

Arkusz kalkulacyjny

Generated by Doxygen 1.9.1

1 Hierarchical Index	1
1.1 Class Hierarchy	1
2 Class Index	3
2.1 Class List	3
3 File Index	5
3.1 File List	5
4 Class Documentation	7
4.1 Arkusz Class Reference	7
4.1.1 Detailed Description	7
4.1.2 Constructor & Destructor Documentation	8
4.1.2.1 Arkusz()	8
4.1.3 Member Function Documentation	8
4.1.3.1 modyfikacjaWartosci()	8
4.1.3.2 rozmiarX()	9
4.1.3.3 rozmiarY()	10
4.1.3.4 rozszerzArkusz()	10
4.1.3.5 tworzTablica()	11
4.1.3.6 zwrocWartosc()	12
4.2 Cell Class Reference	13
4.2.1 Detailed Description	14
4.2.2 Constructor & Destructor Documentation	14
4.2.2.1 Cell()	14
4.2.3 Member Function Documentation	14
4.2.3.1 getValue()	14
4.2.3.2 isCalculable()	15
4.2.3.3 operator+() [1/2]	15
4.2.3.4 operator+() [2/2]	15
4.2.3.5 toString()	16
4.3 intCell Class Reference	17
4.3.1 Detailed Description	18
4.3.2 Constructor & Destructor Documentation	18
4.3.2.1 intCell()	18
4.3.3 Member Function Documentation	18
4.3.3.1 getValue()	18
4.3.3.2 setValue()	19
4.3.3.3 toString()	19
4.4 stringCell Class Reference	19
4.4.1 Detailed Description	20
4.4.2 Constructor & Destructor Documentation	20
4.4.2.1 stringCell()	20

4.4.3 Member Function Documentation	21
4.4.3.1 setValue()	21
4.4.3.2 toString()	21
5 File Documentation	23
5.1 ProgramowanieObiektowe/arkusz/komorka/cell.cpp File Reference	23
5.2 ProgramowanieObiektowe/arkusz/komorka/cell.hpp File Reference	23
5.3 ProgramowanieObiektowe/arkusz/komorka/intCell.cpp File Reference	24
5.4 ProgramowanieObiektowe/arkusz/komorka/intCell.hpp File Reference	25
5.5 ProgramowanieObiektowe/arkusz/komorka/stringCell.cpp File Reference	26
5.6 ProgramowanieObiektowe/arkusz/komorka/stringCell.hpp File Reference	27
5.7 ProgramowanieObiektowe/arkusz/tablica/tablica.cpp File Reference	28
5.8 ProgramowanieObiektowe/arkusz/tablica/tablica.hpp File Reference	28
5.9 ProgramowanieObiektowe/arkusz/tablica/tablica_wysw.cpp File Reference	29
5.9.1 Function Documentation	30
5.9.1.1 wprowadzWartosc()	30
5.9.1.2 wyswietlTablica()	31
5.10 ProgramowanieObiektowe/error.hpp File Reference	32
5.10.1 Enumeration Type Documentation	32
5.10.1.1 Wyjatki	32
5.11 ProgramowanieObiektowe/io/zapisOdczyt.cpp File Reference	33
5.12 ProgramowanieObiektowe/io/zapisOdczyt.hpp File Reference	33
5.12.1 Function Documentation	34
5.12.1.1 wczytajPlik()	35
5.12.1.2 zapisPliku()	35
5.13 ProgramowanieObiektowe/main.cpp File Reference	36
5.14 ProgramowanieObiektowe/menu/menu.cpp File Reference	37
5.14.1 Function Documentation	37
5.14.1.1 generujMenu()	38
5.14.1.2 obslugaBledow()	38
5.14.1.3 obslugaMenu()	38
5.14.1.4 parametry()	39
5.14.1.5 parametryKolumny()	40
5.14.1.6 parametryWiersza()	41
5.14.1.7 rozszerzArkusz()	42
5.14.1.8 tworzArkusz()	43
5.14.1.9 wczytanie()	43
5.14.1.10 zapis()	44
5.15 ProgramowanieObiektowe/menu/menu.hpp File Reference	45
5.15.1 Function Documentation	46
5.15.1.1 generujMenu()	46
5.15.1.2 obslugaBledow()	46

5.15.1.3 obslugaMenu()	47
5.15.1.4 parametry()	47
5.15.1.5 parametryKolumny()	48
5.15.1.6 parametryWiersza()	49
5.15.1.7 rozszerzArkusz()	50
5.15.1.8 tworzArkusz()	51
5.15.1.9 wczytanie()	51
5.15.1.10 zapis()	52

Index	53
--------------	-----------

Chapter 1

Hierarchical Index

1.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

Arkusz	7
Cell	13
intCell	17
stringCell	19

Chapter 2

Class Index

2.1 Class List

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

Arkusz	Struktura Arkusz Struktura Arkusz przechowująca tablicę i jej rozmiar	7
Cell	Prototyp klasy komórki	13
intCell	Klasa komórki wartości całkowitych	17
stringCell	Klasa komórki tekstowej	19

Chapter 3

File Index

3.1 File List

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

ProgramowanieObiektowe/ error.hpp	32
ProgramowanieObiektowe/ main.cpp	36
ProgramowanieObiektowe/arkusz/komorka/ cell.cpp	23
ProgramowanieObiektowe/arkusz/komorka/ cell.hpp	23
ProgramowanieObiektowe/arkusz/komorka/ intCell.cpp	24
ProgramowanieObiektowe/arkusz/komorka/ intCell.hpp	25
ProgramowanieObiektowe/arkusz/komorka/ stringCell.cpp	26
ProgramowanieObiektowe/arkusz/komorka/ stringCell.hpp	27
ProgramowanieObiektowe/arkusz/tablica/ tablica.cpp	28
ProgramowanieObiektowe/arkusz/tablica/ tablica.hpp	28
ProgramowanieObiektowe/arkusz/tablica/ tablica_wysw.cpp	29
ProgramowanieObiektowe/arkusz/tablica/ tablica_wysw.hpp	??
ProgramowanieObiektowe/io/ zapisOdczyt.cpp	33
ProgramowanieObiektowe/io/ zapisOdczyt.hpp	33
ProgramowanieObiektowe/menu/ menu.cpp	37
ProgramowanieObiektowe/menu/ menu.hpp	45
ProgramowanieObiektowe/operacje/ operacje.hpp	??
ProgramowanieObiektowe/utility/ utility.hpp	??

Chapter 4

Class Documentation

4.1 Arkusz Class Reference

Struktura [Arkusz](#) Struktura [Arkusz](#) przechowująca tablicę i jej rozmiar.

```
#include <tablica.hpp>
```

Public Member Functions

- [Arkusz](#) (size_t x, size_t y)
Konstruktor tworzący arkusz z tablicą o wyznaczonym rozmiarze Konstruktor tworzący arkusz z tablicą o wyznaczonej ilości kolumn i wierszy.
- [Wyjątki modyfikacjaWartosci](#) (size_t x, size_t y, [Komorka](#) wartosc)
Wstawia określoną wartość do komórki tablicy w arkuszu.
- [Wyjątki rozszerzArkusz](#) (size_t x, size_t y)
Modyfikacja rozmiaru tablicy.
- [Komorka zwrocWartosc](#) (size_t x, size_t y)
Zwraca wartość wybranej komórki
- size_t [rozmiarX](#) ()
Getter ilości kolumn.
- size_t [rozmiarY](#) ()
Getter ilości wierszy.

Static Public Member Functions

- static [Tablica](#) [tworzTablica](#) (size_t kolumny, size_t wiersze)
Tworzy nową dwuwymiarową tablice.

4.1.1 Detailed Description

Struktura [Arkusz](#) Struktura [Arkusz](#) przechowująca tablicę i jej rozmiar.

4.1.2 Constructor & Destructor Documentation

4.1.2.1 Arkusz()

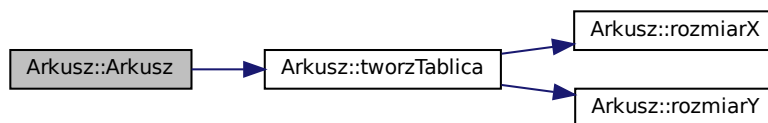
```
Arkusz::Arkusz (
    size_t x,
    size_t y )
```

Konstruktor tworzący arkusz z tablicą o wyznaczonym rozmiarze Konstruktor tworzący arkusz z tablicą o wyznaczonej ilości kolumn i wierszy.

