

FACULTY OF INFORMATION TECHNOLOGY
BRNO UNIVERSITY OF TECHNOLOGY

PRACTICAL ASPECTS OF SOFTWARE DESIGN – CALCULATOR
SOFTWARE DOCUMENTATION
2019/2020

Generated by Doxygen 1.8.18

1 Namespace Index	1
1.1 Packages	1
2 Hierarchical Index	3
2.1 Class Hierarchy	3
3 Class Index	5
3.1 Class List	5
4 Namespace Documentation	7
4.1 Calculator Namespace Reference	7
4.2 Calculator.Properties Namespace Reference	7
4.3 ExpressionProcessor Namespace Reference	7
4.4 MathLibrary Namespace Reference	8
4.5 MathLibraryTests Namespace Reference	8
4.6 StandardDeviation Namespace Reference	8
5 Class Documentation	9
5.1 MathLibraryTests.AdvancedFunctionsTests Class Reference	9
5.1.1 Detailed Description	9
5.1.2 Member Function Documentation	9
5.1.2.1 Abs()	10
5.1.2.2 Fact()	10
5.1.2.3 Fact_ShouldThrowArgumentOutOfRangeException()	10
5.1.2.4 Pow()	10
5.1.2.5 Pow_ShouldThrowArgumentOutOfRangeException()	11
5.1.2.6 Rnd()	11
5.1.2.7 Root()	11
5.1.2.8 Root_ShouldThrowArgumentOutOfRangeException()	12
5.2 Calculator.App Class Reference	12
5.2.1 Detailed Description	12
5.2.2 Member Function Documentation	13
5.2.2.1 InitializeComponent()	13
5.2.2.2 Main()	13
5.3 MathLibraryTests.BasicFunctionsTests Class Reference	13
5.3.1 Detailed Description	13
5.3.2 Member Function Documentation	13
5.3.2.1 Add()	13
5.3.2.2 Div()	14
5.3.2.3 Div_ShouldThrowDivideByZeroException()	14
5.3.2.4 Mul()	14
5.3.2.5 Sub()	15
5.4 ExpressionProcessor.ExpressionProcessor Class Reference	15
5.5 Calculator.MainWindow Class Reference	15

5.5.1 Detailed Description	16
5.5.2 Member Function Documentation	16
5.5.2.1 Abs()	16
5.5.2.2 Get_Equation()	16
5.5.2.3 InitializeComponent()	17
5.6 StandardDeviation.Program Class Reference	17
5.7 ExpressionProcessor.ShuntingYard Class Reference	17
5.7.1 Detailed Description	17
5.7.2 Member Function Documentation	17
5.7.2.1 ToPostfix()	17
5.8 ExpressionProcessor.Solver Class Reference	18
5.8.1 Member Function Documentation	18
5.8.1.1 Solve()	18
5.9 ExpressionProcessor.Splitter Class Reference	19
5.9.1 Member Function Documentation	19
5.9.1.1 SplitToTokens()	19
5.10 ExpressionProcessor.Validator Class Reference	19
5.10.1 Member Function Documentation	19
5.10.1.1 IsValid()	19
5.11 Calculator.Windowhelp Class Reference	20
5.11.1 Detailed Description	20
5.11.2 Member Function Documentation	20
5.11.2.1 InitializeComponent()	20
Index	21

Chapter 1

Namespace Index

1.1 Packages

Here are the packages with brief descriptions (if available):

Calculator	7
Calculator.Properties	7
ExpressionProcessor	7
MathLibrary	8
MathLibraryTests	8
StandardDeviation	8

Chapter 2

Hierarchical Index

2.1 Class Hierarchy

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

MathLibraryTests.AdvancedFunctionsTests	9
Application	
Calculator.App	12
MathLibraryTests.BasicFunctionsTests	13
ExpressionProcessor.ExpressionProcessor	15
IComponentConnector	
Calculator.Windowhelp	20
StandardDeviation.Program	17
ExpressionProcessor.ShuntingYard	17
ExpressionProcessor.Solver	18
ExpressionProcessor.Splitter	19
ExpressionProcessor.Validator	19
Window	
Calculator.Windowhelp	20
Window	
Calculator.MainWindow	15

