

tAltris
v1.0

Generated by Doxygen 1.8.13

Contents

1	Data Structure Index	1
1.1	Data Structures	1
2	File Index	3
2.1	File List	3
3	Data Structure Documentation	5
3.1	AiCoefs Struct Reference	5
3.1.1	Field Documentation	5
3.1.1.1	agg_height	5
3.1.1.2	bumpiness	5
3.1.1.3	clears	5
3.1.1.4	holes	6
3.2	Board Struct Reference	6
3.2.1	Field Documentation	6
3.2.1.1	cells	6
3.2.1.2	height	6
3.2.1.3	width	6
3.3	Piece Struct Reference	7
3.3.1	Field Documentation	7
3.3.1.1	angle	7
3.3.1.2	shape	7
3.3.1.3	type	7
3.3.1.4	x	8

3.3.1.5	y	8
3.4	PieceQueue Struct Reference	8
3.4.1	Field Documentation	8
3.4.1.1	data	8
3.4.1.2	length	8
3.5	PieceShape Struct Reference	8
3.5.1	Field Documentation	9
3.5.1.1	fill	9
3.5.1.2	shape	9
3.6	State Struct Reference	9
3.6.1	Field Documentation	10
3.6.1.1	board	10
3.6.1.2	broken_lines	10
3.6.1.3	current_piece	10
3.6.1.4	input_counts	10
3.6.1.5	level	10
3.6.1.6	next_piece	11
3.6.1.7	piece_queue	11
3.6.1.8	piece_queue_index	11
3.6.1.9	score	11
3.6.1.10	step	11

4	File Documentation	13
4.1	src/ai/genetic/core.c File Reference	13
4.1.1	Detailed Description	14
4.1.2	Function Documentation	14
4.1.2.1	genetic_show_stats()	14
4.2	src/ai/genetic/core.h File Reference	14
4.2.1	Detailed Description	15
4.2.2	Function Documentation	15
4.2.2.1	genetic_show_stats()	15
4.3	src/ai/genetic/engine.c File Reference	16
4.3.1	Detailed Description	16
4.3.2	Function Documentation	16
4.3.2.1	get_coefs()	17
4.3.2.2	get_rank()	17
4.4	src/ai/genetic/engine.h File Reference	17
4.4.1	Detailed Description	18
4.4.2	Function Documentation	18
4.4.2.1	get_coefs()	18
4.4.2.2	get_rank()	18
4.5	src/ai/genetic/tools.c File Reference	19
4.5.1	Detailed Description	19
4.5.2	Function Documentation	19
4.5.2.1	aggregate_height()	20
4.5.2.2	board_height()	20
4.5.2.3	board_heights()	20
4.5.2.4	bumpiness()	20
4.5.2.5	clears()	20
4.5.2.6	hole()	20
4.5.2.7	holes()	21
4.5.2.8	show_features()	21

4.6	src/ai/genetic/tools.h File Reference	21
4.6.1	Detailed Description	22
4.6.2	Macro Definition Documentation	23
4.6.2.1	ABS	23
4.6.3	Function Documentation	23
4.6.3.1	aggregate_height()	23
4.6.3.2	board_height()	23
4.6.3.3	board_heights()	23
4.6.3.4	bumpiness()	23
4.6.3.5	clears()	24
4.6.3.6	coalescent_clears()	24
4.6.3.7	hole()	24
4.6.3.8	holes()	24
4.6.3.9	show_features()	24
4.7	src/debug/debug.h File Reference	25
4.7.1	Detailed Description	25
4.7.2	Macro Definition Documentation	26
4.7.2.1	DEBUG_TAG	26
4.8	src/debug/engine/debug_state.c File Reference	26
4.8.1	Detailed Description	27
4.8.2	Function Documentation	27
4.8.2.1	debug_state_print()	27
4.8.2.2	debug_state_print_cell()	27
4.8.2.3	debug_state_print_infos()	27
4.8.2.4	debug_state_print_line_number()	28
4.8.2.5	debug_state_print_next_piece()	28
4.9	src/debug/engine/debug_state.h File Reference	28
4.9.1	Detailed Description	29
4.9.2	Macro Definition Documentation	29
4.9.2.1	DEBUG_STATE_COLOR	29

4.9.2.2	DEBUG_STATE_NAME	29
4.9.2.3	DEBUG_STATE_TAG	29
4.9.3	Function Documentation	30
4.9.3.1	debug_state_print()	30
4.9.3.2	debug_state_print_cell()	30
4.9.3.3	debug_state_print_infos()	30
4.9.3.4	debug_state_print_line_number()	30
4.9.3.5	debug_state_print_next_piece()	30
4.10	src/engine/angle.h File Reference	31
4.10.1	Detailed Description	31
4.10.2	Macro Definition Documentation	32
4.10.2.1	ANGLE_ESIZE	32
4.10.3	Enumeration Type Documentation	32
4.10.3.1	Angle	32
4.10.3.2	Rotation	32
4.11	src/engine/board.c File Reference	32
4.11.1	Detailed Description	33
4.11.2	Function Documentation	33
4.11.2.1	board_break_lines()	34
4.11.2.2	board_copy()	34
4.11.2.3	board_create()	34
4.11.2.4	board_free()	34
4.11.2.5	board_get_completed_lines()	34
4.11.2.6	board_init()	34
4.11.2.7	board_merge_piece()	35
4.12	src/engine/board.h File Reference	35
4.12.1	Detailed Description	36
4.12.2	Macro Definition Documentation	36
4.12.2.1	BOARD_HEIGHT	36
4.12.2.2	BOARD_HIDDEN	36

4.12.2.3	board_reverse_y	37
4.12.2.4	BOARD_WIDTH	37
4.12.3	Function Documentation	37
4.12.3.1	board_break_lines()	37
4.12.3.2	board_copy()	37
4.12.3.3	board_create()	37
4.12.3.4	board_free()	37
4.12.3.5	board_get_completed_lines()	38
4.12.3.6	board_init()	38
4.12.3.7	board_merge_piece()	38
4.13	src/engine/cell.h File Reference	38
4.13.1	Detailed Description	39
4.13.2	Macro Definition Documentation	39
4.13.2.1	CELL_ESIZE	39
4.13.3	Enumeration Type Documentation	39
4.13.3.1	Cell	39
4.14	src/engine/input.h File Reference	40
4.14.1	Detailed Description	40
4.14.2	Macro Definition Documentation	40
4.14.2.1	INPUT_ESIZE	41
4.14.3	Enumeration Type Documentation	41
4.14.3.1	Input	41
4.15	src/engine/motion.c File Reference	41
4.15.1	Detailed Description	42
4.15.2	Function Documentation	42
4.15.2.1	motion_can_move()	42
4.15.2.2	motion_can_rotate()	42
4.15.2.3	motion_is_valid()	42
4.15.2.4	motion_try_down()	43
4.15.2.5	motion_try_move()	43

4.15.2.6	<code>motion_try_rotate()</code>	43
4.16	<code>src/engine/motion.h</code> File Reference	43
4.16.1	Detailed Description	44
4.16.2	Function Documentation	44
4.16.2.1	<code>motion_can_move()</code>	44
4.16.2.2	<code>motion_can_rotate()</code>	45
4.16.2.3	<code>motion_is_valid()</code>	45
4.16.2.4	<code>motion_try_down()</code>	45
4.16.2.5	<code>motion_try_move()</code>	45
4.16.2.6	<code>motion_try_rotate()</code>	45
4.17	<code>src/engine/piece/piece.c</code> File Reference	46
4.17.1	Detailed Description	46
4.17.2	Function Documentation	46
4.17.2.1	<code>piece_copy()</code>	47
4.17.2.2	<code>piece_create()</code>	47
4.17.2.3	<code>piece_free()</code>	47
4.17.2.4	<code>piece_random()</code>	47
4.18	<code>src/engine/piece/piece.h</code> File Reference	47
4.18.1	Detailed Description	49
4.18.2	Function Documentation	49
4.18.2.1	<code>piece_copy()</code>	49
4.18.2.2	<code>piece_create()</code>	49
4.18.2.3	<code>piece_free()</code>	49
4.18.2.4	<code>piece_random()</code>	49
4.19	<code>src/engine/piece/piece_queue.c</code> File Reference	50
4.19.1	Detailed Description	50
4.19.2	Function Documentation	51
4.19.2.1	<code>piece_queue_create()</code>	51
4.19.2.2	<code>piece_queue_extend()</code>	51
4.19.2.3	<code>piece_queue_fill_data()</code>	51

4.19.2.4	piece_queue_free()	51
4.19.2.5	piece_queue_get()	51
4.20	src/engine/piece/piece_queue.h File Reference	52
4.20.1	Detailed Description	53
4.20.2	Macro Definition Documentation	53
4.20.2.1	PIECE_QUEUE_LENGTH	53
4.20.3	Function Documentation	53
4.20.3.1	piece_queue_create()	53
4.20.3.2	piece_queue_extend()	54
4.20.3.3	piece_queue_fill_data()	54
4.20.3.4	piece_queue_free()	54
4.20.3.5	piece_queue_get()	54
4.21	src/engine/piece/piece_shape.h File Reference	54
4.21.1	Detailed Description	55
4.21.2	Macro Definition Documentation	55
4.21.2.1	PIECE_SHAPE_HEIGHT	56
4.21.2.2	PIECE_SHAPE_WIDTH	56
4.22	src/engine/piece/piece_type.h File Reference	56
4.22.1	Detailed Description	57
4.22.2	Macro Definition Documentation	57
4.22.2.1	PIECE_TYPE_ESIZE	57
4.22.3	Enumeration Type Documentation	57
4.22.3.1	PieceType	57
4.23	src/engine/piece/seven_bag.c File Reference	58
4.23.1	Detailed Description	58
4.23.2	Function Documentation	59
4.23.2.1	seven_bag_draw()	59
4.23.2.2	seven_bag_init()	59
4.23.2.3	seven_bag_shuffle()	59
4.23.2.4	seven_bag_swap()	59

4.24	src/engine/piece/seven_bag.h File Reference	59
4.24.1	Detailed Description	60
4.24.2	Function Documentation	60
4.24.2.1	seven_bag_draw()	61
4.24.2.2	seven_bag_init()	61
4.24.2.3	seven_bag_shuffle()	61
4.24.2.4	seven_bag_swap()	61
4.25	src/engine/score.c File Reference	61
4.25.1	Detailed Description	62
4.25.2	Function Documentation	62
4.25.2.1	score_compute_break()	62
4.26	src/engine/score.h File Reference	62
4.26.1	Detailed Description	63
4.26.2	Macro Definition Documentation	63
4.26.2.1	SCORE_DOUBLE	64
4.26.2.2	SCORE_HDROP	64
4.26.2.3	SCORE_SDROP	64
4.26.2.4	SCORE_SINGLE	64
4.26.2.5	SCORE_TETRIS	64
4.26.2.6	SCORE_TRIPLE	64
4.26.3	Function Documentation	64
4.26.3.1	score_compute_break()	64
4.27	src/engine/state.c File Reference	65
4.27.1	Detailed Description	65
4.27.2	Function Documentation	66
4.27.2.1	state_apply_input()	66
4.27.2.2	state_apply_inputs()	66
4.27.2.3	state_can_apply_input()	66
4.27.2.4	state_can_apply_inputs()	66
4.27.2.5	state_copy()	66

4.27.2.6	state_create()	67
4.27.2.7	state_create_piece()	67
4.27.2.8	state_free()	67
4.27.2.9	state_init()	67
4.27.2.10	state_next_piece()	67
4.27.2.11	state_step()	67
4.28	src/engine/state.h File Reference	68
4.28.1	Detailed Description	69
4.28.2	Function Documentation	69
4.28.2.1	state_apply_input()	69
4.28.2.2	state_apply_inputs()	69
4.28.2.3	state_can_apply_input()	70
4.28.2.4	state_can_apply_inputs()	70
4.28.2.5	state_copy()	70
4.28.2.6	state_create()	70
4.28.2.7	state_create_piece()	70
4.28.2.8	state_free()	70
4.28.2.9	state_init()	71
4.28.2.10	state_next_piece()	71
4.28.2.11	state_step()	71
4.29	src/tAltris.c File Reference	71
4.29.1	Detailed Description	72
4.29.2	Function Documentation	72
4.29.2.1	main()	72
4.30	src/utils/ansi_code.h File Reference	72
4.30.1	Detailed Description	74
4.30.2	Macro Definition Documentation	74
4.30.2.1	ANSI_BG_BBLACK	74
4.30.2.2	ANSI_BG_BBLUE	74
4.30.2.3	ANSI_BG_BCYAN	74

4.30.2.4	ANSI_BG_BGREEN	74
4.30.2.5	ANSI_BG_BLACK	74
4.30.2.6	ANSI_BG_BLUE	75
4.30.2.7	ANSI_BG_BMAGENTA	75
4.30.2.8	ANSI_BG_BRED	75
4.30.2.9	ANSI_BG_BWHITE	75
4.30.2.10	ANSI_BG_BYELLOW	75
4.30.2.11	ANSI_BG_CYAN	75
4.30.2.12	ANSI_BG_DEFAULT	75
4.30.2.13	ANSI_BG_GREEN	75
4.30.2.14	ANSI_BG_MAGENTA	76
4.30.2.15	ANSI_BG_RED	76
4.30.2.16	ANSI_BG_WHITE	76
4.30.2.17	ANSI_BG_YELLOW	76
4.30.2.18	ANSI_BOLD	76
4.30.2.19	ANSI_CROSSEDOUT	76
4.30.2.20	ANSI_ENCIRCLED	76
4.30.2.21	ANSI_ESC	76
4.30.2.22	ANSI_FAINT	77
4.30.2.23	ANSI_FG_BBLACK	77
4.30.2.24	ANSI_FG_BBLUE	77
4.30.2.25	ANSI_FG_BCYAN	77
4.30.2.26	ANSI_FG_BGREEN	77
4.30.2.27	ANSI_FG_BLACK	77
4.30.2.28	ANSI_FG_BLUE	77
4.30.2.29	ANSI_FG_BMAGENTA	77
4.30.2.30	ANSI_FG_BRED	78
4.30.2.31	ANSI_FG_BWHITE	78
4.30.2.32	ANSI_FG_BYELLOW	78
4.30.2.33	ANSI_FG_CYAN	78

4.30.2.34 ANSI_FG_DEFAULT	78
4.30.2.35 ANSI_FG_GREEN	78
4.30.2.36 ANSI_FG_MAGENTA	78
4.30.2.37 ANSI_FG_RED	78
4.30.2.38 ANSI_FG_WHITE	79
4.30.2.39 ANSI_FG_YELLOW	79
4.30.2.40 ANSI_FRAMED	79
4.30.2.41 ANSI_ITALIC	79
4.30.2.42 ANSI_OVERLINED	79
4.30.2.43 ANSI_RBLINK	79
4.30.2.44 ANSI_RESET	79
4.30.2.45 ANSI_SBLINK	79
4.30.2.46 ANSI_SGR	80
4.30.2.47 ANSI_UNDERLINE	80
4.31 src/utls/random.h File Reference	80
4.31.1 Detailed Description	81
4.32 src/utls/safe_op.h File Reference	81
4.32.1 Detailed Description	82
4.32.2 Macro Definition Documentation	83
4.32.2.1 SAFE_OP_OVERFLOW	83
4.32.2.2 SAFE_OP_SUCCESS	83
4.32.2.3 SAFE_OP_UNDERFLOW	83
Index	85

Chapter 1

Data Structure Index

1.1 Data Structures

Here are the data structures with brief descriptions:

AI Coefs	5
Board	6
Piece	7
PieceQueue	8
PieceShape	8
State	9

Chapter 2

File Index

2.1 File List

Here is a list of all files with brief descriptions:

src/ tAltris.c	
Main file	71
src/ai/genetic/ core.c	
Core of the genetic algorithm	13
src/ai/genetic/ core.h	
Core of the genetic algorithm	14
src/ai/genetic/ engine.c	
Engine for the genetic algorithm	16
src/ai/genetic/ engine.h	
Engine for the genetic algorithm	17
src/ai/genetic/ tools.c	
Tools for the genetic algorithm	19
src/ai/genetic/ tools.h	
Tools for the genetic algorithm	21
src/debug/ debug.h	
Debug	25
src/debug/engine/ debug_state.c	
Debug state	26
src/debug/engine/ debug_state.h	
Debug state	28
src/engine/ angle.h	
Angle	31
src/engine/ board.c	
Board (p. 6)	32
src/engine/ board.h	
Board (p. 6)	35
src/engine/ cell.h	
Cell	38
src/engine/ input.h	
Input	40
src/engine/ motion.c	
Motion	41
src/engine/ motion.h	
Motion	43
src/engine/ score.c	
Scoring system	61

src/engine/ score.h	
Scoring system	62
src/engine/ state.c	
State (p. 9)	65
src/engine/ state.h	
State (p. 9)	68
src/engine/piece/ piece.c	
Piece (p. 7)	46
src/engine/piece/ piece.h	
Piece (p. 7)	47
src/engine/piece/ piece_queue.c	
Piece (p. 7) queue	50
src/engine/piece/ piece_queue.h	
Piece (p. 7) queue	52
src/engine/piece/ piece_shape.h	
Piece (p. 7) shape	54
src/engine/piece/ piece_type.h	
Piece (p. 7) type	56
src/engine/piece/ seven_bag.c	
7-Bag generator	58
src/engine/piece/ seven_bag.h	
7-Bag generator	59
src/utills/ ansi_code.h	
ANSI escape code	72
src/utills/ random.h	
Random number generation	80
src/utills/ safe_op.h	
Safe operations	81

Chapter 3

Data Structure Documentation

3.1 AiCoefs Struct Reference

```
#include <engine.h>
```

Data Fields

- double **agg_height**
- double **holes**
- double **clears**
- double **bumpiness**

3.1.1 Field Documentation

3.1.1.1 agg_height

```
double agg_height
```

3.1.1.2 bumpiness

```
double bumpiness
```

3.1.1.3 clears

```
double clears
```

3.1.1.4 holes

```
double holes
```

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

- `src/ai/genetic/ engine.h`

3.2 Board Struct Reference

```
#include <board.h>
```

Data Fields

- int **width**
- int **height**
- **Cell** * **cells**

3.2.1 Field Documentation

3.2.1.1 cells

```
Cell* cells
```

3.2.1.2 height

```
int height
```

3.2.1.3 width

```
int width
```

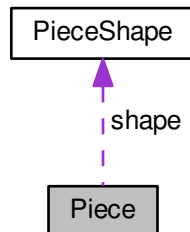
The documentation for this struct was generated from the following file:

- `src/engine/ board.h`

3.3 Piece Struct Reference

```
#include <piece.h>
```

Collaboration diagram for Piece:



Data Fields

- **PieceType** type
- const **PieceShape** * shape
- int x
- int y
- **Angle** angle

3.3.1 Field Documentation

3.3.1.1 angle

Angle angle

3.3.1.2 shape

const **PieceShape*** shape

3.3.1.3 type

PieceType type

3.3.1.4 x

```
int x
```

3.3.1.5 y

```
int y
```

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

- src/engine/piece/ **piece.h**

3.4 PieceQueue Struct Reference

```
#include <piece_queue.h>
```

Data Fields

- `size_t` **length**
- **PieceType** * **data**

3.4.1 Field Documentation

3.4.1.1 data

```
PieceType* data
```

3.4.1.2 length

```
size_t length
```

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

- src/engine/piece/ **piece_queue.h**

3.5 PieceShape Struct Reference

```
#include <piece_shape.h>
```

Data Fields

- int **shape** [**ANGLE_ESIZE**][**PIECE_SHAPE_HEIGHT**][**PIECE_SHAPE_WIDTH**]
- Cell **fill**

3.5.1 Field Documentation

3.5.1.1 fill

Cell **fill**

3.5.1.2 shape

```
int shape[ ANGLE_ESIZE][ PIECE_SHAPE_HEIGHT][ PIECE_SHAPE_WIDTH]
```

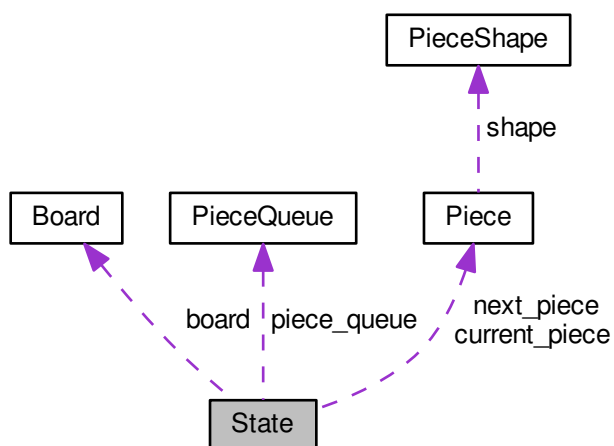
The documentation for this struct was generated from the following file:

- src/engine/piece/ **piece_shape.h**

3.6 State Struct Reference

```
#include <state.h>
```

Collaboration diagram for State:



Data Fields

- unsigned int **score**
- unsigned int **level**
- unsigned int **broken_lines**
- unsigned int **step**
- unsigned int **input_counts**
- **Board** * **board**
- **PieceQueue** * **piece_queue**
- size_t **piece_queue_index**
- **Piece** * **current_piece**
- **Piece** * **next_piece**

3.6.1 Field Documentation

3.6.1.1 board

Board* board

3.6.1.2 broken_lines

unsigned int broken_lines

3.6.1.3 current_piece

Piece* current_piece

3.6.1.4 input_counts

unsigned int input_counts

3.6.1.5 level

unsigned int level

3.6.1.6 next_piece

Piece* next_piece

3.6.1.7 piece_queue

PieceQueue* piece_queue

3.6.1.8 piece_queue_index

size_t piece_queue_index

3.6.1.9 score

unsigned int score

3.6.1.10 step

unsigned int step

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

- src/engine/ **state.h**

Chapter 4

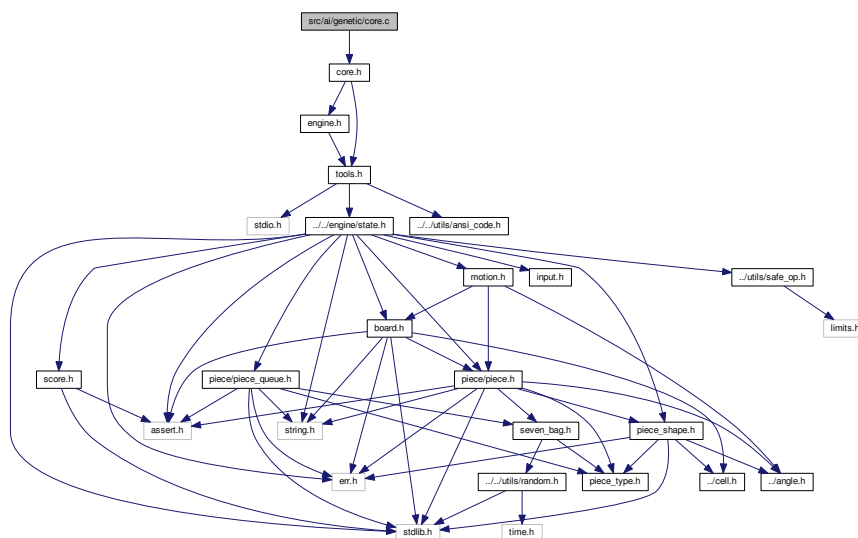
File Documentation

4.1 src/ai/genetic/core.c File Reference

Core of the genetic algorithm.

```
#include "core.h"
```

Include dependency graph for core.c:



Functions

- void **genetic_show_stats** (**State** *state)

4.1.1 Detailed Description

Core of the genetic algorithm.

Author

S4MasterRace

Version

2.0

4.1.2 Function Documentation

4.1.2.1 genetic_show_stats()

```
void genetic_show_stats (
    State * state )
```

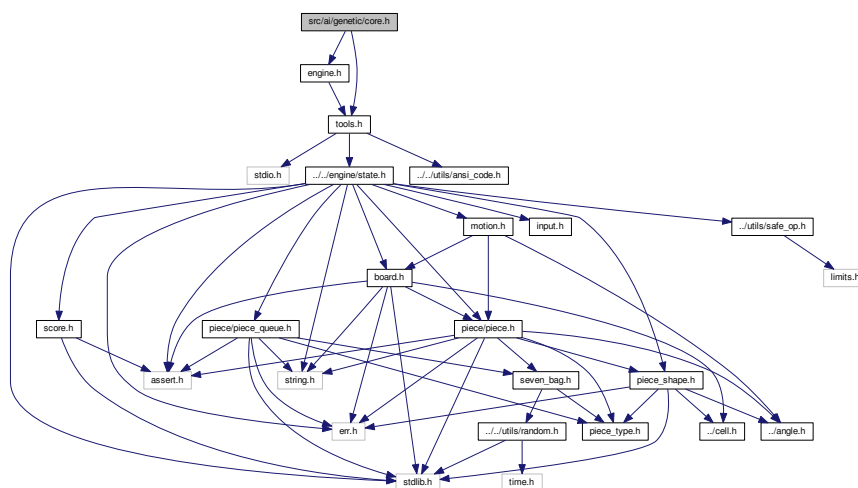
4.2 src/ai/genetic/core.h File Reference

Core of the genetic algorithm.

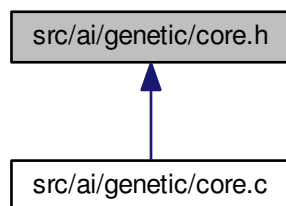
```
#include "engine.h"
```

```
#include "tools.h"
```

Include dependency graph for core.h:



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



Functions

- void **genetic_show_stats** (**State** *state)

4.2.1 Detailed Description

Core of the genetic algorithm.

Author

S4MasterRace

Version

2.0

4.2.2 Function Documentation

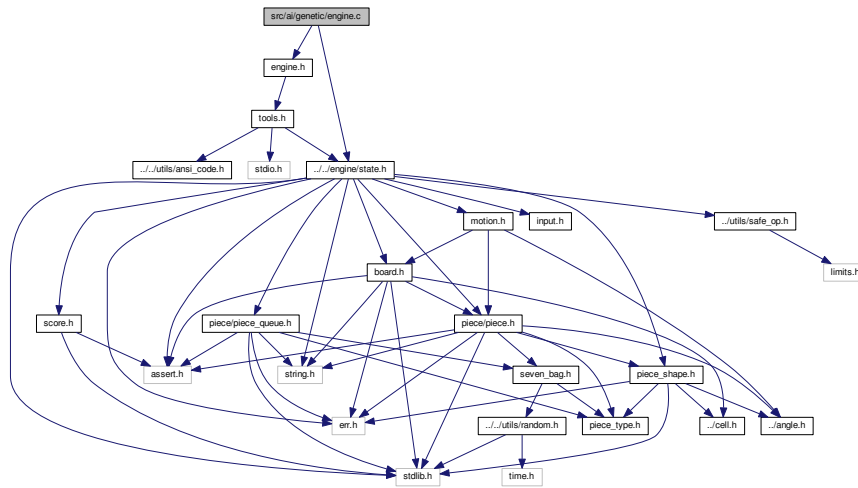
4.2.2.1 genetic_show_stats()

```
void genetic_show_stats (
    State * state )
```

4.3 src/ai/genetic/engine.c File Reference

Engine for the genetic algorithm.

```
#include "engine.h"
#include "../../engine/state.h"
Include dependency graph for engine.c:
```



Functions

- **AiCoefs** * **get_coefs** ()
- double **get_rank** (**State** *state)

4.3.1 Detailed Description

Engine for the genetic algorithm.

Author

S4MasterRace

Version

2.0

4.3.2 Function Documentation

4.3.2.1 get_coefs()

```
AiCoefs* get_coefs ( )
```

4.3.2.2 get_rank()

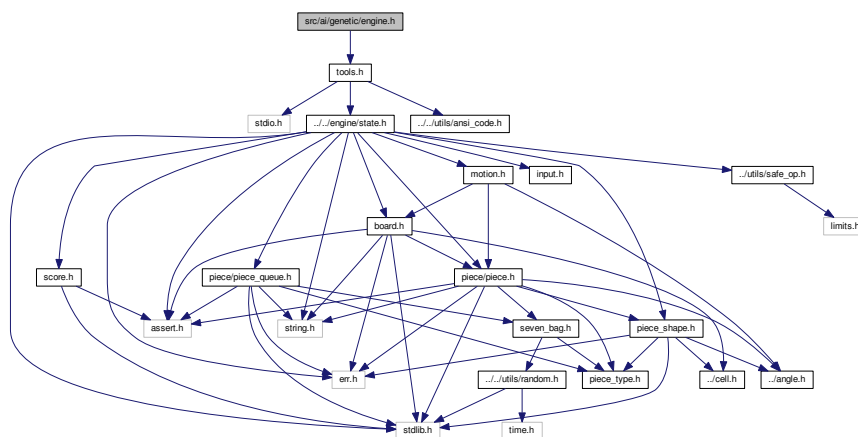
```
double get_rank (
    State * state )
```

4.4 src/ai/genetic/engine.h File Reference

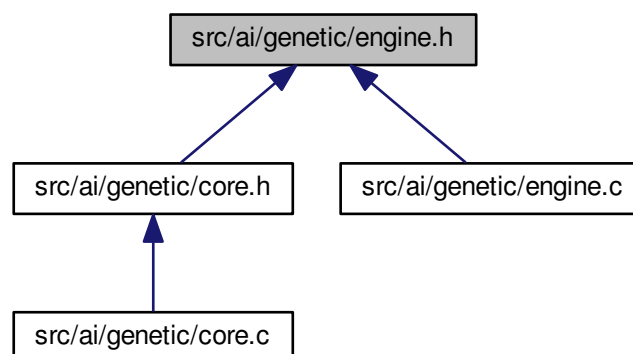
Engine for the genetic algorithm.

```
#include "tools.h"
```

Include dependency graph for engine.h:



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



Data Structures

- struct **AiCoefs**

Functions

- **AiCoefs** * **get_coefs** ()
- double **get_rank** (**State** *state)

4.4.1 Detailed Description

Engine for the genetic algorithm.

Author

S4MasterRace

Version

2.0

4.4.2 Function Documentation

4.4.2.1 **get_coefs**()

```
AiCoefs* get_coefs ( )
```

4.4.2.2 **get_rank**()

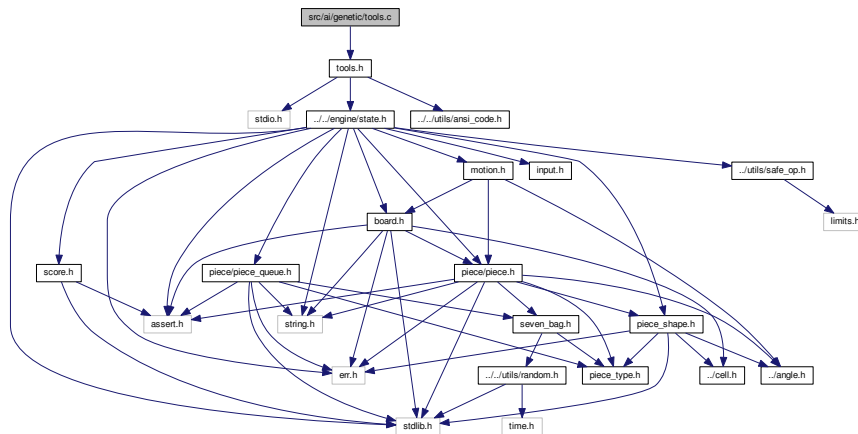
```
double get_rank (
    State * state )
```


4.5 src/ai/genetic/tools.c File Reference

Tools for the genetic algorithm.

```
#include "tools.h"
```

Include dependency graph for tools.c:



Functions

- void **board_heights** (const **Board** *brd, int *heights)
- int **board_height** (const **Board** *brd, int x)
- int **bumpiness** (const **Board** *brd)
- int **aggregate_height** (const **Board** *brd)
- int **hole** (const **Board** *brd, int x)
- int **holes** (const **Board** *brd)
- int **clears** (const **Board** *brd)
- void **show_features** (const **Board** *brd)

4.5.1 Detailed Description

Tools for the genetic algorithm.