Parameters

in	<i>x</i>	Szerokość tablicy nowego arkusza
in	<i>y</i>	Wysokość tablicy nowego arkusza

Here is the call graph for this function:



4.1.3 Member Function Documentation

4.1.3.1 modyfikacjaWartosci()

```
Wyjatk Arkusz::modyfikacjaWartosci (
    size_t x,
    size_t y,
    Komorka wartosc )
```

Wstawia określoną wartość do komórki tablicy w arkuszu.

Funkcja wstawia wartość do komórki o określonym adresie

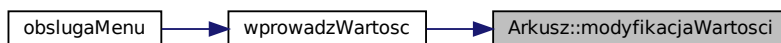
Parameters

in	<i>x</i>	określona kolumna
in	<i>y</i>	określony wiersz
in	<i>wart</i>	wprowadzana wartość

Returns

Kod błędu, BRAK - Brak błędu, [TABLICA_ZAKR](#) - indeks spoza zakresu tablicy

Here is the caller graph for this function:



4.1.3.2 rozmiarX()

```
size_t Arkusz::rozmiarX ( )
```

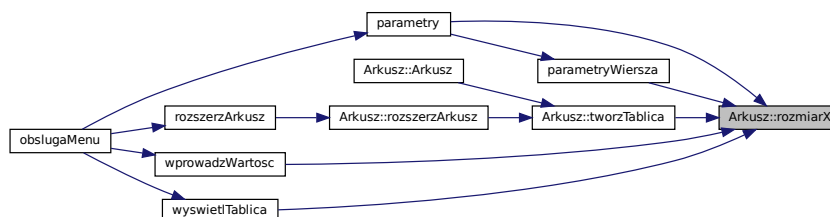
Getter ilości kolumn.

Funkcja zwraca ilość kolumn danego arkusza

Returns

Ilość kolumn arkusza

Here is the caller graph for this function:



4.1.3.3 rozmiarY()

```
size_t Arkusz::rozmiarY ( )
```

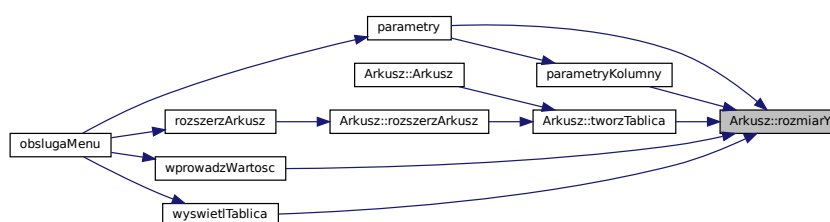
Getter ilości wierszy.

Funkcja zwraca ilość wierszy danego arkusza

Returns

Ilość wierszy arkusza

Here is the caller graph for this function:



4.1.3.4 rozszerzArkusz()

```
Wyjatkami Arkusz::rozszerzArkusz (
    size_t x,
    size_t y )
```

Modyfikacja rozmiaru tablicy.

Funkcja modyfikująca rozmiar arkusza, wartości modyfikowanego arkusza zostają zachowane wraz z rozszerzaniem bądź redukcją rozmiaru (przy redukcji możliwa utrata kolumn/wierszy)

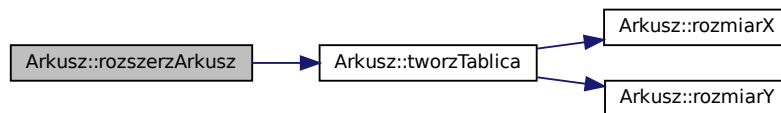
Parameters

in	<i>nowyX</i>	nowa szerokość tablicy
in	<i>nowyY</i>	nowa wysokość tablicy

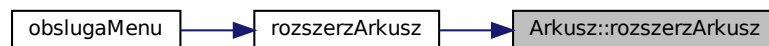
Returns

Kod błędu, BRAK - Brak błędu, [TABLICA_SIZE](#) - Niepoprawny rozmiar tablicy

Here is the call graph for this function:



Here is the caller graph for this function:



4.1.3.5 tworzTablica()

```

Tablica Arkusz::tworzTablica (
    size_t kolumny,
    size_t wiersze ) [static]
  
```

Tworzy nową dwuwymiarową tablicę.

Funkcja generująca tablicę o określonym rozmiarze

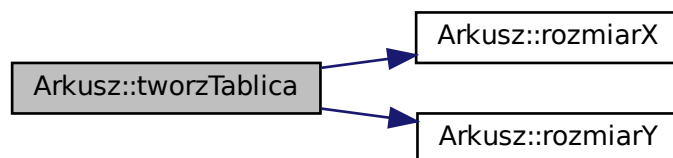
Parameters

in	<i>rozmiarX</i>	Szerokość nowej tablicy
in	<i>rozmiarY</i>	Wysokość nowej tablicy

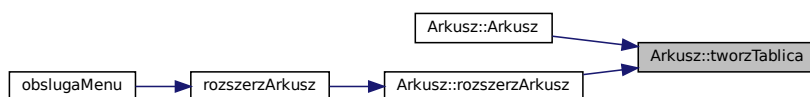
Returns

Nową tablicę dwuwymiarową o wyznaczonych rozmiarach

Here is the call graph for this function:



Here is the caller graph for this function:

**4.1.3.6 zwrocWartosc()**

```

Komorka Arkusz::zwrocWartosc (
    size_t x,
    size_t y )
  
```

Zwraca wartość wybranej komórki

Funkcja zwraca wartość wybranej komórki... jeśli koordynaty x lub y są poza zakresem funkcja zwróci najmniejszą możliwą wartość komórki

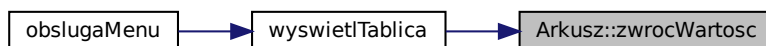
Parameters

in	x	Wybrana kolumna
in	y	Wybrany wiersz

Returns

Wartość komórki w przeciwnym wypadku najmniejsza możliwa wartość

Here is the caller graph for this function:



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

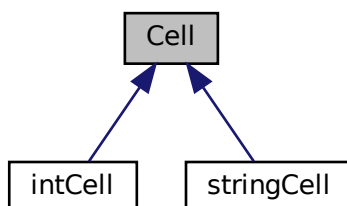
- [ProgramowanieObiektowe/arkusz/tablica/tablica.hpp](#)
- [ProgramowanieObiektowe/arkusz/tablica/tablica.cpp](#)

4.2 Cell Class Reference

Prototyp klasy komórki.

```
#include <cell.hpp>
```

Inheritance diagram for Cell:



Public Member Functions

- [Cell](#) ()
Konstruktor domyślny komórki konstruktor tworzy nową "nieobliczalną" komórkę
- [Cell](#) (bool isCalc)
Konstruktor komórki Konstruktor komórki określający czy komórka może być "obliczalna".
- virtual int [getValue](#) ()
Metoda od pobierania wartości matematycznej Zwraca wartość typu Int.
- virtual std::string [toString](#) ()
Metoda od pobierania wartości ciągu znaków Zwraca wartość typu String - ciąg znaków do wykorzystywania przykładowo w wypisywaniu.
- bool [isCalculable](#) ()
Sprawdź czy komórka może posłużyć w obliczeniach Metoda służy do sprawdzania czy typ komórki umożliwia jej branie udziału w funkcjach arytmetycznych.
- int [operator+](#) ([Cell](#) c1)
Zwraca arytmetyczną sumę komórek Zwraca sumę komórek uzyskanych za pomocą wartości zwracanych przez [getValue\(\)](#) obu sumowanych komórek.
- int [operator+](#) (int wart)
Zwraca sumę wartości Int z komórką Zwraca sumę komórki i zmiennej całkowitej.

4.2.1 Detailed Description

Prototyp klasy komórki.

Klasa prototypowa komórki, posiadająca ogólny szkielet tego co komórka powinna zawierać

4.2.2 Constructor & Destructor Documentation

4.2.2.1 Cell()

```
Cell::Cell (
    bool isCalc ) [inline]
```

Konstruktor komórki Konstruktor komórki określający czy komórka może być "obliczalna".

Parameters

in	<i>isCalc</i>	czy komórka jest obliczalna
----	---------------	-----------------------------

4.2.3 Member Function Documentation

4.2.3.1 getValue()

```
int Cell::getValue ( ) [virtual]
```

Metoda od pobierania wartości matematycznej Zwraca wartość typu Int.

Returns

Zwraca wartość typu int, jeśli komórka jest innego typu niż int zwraca 0.

Reimplemented in [intCell](#).

Here is the caller graph for this function:



4.2.3.2 isCalcualteable()

```
bool Cell::isCalcualteable ( )
```

Sprawdź czy komórka może posłużyć w obliczeniach Metoda służy do sprawdzania czy typ komórki umożliwia jej branie udziału w funkcjach arytmetycznych.

