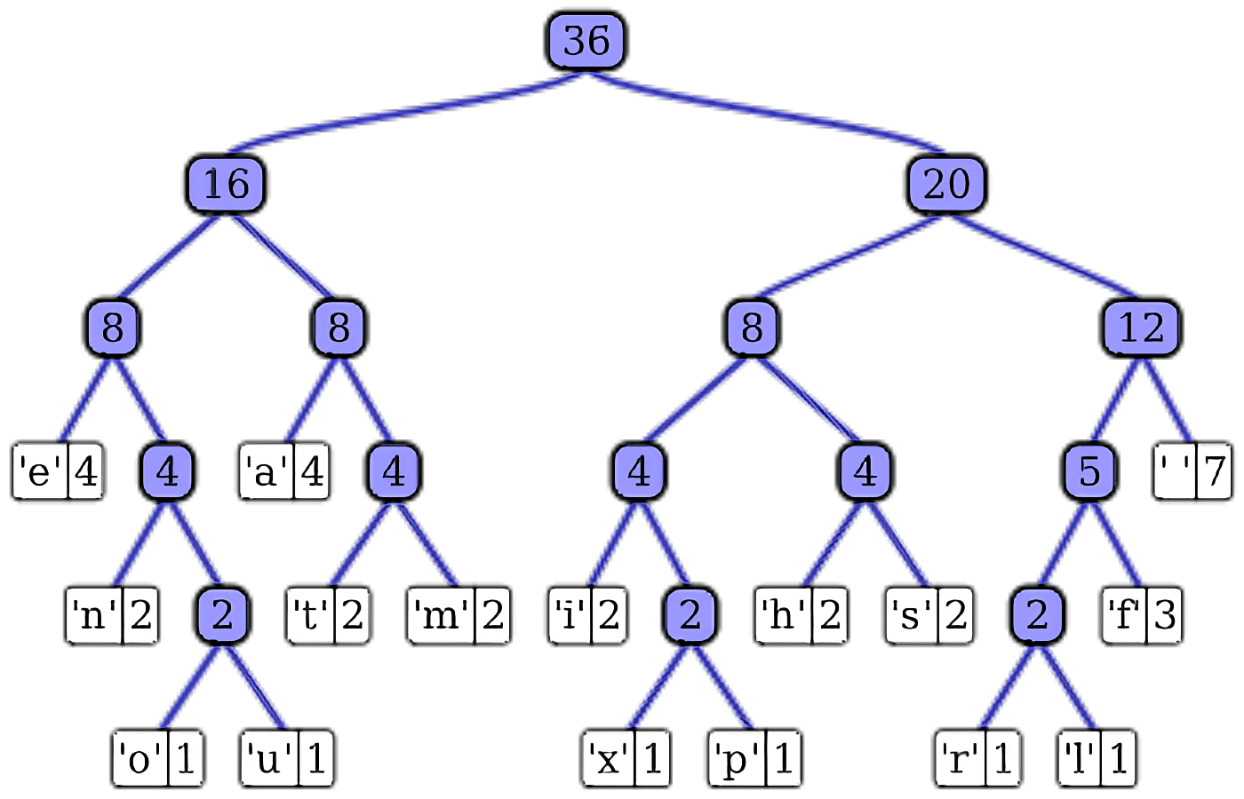


**Rapport de projet**

Année scolaire : 2021-2022

PROJET HUFFMAN



Pré-ing 2 Groupe 5 : PAUL JUMEL, COURTOIS LOUIS, KALMUK MARC-ANTOINE

Professeur référent : ROMUALD GRIGNON

**Sommaire**

* **I.** **INTRODUCTION**
* **II. REPARTTION DES TACHES ET ORGANISATION**
* **III. AVANTAGES ET LIMITES/DIFFICULTES**
* **IV. README**
* **V. CONCLUSION**

# Introduction

**Qu’est-ce que l’algorithme Huffman?**

La compression de données est un aspect important en informatique. Le but de ce projet est de comprendre et d’implémenter le comportement d’un algorithme par l’écriture un programme permettant de compresser des fichiers textes afin qu'ils occupent moins d'espace sur un disque. On demande également d'écrire un décompresseur qui devra restaurer le fichier original. Il existe beaucoup de méthodes de compression. Celui proposé ici est l'algorithme de Huffman.

