



# ALGORITHMIE

Par K vin Ansard

# Qui suis-je ?



DÉVELOPPEUR SPÉCIALISÉ  
EN CLOUD APP DATA



PROFESSEUR FREELANCE  
EN INFORMATIQUE



EXPERT EN R&D

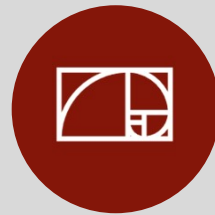
# Qu'est ce qu'un algorithme ?



INSTRUCTIONS



CONDITIONS



RESULTAT

Un algorithme, est une suite d'opérations élémentaires aussi appelé instructions qui une fois exécutés correctement conduits à un résultat voulu

# A quoi ça sert ?

On retrouve des algorithmes partout

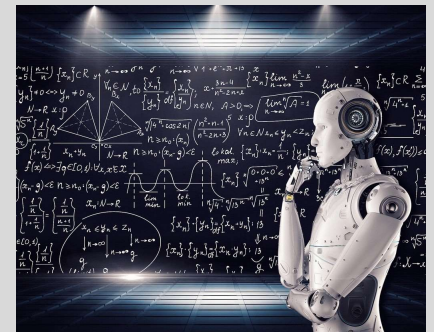
**Assurance**

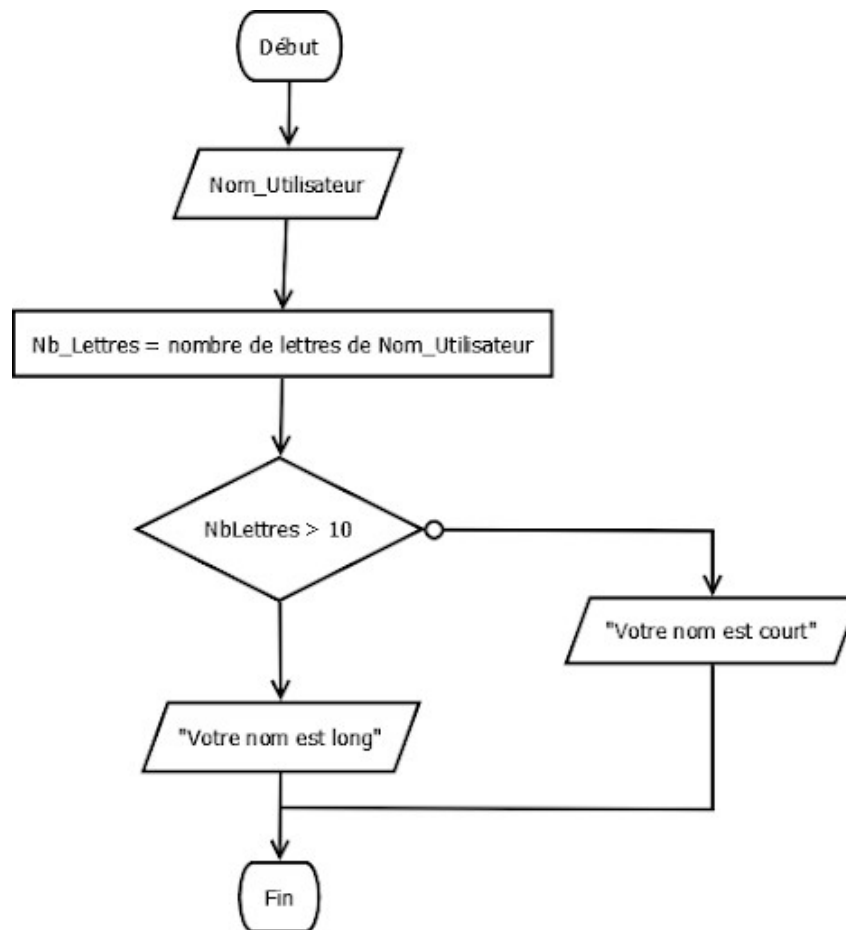


**Bitcoin**



**intelligence artificielle**





18/01/2021

# Comment concevoir un algorithme ?

## 1- L' organigramme

Un organigramme est une représentation schématique des liens et des relations fonctionnelles

