

Gprof

10 čísel

```
Flat profile:
```

Each sample counts as 0.01 seconds.
no time accumulated

% time	cumulative seconds	self seconds	calls	self Ts/call	total Ts/call	name
0.00	0.00	0.00	20	0.00	0.00	CalcMath::add(double, double)
0.00	0.00	0.00	11	0.00	0.00	CalcMath::pow(double, double)
0.00	0.00	0.00	2	0.00	0.00	CalcMath::div(double, double)
0.00	0.00	0.00	2	0.00	0.00	CalcMath::sub(double, double)
0.00	0.00	0.00	1	0.00	0.00	_GLOBAL_sub_I_main
0.00	0.00	0.00	1	0.00	0.00	__static_initialization_and_destruction_0(int, int)
0.00	0.00	0.00	1	0.00	0.00	CalcMath::mult(double, double)
0.00	0.00	0.00	1	0.00	0.00	CalcMath::root(double, double)

Call graph (explanation follows)

granularity: each sample hit covers 2 byte(s) no time propagated

index	% time	self	children	called	name
		0.00	0.00	20/20	main [6]
[8]	0.0	0.00	0.00	20	CalcMath::add(double, double) [8]

		0.00	0.00	11/11	main [6]
[9]	0.0	0.00	0.00	11	CalcMath::pow(double, double) [9]

		0.00	0.00	2/2	main [6]
[10]	0.0	0.00	0.00	2	CalcMath::div(double, double) [10]

		0.00	0.00	2/2	main [6]
[11]	0.0	0.00	0.00	2	CalcMath::sub(double, double) [11]

		0.00	0.00	1/1	libc_csu_init [21]
[12]	0.0	0.00	0.00	1	_GLOBAL_sub_I_main [12]
		0.00	0.00	1/1	__static_initialization_and_destruction_0(int, int) [13]

		0.00	0.00	1/1	_GLOBAL_sub_I_main [12]
[13]	0.0	0.00	0.00	1	__static_initialization_and_destruction_0(int, int) [13]

		0.00	0.00	1/1	main [6]
[14]	0.0	0.00	0.00	1	CalcMath::mult(double, double) [14]

		0.00	0.00	1/1	main [6]
[15]	0.0	0.00	0.00	1	CalcMath::root(double, double) [15]

100 čísel

```
Flat profile:
Each sample counts as 0.01 seconds.
no time accumulated

%   cumulative   self           self         total
time  seconds    seconds   calls   Ts/call   Ts/call  name
-----
0.00      0.00      0.00        200      0.00      0.00  CalcMath::add(double, double)
0.00      0.00      0.00       101      0.00      0.00  CalcMath::pow(double, double)
0.00      0.00      0.00         2      0.00      0.00  CalcMath::div(double, double)
0.00      0.00      0.00         2      0.00      0.00  CalcMath::sub(double, double)
0.00      0.00      0.00         1      0.00      0.00  _GLOBAL__sub_I main
0.00      0.00      0.00         1      0.00      0.00  __static_initialization_and_destruction_0(int, int)
0.00      0.00      0.00         1      0.00      0.00  CalcMath::mult(double, double)
0.00      0.00      0.00         1      0.00      0.00  CalcMath::root(double, double)

Call graph (explanation follows)

granularity: each sample hit covers 2 byte(s) no time propagated

index % time    self  children   called    name
-----
[8]    0.0      0.00    0.00   200/200    main [6]
-----
[9]    0.0      0.00    0.00  101/101    CalcMath::add(double, double) [8]
-----
[10]   0.0      0.00    0.00    2/2        main [6]
-----
[11]   0.0      0.00    0.00    2/2        CalcMath::pow(double, double) [9]
-----
[12]   0.0      0.00    0.00    2/2        main [6]
-----
[13]   0.0      0.00    0.00    2/2        CalcMath::div(double, double) [10]
-----
[14]   0.0      0.00    0.00    2/2        main [6]
-----
[15]   0.0      0.00    0.00    2/2        CalcMath::sub(double, double) [11]
-----
[16]   0.0      0.00    0.00    1/1        libc_csu_init [21]
-----
[17]   0.0      0.00    0.00     1/1        _GLOBAL__sub_I main [12]
-----
[18]   0.0      0.00    0.00     1/1        __static_initialization_and_destruction_0(int, int) [13]
-----
[19]   0.0      0.00    0.00     1/1        _GLOBAL__sub_I main [12]
-----
[20]   0.0      0.00    0.00     1/1        __static_initialization_and_destruction_0(int, int) [13]
-----
[21]   0.0      0.00    0.00     1/1        main [6]
-----
[22]   0.0      0.00    0.00     1/1        CalcMath::mult(double, double) [14]
-----
[23]   0.0      0.00    0.00     1/1        main [6]
-----
[24]   0.0      0.00    0.00     1/1        CalcMath::root(double, double) [15]
-----
```

