

My Project

Wygenerowano przez Doxygen 1.8.16

1 Indeks klas	1
1.1 Lista klas	1
2 Indeks plików	3
2.1 Lista plików	3
3 Dokumentacja klas	5
3.1 Dokumentacja struktury threadInfo	5
3.1.1 Opis szczegółowy	5
3.2 Dokumentacja struktury tileInfo	5
3.2.1 Opis szczegółowy	6
4 Dokumentacja plików	9
4.1 Dokumentacja pliku main.c	9
4.1.1 Dokumentacja definicji typów	10
4.1.1.1 threadInfo	10
4.1.1.2 tileInfo	10
4.1.2 Dokumentacja funkcji	11
4.1.2.1 checkColumn()	11
4.1.2.2 checkRow()	11
4.1.2.3 checkSquare()	12
4.1.2.4 displayString()	12
4.1.2.5 fillSudoku()	13
4.1.2.6 generateSudoku()	13
4.1.2.7 load()	13
4.1.2.8 removeElements()	14
4.1.2.9 save()	14
4.1.2.10 solveSudoku()	15
Indeks	17

Rozdział 1

Indeks klas

1.1 Lista klas

Tutaj znajdują się klasy, struktury, unie i interfejsy wraz z ich krótkimi opisami:

threadInfo	5
tileInfo	5

Rozdział 2

Indeks plików

2.1 Lista plików

Tutaj znajduje się lista wszystkich udokumentowanych plików z ich krótkimi opisami:

main.c	9
----------------------------------	---

Rozdział 3

Dokumentacja klas

3.1 Dokumentacja struktury threadInfo

Atrybuty publiczne

- [tileInfo](#)(* **tab**)[sudokuSize]
- short int **more**
- bool **decremented**
- bool **inside**

3.1.1 Opis szczegółowy

Informacje przesyłane do wątku funkcji solveSudoku

Parametry

<i>tab</i>	Tablica sudoku
<i>more</i>	Ile więcej rozwiązań ma rozwiązać
<i>decremented</i>	Czy zmienna more została już zdekrementowana
<i>inside</i>	Czy aktualnie jesteśmy w wątku

Dokumentacja dla tej struktury została wygenerowana z pliku:

- [main.c](#)

3.2 Dokumentacja struktury tileInfo

Atrybuty publiczne

- int **value**
- bool **canBeModified**

3.2.1 Opis szczegółowy

Informacje o jednym polu sudoku

Parametry

<i>value</i>	Przechowywana wartość w polu
<i>canBeModified</i>	Czy użytkownik może edytować dane pole, czy zostało ono wygenerowane podczas generowania tablicy Sudoku

Dokumentacja dla tej struktury została wygenerowana z pliku:

- [main.c](#)

Rozdział 4

Dokumentacja plików

4.1 Dokumentacja pliku main.c

```
#include <SFML/Graphics.h>
#include <stdbool.h>
#include <stdlib.h>
#include <stdio.h>
#include <string.h>
#include <time.h>
```

Komponenty

- struct [tileInfo](#)
- struct [threadInfo](#)

Definicje

- #define **sudokuSize** 9
- #define **elementsToRemove** 25
- #define **WIDTH** 453
- #define **HEIGHT** 542

Definicje typów

- typedef struct [tileInfo](#) [tileInfo](#)
- typedef struct [threadInfo](#) [threadInfo](#)

