

Arkusz kalkulacyjny

Generated by Doxygen 1.9.1



<b>1 Hierarchical Index</b>	<b>1</b>
1.1 Class Hierarchy	1
<b>2 Class Index</b>	<b>3</b>
2.1 Class List	3
<b>3 File Index</b>	<b>5</b>
3.1 File List	5
<b>4 Class Documentation</b>	<b>7</b>
4.1 AbstractCell Struct Reference	7
4.1.1 Detailed Description	7
4.2 Arkusz Class Reference	7
4.2.1 Detailed Description	8
4.2.2 Constructor & Destructor Documentation	8
4.2.2.1 Arkusz()	8
4.2.3 Member Function Documentation	9
4.2.3.1 modyfikacjaWartosci()	9
4.2.3.2 rozmiarX()	10
4.2.3.3 rozmiarY()	10
4.2.3.4 rozszerzArkusz()	11
4.2.3.5 tworzTablica()	12
4.2.3.6 zwrocWartosc()	13
4.3 Cell< T > Class Template Reference	14
4.3.1 Detailed Description	15
4.3.2 Constructor & Destructor Documentation	15
4.3.2.1 Cell()	15
4.3.3 Member Function Documentation	15
4.3.3.1 Calculateable()	15
4.3.3.2 getString()	16
4.3.3.3 getValue()	16
4.3.3.4 setValue()	16
<b>5 File Documentation</b>	<b>19</b>
5.1 ProgramowanieObiektowe/arkusz/komorka/cell.cpp File Reference	19
5.2 ProgramowanieObiektowe/arkusz/komorka/cell.hpp File Reference	19
5.3 ProgramowanieObiektowe/arkusz/tablica/tablica.cpp File Reference	20
5.4 ProgramowanieObiektowe/arkusz/tablica/tablica.hpp File Reference	21
5.4.1 Function Documentation	22
5.4.1.1 wprowadzWartosc()	22
5.4.1.2 wyswietlTablica()	23
5.5 ProgramowanieObiektowe/arkusz/tablica/tablica_wysw.cpp File Reference	24
5.5.1 Function Documentation	25
5.5.1.1 wprowadzWartosc()	25

5.5.1.2	wyswietlTablica()	26
5.6	ProgramowanieObiektowe/error.hpp File Reference	27
5.6.1	Enumeration Type Documentation	28
5.6.1.1	Wyjatk	28
5.7	ProgramowanieObiektowe/io/zapisOdczyt.cpp File Reference	28
5.8	ProgramowanieObiektowe/io/zapisOdczyt.hpp File Reference	29
5.8.1	Function Documentation	30
5.8.1.1	wczytajPlik()	30
5.8.1.2	zapisPliku()	30
5.9	ProgramowanieObiektowe/main.cpp File Reference	31
5.10	ProgramowanieObiektowe/menu/menu.cpp File Reference	32
5.10.1	Function Documentation	33
5.10.1.1	generujMenu()	33
5.10.1.2	obsługaBledow()	33
5.10.1.3	obsługaMenu()	34
5.10.1.4	parametry()	34
5.10.1.5	parametryKolumny()	35
5.10.1.6	parametryWiersza()	36
5.10.1.7	rozszerzArkusz()	37
5.10.1.8	tworzArkusz()	38
5.10.1.9	wczytanie()	38
5.10.1.10	zapis()	39
5.11	ProgramowanieObiektowe/menu/menu.hpp File Reference	40
5.11.1	Function Documentation	41
5.11.1.1	generujMenu()	41
5.11.1.2	obsługaBledow()	41
5.11.1.3	obsługaMenu()	42
5.11.1.4	parametry()	42
5.11.1.5	parametryKolumny()	43
5.11.1.6	parametryWiersza()	44
5.11.1.7	rozszerzArkusz()	45
5.11.1.8	tworzArkusz()	46
5.11.1.9	wczytanie()	46
5.11.1.10	zapis()	47

<b>Index</b>	<b>49</b>
--------------	-----------

# Chapter 1

## Hierarchical Index

### 1.1 Class Hierarchy

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

AbstractCell . . . . .	7
Cell< T > . . . . .	14
Arkusz . . . . .	7



## Chapter 2

# Class Index

### 2.1 Class List

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

<a href="#">AbstractCell</a>	Definicja pustej klasy do generalizacji <a href="#">Cell</a> 'a Pusta klasa utworzona w celu generalizacji klasy	
<a href="#">Cell</a>	.....	7
<a href="#">Arkusz</a>	Struktura <a href="#">Arkusz</a> Struktura <a href="#">Arkusz</a> przechowująca tablicę i jej rozmiar	7
<a href="#">Cell&lt; T &gt;</a>	Szablon klasy komórki Klasa komórka przechowująca pewną wartość określonego typu	14





## Chapter 3

# File Index

### 3.1 File List

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

ProgramowanieObiektowe/ <a href="#">error.hpp</a> . . . . .	27
ProgramowanieObiektowe/ <a href="#">main.cpp</a> . . . . .	31
ProgramowanieObiektowe/arkusz/komorka/ <a href="#">cell.cpp</a> . . . . .	19
ProgramowanieObiektowe/arkusz/komorka/ <a href="#">cell.hpp</a> . . . . .	19
ProgramowanieObiektowe/arkusz/tablica/ <a href="#">tablica.cpp</a> . . . . .	20
ProgramowanieObiektowe/arkusz/tablica/ <a href="#">tablica.hpp</a> . . . . .	21
ProgramowanieObiektowe/arkusz/tablica/ <a href="#">tablica_wysw.cpp</a> . . . . .	24
ProgramowanieObiektowe/io/ <a href="#">zapisOdczyt.cpp</a> . . . . .	28
ProgramowanieObiektowe/io/ <a href="#">zapisOdczyt.hpp</a> . . . . .	29
ProgramowanieObiektowe/menu/ <a href="#">menu.cpp</a> . . . . .	32
ProgramowanieObiektowe/menu/ <a href="#">menu.hpp</a> . . . . .	40
ProgramowanieObiektowe/operacje/ <b>operacje.hpp</b> . . . . .	??
ProgramowanieObiektowe/utility/ <b>utility.hpp</b> . . . . .	??



