

Projekt

AUTHOR
Version
Thu Jul 9 2020

Table of Contents

Table of contents

Class Index

Class List

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

Game	4
Piece	5

File Index

File List

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

gameFunctions.c (Definicje funkcji z pliku gameFunctions.h)	6
gameFunctions.h (Plik z deklaracjami funkcji)	10
main.c (Plik z funkcj¹ main)	14

Class Documentation

Game Struct Reference

```
#include <gameFunctions.h>
```

Public Attributes

- **GameBoard board**
dwuwymiarowa tablica opisuj¹ce pole gry
- **int activeWhite**
Iloœæ aktywnych bia³ych pionków
- **int activeBlack**
Iloœæ aktywnych czarnych pionków
- **bool running**
Sprawdzanie czy p³etla gry winna siê odbywaæ

Detailed Description

Struktura opisuj¹ca grê

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

- **gameFunctions.h**

Piece Struct Reference

```
#include <gameFunctions.h>
```

Public Attributes

- **bool active**
mówi czy jest aktywne
- **char colour**
opisuje kolor w - bia³y, W - bia³a królowa, b - czarny, B - czarna królowa
- **bool isQueen**
mówi czy jest królow¹
- **int ypos**
opisuje pozycje y
- **int xpos**
opisuje pozycje x

Detailed Description

Struktura opisuj¹ca pionek

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

- **gameFunctions.h**

File Documentation

gameFunctions.c File Reference

Definicje funkcji z pliku **gameFunctions.h**.

```
#include <stdio.h>
#include <stdlib.h>
#include <stdbool.h>
#include "gameFunctions.h"
```

Functions

- void **initializeGame** (**Game** *game)
- void **initializeBlackPieces** (**Piece** *pieces)
- void **initializeWhitePieces** (**Piece** *pieces)
- **Piece** * **InputFindPiece** (**Piece** *pieces, int *xpos, int *ypos, char colour)
- void **movePieces** (**Piece** *pieces, char colour, **Piece** *enemyPieces)
- bool **checkForEnemiesAsNotQueenBlack** (**Piece** *foundPiece, **Piece** *enemyPieces, **Piece** *ownPieces, int *newxpos, int *newypos)
- bool **checkForEnemiesAsNotQueenWhite** (**Piece** *foundPiece, **Piece** *enemyPieces, **Piece** *ownPieces, int *newxpos, int *newypos)
- bool **checkForFurtherEnemiesAsNotQueenBlack** (**Piece** *foundPiece, **Piece** *enemyPieces, **Piece** *ownPieces, int *newxpos, int *newypos)
- bool **checkForFurtherEnemiesAsNotQueenWhite** (**Piece** *foundPiece, **Piece** *enemyPieces, **Piece** *ownPieces, int *newxpos, int *newypos)
- bool **checkForEnemiesAsQueen** (**Piece** *foundPiece, **Piece** *enemyPieces, **Piece** *ownPieces, int *newxpos, int *newypos)
- bool **checkForFurtherEnemiesAsQueen** (**Piece** *foundPiece, **Piece** *enemyPieces, **Piece** *ownPieces, int *newxpos, int *newypos)
- void **checkIfBecameQueen** (**Piece** *foundpiece)
- void **initializeBoard** (**GameBoard** *board)
- void **updateBoard** (**Piece** *blacks, **Piece** *whites, **GameBoard** *board)
- void **displayBoard** (**GameBoard** board)
- bool **checkIfFinish** (**Piece** *blacks, **Piece** *whites)

Variables

- const char **defaultGameBoard** [8][8]

Detailed Description

Definicje funkcji z pliku **gameFunctions.h**.