Returns

Zwraca wartość logiczną 'true' dla elementów które można obliczać a 'false' dla tych które nie mogą być obliczane.

4.2.3.3 operator+() [1/2]

```
int Cell::operator+ (
    Cell c1 )
```

Zwraca arytmetyczną sumę komórek Zwraca sumę komórek uzyskanych za pomocą wartości zwracanych przez [getValue\(\)](#) obu sumowanych komórek.

Parameters

in	c1	druga sumowana komórka
----	----	------------------------

Returns

int Suma wartości komórek

Here is the call graph for this function:



4.2.3.4 operator+() [2/2]

```
int Cell::operator+ (
    int wart )
```

Zwraca sumę wartości Int z komórką Zwraca sumę komórki i zmiennej całkowitej.

Parameters

<code>in</code>	<code>wart</code>	wartość całkowita
-----------------	-------------------	-------------------

Returns

int Suma wartości komórek

Here is the call graph for this function:

**4.2.3.5 toString()**

```
std::string Cell::toString ( ) [virtual]
```

Metoda od pobierania wartości ciągu znaków Zwraca wartość typu String - ciąg znaków do wykorzystywania przykładowo w wypisywaniu.

Returns

Zwraca wartość typu string

Reimplemented in [stringCell](#), and [intCell](#).

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

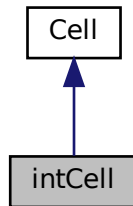
- ProgramowanieObiektowe/arkusz/komorka/[cell.hpp](#)
- ProgramowanieObiektowe/arkusz/komorka/[cell.cpp](#)

4.3 intCell Class Reference

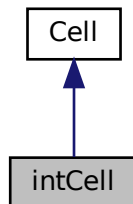
Klasa komórki wartości całkowitych.

```
#include <intCell.hpp>
```

Inheritance diagram for intCell:



Collaboration diagram for intCell:



Public Member Functions

- `intCell` (int Val)
Konstruktor komórki całkowitej Konstruktor komórki z wartością początkową
- int `getValue` ()
Metoda od pobierania wartości matematycznej Zwraca wartość typu Int.
- std::string `toString` ()
Metoda od pobierania wartości ciągu znaków Zwraca wartość typu String - ciąg znaków do wykorzystywania przykładowo w wypisywaniu.
- void `setValue` (int Val)
setter wartości komórki Setter umożliwiający ustawienie wartości całkowitej w komórce

4.3.1 Detailed Description

Klasa komórki wartości całkowitych.

Klasa komórki posiadającej wartość typu Int

4.3.2 Constructor & Destructor Documentation

4.3.2.1 intCell()

```
intCell::intCell (  
    int Val )
```

Konstruktor komórki całkowitej Konstruktor komórki z wartością początkową

Parameters

in	Val	Wartość początkowa komórki
----	-----	----------------------------

Here is the call graph for this function:



4.3.3 Member Function Documentation

4.3.3.1 getValue()

```
int intCell::getValue ( ) [virtual]
```

Metoda od pobierania wartości matematycznej Zwraca wartość typu Int.

Returns

Zwraca wartość typu int, jeśli komórka jest innego typu niż int zwraca 0.

Reimplemented from [Cell](#).

4.3.3.2 setValue()

```
void intCell::setValue (
    int Val )
```

setter wartości komórki Setter umożliwiający ustawienie wartości całkowitej w komórce

Parameters

in	Val	umożliwia modyfikację wartości komórki
----	-----	----------------------------------------

4.3.3.3 toString()

```
std::string intCell::toString ( ) [virtual]
```

Metoda od pobierania wartości ciągu znaków Zwraca wartość typu String - ciąg znaków do wykorzystywania przykładowo w wypisywaniu.

Returns

Zwraca wartość typu string

Reimplemented from [Cell](#).

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

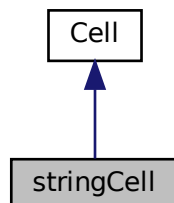
- ProgramowanieObiektowe/arkusz/komorka/[intCell.hpp](#)
- ProgramowanieObiektowe/arkusz/komorka/[intCell.cpp](#)

4.4 stringCell Class Reference

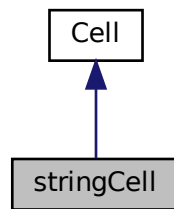
Klasa komórki tekstowej.

```
#include <stringCell.hpp>
```

Inheritance diagram for stringCell:



Collaboration diagram for stringCell:



Public Member Functions

- [stringCell](#) (std::string Val)
Konstruktor komórki tekstowej Konstruktor komórki z wartością początkową
- std::string [toString](#) ()
Metoda od pobierania wartości ciągu znaków Zwraca wartość typu String - ciąg znaków do wykorzystywania przykładowo w wypisywaniu.
- void [setValue](#) (std::string Val)
setter wartości komórki Setter umożliwiający ustawienie wartości tekstowej w komórce

4.4.1 Detailed Description

Klasa komórki tekstowej.

Klasa komórki posiadającej wartość typu string

4.4.2 Constructor & Destructor Documentation

4.4.2.1 stringCell()

```
stringCell::stringCell (
    std::string Val )
```

Konstruktor komórki tekstowej Konstruktor komórki z wartością początkową

Parameters

in	Val	Wartość początkowa komórki
----	-----	----------------------------

4.4.3 Member Function Documentation

4.4.3.1 setValue()

```
void stringCell::setValue (
    std::string Val )
```

setter wartości komórki Setter umożliwiający ustawienie wartości tekstowej w komórce

Parameters

in	Val	umożliwia modyfikację wartości komórki
----	-----	----------------------------------------

4.4.3.2 toString()

```
std::string stringCell::toString ( ) [virtual]
```

Metoda od pobierania wartości ciągu znaków Zwraca wartość typu String - ciąg znaków do wykorzystywania przykładowo w wypisywaniu.

Returns

Zwraca wartość typu string

Reimplemented from [Cell](#).

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

- ProgramowanieObiektowe/arkusz/komorka/[stringCell.hpp](#)
- ProgramowanieObiektowe/arkusz/komorka/[stringCell.cpp](#)

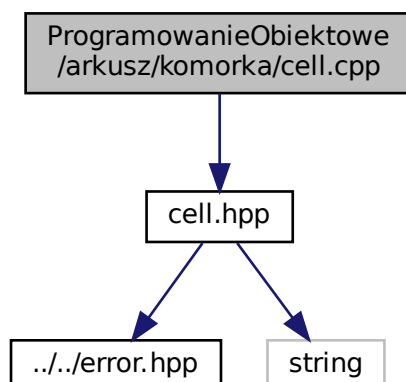
Chapter 5

File Documentation

5.1 ProgramowanieObiektowe/arkusz/komorka/cell.cpp File Reference

```
#include "cell.hpp"
```

Include dependency graph for cell.cpp:

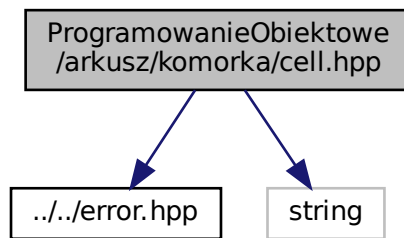


5.2 ProgramowanieObiektowe/arkusz/komorka/cell.hpp File Reference

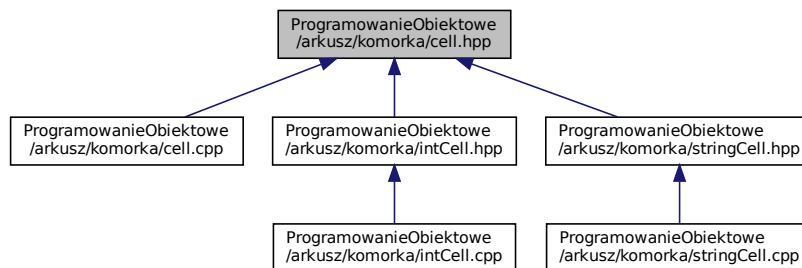
```
#include "../error.hpp"
```

```
#include <string>
```

Include dependency graph for cell.hpp:



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



Classes

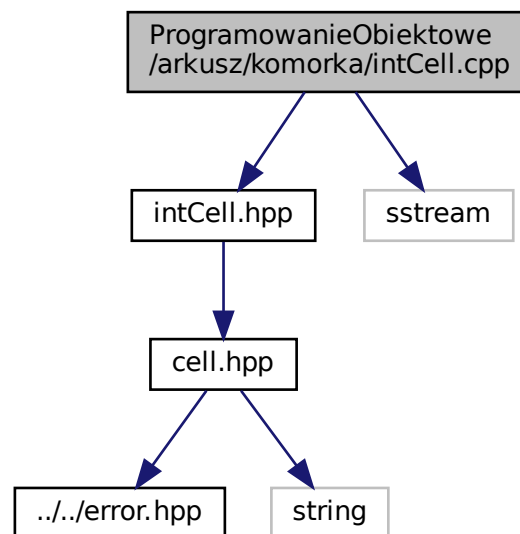
- class [Cell](#)

Prototyp klasy komórki.

