

My Project

Generated by Doxygen 1.8.11

Contents

1	termproject	1
2	Hierarchical Index	3
2.1	Class Hierarchy	3
3	Class Index	5
3.1	Class List	5
4	File Index	7
4.1	File List	7
5	Class Documentation	9
5.1	main_savitch_14::game Class Reference	9
5.1.1	Member Function Documentation	10
5.1.1.1	display_message(const std::string &message) const	10
5.1.1.2	play(char level)	10
5.1.1.3	winning() const	10
5.2	main_savitch_14::Othello Class Reference	11
5.2.1	Member Function Documentation	12
5.2.1.1	all_legal(int a, int b, int c, int d) const	12
5.2.1.2	compute_moves(std::queue< std::string > &moves) const	13
5.2.1.3	computer_legal() const	13
5.2.1.4	display_message(const std::string &message) const	13
5.2.1.5	display_status() const	13
5.2.1.6	down_legal(int a, int b, int c, int d) const	14

5.2.1.7	<code>evaluate(char level) const</code>	14
5.2.1.8	<code>flip_flip(int a, int b, int c, int d)</code>	14
5.2.1.9	<code>is_game_over() const</code>	15
5.2.1.10	<code>is_legal(const std::string &move) const</code>	15
5.2.1.11	<code>left_down_legal(int a, int b, int c, int d) const</code>	15
5.2.1.12	<code>left_legal(int a, int b, int c, int d) const</code>	16
5.2.1.13	<code>left_up_legal(int a, int b, int c, int d) const</code>	16
5.2.1.14	<code>pass(std::string move)</code>	16
5.2.1.15	<code>right_down_legal(int a, int b, int c, int d) const</code>	16
5.2.1.16	<code>right_legal(int a, int b, int c, int d) const</code>	17
5.2.1.17	<code>right_up_legal(int a, int b, int c, int d) const</code>	17
5.2.1.18	<code>up_legal(int a, int b, int c, int d) const</code>	17
5.2.1.19	<code>winning() const</code>	17
5.3	<code>main_savitch_14::Piece Class Reference</code>	18
6	File Documentation	19
6.1	<code>game.cc File Reference</code>	19
6.1.1	Detailed Description	19
6.2	<code>main.cc File Reference</code>	20
6.2.1	Detailed Description	20
6.3	<code>othello.cc File Reference</code>	21
6.3.1	Detailed Description	21
Index		23

Chapter 1

termproject

Chapter 2

Hierarchical Index

2.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

main_savitch_14::game	9
main_savitch_14::Othello	11
main_savitch_14::Piece	18

Chapter 3

Class Index

3.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

main_savitch_14::game	9
main_savitch_14::Othello	11
main_savitch_14::Piece	18

Chapter 4

File Index

4.1 File List

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

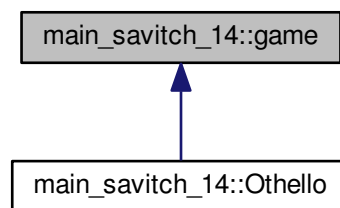
colors.h	??
game.cc	This is a file includes moves, winer and some other functions	19
game.h	??
main.cc	This is the main function	20
othello.cc	This is a file includes check_move check game over and some other functions	21
othello.h	??
piece.h	??

Chapter 5

Class Documentation

5.1 main_savitch_14::game Class Reference

Inheritance diagram for main_savitch_14::game:



Public Types

- enum **who** { **HUMAN**, **NEUTRAL**, **COMPUTER** }

Public Member Functions

- who **play** (char level)

Protected Member Functions

- virtual void **display_message** (const std::string &message) const
display message a message
- virtual std::string **get_user_move** () const
- virtual who **last_mover** () const
- virtual int **moves_completed** () const
- virtual who **next_mover** () const

