



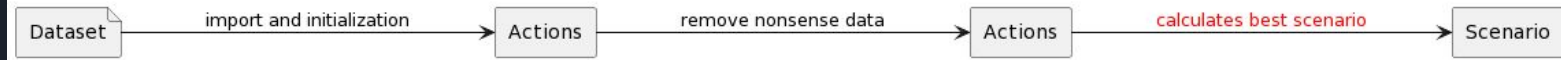
# AlgoInvest&Trade

*Python scripts*

# Bruteforce Python Script

- Complexité :  $O(n^2 * 2^n)$
- Temps d'exécution (pour 20 termes) : inférieur à 3 minutes
- Extraire les données du fichier CSV fournit
- Initialiser les objets "Actions" avec les données chargées
- Calculer le bénéfice de chaque Action en fonction de son cout et de sa rentabilité
- Supprimer les Actions avec un prix négatif ou nul, pareil pour le bénéfice
- Calculer le meilleur "Scénario" en itérant dans des listes d'Actions de différentes longueurs
- Afficher le résultat et le temps d'exécution

Bruteforce script : UML





# Bruteforce Python Script : **résultats**

```
Total benefits : 99.08€
```

```
Total coast : 498.0€
```

```
Number of actions : 10
```

```
Actions name : ['Action-4', 'Action-5', 'Action-6', 'Action-8', 'Action-10',  
'Action-11', 'Action-13', 'Action-18', 'Action-19', 'Action-20']
```

```
-----  
Execution time: 69.68 seconds
```

# Optimized Python Script

- Complexité :  $O(14) + O(n \log n) \rightarrow O(n \log n)$
- Temps d'exécution (pour 1000 termes) : inférieur à 10 secondes
- Extraire les données du fichier CSV fournit
- Initialiser les objets "Actions" avec les données chargées
- Calculer le bénéfice de chaque Action en fonction de son cout et de sa rentabilité
- Supprimer les Actions avec un prix négatif ou nul, pareil pour le bénéfice
- Trier les Actions en fonction de leur bénéfice
- Initialiser les objets "Scenarios" contenant chacun une liste d'Actions, en itérant seulement dans **les premières Actions**
- Afficher les résultats et le temps d'exécution





# Optimized Python Script : **résultats**

Python script

```
SCENARIO 1
Total benefits : 197.09€
Total coast : 499.98€
Number of actions : 2
Actions name : ['Share-GRUT', 'Share-CBNY']
-----
Execution time: 5.70 seconds
```

Sienna

Sienna bought:

Share-GRUT

Total cost: 498.76â,-

Total return: 196.61â,-