

OwnExcel

Generated by Doxygen 1.8.17

1 Programowanie Obiektowe	1
1.0.0.1 Projekt arkusza kalkulacyjnego	1
1.1 Interfejs użytkownika	1
1.2 Operacyjna	1
1.2.0.1 obliczenia i zmiany rozmiaru tablicy:	1
1.2.0.2 prezentacja oraz zapisywanie:	2
2 Hierarchical Index	3
2.1 Class Hierarchy	3
3 Class Index	5
3.1 Class List	5
4 Class Documentation	7
4.1 Array Class Reference	7
4.1.1 Detailed Description	8
4.1.2 Constructor & Destructor Documentation	8
4.1.2.1 Array()	8
4.1.3 Member Function Documentation	8
4.1.3.1 average()	8
4.1.3.2 changeValue()	8
4.1.3.3 columns()	10
4.1.3.4 divide()	10
4.1.3.5 getMaxLengthValue()	10
4.1.3.6 getNumberFromSheet()	11
4.1.3.7 getStringFromSheet()	11
4.1.3.8 loadDataFromFile()	11
4.1.3.9 multiplication()	12
4.1.3.10 resizeSheet()	12
4.1.3.11 rows()	12
4.1.3.12 saveDataToFile()	13
4.1.3.13 subtract()	13
4.1.3.14 sum()	13
4.1.3.15 to_string_with_precision()	14
4.2 ArrayDisplay Class Reference	14
4.2.1 Detailed Description	14
4.2.2 Member Function Documentation	14
4.2.2.1 Display()	14
4.2.2.2 to_string_with_precision()	15
4.3 Cell Class Reference	15
4.3.1 Member Function Documentation	16
4.3.1.1 changeValue() [1/2]	16
4.3.1.2 changeValue() [2/2]	17

4.3.1.3	getValue()	17
4.3.2	Member Data Documentation	17
4.3.2.1	Value	17
4.4	CellValue Class Reference	18
4.4.1	Member Function Documentation	18
4.4.1.1	areDecimalOperationsAllowed()	18
4.4.1.2	getDecimalValue()	18
4.4.1.3	getTextValue()	19
4.5	DecimalCell Class Reference	19
4.5.1	Member Function Documentation	20
4.5.1.1	changeValue() [1/2]	20
4.5.1.2	changeValue() [2/2]	20
4.5.1.3	getValue()	21
4.6	Identifier Class Reference	21
4.7	Menu Class Reference	21
4.7.1	Detailed Description	22
4.7.2	Member Function Documentation	22
4.7.2.1	alert()	22
4.7.2.2	clear()	22
4.7.2.3	getCell()	23
4.7.2.4	getch()	23
4.7.2.5	getIdentifier()	23
4.7.2.6	getNumber()	24
4.7.2.7	message()	24
4.7.2.8	showFunctions()	24
4.8	OperationHandler Class Reference	25
4.8.1	Detailed Description	25
4.8.2	Member Function Documentation	25
4.8.2.1	decimalOperation()	25
4.9	TextCell Class Reference	26
4.9.1	Member Function Documentation	26
4.9.1.1	changeValue() [1/2]	27
4.9.1.2	changeValue() [2/2]	27
4.9.1.3	getValue()	27

Index	29
--------------	-----------

Chapter 1

Programowanie Obiektowe

1.0.0.1 Projekt arkusza kalkulacyjnego

Prosty arkusz kalkulacyjny obsługiwany w konsoli.

1.1 Interfejs użytkownika

- [] Wyświetlanie informacji na temat funkcji
- [] prezentacja arkusza
- [] możliwość zmiany rozmiaru tablicy
- [] możliwość zmiany danych w arkuszu
- [] możliwość wykonywania operacji na arkuszu
- [] możliwość wrócenia do menu

1.2 Operacyjna

1.2.0.1 obliczenia i zmiany rozmiaru tablicy:

- [] wprowadzanie danych do arkusza
- [] dodawanie komórek
- [] odejmowanie komórek
- [] mnożenie komórek
- [] dzielenie komórek
- [] średnia z komórek
- [] zapisywanie wyniku do konkretnej komórki
- [] zmiana rozmiaru tablicy bez utraty danych

