



[www.phoronix-test-suite.com](http://www.phoronix-test-suite.com)

**2022-06-17-0443**

microsoft testing on Ubuntu 18.04 via the Phoronix Test Suite.

#### **Automated Executive Summary**

*D4dsv5 had the most wins, coming in first place for 88% of the tests.*

*Based on the geometric mean of all complete results, the fastest (D4dsv5) was 1.85x the speed of the slowest (D4pdsv5).*

*The results with the greatest spread from best to worst included:*

*NAS Parallel Benchmarks (Test / Class: LU.C) at 7.019x*

*NAS Parallel Benchmarks (Test / Class: SP.C) at 5.638x*

*NAS Parallel Benchmarks (Test / Class: MG.C) at 5.448x*

*NAS Parallel Benchmarks (Test / Class: SP.B) at 5.104x*

*NAS Parallel Benchmarks (Test / Class: CG.C) at 4.231x*

*NAS Parallel Benchmarks (Test / Class: FT.C) at 3.135x*

*NAS Parallel Benchmarks (Test / Class: BT.C) at 2.433x*

*Botan (Test: ChaCha20Poly1305 - Decrypt) at 2.306x*

*Botan (Test: ChaCha20Poly1305) at 2.303x*

*Sysbench (Test: CPU) at 2.281x.*

## Test Systems:

### D4pdsv5

Processor: ARMv8 Neoverse-N1 (4 Cores), Motherboard: Microsoft Virtual Machine (Hyper-V UEFI v4.1 BIOS), Memory: 16GB, Disk: 275GB Virtual Disk + 161GB Virtual Disk, Graphics: hyperv\_fb, Network: Mellanox MT27800

OS: Ubuntu 18.04, Kernel: 5.4.0-1083-azure (aarch64), Compiler: GCC 7.5.0 + Clang 6.0.0-1ubuntu2, File-System: ext4, Screen Resolution: 1152x864, System Layer: microsoft

Kernel Notes: Transparent Huge Pages: always  
 Compiler Notes: --build=aarch64-linux-gnu --disable-libquadmath --disable-libquadmath-support --disable-werror --enable-bootstrap --enable-checking=release --enable-clocale-gnu --enable-default-pie --enable-fix-cortex-a53-843419 --enable-gnu-unique-object --enable-languages=c,ada,c++,go,d,fortran,objc,obj-c++ --enable-libstdcxx-debug --enable-libstdcxx-time=yes --enable-multiarch --enable-nls --enable-plugin --enable-shared --enable-threads=posix --host=aarch64-linux-gnu --program-prefix=aarch64-linux-gnu- --target=aarch64-linux-gnu --with-default-libstdcxx-abi=new --with-gcc-major-version-only -v  
 Security Notes: itlb\_multihit: Not affected + l1tf: Not affected + mds: Not affected + meltdown: Mitigation of PTI + spec\_store\_bypass: Vulnerable + spectre\_v1: Mitigation of \_\_user pointer sanitization + spectre\_v2: Vulnerable + srbds: Not affected + tsx\_async\_abort: Not affected

### D4dsv5

Processor: Intel Xeon Platinum 8370C @ 2.80GHz (2 Cores / 4 Threads), Motherboard: Microsoft Virtual Machine (Hyper-V UEFI v4.1 BIOS), Memory: 16GB, Disk: 275GB Virtual Disk + 161GB Virtual Disk, Graphics: hyperv\_fb, Network: Mellanox MT27800

OS: Ubuntu 18.04, Kernel: 5.4.0-1085-azure (x86\_64), Compiler: GCC 7.5.0 + Clang 6.0.0-1ubuntu2, File-System: ext4, Screen Resolution: 1152x864, System Layer: microsoft

