$? oui
  - $temp \leftarrow 12$
  - $b \leftarrow 6$
  - $a \leftarrow 12$
- $6 \neq 0$? oui
  - $temp \leftarrow 6$
  - $b \leftarrow 0$
  - $a \leftarrow 6$
- $0 \neq 0$? non
  - sortir 6

$a = 30$, $b = 12$

$30 = 2 \cdot 3 \cdot 5$ | $12 = 2 \cdot 2 \cdot 3$

$\rightarrow$ pgcd $= 6$

$$\text{pgcd}(a,b) = \text{pgcd}(a-b, b)$$
$$= \text{pgcd}(a - k \cdot b, b)$$
$$= \text{pgcd}(a \bmod b, b)$$