Author

S4MasterRace

Version

2.0

4.5.2 Function Documentation

4.5.2.1 aggregate_height()

```
int aggregate_height (
    const Board * brd )
```

4.5.2.2 board_height()

```
int board_height (
    const Board * brd,
    int x )
```

4.5.2.3 board_heights()

```
void board_heights (
    const Board * brd,
    int * heights )
```

4.5.2.4 bumpiness()

```
int bumpiness (
    const Board * brd )
```

4.5.2.5 clears()

```
int clears (
    const Board * brd )
```

4.5.2.6 hole()

```
int hole (
    const Board * brd,
    int x )
```

4.5.2.7 holes()

```
int holes (
    const Board * brd )
```

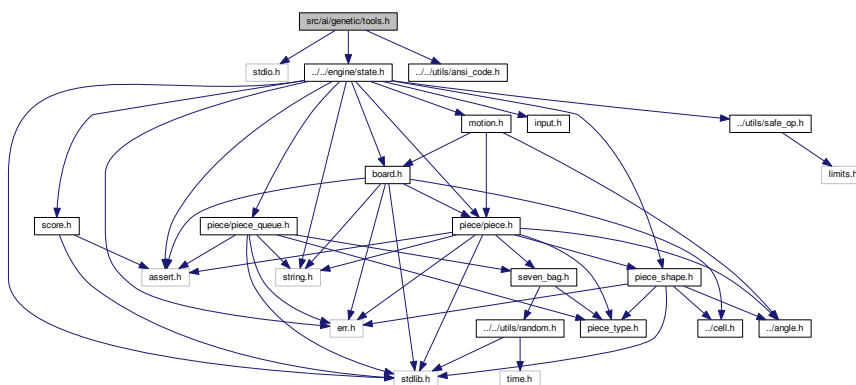
4.5.2.8 show_features()

```
void show_features (
    const Board * brd )
```

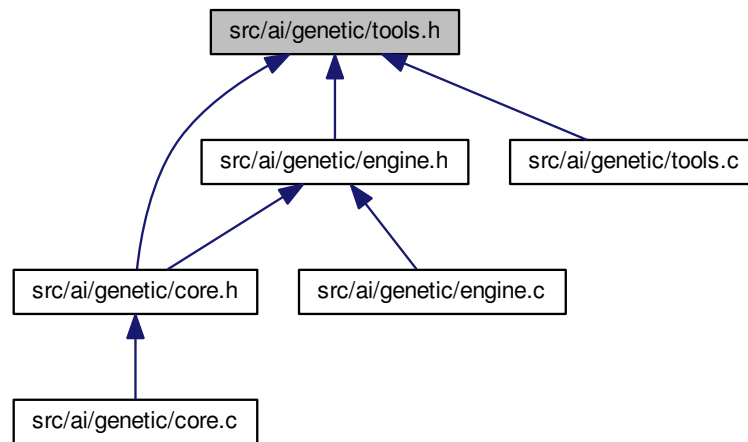
4.6 src/ai/genetic/tools.h File Reference

Tools for the genetic algorithm.

```
#include <stdio.h>
#include "../engine/state.h"
#include "../utils/ansi_code.h"
Include dependency graph for tools.h:
```



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



Macros

- `#define ABS(X) (((X) < 0) ? (-1 * (X)) : (X))`

Functions

- `int board_height (const Board *brd, int x)`
- `void board_heights (const Board *brd, int *heights)`
- `int bumpiness (const Board *brd)`
- `int aggregate_height (const Board *brd)`
- `int hole (const Board *brd, int x)`
- `int holes (const Board *brd)`
- `size_t coalescent_clears (const Board *brd)`
- `int clears (const Board *brd)`
- `void show_features (const Board *brd)`

4.6.1 Detailed Description

Tools for the genetic algorithm.

Author

S4MasterRace

Version

2.0

4.6.2 Macro Definition Documentation

4.6.2.1 ABS

```
#define ABS(  
    X ) ((X) < 0) ?  (-1 * (X)) : (X)
```

4.6.3 Function Documentation

4.6.3.1 aggregate_height()

```
int aggregate_height (  
    const Board * brd )
```

4.6.3.2 board_height()

```
int board_height (  
    const Board * brd,  
    int x )
```

4.6.3.3 board_heights()

```
void board_heights (  
    const Board * brd,  
    int * heights )
```

4.6.3.4 bumpiness()

```
int bumpiness (  
    const Board * brd )
```

4.6.3.5 clears()

```
int clears (
    const Board * brd )
```

4.6.3.6 coalescent_clears()

```
size_t coalescent_clears (
    const Board * brd )
```

4.6.3.7 hole()

```
int hole (
    const Board * brd,
    int x )
```

4.6.3.8 holes()

```
int holes (
    const Board * brd )
```

4.6.3.9 show_features()

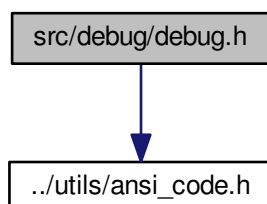
```
void show_features (
    const Board * brd )
```

4.7 src/debug/debug.h File Reference

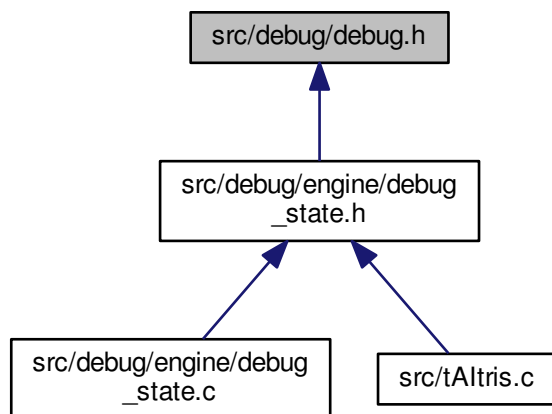
Debug.

```
#include "../utils/ansi_code.h"
```

Include dependency graph for debug.h:



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



Macros

- `#define DEBUG_TAG(_name_, _color_)`

4.7.1 Detailed Description

Debug.

Functions

- void **debug_state_print_line_number** (const **Board** *brd, int y)
- void **debug_state_print_cell** (**Cell** c)
- void **debug_state_print_infos** (const **State** *state, int y)
- void **debug_state_print_next_piece** (const **Piece** *pc, int y)
- void **debug_state_print** (const **State** *state)

4.8.1 Detailed Description

Debug state.

Author

S4MasterRace

Version

2.0

4.8.2 Function Documentation

4.8.2.1 debug_state_print()

```
void debug_state_print (
    const State * state )
```

4.8.2.2 debug_state_print_cell()

```
void debug_state_print_cell (
    Cell c )
```

4.8.2.3 debug_state_print_infos()

```
void debug_state_print_infos (
    const State * state,
    int y )
```

4.8.2.4 debug_state_print_line_number()

```
void debug_state_print_line_number (
    const Board * brd,
    int y )
```

4.8.2.5 debug_state_print_next_piece()

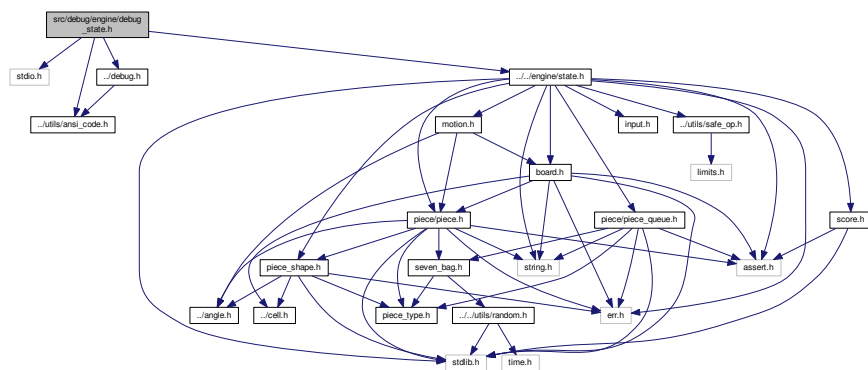
```
void debug_state_print_next_piece (
    const Piece * pc,
    int y )
```

4.9 src/debug/engine/debug_state.h File Reference

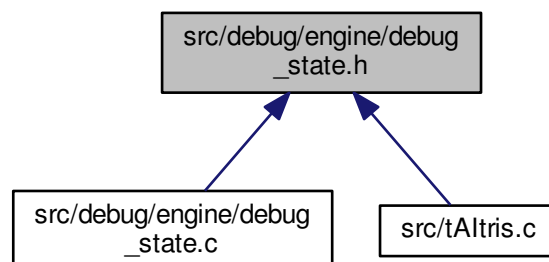
Debug state.

```
#include <stdio.h>
#include "../debug.h"
#include "../../engine/state.h"
#include "../../utils/ansi_code.h"
```

Include dependency graph for debug_state.h:



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



Macros

- `#define DEBUG_STATE_NAME "State"`
- `#define DEBUG_STATE_COLOR ANSI_FG_MAGENTA`
- `#define DEBUG_STATE_TAG DEBUG_TAG(DEBUG_STATE_NAME, DEBUG_STATE_COLOR)`

Functions

- void `debug_state_print_line_number` (const **Board** *brd, int y)
- void `debug_state_print_cell` (**Cell** c)
- void `debug_state_print_infos` (const **State** *state, int y)
- void `debug_state_print_next_piece` (const **Piece** *pc, int y)
- void `debug_state_print` (const **State** *state)

4.9.1 Detailed Description

Debug state.

Author

S4MasterRace

Version

2.0

4.9.2 Macro Definition Documentation

4.9.2.1 DEBUG_STATE_COLOR

```
#define DEBUG_STATE_COLOR ANSI_FG_MAGENTA
```

4.9.2.2 DEBUG_STATE_NAME

```
#define DEBUG_STATE_NAME "State"
```

4.9.2.3 DEBUG_STATE_TAG

```
#define DEBUG_STATE_TAG DEBUG_TAG( DEBUG_STATE_NAME, DEBUG_STATE_COLOR)
```

4.9.3 Function Documentation

4.9.3.1 debug_state_print()

```
void debug_state_print (
    const State * state )
```

4.9.3.2 debug_state_print_cell()

```
void debug_state_print_cell (
    Cell c )
```

4.9.3.3 debug_state_print_infos()

```
void debug_state_print_infos (
    const State * state,
    int y )
```

4.9.3.4 debug_state_print_line_number()

```
void debug_state_print_line_number (
    const Board * brd,
    int y )
```

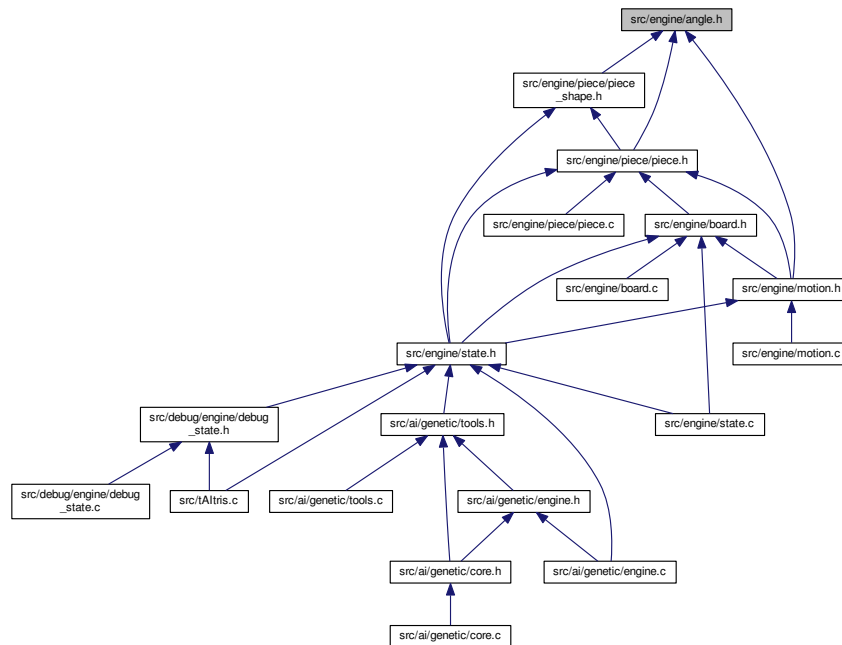
4.9.3.5 debug_state_print_next_piece()

```
void debug_state_print_next_piece (
    const Piece * pc,
    int y )
```

4.10 src/engine/angle.h File Reference

Angle.

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



Macros

- `#define` **ANGLE_ESIZE** 4

Enumerations

- `enum` **Angle** { **ANGLE_UP**, **ANGLE_RIGHT**, **ANGLE_DOWN**, **ANGLE_LEFT** }
- `enum` **Rotation** { **ROTATE_LEFT** = -1, **ROTATE_RIGHT** = 1 }

4.10.1 Detailed Description

Angle.

Author

S4MasterRace

Version

2.0

4.10.2 Macro Definition Documentation

4.10.2.1 ANGLE_ESIZE

```
#define ANGLE_ESIZE 4
```

4.10.3 Enumeration Type Documentation

4.10.3.1 Angle

```
enum Angle
```

Enumerator

ANGLE_UP	
ANGLE_RIGHT	
ANGLE_DOWN	
ANGLE_LEFT	

4.10.3.2 Rotation

```
enum Rotation
```

Enumerator

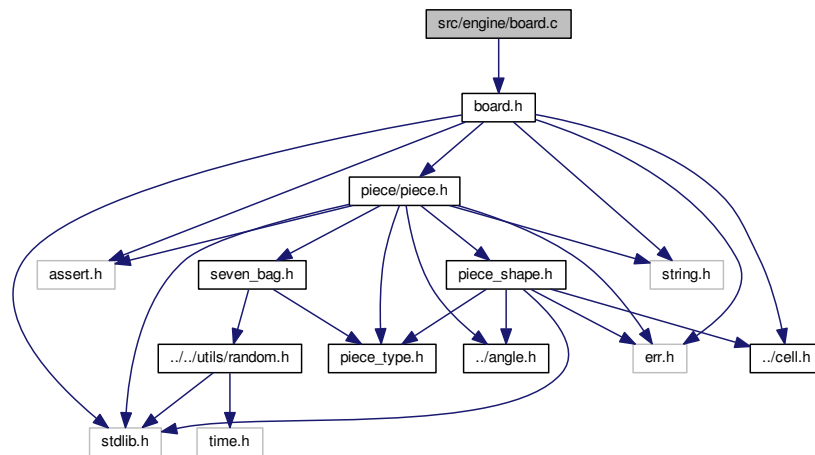
ROTATE_LEFT	
ROTATE_RIGHT	

4.11 src/engine/board.c File Reference

Board (p. 6).

```
#include "board.h"
```

Include dependency graph for board.c:



Functions

- **Board *** **board_create** (int width, int height)
- void **board_init** (**Board** *brd)
- void **board_free** (**Board** *brd)
- **Board *** **board_copy** (**Board** *brd)
- size_t **board_get_completed_lines** (const **Board** *brd, int *hist)
- void **board_break_lines** (**Board** *brd, const int *hist)
- int **board_merge_piece** (**Board** *brd, const **Piece** *pc)

4.11.1 Detailed Description

Board (p. 6).