≡ Peuso-Code.txt ✕

CHAP01 > terraform-docker > ≡ Peuso-Code.txt

```
1  Variable
2  |    nb_lettre : Entier
3
4  Début:
5  |    Recuperer le nom d'utilisateur (input)
6  |    Lire(le nom)
7  |    nb_lettre = nombre de lettre dans le nom
8  |    SI nb_lettre > 10
9  |    |    Afficher(votre nom est long)
10 |    SINON
11 |    |    Afficher(Votre nom est court)
12 Fin
```

# Comment concevoir un algorithme ?

## 2- Le pseudo code

En programmation, le pseudo-code, également appelé LDA est une façon de décrire un algorithme en langage presque naturel, sans référence à un langage de programmation en particulier

# Live Coding Python les bases

Les points abordés :

- Les Variables
- Les Boucles
- Les Conditions
- Les Fonctions
- Mise en place d'un algorithme simple

# Passons à la pratique !

---

Les prérequis

Python

---

Un IDE (VS code / Pycharm)

---

Groupe de 2 imposés

---



# Qu'est ce qu'un algorithme de tri ?

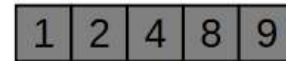
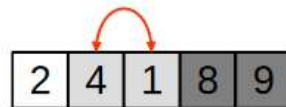
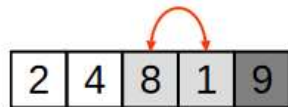
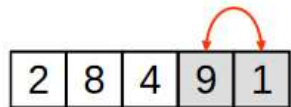
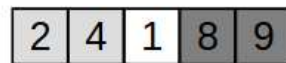
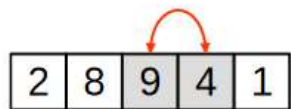
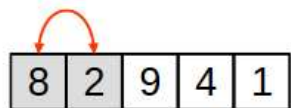
Les algorithmes de tri les plus connus :

- ❑ tri à Bulle (*Bubble sort*)
- ❑ tri par Sélection (*Selection sort*)
- ❑ tri par Insertion (*Insertion sort*)
- ❑ tri Fusion (*Merge sort*)
- ❑ tri Rapide (*Quick sort*)

Exercices

# A VOUS DE RECODER CES ALGORITHMES

Un sujet vous sera donné en PDF vous expliquant l'algorithme de chaque exercice.



(a)  $i = 1$

(b)  $i = 2$

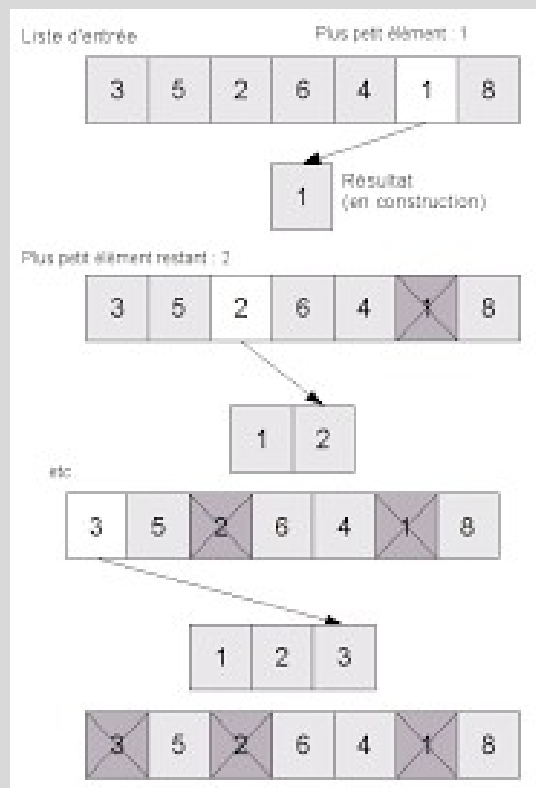
(c)  $i = 3$

(d)  $i = 4$

## Exercice 1- Le tri à bulle

Le tri à bulles ou tri par propagation est un algorithme de tri. Il consiste à comparer répétitivement les éléments consécutifs d'un tableau, et à les permuter lorsqu'ils sont mal triés.