- virtual who **opposite** (who player) const
- virtual who **winning** () const
check who is the winner
- virtual void **make_move** (const std::string &move)
- virtual void **restart** ()
- virtual **game** * **clone** () const =0
- virtual void **compute_moves** (std::queue< std::string > &moves) const =0
- virtual void **display_status** () const =0
- virtual int **evaluate** (char level) const =0
- virtual bool **is_game_over** () const =0
- virtual bool **is_legal** (const std::string &move) const =0

5.1.1 Member Function Documentation

5.1.1.1 void main_savitch_14::game::display_message (const std::string & *message*) const [protected],
[virtual]

display message a message

Parameters

<i>message</i>	the message you want to display
----------------	---------------------------------

Reimplemented in [main_savitch_14::Othello](#).

5.1.1.2 game::who main_savitch_14::game::play (char *level*)

The play function should not be overridden. It plays one round of the game, with the human player moving first and the computer second. The return value is the winner of the game (or NEUTRAL for a tie). The commenting you see below sets this up for Phase One

5.1.1.3 game::who main_savitch_14::game::winning () const [protected],[virtual]

check who is the winner

See also

next_mover()
last_mover()

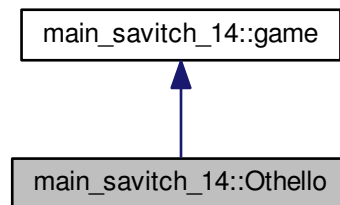
Reimplemented in [main_savitch_14::Othello](#).

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

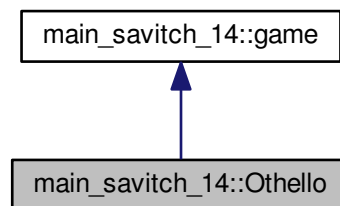
- game.h
- [game.cc](#)

5.2 main_savitch_14::Othello Class Reference

Inheritance diagram for main_savitch_14::Othello:



Collaboration diagram for main_savitch_14::Othello:



Public Member Functions

- **Othello** (const [Othello](#) &other)
- [game](#) * **clone** () const
- void **make_move** (const std::string &move)
- void [restart](#) ()
restart the game, set up four original locations
- void [display_status](#) () const
display current status
- bool [is_legal](#) (const std::string &move) const
check if the location player choose is legal or not
- void [display_message](#) (const std::string &message) const
display the message for next step
- bool [is_game_over](#) () const
check if game is over
- void [compute_moves](#) (std::queue< std::string > &moves) const
make computer move

- int [evaluate](#) (char level) const
for AI the make sure different location has different value
- void [flip_flip](#) (int a, int b, int c, int d)
flip_flip after someone make moves
- void [pass](#) (std::string move)
check if someone can move next step move
- bool [up_legal](#) (int a, int b, int c, int d) const
check if up side is legal
- bool [down_legal](#) (int a, int b, int c, int d) const
check if down side is legal
- bool [left_legal](#) (int a, int b, int c, int d) const
check if left side is legal
- bool [right_legal](#) (int a, int b, int c, int d) const
check if right side is legal
- bool [left_up_legal](#) (int a, int b, int c, int d) const
check if left up side is legal
- bool [right_up_legal](#) (int a, int b, int c, int d) const
check if right up side is legal
- bool [left_down_legal](#) (int a, int b, int c, int d) const
check if left down side is legal
- bool [right_down_legal](#) (int a, int b, int c, int d) const
check if right down side is legal
- bool [all_legal](#) (int a, int b, int c, int d) const
combine all check sides
- bool [human_legal](#) () const
- bool [computer_legal](#) () const
check computer step is legal
- who [winning](#) () const
check who is the winnner

Additional Inherited Members

5.2.1 Member Function Documentation

5.2.1.1 bool main_savitch_14::Othello::all_legal (int a, int b, int c, int d) const

combine all check sides

See also

[up_legal\(\)](#)
[down_legal\(\)](#)
[left_legal\(\)](#)
[right_legal\(\)](#)
[left_up_legal\(\)](#)
[right_up_legal\(\)](#)
[left_down_legal\(\)](#)
[right_down_legal\(\)](#)
[human_legal\(\)](#)
[get_piece\(\)](#)
[all_legal\(\)](#)

Returns

return true if it is legal

5.2.1.2 void main_savitch_14::Othello::compute_moves (std::queue< std::string > & moves) const [virtual]

make computer move

See also

[is_legal\(\)](#)

Implements [main_savitch_14::game](#).