Chapter 3

Class Index

3.1 Class List

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

MathLibraryTests.AdvancedFunctionsTests	
Tests of basic functions of mathematical library	9
Calculator.App	
Interaction logic for App.xaml	12
MathLibraryTests.BasicFunctionsTests	
Tests of basic functions of mathematical library	13
ExpressionProcessor.ExpressionProcessor	15
Calculator.MainWindow	
Interaction logic for MainWindow.xaml	15
StandardDeviation.Program	17
ExpressionProcessor.ShuntingYard	
Class implementing a shunting-yard algorithm	17
ExpressionProcessor.Solver	18
ExpressionProcessor.Splitter	19
ExpressionProcessor.Validator	19
Calculator.Windowhelp	
Windowhelp	20

Chapter 4

Namespace Documentation

4.1 Calculator Namespace Reference

Classes

- class [App](#)
Interaction logic for App.xaml
- class [MainWindow](#)
Interaction logic for MainWindow.xaml
- class [Windowhelp](#)
Windowhelp

4.2 Calculator.Properties Namespace Reference

Classes

- class **Resources**
Třída prostředků se silnými typy pro vyhledávání lokalizovaných řetězců atp.
- class **Settings**

4.3 ExpressionProcessor Namespace Reference

Classes

- class [ExpressionProcessor](#)
- class [ShuntingYard](#)
Class implementing a shunting-yard algorithm
- class [Solver](#)
- class [Splitter](#)
- class [Validator](#)

4.4 MathLibrary Namespace Reference

Classes

- class **MathLib**

4.5 MathLibraryTests Namespace Reference

Classes

- class [AdvancedFunctionsTests](#)
Tests of basic functions of mathematical library
- class [BasicFunctionsTests](#)
Tests of basic functions of mathematical library

4.6 StandardDeviation Namespace Reference

Classes

- class [Program](#)

Chapter 5

Class Documentation

5.1 MathLibraryTests.AdvancedFunctionsTests Class Reference

Tests of basic functions of mathematical library

Public Member Functions

- void [Fact](#) (double num, double expected)
Tests of factorial
- void [Fact_ShouldThrowArgumentOutOfRangeException](#) (double num)
Tests of factorial, that should throw ArgumentOutOfRangeException when num is non Natural number except zero
- void [Pow](#) (double number, double exponent, double expected)
Tests of exponentiation
- void [Pow_ShouldThrowArgumentOutOfRangeException](#) (double number, double exponent)
Tests of exponentiation, that should throw ArgumentOutOfRangeException when exponent is non Natural number except zero
- void [Root](#) (double number, double degree, double expected)
Tests of nth-Root
- void [Root_ShouldThrowArgumentOutOfRangeException](#) (double number, double degree)
Tests of nth-Root, that should throw ArgumentOutOfRangeException, when degree is non Natural number or number is a negative number
- void [Abs](#) (double number, double expected)
Tests of absolute value
- void [Rnd](#) ()
Tests of random number generation

5.1.1 Detailed Description

Tests of basic functions of mathematical library

5.1.2 Member Function Documentation

5.1.2.1 Abs()

```
void MathLibraryTests.AdvancedFunctionsTests.Abs (
    double number,
    double expected )
```

Tests of absolute value

Parameters

<i>number</i>	Number
<i>expected</i>	Expected result of test

5.1.2.2 Fact()

```
void MathLibraryTests.AdvancedFunctionsTests.Fact (
    double num,
    double expected )
```

Tests of factorial

Parameters

<i>num</i>	Number
<i>expected</i>	Expected result of test

5.1.2.3 Fact_ShouldThrowArgumentOutOfRangeException()

```
void MathLibraryTests.AdvancedFunctionsTests.Fact_ShouldThrowArgumentOutOfRangeException (
    double num )
```

Tests of factorial, that should throw `ArgumentOutOfRangeException` when *num* is non Natural number except zero

Parameters

<i>num</i>	Number
------------	--------

5.1.2.4 Pow()

```
void MathLibraryTests.AdvancedFunctionsTests.Pow (
    double number,
```

```
double exponent,  
double expected )
```

Tests of exponentiation

Parameters

<i>number</i>	Base
<i>exponent</i>	Exponent
<i>expected</i>	Expected result of test

5.1.2.5 Pow_ShouldThrowArgumentOutOfRangeException()

```
void MathLibraryTests.AdvancedFunctionsTests.Pow_ShouldThrowArgumentOutOfRangeException (   
    double number,  
    double exponent )
```

Tests of exponentiation, that should throw `ArgumentOutOfRangeException` when *exponent* is non Natural number except zero