5.3 ProgramowanieObiektowe/arkusz/komorka/intCell.cpp File Reference

```
#include "intCell.hpp"
#include <sstream>
```

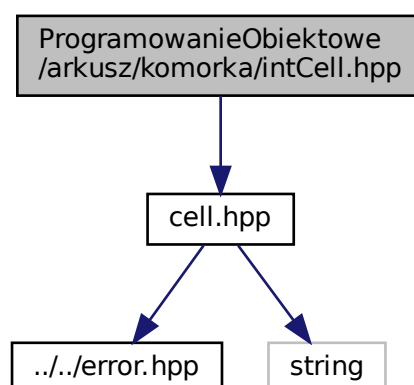
Include dependency graph for intCell.cpp:



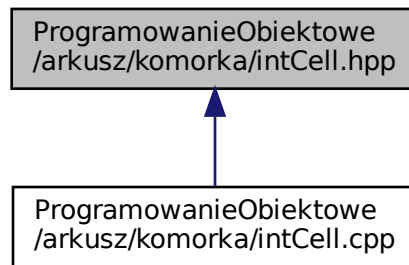
5.4 ProgramowanieObiektowe/arkusz/komorka/intCell.hpp File Reference

```
#include "cell.hpp"
```

Include dependency graph for intCell.hpp:



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



Classes

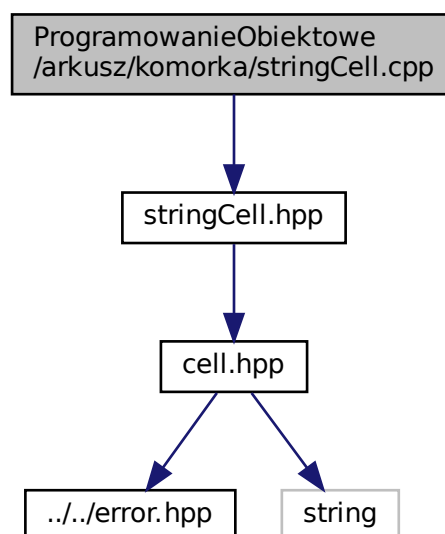
- class `intCell`

Klasa komórki wartości całkowitych.

5.5 ProgramowanieObiektowe/arkusz/komorka/stringCell.cpp File Reference

```
#include "stringCell.hpp"
```

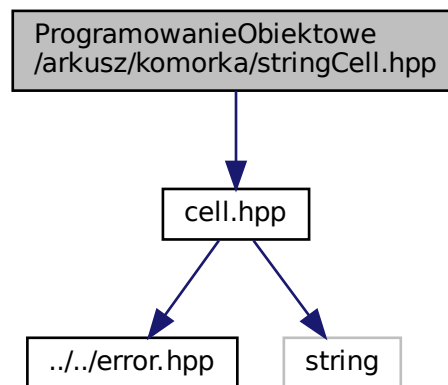
Include dependency graph for `stringCell.cpp`:



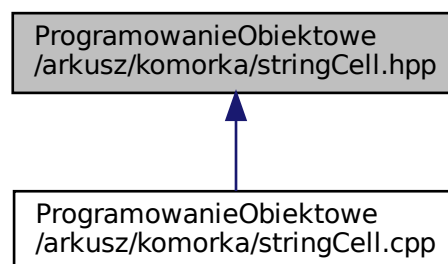
5.6 ProgramowanieObiektowe/arkusz/komorka/stringCell.hpp File Reference

```
#include "cell.hpp"
```

Include dependency graph for stringCell.hpp:



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



Classes

- class [stringCell](#)

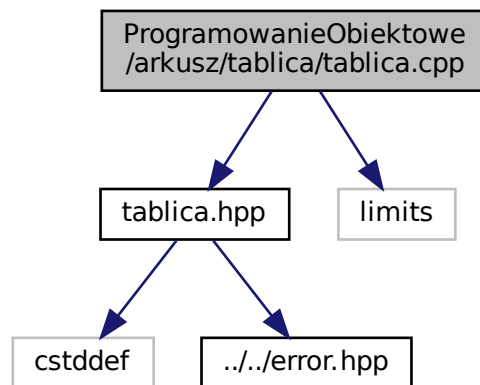
Klasa komórki tekstowej.

5.7 ProgramowanieObiektowe/arkusz/tablica/tablica.cpp File Reference

```
#include "tablica.hpp"
```

```
#include <limits>
```

Include dependency graph for tablica.cpp:

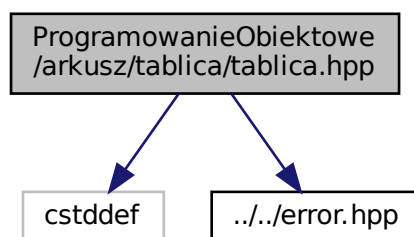


5.8 ProgramowanieObiektowe/arkusz/tablica/tablica.hpp File Reference

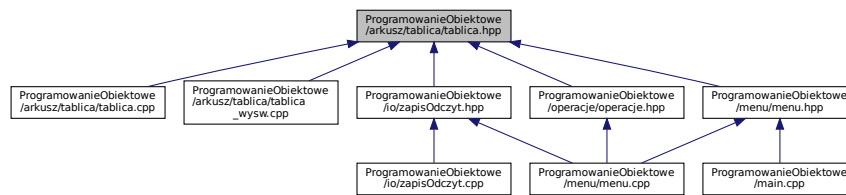
```
#include <cstddef>
```

```
#include "../..error.hpp"
```

Include dependency graph for tablica.hpp:



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



Classes

- class [Arkusz](#)

Struktura [Arkusz](#) Struktura [Arkusz](#) przechowująca tablicę i jej rozmiar.

Typedefs

- typedef int [Komorka](#)

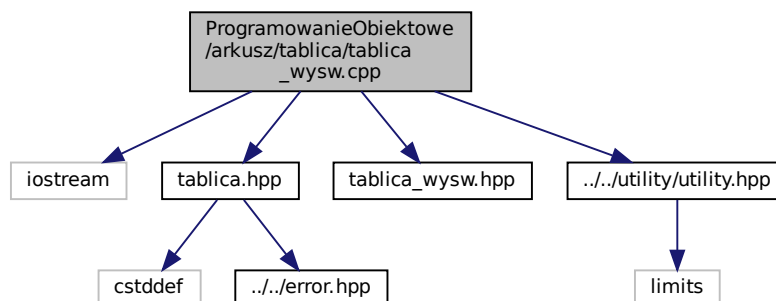
Definicja/alias typu int jako typ określający komórkę

- typedef [Komorka](#) ** [Tablica](#)

Definicja/alias typu Komorka** jako Tablica komórek.

5.9 ProgramowanieObiektowe/arkusz/tablica/tablica_wysw.cpp File Reference

```
#include <iostream>
#include "tablica.hpp"
#include "tablica_wysw.hpp"
#include "../utility/utility.hpp"
Include dependency graph for tablica_wysw.cpp:
```



Functions

- void `wyswietlTablica` (`Arkusz` arkusz)
Generuje podgląd tablicy.
- void `wprowadzWartosc` (`Arkusz` *arkusz)
Interfejs użytkownika wprowadzania nowej wartości.

5.9.1 Function Documentation

5.9.1.1 `wprowadzWartosc()`

```
void wprowadzWartosc (  
    Arkusz * arkusz )
```

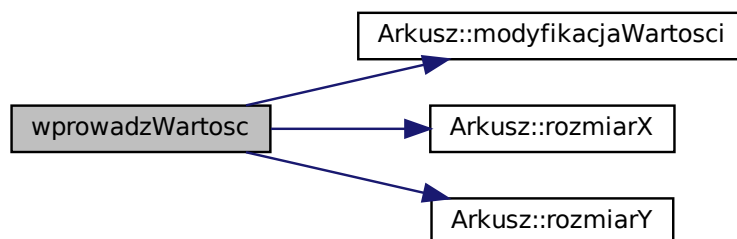
Interfejs użytkownika wprowadzania nowej wartości.

