

Compresseur / Décompresseur de Huffman

1.1

Généré par Doxygen 1.8.13

Table des matières

1	Index des classes	1
1.1	Liste des classes	1
2	Index des fichiers	3
2.1	Liste des fichiers	3
3	Documentation des classes	5
3.1	Référence de la structure Code	5
3.1.1	Description détaillée	5
3.1.2	Documentation des données membres	5
3.1.2.1	bit	5
3.2	Référence de la structure Noeud	5
3.2.1	Description détaillée	6
4	Documentation des fichiers	7
4.1	Référence du fichier compress.h	7
4.1.1	Description détaillée	8
4.1.2	Documentation des fonctions	8
4.1.2.1	compression()	8
4.2	Référence du fichier decompress.h	9
4.2.1	Description détaillée	10
4.2.2	Documentation des fonctions	10
4.2.2.1	decompression()	10
4.3	Référence du fichier header.h	10
4.3.1	Description détaillée	12
4.3.2	Documentation des fonctions	12
4.3.2.1	frequency()	12
4.3.2.2	getHeader()	12
4.3.2.3	saveHeader()	13
4.3.2.4	tableauHuffmann()	13
4.4	Référence du fichier main.c	14
4.4.1	Description détaillée	14
4.4.2	Documentation des fonctions	15
4.4.2.1	help()	15
4.4.2.2	main()	15

Index	17
-----------------------	----

Chapitre 1

Index des classes

1.1 Liste des classes

Liste des classes, structures, unions et interfaces avec une brève description :

Code	Codage d'un caractère	5
Noeud	Noeud de l'arbre d'Huffman	5

Chapitre 2

Index des fichiers

2.1 Liste des fichiers

Liste de tous les fichiers documentés avec une brève description :

compress.h	Fonction de compilation Huffman	7
decompress.h	Fonction de décompilation Huffman	9
header.h	Fonction de Création d'index Huffman	10
main.c	Programme de Création d'index, compilation et décompilation Huffman	14

Chapitre 3

Documentation des classes

3.1 Référence de la structure Code

```
#include <header.h>
```

Attributs publics

- int [bit](#)
- unsigned char * **code**

3.1.1 Description détaillée

bit est le nombre de bit du code. code est le code.

3.1.2 Documentation des données membres

3.1.2.1 bit

```
int Code::bit
```

bit != (byte||octet)

La documentation de cette structure a été générée à partir du fichier suivant :

- [header.h](#)

3.2 Référence de la structure Noeud

```
#include <header.h>
```

Attributs publics

- int **pere**
- int **fg**
- int **fd**
- double **freq**

3.2.1 Description détaillée

pere est le père du [Noeud](#). fg est le fils gauche du [Noeud](#). fd est le fils droit du [Noeud](#). freq est le poid du noeud. Le poid peut être négligé dans certaine condition d'utilisation comme la décompression.

La documentation de cette structure a été générée à partir du fichier suivant :

- [header.h](#)

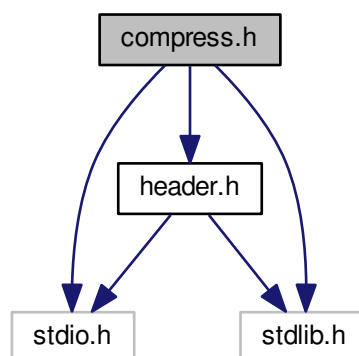
Chapitre 4

Documentation des fichiers

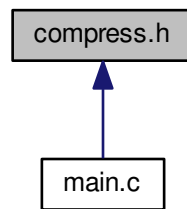
4.1 Référence du fichier compress.h

```
#include <stdio.h>
#include <stdlib.h>
#include "header.h"
```

Graphe des dépendances par inclusion de compress.h :



Ce graphe montre quels fichiers incluent directement ou indirectement ce fichier :



Fonctions

— void `compression` (FILE *input, FILE *output, Code *codes)

4.1.1 Description détaillée

Auteur

Hareski

Version

1.1

Date

16 décembre 2017

4.1.2 Documentation des fonctions

4.1.2.1 `compression()`

```
void compression (
    FILE * input,
    FILE * output,
    Code * codes )
```

Fonction de compilation Huffman.

