GCC

Switch	Description
`-c`	Compile the source file but do not link
`-x language`	Set the specific language instead of letting the compiler decide based on the source file suffix. Useful for FORTRAN i.e. `language` can be replaced with `f77`, `f77-cpp-input`, `f95` or `f95-cpp-input`
`-o file`	Change the name of the binary file from `a.out` to `file`
`-v`	Print version of the compiler with options used for its configuration
`-fopenmp`	Enables OpenMP directives to create a multithreaded code
`-fno-gnu-keywords`	Turns off GNU specific language extensions
`-w`	Suspend add warnings
`-Wall`	Enables all warnings
`-g`	Enables debugging information
`-p` or `-pg`	Turns on `gprof` profiling information
`-ftree-vectorizer-verbose`	Enables verbose mode for GCC vectorization
`-O`, `-O1`, `-O2` or `-O3`	Different levels of code optimization; -O3 is the most aggressive
`-Ofast`	`-Ofast` enables all `-O3` optimizations. It also enables optimizations that are not valid for all standard-compliant programs
`-Og`	Turns on debugging safe optimization mode
`-floop-block`	Perform loop blocking transformations on loops
`-floop-interchange`	Perform loop interchange transformations on loops
`-ftree-vectorize`	Perform loop vectorization on trees. This flag is enabled by default at -O3
`-funroll-loops`	Unroll loops whose number of iterations can be determined at compile time or upon entry to the loop
`-fprefetch-`	Generate instructions to prefetch memory to improve the performance of

`loop-arrays`	loops that access large arrays
`-ffast-math`	Use faster but less accurate mathematical functions
`-D`	Define a macro
`-I<dir>`	Add `<dir>` to compilers path for included files
`-l<lib>`	Pass a library `<lib>` to the linker
`-L<dir>`	Add directory `<dir>` to the linker library path
`-march=native`	This selects the CPU to generate code for at compilation time by determining the processor type of the compiling machine
`-ffree-form`	FORTRAN : Specify the layout used by the source file
`-cpp`	FORTRAN : Enable preprocessor for FORTRAN files
`-fno-underscoring`	FORTRAN : Do not transform names of entities specified in the Fortran source file by appending underscores to them
`-fexternal-blas`	FORTRAN : This option will make gfortran generate calls to BLAS functions for some matrix operations like "MATMUL"
`-floop-parallelize-all`	Identify loops that can be parallelized
`-ftree-parallelize-loops=n`	Parallelize loops, i.e., split their iteration space to run in n threads. This option implies `-pthread`

More information

- GCC autoparallelization http://gcc.gnu.org/wiki/Graphite/Parallelization

PGI compiler options

Switch	Description
`-mp`	Enables OpenMP directives to explicitly parallelize the code regions for multithreaded execution
`-O0`	Turns off all optimizations
`-O`	Basic code optimization, no SIMD instructions are used, loop optimization.
`-O2`	All `-O` optimizations with SIMD instructions, cache alignment and redundancy elimination
`-g`	Generates debug information.
`-pg`	Enables `gprof` profiling
`-acc`	Enables OpenACC directives for accelerator offload
`-ta=tesla`	Specifies the target accelerator `tesla` can be specialized as `tesla:fermi` or `tesla:pgckepler` on Palmetto for enabling 2.0 or 3.x compute capability of the NVIDIA accelerators
`-Mvect`	Enable vectorization
`-Mprefetch`	Enable prefetching
`-Minfo`	Enables verbose mode, `-Minfo=loop,accel,vect` will display information about loop, accelerator and vectorization
`-fast`	Chooses generally optimal flags for optimization on the target platform
`-c`	Skip linking, compile and assemble only
`-o file`	Use `file` as an executable name
`-w`	Do not print warning messages
`-Mconcur`	Instructs the compiler to auto-parallelize loops
`-I<dir>`	Add `<dir>` to compilers path for included files
`-D`	Define a macro
`-V` or `--version`	Prints information about the compiler
`-Mpreprocess`	Turns on the preprocessor (useful for FORTRAN code)
`-acclibs`	Link time option to add the accelerator libraries to the linker
`-l<lib>`	Pass a library `<lib>` to the linker
`-L<dir>`	Add directory `<dir>` to the linker library path

Intel compiler options

Switch	Description
`-parallel`	enable the auto-parallelizer to generate multi-threaded code for loops that can be safely executed in parallel
`-par-reportX`	(X=1,2,3) control the auto-parallelizer diagnostic level
`-openmp`	enable the compiler to generate multi-threaded code based on the OpenMP* directives
`-openmp-reportX`	(X=1,2,3) control the OpenMP parallelizer diagnostic level
`-fast`	enable `-xHOST -O3 -ipo -no-prec-div -static` options set by -fast cannot be overridden with the exception of `-xHOST` , list options separately to change behavior
`-O0`	disable optimizations
`-O1`	optimize for maximum speed, but disable some optimizations which increase code size for a small speed benefit
`-O2`	optimize for maximum speed (DEFAULT)
`-O3`	optimize for maximum speed and enable more aggressive optimizations that may not improve performance on some programs
`-threads`	specify that multi-threaded libraries should be linked against `-nothreads` disables multi-threaded libraries
`-vec`	enables (DEFAULT) vectorization (`-novec` disables vectorization)
`-mkl`	link to the Math Kernel Library (MKL) and bring in the associated headers, `-mkl=` one of `parallel` (default), `sequential` or `cluster`
`-opt-matmul`	replace matrix multiplication with calls to intrinsics and threading libraries for improved performance (DEFAULT at `-O3 -parallel`)
`-xHost`	generate instructions for the highest instruction set and processor available on the compilation host machine
`-integer-size XX`	specifies the default size of integer and logical variables size: 16, 32, 64
`-real-size XX`	specify the size of REAL and COMPLEX declarations, constants, functions, and intrinsics size: 32, 64, 128
`-w`	disable all warnings, equivalent to `-warn none` or `-nowarn`

`-warn all`	enables all warnings
`-L<dir>`	instruct linker to search `<dir>` for libraries
`-l<string>`	instruct the linker to link in the `-l<string>` library
`-Wl,<o1>[,<o2>,...]`	pass options `o1`, `o2`, etc. to the linker for processing
`-V`	display compiler version information
`-multiple-processes[=<n>]`	create multiple processes that can be used to compile large numbers of source files at the same time
`-c`	compile to object (.o) only, do not link
`-S`	compile to assembly (.s) only, do not link
`-o <file>`	name output executable file
`-g`	produce symbolic debug information in object file (implies `-O0` when another optimization option is not explicitly set)
`-p`	compile and link for function profiling with UNIX gprof tool, `-pg` is also valid
`-D<name>[=<text>]`	define macro
`-I<dir>`	add directory to include file search path
`-fpp`	run Fortran preprocessor on source files prior to compilation