

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

Fakulta informačních technologií

**Jan Kalina (xkalinj00)**

Praktické aspekty vývoje software

2023/2024

**Protokol z profilování**

Farkašovský Lukáš (xfarkal00)

Hýža Pavel (xhyzapa00)

Kalina Jan (xkalinj00)

Krejčí David (xkrejcd00)

Brno, 22. dubna 2024

# OBSAH

<b>OBSAH .....</b>	<b>2</b>
<b>1 PROFILOVACÍ PROSTŘEDÍ .....</b>	<b>3</b>
1.1 INFORMACE O SYSTÉMU – HONZIKSICK@CAREN .....	3
1.2 PROFILOVACÍ SOFTWARE .....	3
1.3 ZPŮSOB PŘEKladu PROGRAMŮ A KNIHOVEN .....	3
1.4 LEGENDY DATOVÝCH TABULEK .....	4
1.4.1 TABULKA ČASU STRÁVENÉHO V JEDNOTLIVÝCH FUNKCÍCH .....	4
1.4.2 GRAF VOLÁNÍ FUNKCÍ (CALL GRAPH) .....	5
<b>2 VÝSLEDKY PROFILOVÁNÍ .....</b>	<b>7</b>
2.1 PRO POČET $10^1$ VZORKŮ .....	7
2.1.1 TABULKA ČASU STRÁVENÉHO V JEDNOTLIVÝCH FUNKCÍCH .....	7
2.1.2 GRAF VOLÁNÍ FUNKCÍ (CALL GRAPH) .....	8
2.1.3 INDEXY FUNKCÍ .....	11
2.2 PRO POČET $10^3$ VZORKŮ .....	12
2.2.1 TABULKA ČASU STRÁVENÉHO V JEDNOTLIVÝCH FUNKCÍCH .....	12
2.2.2 GRAF VOLÁNÍ FUNKCÍ (CALL GRAPH) .....	13
2.2.3 INDEXY FUNKCÍ .....	16
2.3 PRO POČET $10^6$ VZORKŮ .....	17
2.3.1 TABULKA ČASU STRÁVENÉHO V JEDNOTLIVÝCH FUNKCÍCH .....	17
2.3.2 GRAF VOLÁNÍ FUNKCÍ (CALL GRAPH) .....	18
2.3.3 INDEXY FUNKCÍ .....	21
<b>3 SOUHRNNÉ STATISTIKY A ANALÝZA VÝSLEDKŮ .....</b>	<b>22</b>
3.1 TABULKA ČASU STRÁVENÉHO V JEDNOTLIVÝCH FUNKCÍCH .....	22
3.1.1 SEŘAZENO DLE TIME [%] .....	22
3.1.2 SEŘAZENO DLE COMULATIVE [s] .....	24
3.2 VÝSLEDKY ANALÝZY PROFILOVÁNÍ .....	26
3.2.1 FUNKCE CATMATH: : PARSE ( ) .....	26
3.2.2 FUNKCE CATMATH: : DOUBLETOSTRING ( ) .....	26
3.2.3 FUNKCE CATMATH: : EVALUATE ( ) .....	26
3.2.4 DALŠÍ MOŽNOSTI OPTIMALIZACE .....	27
3.3 ZÁVĚR .....	27

# 1 PROFILOVACÍ PROSTŘEDÍ

## 1.1 INFORMACE O SYSTÉMU – HONZIKSICK@CAREN

- **OS:** Ubuntu 22.04.4 LTS on Windows 10 x86\_64
- **Kernel:** 5.15.146.1-microsoft-standard-WSL2
- **Uptime:** 4 hours, 9 mins
- **Packages:** 1979 (dpkg), 10 (snap)
- **Shell:** bash 5.1.16
- **Terminal:** Windows Terminal
- **CPU:** AMD Ryzen 5 2600 Six-Core Processor 3.40 GHz
- **GPU:** MSI GeForce GTX 1660 Ti GAMING X 6G, 6 GB GDDR6  
(6fc0:00:00:0 Microsoft Corporation Device 008e)
- **Memory:** 1751MiB / 7921MiB

## 1.2 PROFILOVACÍ SOFTWARE

- **Profilovací software:** GNU gprof (GNU Binutils for Ubuntu) 2.38
- **Software generování grafů:** gprof2dot v2022.07.29
- **Počet cyklů profilování:** 20

## 1.3 ZPŮSOB PŘEKLADU PROGRAMŮ A KNIHOVEN

- **Programy a knihovny:**
  - **stddev:** Program na výpočet výběrové směrodatné odchylky.
  - **Calm CatCalc Core:** Matematická knihovna nad programem stddev.
- **Standardy jazyka:**
  - **-std=c++17:** Nastavuje standard jazyka na C++17.
- **Varování:**
  - **-Wall:** Zapne všechna standardní varování.
  - **-Wextra:** Zapne další varování, která nejsou zapnuta pomocí -Wall.
  - **-Werror:** Převeď všechna varování na chyby.
  - **-pedantic:** Vydá všechna varování, která vyžaduje C a C++.

- **Paralelizace:**
  - `-pthread`: Přidá podporu pro multithreading s použitím POSIX threads.
- **Ladění:**
  - `-g`: Generuje ladící informace pro použití s GDB debuggerem.
- **Profilování:**
  - `-pg`: Generuje extra kód pro použití s nástrojem gprof pro profilování výkonu.
- **Optimalizace:**
  - `-O0`: Optimalizace vypnuta.

## 1.4 LEGENDY DATOVÝCH TABULEK

### 1.4.1 TABULKA ČASU STRÁVENÉHO V JEDNOTLIVÝCH FUNKCÍCH

Každý vzorek se počítá jako 0,01 sekundy.

- **Time [%]:**
  - Procento celkové doby běhu programu využité touto funkcí.
- **Cumulative [s]:**
  - Celkový součet sekund strávených ve všech funkcích dohromady.
- **Self [s]:**
  - Počet sekund přiřazených pouze této funkci.
  - Toto je hlavní kritérium pro řazení v rámci této tabulky.
- **Calls:**
  - Počet volání této funkce, pokud je tato funkce profilována, jinak prázdné.
- **Self [ms/call]:**
  - Průměrný počet milisekund strávených v této funkci na volání, pokud je tato funkce profilována, jinak prázdné.
- **Total [ms/call]:**
  - Průměrný počet milisekund strávených v této funkci a jejích potomcích na volání, pokud je tato funkce profilována, jinak prázdné.
- **Name:**
  - Název funkce.
  - Toto je vedlejší kritérium pro řazení v rámci tabulky.
  - Index ukazuje umístění funkce v seznamu gprof.