**Comment fonctionne-t-il ?**

L'algorithme construit de façon itérative un arbre de codage correspondant au code optimal, en regroupant à chaque itération les deux nœuds possédant le moins de fréquence dans le texte.

Pour chaque caractère ayant une fréquence d’apparition, on construit un nœud contenant le caractère et sa fréquence. On insère ces nœuds, de manière ordonnée croissante par rapport à la fréquence, dans une liste de nœuds.

On parcourt cette liste pour construire l’arbre, de la manière suivante. On prend les deux premiers nœuds (en les enlevant de la liste), on les fusionne de sorte qu’ils se retrouvent fils droit et fils gauche d’un nouveau nœud qui contient la somme de leurs fréquences, puis on l’insère ce nœud créé dans la liste.

Le code en bit de chaque caractère est donné par le parcours de l’arbre, on note zéro lorsque l’on part à gauche, 1 lorsque l’on part à droite. Ainsi, plus un caractère est fréquent dans le texte, plus le nombre de bits sera petit.

Enfin l’algorithme génère un fichier contenant : en entête, le nombre de lettres, le nombre de caractères codés, les caractères avec leurs fréquences, leur nombre de bits et la valeur en base 10. Puis, le contenu du fichier originel codé avec le nouveau code de chaque caractère.

Répartition des tâches et organisation

* Lorsque nous avons su qu'il fallait réaliser ce projet, nous avions deux ou trois semaines pour le faire, ce qui nous semblait très complexe au vu des différents DS que nous devions réviser en parallèle. Le fait d'avoir reculé l'échéance nous a grandement soulagé mais la quantité de travail restait la même, nous savions qu'il faudrait beaucoup travailler.
* Nous avons donc travaillé de manière régulière afin de ne pas être submergé par la masse de travail dus à un potentiel retard. Le travail fourni était sérieux et nous restions parfois plusieurs heures après les cours pour travailler le projet. Nous avions estimé qu'un partage des tâches nous aurait grandement desservi, ainsi, nous avons fait le projet à trois en entier, parfois à deux quand l'un d'entre nous n'était pas là. La dernière semaine nous avons utilisé la plateforme en ligne Repl.it afin de pouvoir écrire sur le même code à partir d'ordinateurs différents.

Avantages et limités / difficultés

**Avantages :**

* Nous avons créé un menu pour faciliter les choix.
  + 1. Compression
  + 2. Décompression
  + 3. Manuel
* La facilité d’utilisation du programme, il suffit de rentrer le nom du fichier à compresser ou à décompresser, puis de rentrer le nom du fichier à créer.

**Limites :**

* Le fichier ne compresse pas réellement, il fait même l’inverse. Mais la méthode Huffman est respectée, dans le fichier compressé, nous retrouvons bien un train binaire.
* Dans le texte compressé, il ne faut pas de mot avec des accents.
* On peut seulement compresser des fichiers au format txt.

**Difficultés**

* Nous n’avons pas réussi à faire le décalage de bits pour avoir un fichier réellement compresser.
* Nous avons rencontré aussi plusieurs difficultés sur la création des fichiers. Nous avons donc utilisé les différentes fonctions associées à la création et à la lecture de fichier. Pour cela on s’est renseigné sur différents sites. On a découvert pour la première fois les fonctions fprintf et fscanf, des fonctions exigeantes.

**Readme**

* Si vous voulez compresser un fichier txt, il faut taper 1, puis rentrer le nom de votre fichier avec .txt, puis nommé le nom du fichier compressé.

Une image contenant texte

Description générée automatiquement

* Pour décompresser le fichier créé, il faut taper 2, puis rentrer le nom du fichier compressé, puis nommé le nouveau nom du fichier décompressé.

Une image contenant texte

Description générée automatiquement

**Conclusion**

* En conclusion, ce projet nous a permis de considérablement augmenter notre niveau global en codage et notre autonomie car nous avons dû nous-mêmes réfléchir à la transcription de ce que nous voulions faire ainsi que chercher à nous renseigner sur les outils permettant de mettre en œuvre ce que nous imaginions.

Cela a été une belle expérience et, bien que notre compression soit incomplète, nous avons beaucoup apprécié travailler sur ce projet. Nous sommes pour ainsi dire assez fiers du résultat.

* Un grand merci à M. Grignon de nous avoir aiguillé tout au long du projet.

Paul, Louis, Marc-Antoine.

Pré-ING 2 groupe 5.

2020-2021.