Plik zawiera definicje wszystkich funkcji z **gameFunctions.h**

Function Documentation

bool checkForEnemiesAsNotQueenBlack (**Piece** * *foundPiece*, **Piece** * *enemyPieces*, **Piece** * *ownPieces*, int * *newxpos*, int * *newypos*)

Funkcja odpowiedzialna za szukanie przeciwników jako czarna nie-królowa

Parameters

<i>foundPiece</i>	pionek dla którego rozpatrujemy sytuacje
<i>enemyPieces</i>	wskaŹnik na pionki przeciwnika

<i>ownPieces</i>	wskaźnik na własne pionki
<i>newxpos</i>	nowa pozycja na osi x
<i>newypos</i>	nowa pozycja na osi y

bool checkForEnemiesAsNotQueenWhite (Piece * foundPiece, Piece * enemyPieces, Piece * ownPieces, int * newxpos, int * newypos)

Funkcja odpowiedzialna za szukanie przeciwników jako biała nie-królowa

Parameters

<i>foundPiece</i>	pionek dla którego rozpatrujemy sytuację
<i>enemyPieces</i>	wskaźnik na pionki przeciwnika
<i>ownPieces</i>	wskaźnik na własne pionki
<i>newxpos</i>	nowa pozycja na osi x
<i>newypos</i>	nowa pozycja na osi y

bool checkForEnemiesAsQueen (Piece * foundPiece, Piece * enemyPieces, Piece * ownPieces, int * newxpos, int * newypos)

Funkcja odpowiedzialna za szukanie przeciwników jako królowa dowolnego koloru

Parameters

<i>foundPiece</i>	pionek dla którego rozpatrujemy sytuację
<i>enemyPieces</i>	wskaźnik na pionki przeciwnika
<i>ownPieces</i>	wskaźnik na własne pionki
<i>newxpos</i>	nowa pozycja na osi x
<i>newypos</i>	nowa pozycja na osi y

bool checkForFurtherEnemiesAsNotQueenBlack (Piece * foundPiece, Piece * enemyPieces, Piece * ownPieces, int * newxpos, int * newypos)

Funkcja odpowiedzialna za szukanie dalszych (po zabicu) przeciwników jako czarna nie-królowa

Parameters

<i>foundPiece</i>	pionek dla którego rozpatrujemy sytuację
<i>enemyPieces</i>	wskaźnik na pionki przeciwnika
<i>ownPieces</i>	wskaźnik na własne pionki
<i>newxpos</i>	nowa pozycja na osi x
<i>newypos</i>	nowa pozycja na osi y

bool checkForFurtherEnemiesAsNotQueenWhite (Piece * foundPiece, Piece * enemyPieces, Piece * ownPieces, int * newxpos, int * newypos)

Funkcja odpowiedzialna za szukanie dalszych (po zabicu) przeciwników jako biała nie-królowa

Parameters

<i>foundPiece</i>	pionek dla którego rozpatrujemy sytuację
<i>enemyPieces</i>	wskaźnik na pionki przeciwnika
<i>ownPieces</i>	wskaźnik na własne pionki
<i>newxpos</i>	nowa pozycja na osi x
<i>newypos</i>	nowa pozycja na osi y

bool checkForFurtherEnemiesAsQueen (Piece * foundPiece, Piece * enemyPieces, Piece * ownPieces, int * newxpos, int * newypos)

Funkcja odpowiedzialna za szukanie dalszych (po zabicu) przeciwników jako królowa dowolnego koloru

Parameters

<i>foundPiece</i>	pionek dla którego rozpatrujemy sytuację
-------------------	--

<i>enemyPieces</i>	wskaźnik na pionki przeciwnika
<i>ownPieces</i>	wskaźnik na własne pionki
<i>newxpos</i>	nowa pozycja na osi x
<i>newypos</i>	nowa pozycja na osi y

void checkIfBecameQueen (Piece * *foundpiece*)

Funkcja odpowiedzialna za sprawdzenie czy dany pionek kończący swój turę został królową¹

Parameters

<i>foundpiece</i>	pionek dla którego rozpatrujemy sytuację
-------------------	--

bool checkIfFinish (Piece * *blacks*, Piece * *whites*)

Funkcja odpowiedzialna za sprawdzenie czy gra się zakończyła³

Parameters

<i>blacks</i>	wskaźnik na czarne pionki
<i>whites</i>	wskaźnik na białe pionki

void displayBoard (GameBoard *board*)