## Chapter 4

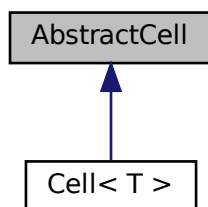
# Class Documentation

### 4.1 AbstractCell Struct Reference

definicja pustej klasy do generalizacji [Cell](#)'a Pusta klasa utworzona w celu generalizacji klasy [Cell](#)

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

Inheritance diagram for AbstractCell:



#### 4.1.1 Detailed Description

definicja pustej klasy do generalizacji [Cell](#)'a Pusta klasa utworzona w celu generalizacji klasy [Cell](#)

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

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

### 4.2 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, int wartosc)  
*Wstawia określoną wartość do komórki tablicy w arkuszu.*
- [Wyjątki rozszerzArkusz](#) (size\_t x, size\_t y)  
*Modyfikacja rozmiaru tablicy.*
- int [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 tworTablica](#) (size\_t kolumny, size\_t wiersze)  
*Tworzy nową dwuwymiarową tablice.*

### 4.2.1 Detailed Description

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

### 4.2.2 Constructor & Destructor Documentation

#### 4.2.2.1 Arkusz()

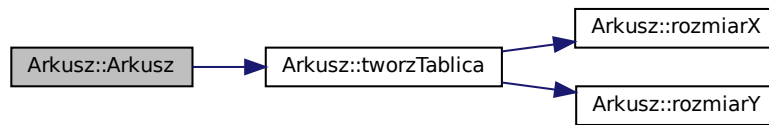
```
Arkusz::Arkusz (
    size_t x,
    size_t y )
```

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:



## 4.2.3 Member Function Documentation

### 4.2.3.1 modyfikacjaWartosci()

```
Wyjatk: Arkusz::modyfikacjaWartosci (
    size_t x,
    size_t y,
    int 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

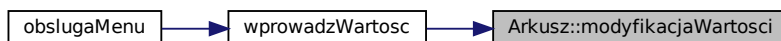
#### Returns

Kod błędu, BRAK - Brak błędu, [TABLICA\\_ZAKR](#) - indeks spoza zakresu tablicy

Here is the call graph for this function:



Here is the caller graph for this function:



#### 4.2.3.2 rozmiarX()

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

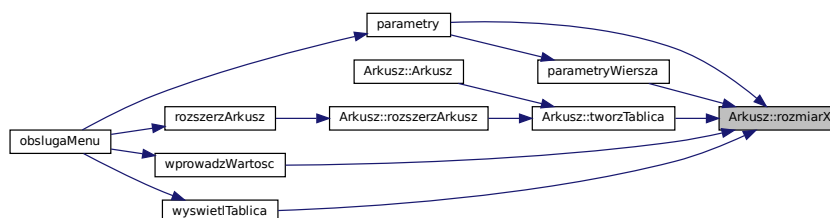
Getter ilości kolumn.

Funkcja zwraca ilość kolumn danego arkusza

##### Returns

Ilość kolumn arkusza

Here is the caller graph for this function:



#### 4.2.3.3 rozmiarY()

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

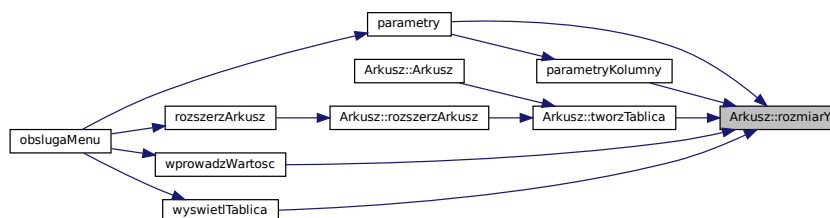
Getter ilości wierszy.

Funkcja zwraca ilość wierszy danego arkusza

## Returns

Ilość wierszy arkusza

Here is the caller graph for this function:



## 4.2.3.4 rozszerzArkusz()

```
Wyjatk: 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)

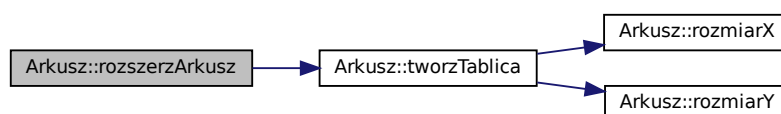
## Parameters

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

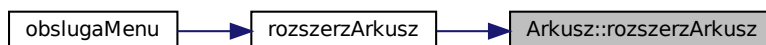
## Returns

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:



#### 4.2.3.5 tworzTablica()

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

Tworzy nową dwuwymiarową tablicę.

Funkcja generująca tablicę o określonym rozmiarze

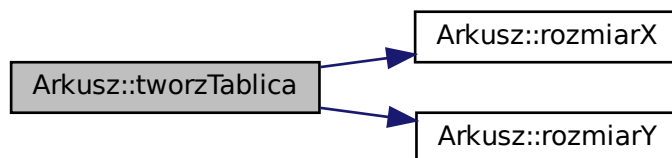
##### Parameters

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

##### Returns

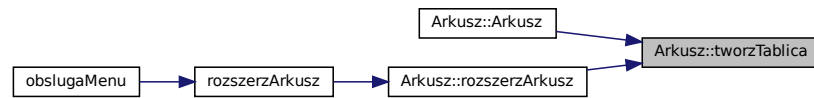
Nową tablicę dwuwymiarową o wyznaczonych rozmiarach

Here is the call graph for this function:





Here is the caller graph for this function:



#### 4.2.3.6 zwrocWartosc()

```
int 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

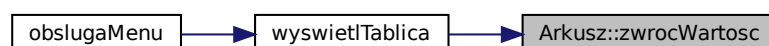
##### Returns

Wartość komórki w przeciwnym wypadku najmniejsza możliwa wartość

Here is the call graph for this function:



Here is the caller graph for this function:



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

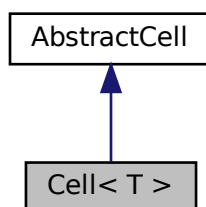
- [ProgramowanieObiektowe/arkusz/tablica/tablica.hpp](#)
- [ProgramowanieObiektowe/arkusz/tablica/tablica.cpp](#)

### 4.3 Cell< T > Class Template Reference

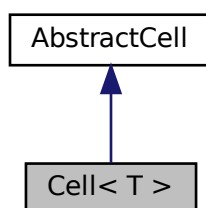
Szablon klasy komórki Klasa komórka przechowująca pewną wartość określonego typu.

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

Inheritance diagram for Cell< T >:



Collaboration diagram for Cell< T >:



#### Public Member Functions

- [Cell](#) (T val)  
*Konstruktor komórki Konstruktor komórki nadający jej wartość początkową określonego typu.*
- [int](#) [getValue](#) ()  
*Metoda od pobierania wartości matematycznej Zwraca wartość typu Int.*
- [std::string](#) [getString](#) ()

Metoda od pobierania wartości ciągu znaków Zwraca wartość typu String - ciąg znaków do wykorzystywania przykładowo w wypisywaniu Wartości innego typu niż string są przepuszczane przez stringstream celem "rzutowania" na String.

- void `setValue` (T val)

Metoda od ustawienia wartości komórki Metoda ustawia wartość komórki określonego typu.

- bool `Calculateable` ()

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

- bool `Calculateable` ()
- int `getValue` ()
- std::string `getString` ()
- bool `Calculateable` ()
- int `getValue` ()
- std::string `getString` ()

### 4.3.1 Detailed Description

