Intel Software Development Software Support Renewals (SSR)

Intel listens to their customers and continually generates revisions of industry-leading tools for developers seeking new features on the latest processors. Secure your software development with an investment in the most up-to-date professional compilers and performance tools.

Renewing can save you on costs, time and more:

- Renewing can save you between 30% and 70% (depending on the time the license is renewed)
- Save valuable time on Software Development Tool problems
- Maintain access to Premier Support and receive technical support directly from Intel
- Ensure your application investment extends to tomorrow's platforms since technology is rapidly advancing
- Receive access to latest updates and free performance increase through service pack updates
- Benefit from feature enhancements and bug fixes
- $ Be eligible for product upgrades at discounted prices
- OS Get support for new OS releases

Support renewal options:

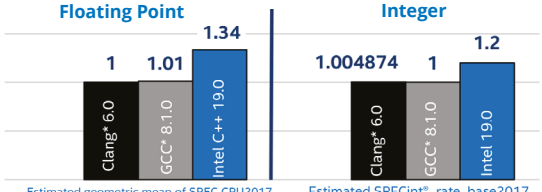
- ✓ ***SSR Pre-Expiry*** **- for purchase before the support expiration date. Support for the existing serial number will extend by 12 months, starting from the previous support expiration date.**
- ✓ ***SSR Post-Expiry*** **- for purchase when support has expired by longer than 12 months. The new 12-month support period for the existing serial number will start from the date of purchase.**
- ✓ **If support has expired by more than 12 months, the user no longer has the option to renew support but must purchase a new license.**
- ✓ **A valid serial number is required when purchasing all Intel Service & Support Renewal (SSR) SKUs.**

Additional Product-Specific Benefits

The following are graphs that depict the performance gains provided by using Intel compilers:

Industry-leading Application Performance on Linux* using Intel® C++ & Fortran Compilers (higher is better)

Boost C++ Application Performance on Linux* using Intel® C++ Compiler

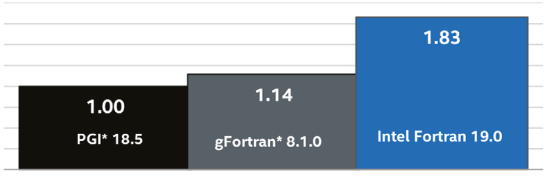


Estimated geometric mean of SPEC CPU2017 Floating Point rate base C/C++ benchmarks

Estimated SPECint®_rate_base2017

Relative geomean performance, SPEC* benchmark - higher is better

Boost Fortran Application Performance on Linux* using Intel® Fortran Compiler



Estimated relative geomean performance, Polyhedron* benchmark – higher is better

Performance results are based on testing as of Aug. 26, 2018 and may not reflect all publicly available security updates. See configuration disclosure for details. No product can be absolutely secure. Software and workloads used in performance tests may have been optimized for performance only on Intel microprocessors. Performance tests, such as SYSmark and MobileMark, are measured using specific computer systems, components, software, operations and functions. Any change to any of those factors may cause the results to vary. You should consult other information and performance tests to assist you in fully evaluating your contemplated purchases, including the performance of that product when combined with other products. For more complete information, see Performance Benchmark Test Disclosure

Testing by Intel as of Aug. 26, 2018. Configuration: Linux hardware: Intel(R) Xeon(R) Platinum 8180 CPU @ 2.50GHz, 384 GB RAM, HyperThreading is on. Software: Intel compilers 19.0, GCC 8.1.0, PGI 18.5, Clang/LLVM 6.0. Linux OS: Red Hat Enterprise Linux Server release 7.4 (Maipo), 3.10.0-693.el7.x86_64. SPEC* Benchmark (www.spec.org). SmartHeap 10 was used for CXX tests when measuring SPECint® benchmarks.SPECint®_rate_base_2017 compiler switches: SmartHeap 10 were used for C++ tests. Intel C/C++ compiler 19.0: -xCORE-AVX512 -ipo -O3 -no-prec-div -qopt-mem-layout-trans=3. GCC 8.1.0 -march=znver1 -mfpmath=sse -Ofast -funroll-loops -flto. Clang 6.0: -march=core-avx2 -mfpmath=sse -Ofast -funroll-loops -flto.SPECfp®_rate_base_2017 compiler switches: Intel C/C++ compiler 19.0: -xCORE-AVX512 -ipo -O3 -no-prec-div -qopt-prefetch -ffinite-math-only -qopt-mem-layout-trans=3. GCC 8.1.0: -march=skylake-avx512 -mfpmath=sse -Ofast -fno-associative-math -funroll-loops -flto. Clang 6.0: -march=znver1 -mfpmath=sse -Ofast -funroll-loops -flto.SPECint®_speed_base_2017 compiler switches: SmartHeap 10 were used for C++ tests. Intel C/C++ compiler 19.0: -xCORE-AVX512 -ipo -O3 -no-prec-div -qopt-mem-layout-trans=3 -qopenmp. GCC 8.1.0 -march=znver1 -mfpmath=sse -Ofast -funroll-loops -flto -fopenmp. Clang 6.0: -march=core-avx2 -mfpmath=sse -Ofast -funroll-loops -flto -fopenmp=libomp. SPECfp®_speed_base_2017 compiler switches: Intel C/C++ compiler 19.0: -xCORE-AVX512 -ipo -O3 -no-prec-div -qopt-prefetch -ffinite-math-only -qopenmp. GCC 8.1.0: -march=skylake-avx512 -mfpmath=sse -Ofast -fno-associative-math -funroll-loops -flto -fopenmp. Clang 6.0: -march=skylake-avx512 -mfpmath=sse -Ofast -funroll-loops -flto -fopenmp=libomp.

Testing by Intel as of Aug. 26, 2018. Configuration: Hardware: Intel® Core™ i7-8700K CPU @ 3.70GHz, 64 GB RAM, HyperThreading is on. Software: Intel Fortran compiler 19.0. PGI Fortran* 18.5. gFortran* 8.1.0. Linux OS: Red Hat Enterprise Linux Server release 7.4 (Maipo), 3.10.0-693.el7.x86_64 Polyhedron Fortran Benchmark (www.fortran.uk). Linux compiler switches: Gfortran: -Ofast -mfpmath=sse -flto -march=haswell -funroll-loops -ftree-parallelize-loops=6. Intel Fortran compiler: -fast -parallel -xCORE-AVX2 -nostandard-realloc-lhs. PGI Fortran: -fast -Mipa=fast,inline -Msmartalloc -Mfprelaxed -Mstack_arrays -Mconcur=bind -tp haswell

OPTIMIZATION NOTICE: Copyright © 2019, Intel Corporation. All rights reserved. *Other names and brands may be claimed as the property of others.

Intel's compilers may or may not optimize to the same degree for non-Intel microprocessors for optimizations that are not unique to Intel microprocessors. These optimizations include SSE2, SSE3, and SSSE3 instruction sets and other optimizations. Intel does not guarantee the availability, functionality, or effectiveness of any optimization on microprocessors not manufactured by Intel. Microprocessor-dependent optimizations in this product are intended for use with Intel microprocessors. Certain optimizations not specific to Intel microarchitecture are reserved for Intel microprocessors. Please refer to the applicable product User and Reference Guides for more information regarding the specific instruction sets covered by this notice. Notice revision #20110804. For more complete information about compiler optimizations, see our Optimization Notice

Your contact person
Holger Brunken
email holger.brunken@compuwave.de
phone +49-89-2314142-31

COMPUWAVE BUSINESS SOFTWARE DISTRIBUTION
OFFICIAL PARTNER