

Arkusz kalkulacyjny

Wygenerowano przez Doxygen 1.9.1

1 Arkusz kalkulacyjny	1
1.1 1. main.cpp	1
1.2 2. menu.cpp:	1
1.3 3. tablica.cpp:	1
1.4 4. tablica_wysw.cpp:	2
1.5 5. operacje_na_plikach.cpp:	2
2 Indeks hierarchiczny	3
2.1 Hierarchia klas	3
3 Indeks klas	5
3.1 Lista klas	5
4 Indeks plików	7
4.1 Lista plików	7
5 Dokumentacja klas	9
5.1 Dokumentacja klasy Cell	9
5.1.1 Opis szczegółowy	9
5.1.2 Dokumentacja funkcji składowych	9
5.1.2.1 getNum()	10
5.1.2.2 getValue()	10
5.1.2.3 setValue()	10
5.2 Dokumentacja klasy DoubleCell	11
5.2.1 Opis szczegółowy	11
5.2.2 Dokumentacja funkcji składowych	12
5.2.2.1 getNum()	12
5.2.2.2 getValue()	12
5.2.2.3 setValue()	12
5.2.3 Dokumentacja atrybutów składowych	12
5.2.3.1 doubleValue	12
5.3 Dokumentacja klasy Sheet	13
5.3.1 Opis szczegółowy	14
5.3.2 Dokumentacja funkcji składowych	14
5.3.2.1 average_value()	14
5.3.2.2 copy_tab()	14
5.3.2.3 create_tab()	15
5.3.2.4 edit()	15
5.3.2.5 edit_size()	15
5.3.2.6 greatest_value()	16
5.3.2.7 math()	16
5.3.2.8 new_tab()	16
5.3.2.9 not_empty()	17
5.3.2.10 read()	17

5.3.2.11 read_sheet()	17
5.3.2.12 reset_sheet() [1/2]	18
5.3.2.13 reset_sheet() [2/2]	18
5.3.2.14 smallest_value()	18
5.3.2.15 sum()	19
5.3.2.16 tab_disp()	19
5.3.2.17 update_tab()	19
5.3.2.18 update_tabsize()	19
5.3.2.19 write()	20
5.3.2.20 write_sheet()	20
5.3.3 Dokumentacja atrybutów składowych	20
5.3.3.1 r1	21
5.3.3.2 r2	21
5.3.3.3 tab	21
5.4 Dokumentacja klasy StringCell	21
5.4.1 Opis szczegółowy	22
5.4.2 Dokumentacja funkcji składowych	22
5.4.2.1 getNum()	22
5.4.2.2 getValue()	23
5.4.2.3 setValue()	23
5.4.3 Dokumentacja atrybutów składowych	23
5.4.3.1 stringValue	23
6 Dokumentacja plików	25
6.1 Dokumentacja pliku main.cpp	25
6.1.1 Dokumentacja funkcji	25
6.1.1.1 main()	26
6.2 Dokumentacja pliku menu.cpp	26
6.2.1 Dokumentacja funkcji	26
6.2.1.1 choice_math()	26
6.2.1.2 columnType()	27
6.2.1.3 confirm()	27
6.2.1.4 menu()	27
6.2.1.5 welcome()	27
6.3 Dokumentacja pliku menu.h	28
6.3.1 Dokumentacja funkcji	28
6.3.1.1 choice_math()	29
6.3.1.2 columnType()	29
6.3.1.3 confirm()	29
6.3.1.4 menu()	29
6.3.1.5 welcome()	29
6.4 Dokumentacja pliku operacje_na_plikach.cpp	30

6.4.1 Dokumentacja funkcji	30
6.4.1.1 del()	30
6.5 Dokumentacja pliku operacje_na_plikach.h	31
6.5.1 Dokumentacja funkcji	31
6.5.1.1 del()	31
6.6 Dokumentacja pliku README.md	32
6.7 Dokumentacja pliku tablica.cpp	32
6.7.1 Dokumentacja funkcji	32
6.7.1.1 delete_sheet()	32
6.8 Dokumentacja pliku tablica.h	33
6.8.1 Dokumentacja funkcji	33
6.8.1.1 delete_sheet()	33
6.9 Dokumentacja pliku tablica_wysw.cpp	34
6.10 Dokumentacja pliku tablica_wysw.h	34
6.10.1 Dokumentacja funkcji	35
6.10.1.1 tab_disp()	35
Indeks	37

Rozdział 1

Arkusz kalkulacyjny

1.1 1. main.cpp

- start działania programu - wywołanie funkcji menu

1.2 2. menu.cpp:

- funkcja powitalna i wybór akcji
 - utworzenie arkusza kalkulacyjnego o danych wymiarach
 - wprowadzenie/aktualizacja danych do arkusza
 - wyświetlanie zawartości arkusza
 - zmiana wymiarów arkusza
 - zapis arkusza do pliku
 - odczyt arkusza z pliku
 - usunięcie wybranego pliku z arkuszem
 - wybór operacji matematycznej na arkuszu
 - zakończenie pracy z programem

1.3 3. tablica.cpp:

