

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	8
4.1.2 Constructor & Destructor Documentation	8
4.1.2.1 Arkusz()	8
4.1.3 Member Function Documentation	8
4.1.3.1 czyTekstowa()	8
4.1.3.2 modyfikacjaWartosci() [1/2]	9
4.1.3.3 modyfikacjaWartosci() [2/2]	10
4.1.3.4 rozmiarX()	10
4.1.3.5 rozmiarY()	11
4.1.3.6 rozszerzArkusz()	11
4.1.3.7 tworzTablica()	12
4.1.3.8 tworzTablicaInt()	13
4.1.3.9 tworzTablicaString()	14
4.1.3.10 zwrocWartosc()	15
4.2 Cell Class Reference	16
4.2.1 Detailed Description	17
4.2.2 Constructor & Destructor Documentation	17
4.2.2.1 Cell()	17
4.2.3 Member Function Documentation	17
4.2.3.1 getValue()	17
4.2.3.2 isCalcualteable()	18
4.2.3.3 operator+() [1/2]	18
4.2.3.4 operator+() [2/2]	19
4.2.3.5 toString()	19
4.3 intCell Class Reference	20
4.3.1 Detailed Description	21
4.3.2 Constructor & Destructor Documentation	21
4.3.2.1 intCell()	21
4.3.3 Member Function Documentation	22
4.3.3.1 getValue()	22
4.3.3.2 setValue()	22
4.3.3.3 toString()	23

4.4 stringCell Class Reference	23
4.4.1 Detailed Description	24
4.4.2 Constructor & Destructor Documentation	24
4.4.2.1 stringCell()	24
4.4.3 Member Function Documentation	25
4.4.3.1 setValue()	25
4.4.3.2 toString()	25
5 File Documentation	27
5.1 ProgramowanieObiektowe/arkusz/komorka/cell.cpp File Reference	27
5.2 ProgramowanieObiektowe/arkusz/komorka/cell.hpp File Reference	27
5.3 ProgramowanieObiektowe/arkusz/komorka/intCell.cpp File Reference	28
5.4 ProgramowanieObiektowe/arkusz/komorka/intCell.hpp File Reference	29
5.5 ProgramowanieObiektowe/arkusz/komorka/stringCell.cpp File Reference	30
5.6 ProgramowanieObiektowe/arkusz/komorka/stringCell.hpp File Reference	31
5.7 ProgramowanieObiektowe/arkusz/tablica/tablica.cpp File Reference	31
5.8 ProgramowanieObiektowe/arkusz/tablica/tablica.hpp File Reference	32
5.9 ProgramowanieObiektowe/arkusz/tablica/tablica_wysw.cpp File Reference	33
5.9.1 Function Documentation	34
5.9.1.1 wprowadzWartosc()	34
5.9.1.2 wyswietlTablica()	35
5.10 ProgramowanieObiektowe/error.hpp File Reference	36
5.10.1 Enumeration Type Documentation	36
5.10.1.1 Wyjatki	36
5.11 ProgramowanieObiektowe/io/zapisOdczyt.cpp File Reference	37
5.12 ProgramowanieObiektowe/io/zapisOdczyt.hpp File Reference	37
5.12.1 Function Documentation	38
5.12.1.1 wczytajPlik()	39
5.12.1.2 zapisPliku()	39
5.13 ProgramowanieObiektowe/main.cpp File Reference	40
5.14 ProgramowanieObiektowe/menu/menu.cpp File Reference	41
5.14.1 Function Documentation	42
5.14.1.1 generujMenu()	42
5.14.1.2 obslugaBledow()	42
5.14.1.3 obslugaMenu()	43
5.14.1.4 parametry()	43
5.14.1.5 parametryKolumny()	44
5.14.1.6 parametryWiersza()	45
5.14.1.7 rozszerzArkusz()	46
5.14.1.8 tworzyArkusz()	47
5.14.1.9 wczytanie()	47
5.14.1.10 zapis()	48

5.15 ProgramowanieObiektowe/menu/menu.hpp File Reference	49
5.15.1 Function Documentation	50
5.15.1.1 generujMenu()	50
5.15.1.2 obslugaBledow()	50
5.15.1.3 obslugaMenu()	51
5.15.1.4 parametry()	52
5.15.1.5 parametryKolumny()	52
5.15.1.6 parametryWiersza()	53
5.15.1.7 rozszerzArkusz()	54
5.15.1.8 tworzArkusz()	55
5.15.1.9 wczytanie()	55
5.15.1.10 zapis()	56
Index	59

Chapter 1

Hierarchical Index

1.1 Class Hierarchy

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

Arkusz	7
Cell	16
intCell	20
stringCell	23

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	16
intCell	Klasa komórki wartości całkowitych	20
stringCell	Klasa komórki tekstowej	23

Chapter 3

File Index

3.1 File List

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

ProgramowanieObiektowe/ error.hpp	36
ProgramowanieObiektowe/ main.cpp	40
ProgramowanieObiektowe/arkusz/komorka/ cell.cpp	27
ProgramowanieObiektowe/arkusz/komorka/ cell.hpp	27
ProgramowanieObiektowe/arkusz/komorka/ intCell.cpp	28
ProgramowanieObiektowe/arkusz/komorka/ intCell.hpp	29
ProgramowanieObiektowe/arkusz/komorka/ stringCell.cpp	30
ProgramowanieObiektowe/arkusz/komorka/ stringCell.hpp	31
ProgramowanieObiektowe/arkusz/tablica/ tablica.cpp	31
ProgramowanieObiektowe/arkusz/tablica/ tablica.hpp	32
ProgramowanieObiektowe/arkusz/tablica/ tablica_wysw.cpp	33
ProgramowanieObiektowe/arkusz/tablica/ tablica_wysw.hpp	??
ProgramowanieObiektowe/io/ zapisOdczyt.cpp	37
ProgramowanieObiektowe/io/ zapisOdczyt.hpp	37
ProgramowanieObiektowe/menu/ menu.cpp	41
ProgramowanieObiektowe/menu/ menu.hpp	49
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, bool [czyTekstowa](#)=false)
Konstruktor tworzący arkusz z tablicą o wyznaczonym rozmiarze i wybranym typie Konstruktor tworzący arkusz z tablicą o wyznaczonej ilości kolumn i wierszy.
- [Wyjątki modyfikacjaWartosci](#) (size_t x, size_t y, std::string wartosc)
Wstawia określoną wartość do komórki tekstowej tablicy w arkuszu.
- [Wyjątki modyfikacjaWartosci](#) (size_t x, size_t y, int wartosc)
Wstawia określoną wartość do komórki liczbowej tablicy w arkuszu.
- [Wyjątki rozszerzArkusz](#) (size_t x, size_t y)
Modyfikacja rozmiaru tablicy.
- [Komorka & zwrocWartosc](#) (size_t x, size_t y)
Zwraca wybraną komórkę
- size_t [rozmiarX](#) ()
Getter ilości kolumn.
- size_t [rozmiarY](#) ()
Getter ilości wierszy.
- bool [czyTekstowa](#) ()
zwraca czy tablica jest typu tekstowego Funkcja zwraca czy tablica jest typu tekstowego

Static Public Member Functions

- static [Tablica tworzTablica](#) (size_t kolumny, size_t wiersze)
Określa czy tabela jest tylko typu tekstowego.
- static [Tablica tworzTablicaInt](#) (size_t kolumny, size_t wiersze)
Tworzy nową dwuwymiarową tablicę typu komórek Int.
- static [Tablica tworzTablicaString](#) (size_t kolumny, size_t wiersze)
Tworzy nową dwuwymiarową tablicę typu tekstowego.

