

2o48.hackable.c

16D0

Generated by Doxygen 1.8.10

Wed Dec 9 2015 21:38:21

Contents

1	2o48.hackable.c	1
2	File Index	3
2.1	File List	3
3	File Documentation	5
3.1	2048.c File Reference	5
3.1.1	Macro Definition Documentation	8
3.1.1.1	EDOWN	8
3.1.1.2	ELEFT	8
3.1.1.3	ERIGHT	8
3.1.1.4	EUP	8
3.1.1.5	MAX_BOARD_NUM	8
3.1.1.6	MAX_BOARD_SIZE	8
3.1.1.7	MENU_POSITION_X	8
3.1.1.8	MENU_POSITION_Y	8
3.1.1.9	PWD	8
3.1.1.10	PWD_LEN	9
3.1.1.11	WARNING_POSITION_X	9
3.1.1.12	WARNING_POSITION_Y	9
3.1.2	Typedef Documentation	9
3.1.2.1	Attr	9
3.1.2.2	Cond	9
3.1.2.3	Mut	9
3.1.2.4	Thrd	9
3.1.3	Function Documentation	9
3.1.3.1	AlignCol(int curcol, int direction)	9
3.1.3.2	AlignLine(int curline, int direction)	10
3.1.3.3	c_boom(int y, int x)	10
3.1.3.4	c_checksum()	11
3.1.3.5	c_currentStr(bool show)	11
3.1.3.6	c_forceQuit()	12

3.1.3.7	<code>c_info(char *msg)</code>	12
3.1.3.8	<code>c_loadStr()</code>	13
3.1.3.9	<code>c_loadStr(int iptN, FILE *fp)</code>	13
3.1.3.10	<code>c_readBoard(int from)</code>	14
3.1.3.11	<code>c_readFromDisk(int boards)</code>	15
3.1.3.12	<code>c_saveBoard(int to, bool jmp)</code>	16
3.1.3.13	<code>c_tryQuit()</code>	16
3.1.3.14	<code>c_version()</code>	17
3.1.3.15	<code>c_warning(char *msg)</code>	18
3.1.3.16	<code>c_writeBoardToDisk(char boards)</code>	18
3.1.3.17	<code>CheckEat(char *a, char *b)</code>	19
3.1.3.18	<code>Clrboard(int boardToClr)</code>	20
3.1.3.19	<code>command()</code>	21
3.1.3.20	<code>die()</code>	21
3.1.3.21	<code>Display(char in)</code>	22
3.1.3.22	<code>Eat(bool isH, int direction)</code>	23
3.1.3.23	<code>EatCol(int curcol, int direction)</code>	24
3.1.3.24	<code>EatLine(const int curline, int direction)</code>	25
3.1.3.25	<code>GetRandNums()</code>	26
3.1.3.26	<code>main()</code>	27
3.1.3.27	<code>play()</code>	28
3.1.3.28	<code>Rando(int N)</code>	30
3.1.3.29	<code>settings()</code>	30
3.1.3.30	<code>showBoard(WINDOW *win, int offy, int offx)</code>	31
3.1.3.31	<code>showInfo(WINDOW *win)</code>	32
3.1.3.32	<code>t_Info(void *arg)</code>	33
3.1.3.33	<code>t_Show(void *arg)</code>	34
3.1.3.34	<code>welcome()</code>	34
3.1.4	Variable Documentation	35
3.1.4.1	<code>AThread</code>	35
3.1.4.2	<code>board</code>	35
3.1.4.3	<code>boardseed</code>	35
3.1.4.4	<code>boardstr</code>	35
3.1.4.5	<code>BoardWin</code>	35
3.1.4.6	<code>CBoard</code>	35
3.1.4.7	<code>CInfo</code>	35
3.1.4.8	<code>col</code>	36
3.1.4.9	<code>cs_pwd</code>	36
3.1.4.10	<code>curs</code>	36
3.1.4.11	<code>display</code>	36

3.1.4.12	eat	36
3.1.4.13	iswarn	36
3.1.4.14	MAX_RANDOM	36
3.1.4.15	MBoard	36
3.1.4.16	MenuWin	36
3.1.4.17	MInfo	37
3.1.4.18	N	37
3.1.4.19	NA	37
3.1.4.20	P_RANDOM	37
3.1.4.21	point	37
3.1.4.22	row	37
3.1.4.23	sinfo	37
3.1.4.24	TInfo	37
3.1.4.25	TShow	37
3.2	README.md File Reference	38
Index		39

Chapter 1

2048.hackable.c

An hackable 2048 game written in C

Used ncurses.h to control the input and output

USAGE

Welcome:

Enter the size of board that you want to play. The size must be 1~9, The version number of game executable will be displayed at center of screen;

Play:

Use arrow left/down/up/right or key hjkl to move Type : to input commands;

Command:

```
#### Stash in memory:
##### s | save [n]
    Save current board and continuing current game in a new board
    n (optional): board number to jump to;
##### s! | saveto [n]
    Save current board to a new board and continuing current game in a current board
    n (optional): board number to save to;
##### r | read [n]
    Read from a board saved in memory
    n (optional):board number to read from;

#### Save in file:
    The file name will be 2048[.d].%X.save,
    where %d is the board number and %X is version number in hexadecimal;
##### w | write
    Write current board to file;
##### wb | writeboard [n]
    Write board specfied to file
    n :board number to save;
##### o | open [n]
    Open a board saved in file
    n (optional): board number to read;

####Gaming option
##### q | quit | (ctrl+c)
    Quit the game with a confirming;
##### q! | quit!
    Quit the game without confirming;
##### wq | writequit
```

Write current board to file and quit without confirming

License

The MIT License (MIT)

Copyright (c) 2015 Librazy

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Chapter 2

File Index

2.1 File List

Here is a list of all files with brief descriptions:

2048.c	5
----------------------------------	---

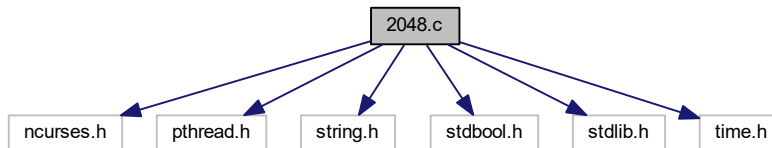
Chapter 3

File Documentation

3.1 2048.c File Reference

```
#include <ncurses.h>
#include <pthread.h>
#include <string.h>
#include <stdbool.h>
#include <stdlib.h>
#include <time.h>
```

