



# Meetup

# Etude des algorithmes

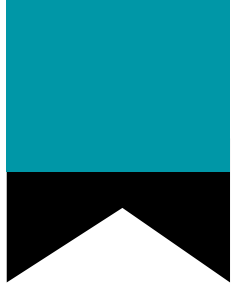
# fondamentaux

Session du lundi 7 novembre 2016

# Notre but

Apprendre ensemble des algorithmes  
fondamentaux

En s'aidant les uns les autres



# Au programme, ce soir

- Qu'est-ce qu'un graphe?
- Des algorithmes de recherche de parcours
  - Algorithmes naifs
  - Bellman-Ford
  - Dijkstra
  - A\*
- Un exemple avec Neo4j et métro parisien

Pour la présentation : <https://goo.gl/aydtl0>

[illegible]

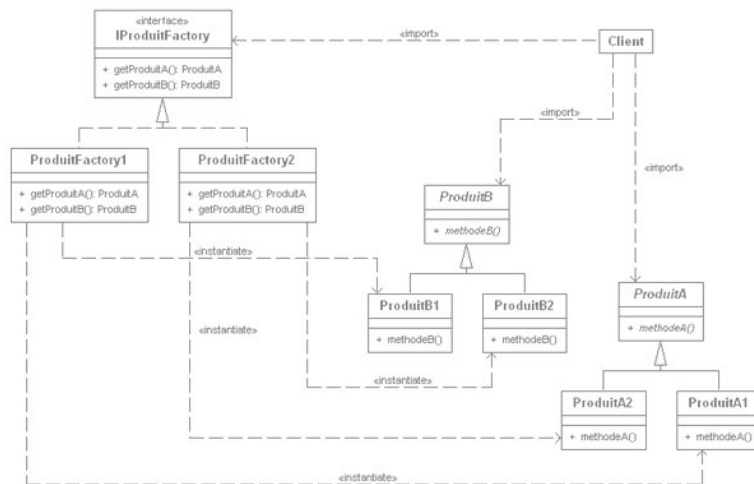
**Welcome chez Arolla!**

**TDD, BDD, DDD, Clean Code, Software Craftmanship, ...**

**SOFTWARE CRAFTSMANSHIP**

# On fait d'autres trucs!

## Meetup Étude des Design Patterns

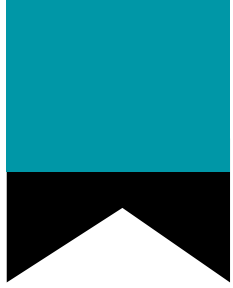


<http://www.meetup.com/fr-FR/design-patterns/>



# Qu'est-ce qu'un graphe?

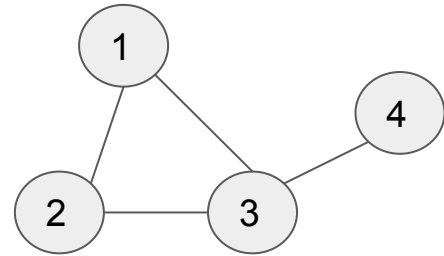
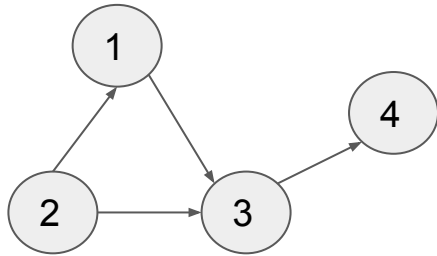
Un graphe est un ensemble de noeuds ou sommets (en: node, vertex/vertices, points) liés par des arcs (en: edges, arcs, or lines).



# Qu'est-ce qu'un graphe?

orienté (en: directed) et non-orienté (en: undirected)

Les arcs peuvent être valués ou non



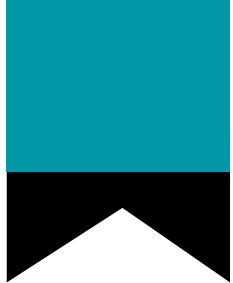


# Qu'est-ce qu'un graphe?

Implémentons notre graphe!

