

Csuper - Compteur de Score Universel Permettant l'Exemption de Reflexion  
2.1.0

Generated by Doxygen 1.8.6

Fri Mar 21 2014 19:37:28



# Contents

<b>1</b>	<b>Data Structure Index</b>	<b>1</b>
1.1	Data Structures . . . . .	1
<b>2</b>	<b>File Index</b>	<b>3</b>
2.1	File List . . . . .	3
<b>3</b>	<b>Data Structure Documentation</b>	<b>5</b>
3.1	Fichier_Jeu Struct Reference . . . . .	5
3.1.1	Detailed Description . . . . .	5
3.1.2	Field Documentation . . . . .	5
3.1.2.1	annee . . . . .	5
3.1.2.2	distribue . . . . .	5
3.1.2.3	jour . . . . .	6
3.1.2.4	mois . . . . .	6
3.1.2.5	nb_joueur . . . . .	6
3.1.2.6	nb_max . . . . .	6
3.1.2.7	nb_tour . . . . .	6
3.1.2.8	nom_joueur . . . . .	6
3.1.2.9	point . . . . .	6
3.1.2.10	point_tot . . . . .	6
3.1.2.11	position . . . . .	6
3.1.2.12	sens_premier . . . . .	6
3.1.2.13	taille_max_nom . . . . .	6
3.1.2.14	version . . . . .	6
<b>4</b>	<b>File Documentation</b>	<b>9</b>
4.1	affichage.c File Reference . . . . .	9
4.1.1	Detailed Description . . . . .	10
4.1.2	Function Documentation . . . . .	10
4.1.2.1	afficherChaineTroisTab . . . . .	10
4.1.2.2	afficherDistribue . . . . .	10
4.1.2.3	afficherEnTete . . . . .	11

4.1.2.4	<a href="#">afficherLicense</a>	11
4.1.2.5	<a href="#">afficherLigne</a>	12
4.1.2.6	<a href="#">afficherNom</a>	12
4.1.2.7	<a href="#">afficherPartieFinie</a>	13
4.1.2.8	<a href="#">afficherPosition</a>	13
4.1.2.9	<a href="#">afficherScore</a>	14
4.1.2.10	<a href="#">afficherScoreEntier</a>	15
4.1.2.11	<a href="#">afficherScoreTotal</a>	15
4.1.2.12	<a href="#">afficherStruct</a>	16
4.2	<a href="#">affichage.h File Reference</a>	17
4.2.1	<a href="#">Detailed Description</a>	18
4.2.2	<a href="#">Function Documentation</a>	19
4.2.2.1	<a href="#">afficherChaineTroisTab</a>	19
4.2.2.2	<a href="#">afficherDistribue</a>	19
4.2.2.3	<a href="#">afficherEnTete</a>	19
4.2.2.4	<a href="#">afficherLicense</a>	20
4.2.2.5	<a href="#">afficherLigne</a>	20
4.2.2.6	<a href="#">afficherNom</a>	21
4.2.2.7	<a href="#">afficherPartieFinie</a>	21
4.2.2.8	<a href="#">afficherPosition</a>	22
4.2.2.9	<a href="#">afficherScore</a>	22
4.2.2.10	<a href="#">afficherScoreEntier</a>	23
4.2.2.11	<a href="#">afficherScoreTotal</a>	24
4.2.2.12	<a href="#">afficherStruct</a>	24
4.3	<a href="#">argument_main.c File Reference</a>	25
4.3.1	<a href="#">Detailed Description</a>	26
4.3.2	<a href="#">Function Documentation</a>	26
4.3.2.1	<a href="#">searchArgument</a>	26
4.4	<a href="#">argument_main.h File Reference</a>	27
4.4.1	<a href="#">Detailed Description</a>	28
4.4.2	<a href="#">Macro Definition Documentation</a>	28
4.4.2.1	<a href="#">CHAINE_LECTURE_FICHER</a>	28
4.4.2.2	<a href="#">CHAINE_LECTURE_FICHER_RED</a>	29
4.4.2.3	<a href="#">CHAINE_OUVERTURE_FICHER</a>	29
4.4.2.4	<a href="#">CHAINE_OUVERTURE_FICHER_RED</a>	29
4.4.2.5	<a href="#">LECTURE_FICHER</a>	29
4.4.2.6	<a href="#">OUVERTURE_FICHER</a>	29
4.4.3	<a href="#">Function Documentation</a>	29
4.4.3.1	<a href="#">searchArgument</a>	29
4.5	<a href="#">emplacement_fichier.c File Reference</a>	29

4.5.1	Detailed Description	30
4.5.2	Function Documentation	31
4.5.2.1	changerCheminFichier	31
4.5.2.2	creationPreferences	31
4.5.2.3	lectureCheminFichier	32
4.5.2.4	lecturePreferences	33
4.6	emplacement_fichier.h File Reference	34
4.6.1	Detailed Description	36
4.6.2	Macro Definition Documentation	37
4.6.2.1	NOM_DOSSIER	37
4.6.2.2	NOM_FICHIER	37
4.6.3	Function Documentation	37
4.6.3.1	changerCheminFichier	37
4.6.3.2	creationPreferences	37
4.6.3.3	lectureCheminFichier	38
4.6.3.4	lecturePreferences	39
4.7	fichier.c File Reference	40
4.7.1	Detailed Description	41
4.7.2	Function Documentation	42
4.7.2.1	ecrireFichier	42
4.7.2.2	lireFichier	42
4.7.2.3	nouveauScore	43
4.7.2.4	ouvrirFichierExtension	44
4.7.2.5	renommerFichier	45
4.7.2.6	supprimerFichier	45
4.8	fichier.h File Reference	46
4.8.1	Detailed Description	47
4.8.2	Macro Definition Documentation	48
4.8.2.1	EXTENSION_FICHIER	48
4.8.2.2	TAILLE_MAX_NOM_FICHIER	48
4.8.2.3	TYPE_FICHIER	48
4.8.3	Function Documentation	48
4.8.3.1	ecrireFichier	48
4.8.3.2	lireFichier	49
4.8.3.3	nouveauScore	50
4.8.3.4	ouvrirFichierExtension	51
4.8.3.5	renommerFichier	51
4.8.3.6	supprimerFichier	52
4.9	fonction.c File Reference	52
4.9.1	Detailed Description	53

4.9.2	Function Documentation	54
4.9.2.1	compareFlottantCroissant	54
4.9.2.2	compareFlottantDecroissant	55
4.9.2.3	fermerFichier	55
4.9.2.4	lireTailleFichier	56
4.9.2.5	mauvais_choix	56
4.9.2.6	ouvrirFichier	57
4.9.2.7	systemEfface	57
4.9.2.8	systemPause	58
4.10	fonction.h File Reference	59
4.10.1	Detailed Description	60
4.10.2	Macro Definition Documentation	61
4.10.2.1	FAUX	61
4.10.2.2	VRAI	61
4.10.3	Function Documentation	61
4.10.3.1	compareFlottantCroissant	61
4.10.3.2	compareFlottantDecroissant	61
4.10.3.3	fermerFichier	61
4.10.3.4	lireTailleFichier	62
4.10.3.5	mauvais_choix	63
4.10.3.6	ouvrirFichier	63
4.10.3.7	systemEfface	63
4.10.3.8	systemPause	64
4.11	interface.c File Reference	65
4.11.1	Detailed Description	66
4.11.2	Function Documentation	67
4.11.2.1	afficheFichier	67
4.11.2.2	afficheFichierLocale	68
4.11.2.3	chargerPartie	69
4.11.2.4	chargerPartieLocale	70
4.11.2.5	jouer	71
4.11.2.6	lireCheminFichier	72
4.11.2.7	listerFichier	72
4.11.2.8	menuPrincipal	73
4.11.2.9	nouveauCheminFichier	74
4.11.2.10	nouvellePartie	75
4.11.2.11	supprimerFichierNom	76
4.12	interface.h File Reference	77
4.12.1	Detailed Description	79
4.12.2	Enumeration Type Documentation	79

4.12.2.1	Menu	79
4.12.3	Function Documentation	79
4.12.3.1	afficheFichier	79
4.12.3.2	afficheFichierLocale	80
4.12.3.3	chargerPartie	81
4.12.3.4	chargerPartieLocale	82
4.12.3.5	jouer	83
4.12.3.6	lireCheminFichier	84
4.12.3.7	listerFichier	85
4.12.3.8	menuPrincipal	86
4.12.3.9	nouveauCheminFichier	87
4.12.3.10	nouvellePartie	87
4.12.3.11	supprimerFichierNom	88
4.13	main.c File Reference	89
4.13.1	Detailed Description	90
4.13.2	Function Documentation	90
4.13.2.1	main	90
4.14	main.h File Reference	92
4.14.1	Detailed Description	93
4.14.2	Function Documentation	93
4.14.2.1	main	93
4.15	menu.c File Reference	94
4.15.1	Detailed Description	95
4.15.2	Function Documentation	96
4.15.2.1	menuContinuer	96
4.15.2.2	menuDebutPartie	96
4.15.2.3	menuDistribue	97
4.15.2.4	menuNomFichier	97
4.15.2.5	menuNomJoueur	98
4.15.2.6	menuNouveauChemin	99
4.15.2.7	menuPointsJoueur	99
4.15.2.8	menuSupprimer	100
4.16	menu.h File Reference	101
4.16.1	Detailed Description	102
4.16.2	Function Documentation	103
4.16.2.1	menuContinuer	103
4.16.2.2	menuDebutPartie	103
4.16.2.3	menuDistribue	104
4.16.2.4	menuNomFichier	104
4.16.2.5	menuNomJoueur	105

4.16.2.6	menuNouveauChemin . . . . .	106
4.16.2.7	menuPointsJoueur . . . . .	106
4.16.2.8	menuSupprimer . . . . .	107
4.17	saisie_clavier.c File Reference . . . . .	108
4.17.1	Detailed Description . . . . .	109
4.17.2	Function Documentation . . . . .	109
4.17.2.1	clean_stdin . . . . .	109
4.17.2.2	saisieClavierCaractere . . . . .	109
4.17.2.3	saisieClavierChaine . . . . .	110
4.17.2.4	saisieClavierDouble . . . . .	111
4.17.2.5	saisieClavierEntier . . . . .	111
4.17.2.6	saisieClavierFlottant . . . . .	112
4.17.2.7	saisieClavierFlottantSansVirgule . . . . .	113
4.18	saisie_clavier.h File Reference . . . . .	114
4.18.1	Detailed Description . . . . .	115
4.18.2	Macro Definition Documentation . . . . .	116
4.18.2.1	NB_CARACT_DOUB . . . . .	116
4.18.2.2	NB_CARACT_FLOT . . . . .	116
4.18.2.3	NB_CARACT_INT . . . . .	116
4.18.3	Function Documentation . . . . .	116
4.18.3.1	clean_stdin . . . . .	116
4.18.3.2	saisieClavierCaractere . . . . .	116
4.18.3.3	saisieClavierChaine . . . . .	117
4.18.3.4	saisieClavierDouble . . . . .	118
4.18.3.5	saisieClavierEntier . . . . .	118
4.18.3.6	saisieClavierFlottant . . . . .	119
4.18.3.7	saisieClavierFlottantSansVirgule . . . . .	120
4.19	structure.c File Reference . . . . .	121
4.19.1	Detailed Description . . . . .	122
4.19.2	Function Documentation . . . . .	123
4.19.2.1	ajoutDistribueStruct . . . . .	123
4.19.2.2	calculPosition . . . . .	124
4.19.2.3	creerFichierStruct . . . . .	124
4.19.2.4	debNouvTour . . . . .	125
4.19.2.5	depScoreMax . . . . .	125
4.19.2.6	fermeeFichierStruct . . . . .	126
4.19.2.7	finNouvTour . . . . .	126
4.20	structure.h File Reference . . . . .	127
4.20.1	Detailed Description . . . . .	128
4.20.2	Macro Definition Documentation . . . . .	129



---

4.20.2.1	TAILLE_MAX_NOM . . . . .	129
4.20.2.2	VERSION . . . . .	129
4.20.3	Function Documentation . . . . .	129
4.20.3.1	ajoutDistribueStruct . . . . .	129
4.20.3.2	calculPosition . . . . .	129
4.20.3.3	creerFichierStruct . . . . .	130
4.20.3.4	debNouvTour . . . . .	130
4.20.3.5	depScoreMax . . . . .	131
4.20.3.6	fermeeFichierStruct . . . . .	131
4.20.3.7	finNouvTour . . . . .	132
 <b>Index</b>		 <b>133</b>



# Chapter 1

## Data Structure Index

### 1.1 Data Structures

Here are the data structures with brief descriptions:

<a href="#">Fichier_Jeu</a> . . . . .	5
---------------------------------------	---



## Chapter 2

# File Index

### 2.1 File List

Here is a list of all files with brief descriptions:

<a href="#">affichage.c</a>	Fonctions qui gerent l'affichage des scores du logiciel . . . . .	9
<a href="#">affichage.h</a>	Prototypes des fonctions qui gerent l'affichage des scores du logiciel . . . . .	17
<a href="#">argument_main.c</a>	Lancement du programme . . . . .	25
<a href="#">argument_main.h</a>	Lancement du programme . . . . .	27
<a href="#">emplacement_fichier.c</a>	Fonctions qui l'emplacement des fichiers sauvegardes . . . . .	29
<a href="#">emplacement_fichier.h</a>	Prototypes des fonctions qui l'emplacement des fichiers sauvegardes . . . . .	34
<a href="#">fichier.c</a>	Fonction de gestion des fichiers . . . . .	40
<a href="#">fichier.h</a>	Prototypes des fonction de gestion des fichiers . . . . .	46
<a href="#">fonction.c</a>	Fonctions essentielles au fonctionnement du programme . . . . .	52
<a href="#">fonction.h</a>	Prototypes des fonctions essentielles au fonctionnement du programme . . . . .	59
<a href="#">interface.c</a>	Fonctions qui gerent l'interface graphique du logiciel . . . . .	65
<a href="#">interface.h</a>	Prototypes des fonctions qui gerent l'interface graphique du logiciel . . . . .	77
<a href="#">main.c</a>	Lancement du programme . . . . .	89
<a href="#">main.h</a>	Lancement du programme . . . . .	92
<a href="#">menu.c</a>	Fonctions qui gerent les menus demandant de rentrer des valeurs du logiciel . . . . .	94
<a href="#">menu.h</a>	Prototypes des fonctions qui gerent les menus demandant de rentrer des valeurs du logiciel . . . . .	101
<a href="#">saisie_clavier.c</a>	Fonctions de saisie clavier . . . . .	108
<a href="#">saisie_clavier.h</a>	Prototypes des fonctions de saisie clavier . . . . .	114
<a href="#">structure.c</a>	Fonction de gestion des fichiers de la struction contenant les informations . . . . .	121

[structure.h](#)

Prototypes des fonction de gestion des fichiers de la struction contenant les informations . . . [127](#)

## Chapter 3

# Data Structure Documentation

### 3.1 Fichier\_Jeu Struct Reference

```
#include <structure.h>
```

#### Data Fields

- float [version](#)
- float [taille\\_max\\_nom](#)
- float [jour](#)
- float [mois](#)
- float [annee](#)
- float [nb\\_joueur](#)
- float [nb\\_max](#)
- char [sens\\_premier](#)
- char \*\* [nom\\_joueur](#)
- float \* [point\\_tot](#)
- float \* [position](#)
- float [nb\\_tour](#)
- float [distribue](#)
- float \* [point](#)

#### 3.1.1 Detailed Description

Type representant un fichier .jeu

#### 3.1.2 Field Documentation

##### 3.1.2.1 float annee

Annee de creation de la structure.

##### 3.1.2.2 float distribue

Numero de la personne qui doit distribuer.

### 3.1.2.3 float jour

Jour de creation de la structure.

### 3.1.2.4 float mois

Mois de creation de la structure.

### 3.1.2.5 float nb\_joueur

Nombre de joueurs.

### 3.1.2.6 float nb\_max

Nombre maximum que peut prendre un joueur.

### 3.1.2.7 float nb\_tour

Nombre de tour dans le jeu.

### 3.1.2.8 char\*\* nom\_joueur

Tableau contenant tout les noms de joueurs.

### 3.1.2.9 float\* point

Tableau contenant les points de chaque joueur a chaque tour.

### 3.1.2.10 float\* point\_tot

Tableau contenant tout les points totaux des joueurs.

### 3.1.2.11 float\* position

Tableau contenant la position des joueurs.

### 3.1.2.12 char sens\_premier

Vaut 1 si le premier est celui qui a le plus de points, -1 sinon

### 3.1.2.13 float taille\_max\_nom

Taille maximum que peut prendre un nom de joueur.

### 3.1.2.14 float version

Version de la structure.

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



- [structure.h](#)



## Chapter 4

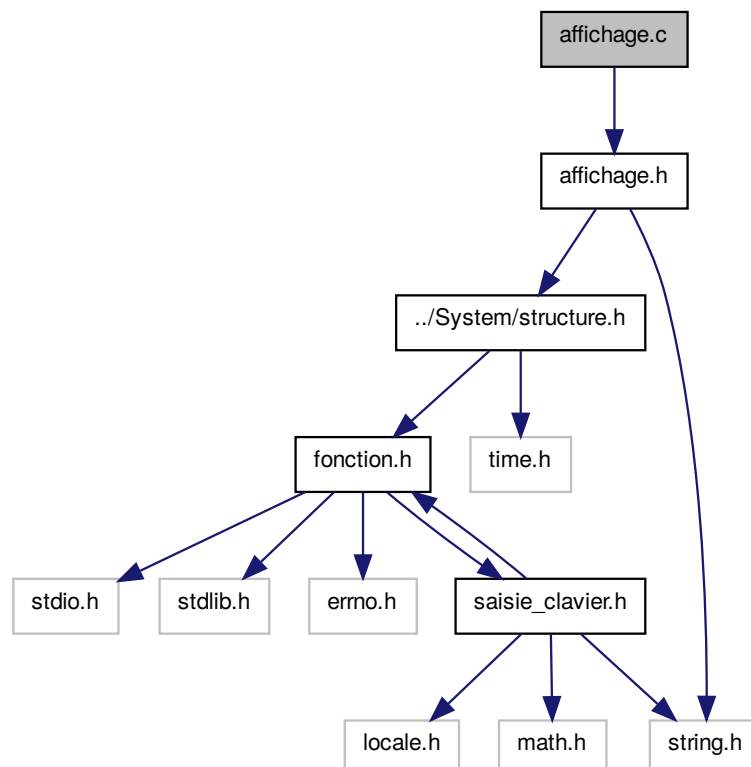
# File Documentation

### 4.1 affichage.c File Reference

Fonctions qui gerent l'affichage des scores du logiciel.

```
#include "affichage.h"
```

Include dependency graph for affichage.c:



## Functions

- void [afficherNom](#) ([Fichier\\_Jeu](#) \*ptr\_struct\_fichier, int \*ptr\_taille\_ligne)
- void [afficherLigne](#) (int taille\_ligne)
- void [afficherScoreTotal](#) ([Fichier\\_Jeu](#) \*ptr\_struct\_fichier)
- void [afficherDistribue](#) ([Fichier\\_Jeu](#) \*ptr\_struct\_fichier)
- void [afficherEnTete](#) ([Fichier\\_Jeu](#) \*ptr\_struct\_fichier)
- void [afficherScoreEntier](#) ([Fichier\\_Jeu](#) \*ptr\_struct\_fichier)
- void [afficherPosition](#) ([Fichier\\_Jeu](#) \*ptr\_struct\_fichier)
- void [afficherScore](#) ([Fichier\\_Jeu](#) \*ptr\_struct\_fichier)
- void [afficherStruct](#) ([Fichier\\_Jeu](#) \*ptr\_struct\_fichier)
- void [afficherPartieFinie](#) ([Fichier\\_Jeu](#) \*ptr\_struct\_fichier)
- void [afficherChaineTroisTab](#) (char \*chaine)
- void [afficherLicense](#) ()

### 4.1.1 Detailed Description

Fonctions qui gerent l'affichage des scores du logiciel.

#### Author

Remi BERTHO

#### Date

13/02/14

#### Version

2.0

### 4.1.2 Function Documentation

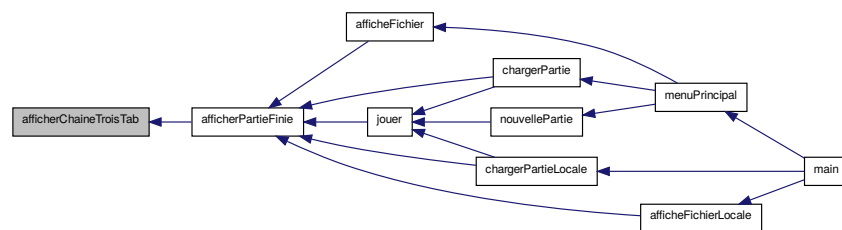
#### 4.1.2.1 void [afficherChaineTroisTab](#) ( char \* *chaine* )

Affiche la chaine de caractere passe en parametre centrer sur un espace de trois tabulations

#### Parameters

in	* <i>chaine</i>	une chaine de caractere
----	-----------------	-------------------------

Here is the caller graph for this function:



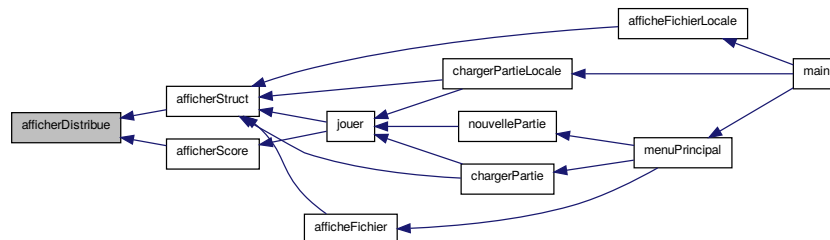
#### 4.1.2.2 void [afficherDistribue](#) ( [Fichier\\_Jeu](#) \* *ptr\_struct\_fichier* )

Affiche la personne devant distribuer

## Parameters

in	<i>*ptr_struct_fichier</i>	un pointeur sur la structure <a href="#">Fichier_Jeu</a>
----	----------------------------	--

Here is the caller graph for this function:



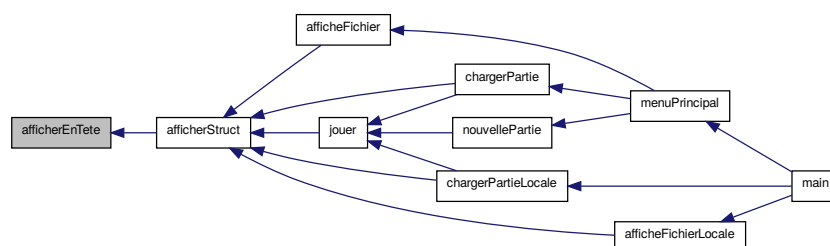
## 4.1.2.3 void afficherEnTete ( Fichier\_Jeu \* ptr\_struct\_fichier )

Affiche l'en tete de la structure

## Parameters

in	<i>*ptr_struct_fichier</i>	un pointeur sur la structure <a href="#">Fichier_Jeu</a>
----	----------------------------	--

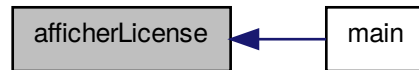
Here is the caller graph for this function:



## 4.1.2.4 void afficherLicense ( )

Affiche la license

Here is the caller graph for this function:



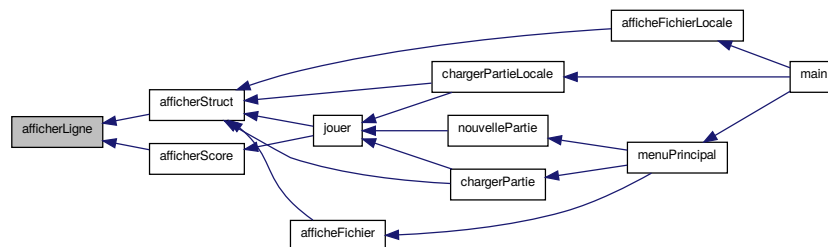
#### 4.1.2.5 void afficherLigne ( int *taille\_ligne* )

Affiche *taille\_ligne* - apres une tabulation

##### Parameters

in	<i>taille_ligne</i>	la taille de la ligne
----	---------------------	-----------------------

Here is the caller graph for this function:



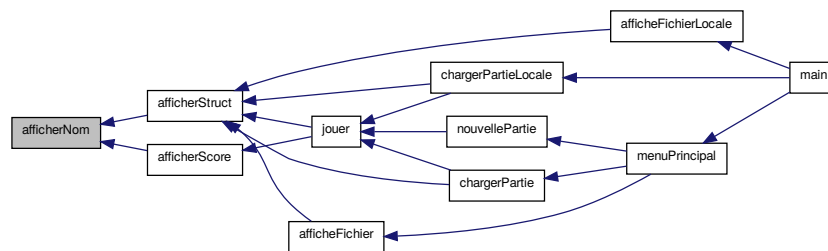
#### 4.1.2.6 void afficherNom ( Fichier\_Jeu \* *ptr\_struct\_fichier*, int \* *ptr\_taille\_ligne* )

Affiche les noms des joueurs et calcule la taille de la ligne

##### Parameters

in	* <i>ptr_struct_fichier</i>	un pointeur sur la structure <a href="#">Fichier_Jeu</a>
in, out	* <i>ptr_taille_ligne</i>	un pointeur sur la taille de la ligne a modifier

Here is the caller graph for this function:



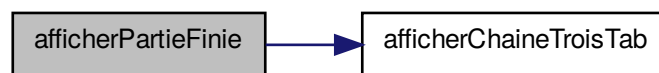
#### 4.1.2.7 void afficherPartieFinie ( Fichier\_Jeu \* ptr\_struct\_fichier )

Affiche un podium des resultats

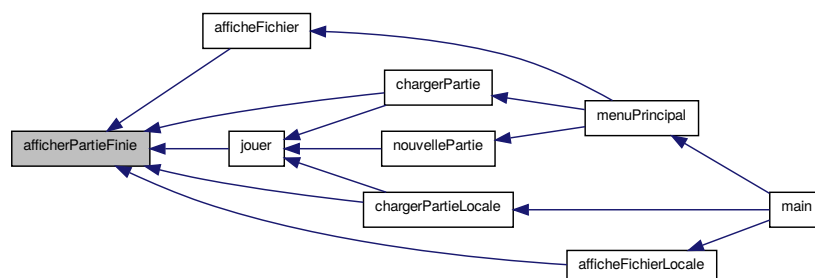
##### Parameters

in	*ptr_struct_fichier	un pointeur sur la structure <a href="#">Fichier_Jeu</a>
----	---------------------	--

Here is the call graph for this function:



Here is the caller graph for this function:



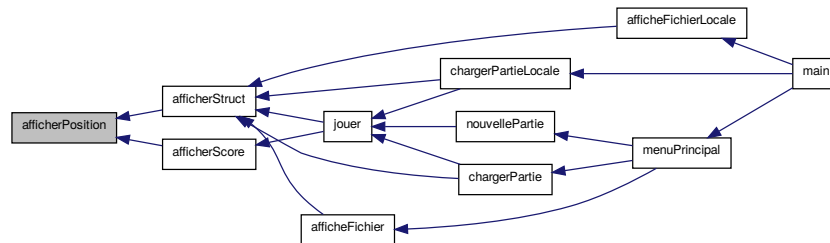
#### 4.1.2.8 void afficherPosition ( Fichier\_Jeu \* ptr\_struct\_fichier )

Affiche les positions des joueurs

## Parameters

in	<i>*ptr_struct_fichier</i>	un pointeur sur la structure <a href="#">Fichier_Jeu</a>
----	----------------------------	--

Here is the caller graph for this function:



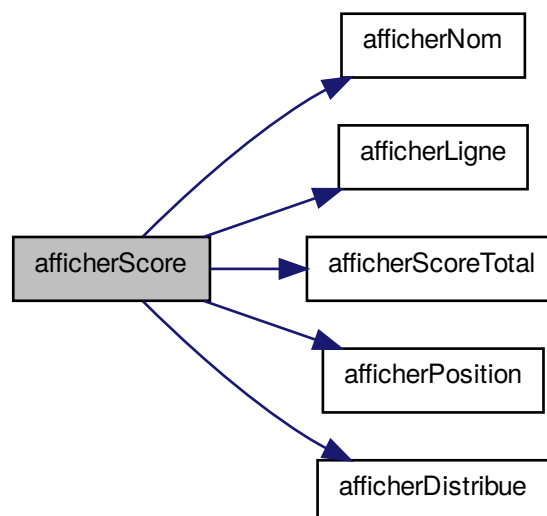
#### 4.1.2.9 void afficherScore ( Fichier\_Jeu \* ptr\_struct\_fichier )

Affiche les noms des personnes avec leurs scores totaux ainsi que la personne devant distribuer a partir d'une structure [Fichier\\_Jeu](#)

## Parameters

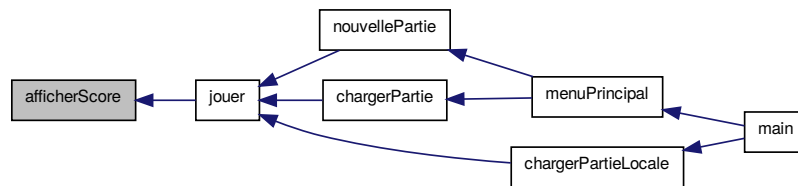
in	<i>*ptr_struct_fichier</i>	un pointeur sur la structure <a href="#">Fichier_Jeu</a>
----	----------------------------	--

Here is the call graph for this function:





Here is the caller graph for this function:



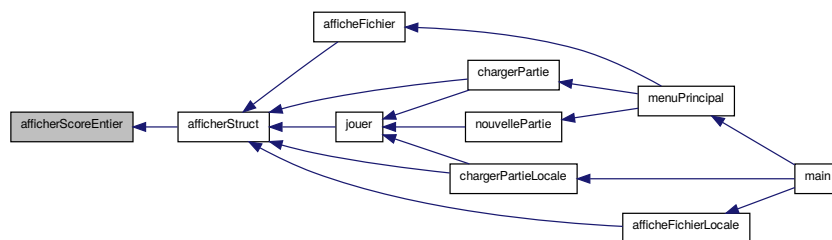
#### 4.1.2.10 void afficherScoreEntier ( Fichier\_Jeu \* ptr\_struct\_fichier )

Affiche tout les scores de tous les joueurs a chaque tour.

##### Parameters

in	<i>*ptr_struct_fichier</i>	un pointeur sur la structure <a href="#">Fichier_Jeu</a>
----	----------------------------	--

Here is the caller graph for this function:



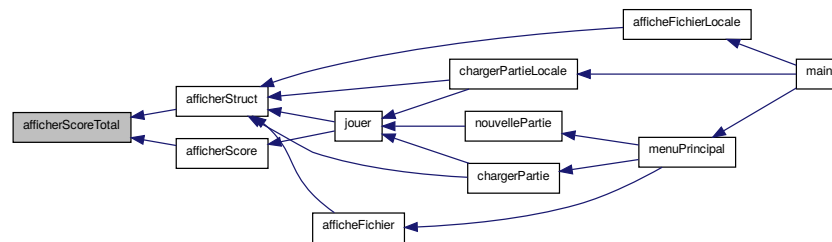
#### 4.1.2.11 void afficherScoreTotal ( Fichier\_Jeu \* ptr\_struct\_fichier )

Affiche le score total des joueurs

##### Parameters

in	<i>*ptr_struct_fichier</i>	un pointeur sur la structure <a href="#">Fichier_Jeu</a>
----	----------------------------	--

Here is the caller graph for this function:



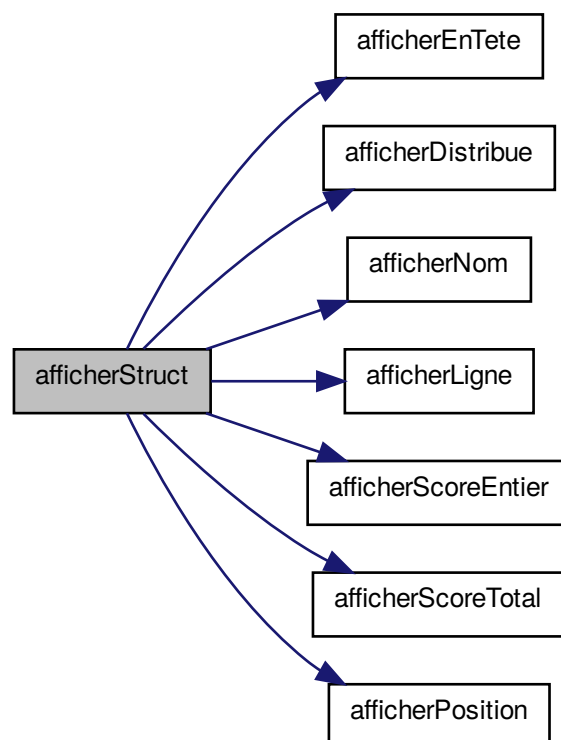
#### 4.1.2.12 void afficherStruct ( Fichier\_Jeu \* ptr\_struct\_fichier )

Affiche toutes les donnes d'une structure [Fichier\\_Jeu](#)

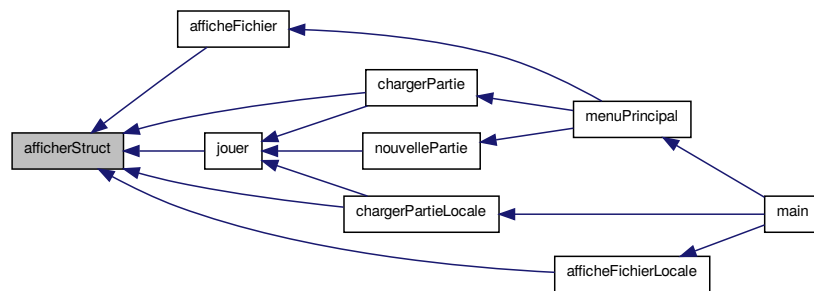
Parameters

in	*ptr_struct_fichier	un pointeur sur la structure <a href="#">Fichier_Jeu</a>
----	---------------------	--

Here is the call graph for this function:



Here is the caller graph for this function:



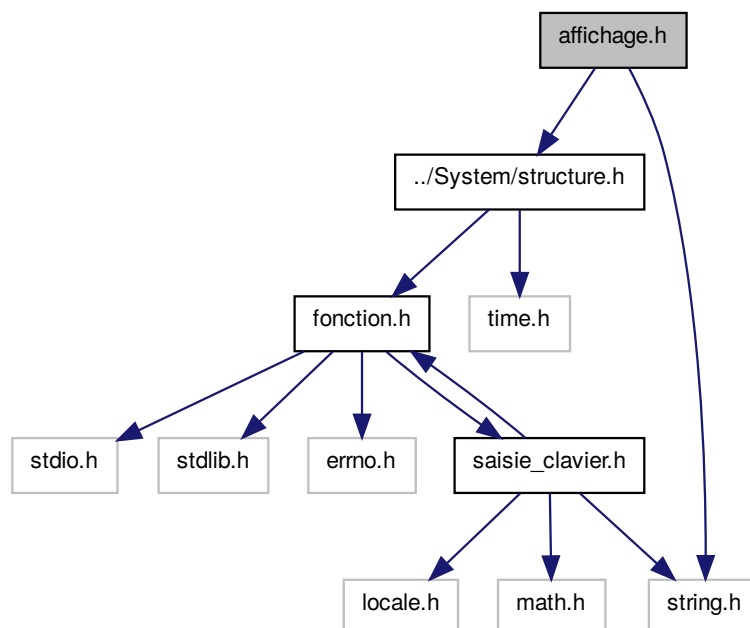
## 4.2 affichage.h File Reference

Prototypes des fonctions qui gerent l'affichage des scores du logiciel.