Include dependency graph for 2048.c:



Macros

- `#define PWD ":2048"`
- `#define PWD_LEN 5`
- `#define MAX_BOARD_NUM 16`
- `#define MAX_BOARD_SIZE 16`
- `#define EUP false,1`
- `#define EDOWN false,-1`
- `#define ELEFT true,1`
- `#define ERIGHT true,-1`
- `#define MENU_POSITION_Y 0`
- `#define MENU_POSITION_X 0`
- `#define WARNING_POSITION_Y 0`
- `#define WARNING_POSITION_X 0`

Typedefs

- `typedef pthread_mutex_t Mut`

- typedef pthread_cond_t [Cond](#)
- typedef pthread_attr_t [Attr](#)
- typedef pthread_t [Thrd](#)

Functions

- void [settings](#) ()
Set the global settings.
- char [AlignCol](#) (int curcol, int direction)
Align the vertical direction.
- char [AlignLine](#) (int curline, int direction)
Align the horizontal direction.
- char [CheckEat](#) (char *a, char *b)
Check while the two number can be eaten.
- void [Clrboard](#) (int boardToClr)
Empty the board specified.
- char * [Display](#) (char in)
Get the grid's display string.
- char [Eat](#) (bool isH, int direction)
Eat the board at the direction specified.
- char [EatCol](#) (int curcol, int direction)
Eat the vertical direction.
- char [EatLine](#) (const int curline, int direction)
Eat the horizontal direction.
- int [GetRandNums](#) ()
Get random grid ranged from 0 to MAX_RANDOMUM on board.
- unsigned int [Rando](#) (int N)
Generate random num ranged from 0 to N-1.
- void * [t_Show](#) (void *arg)
The thread to print Board to screen.
- void * [t_Info](#) (void *arg)
The thread to print infos to screen.
- void [command](#) ()
Show and handle commands inputed by :
- bool [die](#) ()
Handle when no empty grid present.
- bool [play](#) ()
Handle for main game.
- void [showBoard](#) (WINDOW *win, int offy, int offx)
Print the board to screen.
- void [welcome](#) ()
Print welcome message and input the size of the board.
- void [c_boom](#) (int y, int x)
Make (y,x) a 'boom'.
- int [c_checksum](#) ()
Calculate the checksum for saving.
- void [c_currentStr](#) (bool show)
Genetate the string representing current board.
- void [c_forceQuit](#) ()
Quit the game.
- void [c_loadStr](#) ()

- void `c_readBoard` (int from)
Read the saved board.
- void `c_readFromDisk` (int boards)
Read the saved file.
- void `c_saveBoard` (int to, bool jmp)
Save the board in memory.
- void `c_tryQuit` ()
Ask player whether to quit.
- int `c_version` ()
Calculate the game's version.
- void `c_warning` (char *msg)
To print a warning to screen.
- void `c_info` (char *msg)
To print a info to screen.
- bool `c_writeBoardToDisk` (char boards)
Write the board to disk.
- void `showInfo` (WINDOW *win)
Print the info to screen.
- void `c_loadStr` (int iptN, FILE *fp)
Load the string representing saved board.
- int `main` ()
Main executable.

Variables

- const int `NA` =126
- int `P_RANDOMUM` =30
- int `MAX_RANDOMUM` =2
- const char `cs_pwd` [PWD_LEN+1] =PWD
- char `board` [MAX_BOARD_NUM][MAX_BOARD_SIZE][MAX_BOARD_SIZE]
- char `boardstr` [MAX_BOARD_SIZE][MAX_BOARD_SIZE][5]
- int `boardseed` [MAX_BOARD_NUM]
- unsigned char `curs` =0
- char `eat` [256][256][2]
Eat table(TODO:use eat array in CheckEat)
- char `display` [256][16]
- int `point` [256]
Point table(TODO:use eat array in CheckEat)
- char `N` =5
- int `row`
- int `col`
- char `sinfo` [512]
- bool `iswarn`
- `Thrd TShow`
- `Thrd TInfo`
- `Mut MBoard`
- `Mut MInfo`
- `Cond CBoard`
- `Cond CInfo`
- `Attr AThread`
- WINDOW * `BoardWin`
- WINDOW * `MenuWin`

3.1.1 Macro Definition Documentation

3.1.1.1 `#define EDOWN false,-1`

Eat down

Definition at line 49 of file 2048.c.

3.1.1.2 `#define ELEFT true,1`

Eat left

Definition at line 50 of file 2048.c.

3.1.1.3 `#define ERIGHT true,-1`

Rat right

Definition at line 51 of file 2048.c.

3.1.1.4 `#define EUP false,1`

Eat up

Definition at line 48 of file 2048.c.

3.1.1.5 `#define MAX_BOARD_NUM 16`

The maxium board num (for saving in game)

Definition at line 29 of file 2048.c.

3.1.1.6 `#define MAX_BOARD_SIZE 16`

The maxium board size

Definition at line 33 of file 2048.c.

3.1.1.7 `#define MENU_POSITION_X 0`

the X position to print menu

Definition at line 64 of file 2048.c.

3.1.1.8 `#define MENU_POSITION_Y 0`

the Y position to print menu

Definition at line 63 of file 2048.c.

3.1.1.9 `#define PWD "2048"`

The password for the save file and represent the version of the game Should and only be changed when the saved file isn't/shouldn't compatible with others

Definition at line 5 of file 2048.c.

3.1.1.10 `#define PWD_LEN 5`

The length of PWD

Definition at line 9 of file 2048.c.

3.1.1.11 `#define WARNING_POSITION_X 0`

the X position to print warning

Definition at line 66 of file 2048.c.

3.1.1.12 `#define WARNING_POSITION_Y 0`

the Y position to print warning

Definition at line 65 of file 2048.c.

3.1.2 Typedef Documentation

3.1.2.1 `typedef pthread_attr_t Attr`

Definition at line 19 of file 2048.c.

3.1.2.2 `typedef pthread_cond_t Cond`

Definition at line 18 of file 2048.c.

3.1.2.3 `typedef pthread_mutex_t Mut`

Definition at line 17 of file 2048.c.