Funkcja odpowiedzialna za wyświetlanie planszy

Parameters

<i>board</i>	wskaźnik na planszę
--------------	---------------------

void initializeBlackPieces (Piece * *pieces*)

Funkcja odpowiedzialna za ustawienie czarnych pionków

Parameters

<i>pieces</i>	wskaźnik do czarnych pionków
---------------	------------------------------

void initializeBoard (GameBoard * *board*)

Funkcja odpowiedzialna za inicjalizację planszy

Parameters

<i>game</i>	wskaźnik na planszę
-------------	---------------------

void initializeGame (Game * *game*)

Funkcja odpowiedzialna za inicjalizację gry

Parameters

<i>game</i>	wskaźnik na strukturę odpowiedzialną "nadzorowanie" gry
-------------	---

void initializeWhitePieces (Piece * *pieces*)

Funkcja odpowiedzialna za ustawienie białych pionków

Parameters

<i>pieces</i>	wskaźnik do białych pionków
---------------	-----------------------------

void movePieces (Piece * *pieces*, char *colour*, Piece * *enemyPieces*)

Funkcja odpowiedzialna za wykonywanie ruchu pionków

Parameters

<i>pieces</i>	wskaźnik do pionków
<i>colour</i>	kolory pionka
<i>enemyPieces</i>	wskaźnik do pionków przeciwnika

void updateBoard (Piece * *blacks*, Piece * *whites*, GameBoard * *board*)

Funkcja odpowiedzialna za odświeżenie wyświetlanej planszy

Parameters

<i>black</i>	wskaźnik do czarnych pionków
<i>whites</i>	wskaźnik do białych pionków
<i>Gameboard</i>	wskaźnik na planszę gry

Variable Documentation

const char defaultGameBoard[8][8]

```
Initial value:=
{
    {'0', 'w', '0', 'w', '0', 'w', '0', 'w'},
    {'w', '0', 'w', '0', 'w', '0', 'w', '0'},
    {'0', 'w', '0', 'w', '0', 'w', '0', 'w'},
    {'0', '0', '0', '0', '0', '0', '0', '0'},
    {'0', '0', '0', '0', '0', '0', '0', '0'},
    {'b', '0', 'b', '0', 'b', '0', 'b', '0'},
    {'0', 'b', '0', 'b', '0', 'b', '0', 'b'},
    {'b', '0', 'b', '0', 'b', '0', 'b', '0'}
}
```

gameFunctions.h File Reference

Plik z deklaracjami funkcji.

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

Classes

- struct **Piece**
- struct **Game**

Typedefs

- typedef char ** **GameBoard**

Functions

- void **initializeBlackPieces** (**Piece** *pieces)
- void **initializeWhitePieces** (**Piece** *pieces)
- void **movePieces** (**Piece** *pieces, char colour, **Piece** *enemyPieces)
- void **updateBoard** (**Piece** *blacks, **Piece** *whites, **GameBoard** *board)
- void **initializeGame** (**Game** *game)
- void **initializeBoard** (**GameBoard** *board)
- void **displayBoard** (**GameBoard** board)
- bool **checkForEnemiesAsNotQueenBlack** (**Piece** *foundPiece, **Piece** *enemyPieces, **Piece** *ownPieces, int *newxpos, int *newypos)
- bool **checkForFurtherEnemiesAsNotQueenBlack** (**Piece** *foundPiece, **Piece** *enemyPieces, **Piece** *ownPieces, int *newxpos, int *newypos)
- bool **checkForEnemiesAsNotQueenWhite** (**Piece** *foundPiece, **Piece** *enemyPieces, **Piece** *ownPieces, int *newxpos, int *newypos)
- bool **checkForFurtherEnemiesAsNotQueenWhite** (**Piece** *foundPiece, **Piece** *enemyPieces, **Piece** *ownPieces, int *newxpos, int *newypos)
- bool **checkForEnemiesAsQueen** (**Piece** *foundPiece, **Piece** *enemyPieces, **Piece** *ownPieces, int *newxpos, int *newypos)
- bool **checkForFurtherEnemiesAsQueen** (**Piece** *foundPiece, **Piece** *enemyPieces, **Piece** *ownPieces, int *newxpos, int *newypos)
- void **checkIfBecameQueen** (**Piece** *foundpiece)
- **Piece** * **InputFindPiece** (**Piece** *pieces, int xpos, int ypos, char colour)
- bool **checkIfFinish** (**Piece** *blacks, **Piece** *whites)

Variables

- const char **defaultGameBoard** [8][8]

Detailed Description

Plik z deklaracjami funkcji.

