Fifteen_Puzzle_Game

Generated by Doxygen 1.8.17

Chapter 1

Data Structure Index

1.1 Data Structures

Here are the data structures with brief descriptions:

Chapter 2

File Index

2.1 File List

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

Chapter 3

Data Structure Documentation

3.1 listElement Struct Reference

Data Fields

- int number

  The number stored in the linked list element.
- struct listElement * next

  The pointer on the next list element.

3.1.1 Detailed Description

Definition at line 19 of file structures.h.

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

- structures.h

Chapter 4

File Documentation

4.1 functions.c File Reference

```
#include <stdlib.h>
#include <stdio.h>
#include <time.h>
#include <stdbool.h>
#include "functions.h"
#include "structures.h"
```

Functions

- listElement * create_linked_list ()
- int ** create_game_board (listElement *head)
- void show_game_details ()
- void print_game_board (int **game_board)
- enum Move print_move_and_get_move (int **game_board, listElement **head)
- int ** board_update (enum Move move, int **game_board)
- bool is_game_finished (int **game_board)
- void delete_game_board (int **game_board)
- void delete_linked_list (listElement **head)

4.1.1 Function Documentation

4.1.1.1 board_update()

```
int** board_update (
            enum Move move,
            int ** game_board )
```

Function updates the game board based on user's next move.

Parameters

move	Next user's move.
game_board	Two-dimensional array 4x4 that will be updated based on user's next move.

Returns

Updated board

Definition at line 137 of file functions.c.

4.1.1.2 create_game_board()

```
int** create_game_board (
            listElement * head )
```

Function creates a two-dimensional array 4x4 that is filled with the numbers from sent linked list.

Parameters

head	Head of the linked list.

Returns

Two dimensional array 4x4 that is filled with the numbers from sent linked list.

Definition at line 51 of file functions.c.

4.1.1.3 create_linked_list()

```
listElement* create_linked_list ( )
```

Function creates a linked list and fill it with the numbers from 0 to 15 in a random sequence.

Returns

Created linked list filled with the numbers from 0 to 15 in a random sequence.

Definition at line 10 of file functions.c.

4.1.1.4 delete_game_board()

```
void delete_game_board (
            int ** board )
```

Function deletes the two-dimensional array 4x4 that is the game board.

Parameters

game_board	Two-dimensional array 4x4 that is the game board.

Definition at line 194 of file functions.c.

4.1.1.5 delete_linked_list()

```
void delete_linked_list (
            listElement ** head )
```

Function deletes the linked list.

Parameters

head	Head of the linked list.

Definition at line 201 of file functions.c.

4.1.1.6 is_game_finished()

```
bool is_game_finished (
            int ** game_board )
```

Function checks if the game board is ordered according to the game's rules.

Parameters

game_board	Two dimensional 4x4 Game board that is sent to check if it is well-ordered.

Returns

True if the game board is well-ordered, False if it's not.

Definition at line 169 of file functions.c.

4.1.1.7 print_game_board()

```
void print_game_board (
            int ** game_board )
```

Function prints sent game-board on the screen.

Parameters

board	Two-dimensional array 4x4 that is the board of the game.

Definition at line 79 of file functions.c.

4.1.1.8 print_move_and_get_move()

```
enum Move print_move_and_get_move (
            int ** game_board,
            listElement ** head )
```

Function prints the list of the possbile moves and receives a char from the user of his next move. If the user chooses to quit the game, function deletes the 2D array, the linked list and then it closes the application.

Parameters

game_board	Two-dimensional array 4x4, which is deleted if the user chooses to quit the game.
head	Linked list, which is deleted if the user chooses to quit the game.

Returns

Next user's move.

Definition at line 95 of file functions.c.

4.1.1.9 show_game_details()

```
void show_game_details ( )
```

Function shows the title and rules of the game.

Definition at line 73 of file functions.c.