3.1.2.4 `typedef pthread_t Thrd`

Definition at line 20 of file 2048.c.

3.1.3 Function Documentation

3.1.3.1 `char AlignCol (int curcol, int direction)`

Align the vertical direction.

Parameters

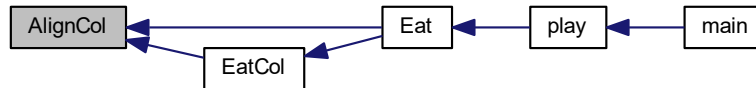
<i>curcol</i>	Current column to align
<i>direction</i>	The direction to align to positive for up and negative for down

Returns

The number of blank grid in the column

Definition at line 212 of file 2048.c.

Here is the caller graph for this function:



3.1.3.2 `char AlignLine (int curline, int direction)`

Align the horizontal direction.

Parameters

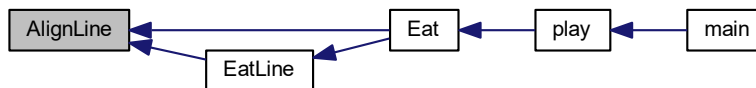
<i>curline</i>	Current line to align
<i>direction</i>	The direction to align to positive for left and negative for right

Returns

The number of blank grid in the column

Definition at line 242 of file 2048.c.

Here is the caller graph for this function:



3.1.3.3 `void c_boom (int y, int x)`

Make (y,x) a 'boom'.

Returns

void

Definition at line 588 of file 2048.c.

Here is the call graph for this function:



Here is the caller graph for this function:

**3.1.3.4 int c_checksum ()**

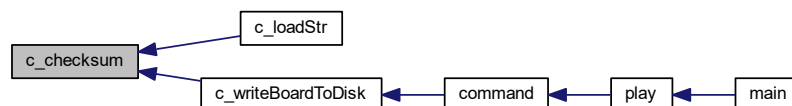
Calculate the checksum for saving.

Returns

The checksum for current board

Definition at line 595 of file 2048.c.

Here is the caller graph for this function:

**3.1.3.5 void c_currentStr (bool show)**

Genetate the string representing current board.

Parameters

<i>show</i>	If need to print the string to screen
-------------	---------------------------------------

Returns

void

Definition at line 622 of file 2048.c.

Here is the caller graph for this function:

**3.1.3.6 void c_forceQuit ()**

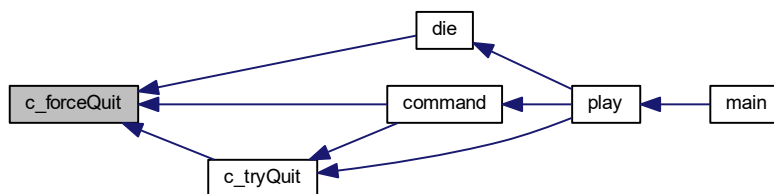
Quit the game.

Returns

void

Definition at line 651 of file 2048.c.

Here is the caller graph for this function:

**3.1.3.7 void c_info (char * msg)**

To print a info to screen.

The function storage the message in a buffer and call up the TInfo thread to print it

Parameters

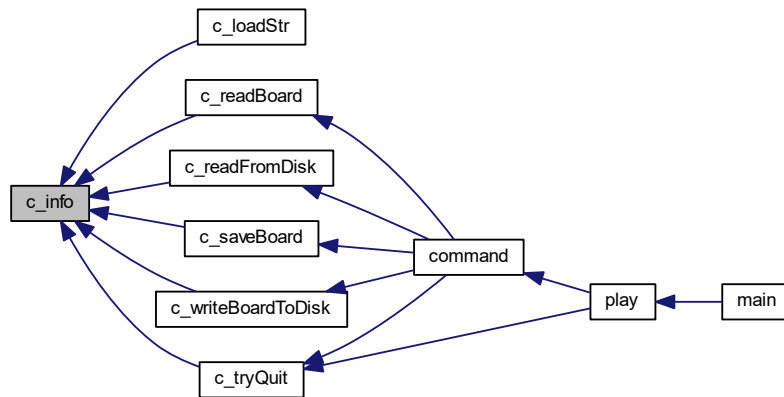
<i>msg</i>	The string to print
------------	---------------------

Returns

void

Definition at line 863 of file 2048.c.

Here is the caller graph for this function:



3.1.3.8 void c_loadStr ()

Here is the caller graph for this function:

3.1.3.9 void c_loadStr (int *iptN*, FILE * *fp*)

Load the string representing saved board.

Parameters

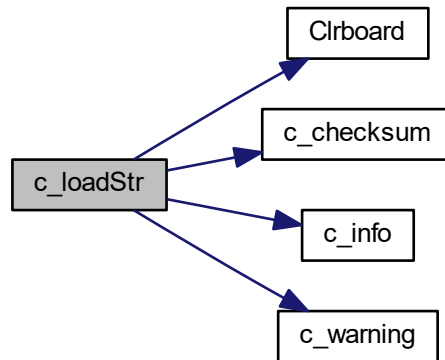
<i>iptN</i>	The N in the saved game
<i>fp</i>	The file stream to read from

Returns

void

Definition at line 660 of file 2048.c.

Here is the call graph for this function:

**3.1.3.10 void c_readBoard (int *from*)**

Read the saved board.

Parameters

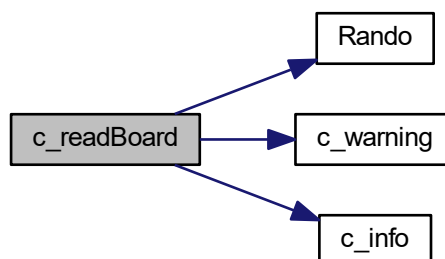
<i>from</i>	The number of board to read from
-------------	----------------------------------

Returns

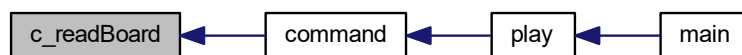
void

Definition at line 701 of file 2048.c.

Here is the call graph for this function:



Here is the caller graph for this function:



3.1.3.11 void c_readFromDisk (int *boards*)

Read the saved file.

