Algorithmique
Travaux dirigés, 8 octobre 2004
Louis Granboulan

1 La suite de Fibonacci

Définissons par récurrence $u_0 = 0$, $u_1 = 1$ et $\forall n > 1, u_n = u_{n-1} + u_{n+2}$.
- **Algorithme récursif.** Étudier la complexité (en fonction de n) du calcul de u_n par un algorithme récursif. Commenter.
- **Algorithme linéaire.** Proposer une variante où le calcul de chaque u_i est fait au plus une fois. Quelle est la mémoire nécessaire ?
- **Algorithme logarithmique.** Proposer une variante dont la complexité est logarithmique en fonction de n. Commenter.

2 Huffman

2.1 Code de Huffman

On se donne un alphabet $\mathcal{A} = \{a_1, ..., a_k\}$ et on veut coder par une suite binaire les suites d'éléments de $\mathcal{A}$. Un codage sera une fonction ϕ qui à tout élément de $\mathcal{A}$ associe une suite finie de 0 et de 1.
Un exemple est le codage de Huffman, qu'on peut définir récursivement comme suit : si $k = 2$, on code $\phi(a_1) = 0$ et $\phi(a_2) = 1$. Si la probabilité d'occurence du symbole a_i est notée p_i, et si a_{k-1} et a_{k-2} sont les deux symboles les moins probables, on construit le code de Huffman ψ de l'alphabet $\{a_1, ..., a_{k-2}, b_{k-1}\}$ où la probabilité d'occurence des a_i est p_i et où la probabilité d'occurence de b_{k-1} est $p_{k-1} + p_{k-2}$, puis on définit $\phi(a_k) = \psi(b_{k-1})\|1$, $\phi(a_{k-1}) = \psi(b_{k-1})\|0$ et $\phi(a_i) = \psi(a_i)$ pour $i < k - 1$.
- **Correction.** Montrer que la fonction qui à une suite finie d'éléments de $\mathcal{A}$ associe la concaténation de leurs codages est inversible.
- **Complexité.** Comment décrire les algorithmes de codage et de décodage ? Quelle est leur complexité ?
- **Optimalité.** Définir ce qu'est un code préfixe. Définir ce qu'est un code préfixe optimal. Montrer que le code de Huffman l'est.

2.2 Code de Huffman adaptatif

Appelons *code de Huffman fourre-tout* le code de Huffman d'un alphabet dont un et un seul symbole a probabilité 0, le symbole fourre-tout.
Le codage de Huffman adaptatif sert dans le cas où la probabilité d'occurence de chaque symbole est inconnue. On code les symboles un par un. Pour coder le symbole suivant, on construit le code de Huffman fourre-tout correspondant aux symboles déjà apparus, avec leur fréquence. Si ce symbole n'est pas déjà apparu, il est codé par le codage du symbole fourre-tout.
- **Correction.** Montrer que la fonction qui à une suite finie d'éléments de $\mathcal{A}$ associe la concaténation de leurs codages est inversible.
- **Complexité.** Comment décrire les algorithmes de codage et de décodage ? Quelle est leur complexité ?

3 Recouvrement de sommets

Étant donné un graphe non orienté connexe valué, un *arbre minimum de recouvrement* est un sous-graphe connexe sans cycle, contenant tous les sommets et dont la somme des poids des arêtes est minimale.
Proposer des algorithmes résolvant ce problème. Les comparer.
Que penser du cas où les sommets sont des points du plan et le graphe est complet ?

4 Voyageur de commerce

Étant donné un graphe fortement connexe valué, une solution du problème du voyageur de commerce est un chemin de poids minimal passant par tous les sommets.
Proposer des algorithmes et en étudier la complexité.