- funkcja tworząca dwuwymiarową tablicę dynamiczną
- funkcja wprowadzająca/aktualizująca dane w tablicy
- funkcja aktualizująca rozmiar tablicy
 - funkcja zerująca arkusz
 - * funkcja kopiująca zawartość arkusza o starych wymiarach do arkusza o nowych wymiarach
- funkcja wysyłająca dane do wykonania zapisu arkusza do pliku
- funkcja odbierająca dane z wykonanego odczytu arkusza z pliku
- funkcja wysyłająca nazwę pliku do usunięcia oraz informację o błędzie
- funkcje realizujące operacje matematyczne na arkuszu
 - funkcja sumująca po wierszach/kolumnach
 - funkcja znajdująca wartość największą w wierszu/kolumnie
 - funkcja znajdująca wartość najmniejszą w wierszu/kolumnie
 - funkcja obliczająca wartość średnią wiersza/kolumny

1.4 4. tablica_wysw.cpp:

- funkcja wyświetlająca zawartość arkusza na ekranie

1.5 5. operacje_na_plikach.cpp:

- funkcja zapisująca zawartość arkusza do pliku
- funkcja odczytująca zawartość arkusza z pliku
- funkcja usuwająca wybrany plik z arkuszem

Rozdział 2

Indeks hierarchiczny

2.1 Hierarchia klas

Ta lista dziedziczenia posortowana jest z grubsza, choć nie całkowicie, alfabetycznie:

Cell	9
DoubleCell	11
StringCell	21
Sheet	13

Rozdział 3

Indeks klas

3.1 Lista klas

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

Cell	9
DoubleCell	11
Sheet	13
StringCell	21

Rozdział 4

Indeks plików

4.1 Lista plików

Tutaj znajduje się lista wszystkich plików z ich krótkimi opisami:

main.cpp	25
menu.cpp	26
menu.h	28
operacje_na_plikach.cpp	30
operacje_na_plikach.h	31
tablica.cpp	32
tablica.h	33
tablica_wysw.cpp	34
tablica_wysw.h	34

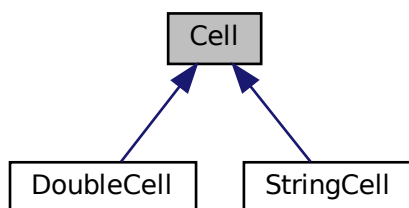
Rozdział 5

Dokumentacja klas

5.1 Dokumentacja klasy Cell

```
#include <tablica.h>
```

Diagram dziedziczenia dla Cell



Metody publiczne

- virtual bool `getNum` (void)=0
- virtual std::string `getValue` (void)=0
- virtual void `setValue` (std::string value)=0

5.1.1 Opis szczegółowy

Klasa przechowująca typy wartości w komórkach arkusza

5.1.2 Dokumentacja funkcji składowych

5.1.2.1 `getNum()`

```
virtual bool Cell::getNum (
    void ) [pure virtual]
```

Funkcja informująca czy dana komórka jest numeryczna

Zwraca

prawda albo fałsz

Implementowany w [StringCell](#) i [DoubleCell](#).

5.1.2.2 `getValue()`

```
virtual std::string Cell::getValue (
    void ) [pure virtual]
```

Getter

Implementowany w [StringCell](#) i [DoubleCell](#).

5.1.2.3 `setValue()`

```
virtual void Cell::setValue (
    std::string value ) [pure virtual]
```

Setter

Implementowany w [StringCell](#) i [DoubleCell](#).

Dokumentacja dla tej klasy została wygenerowana z pliku:

- [tablica.h](#)

5.2 Dokumentacja klasy DoubleCell

```
#include <tablica.h>
```

Diagram dziedziczenia dla DoubleCell

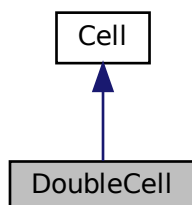
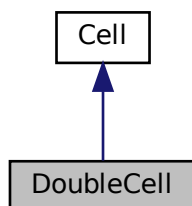


Diagram współpracy dla DoubleCell:



Metody publiczne

- std::string `getValue` (void)
- void `setValue` (std::string value)
- bool `getNum` (void)

Atrybuty prywatne

- double `doubleValue` = 0

5.2.1 Opis szczegółowy

Klasa komórki typu liczbowego

5.2.2 Dokumentacja funkcji składowych

5.2.2.1 getNum()

```
bool DoubleCell::getNum (
    void ) [virtual]
```

Funkcja informująca czy dana komórka jest numeryczna

Zwraca

prawda albo fałsz

Implementuje [Cell](#).

5.2.2.2 getValue()

```
string DoubleCell::getValue (
    void ) [virtual]
```

Getter

Implementuje [Cell](#).

5.2.2.3 setValue()

```
void DoubleCell::setValue (
    std::string value ) [virtual]
```

Setter

Implementuje [Cell](#).