1.2.0.2 prezentacja oraz zapisywanie:

- [] prezentacja tablicy
- [] zapisywanie wyniku do konkretnej komórki
- [] odczytywanie danych z pliku
- [] zapisywanie danych do pliku

Chapter 2

Hierarchical Index

2.1 Class Hierarchy

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

Array	7
ArrayDisplay	14
Cell	15
DecimalCell	19
TextCell	26
CellValue	18
Identifier	21
Menu	21
OperationHandler	25

Chapter 3

Class Index

3.1 Class List

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

Array	This is a class handling all operations on sheet	7
ArrayDisplay	This is a class handling displaying sheet	14
Cell	15
CellValue	18
DecimalCell	19
Identifier	21
Menu	This is a class handling menu operations	21
OperationHandler	This is a class handling menu operations	25
TextCell	26

Chapter 4

Class Documentation

4.1 Array Class Reference

This is a class handling all operations on sheet.

```
#include <array.h>
```

Public Member Functions

- **Array** (int **columns**, int **rows**)
Constructor of class.
- int **columns** () const
- int **rows** () const
- float **sum** (**Identifier** *identifiers, int length)
func that return sum of values depends on each one position in sheet and returns result.
- float **subtract** (**Identifier** minued, **Identifier** subtrahend)
func that subtract two values from sheet and returns result.
- float **divide** (**Identifier** divisor, **Identifier** dividend)
func that divide two values from sheet and returns result.
- float **multiplication** (**Identifier** *identifiers, int length)
func that return multiplication of values depends on each one position in sheet and returns result.
- float **average** (**Identifier** *identifiers, int length)
func that return average value of values depends on each one position in sheet and returns result.
- void **changeValue** (**Identifier** identifier, **CellValue** value)
func that change value single position in sheet and return true or false depends on result.
- void **resizeSheet** (int **columns**, int **rows**)
func that resize sheet with no data lose.
- float **getNumberFromSheet** (**Identifier** identifier)
func that return single value from sheet depends on it position.
- **CellValue** **getCellFromSheet** (**Identifier** identifier)
- std::string **getStringFromSheet** (**Identifier** identifier)
func that return single value from sheet depends on it position.
- void **saveDataToFile** ()
func that save data from array to file (NOT IMPLEMENTED)
- void **loadDataFromFile** ()
func that load data from file (NOT IMPLEMENTED)
- int **getMaxLengthValue** (int precision)
- template<typename T >
std::string **to_string_with_precision** (const T a_value, const int n)

4.1.1 Detailed Description

This is a class handling all operations on sheet.

4.1.2 Constructor & Destructor Documentation

4.1.2.1 Array()

```
Array::Array (
    int columns,
    int rows ) [inline]
```

Constructor of class.

Just in time of creating array object constructor configure sheet variable to proper value of rows and columns. Construction also configure columns and rows properties.

4.1.3 Member Function Documentation

4.1.3.1 average()

```
float Array::average (
    Identifier * identifiers,
    int length )
```

func that return average value of values depends on each one position in sheet and returns result.

Parameters

in	<i>identifiers</i>	- array that holds position of cells.
in	<i>length</i>	- length of array.

Returns

- float value

4.1.3.2 changeValue()

```
void Array::changeValue (
    Identifier identifier,
    CellValue value )
```

func that change value single position in sheet and return true or false depends on result.

Parameters

in	<i>identifier</i>	- position of cell that gonna be changed.
in	<i>value</i>	- float value to write up.

Returns

- nothing or exception

4.1.3.3 columns()

```
int Array::columns ( ) const
```

Returns

numbers of columns

4.1.3.4 divide()

```
float Array::divide (
    Identifier divisor,
    Identifier dividend )
```

func that divide two values from sheet and returns result.

Parameters

in	<i>divisor</i>	- first number identifier
in	<i>dividend</i>	- second number identifier

Returns

- float value or exception.

4.1.3.5 getMaxLengthValue()

```
int Array::getMaxLengthValue (
    int precision )
```

func search max value stored in sheet.