Parameters

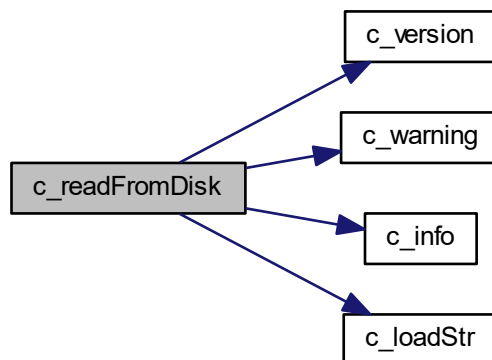
<i>boards</i>	The number of the saved board.NA for not to use
---------------	---

Returns

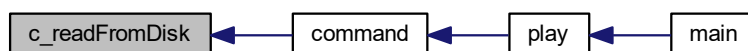
void

Definition at line 721 of file 2048.c.

Here is the call graph for this function:



Here is the caller graph for this function:



3.1.3.12 void c_saveBoard (int *to*, bool *jmp*)

Save the board in memory.

Parameters

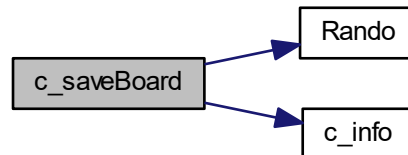
<i>to</i>	The number of the board to save to.NA for auto find nnext
<i>jmp</i>	If should jump to new board

Returns

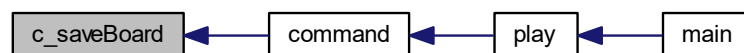
void

Definition at line 758 of file 2048.c.

Here is the call graph for this function:



Here is the caller graph for this function:



3.1.3.13 void c_tryQuit ()

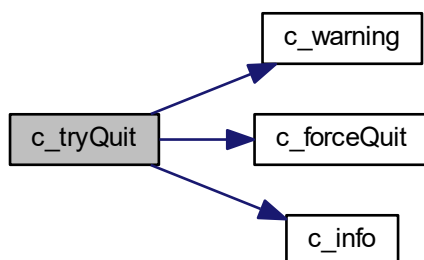
Ask player whether to quit.

Returns

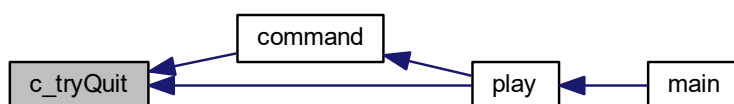
void

Definition at line 789 of file 2048.c.

Here is the call graph for this function:



Here is the caller graph for this function:

**3.1.3.14 int c_version ()**

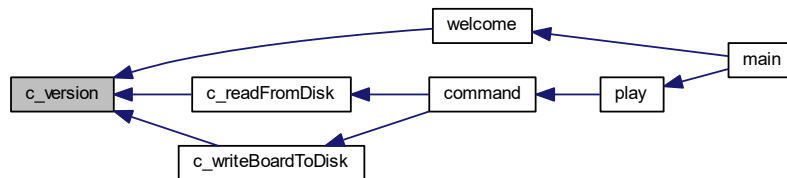
Calculate the game's version.

Returns

Version

Definition at line 780 of file 2048.c.

Here is the caller graph for this function:



3.1.3.15 void c_warning (char * msg)

To print a warning to screen.

The function storage the message in a buffer and call up the TInfo thread to print it

Parameters

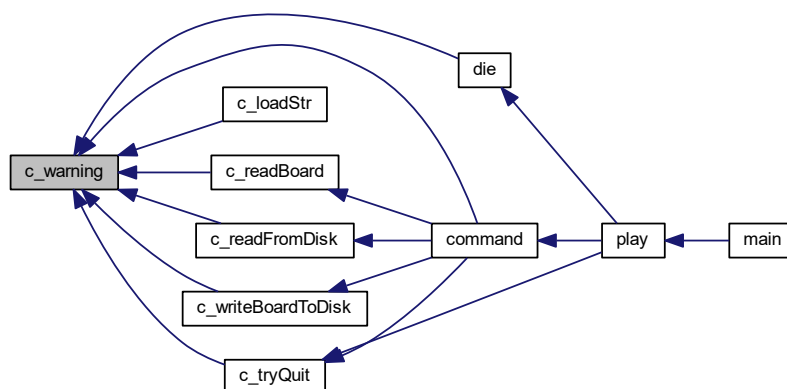
<i>msg</i>	The string to print
------------	---------------------

Returns

void

Definition at line 850 of file 2048.c.

Here is the caller graph for this function:



3.1.3.16 bool c_writeBoardToDisk (char boards)

Write the board to disk.

Parameters

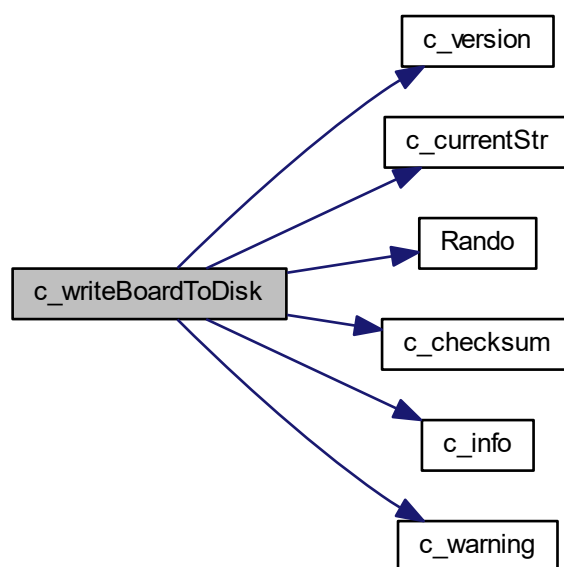
<i>boards</i>	The number of board to save
---------------	-----------------------------

Returns

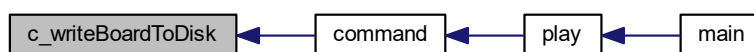
Whether the file is saved successfully

Definition at line 801 of file 2048.c.

Here is the call graph for this function:



Here is the caller graph for this function:



3.1.3.17 char CheckEat (char * a, char * b)

Check while the two number can be eaten.

Parameters

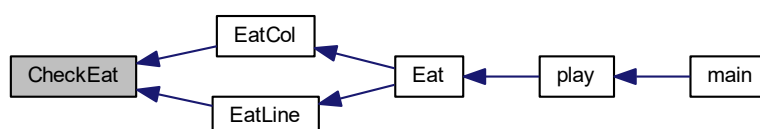
<i>a</i>	Value a
<i>b</i>	Value b

Returns

The final value of a

Definition at line 270 of file 2048.c.