Kernel Notes: Transparent Huge Pages: always  
 Compiler Notes: --build=x86\_64-linux-gnu --disable-vtable-verify --disable-werror --enable-bootstrap --enable-checking=release --enable-clocale-gnu --enable-default-pie --enable-gnu-unique-object --enable-languages=c,ada,c++,go,brig,d,fortran,objc,obj-c++ --enable-libmpx --enable-libstdcxx-debug --enable-libstdcxx-time=yes --enable-multiarch --enable-multilib --enable-nls --enable-objc-gc=auto --enable-offload-targets=nvptx-none --enable-plugin --enable-shared --enable-threads=posix --host=x86\_64-linux-gnu --program-prefix=x86\_64-linux-gnu- --target=x86\_64-linux-gnu --with-abi=m64 --with-arch-32=i686 --with-default-libstdcxx-abi=new --with-gcc-major-version-only --with-multilib-list=m32,m64,mx32 --with-target-system-zlib --with-tune=generic --without-cuda-driver -v  
 Processor Notes: Scaling Governor: intel\_pstate performance - CPU Microcode: 0xffffffff  
 Security Notes: itlb\_multihit: Not affected + l1tf: Not affected + mds: Not affected + meltdown: Not affected + mmio\_stale\_data: Vulnerable: Clear buffers attempted no microcode; SMT Host state unknown + spec\_store\_bypass: Vulnerable + spectre\_v1: Mitigation of usercopy/swaps barriers and \_\_user pointer sanitization + spectre\_v2: Mitigation of Retpolines STIBP: disabled RSB filling + srbds: Not affected + tsx\_async\_abort: Not affected

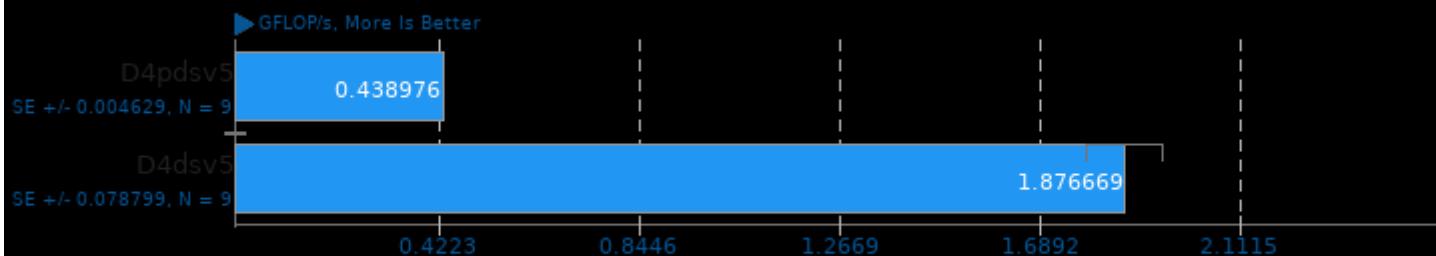
	D4pdsv5	D4dsv5
<b>ACES DGEMM - S.F.P.R (GFLOP/s)</b>	<b>0.438976</b>	<b>1.876669</b>
Normalized	23.39%	100%
Standard Deviation	3.2%	12.6%
<b>Botan - KASUMI (MiB/s)</b>	<b>67.494</b>	<b>78.491</b>
Normalized	85.99%	100%
Standard Deviation	0%	0.1%
<b>Botan - KASUMI - Decrypt (MiB/s)</b>	<b>64.858</b>	<b>75.312</b>
Normalized	86.12%	100%
Standard Deviation	0.1%	0.1%
<b>Botan - AES-256 (MiB/s)</b>	<b>2764</b>	<b>5646</b>
Normalized	48.95%	100%
Standard Deviation	0.1%	0.2%
<b>Botan - AES-256 - Decrypt (MiB/s)</b>	<b>2767</b>	<b>5643</b>
Normalized	49.04%	100%

