Compression & Algo du Texte

– M1 –

2015-2016

Examen : durée 2h. Aucun document électronique n'est autorisé. Les notes de cours sont autorisées.

▶ **Exercice 1** ◀ On considère le message suivant : $u = ceciestletextedelexam$

(a) Quelle sera la taille du codage du message en code ASCII ?

(b) Quelle sera la taille minimale du codage du message en code de longueur fixe ?

(c) Appliquez le procédé de Huffman et proposez un code pour représenter le message. Quelle est la taille du message une fois encodé ?

(d) Comment encoder l'arbre de Huffman ? Quelle est la taille totale de l'encodage de l'arbre et du message ?

▶ **Exercice 2** ◀ Dans tout l'exercice, la taille du buffer et la longueur maximale sont fixés à 8.

(a) Donnez le résultat de l'encodage LZ77 appliqué au mot $u = bbaabababababb$.

(b) On reçoit le message $[b, a, r, a, (3, 2), (1, 2), r]$ qui a été encodé par LZSS avec un seuil de 2. Décompressez-le.

▶ **Exercice 3** ◀ Dans cet exercice, on considère que les dictionnaires sont de tailles illimitées.

(a) Compressez le message *bbaaababbaba* sur l'alphabet $\{a, b\}$ avec l'algorithme de LZ78.

(b) On a encodé un message avec la méthode LZW, en initialisant le dictionnaire à $a \mapsto 0$ et $b \mapsto 1$. On a obtenu $[1, 0, 2, 4, 2]$. Retrouvez le message initial.

▶ **Exercice 4** ◀ On part du message $u = abaabcabdab$.

(a) Calculez la transformée de Burrows-Wheeler v de u.

(b) Calculez la transformation Move-to-Front w de v.

(c) Appliquez l'algorithme de Huffman à w. Combien de bits faut-il pour encoder w ?

▶ **Exercice 5** ◀ On rappelle que la suite de Fibonacci est la suite définie par

$$\begin{cases} u_0 = 0, \ u_1 = 1 \\ u_n = u_{n-1} + u_{n-2}, \quad n \geq 2 \end{cases}$$

(a) Calculez les valeurs de u_i pour i de 0 à 6.

Si $(b_0, b_1, \ldots, b_{\ell-1})$ est une séquence de ℓ bits, on note $f((b_0, b_1, \ldots, b_{\ell-1}))$ le nombre :

$$f((b_0, b_1, \ldots, b_{\ell-1})) = b_0 u_0 + b_1 u_1 + \ldots + b_{\ell-1} u_{\ell-1}.$$

On dit que $(b_0, b_1, \ldots, b_{\ell-1})$ est l'encodage de Fibonacci du nombre $f((b_0, b_1, \ldots, b_{\ell-1}))$.

(b) Calculez $f((1, 1, 1, 0, 1))$. Trouvez deux suite binaires différentes encodant le même nombre.

(c) Ecrire un algorithme qui calcule le nombre encodé par l'encodage de Fibonacci d'une suite de bits (donnée par une chaîne de caractères '0' ou '1').

(d) On admet que tout nombre peut être encodé par une suite binaire. Montrez que tout nombre peut être encodé par une suite binaire **qui ne possède pas deux 1 consécutifs.**

Le *codage par tailles des plages* consiste à découper le mot en blocs ne contenant que des lettres identiques, puis à encoder chaque bloc par un couple (lettre, nombre d'occurrences). Par exemple *aaaaaa* est encodé par le couple $(a, 6)$.

(e) Donnez l'encodage par taille des plages du mot *bbbaaaaabbbbccccccccaaa*.

(f) On remarque que si on s'arrête au dernier 1 (pas de 0 inutiles à la fin) et qu'on ajoute un 1 à la fin de l'encodage de Fibonacci d'un nombre, on sait exactement quand l'écriture du nombre est finie à cause des deux 1 consécutifs. En utilisant cette propriété, proposez une façon d'encoder la séquence de couples du codage par taille de plages. (On vous demande juste d'expliquer brièvement comment faire, en supposant qu'on a un algorithme pour calculer un encodage de Fibonacci sans deux 1 consécutif d'un nombre).

(g) Appliquez votre méthode à la séquence obtenue à la question **(e)**. On n'encodera pas les caractères en ASCII, on écrira juste $\underline{a}$ pour l'encodage ASCII du caractère a.