```
template<class T>
class Cell< T >
```

Szablon klasy komórki Klasa komórka przechowująca pewną wartość określonego typu.

### 4.3.2 Constructor & Destructor Documentation

#### 4.3.2.1 Cell()

```
template<class T >
Cell< T >::Cell (
    T val )
```

Konstruktor komórki Konstruktor komórki nadający jej wartość początkową określonego typu.

##### Parameters

in	val	Wartość początkowa komórki
----	-----	----------------------------

### 4.3.3 Member Function Documentation

#### 4.3.3.1 Calculateable()

```
template<class T >
bool Cell< T >::Calculateable
```

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.

#### 4.3.3.2 getString()

```
template<class T >
std::string Cell< T >::getString
```

Metoda od pobierania wartości ciągu znaków Zwraca wartość typu String - ciąg znaków do wykorzystywania przykładowo w wypisywaniu Wartości innego typu niż string są przepuszczane przez stringstream celem "rzutowania" na String.

#### Returns

Zwraca wartość typu string

#### 4.3.3.3 getValue()

```
template<class T >
int Cell< T >::getValue
```

Metoda od pobierania wartości matematycznej Zwraca wartość typu Int.

#### Returns

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

Here is the caller graph for this function:



#### 4.3.3.4 setValue()

```
template<class T >
void Cell< T >::setValue (
    T val )
```

Metoda od ustawienia wartości komórki Metoda ustawia wartość komórki określonego typu.

## Parameters

<code>in</code>	<code>val</code>	Nowa wartość komórki
-----------------	------------------	----------------------

Here is the caller graph for this function:



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

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



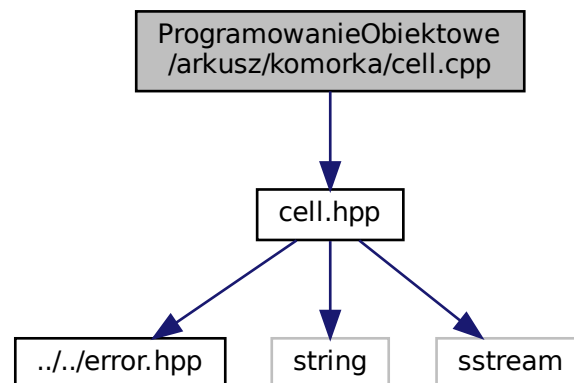
## Chapter 5

# File Documentation

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

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

Include dependency graph for cell.cpp:



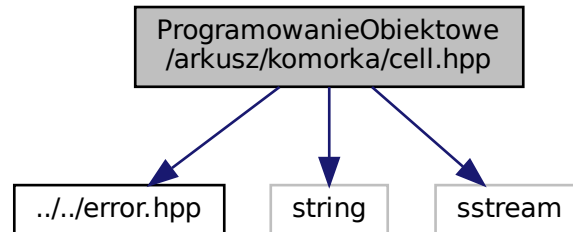
### 5.2 ProgramowanieObiektowe/arkusz/komorka/cell.hpp File Reference

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

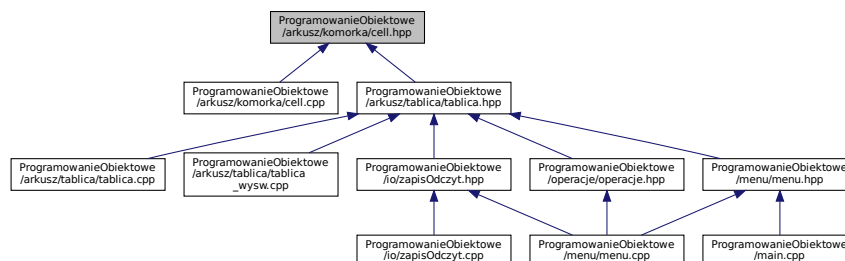
```
#include <string>
```

```
#include <sstream>
```

Include dependency graph for cell.hpp:



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



## Classes

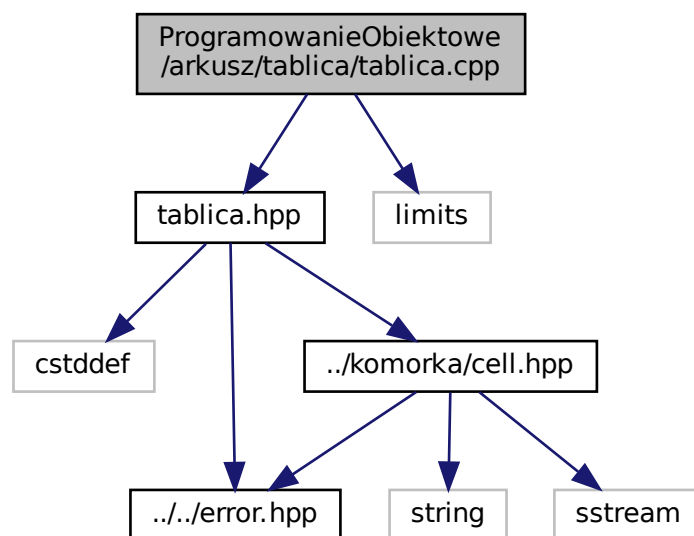
- struct [AbstractCell](#)  
definicja pustej klasy do generalizacji [Cell](#)'a Pusta klasa utworzona w celu generalizacji klasy [Cell](#)
- class [Cell< T >](#)  
Szablon klasy komórki Klasa komórka przechowująca pewną wartość określonego typu.

## 5.3 ProgramowanieObiektowe/arkusz/tablica/tablica.cpp File Reference

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



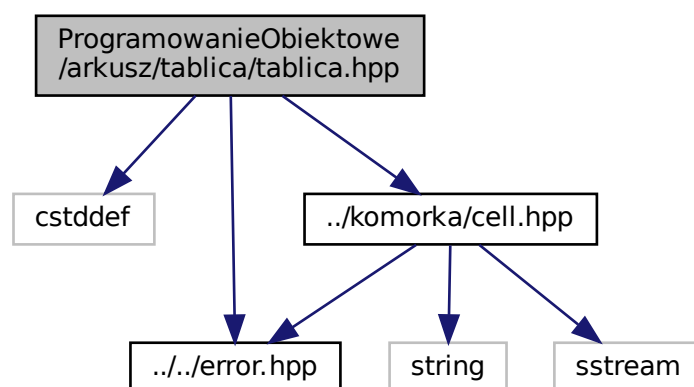
Include dependency graph for tablica.cpp:



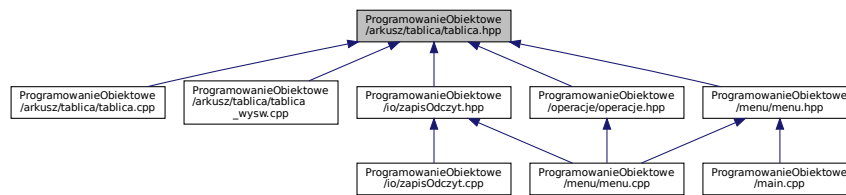
## 5.4 ProgramowanieObiektowe/arkusz/tablica/tablica.hpp File Reference

