

# Compresseur Huffman

Mathieu LADEUIL Jean-Charles ALLA

December 2019

## 1 Introduction

Ce document est un micro-rapport détaillant le fonctionnement et les choix effectués pour la réalisation d'un compresseur/decompresseur Huffman en C.

## 2 Codage de Huffman

Le code est déterminé à partir d'une estimation des probabilités d'apparition des symboles de source, un code court étant associé aux symboles de source les plus fréquents. Dans un premier temps, on récupère les fréquences de chaque caractère du fichier à compresser, on va ensuite créer un arbre de cette façon :

- Tous les caractères vont être associés à des feuilles d'un arbre binaire
- On prends les 2 noeuds (qui peuvent être des feuilles) de fréquence minimum et on les associe à un parent.
- On réitère le processus jusqu'à ce qu'on arrive à la racine. Ensuite pour avoir le codage on associe les branches gauches à 0 et les branches droites à 1.

1

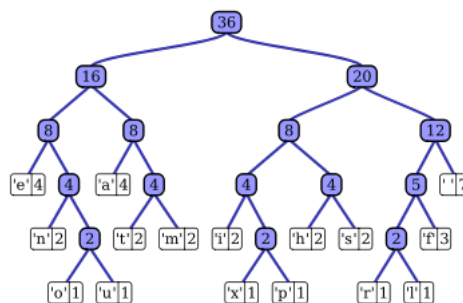


Figure 1: Exemple d'arbre Huffman

---

<sup>1</sup>[https://fr.wikipedia.org/wiki/Codage\\_de\\_Huffman](https://fr.wikipedia.org/wiki/Codage_de_Huffman)

### 3 Tas informatique

Le tas est une structure de données de type arbre qui permet de retrouver directement l'élément que l'on veut traiter en priorité. Pour utiliser un tas, les clés sont ordonnées selon la propriété de tas : la clé d'un nœud parent a une plus haute priorité que les clés de ses enfants. La "priorité" signifie ici que les clés des enfants sont toutes inférieures, suivant que le tas est ordonné pour avoir en racine la clé maximale (max heap).<sup>2</sup> Nous avons choisi le tas, car il paraissait bien adapté au Huffman, et qu'il présentait un intérêt à l'implémenter.

### 4 Compression du fichier

On écrit un entête pour la décompression puis on parcourt le fichier et on remplace les caractères par les codes de l'arbre de Huffman. A la fin, on écrit un pseudo-EOF pour le décompresseur, on ajoute des bits complémentaires pour avoir un octet rempli.

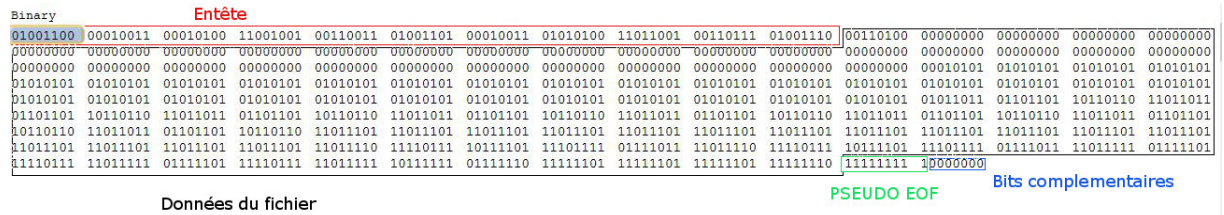


Figure 2: Exemple en binaire de fichier compressé

#### 4.1 Codage de l'entête

Nous avons choisi de coder l'arbre dans le fichier compressé, ainsi nous parcourons l'arbre construit par le tas, on inscrit un 0 si on est sur un nœud, et un 1 si on est sur une feuille, et dans ce dernier cas, on écrit le caractère. Cela nous permet d'avoir un entête plus court qu'avec un dictionnaire et de garder la structure de l'arbre.

### 5 Decompression du fichier

On décode l'entête, puis le reste du fichier en s'arrêtant au EOF. Pour l'entête on reconstruit l'arbre simplement, si on lit un 0 c'est un nœud sinon c'est une feuille et on peut réutiliser le même arbre qu'en compression.

<sup>2</sup>[fr.wikipedia.org/wiki/Tas\\_\(informatique\)](https://fr.wikipedia.org/wiki/Tas_(informatique))