

Mémoïsation

La programmation dynamique consiste à stocker les valeurs des sous-problèmes pour éviter les recalculs. Une stratégie qui peut prendre la forme de la mémoïsation.

En informatique le terme *mémoïsation* désigne la sauvegarde des valeurs de retour d'une fonction selon ses valeurs d'entrée. Dans le cadre de la programmation dynamique, on peut utiliser le mot "mémorisation" et "mémoïsation" de façon interchangeable, le second étant plus précis.

En python, on dispose des dictionnaires qui permettent de rendre la mémoïsation très efficace avec un accès $O(1)$ aux valeurs mémoïsées. On peut améliorer la fonction récursive `fib(n: int)`, en implémentant la programmation dynamique avec un dictionnaire de mémoïsation qui associe la clé n à F_n :

```
def fib(n: int, memo: dict):
    if n == 0 or n == 1:
        memo[n] = n
    if n not in memo:
        memo[n] = fib(n-1, memo) + fib(n-2, memo)
    return memo[n]
```

Chaque terme de la suite de Fibonacci n'est calculé qu'une seule fois (et lu deux fois). Cet algorithme a donc une complexité en $O(n)$, mais en contrepartie, on augmente l'utilisation de l'espace mémoire avec un dictionnaire de taille n .

Rappel sur les algorithmes gloutons

Un algorithme glouton est un algorithme qui suit le principe de faire, étape par étape, un choix optimal local, dans l'espoir d'obtenir un résultat optimal global. La solution globale n'est pas nécessairement optimale. En contrepartie, les algorithmes gloutons sont souvent simples à implémenter et disposent d'une bonne complexité en temps et en mémoire.

Par exemple, dans le problème du rendu de monnaie (donner une somme avec le moins possible de pièces), l'algorithme consistant à répéter le choix de la pièce de plus grande valeur qui ne dépasse pas la somme restante est un algorithme glouton. Exemple de déroulé de l'algorithme:

Système monétaire: {10, 5, 2, 1}

Somme à rendre: 8

| Étape | Choix de pièce | Reste à rendre |
|-------|----------------|----------------|
| 1 | 5 | 3 |
| 2 | 2 | 1 |
| 3 | 1 | 0 |