Author

S4MasterRace

Version

2.0

4.11.2 Function Documentation

4.11.2.1 board_break_lines()

```
void board_break_lines (
    Board * brd,
    const int * hist )
```

4.11.2.2 board_copy()

```
Board* board_copy (
    Board * brd )
```

4.11.2.3 board_create()

```
Board* board_create (
    int width,
    int height )
```

4.11.2.4 board_free()

```
void board_free (
    Board * brd )
```

4.11.2.5 board_get_completed_lines()

```
size_t board_get_completed_lines (
    const Board * brd,
    int * hist )
```

4.11.2.6 board_init()

```
void board_init (
    Board * brd )
```


4.11.2.7 board_merge_piece()

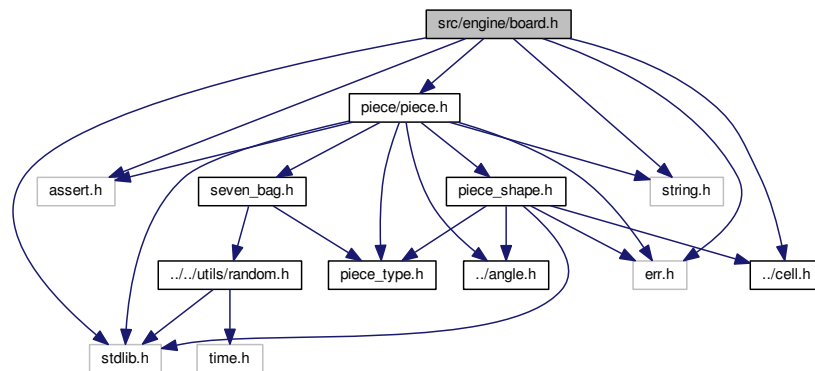
```
int board_merge_piece (
    Board * brd,
    const Piece * pc )
```

4.12 src/engine/board.h File Reference

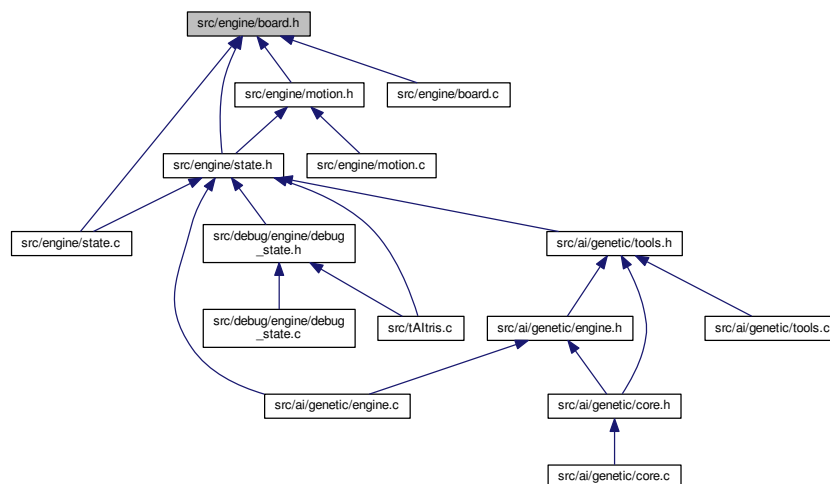
Board (p. 6).

```
#include <stdlib.h>
#include <assert.h>
#include <string.h>
#include <err.h>
#include "piece/piece.h"
#include "cell.h"
```

Include dependency graph for board.h:



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



Data Structures

- struct **Board**

Macros

- `#define BOARD_WIDTH 10`
- `#define BOARD_HEIGHT 20`
- `#define BOARD_HIDDEN 2`
- `#define board_reverse_y(_brd_, _y_) ((_brd_)->height - 1 - (_y_))`

Functions

- **Board** * **board_create** (int width, int height)
- void **board_init** (**Board** *brd)
- void **board_free** (**Board** *brd)
- **Board** * **board_copy** (**Board** *brd)
- size_t **board_get_completed_lines** (const **Board** *brd, int *hist)
- void **board_break_lines** (**Board** *brd, const int *hist)
- int **board_merge_piece** (**Board** *brd, const **Piece** *pc)

4.12.1 Detailed Description

Board (p. 6).

Author

S4MasterRace

Version

2.0

4.12.2 Macro Definition Documentation

4.12.2.1 BOARD_HEIGHT

```
#define BOARD_HEIGHT 20
```

4.12.2.2 BOARD_HIDDEN

```
#define BOARD_HIDDEN 2
```

4.12.2.3 board_reverse_y

```
#define board_reverse_y(  
    _brd_,  
    _y_ ) ((_brd_)->height - 1 - (_y_))
```

4.12.2.4 BOARD_WIDTH

```
#define BOARD_WIDTH 10
```

4.12.3 Function Documentation

4.12.3.1 board_break_lines()

```
void board_break_lines (  
    Board * brd,  
    const int * hist )
```

4.12.3.2 board_copy()

```
Board* board_copy (  
    Board * brd )
```

4.12.3.3 board_create()

```
Board* board_create (  
    int width,  
    int height )
```

4.12.3.4 board_free()

```
void board_free (  
    Board * brd )
```

4.12.3.5 board_get_completed_lines()

```
size_t board_get_completed_lines (
    const Board * brd,
    int * hist )
```

4.12.3.6 board_init()

```
void board_init (
    Board * brd )
```

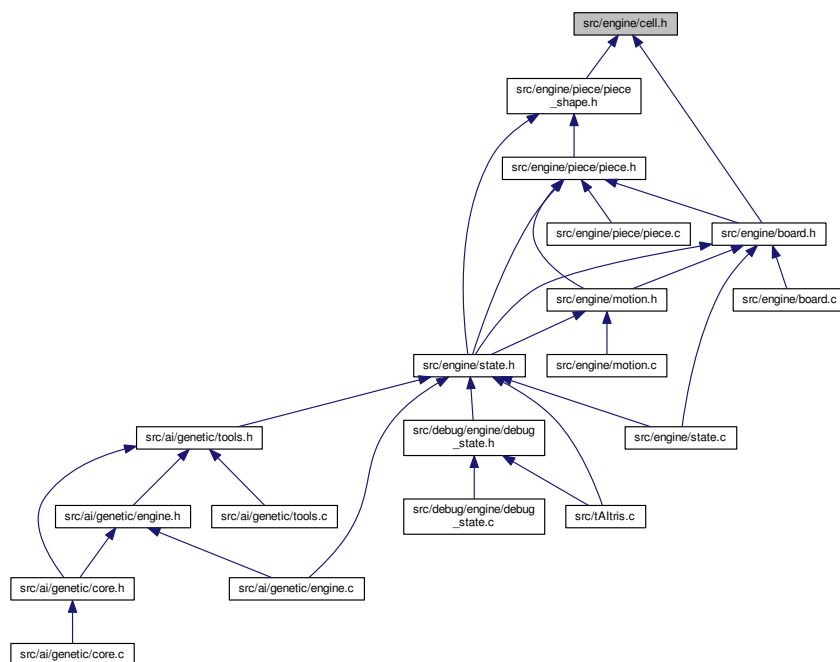
4.12.3.7 board_merge_piece()

```
int board_merge_piece (
    Board * brd,
    const Piece * pc )
```

4.13 src/engine/cell.h File Reference

Cell.

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



Macros

- `#define CELL_ESIZE 8`

Enumerations

- `enum Cell {
CELL_EMPTY, CELL_CYAN, CELL_YELLOW, CELL_PURPLE,
CELL_GREEN, CELL_RED, CELL_BLUE, CELL_ORANGE }`

4.13.1 Detailed Description

Cell.

Author

S4MasterRace

Version

2.0

4.13.2 Macro Definition Documentation

4.13.2.1 CELL_ESIZE

```
#define CELL_ESIZE 8
```

4.13.3 Enumeration Type Documentation

4.13.3.1 Cell

```
enum Cell
```

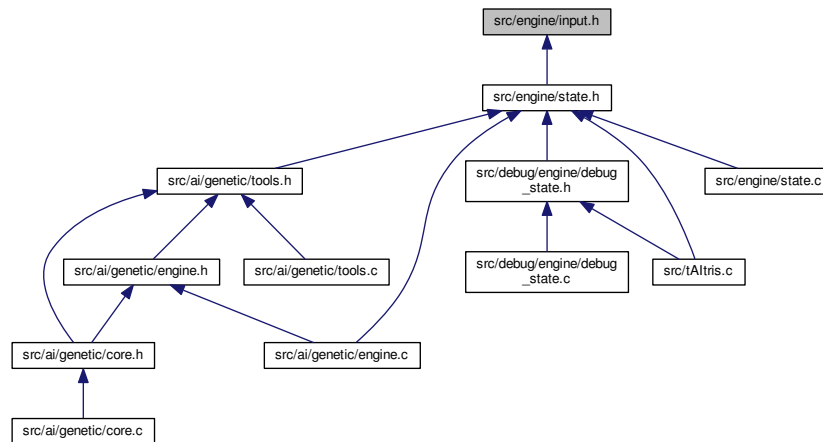
Enumerator

CELL_EMPTY	
CELL_CYAN	
CELL_YELLOW	
CELL_PURPLE	
CELL_GREEN	
CELL_RED	
CELL_BLUE	
CELL_ORANGE	

4.14 src/engine/input.h File Reference

Input.

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



Macros

- `#define INPUT_ESIZE 6`

Enumerations

- `enum Input {
 INPUT_MOVE_LEFT, INPUT_MOVE_RIGHT, INPUT_ROTATE_RIGHT, INPUT_ROTATE_LEFT,
 INPUT_SOFT_DROP, INPUT_HARD_DROP }`

4.14.1 Detailed Description

Input.

Author

S4MasterRace

Version

2.0

4.14.2 Macro Definition Documentation

4.14.2.1 INPUT_ESIZE

```
#define INPUT_ESIZE 6
```

4.14.3 Enumeration Type Documentation

4.14.3.1 Input

```
enum Input
```

Enumerator

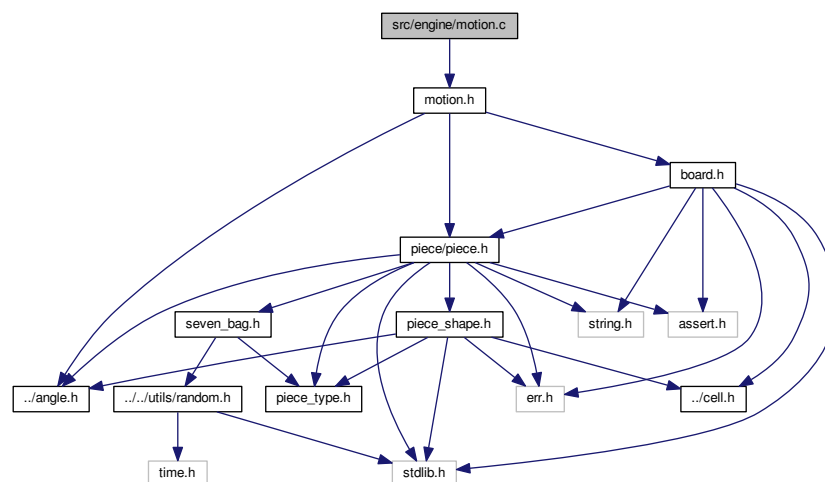
INPUT_MOVE_LEFT	
INPUT_MOVE_RIGHT	
INPUT_ROTATE_RIGHT	
INPUT_ROTATE_LEFT	
INPUT_SOFT_DROP	
INPUT_HARD_DROP	

4.15 src/engine/motion.c File Reference

Motion.

```
#include "motion.h"
```

Include dependency graph for motion.c:



Functions

- int **motion_is_valid** (const **Piece** *pc, const **Board** *brd)
- int **motion_try_move** (**Piece** *pc, const **Board** *brd, int dx, int dy)
- int **motion_try_down** (**Piece** *pc, const **Board** *brd)
- int **motion_try_rotate** (**Piece** *pc, const **Board** *brd, **Rotation** r)
- int **motion_can_move** (const **Piece** *pc, const **Board** *brd, int dx, int dy)
- int **motion_can_rotate** (const **Piece** *pc, const **Board** *brd, **Rotation** r)

4.15.1 Detailed Description

Motion.

Author

S4MasterRace

Version

2.0

4.15.2 Function Documentation

4.15.2.1 motion_can_move()

```
int motion_can_move (
    const Piece * pc,
    const Board * brd,
    int dx,
    int dy )
```

4.15.2.2 motion_can_rotate()

```
int motion_can_rotate (
    const Piece * pc,
    const Board * brd,
    Rotation r )
```

4.15.2.3 motion_is_valid()

```
int motion_is_valid (
    const Piece * pc,
    const Board * brd )
```


4.15.2.4 motion_try_down()

```
int motion_try_down (
    Piece * pc,
    const Board * brd )
```

4.15.2.5 motion_try_move()

```
int motion_try_move (
    Piece * pc,
    const Board * brd,
    int dx,
    int dy )
```

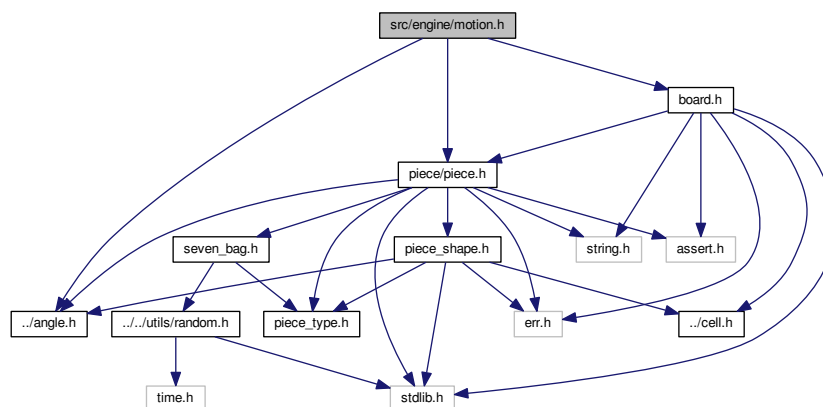
4.15.2.6 motion_try_rotate()

```
int motion_try_rotate (
    Piece * pc,
    const Board * brd,
    Rotation r )
```

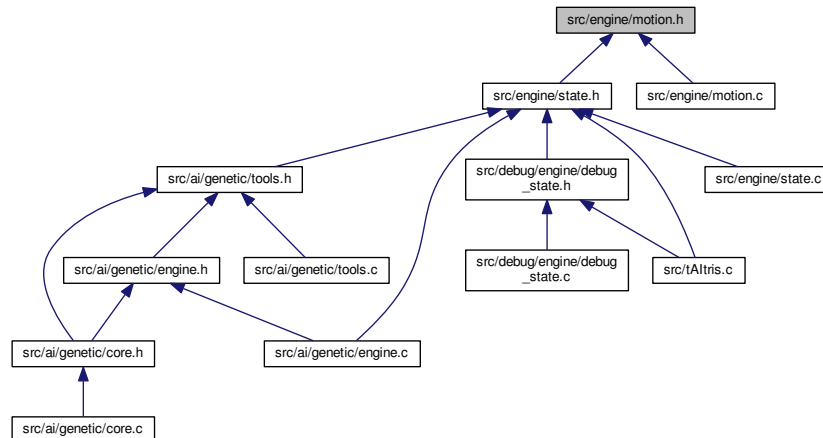
4.16 src/engine/motion.h File Reference

Motion.

```
#include "piece/piece.h"
#include "board.h"
#include "angle.h"
Include dependency graph for motion.h:
```



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



Functions

- int **motion_is_valid** (const **Piece** *pc, const **Board** *brd)
- int **motion_try_move** (**Piece** *pc, const **Board** *brd, int dx, int dy)
- int **motion_try_rotate** (**Piece** *pc, const **Board** *brd, **Rotation** r)
- int **motion_try_down** (**Piece** *pc, const **Board** *brd)
- int **motion_can_move** (const **Piece** *pc, const **Board** *brd, int dx, int dy)
- int **motion_can_rotate** (const **Piece** *pc, const **Board** *brd, **Rotation** r)

4.16.1 Detailed Description

Motion.