```
#include <cstddef>
#include "../error.hpp"
#include "../komorka/cell.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 [Cell](#)< int > [Komorka](#)

Definicja/alias typu [int](#) jako typ określający komórkę

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

Definicja/alias typu [Komorka](#)\*\* jako [Tablica](#) komórek.

## Functions

- void [wyswietlTablica](#) ([Arkusz](#) arkusz)

Generuje podgląd tablicy.

- void [wprowadzWartosc](#) ([Arkusz](#) \*arkusz)

Interfejs użytkownika wprowadzania nowej wartości.

## 5.4.1 Function Documentation

### 5.4.1.1 wprowadzWartosc()

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

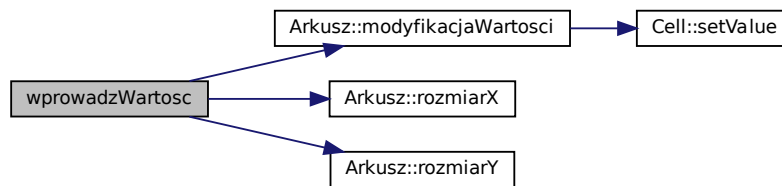
Interfejs użytkownika wprowadzania nowej wartości.

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

#### Parameters

in, out	<a href="#">arkusz</a>	arkusz która zostaje zmodyfikowany
---------	------------------------	------------------------------------

Here is the call graph for this function:



Here is the caller graph for this function:



#### 5.4.1.2 wyswietlTablica()

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

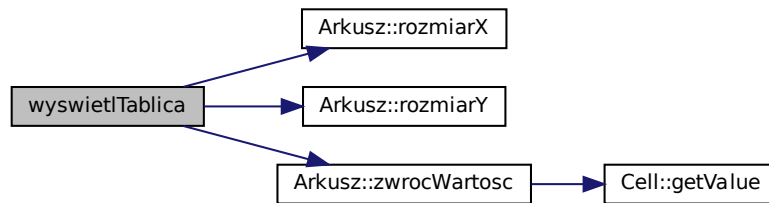
Generuje podgląd tablicy.

Generuje tablicę na konsoli wyświetlając wszystkie zawarte w niej komórki

##### Parameters

in	<i>arkusz</i>	pobiera arkusz celem jego wyświetlenia
----	---------------	--

Here is the call graph for this function:



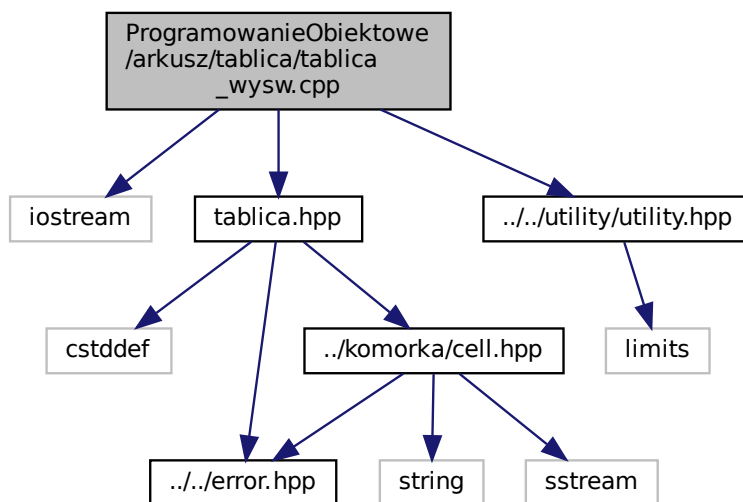
Here is the caller graph for this function:



## 5.5 ProgramowanieObiektowe/arkusz/tablica/tablica\_wysw.cpp File Reference

```
#include <iostream>
#include "tablica.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.*

### 5.5.1 Function Documentation

#### 5.5.1.1 wprowadzWartosc()

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

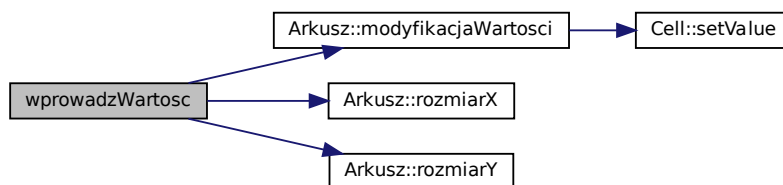
Interfejs użytkownika wprowadzania nowej wartości.

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

#### Parameters

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

Here is the call graph for this function:



Here is the caller graph for this function:



### 5.5.1.2 wyświetlTablica()

