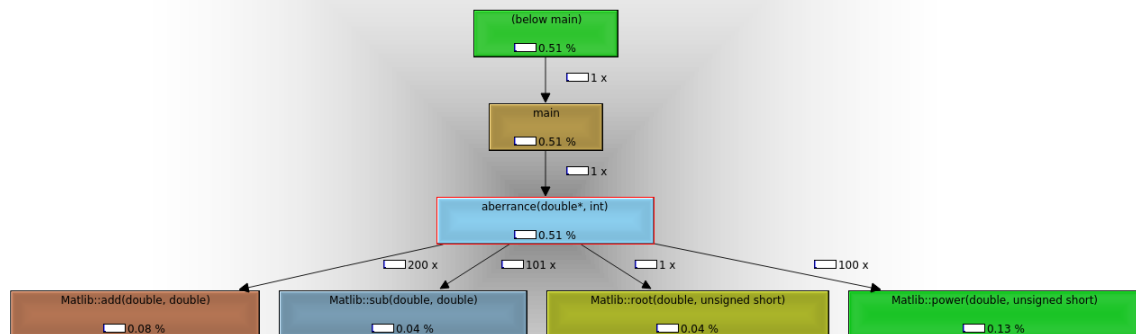


Výstup pro 100 čísel:



1 - graf volání funkcí z matlib.h

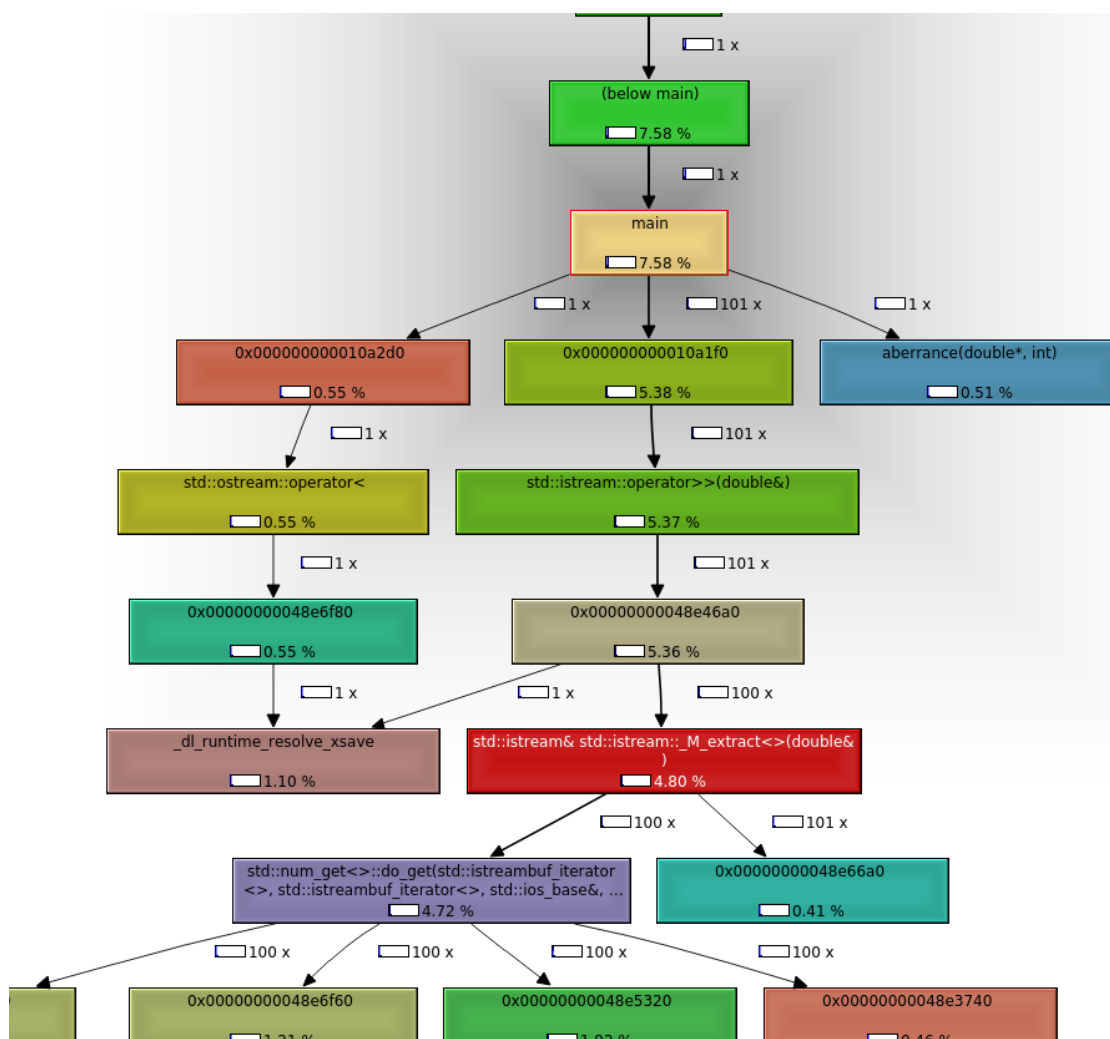
aberrance(double*, int)		
Types	Callers	All Callers
Types	Callers	Callee Map
Types	Callers	Source Code
#	Ir	Source
0		--- From '/home/jonys/CalculandumMachina-main/profiling/stddeviation.cpp' ---
26		
27		
28		double aberrance(double numbers[], int numSize)
29	0.00	{
30		Matlib math; //creating a instation of class
31	0.00	long double sum = 0;
32	0.00	double tmp = 0;
33		
34	0.02	for (int i = 0; i < numSize; i++)
35		{
36		//count average
37	0.03	tmp = numbers[i];
38	0.04	sum = math.add(sum, tmp);
0.04		100 call(s) to 'Matlib::add(double, double)' (stddeviation: matlib.cpp)
39		}
40	0.00	double average = math.div(sum,numSize);
0.00		1 call(s) to 'Matlib::div(double, double)' (stddeviation: matlib.cpp)
41		
42	0.00	long double sumOfNumbers = 0;
43		
44	0.02	for (int j = 0; j < numSize; j++)
45		{