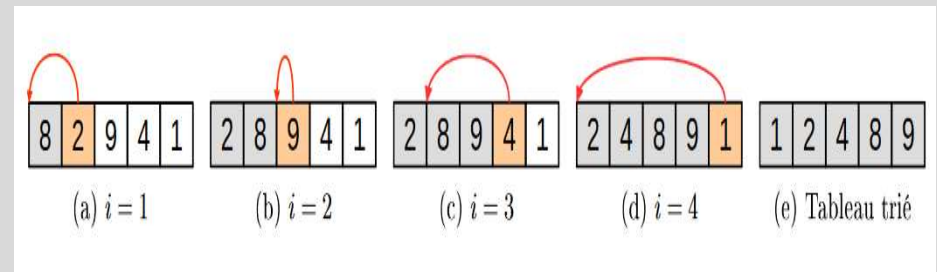
## Exercice 2- Le tri par sélection

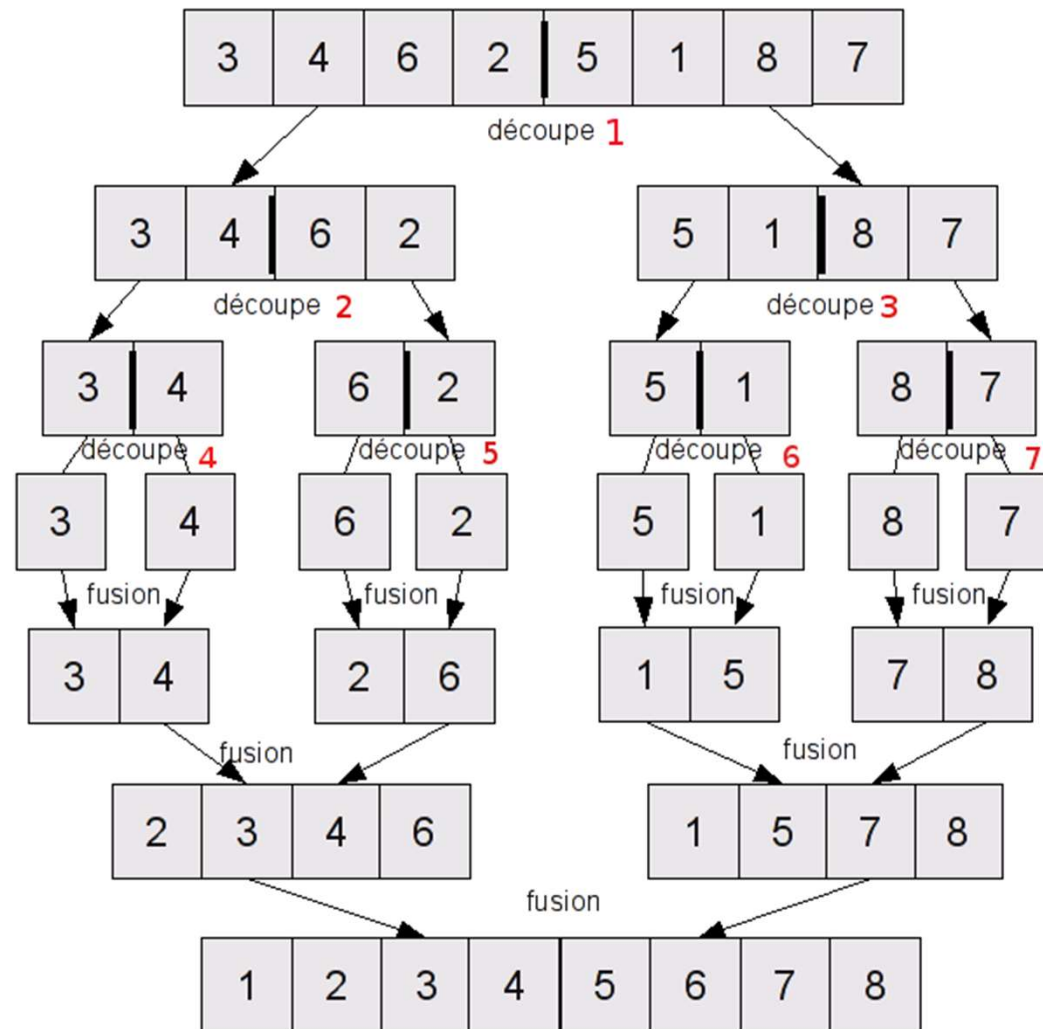


- Le tri par sélection est un algorithme de tri par comparaison. Cet algorithme est simple, mais considéré comme inefficace car il s'exécute en temps quadratique en le nombre d'éléments à trier, et non en temps pseudo linéaire

