

Tri par insertion

Le sommaire

- L'objectif de l'algorithme
- Comment l'algorithme fonctionne
- Quelques définitions
- Quelques exemples où cet algorithme peut être utile

L'objectif de l'algorithme

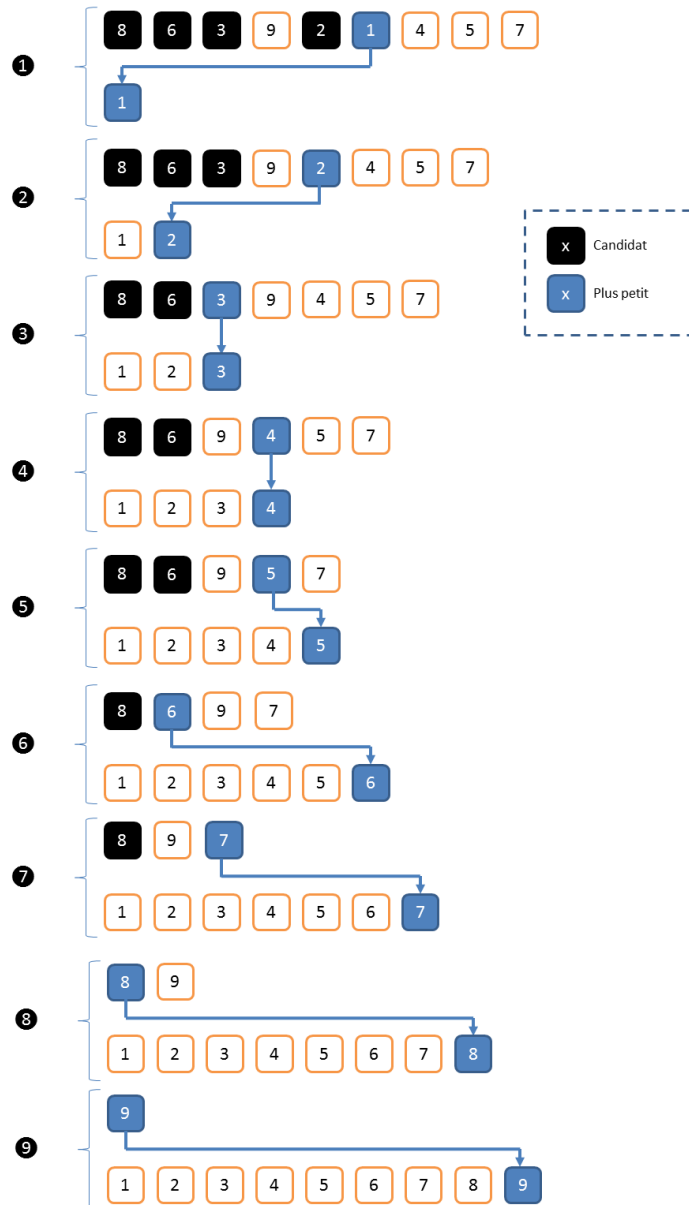
L'objectif de l'algorithme de tri par insertion(et par sélection) est, comme son nom l'indique, de trier une liste.

Comment l'algorithme fonctionne

L'algorithme ou du moins les algorithmes de tri par insertion et par sélection ont le même but mais fonctionnent différemment. En effet, le tri par sélection passe en revue tous les éléments d'une liste, en extrait le plus petit et le sort et ça jusqu'au dernier élément de la liste. Pour l'algorithme de tri par insertion, on a une liste non triée, on extrait le premier élément de la liste et on le place dans une liste vide puis on réitère l'opération avec le deuxième élément et si il est plus petit que le premier élément alors il sera placé avant lui sinon après on fait cela avec le reste de la liste.

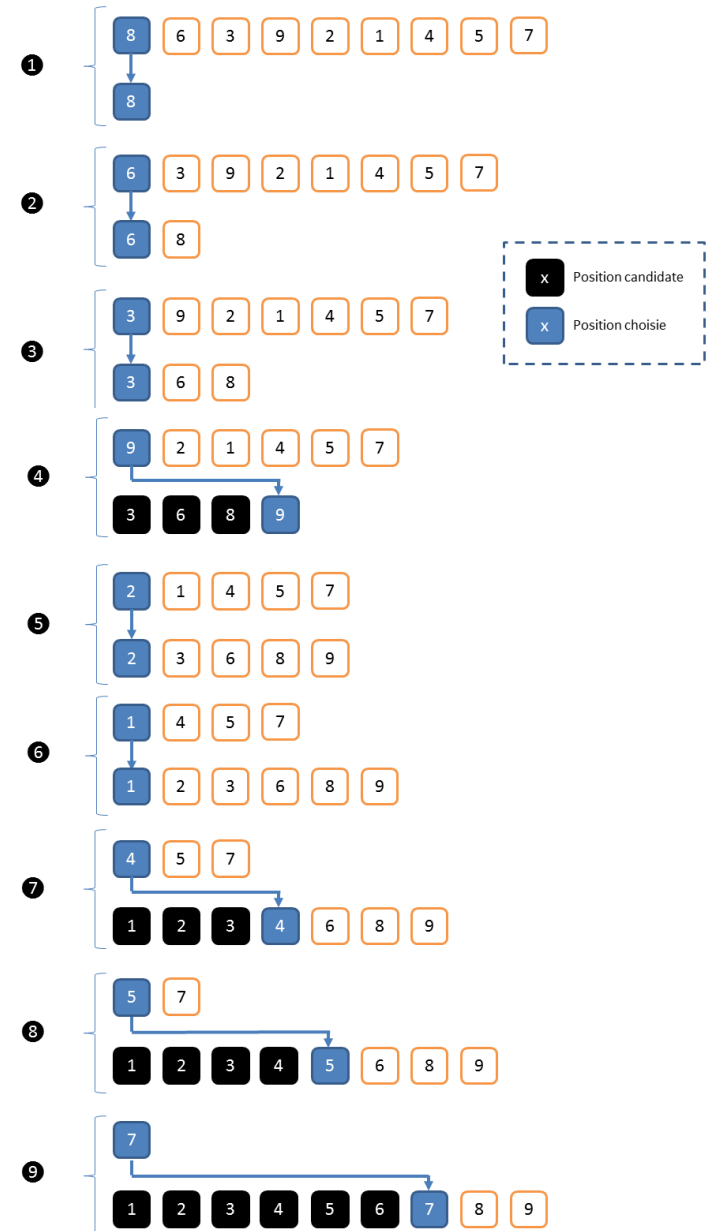
Tri par sélection

Liste initiale: 8 6 3 9 2 1 4 5 7



Tri par insertion

Liste initiale: 8 6 3 9 2 1 4 5 7



Quelque définitions

- Invariant de boucle :

Un invariant de boucle nous permet de « vérifier » notre algorithme. Il a une valeur qui demeure la même l'après chaque itération de la boucle.

- Pourquoi l'algorithme s'arrête ?

Si l'algorithme s'arrête ce que la condition de la boucle (en python while) est remplie .

Quelque exemples où cet algorithme peut être utile

- L'algorithme de tri par insertion et par sélection sont utiles pour trier une liste dans l'ordre croissant. En réalité nous avons à faire à cet algorithme dans la vie de tous les jours.

En effet, par exemple sur vos relevés bancaires les opérations sont listées par dates.