Parameters

<i>precision</i>	- precision to round off
------------------	--------------------------

Returns

- max value as float

4.1.3.6 getNumberFromSheet()

```
float Array::getNumberFromSheet (
    Identifier identifier )
```

func that return single value from sheet depends on it position.

Parameters

<i>identifier</i>	- position of cell that value func gonna return.
-------------------	--

Returns

return float or exception

4.1.3.7 getStringFromSheet()

```
std::string Array::getStringFromSheet (
    Identifier identifier )
```

func that return single value from sheet depends on it position.

Parameters

<i>identifier</i>	- position of cell that value func gonna return.
-------------------	--

Returns

return string or exception

4.1.3.8 loadDataFromFile()

```
void Array::loadDataFromFile ( )
```

func that load data from file (NOT IMPLEMENTED)

Returns

- nothing

4.1.3.9 multiplication()

```
float Array::multiplication (
    Identifier * identifiers,
    int length )
```

func that return multiplication of values depends on each one position in sheet and returns result.

Parameters

in	<i>identifiers</i>	- array that holds position of cells.
in	<i>length</i>	- length of array.

Returns

- float value

4.1.3.10 resizeSheet()

```
void Array::resizeSheet (
    int columns,
    int rows )
```

func that resize sheet with no data lose.

Parameters

in	<i>columns</i>	- new numbers of columns in sheet.
in	<i>rows</i>	- new numbers of rows in sheet.

Returns

return nothing or exception

4.1.3.11 rows()

```
int Array::rows ( ) const
```

Returns

number of rows

4.1.3.12 saveDataToFile()

```
void Array::saveDataToFile ( )
```

func that save data from array to file (NOT IMPLEMENTED)

Returns

- nothing

4.1.3.13 subtract()

```
float Array::subtract (
    Identifier minued,
    Identifier subtrahend )
```

func that subtract two values from sheet and returns result.

Parameters

in	<i>minued</i>	- first number identifier
in	<i>subtrahend</i>	- second number identifier

Returns

- float value or exception

4.1.3.14 sum()

```
float Array::sum (
    Identifier * identifiers,
    int length )
```

func that return sum of values depends on each one position in sheet and returns result.

Parameters

in	<i>identifiers</i>	- array that holds position of cells.
in	<i>length</i>	- holds length of identifiers array

Returns

- float value or exception

4.1.3.15 to_string_with_precision()

```
template<typename T >
std::string Array::to_string_with_precision (
    const T a_value,
    const int n )
```

func search max value stored in sheet.

Parameters

<i>n</i>	- precision to round off
<i>a_value</i>	- value to round off

Returns

- rounded value as string

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

- Header Files/array.h
- Source Files/array.cpp

4.2 ArrayDisplay Class Reference

This is a class handling displaying sheet.

```
#include <array_display.h>
```

Public Member Functions

- void [Display](#) ([Array](#) *arr)
func that display sheet.
- template<typename T >
std::string [to_string_with_precision](#) (const T a_value, const int n=6)

4.2.1 Detailed Description

This is a class handling displaying sheet.

4.2.2 Member Function Documentation

4.2.2.1 Display()

```
void ArrayDisplay::Display (
    Array * arr )
```

func that display sheet.

Parameters

in	Array	- 2D array that holds sheet.
----	-----------------------	------------------------------

Returns

- void or exception

4.2.2.2 to_string_with_precision()

```
template<typename T >
std::string ArrayDisplay::to_string_with_precision (
    const T a_value,
    const int n = 6 )
```

func search max value stored in sheet.

