

Super Martin

Generated by Doxygen 1.8.6

Sun Mar 16 2014 19:48:16

Contents

1	Data Structure Index	1
1.1	Data Structures	1
2	File Index	3
2.1	File List	3
3	Data Structure Documentation	5
3.1	Character Struct Reference	5
3.1.1	Field Documentation	5
3.1.1.1	isOnGround	5
3.2	Level Struct Reference	5
3.3	Map Struct Reference	6
3.4	Sound Struct Reference	6
4	File Documentation	7
4.1	const.h File Reference	7
4.1.1	Detailed Description	8
4.2	file.c File Reference	8
4.2.1	Detailed Description	9
4.2.2	Function Documentation	9
4.2.2.1	closeFile	9
4.2.2.2	openFile	9
4.2.2.3	readFileSize	9
4.3	file.h File Reference	10
4.3.1	Detailed Description	11
4.3.2	Function Documentation	11
4.3.2.1	closeFile	11
4.3.2.2	openFile	11
4.3.2.3	readFileSize	11
4.4	jeu.c File Reference	12
4.4.1	Detailed Description	12
4.4.2	Function Documentation	12

4.4.2.1	initMap	12
4.4.2.2	jouer	13
4.4.2.3	move	13
4.4.2.4	printGameOver	14
4.4.2.5	scrolling	14
4.5	jeu.h File Reference	15
4.5.1	Detailed Description	16
4.5.2	Function Documentation	16
4.5.2.1	initMap	16
4.5.2.2	jouer	16
4.5.2.3	move	17
4.5.2.4	printGameOver	17
4.5.2.5	scrolling	18
4.6	menu.c File Reference	18
4.6.1	Detailed Description	19
4.7	menu.h File Reference	19
4.7.1	Detailed Description	20
4.8	menu_level.c File Reference	20
4.8.1	Detailed Description	21
4.9	menu_level.h File Reference	21
4.9.1	Detailed Description	22
4.10	player.c File Reference	23
4.10.1	Detailed Description	23
4.10.2	Function Documentation	24
4.10.2.1	blitCharacter	24
4.10.2.2	collisionSprite	24
4.10.2.3	creatorCharacter	24
4.10.2.4	moveCharacter	24
4.10.2.5	movementVector	25
4.11	player.h File Reference	25
4.11.1	Detailed Description	27
4.11.2	Macro Definition Documentation	27
4.11.2.1	ABS	27
4.11.2.2	SGN	27
4.11.3	Function Documentation	27
4.11.3.1	blitCharacter	27
4.11.3.2	collisionSprite	27
4.11.3.3	creatorCharacter	28
4.11.3.4	moveCharacter	28
4.11.3.5	movementVector	28

4.12 sound.c File Reference	29
4.12.1 Detailed Description	29
4.12.2 Function Documentation	30
4.12.2.1 createSound	30
4.12.2.2 freeSound	30
4.12.2.3 playMusic	30
4.12.2.4 stopSound	30
4.13 sound.h File Reference	30
4.13.1 Detailed Description	31
4.13.2 Function Documentation	32
4.13.2.1 createSound	32
4.13.2.2 freeSound	32
4.13.2.3 playMusic	32
4.13.2.4 stopSound	32
4.14 text.c File Reference	32
4.14.1 Detailed Description	33
4.14.2 Function Documentation	33
4.14.2.1 printText	33
4.15 text.h File Reference	33
4.15.1 Detailed Description	34
4.15.2 Function Documentation	35
4.15.2.1 printText	35
Index	36

Chapter 1

Data Structure Index

1.1 Data Structures

Here are the data structures with brief descriptions:

Character	5
Level	5
Map	6
Sound	6

Chapter 2

File Index

2.1 File List

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

const.h	Contient les constantes du programme	7
file.c	Fonctions d'accès aux fichiers	8
file.h	Prototypes des fonctions d'accès aux fichiers	10
file_level.h	??
jeu.c	Contient les fonctions liées au jeu	12
jeu.h	Header de jeu.c	15
menu.c	Contient les fonctions liées au menu	18
menu.h	Header de menu.c	19
menu_level.c	Menu gérant le choix du niveau	20
menu_level.h	Menu gérant le choix du niveau	21
player.c	Contient les fonctions pour manipuler le joueur	23
player.h	Header de player.c	25
share.h	??
sound.c	Contient les fonctions pour jouer du son	29
sound.h	Header de sound.c	30
text.c	Contient les fonctions pour afficher du texte à l'écran	32
text.h	Header de text.c	33

Chapter 3

Data Structure Documentation

3.1 Character Struct Reference

Data Fields

- SDL_Surface * **spriteR**
- SDL_Surface * **spriteL**
- SDL_Rect **location**
- int **isRight**
- int [isOnGround](#)

3.1.1 Field Documentation

3.1.1.1 int isOnGround

indique la direction de regard du personnage (1 droite, 0 gauche)

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

- [player.h](#)

3.2 Level Struct Reference

Data Fields

- unsigned char ** **map**
- int **width**
- int **height**
- char **background** [TAILLE_MAX_NOM_FICHIER]

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

- [const.h](#)

3.3 Map Struct Reference

Collaboration diagram for Map:



Data Fields

- [Level](#) * **lvl**
- int **xScroll**
- int **screenWidth**
- int **screenHeight**

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

- [const.h](#)

3.4 Sound Struct Reference

Data Fields

- FMOD_SYSTEM * **sys**
- FMOD_SOUND * **sound**
- FMOD_CHANNELGROUP * **channel**

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

- [sound.h](#)

