

PROGRAMMATION DYNAMIQUE

Dans ce chapitre, nous sommes revenus sur les algorithmes d'optimisation avec l'exemple du **rendu de monnaie**.

- L'**approche gloutonne** a été vue en première. Dans l'exemple du rendu de monnaie, elle consiste à prendre la pièce de plus grande valeur, en prendre autant qu'il est possible puis passer à la suivante etc. Il arrive que cette approche ne donne pas le résultat optimal.
- Pour être sûr d'obtenir le résultat optimal, on peut utiliser la **force brute**, c'est à dire calculer toutes les configurations possibles et chercher la plus optimale. Nous avons écrit l'algorithme de force brute récursivement. La complexité de cette approche est *exponentielle* et donc bien trop coûteuse en termes de temps de calcul.
- Pour contourner cette difficulté, nous avons remarqué que, en force brute, certains calculs sont faits plusieurs fois. L'approche dynamique se propose de garder en mémoire les calculs déjà effectués, cette technique se nomme *mémoïsation*. On peut montrer que la complexité de l'algorithme de **programmation dynamique** est en $O(n \times \log_2(n))$.