Plik nagłówkowy zawieraj¹cy deklaracje wszystkich funkcji i struktur używanych w programie

Typedef Documentation

typedef char** GameBoard

Nazwanie wskaŹnika na wskaŹnik charów(potrzebny do opisu dwuwymiarowej tablicy)
gameBoard

Function Documentation

bool checkForEnemiesAsNotQueenBlack (Piece * *foundPiece*, Piece * *enemyPieces*, Piece * *ownPieces*, int * *newxpos*, int * *newypos*)

Funkcja odpowiedzialna za szukanie przeciwników jako czarna nie-królowa

Parameters

<i>foundPiece</i>	pionek dla którego rozpatrujemy sytuacje
<i>enemyPieces</i>	wskaŹnik na pionki przeciwnika
<i>ownPieces</i>	wskaŹnik na własne pionki
<i>newxpos</i>	nowa pozycja na osi x
<i>newypos</i>	nowa pozycja na osi y

bool checkForEnemiesAsNotQueenWhite (Piece * *foundPiece*, Piece * *enemyPieces*, Piece * *ownPieces*, int * *newxpos*, int * *newypos*)

Funkcja odpowiedzialna za szukanie przeciwników jako biała nie-królowa

Parameters

<i>foundPiece</i>	pionek dla którego rozpatrujemy sytuacje
<i>enemyPieces</i>	wskaŹnik na pionki przeciwnika
<i>ownPieces</i>	wskaŹnik na własne pionki
<i>newxpos</i>	nowa pozycja na osi x
<i>newypos</i>	nowa pozycja na osi y

bool checkForEnemiesAsQueen (Piece * *foundPiece*, Piece * *enemyPieces*, Piece * *ownPieces*, int * *newxpos*, int * *newypos*)

Funkcja odpowiedzialna za szukanie przeciwników jako królowa dowolnego koloru

Parameters

<i>foundPiece</i>	pionek dla którego rozpatrujemy sytuacje
<i>enemyPieces</i>	wskaŹnik na pionki przeciwnika
<i>ownPieces</i>	wskaŹnik na własne pionki
<i>newxpos</i>	nowa pozycja na osi x
<i>newypos</i>	nowa pozycja na osi y

bool checkForFurtherEnemiesAsNotQueenBlack (Piece * *foundPiece*, Piece * *enemyPieces*, Piece * *ownPieces*, int * *newxpos*, int * *newypos*)

Funkcja odpowiedzialna za szukanie dalszych (po zbiciu) przeciwników jako czarna nie-królowa

Parameters

<i>foundPiece</i>	pionek dla którego rozpatrujemy sytuacje
<i>enemyPieces</i>	wskaŹnik na pionki przeciwnika
<i>ownPieces</i>	wskaŹnik na własne pionki
<i>newxpos</i>	nowa pozycja na osi x
<i>newypos</i>	nowa pozycja na osi y

bool checkForFurtherEnemiesAsNotQueenWhite (Piece * foundPiece, Piece * enemyPieces, Piece * ownPieces, int * newxpos, int * newypos)

Funkcja odpowiedzialna za szukanie dalszych (po zbiciu) przeciwników jako biała nie-królowa

Parameters

<i>foundPiece</i>	pionek dla którego rozpatrujemy sytuację
<i>enemyPieces</i>	wskaźnik na pionki przeciwnika
<i>ownPieces</i>	wskaźnik na własne pionki
<i>newxpos</i>	nowa pozycja na osi x
<i>newypos</i>	nowa pozycja na osi y

bool checkForFurtherEnemiesAsQueen (Piece * foundPiece, Piece * enemyPieces, Piece * ownPieces, int * newxpos, int * newypos)