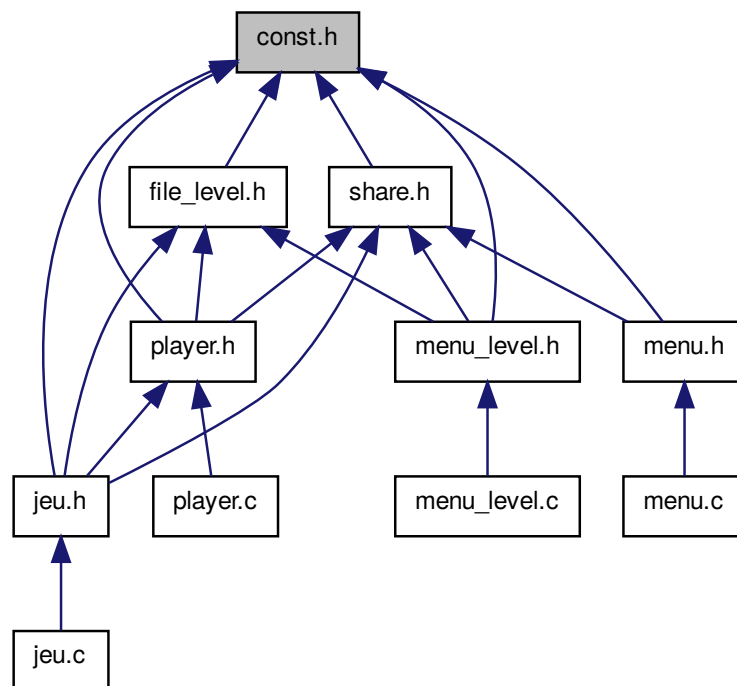
Chapter 4

File Documentation

4.1 const.h File Reference

contient les constantes du programme

This graph shows which files directly or indirectly include this file:



Data Structures

- struct [Level](#)
- struct [Map](#)

Macros

- `#define TAILLE_BLOC 20`
- `#define NB_BLOCS_LARGEUR 48`
- `#define NB_BLOCS_HAUTEUR 27`
- `#define LARGEUR_FENETRE TAILLE_BLOC * NB_BLOCS_LARGEUR`
- `#define HAUTEUR_FENETRE TAILLE_BLOC * NB_BLOCS_HAUTEUR`
- `#define FPS 60`
- `#define TAILLE_MAX_NOM_FICHIER 100`

Enumerations

- `enum { VOID =0, GRASS1 =1, GROUND1 =2, GREY_WALL =3 }`
- `enum { RIGHT, LEFT, UP, DOWN }`

4.1.1 Detailed Description

contient les constantes du programme

Author

Xavier COPONET

Date

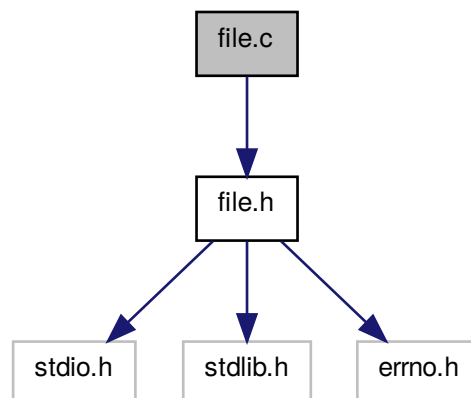
2014-02-27

4.2 file.c File Reference

Fonctions d'accès au fichiers.

```
#include "file.h"
```

Include dependency graph for file.c:



Functions

- FILE * [openFile](#) (char nom[], char mode[])
- int [closeFile](#) (FILE *ptr_fichier)
- int [readFileSize](#) (FILE *ptr_fichier)

4.2.1 Detailed Description

Fonctions d'accès aux fichiers.

Author

Remi BERTHO

Date

15/03/14

4.2.2 Function Documentation

4.2.2.1 int closeFile (FILE * ptr_fichier)

Ferme le fichier

Parameters

in	*ptr_fichier	le fichier
----	--------------	------------

Returns

entier 0 si tout s'est bien passé, 1 sinon

4.2.2.2 FILE * openFile (char nom[], char mode[])

Ouvre un fichier à partir de son nom (nom[]) et du mode voulu (mode[])

Parameters

in	nom[]	le nom du fichier
in	mode[]	le mode voulu

Returns

un pointeur sur le fichier ouvert, NULL s'il y a eu un problème

4.2.2.3 int readFileSize (FILE * ptr_fichier)

Lis la taille du fichier

Parameters

in	*ptr_fichier	le fichier
----	--------------	------------

Returns

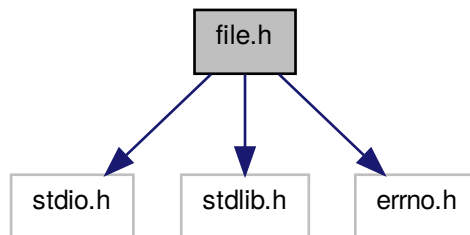
entier ayant la taille du fichier

4.3 file.h File Reference

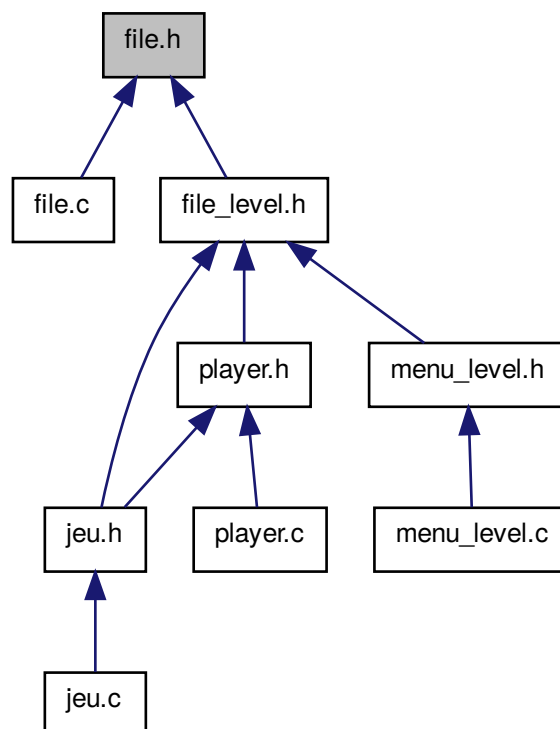
Prototypes des fonctions d'accès aux fichiers.

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

Include dependency graph for file.h:



This graph shows which files directly or indirectly include this file:



Functions

- FILE * [openFile](#) (char nome[], char mode[])
- int [closeFile](#) (FILE *ptr_fichier)
- int [readFileSize](#) (FILE *ptr_fichier)

4.3.1 Detailed Description

Prototypes des fonctions d'accès aux fichiers.

