

WinDos Kalkulačka

1.0

Generováno programem Doxygen 1.8.13

Obsah

1	WinDos-Kalkulačka	1
2	Rejstřík prostorů jmen	3
2.1	Seznam prostorů jmen	3
3	Rejstřík tříd	5
3.1	Seznam tříd	5
4	Rejstřík souborů	7
4.1	Seznam souborů	7
5	Dokumentace prostorů jmen	9
5.1	Dokumentace prostoru jmen CalculatorUnit	9
5.1.1	Detailní popis	9
5.1.2	Dokumentace výčtových typů	10
5.1.2.1	CalcErrorType	10
5.1.2.2	NumSystem	10
5.2	Dokumentace prostoru jmen MathLib	10
5.2.1	Detailní popis	11

6	Dokumentace tříd	13
6.1	Dokumentace třídy MathLib.CalcMath	13
6.1.1	Detailní popis	13
6.1.2	Dokumentace k metodám	14
6.1.2.1	Add()	14
6.1.2.2	Divide()	14
6.1.2.3	Fact()	15
6.1.2.4	Log()	15
6.1.2.5	Modulo()	15
6.1.2.6	Multiply()	16
6.1.2.7	Pow()	16
6.1.2.8	Root()	17
6.1.2.9	Subtract()	17
6.2	Dokumentace třídy CalculatorUnit.Calculation	18
6.2.1	Detailní popis	18
6.2.2	Dokumentace konstruktoru a destruktoru	19
6.2.2.1	Calculation()	19
6.2.3	Dokumentace k metodám	19
6.2.3.1	GetAsString()	19
6.2.4	Dokumentace k vlastnosti	19
6.2.4.1	ErrorType	20
6.2.4.2	Expression	20
6.2.4.3	Value	20
6.3	Dokumentace třídy CalculatorUnit.NumberConverter	21
6.3.1	Detailní popis	21
6.3.2	Dokumentace k metodám	21
6.3.2.1	ToDouble()	21
6.3.2.2	ToString()	22
7	Dokumentace souborů	23
7.1	Dokumentace souboru D:/IVS-Kalkulacka/IVS-Kalkulacka/README.md	23
7.2	Dokumentace souboru CalculatorProject/CalculatorUnit/Calculation.cs	23
7.2.1	Detailní popis	23
7.3	Dokumentace souboru CalculatorProject/CalculatorUnit/CalcUnitEnums.cs	24
7.3.1	Detailní popis	24
7.4	Dokumentace souboru CalculatorProject/CalculatorUnit/NumberConverter.cs	24
7.4.1	Detailní popis	25
7.5	Dokumentace souboru CalculatorProject/MathLib/CalcMath.cs	25
7.5.1	Detailní popis	25

Kapitola 1

WinDos-Kalkulačka

Týmový projekt do předmětu IVS na FIT VUT

Prostředí

Windows 64bit

Autoři

WinDos

- xblaha28 Radim Blaha
- xfrane16 Jaromír Franěk
- xfusek08 Petr Fusek
- xvosyk00 Pavel Vosyka

Licence

Tento program je poskytován pod licencí GPLv3.0.

Testování

Testování je prováděno pomocí frameworku MSTest ve visual studiu. Jednotkové testy jsou součástí projektu v "src\CalculatorProject\ProjectTesting\".

Kapitola 2

Rejstřík prostorů jmen

2.1 Seznam prostorů jmen

Zde naleznete seznam všech prostorů jmen se stručným popisem:

CalculatorUnit	
Modul výpočetní jednotky	9
MathLib	
Matematická knihovna	10

Kapitola 3

Rejstřík tříd

3.1 Seznam tříd

Následující seznam obsahuje především identifikace tříd, ale nacházejí se zde i další netriviální prvky, jako jsou struktury (struct), unie (union) a rozhraní (interface). V seznamu jsou uvedeny jejich stručné popisy:

MathLib.CalcMath	
Matematická knihovna	13
CalculatorUnit.Calculation	
Objekt představující jeden výpočet	18
CalculatorUnit.NumberConverter	
Třída poskytující metody pro konverzi čísel na řetězce a naopak	21