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()

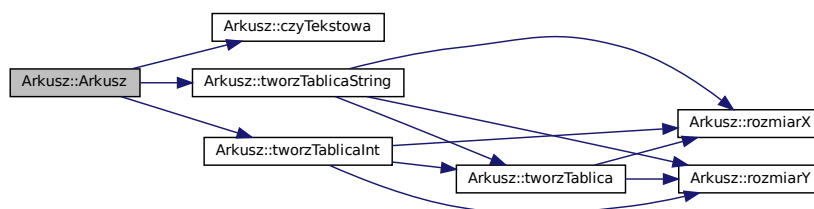
```
Arkusz::Arkusz (
    size_t x,
    size_t y,
    bool czyTekstowa = false )
```

Konstruktor tworzący arkusz z tablicą o wyznaczonym rozmiarze i wybranym typie 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
in	<i>czyTekstowa</i>	Określa czy tworzony arkusz jest tekstowy czy liczbowy

Here is the call graph for this function:



4.1.3 Member Function Documentation

4.1.3.1 czyTekstowa()

```
bool Arkusz::czyTekstowa ( )
```

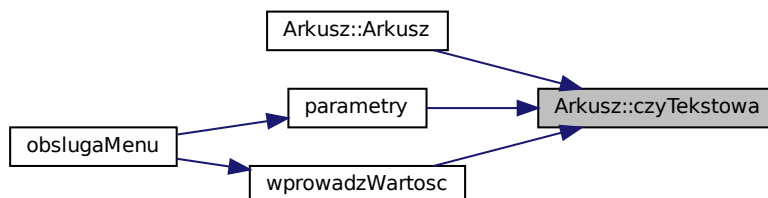
zwraca czy tablica jest typu tekstowego Funkcja zwraca czy tablica jest typu tekstowego

Returns

true komórki tablicy są typu std::string

false komórki są typu int

Here is the caller graph for this function:

**4.1.3.2 modyfikacjaWartosci() [1/2]**

```

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

Wstawia określoną wartość do komórki liczbowej tablicy w arkuszu.

Funkcja wstawia wartość do komórki o określonym adresie

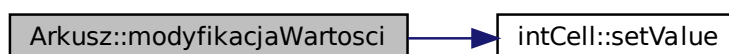
Parameters

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:



4.1.3.3 modyfikacjaWartosci() [2/2]

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

Wstawia określoną wartość do komórki tekstowej tablicy w arkuszu.

Funkcja wstawia wartość do komórki o określonym adresie

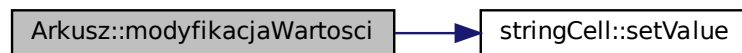
Parameters

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

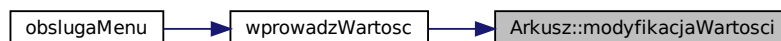
Returns

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:



4.1.3.4 rozmiarX()

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

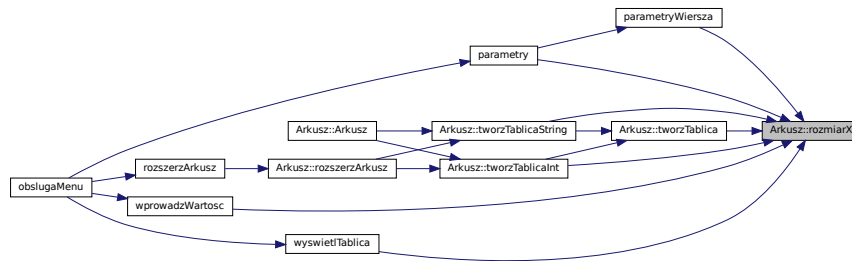
Getter ilości kolumn.

Funkcja zwraca ilość kolumn danego arkusza

Returns

Ilość kolumn arkusza

Here is the caller graph for this function:



4.1.3.5 rozmiarY()

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

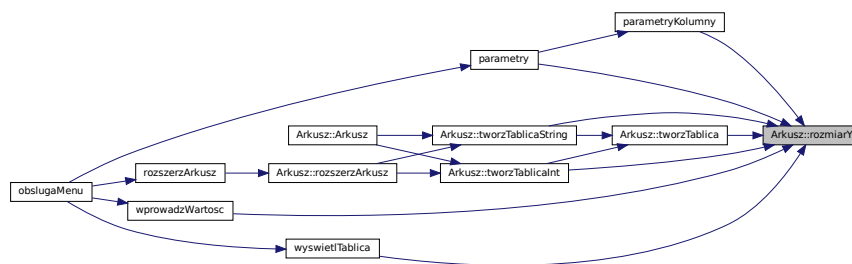
Getter ilości wierszy.

Funkcja zwraca ilość wierszy danego arkusza

Returns

Ilość wierszy arkusza

Here is the caller graph for this function:



4.1.3.6 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)

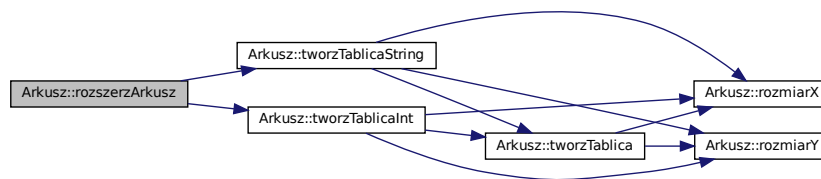
Parameters

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

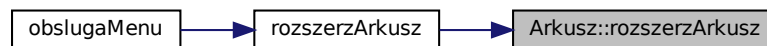
Returns

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:



4.1.3.7 tworzTablica()

```

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

Określa czy tabela jest tylko typu tekstowego.

Tworzy nową dwuwymiarową tablicę.