Des idées?





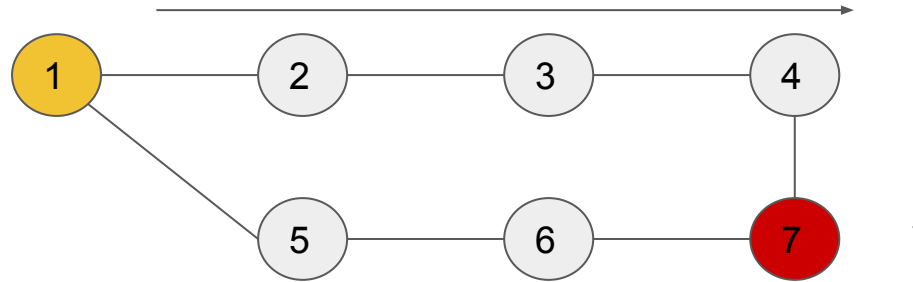
# **Le problème de ce soir**

Comment trouver le chemin optimal en un minimum de temps pour aller d'un noeud à l'autre?

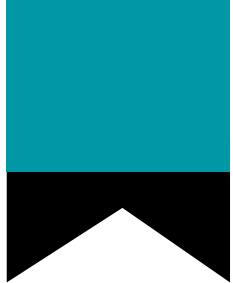


# Algorithme naïf

On cherche à avancer le plus possible et si on est coincé, on revient en arrière

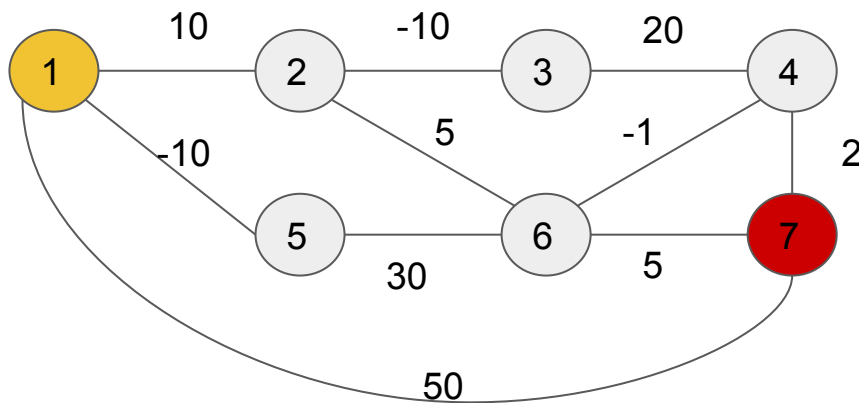


Deux manières de faire...



# Algorithme Bellman-Ford

Plus court chemin d'un noeud vers tous les autres

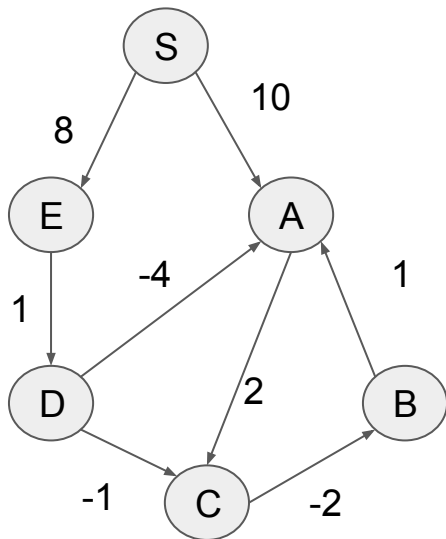


👍 Arc négatif

👎 Cycle négatif

# Algorithme Bellman-Ford

Départ



$n := \text{nb Noeud}$ ,  $E$  : ensemble des arcs

Init:  $\text{dist}[n] = \infty$

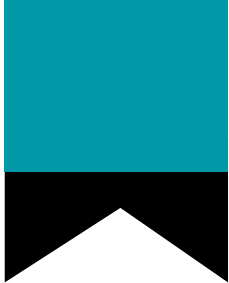
Répéter  $n - 1$  fois

Pour chaque arc  $(u,v) \in E$

if  $\text{dist}(v) > \text{dist}(u) + l(u,v)$

$\text{dist}(v) := \text{dist}(u) + l(u,v)$

⇒ Complexité du pire cas?