46		// sumOfNumbers = sumOfNumbers + ((x - average)^2)
47	0.10	sumOfNumbers = math.add((math.power(math.sub(numbers[j], average), 2)), sumOfNumbers);
0.04		100 call(s) to 'Matlib::add(double, double)' (stddeviation: matlib.cpp)
0.04		100 call(s) to 'Matlib::sub(double, double)' (stddeviation: matlib.cpp)
0.13		100 call(s) to 'Matlib::power(double, unsigned short)' (stddeviation: matlib.cpp)
48		}
49		// total = 1 / (N-1) * sumOfNumbers
50	0.00	double total = 0;
51	0.00	total = math.mul(sumOfNumbers, math.div(1, math.sub(numSize, 1)));
0.00		1 call(s) to 'Matlib::mul(double, double)' (stddeviation: matlib.cpp)
0.00		1 call(s) to 'Matlib::sub(double, double)' (stddeviation: matlib.cpp)
0.00		1 call(s) to 'Matlib::div(double, double)' (stddeviation: matlib.cpp)
52	0.00	double result = math.root(total, 2);
0.04		1 call(s) to 'Matlib::root(double, unsigned short)' (stddeviation: matlib.cpp)
53	0.00	return result;
54	0.00	}
55		
56		//main function - to read numbers from file
57		