### 1.4.2 GRAF VOLÁNÍ FUNKCÍ (CALL GRAPH)

Tato tabulka popisuje strom volání programu a byla seřazena podle celkového času stráveného v každé funkci a jejích potomcích. Každý záznam v této tabulce se skládá z několika řádků.

**Řádek s indexovým číslem na levém okraji** uvádí aktuální funkci. Řádky nad ním uvádějí funkce, které tuto funkci volaly, a řádky pod ním uvádějí funkce, které tato funkce volala. Tento řádek uvádí:

- **Index**
  - Jedinečné číslo přidělené každému prvku tabulky.
  - Indexová čísla jsou seřazena numericky.
  - Indexové číslo je vytisknuto vedle každého názvu funkce, aby bylo snazší zjistit, kde je funkce v tabulce.
- **Time [%]**
  - Toto je procento z celkového času, které bylo stráveno v této funkci a jejích potomcích.
  - Vzhledem k různým hlediskům, vyloučenými funkcemi atd. se tyto čísla nesečtou na 100 %.
- **Self**
  - Toto je celkové množství času stráveného v této funkci.
- **Children**
  - Toto je celkové množství času, které do této funkce přenesly její potomci.
- **Called**
  - Toto je počet volání funkce.
  - Pokud funkce volala sama sebe rekurzivně, číslo zahrnuje pouze nerekurzivní volání a je následováno „+“ a počtem rekurzivních volání.
- **Name**
  - Název aktuální funkce.
  - Indexové číslo je vytisknuto za ním.
  - Pokud je funkce členem cyklu, číslo cyklu je vytisknuto mezi názvem funkce a indexovým číslem.

**Pro rodiče funkce mají pole následující významy:**

- **self**
  - Toto je množství času, které bylo přímo přeneseno z funkce do tohoto rodiče.
- **children**
  - Toto je množství času, které bylo přeneseno z potomků funkce do tohoto rodiče.
- **called**
  - Toto je počet volání, které tento rodič provedl funkci / celkový počet volání funkce.
  - Rekurzivní volání funkce nejsou zahrnuta v čísle za /.
- **name**
  - Toto je název rodiče.
  - Indexové číslo rodiče je vytisknuto za ním.
  - Pokud je rodič členem cyklu, číslo cyklu je vytisknuto mezi názvem a indexovým číslem.
  - Pokud nelze určit rodiče funkce, pole jsou prázdná.

**Pro potomky funkce mají pole následující významy:**

- **self**
  - Toto je množství času, které bylo přímo přeneseno z potomka do funkce.
- **children**
  - Toto je množství času, které bylo přeneseno z potomků potomka do funkce.
- **called**
  - Toto je počet volání, které funkce provedla tomuto potomku / celkový počet volání potomka.
  - Rekurzivní volání potomkem nejsou uvedena v čísle za /.
- **name**
  - Toto je název potomka.
  - Indexové číslo potomka je vytisknuto za ním.
  - Pokud je potomek členem cyklu, číslo cyklu je vytisknuto mezi názvem a indexovým číslem.

Pokud jsou v grafu volání nějaké cykly, existuje záznam pro celý cyklus. Tento záznam ukazuje, kdo volal cyklus (jako rodiče) a členy cyklu (jako potomky). Záznam „+“ rekurzivních volání ukazuje počet volání funkcí, které byly interní pro cyklus, a záznam volání pro každého člena ukazuje, kolikrát byl volán od ostatních členů cyklu.

## 2 VÝSLEDKY PROFILOVÁNÍ

### 2.1 PRO POČET $10^1$ VZORKŮ

#### 2.1.1 TABULKA ČASU STRÁVENÉHO V JEDNOTLIVÝCH FUNKCÍCH

Time [%]	Cumulative [s]	Self [s]	Calls	Self [ms/call]	Total [ms/call]	Name
0,00	0,00	0,00	64617	0,00	0,00	catMath::absVal()
0,00	0,00	0,00	19362	0,00	0,00	catMath::isOperator()
0,00	0,00	0,00	8652	0,00	0,00	catMath::power()
0,00	0,00	0,00	1176	0,00	0,00	catMath::priority()
0,00	0,00	0,00	903	0,00	0,00	catMath::evaluateOperation()
0,00	0,00	0,00	882	0,00	0,00	catMath::doubleToString()
0,00	0,00	0,00	525	0,00	0,00	catMath::formatInput()
0,00	0,00	0,00	525	0,00	0,00	catMath::pairParenthesis()
0,00	0,00	0,00	525	0,00	0,00	catMath::removeMultSpaces()
0,00	0,00	0,00	525	0,00	0,00	catMath::parse()
0,00	0,00	0,00	525	0,00	0,00	catMath::postfix()
0,00	0,00	0,00	525	0,00	0,00	catMath::evaluate()
0,00	0,00	0,00	525	0,00	0,00	catMath::calculate()
0,00	0,00	0,00	420	0,00	0,00	catMath::evalAdd()
0,00	0,00	0,00	105	0,00	0,00	catMath::evalNeg()
0,00	0,00	0,00	42	0,00	0,00	catMath::evalDiv()
0,00	0,00	0,00	42	0,00	0,00	catMath::evalMul()
0,00	0,00	0,00	42	0,00	0,00	catMath::evalSub()
0,00	0,00	0,00	21	0,00	0,00	catMath::root()
0,00	0,00	0,00	21	0,00	0,00	catStddev::readDataFromStdin()
0,00	0,00	0,00	21	0,00	0,00	catStddev::standardDeviation()
0,00	0,00	0,00	21	0,00	0,00	catStddev::readData()