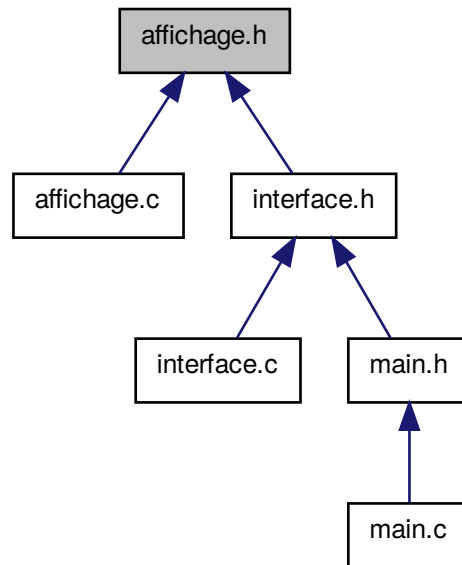
```
#include "../System/structure.h"
```

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

Include dependency graph for `affichage.h`:



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



## Functions

- void [afficherNom](#) ([Fichier\\_Jeu](#) \*ptr\_struct\_fichier, int \*ptr\_taille\_ligne)
- void [afficherLigne](#) (int taille\_ligne)
- void [afficherScoreTotal](#) ([Fichier\\_Jeu](#) \*ptr\_struct\_fichier)
- void [afficherDistribue](#) ([Fichier\\_Jeu](#) \*ptr\_struct\_fichier)
- void [afficherEnTete](#) ([Fichier\\_Jeu](#) \*ptr\_struct\_fichier)
- void [afficherScoreEntier](#) ([Fichier\\_Jeu](#) \*ptr\_struct\_fichier)
- void [afficherPosition](#) ([Fichier\\_Jeu](#) \*ptr\_struct\_fichier)
- void [afficherScore](#) ([Fichier\\_Jeu](#) \*ptr\_struct\_fichier)
- void [afficherStruct](#) ([Fichier\\_Jeu](#) \*ptr\_struct\_fichier)
- void [afficherPartieFinie](#) ([Fichier\\_Jeu](#) \*ptr\_struct\_fichier)
- void [afficherChaineTroisTab](#) (char \*chaine)
- void [afficherLicense](#) ()

### 4.2.1 Detailed Description

Prototypes des fonctions qui gerent l'affichage des scores du logiciel.

Author

Remi BERTHO

Date

13/02/14

Version

2.0

## 4.2.2 Function Documentation

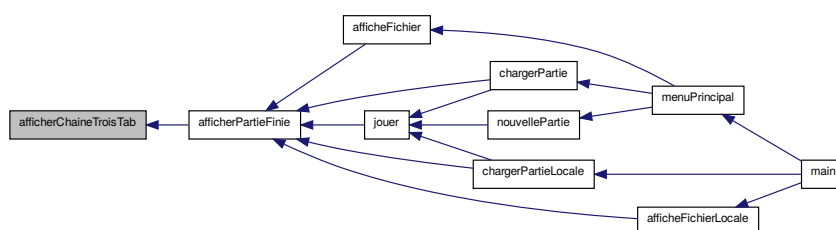
### 4.2.2.1 void afficherChaineTroisTab ( char \* *chaine* )

Affiche la chaine de caractere passe en parametre centrer sur un espace de trois tabulations

Parameters

in	* <i>chaine</i>	une chaine de caractere
----	-----------------	-------------------------

Here is the caller graph for this function:



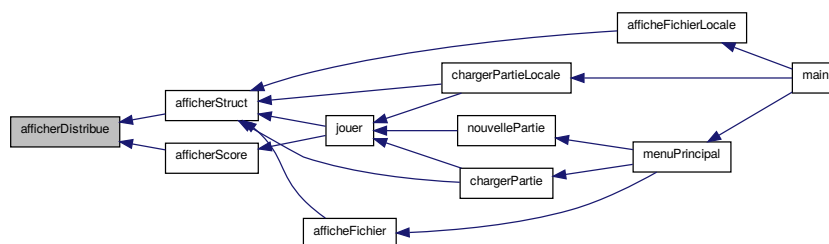
### 4.2.2.2 void afficherDistribue ( Fichier\_Jeu \* *ptr\_struct\_fichier* )

Affiche la personne devant distribuer

Parameters

in	* <i>ptr_struct_fichier</i>	un pointeur sur la structure <a href="#">Fichier_Jeu</a>
----	-----------------------------	--

Here is the caller graph for this function:



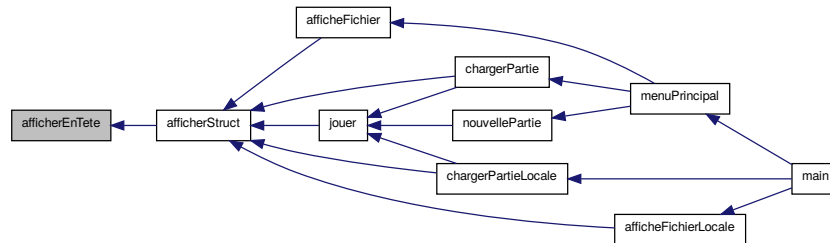
### 4.2.2.3 void afficherEnTete ( Fichier\_Jeu \* *ptr\_struct\_fichier* )

Affiche l'en tete de la structure

## Parameters

in	<i>*ptr_struct_ - fichier</i>	un pointeur sur la structure <a href="#">Fichier_Jeu</a>
----	-----------------------------------	--

Here is the caller graph for this function:



## 4.2.2.4 void afficherLicense ( )

Affiche la license

Here is the caller graph for this function:



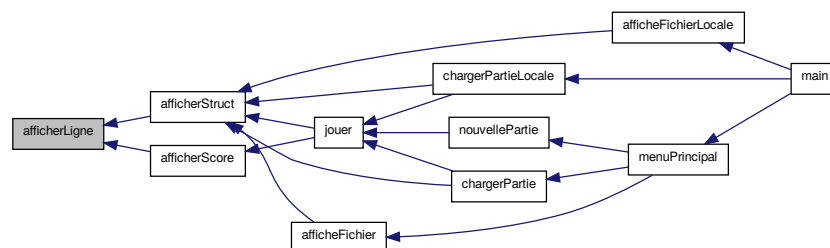
## 4.2.2.5 void afficherLigne ( int taille\_ligne )

Affiche taille\_ligne - apres une tabulation

## Parameters

in	<i>taille_ligne</i>	la taille de la ligne
----	---------------------	-----------------------

Here is the caller graph for this function:



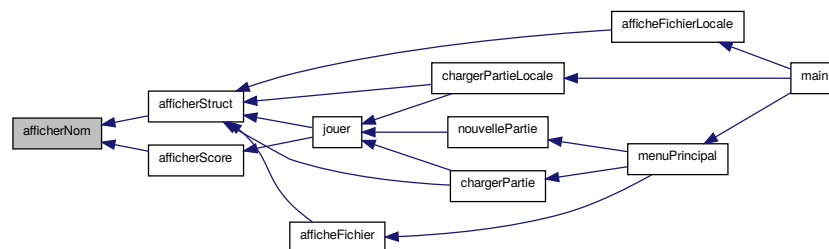
#### 4.2.2.6 void afficherNom ( Fichier\_Jeu \* ptr\_struct\_fichier, int \* ptr\_taille\_ligne )

Affiche les noms des joueurs et calcule la taille de la ligne

##### Parameters

in	*ptr_struct_fichier	un pointeur sur la structure <a href="#">Fichier_Jeu</a>
in, out	*ptr_taille_ligne	un pointeur sur la taille de la ligne a modifier

Here is the caller graph for this function:



#### 4.2.2.7 void afficherPartieFinie ( Fichier\_Jeu \* ptr\_struct\_fichier )

Affiche un podium des resultats

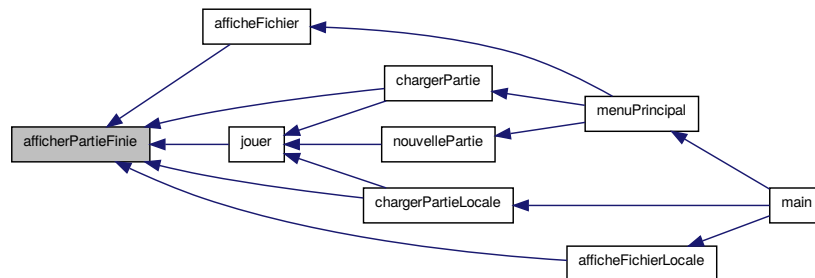
##### Parameters

in	*ptr_struct_fichier	un pointeur sur la structure <a href="#">Fichier_Jeu</a>
----	---------------------	--

Here is the call graph for this function:



Here is the caller graph for this function:



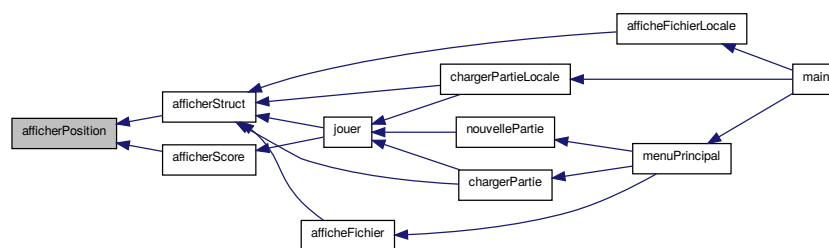
#### 4.2.2.8 void afficherPosition ( Fichier\_Jeu \* ptr\_struct\_fichier )

Affiche les positions des joueurs

##### Parameters

in	*ptr_struct_fichier	un pointeur sur la structure <a href="#">Fichier_Jeu</a>
----	---------------------	--

Here is the caller graph for this function:



#### 4.2.2.9 void afficherScore ( Fichier\_Jeu \* ptr\_struct\_fichier )

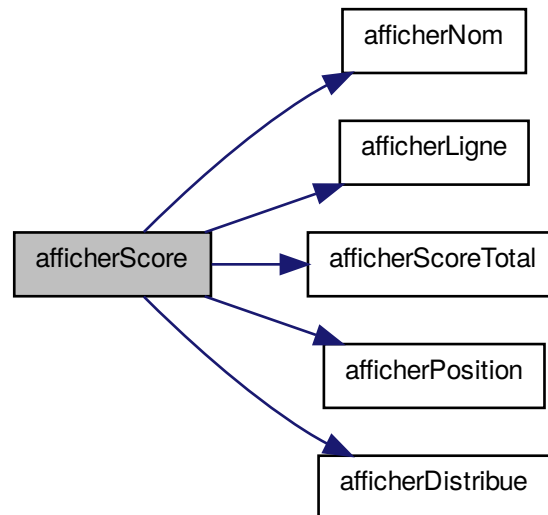
Affiche les noms des personnes avec leurs scores totaux ainsi que la personne devant distribuer à partir d'une structure [Fichier\\_Jeu](#)

##### Parameters

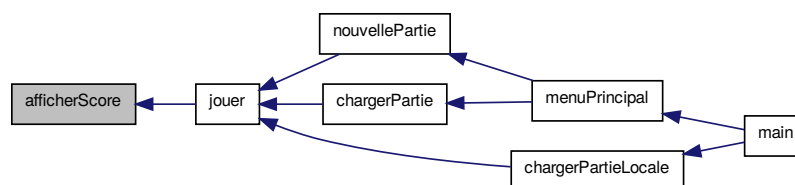
in	*ptr_struct_fichier	un pointeur sur la structure <a href="#">Fichier_Jeu</a>
----	---------------------	--



Here is the call graph for this function:



Here is the caller graph for this function:



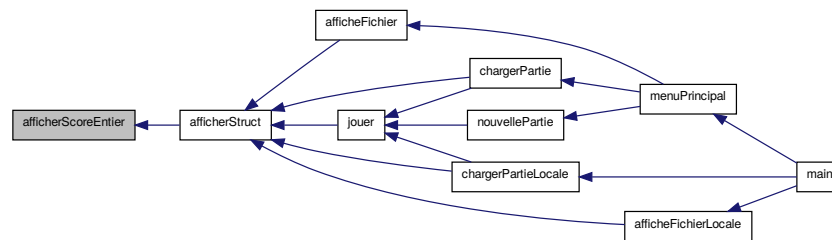
#### 4.2.2.10 void afficherScoreEntier ( Fichier\_Jeu \* ptr\_struct\_fichier )

Affiche tout les scores de tous les joueurs a chaque tour.

##### Parameters

in	<i>*ptr_struct_fichier</i>	un pointeur sur la structure <a href="#">Fichier_Jeu</a>
----	----------------------------	--

Here is the caller graph for this function:



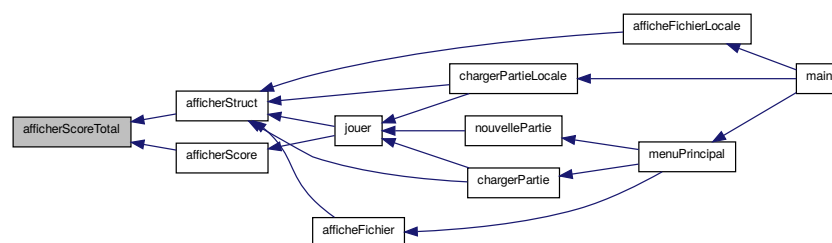
#### 4.2.2.11 void afficherScoreTotal ( Fichier\_Jeu \* ptr\_struct\_fichier )

Affiche le score total des joueurs

Parameters

in	*ptr_struct_fichier	un pointeur sur la structure <a href="#">Fichier_Jeu</a>
----	---------------------	--

Here is the caller graph for this function:



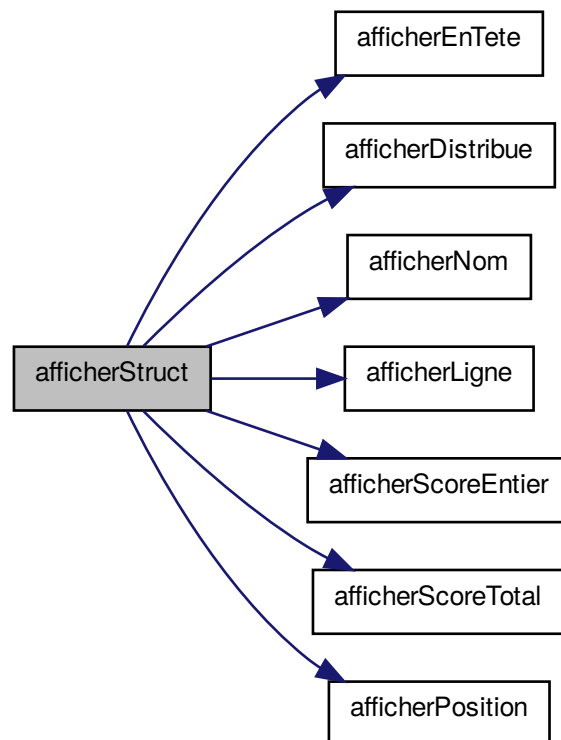
#### 4.2.2.12 void afficherStruct ( Fichier\_Jeu \* ptr\_struct\_fichier )

Affiche toutes les donnes d'une structure [Fichier\\_Jeu](#)

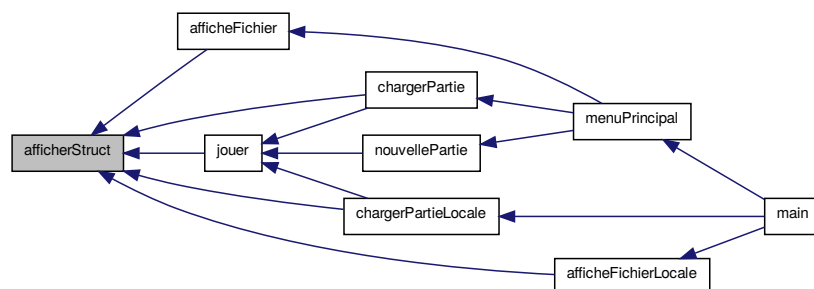
Parameters

in	*ptr_struct_fichier	un pointeur sur la structure <a href="#">Fichier_Jeu</a>
----	---------------------	--

Here is the call graph for this function:



Here is the caller graph for this function:

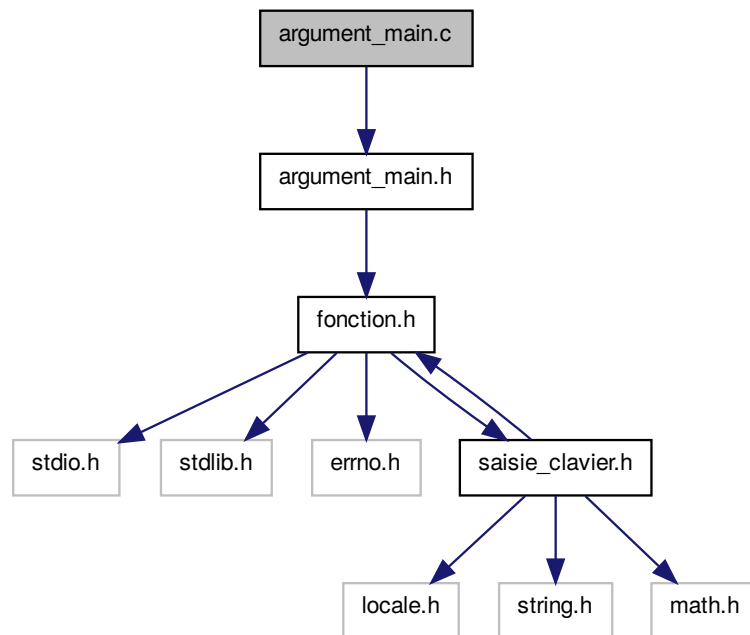


## 4.3 argument\_main.c File Reference

Lancement du programme.

```
#include "argument_main.h"
```

Include dependency graph for argument\_main.c:



## Functions

- int [searchArgument](#) (int argc, char \*argv[], int \*fonction, int \*emplacement\_fichier)

### 4.3.1 Detailed Description

Lancement du programme.

Author

Remi BERTHO

Date

21/03/14

Version

2.1.0

### 4.3.2 Function Documentation

4.3.2.1 int `searchArgument` ( int *argc*, char \* *argv*[], int \* *fonction*, int \* *emplacement\_fichier* )

Lance le programme

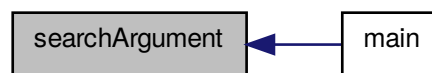
## Parameters

in	<i>argc</i>	le nombre d 'argument
in	<i>argv</i>	le tableau des arguments
in	<i>fonction</i>	entier determinant quelle fonction lancer
in	<i>emplacement_ - fichier</i>	entier donnant l'emplacement du fichier a ouvrir

## Returns

VRAI si la fonction a trouve un argument, FAUX sinon

Here is the caller graph for this function:

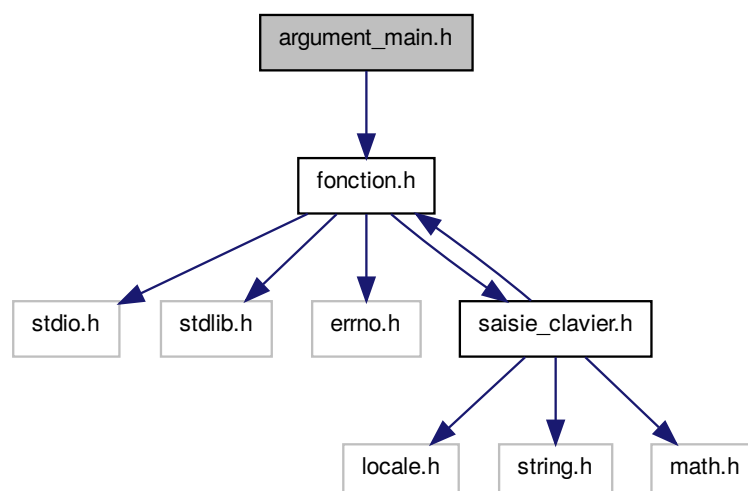


## 4.4 argument\_main.h File Reference

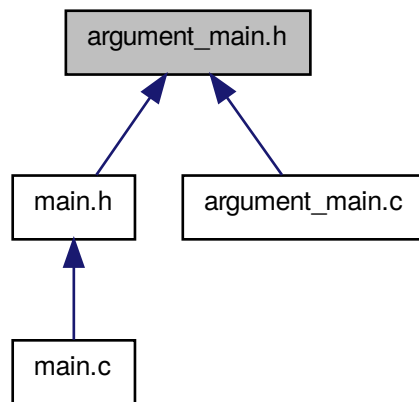
Lancement du programme.

```
#include "fonction.h"
```

Include dependency graph for argument\_main.h:



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



## Macros

- `#define CHAINE_Lecture_Fichier "--read"`
- `#define CHAINE_Lecture_Fichier_Red "-r"`
- `#define LECTURE_FICHIER 0`
- `#define CHAINE_Ouverture_Fichier "--open"`
- `#define CHAINE_Ouverture_Fichier_Red "-o"`
- `#define OUVERTURE_FICHIER 1`

## Functions

- `int searchArgument (int argc, char *argv[], int *fonction, int *emplacement_fichier)`

### 4.4.1 Detailed Description

Lancement du programme.

#### Author

Remi BERTHO

#### Date

21/03/14

#### Version

2.1.0

### 4.4.2 Macro Definition Documentation

#### 4.4.2.1 `#define CHAINE_Lecture_Fichier "--read"`

Définit l'appel à la lecture du fichier à "--read"

#### 4.4.2.2 #define CHAINE\_LECTURE\_FICHER\_RED "-r"

Definit l'appel a la lecture du fichier a "-r"

#### 4.4.2.3 #define CHAINE\_OUVERTURE\_FICHER "--open"

Definit l'appel a l'ouverture du fichier a "--open"

#### 4.4.2.4 #define CHAINE\_OUVERTURE\_FICHER\_RED "-o"

Definit l'appel a l'ouverture du fichier a "-o"

#### 4.4.2.5 #define LECTURE\_FICHER 0

Definit l'appel a la lecture du fichier a 0

#### 4.4.2.6 #define OUVERTURE\_FICHER 1

Definit l'appel a l'ouverture du fichier a 1

### 4.4.3 Function Documentation

#### 4.4.3.1 int searchArgument ( int *argc*, char \* *argv*[], int \* *fonction*, int \* *emplacement\_fichier* )

Lance le programme

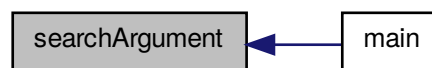
##### Parameters

in	<i>argc</i>	le nombre d 'argument
in	<i>argv</i>	le tableau des arguments
in	<i>fonction</i>	entier determinant quelle fonction lancer
in	<i>emplacement_ - fichier</i>	entier donnant l'emplacement du fichier a ouvrir

##### Returns

VRAI si la fonction a trouve un argument, FAUX sinon

Here is the caller graph for this function:

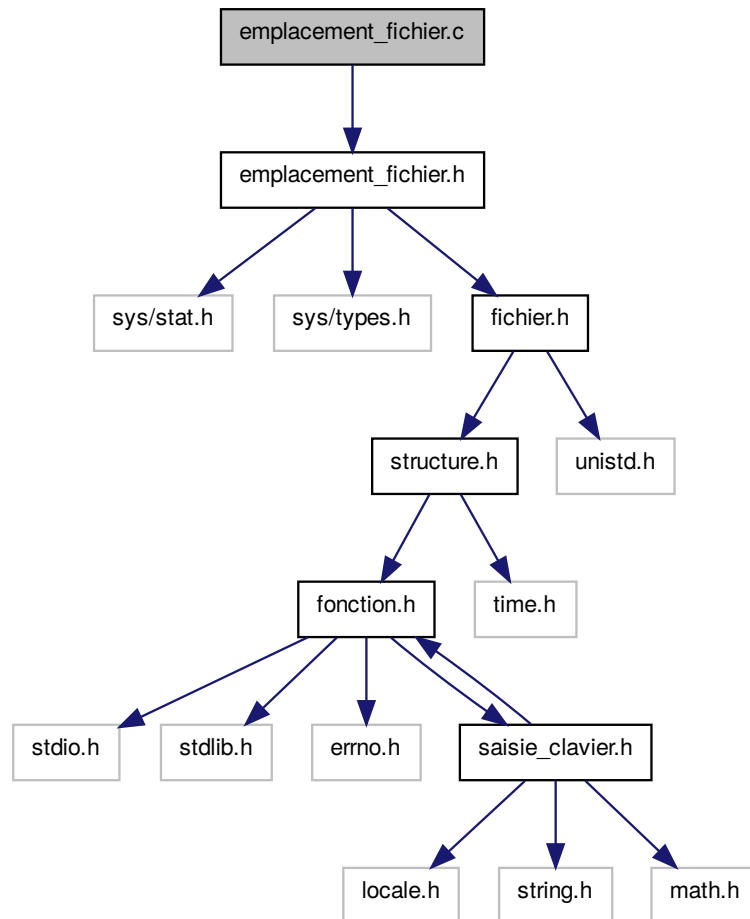


## 4.5 emplacement\_fichier.c File Reference

Fonctions qui l'emplacement des fichiers sauvegardes.

```
#include "emplacement_fichier.h"
```

Include dependency graph for emplacement\_fichier.c:



## Functions

- int [creationPreferences](#) ()
- int [lecturePreferences](#) (char \*nom\_fichier)
- int [lectureCheminFichier](#) (char \*nom\_fichier)
- int [changerCheminFichier](#) (char \*nouveauChemin)

### 4.5.1 Detailed Description

Fonctions qui l'emplacement des fichiers sauvegardes.

#### Author

Remi BERTHO



## Date

13/02/14

## Version

2.0

## 4.5.2 Function Documentation

4.5.2.1 int changerCheminFichier ( char \* *nouveauChemin* )

Changer le chemin de sauvegarde des fichier par nouveauChemin

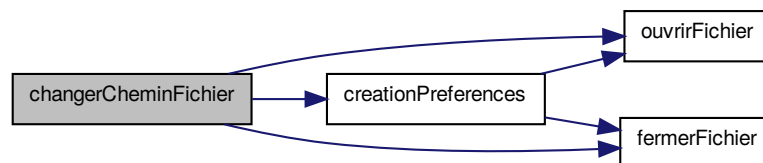
## Parameters

in, out	* <i>nouveau-Chemin</i>	le nomveau chemin
---------	-------------------------	-------------------

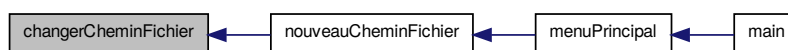
## Returns

VRAI si tout s'est bien passe, FAUX sinon

Here is the call graph for this function:



Here is the caller graph for this function:



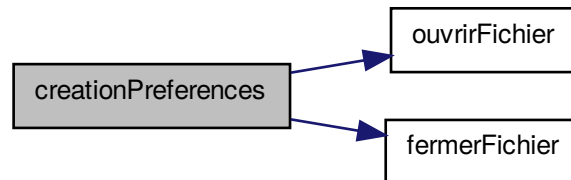
## 4.5.2.2 void creationPreferences ( )

Cree le dossier et le fichier qui va contenir les preferences

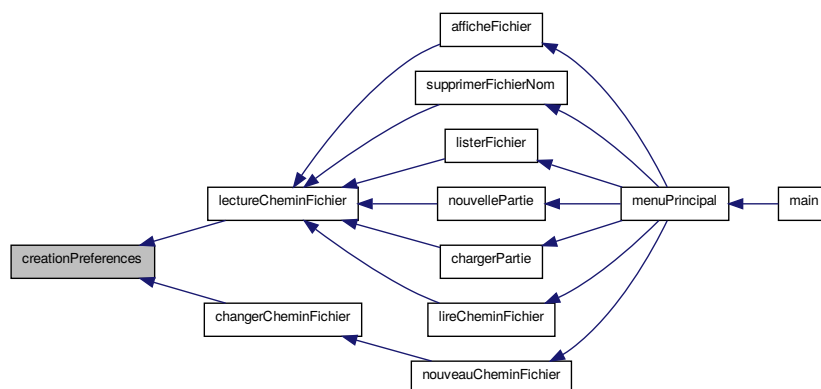
**Returns**

VRAI si tout s'est bien passe, FAUX sinon

Here is the call graph for this function:



Here is the caller graph for this function:



#### 4.5.2.3 int lectureCheminFichier ( char \* nom\_fichier )

Ajoute le chemin du fichier dans le nom du fichier a partir des preferences, si le fichier de preference n'existe pas il le cree Attention ne fait quelque chose uniquement si la constant PORTABLE n'est pas defini

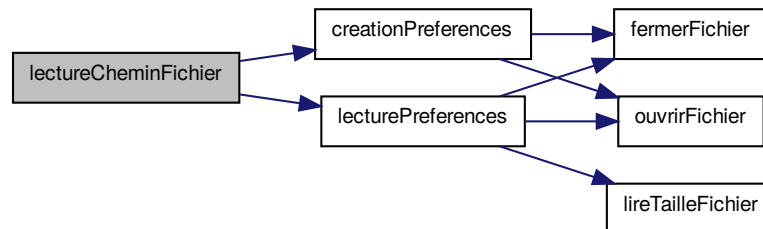
**Parameters**

<code>in, out</code>	<code>*nom_fichier</code>	le nom du fichier a qui on va ajouter son chemin
----------------------	---------------------------	--

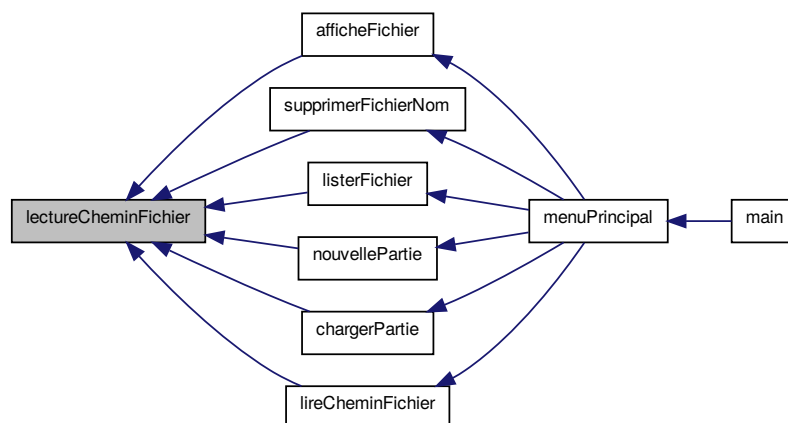
## Returns

VRAI si tout s'est bien passe, FAUX sinon

Here is the call graph for this function:



Here is the caller graph for this function:



#### 4.5.2.4 int lecturePreferences ( char \* nom\_fichier )

lis les preferences du fichier et ajoute le chemin lu dans le nom de fichier passe en parametre

