Travaux dirigés

Codage de Huffman

On considère des codes binaires, c'est-à-dire une fonction injective qui, à chaque caractère d'un alphabet, associe un mot binaire. Un code préfixe est un code tel qu'aucun mot de ce code n'est préfixe d'un autre. Un tel code est toujours représentable par un arbre : on peut lire la représentation d'un caractère porté par une feuille par la suite de choix fils gauche/fils droit y menant depuis la racine.

1. Un arbre $\mathcal{T}$ est dit optimal pour l'alphabet A lorsque la quantité $Q_A(T) = \sum_{a \in A} p(a)l(a)$ est minimale, où $p(a)$ est la probabilité de lire la lettre a, et $l(a)$ la longueur du mot codant a, donc la profondeur de a dans l'arbre T. Montrer qu'un arbre optimal est nécessairement binaire strict.
2. Soient x et y deux lettres de A de plus petit poids. Montrer qu'il existe un arbre (de code) optimal pour lequel ces deux lettres ont le même père et sont de profondeur maximale.
3. Montrer que l'algorithme de Huffman produit bien un arbre optimal.