Otto-von-Guericke-University Magdeburg
Max Planck Institute for Dynamics of Complex Technical Systems
Computational Methods for Systems and Control Theory

Dr. Jens Saak, Dipl.-Math. Martin Köhler
Website: http://www.mpi-magdeburg.mpg.de/csc/teaching/16ws/sc1/

Winterterm 2016/2017

Scientific Computing 1
Handout 3
October 17, 2016

Common GCC Options

Binary code optimization:

`-Os`	Optimize the code to reduce the size of the binary.
`-O1`	Turn on basic optimizations. The compiler tries to reduce code size and execution time, without performing any optimizations that take a great deal of compilation time.
`-O2`	Optimize even more. GCC performs nearly all optimizations that do not involve a space-speed trade-off. As compared to -O1, this option increases both compilation time and the performance.
`-O3`	Aggressive optimization. It tries to unroll loops constructs and inlines small functions. It can cause unexpected effects in the program. The output is usually larger then using `-O2`.
`-march=native`	Automatically determines the code generation options to optimally exploit your local CPU features. **Code may not be executable on other machines.**

Debugging:

`-g`	Include the debug symbols in the output. This is necessary for tools like `gdb`, `ddd` or `valgrind`.
`-pg`	Include the profiling information for the GNU profiler. Execution in `gprof` then produces the desired information.

Floating Point Arithmetics related:

`-ffast-math`	Turns off the IEEE754 floating point arithmetics. **This option is dangerous.**
`-ffloat-store`	Floating point operations store the results to the memory instead of keeping them in high accuracy CPU registers.
`-mfpmath=sse` `-msse2`	Use the SSE2 registers for floating point operations instead of the classical x86/x87 floating point unit. **Only available on x86 and x86_64 plaforms.-`mfpmath=sse` default on x86_64.**
`-mavx` `-mavx2`	as above but for the more recent AVX and AVX2 registers.

Warnings and C Standards:

`-Wall`	The compiler displays all warnings about malformed code.
`-std=XXX`	Defines the C standard to use. Normally explicit usage is not necessary. possible values: `c89`, `c99` or `c11`.

Finding libraries and header files:

`-I`*path*	Set an addtional search path for the `include` directive. This can be used multiple times.
`-L`*path*	Set an additional search path for the linker.
`-l`*NAME*	Link a specified library to the program. The `lib` prefix is automatically added to the library.

Compilation of own libraries:

`-c`	Compile the source code to object files without linking it. The default output name is `inputname.o`.
`-fPIC`	Generate *position independent code*. This flag influence the assembler code production to use relative addresses. It is necessary for libraries.

Code Preprocessing and basic shared memory parallelism:

`-D`*NAME*`=`*VALUE*	Defines a preprocessor variable *NAME* and sets it to *VALUE*
`-fopenmp`	The OpenMP support is enabled.
`-pthread`	The PThread support is enabled.