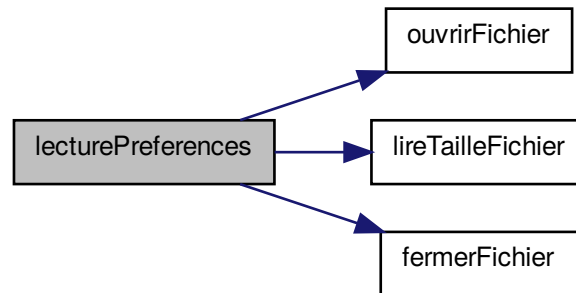
## Parameters

in, out	*nom_fichier	le nom du fichier a qui on va ajouter son chemin
---------	--------------	--

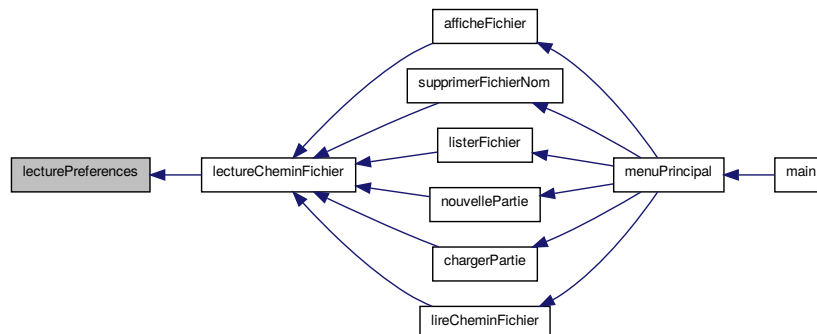
**Returns**

VRAI si tout s'est bien passe, FAUX sinon

Here is the call graph for this function:



Here is the caller graph for this function:

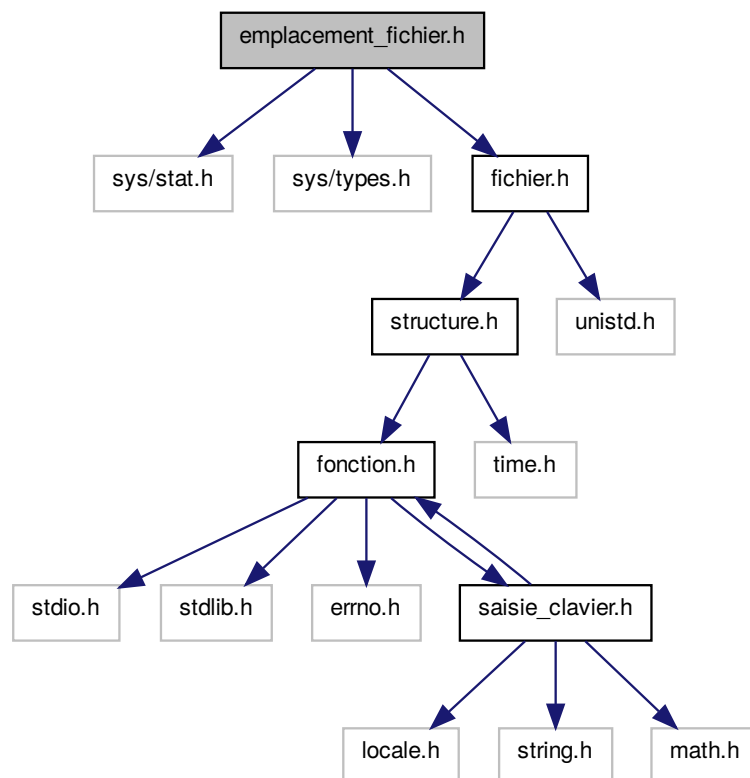


## 4.6 emplacement\_fichier.h File Reference

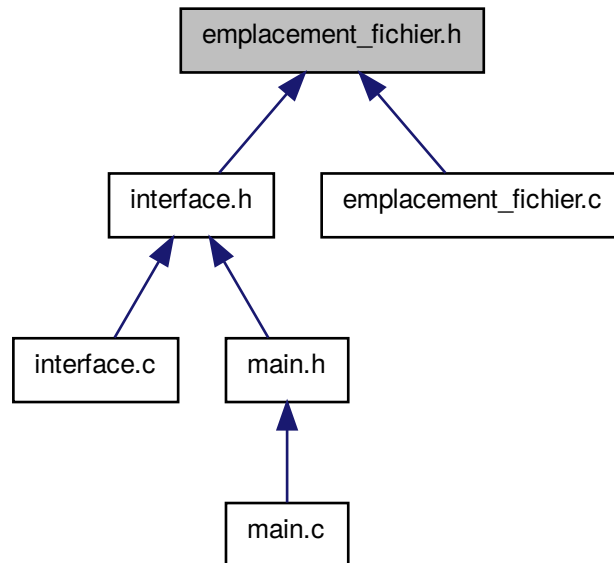
Prototypes des fonctions qui l'emplacement des fichiers sauvegardes.

```
#include <sys/stat.h>
#include <sys/types.h>
#include "fichier.h"
```

Include dependency graph for emplacement\_fichier.h:



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



## Macros

- `#define NOM_FICHIER "preferences.txt"`
- `#define NOM_DOSSIER ".csuper"`

## Functions

- `int creationPreferences ()`
- `int lecturePreferences (char *nom_fichier)`
- `int lectureCheminFichier (char *nom_fichier)`
- `int changerCheminFichier (char *nouveauChemin)`

### 4.6.1 Detailed Description

Prototypes des fonctions qui l'emplacement des fichiers sauvegardes.

#### Author

Remi BERTHO

#### Date

13/02/14

#### Version

2.0

## 4.6.2 Macro Definition Documentation

### 4.6.2.1 #define NOM\_DOSSIER ".csuper"

Definit NOM\_DOSSIER a ".csuper"

### 4.6.2.2 #define NOM\_FICHER "preferences.txt"

Definit NOM\_FICHER a "preferences.txt"

## 4.6.3 Function Documentation

### 4.6.3.1 int changerCheminFichier ( char \* *nouveauChemin* )

Changer le chemin de sauvegarde des fichier par nouveauChemin

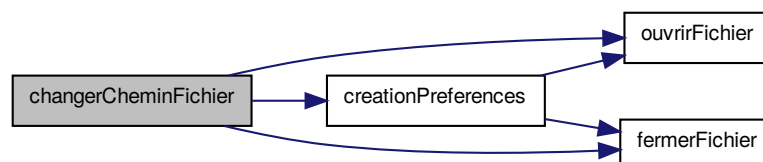
#### Parameters

in, out	* <i>nouveau-Chemin</i>	le nouveau chemin
---------	-------------------------	-------------------

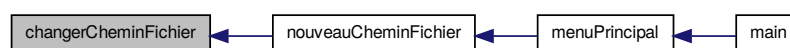
#### Returns

VRAI si tout s'est bien passe, FAUX sinon

Here is the call graph for this function:



Here is the caller graph for this function:



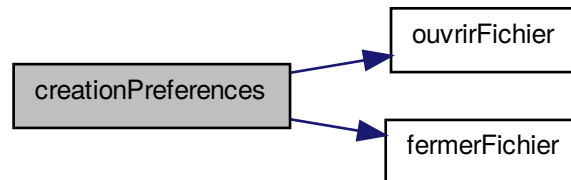
### 4.6.3.2 int creationPreferences ( )

Cree le dossier et le fichier qui va contenir les preferences

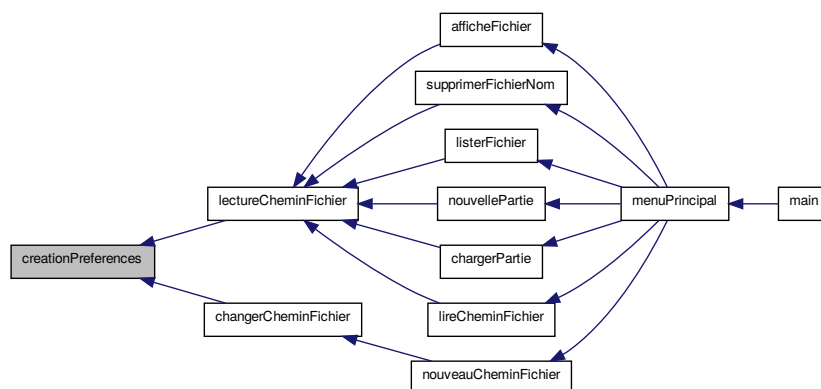
**Returns**

VRAI si tout s'est bien passe, FAUX sinon

Here is the call graph for this function:



Here is the caller graph for this function:



#### 4.6.3.3 int lectureCheminFichier ( char \* *nom\_fichier* )

Ajoute le chemin du fichier dans le nom du fichier a partir des preferences, si le fichier de preference n'existe pas il le cree Attention ne fait quelque chose uniquement si la constant PORTABLE n'est pas defini

**Parameters**

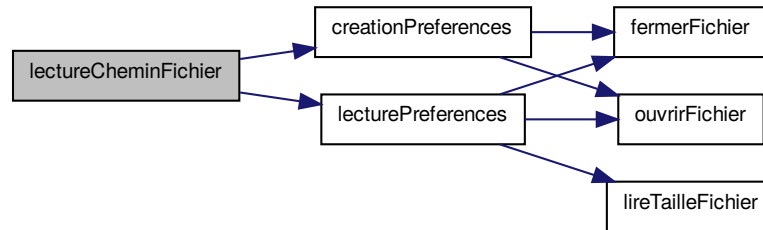
<code>in, out</code>	<code>*<i>nom_fichier</i></code>	le nom du fichier a qui on va ajouter son chemin
----------------------	----------------------------------	--



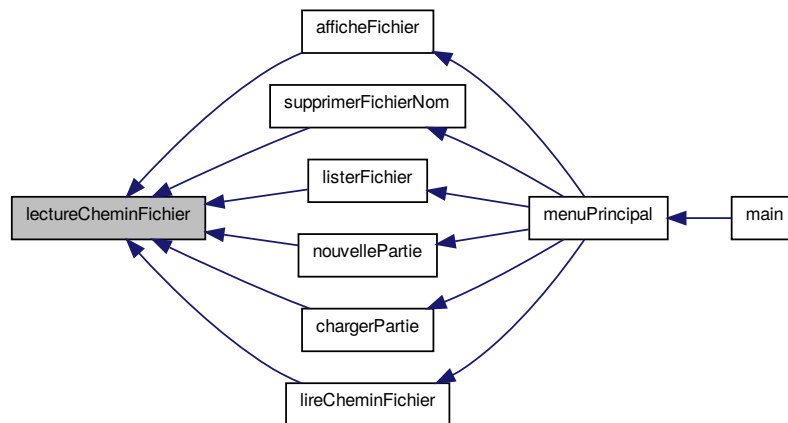
## Returns

VRAI si tout s'est bien passe, FAUX sinon

Here is the call graph for this function:



Here is the caller graph for this function:



#### 4.6.3.4 int lecturePreferences ( char \* nom\_fichier )

lis les preferences du fichier et ajoute le chemin lu dans le nom de fichier passe en parametre

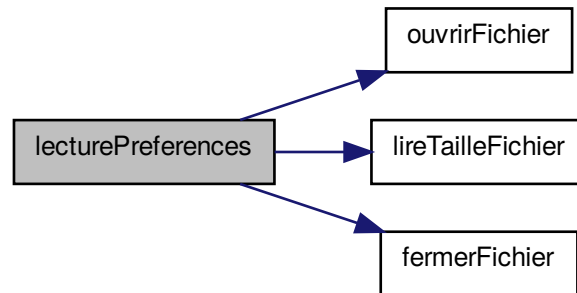
## Parameters

in, out	*nom_fichier	le nom du fichier a qui on va ajouter son chemin
---------	--------------	--

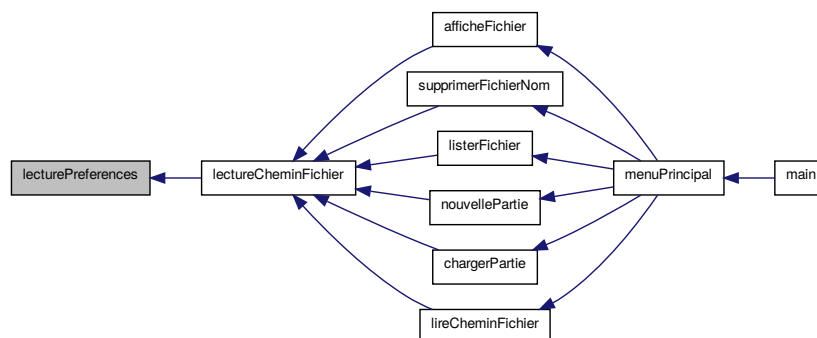
**Returns**

VRAI si tout s'est bien passe, FAUX sinon

Here is the call graph for this function:



Here is the caller graph for this function:

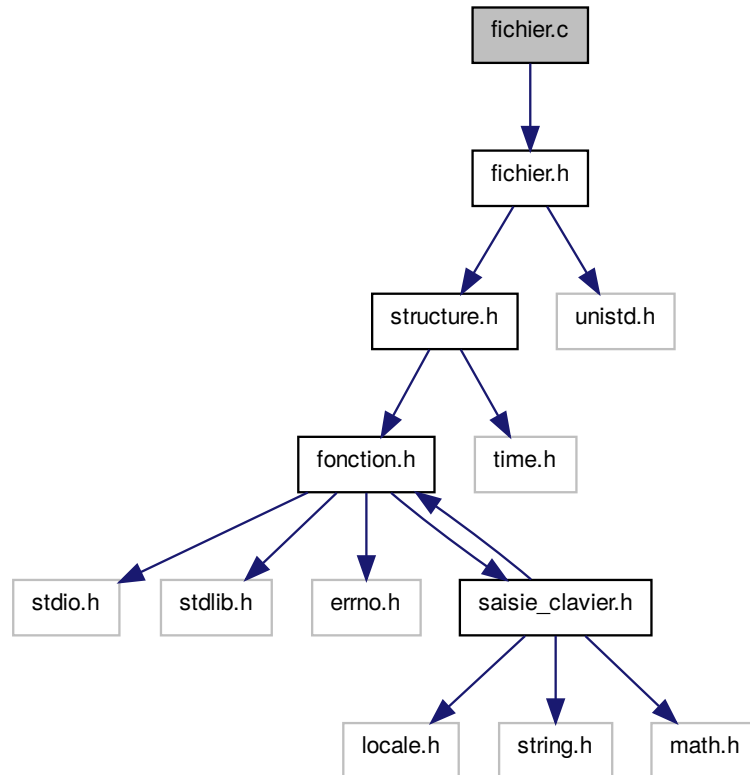


## 4.7 fichier.c File Reference

Fonction de gestion des fichiers.

```
#include "fichier.h"
```

Include dependency graph for fichier.c:



## Functions

- `FILE * ouvrirFichierExtension` (`char nom[]`, `char mode[]`)
- `Fichier_Jeu * lireFichier` (`char *nom`)
- `int ecrireFichier` (`char *nom`, `Fichier_Jeu *ptr_struct_fichier`)
- `int nouveauScore` (`char *nom`, `Fichier_Jeu *ptr_struct_fichier`)
- `int supprimerFichier` (`char *nom`)
- `int renommerFichier` (`char *nom_ancien`, `char *nom_nouveau`)

### 4.7.1 Detailed Description

Fonction de gestion des fichiers.

#### Author

Remi BERTHO

#### Date

09/03/14

## Version

2.1.0

## 4.7.2 Function Documentation

## 4.7.2.1 int ecrireFichier ( char \* nom, Fichier\_Jeu \* ptr\_struct\_fichier )

Crée un fichier .jeu qui contient toutes les informations de la structure [Fichier\\_Jeu](#) mis en paramètre

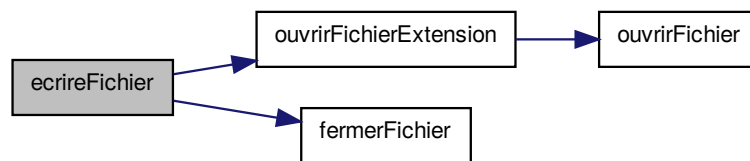
## Parameters

in	*nom	le nom du fichier
in	*ptr_struct_fichier	la structure du fichier avec lequel on veut créer le fichier

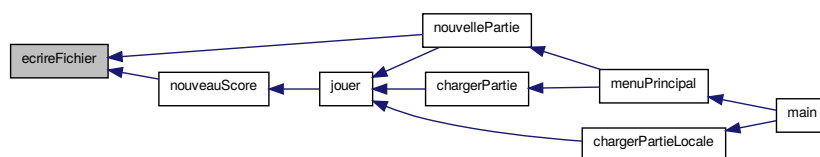
## Returns

VRAI si tout s'est bien passé, FAUX sinon

Here is the call graph for this function:



Here is the caller graph for this function:



## 4.7.2.2 Fichier\_Jeu \* lireFichier ( char \* nom )

Lis ce qu'il y a dans le fichier avec le nom donné en paramètre et le met dans une structure [Fichier\\_Jeu](#) rendu par la fonction

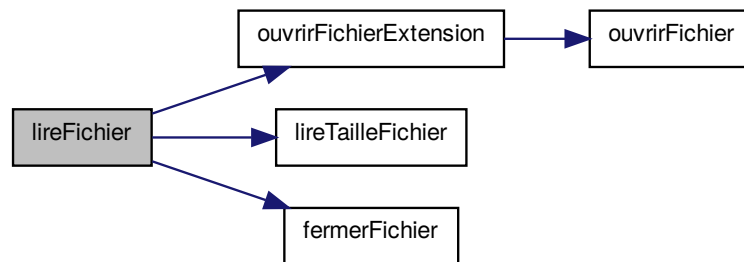
## Parameters

in	<i>nom[]</i>	le nom du fichier
----	--------------	-------------------

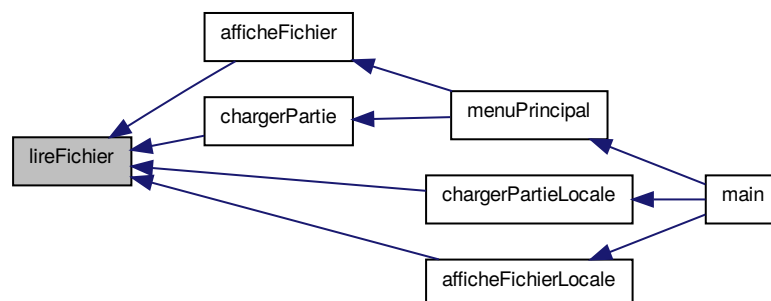
**Returns**

un pointeur sur la structure [Fichier\\_Jeu](#) cree, NULL s'il y a un probleme d'ouverture du fichier

Here is the call graph for this function:



Here is the caller graph for this function:



#### 4.7.2.3 void nouveauScore ( char \* nom, Fichier\_Jeu \* ptr\_struct\_fichier )

Mets a jour le fichier avec les nouveaux scores

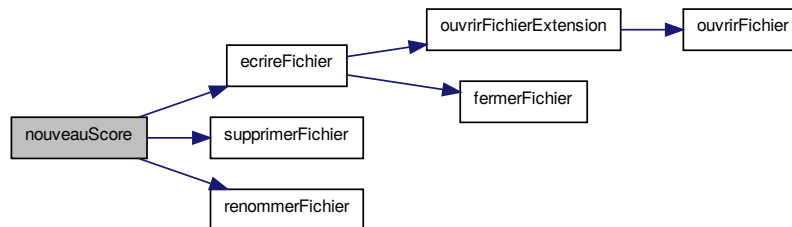
**Parameters**

in	<i>*nom</i>	le nom du fichier
in	<i>*ptr_struct_fichier</i>	la structure du fichier avec lequel on veut mettre un nouveau score

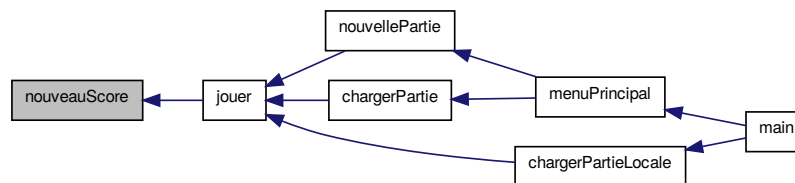
**Returns**

VRAI si tout s'est bien passe, FAUX sinon

Here is the call graph for this function:



Here is the caller graph for this function:



#### 4.7.2.4 FILE \* ouvrirFichierExtension ( char *nom*[], char *mode*[] )

Ouvre un fichier a partir de son nom et du mode voulu en y ajouter l'extension du fichier si necessaire

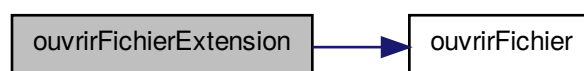
**Parameters**

in	<i>nom</i> []	le nom du fichier
in	<i>mode</i> []	le mode voulu

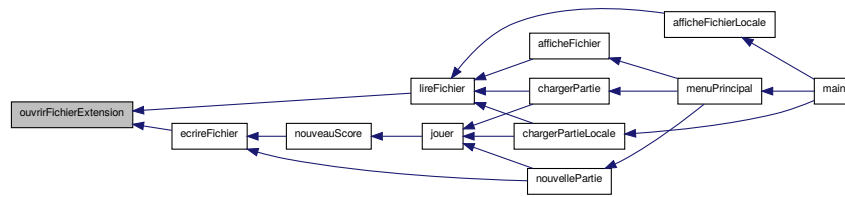
**Returns**

un pointeur sur le fichier ouvert, NULL s'il y a eut un probleme

Here is the call graph for this function:



Here is the caller graph for this function:



#### 4.7.2.5 int renommerFichier ( char \* *nom\_ancien*, char \* *nom\_nouveau* )

Renomme le fichier dont le nom est en parametre

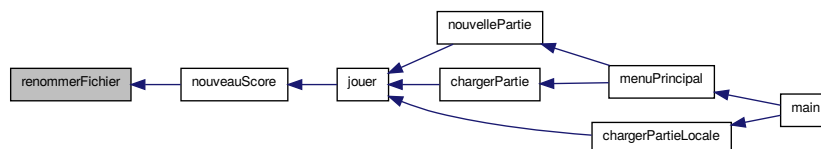
##### Parameters

in	* <i>nom_ancien</i>	l'ancien nom du fichier
in	* <i>nom_nouveau</i>	le nouveau nom du fichier

##### Returns

VRAI si tout s'est bien passe, FAUX sinon

Here is the caller graph for this function:



#### 4.7.2.6 int supprimerFichier ( char \* *nom* )

Supprime le fichier dont le nom est en parametre

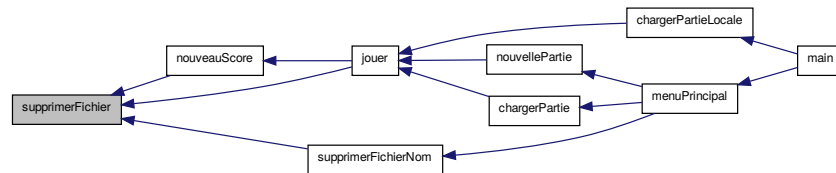
##### Parameters

in	* <i>nom</i>	le nom du fichier
----	--------------	-------------------

## Returns

VRAI si tout s'est bien passe, FAUX sinon

Here is the caller graph for this function:



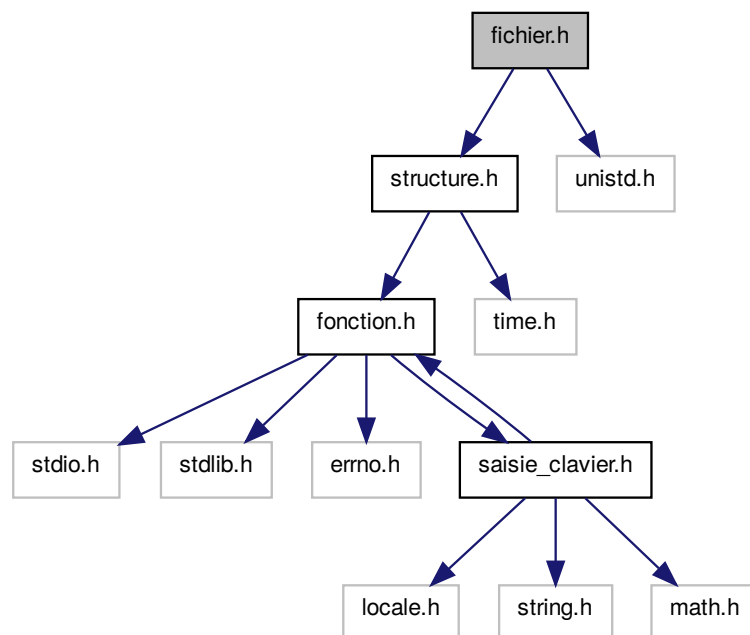
## 4.8 fichier.h File Reference

Prototypes des fonction de gestion des fichiers.

```
#include "structure.h"
```

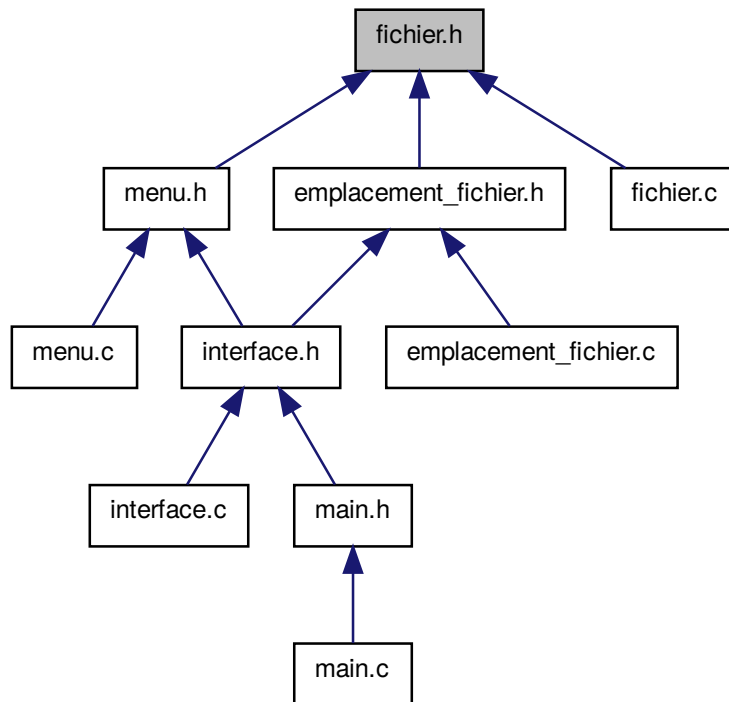
```
#include <unistd.h>
```

Include dependency graph for fichier.h:





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



## Macros

- `#define TAILLE_MAX_NOM_FICHIER 250`
- `#define EXTENSION_FICHIER "csu"`
- `#define TYPE_FICHIER "CompteurScoreUniversel"`

## Functions

- `FILE * ouvrirFichierExtension (char nome[], char mode[])`
- `Fichier_Jeu * lireFichier (char *nom)`
- `int ecrireFichier (char *nom, Fichier_Jeu *ptr_struct_fichier)`
- `int nouveauScore (char *nom, Fichier_Jeu *ptr_struct_fichier)`
- `int supprimerFichier (char *nom)`
- `int renommerFichier (char *nom_ancien, char *nom_nouveau)`

### 4.8.1 Detailed Description

Prototypes des fonction de gestion des fichiers.

#### Author

Remi BERTHO

## Date

13/02/14

## Version

2.0

## 4.8.2 Macro Definition Documentation

### 4.8.2.1 #define EXTENSION\_FICHIER "csu"

Definit l'extension du fichier a "csu"

### 4.8.2.2 #define TAILLE\_MAX\_NOM\_FICHIER 250

Definit la taille max d'un nom a 250

### 4.8.2.3 #define TYPE\_FICHIER "CompteurScoreUniversel"

Definit la chaine de caractere permettant de verifier le type de fichier a "CompteurScoreUniversel"

## 4.8.3 Function Documentation

### 4.8.3.1 int ecrireFichier ( char \* *nom*, Fichier\_Jeu \* *ptr\_struct\_fichier* )

Cree un fichier .jeu qui contient toutes les informations de la structure [Fichier\\_Jeu](#) mis en parametre

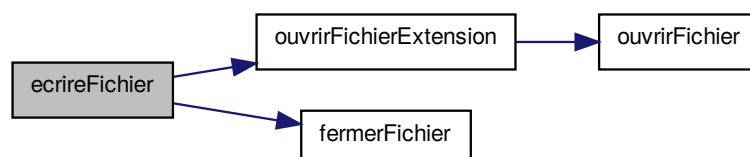
**Parameters**

in	<i>*nom</i>	le nom du fichier
in	<i>*ptr_struct_fichier</i>	la structure du fichier avec lequel on veut cree le fichier

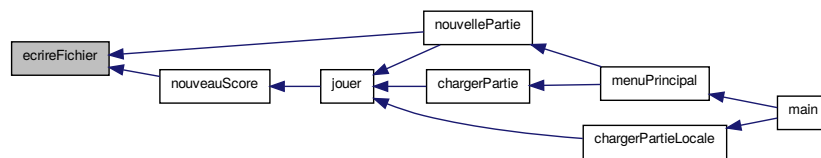
**Returns**

VRAI si tout s'est bien passe, FAUX sinon

Here is the call graph for this function:



Here is the caller graph for this function:



#### 4.8.3.2 Fichier\_Jeu\* lireFichier ( char \* nom )

Lis ce qu'il y a dans le fichier avec le nom donne en parametre et le met dans une structure [Fichier\\_Jeu](#) rendu par la fonction

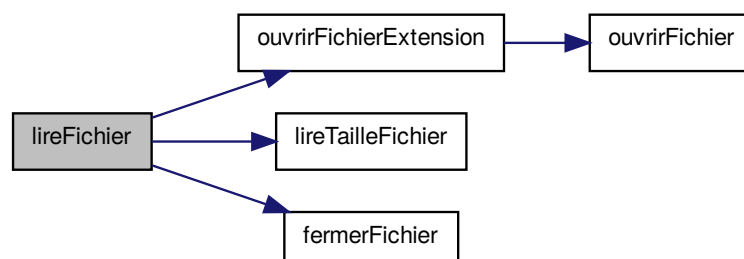
##### Parameters

in	nom[]	le nom du fichier
----	-------	-------------------

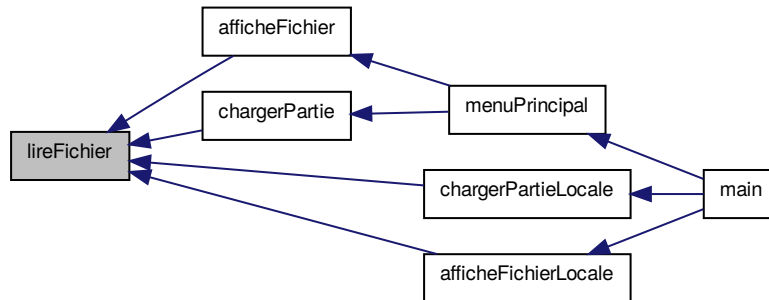
##### Returns

un pointeur sur la structure [Fichier\\_Jeu](#) cree, NULL s'il y a un probleme d'ouverture du fichier

Here is the call graph for this function:



Here is the caller graph for this function:



#### 4.8.3.3 int nouveauScore ( char \* nom, Fichier\_Jeu \* ptr\_struct\_fichier )

Mets a jour le fichier avec les nouveaux scores

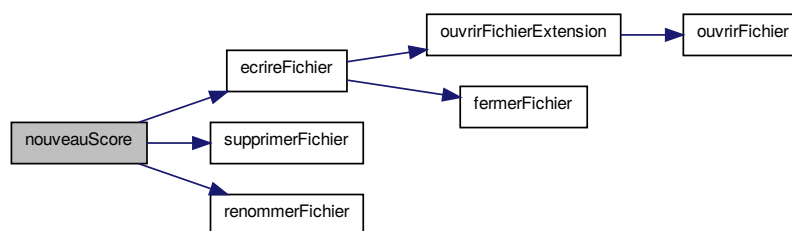
##### Parameters

in	*nom	le nom du fichier
in	*ptr_struct_fichier	la structure du fichier avec lequel on veut mettre un nouveau score

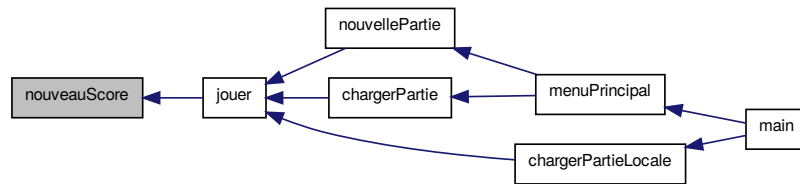
##### Returns

VRAI si tout s'est bien passe, FAUX sinon

Here is the call graph for this function:



Here is the caller graph for this function:



#### 4.8.3.4 FILE\* ouvrirFichierExtension ( char *nom*[], char *mode*[] )

Ouvre un fichier a partir de son nom et du mode voulu en y ajouter l'extension du fichier si necessaire

##### Parameters

in	<i>nom</i> []	le nom du fichier
in	<i>mode</i> []	le mode voulu

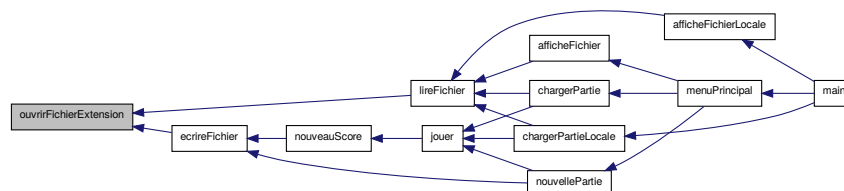
##### Returns

un pointeur sur le fichier ouvert, NULL s'il y a eut un probleme

Here is the call graph for this function:



Here is the caller graph for this function:



#### 4.8.3.5 int renommerFichier ( char \* *nom\_ancien*, char \* *nom\_nouveau* )

Renomme le fichier dont le nom est en parametre

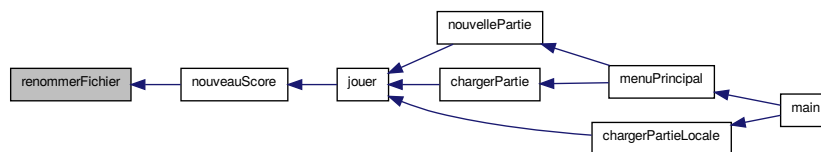
**Parameters**

in	<i>*nom_ancien</i>	l'ancien nom du fichier
in	<i>*nom_nouveau</i>	le nouveau nom du fichier

**Returns**

VRAI si tout s'est bien passe, FAUX sinon

Here is the caller graph for this function:

**4.8.3.6 int supprimerFichier ( char \* nom )**

Supprime le fichier dont le nom est en parametre

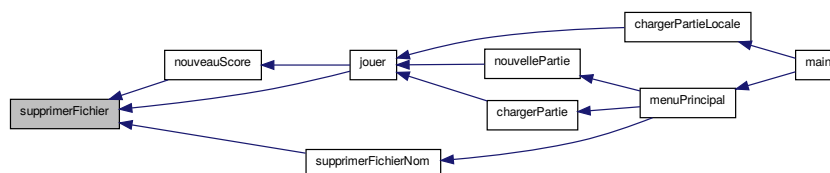
**Parameters**

in	<i>*nom</i>	le nom du fichier
----	-------------	-------------------

**Returns**

VRAI si tout s'est bien passe, FAUX sinon

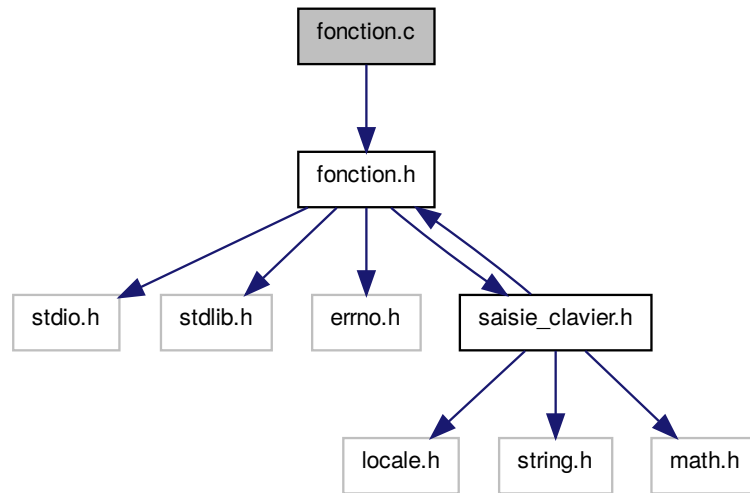
Here is the caller graph for this function:

**4.9 fonction.c File Reference**

Fonctions essentielles au fonctionnement du programme.

```
#include "fonction.h"
```

Include dependency graph for fonction.c:



## Functions

- void [mauvais\\_choix](#) ()
- void [systemPause](#) ()
- void [systemEfface](#) ()
- int [compareFlottantCroissant](#) (void const \*a, void const \*b)
- int [compareFlottantDecroissant](#) (void const \*a, void const \*b)
- FILE \* [ouvrirFichier](#) (char nom[], char mode[])
- int [fermerFichier](#) (FILE \*ptr\_fichier)
- int [lireTailleFichier](#) (FILE \*ptr\_fichier)

### 4.9.1 Detailed Description

Fonctions essentielles au fonctionnement du programme.

