Huffman

Generated by Doxygen 1.8.20

Chapter 1

Data Structure Index

1.1 Data Structures

Here are the data structures with brief descriptions:

Chapter 2

File Index

2.1 File List

Here is a list of all documented files with brief descriptions:

Chapter 3

Data Structure Documentation

3.1 indexi Struct Reference

structure de mon index pour le recuperer du fichier compresser

Data Fields

- int **ascii**
- char **c** [255]
- unsigned char **ascii**
- char * **c**

3.1.1 Detailed Description

structure de mon index pour le recuperer du fichier compresser

structure de mon index pour le transmettre dans le fichier compresser avec l'indice ascii et son code binaire en chaine de caratère

The documentation for this struct was generated from the following files:

- dehuff.c
- huff.c

3.2 noeud Struct Reference

structure d'un noeud d'un arbre avec le nombre d'occurence du caratère à son indice ascii, l'indice de son parent et de ses fils gauche et droit

Data Fields

- int **nbOcc**
- int **gauche**
- int **droit**
- int **parent**

3.2.1 Detailed Description

structure d'un noeud d'un arbre avec le nombre d'occurence du caratère à son indice ascii, l'indice de son parent et de ses fils gauche et droit

The documentation for this struct was generated from the following file:

- huff.c

Chapter 4

File Documentation

4.1 dehuff.c File Reference

Decompresseur de Huffman d'un fichier passé en paramètre sur la sortie standard.

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
```

Data Structures

- struct indexi

  structure de mon index pour le recuperer du fichier compresser

Functions

- int **main** (int argc, char ∗argv[ ])

4.1.1 Detailed Description

Decompresseur de Huffman d'un fichier passé en paramètre sur la sortie standard.

Author

Ryan Bengoufa et Thomas Bergon

Version

0.1

Date

13/12/2020

4.2 huff.c File Reference

Compresseur de Huffman d'un fichier passé en paramètre, génere le codage dans un autre fichier passé en paramètre.

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <sys/stat.h>
```

Data Structures

- struct noeud

  structure d'un noeud d'un arbre avec le nombre d'occurence du caratère à son indice ascii, l'indice de son parent et de ses fils gauche et droit
- struct indexi

  structure de mon index pour le recuperer du fichier compresser

Functions

- void ajoutChar (char ∗s, char c)

  ajout d'un caractere c à la fin d'une chaine de caratère s
- void reverse (char ∗s)

  inverse une chaine de caratère
- void encodage (char ∗fichiers, char ∗fichierd, char ∗∗tabBin)

  encodage du fichier avec la methode de Huffman
- void codeFeuille (noeud ∗arbre, int feuille, char ∗chaine)

  Ajoute le code binaire en suivant l'arbre de Huffman depuis la feuille donc le code est à l'envers.
- void codeBin (noeud ∗arbre, char ∗∗tabBin)

  Cherche les code binaire necessaires pour coder le fichier et les enregistre dans un tableau.
- int comptage (char ∗fichier, int ∗nbOcc)

  compte le nombre de caractere différents et enregistre à l'indice ascii son nombre d'occurence
- void affichageArbre (noeud ∗arbre, int taille)

  affiche l'arbre binaire
- void construction (int ∗nbOcc, noeud ∗arbre, int taille)

  construit un arbre binaire avec les différents caratères enregistrer dans la fonction comptage
- int main (int argc, char ∗argv[ ])

  fonction principal du compresseur

4.2.1 Detailed Description

Compresseur de Huffman d'un fichier passé en paramètre, génere le codage dans un autre fichier passé en paramètre.

Author

Ryan Bengoufa et Thomas Bergon

Version

0.1

Date

13/12/2020

4.2.2 Function Documentation

4.2.2.1 affichageArbre()

```
void affichageArbre (
            noeud * arbre,
            int taille )
```

affiche l'arbre binaire

Parameters

arbre	arbre binaire
taille	taille de l'arbre

4.2.2.2 ajoutChar()

```
void ajoutChar (
            char * s,
            char c )
```

ajout d'un caractere c à la fin d'une chaine de caratère s

Parameters

s	chaine de caratère
c	caratère

4.2.2.3 codeBin()

```
void codeBin (
            noeud * arbre,
            char ** tabBin )
```

Cherche les code binaire necessaires pour coder le fichier et les enregistre dans un tableau.

Parameters

arbre	arbre de Huffman
tabBin	tableau avec les différents code bianires (vide avant d'avoir lancer la focntion)

4.2.2.4 codeFeuille()

```
void codeFeuille (
            noeud * arbre,
            int feuille,
            char * chaine )
```

Ajoute le code binaire en suivant l'arbre de Huffman depuis la feuille donc le code est à l'envers.

Parameters

arbre	arbre de Huffman
feuille	code ascii de la feuille dont je recherche le code
chaine	chaine de caratère où sera stocké le code binaire

4.2.2.5 comptage()

```
int comptage (
            char * fichier,
            int * nbOcc )
```

compte le nombre de caractere différents et enregistre à l'indice ascii son nombre d'occurence

Parameters

fichier	fichier en lecture
nbOcc	tableau de int

Returns

nbFeuille le nombre de caractere différents dans le fichier

4.2.2.6 construction()

```
void construction (
            int * nbOcc,
            noeud * arbre,
            int taille )
```

construit un arbre binaire avec les différents caratères enregistrer dans la fonction comptage

Parameters

nbOcc	tableau de int avec les différents code ascii et leur fréquence
arbre	arbre binaire où sera stocker
taille	taille de l'arbre

4.2.2.7 encodage()

```
void encodage (
            char * fichiers,
            char * fichierd,
            char ** tabBin )
```

encodage du fichier avec la methode de Huffman

Parameters

fichiers	fichier qui correspond à la source
fichiers	fichier qui correspond à la destination
tabBin	tableau de chaine de caractère qui contient les codes binaires des différents code ascii qu'il y a dans le fichier source

4.2.2.8 main()

```
int main (
            int argc,
            char * argv[] )
```

fonction principal du compresseur

Parameters

argc	nombre d'argument passé en paramètre
argv	tableau avec le nom des differents paramètres

4.2.2.9 reverse()

```
void reverse (
            char * s )
```

inverse une chaine de caratère

Parameters

s	chaine de caratère

Index