Feuille d'exercices no. 9

Codage de Huffman

Exercice 1. **Huffman et Fibonacci**
On rappelle que la suite de Fibonacci est définie comme suit : $F(1) = 1$, $F(2) = 1$ et $F(i) = F(i-2) + F(i-1)$ pour tout $i \geq 2$.

1. Déterminez le codage de Huffman de l'ensemble des lettres $\{a, b, c, d, e, f, g, h\}$ dont les fréquences correspondent aux 8 premiers nombres de Fibonacci :

$$a:1 \quad b:1 \quad c:2 \quad d:3 \quad e:5 \quad f:8 \quad g:13 \quad h:21$$

2. Calculez le gain en nombre de bits de ce codage par rapport à un codage de longueur fixe.
3. Généralisez la réponse en considérant n lettres dont les fréquences correspondent aux n premiers nombres de Fibonacci.

Exercice 2. **Codage**
Pour identifier le code d'un caractère à partir d'un arbre de codage Huffman, il faut parcourir l'arbre en recherchant quelle feuille porte le caractère que l'on veut coder et quel est le chemin qui y mène. Nous souhaitons coder un texte T à partir de son arbre de Huffman A_T. Pour éviter d'effectuer un parcours de l'arbre pour chaque caractère, nous pouvons réaliser un seul parcours et nous construisons un tableau C dont les indices sont les caractères et les valeurs sont leurs codes binaires correspondants.

1. Écrire une fonction ConstruireCode qui crée le tableau C à partir d'un arbre A_T correspondant à un texte T. Calculer la complexité de cette fonction.
2. Comparer la complexité de l'algorithme de codage ainsi obtenu avec l'algorithme de codage qui consiste à parcourir l'arbre chaque fois qu'une lettre doit être codée.

Exercice 3. **Code de Huffman canonique**
Dans un établissement scolaire, les élèves sont évalués avec un système de quantification par lettres : A, B, C, D, E, F. On souhaite coder ces notes en binaire. Sur l'ensemble des élèves, les notes apparaissent avec des probabilités respectives de 0.1, 0.2, 0.55, 0.06, 0.07 et 0.02.

1. Déterminez le codage de Huffman pour cet ensemble de notes. Est-ce que l'arbre de Huffman obtenu est un arbre de Huffman canonique ou non ? Si non, transformez-le en un arbre de Huffman canonique à la main, mais en suivant la démarche proposée dans le cours.
2. Ecrire un algorithme qui, étant donnée une liste L d'entiers triés par ordre croissant, construit un arbre binaire dont les feuilles ont les profondeurs indiquées dans la liste L, dans l'ordre de gauche à droite des feuilles. *Note.* On suppose que le contenu de la liste L assure l'existence de l'arbre.
3. Supposons maintenant qu'un texte (représentant par exemple les notes d'un groupe d'élèves) a été codé à l'aide de l'arbre Huffman canonique pour l'exemple ci-dessus. Le texte codé est :

$$S = 1110111111100$$

Vous ne connaissez pas l'arbre, mais on vous transmet pour le décodage les fréquences des lettres et l'information suivante : $min = 1$, $max = 5$, distribution : 1, 1, 1, 1, 2. Décodez le texte encodé par S.

Pour aller plus loin

Exercice 4. [1] On rappelle que les nombres premiers plus petits que 100 sont : 2 3 5 7 11 13 17 19 23 29 31 37 41 43 47 53 59 61 67 71 73 79 83 89 et 97. On considere le mot u, de longueur 100, sur l'alphabet binaire $B = \{0, 1\}$ dont les lettres sont $u_1 u_2 \ldots u_{100}$, avec $u_i = 1$ si i est un nombre premier et 0 sinon (il n'est pas nécessaire d'écrire le mot en entier pour répondre aux questions suivantes). On découpe le mot en blocs de taille 2, de la forme $u_{2n-1}u_{2n}$, pour $1 \leq n \leq 50$.

1. Indiquez le nombre d'occurrences de chacun de ces blocs dans u (attention au début du mot avec la particularité de 2).
2. Construisez un arbre de Huffman sur les blocs de longueur 2.
3. Quelle sera la taille du message une fois compressé par la méthode d'Huffman ?

1. ©pageperso.lif.univ-mrs.fr/∼laurent.braud/man/