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 Class Index **3**

2.1 Class List . . . 3

3 Class Documentation **5**

3.1 Array Class Reference . . . 5

3.1.1 Detailed Description . . . 6

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() . . . 8

3.1.3.4 getNumberFromSheet() . . . 8

3.1.3.5 multiplication() . . . 9

3.1.3.6 resizeSheet() . . . 9

3.1.3.7 subtract() . . . 9

3.1.3.8 sum() . . . 10

3.1.4 Member Data Documentation . . . 10

3.1.4.1 sheet . . . 10

3.1.4.2 sheetColumns . . . 10

3.1.4.3 sheetRows . . . 10

3.2 Identifier Class Reference . . . 11

3.3 Menu Class Reference . . . 11

Index **13**

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:

Array
This is a class handling all operations on sheet 5
Identifier 11
Menu 11

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.
- bool changeValue (Identifier identifier, float value)

  func that change value single position in sheet and return true or false depends on result.
- bool 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.
- bool saveDataToFile ()

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

  func that load data from file (NOT IMPLEMENTED)

Public Attributes

- int sheetColumns
- int sheetRows
- float ** sheet

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	*identifiers*	- array that holds position of cells.
in	*length*	- length of array.

Returns

- float value or exception

3.1.3.2 changeValue()

```
bool Array::changeValue (
            Identifier identifier,
            float value )
```

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

Parameters

in	*identifier*	- position of cell that gonna be changed.
in	*value*	- float value to write up.

Returns

- bool value 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	*divisor*	- first number identifier
in	*dividend*	- 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

identifier	- position of cell that value func gonna return.

Returns

value of single cell or exception

3.1.3.5 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	*identifiers*	- array that holds position of cells.
in	*length*	- length of array.

Returns

- float value or exception

3.1.3.6 resizeSheet()

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

func that resize sheet with no data lose.

Parameters

in	*columns*	- new numbers of columns in sheet.
in	*rows*	- new numbers of rows in sheet.

Returns

bool value or exception

3.1.3.7 subtract()

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

func that subtract two values from sheet and returns result.

Parameters

in	*minued*	- first number identifier
in	*subtrahend*	- second number identifier

Returns

- float value or exception

3.1.3.8 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	*identifiers*	- array that holds position of cells.
in	*length*	- holds length of identifiers array

3.1.4 Member Data Documentation

3.1.4.1 sheet

```
float** Array::sheet
```

Parameters

this	variable store sheet using float array of arrays

3.1.4.2 sheetColumns

```
int Array::sheetColumns
```

Parameters

this	variable store actual number of columns in sheet

3.1.4.3 sheetRows

```
int Array::sheetRows
```

Parameters

this	variable store actual number of rows in sheet

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 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, 8
    getNumberFromSheet, 8
    multiplication, 8
    resizeSheet, 9
    sheet, 10
    sheetColumns, 10
    sheetRows, 10
    subtract, 9
    sum, 10
average
    Array, 6

changeValue
    Array, 6

divide
    Array, 8

getNumberFromSheet
    Array, 8

Identifier, 11

Menu, 11
multiplication
    Array, 8

resizeSheet
    Array, 9

sheet
    Array, 10
sheetColumns
    Array, 10
sheetRows
    Array, 10
subtract
    Array, 9
sum
    Array, 10