Author

S4MasterRace

Version

2.0

4.16.2 Function Documentation

4.16.2.1 motion_can_move()

```

int motion_can_move (
    const Piece * pc,
    const Board * brd,
    int dx,
    int dy )
  
```

4.16.2.2 motion_can_rotate()

```
int motion_can_rotate (
    const Piece * pc,
    const Board * brd,
    Rotation r )
```

4.16.2.3 motion_is_valid()

```
int motion_is_valid (
    const Piece * pc,
    const Board * brd )
```

4.16.2.4 motion_try_down()

```
int motion_try_down (
    Piece * pc,
    const Board * brd )
```

4.16.2.5 motion_try_move()

```
int motion_try_move (
    Piece * pc,
    const Board * brd,
    int dx,
    int dy )
```

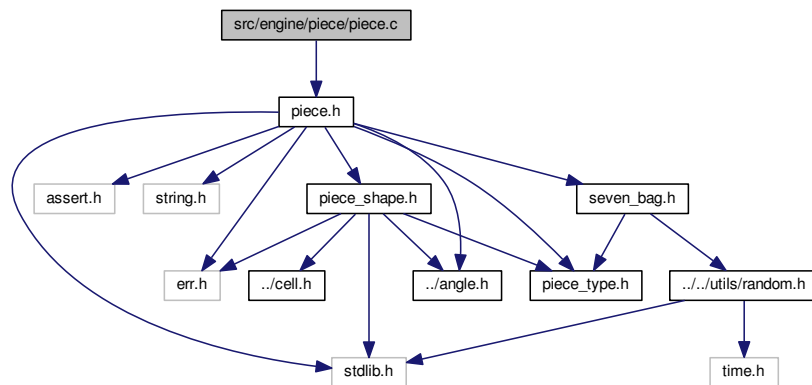
4.16.2.6 motion_try_rotate()

```
int motion_try_rotate (
    Piece * pc,
    const Board * brd,
    Rotation r )
```

4.17 src/engine/piece/piece.c File Reference

Piece (p. 7).

```
#include "piece.h"
Include dependency graph for piece.c:
```



Functions

- **Piece * piece_create** (**PieceType** type, int x, int y, **Angle** angle)
- void **piece_free** (**Piece** *pc)
- **Piece * piece_copy** (const **Piece** *pc)
- **Piece * piece_random** (int x, int y, **Angle** angle)

4.17.1 Detailed Description

Piece (p. 7).

Author

S4MasterRace

Version

2.0

4.17.2 Function Documentation

4.17.2.1 piece_copy()

```
Piece* piece_copy (
    const Piece * pc )
```

4.17.2.2 piece_create()

```
Piece* piece_create (
    PieceType type,
    int x,
    int y,
    Angle angle )
```

4.17.2.3 piece_free()

```
void piece_free (
    Piece * pc )
```

4.17.2.4 piece_random()

```
Piece* piece_random (
    int x,
    int y,
    Angle angle )
```

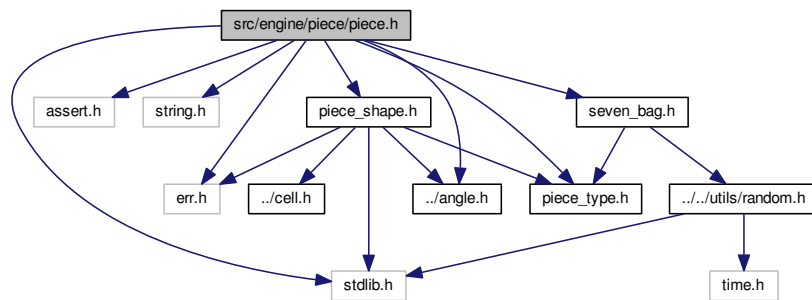
4.18 src/engine/piece/piece.h File Reference

Piece (p. 7).

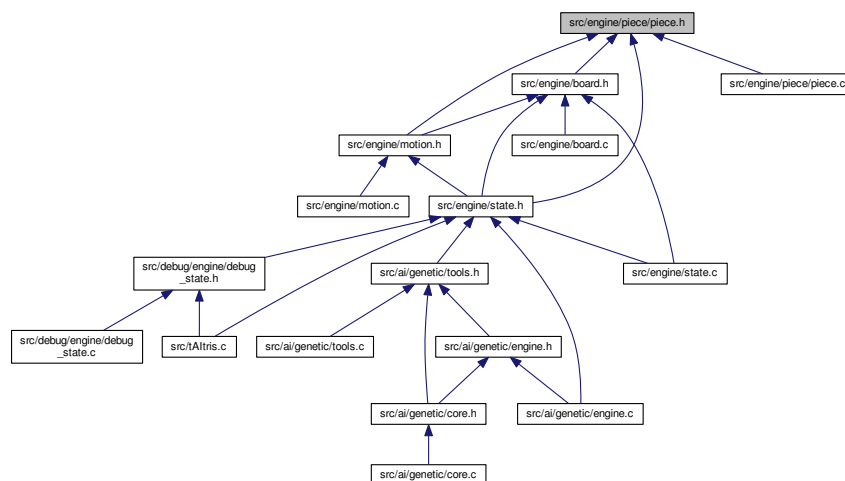
```
#include <stdlib.h>
#include <assert.h>
#include <string.h>
#include <err.h>
#include "piece_type.h"
#include "piece_shape.h"
#include "../angle.h"
```

```
#include "seven_bag.h"
```

Include dependency graph for piece.h:



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



Data Structures

- struct **Piece**

Functions

- **Piece *** **piece_create** (**PieceType** type, int x, int y, **Angle** angle)
- void **piece_free** (**Piece** *pc)
- **Piece *** **piece_copy** (const **Piece** *pc)
- **Piece *** **piece_random** (int x, int y, **Angle** angle)

4.18.1 Detailed Description

Piece (p. 7).

Author

S4MasterRace

Version

2.0

4.18.2 Function Documentation

4.18.2.1 piece_copy()

```
Piece* piece_copy (
    const Piece * pc )
```

4.18.2.2 piece_create()

```
Piece* piece_create (
    PieceType type,
    int x,
    int y,
    Angle angle )
```

4.18.2.3 piece_free()

```
void piece_free (
    Piece * pc )
```

4.18.2.4 piece_random()

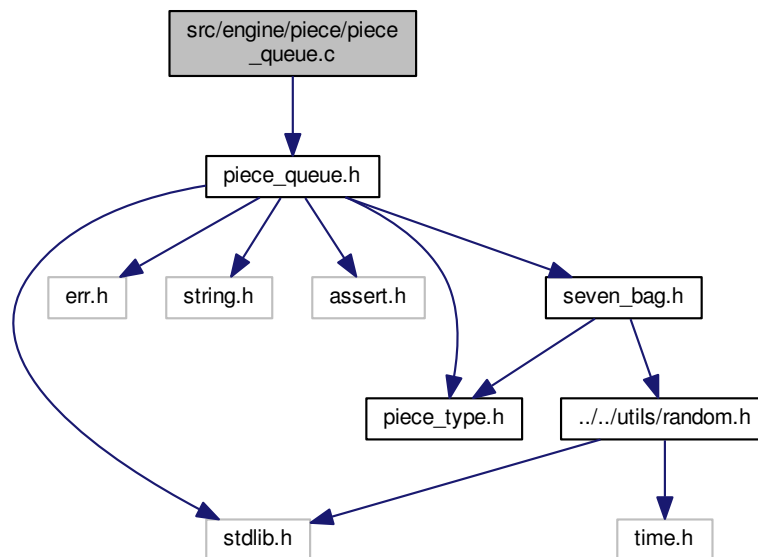
```
Piece* piece_random (
    int x,
    int y,
    Angle angle )
```

4.19 src/engine/piece/piece_queue.c File Reference

Piece (p. 7) queue.

```
#include "piece_queue.h"
```

Include dependency graph for piece_queue.c:



Functions

- **PieceQueue** * `piece_queue_create` ()
- void `piece_queue_free` (**PieceQueue** *q)
- void `piece_queue_fill_data` (**PieceType** *data, size_t length)
- void `piece_queue_extend` (**PieceQueue** *q)
- **PieceType** `piece_queue_get` (**PieceQueue** *q, size_t index)

4.19.1 Detailed Description

Piece (p. 7) queue.

Author

S4MasterRace

Version

2.0

4.19.2 Function Documentation

4.19.2.1 piece_queue_create()

```
PieceQueue* piece_queue_create ( )
```

4.19.2.2 piece_queue_extend()

```
void piece_queue_extend (
    PieceQueue * q )
```

4.19.2.3 piece_queue_fill_data()

```
void piece_queue_fill_data (
    PieceType * data,
    size_t length )
```

4.19.2.4 piece_queue_free()

```
void piece_queue_free (
    PieceQueue * q )
```

4.19.2.5 piece_queue_get()

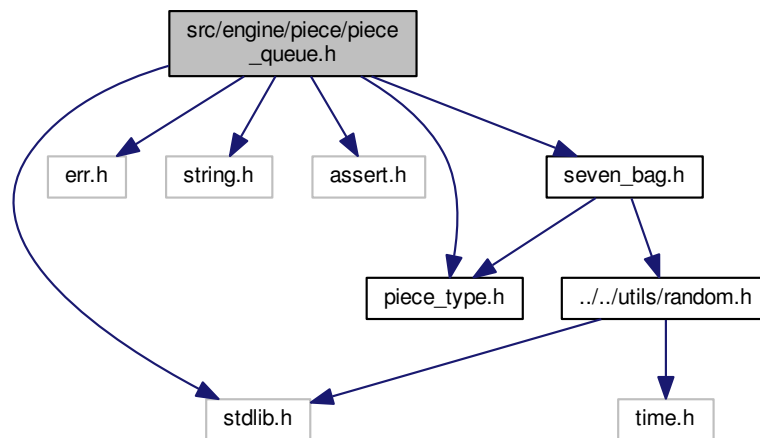
```
PieceType piece_queue_get (
    PieceQueue * q,
    size_t index )
```

4.20 src/engine/piece/piece_queue.h File Reference

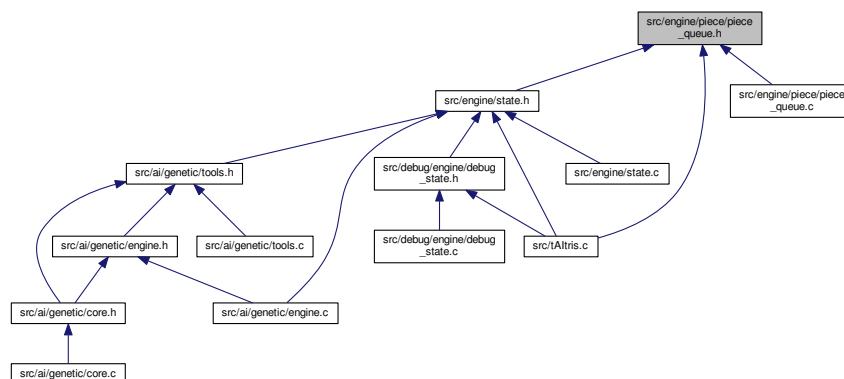
Piece (p. 7) queue.

```
#include <stdlib.h>
#include <err.h>
#include <string.h>
#include <assert.h>
#include "piece_type.h"
#include "seven_bag.h"
```

Include dependency graph for piece_queue.h:



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



Data Structures

- struct **PieceQueue**

Macros

- `#define PIECE_QUEUE_LENGTH 100`

Functions

- `PieceQueue * piece_queue_create ()`
- `void piece_queue_free (PieceQueue *q)`
- `void piece_queue_fill_data (PieceType *data, size_t length)`
- `void piece_queue_extend (PieceQueue *q)`
- `PieceType piece_queue_get (PieceQueue *q, size_t index)`

4.20.1 Detailed Description

Piece (p. 7) queue.

Author

S4MasterRace

Version

2.0

4.20.2 Macro Definition Documentation

4.20.2.1 PIECE_QUEUE_LENGTH

```
#define PIECE_QUEUE_LENGTH 100
```

4.20.3 Function Documentation

4.20.3.1 piece_queue_create()

```
PieceQueue* piece_queue_create ( )
```

4.20.3.2 piece_queue_extend()

```
void piece_queue_extend (  
    PieceQueue * q )
```

4.20.3.3 piece_queue_fill_data()

```
void piece_queue_fill_data (  
    PieceType * data,  
    size_t length )
```

4.20.3.4 piece_queue_free()

```
void piece_queue_free (  
    PieceQueue * q )
```

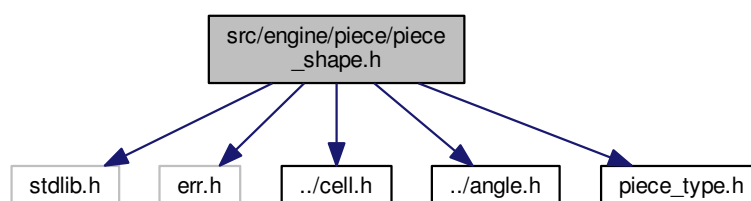
4.20.3.5 piece_queue_get()

```
PieceType piece_queue_get (  
    PieceQueue * q,  
    size_t index )
```

4.21 src/engine/piece/piece_shape.h File Reference

Piece (p. 7) shape.

```
#include <stdlib.h>  
#include <err.h>  
#include "../cell.h"  
#include "../angle.h"  
#include "piece_type.h"  
Include dependency graph for piece_shape.h:
```



4.21.2.1 PIECE_SHAPE_HEIGHT

```
#define PIECE_SHAPE_HEIGHT 4
```

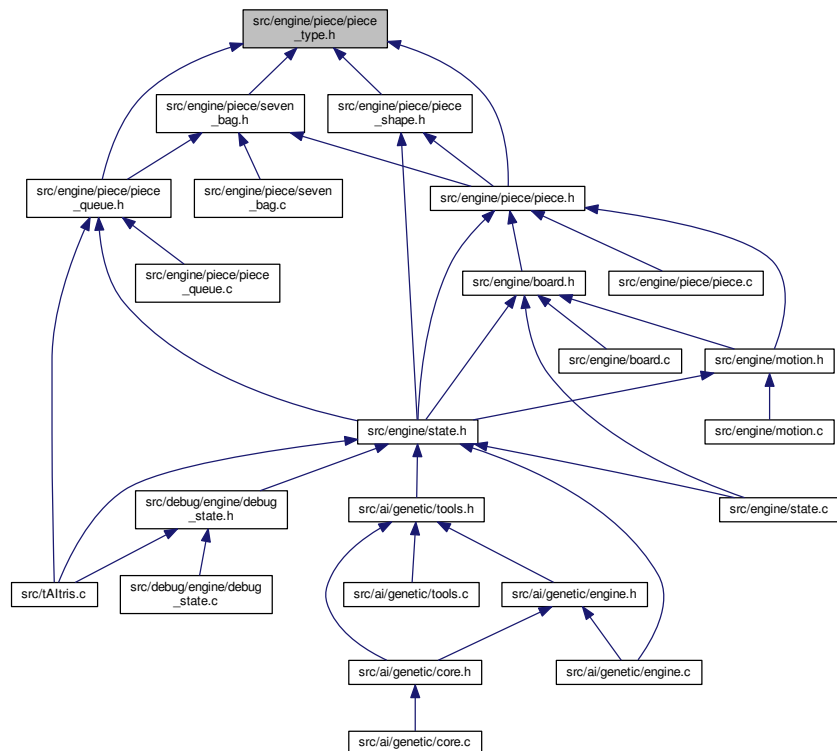
4.21.2.2 PIECE_SHAPE_WIDTH

```
#define PIECE_SHAPE_WIDTH 4
```

4.22 src/engine/piece/piece_type.h File Reference

Piece (p. 7) type.

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



Macros

- `#define PIECE_TYPE_ESIZE 7`

Enumerations

- `enum PieceType {`
`PIECE_TYPE_I, PIECE_TYPE_O, PIECE_TYPE_T, PIECE_TYPE_L,`
`PIECE_TYPE_J, PIECE_TYPE_Z, PIECE_TYPE_S }`

4.22.1 Detailed Description

Piece (p. 7) type.