## 2.1.2 GRAF VOLÁNÍ FUNKCÍ (CALL GRAPH)

Index	Time [%]	Self	Children	Called	Name
[9]	0.0	0.00	0.00	42/64617	catMath::evalMul()
		0.00	0.00	12663/64617	catMath::root()
		0.00	0.00	51912/64617	catMath::power()
		0.00	0.00	64617	catMath::absVal()
[28]	0.0	0.00	0.00	1722/19362	catMath::evaluate()
		0.00	0.00	17640/19362	catMath::parse()
		0.00	0.00	19362	catMath::isOperator()
[55]	0.0	0.00	0.00	231/8652	catMath::evaluateOperation()
		0.00	0.00	8421/8652	catMath::root()
		0.00	0.00	8652	catMath::power()
		0.00	0.00	51912/64617	catMath::absVal()
[110]	0.0	0.00	0.00	1176/1176	catMath::postfix()
		0.00	0.00	1176	catMath::priority()
[120]	0.0	0.00	0.00	903/903	catMath::evaluate()
		0.00	0.00	903	catMath::evaluateOperation()
		0.00	0.00	420/420	catMath::evalAdd()
		0.00	0.00	252/882	catMath::doubleToString()
		0.00	0.00	231/8652	catMath::power()
		0.00	0.00	105/105	catMath::evalNeg()
		0.00	0.00	42/42	catMath::evalSub()
		0.00	0.00	42/42	catMath::evalMul()
		0.00	0.00	42/42	catMath::evalDiv()
		0.00	0.00	21/21	catMath::root()
		0.00	0.00	21/882	catMath::evalSub()
		0.00	0.00	42/882	catMath::evalMul()
		0.00	0.00	42/882	catMath::evalDiv()
		0.00	0.00	105/882	catMath::evalNeg()



## VÝSLEDKY PROFILOVÁNÍ

[134]	0.0	0.00	0.00	252/882	catMath::evaluateOperation()
		0.00	0.00	420/882	catMath::evalAdd()
		0.00	0.00	882	catMath::doubleToString()
[150]	0.0	0.00	0.00	525/525	catMath::calculate()
		0.00	0.00	525	catMath::formatInput()
		0.00	0.00	525/525	catMath::removeMultSpaces()
		0.00	0.00	525/525	catMath::pairParenthesis()
[151]	0.0	0.00	0.00	525/525	catMath::formatInput()
		0.00	0.00	525	catMath::pairParenthesis()
[152]	0.0	0.00	0.00	525/525	catMath::formatInput()
		0.00	0.00	525	catMath::removeMultSpaces()
[153]	0.0	0.00	0.00	525/525	catMath::calculate()
		0.00	0.00	525	catMath::parse()
		0.00	0.00	17640/19362	catMath::isOperator()
[154]	0.0	0.00	0.00	525/525	catMath::calculate()
		0.00	0.00	525	catMath::postfix()
		0.00	0.00	1176/1176	catMath::priority()
[155]	0.0	0.00	0.00	525/525	catMath::calculate()
		0.00	0.00	525	catMath::evaluate()
		0.00	0.00	1722/19362	catMath::isOperator()
		0.00	0.00	903/903	catMath::evaluateOperation()
[156]	0.0	0.00	0.00	105/525	catStddev::standardDeviation()
		0.00	0.00	420/525	catStddev::readData()
		0.00	0.00	525	catMath::calculate()
		0.00	0.00	525/525	catMath::formatInput()
		0.00	0.00	525/525	catMath::parse()
		0.00	0.00	525/525	catMath::postfix()
		0.00	0.00	525/525	catMath::evaluate()

[166]	0.0	0.00	0.00	420/420	catMath::evaluateOperation()
		0.00	0.00	420	catMath::evalAdd()
		0.00	0.00	420/882	catMath::doubleToString()
[173]	0.0	0.00	0.00	105/105	catMath::evaluateOperation()
		0.00	0.00	105	catMath::evalNeg()
		0.00	0.00	105/882	catMath::doubleToString()
[183]	0.0	0.00	0.00	42/42	catMath::evaluateOperation()
		0.00	0.00	42	catMath::evalDiv()
		0.00	0.00	42/882	catMath::doubleToString()
[184]	0.0	0.00	0.00	42/42	catMath::evaluateOperation()
		0.00	0.00	42	catMath::evalMul()
		0.00	0.00	42/882	catMath::doubleToString()
		0.00	0.00	42/64617	catMath::absVal()
[185]	0.0	0.00	0.00	42/42	catMath::evaluateOperation()
		0.00	0.00	42	catMath::evalSub()
		0.00	0.00	21/882	catMath::doubleToString()
[188]	0.0	0.00	0.00	21/21	catMath::evaluateOperation()
		0.00	0.00	21	catMath::root()
		0.00	0.00	12663/64617	catMath::absVal()
		0.00	0.00	8421/8652	catMath::power()
[190]	0.0	0.00	0.00	21/21	catStddev::standardDeviation()
		0.00	0.00	21	catStddev::readDataFromStdin()
		0.00	0.00	21/21	catStddev::readData()
[191]	0.0	0.00	0.00	21/21	catStddev::main
		0.00	0.00	21	catStddev::standardDeviation()
		0.00	0.00	105/525	catMath::calculate()
		0.00	0.00	21/21	catStddev::readDataFromStdin()
[192]	0.0	0.00	0.00	21/21	catStddev::readDataFromStdin()
		0.00	0.00	21	catStddev::readData()
		0.00	0.00	420/525	catMath::calculate()

### 2.1.3 INDEXY FUNKCÍ

<b>[9]</b> catMath::absVal()	<b>[155]</b> catMath::evaluate()
<b>[28]</b> catMath::isOperator()	<b>[156]</b> catMath::calculate()
<b>[55]</b> catMath::power()	<b>[166]</b> catMath::evalAdd()
<b>[110]</b> catMath::priority()	<b>[173]</b> catMath::evalNeg()
<b>[120]</b> catMath::evaluateOperation()	<b>[183]</b> catMath::evalDiv()
<b>[134]</b> catMath::doubleToString()	<b>[184]</b> catMath::evalMul()
<b>[150]</b> catMath::formatInput()	<b>[185]</b> catMath::evalSub()
<b>[151]</b> catMath::pairParenthesis()	<b>[188]</b> catMath::root()
<b>[152]</b> catMath::removeMultSpaces()	<b>[190]</b> catStddev::readDataFromStdin()
<b>[153]</b> catMath::parse()	<b>[191]</b> catStddev::standardDeviation()
<b>[154]</b> catMath::postfix()	<b>[192]</b> catStddev::readData()