Standard Deviation	0.1%	0.3%
<b>Botan - Twofish (MiB/s)</b>	<b>230.327</b>	<b>300.167</b>
Normalized	76.73%	100%
Standard Deviation	0.1%	0%
<b>Botan - Twofish - Decrypt (MiB/s)</b>	<b>245.634</b>	<b>296.392</b>
Normalized	82.87%	100%
Standard Deviation	0.1%	0%
<b>Botan - Blowfish (MiB/s)</b>	<b>289.749</b>	<b>373.340</b>
Normalized	77.61%	100%
Standard Deviation	0.1%	0.1%
<b>Botan - Blowfish - Decrypt (MiB/s)</b>	<b>291.005</b>	<b>372.691</b>
Normalized	78.08%	100%
Standard Deviation	0.6%	0%
<b>Botan - CAST-256 (MiB/s)</b>	<b>112.842</b>	<b>117.917</b>
Normalized	95.7%	100%
Standard Deviation	0.1%	0%
<b>Botan - CAST-256 - Decrypt (MiB/s)</b>	<b>112.450</b>	<b>117.916</b>
Normalized	95.36%	100%
Standard Deviation	0.1%	0.1%
<b>Botan - ChaCha20Poly1305 (MiB/s)</b>	<b>291.244</b>	<b>670.712</b>
Normalized	43.42%	100%
Standard Deviation	0%	0.1%
<b>Botan - ChaCha20Poly1305 - Decrypt (MiB/s)</b>	<b>288.990</b>	<b>666.297</b>
Normalized	43.37%	100%
Standard Deviation	0.1%	0.1%
<b>BYTE Unix Benchmark - Dhrystone 2 (LPS)</b>	<b>27428886</b>	<b>38333509</b>
Normalized	71.55%	100%
Standard Deviation	0.1%	0.5%
<b>Coremark - CoreMark Size 666 - I.P.S (Iterations/Sec)</b>	<b>93441</b>	<b>69492</b>
Normalized	100%	74.37%
Standard Deviation	0%	0.3%
<b>Fhourstones - C.C.4.S (Kpos / sec)</b>	<b>7198</b>	<b>12464</b>
Normalized	57.75%	100%
Standard Deviation	0.8%	0.1%
<b>NAS Parallel Benchmarks - BT.C (Mop/s)</b>	<b>1734</b>	<b>4219</b>
Normalized	41.11%	100%
Standard Deviation	0.5%	0.3%
<b>NAS Parallel Benchmarks - CG.C (Mop/s)</b>	<b>750.97</b>	<b>3177</b>
Normalized	23.64%	100%
Standard Deviation	2.3%	1.3%
<b>NAS Parallel Benchmarks - EP.C (Mop/s)</b>	<b>124.46</b>	<b>106.33</b>
Normalized	100%	85.43%
Standard Deviation	0.3%	0.3%
<b>NAS Parallel Benchmarks - EP.D (Mop/s)</b>	<b>124.14</b>	<b>106.47</b>
Normalized	100%	85.77%
Standard Deviation	0.8%	0.3%
<b>NAS Parallel Benchmarks - FT.C (Mop/s)</b>	<b>1784</b>	<b>5593</b>
Normalized	31.9%	100%
Standard Deviation	0.2%	0.3%
<b>NAS Parallel Benchmarks - LU.C (Mop/s)</b>	<b>1593</b>	<b>11180</b>
Normalized	14.25%	100%
Standard Deviation	0.3%	0.2%
<b>NAS Parallel Benchmarks - MG.C (Mop/s)</b>	<b>1951</b>	<b>10631</b>
Normalized	18.36%	100%
Standard Deviation	0.2%	0.3%

<b>NAS Parallel Benchmarks - SP.B (Mop/s)</b>	<b>569.03</b>	<b>2904</b>
Normalized	19.59%	100%
Standard Deviation	0.4%	0.1%
<b>NAS Parallel Benchmarks - SP.C (Mop/s)</b>	<b>547.75</b>	<b>3088</b>
Normalized	17.74%	100%
Standard Deviation	0.7%	1.7%
<b>OSBench - Create Files (us/Event)</b>	<b>27.724865</b>	<b>15.218015</b>
Normalized	54.89%	100%
Standard Deviation	1.1%	0.4%
<b>OSBench - Create Threads (us/Event)</b>	<b>43.412050</b>	<b>15.340646</b>
Normalized	35.34%	100%
Standard Deviation	33.7%	0.9%
<b>OSBench - Launch Programs (us/Event)</b>	<b>185.763836</b>	<b>85.636775</b>
Normalized	46.1%	100%
Standard Deviation	0.9%	1.1%
<b>OSBench - Create Processes (us/Event)</b>	<b>59.537888</b>	<b>31.185945</b>
Normalized	52.38%	100%
Standard Deviation	0.7%	2%
<b>OSBench - Memory Allocations (Ns/Event)</b>	<b>164.429347</b>	<b>75.611274</b>
Normalized	45.98%	100%
Standard Deviation	0.6%	0.6%
<b>Stockfish - Total Time (Nodes/s)</b>	<b>4622524</b>	<b>5172713</b>
Normalized	89.36%	100%
Standard Deviation	1.3%	1.4%
<b>Swet - Average (Operations/sec)</b>	<b>363861708</b>	<b>631771984</b>
Normalized	57.59%	100%
Standard Deviation	0.2%	1.6%
<b>Sysbench - RAM / Memory (MiB/sec)</b>	<b>5647</b>	<b>10165</b>
Normalized	55.56%	100%
Standard Deviation	2.3%	2.3%
<b>Sysbench - CPU (Events/sec)</b>	<b>14232</b>	<b>6239</b>
Normalized	100%	43.84%
Standard Deviation	0%	0%