Author

Remi BERTHO

Date

15/03/14

4.3.2 Function Documentation

4.3.2.1 int closeFile (FILE * ptr_fichier)

Ferme le fichier

Parameters

in	*ptr_fichier	le fichier
----	--------------	------------

Returns

entier 0 si tout s'est bien passé, 1 sinon

4.3.2.2 FILE* openFile (char nom[], char mode[])

Ouvre un fichier à partir de son nom (nom[]) et du mode voulu (mode[])

Parameters

in	nom[]	le nom du fichier
in	mode[]	le mode voulu

Returns

un pointeur sur le fichier ouvert, NULL s'il y a eu un problème

4.3.2.3 int readFileSize (FILE * ptr_fichier)

Lis la taille du fichier

Parameters

in	*ptr_fichier	le fichier
----	--------------	------------

Returns

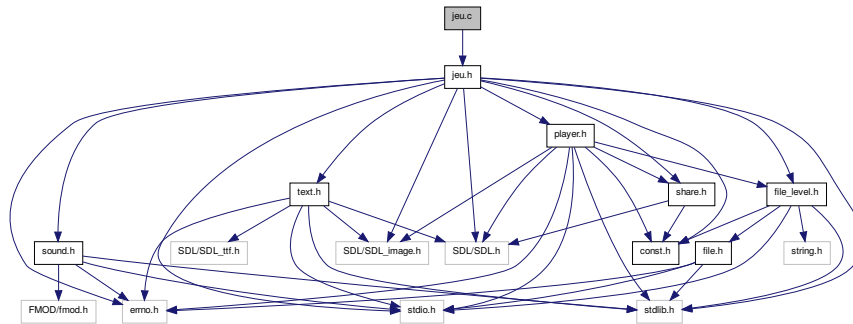
entier ayant la taille du fichier

4.4 jeu.c File Reference

contient les fonction liées au jeu

```
#include "jeu.h"
```

Include dependency graph for jeu.c:



Functions

- void [jouer](#) (SDL_Surface *screen, char *level_name)
- void **updateScreenMap** (SDL_Surface *screen, [Map](#) *m)
- void [scrolling](#) ([Map](#) *m, int direction, float speed)
- Uint32 **decomptage** (Uint32 intervalle, void *parametre)
- [Map](#) * [initMap](#) ([Level](#) *lvl, SDL_Surface *screen)
- void **freeMap** ([Map](#) *m)
- void [printGameOver](#) (SDL_Surface *screen, int *continuer)
- void [move](#) (int move_left, int move_right, [Character](#) *player, [Map](#) *m, float speed)

4.4.1 Detailed Description

contient les fonction liées au jeu

Author

Xavier COPONET

Date

2014-02-27

4.4.2 Function Documentation

4.4.2.1 [Map](#) * [initMap](#) ([Level](#) * *lvl*, SDL_Surface * *screen*)

initialise la carte

Parameters

in	<i>screen</i>	l'écran de jeu
in	<i>level</i>	le niveau

Returns

un pointeur sur la carte initialisée

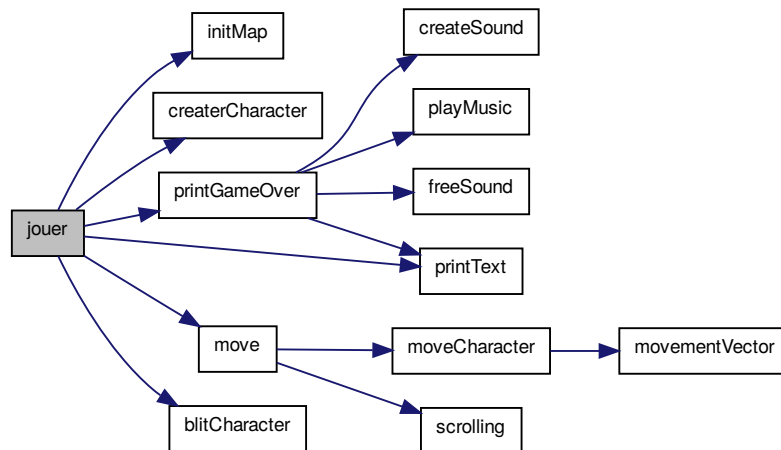
4.4.2.2 void jouer (SDL_Surface * screen, char * level_name)

contient la boucle principale du jeu qui appelle les fonctions

Parameters

in, out	<i>screen</i>	L'écran de jeu
in	<i>level_name</i>	le nom du niveau

Here is the call graph for this function:

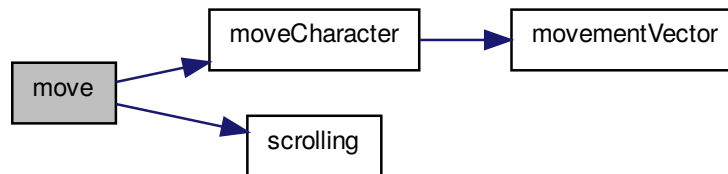
**4.4.2.3 void move (int move_left, int move_right, Character * player, Map * m, float speed)**

Deplace le joueur et scrolle l'écran si besoin

Parameters

in	<i>move_left</i>	booleen pour savoir si l'on bouge a gauche
in	<i>move_right</i>	booleen pour savoir si l'on bouge a droite
in	<i>player</i>	le joueur
in	<i>m</i>	la carte

Here is the call graph for this function:



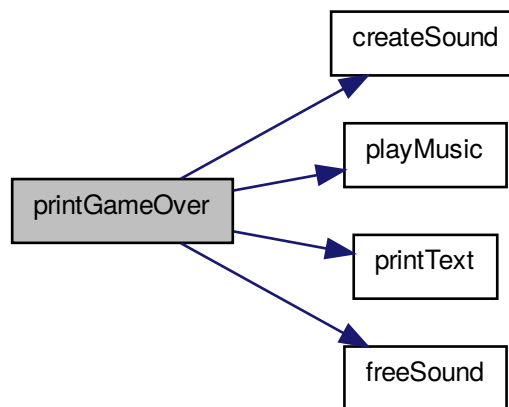
4.4.2.4 void printGameOver (SDL_Surface * *screen*, int * *continuer*)

affiche le message de game overflow_error

Parameters

out	<i>screen</i>	l'écran de jeu
-----	---------------	----------------

Here is the call graph for this function:



4.4.2.5 void scrolling (Map * *m*, int *direction*, float *speed*)

effectue un scrolling

Parameters

Functions

- void [jouer](#) (SDL_Surface *screen, char *level_name)
- void **updateScreenMap** (SDL_Surface *screen, [Map](#) *m)
- void [scrolling](#) ([Map](#) *m, int direction, float speed)
- Uint32 **decomptage** (Uint32 intervalle, void *parametre)
- [Map](#) * **initMap** ([Level](#) *lvl, SDL_Surface *screen)
- void **freeMap** ([Map](#) *m)
- void [printGameOver](#) (SDL_Surface *screen, int *continuer)
- void [move](#) (int move_left, int move_right, [Character](#) *player, [Map](#) *m, float speed)

4.5.1 Detailed Description

header de [jeu.c](#)

Author

Xavier COPONET

Date

2014-02-27

4.5.2 Function Documentation

4.5.2.1 **Map*** initMap ([Level](#) * *lvl*, SDL_Surface * *screen*)

initialise la carte

Parameters

<i>in</i>	<i>screen</i>	l'écran de jeu
<i>in</i>	<i>level</i>	le niveau

Returns

un pointeur sur la carte initialisée

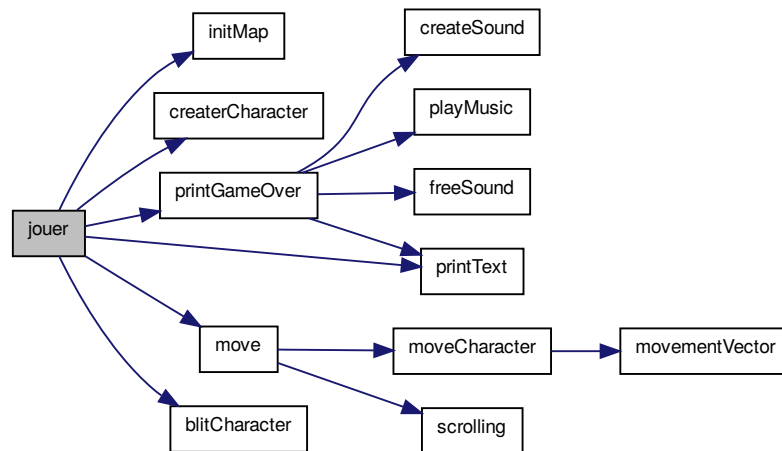
4.5.2.2 void **jouer** (SDL_Surface * *screen*, char * *level_name*)

contient la boucle principale du jeu qui appelle les fonctions

Parameters

<i>in, out</i>	<i>screen</i>	L'écran de jeu
<i>in</i>	<i>lvl_name</i>	le nom du niveau

Here is the call graph for this function:



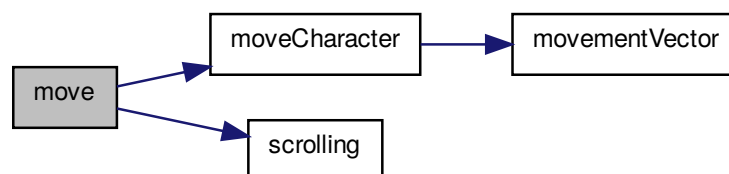
4.5.2.3 void move (int *move_left*, int *move_right*, **Character** * *player*, **Map** * *m*, float *speed*)

Deplace le joueur et scrolle l'ecran si besoin

Parameters

in	<i>move_left</i>	booleen pour savoir si l'on bouge a gauche
in	<i>move_right</i>	booleen pour savoir si l'on bouge a droite
in	<i>player</i>	le joueur
in	<i>m</i>	la carte

Here is the call graph for this function:



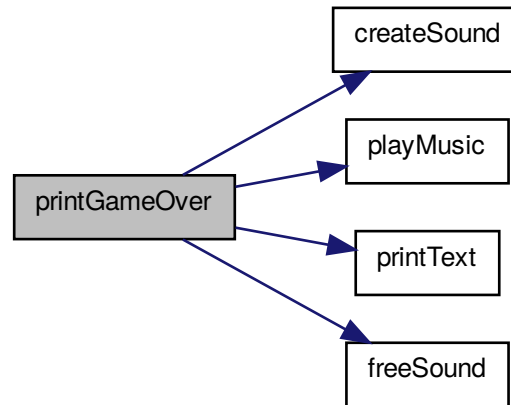
4.5.2.4 void printGameOver (**SDL_Surface** * *screen*, int * *continuer*)

affiche le message de game overflow_error

Parameters

out	screen	l'écran de jeu
-----	--------	----------------

Here is the call graph for this function:



4.5.2.5 void scrolling (Map * m, int direction, float speed)

effectue un scrolling

Parameters

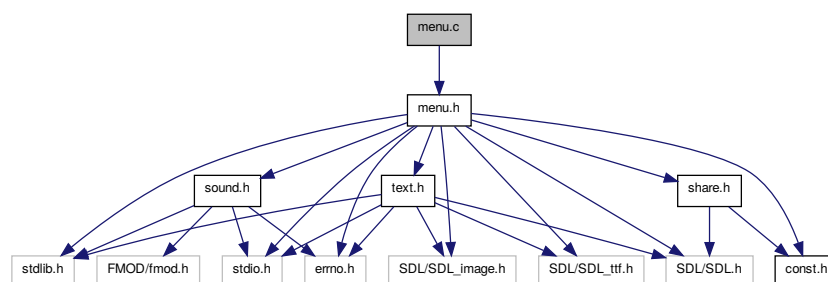
in, out	map	Le niveau à gérer
in	direction	La direction de scrolling
in	speed	la vitesse de scrolling

4.6 menu.c File Reference

contient les fonction liées au menu

```
#include "menu.h"
```

Include dependency graph for menu.c:



Functions

- int **menu** (SDL_Surface *screen, int *continuer, [Sound](#) *s)
- Uint32 **blinkText** (Uint32 intervalle, void *parametre)

4.6.1 Detailed Description

contient les fonction liées au menu

Author

Xavier COPONET

Date

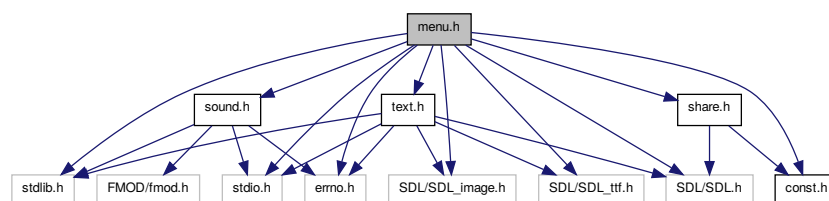
2014-02-27

4.7 menu.h File Reference

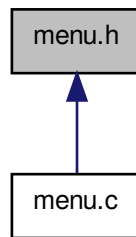
header de [menu.c](#)

```
#include <stdlib.h>
#include <stdio.h>
#include <errno.h>
#include <SDL/SDL.h>
#include <SDL/SDL_image.h>
#include <SDL/SDL_ttf.h>
#include "const.h"
#include "text.h"
#include "sound.h"
#include "share.h"
```