Author

S4MasterRace

Version

2.0

4.22.2 Macro Definition Documentation

4.22.2.1 PIECE_TYPE_ESIZE

```
#define PIECE_TYPE_ESIZE 7
```

4.22.3 Enumeration Type Documentation

4.22.3.1 PieceType

enum **PieceType**

Enumerator

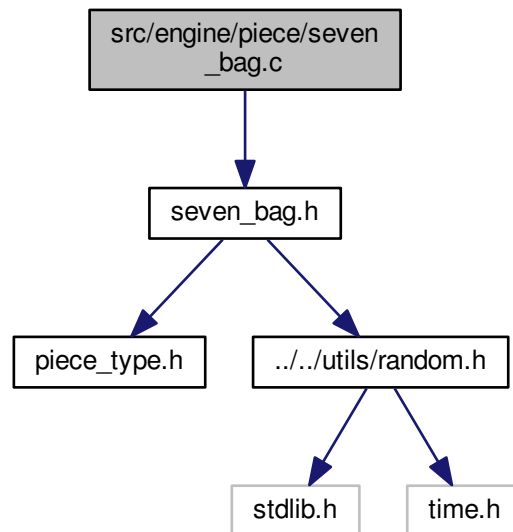
PIECE_TYPE_I	
PIECE_TYPE_O	
PIECE_TYPE_T	
PIECE_TYPE_L	
PIECE_TYPE_J	
PIECE_TYPE_Z	
PIECE_TYPE_S	

4.23 src/engine/piece/seven_bag.c File Reference

7-Bag generator

```
#include "seven_bag.h"
```

Include dependency graph for seven_bag.c:



Functions

- void **seven_bag_init** (**PieceType** *bag)
- void **seven_bag_swap** (**PieceType** *a, **PieceType** *b)
- void **seven_bag_shuffle** (**PieceType** *bag)
- **PieceType** **seven_bag_draw** ()

4.23.1 Detailed Description

7-Bag generator

Author

S4MasterRace

Version

2.0

4.23.2 Function Documentation

4.23.2.1 seven_bag_draw()

```
PieceType seven_bag_draw ( )
```

4.23.2.2 seven_bag_init()

```
void seven_bag_init (
    PieceType * bag )
```

4.23.2.3 seven_bag_shuffle()

```
void seven_bag_shuffle (
    PieceType * bag )
```

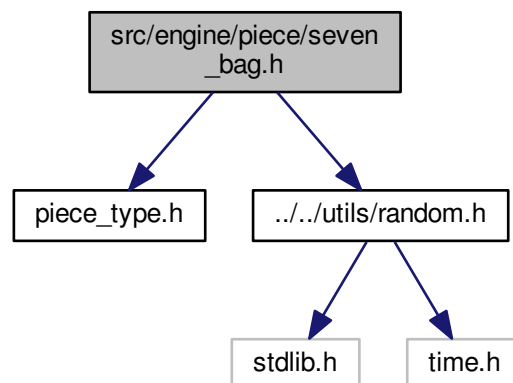
4.23.2.4 seven_bag_swap()

```
void seven_bag_swap (
    PieceType * a,
    PieceType * b )
```

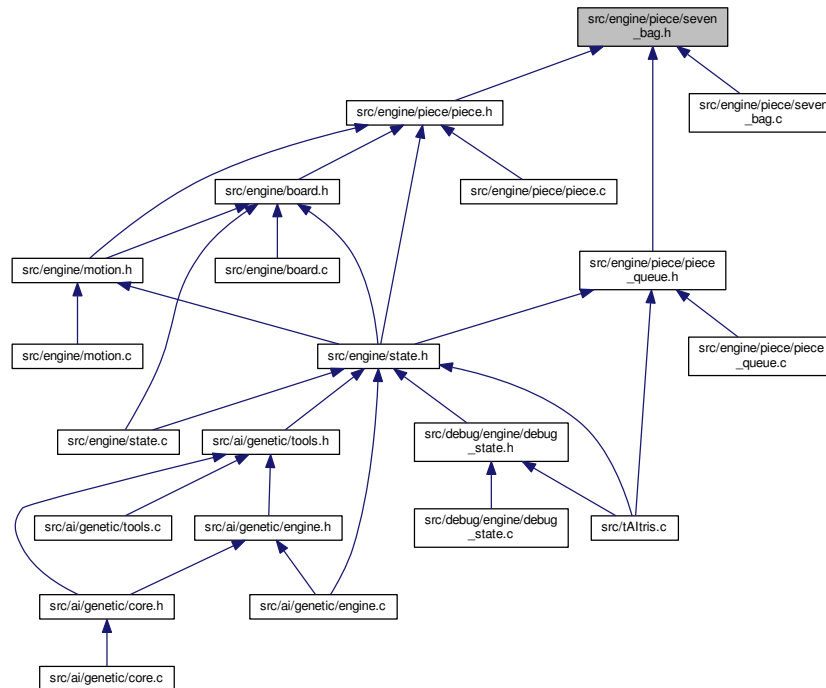
4.24 src/engine/piece/seven_bag.h File Reference

7-Bag generator

```
#include "piece_type.h"
#include "../utils/random.h"
Include dependency graph for seven_bag.h:
```



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



Functions

- void **seven_bag_init** (PieceType *bag)
- void **seven_bag_swap** (PieceType *a, PieceType *b)
- void **seven_bag_shuffle** (PieceType *bag)
- PieceType **seven_bag_draw** ()

4.24.1 Detailed Description

7-Bag generator

Author

S4MasterRace

Version

2.0

4.24.2 Function Documentation

4.24.2.1 seven_bag_draw()

```
PieceType seven_bag_draw ( )
```

4.24.2.2 seven_bag_init()

```
void seven_bag_init (
    PieceType * bag )
```

4.24.2.3 seven_bag_shuffle()

```
void seven_bag_shuffle (
    PieceType * bag )
```

4.24.2.4 seven_bag_swap()

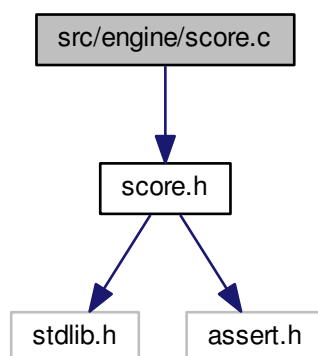
```
void seven_bag_swap (
    PieceType * a,
    PieceType * b )
```

4.25 src/engine/score.c File Reference

Scoring system.

```
#include "score.h"
```

Include dependency graph for score.c:



Functions

- unsigned int **score_compute_break** (const int hist[], size_t len, unsigned int level)

4.25.1 Detailed Description

Scoring system.

Author

S4MasterRace

Version

2.0

4.25.2 Function Documentation

4.25.2.1 score_compute_break()

```
unsigned int score_compute_break (  
    const int hist[],  
    size_t len,  
    unsigned int level )
```

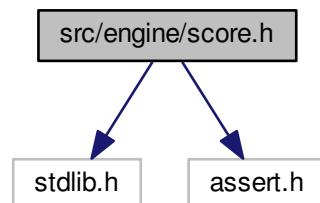
4.26 src/engine/score.h File Reference

Scoring system.

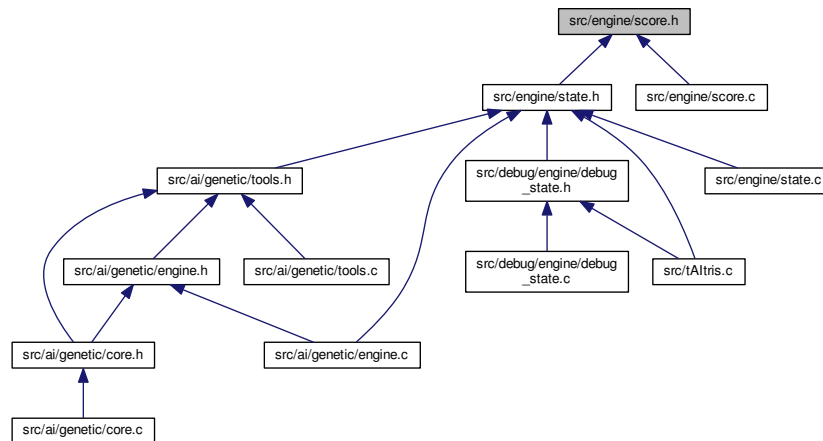
```
#include <stdlib.h>
```

```
#include <assert.h>
```

Include dependency graph for score.h:



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



Macros

- `#define SCORE_SINGLE 100`
- `#define SCORE_DOUBLE 300`
- `#define SCORE_TRIPLE 500`
- `#define SCORE_TETRIS 800`
- `#define SCORE_SDROP 1`
- `#define SCORE_HDROP 2`

Functions

- unsigned int **score_compute_break** (const int hist[], size_t len, unsigned int level)

4.26.1 Detailed Description

Scoring system.

Author

S4MasterRace

Version

2.0

4.26.2 Macro Definition Documentation

4.26.2.1 SCORE_DOUBLE

```
#define SCORE_DOUBLE 300
```

4.26.2.2 SCORE_HDROP

```
#define SCORE_HDROP 2
```

4.26.2.3 SCORE_SDROP

```
#define SCORE_SDROP 1
```

4.26.2.4 SCORE_SINGLE

```
#define SCORE_SINGLE 100
```

4.26.2.5 SCORE_TETRIS

```
#define SCORE_TETRIS 800
```

4.26.2.6 SCORE_TRIPLE

```
#define SCORE_TRIPLE 500
```

4.26.3 Function Documentation

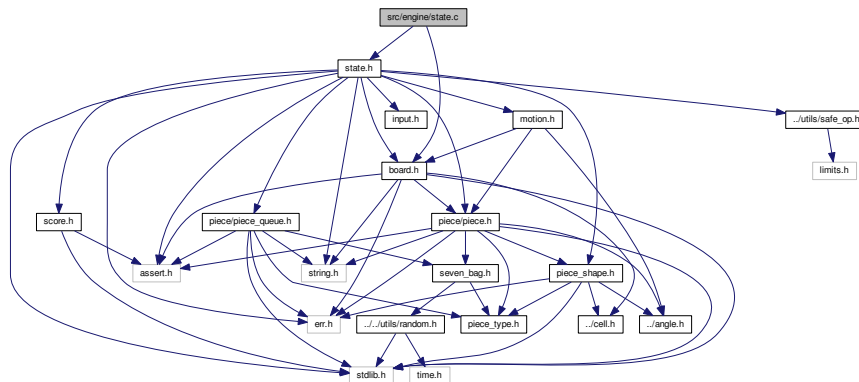
4.26.3.1 score_compute_break()

```
unsigned int score_compute_break (
    const int hist[],
    size_t len,
    unsigned int level )
```

4.27 src/engine/state.c File Reference

State (p. 9).

```
#include "state.h"
#include "board.h"
Include dependency graph for state.c:
```



Functions

- **State * state_create** ()
- **void state_init** (**State** *state, **PieceQueue** *q)
- **void state_free** (**State** *state)
- **State * state_copy** (const **State** *state)
- **Piece * state_create_piece** (**State** *state)
- **void state_next_piece** (**State** *state)
- **int state_step** (**State** *state)
- **int state_apply_input** (**State** *state, **Input** input)
- **int state_apply_inputs** (**State** *state, **Input** input[], size_t len)
- **int state_can_apply_input** (**State** *state, **Input** input)
- **int state_can_apply_inputs** (**State** *state, **Input** input[], size_t len)

4.27.2 Function Documentation

4.27.2.1 state_apply_input()

```
int state_apply_input (
    State * state,
    Input input )
```

4.27.2.2 state_apply_inputs()

```
int state_apply_inputs (
    State * state,
    Input input[],
    size_t len )
```

4.27.2.3 state_can_apply_input()

```
int state_can_apply_input (
    State * state,
    Input input )
```

4.27.2.4 state_can_apply_inputs()

```
int state_can_apply_inputs (
    State * state,
    Input input[],
    size_t len )
```

4.27.2.5 state_copy()

```
State* state_copy (
    const State * state )
```


4.27.2.6 state_create()

```
State* state_create ( )
```

4.27.2.7 state_create_piece()

```
Piece* state_create_piece (
    State * state )
```

4.27.2.8 state_free()

```
void state_free (
    State * state )
```

4.27.2.9 state_init()

```
void state_init (
    State * state,
    PieceQueue * q )
```

4.27.2.10 state_next_piece()

```
void state_next_piece (
    State * state )
```

4.27.2.11 state_step()

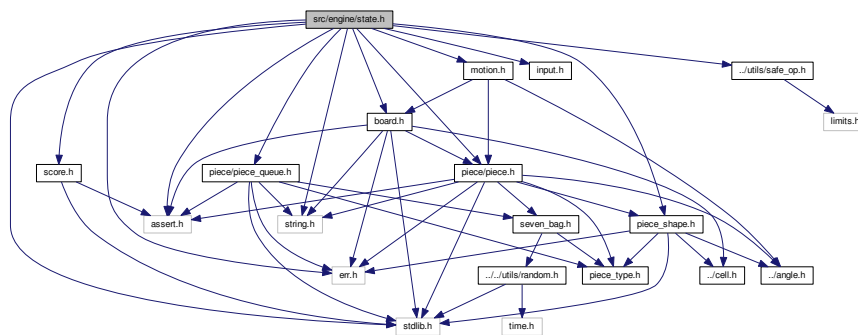
```
int state_step (
    State * state )
```

4.28 src/engine/state.h File Reference

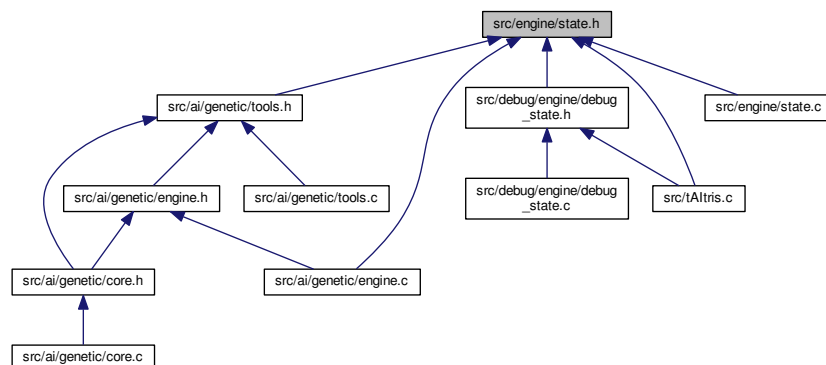
State (p. 9).

```
#include <stdlib.h>
#include <err.h>
#include <string.h>
#include <assert.h>
#include "board.h"
#include "piece/piece.h"
#include "piece/piece_shape.h"
#include "piece/piece_queue.h"
#include "motion.h"
#include "input.h"
#include "score.h"
#include "../utils/safe_op.h"
```

Include dependency graph for state.h:



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



Data Structures

- struct **State**

Functions

- **State** * **state_create** ()
- void **state_init** (**State** *state, **PieceQueue** *q)
- void **state_free** (**State** *state)
- **State** * **state_copy** (const **State** *state)
- **Piece** * **state_create_piece** (**State** *state)
- void **state_next_piece** (**State** *state)
- int **state_step** (**State** *state)
- int **state_apply_input** (**State** *state, **Input** input)
- int **state_apply_inputs** (**State** *state, **Input** input[], size_t len)
- int **state_can_apply_input** (**State** *state, **Input** input)
- int **state_can_apply_inputs** (**State** *state, **Input** input[], size_t len)

4.28.1 Detailed Description

State (p. 9).

