nulocks-core 0.1

Generated by Doxygen 1.8.5

Tue May 6 2014 13:38:46

Contents

1	nulo	cks-cor	е														1
2	Nam	espace	Index														3
	2.1	Names	space List							 	 	 		 			3
3	Clas	s Index															5
	3.1	Class I	_ist							 	 	 		 			5
4	File	Index															7
	4.1	File Lis	st							 	 	 		 			7
5	Nam	espace	Docume	ntation	ı												9
	5.1	Nulock	s Namesp	oace Re	eferenc	ce				 	 	 		 			9
		5.1.1	Detailed	Descri	ption					 	 	 		 			9
	5.2	Nulock	s::Core N	amespa	ace Re	eferen	ice			 	 	 		 			9
		5.2.1	Detailed	Descri	ption					 	 	 		 			9
6	Clas	s Docu	mentatio	n													11
	6.1	Nulock	s::Core::E	Block Cl	ass Re	eferer	nce			 	 	 		 			11
		6.1.1	Detailed	Descri	ption					 	 	 		 			11
		6.1.2	Member	Function	on Doc	cumer	ntatio	n .		 	 	 		 			12
			6.1.2.1	getLe	evel .					 	 	 		 			12
			6.1.2.2	getVa	alue .					 	 	 		 			12
			6.1.2.3	isEm	pty .					 	 	 		 			12
			6.1.2.4	setLe	vel .					 	 	 		 			12
			6.1.2.5	wasN	loved					 	 	 		 			12
	6.2	Nulock	s::Core::E	Board C	lass R	lefere	nce			 	 	 		 			12
		6.2.1	Detailed	Descri	ption					 	 	 		 			13
		6.2.2	Construc	ctor & D)estruc	ctor D	ocun	nenta	tion	 	 	 		 			13
			6.2.2.1	Board	d					 	 	 		 			13
			6.2.2.2	Board	d					 	 	 		 			13
		6.2.3	Member	Function	on Doc	cumer	ntatio	n .		 	 	 		 			14
			6.2.3.1	game	InProd	aress				 	 	 					14

iv CONTENTS

			6.2.3.2		neLost															
			6.2.3.3	gan	neWon			٠	 			 	•		 ٠	 		•		14
			6.2.3.4	get	Repres	enta	tion		 			 		 		 				14
			6.2.3.5	get	Score				 			 				 				14
			6.2.3.6	get	Size .				 			 		 		 				14
			6.2.3.7	get	WinPov	ver .			 			 		 		 				15
7	File	Docum	entation																	17
	7.1	Block.l	n File Refe	erence	e				 			 		 		 				17
		7.1.1	Detailed	Desc	ription				 			 		 		 				17
	7.2	Board.	h File Ref	erenc	е				 			 		 		 				17
		7.2.1	Detailed	Desc	ription				 			 		 		 				18
In	dex																			19

nulocks-core

nulocks - NUmerical bLOCKS is a modular and customizable clone of 2048 game in C++. This is a repository with core engine of the game.

The main purpose of Nulocks is to provide an engine of a game similar to original 2048 and an option to visualize it in any interface you want to implement it in. This repository contains this very engine that can be used to implement 2048 and/or use extended Nulocks possibilities such as determined by user size of the board and ending level of block in any user interface you want.

If you want to contribute, start with the documentation at: http://www.phitherek.mooo.com/doxy/nulocks-core/. I would be happy to see Nulocks implemented for many platforms in many graphical libraries.

Installation

make

sudo make install

Make macros: LIBPREFIX - determines a directory in which the shared library should be placed, default: /usr/lib, INCLUDEPREFIX - determines a directory in which the header files should be placed, default: /usr/include. Use them with the install command.

Usage

In your program include the main header file:

#include "nulocks-core/Board.h"

Compile your program with:

g++ -o your_executable your_source.cpp -lnulockscore

2 nulocks-core

Namespace Index

2.1 Namespace List

Here	is a	list of	all	documented	namespaces	with	brief	descriptions
1 1010	io a	IISL O	an	accumented	Hairiespaces	VVILII	וטווטו	ucscriptions.

Nulocks	
A global namespace for all Nulocks game components	9
Nulocks::Core	
A namespace for Nulocks core game engine	9

Namespace Index

Class Index

3.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:	
Nulocks::Core::Block	
A class representing a block on the game board	11
Nulocks::Core::Board	

6 Class Index

File Index

4.1 File List

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

Block.h		
	A file that contains the class representing a block on the game board	17
Board.h		
	A file that contains main core engine class of Nulocks - modular and customizable clone of 2048	
	game	17

8 File Index

Namespace Documentation

5.1 Nulocks Namespace Reference

A global namespace for all Nulocks game components.

Namespaces

Core

A namespace for Nulocks core game engine.

5.1.1 Detailed Description

A global namespace for all Nulocks game components.

5.2 Nulocks::Core Namespace Reference

A namespace for Nulocks core game engine.

Classes

• class Block

A class representing a block on the game board.

class Board

A class that represents Nulocks board.

5.2.1 Detailed Description

A namespace for Nulocks core game engine.

Names	pace	Docu	ment	tation

Class Documentation

6.1 Nulocks::Core::Block Class Reference

A class representing a block on the game board.

```
#include <Block.h>
```

Public Member Functions

• Block ()

Construct an empty block.

• ∼Block ()

Destroy the block.

void setMoved ()

Indicate the block as moved this turn.

void unsetMoved ()

Indicate the block as not moved this turn.

• bool wasMoved ()

Check if the block was moved this turn.

• int getValue ()

Get block value.

• int getLevel ()

Get block level.

• void levelUp ()

Increase level of the block.

· void empty ()

Make the block empty.

• bool isEmpty ()

Check if the block is empty.

void setLevel (int level)

Set level of the block.