Include dependency graph for menu.h:



This graph shows which files directly or indirectly include this file:



Functions

- int **menu** (SDL_Surface *screen, int *continuer, [Sound](#) *s)
- Uint32 **blinkText** (Uint32 intervalle, void *parametre)

4.7.1 Detailed Description

header de [menu.c](#)

Author

Xavier COPONET

Date

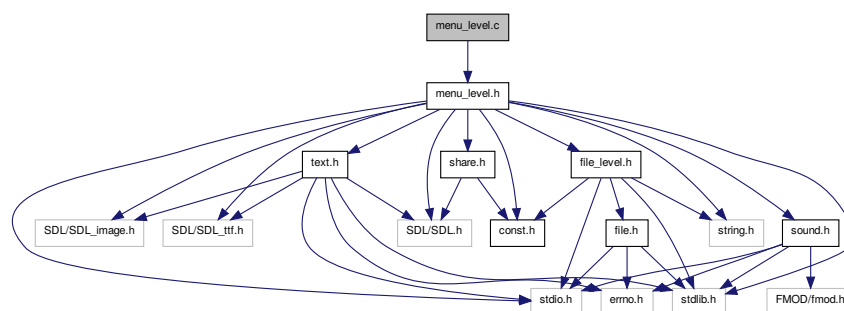
2014-02-27

4.8 menu_level.c File Reference

Menu gerant le choix du niveau.

```
#include "menu_level.h"
```

Include dependency graph for menu_level.c:



Functions

- int **menuLevel** (SDL_Surface *screen, char level_name[TAILLE_MAX_NOM_FICHIER], [Sound](#) *s)

4.8.1 Detailed Description

Menu gerant le choix du niveau.

Author

Remi BERTHO

Date

15/03/14

Version

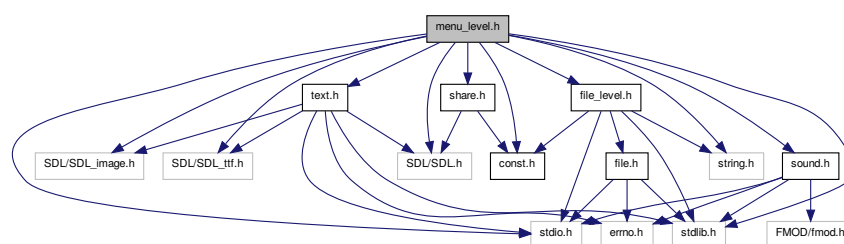
1.0

4.9 menu_level.h File Reference

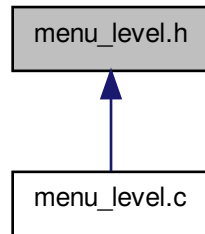
Menu gerant le choix du niveau.

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <SDL/SDL.h>
#include <SDL/SDL_image.h>
#include <SDL/SDL_ttf.h>
#include "const.h"
#include "file_level.h"
#include "share.h"
#include "text.h"
#include "sound.h"
```

Include dependency graph for menu_level.h:



This graph shows which files directly or indirectly include this file:



Functions

- int **menuLevel** (SDL_Surface *screen, char level_name[TAILLE_MAX_NOM_FICHIER], [Sound](#) *s)

4.9.1 Detailed Description

Menu gerant le choix du niveau.

Author

Remi BERTHO

Date

15/03/14

Version

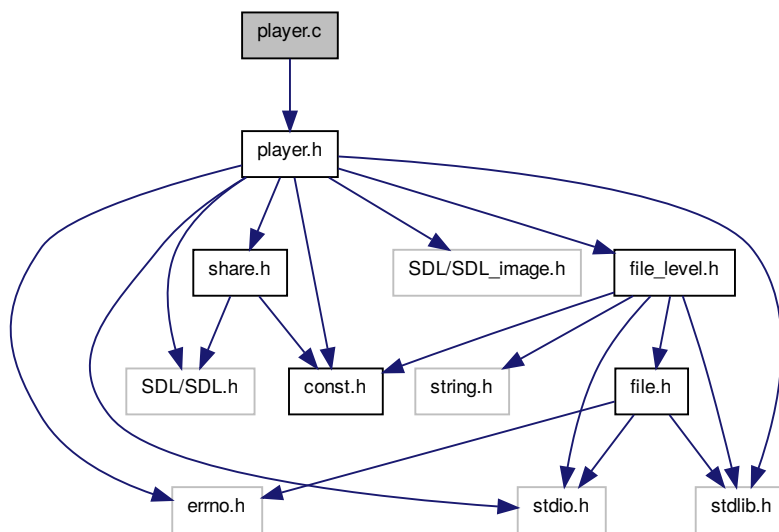
1.0

4.10 player.c File Reference

contient les fonction pour manipuler le joueur