Author

S4MasterRace

Version

2.0

4.28.2 Function Documentation

4.28.2.1 state_apply_input()

```
int state_apply_input (
    State * state,
    Input input )
```

4.28.2.2 state_apply_inputs()

```
int state_apply_inputs (
    State * state,
    Input input[],
    size_t len )
```

4.28.2.3 state_can_apply_input()

```
int state_can_apply_input (
    State * state,
    Input input )
```

4.28.2.4 state_can_apply_inputs()

```
int state_can_apply_inputs (
    State * state,
    Input input[],
    size_t len )
```

4.28.2.5 state_copy()

```
State* state_copy (
    const State * state )
```

4.28.2.6 state_create()

```
State* state_create ( )
```

4.28.2.7 state_create_piece()

```
Piece* state_create_piece (
    State * state )
```

4.28.2.8 state_free()

```
void state_free (
    State * state )
```


Macros

- #define **ANSI_ESC** "\x1b"
- #define **ANSI_SGR**(_code_) **ANSI_ESC** "[" #_code_ "m"
- #define **ANSI_RESET** **ANSI_SGR**(0)
- #define **ANSI_BOLD** **ANSI_SGR**(1)
- #define **ANSI_FAINT** **ANSI_SGR**(2)
- #define **ANSI_ITALIC** **ANSI_SGR**(3)
- #define **ANSI_UNDERLINE** **ANSI_SGR**(4)
- #define **ANSI_SBLINK** **ANSI_SGR**(5)
- #define **ANSI_RBLINK** **ANSI_SGR**(6)
- #define **ANSI_CROSSEDOUT** **ANSI_SGR**(9)
- #define **ANSI_FRAMED** **ANSI_SGR**(51)
- #define **ANSI_ENCIRCLED** **ANSI_SGR**(52)
- #define **ANSI_OVERLINED** **ANSI_SGR**(53)
- #define **ANSI_FG_DEFAULT** **ANSI_SGR**(39)
- #define **ANSI_FG_BLACK** **ANSI_SGR**(30)
- #define **ANSI_FG_RED** **ANSI_SGR**(31)
- #define **ANSI_FG_GREEN** **ANSI_SGR**(32)
- #define **ANSI_FG_YELLOW** **ANSI_SGR**(33)
- #define **ANSI_FG_BLUE** **ANSI_SGR**(34)
- #define **ANSI_FG_MAGENTA** **ANSI_SGR**(35)
- #define **ANSI_FG_CYAN** **ANSI_SGR**(36)
- #define **ANSI_FG_WHITE** **ANSI_SGR**(37)
- #define **ANSI_FG_BBLACK** **ANSI_SGR**(90)
- #define **ANSI_FG_BRED** **ANSI_SGR**(91)
- #define **ANSI_FG_BGREEN** **ANSI_SGR**(92)
- #define **ANSI_FG_BYELLOW** **ANSI_SGR**(93)
- #define **ANSI_FG_BBLUE** **ANSI_SGR**(94)
- #define **ANSI_FG_BMAGENTA** **ANSI_SGR**(95)
- #define **ANSI_FG_BCYAN** **ANSI_SGR**(96)
- #define **ANSI_FG_BWHITE** **ANSI_SGR**(97)
- #define **ANSI_BG_DEFAULT** **ANSI_SGR**(49)
- #define **ANSI_BG_BLACK** **ANSI_SGR**(40)
- #define **ANSI_BG_RED** **ANSI_SGR**(41)
- #define **ANSI_BG_GREEN** **ANSI_SGR**(42)
- #define **ANSI_BG_YELLOW** **ANSI_SGR**(43)
- #define **ANSI_BG_BLUE** **ANSI_SGR**(44)
- #define **ANSI_BG_MAGENTA** **ANSI_SGR**(45)
- #define **ANSI_BG_CYAN** **ANSI_SGR**(46)
- #define **ANSI_BG_WHITE** **ANSI_SGR**(47)
- #define **ANSI_BG_BBLACK** **ANSI_SGR**(100)
- #define **ANSI_BG_BRED** **ANSI_SGR**(101)
- #define **ANSI_BG_BGREEN** **ANSI_SGR**(102)
- #define **ANSI_BG_BYELLOW** **ANSI_SGR**(103)
- #define **ANSI_BG_BBLUE** **ANSI_SGR**(104)
- #define **ANSI_BG_BMAGENTA** **ANSI_SGR**(105)
- #define **ANSI_BG_BCYAN** **ANSI_SGR**(106)
- #define **ANSI_BG_BWHITE** **ANSI_SGR**(107)

4.30.1 Detailed Description

ANSI escape code.

Author

S4MasterRace

Version

2.0

4.30.2 Macro Definition Documentation

4.30.2.1 ANSI_BG_BBLACK

```
#define ANSI_BG_BBLACK ANSI_SGR(100)
```

4.30.2.2 ANSI_BG_BBLUE

```
#define ANSI_BG_BBLUE ANSI_SGR(104)
```

4.30.2.3 ANSI_BG_BCYAN

```
#define ANSI_BG_BCYAN ANSI_SGR(106)
```

4.30.2.4 ANSI_BG_BGREEN

```
#define ANSI_BG_BGREEN ANSI_SGR(102)
```

4.30.2.5 ANSI_BG_BLACK

```
#define ANSI_BG_BLACK ANSI_SGR(40)
```


4.30.2.6 ANSI_BG_BLUE

```
#define ANSI_BG_BLUE    ANSI_SGR(44)
```

4.30.2.7 ANSI_BG_BMAGENTA

```
#define ANSI_BG_BMAGENTA ANSI_SGR(105)
```

4.30.2.8 ANSI_BG_BRED

```
#define ANSI_BG_BRED    ANSI_SGR(101)
```

4.30.2.9 ANSI_BG_BWHITE

```
#define ANSI_BG_BWHITE  ANSI_SGR(107)
```

4.30.2.10 ANSI_BG_BYELLOW

```
#define ANSI_BG_BYELLOW ANSI_SGR(103)
```

4.30.2.11 ANSI_BG_CYAN

```
#define ANSI_BG_CYAN    ANSI_SGR(46)
```

4.30.2.12 ANSI_BG_DEFAULT

```
#define ANSI_BG_DEFAULT ANSI_SGR(49)
```

4.30.2.13 ANSI_BG_GREEN

```
#define ANSI_BG_GREEN    ANSI_SGR(42)
```

4.30.2.14 ANSI_BG_MAGENTA

```
#define ANSI_BG_MAGENTA  ANSI_SGR(45)
```

4.30.2.15 ANSI_BG_RED

```
#define ANSI_BG_RED  ANSI_SGR(41)
```

4.30.2.16 ANSI_BG_WHITE

```
#define ANSI_BG_WHITE  ANSI_SGR(47)
```

4.30.2.17 ANSI_BG_YELLOW

```
#define ANSI_BG_YELLOW  ANSI_SGR(43)
```

4.30.2.18 ANSI_BOLD

```
#define ANSI_BOLD  ANSI_SGR(1)
```

4.30.2.19 ANSI_CROSSEDOUT

```
#define ANSI_CROSSEDOUT  ANSI_SGR(9)
```

4.30.2.20 ANSI_ENCIRCLED

```
#define ANSI_ENCIRCLED  ANSI_SGR(52)
```

4.30.2.21 ANSI_ESC

```
#define ANSI_ESC  "\x1b"
```

4.30.2.22 ANSI_FAINT

```
#define ANSI_FAINT ANSI_SGR(2)
```

4.30.2.23 ANSI_FG_BBLACK

```
#define ANSI_FG_BBLACK ANSI_SGR(90)
```

4.30.2.24 ANSI_FG_BBLUE

```
#define ANSI_FG_BBLUE ANSI_SGR(94)
```

4.30.2.25 ANSI_FG_BCYAN

```
#define ANSI_FG_BCYAN ANSI_SGR(96)
```

4.30.2.26 ANSI_FG_BGREEN

```
#define ANSI_FG_BGREEN ANSI_SGR(92)
```

4.30.2.27 ANSI_FG_BLACK

```
#define ANSI_FG_BLACK ANSI_SGR(30)
```

4.30.2.28 ANSI_FG_BLUE

```
#define ANSI_FG_BLUE ANSI_SGR(34)
```

4.30.2.29 ANSI_FG_BMAGENTA

```
#define ANSI_FG_BMAGENTA ANSI_SGR(95)
```

4.30.2.30 ANSI_FG_BRED

```
#define ANSI_FG_BRED    ANSI_SGR(91)
```

4.30.2.31 ANSI_FG_BWHITE

```
#define ANSI_FG_BWHITE  ANSI_SGR(97)
```

4.30.2.32 ANSI_FG_BYELLOW

```
#define ANSI_FG_BYELLOW ANSI_SGR(93)
```

4.30.2.33 ANSI_FG_CYAN

```
#define ANSI_FG_CYAN    ANSI_SGR(36)
```

4.30.2.34 ANSI_FG_DEFAULT

```
#define ANSI_FG_DEFAULT ANSI_SGR(39)
```

4.30.2.35 ANSI_FG_GREEN

```
#define ANSI_FG_GREEN    ANSI_SGR(32)
```

4.30.2.36 ANSI_FG_MAGENTA

```
#define ANSI_FG_MAGENTA  ANSI_SGR(35)
```

4.30.2.37 ANSI_FG_RED

```
#define ANSI_FG_RED      ANSI_SGR(31)
```

4.30.2.38 ANSI_FG_WHITE

```
#define ANSI_FG_WHITE  ANSI_SGR(37)
```

4.30.2.39 ANSI_FG_YELLOW

```
#define ANSI_FG_YELLOW  ANSI_SGR(33)
```

4.30.2.40 ANSI_FRAMED

```
#define ANSI_FRAMED  ANSI_SGR(51)
```

4.30.2.41 ANSI_ITALIC

```
#define ANSI_ITALIC  ANSI_SGR(3)
```

4.30.2.42 ANSI_OVERLINED

```
#define ANSI_OVERLINED  ANSI_SGR(53)
```

4.30.2.43 ANSI_RBLINK

```
#define ANSI_RBLINK  ANSI_SGR(6)
```

4.30.2.44 ANSI_RESET

```
#define ANSI_RESET  ANSI_SGR(0)
```

4.30.2.45 ANSI_SBLINK

```
#define ANSI_SBLINK  ANSI_SGR(5)
```

4.30.2.46 ANSI_SGR

```
#define ANSI_SGR(  
    _code_ )  ANSI_ESC "[" #_code_ "m"
```

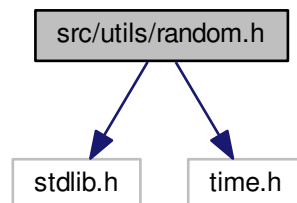
4.30.2.47 ANSI_UNDERLINE

```
#define ANSI_UNDERLINE  ANSI_SGR(4)
```

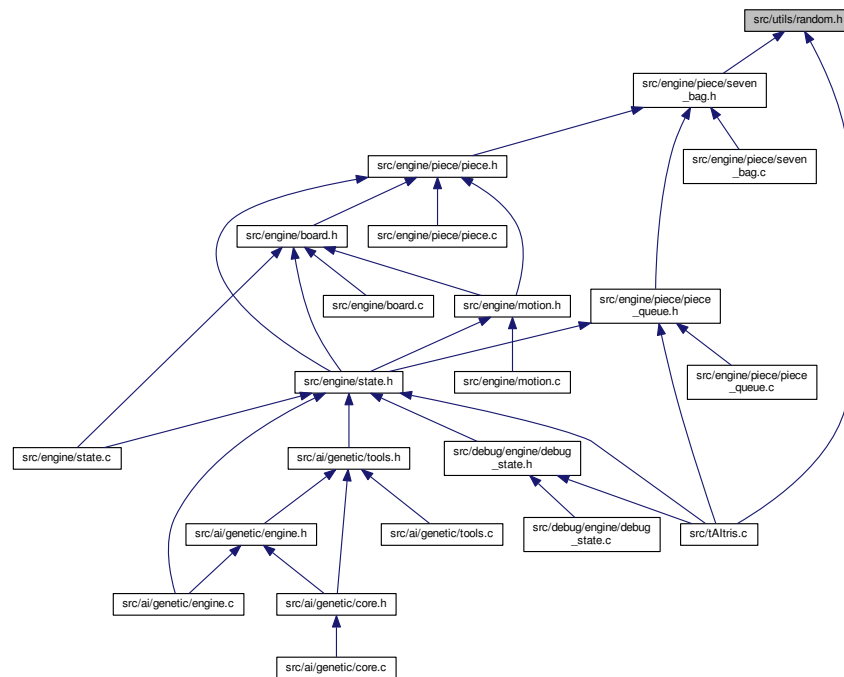
4.31 src/utls/random.h File Reference

Random number generation.

```
#include <stdlib.h>  
#include <time.h>  
Include dependency graph for random.h:
```



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



4.31.1 Detailed Description

Random number generation.

Author

S4MasterRace

Version

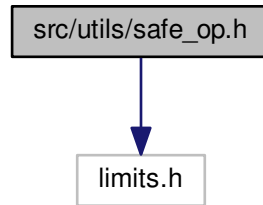
2.0

4.32 src/utils/safe_op.h File Reference

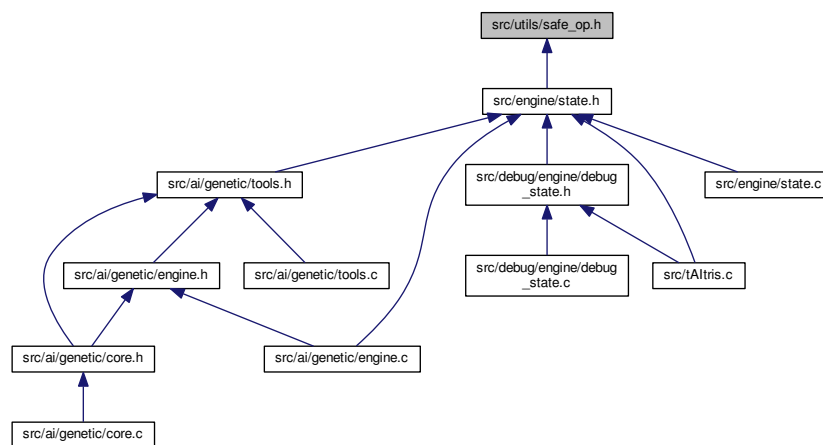
Safe operations.

```
#include <limits.h>
```

Include dependency graph for safe_op.h:



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



Macros

- `#define SAFE_OP_SUCCESS 0`
- `#define SAFE_OP_OVERFLOW 1`
- `#define SAFE_OP_UNDERFLOW (-1)`

4.32.1 Detailed Description

Safe operations.