Funkcje

- bool `save` (const `tileInfo` `tab[sudokuSize][sudokuSize]`)
- bool `load` (`tileInfo` `tab[sudokuSize][sudokuSize]`)
- void `displayString` (const `sfRenderWindow *win`, const `sfColor` `color`, const `sfFont *font`, const int `x`, const int `y`, const int `charSize`, const char `*napis`, const bool `center`)
- bool `checkColumn` (`tileInfo` `tab[sudokuSize][sudokuSize]`, int `i`, int `number`)
- bool `checkRow` (`tileInfo` `tab[sudokuSize][sudokuSize]`, int `i`, int `number`)
- bool `checkSquare` (`tileInfo` `tab[sudokuSize][sudokuSize]`, int `x`, int `y`, int `number`)
- void `fillSudoku` (`tileInfo` `tab[sudokuSize][sudokuSize]`, int `number`)
- void `removeElements` (`tileInfo` `tab[sudokuSize][sudokuSize]`, int `elements`)
- void `generateSudoku` (`tileInfo` `tab[sudokuSize][sudokuSize]`, bool `*redo`)
- void `solveSudoku` (`threadInfo` `*tInfo`)
- bool `hasWon` (`tileInfo` `tab[sudokuSize][sudokuSize]`)
- void `changeDisplayedText` (char `**dst`, char `*src`, int `**positions`)
- void `handleWindow` (`tileInfo` `tab[sudokuSize][sudokuSize]`)
- int `main` ()

4.1.1 Dokumentacja definicji typów

4.1.1.1 threadInfo

```
typedef struct threadInfo threadInfo
```

Informacje przesyłane do wątku funkcji `solveSudoku`

Parametry

<i>tab</i>	Tablica sudoku
<i>more</i>	Ile więcej rozwiązań ma rozwiązać
<i>decremented</i>	Czy zmienna <i>more</i> została już zdekrementowana
<i>inside</i>	Czy aktualnie jesteśmy w wątku

4.1.1.2 tileInfo

```
typedef struct tileInfo tileInfo
```

Informacje o jednym polu sudoku

Parametry

<i>value</i>	Przechowywana wartość w polu
<i>canBeModified</i>	Czy użytkownik może edytować dane pole, czy zostało ono wygenerowane podczas generowania tablicy Sudoku

4.1.2 Dokumentacja funkcji

4.1.2.1 checkColumn()

```
bool checkColumn (
    tileInfo tab[sudokuSize][sudokuSize],
    int i,
    int number )
```

Funkcja sprawdza, czy możliwe jest umieszczenie danej liczby w danej kolumnie

Parametry

<i>tab</i>	Informacje o tablicy sudoku
<i>i</i>	Index kolumny
<i>number</i>	Liczba do sprawdzenia

Zwraca

Zwraca, czy można wstawić podaną liczbę do danej kolumny

4.1.2.2 checkRow()

```
bool checkRow (
    tileInfo tab[sudokuSize][sudokuSize],
    int i,
    int number )
```

Funkcja sprawdza, czy możliwe jest umieszczenie danej liczby w danym wierszu

Parametry

<i>tab</i>	Informacje o tablicy sudoku
<i>i</i>	Index wiersza
<i>number</i>	Liczba do sprawdzenia

Zwraca

Zwraca, czy można wstawić podaną liczbę do danego wierszu

4.1.2.3 checkSquare()

```
bool checkSquare (
    tileInfo tab[sudokuSize][sudokuSize],
    int x,
    int y,
    int number )
```

Funkcja sprawdza, czy możliwe jest umieszczenie danej liczby w danym kwadracie 3x3

Parametry

<i>tab</i>	Informacje o tablicy sudoku
<i>x</i>	Index x kwadratu
<i>y</i>	Index y kwadratu
<i>number</i>	Liczba do sprawdzenia

Zwraca

Zwraca, czy można wstawić podaną liczbę do danego kwadratu 3x3

4.1.2.4 displayString()

```
void displayString (
    const sfRenderWindow * win,
    const sfColor color,
    const sfFont * font,
    const int x,
    const int y,
    const int charSize,
    const char * napis,
    const bool center )
```

Funkcja wyświetla na ekran podany napis

Parametry

<i>win</i>	Wskaźnik na okno, na którym ma być napis narysowany
<i>color</i>	Jakiego koloru ma być napis
<i>font</i>	Wskaźnik na czcionkę
<i>x</i>	Pozycja x na oknie
<i>y</i>	Pozycja y na oknie
<i>charSize</i>	Rozmiar czcionki
<i>napis</i>	Napis do wyświetlenia
<i>center</i>	Czy tekst ma zostać wyśrodkowany