# Algorithme Bellman-Ford

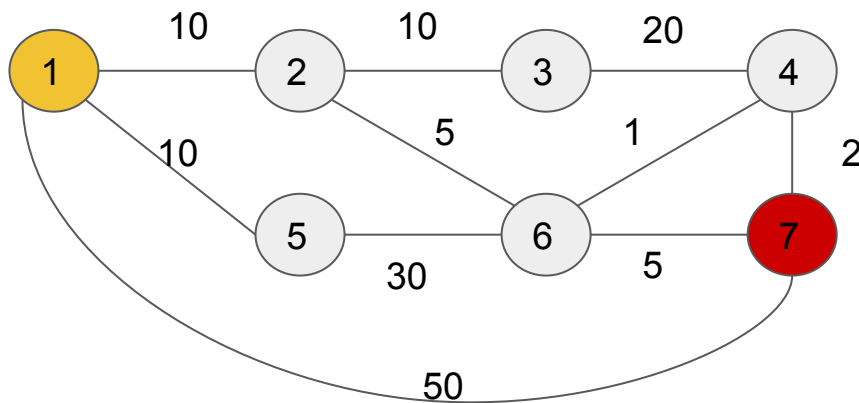
Itération 1

Noeud	S	A	B	C	D	E	Commentaires
S	0	10	$\infty$	$\infty$	$\infty$	8	
A	0	10	$\infty$	12	$\infty$	8	$\text{dist}(C) = \infty, \text{dist}(A) + l(A, C) = 12$
B	0	10	$\infty$	12	$\infty$	8	$\text{dist}(B) = \infty$
C	0	10	10	12	$\infty$	8	$\text{dist}(B) = \infty, \text{dist}(C) + l(C, B) = -2$
D	0	10	10	12	$\infty$	8	
E	0	10	10	12	9	8	$\text{dist}(D) = \infty, \text{dist}(E) + l(E, D) = 1$