Funkcja interfejsu wprowadzenia wartości do tablicy. Wprowadzana wartość spoza przedziału zostanie zablokowana

Parameters

<code>in, out</code>	<code>arkusz</code>	arkusz która zostaje zmodyfikowany
----------------------	---------------------	------------------------------------

Here is the call graph for this function:



Here is the caller graph for this function:



5.9.1.2 wyswietlTablica()

```
void wyswietlTablica (  
    Arkusz arkusz )
```

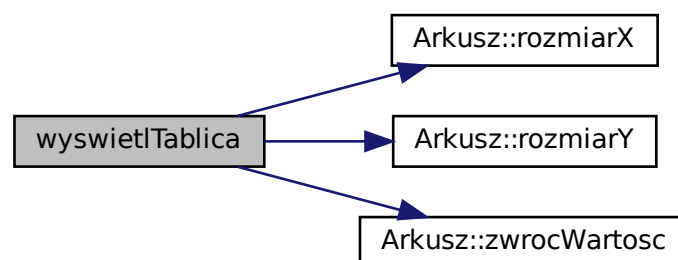
Generuje podgląd tablicy.

Generuje tablicę na konsoli wyświetlając wszystkie zawarte w niej komórki

Parameters

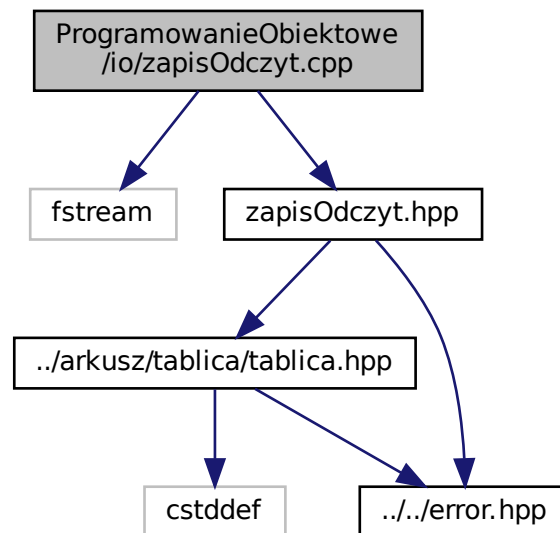
in	arkusz	pobiera arkusz celem jego wyświetlenia
----	--------	----------------------------------------

Here is the call graph for this function:



5.11 ProgramowanieObiektowe/io/zapisOdczyt.cpp File Reference

```
#include <fstream>
#include "zapisOdczyt.hpp"
Include dependency graph for zapisOdczyt.cpp:
```



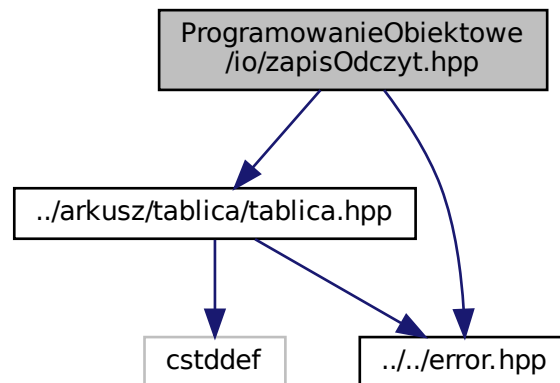
Functions

- Wyjatk `zapisPliku` (`Arkusz` arkusz, string nazwa)
- Wyjatk `wczytajPlik` (`Arkusz` *arkusz, string nazwa)

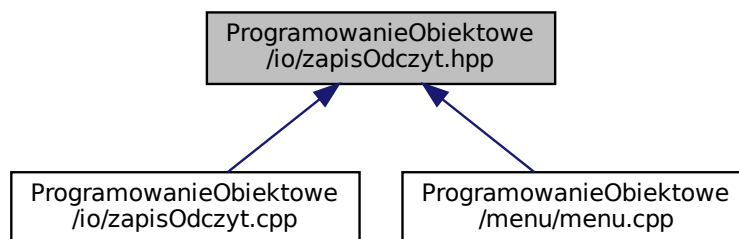
5.12 ProgramowanieObiektowe/io/zapisOdczyt.hpp File Reference

```
#include "../arkusz/tablica/tablica.hpp"
#include "../error.hpp"
```

Include dependency graph for zapisOdczyt.hpp:



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



Functions

- [Wyjątki zapisPliku](#) ([Arkusz](#) arkusz, std::string nazwa)
Funkcja zapisu do pliku.
- [Wyjątki wczytajPlik](#) ([Arkusz](#) *arkusz, std::string nazwa)
Funkcja wczytywania tablicy z pliku.

5.12.1 Function Documentation

5.12.1.1 wczytajPlik()

```
WyjatkI wczytajPlik (
    Arkusz * arkusz,
    std::string nazwa )
```

Funkcja wczytywania tablicy z pliku.

Funkcja wykonuje wczytanie arkusza z wybranego pliku, w przypadku niepowodzenia zwraca wartość różną od 0.

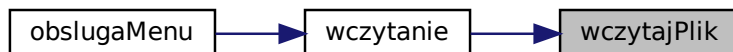
Parameters

in, out	arkusz	Arkusz do nadpisania wczytywaną tablicą
in	nazwa	Nazwa wczytywanego pliku

Returns

Zwraca stan funkcji "::BRAK - Funkcja wykonana bez błędnie", "1::PLIK_ACCESS - Niepoprawna nazwa bądź brak dostępu do pliku", "::PLIK_FORMAT - Niepoprawny format", "::PLIK_ROZMIAR - Niepoprawny rozmiar wczytywanej tablicy"

Here is the caller graph for this function:



5.12.1.2 zapisPliku()

```
WyjatkI zapisPliku (
    Arkusz arkusz,
    std::string nazwa )
```

Funkcja zapisu do pliku.

Funkcja wykonuje zapis do wybranego przez nas pliku, w przypadku błędu zwraca wartość różną od 0.

Parameters

in	arkusz	Arkusz przeznaczony do zapisu
in	nazwa	Nazwa zapisywanego pliku

Returns

Zwraca stan funkcji "::BRAK - Funkcja wykonana bez błędnie", "::PLIK_ACCESS - Niepoprawna nazwa bądź brak dostępu do pliku"

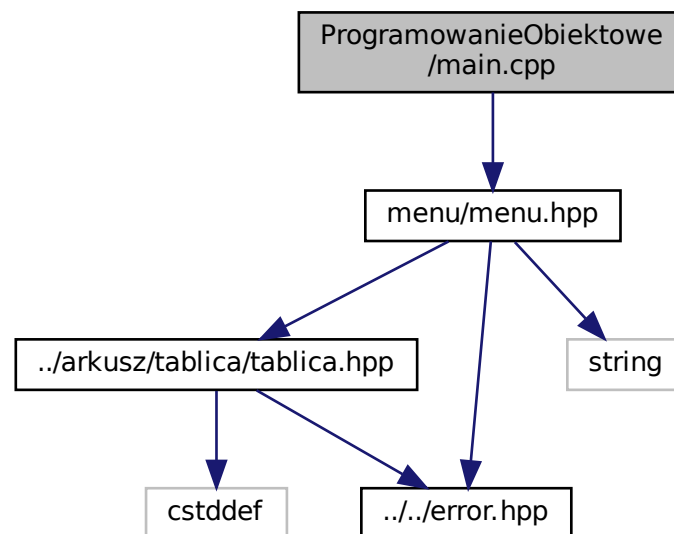
Here is the caller graph for this function:



5.13 ProgramowanieObiektowe/main.cpp File Reference

```
#include "menu/menu.hpp"
```

Include dependency graph for main.cpp:

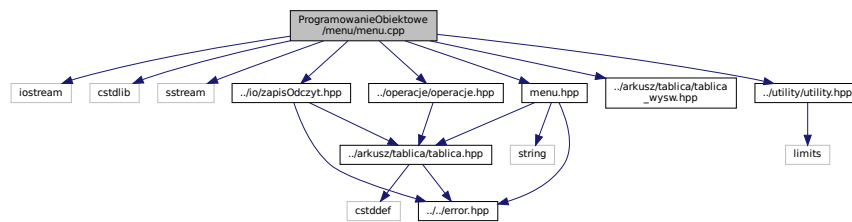
**Functions**

- `int main ()`

5.14 ProgramowanieObiektowe/menu/menu.cpp File Reference

```
#include <iostream>
#include <cstdlib>
#include <sstream>
#include "menu.hpp"
#include "../io/zapisOdczyt.hpp"
#include "../arkusz/tablica/tablica_wysw.hpp"
#include "../utility/utility.hpp"
#include "../operacje/operacje.hpp"
```