Kapitola 4

Rejstřík souborů

4.1 Seznam souborů

Zde naleznete seznam všech souborů se stručnými popisy:

CalculatorProject/CalculatorUnit/ Calculation.cs	
Třída výpočtu výpočetní jednotky	23
CalculatorProject/CalculatorUnit/ CalcUnitEnums.cs	
Soubor združující výčtové typy používané v Namespace CalculatorUnit	24
CalculatorProject/CalculatorUnit/ NumberConverter.cs	
Pomocná statická třída pro konvertování čísla na řetězec	24
CalculatorProject/MathLib/ CalcMath.cs	
Matematická knihovna	25

Kapitola 5

Dokumentace prostorů jmen

5.1 Dokumentace prostoru jmen CalculatorUnit

Modul výpočetní jednotky.

Třídy

- class [Calculation](#)
Objekt představující jeden výpočet.
- class [NumberConverter](#)
Třída poskytující metody pro konverzi čísel na řetězce a naopak.

Výčty

- enum [CalcErrorType](#) {
 [CalcErrorType.None](#), [CalcErrorType.FuncDomainError](#), [CalcErrorType.DataTypeOverflow](#), [CalcErrorType.ExprFormatError](#),
 [CalcErrorType.UnknownError](#) }
Druhy chyb, které mohou nastat během výpočtu.
- enum [NumSystem](#) { [NumSystem.Dec](#) = 10, [NumSystem.Bin](#) = 2, [NumSystem.Hex](#) = 16, [NumSystem.Oct](#) = 8 }
Typ číselné soustavy.

5.1.1 Detailní popis

Modul výpočetní jednotky.

Modul představuje mezivrstvu mezi uživatelským rozhraním a matematickou knihovnou. Obsahuje prostředky pro vyhodnocování matematických výrazů, pomocí funkcí z matematické knihovny [MathLib.CalcMath](#).

Autor

Petr Fusek
Pavel Vosyka

5.1.2 Dokumentace výčtových typů

5.1.2.1 CalcErrorType

```
enum CalculatorUnit.CalcErrorType [strong]
```

Druhy chyb, které mohou nastat během výpočtu.

Hodnoty výčtu

None	Žádná chybaVýpočet proběhl v pořádku.
FuncDomainError	Chyba definičního oboru funkceVe výrazu jsou hodnoty operátorů, které nemají definovanou hodnotu v konkrétní funkci.
DataTypeOverflow	Během výpočtu došlo k přetečení. V případě, že se během výpočtů dostaneme za hranice rozsahu typu double .
ExprFormatError	Chyba formátování výrazuChyba pokud vyhodnocovaný řetězec není platný matematický výraz. např.: <ul style="list-style-type: none">• Neukončené závorky• Operátory bez operandů• neznámé znaky• ...
UnknownError	Ostatní neznámé chyby

5.1.2.2 NumSystem

```
enum CalculatorUnit.NumSystem [strong]
```

Typ číselné soustavy.

Hodnoty výčtu

Dec	Desítková soustava
Bin	Binární soustava
Hex	Šestnáctková soustava
Oct	Osmičková soustava

5.2 Dokumentace prostoru jmen MathLib

Matematická knihovna.

Třídy

- class [CalcMath](#)

Matematická knihovna

5.2.1 Detailní popis

Matematická knihovna.

Matematická knihovna pro výpočet základních matematických operací

Autor

Jaromír Franěk

Kapitola 6

Dokumentace tříd

6.1 Dokumentace třídy MathLib.CalcMath

Matematická knihovna

Veřejné metody

- double **Add** (double a, double b)
Sčítání
- double **Subtract** (double a, double b)
Odčítání
- double **Divide** (double a, double b)
Dělení
- double **Multiply** (double a, double b)
Násobení
- double **Pow** (double x, double y)
Mocnina
- double **Root** (double a, double x)
Odmocnina
- double **Fact** (Byte x)
Factorial
- double **Modulo** (double a, double b)
Zbytek
- double **Log** (double x)
Logaritmus