Here is the caller graph for this function:



3.1.3.18 void Clrboard (int *boardToClr*)

Empty the board specified.

Parameters

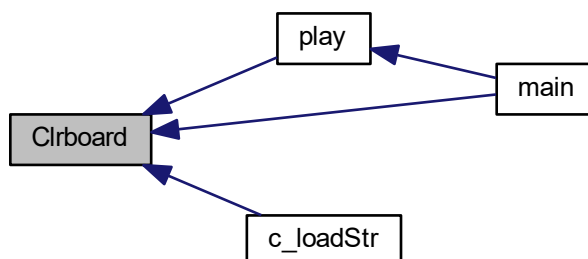
<i>boardToClr</i>	The board to empty
-------------------	--------------------

Returns

void

Definition at line 291 of file 2048.c.

Here is the caller graph for this function:



3.1.3.19 void command ()

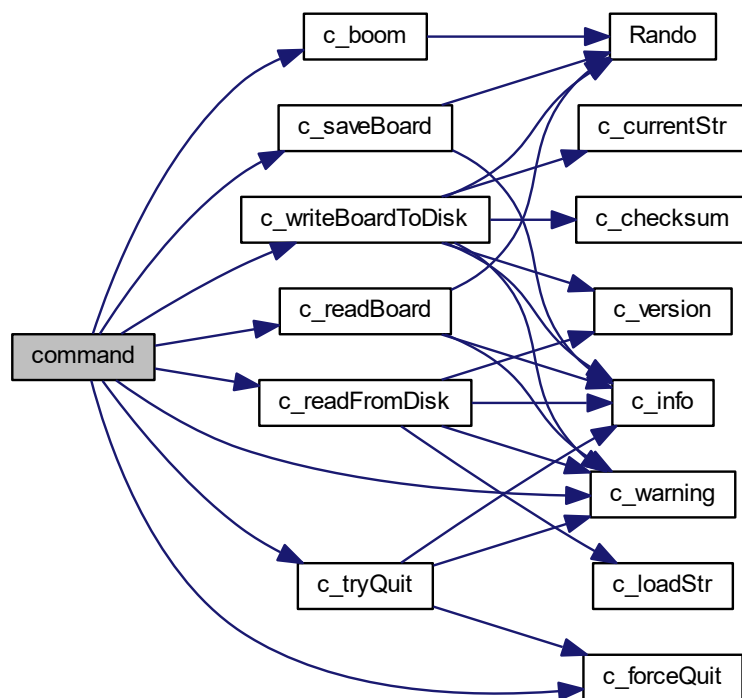
Show and handle commands inputed by :

Returns

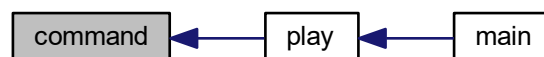
void

Definition at line 398 of file 2048.c.

Here is the call graph for this function:



Here is the caller graph for this function:



3.1.3.20 bool die ()

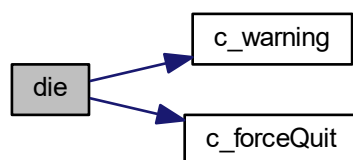
Handle when no empty grid present.

Returns

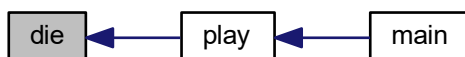
If restart

Definition at line 444 of file 2048.c.

Here is the call graph for this function:



Here is the caller graph for this function:



3.1.3.21 `char * Display (char in)`

Get the grid's display string.

Parameters

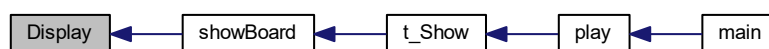
<i>in</i>	The grid's value
-----------	------------------

Returns

The string to display

Definition at line 297 of file 2048.c.

Here is the caller graph for this function:



3.1.3.22 char Eat (bool *isH*, int *direction*)

Eat the board at the direction specified.

Parameters

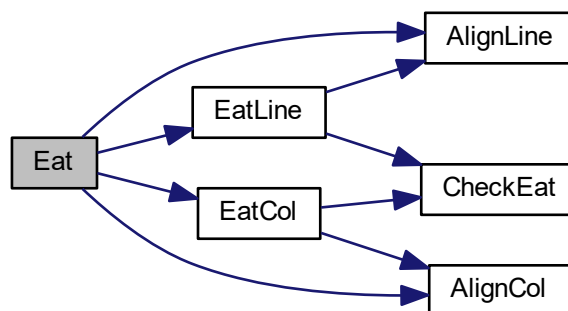
<i>isH</i>	Is horizontal true for horizontal and false for vertical
<i>direction</i>	The direction to eat to positive for left/up and negative for right/down

Returns

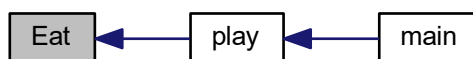
The number of empty grids

Definition at line 309 of file 2048.c.

Here is the call graph for this function:



Here is the caller graph for this function:



3.1.3.23 char EatCol (int *curcol*, int *direction*)

Eat the vertical direction.

Parameters

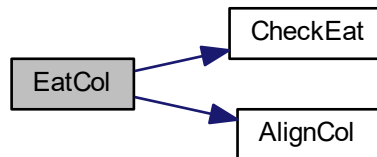
<i>curcol</i>	Current column to eat
<i>direction</i>	The direction to eat positive for up and negative for down

Returns

The number of blank grid in the column

Definition at line 330 of file 2048.c.

Here is the call graph for this function:



Here is the caller graph for this function:



3.1.3.24 char EatLine (const int *curline*, int *direction*)

Eat the horizontal direction.

Parameters

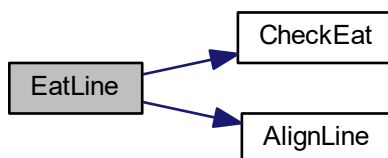
<i>curline</i>	Current line to eat
<i>direction</i>	The direction to eat positive for left and negative for right

Returns

The number of blank grid in the column

Definition at line 345 of file 2048.c.

Here is the call graph for this function:



Here is the caller graph for this function:

**3.1.3.25 int GetRandNums ()**

Get random grid ranged from 0 to MAX_RANDNUM on board.

Returns

The number of random grid generated

Definition at line 356 of file 2048.c.

