

# Algorithmes gloutons

## Capacités attendues

- ✓ Résoudre un problème grâce à un algorithme glouton.

Exemples : problèmes du sac à dos ou du rendu de monnaie.



## 1 Introduction : les problèmes d'optimisation

En informatique, on rencontre souvent des problèmes d'**optimisation**, c'est-à-dire des problèmes pour lesquels on cherche la meilleure solution possible satisfaisant un certain nombre de contraintes.

Exemple :

- la recherche du plus court chemin reliant deux villes en utilisant un réseau routier existant ;
- le nombre minimum de pièces de monnaie que l'on peut rendre à partir du contenu d'une caisse enregistreuse commerciale ;
- le nombre maximum de cours qu'on peut organiser dans une même journée dans un établissement ayant un nombre fixé de salles et connaissant la durée et les horaires de chaque cours ;
- etc.

En pratique, on cherche à trouver une solution algorithmique pour résoudre des problèmes d'optimisation quand :

1. le problème possède un très grand nombre de solutions ;
2. on sait évaluer la qualité de chacune des solutions (et donc dire quelle solution est meilleure parmi plusieurs).

## 2 Le problème du rendu de monnaie

On veut programmer une caisse automatique pour qu'elle rende la monnaie de façon optimale, c'est-à-dire avec le nombre minimal de pièces et de billets.

Les valeurs des pièces et billets à disposition sont : 1, 2, 5, 10, 20, 50, 100 et 200 euros. On suppose que la machine dispose d'autant d'exemplaire de pièces et billets que nécessaires pour rendre la monnaie.

### Exercice 1

Un objet coûte 34 euros et on paie avec un billet de 50. La machine doit alors rendre 16 euros.

De quoi est composé le rendu de 16 € par la machine si on veut rendre un minimum de pièces et de billets ?

## 2.1 La force brute pour déterminer toutes les solutions

Une solution naïve pour déterminer la solution optimale est d'énumérer toutes les solutions possibles, c'est-à-dire toutes les manières possibles d'obtenir la somme 16 à l'aide des nombres 1, 2, 5, 10, 20, 50 et 100.

Ainsi pour rendre 16 euros on peut, par exemple, utiliser (toutes les solutions ne sont pas listées ici) :

Rendu de monnaie	Nombre de pièces et billet	Rendu de monnaie	Nombre de pièces et billet
$16 \times 1\text{€}$	16	$2 \times 1\text{€}, 7 \times 2\text{€}$	9
$14 \times 1\text{€}, 1 \times 2\text{€}$	15	$6 \times 1\text{€}, 2 \times 5\text{€}$	8
$12 \times 1\text{€}, 2 \times 2\text{€}$	14	$6 \times 1\text{€}, 1 \times 10\text{€}$	7
$10 \times 1\text{€}, 3 \times 2\text{€}$	13	$2 \times 1\text{€}, 2 \times 2\text{€}, 2 \times 5\text{€}$	6
$8 \times 1\text{€}, 4 \times 2\text{€}$	12	$3 \times 2\text{€}, 2 \times 5\text{€}$	5
$6 \times 1\text{€}, 5 \times 2\text{€}$	11	$3 \times 2\text{€}, 1 \times 10\text{€}$	4
$4 \times 1\text{€}, 6 \times 2\text{€}$	10	$1 \times 1\text{€}, 1 \times 5\text{€}, 1 \times 10\text{€}$	3

## 3 Une méthode moins gourmande en calculs : un algorithme glouton

On fait à chaque étape le choix qui semble meilleur en espérant obtenir une solution optimale. Autrement dit, ici on rend le billet ou la pièce de la plus grande valeur possible. On recommence sur la somme à laquelle on a soustrait ce qu'on a déjà rendu.

**Exemple :** Pour rendre 16€ :

1. on choisit de rendre un billet de 10€ (plus grande valeur possible alors) ;
2. on doit encore rendre 6€ donc on choisit le billet de 5€ ,
3. il reste enfin 1€ à rendre avec une pièce de 1.

Au final, en trois étapes on obtient la solution :  $1 \times 1\text{€}$  ,  $1 \times 5\text{€}$  ,  $1 \times 10\text{€}$  soit la meilleure solution proposée ci-dessus puisque c'est celle qui rend le moins de pièces et billets.

## 4 Le rendu de monnaie glouton produit-il toujours une solution optimale ?

Les euros forment ce qu'on appelle un **système canonique**, c'est-à-dire un système pour lequel l'algorithme glouton trouve un rendu optimal. Pour le démontrer on peut analyser quelques combinaisons de pièces :

- les pièces et billets de 1, 5, 10, 50 et 100 ne peuvent pas apparaître deux fois dans un rendu optimal : en effet, on peut améliorer le rendu en remplaçant deux exemplaires de chacun de ces montants par un seul exemplaire du double du montant, par exemple deux billets de 5 par un billet de 10€ ;
- les pièces de 2€ et les billets de 20€ ne peuvent être utilisés que 2 fois au maximum pour un rendu optimal, par exemple, le rendu optimal de 6 € est un billet de 5 € et une pièce de 1 € plutôt que 3 pièces de 2 € ;
- dans un rendu optimal on ne peut avoir 2 exemplaires de 20€ (respectivement 2 € ) et un exemplaire de 10 € (respectivement 1 € ) car un billet de 50 € (respectivement un billet de 5 € ) serait un meilleur rendu.

## Exercice 2

Supposons qu'on paye 37 € à cette caisse automatique avec un billet de 100 €.

1. Quelle sera la composition de la monnaie rendue par la machine en utilisant l'algorithme glouton ?  
.....
2. Supposons maintenant que la machine n'ait plus à disposition de billets de 5 € et de 10 €. Quelle sera la monnaie rendue par la machine en utilisant l'algorithme glouton ? Le rendu est-il le meilleur possible ?  
.....

## 5 Synthèse : finalement un algorithme glouton c'est quoi ?

Les algorithmes gloutons sont des algorithmes permettant de déterminer des solutions de problèmes d'optimisation (problème dans lesquels on cherche un minimum ou un maximum) en faisant des choix qui semblent les meilleurs sur le moment.

Les solutions produites ne sont pas toujours les meilleures possibles et ne sont donc pas adaptées à toutes les situations. La stratégie gloutonne n'envisage pas toutes les solutions possibles mais elle a le mérite d'être réalisable en temps machine raisonnable.

*(Faire le TP correspondant dans Capytale.)*