6.1.1 Detailní popis

Matematická knihovna

Třída obsahující metody základních matematických operací

6.1.2 Dokumentace k metodám

6.1.2.1 Add()

```
double MathLib.CalcMath.Add (
    double a,
    double b ) [inline]
```

Sčítání

Sečte argument a s argumentem b

Parametry

<i>a</i>	Předá první číselník typu double
<i>b</i>	Předá druhý číselník typu double

Návratová hodnota

Vrací součet argumentů

6.1.2.2 Divide()

```
double MathLib.CalcMath.Divide (
    double a,
    double b ) [inline]
```

Dělení

Vydělí argument a argumentem b

Parametry

<i>a</i>	Předá dělenec typu double
<i>b</i>	Předá dělitel typu double

Návratová hodnota

Vrací podíl argumentů

6.1.2.3 Fact()

```
double MathLib.CalcMath.Fact (
    Byte x ) [inline]
```

Factorial

Factorial z čísla x, o maximální hodnotě 170

Parametry

<i>b</i>	Předá hodnotu factorialu typu byte
----------	------------------------------------

Návratová hodnota

Pokud je $x < 170$, vrací factorial z čísla x, pokud je $x > 170$, vrací PositiveInfinity.

6.1.2.4 Log()

```
double MathLib.CalcMath.Log (
    double x ) [inline]
```

Logaritmus

Logaritmus při základu 10 argumentu x

Parametry

<i>x</i>	Předá hodnotu typu double, ze které bude logaritmus vypočítán
----------	---

Návratová hodnota

Vrací výpočet logaritmu z čísla x o základu 10

6.1.2.5 Modulo()

```
double MathLib.CalcMath.Modulo (
    double a,
    double b ) [inline]
```

Zbytek

Zbytek po dělení argumentu a argumentem b

Parametry

<i>a</i>	Předá dělenec typu double
<i>b</i>	Předá dělitel typu double

Návratová hodnota

Vrací zbytek po podílu čísla *a* číslem *b*

6.1.2.6 Multiply()

```
double MathLib.CalcMath.Multiply (  
    double a,  
    double b ) [inline]
```

Násobení

Vynásobí argument *a* argumentem *b*

Parametry

<i>a</i>	Předá první činitel typu double
<i>b</i>	Předá druhý činitel typu double

Návratová hodnota

Vrací násobek argumentů

6.1.2.7 Pow()

```
double MathLib.CalcMath.Pow (  
    double x,  
    double y ) [inline]
```

Mocnina

Obecná *y*-tá mocnina z "*x*"

Parametry

<i>x</i>	Předá základ typu double
<i>y</i>	Předá exponent typu double

Návratová hodnota

Vrací y-tou mocninu argumentu x

6.1.2.8 Root()

```
double MathLib.CalcMath.Root (
    double a,
    double x ) [inline]
```

Odmocnina

Obecná x-tá odmocnina z "a"

Parametry

<i>a</i>	Předá základ typu double
<i>x</i>	Předá exponent typu double

Návratová hodnota

Vrací x-tou odmocninu argumentu a

6.1.2.9 Subtract()

```
double MathLib.CalcMath.Subtract (
    double a,
    double b ) [inline]
```

Odčítání

Odečte argument a od argumentu b

Parametry

<i>a</i>	Předá menšenec typu double
<i>b</i>	Předá menšitel typu double

Návratová hodnota

Vrací rozdíl argumentů

Dokumentace pro tuto třídu byla generována z následujícího souboru:

- CalculatorProject/MathLib/[CalcMath.cs](#)

6.2 Dokumentace třídy CalculatorUnit.Calculation

Objekt představující jeden výpočet.

Veřejné metody

- **Calculation** (string expr)
Konstruktor objektu
- string **GetAsString** (NumSystem numbase, string format)
Vrací Value jako řetězec.

Vlastnosti

- double **Value** [get]
Výsledek výpočtu
- string **Expression** [get, set]
Matematický výraz
- **CalcErrorType ErrorType** [get]
Typ chyby, která nastala během výpočtu

6.2.1 Detailní popis

Objekt představující jeden výpočet.