Funkcja odpowiedzialna za szukanie dalszych (po zbiciu) przeciwników jako królowa dowolnego koloru

Parameters

<i>foundPiece</i>	pionek dla którego rozpatrujemy sytuację
<i>enemyPieces</i>	wskaźnik na pionki przeciwnika
<i>ownPieces</i>	wskaźnik na własne pionki
<i>newxpos</i>	nowa pozycja na osi x
<i>newypos</i>	nowa pozycja na osi y

void checkIfBecameQueen (Piece * foundpiece)

Funkcja odpowiedzialna za sprawdzenie czy dany pionek kończący swą turę został królową

Parameters

<i>foundpiece</i>	pionek dla którego rozpatrujemy sytuację
-------------------	--

bool checkIfFinish (Piece * blacks, Piece * whites)

Funkcja odpowiedzialna za sprawdzenie czy gra się zakończyła

Parameters

<i>blacks</i>	wskaźnik na czarne pionki
<i>whites</i>	wskaźnik na białe pionki

void displayBoard (GameBoard board)

Funkcja odpowiedzialna za wyświetlanie planszy

Parameters

<i>board</i>	wskaźnik na planszę
--------------	---------------------

void initializeBlackPieces (Piece * pieces)

Funkcja odpowiedzialna za ustawienie czarnych pionków

Parameters

<i>pieces</i>	wskaźnik do czarnych pionków
---------------	------------------------------

void initializeBoard (GameBoard * board)

Funkcja odpowiedzialna za inicjalizację planszy

Parameters

<i>game</i>	wskaźnik na planszę
-------------	---------------------

void initializeGame (Game * game)

Funkcja odpowiedzialna za inicjalizację gry

Parameters

<i>game</i>	wskaźnik na strukturę odpowiedzialną "nadzorowanie" gry
-------------	---

void initializeWhitePieces (Piece * pieces)

Funkcja odpowiedzialna za ustawienie białych pionków

Parameters

<i>pieces</i>	wskaźnik do białych pionków
---------------	-----------------------------

Piece* InputFindPiece (Piece * pieces, int xpos, int ypos, char colour)

Funkcja odpowiedzialna za znajdowanie pionka którym chcemy ruszyć

Parameters

<i>pieces</i>	wskaźnik na pionki których jest tura
<i>xpos</i>	pozycja na osi x
<i>ypos</i>	pozycja na osi y
<i>colour</i>	kolor pionka

void movePieces (Piece * pieces, char colour, Piece * enemyPieces)

Funkcja odpowiedzialna za wykonywanie ruchu pionków

Parameters

<i>pieces</i>	wskaźnik do pionków
<i>colour</i>	kolor pionka
<i>enemyPieces</i>	wskaźnik do pionków przeciwnika

void updateBoard (Piece * blacks, Piece * whites, GameBoard * board)

Funkcja odpowiedzialna za odświeżenie wyświetlanej planszy

Parameters

<i>black</i>	wskaźnik do czarnych pionków
<i>whites</i>	wskaźnik do białych pionków
<i>Gameboard</i>	wskaźnik na planszę gry

main.c File Reference

Plik z funkcj¹ main.

```
#include <stdlib.h>
#include <stdio.h>
#include <stdbool.h>
#include <windows.h>
#include "gameFunctions.h"
```

Functions

- `int main ()`

Variables

- `Piece blacks` [12]
 - `Piece whites` [12]
-

Detailed Description

Plik z funkcj¹ main.

Plik z funkcj¹ g³ówna, przyjmuje argumenty zawieraj¹ce nazwy plików i innych danych wejœciowych na których program bêdzie operowaæ

Function Documentation

`int main ()`

Funkcj¹ g³ówna. W niej dzieje siê ca³a pêtla gry oraz zapis rozpoczêtech rozgrywek.

Index

INDEX