```
#include "player.h"
```

Include dependency graph for player.c:



Functions

- [Character](#) * [creatorCharacter](#) (char *spR, char *spL)
- int [moveCharacter](#) ([Character](#) *c, int direction, [Map](#) *m, float speed)
- int [tryMovement](#) ([Character](#) *c, int vx, int vy, [Map](#) *m)
- void [movementVector](#) (int direction, int *vx, int *vy, int speed, [Character](#) *c)
- void [blitCharacter](#) (SDL_Surface *screen, [Character](#) *c, [Map](#) *m)
- int [collisionSprite](#) (SDL_Rect r, [Map](#) *m)
- void [gravity](#) ([Character](#) *c, [Map](#) *m, SDL_Surface *screen)
- void [presiseMoveCharacter](#) ([Character](#) *c, int vx, int vy, [Map](#) *m)

4.10.1 Detailed Description

contient les fonction pour manipuler le joueur

Author

Xavier COPONET

Date

2014-02-27

4.10.2 Function Documentation

4.10.2.1 void blitCharacter (SDL_Surface * *screen*, Character * *c*, Map * *m*)

blit le personnage à l'écran

Parameters

in, out	<i>screen</i>	L'écran
in	<i>c</i>	Le personnage
in	<i>m</i>	la carte du jeu

4.10.2.2 int collisionSprite (SDL_Rect *r*, Map * *m*)

détermine s'il y a collision entre une sprite et le décor

Parameters

in	<i>r</i>	le SDL_Rect correspondant à la sprite
in	<i>m</i>	la carte contenant le décor

Returns

1 s'il y a collision ou si en dehors du monde, 0 sinon

4.10.2.3 Character * creatorCharacter (char * *spR*, char * *spL*)

créer un personnage

Parameters

in	<i>spR</i>	l'adresse de la sprite droite
in	<i>spL</i>	l'adresse de la sprite gauche

Returns

le pointeur sur la structure créée

4.10.2.4 void moveCharacter (Character * *c*, int *direction*, Map * *m*, float *speed*)

déplace le personnage selon la direction

Parameters

in, out	<i>c</i>	Le personnage
in	<i>direction</i>	La direction du déplacement
in	<i>m</i>	la carte sur laquelle le personnage se déplace
in	<i>speed</i>	la vitesse de déplacement

Returns

1 si le personnage a pu se déplacer normalement, 0 s'il a fallut affiner

Here is the call graph for this function:



4.10.2.5 void movementVector (int *direction*, int * *vx*, int * *vy*, int *speed*, Character * *c*)

create a movement vector

Parameters

in	<i>direction</i>	The movement's direction
out	<i>vx</i>	the horizontal component of the vector
out	<i>vy</i>	the vertical component of the vector
in	<i>speed</i>	the speed of the move
out	<i>c</i>	the Character you have to move

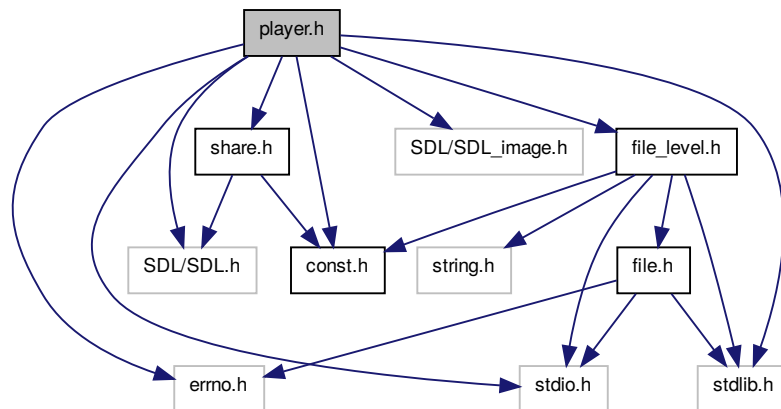
4.11 player.h File Reference

header de [player.c](#)

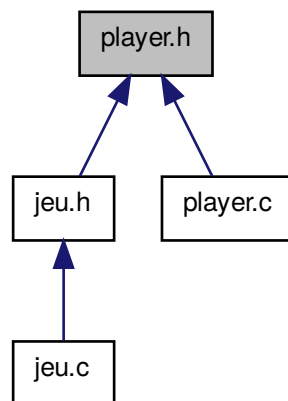
```

#include "const.h"
#include <stdlib.h>
#include <stdio.h>
#include <errno.h>
#include <SDL/SDL.h>
#include <SDL/SDL_image.h>
#include "file_level.h"
#include "share.h"
  
```

Include dependency graph for player.h:



This graph shows which files directly or indirectly include this file:



Data Structures

- struct [Character](#)

Macros

- #define [SGN](#)(X) (((X)==0)?(0):(((X)<0)?(-1):(1)))
- #define [ABS](#)(X) (((X)<0)?(-(X)):(X))

Functions

- [Character](#) * [creatorCharacter](#) (char *spR, char *spL)
- int [moveCharacter](#) ([Character](#) *c, int direction, [Map](#) *m, float speed)
- void [blitCharacter](#) (SDL_Surface *screen, [Character](#) *c, [Map](#) *m)
- int [collisionSprite](#) (SDL_Rect r, [Map](#) *m)
- void [gravity](#) ([Character](#) *c, [Map](#) *m, SDL_Surface *screen)
- void [movementVector](#) (int direction, int *vx, int *vy, int speed, [Character](#) *c)
- int [tryMovement](#) ([Character](#) *c, int vx, int vy, [Map](#) *m)
- void [presiseMoveCharacter](#) ([Character](#) *c, int vx, int vy, [Map](#) *m)

4.11.1 Detailed Description

header de [player.c](#)

Author

Xavier COPONET

Date

2014-02-27

4.11.2 Macro Definition Documentation

4.11.2.1 #define ABS(X) (((X)<0)?(-(X)):(X))

X absolute value

4.11.2.2 #define SGN(X) (((X)==0)?(0):(((X)<0)?(-1):(1)))

X sign

4.11.3 Function Documentation

4.11.3.1 void blitCharacter (SDL_Surface * screen, Character * c, Map * m)

blit le personnage à l'écran

Parameters

in, out	<i>screen</i>	L'écran
in	<i>c</i>	Le personnage
in	<i>m</i>	la carte du jeu

4.11.3.2 int collisionSprite (SDL_Rect r, Map * m)

détermine s'il y a collision entre une sprite et le décor

Parameters

in	<i>r</i>	le SDL_Rect correspondant à la sprite
in	<i>m</i>	la carte contenant le décor

Returns

1 s'il y a collision ou si en dehors du monde, 0 sinon

4.11.3.3 Character* creatorCharacter (char * *spR*, char * *spL*)

créer un personnage

Parameters

in	<i>spR</i>	l'adresse de la sprite droite
in	<i>spL</i>	l'adresse de la sprite gauche

Returns

le pointeur sur la structure créée

4.11.3.4 int moveCharacter (Character * *c*, int *direction*, Map * *m*, float *speed*)

déplace le personnage selon la direction

Parameters

in, out	<i>c</i>	Le personnage
in	<i>direction</i>	La direction du déplacement
in	<i>m</i>	la carte sur laquelle le personnage se déplace
in	<i>speed</i>	la vitesse de déplacement

Returns

1 si le personnage a pu se déplacer normalement, 0 s'il a fallut affiner

Here is the call graph for this function:

**4.11.3.5 void movementVector (int *direction*, int * *vx*, int * *vy*, int *speed*, Character * *c*)**

create a movement vector

Parameters

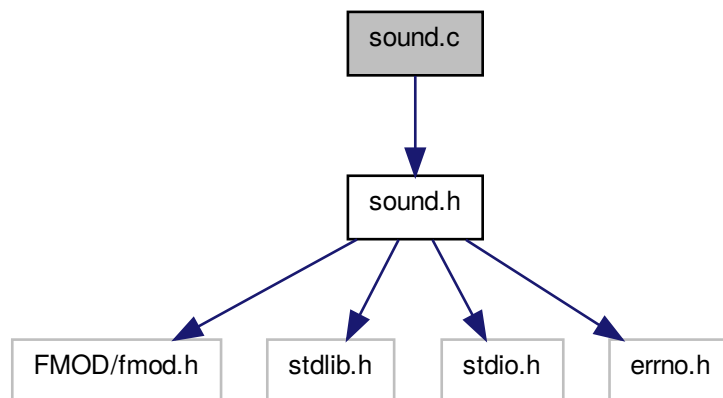
in	<i>direction</i>	The movement's direction
out	<i>vx</i>	the horizontal component of the vector
out	<i>vy</i>	the vertical component of the vector
in	<i>speed</i>	the speed of the move
out	<i>c</i>	the Character you have to move

4.12 sound.c File Reference

contient les fonction pour jouer du son