Přijímá matematický výraz, který implicitně vyhodnotí a změní stav svých vlastností podle výsledku operace.

Výrazy

Výrazy jsou klasické řetězce znaků, představující matematický zápis. rozšířený o dodatečné operátory, představující další funkce kalkulačky.

Příklad složitějšího výrazu:

```
(- (7-3) ! / (5+1)) - ((6-3) @ (30 - (L(200-50*2) + 1)) * -1)
```

Všechny mezery ve výrazu jsou ignorovány.

Podporované operátory

Operátor	Výraz	Význam	Odpovídající funkce z CalcMath
!	x !	faktorial x	Fact()
L	Lx	logaritmus z x při základu 10	Log()
^	x^y	x umocněné na y	Pow()
@	x@y	x-tá odmocnina z y	Root()
%	x y	x modulo y	Modulo()
*	x*y	x krát y	Multiply()
/	x/y	x děleno y	Divide()
-	x-y	x mínus y	Subtract()
+	x+y	x plus y	Add()

viz. [MathLib.CalcMath](#)

6.2.2 Dokumentace konstruktoru a destruktoru

6.2.2.1 Calculation()

```
CalculatorUnit.Calculation.Calculation (
    string expr ) [inline]
```

Konstruktor objektu

Inicializace objektu

Parametry

<i>expr</i>	matematický výraz, který je nastaven do vlastnosti Expression , a jeho vyhodnocení je v Value
-------------	---

6.2.3 Dokumentace k metodám

6.2.3.1 GetAsString()

```
string CalculatorUnit.Calculation.GetAsString (
    NumSystem numbase,
    string format ) [inline]
```

Vrací Value jako řetězec.

Metoda vrací výsledek výrazu Expression v podobě formátovaného řetězce v zadané číselné soustavě.

Parametry

<i>numbase</i>	Číselná soustava ve které bude výsledek vypsán. Definuje se: NumSystem
<i>format</i>	Formátovací řetězec <ul style="list-style-type: none">Odpovídá parametru format v NumberConverter.ToString

Návratová hodnota

Výsledný řetězec

6.2.4 Dokumentace k vlastnosti

6.2.4.1 `ErrorType`

```
CalcErrorType CalculatorUnit.Calculation.ErrorType [get]
```

Typ chyby, která nastala během výpočtu

- Read-only vlastnost.
- Hodnota se aktualizuje při každém výpočtu.

6.2.4.2 `Expression`

```
string CalculatorUnit.Calculation.Expression [get], [set]
```

Matematický výraz

Atribut objektu udržující řetězec odpovídající matematickému výrazu, který třída vyhodnocuje.

Chování:

- **get** - vrátí text výrazu
- **set**
 - všechny bílé znaky jsou vymazány
 - provede vyhodnocení a aktualizuje hodnoty vlastností:
 - * [Value](#)
 - * [ErrorType](#)

6.2.4.3 `Value`

```
double CalculatorUnit.Calculation.Value [get]
```

Výsledek výpočtu

Vrací výslednou hodnotu vyhodnoceného matematického výrazu v `Expression`.

- Read-only vlastnost.
- Hodnota se mění pouze v případě změny výrazu a to bezprostředně.
- Hodnota **NaN** znamená chybu během výpočtu a její typ je uložen v [ErrorType](#)

Dokumentace pro tuto třídu byla generována z následujícího souboru:

- CalculatorProject/CalculatorUnit/[Calculation.cs](#)

6.3 Dokumentace třídy CalculatorUnit.NumberConverter

Třída poskytující metody pro konverzi čísel na řetězce a naopak.

Statické veřejné metody

- static double [ToDouble](#) (string input, [NumSystem](#) numsystem)
Konvertuje řetězec na číslo s pohyblivou desetinnou čárkou.
- static string [ToString](#) (double value, [NumSystem](#) numbase, string format)
Konvertuje číslo na řetězec.