Parameters

<i>number</i>	Base
<i>exponent</i>	Exponent

5.1.2.6 Rnd()

```
void MathLibraryTests.AdvancedFunctionsTests.Rnd ( )
```

Tests of random number generation

5.1.2.7 Root()

```
void MathLibraryTests.AdvancedFunctionsTests.Root (   
    double number,  
    double degree,  
    double expected )
```

Tests of nth-Root

Parameters

<i>number</i>	Radicant
<i>degree</i>	Degree
<i>expected</i>	Expected result of test

5.1.2.8 Root_ShouldThrowArgumentOutOfRangeException()

```
void MathLibraryTests.AdvancedFunctionsTests.Root_ShouldThrowArgumentOutOfRangeException (
    double number,
    double degree )
```

Tests of nth-Root, that should throw `ArgumentOutOfRangeException`, when *degree* is non Natural number or *number* is a negative number

Parameters

<i>number</i>	Radicant
<i>degree</i>	Degree

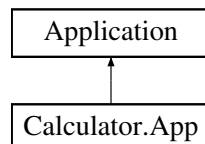
The documentation for this class was generated from the following file:

- `MathLibraryTests/AdvancedFunctionsTests.cs`

5.2 Calculator.App Class Reference

Interaction logic for App.xaml

Inheritance diagram for Calculator.App:



Public Member Functions

- void [InitializeComponent](#) ()
InitializeComponent

Static Public Member Functions

- static void [Main](#) ()
Application Entry Point.

5.2.1 Detailed Description

Interaction logic for App.xaml

[App](#)

5.2.2 Member Function Documentation

5.2.2.1 InitializeComponent()

```
void Calculator.App.InitializeComponent ( )  
InitializeComponent
```

5.2.2.2 Main()

```
static void Calculator.App.Main ( ) [static]  
Application Entry Point.
```

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

- Calculator/App.xaml.cs
- Calculator/obj/Debug/netcoreapp3.1/App.g.i.cs

5.3 MathLibraryTests.BasicFunctionsTests Class Reference

Tests of basic functions of mathematical library

Public Member Functions

- void [Add](#) (double num1, double num2, double expected)
Tests of addition
- void [Sub](#) (double num1, double num2, double expected)
Tests of subtraction
- void [Mul](#) (double num1, double num2, double expected)
Tests of multiplication
- void [Div](#) (double num1, double num2, double expected)
Tests of division
- void [Div_ShouldThrowDivideByZeroException](#) (double num1, double num2)
Test of division, that should throw DivideByZeroException

5.3.1 Detailed Description

Tests of basic functions of mathematical library

5.3.2 Member Function Documentation

5.3.2.1 Add()

```
void MathLibraryTests.BasicFunctionsTests.Add (  
    double num1,  
    double num2,  
    double expected )
```

Tests of addition

Parameters

<i>num1</i>	Addend
<i>num2</i>	Addend
<i>expected</i>	Expected result of test

5.3.2.2 Div()

```
void MathLibraryTests.BasicFunctionsTests.Div (
    double num1,
    double num2,
    double expected )
```

Tests of division

Parameters

<i>num1</i>	Dividend
<i>num2</i>	Divisor
<i>expected</i>	Expected result of test

5.3.2.3 Div_ShouldThrowDivideByZeroException()

```
void MathLibraryTests.BasicFunctionsTests.Div_ShouldThrowDivideByZeroException (
    double num1,
    double num2 )
```

Test of division, that should throw DivideByZeroException

Parameters

<i>num1</i>	Dividend
<i>num2</i>	Divisor

5.3.2.4 Mul()

```
void MathLibraryTests.BasicFunctionsTests.Mul (
    double num1,
    double num2,
    double expected )
```

Tests of multiplication

Parameters

<i>num1</i>	Factor
<i>num2</i>	Factor
<i>expected</i>	Expected result of test

5.3.2.5 Sub()

```
void MathLibraryTests.BasicFunctionsTests.Sub (
    double num1,
    double num2,
    double expected )
```

Tests of subtraction

Parameters

<i>num1</i>	Minuend
<i>num2</i>	Subtrahend
<i>expected</i>	Expected result of test

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

- MathLibraryTests/BasicFunctionsTests.cs

5.4 ExpressionProcessor.ExpressionProcessor Class Reference

Static Public Member Functions

- static string **Process** (string expression)

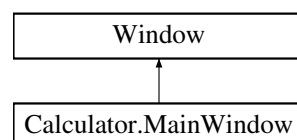
The documentation for this class was generated from the following file:

- ExpressionProcessor/ExpressionProcessor.cs

5.5 Calculator.MainWindow Class Reference

Interaction logic for MainWindow.xaml

Inheritance diagram for Calculator.MainWindow:



Public Member Functions

- void [Get_Equation](#) (object sender, RoutedEventArgs e)
Actions to take when "=" button is clicked.
- void [Abs](#) (object sender, RoutedEventArgs e)
Actions to take when "Abs" button is clicked.
- void [InitializeComponent](#) ()
InitializeComponent

5.5.1 Detailed Description

Interaction logic for MainWindow.xaml

[MainWindow](#)

5.5.2 Member Function Documentation

5.5.2.1 Abs()

```
void Calculator.MainWindow.Abs (
    object sender,
    RoutedEventArgs e )
```

Actions to take when "Abs" button is clicked.

Parameters

<i>sender</i>	
<i>e</i>	

5.5.2.2 Get_Equation()

```
void Calculator.MainWindow.Get_Equation (
    object sender,
    RoutedEventArgs e )
```

Actions to take when "=" button is clicked.

Parameters

<i>sender</i>	
<i>e</i>	

5.5.2.3 InitializeComponent()

```
void Calculator.MainWindow.InitializeComponent ( )
```

InitializeComponent

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

- Calculator/MainWindow.xaml.cs
- Calculator/obj/Debug/netcoreapp3.1/MainWindow.g.i.cs

5.6 StandardDeviation.Program Class Reference

Static Public Member Functions

- static void **Main** (string[] args)

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

- StandardDeviation/Program.cs

5.7 ExpressionProcessor.ShuntingYard Class Reference

Class implementing a shunting-yard algorithm

Static Public Member Functions

- static List< string > **ToPostfix** (List< string > tokens)
Convert mathematical expression to postfix notation

5.7.1 Detailed Description

Class implementing a shunting-yard algorithm

5.7.2 Member Function Documentation

5.7.2.1 ToPostfix()

```
static List<string> ExpressionProcessor.ShuntingYard.ToPostfix (
    List< string > tokens ) [static]
```

Convert mathematical expression to postfix notation

Parameters

<i>tokens</i>	List of tokens of mathematical expression
---------------	---

Returns

List of tokens in postfix notation

Exceptions

<i>ArgumentException</i>	Thrown when <i>tokens</i> contains unclosed expression
--------------------------	--

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

- ExpressionProcessor/ShuntingYard.cs

5.8 ExpressionProcessor.Solver Class Reference

Static Public Member Functions

- static string [Solve](#) (List< string > tokens)
Calculates the given expression which is splitted into tokens.

5.8.1 Member Function Documentation

5.8.1.1 Solve()

```
static string ExpressionProcessor.Solver.Solve (  
    List< string > tokens ) [static]
```

Calculates the given expression which is splitted into tokens.

Expression must be in postfix notation

Parameters

<i>tokens</i>	List of tokens
---------------	----------------

Returns

Calculation result

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

- ExpressionProcessor/Solver.cs

5.9 ExpressionProcessor.Splitter Class Reference

Static Public Member Functions

- static List< string > [SplitToTokens](#) (string exp)
Splits a given string into tokens

5.9.1 Member Function Documentation

5.9.1.1 SplitToTokens()

```
static List<string> ExpressionProcessor.Splitter.SplitToTokens (
    string exp ) [static]
```

Splits a given string into tokens

Parameters

<i>exp</i>	The string to be splitted
------------	---------------------------

Returns

List of tokens

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

- ExpressionProcessor/Splitter.cs

5.10 ExpressionProcessor.Validator Class Reference

Static Public Member Functions

- static bool [IsValid](#) (string exp, int num_of_param=1)
Checks if a given string is a valid expression.

5.10.1 Member Function Documentation

5.10.1.1 IsValid()

```
static bool ExpressionProcessor.Validator.IsValid (
    string exp,
    int num_of_param = 1 ) [static]
```

Checks if a given string is a valid expression.