#### Author

Remi BERTHO

#### Date

13/02/14

#### Version

2.0

## 4.9.2 Function Documentation

4.9.2.1 `int compareFlottantCroissant ( void const * a, void const * b )`

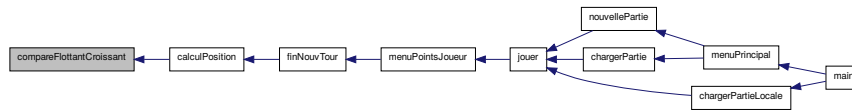
Compare 2 Flottant, renvoie 1 si  $a > b$ , 0 si  $a = b$  et -1 si  $a < b$



## Parameters

in	<i>*a</i>	un pointeur sur un flottant
in	<i>*b</i>	un pointeur sur un flottant

Here is the caller graph for this function:

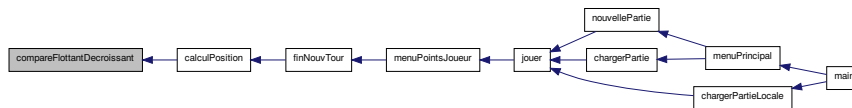
4.9.2.2 int int compareFlottantDecroissant ( void const \* *a*, void const \* *b* )

Compare 2 Flottant, renvoie 1 si  $a < b$ , 0 si  $a = b$  et -1 si  $a > b$

## Parameters

in	<i>*a</i>	un pointeur sur un flottant
in	<i>*b</i>	un pointeur sur un flottant

Here is the caller graph for this function:

4.9.2.3 int fermerFichier ( FILE \* *ptr\_fichier* )

Ferme le fichier

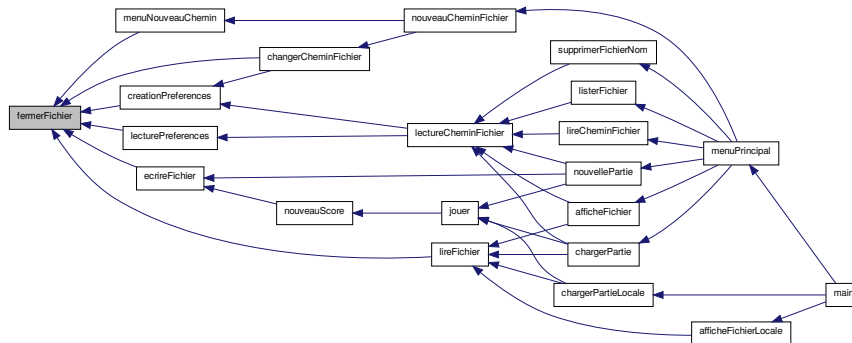
## Parameters

in	<i>*ptr_fichier</i>	le fichier
----	---------------------	------------

**Returns**

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

Here is the caller graph for this function:



#### 4.9.2.4 int lireTailleFichier ( FILE \* *ptr\_fichier* )

Lis la taille du fichier

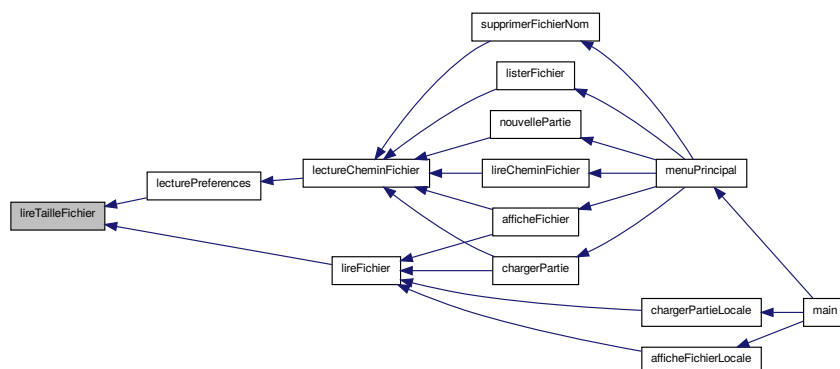
**Parameters**

in	* <i>ptr_fichier</i>	le fichier
----	----------------------	------------

**Returns**

entier ayant la taille du fichier

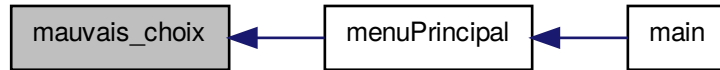
Here is the caller graph for this function:



#### 4.9.2.5 void mauvais\_choix ( )

Affiche un message d'erreur.

Here is the caller graph for this function:



#### 4.9.2.6 FILE \* ouvrirFichier ( char nom[], char mode[] )

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

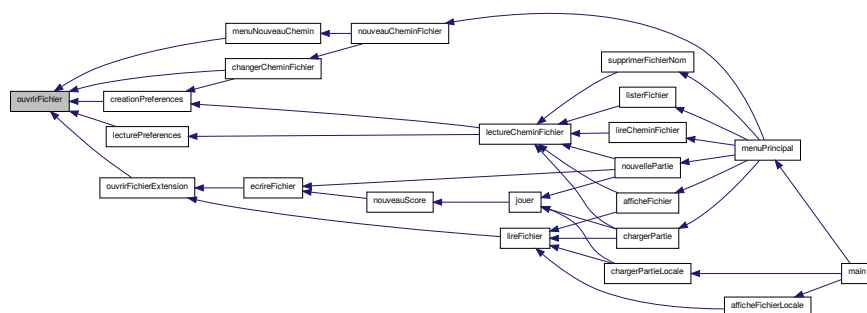
##### Parameters

in	<i>nom[]</i>	le nom du fichier
in	<i>mode[]</i>	le mode voulu

##### Returns

un pointeur sur le fichier ouvert, NULL s'il y a eut un probleme

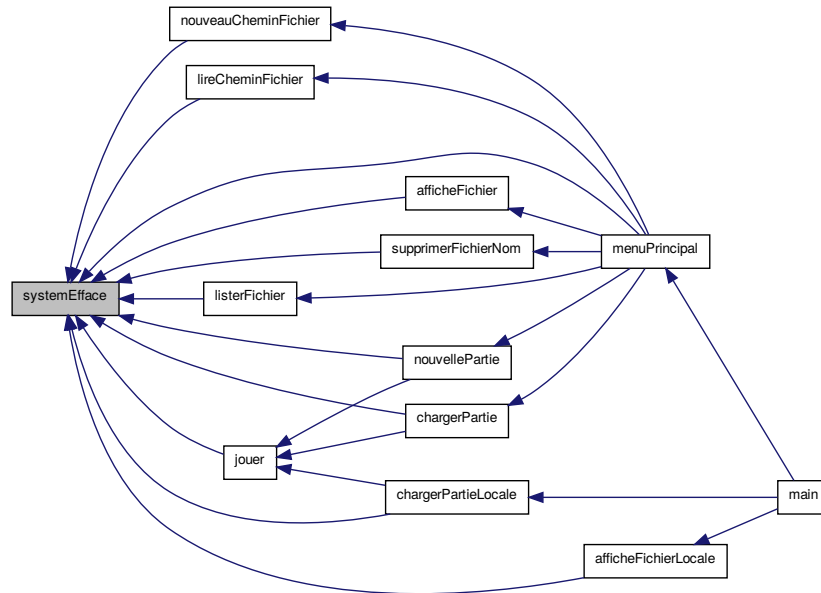
Here is the caller graph for this function:



#### 4.9.2.7 void systemEfface ( )

Efface la console de l'utilisateur.

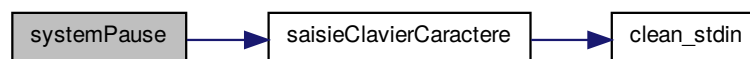
Here is the caller graph for this function:



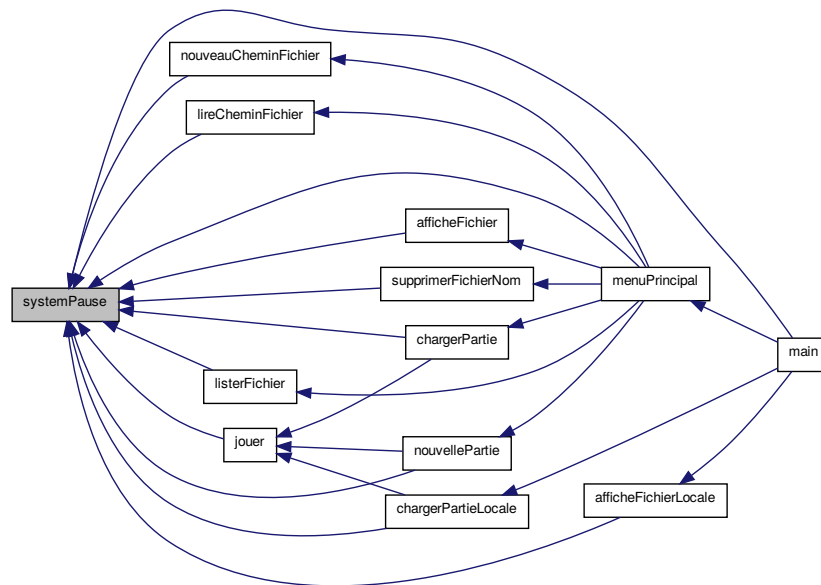
#### 4.9.2.8 void systemPause ( )

Demande a l'utilisateur d'appuyer sur entrer pour continuer le programme.

Here is the call graph for this function:



Here is the caller graph for this function:



## 4.10 fonction.h File Reference

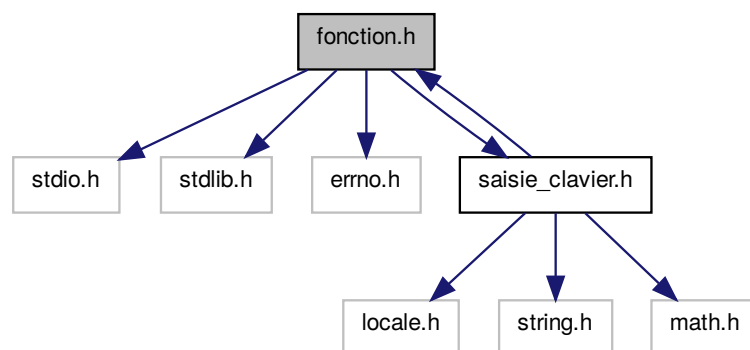
Prototypes des fonctions essentielles au fonctionnement du programme.

```

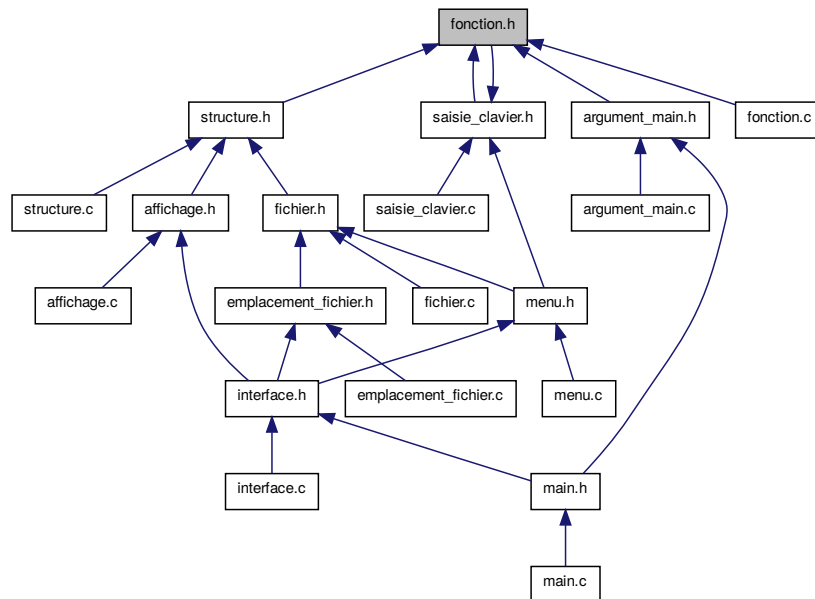
#include <stdio.h>
#include <stdlib.h>
#include <errno.h>
#include "saisie_clavier.h"

```

Include dependency graph for fonction.h:



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



## Macros

- `#define` [VRAI](#) 1
- `#define` [FAUX](#) 0

## Functions

- void [mauvais\\_choix](#) ()
- void [systemPause](#) ()
- void [systemEfface](#) ()
- int [compareFlottantDecroissant](#) (void const \*a, void const \*b)
- int [compareFlottantCroissant](#) (void const \*a, void const \*b)
- FILE \* [ouvrirFichier](#) (char nome[], char mode[])
- int [fermerFichier](#) (FILE \*ptr\_fichier)
- int [lireTailleFichier](#) (FILE \*ptr\_fichier)

### 4.10.1 Detailed Description

Prototypes des fonctions essentielles au fonctionnement du programme.

#### Author

Remi BERTHO

#### Date

13/02/14

#### Version

2.0

## 4.10.2 Macro Definition Documentation

### 4.10.2.1 #define FAUX 0

Definit FAUX a 0

### 4.10.2.2 #define VRAI 1

Definit VRAI a 1

## 4.10.3 Function Documentation

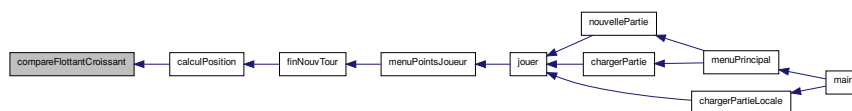
### 4.10.3.1 int compareFlottantCroissant ( void const \* *a*, void const \* *b* )

Compare 2 Flottant, renvoie 1 si  $a > b$ , 0 si  $a = b$  et -1 si  $a < b$

Parameters

in	<i>*a</i>	un pointeur sur un flottant
in	<i>*b</i>	un pointeur sur un flottant

Here is the caller graph for this function:



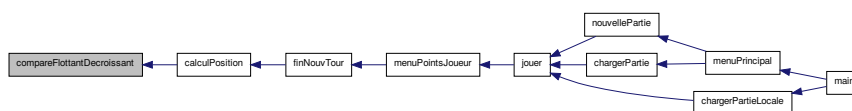
### 4.10.3.2 int compareFlottantDecroissant ( void const \* *a*, void const \* *b* )

Compare 2 Flottant, renvoie 1 si  $a < b$ , 0 si  $a = b$  et -1 si  $a > b$

Parameters

in	<i>*a</i>	un pointeur sur un flottant
in	<i>*b</i>	un pointeur sur un flottant

Here is the caller graph for this function:



### 4.10.3.3 int fermerFichier ( FILE \* *ptr\_fichier* )

Ferme le fichier

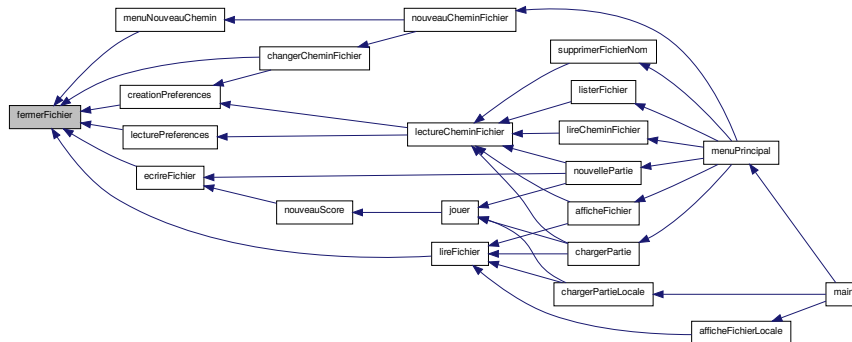
## Parameters

in	<i>*ptr_fichier</i>	le fichier
----	---------------------	------------

## Returns

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

Here is the caller graph for this function:



## 4.10.3.4 int lireTailleFichier ( FILE \* ptr\_fichier )

Lis la taille du fichier

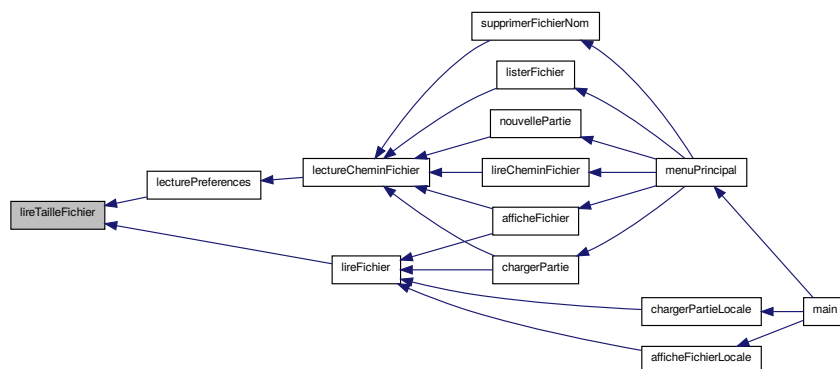
## Parameters

in	<i>*ptr_fichier</i>	le fichier
----	---------------------	------------

## Returns

entier ayant la taille du fichier

Here is the caller graph for this function:

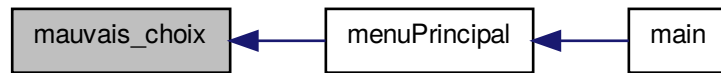




## 4.10.3.5 void mauvais\_choix ( )

Affiche un message d'erreur.

Here is the caller graph for this function:



## 4.10.3.6 FILE\* ouvrirFichier ( char nom[], char mode[] )

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

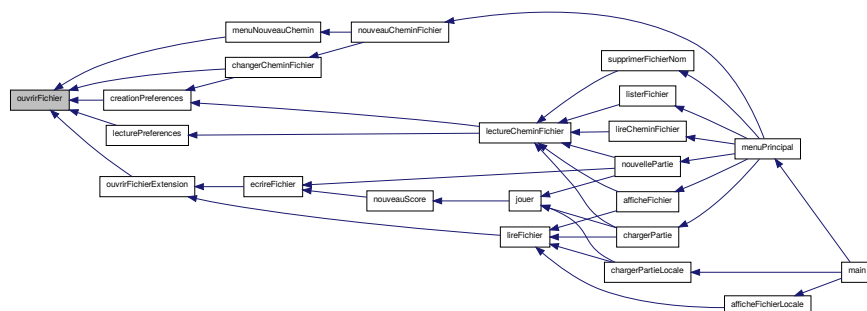
## Parameters

in	<i>nom[]</i>	le nom du fichier
in	<i>mode[]</i>	le mode voulu

## Returns

un pointeur sur le fichier ouvert, NULL s'il y a eut un probleme

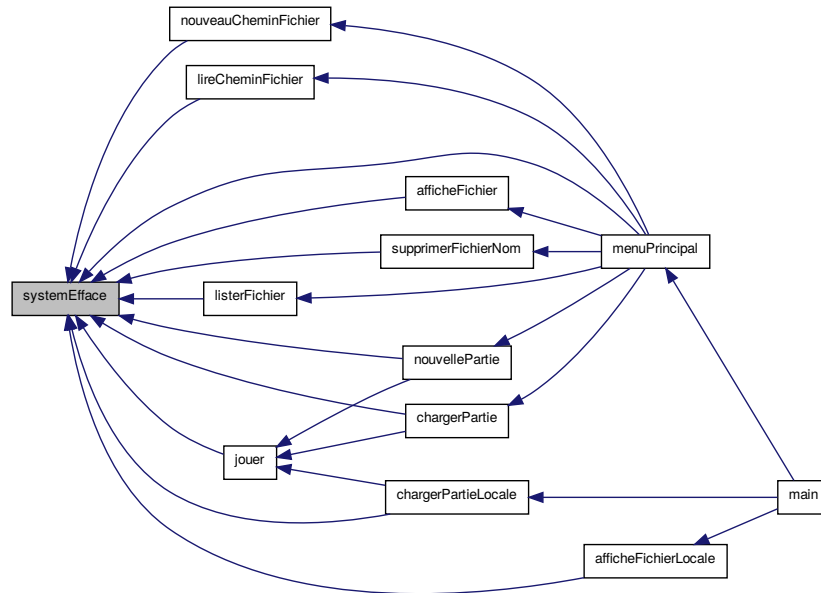
Here is the caller graph for this function:



## 4.10.3.7 void systemEfface ( )

Efface la console de l'utilisateur.

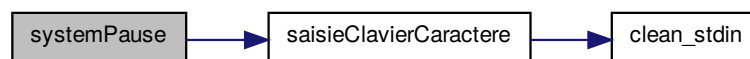
Here is the caller graph for this function:



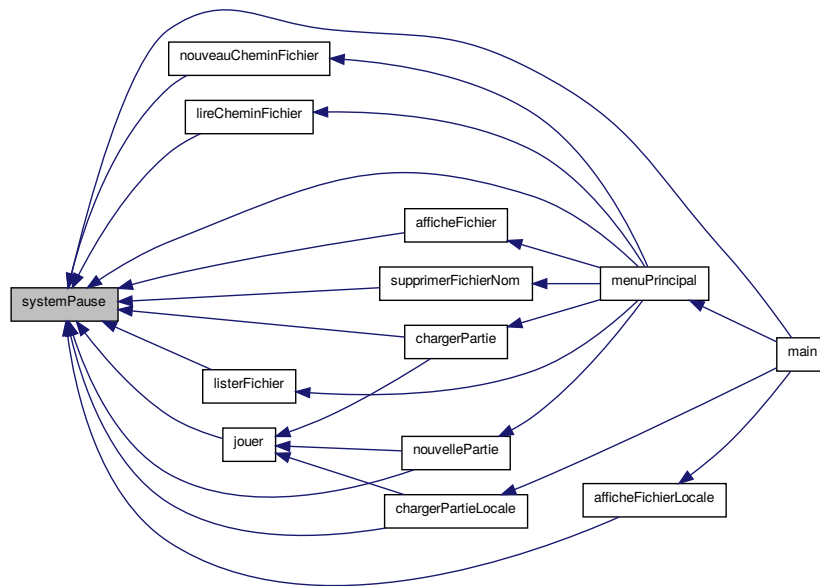
#### 4.10.3.8 void systemPause ( )

Demande a l'utilisateur d'appuyer sur entrer pour continuer le programme.

Here is the call graph for this function:



Here is the caller graph for this function:

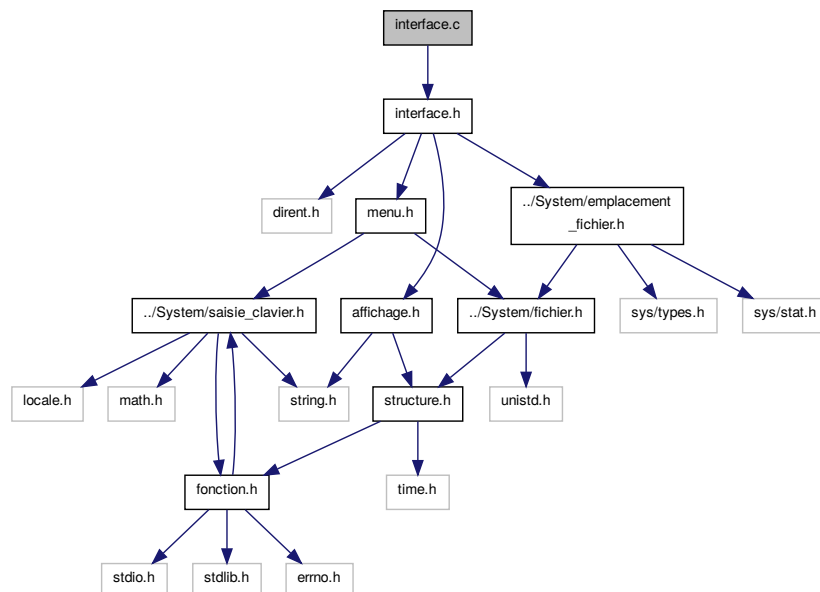


## 4.11 interface.c File Reference

Fonctions qui gerent l'interface graphique du logiciel.

```
#include "interface.h"
```

Include dependency graph for interface.c:



## Functions

- void `afficheFichier ()`
- void `supprimerFichierNom ()`
- void `listerFichier ()`
- void `jouer (Fichier_Jeu *ptr_struct_fichier, char *nom_fichier)`
- void `nouvellePartie ()`
- void `chargerPartie ()`
- void `menuPrincipal ()`
- void `nouveauCheminFichier ()`
- void `lireCheminFichier ()`
- void `chargerPartieLocale (char *nom_fichier)`
- void `afficheFichierLocale (char *nom_fichier)`

### 4.11.1 Detailed Description

Fonctions qui gerent l'interface graphique du logiciel.