5.2.3 Dokumentacja atrybutów składowych

5.2.3.1 doubleValue

```
double DoubleCell::doubleValue = 0 [private]
```

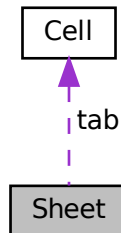
Dokumentacja dla tej klasy została wygenerowana z plików:

- [tablica.h](#)
- [tablica.cpp](#)

5.3 Dokumentacja klasy Sheet

```
#include <tablica.h>
```

Diagram współpracy dla Sheet:



Metody publiczne

- void `new_tab` (int x, int y, int *colType)
- void `update_tab` (int x, int y, `Cell` *value)
- void `update_tabsize` (int nr1, int nr2, int *colType)
- void `tab_disp` (void)
- void `reset_sheet` (void)
- bool `read_sheet` (std::string file_name)
- bool `write_sheet` (std::string file_name)
- `Cell` * `edit` (int *wskx, int *wsky)
- int * `edit_size` (int *wsknewr1, int *wsknewr2, int *colType)
- void `math` (int dim_choice, int math_choice)

Metody prywatne

- `Cell` *** `create_tab` (int r1, int r2, int *colType)
- void `copy_tab` (`Cell` ***t1, `Cell` ***t2, int r1, int r2)
- void `reset_sheet` (`Cell` ***t, int r1, int r2)
- bool `read` (std::string file_name)
- bool `write` (std::string file_name)
- `Cell` ** `sum` (int dim_choice)
- `Cell` * `greatest_value` (int dim_choice, int x)
- `Cell` * `smallest_value` (int dim_choice, int x)
- `Cell` * `average_value` (int dim_choice, int x)
- double `not_empty` (int dim_choice, int x)

Atrybuty prywatne

- `Cell` *** `tab`
- int r1 = 0
- int r2 = 0

5.3.1 Opis szczegółowy

Klasa przechowująca dane o arkuszu

5.3.2 Dokumentacja funkcji składowych

5.3.2.1 average_value()

```
Cell * Sheet::average_value (
    int dim_choice,
    int x ) [private]
```

Funkcja realizująca operację obliczania wartości średniej w wierszach/kolumnach

Parametry

in	<i>dim_choice</i>	zmienna przechowująca dane o wybranym wymiarze
in	<i>x</i>	zmienna przechowująca numer wybranego wiersza/kolumny

Zwraca

obiekt z wynikiem

5.3.2.2 copy_tab()

```
void Sheet::copy_tab (
    Cell *** t1,
    Cell *** t2,
    int r1,
    int r2 ) [private]
```

Funkcja kopiująca zawartość między dwiema tablicami

Parametry

in, out	<i>t1</i>	wskaźnik do dwuwymiarowej tablicy dynamicznej obiektów klasy Cell (arkusz) docelowej
in	<i>t2</i>	wskaźnik do dwuwymiarowej tablicy dynamicznej obiektów klasy Cell (arkusz) kopiowanej
in	<i>r1</i>	pierwszy wymiar tablicy docelowej (wiersze)
in	<i>r2</i>	drugi wymiar tablicy docelowej (kolumny)

5.3.2.3 create_tab()

```
Cell *** Sheet::create_tab (
    int r1,
    int r2,
    int * colType ) [private]
```

Funkcja tworząca arkusz kalkulacyjny o danych wymiarach

Parametry

in	<i>r1</i>	pierwszy wymiar arkusza (wiersze)
in	<i>r2</i>	drugi wymiar arkusza (kolumny)
in	<i>colType</i>	wskaźnik do tablicy przechowującej dane o typie kolumn

Zwraca

wskaźnik do dwuwymiarowej tablicy dynamicznej obiektów klasy [Cell](#) (arkusz)

5.3.2.4 edit()

```
Cell * Sheet::edit (
    int * wskx,
    int * wsky )
```

Funkcja pobierająca współrzędne komórki i wartość od użytkownika

Parametry

in, out	<i>wskx</i>	wskaźnik do zmiennej odpowiedzialnej za numer wiersza
in, out	<i>wsky</i>	wskaźnik do zmiennej odpowiedzialnej za numer kolumny

5.3.2.5 edit_size()

```
int * Sheet::edit_size (
    int * wsknewr1,
    int * wsknewr2,
    int * colType )
```

Funkcja pobierająca nowe rozmiary arkusza od użytkownika

Parametry

in, out	<i>wsknewr1</i>	wskaźnik do zmiennej odpowiedzialnej za nowy rozmiar arkusza (wiersze)
in, out	<i>wsknewr2</i>	wskaźnik do zmiennej odpowiedzialnej za nowy rozmiar arkusza (kolumny)

5.3.2.6 greatest_value()

```
Cell * Sheet::greatest_value (
    int dim_choice,
    int x ) [private]
```

Funkcja realizująca operację znajdowania wartości największej w wierszach/kolumnach

Parametry

in	<i>dim_choice</i>	zmienna przechowująca dane o wybranym wymiarze
in	<i>x</i>	zmienna przechowująca numer wybranego wiersza/kolumny

Zwraca

obiekt z wynikiem

5.3.2.7 math()

```
void Sheet::math (
    int dim_choice,
    int math_choice )
```

Funkcja realizująca poszczególne działania matematyczne

Parametry

in	<i>dim_choice</i>	zmienna przechowująca dane o wybranym wymiarze
in	<i>math_choice</i>	zmienna przechowująca dane o wybranym działaniu matematycznym