Funkcja generująca pustą tablicę komórek o określonym rozmiarze

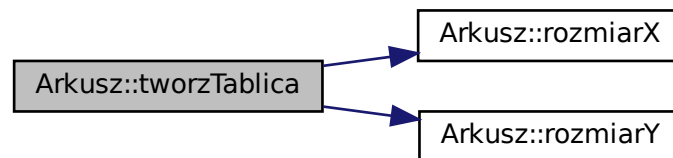
Parameters

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

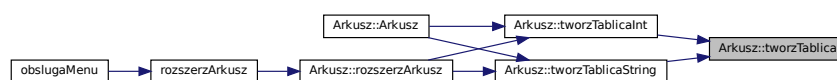
Returns

Nową tablicę dwuwymiarową z komórkami liczbowymi o wyznaczonych rozmiarach

Here is the call graph for this function:



Here is the caller graph for this function:

**4.1.3.8 tworzTablicaInt()**

```

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

Tworzy nową dwuwymiarową tablicę typu komórek `Int`.

Funkcja generująca tablicę tekstową o określonym rozmiarze

Parameters

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

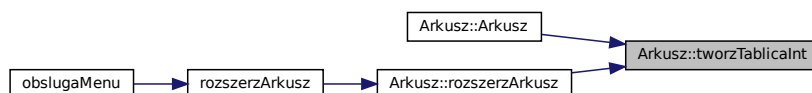
Returns

Nową tablicę dwuwymiarową z komórkami liczbowymi o wyznaczonych rozmiarach

Here is the call graph for this function:



Here is the caller graph for this function:

**4.1.3.9 tworzTablicaString()**

```

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

Tworzy nową dwuwymiarową tablicę typu tekstowego.

Funkcja generująca tablicę tekstową o określonym rozmiarze

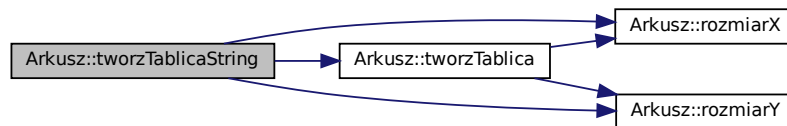
Parameters

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

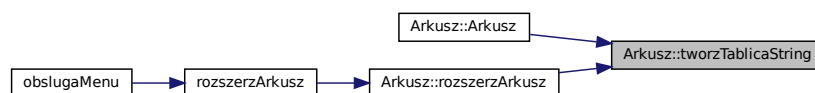
Returns

Nową tablicę dwuwymiarową z komórkami tekstowymi o wyznaczonych rozmiarach

Here is the call graph for this function:



Here is the caller graph for this function:

**4.1.3.10 zwrocWartosc()**

```

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

Zwraca wybraną komórkę

Funkcja zwraca referencje wybranej komórki

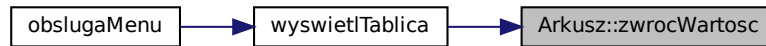
Parameters

in	x	Wybrana kolumna
in	y	Wybrany wiersz

Returns

Wybrana komórka

Here is the caller graph for this function:



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

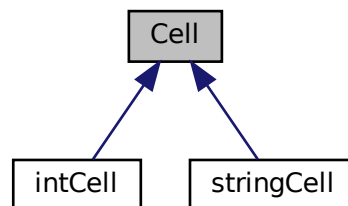
- [ProgramowanieObiektowe/arkusz/tablica/tablica.hpp](#)
- [ProgramowanieObiektowe/arkusz/tablica/tablica.cpp](#)

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 `isCalcuable` ()

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	<i>c1</i>	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

in	<i>wart</i>	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]
```

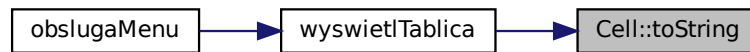
Metoda od pobierania wartości ciągu znaków Zwraca wartość typu String - ciąg znaków do wykorzystywania przykładowo w wypisywaniu.

Returns

Zwraca wartość typu string

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

Here is the caller graph for this function:



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

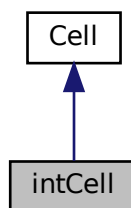
- [ProgramowanieObiektowe/arkusz/komorka/cell.hpp](#)
- [ProgramowanieObiektowe/arkusz/komorka/cell.cpp](#)

4.3 intCell Class Reference

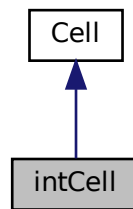
Klasa komórki wartości całkowitych.

```
#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()

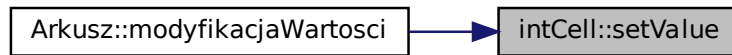
```
void intCell::setValue (
    int Val )
```

setter wartości komórki Setter umożliwiający ustawienie wartości całkowitej w komórce

Parameters

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

Here is the caller graph for this function:



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:

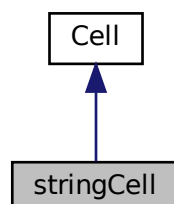
- [ProgramowanieObiektowe/arkusz/komorka/intCell.hpp](#)
- [ProgramowanieObiektowe/arkusz/komorka/intCell.cpp](#)

4.4 stringCell Class Reference

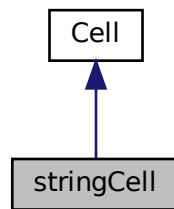
Klasa komórki tekstowej.

```
#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()

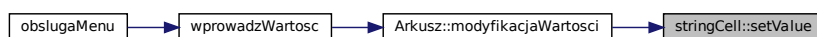
```
void stringCell::setValue (
    std::string Val )
```

setter wartości komórki Setter umożliwiający ustawienie wartości tekstowej w komórce

Parameters

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

Here is the caller graph for this function:



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](#)

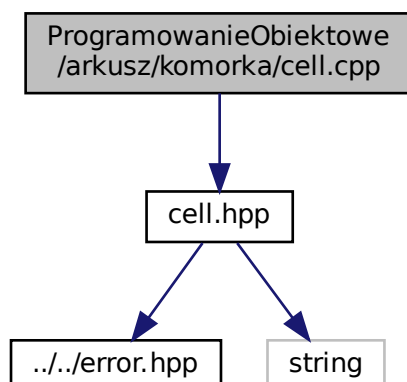
Chapter 5

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

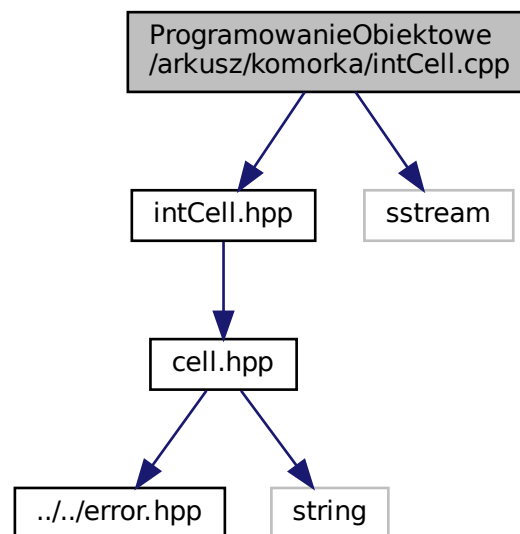
```
#include "../..error.hpp"
```

```
#include <string>
```