#### Author

Remi BERTHO

#### Date

09/03/14

Version

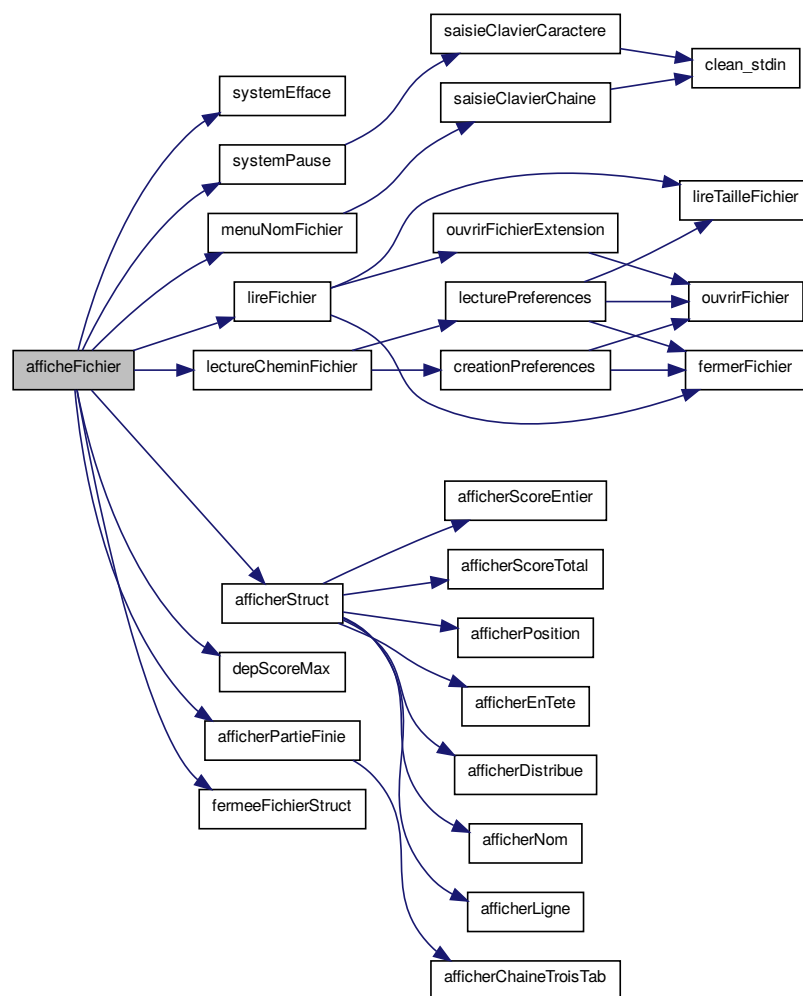
2.1.0

## 4.11.2 Function Documentation

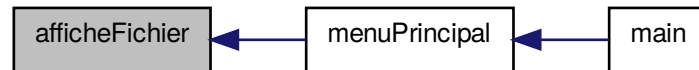
### 4.11.2.1 void afficheFichier ( )

Demande le nom d'un fichier et l'affiche

Here is the call graph for this function:



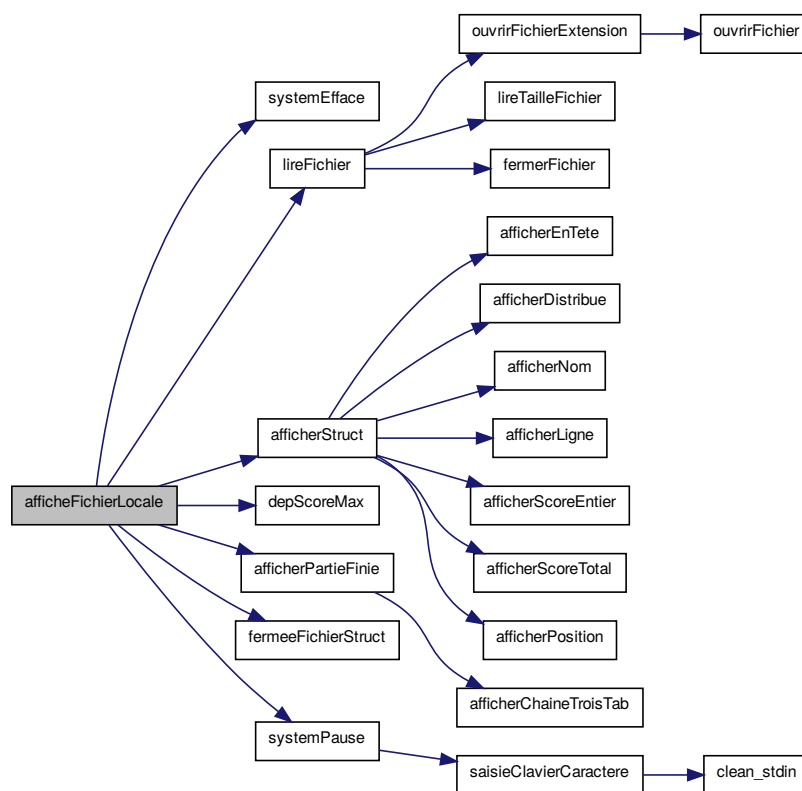
Here is the caller graph for this function:



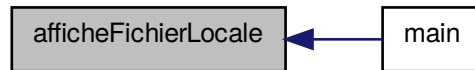
#### 4.11.2.2 void afficheFichierLocale ( char \* *nom\_fichier* )

Affiche le fichier dont le nom a ete donne en parametre

Here is the call graph for this function:



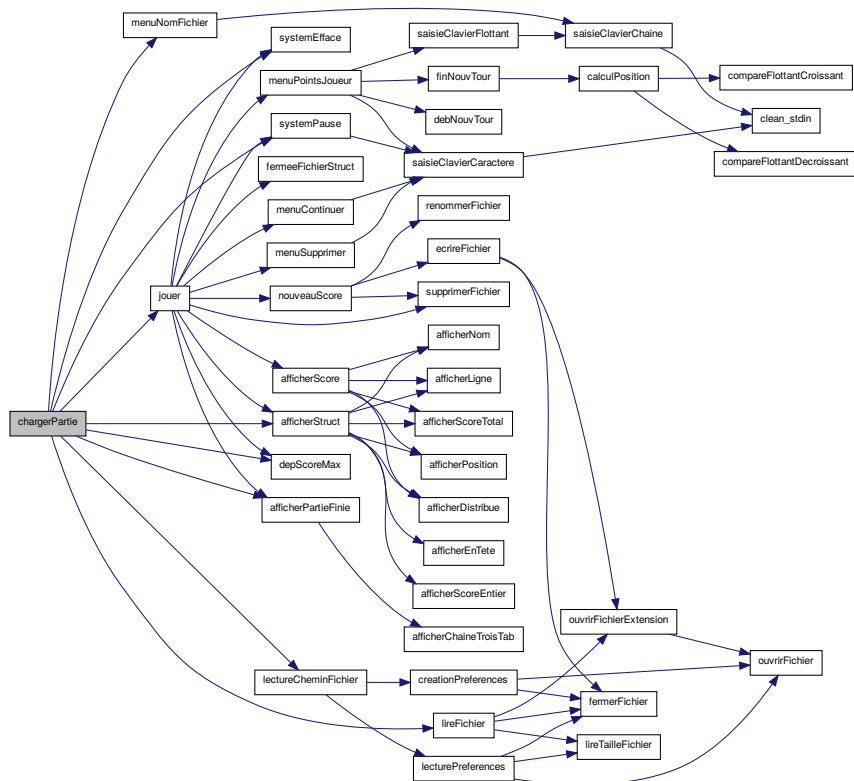
Here is the caller graph for this function:



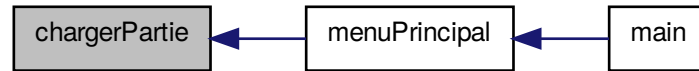
#### 4.11.2.3 void chargerPartie ( )

Charge une partie a partir d'un fichier et lance la fonction de comptage des points

Here is the call graph for this function:



Here is the caller graph for this function:



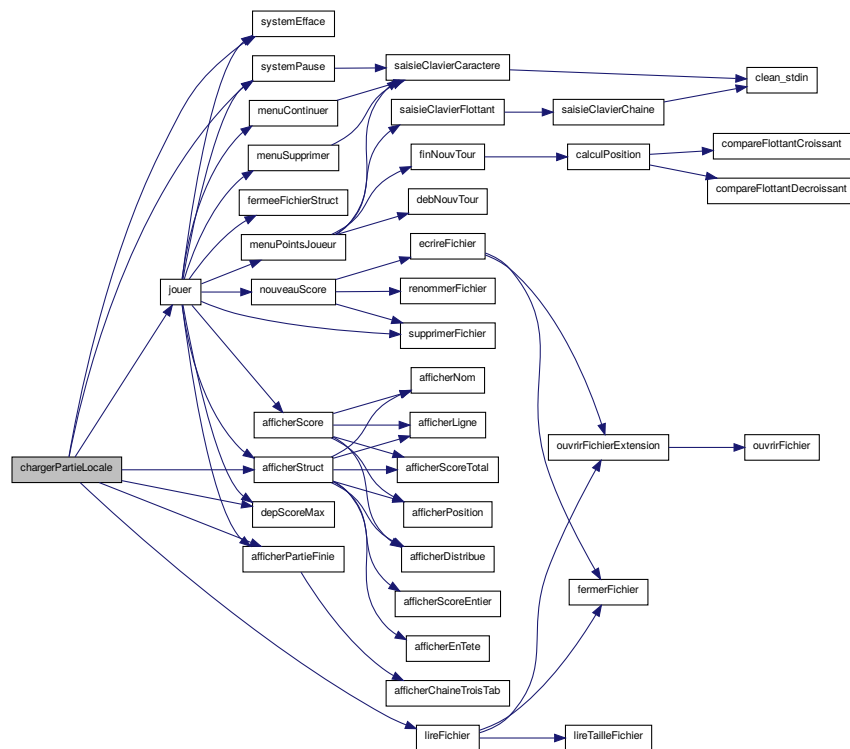
#### 4.11.2.4 void chargerPartieLocale ( char \* nom\_fichier )

Charge une partie a partir d'un fichier dont le nom a ete donne et lance la fonction de comptage des points

##### Parameters

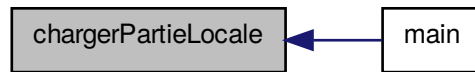
in	nom_fichier,le	nom du fichier
----	----------------	----------------

Here is the call graph for this function:





Here is the caller graph for this function:



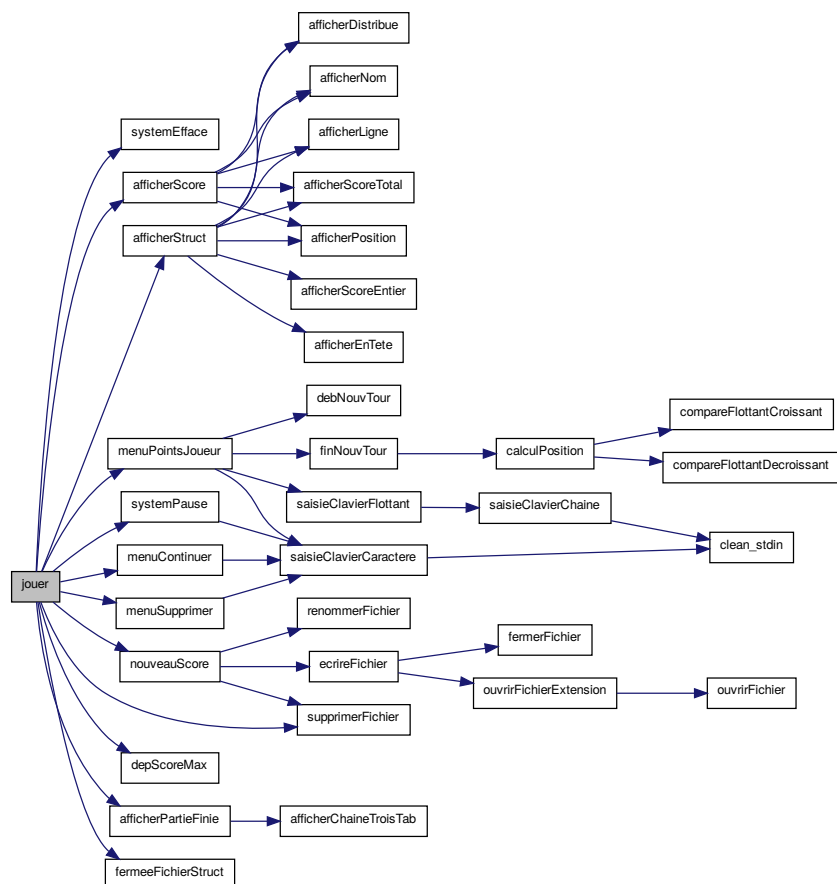
#### 4.11.2.5 void jouer ( Fichier\_Jeu \* ptr\_struct\_fichier, char \* nom\_fichier )

Lance la partie de comptage jusqu'à la fin

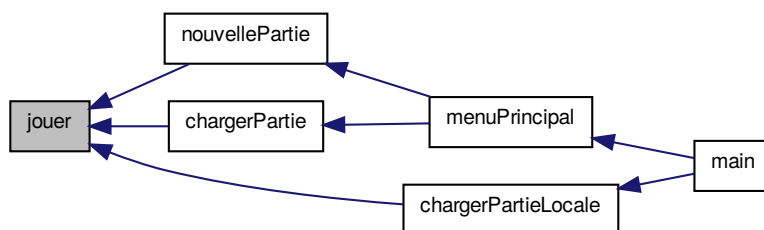
##### Parameters

in, out	*ptr_struct_fichier	la structure <a href="#">Fichier_Jeu</a>
in	*nom_fichier	le nom du fichier dans lequel on stocke les informations

Here is the call graph for this function:



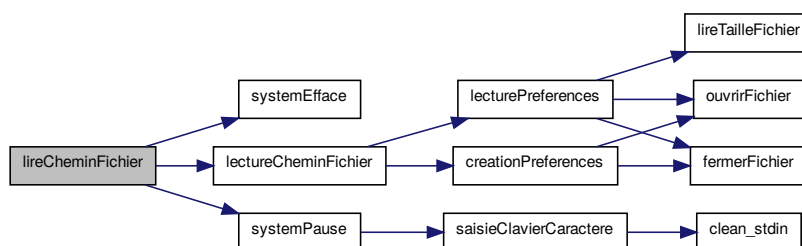
Here is the caller graph for this function:



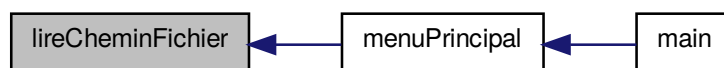
#### 4.11.2.6 void lireCheminFichier ( )

Lis le chemin de fichier et l'affiche

Here is the call graph for this function:



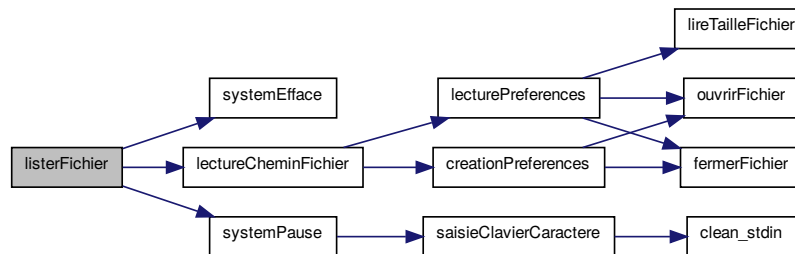
Here is the caller graph for this function:



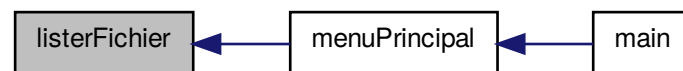
#### 4.11.2.7 void listerFichier ( )

Liste tout les fichier du dossier courant contenant l'extension .jeu

Here is the call graph for this function:



Here is the caller graph for this function:

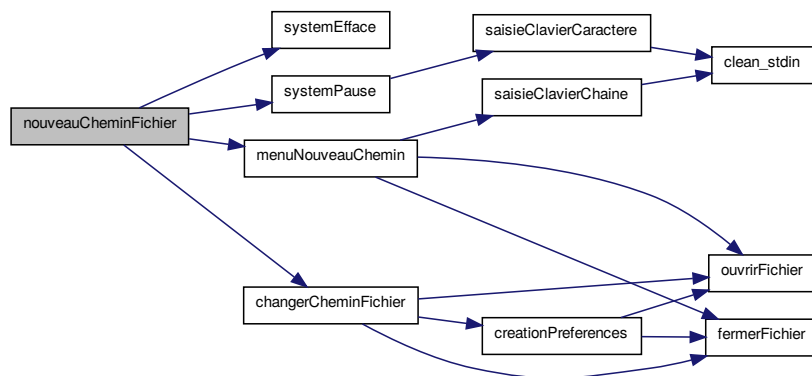


#### 4.11.2.8 void menuPrincipal ( )

Lance un menu que redirige vers l'action que l'on veut effectuer



Here is the call graph for this function:



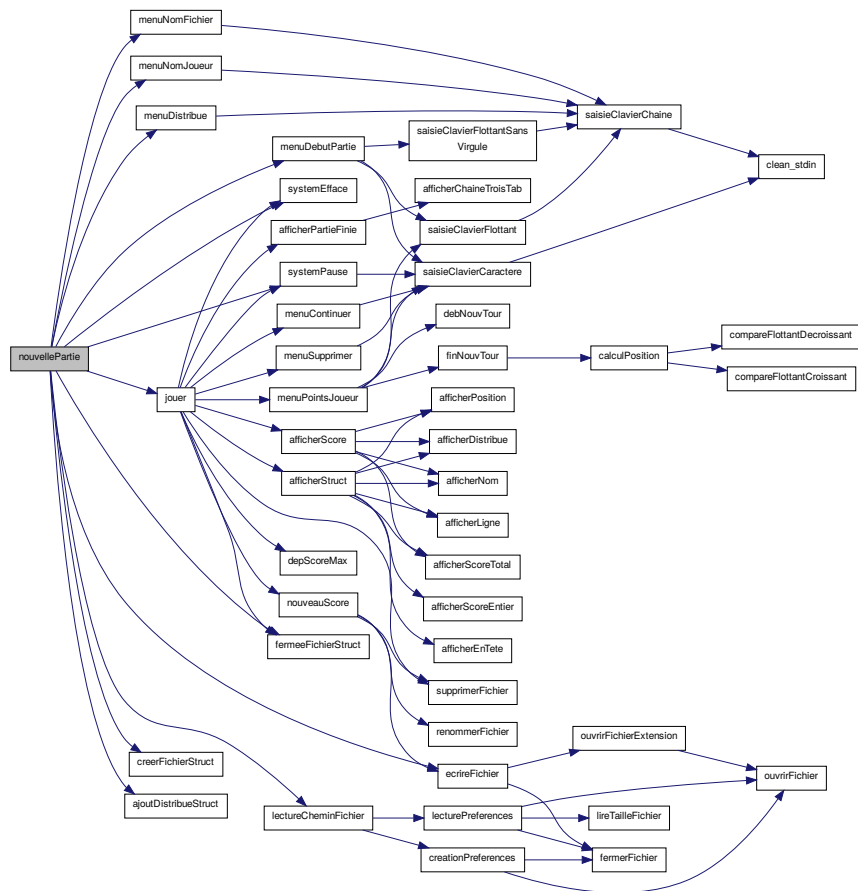
Here is the caller graph for this function:



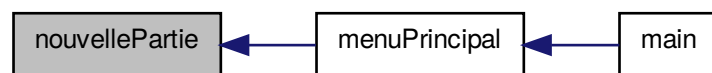
#### 4.11.2.10 void nouvellePartie ( )

Initialise une nouvelle partie et lance la fonction de comptage des points

Here is the call graph for this function:



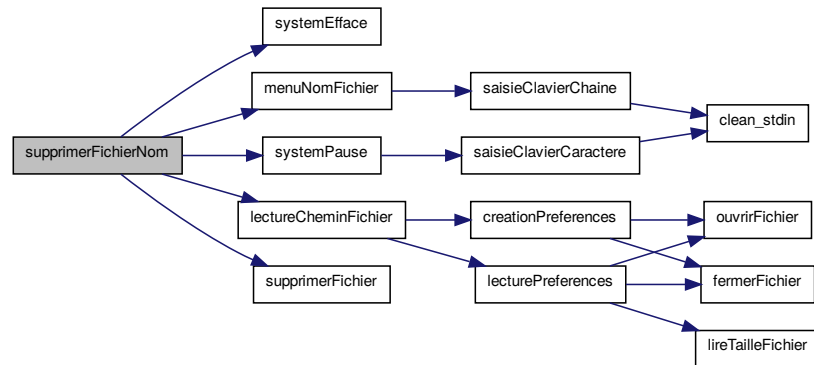
Here is the caller graph for this function:



#### 4.11.2.11 void supprimerFichierNom ( )

Demande le nom d'un fichier et le supprime

Here is the call graph for this function:



Here is the caller graph for this function:



## 4.12 interface.h File Reference

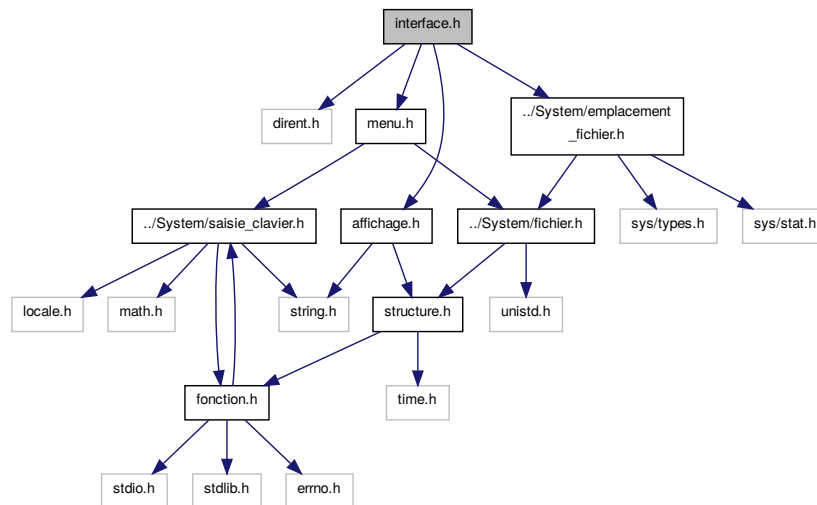
Prototypes des fonctions qui gerent l'interface graphique du logiciel.

```

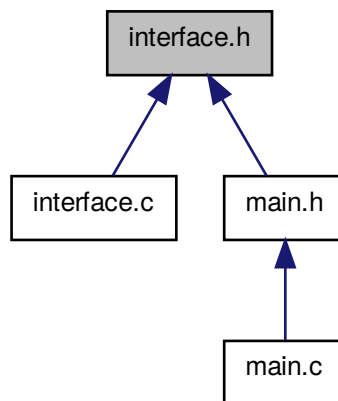
#include <dirent.h>
#include "menu.h"
#include "affichage.h"
#include "../System/emplacement_fichier.h"

```

Include dependency graph for interface.h:



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



## Enumerations

- enum `Menu` {  
`nouvPart` =1, `charPart` =2, `affFich` =3, `supprFich` =4,  
`listFich` =5, `nouvChem` =6, `lireChem` =7, `quit` =8,  
`easterEggs` = 42 }

## Functions

- void `afficheFichier` ()



- void [supprimerFichierNom](#) ()
- void [listerFichier](#) ()
- void [jouer](#) ([Fichier\\_Jeu](#) \*ptr\_struct\_fichier, char \*nom\_fichier)
- void [nouvellePartie](#) ()
- void [chargerPartie](#) ()
- void [menuPrincipal](#) ()
- void [nouveauCheminFichier](#) ()
- void [lireCheminFichier](#) ()
- void [chargerPartieLocale](#) (char \*nom\_fichier)
- void [afficheFichierLocale](#) (char \*nom\_fichier)

### 4.12.1 Detailed Description

Prototypes des fonctions qui gerent l'interface graphique du logiciel.

Author

Remi BERTHO

Date

09/03/14

Version

2.1.0

### 4.12.2 Enumeration Type Documentation

#### 4.12.2.1 enum Menu

Enumerator

***nouvPart***

***charPart***

***affFich***

***supprFich***

***listFich***

***nouvChem***

***lireChem***

***quit***

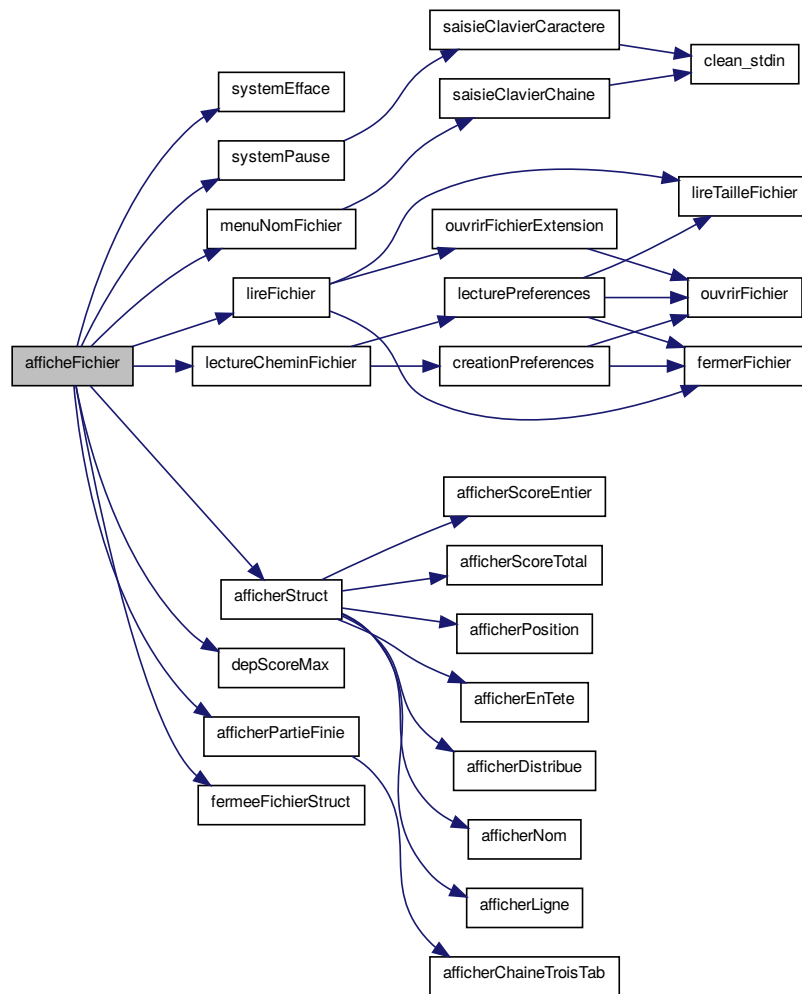
***easterEggs***

### 4.12.3 Function Documentation

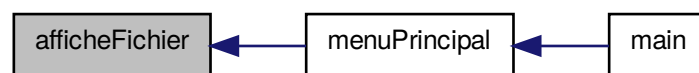
#### 4.12.3.1 void afficheFichier ( )

Demande le nom d'un fichier et l'affiche

Here is the call graph for this function:



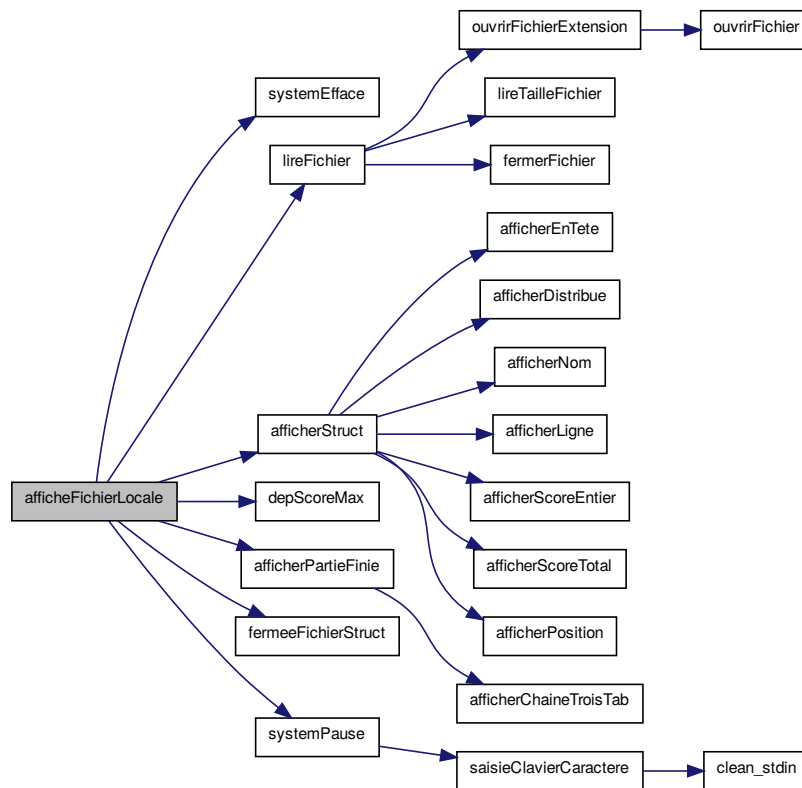
Here is the caller graph for this function:



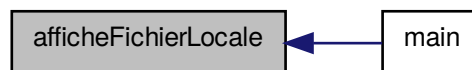
#### 4.12.3.2 void afficheFichierLocale ( char \* nom\_fichier )

Affiche le fichier dont le nom a été donné en paramètre

Here is the call graph for this function:



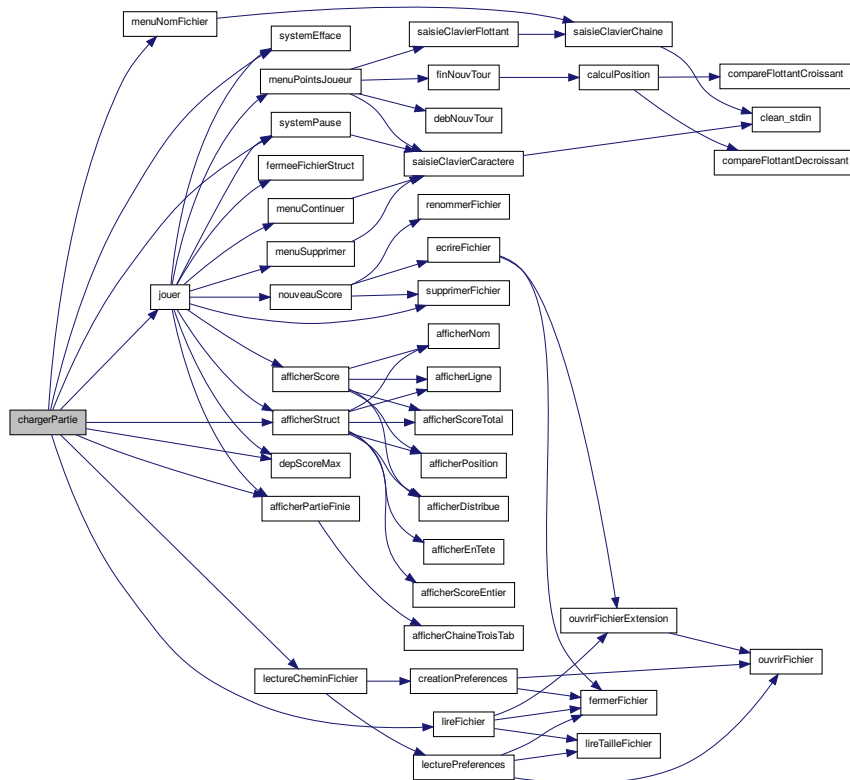
Here is the caller graph for this function:



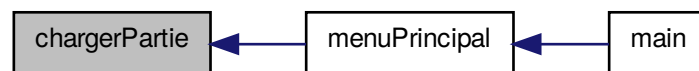
#### 4.12.3.3 void chargerPartie ( )

Charge une partie a partir d'un fichier et lance la fonction de comptage des points

Here is the call graph for this function:



Here is the caller graph for this function:



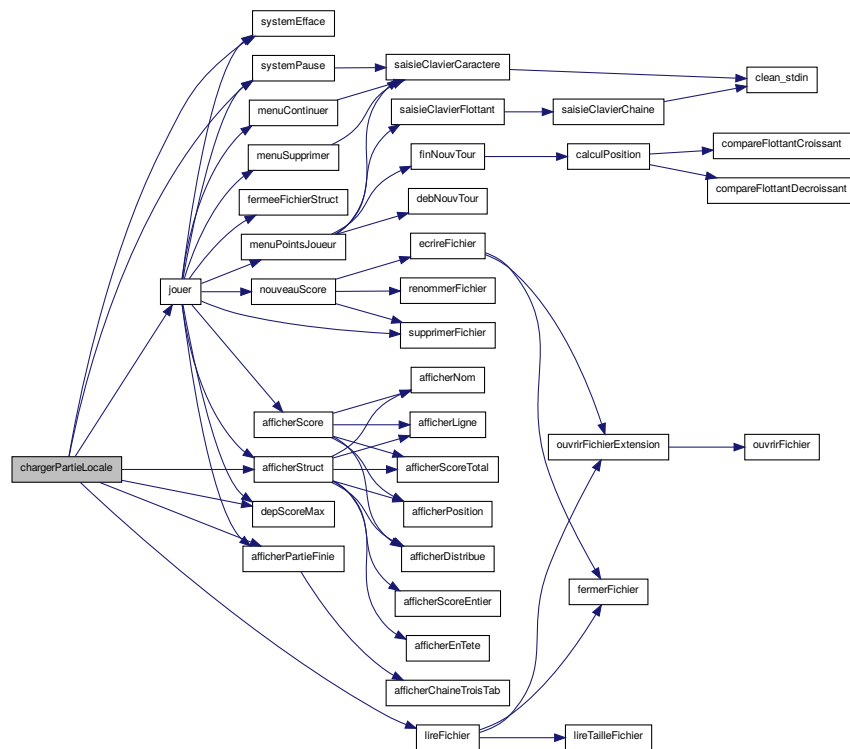
#### 4.12.3.4 void chargerPartieLocale ( char \* nom\_fichier )

Charge une partie a partir d'un fichier dont le nom a ete donne et lance la fonction de comptage des points

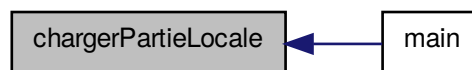
##### Parameters

in	nom_fichier,le	nom du fichier
----	----------------	----------------

Here is the call graph for this function:



Here is the caller graph for this function:



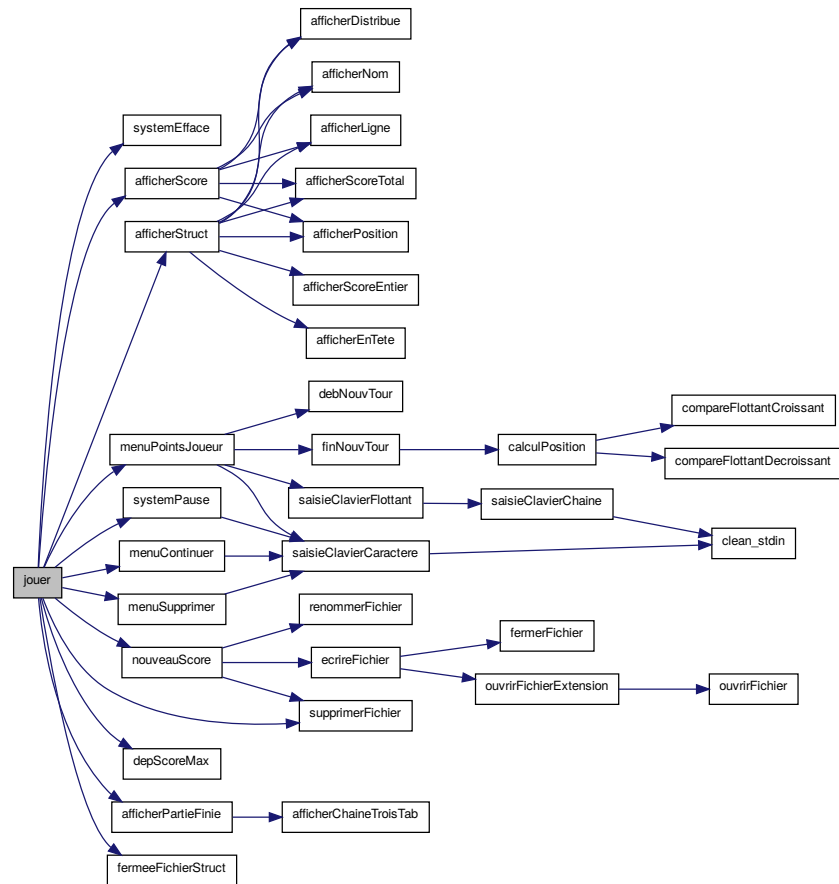
#### 4.12.3.5 void jouer ( Fichier\_Jeu \* ptr\_struct\_fichier, char \* nom\_fichier )

Lance la partie de comptage jusqu'à la fin

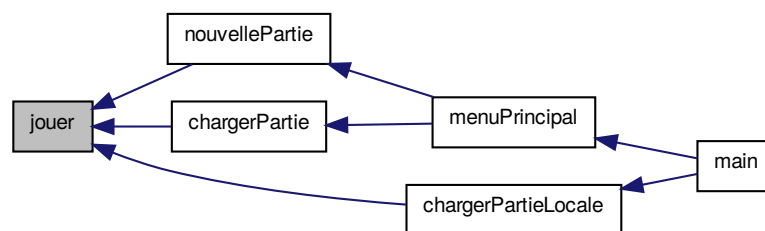
##### Parameters

in, out	*ptr_struct_fichier	la structure <a href="#">Fichier_Jeu</a>
in	*nom_fichier	le nom du fichier dans lequel on stocke les informations

Here is the call graph for this function:



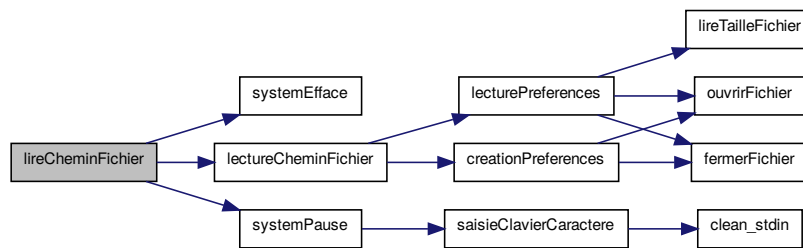
Here is the caller graph for this function:



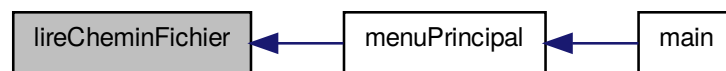
#### 4.12.3.6 void lireCheminFichier ( )

Lis le chemin de fichier et l'affiche

Here is the call graph for this function:



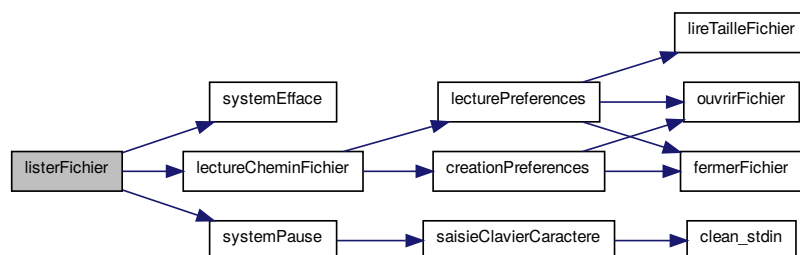
Here is the caller graph for this function:



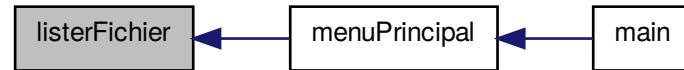
#### 4.12.3.7 void listerFichier ( )

Liste tout les fichier du dossier courant contenant l'extension .jeu

Here is the call graph for this function:



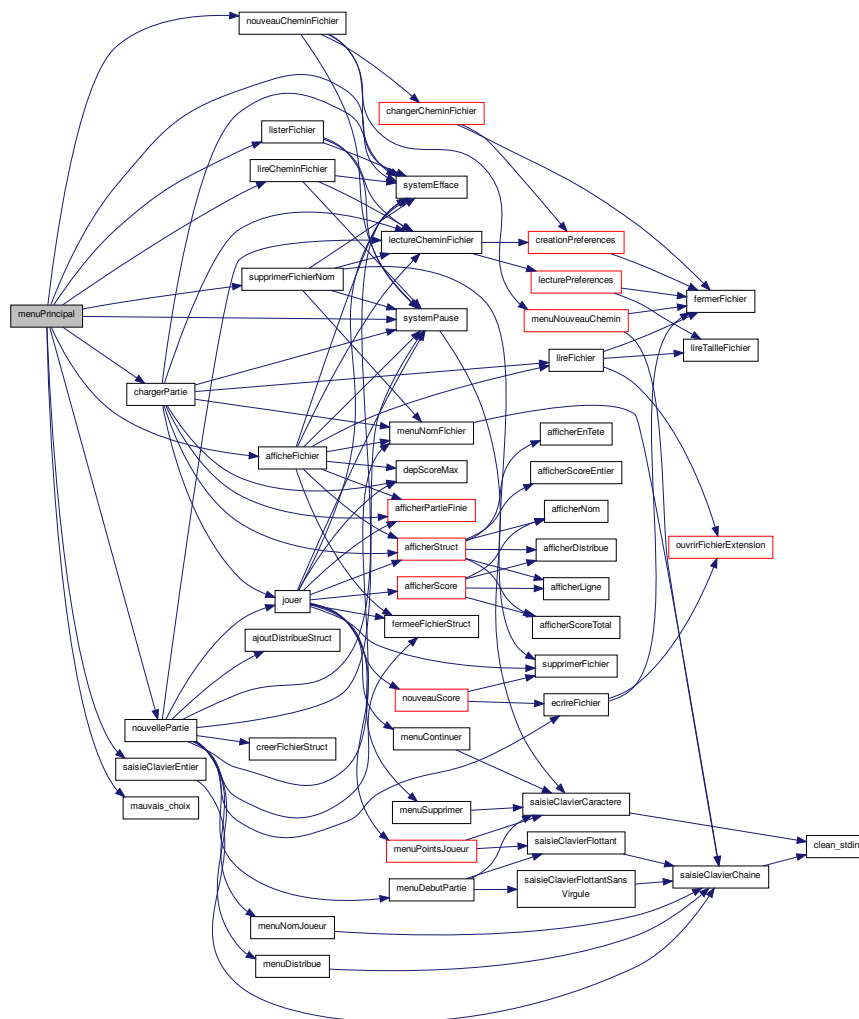
Here is the caller graph for this function:



#### 4.12.3.8 void menuPrincipal ( )

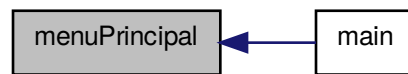
Lance un menu que redirige vers l'action que l'on veut effectuer

Here is the call graph for this function:





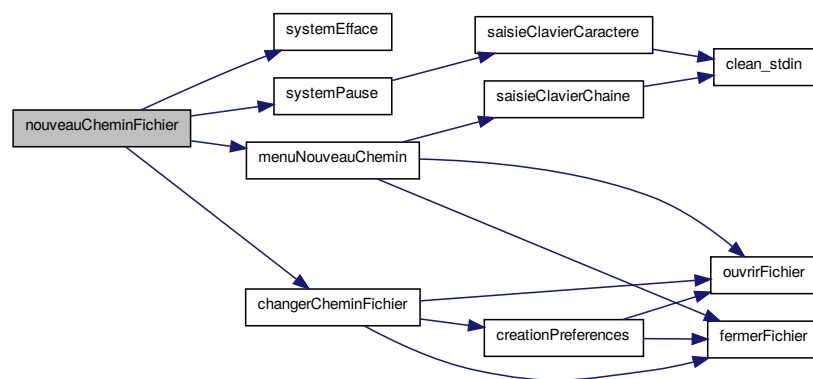
Here is the caller graph for this function:



#### 4.12.3.9 void nouveauCheminFichier ( )

Charge un nouveau chemin que l'on demande a l'utilisateur

Here is the call graph for this function:



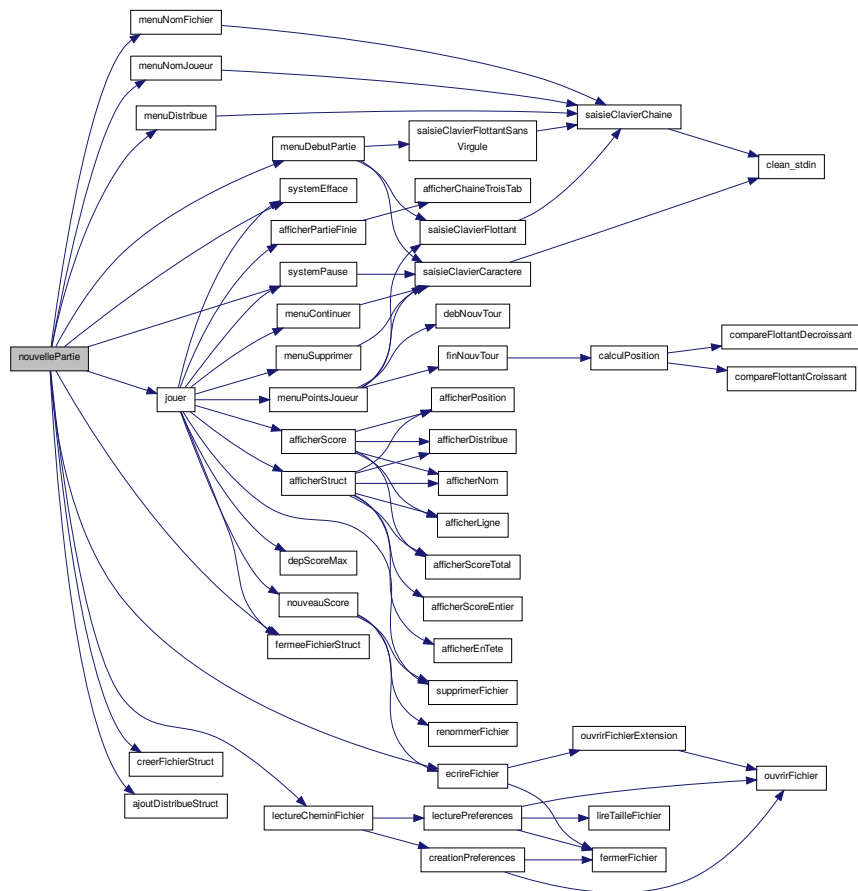
Here is the caller graph for this function:



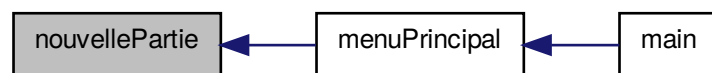
#### 4.12.3.10 void nouvellePartie ( )

Initialise une nouvelle partie et lance la fonction de comptage des points

Here is the call graph for this function:



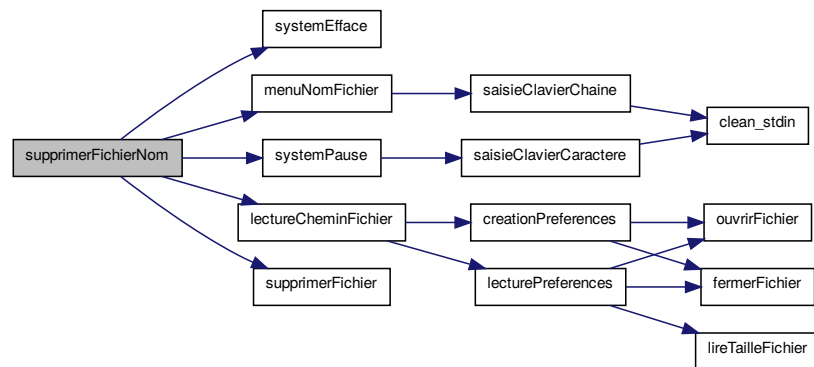
Here is the caller graph for this function:



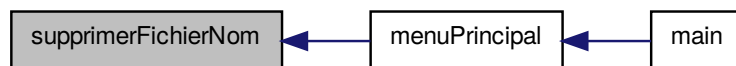
#### 4.12.3.11 void supprimerFichierNom ( )

Demande le nom d'un fichier et le supprime

Here is the call graph for this function:



Here is the caller graph for this function:

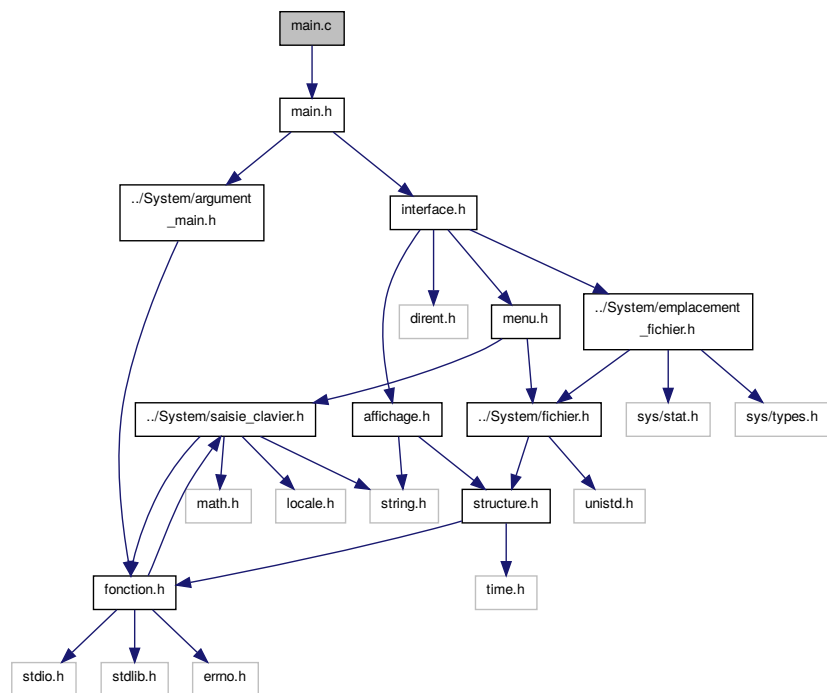


## 4.13 main.c File Reference

Lancement du programme.

```
#include "main.h"
```

Include dependency graph for main.c:



## Functions

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

### 4.13.1 Detailed Description

Lancement du programme.

Author

Remi BERTHO

Date

09/03/14

Version

2.1.0

### 4.13.2 Function Documentation

#### 4.13.2.1 int main ( int argc, char \* argv[] )

Lance le programme

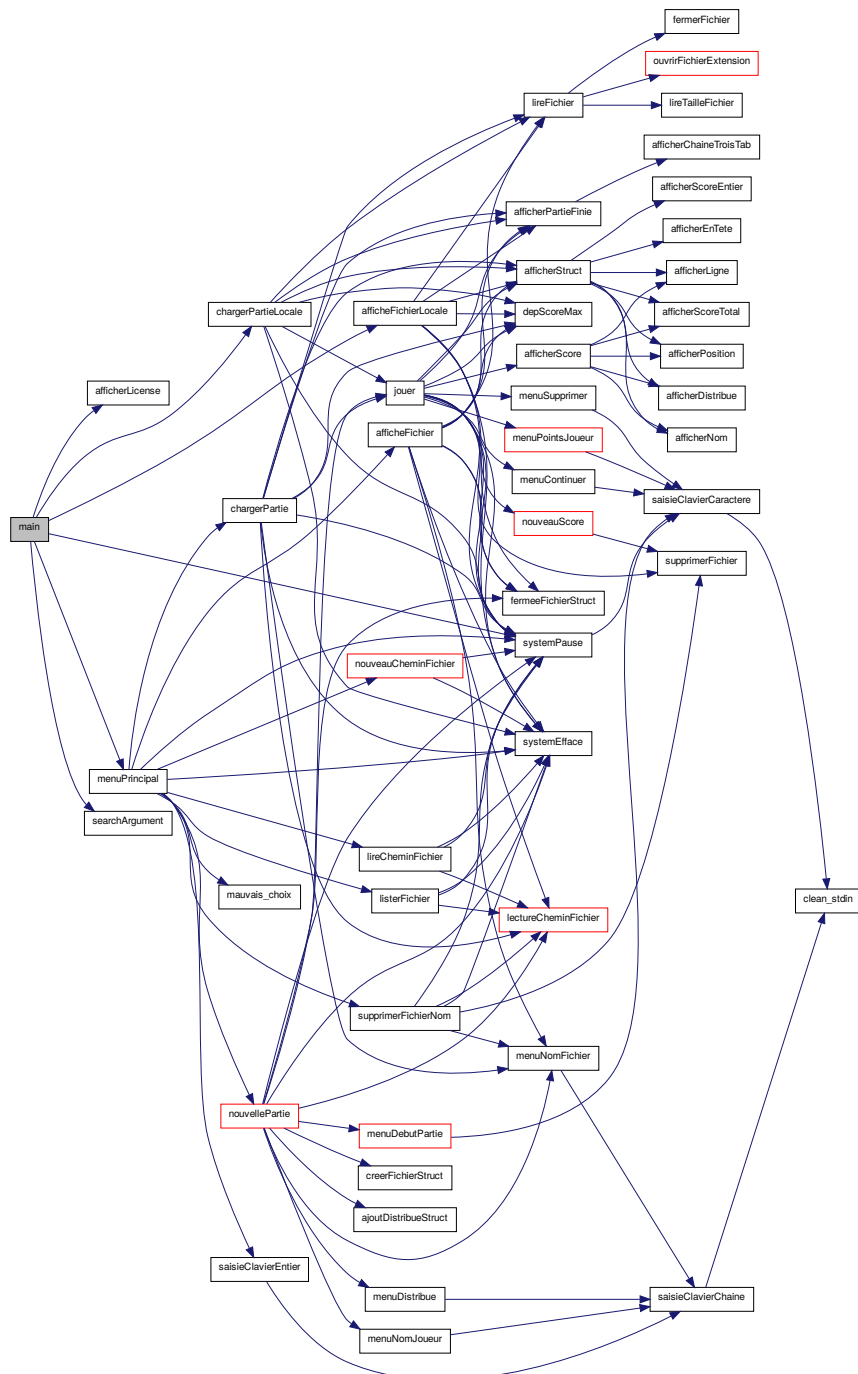
### Parameters

in	<i>argc</i>	le nombre d'argument
in	<i>argv</i>	le tableau des arguments

## Returns

0 si tout s'est bien passé

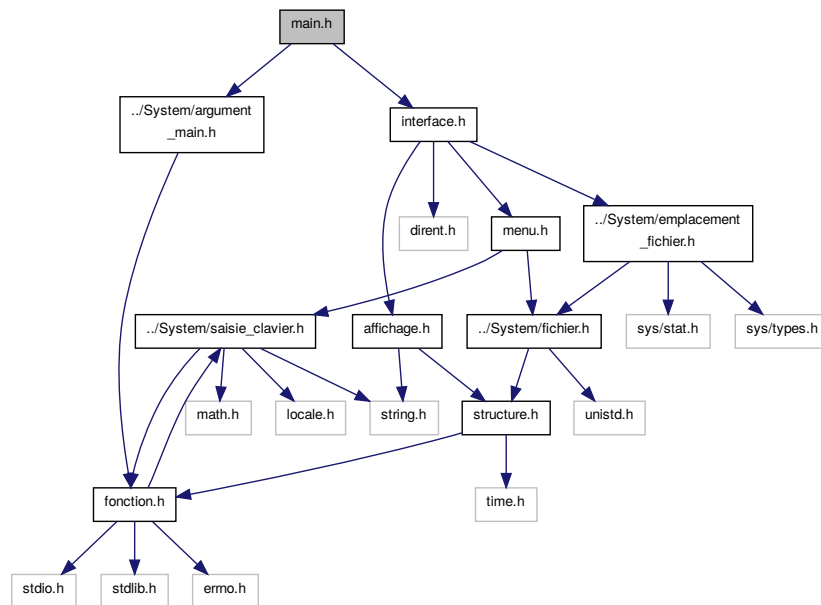
Here is the call graph for this function:



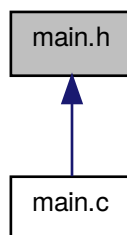
## 4.14 main.h File Reference

Lancement du programme.

```
#include "interface.h"
#include "../System/argument_main.h"
Include dependency graph for main.h:
```



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



## Functions

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

### 4.14.1 Detailed Description

Lancement du programme.

#### Author

Remi BERTHO

#### Date

09/03/14

#### Version

2.1.0

### 4.14.2 Function Documentation

#### 4.14.2.1 int main ( int *argc*, char \* *argv*[ ] )

Lance le programme

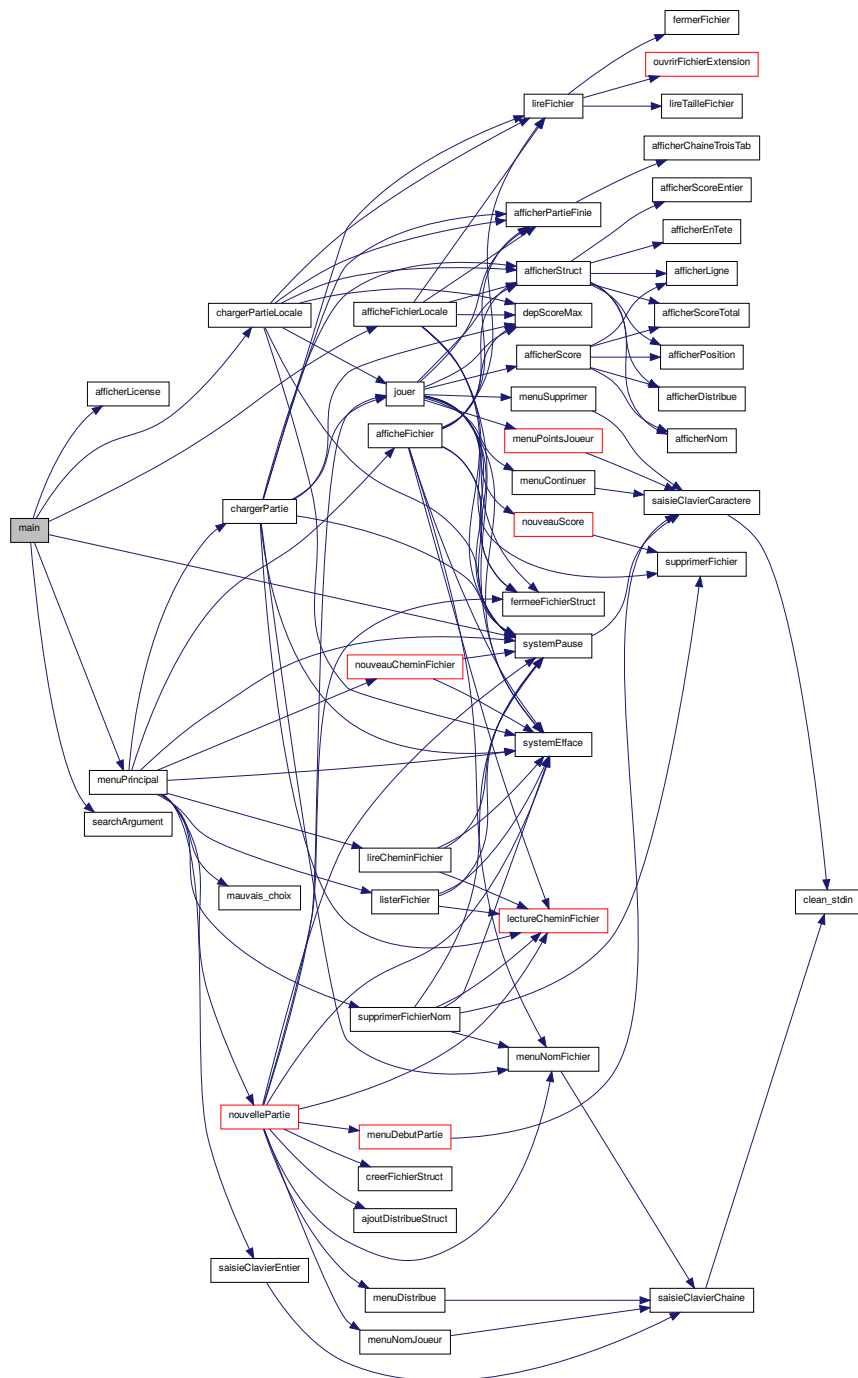
#### Parameters

in	<i>argc</i>	le nombre d 'argument
in	<i>argv</i>	le tableau des arguments

#### Returns

0 si tout s'est bien passe

Here is the call graph for this function:



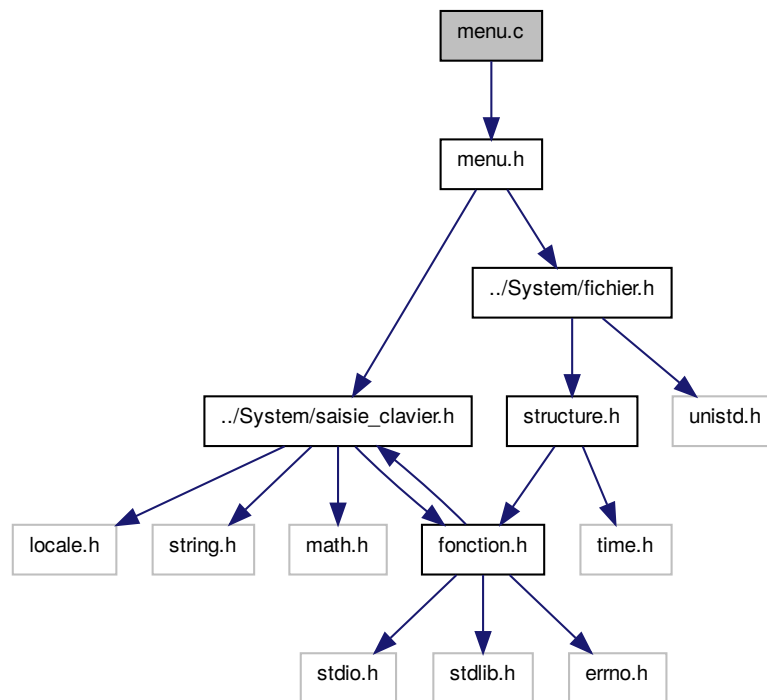
## 4.15 menu.c File Reference

Fonctions qui gerent les menus demandant de rentrer des valeurs du logiciel.



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

Include dependency graph for menu.c:



## Functions

- char \* [menuNomFichier](#) (char nom\_fichier[TAILLE\_MAX\_NOM\_FICHIER])
- void [menuDebutPartie](#) (float \*ptr\_nb\_joueur, float \*ptr\_nb\_max, char \*ptr\_sens\_premier)
- void [menuDistribue](#) (char \*nom\_distribue)
- void [menuNomJoueur](#) (Fichier\_Jeu \*ptr\_struct\_fichier)
- void [menuPointsJoueur](#) (Fichier\_Jeu \*ptr\_struct\_fichier)
- int [menuContinuer](#) ()
- int [menuSupprimer](#) ()
- void [menuNouveauChemin](#) (char \*nouveauChemin)

### 4.15.1 Detailed Description

Fonctions qui gerent les menus demandant de rentrer des valeurs du logiciel.

#### Author

Remi BERTHO

#### Date

09/03/14

#### Version

2.1.0

## 4.15.2 Function Documentation

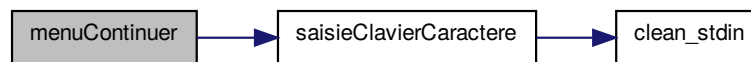
### 4.15.2.1 int menuContinuer ( )

Demande si l'on veut continuer ou pas et l'enregistre dans la variable arret

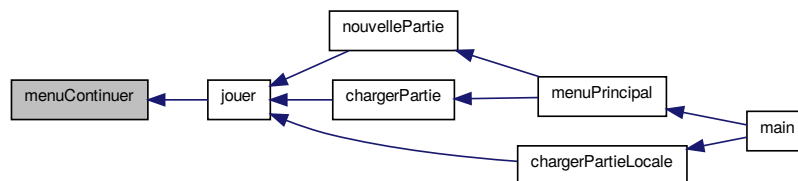
#### Returns

VRAI si l'on veut continuer, FAUX sinon

Here is the call graph for this function:



Here is the caller graph for this function:



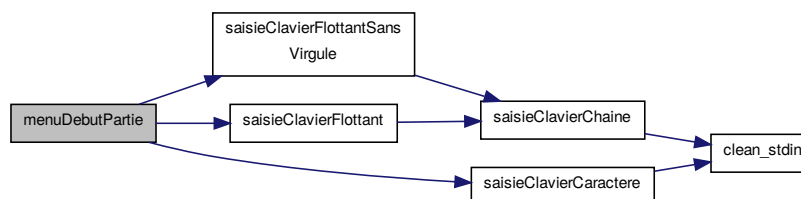
### 4.15.2.2 void menuDebutPartie ( float \* ptr\_nb\_joueur, float \* ptr\_nb\_max, char \* ptr\_sens\_premier )

Demande et enregistre le nombre de joueur, le nombre maximum et le nom de la personne qui comme a distribuer

#### Parameters

in, out	*ptr_nb_joueur	le nombre de joueur
in, out	*ptr_nb_max	le nombre maximum
in, out	*ptr_sens_premier	definit le sens du premier

Here is the call graph for this function:



Here is the caller graph for this function:



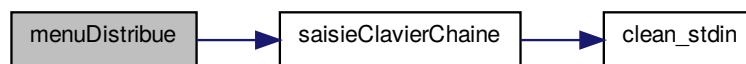
#### 4.15.2.3 void menuDistribue ( char \* *nom\_distribue* )

Demande et enregistre le nom de la personne qui comme a distribuer

##### Parameters

<i>in, out</i>	<i>*nom_distribue</i>	le nom de la personne qui commence a distribuer
----------------	-----------------------	---

Here is the call graph for this function:



Here is the caller graph for this function:



#### 4.15.2.4 char \* menuNomFichier ( char *nom\_fichier*[TAILLE\_MAX\_NOM\_FICHIER] )

Demande et enregistre le nom du fichier

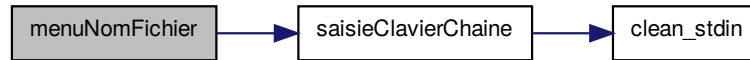
##### Parameters

<i>in, out</i>	<i>nom_fichier</i>	la chaine de caractere contenant le nom du fichier
----------------	--------------------	--

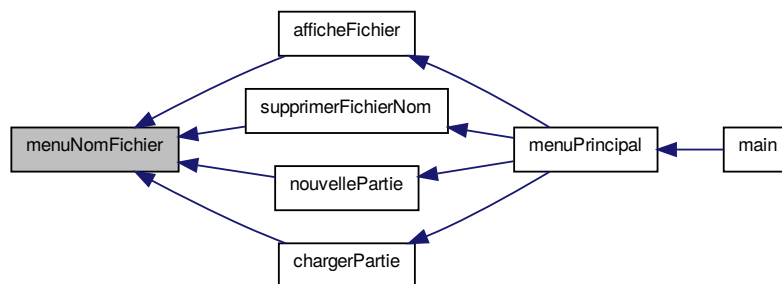
**Returns**

la chaine de caractere contenant le nom du fichier

Here is the call graph for this function:



Here is the caller graph for this function:



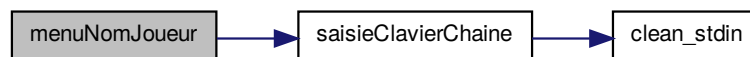
#### 4.15.2.5 void menuNomJoueur ( Fichier\_Jeu \* ptr\_struct\_fichier )

Demande et enregistre le nom des joueurs

**Parameters**

in, out	<i>*ptr_struct_fichier</i>	la structure <a href="#">Fichier_Jeu</a> ou l'on veut enregistrer le nom des joueurs
---------	----------------------------	--

Here is the call graph for this function:



Here is the caller graph for this function:



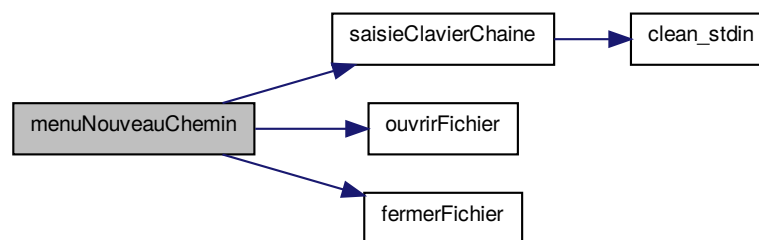
#### 4.15.2.6 void menuNouveauChemin ( char \* *nouveauChemin* )

Demande et enregistre le nouveau chemin

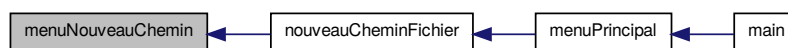
##### Parameters

<code>in, out</code>	<code>*nouveau-Chemin</code>	le nouveau chemin
----------------------	------------------------------	-------------------

Here is the call graph for this function:



Here is the caller graph for this function:



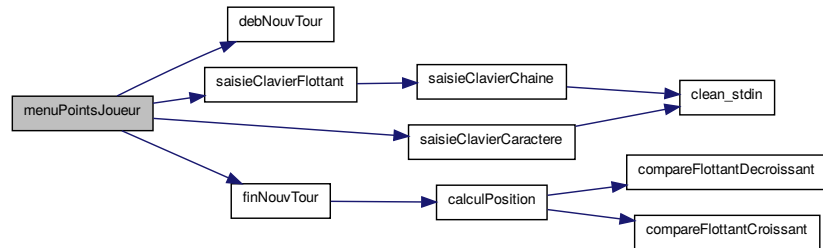
#### 4.15.2.7 void menuPointsJoueur ( Fichier\_Jeu \* *ptr\_struct\_fichier* )

Debut un nouveau tour, demande et enregistre les points et fini le tour

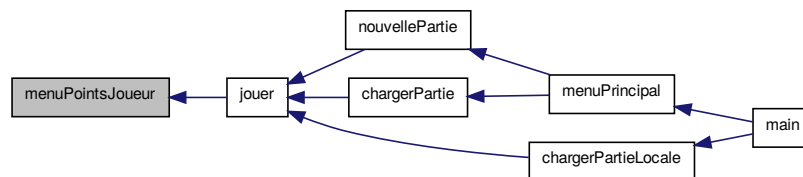
##### Parameters

in, out	<i>*ptr_struct_ - fichier</i>	la structure <a href="#">Fichier_Jeu</a> ou l'on veut faire un nouveau tour
---------	-----------------------------------	---

Here is the call graph for this function:



Here is the caller graph for this function:



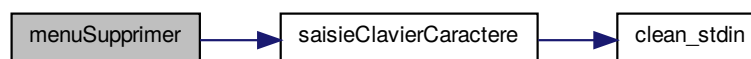
#### 4.15.2.8 int menuSupprimer ( )

Demande si l'on veut supprimer le fichier ou pas et l'enregistre dans la variable `suppr`

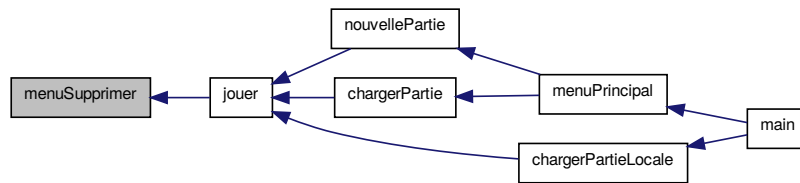
##### Returns

VRAI si l'on veut supprimer le fichier, FAUX sinon

Here is the call graph for this function:



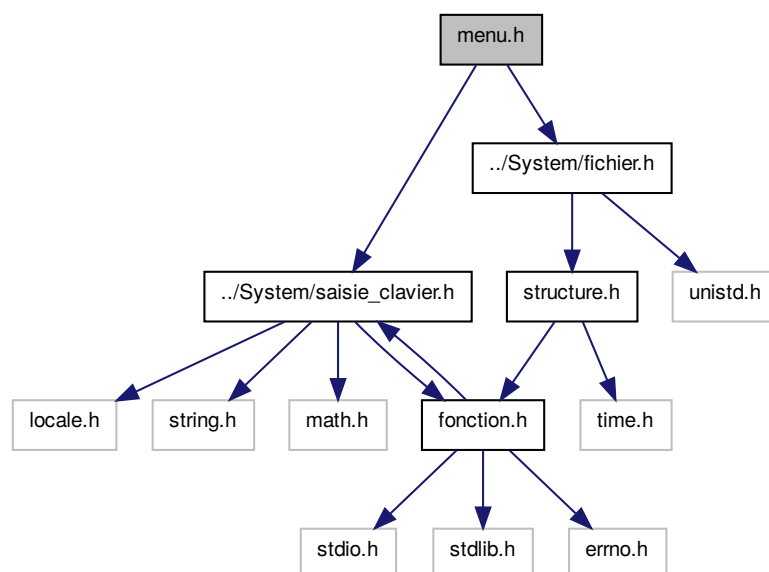
Here is the caller graph for this function:



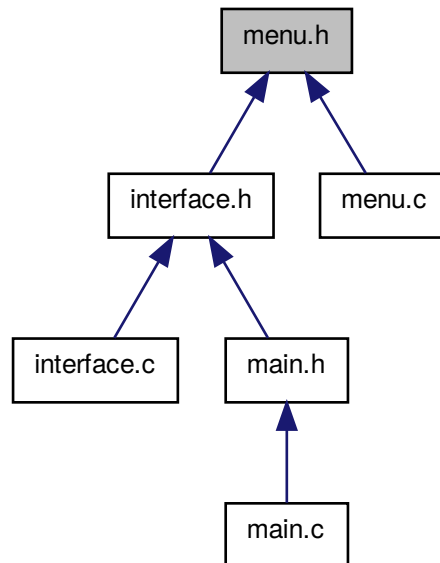
## 4.16 menu.h File Reference

Prototypes des fonctions qui gerent les menus demandant de rentrer des valeurs du logiciel.

```
#include "../System/saisie_clavier.h"
#include "../System/fichier.h"
Include dependency graph for menu.h:
```



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



## Functions

- char \* [menuNomFichier](#) (char nom\_fichier[TAILLE\_MAX\_NOM\_FICHIER])
- void [menuDebutPartie](#) (float \*ptr\_nb\_joueur, float \*ptr\_nb\_max, char \*ptr\_sens\_premier)
- void [menuDistribue](#) (char \*nom\_distribue)
- void [menuNomJoueur](#) (Fichier\_Jeu \*ptr\_struct\_fichier)
- void [menuPointsJoueur](#) (Fichier\_Jeu \*ptr\_struct\_fichier)
- int [menuContinuer](#) ()
- int [menuSupprimer](#) ()
- void [menuNouveauChemin](#) (char \*nouveauChemin)

### 4.16.1 Detailed Description

Prototypes des fonctions qui gerent les menus demandant de rentrer des valeurs du logiciel.

