

TD n°6

Le rendu de monnaie

On considère le problème suivant. Dans une monnaie donnée, il y a un certain ensemble P de pièces et billets.

Par exemple pour l'Euro $P = \{500, 200, 100, 50, 20, 10, 5, 2, 1, 0.5, 0.2, 0.1, 0.05, 0.02, 0.01\}$

Le but de ce TD est de parvenir à produire une somme S en utilisant le moins de pièces et billets possible. Ainsi pour $S = 60$ euros on préférera $50 + 10$ à $20 + 20 + 20$.

Exercice 1 Proposer un algorithme pour les pièces en euros. Analysez sa complexité.

Exceptionnellement, la preuve de sa correction vous est épargnée (car elle sera vue en cours dans 8 jours).

Exercice 2 Imaginer un pays bizarre où l'algorithme de l'exercice 1 ne marche pas (il faut produire un P et un S où il échoue)

Exercice 3 Proposer une fonction `rendreMonnaie(entier[] P, entier S)` qui résout le problème même pour les pays bizarres. La fonction sera récursive et renvoie le nombre de pièces et billets à considérer. Analysez sa complexité.

Exercice 4 En remarquant que ce TD est sur la programmation dynamique, améliorez la fonction précédente. Analyser la complexité en fonction de S

Exercice 5 Comment faire pour avoir la liste et non plus le nombre de pièces et billets à utiliser ?

Exercice 6 On considère maintenant le problème où les *deux* personnes peuvent donner des pièces.

Ainsi pour faire $S = 19.5$ euros il vaut mieux faire $20 - 0.50$ (je donne 50 centimes et reçois 20 euros) que $10 + 5 + 2 + 2 + 0.50$.

Que faire ?