## ACES DGEMM 1.0

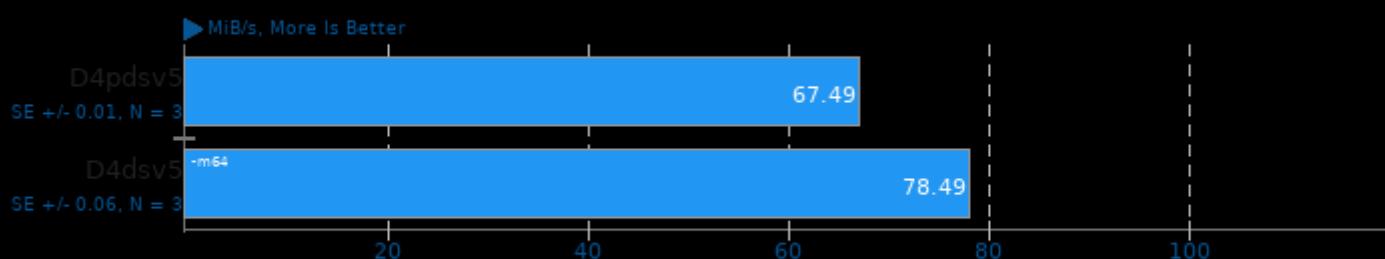
Sustained Floating-Point Rate



1. (CC) gcc options: -O3 -march=native -fopenmp

## Botan 2.17.3

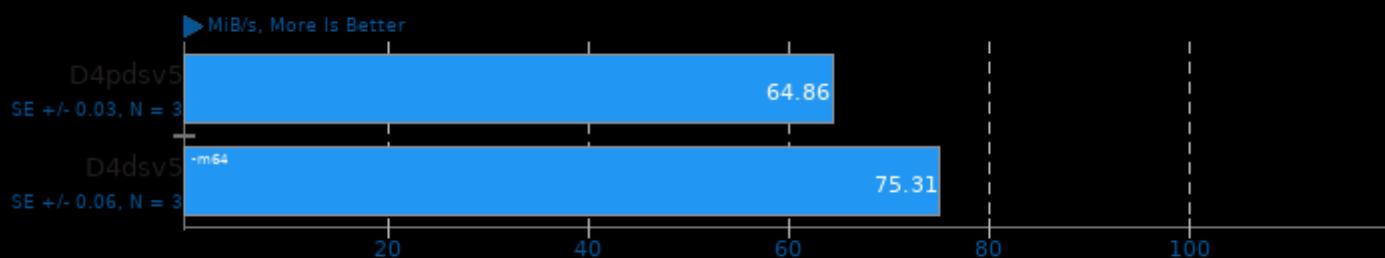
Test: KASUMI



1. (CXX) g++ options: -fstack-protector -pthread -lbotan-2 -ldl -lrt

## Botan 2.17.3

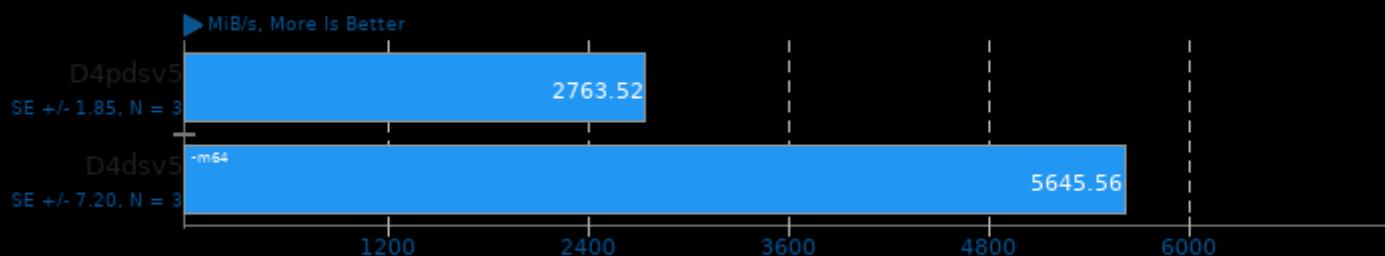
Test: KASUMI - Decrypt



1. (CXX) g++ options: -fstack-protector -pthread -lbotan-2 -ldl -lrt