5.2.1.3 bool main_savitch_14::Othello::computer_legal () const

check computer step is legal

See also

[get_piece\(\)](#)

[all_legal\(\)](#)

Returns

return true if it is legal

5.2.1.4 void main_savitch_14::Othello::display_message (const std::string & message) const [virtual]

display the message for next step

Parameters

<i>message</i>	it is used for checking if it is legal move
----------------	---

See also

[next_move\(\)](#)

[display_message\(\)](#)

Reimplemented from [main_savitch_14::game](#).

5.2.1.5 void main_savitch_14::Othello::display_status () const [virtual]

display current status

See also

[is_black\(\)](#)

[is_white\(\)](#)

[is_empty\(\)](#)

[is_game_over\(\)](#)

Implements [main_savitch_14::game](#).

5.2.1.6 `bool main_savitch_14::Othello::down_legal (int a, int b, int c, int d) const`

check if down side is legal

See also

`get_piece()`

Returns

return true if it is legal

5.2.1.7 `int main_savitch_14::Othello::evaluate (char level) const` `[virtual]`

for AI the make sure different location has different value

See also

`is_black()`

`is_white()`

Returns

the total evalueta value to know which location the best choice

Implements [main_savitch_14::game](#).

5.2.1.8 `void main_savitch_14::Othello::flip_flip (int a, int b, int c, int d)`

flip_flip after someone make moves

See also

`get_piece()`

`up_legal()`

`down_legal()`

`left_legal()`

`right_legal()`

`left_up_legal()`

`right_up_legal()`

`left_down_legal()`

`right_down_legal()`

5.2.1.9 bool main_savitch_14::Othello::is_game_over () const [virtual]

check if game is over

Returns

return true if game is over

See also

get_piese()
human_legal()
[computer_legal\(\)](#)

Returns

return true if the game is over

Implements [main_savitch_14::game](#).

5.2.1.10 bool main_savitch_14::Othello::is_legal (const std::string & move) const [virtual]

check if the location player choose is legal or not

Parameters

<i>move</i>	the location which player choose
-------------	----------------------------------

See also

next_move()
get_piese()
[all_legal\(\)](#)

Returns

return true if the move is legal otherwise false

Implements [main_savitch_14::game](#).

5.2.1.11 bool main_savitch_14::Othello::left_down_legal (int a, int b, int c, int d) const

check if left down side is legal

See also

get_piece()

Returns

return true if it is legal

5.2.1.12 `bool main_savitch_14::Othello::left_legal (int a, int b, int c, int d) const`

check if left side is legal

See also

`get_piece()`

Returns

return true if it is legal

5.2.1.13 `bool main_savitch_14::Othello::left_up_legal (int a, int b, int c, int d) const`

check if left up side is legal

See also

`get_piece()`

Returns

return true if it is legal

5.2.1.14 `void main_savitch_14::Othello::pass (std::string move)`

check if someone can move next step move

See also

[is_game_over\(\)](#)
[make_move\(\)](#)
[computer_legal\(\)](#)
[human_legal\(\)](#)