```
void wyświetlTablica (
    Arkusz arkusz )
```

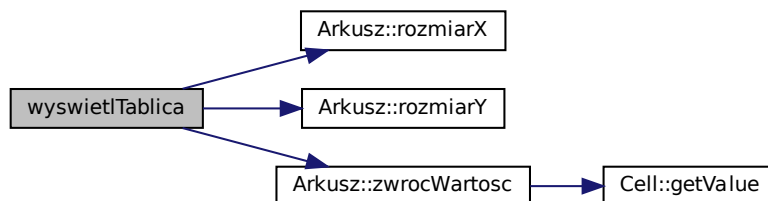
Generuje podgląd tablicy.

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:

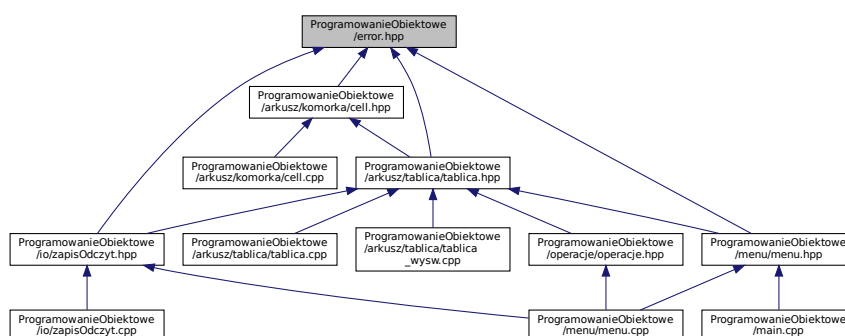


Here is the caller graph for this function:



## 5.6 ProgramowanieObiektowe/error.hpp File Reference

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



## Enumerations

- enum class `Wyjatki` : unsigned int {  
**BRAK** = 0 , **TABLICA\_SIZE** = 1 , **TABLICA\_ZAKR** = 2 , **PLIK\_ACCESS** = 10 ,  
**PLIK\_FORMAT** = 11 , **PLIK\_ROZMIAR** = 12 }

*Wyjątki występujące w programie Typ wyliczeniowy który zawiera wszystkie występujące wyjątki.*

## 5.6.1 Enumeration Type Documentation

### 5.6.1.1 Wyjątki

```
enum Wyjatki : unsigned int [strong]
```

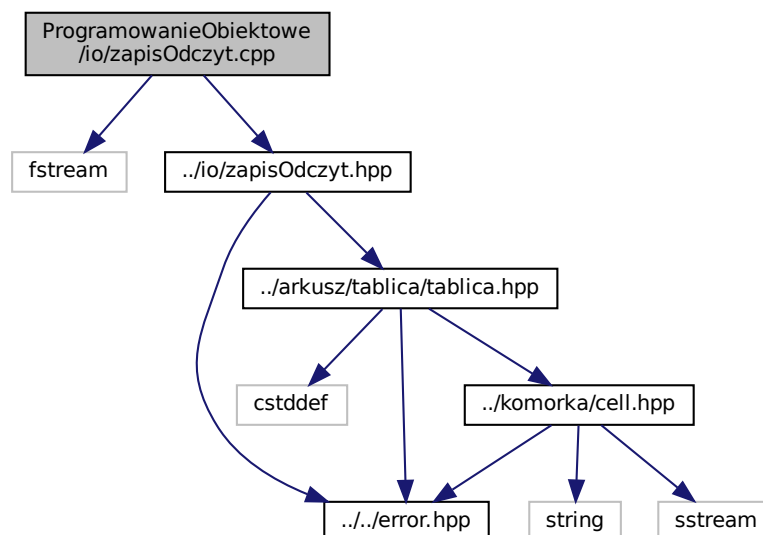
Wyjątki występujące w programie Typ wyczeniowy który zawiera wszystkie występujące wyjątki.

Enumerator

TABLICA_SIZE	Brak błędów.
TABLICA_ZAKR	Próba dostępu do elementu poza zakresem tablicy.
PLIK_ACCESS	Próba utworzenia tablicy o niepoprawnym rozmiarze.
PLIK_FORMAT	Niepoprawna nazwa lub brak dostępu do pliku.
PLIK_ROZMIAR	Niepoprawny format wczytywanego pliku.

## 5.7 ProgramowanieObiektowe/io/zapisOdczyt.cpp File Reference

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



## Functions

- [Wyjątki zapisPliku](#) ([Arkusz](#) arkusz, string nazwa)
- [Wyjątki wczytajPlik](#) ([Arkusz](#) \*arkusz, string nazwa)

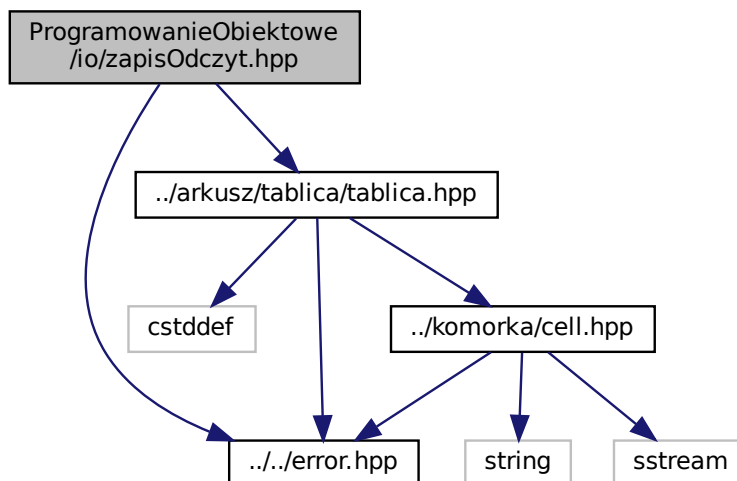


## 5.8 ProgramowanieObiektowe/io/zapisOdczyt.hpp File Reference

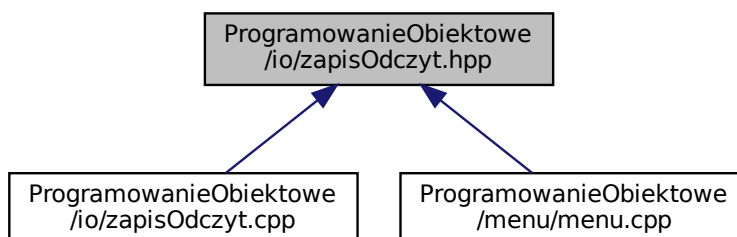
```
#include "../arkusz/tablica/tablica.hpp"
```

```
#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.*

## 5.8.1 Function Documentation

### 5.8.1.1 wczytajPlik()

```
Wyjatk 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.

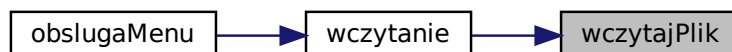
#### Parameters

in, out	<i>arkusz</i>	Arkusz do nadpisania wczytywaną tablicą
in	<i>nazwa</i>	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:



### 5.8.1.2 zapisPliku()

```
Wyjatk 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.