Here is the call graph for this function:



Here is the caller graph for this function:



3.1.3.26 `int main ()`

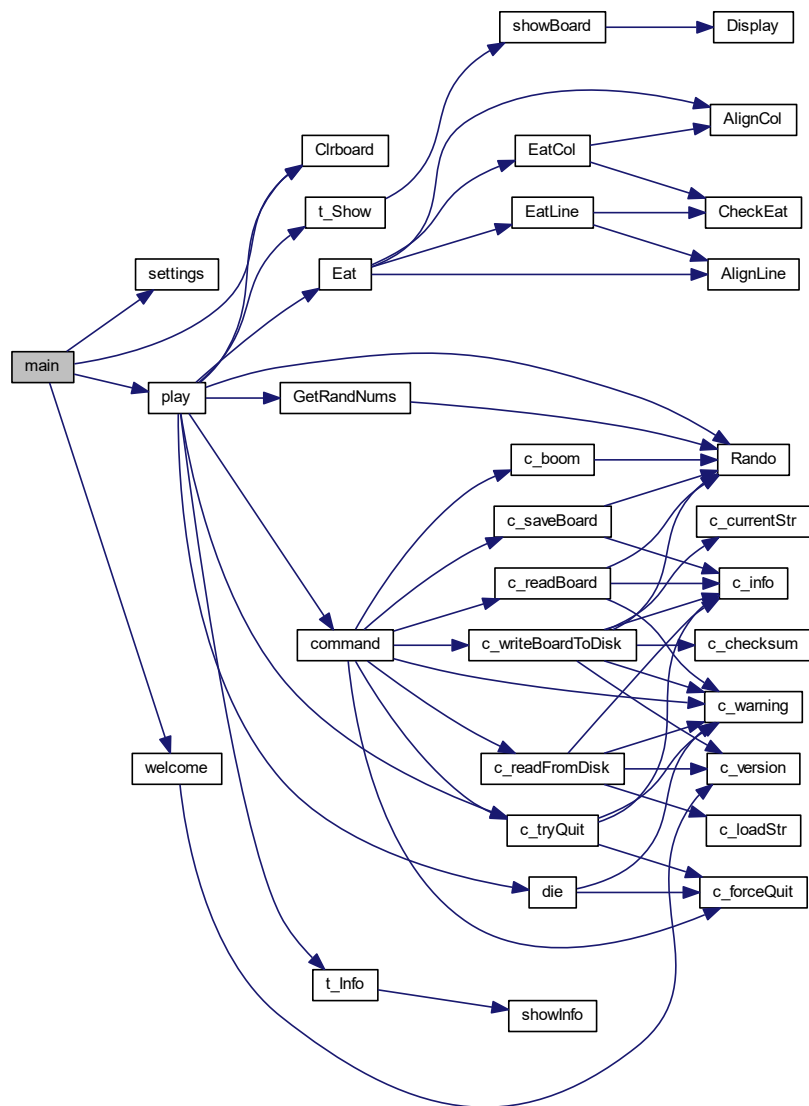
Main executable.

Returns

0

Definition at line 915 of file 2048.c.

Here is the call graph for this function:



3.1.3.27 bool play ()

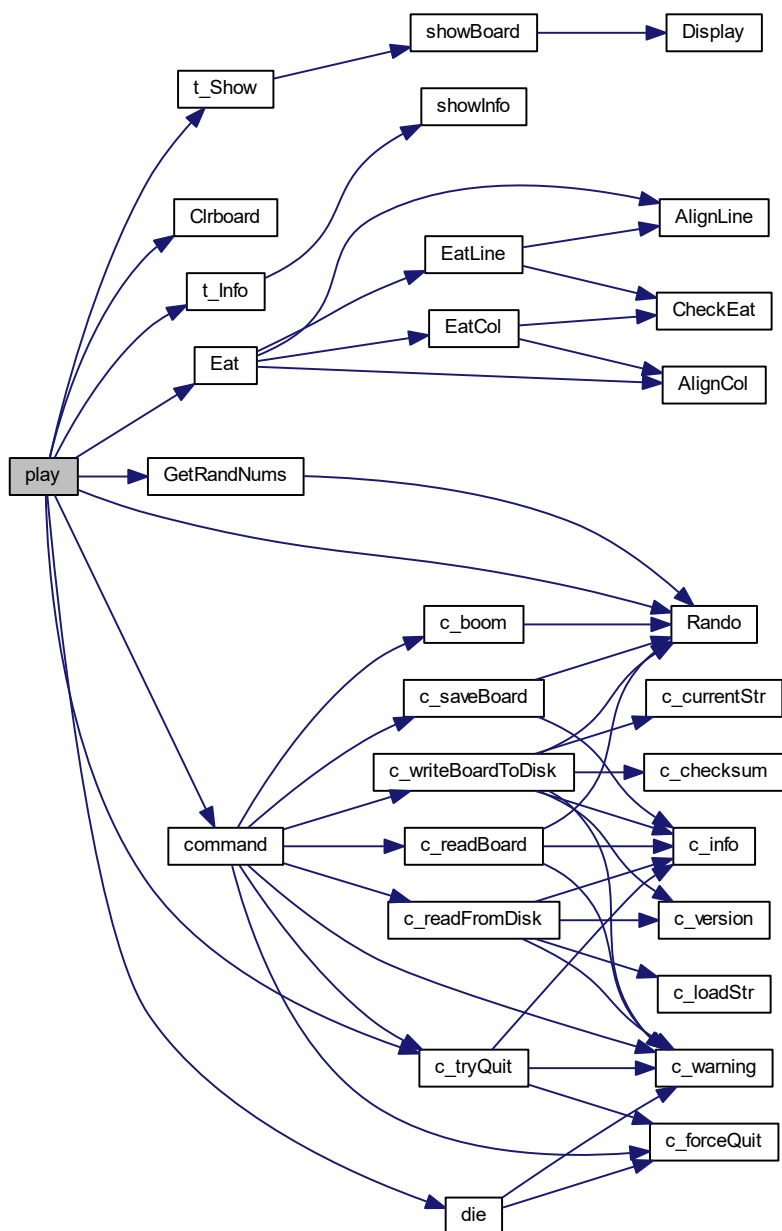
Handle for main game.

Returns

If restart

Definition at line 461 of file 2048.c.