5.2.1.15 `bool main_savitch_14::Othello::right_down_legal (int a, int b, int c, int d) const`

check if right down side is legal

See also

`get_piece()`

Returns

return true if it is legal

5.2.1.16 `bool main_savitch_14::Othello::right_legal (int a, int b, int c, int d) const`

check if right side is legal

See also

`get_piece()`

Returns

return true if it is legal

5.2.1.17 `bool main_savitch_14::Othello::right_up_legal (int a, int b, int c, int d) const`

check if right up side is legal

See also

`get_piece()`

Returns

return true if it is legal

5.2.1.18 `bool main_savitch_14::Othello::up_legal (int a, int b, int c, int d) const`

check if up side is legal

See also

`get_piece()`

Returns

return true if it is legal

5.2.1.19 `game::who main_savitch_14::Othello::winning () const` `[virtual]`

check who is the winner

See also

`get_piece()`

Returns

return who wins this game otherwise return neutral

Reimplemented from [main_savitch_14::game](#).

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

- `othello.h`
- [othello.cc](#)

5.3 main_savitch_14::Piece Class Reference

Public Member Functions

- int **get_horizontal** ()
- int **get_vertical** ()
- int **get_piece** (int i, int j)
- void **flip** (int i, int j, int p)
- bool **is_empty** (int i, int j)
- bool **is_black** (int i, int j)
- bool **is_white** (int i, int j)

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

- piece.h

Chapter 6

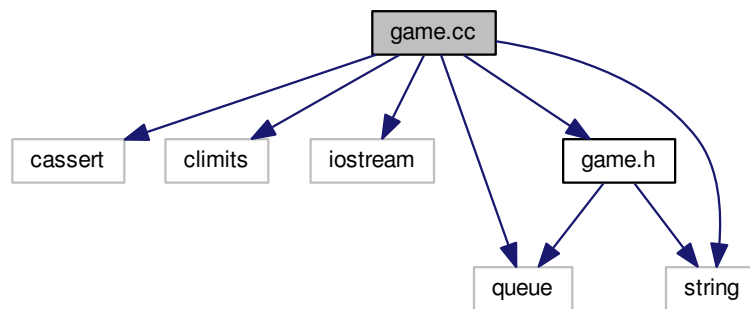
File Documentation

6.1 game.cc File Reference

This is a file includes moves, winer and some other functions.

```
#include <cassert>
#include <climits>
#include <iostream>
#include <queue>
#include <string>
#include "game.h"
```

Include dependency graph for game.cc:



6.1.1 Detailed Description

This is a file includes moves, winer and some other functions.

Author

Shipeng Yang, Zhaojie Chen, Bohong Li, Xudong Yuan

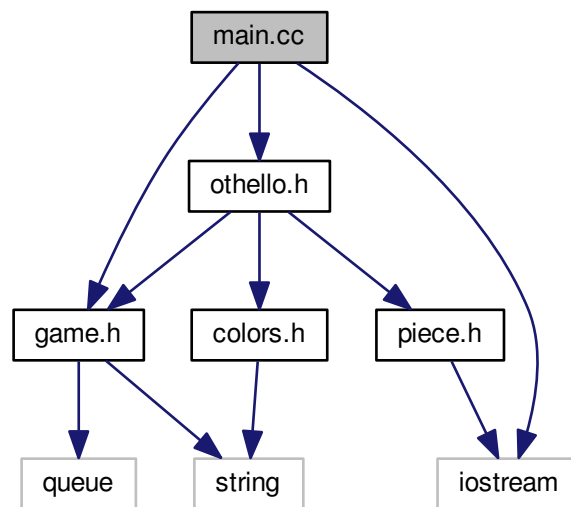
Date

2017/11/12

6.2 main.cc File Reference

This is the main function.

```
#include "game.h"  
#include "othello.h"  
#include <iostream>  
Include dependency graph for main.cc:
```



Functions

- `int main ()`

6.2.1 Detailed Description

This is the main function.