Paramètres

<i>input</i>	FILE* : fichier à analyser au niveau de l'encodage
<i>output</i>	FILE* : fichier pour la compression
<i>codes</i>	Code* : codage des caractères ASCII

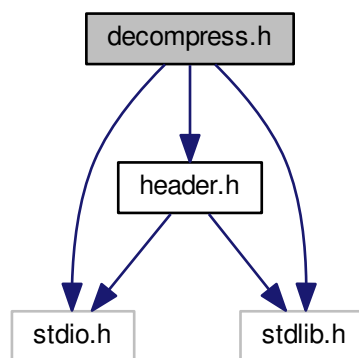
Renvoie

void

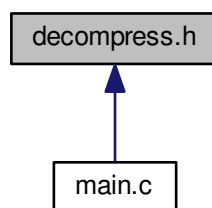
4.2 Référence du fichier decompress.h

```
#include <stdio.h>
#include <stdlib.h>
#include "header.h"
```

Graphe des dépendances par inclusion de decompress.h :



Ce graphe montre quels fichiers incluent directement ou indirectement ce fichier :



Fonctions

— void [decompression](#) (FILE *input, FILE *output, [Noeud *tableauHuffman](#), int nbrChar, int nbrASCII)

4.2.1 Description détaillée

Auteur

Hareski

Version

1.1

Date

16 décembre 2017

4.2.2 Documentation des fonctions

4.2.2.1 decompression()

```
void decompression (
    FILE * input,
    FILE * output,
    Noeud * tableauHuffmann,
    int nbrChar,
    int nbrASCII )
```

Fonction de décompilation Huffman.

Paramètres

<i>input</i>	FILE* : fichier à analyser au niveau de l'encodage
<i>output</i>	FILE* : fichier pour l'extraction
<i>tableauHuffmann</i>	Noeud* : tableau de Huffmann de l'index
<i>nbrChar</i>	int : nombre de caractères dans input

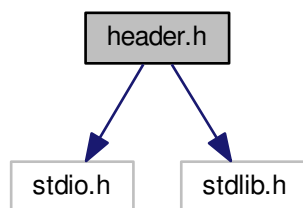
Renvoie

void

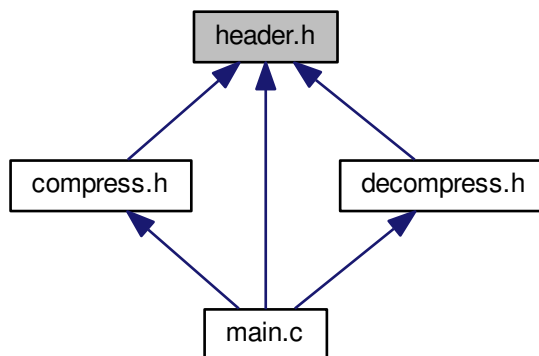
4.3 Référence du fichier header.h

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

Graphe des dépendances par inclusion de header.h :



Ce graphe montre quels fichiers incluent directement ou indirectement ce fichier :



Classes

- struct [Noeud](#)
- struct [Code](#)

Fonctions

- double * [frequency](#) (FILE *input, int *nbrChar, int *nbrASCII)
- [Noeud](#) * [tableauHuffman](#) (double *freqAscii)
- [Code](#) * [saveHeader](#) (FILE *output, [Noeud](#) *[tableauHuffman](#), int nbrChar, int nbrASCII)
- [Noeud](#) * [getHeader](#) (FILE *input, int *nbrChar, int *nbrASCII)

4.3.1 Description détaillée

Auteur

Hareski

Version

1.1

Date

16 décembre 2017

4.3.2 Documentation des fonctions

4.3.2.1 frequency()

```
double* frequency (
    FILE * input,
    int * nbrChar,
    int * nbrASCII )
```

Fonction de calcul de la fréquences d'ASCII Etendu.

Paramètres

<i>input</i>	FILE* : fichier à analyser
<i>nbrChar</i>	int& : nombre de caractères dans input
<i>nbrASCII</i>	int& : nombre de caractères distinct dans input

Renvoie

void

4.3.2.2 getHeader()

```
Noeud* getHeader (
    FILE * input,
    int * nbrChar,
    int * nbrASCII )
```

Fonction de récupération sur fichier d'arbre Huffman.

