

OwnExcel

Generated by Doxygen 1.8.17



<b>1 Programowanie Obiektowe</b>	<b>1</b>
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
<b>2 Class Index</b>	<b>3</b>
2.1 Class List	3
<b>3 Class Documentation</b>	<b>5</b>
3.1 Array Class Reference	5
3.1.1 Detailed Description	5
3.1.2 Constructor & Destructor Documentation	6
3.1.2.1 Array()	6
3.1.3 Member Function Documentation	6
3.1.3.1 average()	6
3.1.3.2 changeValue()	6
3.1.3.3 divide()	7
3.1.3.4 getNumberFromSheet()	7
3.1.3.5 loadDataFromFile()	8
3.1.3.6 multiplication()	8
3.1.3.7 resizeSheet()	8
3.1.3.8 saveDataToFile()	9
3.1.3.9 subtract()	9
3.1.3.10 sum()	9
3.2 Identifier Class Reference	10
3.3 Menu Class Reference	10
<b>Index</b>	<b>11</b>



## 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

# Class Index

### 2.1 Class List

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

<a href="#">Array</a>	This is a class handling all operations on sheet . . . . .	5
<a href="#">Identifier</a>	. . . . .	10
<a href="#">Menu</a>	. . . . .	10





## Chapter 3

# Class Documentation

### 3.1 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.*
- 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, float 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.*
- void [saveDataToFile](#) ()  
*func that save data from array to file (NOT IMPLEMENTED)*
- void [loadDataFromFile](#) ()  
*func that load data from file (NOT IMPLEMENTED)*

#### 3.1.1 Detailed Description

This is a class handling all operations on sheet.

### 3.1.2 Constructor & Destructor Documentation

#### 3.1.2.1 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.

### 3.1.3 Member Function Documentation

#### 3.1.3.1 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

#### 3.1.3.2 changeValue()

```
void Array::changeValue (
    Identifier identifier,
    float 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

**3.1.3.3 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.

**3.1.3.4 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 nothing or exception

### 3.1.3.5 loadDataFromFile()

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

func that load data from file (NOT IMPLEMENTED)

#### Returns

- nothing

### 3.1.3.6 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

### 3.1.3.7 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

### 3.1.3.8 saveDataToFile()

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

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

#### Returns

- nothing

### 3.1.3.9 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

### 3.1.3.10 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

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

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

## 3.2 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

## 3.3 Menu Class Reference

### Public Member Functions

- void **Show** ()

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

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

# Index

- Array, [5](#)
  - Array, [6](#)
  - average, [6](#)
  - changeValue, [6](#)
  - divide, [7](#)
  - getNumberFromSheet, [7](#)
  - loadDataFromFile, [7](#)
  - multiplication, [8](#)
  - resizeSheet, [8](#)
  - saveDataToFile, [8](#)
  - subtract, [9](#)
  - sum, [9](#)
- average
  - Array, [6](#)
- changeValue
  - Array, [6](#)
- divide
  - Array, [7](#)
- getNumberFromSheet
  - Array, [7](#)
- Identifier, [10](#)
- loadDataFromFile
  - Array, [7](#)
- Menu, [10](#)
- multiplication
  - Array, [8](#)
- resizeSheet
  - Array, [8](#)
- saveDataToFile
  - Array, [8](#)
- subtract
  - Array, [9](#)
- sum
  - Array, [9](#)