- class **Cell**

- Prototyp klasy komórk.*

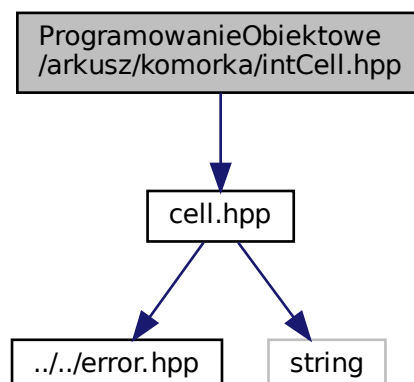
Include dependency graph for intCell.cpp:



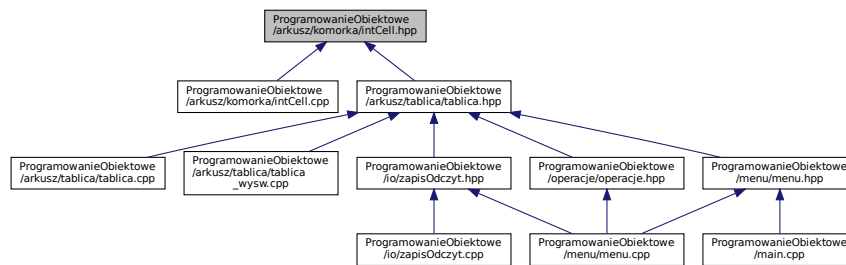
5.4 ProgramowanieObiektowe/arkusz/komorka/intCell.hpp File Reference

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

Include dependency graph for intCell.hpp:



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



Classes

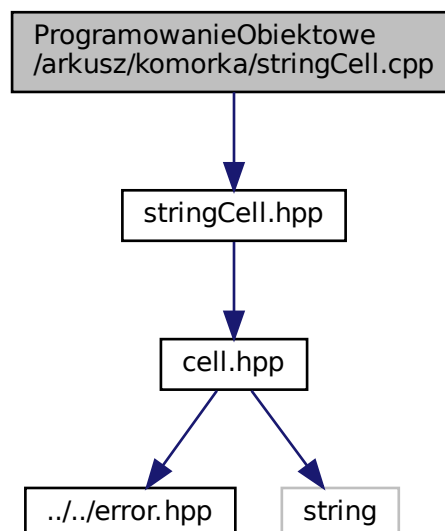
- class [intCell](#)

Klasa komórki wartości całkowitych.

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

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

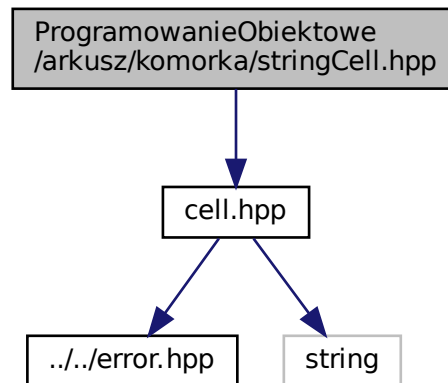
Include dependency graph for stringCell.cpp:



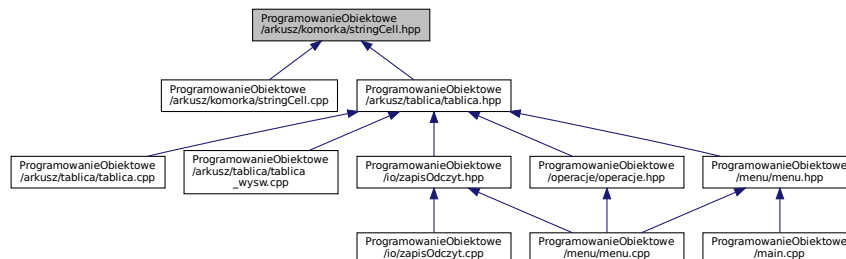
5.6 ProgramowanieObiektowe/arkusz/komorka/stringCell.hpp File Reference

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

Include dependency graph for stringCell.hpp:



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



Classes

- class [stringCell](#)

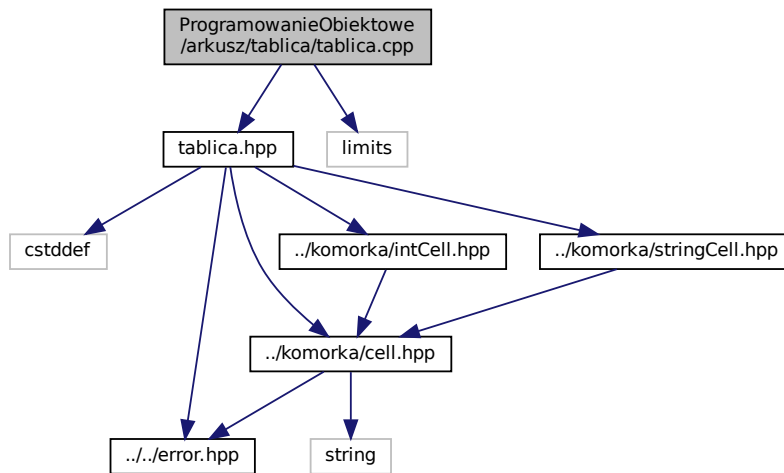
Klasa komórki tekstowej.

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

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

```
#include <limits>
```

Include dependency graph for `tablica.cpp`:



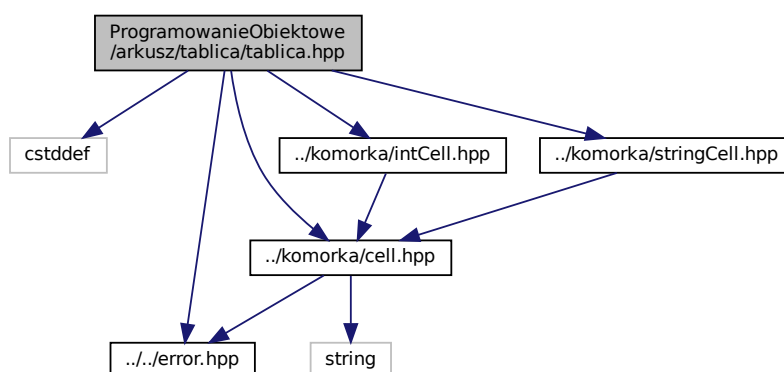
5.8 ProgramowanieObiektowe/arkusz/tablica/tablica.hpp File Reference

```

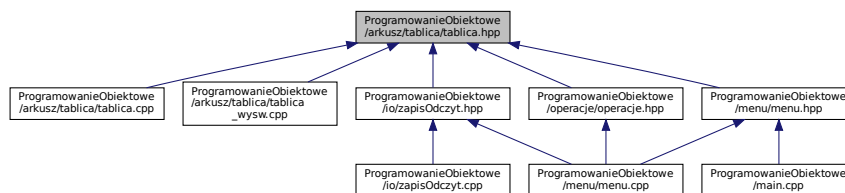
#include <cstddef>
#include "../error.hpp"
#include "../komorka/cell.hpp"
#include "../komorka/intCell.hpp"
#include "../komorka/stringCell.hpp"

```

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](#) [Komorka](#)

Definicja/alias typu `int` jako typ określający komórkę

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

Definicja/alias typu `Komorka**` jako `Tablica` komórek.