4.2 functions.h File Reference

```
#include <stdio.h>
#include <stdbool.h>
#include "structures.h"
```

Functions

- listElement * create_linked_list ()
- int ** create_game_board (listElement *head)
- void show_game_details ()
- void print_game_board (int **game_board)
- enum Move print_move_and_get_move (int **game_board, listElement **head)
- int ** board_update (enum Move move, int **game_board)
- bool is_game_finished (int **game_board)
- void delete_game_board (int **board)
- void delete_linked_list (listElement **head)

4.2.1 Function Documentation

4.2.1.1 board_update()

```
int** board_update (
            enum Move move,
            int ** game_board )
```

Function updates the game board based on user's next move.

Parameters

move	Next user's move.
game_board	Two-dimensional array 4x4 that will be updated based on user's next move.

Returns

Updated board

Definition at line 137 of file functions.c.

4.2.1.2 create_game_board()

```
int** create_game_board (
            listElement * head )
```

Function creates a two-dimensional array 4x4 that is filled with the numbers from sent linked list.

Parameters

head	Head of the linked list.

Returns

Two dimensional array 4x4 that is filled with the numbers from sent linked list.

Definition at line 51 of file functions.c.

4.2.1.3 create_linked_list()

```
listElement* create_linked_list ( )
```

Function creates a linked list and fill it with the numbers from 0 to 15 in a random sequence.

Returns

Created linked list filled with the numbers from 0 to 15 in a random sequence.

Definition at line 10 of file functions.c.

4.2.1.4 delete_game_board()

```
void delete_game_board (
            int ** board )
```

Function deletes the two-dimensional array 4x4 that is the game board.

Parameters

game_board	Two-dimensional array 4x4 that is the game board.

Definition at line 194 of file functions.c.

4.2.1.5 delete_linked_list()

```
void delete_linked_list (
            listElement ** head )
```

Function deletes the linked list.

Parameters

head	Head of the linked list.

Definition at line 201 of file functions.c.

4.2.1.6 is_game_finished()

```
bool is_game_finished (
            int ** game_board )
```

Function checks if the game board is ordered according to the game's rules.

Parameters

game_board	Two dimensional 4x4 Game board that is sent to check if it is well-ordered.

Returns

True if the game board is well-ordered, False if it's not.

Definition at line 169 of file functions.c.

4.2.1.7 print_game_board()

```
void print_game_board (
            int ** game_board )
```

Function prints sent game-board on the screen.

Parameters

board	Two-dimensional array 4x4 that is the board of the game.

Definition at line 79 of file functions.c.

4.2.1.8 print_move_and_get_move()

```
enum Move print_move_and_get_move (
            int ** game_board,
            listElement ** head )
```

Function prints the list of the possbile moves and receives a char from the user of his next move. If the user chooses to quit the game, function deletes the 2D array, the linked list and then it closes the application.

Parameters

game_board	Two-dimensional array 4x4, which is deleted if the user chooses to quit the game.
head	Linked list, which is deleted if the user chooses to quit the game.

Returns

Next user's move.

Definition at line 95 of file functions.c.

4.2.1.9 show_game_details()

```
void show_game_details ( )
```

Function shows the title and rules of the game.

Definition at line 73 of file functions.c.

4.3 main.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <stdbool.h>
#include <time.h>
#include "functions.h"
#include "structures.h"
```

Functions

- int **main** ()

4.4 structures.h File Reference

```
#include <stdio.h>
```

Data Structures

- struct listElement

Macros

- #define **_CRT_SECURE_NO_WARNINGS**
- #define **NROWS** 4
- #define **NCOLUMNS** 4

Typedefs

- typedef struct listElement **listElement**

Enumerations

- enum Move { moveUp = 0, moveDown = 1, moveLeft = 2, moveRight = 3 }

4.4.1 Enumeration Type Documentation

4.4.1.1 Move

`enum Move`

Enumerative type used to choose next user's move and update the game board.

Enumerator

moveUp	If it's 0, the function called "board_update" moves up a number below the blank space.
moveDown	If it's 1, the function called "board_update" moves down a number above the blank space.
moveLeft	If it's 2, the function called "board_update" moves left a number on the right of the blank space.
moveRight	If it's 3, the function called "board_update" moves right a number on the left of the blank space.

Definition at line 7 of file structures.h.

Index