5.3.2.8 new_tab()

```
void Sheet::new_tab (
    int x,
    int y,
    int * colType )
```

Funkcja tworząca nowy arkusz

Parametry

in	<i>x</i>	pierwszy wymiar arkusza (wiersze)
in	<i>y</i>	drugi wymiar arkusza (kolumny)
in	<i>colType</i>	

5.3.2.9 not_empty()

```
double Sheet::not_empty (
    int dim_choice,
    int x ) [private]
```

Funkcja zwracająca liczbę niepustych komórek danego wiersza/kolumny

Parametry

in	<i>dim_choice</i>	zmienna przechowująca dane o wybranym wymiarze
in	<i>x</i>	zmienna przechowująca numer wybranego wiersza/kolumny

Zwraca

liczba niepustych komórek danego wiersza/kolumny

5.3.2.10 read()

```
bool Sheet::read (
    std::string file_name ) [private]
```

Funkcja odczytująca zawartość arkusza z pliku

Parametry

in	<i>file_name</i>	nazwa pliku do odczytu
----	------------------	------------------------

Zwraca

1 gdy plik istnieje, 0 gdy plik nie istnieje

5.3.2.11 read_sheet()

```
bool Sheet::read_sheet (
    std::string file_name )
```

Funkcja odczytująca arkusz z pliku

Parametry

in	<i>file_name</i>	nazwa pliku do odczytu
----	------------------	------------------------

Zwraca

1 gdy odczyt został wykonany poprawnie, 0 gdy odczyt nie został wykonany

5.3.2.12 reset_sheet() [1/2]

```
void Sheet::reset_sheet (
    Cell *** t,
    int r1,
    int r2 ) [private]
```

Funkcja zerująca zawartość arkusza

Parametry

in, out	<i>t</i>	wskaźnik do dwuwymiarowej tablicy dynamicznej obiektów klasy Cell (arkusz)
in	<i>r1</i>	pierwszy wymiar tablicy (wiersze)
in	<i>r2</i>	drugi wymiar arkusza (kolumny)

5.3.2.13 reset_sheet() [2/2]

```
void Sheet::reset_sheet (
    void )
```

Funkcja zerująca zawartość arkusza

5.3.2.14 smallest_value()

```
Cell * Sheet::smallest_value (
    int dim_choice,
    int x ) [private]
```

Funkcja realizująca operację znajdowania wartości najmniejszej w wierszach/kolumnach

Parametry

in	<i>dim_choice</i>	zmienna przechowująca dane o wybranym wymiarze
in	<i>x</i>	zmienna przechowująca numer wybranego wiersza/kolumny

Zwraca

obiekt z wynikiem

5.3.2.15 sum()

```
Cell ** Sheet::sum (
    int dim_choice ) [private]
```

Funkcja realizująca operację sumowania wierszy/kolumn

Parametry

in	<i>dim_choice</i>	zmienna przechowująca dane o wybranym wymiarze
----	-------------------	--

Zwraca

adres obiektu tablicy z wynikiem

5.3.2.16 tab_disp()

```
void Sheet::tab_disp (
    void )
```

Funkcja wyświetlająca zawartość arkusza na ekranie

5.3.2.17 update_tab()

```
void Sheet::update_tab (
    int x,
    int y,
    Cell * value )
```

Funkcja aktualizująca zawartość arkusza

Parametry

in	<i>x</i>	numer wiersza w arkuszu
in	<i>y</i>	numer kolumny w arkuszu
in	<i>value</i>	wartość wpisywana do wskazanej komórki arkusza

5.3.2.18 update_tabsize()

```
void Sheet::update_tabsize (
    int nr1,
    int nr2,
    int * colType )
```

Funkcja aktualizująca rozmiar arkusza

Parametry

<i>in, out</i>	<i>sheet</i>	wskaźnik do zmiennej strukturalnej przechowującej arkusz i jego wymiary
<i>in</i>	<i>nr1</i>	nowy pierwszy wymiar arkusza (wiersze)
<i>in</i>	<i>nr2</i>	nowy drugi wymiar arkusza (kolumny)

5.3.2.19 write()

```
bool Sheet::write (
    std::string file_name ) [private]
```

Funkcja zapisująca zawartość arkusza do pliku

Parametry

<i>in, out</i>	<i>file_name</i>	nazwa pliku docelowego
----------------	------------------	------------------------

Zwraca

1 gdy zapis został wykonany poprawnie, 0 gdy zapis nie został wykonany

5.3.2.20 write_sheet()

```
bool Sheet::write_sheet (
    std::string file_name )
```

Funkcja zapisująca arkusz do pliku

Parametry

<i>in, out</i>	<i>sheet</i>	zmienna strukturalna przechowująca arkusz i jego wymiary
<i>in, out</i>	<i>file_name</i>	nazwa pliku docelowego

Zwraca

1 gdy zapis został wykonany poprawnie, 0 gdy zapis nie został wykonany

5.3.3 Dokumentacja atrybutów składowych

5.3.3.1 r1

```
int Sheet::r1 = 0 [private]
```

Pierwszy wymiar arkusza (wiersze)