Author

Shipeng Yang, Zhaojie Chen, Bohong Li, Xudong Yuan

Date

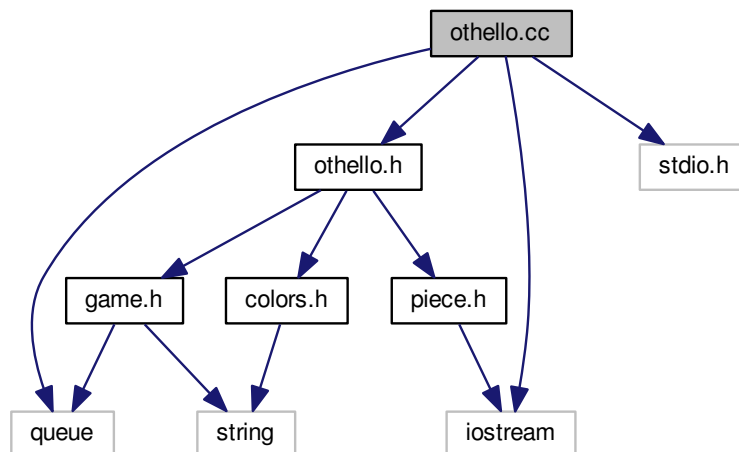
2017/11/12

6.3 othello.cc File Reference

This is a file includes check_move check game over and some other functions.

```
#include "othello.h"  
#include <iostream>  
#include <stdio.h>  
#include <queue>
```

Include dependency graph for othello.cc:



6.3.1 Detailed Description

This is a file includes check_move check game over and some other functions.

Author

Shipeng Yang, Zhaojie Chen, Bohong Li, Xudong Yuan

Date

2017/11/12

Index

- all_legal
 - main_savitch_14::Othello, [12](#)
- compute_moves
 - main_savitch_14::Othello, [12](#)
- computer_legal
 - main_savitch_14::Othello, [13](#)
- display_message
 - main_savitch_14::Othello, [13](#)
 - main_savitch_14::game, [10](#)
- display_status
 - main_savitch_14::Othello, [13](#)
- down_legal
 - main_savitch_14::Othello, [13](#)
- evaluate
 - main_savitch_14::Othello, [14](#)
- flip_flip
 - main_savitch_14::Othello, [14](#)
- game.cc, [19](#)
- is_game_over
 - main_savitch_14::Othello, [14](#)
- is_legal
 - main_savitch_14::Othello, [15](#)
- left_down_legal
 - main_savitch_14::Othello, [15](#)
- left_legal
 - main_savitch_14::Othello, [15](#)
- left_up_legal
 - main_savitch_14::Othello, [16](#)
- main.cc, [20](#)
- main_savitch_14::Othello, [11](#)
 - all_legal, [12](#)
 - compute_moves, [12](#)
 - computer_legal, [13](#)
 - display_message, [13](#)
 - display_status, [13](#)
 - down_legal, [13](#)
 - evaluate, [14](#)
 - flip_flip, [14](#)
 - is_game_over, [14](#)
 - is_legal, [15](#)
 - left_down_legal, [15](#)
 - left_legal, [15](#)
 - left_up_legal, [16](#)
 - pass, [16](#)
 - right_down_legal, [16](#)
 - right_legal, [16](#)
 - right_up_legal, [17](#)
 - up_legal, [17](#)
 - winning, [17](#)
- main_savitch_14::Piece, [18](#)
- main_savitch_14::game, [9](#)
 - display_message, [10](#)
 - play, [10](#)
 - winning, [10](#)
- othello.cc, [21](#)
- pass
 - main_savitch_14::Othello, [16](#)
- play
 - main_savitch_14::game, [10](#)
- right_down_legal
 - main_savitch_14::Othello, [16](#)
- right_legal
 - main_savitch_14::Othello, [16](#)
- right_up_legal
 - main_savitch_14::Othello, [17](#)
- up_legal
 - main_savitch_14::Othello, [17](#)
- winning
 - main_savitch_14::Othello, [17](#)
 - main_savitch_14::game, [10](#)