Here is the call graph for this function:



Here is the caller graph for this function:



3.1.3.28 unsigned int Rando (int *N*)

Generate random num ranged from 0 to *N*-1.

Parameters

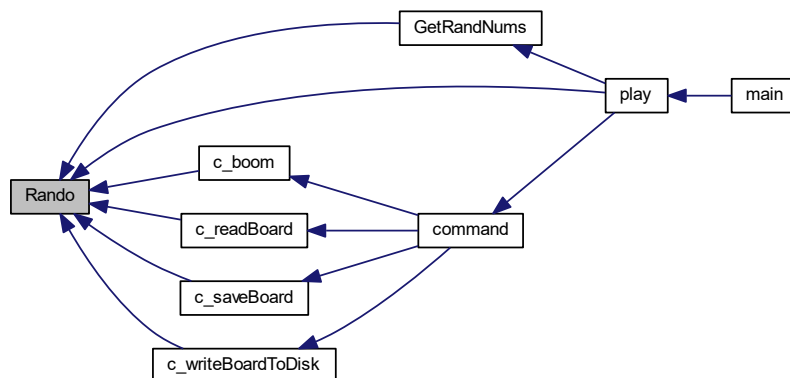
<i>N</i>	The upper bound of the random number
----------	--------------------------------------

Returns

The random number

Definition at line 368 of file 2048.c.

Here is the caller graph for this function:



3.1.3.29 void settings ()

Set the global settings.

TODO:use a ini instead?

Returns

void

Definition at line 125 of file 2048.c.

Here is the caller graph for this function:

**3.1.3.30 void showBoard (WINDOW * win, int offy, int offx)**

Print the board to screen.

Parameters

<i>win</i>	The window to draw on
<i>offy</i>	The y position for the left-up corner
<i>offx</i>	The x position for the left-up corner

Returns

void

Definition at line 535 of file 2048.c.

Here is the call graph for this function:



Here is the caller graph for this function:



3.1.3.31 `void showInfo (WINDOW * win)`

Print the info to screen.

Parameters

<i>win</i>	The window to draw on
------------	-----------------------

Returns

void

Definition at line 557 of file 2048.c.

Here is the caller graph for this function:



3.1.3.32 void * t_Info (void * *arg*)

The thread to print infos to screen.

Parameters

<i>arg</i>	Void
------------	------

Returns

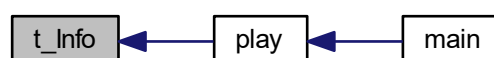
void* NULL

Definition at line 874 of file 2048.c.

Here is the call graph for this function:



Here is the caller graph for this function:



3.1.3.33 void * t_Show (void * arg)

The thread to print Board to screen.

Parameters

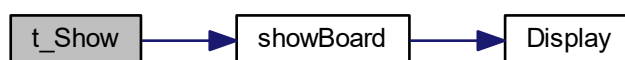
<i>arg</i>	Void
------------	------

Returns

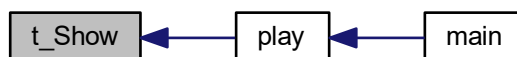
void* NULL

Definition at line 894 of file 2048.c.

Here is the call graph for this function:



Here is the caller graph for this function:



3.1.3.34 void welcome ()

Print welcome message and input the size of the board.

Returns

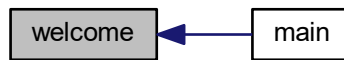
void

Definition at line 569 of file 2048.c.

Here is the call graph for this function:



Here is the caller graph for this function:



3.1.4 Variable Documentation

3.1.4.1 Attr AThread

The attr of threads

Definition at line 115 of file 2048.c.

3.1.4.2 char board[MAX_BOARD_NUM][MAX_BOARD_SIZE][MAX_BOARD_SIZE]

The mutex lock of the board

The boards to storage game progress

Definition at line 71 of file 2048.c.

3.1.4.3 int boardseed[MAX_BOARD_NUM]

The random seed of the boards

Definition at line 75 of file 2048.c.

3.1.4.4 char boardstr[MAX_BOARD_SIZE][MAX_BOARD_SIZE][5]

The output string

Definition at line 73 of file 2048.c.

3.1.4.5 WINDOW* BoardWin

The window to display board

Definition at line 117 of file 2048.c.

3.1.4.6 Cond CBoard

The condition of board

Definition at line 111 of file 2048.c.

3.1.4.7 Cond CInfo

The condition of info

Definition at line 113 of file 2048.c.

3.1.4.8 int col

Definition at line 96 of file 2048.c.

3.1.4.9 const char cs_pwd[PWD_LEN+1] =PWD

The password when generating the checksum

Definition at line 68 of file 2048.c.

3.1.4.10 unsigned char curs =0

Current board

Definition at line 77 of file 2048.c.

3.1.4.11 char display[256][16]

Display table

Will display a as string display[a]

Definition at line 87 of file 2048.c.

3.1.4.12 char eat[256][256][2]

Eat table(TODO:use eat array in CheckEat)

Will set a=eat[a][b][0] and b=eat[a][b][1] when eating a and b

Definition at line 83 of file 2048.c.

3.1.4.13 bool iswarn

If the message is a warning

Definition at line 100 of file 2048.c.

3.1.4.14 int MAX_RANDOM =2

The maxium level of filling an grid.

Definition at line 35 of file 2048.c.

3.1.4.15 Mut MBoard

The mutex of Board

Definition at line 107 of file 2048.c.

3.1.4.16 WINDOW* MenuWin

The window to display info

Definition at line 119 of file 2048.c.

3.1.4.17 Mut MInfo

The mutex of Info

Definition at line 109 of file 2048.c.

3.1.4.18 char N =5

The size of the board

Definition at line 93 of file 2048.c.

3.1.4.19 const int NA =126

Stand for invalid grid.

Definition at line 25 of file 2048.c.

3.1.4.20 int P_RANDNUM =30

The probability of an empty grid becoming filled

Definition at line 34 of file 2048.c.

3.1.4.21 int point[256]

Point table(TODO:use eat array in CheckEat)

Will count a by point[a] when adding up the points

Definition at line 91 of file 2048.c.

3.1.4.22 int row

The size of the screen

Definition at line 96 of file 2048.c.

3.1.4.23 char sinfo[512]

The buffer of the message to print

Definition at line 98 of file 2048.c.