# Algorithme Dijkstra

Plus court chemin d'un noeud vers tous les autres



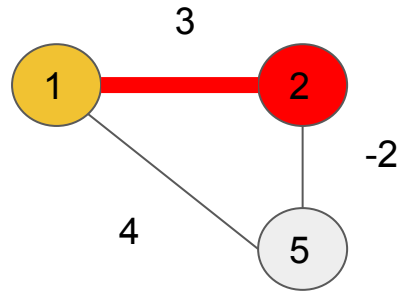
👎 Arc négatif

👎 Cycle négatif

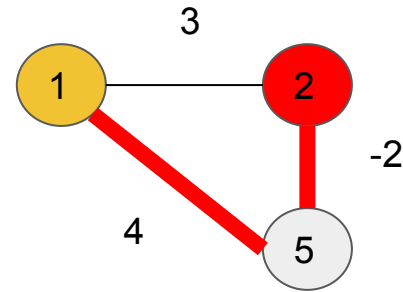


# Algorithmes Dijkstra vs BF

Dijkstra : greedy/glouton



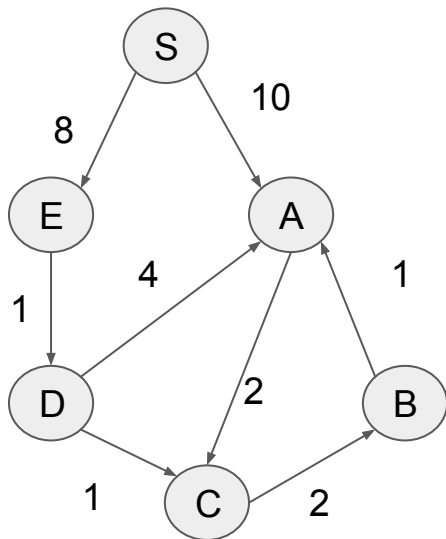
Dijkstra



Bellman-Ford

# Algorithme Dijkstra

Départ



$n := \text{nb Noeud}, N : \{ \text{noeuds} \}$

Init:  $\text{dist}[n] = \infty, \text{précuseur}[n] = \{\}$

Répéter tant que  $N \neq \{\}$

$u \in N / \text{dist}(u) = \min \text{dist}$

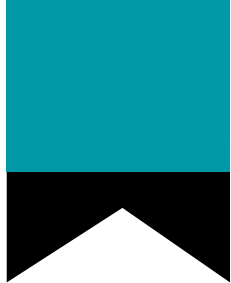
$N := N - \{u\}$

Pour chaque arc  $(u,v) \in E$

if  $\text{dist}(v) > \text{dist}(u) + l(u,v)$

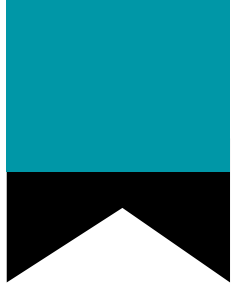
$\text{dist}(v) := \text{dist}(u) + l(u,v)$



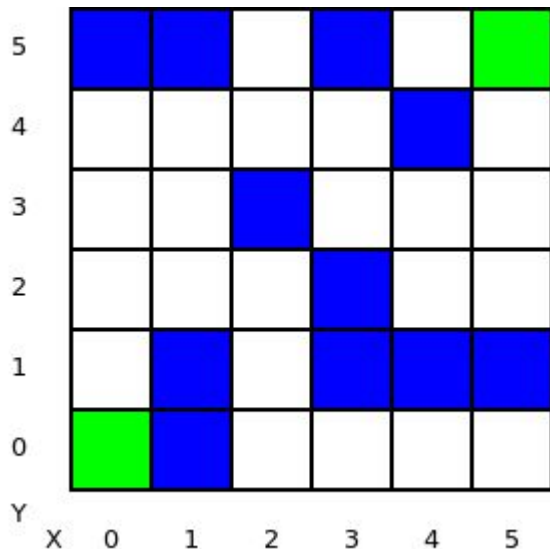


# Algorithme A\*

Approximation du chemin le plus court



# Algorithme A\*



$n := \text{nb Noeud}$ ,  $N : \{ \text{noeuds} \}$

Init:  $\text{dist}[n] = \infty$ ,  $\text{estimated}[n] = ?$ ,  
 $\text{precuseur}[n] = \{ \}$

Répéter tant que  $N \neq \{ \}$

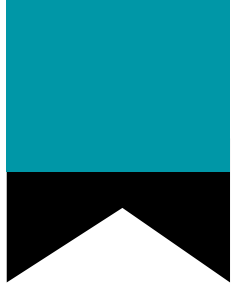
$u \in N / \text{dist}(u) = \min (\text{dist} + \text{estimated})$

$N := N - \{u\}$

Pour chaque arc  $(u,v) \in E$

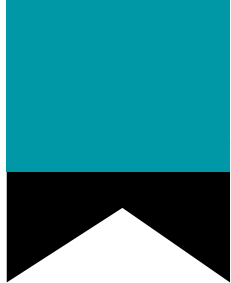
if  $\text{dist}(v) > \text{dist}(u) + l(u,v)$

$\text{dist}(v) := \text{dist}(u) + l(u,v)$



# Un exemple

Neo4j + métro parisien



# Ressources

Cours “introduction to algorithms” de Sedgewick