6.3.1 Detailní popis

Třída poskytující metody pro konverzi čísel na řetězce a naopak.

Podporuje také převody mezi soustavami o základu 2, 8, 10, 16.

6.3.2 Dokumentace k metodám

6.3.2.1 ToDouble()

```
static double CalculatorUnit.NumberConverter.ToDouble (
    string input,
    NumSystem numsystem ) [inline], [static]
```

Konvertuje řetězec na číslo s pohyblivou desetinnou čárkou.

Parametry

<i>input</i>	Řetězec obsahující číslo ve specifikované číselné soustavě
<i>numsystem</i>	Číselná soustava ve které je řetězec zapsán NumSystem

Návratová hodnota

Vrací převedené číslo. Pokud je řetězec roven "NaN" vrací NaN, pokud je "INF" nebo "-INF" vrací double.↔ PositiveInfinity nebo double.NegativeInfinity.

Výjimky

<i>ArgumentNullException</i>	Input je null . (pouze při převodu z desítkového čísla.) viz. ArgumentNullException Class
<i>OverflowException</i>	Číslo přesahuje maximální nebo minimální rozsah. (pouze při převodu z desítkového čísla.) viz. OverflowException Class
<i>FormatException</i>	Výjimka je vyhozena, pokud vstupní řetězec není platné číslo v dané soustavě. Metoda přijímá desetinnou tečku ".", nikoliv čárku ",".

6.3.2.2 ToString()

```
static string CalculatorUnit.NumberConverter.ToString (
    double value,
    NumSystem numbase,
    string format ) [inline], [static]
```

Konvertuje číslo na řetězec.

Konvertuje číslo na řetězec v zadané číselné soustavě na počet zadaných desetinných míst. Převod o jiných soustav než desítkových je limitován velikostí integer, v takovém případě vrátí "NaN".

Parametry

<i>value</i>	Hodnota, která má být převedena na číslo
<i>numbase</i>	Základ soustavy ve které se vypíše viz. NumSystem
<i>format</i>	Formátovací řetězec <ul style="list-style-type: none">• Výchozí nastavení formátu je na 4 desetinná místa.• pokud: numbase je NumSystem.Dec (desítková soustava)<ul style="list-style-type: none">– Očekává formátovací řetězec dle standardu .net– Pokud je prázdný řetězec použije se výchozí nastavení.• jinak: numbase je jiná soustava<ul style="list-style-type: none">– Očekává číslo, které představuje počet desetinných míst.– Pokud nebude validní, použije se výchozí nastavení.

Návratová hodnota

číslo v podobě řetězce v požadované číselné soustavě.

- pokud je `value > 'int max'` a numbase není [NumSystem.Dec](#) potom vrátí "NaN".

Dokumentace pro tuto třídu byla generována z následujícího souboru:

- CalculatorProject/CalculatorUnit/[NumberConverter.cs](#)

Kapitola 7

Dokumentace souborů

7.1 Dokumentace souboru D:/IVS-Kalkulacka/IVS-Kalkulacka/README.md

7.2 Dokumentace souboru CalculatorProject/CalculatorUnit/Calculation.cs

Třída výpočtu výpočetní jednotky.

Třídy

- class [CalculatorUnit.Calculation](#)
Objekt představující jeden výpočet.

Prostory jmen

- namespace [CalculatorUnit](#)
Modul výpočetní jednotky.

7.2.1 Detailní popis

Třída výpočtu výpočetní jednotky.

Obsahuje popis modulu a třídu výpočtu **Calculation**

Autor

Petr Fusek
Pavel Vosyka

Datum

19.04.2017

7.3 Dokumentace souboru CalculatorProject/CalculatorUnit/CalcUnitEnums.cs

Soubor združující výčtové typy používané v Namespace [CalculatorUnit](#).

Prostory jmen

- namespace [CalculatorUnit](#)
Modul výpočetní jednotky.