## 2.2 PRO POČET $10^3$ VZORKŮ

### 2.2.1 TABULKA ČASU STRÁVENÉHO V JEDNOTLIVÝCH FUNKCÍCH

Time [%]	Cumulative [s]	Self [s]	Calls	Self [ms/call]	Total [ms/call]	Name
5,88	0,12	0,01	42105	0,00	0,00	catMath::parse()
0,00	0,17	0,00	1604505	0,00	0,00	catMath::isOperator()
0,00	0,17	0,00	201768	0,00	0,00	catMath::priority()
0,00	0,17	0,00	189357	0,00	0,00	catMath::absVal()
0,00	0,17	0,00	101745	0,00	0,00	catMath::evaluateOperation()
0,00	0,17	0,00	101724	0,00	0,00	catMath::doubleToString()
0,00	0,17	0,00	42105	0,00	0,00	catMath::formatInput()
0,00	0,17	0,00	42105	0,00	0,00	catMath::pairParenthesis()
0,00	0,17	0,00	42105	0,00	0,00	catMath::removeMultSpaces()
0,00	0,17	0,00	42105	0,00	0,00	catMath::postfix()
0,00	0,17	0,00	42105	0,00	0,00	catMath::evaluate()
0,00	0,17	0,00	42105	0,00	0,00	catMath::calculate()
0,00	0,17	0,00	42000	0,00	0,00	catMath::evalAdd()
0,00	0,17	0,00	38577	0,00	0,00	catMath::evalNeg()
0,00	0,17	0,00	29442	0,00	0,00	catMath::power()
0,00	0,17	0,00	42	0,00	0,00	catMath::evalDiv()
0,00	0,17	0,00	42	0,00	0,00	catMath::evalMul()
0,00	0,17	0,00	42	0,00	0,00	catMath::evalSub()
0,00	0,17	0,00	21	0,00	0,00	catMath::root()
0,00	0,17	0,00	21	0,00	7,60	catStddev::readDataFromStdin()
0,00	0,17	0,00	21	0,00	7,86	catStddev::standardDeviation()
0,00	0,17	0,00	21	0,00	7,60	catStddev::readData()

## 2.2.2 GRAF VOLÁNÍ FUNKCÍ (CALL GRAPH)

Index	Time [%]	Self	Children	Called	Name
[1]	97.1	0.00	0.16		main [1]
		0.00	0.16	21/21	catStddev::standardDeviation()
[2]	97.1	0.00	0.16	21/21	main
		0.00	0.16	21	catStddev::standardDeviation()
		0.00	0.16	21/21	catStddev::readDataFromStdin()
		0.00	0.00	105/42105	catMath::calculate()
[3]	93.9	0.00	0.16	21/21	catStddev::standardDeviation()
		0.00	0.16	21	catStddev::readDataFromStdin()
		0.00	0.16	21/21	catStddev::readData()
[4]	93.9	0.00	0.16	21/21	catStddev::readDataFromStdin()
		0.00	0.16	21	catStddev::readData()
		0.00	0.15	42000/42105	catMath::calculate()
[5]	87.6	0.00	0.00	105/42105	catStddev::standardDeviation()
		0.00	0.15	42000/42105	catStddev::readData()
		0.00	0.15	42105	catMath::calculate()
		0.00	0.06	42105/42105	catMath::evaluate()
		0.01	0.02	42105/42105	catMath::parse()
		0.00	0.03	42105/42105	catMath::postfix()
		0.00	0.01	42105/42105	catMath::formatInput()
[8]	33.4	0.00	0.06	42105/42105	catMath::calculate()
		0.00	0.06	42105	catMath::evaluate()
		0.00	0.01	101745/101745	catMath::evaluateOperation()
		0.00	0.00	211890/1604505	catMath::isOperator()
[9]	20.4	0.01	0.02	42105/42105	catMath::calculate()
		0.01	0.02	42105	catMath::parse()
		0.00	0.00	1392615/1604505	catMath::isOperator()

[10]	19.5	0.00	0.03	42105/42105	catMath::calculate()
		0.00	0.03	42105	catMath::postfix()
		0.00	0.00	201768/201768	catMath::priority()
[20]	7.2	0.00	0.01	101745/101745	catMath::evaluate()
		0.00	0.01	101745	catMath::evaluateOperation()
		0.00	0.00	42000/42000	catMath::evalAdd()
		0.00	0.00	38577/38577	catMath::evalNeg()
		0.00	0.00	21042/101724	catMath::doubleToString()
		0.00	0.00	21021/29442	catMath::power()
		0.00	0.00	42/42	catMath::evalSub()
		0.00	0.00	42/42	catMath::evalMul()
		0.00	0.00	42/42	catMath::evalDiv()
		0.00	0.00	21/21	catMath::root()
[39]	3.8	0.00	0.01	42105/42105	catMath::calculate()
		0.00	0.01	42105	catMath::formatInput()
		0.00	0.00	42105/42105	catMath::removeMultSpaces()
		0.00	0.00	42105/42105	catMath::pairParenthesis()
[49]	1.9	0.00	0.00	42105/42105	catMath::formatInput()
		0.00	0.00	42105	catMath::pairParenthesis()
[50]	1.9	0.00	0.00	42105/42105	catMath::formatInput()
		0.00	0.00	42105	catMath::removeMultSpaces()
[92]	0.0	0.00	0.00	211890/1604505	catMath::evaluate()
		0.00	0.00	1392615/1604505	catMath::parse()
		0.00	0.00	1604505	catMath::isOperator()
[136]	0.0	0.00	0.00	201768/201768	catMath::postfix()
		0.00	0.00	201768	catMath::priority()
[137]	0.0	0.00	0.00	42/189357	catMath::evalMul()
		0.00	0.00	12663/189357	catMath::root()
		0.00	0.00	176652/189357	catMath::power()
		0.00	0.00	189357	catMath::absVal()

