

Informatique et Calcul Scientifique

Cours 7.1 : Introduction à l'algorithmique

13.11.2024

Qu'est-ce qu'un algorithme ?

- ▶ Un algorithme est une **procédure** pour résoudre un problème.
 - ▶ Il prend en **entrée (input)** une **instance** de ce problème
 - ▶ et produit la **sortie (output)** correspondant à cette instance.



Qu'est-ce qu'un algorithme ?

- ▶ Un algorithme est une **procédure** pour résoudre un problème.
 - ▶ Il prend en **entrée (input)** une **instance** de ce problème
 - ▶ et produit la **sortie (output)** correspondant à cette instance.

Grand bol de strychnine
Morphine
Un bon verre de pétrole
Gouttes de cigüe
Bave de sangsue
Scorpion coupé très fin
Poivre en grains
...



recette



. Pour la recette complète, voir [le pouding à l'arsenic](#).

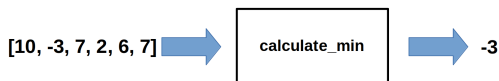
Qu'est-ce qu'un algorithme ?

Par exemple, on peut écrire un algorithme qui calcule le minimum d'une liste de nombres.

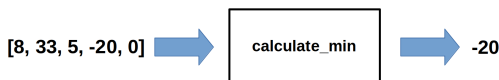
- ▶ **Entrée** : une liste de nombre réels
- ▶ **Sortie** : le minimum des nombres de la liste
- ▶ **Algorithme** (en français) :
 - ▶ Mettre **le_min** = le premier élément de la liste
 - ▶ Passer sur chaque élément de la liste. S'il est plus petit que **le_min**, mettre **le_min** = cet élément
 - ▶ Retourner **le_min**

Qu'est-ce qu'un algorithme ?

- ▶ Une instance de ce problème : la liste $[10, -3, 7, 2, 6, 7]$



- ▶ Une autre instance : la liste $[8, 33, 5, -20, 0]$



- ▶ Ne sont pas des instances valides de ce problème :
 - ▶ $['a', 'b', 'c']$
 - ▶ $[[0, 1], [1, 2]]$

- ▶ A un problème peuvent correspondre plusieurs algorithmes qui le résolvent
 - ▶ Comment caractériser "le meilleur" ?
- ▶ Pour certains problèmes on ne sait pas s'il existe un algorithme qui les résoud¹ ; pour certains problèmes on sait qu'il n'en existe aucun².
- ▶ Pour certains problèmes, on ne connaît encore aucun algorithme efficace qui les résoud.
 - ▶ Que veut dire "efficace" ?

1. https://en.wikipedia.org/wiki/Collatz_conjecture

2. https://en.wikipedia.org/wiki/Halting_problem

Algorithme vs. implémentation

- ▶ Un **algorithme** est une procédure générale pour résoudre un problème.
- ▶ Un **programme** est une implémentation spécifique d'un algorithme dans un langage de programmation donné, par exemple Python, et sur un système donné.
- ▶ A un même algorithme correspondent donc plusieurs implémentations³.
- ▶ Les algorithmes vus dans ce cours seront donnés sous forme de programmes Python.

3. Voir <http://www.rosettacode.org> pour des implémentations de divers algorithmes en plus de 800 langages de programmation.

- ▶ Formalisation de la notion d'algorithme, correctitude d'un algorithme
- ▶ Algorithmes récurrents
- ▶ Complexité algorithmique (analyse mathématique du temps de parcours des algorithmes)
- ▶ Algorithmes de recherche
- ▶ Algorithmes de tri
- ▶ Algorithmes de graphes

- ▶ T. Cormen, C. Leiserson, R. Rivest, C. Stein, *“Introduction to Algorithms (3rd edition)”*, MIT Press and McGraw-Hill, 2009.
- ▶ T. Cormen, *“Algorithms Unlocked”*, MIT Press, 2013.
- ▶ J. Guttag, *“Introduction to Computation and Programming Using Python : With Application to Understanding Data”*, Second Edition, MIT Press, 2016.

- ▶ Correctitude
 - ▶ Slides optionnelles sur Moodle
- ▶ Efficacité