

Les participants :

MASHRA Marwan

CAO Anh

Université de Montpellier

HLIN303 – Système d'exploitation

16 décembre 2019

Rapport de Projet Huffman

L'archive de projet contient 3 fichiers :

1) *compresseur_huffman.c* :

- `./compresseur_huffman [nom de fichier original] [nom de fichier après la compression]`
 - Le compresseur commence par lire le fichier originale, puis construire un tableau de fréquence pour tous les lettres ASCII 256.
 - Un arbre est construit par la suite à base du tableau de fréquence. L'arbre est représenté par un tableau de 511 nœud.
 - Chaque nœud est représenté par l'index de son père, son fils droit, son fils gauche ainsi que sa fréquence.
 - Un code Huffman est généré pour chaque caractère présent dans l'arbre (1 en allant à droite et 0 en allant à gauche).
 - Le fichier compressé est de la forme suivante :
 - Taille : index1 : nbOcc1 : index2 : nbOcc2 : index3 : nbOcc3 : #texte compressé
- Taille : est le nombre de bits écrits dans le texte compressé.
- : est un séparateur entre les nombres de l'entête.
- Index est la valeur ASCII du caractère.
- nbOcc est le nombre d'occurrence du caractère dans le fichier d'origine
- # est le séparateur entre l'entête et le texte compressé
- Le texte compressé est la substitution de chaque caractère par son code Huffman.
- L'écriture du texte compressé se fait en utilisant les opérateurs binaires.

2) *decompresseur_huffman.c* :

- `./decompresseur_huffman [nom de fichier compressé] [nom de fichier après la décompression]`

- Le décompresser commence par lire l'entête pour reconstruire l'arbre.
- Il ouvre ensuite le fichier compressé en mode binaire et il le lit en bit par bit en utilisant les opérateurs binaires.
- Selon ce qu'il lit, il parcourt l'arbre pour retrouver les caractères et les écrire dans le fichier décompressé.

3) *huffman.py* :

- `huffman.py` [option] [nom de fichier]

[option] : vous devez impérativement choisir une des deux option

1. -a : pour archiver
2. -d : pour désarchiver

[nom de fichier] : vous rentrez le nom de fichier (ou de dossier) que vous voulez archivez (ou désarchiver).

- Dans le cas d'archivage :

Le script python crée un fichier de passage qui est composé de deux parties :

1. La concaténation des fichiers textes qui sont dans le dossier à archiver.
2. L'entête qui contient :
 - Le nombre de fichier.
 - Les noms des fichiers.
 - Le nombres de ligne dans chaque fichier.

Le fichier de passage sera par la suite compressé à l'aide de compresseur huffman.

Le fichier de passage et le dossier sont ensuite supprimés.

➤ Dans le cas de désarchivage :

Le script python décompresse le fichier dans un fichier de passage à l'aide de décompresseur huffman.

Il lit l'entête pour récupérer les informations sur les fichiers.

Les dossiers nécessaires sont créés ainsi que les fichiers avec leur texte.

Le fichier de passage et le fichier compressé sont par la suite supprimés.