## VÝSLEDKY PROFILOVÁNÍ

		0.00	0.00	21/101724	catMath::evalSub()
		0.00	0.00	42/101724	catMath::evalMul()
		0.00	0.00	42/101724	catMath::evalDiv()
		0.00	0.00	21042/101724	catMath::evaluateOperation()
		0.00	0.00	38577/101724	catMath::evalNeg()
		0.00	0.00	42000/101724	catMath::evalAdd()
[149]	0.0	0.00	0.00	101724	catMath::doubleToString()
[175]	0.0	0.00	0.00	42000/42000	catMath::evaluateOperation()
		0.00	0.00	42000	catMath::evalAdd()
		0.00	0.00	42000/101724	catMath::doubleToString()
[176]	0.0	0.00	0.00	38577/38577	catMath::evaluateOperation()
		0.00	0.00	38577	catMath::evalNeg()
		0.00	0.00	38577/101724	catMath::doubleToString()
[177]	0.0	0.00	0.00	8421/29442	catMath::root()
		0.00	0.00	21021/29442	catMath::evaluateOperation()
		0.00	0.00	29442	catMath::power()
		0.00	0.00	176652/189357	catMath::absVal()
[189]	0.0	0.00	0.00	42/42	catMath::evaluateOperation()
		0.00	0.00	42	catMath::evalDiv()
		0.00	0.00	42/101724	catMath::doubleToString()
[190]	0.0	0.00	0.00	42/42	catMath::evaluateOperation()
		0.00	0.00	42	catMath::evalMul()
		0.00	0.00	42/101724	catMath::doubleToString()
		0.00	0.00	42/189357	catMath::absVal()
[191]	0.0	0.00	0.00	42/42	catMath::evaluateOperation()
		0.00	0.00	42	catMath::evalSub()
		0.00	0.00	21/101724	catMath::doubleToString()
[194]	0.0	0.00	0.00	21/21	catMath::evaluateOperation()
		0.00	0.00	21	catMath::root()
		0.00	0.00	12663/189357	catMath::absVal()
		0.00	0.00	8421/29442	catMath::power()

### 2.2.3 INDEXY FUNKCÍ

<b>[2]</b> catStddev::standardDeviation()	<b>[92]</b> catMath::isOperator()
<b>[3]</b> catStddev::readDataFromStdin()	<b>[136]</b> catMath::priority()
<b>[4]</b> catStddev::readData()	<b>[137]</b> catMath::absVal()
<b>[5]</b> catMath::calculate()	<b>[149]</b> catMath::doubleToString()
<b>[8]</b> catMath::evaluate()	<b>[175]</b> catMath::evalAdd()
<b>[9]</b> catMath::parse()	<b>[176]</b> catMath::evalNeg()
<b>[10]</b> catMath::postfix()	<b>[177]</b> catMath::power()
<b>[20]</b> catMath::evaluateOperation()	<b>[189]</b> catMath::evalDiv()
<b>[39]</b> catMath::formatInput()	<b>[190]</b> catMath::evalMul()
<b>[49]</b> catMath::pairParenthesis()	<b>[191]</b> catMath::evalSub()
<b>[50]</b> catMath::removeMultSpaces()	<b>[194]</b> catMath::root()



2.3 PRO POČET  $10^6$  VZORKŮ

## 2.3.1 TABULKA ČASU STRÁVENÉHO V JEDNOTLIVÝCH FUNKCÍCH

Time [%]	Cumulative [s]	Self [s]	Calls	Self [ms/call]	Total [ms/call]	Name
2,79	47,52	7,28	42000105	0,00	0,00	catMath::parse()
2,01	94,13	5,25	104995338	0,00	0,00	catMath::doubleToString()
1,65	107,79	4,30	42000105	0,00	0,00	catMath::evaluate()
1,43	131,52	3,73	42000105	0,00	0,00	catMath::postfix()
1,39	135,16	3,63	1673233275	0,00	0,00	catMath::isOperator()
1,09	143,79	2,84	104995737	0,00	0,00	catMath::evaluateOperation()
0,91	156,51	2,38	42000105	0,00	0,00	catMath::removeMultSpaces()
0,77	178,25	2,01	42000105	0,00	0,00	catMath::formatInput()
0,59	191,92	1,54	42000105	0,00	0,00	catMath::calculate()
0,51	203,28	1,33	209977152	0,00	0,00	catMath::priority()
0,46	212,07	1,20	42000105	0,00	0,00	catMath::pairParenthesis()
0,41	222,09	1,06	21	0,05	11,54	catStddev::readData()
0,26	235,40	0,69	41995569	0,00	0,00	catMath::evalNeg()
0,20	243,88	0,53	21002730	0,00	0,00	catMath::power()
0,20	244,41	0,53	42000000	0,00	0,00	catMath::evalAdd()
0,10	251,79	0,27	-----	-----	-----	catMath::associativity()
0,08	253,68	0,21	126020475	0,00	0,00	catMath::absVal()
0,00	260,84	0,01	-----	-----	-----	catMath::factorial()
0,00	260,86	0,00	42	0,00	0,00	catMath::evalDiv()
0,00	260,86	0,00	42	0,00	0,00	catMath::evalMul()
0,00	260,86	0,00	42	0,00	0,00	catMath::evalSub()
0,00	260,86	0,00	21	0,00	0,00	catMath::root()
0,00	260,86	0,00	21	0,00	11,54	catStddev::readDataFromStdin()
0,00	260,86	0,00	21	0,00	11,65	catStddev::standardDeviation()