1000 čísel

```
Flat profile:
Each sample counts as 0.01 seconds.
no time accumulated

%   cumulative   self           self         total
time  seconds    seconds   calls   Ts/call   Ts/call   name
0.00    0.00    0.00      2000    0.00      0.00   CalcMath::add(double, double)
0.00    0.00    0.00      1001    0.00      0.00   CalcMath::pow(double, double)
0.00    0.00    0.00         2    0.00      0.00   CalcMath::div(double, double)
0.00    0.00    0.00         2    0.00      0.00   CalcMath::sub(double, double)
0.00    0.00    0.00         1    0.00      0.00   _GLOBAL__sub_I_main
0.00    0.00    0.00         1    0.00      0.00   __static_initialization_and_destruction_0(int, int)
0.00    0.00    0.00         1    0.00      0.00   CalcMath::mult(double, double)
0.00    0.00    0.00         1    0.00      0.00   CalcMath::root(double, double)

Call graph (explanation follows)

granularity: each sample hit covers 2 byte(s) no time propagated

index % time    self  children    called      name
[8]    0.0      0.00   0.00    2000/2000    main [6]
-----
[8]    0.0      0.00   0.00    2000      CalcMath::add(double, double) [8]
-----
[9]    0.0      0.00   0.00    1001/1001    main [6]
-----
[9]    0.0      0.00   0.00    1001      CalcMath::pow(double, double) [9]
-----
[10]   0.0      0.00   0.00         2/2      main [6]
-----
[10]   0.0      0.00   0.00         2      CalcMath::div(double, double) [10]
-----
[11]   0.0      0.00   0.00         2/2      main [6]
-----
[11]   0.0      0.00   0.00         2      CalcMath::sub(double, double) [11]
-----
[12]   0.0      0.00   0.00         1/1      libc_csu_init [21]
-----
[12]   0.0      0.00   0.00         1      _GLOBAL__sub_I_main [12]
-----
[12]   0.0      0.00   0.00         1/1      __static_initialization_and_destruction_0(int, int) [13]
-----
[13]   0.0      0.00   0.00         1/1      _GLOBAL__sub_I_main [12]
-----
[13]   0.0      0.00   0.00         1      __static_initialization_and_destruction_0(int, int) [13]
-----
[14]   0.0      0.00   0.00         1/1      main [6]
-----
[14]   0.0      0.00   0.00         1      CalcMath::mult(double, double) [14]
-----
[15]   0.0      0.00   0.00         1/1      main [6]
-----
[15]   0.0      0.00   0.00         1      CalcMath::root(double, double) [15]
-----
```

18000000 čísel

```
Flat profile:
Each sample counts as 0.01 seconds.
%   cumulative   self           self       total
time  seconds    seconds   calls   ns/call  ns/call  name
88.94    0.51     0.51    36000000  0.84     0.84   CalcMath::add(double, double)
 5.28    0.54     0.03    18000001  1.12     1.12   CalcMath::pow(double, double)
 3.52    0.56     0.00         2     0.00     0.00   CalcMath::div(double, double)
 0.00    0.56     0.00         2     0.00     0.00   CalcMath::sub(double, double)
 0.00    0.56     0.00         1     0.00     0.00   _GLOBAL_sub_I_main
 0.00    0.56     0.00         1     0.00     0.00   __static_initialization_and_destruction_0(int, int)
 0.00    0.56     0.00         1     0.00     0.00   CalcMath::mult(double, double)
 0.00    0.56     0.00         1     0.00     0.00   CalcMath::root(double, double)

Call graph (explanation follows)

granularity: each sample hit covers 2 byte(s) for 1.79% of 0.56 seconds

index % time    self  children   called    name
-----
[1]  100.0   0.51   0.05      36000000/36000000  main [1]
      0.03   0.00 36000000/36000000  CalcMath::add(double, double) [2]
      0.02   0.00 18000001/18000001  CalcMath::pow(double, double) [3]
      0.00   0.00         2/2  CalcMath::div(double, double) [10]
      0.00   0.00         2/2  CalcMath::sub(double, double) [11]
      0.00   0.00         1/1  CalcMath::mult(double, double) [14]
      0.00   0.00         1/1  CalcMath::root(double, double) [15]
-----
[2]      0.03   0.00 36000000/36000000  main [1]
      0.03   0.00 36000000  CalcMath::add(double, double) [2]
-----
[3]      0.02   0.00 18000001/18000001  main [1]
      0.02   0.00 18000001  CalcMath::pow(double, double) [3]
-----
[10]     0.00   0.00         2/2  main [1]
      0.00   0.00         2  CalcMath::div(double, double) [10]
-----
[11]     0.00   0.00         2/2  main [1]
      0.00   0.00         2  CalcMath::sub(double, double) [11]
-----
[12]     0.00   0.00         1/1  libc_csu_init [21]
      0.00   0.00         1  _GLOBAL_sub_I_main [12]
      0.00   0.00         1/1  __static_initialization_and_destruction_0(int, int) [13]
-----
[13]     0.00   0.00         1/1  GLOBAL_sub_I_main [12]
      0.00   0.00         1  __static_initialization_and_destruction_0(int, int) [13]
-----
[14]     0.00   0.00         1/1  main [1]
      0.00   0.00         1  CalcMath::mult(double, double) [14]
-----
[15]     0.00   0.00         1/1  main [1]
      0.00   0.00         1  CalcMath::root(double, double) [15]
-----
```

Profiler nebol schopný zistiť čas pri 1000 vstupných hodnotách tak sme použili 18 miliónov vstupných hodnôt.

Program najviac volal funkciu *add* a *pow*.