NIVERSITÉ de Marne-la-Vallée

Travaux Dirigés d'algorithmique n°4

Cours d'informatique de deuxième année

—DEUG Sciences S3-MI—

Problème du sac à dos, tris

Dans un premier temps nous étudierons un problème qui se résoud comme ce lui du sac à dos, puis nous verrons deux tris, le tri à bulles et le tri rapide.

---

1 Monnayeur

L'objet de l'exercice suivant est la simulation d'un monnayeur : étant donnée une somme s, le monnayeur est une machine qui rend la somme s à l'aide du stock de pièces dont il dispose. Le stock de pièces est représenté par un tableau de couples d'entiers positifs dont la première composante est la valeur de la pièce et la seconde le nombre de pièces de cette valeur que le distributeur possède. Pour simplifier nous supposerons que les centimes n'existent pas, et que les valeurs des pièces sont 1, 2, 5, 10 et 20 francs (donc cinq genres de pièces). Nous supposerons également que dans le tableau représentant le stock de pièces les pièces sont triées par ordre croissant de valeur (*i.e.* les pièces de 1 franc sont à la première position, celles de 2 à la seconde, celles de 5 à la troisième...).

▸ **Exercice 1.** *Écrire une fonction récursive* `monnayeur()` *qui, pour une somme donnée, calcule en fonction du stock les pièces à rendre. Le monnayeur devra rendre des pièces de plus grande valeur possible. Par exemple, si le stock est :*

- *0 pièce de 1 franc,*
- *10 pièces de 2 francs,*
- *10 pièces de 5 francs,*
- *0 pièce de 10 francs,*
- *10 pièces de 20 francs,*

et qu'on appelle la fonction en voulant rendre 16 francs, la fonction calculera :

- *3 pièces de 2 francs,*
- *2 pièces de 5 francs.*

Autre exemple :

- *10 pièces de 1 franc,*
- *10 pièces de 2 francs,*
- *10 pièces de 5 francs,*
- *10 pièces de 10 francs,*
- *10 pièces de 20 francs,*

et qu'on appelle la fonction en voulant rendre 16 francs, la fonction calculera :

- *1 pièce de 1 franc,*
- *1 pièce de 5 francs,*
- *1 pièce de 10 francs,*

2 Tris

On souhaite maintenant trier un tableau t de n entiers par ordre croissant. On emploiera deux méthodes différentes : le tri à bulles et le "tri rapide".

▸ **Exercice 2.** *Le tri à bulles du tableau consiste à parcourir plusieurs fois le tableau de l'indice le plus petit vers le plus grand, en comparant à chaque étape d'un parcours l'élément courant avec celui d'indice immédiatement supérieur, et en les permutant éventuellement si l'élément courant est plus grand que l'autre. À la fin du premier parcours, ces permutations éventuelles ont pour effet de placer le plus grand élément du tableau en fin de tableau. Cet élément étant à sa place, le second parcours se fera sur* $t[0 \ldots n-2]$*, et ainsi de suite.*

- *Décrire les différentes étapes du tri à bulles sur le tableau* $4, 10, 2, 9, 1$ *;*
- *Écrire une fonction qui prend en argument un tableau d'entiers et sa taille, et qui pratique le tri à bulles dessus ;*
- *Calculer sa complexité ;*
- *En faire la preuve.*

▸ **Exercice 3.** *On définit le tri rapide de* t *de la façon suivante : on choisit une valeur* p *qui apparaît dans le tableau, et on réarrange le tableau (*p *change éventuellement de place) de telle manière que, à gauche de* p *n'apparaissent que des éléments plus petits que* p *et à droite des éléments plus grands. Il ne reste alors qu'à trier récursivement la partie du tableau à gauche de* p*, et celle à droite. On obtient ainsi un tableau trié.*

- *Décrire les différentes étapes du tri rapide sur le tableau* $4, 10, 2, 9, 1$ *;*
- *Implémenter en C le tri rapide ;*
- *En calculer la complexité dans le pire des cas.*