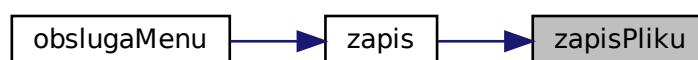
## Parameters

in	<i>arkusz</i>	Arkusz przeznaczony do zapisu
in	<i>nazwa</i>	Nazwa zapisywanego pliku

## Returns

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

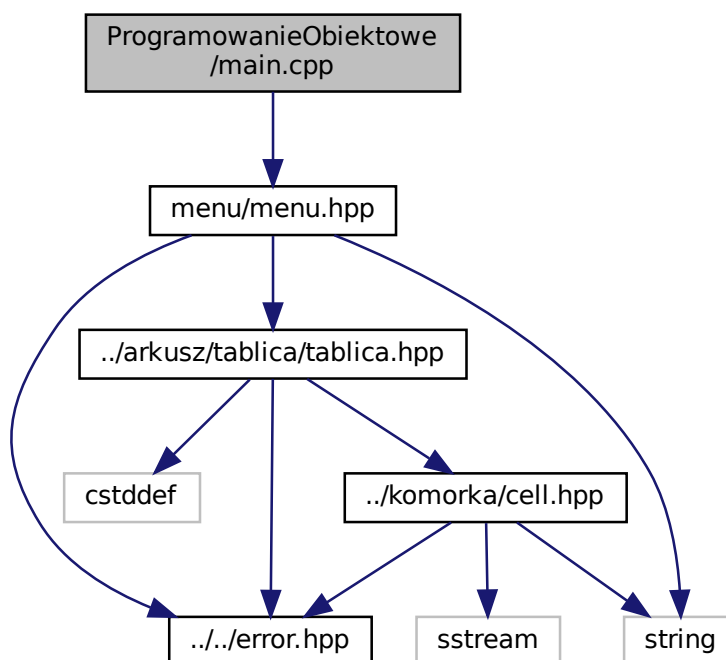
Here is the caller graph for this function:



## 5.9 ProgramowanieObiektowe/main.cpp File Reference

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

Include dependency graph for main.cpp:





## 5.10.1 Function Documentation

### 5.10.1.1 generujMenu()

```
void generujMenu ( )
```

Funkcja tworząca menu.

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



### 5.10.1.2 obslugaBledow()

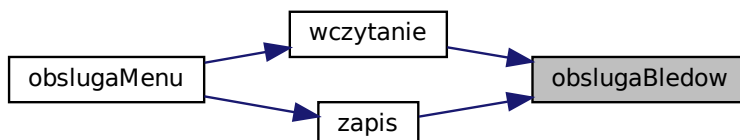
```
void obslugaBledow (
    Wyjatki wyjatek )
```

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:

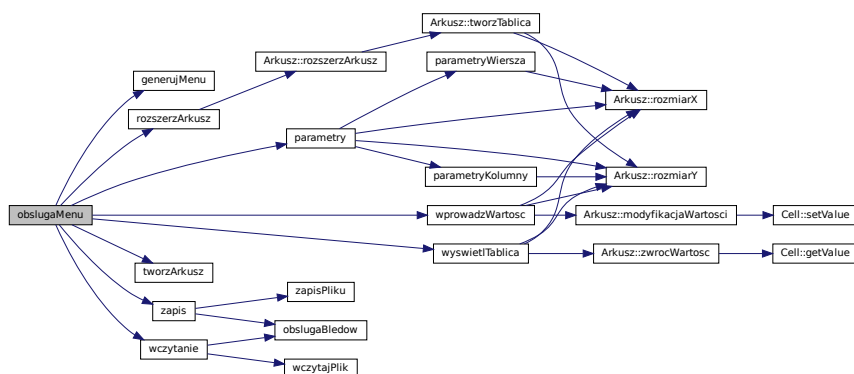


### 5.10.1.3 obslugaMenu()

```
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:



### 5.10.1.4 parametry()

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

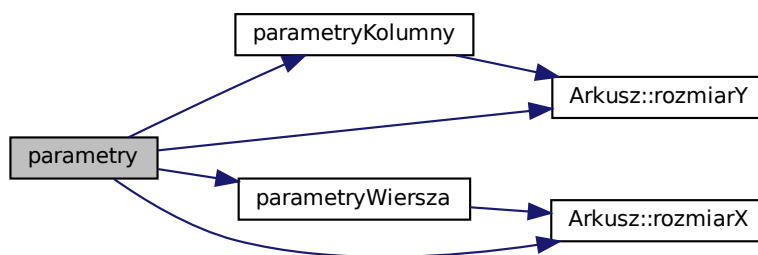
Funkcja menu od wyboru względem czego wyznacza parametry.

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:



#### 5.10.1.5 parametryKolumny()

```

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

Funkcja od wyznaczania parametrów kolumny arkusza.

Funkcja zwraca w postaci tekstowej wszystkie parametry wybranej kolumny

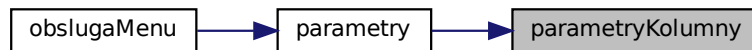
##### Parameters

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:



#### 5.10.1.6 parametryWiersza()

```

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

Funkcja od wyznaczania parametrów wiersza arkusza.

Funkcja zwraca w postaci tekstowej wszystkie parametry wybranego wiersza

##### Parameters

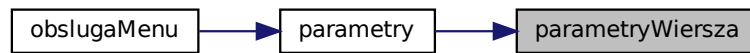
in	<i>arkusz</i>	Przekazywany arkusz do wykorzystania w funkcji obliczania parametrów
in	<i>wiersz</i>	Wiersz względem której zostaną obliczone parametry

Here is the call graph for this function:





Here is the caller graph for this function:



#### 5.10.1.7 rozszerzArkusz()

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

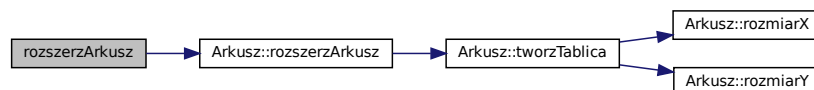
Funkcja modyfikująca rozmiar arkusza.

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:



### 5.10.1.8 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:



### 5.10.1.9 wczytanie()

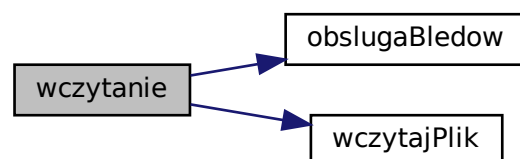
```
void wczytanie (  
    Arkusz * arkusz )
```

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

#### Parameters

in, out	arkusz	<a href="#">Arkusz</a> do którego mogą być wczytane elementy
---------	--------	--

Here is the call graph for this function:



Here is the caller graph for this function:



#### 5.10.1.10 zapis()

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

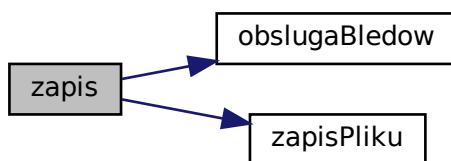
Funkcja menu od zapisu.

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

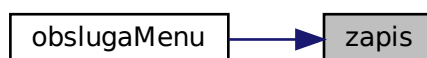
##### Parameters

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:



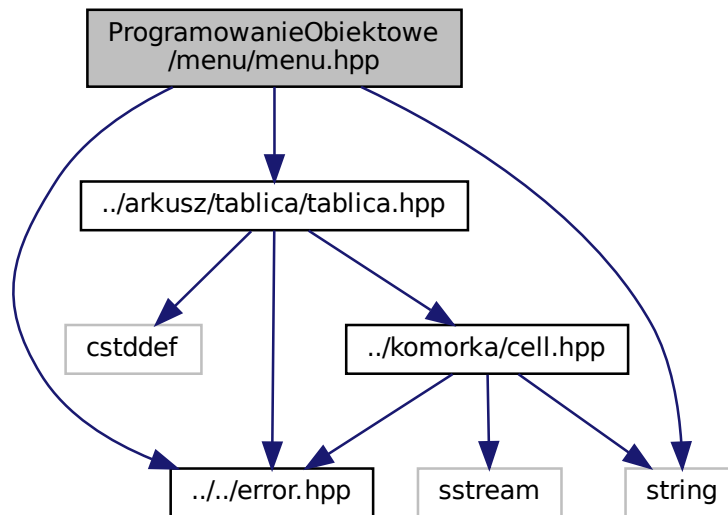
## 5.11 ProgramowanieObiektowe/menu/menu.hpp File Reference

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

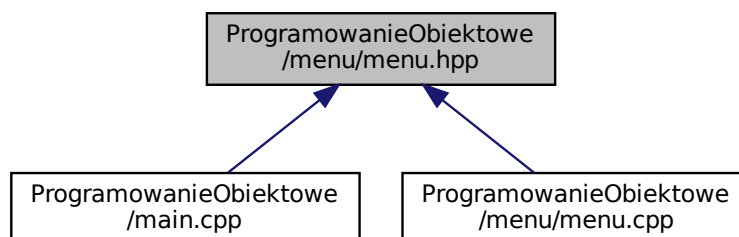
```
#include <string>
```

```
#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.*

## 5.11.1 Function Documentation

### 5.11.1.1 generujMenu()

```
void generujMenu ( )
```

Funkcja tworząca menu.

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



### 5.11.1.2 obsługaBledow()

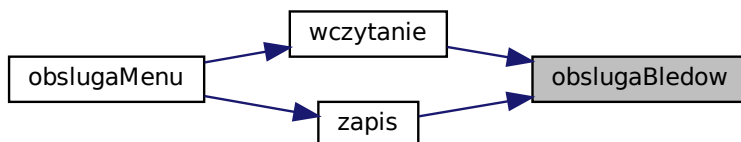
```
void obsługaBledow (
    Wyjatk wyjatek )
```

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:

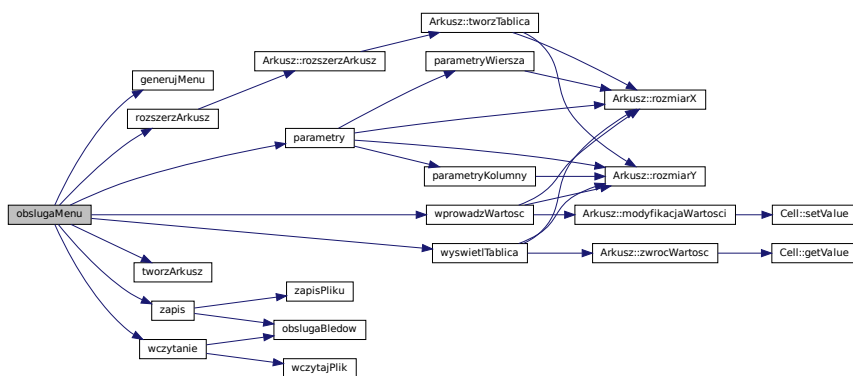


## 5.11.1.3 obslugaMenu()

```
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:



## 5.11.1.4 parametry()

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

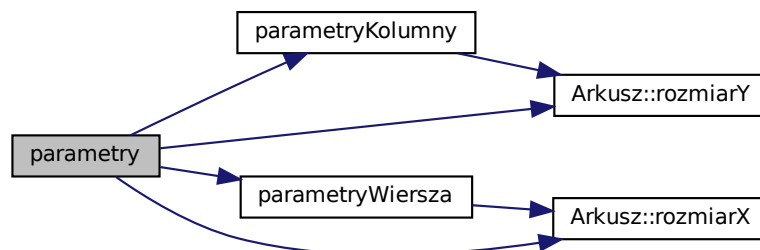
Funkcja menu od wyboru względem czego wyznacza parametry.

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:



## 5.11.1.5 parametryKolumny()