2 - detail volání funkcí z matlib.h zobrazený v kódu

Incl.	Self	Called	Function	Location
100.00	0.00	(0)	0x0000000000001100	ld-2.31.so
84.59	0.04	1	_dl_start	ld-2.31.so: rtld.c, dl-machine.h, get-dynamic-info.h...
84.55	0.02	1	_dl_sysdep_start	ld-2.31.so: dl-sysdep.c, dl-sysdep.c, cpu-features.c...
84.01	0.06	1	dl_main	ld-2.31.so: rtld.c, dl-prop.h, get-dynamic-info.h, se...
81.90	15.16	7	dl_relocate_object	ld-2.31.so: dl-reloc.c, dl-machine.h, do-rel.h, ldsod...
70.44	39.02	2 244	dl_lookup_symbol_x	ld-2.31.so: dl-lookup.c
31.42	23.41	2 244	do_lookup_x	ld-2.31.so: dl-lookup.c, ldsdefs.h
12.06	0.00	1	_start	stddeviation
12.06	0.00	1	(below main)	libc-2.31.so: libc-start.c
11.36	0.16	100	_dl_runtime_resolve_xsave2	ld-2.31.so: dl-trampoline.h
9.31	0.03	21	_dl_runtime_resolve_xsave	ld-2.31.so: dl-trampoline.h
7.86	4.86	2 229	check_match	ld-2.31.so: dl-lookup.c
7.58	0.10	1	main	stddeviation: stddeviation.cpp
5.38	0.01	101	0x0000000000010a1f0	(unknown)
5.37	0.01	101	std::istream::operator>>(d...	libstdc++-so.6.0.28
5.36	0.01	101	0x000000000048e46a0	(unknown)
4.80	0.19	100	std::istream& std::istream::...	libstdc++-so.6.0.28
4.72	0.41	100	std::num_get<>::do_get(st...	libstdc++-so.6.0.28
4.23	0.00	1	_libc_csu_init	stddeviation
4.23	0.00	1	_GLOBAL__sub_I_Z9aberra...	stddeviation: stddeviation.cpp
4.23	0.00	1	_static_initialization_and_d...	stddeviation: stddeviation.cpp, iostream
4.22	0.00	1	0x0000000000010a280	(unknown)
4.22	0.01	1	std::ios_base::Init::Init()	libstdc++-so.6.0.28
4.12	0.39	121	_dl_fixup	ld-2.31.so: dl-runtime.c, dl-machine.h, dl-rel.h
3.45	3.45	2 635	strcmp	ld-2.31.so: strcmp.S
3.35	0.00	1	_dl_init	ld-2.31.so: dl-init.c
3.35	0.02	7	call_init.part.0	ld-2.31.so: dl-init.c
3.25	0.01	102	0x000000000048e3f10	(unknown)
3.06	0.00	1	0x00000000000a6a60	libstdc++-so.6.0.28
3.03	0.00	26	0x000000000048e45c0	(unknown)
3.03	0.00	1	malloc_hook_ini	libc-2.31.so: hooks.c, arena.c, malloc.c
3.00	0.03	1	ptmalloc_init.part.0	libc-2.31.so: arena.c, malloc.c
2.98	0.01	26	0x00000000000b5d0	libstdc++-so.6.0.28
2.94	0.00	1	0x000000000048e4100	(unknown)
2.92	2.88	1	_dl_addr	libc-2.31.so: dl-addr.c
2.92	0.03	(0)	std::locale::_Impl::_Impl(un...	libstdc++-so.6.0.28
1.92	0.01	100	0x000000000048e5320	(unknown)
1.82	0.14	99	void std::_convert_to_v<>...	libstdc++-so.6.0.28
1.73	0.01	100	0x000000000048e5f30	(unknown)
1.68	0.92	100	__strtod_l_internal	libc-2.31.so: strtod_l.c
1.68	0.02	99	strtod_l	libc-2.31.so: strtod_l.c
1.21	0.01	100	0x000000000048e6f60	(unknown)
1.17	0.09	1	_dl_map_object_deps	ld-2.31.so: dl-deps.c, scratch_buffer.h
1.13	0.83	99	std::num_get<>::M_extra...	libstdc++-so.6.0.28
1.12	0.00	12	0x00000000004001090	(unknown)
1.12	0.02	12	_dl_catch_exception	ld-2.31.so: dl-error-skeleton.c
1.08	0.00	1	0x000000000048e38c0	(unknown)
1.08	0.08	12	_dl_map_object	ld-2.31.so: dl-load.c
1.06	0.00	(0)	std::ctype<>::ctype(unsig...	libstdc++-so.6.0.28

main			
Types	Callers	All Callers	Callee Map
Source Code			
#	lr	Source	
0		From '/home/onyon/CalculandumMachina-main/profiling/stddeviation.cpp' ---	
56		//main function - to read numbers from file	
57			
58		int main(int argc, char *argv[])	
59	0.00	{	
60	0.00	ifstream file;	
0.25		1 call(s) to '0x0000000000010a260'	
0.35		1 call(s) to '0x0000000000010a1a0'	
61			
62			
63	0.00	if (argc != 2)	
64		{	
65		cerr << "there must be 1 argument - file with array of numbers" << endl ;	
66		return 0;	
67		}	
68		else	
69		{	
70	0.00	char *fileNum = argv[1];	
71	0.00	file.open(fileNum, ifstream::in);	
0.29		1 call(s) to '0x0000000000010a1d0'	
72	0.00	if (!file.is_open()) //for case program can not open a file	
0.00		1 call(s) to '0x0000000000010a290'	
73		{	
74		throw new std::runtime_error("Failed to open file");	
75		return 0;	
...			
77		}	
78		double num;	
79		double array[10000]; // load numbers to this array	
80	0.00	double numSize = 0;	
81	0.06	for (int i = 0; file >> num ; i++)	
0.02		101 call(s) to '0x0000000000010a270'	
5.38		101 call(s) to '0x0000000000010a1f0'	
82		{	
83	0.02	array[i] = num;	
84	0.02	numSize++;	
85		}	
86			
87	0.00	double deviation = aberrance(array, numSize);	
0.51		1 call(s) to 'aberrance(double, int)' (stddeviation: stddeviation.cpp)	
88	0.00	cout << deviation << endl;	
0.13		1 call(s) to '0x0000000000010a240'	
0.55		1 call(s) to '0x0000000000010a2d0'	
89	0.00	return 0;	
90	0.00	}	
91			
92			
93			

3 - přehled všech běžících procesů



4- přehled všech běžících procesů znázorněný v grafu