

# Fiche d'investigation de fonctionnalité

<b>Fonctionnalité :</b> Recherche Principale	<b>Fonctionnalité #1</b>
<b>Problématique :</b> Obtenir l'algorithme de recherche le plus rapide	

## Option 1 : La méthode natives

Dans cette option, nous utilisons, pour l'algorithme de notre recherche, les méthodes natives (boucles for, while ...). Cette méthode nous permet de voir comment se comporte la fonction et la façon dont les données sont découpées et traitées. L'inconvénient de cette méthode réside dans le fait qu'elle est plus fastidieuse à écrire.

### Avantages

- Compréhension du découpage des données
- Meilleure prise en main pour les autres développeurs

### Inconvénients

- Conception fastidieuse

Réalisation de 3 boucles + une fonction de comparaison des données

## Option 2 : La méthode array

Dans cette option, nous utilisons, pour l'algorithme de notre recherche, les méthodes array (foreach, reduce ...). L'avantage de cette méthode est qu'elle est simple à comprendre et écrire. En revanche, c'est javascript qui s'occupe du traitement des données, il pourrait être difficile de savoir comment elles sont traitées jusqu'au résultat.

### Avantages

- Conception simple

### Inconvénients

- Si les concepts Javascript ne sont pas bien assimilés la méthode peut être mal utilisée.

Utilisation d'une seule boucle et d'un opérateur de comparaison

## Solution retenue :

La solution retenue est celle de la méthode natives. En effectuant un test JSBENCH, c'est cette méthode qui est apparue la plus rapide (cf ci dessous)