

	Séquence 5	1 / 2
	Les algorithmes gloutons	

1 Objectif

Optimiser la résolution d'un problème

2 Principe d'un algorithme glouton

Un algorithme glouton permet d'optimiser la résolution d'un problème en utilisant l'approche suivante :

- on procède étape par étape,
- à chaque étape on fait le choix qui semble le meilleur,
- on ne remet jamais en question les choix passés.

3 Problème posé

3.1 Descriptif

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

Par exemple, on dispose de pièces de 1€, 2€ et des billets de 5€, 10€, 20€, 50€, 100€ et 200€. Pour chaque pièce et billet, le nombre d'exemplaires est illimité.

3.2 Problème

Anaïs veut acheter un objet qui coûte 53€. Elle règle son achat avec un billet de 100€. La caisse automatique doit donc lui rendre 47€.

3.3 Solution

La caisse automatique doit lui rendre sa monnaie de la façon suivante : 2 billets de 20€ plus 1 billet de 5€ et une pièce de 2€.

4 Résolution du problème

4.1 La force brute, solution naïve

Cette solution consiste à énumérer toutes les solutions possibles puis ensuite de choisir celle qui est optimale. C'est-à-dire celle qui minimise le nombre de pièces et de billets.

On peut donc totaliser la monnaie de 47€ de différentes façons.

Question 1. Proposer cinq solutions possibles de rendu de monnaie, **présenter** votre réponse sous la forme d'un tableau.

Il est possible de trouver la solution optimale globale au problème, mais le nombre de possibilités est très élevé.

Il en résulte que l'utilisation d'un algorithme permettant la résolution du problème serait très coûteuse en temps de calculs.

4.2 Application d'un algorithme glouton à notre problème

On définit :

- Les données :
 - o La somme donnée par la cliente à la caisse automatique : **somme**
 - o Liste des pièces et billets disponibles : **liste**
 - o Le coût de l'achat : **cout**
 - o La somme à rendre : **sommeRendre**
- Résultats renvoyé :
 - o une liste des pièces et billets à rendre : **monnaie**

- L'algorithme :
 - Saisir la somme donnée à la caisse automatique par la cliente
 - Initialiser :
 - la liste **liste** des pièces et des billets disponibles
 - la variable **cout** à la valeur de l'achat
 - la liste **monnaie** à une liste vide
 - Calculer la somme à rendre et initialiser la variable **sommeRendre** à cette valeur
 - Tant que la somme restante est strictement positive :
 - On choisit dans la liste la plus grande valeur qui ne dépasse pas la somme restante
 - On ajoute cette valeur à la liste **monnaie**
 - On calcule la nouvelle somme à rendre et on la place dans la variable **sommeRendre**
 - On renvoi la liste contenant les pièces et billets à rendre.

Question 2. **Planter** l'algorithme donné ci-avant. **Valider** votre programme.

4.3 Solution optimale dans tous les cas ?

La caisse automatique ne dispose plus de billets de 5€ et 10€. Anaïs effectue un nouvel achat de 37€. Elle paye toujours avec un billet de 100€.

Question 3. **Donner** la liste des pièces et billets rendu par la machine.

Question 4. **Que constatez-vous ? Que peut-on conclure ?**