Include dependency graph for menu.cpp:



Functions

- void [generujMenu](#) ()
Funkcja tworząca menu.
- void [obsługaMenu](#) ()
Funkcja kontrolująca działanie programu.
- void [wczytanie](#) ([Arkusz](#) *arkusz)
Funkcja wczytywania arkusza Funkcja menu od wczytywania arkusza, ma za zadanie opakowanie funkcji IO wczytajPlik.
- void [zapis](#) ([Arkusz](#) arkusz)
Funkcja menu od zapisu.
- [Arkusz](#) [tworzArkusz](#) ()
Funkcja tworząca nową tablicę.
- void [rozszerzArkusz](#) ([Arkusz](#) *arkusz)
Funkcja modyfikująca rozmiar arkusza.
- void [parametry](#) ([Arkusz](#) arkusz)
Funkcja menu od wyboru względem czego wyznacza parametry.
- string [parametryWiersza](#) ([Arkusz](#) arkusz, int wiersz)
Funkcja od wyznaczania parametrów wiersza arkusza.
- string [parametryKolumny](#) ([Arkusz](#) arkusz, int kolumna)
Funkcja od wyznaczania parametrów kolumny arkusza.
- void [obsługaBledow](#) ([Wyjatk](#) wyjatek)
Funkcja zajmująca się "Przechwytywaniem wyjątków" i ich wypisywaniem Wypisuje napotkany "Wyjątek", Nie wypisze nic jeśli wyjątek ma wartość ::BRAK.

5.14.1 Function Documentation

5.14.1.1 generujMenu()

```
void generujMenu ( )
```

Funkcja tworząca menu.

Funkcja od tworzenia listy dostępnych pozycji menu. Here is the caller graph for this function:



5.14.1.2 obslugaBledow()

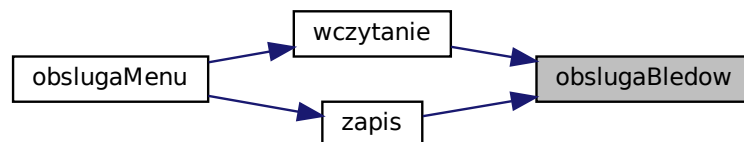
```
void obslugaBledow (
    Wyjatki wyjatek )
```

Funkcja zajmująca się "Przechwytywaniem wyjątków" i ich wypisywaniem Wypisuje napotkany "Wyjątek", Nie wypisze nic jeśli wyjątek ma wartość ::BRAK.

Parameters

in	<i>wyjatek</i>	Wyjątek który zostanie przechwycony
----	----------------	-------------------------------------

Here is the caller graph for this function:

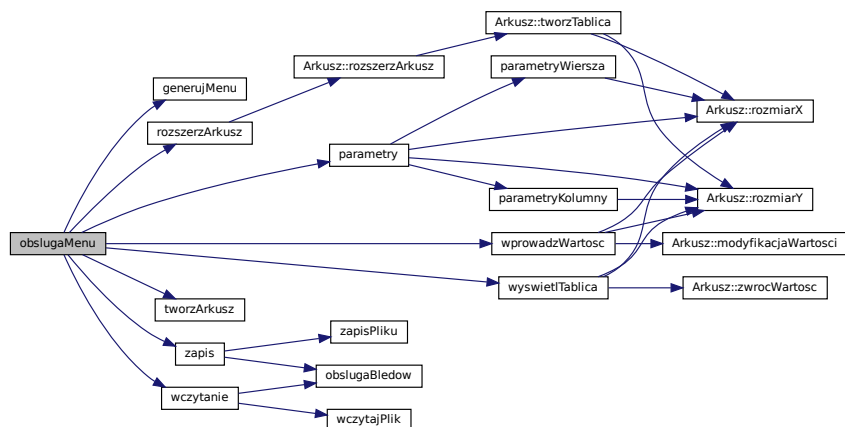


5.14.1.3 obslugaMenu()

```
void obslugaMenu ( )
```

Funkcja kontrolująca działanie programu.

Funkcja zajmująca się obsługą menu programu zarządza tym co będzie wywoływane Here is the call graph for this function:



5.14.1.4 parametry()

```
void parametry (
    Arkusz arkusz )
```

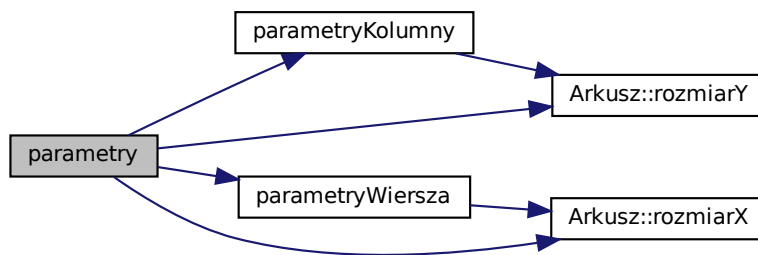
Funkcja menu od wyboru względem czego wyznacza parametry.

Funkcja menu od wyboru atrybutu tablicy (kolumny lub wiersza) która ma za wyświetlenie parametrów wybranego atrybutu.

Parameters

in	arkusz	Przekazywany arkusz do wykorzystania w funkcji wyboru parametrów
----	--------	------------------------------------------------------------------

Here is the call graph for this function:



Here is the caller graph for this function:



5.14.1.5 parametryKolumny()

```

string parametryKolumny (
    Arkusz arkusz,
    int kolumna )
  
```

Funkcja od wyznaczania parametrów kolumny arkusza.

Funkcja zwraca w postaci tekstowej wszystkie parametry wybranej kolumny

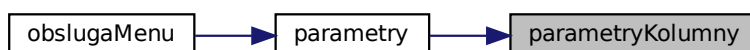
Parameters

in	<i>arkusz</i>	Przekazywany arkusz do wykorzystania w funkcji obliczania parametrów
in	<i>kolumna</i>	Kolumna względem której zostaną obliczone parametry

Here is the call graph for this function:



Here is the caller graph for this function:



5.14.1.6 parametryWiersza()

```
string parametryWiersza (
    Arkusz arkusz,
    int wiersz )
```

Funkcja od wyznaczania parametrów wiersza arkusza.

Funkcja zwraca w postaci tekstowej wszystkie parametry wybranego wiersza

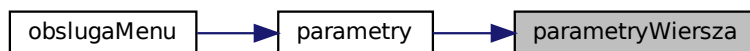
Parameters

in	<i>arkusz</i>	Przekazywany arkusz do wykorzystania w funkcji obliczania parametrów
in	<i>wiersz</i>	Wiersz względem której zostaną obliczone parametry

Here is the call graph for this function:



Here is the caller graph for this function:



5.14.1.7 `rozszerzArkusz()`

```
void rozszerzArkusz (
    Arkusz * arkusz )
```

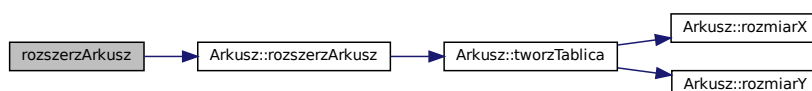
Funkcja modyfikująca rozmiar arkusza.

Interfejs umożliwiający modyfikację rozmiaru istniejącego arkusza.

Parameters

<code>in, out</code>	<code>arkusz</code>	<code>Arkusz</code> przeznaczony do modyfikacji rozmiaru
----------------------	---------------------	----------------------------------------------------------

Here is the call graph for this function:



Here is the caller graph for this function:



5.14.1.8 tworzArkusz()

```
Arkusz tworzArkusz ( )
```

Funkcja tworząca nową tablicę.

Funkcja zawierająca interfejs umożliwiający tworzenie nowego Arkusza z tablicą dwuwymiarową.

Returns

Nowy [Arkusz](#) do wykorzystywania w programie

Here is the caller graph for this function:



5.14.1.9 wczytanie()

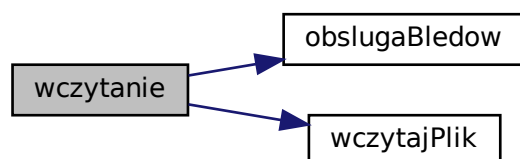
```
void wczytanie (  
    Arkusz * arkusz )
```

Funkcja wczytywania arkusza Funkcja menu od wczytywania arkusza, ma za zadanie opakowanie funkcji IO wczytajPlik.