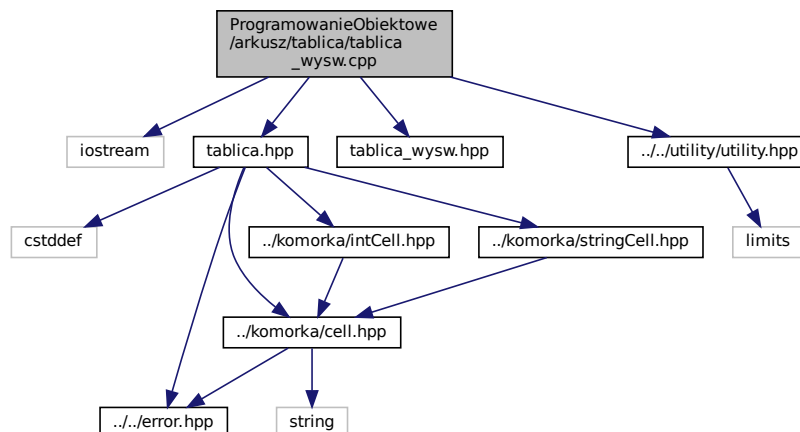
5.9 ProgramowanieObiektowe/arkusz/tablica/tablica_wysw.cpp File Reference

```

#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 )
```

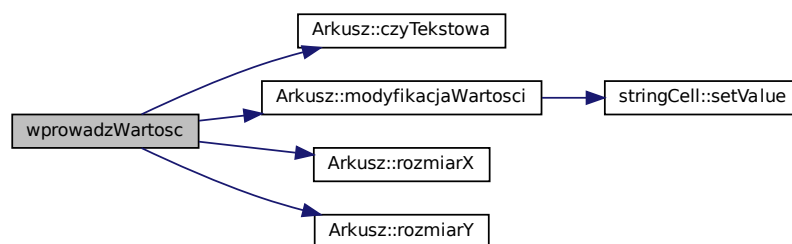
Interfejs użytkownika wprowadzania nowej wartości.

Funkcja interfejsu wprowadzenia wartości do tablicy. Wprowadzana wartość spoza przedziału zostanie zablokowana. W zależności od typu arkusza umożliwia wprowadzenie wartości tekstowej bądź liczbowej.

Parameters

in, out	<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 )
```

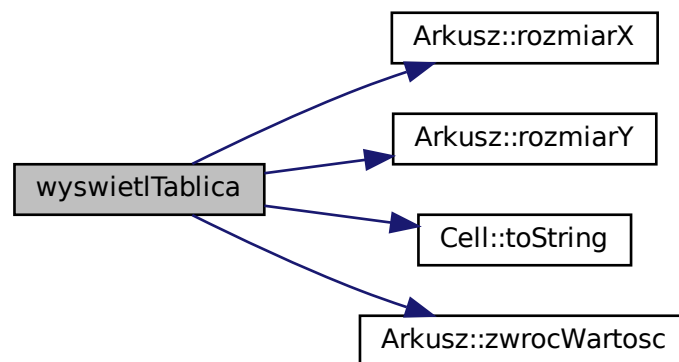
Generuje podgląd tablicy.

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:



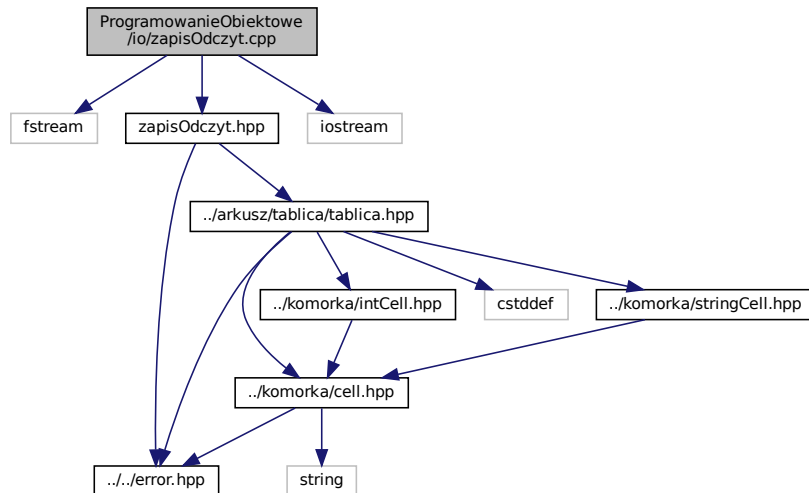
Here is the caller graph for this function:



5.11 ProgramowanieObiektowe/io/zapisOdczyt.cpp File Reference

```
#include <fstream>
#include "zapisOdczyt.hpp"
#include <iostream>
```

Include dependency graph for zapisOdczyt.cpp:



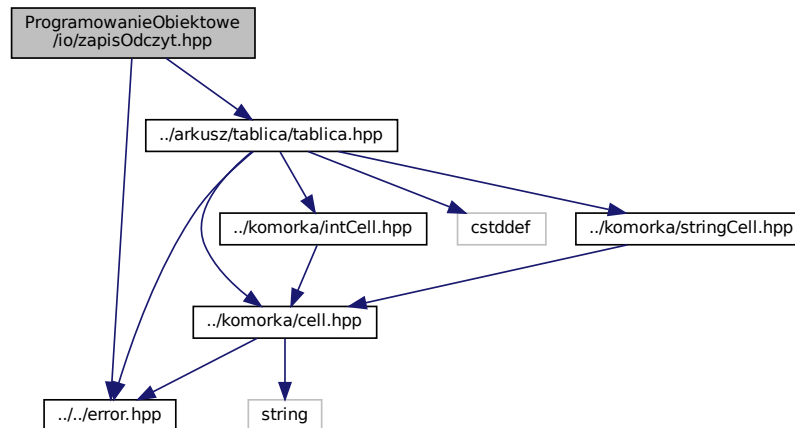
Functions

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

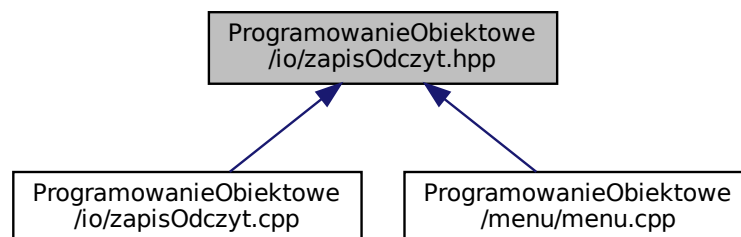
5.12 ProgramowanieObiektowe/io/zapisOdczyt.hpp File Reference

```
#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.