5.3.3.2 r2

```
int Sheet::r2 = 0 [private]
```

Drugi wymiar arkusza (kolumny)

5.3.3.3 tab

```
Cell*** Sheet::tab [private]
```

Wskaźnik do dwuwymiarowej tablicy dynamicznej obiektów klasy [Cell](#) (arkusz)

Dokumentacja dla tej klasy została wygenerowana z plików:

- [tablica.h](#)
- [menu.cpp](#)
- [operacje_na_plikach.cpp](#)
- [tablica.cpp](#)
- [tablica_wysw.cpp](#)

5.4 Dokumentacja klasy StringCell

```
#include <tablica.h>
```

Diagram dziedziczenia dla StringCell

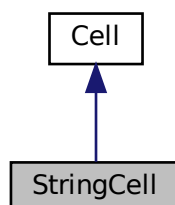
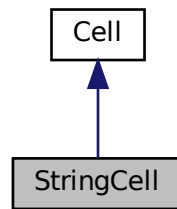


Diagram współpracy dla StringCell:



Metody publiczne

- std::string [getValue](#) (void)
- void [setValue](#) (std::string value)
- bool [getNum](#) (void)

Atrybuty prywatne

- std::string [stringValue](#) = "-"

5.4.1 Opis szczegółowy

Klasa komórki typu tekstowego

5.4.2 Dokumentacja funkcji składowych

5.4.2.1 [getNum\(\)](#)

```
bool StringCell::getNum (  
    void ) [virtual]
```

Funkcja informująca czy dana komórka jest numeryczna

Zwraca

prawda albo fałsz

Implementuje [Cell](#).

5.4.2.2 getValue()

```
string StringCell::getValue (
    void ) [virtual]
```

Getter

Implementuje [Cell](#).

5.4.2.3 setValue()

```
void StringCell::setValue (
    std::string value ) [virtual]
```

Setter

Implementuje [Cell](#).

5.4.3 Dokumentacja atrybutów składowych

5.4.3.1 stringValue

```
std::string StringCell::stringValue = "-" [private]
```

Dokumentacja dla tej klasy została wygenerowana z plików:

- [tablica.h](#)
- [tablica.cpp](#)

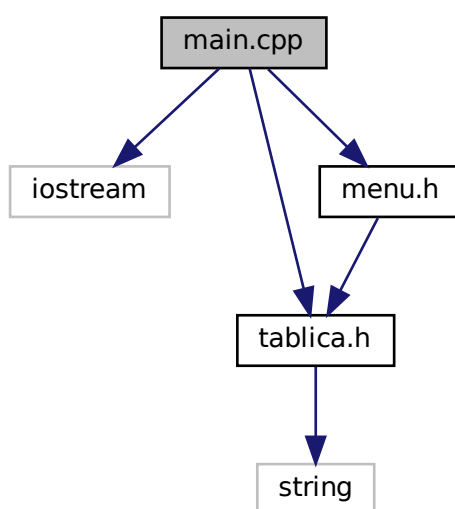
Rozdział 6

Dokumentacja plików

6.1 Dokumentacja pliku main.cpp

```
#include <iostream>
#include "tablica.h"
#include "menu.h"
```

Wykres zależności załączania dla main.cpp:



Funkcje

- int `main` (void)

6.1.1 Dokumentacja funkcji

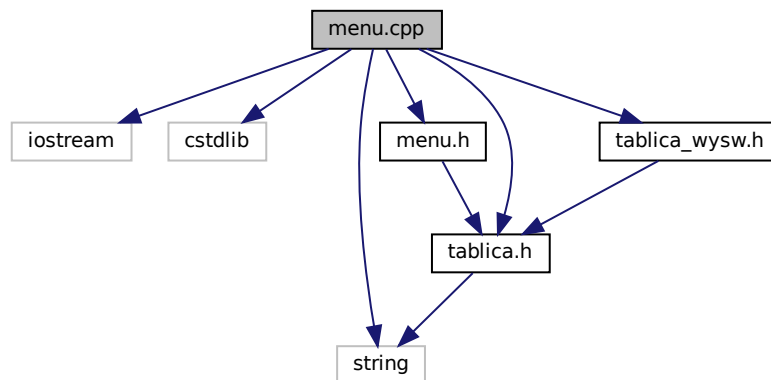
6.1.1.1 main()

```
int main (
    void )
```

6.2 Dokumentacja pliku menu.cpp

```
#include <iostream>
#include <cstdlib>
#include <string>
#include "menu.h"
#include "tablica.h"
#include "tablica_wysw.h"
```

Wykres zależności załączania dla menu.cpp:



Funkcje

- void [welcome](#) (void)
- void [menu](#) ([Sheet](#) sheet)
- bool [confirm](#) (void)
- int * [columnType](#) (int x)
- void [choice_math](#) (int *dim_choice, int *math_choice)

6.2.1 Dokumentacja funkcji

6.2.1.1 choice_math()

```
void choice_math (
    int * dim_choice,
    int * math_choice )
```

Funkcja pobierająca informacje o wyborze wymiaru oraz rodzaju działania matematycznego