#### Author

Remi BERTHO

#### Date

13/02/14

#### Version

2.0



## 4.16.2 Function Documentation

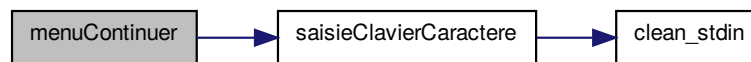
### 4.16.2.1 int menuContinuer ( )

Demande si l'on veut continuer ou pas et l'enregistre dans la variable arret

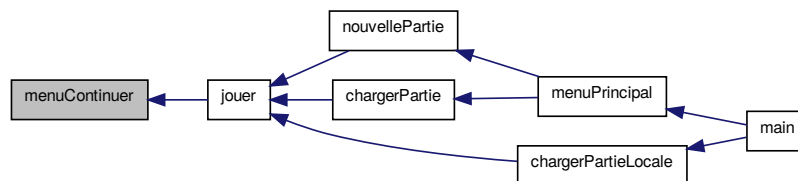
#### Returns

VRAI si l'on veut continuer, FAUX sinon

Here is the call graph for this function:



Here is the caller graph for this function:



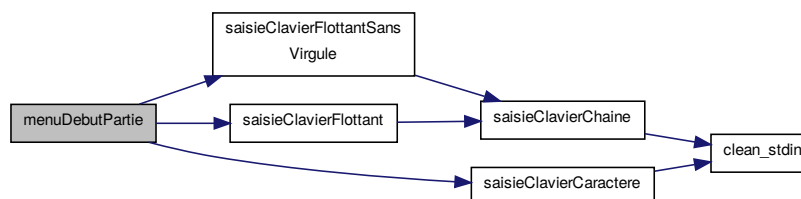
### 4.16.2.2 void menuDebutPartie ( float \* ptr\_nb\_joueur, float \* ptr\_nb\_max, char \* ptr\_sens\_premier )

Demande et enregistre le nombre de joueur, le nombre maximum et le nom de la personne qui comme a distribuer

#### Parameters

in, out	<i>*ptr_nb_joueur</i>	le nombre de joueur
in, out	<i>*ptr_nb_max</i>	le nombre maximum
in, out	<i>*ptr_sens_premier</i>	definit le sens du premier

Here is the call graph for this function:



Here is the caller graph for this function:



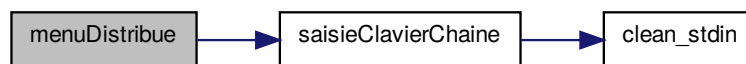
#### 4.16.2.3 void menuDistribue ( char \* nom\_distribue )

Demande et enregistre le nom de la personne qui comme a distribuer

##### Parameters

<code>in, out</code>	<code>*nom_distribue</code>	le nom de la personne qui commence a distribuer
----------------------	-----------------------------	---

Here is the call graph for this function:



Here is the caller graph for this function:



#### 4.16.2.4 char\* menuNomFichier ( char nom\_fichier[TAILLE\_MAX\_NOM\_FICHIER] )

Demande et enregistre le nom du fichier

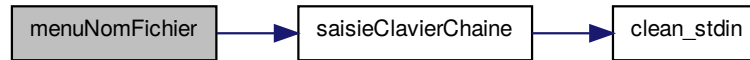
##### Parameters

<code>in, out</code>	<code>nom_fichier</code>	la chaine de caractere contenant le nom du fichier
----------------------	--------------------------	--

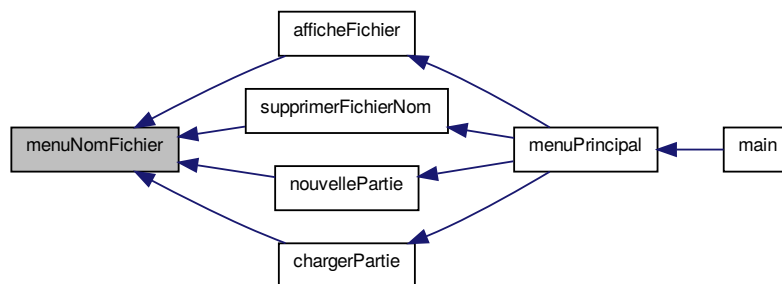
## Returns

la chaine de caractere contenant le nom du fichier

Here is the call graph for this function:



Here is the caller graph for this function:



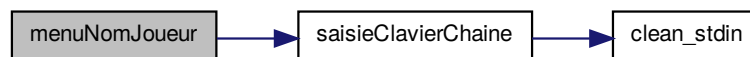
#### 4.16.2.5 void menuNomJoueur ( Fichier\_Jeu \* ptr\_struct\_fichier )

Demande et enregistre le nom des joueurs

## Parameters

<code>in, out</code>	<code>*ptr_struct_fichier</code>	la structure <a href="#">Fichier_Jeu</a> ou l'on veut enregistrer le nom des joueurs
----------------------	----------------------------------	--

Here is the call graph for this function:



Here is the caller graph for this function:



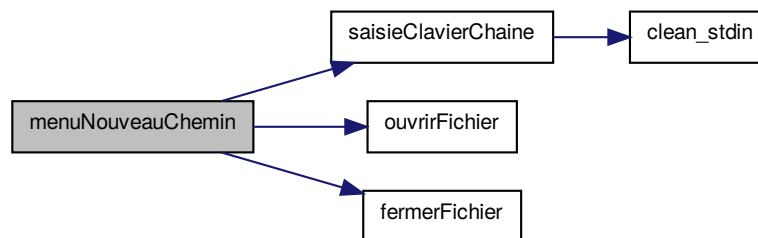
#### 4.16.2.6 void menuNouveauChemin ( char \* *nouveauChemin* )

Demande et enregistre le nouveau chemin

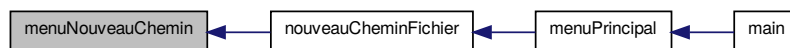
##### Parameters

<code>in, out</code>	<code>*nouveau-Chemin</code>	le nouveau chemin
----------------------	------------------------------	-------------------

Here is the call graph for this function:



Here is the caller graph for this function:



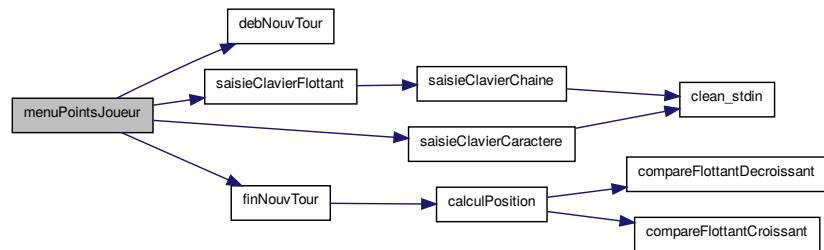
#### 4.16.2.7 void menuPointsJoueur ( Fichier\_Jeu \* *ptr\_struct\_fichier* )

Debut un nouveau tour, demande et enregistre les points et fini le tour

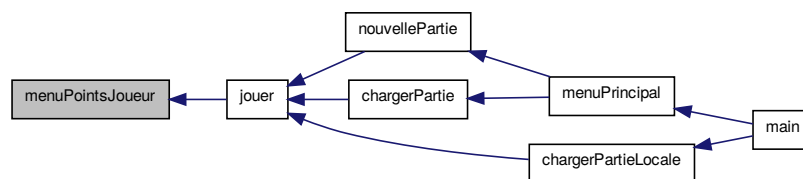
##### Parameters

<code>in, out</code>	<code>*ptr_struct_ - fichier</code>	la structure <code>Fichier_Jeu</code> ou l'on veut faire un nouveau tour
----------------------	-------------------------------------	--

Here is the call graph for this function:



Here is the caller graph for this function:



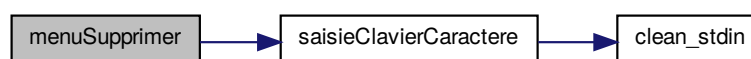
#### 4.16.2.8 int menuSupprimer ( )

Demande si l'on veut supprimer le fichier ou pas et l'enregistre dans la variable `suppr`

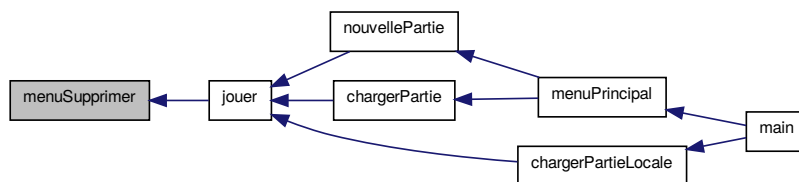
##### Returns

VRAI si l'on veut supprimer le fichier, FAUX sinon

Here is the call graph for this function:



Here is the caller graph for this function:

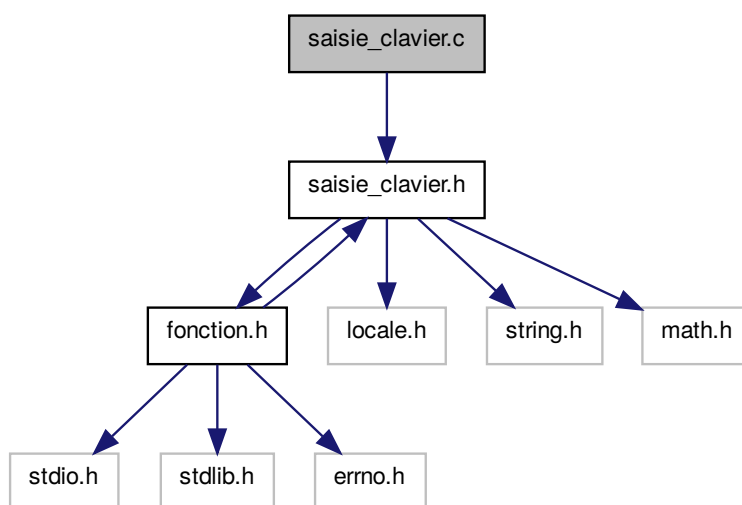


## 4.17 saisie\_clavier.c File Reference

Fonctions de saisie clavier.

```
#include "saisie_clavier.h"
```

Include dependency graph for saisie\_clavier.c:



## Functions

- void [clean\\_stdin](#) (void)
- char \* [saisieClavierChaine](#) (char \*chaine, int nb\_caract\_plus\_un)
- void [saisieClavierEntier](#) (int \*nb)
- void [saisieClavierFlottant](#) (float \*nb)
- void [saisieClavierFlottantSansVirgule](#) (float \*nb)
- void [saisieClavierDouble](#) (double \*nb)
- char \* [saisieClavierCaractere](#) (char \*c)

### 4.17.1 Detailed Description

Fonctions de saisie clavier.

Author

Remi BERTHO

Date

13/02/14

Version

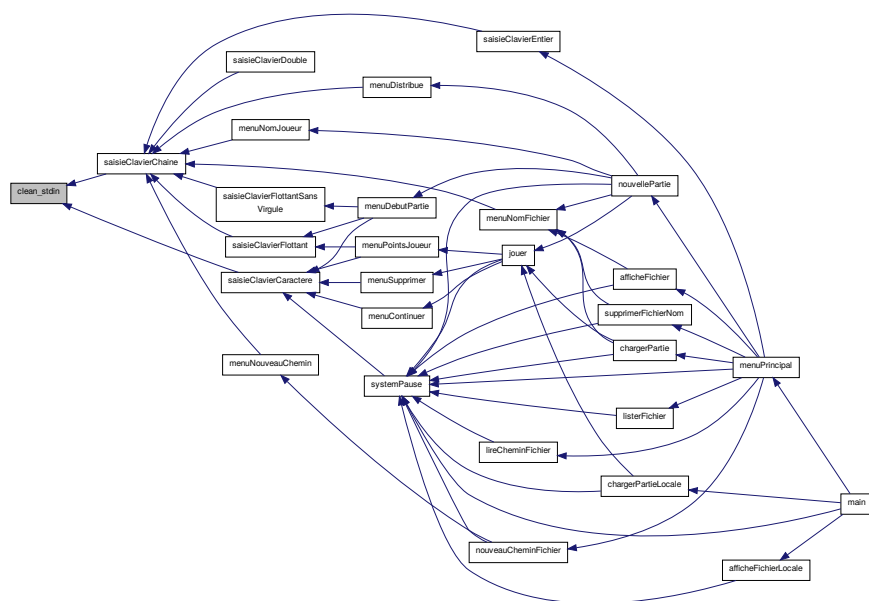
2.0

### 4.17.2 Function Documentation

#### 4.17.2.1 void clean\_stdin ( void )

Vide la cache de stdin.

Here is the caller graph for this function:



#### 4.17.2.2 char \* saisieClavierCaractere ( char \* c )

Fait une saisie clavier d'un caractere.

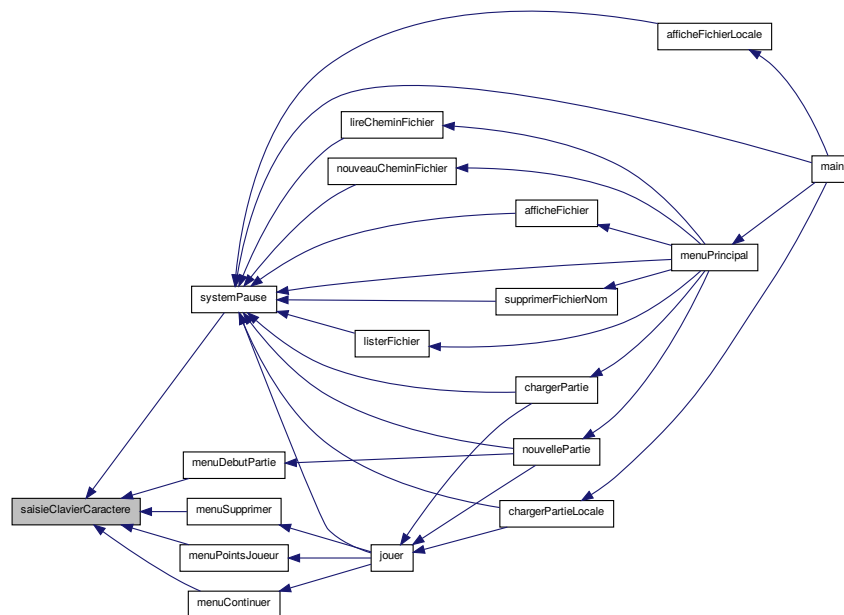
Parameters

in, out	*c	le caractere que l'on veut saisir
---------	----	-----------------------------------

Here is the call graph for this function:



Here is the caller graph for this function:



#### 4.17.2.3 char \* saisieClavierChaine ( char \* chaine, int nb\_caract\_plus\_un )

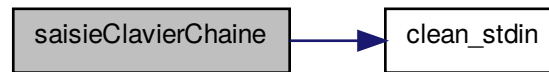
Fait une saisie clavier de chaine de caractere au clavier dans chaine sur nb-caract\_plus\_un moins 1 caractere

##### Parameters

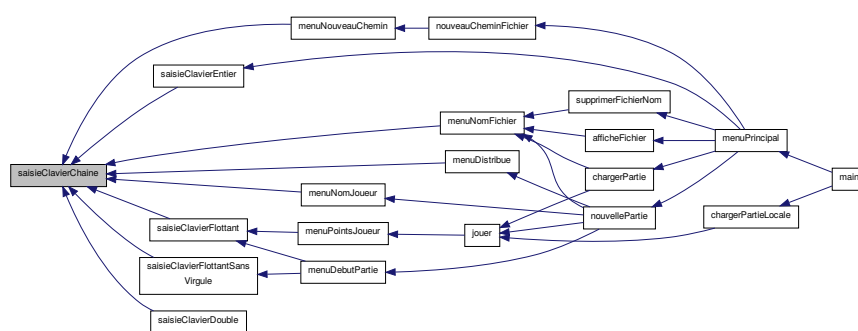
in, out	*chaine	une chaine de caractere
in	nb_caract_plus_un	le nombre de caractere moins un que la fonction va lire



Here is the call graph for this function:



Here is the caller graph for this function:



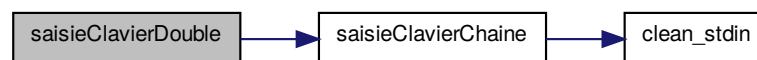
#### 4.17.2.4 void saisieClavierDouble ( double \* nb )

Fait une saisie clavier d'un double.

Parameters

in, out	*nb	le nombre que l'on veut saisir.
---------	-----	---------------------------------

Here is the call graph for this function:



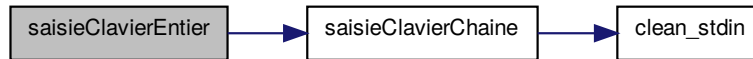
#### 4.17.2.5 void \* saisieClavierEntier ( int \* nb )

Fait une saisie clavier d'un entier.

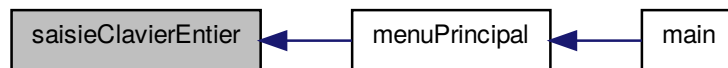
## Parameters

<code>in, out</code>	<code>*nb</code>	le nombre que l'on veut saisir.
----------------------	------------------	---------------------------------

Here is the call graph for this function:



Here is the caller graph for this function:



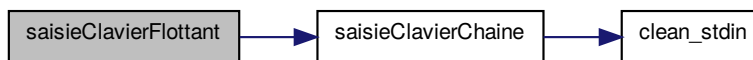
#### 4.17.2.6 void saisieClavierFlottant ( float \* nb )

Fait une saisie clavier d'un flottant.

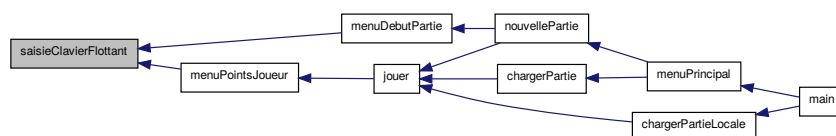
## Parameters

<code>in, out</code>	<code>*nb</code>	le nombre que l'on veut saisir.
----------------------	------------------	---------------------------------

Here is the call graph for this function:



Here is the caller graph for this function:



4.17.2.7 void saisieClavierFlottantSansVirgule ( float \* nb )

Fait une saisie clavier d'un flottant.

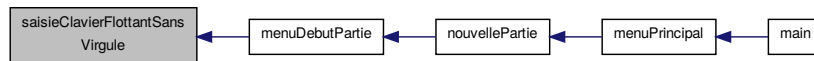
## Parameters

<code>in, out</code>	<code>*nb</code>	le nombre que l'on veut saisir.
----------------------	------------------	---------------------------------

Here is the call graph for this function:



Here is the caller graph for this function:

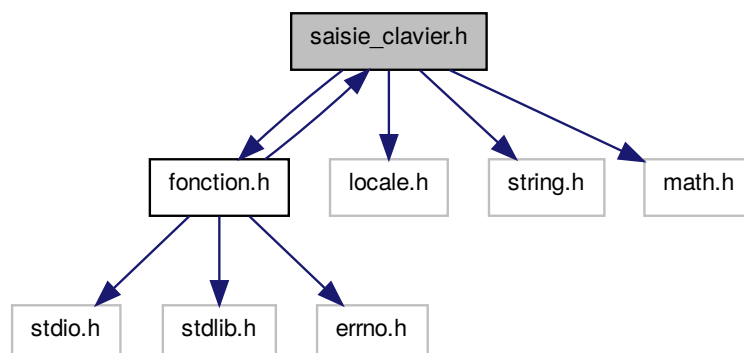


## 4.18 saisie\_clavier.h File Reference

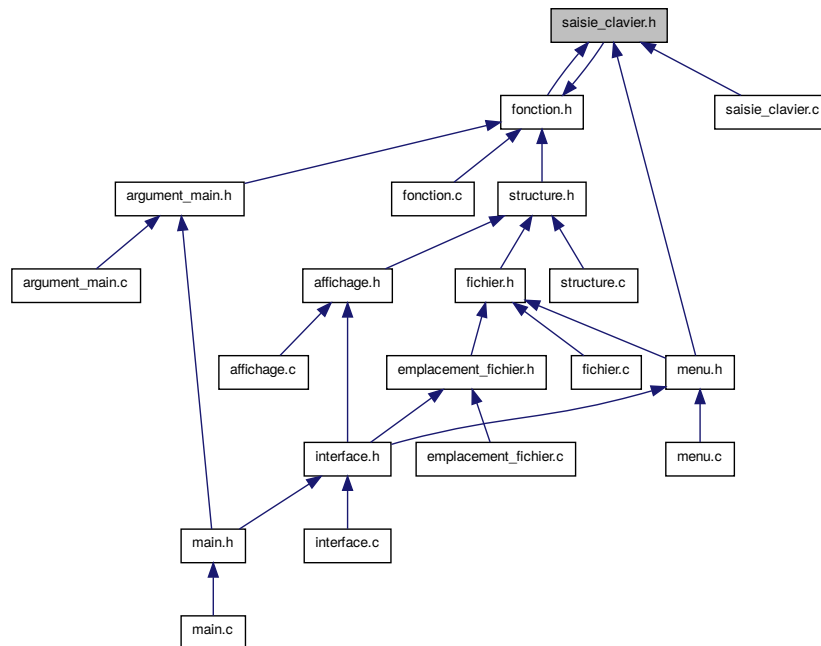
Prototypes des fonctions de saisie clavier.

```
#include "fonction.h"
#include <locale.h>
#include <string.h>
#include <math.h>
```

Include dependency graph for `saisie_clavier.h`:



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



## Macros

- #define `NB_CARACT_INT` 12
- #define `NB_CARACT_FLOT` 39
- #define `NB_CARACT_DOUB` 309

## Functions

- void `clean_stdin` (void)
- char \* `saisieClavierChaine` (char \*chaine, int nb\_caract\_plus\_un)
- void `saisieClavierEntier` (int \*nb)
- void `saisieClavierFlottant` (float \*nb)
- void `saisieClavierFlottantSansVirgule` (float \*nb)
- void `saisieClavierDouble` (double \*nb)
- char \* `saisieClavierCaractere` (char \*c)

### 4.18.1 Detailed Description

Prototypes des fonctions de saisie clavier.

Author

Remi BERTHO

Date

13/02/14



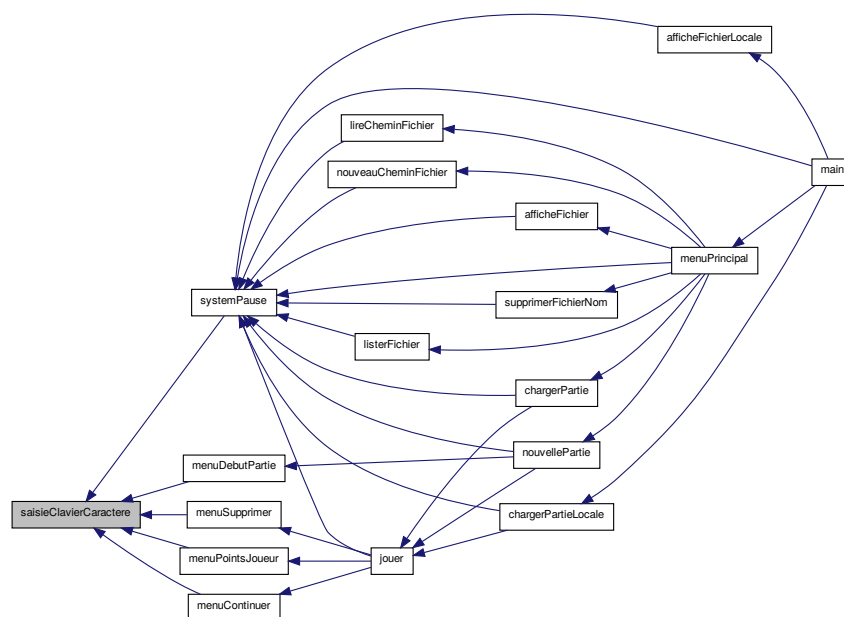
## Parameters

in, out	*c	le caractere que l'on veut saisir
---------	----	-----------------------------------

Here is the call graph for this function:



Here is the caller graph for this function:



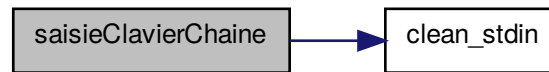
#### 4.18.3.3 char\* saisieClavierChaine ( char \* chaine, int nb\_caract\_plus\_un )

Fait une saisie clavier de chaine de caractere au clavier dans chaine sur nb-caract\_plus\_un moins 1 caractere

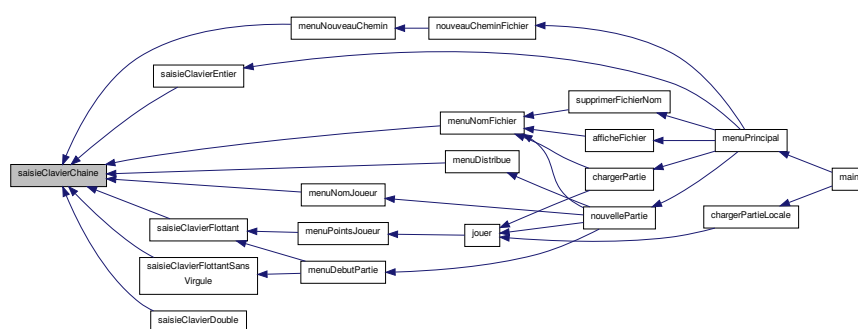
## Parameters

in, out	*chaine	une chaine de caractere
in	nb_caract_plus_un	le nombre de caractere moins un que la fonction va lire

Here is the call graph for this function:



Here is the caller graph for this function:



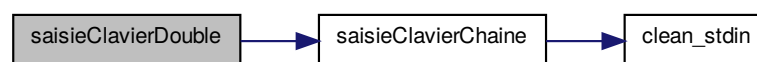
#### 4.18.3.4 void saisieClavierDouble ( double \* nb )

Fait une saisie clavier d'un double.

Parameters

in, out	*nb	le nombre que l'on veut saisir.
---------	-----	---------------------------------

Here is the call graph for this function:



#### 4.18.3.5 void saisieClavierEntier ( int \* nb )

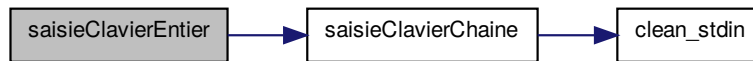
Fait une saisie clavier d'un entier.



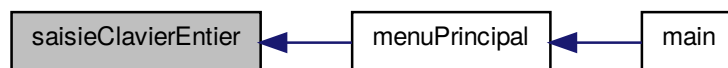
## Parameters

<code>in, out</code>	<code>*nb</code>	le nombre que l'on veut saisir.
----------------------	------------------	---------------------------------

Here is the call graph for this function:



Here is the caller graph for this function:



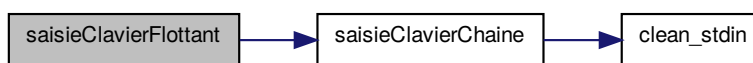
## 4.18.3.6 void saisieClavierFlottant ( float \* nb )

Fait une saisie clavier d'un flottant.

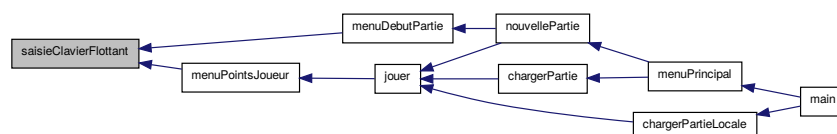
## Parameters

<code>in, out</code>	<code>*nb</code>	le nombre que l'on veut saisir.
----------------------	------------------	---------------------------------

Here is the call graph for this function:



Here is the caller graph for this function:



4.18.3.7 void saisieClavierFlottantSansVirgule ( float \* *nb* )

Fait une saisie clavier d'un flottant.

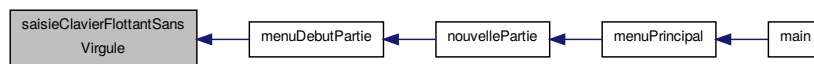
## Parameters

<code>in, out</code>	<code>*nb</code>	le nombre que l'on veut saisir.
----------------------	------------------	---------------------------------

Here is the call graph for this function:



Here is the caller graph for this function:

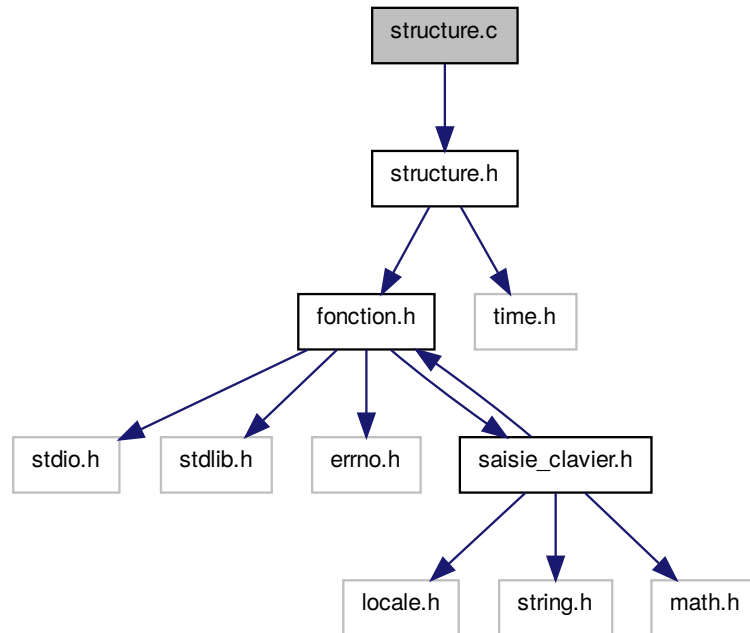


## 4.19 structure.c File Reference

Fonction de gestion des fichiers de la struction contenant les informations.

```
#include "structure.h"
```

Include dependency graph for structure.c:



## Functions

- `Fichier_Jeu * creerFichierStruct` (float nb\_joueur, float nb\_max, char sens\_premier)
- void `fermeeFichierStruct` (`Fichier_Jeu *ptr_struct_fichier`)
- void `debNouvTour` (`Fichier_Jeu *ptr_struct_fichier`)
- void `finNouvTour` (`Fichier_Jeu *ptr_struct_fichier`)
- void `calculPosition` (`Fichier_Jeu *ptr_struct_fichier`)
- void `ajoutDistribueStruct` (`Fichier_Jeu *ptr_struct_fichier`, char \*nom\_distribue)
- int `depScoreMax` (`Fichier_Jeu *ptr_struct_fichier`)

### 4.19.1 Detailed Description

Fonction de gestion des fichiers de la struction contenant les informations.