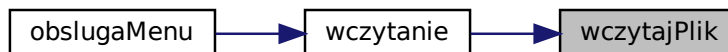
Parameters

in, out	arkusz	Arkusz do nadpisanie 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. Początkowo wczytana tablica jest typu tekstowego, dopiero po wczytaniu wszystkich elementów jest sprawdzana możliwość tego czy tablica jest typu liczbowego (preferowana opcja).

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"

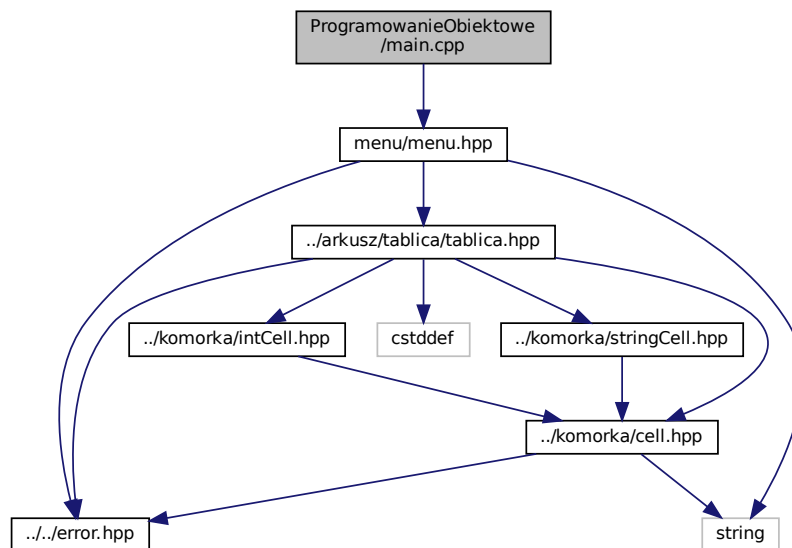
Here is the caller graph for this function:



5.13 ProgramowanieObiektowe/main.cpp File Reference

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

Include dependency graph for main.cpp:



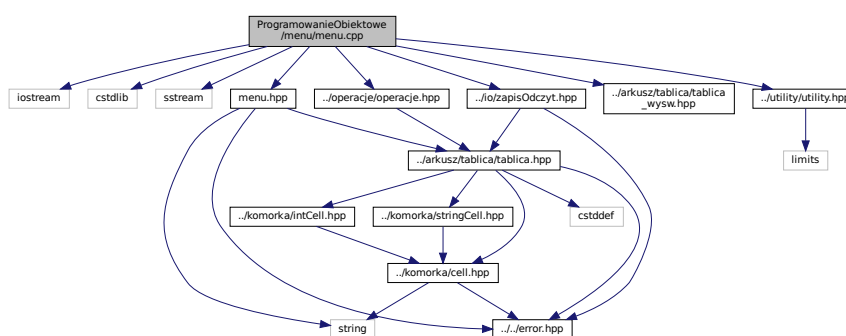
Functions

- `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](#) ([Wyjątki](#) wyjątek)
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()

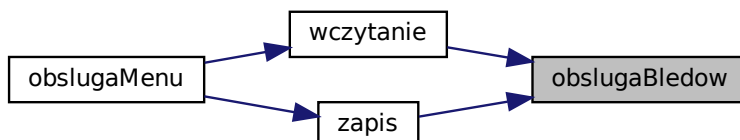
```
void obslugaBledow (
    Wyjatki wyjatek )
```

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:

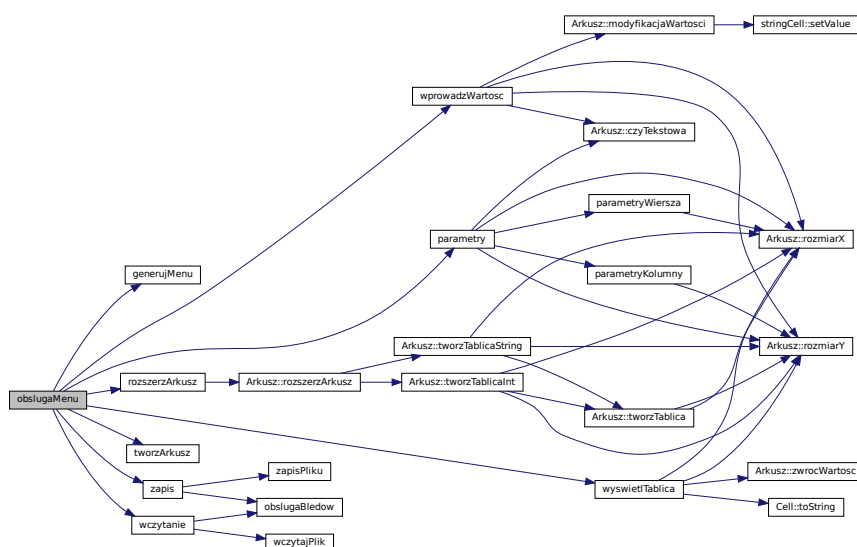


5.14.1.3 obslugaMenu()

```
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 )
```

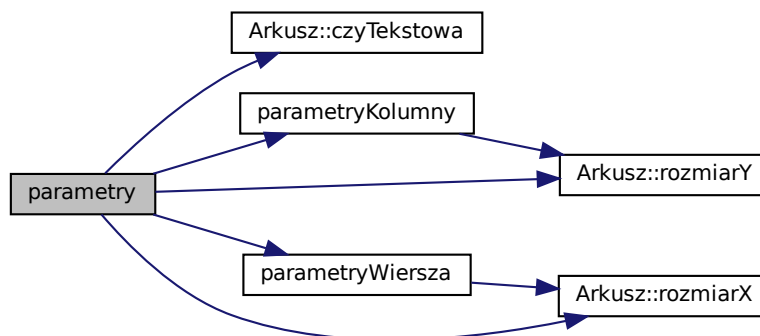
Funkcja menu od wyboru względem czego wyznacza parametry.

Funkcja menu od wyboru atrybutu tablicy (kolumny lub wiersza) która ma za wyświetlenie parametrów wybranego atrybutu. Funkcja nie wykona się dla tablic czysto tekstowych.