Parametry

in, out	<i>dim_choice</i>	wskaźnik do zmiennej służącej do wyboru wymiaru
in, out	<i>math_choice</i>	wskaźnik do zmiennej służącej do wyboru działania matematycznego

6.2.1.2 columnType()

```
int* columnType (
    int x )
```

Funkcja pobierająca od użytkownika typ kolumn arkusza

Zwraca

prawda 0, gdy kolumna liczbowa, 1 gdy kolumna tekstowa

6.2.1.3 confirm()

```
bool confirm (
    void )
```

Funkcja oczekująca potwierdzenia działania przez użytkownika

6.2.1.4 menu()

```
void menu (
    Sheet sheet )
```

Funkcja odpowiedzialna na za działanie menu użytkownika

6.2.1.5 welcome()

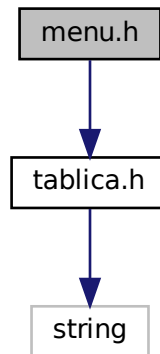
```
void welcome (
    void )
```

Funkcja wyświetlająca powitanie oraz dostępne opcje w programie

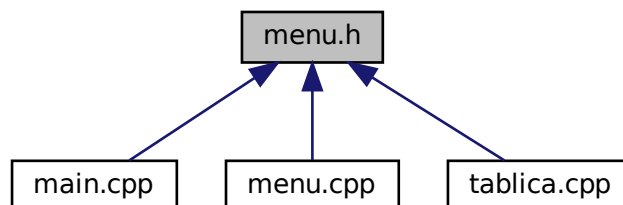
6.3 Dokumentacja pliku menu.h

```
#include "tablica.h"
```

Wykres zależności załączania dla menu.h:



Ten wykres pokazuje, które pliki bezpośrednio lub pośrednio załączają ten plik:



Funkcje

- void [welcome](#) (void)
- void [menu](#) ([Sheet](#) sheet)
- bool [confirm](#) (void)
- int * [columnType](#) (int x)
- void [choice_math](#) (int *dim_choice, int *math_choice)

6.3.1 Dokumentacja funkcji

6.3.1.1 choice_math()

```
void choice_math (
    int * dim_choice,
    int * math_choice )
```

Funkcja pobierająca informacje o wyborze wymiaru oraz rodzaju działania matematycznego

Parametry

in, out	<i>dim_choice</i>	wskaźnik do zmiennej służącej do wyboru wymiaru
in, out	<i>math_choice</i>	wskaźnik do zmiennej służącej do wyboru działania matematycznego

6.3.1.2 columnType()

```
int* columnType (
    int x )
```

Funkcja pobierająca od użytkownika typ kolumn arkusza

Zwraca

prawda 0, gdy kolumna liczbowa, 1 gdy kolumna tekstowa

6.3.1.3 confirm()

```
bool confirm (
    void )
```

Funkcja oczekująca potwierdzenia działania przez użytkownika

6.3.1.4 menu()

```
void menu (
    Sheet sheet )
```

Funkcja odpowiedzialna na za działanie menu użytkownika

6.3.1.5 welcome()

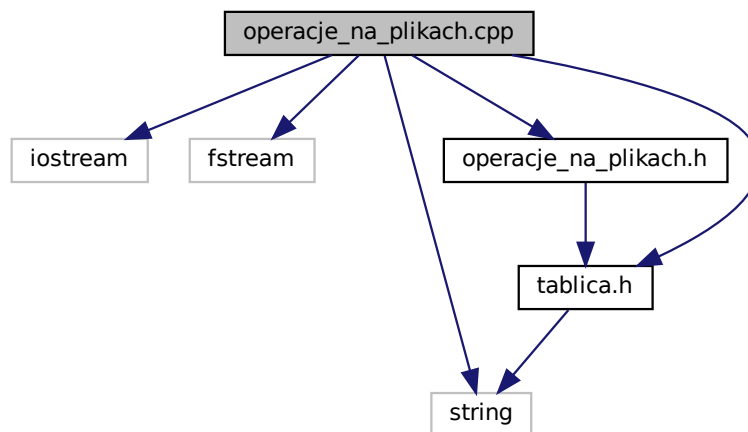
```
void welcome (
    void )
```

Funkcja wyświetlająca powitanie oraz dostępne opcje w programie

6.4 Dokumentacja pliku operacje_na_plikach.cpp

```
#include <iostream>
#include <fstream>
#include <string>
#include "operacje_na_plikach.h"
#include "tablica.h"
```

Wykres zależności załączania dla operacje_na_plikach.cpp:



Funkcje

- bool `del` (string file_name)

6.4.1 Dokumentacja funkcji

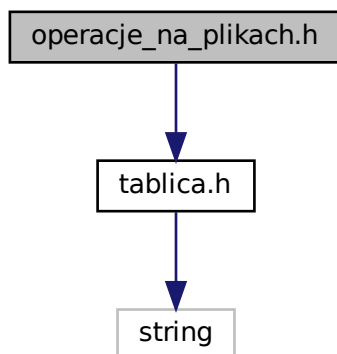
6.4.1.1 del()

```
bool del (
    string file_name )
```