Parameters

<i>n</i>	- precision to round off
<i>a_value</i>	- value to round off

Returns

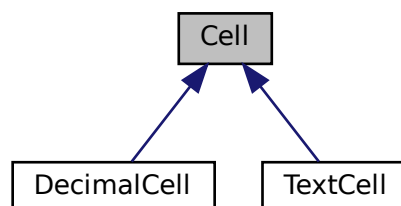
- rounded value as string

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

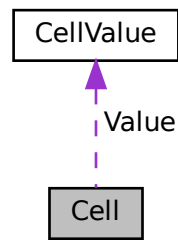
- Header Files/array_display.h
- Source Files/array_display.cpp

4.3 Cell Class Reference

Inheritance diagram for Cell:



Collaboration diagram for Cell:



Public Member Functions

- **Cell** (std::string val)
- **Cell** (float val)
- virtual void [changeValue](#) (std::string val)=0
- virtual void [changeValue](#) (float val)=0
- virtual [CellValue](#) [getValue](#) ()=0

Protected Attributes

- [CellValue](#) [Value](#)

4.3.1 Member Function Documentation

4.3.1.1 [changeValue\(\)](#) [1/2]

```
virtual void Cell::changeValue (
    float val ) [pure virtual]
```

func that change value

Parameters

<i>value</i>	- param that hold value
--------------	-------------------------

Returns

- void

Implemented in [DecimalCell](#), and [TextCell](#).

4.3.1.2 `changeValue()` [2/2]

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

func that change value

Parameters

<i>value</i>	- param that hold value
--------------	-------------------------

Returns

- void

Implemented in [DecimalCell](#), and [TextCell](#).

4.3.1.3 `getValue()`

```
virtual CellValue Cell::getValue ( ) [pure virtual]
```

func that return value as [CellValue](#)

Returns

- [CellValue](#)

Implemented in [DecimalCell](#), and [TextCell](#).

4.3.2 Member Data Documentation

4.3.2.1 Value

```
CellValue Cell::Value [protected]
```

Parameters

<i>this</i>	variable holds value
-------------	----------------------

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

- Models/Cell.h

4.4 CellValue Class Reference

Public Member Functions

- **CellValue** (float value)
- **CellValue** (const std::string value)
- bool [areDecimalOperationsAllowed](#) ()
- float [getDecimalValue](#) ()
- std::string [getTextValue](#) ()

4.4.1 Member Function Documentation

4.4.1.1 [areDecimalOperationsAllowed\(\)](#)

```
bool CellValue::areDecimalOperationsAllowed ( )
```

func check if [Cell](#) value is string or decimal

Returns

- bool statement

4.4.1.2 [getDecimalValue\(\)](#)

```
float CellValue::getDecimalValue ( )
```

func that return float type value

Returns

- float value

4.4.1.3 getTextValue()

```
std::string CellValue::getTextValue ( )
```

func that return string type value

Returns

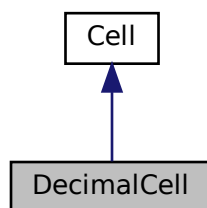
- string value

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

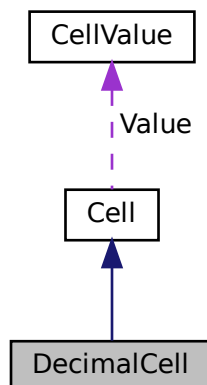
- Models/CellValue.h
- Models/CellValue.cpp

4.5 DecimalCell Class Reference

Inheritance diagram for DecimalCell:



Collaboration diagram for DecimalCell:



Public Member Functions

- void [changeValue](#) (std::string value) override
- void [changeValue](#) (float val) override
- [CellValue](#) [getValue](#) () override
- [DecimalCell](#) (float val)

Additional Inherited Members

4.5.1 Member Function Documentation

4.5.1.1 [changeValue\(\)](#) [1/2]

```
void DecimalCell::changeValue (  
    float val ) [override], [virtual]
```

func that change value

Parameters

<i>value</i>	- param that hold value
--------------	-------------------------

Returns

- void

Implements [Cell](#).

4.5.1.2 [changeValue\(\)](#) [2/2]

```
void DecimalCell::changeValue (  
    std::string val ) [override], [virtual]
```

func that change value

Parameters

<i>value</i>	- param that hold value
--------------	-------------------------

Returns

- void

Implements [Cell](#).

4.5.1.3 getValue()

`CellValue` `DecimalCell::getValue () [override], [virtual]`

func that return value as `CellValue`

Returns

- `CellValue`

Implements `Cell`.