6.1.1 Detailed Description

A class representing a block on the game board.

12 Class Documentation

6.1.2 Member Function Documentation 6.1.2.1 int Nulocks::Core::Block::getLevel () Get block level. Returns Level of the block. 6.1.2.2 int Nulocks::Core::Block::getValue () Get block value. Returns

6.1.2.3 bool Nulocks::Core::Block::isEmpty ()

Check if the block is empty.

Value of the block.

Returns

True if the block is empty, false otherwise.

6.1.2.4 void Nulocks::Core::Block::setLevel (int level)

Set level of the block.

Parameters

level Level of the block.

6.1.2.5 bool Nulocks::Core::Block::wasMoved ()

Check if the block was moved this turn.

Returns

True if block was moved this turn, false otherwise.

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

- Block.h
- · Block.cpp

6.2 Nulocks::Core::Board Class Reference

A class that represents Nulocks board.

#include <Board.h>

Public Member Functions

Board (int size=4, int winpower=11)

Initializes the board for the new game.

∼Board ()

Destroys the board, frees the memory.

Board (const Board &cpy)

A copy constructor.

• void moveUp ()

Performs a move upwards and starts new turn.

• void moveDown ()

Performs a move downwards and starts new turn.

· void moveLeft ()

Performs a move to the left and starts new turn.

• void moveRight ()

Performs a move to the right and starts new turn.

• bool gameInProgress ()

Checks if game is in progress.

• bool gameWon ()

Checks if game has been won.

· bool gameLost ()

Checks if game has been lost.

• int getScore ()

Returns the current score.

• int getWinPower ()

Returns the power of 2 required to win the game.

• int getSize ()

Returns size of the board.

void getRepresentation (int ***repr)

Returns representation of the board as an array of integers.

6.2.1 Detailed Description

A class that represents Nulocks board.

6.2.2 Constructor & Destructor Documentation

6.2.2.1 Nulocks::Core::Board::Board (int size = 4, int winpower = 11)

Initializes the board for the new game.

Parameters

size	Size of the board, defaults to 4 as in original 2048.
winpower	Power of 2 to get to win the game. Defaults to 11, as in original 2048 game.

6.2.2.2 Nulocks::Core::Board::Board (const Board & cpy)

A copy constructor.

14 Class Documentation

Parameters

cpy A Board object to be copied.

6.2.3 Member Function Documentation

6.2.3.1 bool Nulocks::Core::Board::gameInProgress ()

Checks if game is in progress.

Returns

True if game is in progress, false otherwise.

6.2.3.2 bool Nulocks::Core::Board::gameLost()

Checks if game has been lost.

Returns

True if game has been lost, false otherwise.

6.2.3.3 bool Nulocks::Core::Board::gameWon()

Checks if game has been won.

Returns

True if game has been won, false otherwise.

6.2.3.4 void Nulocks::Core::Board::getRepresentation (int *** repr)

Returns representation of the board as an array of integers.

Parameters

in,out	repr	Takes address of a two-dimensional array of integers with the same size as
		the board. Returns integer representation of the board to this array. Memory
		allocation and freeing must be handled outside this method.

6.2.3.5 int Nulocks::Core::Board::getScore ()

Returns the current score.

Returns

Current score of the player.

6.2.3.6 int Nulocks::Core::Board::getSize ()

Returns size of the board.

Returns

Size of the board.

6.2.3.7 int Nulocks::Core::Board::getWinPower ()

Returns the power of 2 required to win the game.

Returns

Power of 2 to get to win the game.

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

- Board.h
- Board.cpp

16 **Class Documentation**

File Documentation

7.1 Block.h File Reference

A file that contains the class representing a block on the game board.

Classes

· class Nulocks::Core::Block

A class representing a block on the game board.

Namespaces

Nulocks

A global namespace for all Nulocks game components.

· Nulocks::Core

A namespace for Nulocks core game engine.

7.1.1 Detailed Description

A file that contains the class representing a block on the game board.

Author

Phitherek_

Date

2014

Version

0.1

7.2 Board.h File Reference

A file that contains main core engine class of Nulocks - modular and customizable clone of 2048 game.

```
#include "Block.h"
```

18 File Documentation

Classes

· class Nulocks::Core::Board

A class that represents Nulocks board.

Namespaces

Nulocks

A global namespace for all Nulocks game components.

• Nulocks::Core

A namespace for Nulocks core game engine.

7.2.1 Detailed Description

A file that contains main core engine class of Nulocks - modular and customizable clone of 2048 game.

Author

Phitherek_

Date

2014

Version

0.1

Index

```
Block.h, 17
Board
    Nulocks::Core::Board, 13
Board.h, 17
gameInProgress
    Nulocks::Core::Board, 14
gameLost
    Nulocks::Core::Board, 14
gameWon
    Nulocks::Core::Board, 14
getLevel
    Nulocks::Core::Block, 12
getRepresentation
    Nulocks::Core::Board, 14
getScore
    Nulocks::Core::Board, 14
getSize
    Nulocks::Core::Board, 14
getValue
    Nulocks::Core::Block, 12
getWinPower
    Nulocks::Core::Board, 14
isEmpty
    Nulocks::Core::Block, 12
Nulocks, 9
Nulocks::Core, 9
Nulocks::Core::Block, 11
    getLevel, 12
    getValue, 12
    isEmpty, 12
    setLevel, 12
    wasMoved, 12
Nulocks::Core::Board, 12
    Board, 13
    gameInProgress, 14
    gameLost, 14
    gameWon, 14
    getRepresentation, 14
    getScore, 14
    getSize, 14
    getWinPower, 14
setLevel
    Nulocks::Core::Block, 12
wasMoved
    Nulocks::Core::Block, 12
```