## 2.3.2 GRAF VOLÁNÍ FUNKCÍ (CALL GRAPH)

Index	Time [%]	Self	Children	Called	Name
[1]	93.8	0.00	244.68		main [1]
		0.00	244.68	21/21	catStddev::standardDeviation()
[2]	93.8	0.00	244.68	21/21	main
		0.00	244.68	21	catStddev::standardDeviation()
		0.00	242.36	21/21	catStddev::readDataFromStdin()
		0.00	0.00	105/42000105	catMath::calculate()
[3]	92.9	0.00	242.36	21/21	catStddev::standardDeviation()
		0.00	242.36	21	catStddev::readDataFromStdin()
		1.06	241.30	21/21	catStddev::readData()
[4]	92.9	1.06	241.30	21/21	catStddev::readDataFromStdin()
		1.06	241.30	21	catStddev::readData()
		1.54	237.49	42000000/42000105	catMath::calculate()
[5]	91.6	0.00	0.00	105/42000105	catStddev::standardDeviation()
		1.54	237.49	42000000/42000105	catStddev::readData()
		1.54	237.49	42000105	catMath::calculate()
		4.30	82.88	42000105/42000105	catMath::evaluate()
		3.73	50.16	42000105/42000105	catMath::postfix()
		7.28	45.29	42000105/42000105	catMath::parse()
		2.01	18.60	42000105/42000105	catMath::formatInput()
[6]	33.4	4.30	82.88	42000105/42000105	catMath::calculate()
		4.30	82.88	42000105	catMath::evaluate()
		2.84	16.94	104995737/104995737	catMath::evaluateOperation()
		0.48	0.00	220485846/1673233275	catMath::isOperator()
[9]	20.7	3.73	50.16	42000105/42000105	catMath::calculate()
		3.73	50.16	42000105	catMath::postfix()
		1.33	1.62	209977152/209977152	catMath::priority()

# VÝSLEDKY PROFILOVÁNÍ

[10]	20.2	7.28	45.29	42000105/42000105	catMath::calculate()
		7.28	45.29	42000105	catMath::parse()
		3.16	0.00	1452747429/1673233275	catMath::isOperator()
[15]	7.9	2.01	18.60	42000105/42000105	catMath::calculate()
		2.01	18.60	42000105	catMath::formatInput()
		2.38	7.25	42000105/42000105	catMath::removeMultSpaces()
		1.20	7.25	42000105/42000105	catMath::pairParenthesis()
[18]	7.6	2.84	16.94	104995737/104995737	catMath::evaluate()
		2.84	16.94	104995737	catMath::evaluateOperation()
		0.53	3.55	42000000/42000000	catMath::evalAdd()
		0.69	2.97	41995569/41995569	catMath::evalNeg()
		1.05	0.14	21000042/104995338	catMath::doubleToString()
		0.53	0.22	21000021/21002730	catMath::power()
		0.00	0.00	21/21	catMath::root()
		0.00	0.00	42/42	catMath::evalMul()
		0.00	0.00	42/42	catMath::evalDiv()
		0.00	0.00	42/42	catMath::evalSub()
[34]	3.7	2.38	7.25	42000105/42000105	catMath::formatInput()
		2.38	7.25	42000105	catMath::removeMultSpaces()
[38]	3.2	1.20	7.25	42000105/42000105	catMath::formatInput()
		1.20	7.25	42000105	catMath::pairParenthesis()
[44]	2.3	0.00	0.00	21/104995338	catMath::evalSub()
		0.00	0.00	42/104995338	catMath::evalMul()
		0.00	0.00	42/104995338	catMath::evalDiv()
		1.05	0.14	21000042/104995338	catMath::evaluateOperation()
		2.10	0.29	41995569/104995338	catMath::evalNeg()
		2.10	0.29	41999622/104995338	catMath::evalAdd()
		5.25	0.72	104995338	catMath::doubleToString()

# **PROTOKOL Z PROFILOVÁNÍ CALM CATCALC**

[57]	1.6	0.53	3.55	42000000/42000000	catMath::evaluateOperation()
		0.53	3.55	42000000	catMath::evalAdd()
		2.10	0.29	41999622/104995338	catMath::doubleToString()
[62]	1.4	0.69	2.97	41995569/41995569	catMath::evaluateOperation()
		0.69	2.97	41995569	catMath::evalNeg()
		2.10	0.29	41995569/104995338	catMath::doubleToString()
[63]	1.4	0.48	0.00	220485846/1673233275	catMath::evaluate()
		3.16	0.00	1452747429/1673233275	catMath::parse()
		3.63	0.00	1673233275	catMath::isOperator()
[67]	1.1	1.33	1.62	209977152/209977152	catMath::postfix()
		1.33	1.62	209977152	catMath::priority()
[152]	0.1	0.00	0.00	42/126020475	catMath::evalMul()
		0.00	0.00	4095/126020475	catMath::root()
		0.21	0.00	126016338/126020475	catMath::power()
		0.21	0.00	126020475	catMath::absVal()
[219]	0.0	0.00	0.00	21/21	catMath::evaluateOperation()
		0.00	0.00	21	catMath::root()
		0.00	0.00	2709/21002730	catMath::power()
		0.00	0.00	4095/126020475	catMath::absVal()
[222]	0.0	0.00	0.00	42/42	catMath::evaluateOperation()
		0.00	0.00	42	catMath::evalMul()
		0.00	0.00	42/104995338	catMath::doubleToString()
		0.00	0.00	42/126020475	catMath::absVal()
[223]	0.0	0.00	0.00	42/42	catMath::evaluateOperation()
		0.00	0.00	42	catMath::evalDiv()
		0.00	0.00	42/104995338	catMath::doubleToString()
[224]	0.0	0.00	0.00	42/42	catMath::evaluateOperation()
		0.00	0.00	42	catMath::evalSub()
		0.00	0.00	21/104995338	catMath::doubleToString()

### 2.3.3 INDEXY FUNKCÍ

<b>[2]</b> catStddev::standardDeviation()	<b>[57]</b> catMath::evalAdd()
<b>[3]</b> catStddev::readDataFromStdin()	<b>[62]</b> catMath::evalNeg()
<b>[4]</b> catStddev::readData()	<b>[63]</b> catMath::isOperator()
<b>[5]</b> catMath::calculate()	<b>[67]</b> catMath::priority()
<b>[6]</b> catMath::evaluate()	<b>[126]</b> catMath::power()
<b>[9]</b> catMath::postfix()	<b>[148]</b> catMath::associativity()
<b>[10]</b> catMath::parse()	<b>[152]</b> catMath::absVal()
<b>[15]</b> catMath::formatInput()	<b>[214]</b> catMath::factorial()
<b>[18]</b> catMath::evaluateOperation()	<b>[219]</b> catMath::root()
<b>[34]</b> catMath::removeMultSpaces()	<b>[222]</b> catMath::evalMul()
<b>[38]</b> catMath::pairParenthesis()	<b>[223]</b> catMath::evalDiv()
<b>[44]</b> catMath::doubleToString()	<b>[224]</b> catMath::evalSub()