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

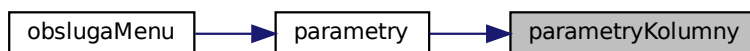
Parameters

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

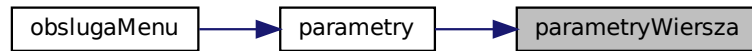
Parameters

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 )
```

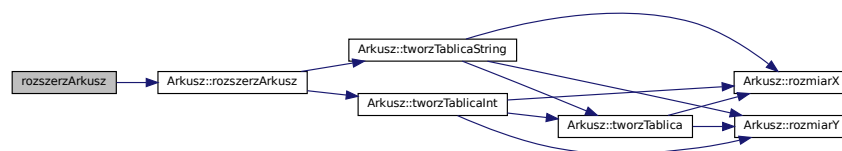
Funkcja modyfikująca rozmiar arkusza.

Interfejs umożliwiający modyfikację rozmiaru istniejącego arkusza.

Parameters

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

Here is the call graph for this function:



Here is the caller graph for this function:



5.14.1.8 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()

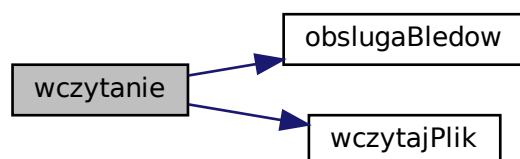
```
void wczytanie (
    Arkusz * arkusz )
```

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

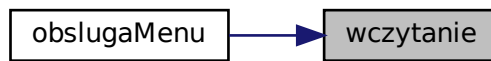
Parameters

<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:



5.14.1.10 zapis()

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

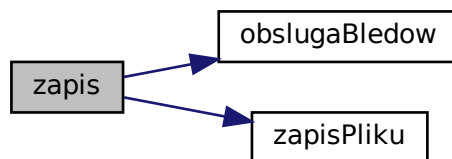
Funkcja menu od zapisu.

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

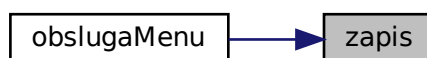
Parameters

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:



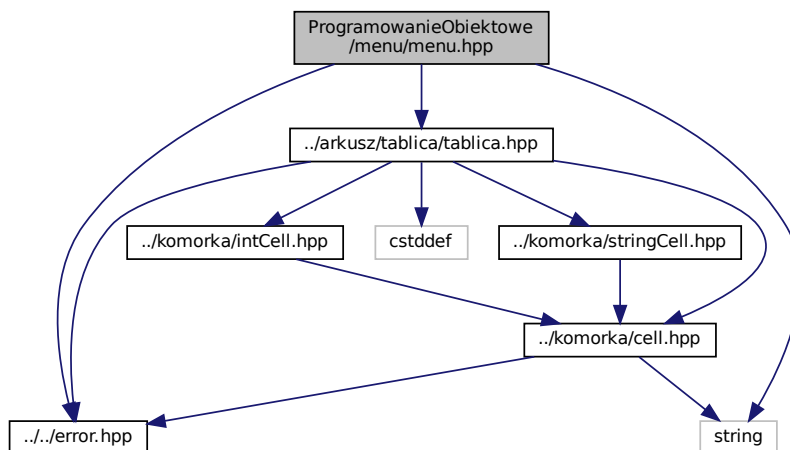
5.15 ProgramowanieObiektowe/menu/menu.hpp File Reference

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

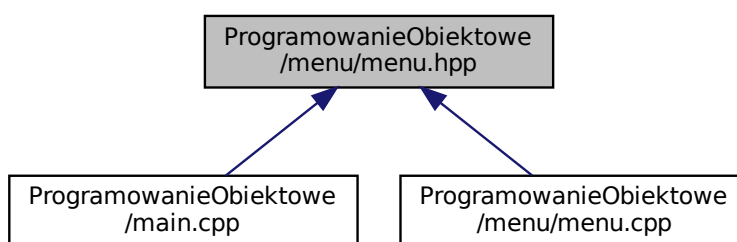
```
#include <string>
```

```
#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 tworArkusz](#) ()
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()

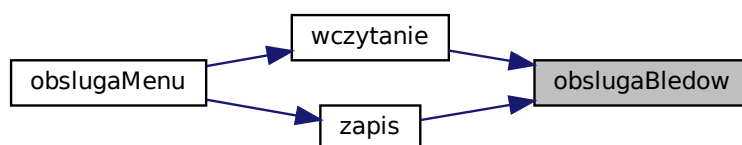
```
void obsługaBledow (
    Wyjatk wyjatek )
```

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:

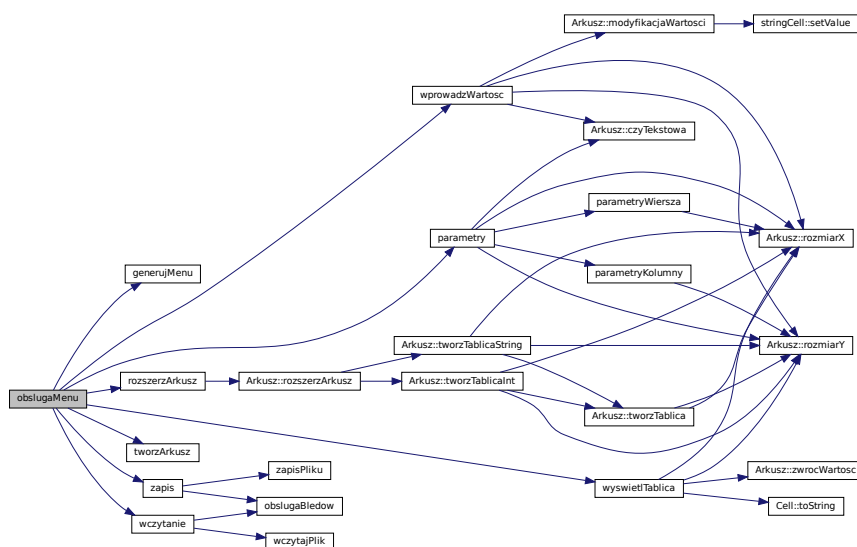


5.15.1.3 obslugaMenu()

```
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 )
```

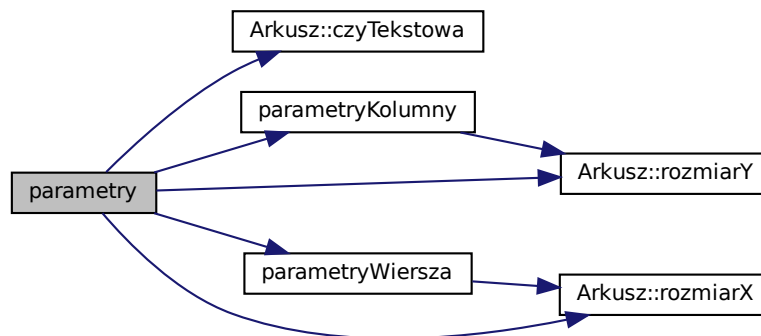
Funkcja menu od wyboru względem czego wyznacza parametry.

Funkcja menu od wyboru atrybutu tablicy (kolumny lub wiersza) która ma za wyświetlenie parametrów wybranego atrybutu. Funkcja nie wykona się dla tablic czysto tekstowych.

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:



5.15.1.5 parametryKolumny()

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

Funkcja od wyznaczania parametrów kolumny arkusza.