Zwraca

Funkcja nic nie zwraca

4.1.2.5 fillSudoku()

```
void fillSudoku (
    tileInfo tab[sudokuSize][sudokuSize],
    int number )
```

Funkcja wypełnia sudoku podaną liczbą

Parametry

<i>tab</i>	Informacje o tablicy sudoku
<i>number</i>	Liczba, którą zostanie wypełniona tablica

Zwraca

Funkcja nic nie zwraca

4.1.2.6 generateSudoku()

```
void generateSudoku (
    tileInfo tab[sudokuSize][sudokuSize],
    bool * redo )
```

Rekurencyjna funkcja tworzy poprawną, w pełni wypełnioną tablicę sudoku

Parametry

<i>tab</i>	Informacje o tablicy sudoku
<i>redo</i>	Wskaźnik na zmienną logiczną, która mówi, czy po znalezieniu jednej rozwiązanej tablicy mamy szukać dalej w poszukiwaniu kolejnych (true = pierwsza tablica wygenerowana kończy funkcję)

Zwraca

Funkcja nic nie zwraca

4.1.2.7 load()

```
bool load (
    tileInfo tab[sudokuSize][sudokuSize] )
```

Funkcja wczytuje planszę sudoku z pliku binarnego

Parametry

<i>tab</i>	Informacje o tablicy sudoku
------------	-----------------------------

Zwraca

Zwraca, czy udało się wczytać tablicę z pliku

4.1.2.8 removeElements()

```
void removeElements (
    tileInfo tab[sudokuSize][sudokuSize],
    int elements )
```

Funkcja usuwa 'elements' elementów z tablicy sudoku

Parametry

<i>tab</i>	Informacje o tablicy sudoku
<i>elements</i>	Ile elementów do usunięcia

Zwraca

Funkcja nic nie zwraca

4.1.2.9 save()

```
bool save (
    const tileInfo tab[sudokuSize][sudokuSize] )
```

Funkcja zapisuje aktualną planszę sudoku do pliku binarnego

Parametry

<i>tab</i>	Informacje o tablicy sudoku
------------	-----------------------------

Zwraca

Zwraca, czy udało się zapisać do pliku

4.1.2.10 solveSudoku()

```
void solveSudoku (
    pthreadInfo * pInfo )
```

Rekurencyjna funkcja, wywoływana na osobnym wątku, która pokazuje możliwe rozwiązania tablicy sudoku. Przy każdym kliknięciu klawisza 'S' pokazuje, kolejne rozwiązanie (o ile istnieje takie)

Parametry

<i>pInfo</i>	Wskaźnik na informacje przesyłane do wątku
--------------	--

Zwraca

Funkcja nic nie zwraca

Indeks

- checkColumn
 - main.c, [11](#)
- checkRow
 - main.c, [11](#)
- checkSquare
 - main.c, [11](#)
- displayString
 - main.c, [12](#)
- fillSudoku
 - main.c, [13](#)
- generateSudoku
 - main.c, [13](#)
- load
 - main.c, [13](#)
- main.c, [9](#)
 - checkColumn, [11](#)
 - checkRow, [11](#)
 - checkSquare, [11](#)
 - displayString, [12](#)
 - fillSudoku, [13](#)
 - generateSudoku, [13](#)
 - load, [13](#)
 - removeElements, [14](#)
 - save, [14](#)
 - solveSudoku, [14](#)
 - threadInfo, [10](#)
 - tileInfo, [10](#)
- removeElements
 - main.c, [14](#)
- save
 - main.c, [14](#)
- solveSudoku
 - main.c, [14](#)
- threadInfo, [5](#)
 - main.c, [10](#)
- tileInfo, [5](#)
 - main.c, [10](#)