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

- Models/DecimalCell.h
- Models/DecimalCell.cpp

4.6 Identifier Class Reference

Public Member Functions

- **Identifier** (int column, int row)

Public Attributes

- int **Column**
- int **Row**

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

- Models/Identifier.h

4.7 Menu Class Reference

This is a class handling menu operations.

```
#include <menu.h>
```

Static Public Member Functions

- static Operations `showFunctions` ()
func that get from user selected operation.
- static `Identifier getIdentifier` (const std::string &message)
func that get cell identifier from user.
- static float `getNumber` (const std::string &message)
func that get number from user.
- static `CellValue getCell` (const std::string &message)
func that get string from user.
- static void `message` (const std::string &message)
func that send message to user.
- static void `alert` (const std::string &message)
func that send alert message to user.
- static int `getch` ()
func that getch char without waiting for enter source: <https://cboard.cprogramming.com/faq-board/27714-faq-there-getch-conio-equivalent-linux-unix.html>
- static void `clear` ()
func that clear console

4.7.1 Detailed Description

This is a class handling menu operations.

4.7.2 Member Function Documentation

4.7.2.1 alert()

```
void Menu::alert (
    const std::string & message ) [static]
```

func that send alert message to user.

Parameters

in	<i>message</i>	- message to user.
----	----------------	--------------------

Returns

return void

4.7.2.2 clear()

```
void Menu::clear ( ) [static]
```

func that clear console

Returns

void

4.7.2.3 getCell()

```
CellValue Menu::getCell (
    const std::string & message ) [static]
```

func that get string from user.

Parameters

in	<i>message</i>	- message to user.
----	----------------	--------------------

Returns

return float value

4.7.2.4 getch()

```
int Menu::getch ( ) [static]
```

func that getch char without waiting for enter source: <https://cboard.cprogramming.com/faq-board/27714-faq-there-getch-conio-equivalent-linux-unix.html>

Returns

return int value of char

4.7.2.5 getIdentifier()

```
Identifier Menu::getIdentifier (
    const std::string & message ) [static]
```

func that get cell identifier from user.

Parameters

in	<i>message</i>	- message to user.
----	----------------	--------------------

Returns

return [Identifier](#) of cell

4.7.2.6 getNumber()

```
float Menu::getNumber (
    const std::string & message ) [static]
```

func that get number from user.

Parameters

in	<i>message</i>	- message to user.
----	----------------	--------------------

Returns

return float value

4.7.2.7 message()

```
void Menu::message (
    const std::string & message ) [static]
```

func that send message to user.

Parameters

in	<i>message</i>	- message to user.
----	----------------	--------------------

Returns

return void

4.7.2.8 showFunctions()

```
Operations Menu::showFunctions ( ) [static]
```

func that get from user selected operation.

Returns

return operation enum

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

- Header Files/menu.h
- Source Files/menu.cpp

4.8 OperationHandler Class Reference

This is a class handling menu operations.

```
#include <operation_handler.h>
```

Static Public Member Functions

- static int [decimalOperation](#) (Operations operation, [Array](#) *arr)
static func that handling decimal operations.

4.8.1 Detailed Description

This is a class handling menu operations.

4.8.2 Member Function Documentation

4.8.2.1 decimalOperation()

```
int OperationHandler::decimalOperation (  
    Operations operation,  
    Array * arr ) [static]
```

static func that handling decimal operations.

Parameters

in, out	<i>arr</i>	- 2D array that holds sheet.
in	<i>operation</i>	- enum instance that holds user operation.

Returns

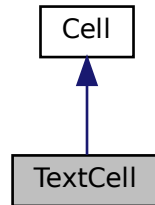
- -1 if exit, 0 if everything is okay.

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

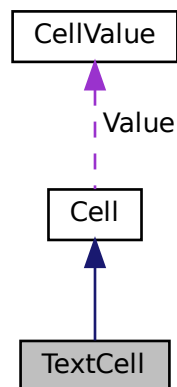
- Header Files/operation_handler.h
- Source Files/operation_handler.cpp

4.9 TextCell Class Reference

Inheritance diagram for TextCell:



Collaboration diagram for TextCell:



Public Member Functions

- void [changeValue](#) (float val) override
- void [changeValue](#) (std::string val) override
- [CellValue](#) [getValue](#) () override
- **TextCell** (const std::string &val)

Additional Inherited Members

4.9.1 Member Function Documentation

4.9.1.1 `changeValue()` [1/2]

```
void TextCell::changeValue (
    float val ) [override], [virtual]
```

func that change value

Parameters

<i>value</i>	- param that hold value
--------------	-------------------------

Returns

- void

Implements [Cell](#).