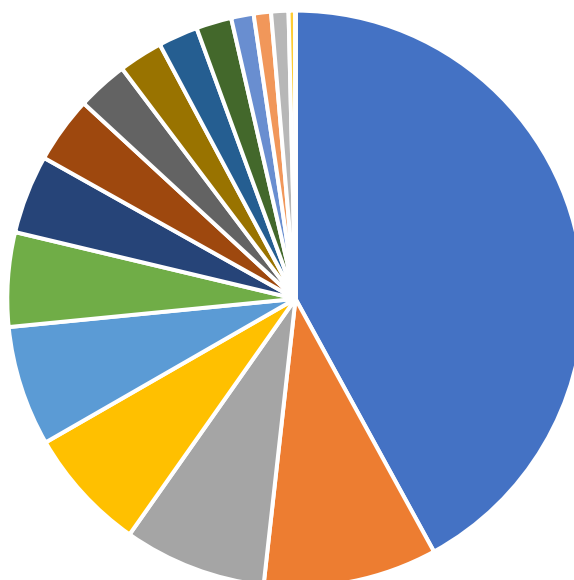
### 3 SOUHRNNÉ STATISTIKY A ANALÝZA VÝSLEDKŮ

#### 3.1 TABULKA ČASU STRÁVENÉHO V JEDNOTLIVÝCH FUNKCÍCH

##### 3.1.1 SEŘAZENO DLE TIME [%]

ID	Name	Time [%]	Cumulative [s]	Self [s]	Calls	Self [ms/call]	Total [ms/call]
1	catMath::parse()	8,67	47,64	7,29	42042735	0	0
2	catMath::doubleToString()	2,01	94,3	5,25	105097944	0	0
3	catMath::evaluate()	1,65	107,96	4,3	42042735	0	0
4	catMath::postfix()	1,43	131,69	3,73	42042735	0	0
5	catMath::isOperator()	1,39	135,33	3,63	1674857142	0	0
6	catMath::evaluateOperation()	1,09	143,96	2,84	105098385	0	0
7	catMath::removeMultSpaces()	0,91	156,68	2,38	42042735	0	0
8	catMath::formatInput()	0,77	178,42	2,01	42042735	0	0
9	catMath::calculate()	0,59	192,09	1,54	42042735	0	0
10	catMath::priority()	0,51	203,45	1,33	210180096	0	0
11	catMath::pairParenthesis()	0,46	212,24	1,2	42042735	0	0
12	catStddev::readData()	0,41	222,26	1,06	63	0,05	19,14
13	catMath::evalNeg()	0,26	235,57	0,69	42034251	0	0
14	catMath::evalAdd()	0,2	244,58	0,53	42042420	0	0
15	catMath::power()	0,2	244,05	0,53	21040824	0	0
16	catMath::absVal()	0,08	253,85	0,21	126274449	0	0
17	catMath::evalDiv()	0	261,03	0	126	0	0
18	catMath::evalMul()	0	261,03	0	126	0	0
19	catMath::evalSub()	0	261,03	0	126	0	0
20	catMath::root()	0	261,03	0	63	0	0
21	catStddev::readDataFromStdin()	0	261,03	0	63	0	19,14
22	catStddev::standardDeviation()	0	261,03	0	63	0	19,51

Čas strávený v jednotlivých funkcích dle Time [%]



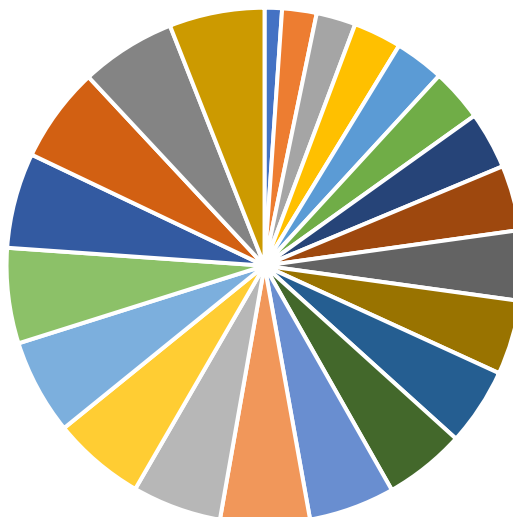
- |                                  |                                  |
|----------------------------------|----------------------------------|
| ■ catMath::parse()               | ■ catMath::doubleToString()      |
| ■ catMath::evaluate()            | ■ catMath::postfix()             |
| ■ catMath::isOperator()          | ■ catMath::evaluateOperation()   |
| ■ catMath::removeMultSpaces()    | ■ catMath::formatInput()         |
| ■ catMath::calculate()           | ■ catMath::priority()            |
| ■ catMath::pairParenthesis()     | ■ catStddev::readData()          |
| ■ catMath::evalNeg()             | ■ catMath::evalAdd()             |
| ■ catMath::power()               | ■ catMath::absVal()              |
| ■ catMath::evalDiv()             | ■ catMath::evalMul()             |
| ■ catMath::evalSub()             | ■ catMath::root()                |
| ■ catStddev::readDataFromStdin() | ■ catStddev::standardDeviation() |