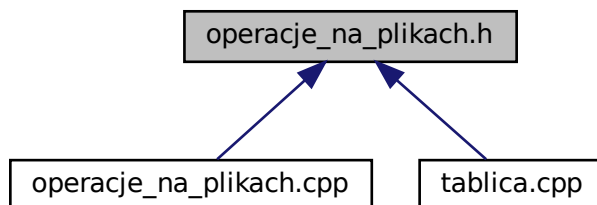
6.5 Dokumentacja pliku operacje_na_plikach.h

```
#include "tablica.h"
```

Wykres zależności załączania dla operacje_na_plikach.h:



Ten wykres pokazuje, które pliki bezpośrednio lub pośrednio załączają ten plik:



Funkcje

- bool `del` (std::string file_name)

6.5.1 Dokumentacja funkcji

6.5.1.1 del()

```
bool del (
    std::string file_name )
```

Funkcja usuwająca wybrany plik *.txt

Parametry

in	<i>file_name</i>	nazwa pliku do usunięcia
----	------------------	--------------------------

Zwraca

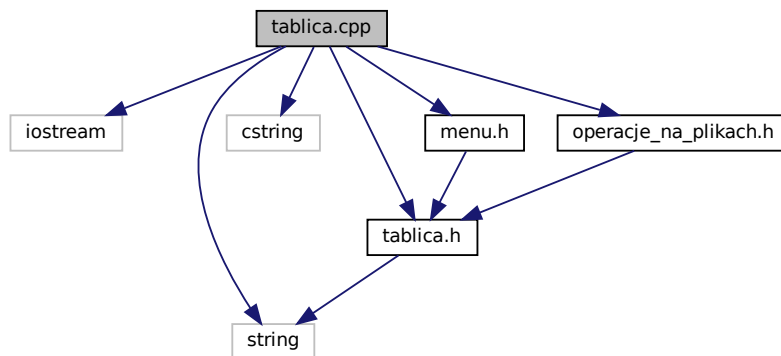
1 gdy plik został usunięty, 0 gdy plik nie został usunięty

6.6 Dokumentacja pliku README.md

6.7 Dokumentacja pliku tablica.cpp

```
#include <iostream>
#include <string>
#include <cstring>
#include "tablica.h"
#include "menu.h"
#include "operacje_na_plikach.h"
```

Wykres zależności załączania dla tablica.cpp:

**Funkcje**

- bool `delete_sheet` (string *file_name*)

6.7.1 Dokumentacja funkcji

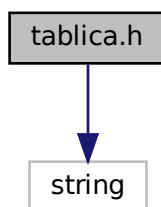
6.7.1.1 `delete_sheet()`

```
bool delete_sheet (
    string file_name )
```

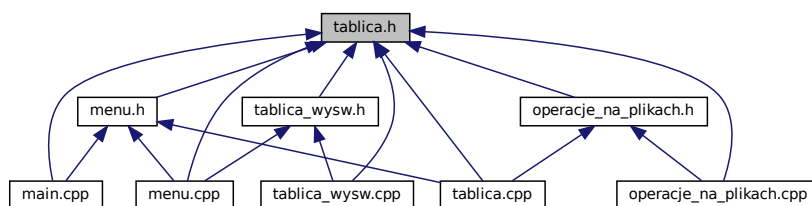
6.8 Dokumentacja pliku tablica.h

```
#include <string>
```

Wykres zależności załączania dla tablica.h:



Ten wykres pokazuje, które pliki bezpośrednio lub pośrednio załączają ten plik:



Komponenty

- class `Cell`
- class `DoubleCell`
- class `StringCell`
- class `Sheet`

Funkcje

- bool `delete_sheet` (std::string file_name)

6.8.1 Dokumentacja funkcji

6.8.1.1 delete_sheet()

```
bool delete_sheet (
    std::string file_name )
```

Funkcja usuwająca wybrany plik z arkuszem

Parametry

in	file_name	nazwa pliku do usunięcia
----	-----------	--------------------------

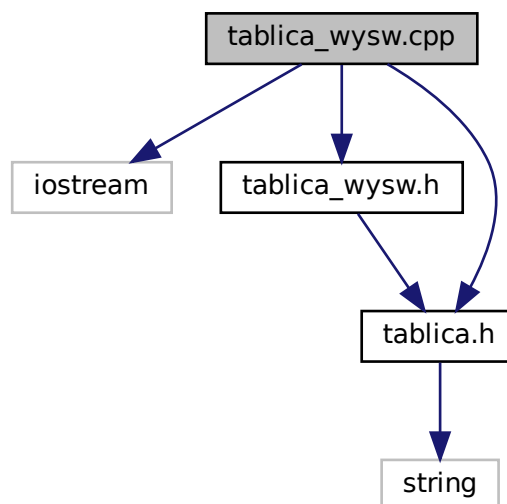
Zwraca

1 gdy plik został usunięty, 0 gdy plik nie został usunięty

6.9 Dokumentacja pliku tablica_wysw.cpp

```
#include <iostream>
#include "tablica_wysw.h"
#include "tablica.h"
```

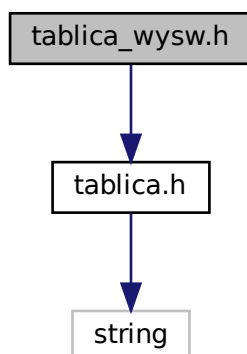
Wykres zależności załączania dla tablica_wysw.cpp:



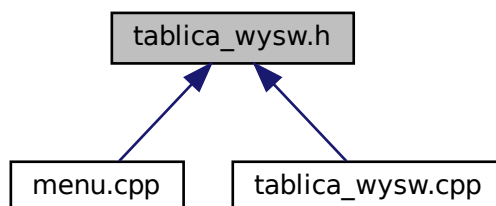
6.10 Dokumentacja pliku tablica_wysw.h

```
#include "tablica.h"
```


Wykres zależności załączania dla tablica_wysw.h:



Ten wykres pokazuje, które pliki bezpośrednio lub pośrednio załączają ten plik:



Funkcje

- void `tab_disp` (void)

6.10.1 Dokumentacja funkcji

6.10.1.1 `tab_disp()`

```
void tab_disp (  
    void )
```

Funkcja wyświetlająca zawartość arkusza na ekranie

Indeks

average_value
 Sheet, [14](#)

Cell, [9](#)
 getNum, [9](#)
 getValue, [10](#)
 setValue, [10](#)

choice_math
 menu.cpp, [26](#)
 menu.h, [28](#)

columnType
 menu.cpp, [27](#)
 menu.h, [29](#)

confirm
 menu.cpp, [27](#)
 menu.h, [29](#)

copy_tab
 Sheet, [14](#)

create_tab
 Sheet, [14](#)

del
 operacje_na_plikach.cpp, [30](#)
 operacje_na_plikach.h, [31](#)

delete_sheet
 tablica.cpp, [32](#)
 tablica.h, [33](#)

DoubleCell, [11](#)
 doubleValue, [12](#)
 getNum, [12](#)
 getValue, [12](#)
 setValue, [12](#)

doubleValue
 DoubleCell, [12](#)

edit
 Sheet, [15](#)

edit_size
 Sheet, [15](#)

getNum
 Cell, [9](#)
 DoubleCell, [12](#)
 StringCell, [22](#)

getValue
 Cell, [10](#)
 DoubleCell, [12](#)
 StringCell, [22](#)

greatest_value
 Sheet, [16](#)

main
 main.cpp, [25](#)
main.cpp, [25](#)
 main, [25](#)

math
 Sheet, [16](#)

menu
 menu.cpp, [27](#)
 menu.h, [29](#)
menu.cpp, [26](#)
 choice_math, [26](#)
 columnType, [27](#)
 confirm, [27](#)
 menu, [27](#)
 welcome, [27](#)
menu.h, [28](#)
 choice_math, [28](#)
 columnType, [29](#)
 confirm, [29](#)
 menu, [29](#)
 welcome, [29](#)

new_tab
 Sheet, [16](#)

not_empty
 Sheet, [17](#)

operacje_na_plikach.cpp, [30](#)
 del, [30](#)
operacje_na_plikach.h, [31](#)
 del, [31](#)

r1
 Sheet, [20](#)

r2
 Sheet, [21](#)

read
 Sheet, [17](#)

read_sheet
 Sheet, [17](#)

README.md, [32](#)

reset_sheet
 Sheet, [18](#)

setValue
 Cell, [10](#)
 DoubleCell, [12](#)
 StringCell, [23](#)
Sheet, [13](#)
 average_value, [14](#)

- copy_tab, [14](#)
- create_tab, [14](#)
- edit, [15](#)
- edit_size, [15](#)
- greatest_value, [16](#)
- math, [16](#)
- new_tab, [16](#)
- not_empty, [17](#)
- r1, [20](#)
- r2, [21](#)
- read, [17](#)
- read_sheet, [17](#)
- reset_sheet, [18](#)
- smallest_value, [18](#)
- sum, [18](#)
- tab, [21](#)
- tab_disp, [19](#)
- update_tab, [19](#)
- update_tabsize, [19](#)
- write, [20](#)
- write_sheet, [20](#)
- smallest_value
 - Sheet, [18](#)
- StringCell, [21](#)
 - getNum, [22](#)
 - getValue, [22](#)
 - setValue, [23](#)
 - stringValue, [23](#)
- stringValue
 - StringCell, [23](#)
- sum
 - Sheet, [18](#)
- tab
 - Sheet, [21](#)
- tab_disp
 - Sheet, [19](#)
 - tablica_wysw.h, [35](#)
- tablica.cpp, [32](#)
 - delete_sheet, [32](#)
- tablica.h, [33](#)
 - delete_sheet, [33](#)
- tablica_wysw.cpp, [34](#)
- tablica_wysw.h, [34](#)
 - tab_disp, [35](#)
- update_tab
 - Sheet, [19](#)
- update_tabsize
 - Sheet, [19](#)
- welcome
 - menu.cpp, [27](#)
 - menu.h, [29](#)
- write
 - Sheet, [20](#)
- write_sheet
 - Sheet, [20](#)