3.1.4.24 Thrd TInfo

The handle of t_Info thread

Definition at line 105 of file 2048.c.

3.1.4.25 Thrd TShow

The handle of t_Show thread

Definition at line 103 of file 2048.c.

3.2 README.md File Reference

Index

2048.c, [5](#)

 AThread, [35](#)

 AlignCol, [9](#)

 AlignLine, [10](#)

 Attr, [9](#)

 board, [35](#)

 BoardWin, [35](#)

 boardseed, [35](#)

 boardstr, [35](#)

 c_boom, [10](#)

 c_checksum, [11](#)

 c_currentStr, [11](#)

 c_forceQuit, [12](#)

 c_info, [12](#)

 c_loadStr, [13](#)

 c_readBoard, [14](#)

 c_readFromDisk, [15](#)

 c_saveBoard, [15](#)

 c_tryQuit, [16](#)

 c_version, [17](#)

 c_warning, [18](#)

 c_writeBoardToDisk, [18](#)

 CBoard, [35](#)

 CInfo, [35](#)

 CheckEat, [19](#)

 Clrboard, [20](#)

 col, [35](#)

 command, [20](#)

 Cond, [9](#)

 cs_pwd, [36](#)

 curs, [36](#)

 die, [21](#)

 Display, [22](#)

 display, [36](#)

 EDOWN, [8](#)

 ELEFT, [8](#)

 ERIGHT, [8](#)

 EUP, [8](#)

 Eat, [22](#)

 eat, [36](#)

 EatCol, [24](#)

 EatLine, [25](#)

 GetRandNums, [26](#)

 iswarn, [36](#)

 MAX_BOARD_NUM, [8](#)

 MAX_BOARD_SIZE, [8](#)

 MAX_RANDOM, [36](#)

 MBoard, [36](#)

 MENU_POSITION_X, [8](#)

 MENU_POSITION_Y, [8](#)

 MInfo, [36](#)

 main, [27](#)

 MenuWin, [36](#)

 Mut, [9](#)

 N, [37](#)

 NA, [37](#)

 P_RANDOM, [37](#)

 PWD, [8](#)

 PWD_LEN, [8](#)

 play, [28](#)

 point, [37](#)

 Rando, [30](#)

 row, [37](#)

 settings, [30](#)

 showBoard, [31](#)

 showInfo, [31](#)

 sinfo, [37](#)

 t_Info, [33](#)

 t_Show, [34](#)

 TInfo, [37](#)

 TShow, [37](#)

 Thrd, [9](#)

 WARNING_POSITION_X, [9](#)

 WARNING_POSITION_Y, [9](#)

 welcome, [34](#)

AThread

 2048.c, [35](#)

AlignCol

 2048.c, [9](#)

AlignLine

 2048.c, [10](#)

Attr

 2048.c, [9](#)

board

 2048.c, [35](#)

BoardWin

 2048.c, [35](#)

boardseed

 2048.c, [35](#)

boardstr

 2048.c, [35](#)

c_boom

 2048.c, [10](#)

c_checksum

 2048.c, [11](#)

c_currentStr

- 2048.c, [11](#)
- c_forceQuit
 - 2048.c, [12](#)
- c_info
 - 2048.c, [12](#)
- c_loadStr
 - 2048.c, [13](#)
- c_readBoard
 - 2048.c, [14](#)
- c_readFromDisk
 - 2048.c, [15](#)
- c_saveBoard
 - 2048.c, [15](#)
- c_tryQuit
 - 2048.c, [16](#)
- c_version
 - 2048.c, [17](#)
- c_warning
 - 2048.c, [18](#)
- c_writeBoardToDisk
 - 2048.c, [18](#)
- CBoard
 - 2048.c, [35](#)
- CInfo
 - 2048.c, [35](#)
- CheckEat
 - 2048.c, [19](#)
- Clrboard
 - 2048.c, [20](#)
- col
 - 2048.c, [35](#)
- command
 - 2048.c, [20](#)
- Cond
 - 2048.c, [9](#)
- cs_pwd
 - 2048.c, [36](#)
- curs
 - 2048.c, [36](#)
- die
 - 2048.c, [21](#)
- Display
 - 2048.c, [22](#)
- display
 - 2048.c, [36](#)
- EDOWN
 - 2048.c, [8](#)
- ELEFT
 - 2048.c, [8](#)
- ERIGHT
 - 2048.c, [8](#)
- EUP
 - 2048.c, [8](#)
- Eat
 - 2048.c, [22](#)
- eat
 - 2048.c, [36](#)
- EatCol
 - 2048.c, [24](#)
- EatLine
 - 2048.c, [25](#)
- GetRandNums
 - 2048.c, [26](#)
- iswarn
 - 2048.c, [36](#)
- MAX_BOARD_NUM
 - 2048.c, [8](#)
- MAX_BOARD_SIZE
 - 2048.c, [8](#)
- MAX_RANDOMUM
 - 2048.c, [36](#)
- MBoard
 - 2048.c, [36](#)
- MENU_POSITION_X
 - 2048.c, [8](#)
- MENU_POSITION_Y
 - 2048.c, [8](#)
- MInfo
 - 2048.c, [36](#)
- main
 - 2048.c, [27](#)
- MenuWin
 - 2048.c, [36](#)
- Mut
 - 2048.c, [9](#)
- N
 - 2048.c, [37](#)
- NA
 - 2048.c, [37](#)
- P_RANDOMUM
 - 2048.c, [37](#)
- PWD
 - 2048.c, [8](#)
- PWD_LEN
 - 2048.c, [8](#)
- play
 - 2048.c, [28](#)
- point
 - 2048.c, [37](#)
- README.md, [38](#)
- Rando
 - 2048.c, [30](#)
- row
 - 2048.c, [37](#)
- settings
 - 2048.c, [30](#)
- showBoard
 - 2048.c, [31](#)
- showInfo
 - 2048.c, [31](#)

sinfo
 2048.c, [37](#)

t_Info
 2048.c, [33](#)

t_Show
 2048.c, [34](#)

TInfo
 2048.c, [37](#)

TShow
 2048.c, [37](#)

Thrd
 2048.c, [9](#)

WARNING_POSITION_X
 2048.c, [9](#)

WARNING_POSITION_Y
 2048.c, [9](#)

welcome
 2048.c, [34](#)