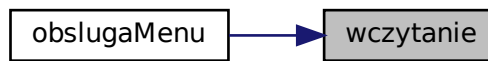
Parameters

in, out	arkusz	Arkusz do którego mogą być wczytane elementy
---------	--------	--------------------------------------------------------------

Here is the call graph for this function:



Here is the caller graph for this function:



5.14.1.10 zapis()

```
void zapis (
    Arkusz arkusz )
```

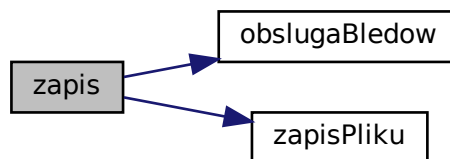
Funkcja menu od zapisu.

Funkcja menu od zapisu która ma za zadanie przetworzenie i opakowanie funkcji IO zapisPliku

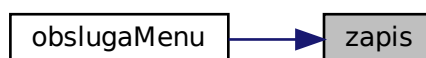
Parameters

in	arkusz	Przekazywany arkusz do wykorzystania w funkcji zapisującej do pliku
----	--------	---------------------------------------------------------------------

Here is the call graph for this function:



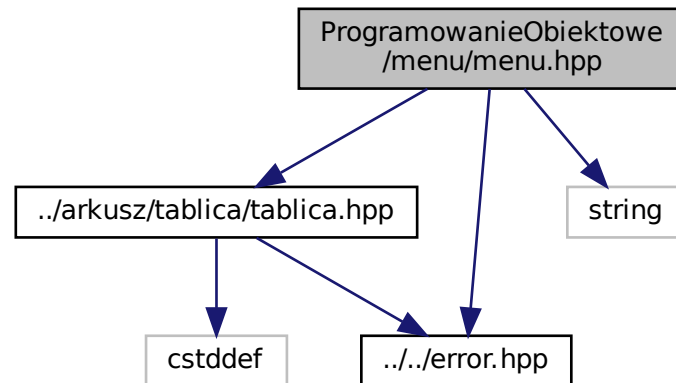
Here is the caller graph for this function:



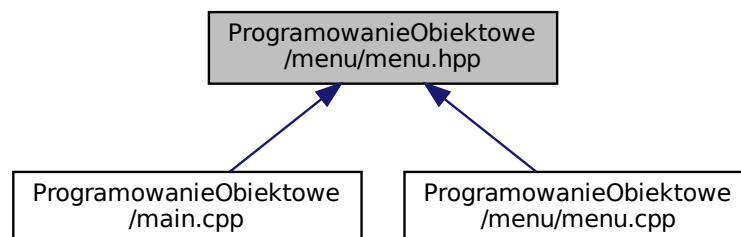
5.15 ProgramowanieObiektowe/menu/menu.hpp File Reference

```
#include "../arkusz/tablica/tablica.hpp"
#include <string>
#include "../error.hpp"
```

Include dependency graph for menu.hpp:



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



Functions

- void `obsługaMenu` ()
Funkcja kontrolująca działanie programu.
- void `generujMenu` ()
Funkcja tworząca menu.
- `Arkusz` `tworzArkusz` ()
Funkcja tworząca nową tablicę.
- void `rozszerzArkusz` (`Arkusz` *arkusz)

Funkcja modyfikująca rozmiar arkusza.

- void `wczytanie` (`Arkusz` *arkusz)

Funkcja wczytywania arkusza Funkcja menu od wczytywania arkusza, ma za zadanie opakowanie funkcji IO wczytajPlik.

- void `zapis` (`Arkusz` arkusz)

Funkcja menu od zapisu.

- void `parametry` (`Arkusz` arkusz)

Funkcja menu od wyboru względem czego wyznacza parametry.

- std::string `parametryWiersza` (`Arkusz` arkusz, int wiersz)

Funkcja od wyznaczania parametrów wiersza arkusza.

- std::string `parametryKolumny` (`Arkusz` arkusz, int kolumna)

Funkcja od wyznaczania parametrów kolumny arkusza.

- void `obsługaBledow` (`Wyjatk` wyjatek)

Funkcja zajmująca się "Przechwytywaniem wyjątków" i ich wypisywaniem Wypisuje napotkany "Wyjątek", Nie wypisze nic jeśli wyjątek ma wartość ::BRAK.

5.15.1 Function Documentation

5.15.1.1 generujMenu()

```
void generujMenu ( )
```

Funkcja tworząca menu.

Funkcja od tworzenia listy dostępnych pozycji menu. Here is the caller graph for this function:



5.15.1.2 obsługaBledow()

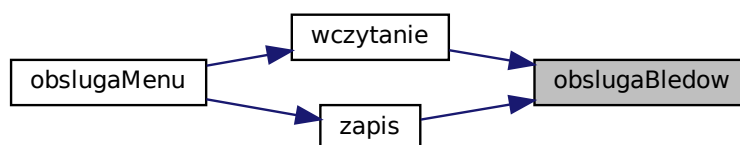
```
void obsługaBledow (
    Wyjatk wyjatek )
```

Funkcja zajmująca się "Przechwytywaniem wyjątków" i ich wypisywaniem Wypisuje napotkany "Wyjątek", Nie wypisze nic jeśli wyjątek ma wartość ::BRAK.

Parameters

in	wyjatek	Wyjątek który zostanie przechwycony
----	---------	-------------------------------------

Here is the caller graph for this function:

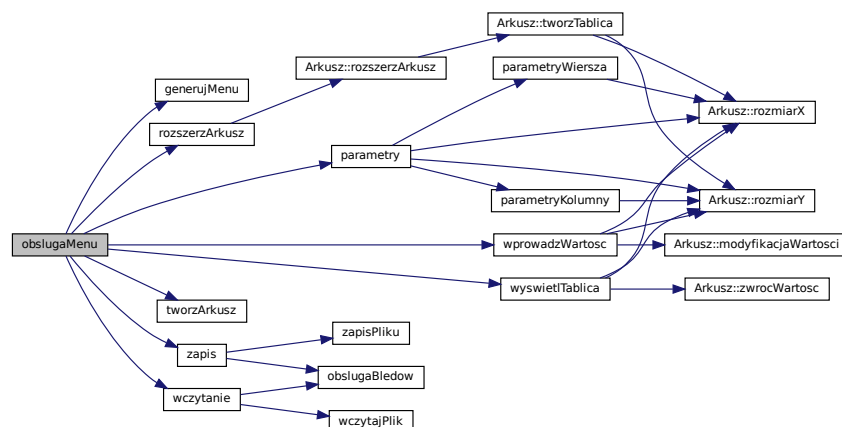


5.15.1.3 obslugaMenu()

```
void obslugaMenu ( )
```

Funkcja kontrolująca działanie programu.

Funkcja zajmująca się obsługą menu programu zarządza tym co będzie wywoływane Here is the call graph for this function:



5.15.1.4 parametry()

```
void parametry (
    Arkusz arkusz )
```

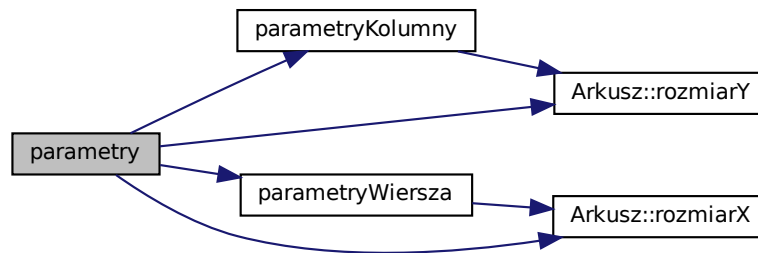
Funkcja menu od wyboru względem czego wyznacza parametry.

Funkcja menu od wyboru atrybutu tablicy (kolumny lub wiersza) która ma za wyświetlenie parametrów wybranego atrybutu.

Parameters

in	<i>arkusz</i>	Przekazywany arkusz do wykorzystania w funkcji wyboru parametrów
----	---------------	------------------------------------------------------------------

Here is the call graph for this function:



Here is the caller graph for this function:



5.15.1.5 parametryKolumny()

```

std::string parametryKolumny (
    Arkusz arkusz,
    int kolumna )

```

Funkcja od wyznaczania parametrów kolumny arkusza.

Funkcja zwraca w postaci tekstowej wszystkie parametry wybranej kolumny

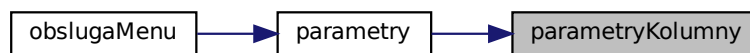
Parameters

in	<i>arkusz</i>	Przekazywany arkusz do wykorzystania w funkcji obliczania parametrów
in	<i>kolumna</i>	Kolumna względem której zostaną obliczone parametry

Here is the call graph for this function:



Here is the caller graph for this function:



5.15.1.6 parametryWiersza()

```
std::string parametryWiersza (
    Arkusz arkusz,
    int wiersz )
```

Funkcja od wyznaczania parametrów wiersza arkusza.