#### Author

Remi BERTHO

#### Date

09/03/14

#### Version

2.1.0

## 4.19.2 Function Documentation

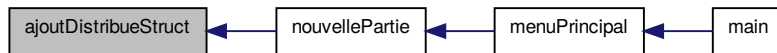
### 4.19.2.1 void ajoutDistribueStruct ( Fichier\_Jeu \* *ptr\_struct\_fichier*, char \* *nom\_distribue* )

Ajoute la personne qui distribue dans la structure.

## Parameters

in	<i>*nom_distribue</i>	le nom de la personne qui distribue
in	<i>*ptr_struct_fichier</i>	la structure du fichier avec lequel on veut mettre la personne qui distribue

Here is the caller graph for this function:



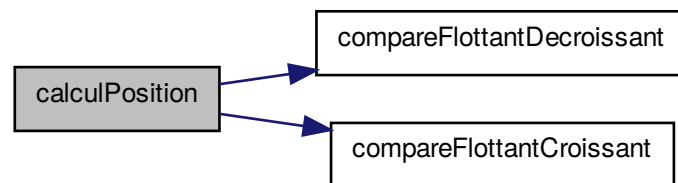
#### 4.19.2.2 void calculPosition ( Fichier\_Jeu \* ptr\_struct\_fichier )

Calcule les positions des joueurs

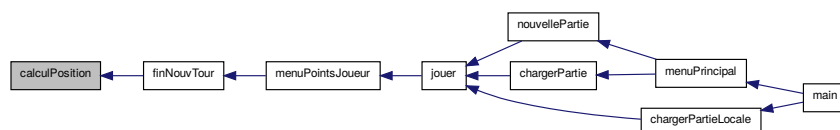
## Parameters

in, out	<i>*ptr_struct_fichier</i>	un pointeur sur la structure <a href="#">Fichier_Jeu</a>
---------	----------------------------	--

Here is the call graph for this function:



Here is the caller graph for this function:



#### 4.19.2.3 Fichier\_Jeu \* creerFichierStruct ( float nb\_joueur, float nb\_max, char sens\_premier )

Cree une structure Fichier\_jeu a partir des donnees de la fonction.

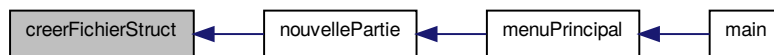
## Parameters

in	<i>nb_joueur</i>	le nombre de joueur
in	<i>nb_max</i>	le nombre maximum de points
in	<i>sens_premier</i>	permet de savoir dans quelle sens est calculer le premier

## Returns

un pointeur sur le structure [Fichier\\_Jeu](#) cree

Here is the caller graph for this function:



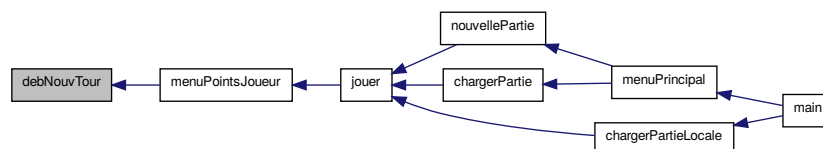
## 4.19.2.4 void debNouvTour ( Fichier\_Jeu \* ptr\_struct\_fichier )

Realloue l'espace memoire dedie au points sur la structure [Fichier\\_Jeu](#) mis en parametre

## Parameters

in, out	<i>*ptr_struct_fichier</i>	un pointeur sur la structure <a href="#">Fichier_Jeu</a>
---------	----------------------------	--

Here is the caller graph for this function:



## 4.19.2.5 int depScoreMax ( Fichier\_Jeu \* ptr\_struct\_fichier )

Verifie si quelqu'un a depasse le score maximum

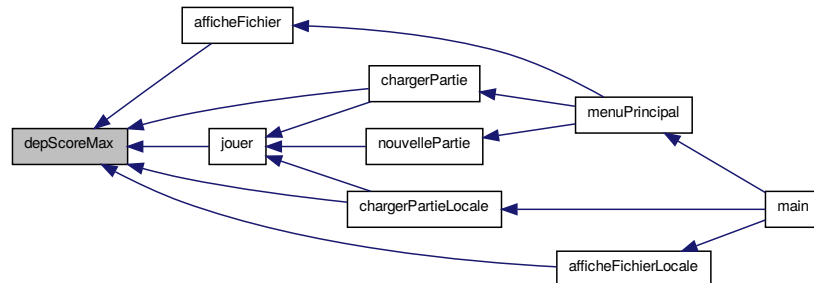
## Parameters

in	<i>*ptr_struct_fichier</i>	la structure du fichier
----	----------------------------	-------------------------

## Returns

VRAI s'il y a un depassement, FAUX sinon

Here is the caller graph for this function:



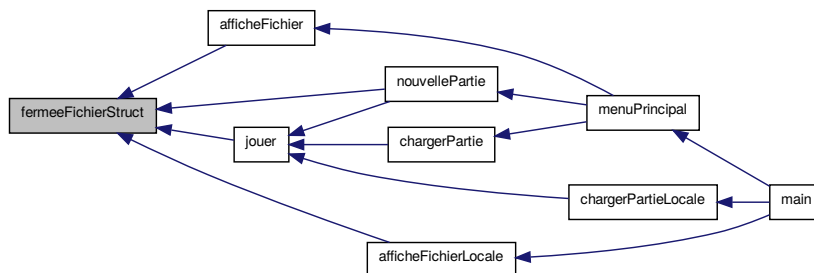
#### 4.19.2.6 void fermeeFichierStruct ( Fichier\_Jeu \* ptr\_struct\_fichier )

Desalloue la memoire attribuee a la structure [Fichier\\_Jeu](#) mis en parametre

## Parameters

in, out	*ptr_struct_fichier	un pointeur sur la structure <a href="#">Fichier_Jeu</a> a fermer
---------	---------------------	---

Here is the caller graph for this function:



#### 4.19.2.7 void finNouvTour ( Fichier\_Jeu \* ptr\_struct\_fichier )

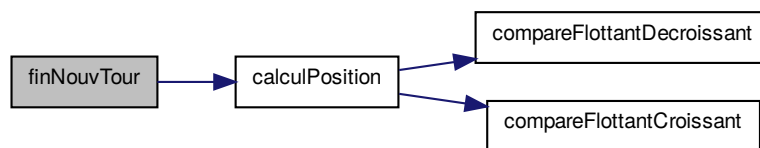
Met a jour les points totaux, le nombre de tour, la personne qui doit distribuer et les positions

## Parameters

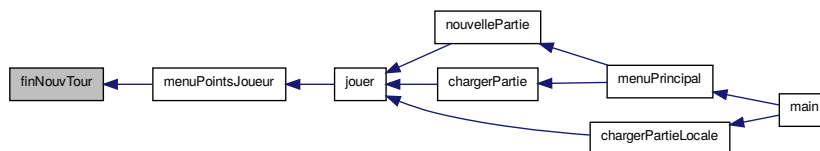
in, out	*ptr_struct_fichier	un pointeur sur la structure <a href="#">Fichier_Jeu</a>
---------	---------------------	--



Here is the call graph for this function:



Here is the caller graph for this function:

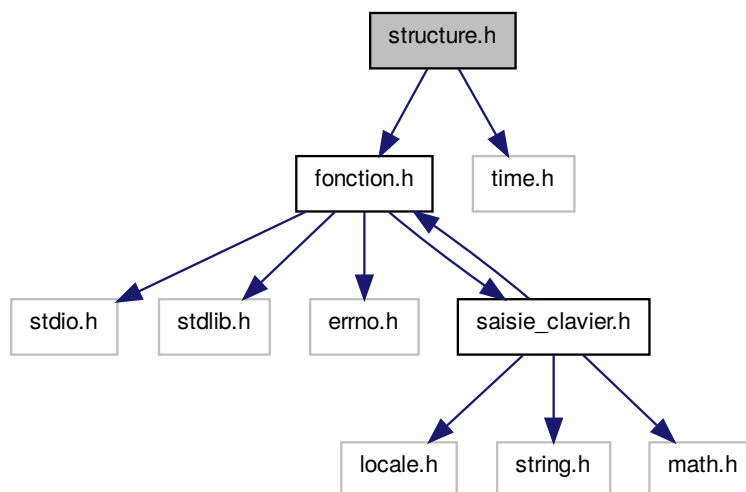


## 4.20 structure.h File Reference

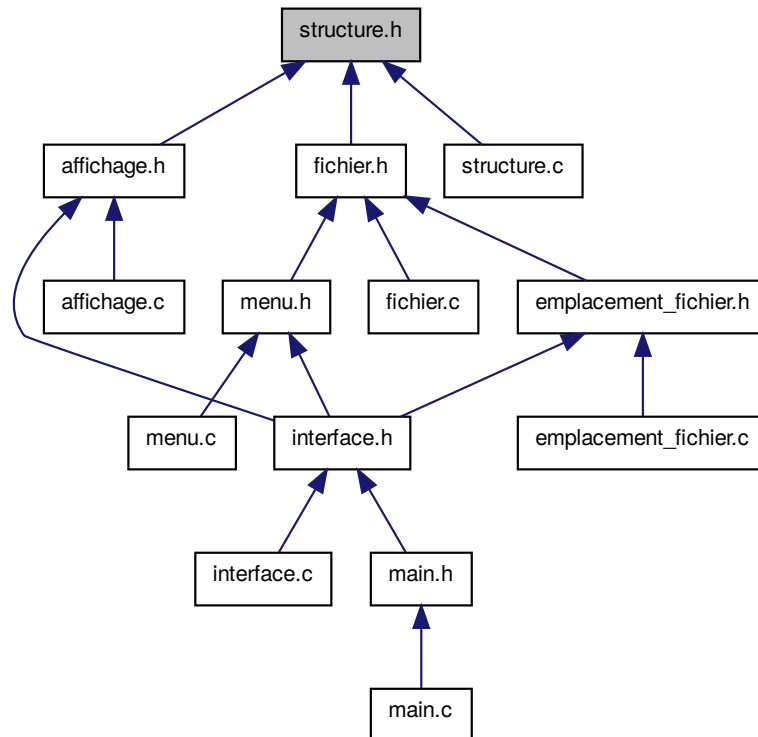
Prototypes des fonction de gestion des fichiers de la struction contenant les informations.

```
#include "fonction.h"
#include <time.h>
```

Include dependency graph for structure.h:



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



## Data Structures

- struct [Fichier\\_Jeu](#)

## Macros

- #define [TAILLE\\_MAX\\_NOM](#) 30
- #define [VERSION](#) 1.4

## Functions

- [Fichier\\_Jeu \\* creerFichierStruct](#) (float nb\_joueur, float nb\_max, char sens\_premier)
- void [fermeeFichierStruct](#) ([Fichier\\_Jeu](#) \*ptr\_struct\_fichier)
- void [debNouvTour](#) ([Fichier\\_Jeu](#) \*ptr\_struct\_fichier)
- void [finNouvTour](#) ([Fichier\\_Jeu](#) \*ptr\_struct\_fichier)
- void [calculPosition](#) ([Fichier\\_Jeu](#) \*ptr\_struct\_fichier)
- void [ajoutDistribueStruct](#) ([Fichier\\_Jeu](#) \*ptr\_struct\_fichier, char \*nom\_distribue)
- int [depScoreMax](#) ([Fichier\\_Jeu](#) \*ptr\_struct\_fichier)

### 4.20.1 Detailed Description

Prototypes des fonction de gestion des fichiers de la struction contenant les informations.

**Author**

Remi BERTHO

**Date**

09/03/14

**Version**

2.1.0

**4.20.2 Macro Definition Documentation****4.20.2.1 #define TAILLE\_MAX\_NOM 30**

Definit la taille max d'un nom a 30

**4.20.2.2 #define VERSION 1.4**

Definit la version a 1.4

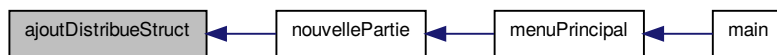
**4.20.3 Function Documentation****4.20.3.1 void ajoutDistribueStruct ( Fichier\_Jeu \* ptr\_struct\_fichier, char \* nom\_distribue )**

Ajoute la personne qui distribue dans la structure.

**Parameters**

in	<i>*nom_distribue</i>	le nom de la personne qui distribue
in	<i>*ptr_struct_fichier</i>	la structure du fichier avec lequel on veut mettre la personne qui distribue

Here is the caller graph for this function:

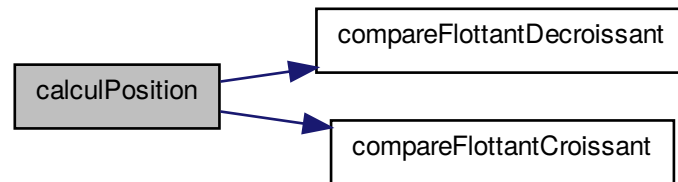
**4.20.3.2 void calculPosition ( Fichier\_Jeu \* ptr\_struct\_fichier )**

Calcule les positions des joueurs

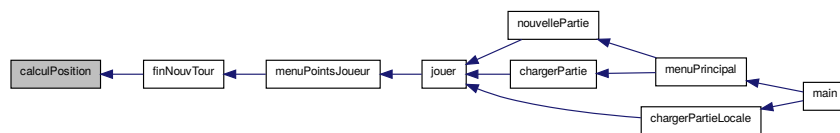
**Parameters**

in, out	<i>*ptr_struct_fichier</i>	un pointeur sur la structure <a href="#">Fichier_Jeu</a>
---------	----------------------------	--

Here is the call graph for this function:



Here is the caller graph for this function:



#### 4.20.3.3 Fichier\_Jeu\* creerFichierStruct ( float nb\_joueur, float nb\_max, char sens\_premier )

Cree une structure Fichier\_jeu a partir des donnees de la fonction.

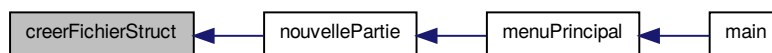
##### Parameters

in	<i>nb_joueur</i>	le nombre de joueur
in	<i>nb_max</i>	le nombre maximum de points
in	<i>sens_premier</i>	permet de savoir dans quelle sens est calculer le premier

##### Returns

un pointeur sur le structure [Fichier\\_Jeu](#) cree

Here is the caller graph for this function:



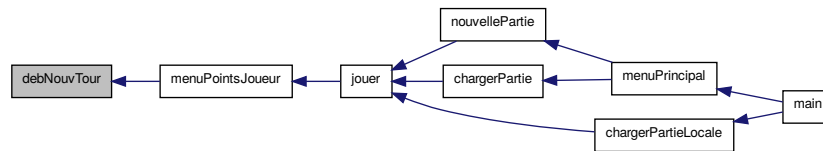
#### 4.20.3.4 void debNouvTour ( Fichier\_Jeu \* ptr\_struct\_fichier )

Realloue l'espace memoire dedie au points sur la structure [Fichier\\_Jeu](#) mis en parametre

## Parameters

in, out	<i>*ptr_struct_fichier</i>	un pointeur sur la structure <a href="#">Fichier_Jeu</a>
---------	----------------------------	--

Here is the caller graph for this function:



#### 4.20.3.5 int depScoreMax ( Fichier\_Jeu \* ptr\_struct\_fichier )

Verifie si quelqu'un a depasse le score maximum

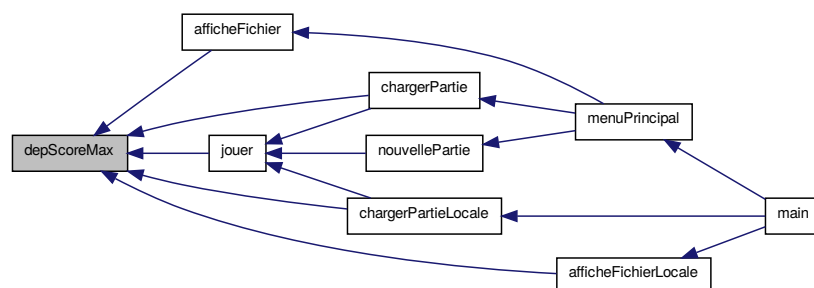
## Parameters

in	<i>*ptr_struct_fichier</i>	la structure du fichier
----	----------------------------	-------------------------

## Returns

VRAI s'il y a un depassement, FAUX sinon

Here is the caller graph for this function:



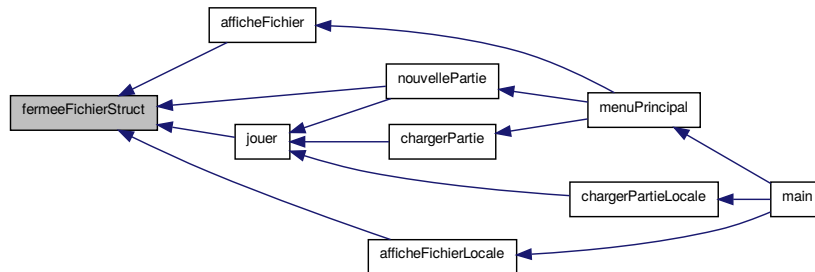
#### 4.20.3.6 void fermeeFichierStruct ( Fichier\_Jeu \* ptr\_struct\_fichier )

Desalloue la memoire attribuee a la structure [Fichier\\_Jeu](#) mis en parametre

## Parameters

in, out	<i>*ptr_struct_fichier</i>	un pointeur sur la structure <a href="#">Fichier_Jeu</a> a fermer
---------	----------------------------	---

Here is the caller graph for this function:



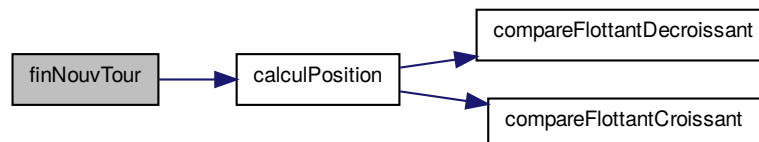
#### 4.20.3.7 void finNouvTour ( Fichier\_Jeu \* ptr\_struct\_fichier )

Met a jour les points totaux, le nombre de tour, la personne qui doit distribuer et les positions

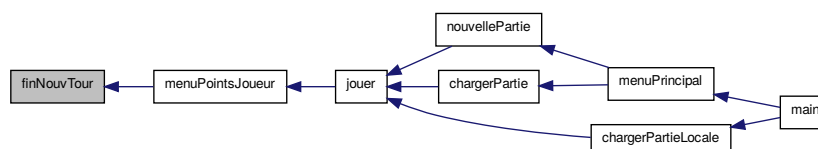
##### Parameters

in, out	<i>*ptr_struct_fichier</i>	un pointeur sur la structure <a href="#">Fichier_Jeu</a>
---------	----------------------------	--

Here is the call graph for this function:



Here is the caller graph for this function:



# Index

affFich  
    interface.h, 79

affichage.c, 9

    afficherChaineTroisTab, 10

    afficherDistribue, 10

    afficherEnTete, 11

    afficherLicense, 11

    afficherLigne, 12

    afficherNom, 12

    afficherPartieFinie, 13

    afficherPosition, 13

    afficherScore, 14

    afficherScoreEntier, 15

    afficherScoreTotal, 15

    afficherStruct, 16

affichage.h, 17

    afficherChaineTroisTab, 19

    afficherDistribue, 19

    afficherEnTete, 19

    afficherLicense, 20

    afficherLigne, 20

    afficherNom, 21

    afficherPartieFinie, 21

    afficherPosition, 22

    afficherScore, 22

    afficherScoreEntier, 23

    afficherScoreTotal, 24

    afficherStruct, 24

afficheFichier

    interface.c, 67

    interface.h, 79

afficheFichierLocale

    interface.c, 68

    interface.h, 80

afficherChaineTroisTab

    affichage.c, 10

    affichage.h, 19

afficherDistribue

    affichage.c, 10

    affichage.h, 19

afficherEnTete

    affichage.c, 11

    affichage.h, 19

afficherLicense

    affichage.c, 11

    affichage.h, 20

afficherLigne

    affichage.c, 12

    affichage.h, 20

afficherNom

    affichage.c, 12

    affichage.h, 21

afficherPartieFinie

    affichage.c, 13

    affichage.h, 21

afficherPosition

    affichage.c, 13

    affichage.h, 22

afficherScore

    affichage.c, 14

    affichage.h, 22

afficherScoreEntier

    affichage.c, 15

    affichage.h, 23

afficherScoreTotal

    affichage.c, 15

    affichage.h, 24

afficherStruct

    affichage.c, 16

    affichage.h, 24

ajoutDistribueStruct

    structure.c, 123

    structure.h, 129

annee

    Fichier\_Jeu, 5

argument\_main.c, 25

    searchArgument, 26

argument\_main.h, 27

    LECTURE\_FICHER, 29

    OUVERTURE\_FICHER, 29

    searchArgument, 29

calculPosition

    structure.c, 124

    structure.h, 129

changerCheminFichier

    emplacement\_fichier.c, 31

    emplacement\_fichier.h, 37

charPart

    interface.h, 79

chargerPartie

    interface.c, 69

    interface.h, 81

chargerPartieLocale

    interface.c, 70

    interface.h, 82

clean\_stdin

    saisie\_clavier.c, 109

    saisie\_clavier.h, 116

- compareFlottantCroissant
  - fonction.c, [54](#)
  - fonction.h, [61](#)
- compareFlottantDecroissant
  - fonction.c, [55](#)
  - fonction.h, [61](#)
- creationPreferences
  - emplacement\_fichier.c, [31](#)
  - emplacement\_fichier.h, [37](#)
- creerFichierStruct
  - structure.c, [124](#)
  - structure.h, [130](#)
- debNouvTour
  - structure.c, [125](#)
  - structure.h, [130](#)
- depScoreMax
  - structure.c, [125](#)
  - structure.h, [131](#)
- distribue
  - Fichier\_Jeu, [5](#)
- EXTENSION\_FICHER
  - fichier.h, [48](#)
- easterEggs
  - interface.h, [79](#)
- ecrireFichier
  - fichier.c, [42](#)
  - fichier.h, [48](#)
- emplacement\_fichier.c, [29](#)
  - changerCheminFichier, [31](#)
  - creationPreferences, [31](#)
  - lectureCheminFichier, [32](#)
  - lecturePreferences, [33](#)
- emplacement\_fichier.h, [34](#)
  - changerCheminFichier, [37](#)
  - creationPreferences, [37](#)
  - lectureCheminFichier, [38](#)
  - lecturePreferences, [39](#)
  - NOM\_DOSSIER, [37](#)
  - NOM\_FICHER, [37](#)
- FAUX
  - fonction.h, [61](#)
- fermeeFichierStruct
  - structure.c, [126](#)
  - structure.h, [131](#)
- fermerFichier
  - fonction.c, [55](#)
  - fonction.h, [61](#)
- fichier.c, [40](#)
  - ecrireFichier, [42](#)
  - lireFichier, [42](#)
  - nouveauScore, [43](#)
  - ouvrirFichierExtension, [44](#)
  - renommerFichier, [45](#)
  - supprimerFichier, [45](#)
- fichier.h, [46](#)
  - EXTENSION\_FICHER, [48](#)
  - ecrireFichier, [48](#)
  - lireFichier, [49](#)
  - nouveauScore, [50](#)
  - ouvrirFichierExtension, [51](#)
  - renommerFichier, [51](#)
  - supprimerFichier, [52](#)
  - TYPE\_FICHER, [48](#)
- Fichier\_Jeu, [5](#)
  - annee, [5](#)
  - distribue, [5](#)
  - jour, [5](#)
  - mois, [6](#)
  - nb\_joueur, [6](#)
  - nb\_max, [6](#)
  - nb\_tour, [6](#)
  - nom\_joueur, [6](#)
  - point, [6](#)
  - point\_tot, [6](#)
  - position, [6](#)
  - sens\_premier, [6](#)
  - taille\_max\_nom, [6](#)
  - version, [6](#)
- finNouvTour
  - structure.c, [126](#)
  - structure.h, [132](#)
- fonction.c, [52](#)
  - compareFlottantCroissant, [54](#)
  - compareFlottantDecroissant, [55](#)
  - fermerFichier, [55](#)
  - lireTailleFichier, [56](#)
  - mauvais\_choix, [56](#)
  - ouvrirFichier, [57](#)
  - systemEfface, [57](#)
  - systemPause, [58](#)
- fonction.h, [59](#)
  - compareFlottantCroissant, [61](#)
  - compareFlottantDecroissant, [61](#)
  - FAUX, [61](#)
  - fermerFichier, [61](#)
  - lireTailleFichier, [62](#)
  - mauvais\_choix, [62](#)
  - ouvrirFichier, [63](#)
  - systemEfface, [63](#)
  - systemPause, [64](#)
  - VRAL, [61](#)
- interface.h
  - affFich, [79](#)
  - charPart, [79](#)
  - easterEggs, [79](#)
  - lireChem, [79](#)
  - listFich, [79](#)
  - nouvChem, [79](#)
  - nouvPart, [79](#)
  - quit, [79](#)
  - supprFich, [79](#)
- interface.c, [65](#)
  - afficheFichier, [67](#)
  - afficheFichierLocale, [68](#)



- chargerPartie, 69
- chargerPartieLocale, 70
- jouer, 71
- lireCheminFichier, 72
- listerFichier, 72
- menuPrincipal, 73
- nouveauCheminFichier, 74
- nouvellePartie, 75
- supprimerFichierNom, 76
- interface.h, 77
  - afficheFichier, 79
  - afficheFichierLocale, 80
  - chargerPartie, 81
  - chargerPartieLocale, 82
  - jouer, 83
  - lireCheminFichier, 84
  - listerFichier, 85
  - Menu, 79
  - menuPrincipal, 86
  - nouveauCheminFichier, 87
  - nouvellePartie, 87
  - supprimerFichierNom, 88
- jouer
  - interface.c, 71
  - interface.h, 83
- jour
  - Fichier\_Jeu, 5
- LECTURE\_FICHER
  - argument\_main.h, 29
- lectureCheminFichier
  - emplacement\_fichier.c, 32
  - emplacement\_fichier.h, 38
- lecturePreferences
  - emplacement\_fichier.c, 33
  - emplacement\_fichier.h, 39
- lireChem
  - interface.h, 79
- lireCheminFichier
  - interface.c, 72
  - interface.h, 84
- lireFichier
  - fichier.c, 42
  - fichier.h, 49
- lireTailleFichier
  - fonction.c, 56
  - fonction.h, 62
- listFich
  - interface.h, 79
- listerFichier
  - interface.c, 72
  - interface.h, 85
- main
  - main.c, 90
  - main.h, 93
- main.c, 89
  - main, 90
- main.h, 92
  - main, 93
- mauvais\_choix
  - fonction.c, 56
  - fonction.h, 62
- Menu
  - interface.h, 79
- menu.c, 94
  - menuContinuer, 96
  - menuDebutPartie, 96
  - menuDistribue, 97
  - menuNomFichier, 97
  - menuNomJoueur, 98
  - menuNouveauChemin, 99
  - menuPointsJoueur, 99
  - menuSupprimer, 100
- menu.h, 101
  - menuContinuer, 103
  - menuDebutPartie, 103
  - menuDistribue, 104
  - menuNomFichier, 104
  - menuNomJoueur, 105
  - menuNouveauChemin, 106
  - menuPointsJoueur, 106
  - menuSupprimer, 107
- menuContinuer
  - menu.c, 96
  - menu.h, 103
- menuDebutPartie
  - menu.c, 96
  - menu.h, 103
- menuDistribue
  - menu.c, 97
  - menu.h, 104
- menuNomFichier
  - menu.c, 97
  - menu.h, 104
- menuNomJoueur
  - menu.c, 98
  - menu.h, 105
- menuNouveauChemin
  - menu.c, 99
  - menu.h, 106
- menuPointsJoueur
  - menu.c, 99
  - menu.h, 106
- menuPrincipal
  - interface.c, 73
  - interface.h, 86
- menuSupprimer
  - menu.c, 100
  - menu.h, 107
- mois
  - Fichier\_Jeu, 6
- NB\_CHARACTER\_DOUB
  - saisie\_clavier.h, 116
- NB\_CHARACTER\_FLOT
  - saisie\_clavier.h, 116

NB\_CARACT\_INT  
     saisie\_clavier.h, 116  
 NOM\_DOSSIER  
     emplacement\_fichier.h, 37  
 NOM\_FICHER  
     emplacement\_fichier.h, 37  
 nb\_joueur  
     Fichier\_Jeu, 6  
 nb\_max  
     Fichier\_Jeu, 6  
 nb\_tour  
     Fichier\_Jeu, 6  
 nom\_joueur  
     Fichier\_Jeu, 6  
 nouvChem  
     interface.h, 79  
 nouvPart  
     interface.h, 79  
 nouveauCheminFichier  
     interface.c, 74  
     interface.h, 87  
 nouveauScore  
     fichier.c, 43  
     fichier.h, 50  
 nouvellePartie  
     interface.c, 75  
     interface.h, 87  
  
 OUVERTURE\_FICHER  
     argument\_main.h, 29  
 ouvrirFichier  
     fonction.c, 57  
     fonction.h, 63  
 ouvrirFichierExtension  
     fichier.c, 44  
     fichier.h, 51  
  
 point  
     Fichier\_Jeu, 6  
 point\_tot  
     Fichier\_Jeu, 6  
 position  
     Fichier\_Jeu, 6  
  
 quit  
     interface.h, 79  
  
 renommerFichier  
     fichier.c, 45  
     fichier.h, 51  
  
 saisie\_clavier.c, 108  
     clean\_stdin, 109  
     saisieClavierCaractere, 109  
     saisieClavierChaine, 110  
     saisieClavierDouble, 111  
     saisieClavierEntier, 111  
     saisieClavierFlottant, 112  
     saisieClavierFlottantSansVirgule, 112  
     saisie\_clavier.h, 114  
     clean\_stdin, 116  
     NB\_CARACT\_DOUB, 116  
     NB\_CARACT\_FLOT, 116  
     NB\_CARACT\_INT, 116  
     saisieClavierCaractere, 116  
     saisieClavierChaine, 117  
     saisieClavierDouble, 118  
     saisieClavierEntier, 118  
     saisieClavierFlottant, 119  
     saisieClavierFlottantSansVirgule, 119  
 saisieClavierCaractere  
     saisie\_clavier.c, 109  
     saisie\_clavier.h, 116  
 saisieClavierChaine  
     saisie\_clavier.c, 110  
     saisie\_clavier.h, 117  
 saisieClavierDouble  
     saisie\_clavier.c, 111  
     saisie\_clavier.h, 118  
 saisieClavierEntier  
     saisie\_clavier.c, 111  
     saisie\_clavier.h, 118  
 saisieClavierFlottant  
     saisie\_clavier.c, 112  
     saisie\_clavier.h, 119  
 saisieClavierFlottantSansVirgule  
     saisie\_clavier.c, 112  
     saisie\_clavier.h, 119  
 searchArgument  
     argument\_main.c, 26  
     argument\_main.h, 29  
 sens\_premier  
     Fichier\_Jeu, 6  
 structure.c, 121  
     ajoutDistribueStruct, 123  
     calculPosition, 124  
     creerFichierStruct, 124  
     debNouvTour, 125  
     depScoreMax, 125  
     fermeeFichierStruct, 126  
     finNouvTour, 126  
 structure.h, 127  
     ajoutDistribueStruct, 129  
     calculPosition, 129  
     creerFichierStruct, 130  
     debNouvTour, 130  
     depScoreMax, 131  
     fermeeFichierStruct, 131  
     finNouvTour, 132  
     TAILLE\_MAX\_NOM, 129  
     VERSION, 129  
 supprFich  
     interface.h, 79  
 supprimerFichier  
     fichier.c, 45  
     fichier.h, 52  
 supprimerFichierNom

- interface.c, [76](#)
  - interface.h, [88](#)
- systemEfface
  - fonction.c, [57](#)
  - fonction.h, [63](#)
- systemPause
  - fonction.c, [58](#)
  - fonction.h, [64](#)
- TAILLE\_MAX\_NOM
  - structure.h, [129](#)
- TYPE\_FICHIER
  - fichier.h, [48](#)
- taille\_max\_nom
  - Fichier\_Jeu, [6](#)
- VERSION
  - structure.h, [129](#)
- VRAI
  - fonction.h, [61](#)
- version
  - Fichier\_Jeu, [6](#)