

# Protokol z profilování aplikace na výpočet směrodatné odchylky

## 1 Úvod

Tento dokument shrnuje výsledky profilování aplikace, která slouží k výpočtu směrodatné odchylky. Aplikace vznikla jako součást projektu do předmětu IVS.

## 2 Podmínky testování

### 2.1 Testovací prostředí

CPU	Intel i5-6200u
RAM	8192 Mb
OS	Windows 10 (64bit)
Software	Visual Studio 2015

### 2.2 Profiler

Při profilování uvedené aplikace byl aplikován přístup instrumentace kódu za pomoci nativního profileru ve VS 2015.

### 2.3 Testovací data

S ohledem na zadání projektu byla aplikace testována pro tři sady čísel o 10, 100 a 1000 prvcích. Čísla byla vygenerována náhodným generátorem čísel v rozsahu 0-100.

## 3 Výsledky

### 3.1 Obecné shrnutí

V případě 10 a 1000 vzorků bylo nejvíce procesorového času využito na parsování vstupních dat. V případě 100 vzorků byla jen mírně náročnější funkce obsahující výpočet. Nejčastěji volanou funkcí ve všech případech byla funkce `addOperand` z třídy `Addition` z matematické knihovny, která je však výpočetně velmi nenáročná, a i z toho důvodu, přesto že byla volána nejčastěji, zabrala z celkového procesorového času ve všech případech maximálně 0,11%.

### 3.2 Závěr

V případě výpočtu směrodatné odchylky se jedná o triviální řešení a profilování takového programu tedy nepřináší nikterak překvapivé výsledky. Prostor pro optimalizace lze naléznout snad jen v oblasti parsování vstupních dat.

## 4 Přílohy

### 4.1 Tabulka volaných funkcí pro 100 prvků

FunctionName	NumCalls	InclusiveElapsedTimePercent	ExclusiveElapsedTimePercent	AvgInclusiveElapsedTime	AvgExclusiveElapsedTime
StandardDeviation.Program.Main(string[])	1	100,00	78,20	25,51	19,95
MathLib.Functions.Basic.Addition.Calculate()	2	10,60	10,60	1,35	1,35
System.IO.StreamReader..ctor(string)	1	5,34	5,34	1,36	1,36
MathLib.Functions.Basic.Division.Calculate(float64,float64)	2	2,24	2,24	0,29	0,29
System.Console.WriteLine(string)	2	1,96	1,96	0,25	0,25
System.Linq.Enumerable.ToArray(class System.Collections.Generic.IEnumerable<T>)	1	0,42	0,42	0,11	0,11
System.Linq.Enumerable.Select(class System.Collections.Generic.IEnumerable<T>,class System.Func<T,U>)	1	0,41	0,41	0,11	0,11
System.IO.TextReader.ReadLine()	21	0,26	0,26	0,00	0,00
StandardDeviation.Program.numParse()	1	6,66	0,18	1,70	0,05
System.String.Concat(object,object)	2	0,14	0,14	0,02	0,02
MathLib.Functions.Basic.Addition.AddOperand(float64)	162	0,11	0,11	0,00	0,00
MathLib.Functions.Basic.Addition..ctor()	2	0,05	0,05	0,01	0,01
StandardDeviation.Program.compute(float64[])	1	14,39	0,03	3,67	0,01
System.String.Concat(string,string)	20	0,02	0,02	0,00	0,00
System.String.Split(char[])	1	0,02	0,02	0,00	0,00
MathLib.Functions.Basic.Subtraction.Calculate(float64,float64)	81	0,01	0,01	0,00	0,00
MathLib.Functions.Advanced.Power.Calculate(float64)	81	0,01	0,01	0,00	0,00
MathLib.Functions.Basic.Division..ctor()	1	0,00	0,00	0,00	0,00
System.Func<T>..ctor(object,non-nullable int)	1	0,00	0,00	0,00	0,00

MathLib.Functions.Basic.Subtraction..ctor()	1	0,00	0,00	0,00	0,00
MathLib.Functions.Advanced.Root..ctor()	1	0,00	0,00	0,00	0,00
MathLib.Functions.Advanced.Power..ctor()	1	0,00	0,00	0,00	0,00
System.String.op_Inequality(string,string)	20	0,00	0,00	0,00	0,00
MathLib.Functions.Advanced.Root.Calculate(float64)	1	0,00	0,00	0,00	0,00