Výčty

- enum [CalculatorUnit.CalcErrorType](#) {
 [CalculatorUnit.CalcErrorType.None](#), [CalculatorUnit.CalcErrorType.FuncDomainError](#), [CalculatorUnit.CalcErrorType.DataTypeOverflow](#), [CalculatorUnit.CalcErrorType.ExprFormatError](#),
 [CalculatorUnit.CalcErrorType.UnknownError](#) }
Druhy chyb, které mohou nastat během výpočtu.
- enum [CalculatorUnit.NumSystem](#) { [CalculatorUnit.NumSystem.Dec](#) = 10, [CalculatorUnit.NumSystem.Bin](#) = 2, [CalculatorUnit.NumSystem.Hex](#) = 16, [CalculatorUnit.NumSystem.Oct](#) = 8 }
Typ číselné soustavy.

7.3.1 Detailní popis

Soubor združující výčtové typy používané v Namespace [CalculatorUnit](#).

Autor

Petr Fusek
Pavel Vosyka

Datum

11.04.2017

7.4 Dokumentace souboru CalculatorProject/CalculatorUnit/NumberConverter.cs

Pomocná statická třída pro konvertování čísla na řetězec.

Třídy

- class [CalculatorUnit.NumberConverter](#)
Třída poskytující metody pro konverzi čísel na řetězce a naopak.

Prostory jmen

- namespace [CalculatorUnit](#)
Modul výpočetní jednotky.

7.4.1 Detailní popis

Pomocná statická třída pro konvertování čísla na řetězec.

Autor

Petr Fusek
Pavel Vasyka

Datum

19.04.2017

7.5 Dokumentace souboru CalculatorProject/MathLib/CalcMath.cs

Matematická knihovna.

Třídy

- class [MathLib.CalcMath](#)
Matematická knihovna

Prostory jmen

- namespace [MathLib](#)
Matematická knihovna.

7.5.1 Detailní popis

Matematická knihovna.

Autor

Jaromír Franěk

Datum

13.04.2017

Rejstřík

Add

MathLib::CalcMath, [14](#)

CalcErrorType

CalculatorUnit, [10](#)

Calculation

CalculatorUnit::Calculation, [19](#)

CalculatorProject/CalculatorUnit/CalcUnitEnums.cs, [24](#)

CalculatorProject/CalculatorUnit/Calculation.cs, [23](#)

CalculatorProject/CalculatorUnit/NumberConverter.cs, [24](#)

CalculatorProject/MathLib/CalcMath.cs, [25](#)

CalculatorUnit, [9](#)

CalcErrorType, [10](#)

NumSystem, [10](#)

CalculatorUnit.Calculation, [18](#)

CalculatorUnit.NumberConverter, [21](#)

CalculatorUnit::Calculation

Calculation, [19](#)

ErrorType, [19](#)

Expression, [20](#)

GetAsString, [19](#)

Value, [20](#)

CalculatorUnit::NumberConverter

ToDouble, [21](#)

ToString, [22](#)

D:/IVS-Kalkulacka/IVS-Kalkulacka/README.md, [23](#)

Divide

MathLib::CalcMath, [14](#)

ErrorType

CalculatorUnit::Calculation, [19](#)

Expression

CalculatorUnit::Calculation, [20](#)

Fact

MathLib::CalcMath, [14](#)

GetAsString

CalculatorUnit::Calculation, [19](#)

Log

MathLib::CalcMath, [15](#)

MathLib, [10](#)

MathLib.CalcMath, [13](#)

MathLib::CalcMath

Add, [14](#)

Divide, [14](#)

Fact, [14](#)

Log, [15](#)

Modulo, [15](#)

Multiply, [16](#)

Pow, [16](#)

Root, [17](#)

Subtract, [17](#)

Modulo

MathLib::CalcMath, [15](#)

Multiply

MathLib::CalcMath, [16](#)

NumSystem

CalculatorUnit, [10](#)

Pow

MathLib::CalcMath, [16](#)

Root

MathLib::CalcMath, [17](#)

Subtract

MathLib::CalcMath, [17](#)

ToDouble

CalculatorUnit::NumberConverter, [21](#)

ToString

CalculatorUnit::NumberConverter, [22](#)

Value

CalculatorUnit::Calculation, [20](#)