Funkcja zwraca w postaci tekstowej wszystkie parametry wybranego wiersza

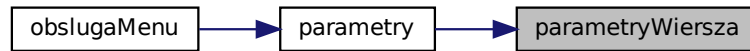
Parameters

in	<i>arkusz</i>	Przekazywany arkusz do wykorzystania w funkcji obliczania parametrów
in	<i>wiersz</i>	Wiersz względem której zostaną obliczone parametry

Here is the call graph for this function:



Here is the caller graph for this function:



5.15.1.7 rozszerzArkusz()

```
void rozszerzArkusz (
    Arkusz * arkusz )
```

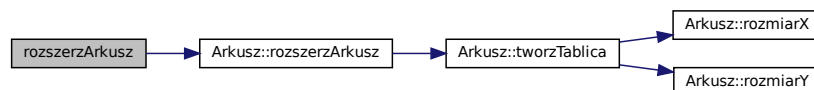
Funkcja modyfikująca rozmiar arkusza.

Interfejs umożliwiający modyfikację rozmiaru istniejącego arkusza.

Parameters

in, out	arkusz	Arkusz przeznaczony do modyfikacji rozmiaru
---------	--------	---------------------------------------------

Here is the call graph for this function:



Here is the caller graph for this function:



5.15.1.8 tworzArkusz()

```
Arkusz tworzArkusz ( )
```

Funkcja tworząca nową tablicę.

Funkcja zawierająca interfejs umożliwiający tworzenie nowego Arkusza z tablicą dwuwymiarową.

Returns

Nowy [Arkusz](#) do wykorzystywania w programie

Here is the caller graph for this function:



5.15.1.9 wczytanie()

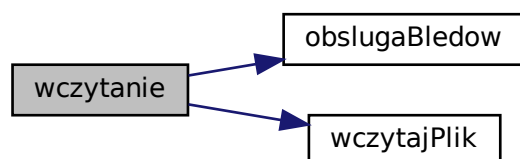
```
void wczytanie (  
    Arkusz * arkusz )
```

Funkcja wczytywania arkusza Funkcja menu od wczytywania arkusza, ma za zadanie opakowanie funkcji IO wczytajPlik.

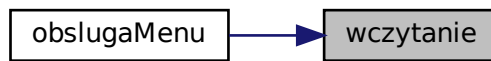
Parameters

<code>in, out</code>	<code>arkusz</code>	Arkusz do którego mogą być wczytane elementy
----------------------	---------------------	--------------------------------------------------------------

Here is the call graph for this function:



Here is the caller graph for this function:



5.15.1.10 zapis()

```
void zapis (  
    Arkusz arkusz )
```

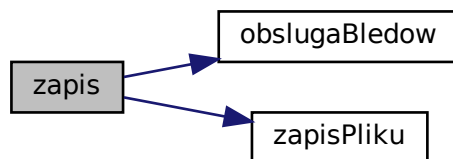
Funkcja menu od zapisu.

Funkcja menu od zapisu która ma za zadanie przetworzenie i opakowanie funkcji IO zapisPliku

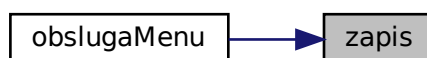
Parameters

in	arkusz	Przekazywany arkusz do wykorzystania w funkcji zapisującej do pliku
----	--------	---------------------------------------------------------------------

Here is the call graph for this function:



Here is the caller graph for this function:



Index

- Arkusz, [7](#)
 - Arkusz, [8](#)
 - modyfikacjaWartosci, [8](#)
 - rozmiarX, [9](#)
 - rozmiarY, [9](#)
 - rozszerzArkusz, [10](#)
 - tworzTablica, [11](#)
 - zwrocWartosc, [12](#)
- Cell, [13](#)
 - Cell, [14](#)
 - getValue, [14](#)
 - isCalcualteable, [14](#)
 - operator+, [15](#)
 - toString, [16](#)
- error.hpp
 - PLIK_ACCESS, [32](#)
 - PLIK_FORMAT, [32](#)
 - PLIK_ROZMIAR, [32](#)
 - TABLICA_SIZE, [32](#)
 - TABLICA_ZAKR, [32](#)
 - Wyjatki, [32](#)
- generujMenu
 - menu.cpp, [37](#)
 - menu.hpp, [46](#)
- getValue
 - Cell, [14](#)
 - intCell, [18](#)
- intCell, [17](#)
 - getValue, [18](#)
 - intCell, [18](#)
 - setValue, [18](#)
 - toString, [19](#)
- isCalcualteable
 - Cell, [14](#)
- menu.cpp
 - generujMenu, [37](#)
 - obsługaBledow, [38](#)
 - obsługaMenu, [38](#)
 - parametry, [39](#)
 - parametryKolumny, [40](#)
 - parametryWiersza, [41](#)
 - rozszerzArkusz, [42](#)
 - tworzArkusz, [42](#)
 - wczytanie, [43](#)
 - zapis, [44](#)
- menu.hpp
 - generujMenu, [46](#)
 - obsługaBledow, [46](#)
 - obsługaMenu, [47](#)
 - parametry, [47](#)
 - parametryKolumny, [48](#)
 - parametryWiersza, [49](#)
 - rozszerzArkusz, [50](#)
 - tworzArkusz, [50](#)
 - wczytanie, [51](#)
 - zapis, [52](#)
- modyfikacjaWartosci
 - Arkusz, [8](#)
- obsługaBledow
 - menu.cpp, [38](#)
 - menu.hpp, [46](#)
- obsługaMenu
 - menu.cpp, [38](#)
 - menu.hpp, [47](#)
- operator+
 - Cell, [15](#)
- parametry
 - menu.cpp, [39](#)
 - menu.hpp, [47](#)
- parametryKolumny
 - menu.cpp, [40](#)
 - menu.hpp, [48](#)
- parametryWiersza
 - menu.cpp, [41](#)
 - menu.hpp, [49](#)
- PLIK_ACCESS
 - error.hpp, [32](#)
- PLIK_FORMAT
 - error.hpp, [32](#)
- PLIK_ROZMIAR
 - error.hpp, [32](#)
- ProgramowanieObiektowe/arkusz/komorka/cell.cpp, [23](#)
- ProgramowanieObiektowe/arkusz/komorka/cell.hpp, [23](#)
- ProgramowanieObiektowe/arkusz/komorka/intCell.cpp, [24](#)
- ProgramowanieObiektowe/arkusz/komorka/intCell.hpp, [25](#)
- ProgramowanieObiektowe/arkusz/komorka/stringCell.cpp, [26](#)
- ProgramowanieObiektowe/arkusz/komorka/stringCell.hpp, [27](#)
- ProgramowanieObiektowe/arkusz/tablica/tablica.cpp, [28](#)
- ProgramowanieObiektowe/arkusz/tablica/tablica.hpp, [28](#)

ProgramowanieObiektowe/arkusz/tablica/tablica_wysw.cpp, 29
wczytajPlik, 34
zapisPliku, 35
zapisPliku
zapisOdczyt.hpp, 35
zwrocWartosc
Arkusz, 12

ProgramowanieObiektowe/error.hpp, 32
ProgramowanieObiektowe/io/zapisOdczyt.cpp, 33
ProgramowanieObiektowe/io/zapisOdczyt.hpp, 33
ProgramowanieObiektowe/main.cpp, 36
ProgramowanieObiektowe/menu/menu.cpp, 37
ProgramowanieObiektowe/menu/menu.hpp, 45

rozmiarX
Arkusz, 9
rozmiarY
Arkusz, 9
rozszerzArkusz
Arkusz, 10
menu.cpp, 42
menu.hpp, 50

setValue
intCell, 18
stringCell, 21
stringCell, 19
setValue, 21
stringCell, 20
toString, 21

TABLICA_SIZE
error.hpp, 32
tablica_wysw.cpp
wprowadzWartosc, 30
wyswietlTablica, 31
TABLICA_ZAKR
error.hpp, 32
toString
Cell, 16
intCell, 19
stringCell, 21
tworzArkusz
menu.cpp, 42
menu.hpp, 50
tworzTablica
Arkusz, 11

wczytajPlik
zapisOdczyt.hpp, 34
wczytanie
menu.cpp, 43
menu.hpp, 51
wprowadzWartosc
tablica_wysw.cpp, 30
Wyjatki
error.hpp, 32
wyswietlTablica
tablica_wysw.cpp, 31

zapis
menu.cpp, 44
menu.hpp, 52
zapisOdczyt.hpp