## Exercice 3- Le tri par insertion

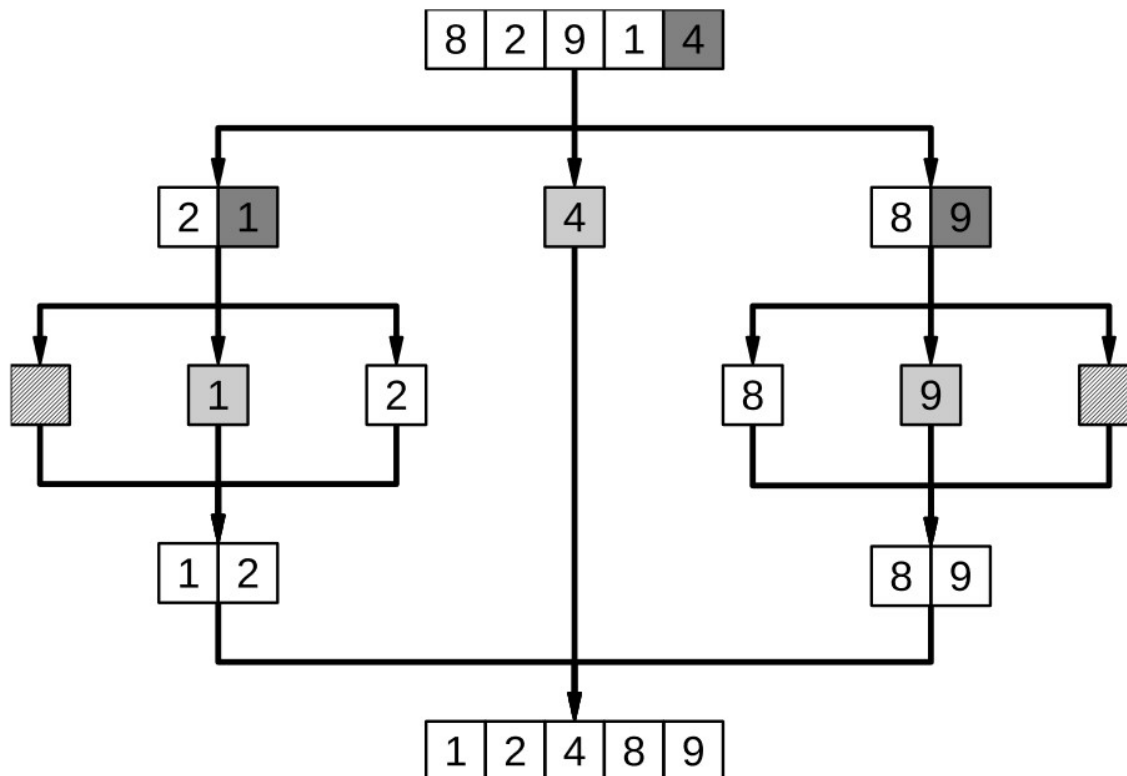
- Le tri par sélection est un algorithme de tri par comparaison. Cet algorithme est simple, mais considéré comme inefficace car il s'exécute en temps quadratique en le nombre d'éléments à trier, et non en temps pseudo linéaire





## Exercice 4- tri fusion

En informatique, le tri fusion est un algorithme de tri par comparaison stable. Sa complexité temporelle pour une entrée de taille  $n$  est de l'ordre de  $n \log n$ , ce qui est asymptotiquement optimal. Ce tri est basé sur la technique algorithmique diviser pour régner.



## Exercice 5- tri rapide

La méthode consiste à placer un élément du tableau (appelé pivot) à sa place définitive, en permutant tous les éléments de telle sorte que tous ceux qui sont inférieurs au pivot soient à sa gauche et que tous ceux qui sont supérieurs au pivot soient à sa droite.

Cette opération s'appelle le partitionnement. Pour chacun des sous-tableaux, on définit un nouveau pivot et on répète l'opération de partitionnement. Ce processus est répété récursivement, jusqu'à ce que l'ensemble des éléments soit trié.

# Exercice 6 – Comparer la Vitesse d'exécution

- Dans cet exercice vous devrez mettre en place un algorithme permettant de calculer la complexité relative de chacun des algorithmes vue précédemment.