## Botan 2.17.3

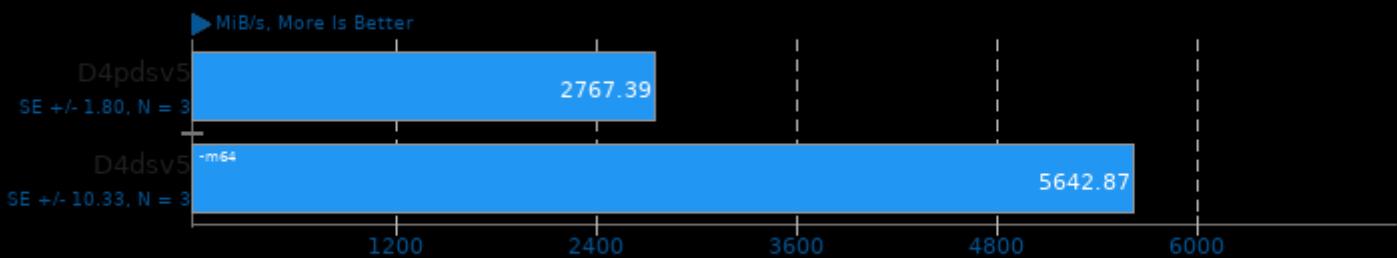
Test: AES-256



1. (CXX) g++ options: -fstack-protector -pthread -lbotan-2 -ldl -lrt

## Botan 2.17.3

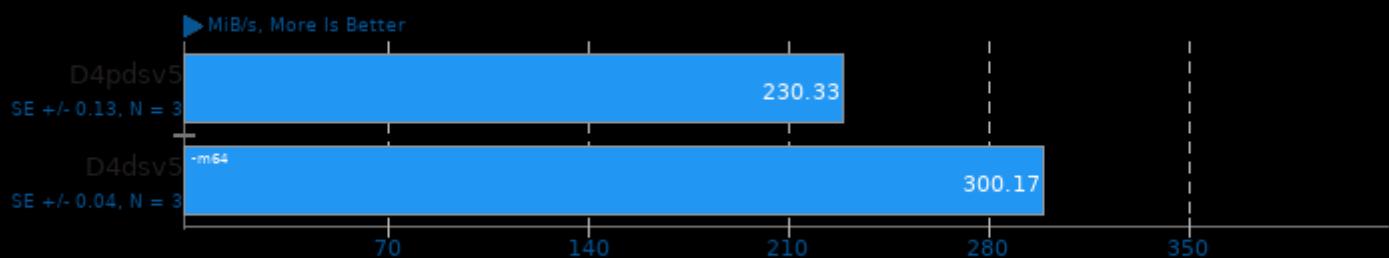
Test: AES-256 - Decrypt



1. (CXX) g++ options: -fstack-protector -pthread -lbotan-2 -ldl -lrt

## Botan 2.17.3

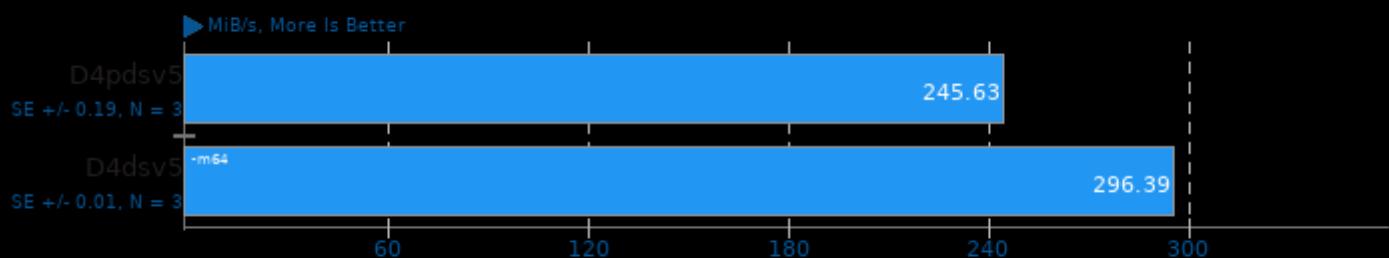
Test: Twofish



1. (CXX) g++ options: -fstack-protector -pthread -lbotan-2 -ldl -lrt