Funkcja zwraca w postaci tekstowej wszystkie parametry wybranej kolumny

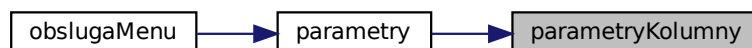
Parameters

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

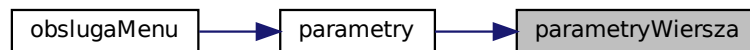
Parameters

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 )
```

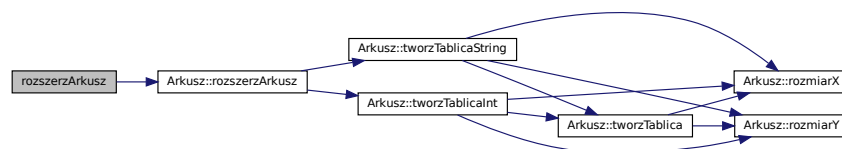
Funkcja modyfikująca rozmiar arkusza.

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()

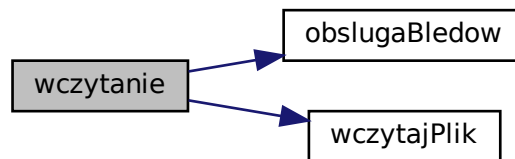
```
void wczytanie (
    Arkusz * arkusz )
```

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 )
```

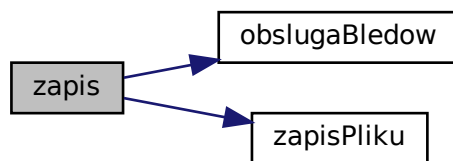
Funkcja menu od zapisu.

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

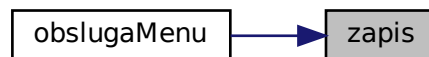
Parameters

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](#)
 - czyTekstowa, [8](#)
 - modyfikacjaWartosci, [9](#), [10](#)
 - rozmiarX, [10](#)
 - rozmiarY, [11](#)
 - rozszerzArkusz, [11](#)
 - tworzTablica, [12](#)
 - tworzTablicaInt, [13](#)
 - tworzTablicaString, [14](#)
 - zwrocWartosc, [15](#)
- Cell, [16](#)
 - Cell, [17](#)
 - getValue, [17](#)
 - isCalcualteable, [18](#)
 - operator+, [18](#), [19](#)
 - toString, [19](#)
- czyTekstowa
 - Arkusz, [8](#)
- error.hpp
 - PLIK_ACCESS, [36](#)
 - PLIK_FORMAT, [36](#)
 - PLIK_ROZMIAR, [36](#)
 - TABLICA_SIZE, [36](#)
 - TABLICA_ZAKR, [36](#)
 - Wyjatk, [36](#)
- generujMenu
 - menu.cpp, [42](#)
 - menu.hpp, [50](#)
- getValue
 - Cell, [17](#)
 - intCell, [22](#)
- intCell, [20](#)
 - getValue, [22](#)
 - intCell, [21](#)
 - setValue, [22](#)
 - toString, [23](#)
- isCalcualteable
 - Cell, [18](#)
- menu.cpp
 - generujMenu, [42](#)
 - obsługaBledow, [42](#)
 - obsługaMenu, [42](#)
 - parametry, [43](#)
 - parametryKolumny, [44](#)
 - parametryWiersza, [45](#)
- rozszerzArkusz, [46](#)
- tworzArkusz, [46](#)
- wczytanie, [47](#)
- zapis, [48](#)
- menu.hpp
 - generujMenu, [50](#)
 - obsługaBledow, [50](#)
 - obsługaMenu, [51](#)
 - parametry, [51](#)
 - parametryKolumny, [52](#)
 - parametryWiersza, [53](#)
 - rozszerzArkusz, [54](#)
 - tworzArkusz, [55](#)
 - wczytanie, [55](#)
 - zapis, [56](#)
- modyfikacjaWartosci
 - Arkusz, [9](#), [10](#)
- obsługaBledow
 - menu.cpp, [42](#)
 - menu.hpp, [50](#)
- obsługaMenu
 - menu.cpp, [42](#)
 - menu.hpp, [51](#)
- operator+
 - Cell, [18](#), [19](#)
- parametry
 - menu.cpp, [43](#)
 - menu.hpp, [51](#)
- parametryKolumny
 - menu.cpp, [44](#)
 - menu.hpp, [52](#)
- parametryWiersza
 - menu.cpp, [45](#)
 - menu.hpp, [53](#)
- PLIK_ACCESS
 - error.hpp, [36](#)
- PLIK_FORMAT
 - error.hpp, [36](#)
- PLIK_ROZMIAR
 - error.hpp, [36](#)
- ProgramowanieObiektowe/arkusz/komorka/cell.cpp, [27](#)
- ProgramowanieObiektowe/arkusz/komorka/cell.hpp, [27](#)
- ProgramowanieObiektowe/arkusz/komorka/intCell.cpp, [28](#)
- ProgramowanieObiektowe/arkusz/komorka/intCell.hpp, [29](#)
- ProgramowanieObiektowe/arkusz/komorka/stringCell.cpp, [30](#)

ProgramowanieObiektowe/arkusz/komorka/stringCell.hpp, Wyjatki
 31 error.hpp, 36
 ProgramowanieObiektowe/arkusz/tablica/tablica.cpp, 31 wyswietlTablica
 ProgramowanieObiektowe/arkusz/tablica/tablica.hpp, tablica_wysw.cpp, 35
 32
 ProgramowanieObiektowe/arkusz/tablica/tablica_wysw.cpp, zapis
 33 menu.cpp, 48
 menu.hpp, 56
 ProgramowanieObiektowe/error.hpp, 36 zapisOdczyt.hpp
 ProgramowanieObiektowe/io/zapisOdczyt.cpp, 37 wczytajPlik, 38
 ProgramowanieObiektowe/io/zapisOdczyt.hpp, 37 zapisPliku, 39
 ProgramowanieObiektowe/main.cpp, 40 zapisPliku
 ProgramowanieObiektowe/menu/menu.cpp, 41 zapisOdczyt.hpp, 39
 ProgramowanieObiektowe/menu/menu.hpp, 49 zwrocWartosc
 Arkusz, 15
 rozmiarX
 Arkusz, 10
 rozmiarY
 Arkusz, 11
 rozszerzArkusz
 Arkusz, 11
 menu.cpp, 46
 menu.hpp, 54
 setValue
 intCell, 22
 stringCell, 25
 stringCell, 23
 setValue, 25
 stringCell, 24
 toString, 25
 TABLICA_SIZE
 error.hpp, 36
 tablica_wysw.cpp
 wprowadzWartosc, 34
 wyswietlTablica, 35
 TABLICA_ZAKR
 error.hpp, 36
 toString
 Cell, 19
 intCell, 23
 stringCell, 25
 tworzArkusz
 menu.cpp, 46
 menu.hpp, 55
 tworzTablica
 Arkusz, 12
 tworzTablicaInt
 Arkusz, 13
 tworzTablicaString
 Arkusz, 14
 wczytajPlik
 zapisOdczyt.hpp, 38
 wczytanie
 menu.cpp, 47
 menu.hpp, 55
 wprowadzWartosc
 tablica_wysw.cpp, 34