### 3.1.2 SEŘAZENO DLE COMULATIVE [S]

ID	Name	Time [%]	Cumulative [s]	Self [s]	Calls	Self [ms/call]	Total [ms/call]
1	catMath::evalDiv()	0	261,03	0	126	0	0
2	catMath::evalMul()	0	261,03	0	126	0	0
3	catMath::evalSub()	0	261,03	0	126	0	0
4	catMath::root()	0	261,03	0	63	0	0
5	catStddev::readDataFromStdin()	0	261,03	0	63	0	19,14
6	catStddev::standardDeviation()	0	261,03	0	63	0	19,51
7	catMath::absVal()	0,08	253,85	0,21	126274449	0	0
8	catMath::evalAdd()	0,2	244,58	0,53	42042420	0	0
9	catMath::power()	0,2	244,05	0,53	21040824	0	0
10	catMath::evalNeg()	0,26	235,57	0,69	42034251	0	0
11	catStddev::readData()	0,41	222,26	1,06	63	0,05	19,14
12	catMath::pairParenthesis()	0,46	212,24	1,2	42042735	0	0
13	catMath::priority()	0,51	203,45	1,33	210180096	0	0
14	catMath::calculate()	0,59	192,09	1,54	42042735	0	0
15	catMath::formatInput()	0,77	178,42	2,01	42042735	0	0
16	catMath::removeMultSpaces()	0,91	156,68	2,38	42042735	0	0
17	catMath::evaluateOperation()	1,09	143,96	2,84	105098385	0	0
18	catMath::isOperator()	1,39	135,33	3,63	1674857142	0	0
19	catMath::postfix()	1,43	131,69	3,73	42042735	0	0
20	catMath::evaluate()	1,65	107,96	4,3	42042735	0	0
21	catMath::doubleToString()	2,01	94,3	5,25	105097944	0	0
22	catMath::parse()	8,67	47,64	7,29	42042735	0	0



Čas strávený v jednotlivých funkcích dle Cumulative [s]



■ catMath::parse()	■ catMath::doubleToString()	■ catMath::evaluate()
■ catMath::postfix()	■ catMath::isOperator()	■ catMath::evaluateOperation()
■ catMath::removeMultSpaces()	■ catMath::formatInput()	■ catMath::calculate()
■ catMath::priority()	■ catMath::pairParenthesis()	■ catStddev::readData()
■ catMath::evalNeg()	■ catMath::evalAdd()	■ catMath::power()
■ catMath::absVal()	■ catMath::evalDiv()	■ catMath::evalMul()
■ catMath::evalSub()	■ catMath::root()	■ catStddev::readDataFromStdin()
■ catStddev::standardDeviation()		

## 3.2 VÝSLEDKY ANALÝZY PROFILOVÁNÍ

### 3.2.1 FUNKCE `CATMATH::PARSE()`

Z tabulky 3.1.1, která je seřazena podle procentuálního času stráveného v dané funkci, vidíme, že nejvíce času (8,67 %) stráví program ve funkci `catMath::parse()`. Tato funkce byla sice zavolána 42042735krát, ale například funkce `catMath::absVal()` byla zavolána přibližně 3krát vícekrát než funkce `catMath::parse()`, a přitom v ní byl z procentuálního hlediska stráven pouze zlomek času (0,08 %).

Funkce `catMath::parse()` prochází vstupní řetězec znak po znaku a rozděluje ho na tokeny. Každý token je buď operátor, závorka nebo číslo.

Optimalizace funkce `catMath::parse()` by mohla být provedena například předalokací paměti pro vektor výsledků nebo použitím pro tokeny `std::stringstream` namísto `std::string`.

### 3.2.2 FUNKCE `CATMATH::DOUBLETOSTRING()`

Další funkce, ve které se stráví významný podíl času, je `catMath::doubleToString()`, která zabírá 2,01 % času. Tato funkce byla zavolána v rámci všech profilování celkem 105097944krát, což je více než dvojnásobek oproti předchozí funkci. I když tato funkce spotřebuje méně času než funkce `catMath::parse()`, tak na základě vysokého počtu volání je tato funkce potenciálním místem pro optimalizaci.

Funkce například zjišťuje počet číslic v čísle X pomocí cyklu `while`. Tento cyklus se může stát velmi náročným na výkon, pokud je X velké číslo.

Jednou z možností optimalizace by bylo použití logaritmické funkce pro výpočet počtu číslic, což by mohlo být rychlejší než postupné dělení číslem 10.

Další možnou optimalizací je použití rychlejších metod pro převod čísel na řetězce, čehož by se dalo docílit například užitím `std::to_string`, ačkoliv by to ale mohlo omezit kontrolu nad zaokrouhlováním.

### 3.2.3 FUNKCE `CATMATH::EVALUATE()`

Ve funkci `catMath::evaluate` se stráví 1,65 % celkového času: Účelem této funkce je vyhodnocování matematických výrazů.

Optimalizace této funkce by mohla zahrnovat použití rychlejších algoritmů pro vyhodnocení výrazů nebo předzpracování výrazů, aby se snížil počet operací potřebných k jejich vyhodnocení.

Další možnou optimalizací by mohlo být předzpracování výrazu, který je funkci předáván. Důvodem je fakt, že funkce obsahuje zanořené cykly `for`, tedy v případě složitého výrazu, může být provádění těl těchto cyklů velmi náročné na celkový výkon.

### 3.2.4 DALŠÍ MOŽNOSTI OPTIMALIZACE

Funkce **`catMath::isOperator()`** byla zavolána nejvícekrát (celkem 1674857142krát), ale strávilo se v ní pouze 1,39 % času, což v porovnání s počtem volání není mnoho. I přesto by mohla být tato funkce kandidátem na optimalizaci, právě kvůli četnosti jejího volání.

Z tabulky 3.1.2, která je seřazena podle kumulativního času, vidíme, že nejvíce času stráví program ve funkcích zajišťující základní matematické operace – **`catMath::evalDiv()`**, **`catMath::evalMul()`**, **`catMath::evalSub()`**, **`catMath::root()`**. Tyto funkce stráví stejný kumulativní čas (261,03 s), ale nebyly zavolány tolikrát jako předchozí funkce. To by mohlo naznačovat, že tyto funkce jsou poměrně náročné na výkon a mohly by být také kandidáty na optimalizaci.

## 3.3 ZÁVĚR

Na základě provedené analýzy je zřejmé, že nejnáročnějšími funkcemi v programu jsou **`catMath::parse`**, **`catMath::doubleToString`** a **`catMath::evaluate`**.

Optimalizace těchto funkcí by mohla významně přispět ke zlepšení celkového výkonu nejen programu `stddev`, ale také ke zrychlení celé matematické knihovny `Calm CatCalc Core`.

Důkladné zkoumání a úprava těchto funkcí by měla být prioritou při dalším vývoji a optimalizaci tohoto softwaru.