4.9.1.2 `changeValue()` [2/2]

```
void TextCell::changeValue (
    std::string val ) [override], [virtual]
```

func that change value

Parameters

<i>value</i>	- param that hold value
--------------	-------------------------

Returns

- void

Implements [Cell](#).

4.9.1.3 `getValue()`

```
CellValue TextCell::getValue ( ) [override], [virtual]
```

func that return value as [CellValue](#)

Returns

- [CellValue](#)

Implements [Cell](#).

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

- Models/TextCell.h
- Models/TextCell.cpp

Index

- alert
 - Menu, [22](#)
- areDecimalOperationsAllowed
 - CellValue, [18](#)
- Array, [7](#)
 - Array, [8](#)
 - average, [8](#)
 - changeValue, [8](#)
 - columns, [10](#)
 - divide, [10](#)
 - getMaxLengthValue, [10](#)
 - getNumberFromSheet, [11](#)
 - getStringFromSheet, [11](#)
 - loadDataFromFile, [11](#)
 - multiplication, [12](#)
 - resizeSheet, [12](#)
 - rows, [12](#)
 - saveDataToFile, [12](#)
 - subtract, [13](#)
 - sum, [13](#)
 - to_string_with_precision, [13](#)
- ArrayDisplay, [14](#)
 - Display, [14](#)
 - to_string_with_precision, [15](#)
- average
 - Array, [8](#)
- Cell, [15](#)
 - changeValue, [16](#), [17](#)
 - getValue, [17](#)
 - Value, [17](#)
- CellValue, [18](#)
 - areDecimalOperationsAllowed, [18](#)
 - getDecimalValue, [18](#)
 - getTextValue, [18](#)
- changeValue
 - Array, [8](#)
 - Cell, [16](#), [17](#)
 - DecimalCell, [20](#)
 - TextCell, [26](#), [27](#)
- clear
 - Menu, [22](#)
- columns
 - Array, [10](#)
- DecimalCell, [19](#)
 - changeValue, [20](#)
 - getValue, [21](#)
- decimalOperation
 - OperationHandler, [25](#)
- Display
 - ArrayDisplay, [14](#)
- divide
 - Array, [10](#)
- getCell
 - Menu, [23](#)
- getch
 - Menu, [23](#)
- getDecimalValue
 - CellValue, [18](#)
- getIdentifier
 - Menu, [23](#)
- getMaxLengthValue
 - Array, [10](#)
- getNumber
 - Menu, [24](#)
- getNumberFromSheet
 - Array, [11](#)
- getStringFromSheet
 - Array, [11](#)
- getTextValue
 - CellValue, [18](#)
- getValue
 - Cell, [17](#)
 - DecimalCell, [21](#)
 - TextCell, [27](#)
- Identifier, [21](#)
- loadDataFromFile
 - Array, [11](#)
- Menu, [21](#)
 - alert, [22](#)
 - clear, [22](#)
 - getCell, [23](#)
 - getch, [23](#)
 - getIdentifier, [23](#)
 - getNumber, [24](#)
 - message, [24](#)
 - showFunctions, [24](#)
- message
 - Menu, [24](#)
- multiplication
 - Array, [12](#)
- OperationHandler, [25](#)
 - decimalOperation, [25](#)
- resizeSheet

- Array, [12](#)
- rows
 - Array, [12](#)
- saveDataToFile
 - Array, [12](#)
- showFunctions
 - Menu, [24](#)
- subtract
 - Array, [13](#)
- sum
 - Array, [13](#)
- TextCell, [26](#)
 - changeValue, [26](#), [27](#)
 - getValue, [27](#)
- to_string_with_precision
 - Array, [13](#)
 - ArrayDisplay, [15](#)
- Value
 - Cell, [17](#)