## Botan 2.17.3

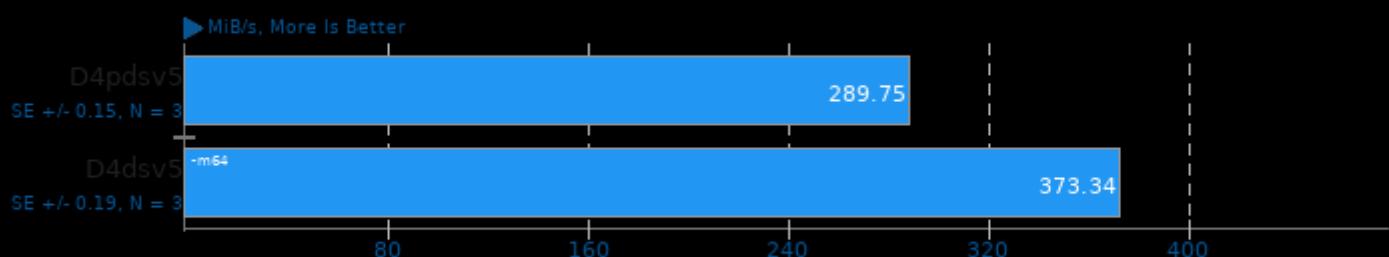
Test: Twofish - Decrypt



1. (CXX) g++ options: -fstack-protector -pthread -lbotan-2 -ldl -lrt

## Botan 2.17.3

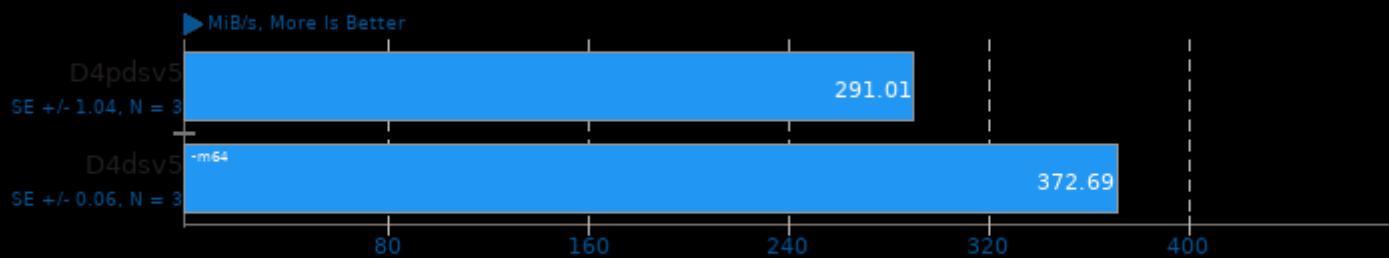
Test: Blowfish



1. (CXX) g++ options: -fstack-protector -pthread -lbotan-2 -ldl -lrt

## Botan 2.17.3

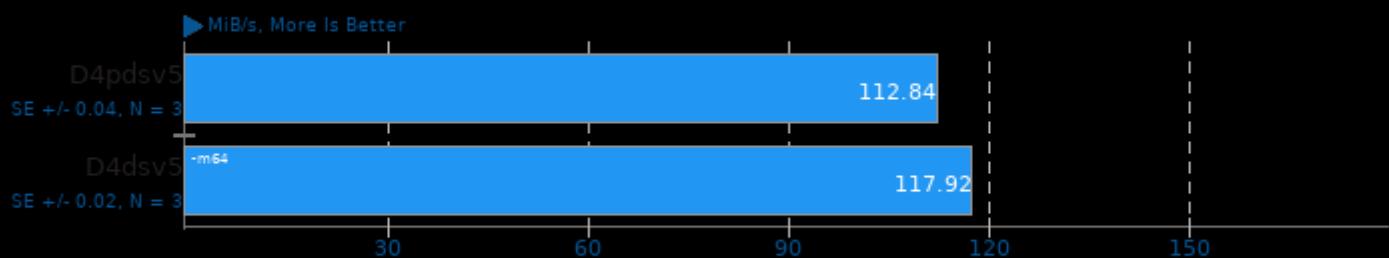
Test: Blowfish - Decrypt



1. (CXX) g++ options: -fstack-protector -pthread -lbotan-2 -ldl -lrt