```
#include "sound.h"
```

Include dependency graph for sound.c:



Functions

- [Sound *](#) [createSound](#) (void)
- void [playMusic](#) ([Sound *](#)s, char *file)
- void [freeSound](#) ([Sound *](#)s)
- void [stopSound](#) ([Sound *](#)s)

4.12.1 Detailed Description

contient les fonction pour jouer du son

Author

Xavier COPONET

Date

2014-02-27

4.12.2 Function Documentation

4.12.2.1 `sound * createSound (void)`

créer une structure son

Returns

la structure son

4.12.2.2 `void freeSound (Sound * s)`

release the sound

Parameters

out	s	the sound
-----	---	-----------

4.12.2.3 `void playMusic (Sound * s, char * file)`

lit un fichier long (musique)

Parameters

in, out	s	la structure son que l'on manipule
in	file	Le fichier son à lire

4.12.2.4 `void stopSound (Sound * s)`

stop the sound

Parameters

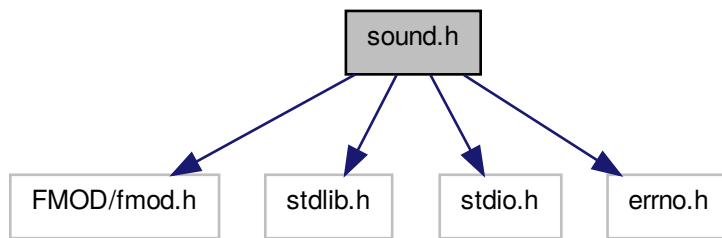
out	the	sound to stop
-----	-----	---------------

4.13 `sound.h` File Reference

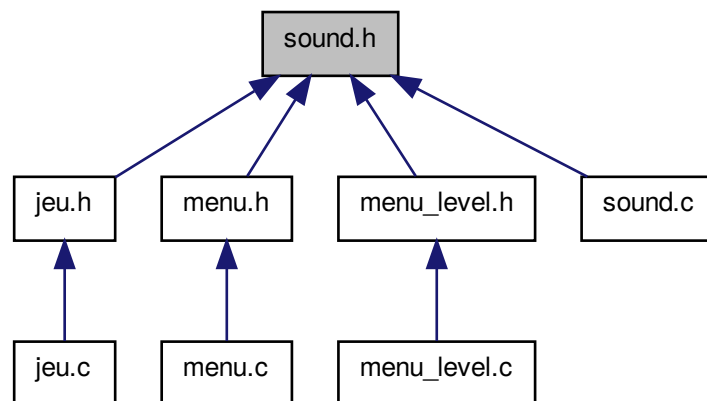
header de [sound.c](#)

```
#include <FMOD/fmod.h>
#include <stdlib.h>
#include <stdio.h>
#include <errno.h>
```

Include dependency graph for sound.h:



This graph shows which files directly or indirectly include this file:



Data Structures

- struct [Sound](#)

Functions

- [Sound](#) * [createSound](#) (void)
- void [playMusic](#) ([Sound](#) *s, char *file)
- void [freeSound](#) ([Sound](#) *s)
- void [stopSound](#) ([Sound](#) *s)

4.13.1 Detailed Description

header de [sound.c](#)

Author

Xavier COPONET

Date

2014-02-27

4.13.2 Function Documentation**4.13.2.1 Sound* createSound (void)**

créer une structure son

Returns

la structure son

4.13.2.2 void freeSound (Sound * s)

release the sound

Parameters

out	s	the sound
-----	---	-----------

4.13.2.3 void playMusic (Sound * s, char * file)

lit un fichier long (musique)

Parameters

in, out	s	la structure son que l'on manipule
in	file	Le fichier son à lire

4.13.2.4 void stopSound (Sound * s)

stop the sound

Parameters

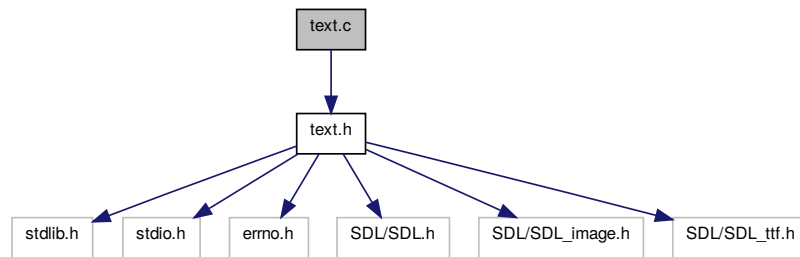
out	the	sound to stop
-----	-----	---------------

4.14 text.c File Reference

contient les fonction pour afficher du texte à l'écran