Parameters

<i>exp</i>	The string to be checked
<i>num_of_param</i>	Number of parameters (Only if the expression is inside a function)

Parameter `num_of_param` is optional

Returns

`True` if a string is a valid expression

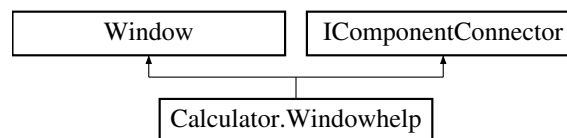
The documentation for this class was generated from the following file:

- `ExpressionProcessor/Validator.cs`

5.11 Calculator.Windowhelp Class Reference

Windowhelp

Inheritance diagram for Calculator.Windowhelp:

**Public Member Functions**

- void `InitializeComponent` ()
InitializeComponent

5.11.1 Detailed Description

Windowhelp

Interaction logic for Windowhelp.xaml

5.11.2 Member Function Documentation

5.11.2.1 InitializeComponent()

```
void Calculator.Windowhelp.InitializeComponent ( )
```

`InitializeComponent`

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

- `Calculator/obj/Debug/netcoreapp3.1/Windowhelp.g.i.cs`
- `Calculator/Windowhelp.xaml.cs`

Index

- Abs
 - Calculator.MainWindow, [16](#)
 - MathLibraryTests.AdvancedFunctionsTests, [9](#)
- Add
 - MathLibraryTests.BasicFunctionsTests, [13](#)
- Calculator, [7](#)
- Calculator.App, [12](#)
 - InitializeComponent, [13](#)
 - Main, [13](#)
- Calculator.MainWindow, [15](#)
 - Abs, [16](#)
 - Get_Equation, [16](#)
 - InitializeComponent, [17](#)
- Calculator.Properties, [7](#)
- Calculator.Windowhelp, [20](#)
 - InitializeComponent, [20](#)
- Div
 - MathLibraryTests.BasicFunctionsTests, [14](#)
- Div_ShouldThrowDivideByZeroException
 - MathLibraryTests.BasicFunctionsTests, [14](#)
- ExpressionProcessor, [7](#)
- ExpressionProcessor.ExpressionProcessor, [15](#)
- ExpressionProcessor.ShuntingYard, [17](#)
 - ToPostfix, [17](#)
- ExpressionProcessor.Solver, [18](#)
 - Solve, [18](#)
- ExpressionProcessor.Splitter, [19](#)
 - SplitToTokens, [19](#)
- ExpressionProcessor.Validator, [19](#)
 - IsValid, [19](#)
- Fact
 - MathLibraryTests.AdvancedFunctionsTests, [10](#)
- Fact_ShouldThrowArgumentOutOfRangeException
 - MathLibraryTests.AdvancedFunctionsTests, [10](#)
- Get_Equation
 - Calculator.MainWindow, [16](#)
- InitializeComponent
 - Calculator.App, [13](#)
 - Calculator.MainWindow, [17](#)
 - Calculator.Windowhelp, [20](#)
- IsValid
 - ExpressionProcessor.Validator, [19](#)
- Main
 - Calculator.App, [13](#)
- MathLibrary, [8](#)
- MathLibraryTests, [8](#)
- MathLibraryTests.AdvancedFunctionsTests, [9](#)
 - Abs, [9](#)
 - Fact, [10](#)
 - Fact_ShouldThrowArgumentOutOfRangeException, [10](#)
 - Pow, [10](#)
 - Pow_ShouldThrowArgumentOutOfRangeException, [11](#)
 - Rnd, [11](#)
 - Root, [11](#)
 - Root_ShouldThrowArgumentOutOfRangeException, [12](#)
- MathLibraryTests.BasicFunctionsTests, [13](#)
 - Add, [13](#)
 - Div, [14](#)
 - Div_ShouldThrowDivideByZeroException, [14](#)
 - Mul, [14](#)
 - Sub, [15](#)
- Mul
 - MathLibraryTests.BasicFunctionsTests, [14](#)
- Pow
 - MathLibraryTests.AdvancedFunctionsTests, [10](#)
- Pow_ShouldThrowArgumentOutOfRangeException
 - MathLibraryTests.AdvancedFunctionsTests, [11](#)
- Rnd
 - MathLibraryTests.AdvancedFunctionsTests, [11](#)
- Root
 - MathLibraryTests.AdvancedFunctionsTests, [11](#)
- Root_ShouldThrowArgumentOutOfRangeException
 - MathLibraryTests.AdvancedFunctionsTests, [12](#)
- Solve
 - ExpressionProcessor.Solver, [18](#)
- SplitToTokens
 - ExpressionProcessor.Splitter, [19](#)
- StandardDeviation, [8](#)
- StandardDeviation.Program, [17](#)
- Sub
 - MathLibraryTests.BasicFunctionsTests, [15](#)
- ToPostfix
 - ExpressionProcessor.ShuntingYard, [17](#)