## Botan 2.17.3

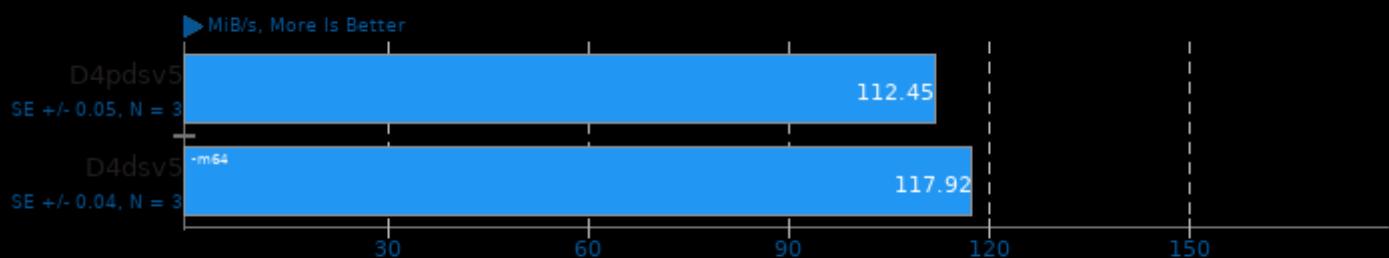
Test: CAST-256



1. (CXX) g++ options: -fstack-protector -pthread -lbotan-2 -ldl -lrt

## Botan 2.17.3

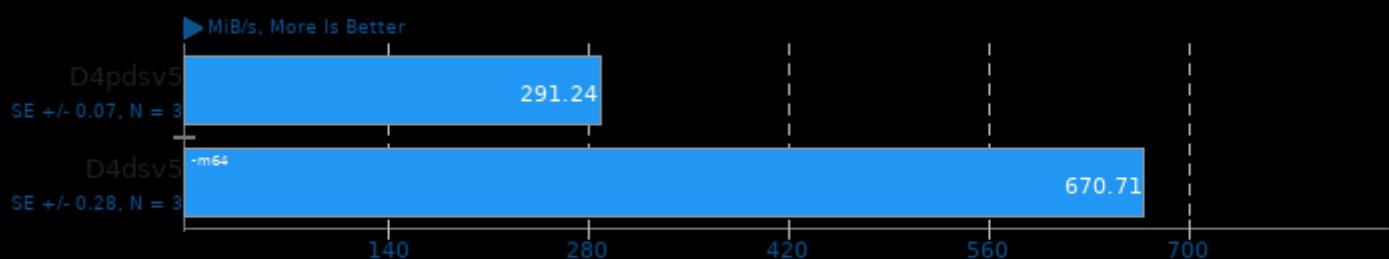
Test: CAST-256 - Decrypt



1. (CXX) g++ options: -fstack-protector -pthread -lbotan-2 -ldl -lrt

## Botan 2.17.3

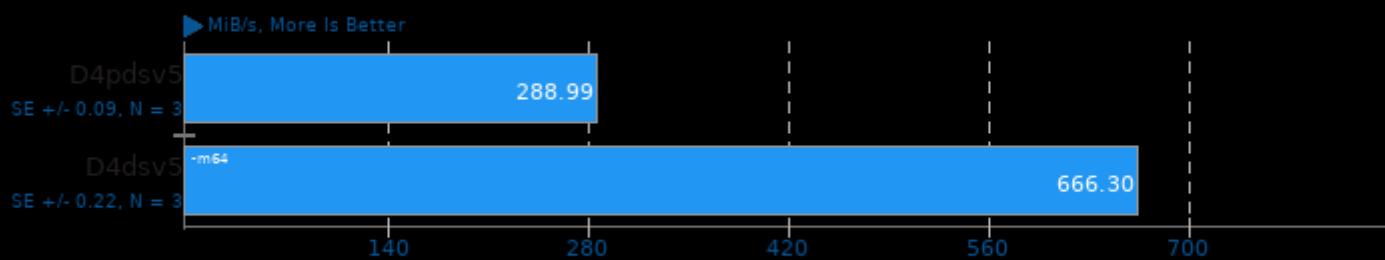
Test: ChaCha20Poly1305



1. (CXX) g++ options: -fstack-protector -pthread -lbotan-2 -ldl -lrt