Author

S4MasterRace

Version

2.0

4.32.2 Macro Definition Documentation

4.32.2.1 SAFE_OP_OVERFLOW

```
#define SAFE_OP_OVERFLOW 1
```

4.32.2.2 SAFE_OP_SUCCESS

```
#define SAFE_OP_SUCCESS 0
```

4.32.2.3 SAFE_OP_UNDERFLOW

```
#define SAFE_OP_UNDERFLOW (-1)
```


Index

ABS
 tools.h, 23

ANGLE_ESIZE
 angle.h, 32

ANSI_BG_BBLACK
 ansi_code.h, 74

ANSI_BG_BBLUE
 ansi_code.h, 74

ANSI_BG_BCYAN
 ansi_code.h, 74

ANSI_BG_BGREEN
 ansi_code.h, 74

ANSI_BG_BLACK
 ansi_code.h, 74

ANSI_BG_BLUE
 ansi_code.h, 74

ANSI_BG_BMAGENTA
 ansi_code.h, 75

ANSI_BG_BRED
 ansi_code.h, 75

ANSI_BG_BWHITE
 ansi_code.h, 75

ANSI_BG_BYELLOW
 ansi_code.h, 75

ANSI_BG_CYAN
 ansi_code.h, 75

ANSI_BG_DEFAULT
 ansi_code.h, 75

ANSI_BG_GREEN
 ansi_code.h, 75

ANSI_BG_MAGENTA
 ansi_code.h, 75

ANSI_BG_RED
 ansi_code.h, 76

ANSI_BG_WHITE
 ansi_code.h, 76

ANSI_BG_YELLOW
 ansi_code.h, 76

ANSI_BOLD
 ansi_code.h, 76

ANSI_CROSSEDOUT
 ansi_code.h, 76

ANSI_ENCIRCLED
 ansi_code.h, 76

ANSI_ESC
 ansi_code.h, 76

ANSI_FAINT
 ansi_code.h, 76

ANSI_FG_BBLACK
 ansi_code.h, 77

ANSI_FG_BBLUE
 ansi_code.h, 77

ANSI_FG_BCYAN
 ansi_code.h, 77

ANSI_FG_BGREEN
 ansi_code.h, 77

ANSI_FG_BLACK
 ansi_code.h, 77

ANSI_FG_BLUE
 ansi_code.h, 77

ANSI_FG_BMAGENTA
 ansi_code.h, 77

ANSI_FG_BRED
 ansi_code.h, 77

ANSI_FG_BWHITE
 ansi_code.h, 78

ANSI_FG_BYELLOW
 ansi_code.h, 78

ANSI_FG_CYAN
 ansi_code.h, 78

ANSI_FG_DEFAULT
 ansi_code.h, 78

ANSI_FG_GREEN
 ansi_code.h, 78

ANSI_FG_MAGENTA
 ansi_code.h, 78

ANSI_FG_RED
 ansi_code.h, 78

ANSI_FG_WHITE
 ansi_code.h, 78

ANSI_FG_YELLOW
 ansi_code.h, 79

ANSI_FRAMED
 ansi_code.h, 79

ANSI_ITALIC
 ansi_code.h, 79

ANSI_OVERLINED
 ansi_code.h, 79

ANSI_RBLINK
 ansi_code.h, 79

ANSI_RESET
 ansi_code.h, 79

ANSI_SBLINK
 ansi_code.h, 79

ANSI_SGR
 ansi_code.h, 79

ANSI_UNDERLINE
 ansi_code.h, 80

- agg_height
 - AiCoefs, 5
- aggregate_height
 - tools.c, 19
 - tools.h, 23
- AiCoefs, 5
 - agg_height, 5
 - bumpiness, 5
 - clears, 5
 - holes, 5
- Angle
 - angle.h, 32
- angle
 - Piece, 7
- angle.h
 - ANGLE_ESIZE, 32
 - Angle, 32
 - Rotation, 32
- ansi_code.h
 - ANSI_BG_BBLACK, 74
 - ANSI_BG_BBLUE, 74
 - ANSI_BG_BCYAN, 74
 - ANSI_BG_BGREEN, 74
 - ANSI_BG_BLACK, 74
 - ANSI_BG_BLUE, 74
 - ANSI_BG_BMAGENTA, 75
 - ANSI_BG_BRED, 75
 - ANSI_BG_BWHITE, 75
 - ANSI_BG_BYELLOW, 75
 - ANSI_BG_CYAN, 75
 - ANSI_BG_DEFAULT, 75
 - ANSI_BG_GREEN, 75
 - ANSI_BG_MAGENTA, 75
 - ANSI_BG_RED, 76
 - ANSI_BG_WHITE, 76
 - ANSI_BG_YELLOW, 76
 - ANSI_BOLD, 76
 - ANSI_CROSSEDOUT, 76
 - ANSI_ENCIRCLED, 76
 - ANSI_ESC, 76
 - ANSI_FAINT, 76
 - ANSI_FG_BBLACK, 77
 - ANSI_FG_BBLUE, 77
 - ANSI_FG_BCYAN, 77
 - ANSI_FG_BGREEN, 77
 - ANSI_FG_BLACK, 77
 - ANSI_FG_BLUE, 77
 - ANSI_FG_BMAGENTA, 77
 - ANSI_FG_BRED, 77
 - ANSI_FG_BWHITE, 78
 - ANSI_FG_BYELLOW, 78
 - ANSI_FG_CYAN, 78
 - ANSI_FG_DEFAULT, 78
 - ANSI_FG_GREEN, 78
 - ANSI_FG_MAGENTA, 78
 - ANSI_FG_RED, 78
 - ANSI_FG_WHITE, 78
 - ANSI_FG_YELLOW, 79
 - ANSI_FRAMED, 79
 - ANSI_ITALIC, 79
 - ANSI_OVERLINED, 79
 - ANSI_RBLINK, 79
 - ANSI_RESET, 79
 - ANSI_SBLINK, 79
 - ANSI_SGR, 79
 - ANSI_UNDERLINE, 80
- BOARD_HEIGHT
 - board.h, 36
- BOARD_HIDDEN
 - board.h, 36
- BOARD_WIDTH
 - board.h, 37
- Board, 6
 - cells, 6
 - height, 6
 - width, 6
- board
 - State, 10
- board.c
 - board_break_lines, 33
 - board_copy, 34
 - board_create, 34
 - board_free, 34
 - board_get_completed_lines, 34
 - board_init, 34
 - board_merge_piece, 34
- board.h
 - BOARD_HEIGHT, 36
 - BOARD_HIDDEN, 36
 - BOARD_WIDTH, 37
 - board_break_lines, 37
 - board_copy, 37
 - board_create, 37
 - board_free, 37
 - board_get_completed_lines, 37
 - board_init, 38
 - board_merge_piece, 38
 - board_reverse_y, 36
- board_break_lines
 - board.c, 33
 - board.h, 37
- board_copy
 - board.c, 34
 - board.h, 37
- board_create
 - board.c, 34
 - board.h, 37
- board_free
 - board.c, 34
 - board.h, 37
- board_get_completed_lines
 - board.c, 34
 - board.h, 37
- board_height
 - tools.c, 20
 - tools.h, 23

- board_heights
 - tools.c, 20
 - tools.h, 23
- board_init
 - board.c, 34
 - board.h, 38
- board_merge_piece
 - board.c, 34
 - board.h, 38
- board_reverse_y
 - board.h, 36
- broken_lines
 - State, 10
- bumpiness
 - AiCoefs, 5
 - tools.c, 20
 - tools.h, 23
- CELL_ESIZE
 - cell.h, 39
- Cell
 - cell.h, 39
- cell.h
 - CELL_ESIZE, 39
 - Cell, 39
- cells
 - Board, 6
- clears
 - AiCoefs, 5
 - tools.c, 20
 - tools.h, 23
- coalescent_clears
 - tools.h, 24
- core.c
 - genetic_show_stats, 14
- core.h
 - genetic_show_stats, 15
- current_piece
 - State, 10
- DEBUG_STATE_COLOR
 - debug_state.h, 29
- DEBUG_STATE_NAME
 - debug_state.h, 29
- DEBUG_STATE_TAG
 - debug_state.h, 29
- DEBUG_TAG
 - debug.h, 26
- data
 - PieceQueue, 8
- debug.h
 - DEBUG_TAG, 26
- debug_state.c
 - debug_state_print, 27
 - debug_state_print_cell, 27
 - debug_state_print_infos, 27
 - debug_state_print_line_number, 27
 - debug_state_print_next_piece, 28
- debug_state.h
 - DEBUG_STATE_COLOR, 29
 - DEBUG_STATE_NAME, 29
 - DEBUG_STATE_TAG, 29
 - debug_state_print, 30
 - debug_state_print_cell, 30
 - debug_state_print_infos, 30
 - debug_state_print_line_number, 30
 - debug_state_print_next_piece, 30
- debug_state_print
 - debug_state.c, 27
 - debug_state.h, 30
- debug_state_print_cell
 - debug_state.c, 27
 - debug_state.h, 30
- debug_state_print_infos
 - debug_state.c, 27
 - debug_state.h, 30
- debug_state_print_line_number
 - debug_state.c, 27
 - debug_state.h, 30
- debug_state_print_next_piece
 - debug_state.c, 28
 - debug_state.h, 30
- engine.c
 - get_coefs, 16
 - get_rank, 17
- engine.h
 - get_coefs, 18
 - get_rank, 18
- fill
 - PieceShape, 9
- genetic_show_stats
 - core.c, 14
 - core.h, 15
- get_coefs
 - engine.c, 16
 - engine.h, 18
- get_rank
 - engine.c, 17
 - engine.h, 18
- height
 - Board, 6
- hole
 - tools.c, 20
 - tools.h, 24
- holes
 - AiCoefs, 5
 - tools.c, 20
 - tools.h, 24
- INPUT_ESIZE
 - input.h, 40
- Input
 - input.h, 41
- input.h
 - INPUT_ESIZE, 40

- INPUT_ESIZE, 40
- Input, 41
- input_counts
 - State, 10
- length
 - PieceQueue, 8
- level
 - State, 10
- main
 - tAltris.c, 72
- motion.c
 - motion_can_move, 42
 - motion_can_rotate, 42
 - motion_is_valid, 42
 - motion_try_down, 42
 - motion_try_move, 43
 - motion_try_rotate, 43
- motion.h
 - motion_can_move, 44
 - motion_can_rotate, 44
 - motion_is_valid, 45
 - motion_try_down, 45
 - motion_try_move, 45
 - motion_try_rotate, 45
- motion_can_move
 - motion.c, 42
 - motion.h, 44
- motion_can_rotate
 - motion.c, 42
 - motion.h, 44
- motion_is_valid
 - motion.c, 42
 - motion.h, 45
- motion_try_down
 - motion.c, 42
 - motion.h, 45
- motion_try_move
 - motion.c, 43
 - motion.h, 45
- motion_try_rotate
 - motion.c, 43
 - motion.h, 45
- next_piece
 - State, 10
- PIECE_QUEUE_LENGTH
 - piece_queue.h, 53
- PIECE_SHAPE_HEIGHT
 - piece_shape.h, 55
- PIECE_SHAPE_WIDTH
 - piece_shape.h, 56
- PIECE_TYPE_ESIZE
 - piece_type.h, 57
- Piece, 7
 - angle, 7
 - shape, 7
 - type, 7
 - x, 7
 - y, 8
- piece.c
 - piece_copy, 46
 - piece_create, 47
 - piece_free, 47
 - piece_random, 47
- piece.h
 - piece_copy, 49
 - piece_create, 49
 - piece_free, 49
 - piece_random, 49
- piece_copy
 - piece.c, 46
 - piece.h, 49
- piece_create
 - piece.c, 47
 - piece.h, 49
- piece_free
 - piece.c, 47
 - piece.h, 49
- piece_queue
 - State, 11
- piece_queue.c
 - piece_queue_create, 51
 - piece_queue_extend, 51
 - piece_queue_fill_data, 51
 - piece_queue_free, 51
 - piece_queue_get, 51
- piece_queue.h
 - PIECE_QUEUE_LENGTH, 53
 - piece_queue_create, 53
 - piece_queue_extend, 53
 - piece_queue_fill_data, 54
 - piece_queue_free, 54
 - piece_queue_get, 54
- piece_queue_create
 - piece_queue.c, 51
 - piece_queue.h, 53
- piece_queue_extend
 - piece_queue.c, 51
 - piece_queue.h, 53
- piece_queue_fill_data
 - piece_queue.c, 51
 - piece_queue.h, 54
- piece_queue_free
 - piece_queue.c, 51
 - piece_queue.h, 54
- piece_queue_get
 - piece_queue.c, 51
 - piece_queue.h, 54
- piece_queue_index
 - State, 11
- piece_random
 - piece.c, 47
 - piece.h, 49
- piece_shape.h

- PIECE_SHAPE_HEIGHT, 55
- PIECE_SHAPE_WIDTH, 56
- piece_type.h
 - PIECE_TYPE_ESIZE, 57
 - PieceType, 57
- PieceQueue, 8
 - data, 8
 - length, 8
- PieceShape, 8
 - fill, 9
 - shape, 9
- PieceType
 - piece_type.h, 57
- Rotation
 - angle.h, 32
- SAFE_OP_OVERFLOW
 - safe_op.h, 83
- SAFE_OP_SUCCESS
 - safe_op.h, 83
- SAFE_OP_UNDERFLOW
 - safe_op.h, 83
- SCORE_DOUBLE
 - score.h, 63
- SCORE_HDROP
 - score.h, 64
- SCORE_SDROP
 - score.h, 64
- SCORE_SINGLE
 - score.h, 64
- SCORE_TETRIS
 - score.h, 64
- SCORE_TRIPLE
 - score.h, 64
- safe_op.h
 - SAFE_OP_OVERFLOW, 83
 - SAFE_OP_SUCCESS, 83
 - SAFE_OP_UNDERFLOW, 83
- score
 - State, 11
- score.c
 - score_compute_break, 62
- score.h
 - SCORE_DOUBLE, 63
 - SCORE_HDROP, 64
 - SCORE_SDROP, 64
 - SCORE_SINGLE, 64
 - SCORE_TETRIS, 64
 - SCORE_TRIPLE, 64
 - score_compute_break, 64
- score_compute_break
 - score.c, 62
 - score.h, 64
- seven_bag.c
 - seven_bag_draw, 59
 - seven_bag_init, 59
 - seven_bag_shuffle, 59
 - seven_bag_swap, 59
- seven_bag.h
 - seven_bag_draw, 60
 - seven_bag_init, 61
 - seven_bag_shuffle, 61
 - seven_bag_swap, 61
- seven_bag_draw
 - seven_bag.c, 59
 - seven_bag.h, 60
- seven_bag_init
 - seven_bag.c, 59
 - seven_bag.h, 61
- seven_bag_shuffle
 - seven_bag.c, 59
 - seven_bag.h, 61
- seven_bag_swap
 - seven_bag.c, 59
 - seven_bag.h, 61
- shape
 - Piece, 7
 - PieceShape, 9
- show_features
 - tools.c, 21
 - tools.h, 24
- src/ai/genetic/core.c, 13
- src/ai/genetic/core.h, 14
- src/ai/genetic/engine.c, 16
- src/ai/genetic/engine.h, 17
- src/ai/genetic/tools.c, 19
- src/ai/genetic/tools.h, 21
- src/debug/debug.h, 25
- src/debug/engine/debug_state.c, 26
- src/debug/engine/debug_state.h, 28
- src/engine/angle.h, 31
- src/engine/board.c, 32
- src/engine/board.h, 35
- src/engine/cell.h, 38
- src/engine/input.h, 40
- src/engine/motion.c, 41
- src/engine/motion.h, 43
- src/engine/piece/piece.c, 46
- src/engine/piece/piece.h, 47
- src/engine/piece/piece_queue.c, 50
- src/engine/piece/piece_queue.h, 52
- src/engine/piece/piece_shape.h, 54
- src/engine/piece/piece_type.h, 56
- src/engine/piece/seven_bag.c, 58
- src/engine/piece/seven_bag.h, 59
- src/engine/score.c, 61
- src/engine/score.h, 62
- src/engine/state.c, 65
- src/engine/state.h, 68
- src/tAltris.c, 71
- src/utils/ansi_code.h, 72
- src/utils/random.h, 80
- src/utils/safe_op.h, 81
- State, 9
 - board, 10
 - broken_lines, 10

- current_piece, 10
- input_counts, 10
- level, 10
- next_piece, 10
- piece_queue, 11
- piece_queue_index, 11
- score, 11
- step, 11
- state.c
 - state_apply_input, 66
 - state_apply_inputs, 66
 - state_can_apply_input, 66
 - state_can_apply_inputs, 66
 - state_copy, 66
 - state_create, 66
 - state_create_piece, 67
 - state_free, 67
 - state_init, 67
 - state_next_piece, 67
 - state_step, 67
- state.h
 - state_apply_input, 69
 - state_apply_inputs, 69
 - state_can_apply_input, 69
 - state_can_apply_inputs, 70
 - state_copy, 70
 - state_create, 70
 - state_create_piece, 70
 - state_free, 70
 - state_init, 70
 - state_next_piece, 71
 - state_step, 71
- state_apply_input
 - state.c, 66
 - state.h, 69
- state_apply_inputs
 - state.c, 66
 - state.h, 69
- state_can_apply_input
 - state.c, 66
 - state.h, 69
- state_can_apply_inputs
 - state.c, 66
 - state.h, 70
- state_copy
 - state.c, 66
 - state.h, 70
- state_create
 - state.c, 66
 - state.h, 70
- state_create_piece
 - state.c, 67
 - state.h, 70
- state_free
 - state.c, 67
 - state.h, 70
- state_init
 - state.c, 67
 - state.h, 70
- state_next_piece
 - state.c, 67
 - state.h, 71
- state_step
 - state.c, 67
 - state.h, 71
- step
 - State, 11
- tAltris.c
 - main, 72
- tools.c
 - aggregate_height, 19
 - board_height, 20
 - board_heights, 20
 - bumpiness, 20
 - clears, 20
 - hole, 20
 - holes, 20
 - show_features, 21
- tools.h
 - ABS, 23
 - aggregate_height, 23
 - board_height, 23
 - board_heights, 23
 - bumpiness, 23
 - clears, 23
 - coalescent_clears, 24
 - hole, 24
 - holes, 24
 - show_features, 24
- type
 - Piece, 7
- width
 - Board, 6
- x
 - Piece, 7
- y
 - Piece, 8