Paramètres

<i>input</i>	FILE* : fichier de l'index
<i>nbrChar</i>	int* : nombre de caractères dans input
<i>nbrASCII</i> <i>CII</i>	int* : nombre de ASCII distinct

Renvoie

Noeud* : tableau d'Huffman

4.3.2.3 saveHeader()

```
Code* saveHeader (
    FILE * output,
    Noeud * tableauHuffman,
    int nbrChar,
    int nbrASCII )
```

Fonction de sauvegarde fichier et tableau d'arbre Huffman.

Paramètres

<i>output</i>	FILE* : fichier pour l'index
<i>tableauHuffman</i>	Noeud* : tableau de Huffman de l'index
<i>nbrChar</i>	int : nombre de caractères dans input
<i>nbrASCII</i>	int : nombre de ASCII distinct

Renvoie

Code* : tableau des codes ASCII selon l'arbre

4.3.2.4 tableauHuffman()

```
Noeud* tableauHuffman (
    double * freqAscii )
```

Fonction de génération d'arbre Huffman.

Paramètres

<i>tabASCII</i> <i>CII</i>	double* : frequences de chaque codage ASCII
-------------------------------	---

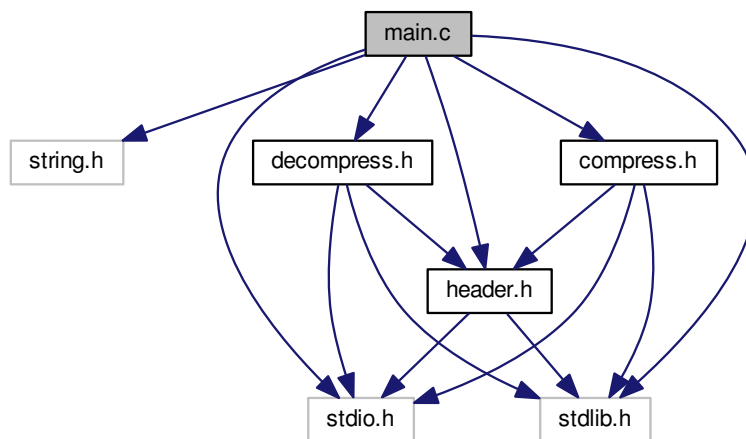
Renvoie

Noeud* : arbre de compression et décompression

4.4 Référence du fichier main.c

```
#include <string.h>
#include <stdio.h>
#include <stdlib.h>
#include "header.h"
#include "compress.h"
#include "decompress.h"
```

Graphe des dépendances par inclusion de main.c :



Fonctions

- void [help](#) (char *exeName)
- int [main](#) (int argc, char *argv[])

4.4.1 Description détaillée

Auteur

Hareski

Version

1.1

Date

16 décembre 2017

4.4.2 Documentation des fonctions

4.4.2.1 help()

```
void help (
    char * exeName )
```

Affiche l'aide.

Paramètres

<i>exeName</i>	char* : nom de l'exécutable
----------------	-----------------------------

Renvoie

void

4.4.2.2 main()

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

Entrée du programme.

Paramètres

<i>argc</i>	int : nombre de paramètres
<i>argv</i>	char** : paramètres

Renvoie

0[Arrêt normal du programme] ou int[Numéro erreur]

Index

- bit
 - Code, [5](#)
- Code, [5](#)
 - bit, [5](#)
- compress.h, [7](#)
 - compression, [8](#)
- compression
 - compress.h, [8](#)
- decompress.h, [9](#)
 - decompression, [10](#)
- decompression
 - decompress.h, [10](#)
- frequency
 - header.h, [12](#)
- getHeader
 - header.h, [12](#)
- header.h, [10](#)
 - frequency, [12](#)
 - getHeader, [12](#)
 - saveHeader, [13](#)
 - tableauHuffman, [13](#)
- help
 - main.c, [15](#)
- main
 - main.c, [15](#)
- main.c, [14](#)
 - help, [15](#)
 - main, [15](#)
- Noeud, [5](#)
- saveHeader
 - header.h, [13](#)
- tableauHuffman
 - header.h, [13](#)