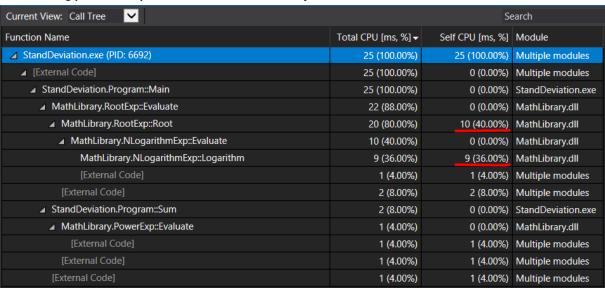# Profiling

Profilování bylo provedeno pomocí vestavěného profileru ve Visual Studiu.

Profilování bylo provedeno pro 10, 100 a 1000 vstupních hodnot (soubory s daty jsou ve složce profiling_data).

Program tráví nejvíce času ve funkcích PowerExp a RootExp (tedy mocnina a odmocnina čísla). V RootExp nejdéle trvá NLogarihtmExp. Při optimalizaci by tedy bylo vhodné zaměřit se právě funkci PoweExp a na funkci NLogarithmExp (čímž by se zoptimalizovala funkce RootExp).

Na následujících stránkách jsou snímky z profilingu, které zobrazují čas strávený ve funkcích a počet volání funkcí (zde je do času zahrnut i čas načítání dat).

Profiling pro 10 vstupních hodnot – čas strávený ve funkcích:

Function Name	Total CPU [ms, %]	Self CPU [ms, %]	Module
◢ StandDeviation.exe (PID: 6692)	25 (100.00%)	25 (100.00%)	Multiple modules
◢ [External Code]	25 (100.00%)	0 (0.00%)	Multiple modules
◢ StandDeviation.Program::Main	25 (100.00%)	0 (0.00%)	StandDeviation.exe
◢ MathLibrary.RootExp::Evaluate	22 (88.00%)	0 (0.00%)	MathLibrary.dll
◢ MathLibrary.RootExp::Root	20 (80.00%)	10 (40.00%)	MathLibrary.dll
◢ MathLibrary.NLogarithmExp::Evaluate	10 (40.00%)	0 (0.00%)	MathLibrary.dll
MathLibrary.NLogarithmExp::Logarithm	9 (36.00%)	9 (36.00%)	MathLibrary.dll
[External Code]	1 (4.00%)	1 (4.00%)	Multiple modules
[External Code]	2 (8.00%)	2 (8.00%)	Multiple modules
◢ StandDeviation.Program::Sum	2 (8.00%)	0 (0.00%)	StandDeviation.exe
◢ MathLibrary.PowerExp::Evaluate	1 (4.00%)	0 (0.00%)	MathLibrary.dll
[External Code]	1 (4.00%)	1 (4.00%)	Multiple modules
[External Code]	1 (4.00%)	1 (4.00%)	Multiple modules
[External Code]	1 (4.00%)	1 (4.00%)	Multiple modules

Profiling pro 10 vstupních hodnot – počet volání:

Function Name	Number of Calls	Elapsed Inclusive Time %	Elapsed Exclusive Time %
◢ StandDeviation.exe	0	100.00%	0.00%
◢ StandDeviation.Program.Main(string[])	1	100.00%	0.04%
▷ StandDeviation.Program.ReadNumbers()	1	99.34%	0.00%
MathLibrary.RootExp.Evaluate()	1	0.55%	0.55%
◢ StandDeviation.Program.Sum(class System.Collect...	2	0.04%	0.00%
MathLibrary.PowerExp.Evaluate()	20	0.03%	0.03%
MathLibrary.PowerExp..ctor(class MathLibrary.IE...	20	0.01%	0.01%
System.Collections.Generic.List`1.GetEnumerator()	2	0.00%	0.00%
MoveNext()	22	0.00%	0.00%
MathLibrary.Number..ctor(float64)	20	0.00%	0.00%
get_Current()	20	0.00%	0.00%
System.IDisposable.Dispose()	2	0.00%	0.00%
System.Console.WriteLine(float64)	1	0.01%	0.01%
MathLibrary.RootExp..ctor(class MathLibrary.IExpr...	1	0.01%	0.01%
System.Collections.Generic.List`1.get_Count()	2	0.00%	0.00%
MathLibrary.PowerExp.Evaluate()	1	0.00%	0.00%
MathLibrary.Number..ctor(float64)	4	0.00%	0.00%
MathLibrary.PowerExp..ctor(class MathLibrary.IExp...	1	0.00%	0.00%

Profiling pro 100 vstupních hodnot – čas strávený ve funkcích:

Function Name	Total CPU [ms, %] ▾	Self CPU [ms, %]
Current View: Call Tree		
⊿ StandDeviation.exe (PID: 13880)	18 (100.00%)	18 (100.00%)
⊿ [External Code]	18 (100.00%)	0 (0.00%)
⊿ StandDeviation.Program::Main	18 (100.00%)	0 (0.00%)
⊿ MathLibrary.RootExp::Evaluate	15 (83.33%)	0 (0.00%)
⊿ MathLibrary.RootExp::Root	14 (77.78%)	5 (27.78%)
▷ MathLibrary.NLogarithmExp::Evaluate	8 (44.44%)	0 (0.00%)
[External Code]	1 (5.56%)	1 (5.56%)
[External Code]	1 (5.56%)	1 (5.56%)
⊿ StandDeviation.Program::Sum	2 (11.11%)	0 (0.00%)
[External Code]	1 (5.56%)	1 (5.56%)
⊿ MathLibrary.PowerExp::Evaluate	1 (5.56%)	0 (0.00%)
[External Code]	1 (5.56%)	1 (5.56%)
[External Code]	1 (5.56%)	1 (5.56%)

Profiling pro 100 vstupních hodnot – počet volání:

Function Name	Number of Calls	Elapsed Inclusive Time % ▾	Elapsed Exclusive Time %
⊿ StandDeviation.exe	0	100.00%	0.00%
⊿ StandDeviation.Program.Main(string[])	1	100.00%	0.03%
▷ StandDeviation.Program.ReadNumbers()	1	99.50%	0.00%
MathLibrary.RootExp.Evaluate()	1	0.41%	0.41%
⊿ StandDeviation.Program.Sum(class System.Col...	2	0.03%	0.00%
MathLibrary.PowerExp.Evaluate()	200	0.03%	0.03%
MathLibrary.PowerExp..ctor(class MathLibrar...	200	0.01%	0.01%
MathLibrary.Number..ctor(float64)	200	0.00%	0.00%
System.Collections.Generic.List`1.GetEnumer...	2	0.00%	0.00%
MoveNext()	202	0.00%	0.00%
get_Current()	200	0.00%	0.00%
System.IDisposable.Dispose()	2	0.00%	0.00%
System.Console.WriteLine(float64)	1	0.01%	0.01%
MathLibrary.RootExp..ctor(class MathLibrary.IE...	1	0.00%	0.00%
System.Collections.Generic.List`1.get_Count()	2	0.00%	0.00%
MathLibrary.PowerExp.Evaluate()	1	0.00%	0.00%
MathLibrary.Number..ctor(float64)	4	0.00%	0.00%
MathLibrary.PowerExp..ctor(class MathLibrary.I...	1	0.00%	0.00%

Profiling pro 1000 vstupních hodnot – čas strávený ve funkcích:

Function Name	Total CPU [ms, %]	Self CPU [ms, %]
Current View: Call Tree		Se
◢ StandDeviation.exe (PID: 1132)	21 (100.00%)	21 (100.00%)
◢ [External Code]	21 (100.00%)	0 (0.00%)
◢ StandDeviation.Program::Main	21 (100.00%)	0 (0.00%)
◢ MathLibrary.RootExp::Evaluate	17 (80.95%)	0 (0.00%)
◢ MathLibrary.RootExp::Root	16 (76.19%)	7 (33.33%)
◢ MathLibrary.NLogarithmExp::Evaluate	8 (38.10%)	0 (0.00%)
MathLibrary.NLogarithmExp::Logarithm	7 (33.33%)	7 (33.33%)
[External Code]	1 (4.76%)	1 (4.76%)
[External Code]	1 (4.76%)	1 (4.76%)
[External Code]	1 (4.76%)	1 (4.76%)
◢ StandDeviation.Program::Sum	2 (9.52%)	0 (0.00%)
[External Code]	2 (9.52%)	2 (9.52%)
[External Code]	2 (9.52%)	2 (9.52%)

Profiling pro 100 vstupních hodnot – počet volání:

Function Name	Number of Calls	Elapsed Inclusive Time %	Elapsed Exclusive Time %
◢ StandDeviation.exe	0	100.00%	0.00%
◢ StandDeviation.Program.Main(string[])	1	100.00%	0.02%
▷ StandDeviation.Program.ReadNumbers()	1	99.59%	0.03%
MathLibrary.RootExp.Evaluate()	1	0.31%	0.31%
◢ StandDeviation.Program.Sum(class System.Collections.Ge...	2	0.07%	0.01%
MathLibrary.PowerExp.Evaluate()	2,000	0.04%	0.04%
MathLibrary.PowerExp..ctor(class MathLibrary.IExpressio...	2,000	0.01%	0.01%
get_Current()	2,000	0.00%	0.00%
MathLibrary.Number..ctor(float64)	2,000	0.00%	0.00%
MoveNext()	2,002	0.00%	0.00%
System.Collections.Generic.List`1.GetEnumerator()	2	0.00%	0.00%
System.IDisposable.Dispose()	2	0.00%	0.00%
MathLibrary.RootExp..ctor(class MathLibrary.IExpression,cl...	1	0.01%	0.01%
System.Console.WriteLine(float64)	1	0.00%	0.00%
System.Collections.Generic.List`1.get_Count()	2	0.00%	0.00%
MathLibrary.PowerExp.Evaluate()	1	0.00%	0.00%
MathLibrary.Number..ctor(float64)	4	0.00%	0.00%
MathLibrary.PowerExp..ctor(class MathLibrary.IExpression,...	1	0.00%	0.00%