```
#include "text.h"
```

Include dependency graph for text.c:



Functions

- void [printText](#) (SDL_Surface *screen, SDL_Rect *posText, char *text, SDL_Color color, char *font, int ptSize, int mode)

4.14.1 Detailed Description

contient les fonction pour afficher du texte à l'écran

Author

Xavier COPONET

Date

2014-02-27

4.14.2 Function Documentation

4.14.2.1 void [printText](#) (SDL_Surface * *screen*, SDL_Rect * *posText*, char * *text*, SDL_Color *color*, char * *font*, int *ptSize*, int *mode*)

affiche le texte sur l'écran à la position donnée

Parameters

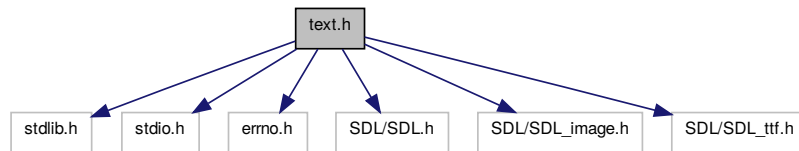
out	<i>screen</i>	L'écran
in	<i>posText</i>	La position du texte à afficher ; si NULL, centré en largeur et hauteur
in	<i>text</i>	Le texte à afficher
in	<i>color</i>	la couleur du texte
in	<i>font</i>	L'adresse de la police d'affichage (.ttf)
in	<i>ptSize</i>	la taille du texte à afficher
in	<i>mode</i>	Le mode d'écriture : 0 (Solid), 1 (Blended)

4.15 text.h File Reference

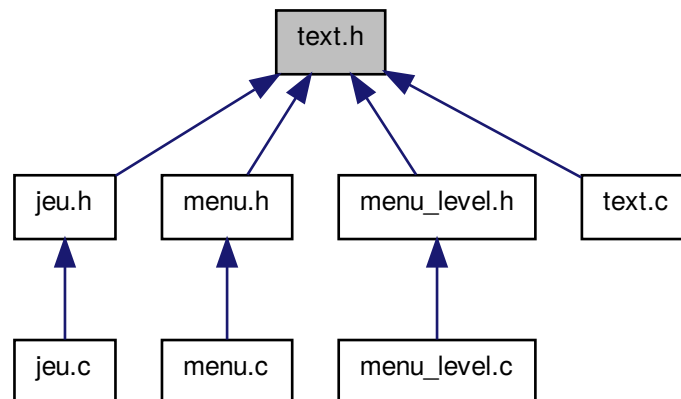
header de [text.c](#)

```
#include <stdlib.h>
#include <stdio.h>
#include <errno.h>
#include <SDL/SDL.h>
#include <SDL/SDL_image.h>
#include <SDL/SDL_ttf.h>
```

Include dependency graph for text.h:



This graph shows which files directly or indirectly include this file:



Functions

- void [printText](#) (SDL_Surface *screen, SDL_Rect *posText, char *text, SDL_Color color, char *font, int ptSize, int mode)

4.15.1 Detailed Description

header de [text.c](#)

Author

Xavier COPONET

Date

2014-02-27

4.15.2 Function Documentation

4.15.2.1 void printText (SDL_Surface * *screen*, SDL_Rect * *posText*, char * *text*, SDL_Color *color*, char * *font*, int *ptSize*, int *mode*)

affiche le texte sur l'écran à la position donnée

Parameters

out	<i>screen</i>	L'écran
in	<i>posText</i>	La position du texte à afficher ; si NULL, centré en largeur et hauteur
in	<i>text</i>	Le texte à afficher
in	<i>color</i>	la couleur du texte
in	<i>font</i>	L'adresse de la police d'affichage (.ttf)
in	<i>ptSize</i>	la taille du texte à afficher
in	<i>mode</i>	Le mode d'écriture : 0 (Solid), 1 (Blended)

Index

- ABS
 - player.h, 27
- blitCharacter
 - player.c, 24
 - player.h, 27
- Character, 5
 - isOnGround, 5
- closeFile
 - file.c, 9
 - file.h, 11
- collisionSprite
 - player.c, 24
 - player.h, 27
- const.h, 7
- createSound
 - sound.c, 30
 - sound.h, 32
- creatorCharacter
 - player.c, 24
 - player.h, 28
- file.c, 8
 - closeFile, 9
 - openFile, 9
 - readFileSize, 9
- file.h, 10
 - closeFile, 11
 - openFile, 11
 - readFileSize, 11
- freeSound
 - sound.c, 30
 - sound.h, 32
- initMap
 - jeu.c, 12
 - jeu.h, 16
- isOnGround
 - Character, 5
- jeu.c, 12
 - initMap, 12
 - jouer, 13
 - move, 13
 - printGameOver, 14
 - scrolling, 14
- jeu.h, 15
 - initMap, 16
 - jouer, 16
 - move, 17
 - printGameOver, 17
 - scrolling, 18
- jouer
 - jeu.c, 13
 - jeu.h, 16
- Level, 5
- Map, 6
 - menu.c, 18
 - menu.h, 19
 - menu_level.c, 20
 - menu_level.h, 21
- move
 - jeu.c, 13
 - jeu.h, 17
- moveCharacter
 - player.c, 24
 - player.h, 28
- movementVector
 - player.c, 25
 - player.h, 28
- openFile
 - file.c, 9
 - file.h, 11
- playMusic
 - sound.c, 30
 - sound.h, 32
- player.c, 23
 - blitCharacter, 24
 - collisionSprite, 24
 - creatorCharacter, 24
 - moveCharacter, 24
 - movementVector, 25
- player.h, 25
 - ABS, 27
 - blitCharacter, 27
 - collisionSprite, 27
 - creatorCharacter, 28
 - moveCharacter, 28
 - movementVector, 28
 - SGN, 27
- printGameOver
 - jeu.c, 14
 - jeu.h, 17
- printText
 - text.c, 33
 - text.h, 35

readFileSize
 file.c, [9](#)
 file.h, [11](#)

SGN
 player.h, [27](#)

scrolling
 jeu.c, [14](#)
 jeu.h, [18](#)

Sound, [6](#)

sound.c, [29](#)
 createSound, [30](#)
 freeSound, [30](#)
 playMusic, [30](#)
 stopSound, [30](#)

sound.h, [30](#)
 createSound, [32](#)
 freeSound, [32](#)
 playMusic, [32](#)
 stopSound, [32](#)

stopSound
 sound.c, [30](#)
 sound.h, [32](#)

text.c, [32](#)
 printText, [33](#)

text.h, [33](#)
 printText, [35](#)