```

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

```

Funkcja od wyznaczania parametrów kolumny arkusza.

Funkcja zwraca w postaci tekstowej wszystkie parametry wybranej kolumny

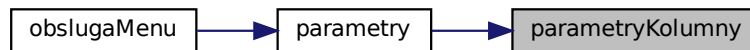
## Parameters

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:



#### 5.11.1.6 parametryWiersza()

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

Funkcja od wyznaczania parametrów wiersza arkusza.

Funkcja zwraca w postaci tekstowej wszystkie parametry wybranego wiersza

##### Parameters

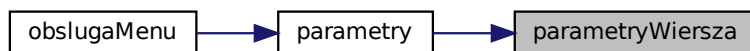
in	<i>arkusz</i>	Przekazywany arkusz do wykorzystania w funkcji obliczania parametrów
in	<i>wiersz</i>	Wiersz względem której zostaną obliczone parametry

Here is the call graph for this function:





Here is the caller graph for this function:



#### 5.11.1.7 rozszerzArkusz()

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

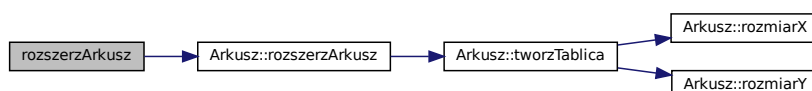
Funkcja modyfikująca rozmiar arkusza.

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:



#### 5.11.1.8 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:



#### 5.11.1.9 wczytanie()

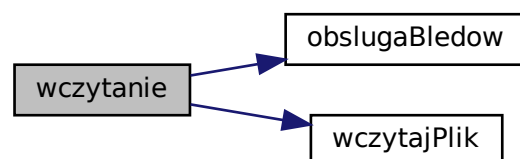
```
void wczytanie (  
    Arkusz * arkusz )
```

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

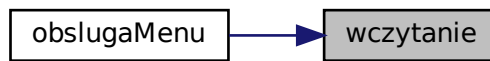
##### Parameters

in, out	arkusz	<a href="#">Arkusz</a> do którego mogą być wczytane elementy
---------	--------	--

Here is the call graph for this function:



Here is the caller graph for this function:



#### 5.11.1.10 zapis()

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

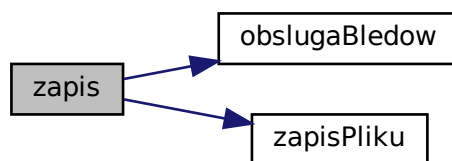
Funkcja menu od zapisu.

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

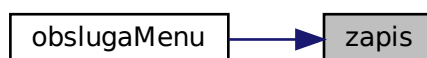
##### Parameters

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

- AbstractCell, [7](#)
- Arkusz, [7](#)
  - Arkusz, [8](#)
  - modyfikacjaWartosci, [9](#)
  - rozmiarX, [10](#)
  - rozmiarY, [10](#)
  - rozszerzArkusz, [11](#)
  - tworzTablica, [12](#)
  - zwrocWartosc, [13](#)
- Calculateable
  - Cell< T >, [15](#)
- Cell
  - Cell< T >, [15](#)
- Cell< T >, [14](#)
  - Calculateable, [15](#)
  - Cell, [15](#)
  - getString, [16](#)
  - getValue, [16](#)
  - setValue, [16](#)
- error.hpp
  - PLIK\_ACCESS, [28](#)
  - PLIK\_FORMAT, [28](#)
  - PLIK\_ROZMIAR, [28](#)
  - TABLICA\_SIZE, [28](#)
  - TABLICA\_ZAKR, [28](#)
  - Wyjatki, [28](#)
- generujMenu
  - menu.cpp, [33](#)
  - menu.hpp, [41](#)
- getString
  - Cell< T >, [16](#)
- getValue
  - Cell< T >, [16](#)
- menu.cpp
  - generujMenu, [33](#)
  - obsługaBledow, [33](#)
  - obsługaMenu, [33](#)
  - parametry, [34](#)
  - parametryKolumny, [35](#)
  - parametryWiersza, [36](#)
  - rozszerzArkusz, [37](#)
  - tworzArkusz, [37](#)
  - wczytanie, [38](#)
  - zapis, [39](#)
- menu.hpp
  - generujMenu, [41](#)
- obsługaBledow, [41](#)
- obsługaMenu, [42](#)
- parametry, [42](#)
- parametryKolumny, [43](#)
- parametryWiersza, [44](#)
- rozszerzArkusz, [45](#)
- tworzArkusz, [45](#)
- wczytanie, [46](#)
- zapis, [47](#)
- modyfikacjaWartosci
  - Arkusz, [9](#)
- obsługaBledow
  - menu.cpp, [33](#)
  - menu.hpp, [41](#)
- obsługaMenu
  - menu.cpp, [33](#)
  - menu.hpp, [42](#)
- parametry
  - menu.cpp, [34](#)
  - menu.hpp, [42](#)
- parametryKolumny
  - menu.cpp, [35](#)
  - menu.hpp, [43](#)
- parametryWiersza
  - menu.cpp, [36](#)
  - menu.hpp, [44](#)
- PLIK\_ACCESS
  - error.hpp, [28](#)
- PLIK\_FORMAT
  - error.hpp, [28](#)
- PLIK\_ROZMIAR
  - error.hpp, [28](#)
- ProgramowanieObiektowe/arkusz/komorka/cell.cpp, [19](#)
- ProgramowanieObiektowe/arkusz/komorka/cell.hpp, [19](#)
- ProgramowanieObiektowe/arkusz/tablica/tablica.cpp, [20](#)
- ProgramowanieObiektowe/arkusz/tablica/tablica.hpp, [21](#)
- ProgramowanieObiektowe/arkusz/tablica/tablica\_wysw.cpp, [24](#)
- ProgramowanieObiektowe/error.hpp, [27](#)
- ProgramowanieObiektowe/io/zapisOdczyt.cpp, [28](#)
- ProgramowanieObiektowe/io/zapisOdczyt.hpp, [29](#)
- ProgramowanieObiektowe/main.cpp, [31](#)
- ProgramowanieObiektowe/menu/menu.cpp, [32](#)
- ProgramowanieObiektowe/menu/menu.hpp, [40](#)
- rozmiarX
  - Arkusz, [10](#)

- rozmiarY
  - Arkusz, [10](#)
- rozszerzArkusz
  - Arkusz, [11](#)
  - menu.cpp, [37](#)
  - menu.hpp, [45](#)
- setValue
  - Cell< T >, [16](#)
- tablica.hpp
  - wprowadzWartosc, [22](#)
  - wyswietlTablica, [23](#)
- TABLICA\_SIZE
  - error.hpp, [28](#)
- tablica\_wysw.cpp
  - wprowadzWartosc, [25](#)
  - wyswietlTablica, [26](#)
- TABLICA\_ZAKR
  - error.hpp, [28](#)
- tworzArkusz
  - menu.cpp, [37](#)
  - menu.hpp, [45](#)
- tworzTablica
  - Arkusz, [12](#)
- wczytajPlik
  - zapisOdczyt.hpp, [30](#)
- wczytanie
  - menu.cpp, [38](#)
  - menu.hpp, [46](#)
- wprowadzWartosc
  - tablica.hpp, [22](#)
  - tablica\_wysw.cpp, [25](#)
- Wyjatk
  - error.hpp, [28](#)
- wyswietlTablica
  - tablica.hpp, [23](#)
  - tablica\_wysw.cpp, [26](#)
- zapis
  - menu.cpp, [39](#)
  - menu.hpp, [47](#)
- zapisOdczyt.hpp
  - wczytajPlik, [30](#)
  - zapisPliku, [30](#)
- zapisPliku
  - zapisOdczyt.hpp, [30](#)
- zwrocWartosc
  - Arkusz, [13](#)