## Botan 2.17.3

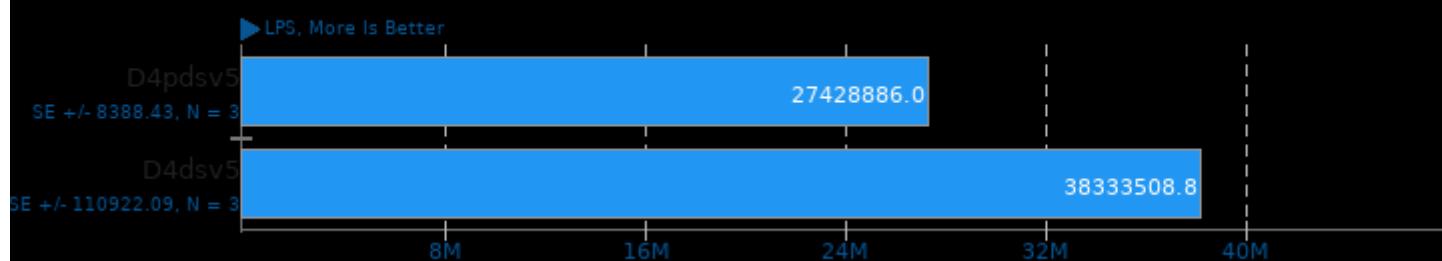
Test: ChaCha20Poly1305 - Decrypt



1. (CXX) g++ options: -fstack-protector -pthread -lbotan-2 -ldl -lrt

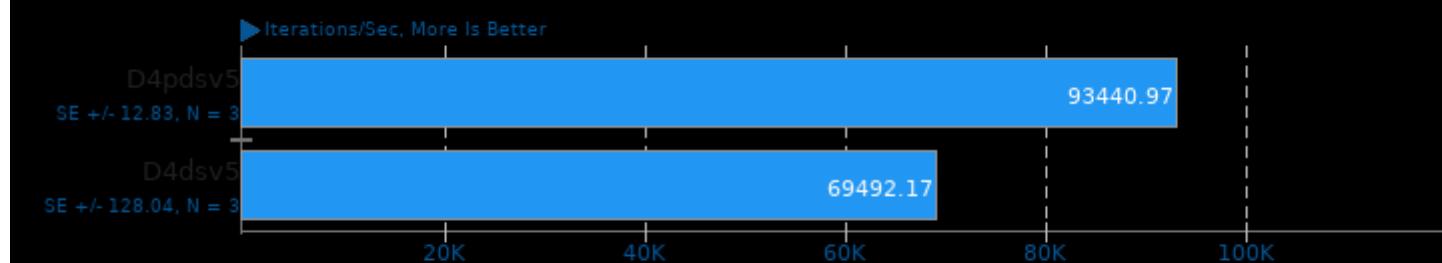
## BYTE Unix Benchmark 3.6

Computational Test: Dhrystone 2



## Coremark 1.0

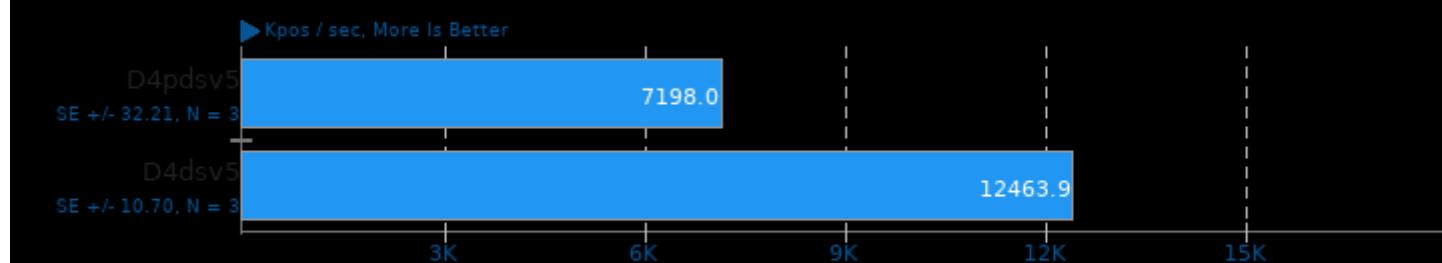
CoreMark Size 666 - Iterations Per Second



1. (CC) gcc options: -O2 -fintc -firt

## Fhourstones 3.1

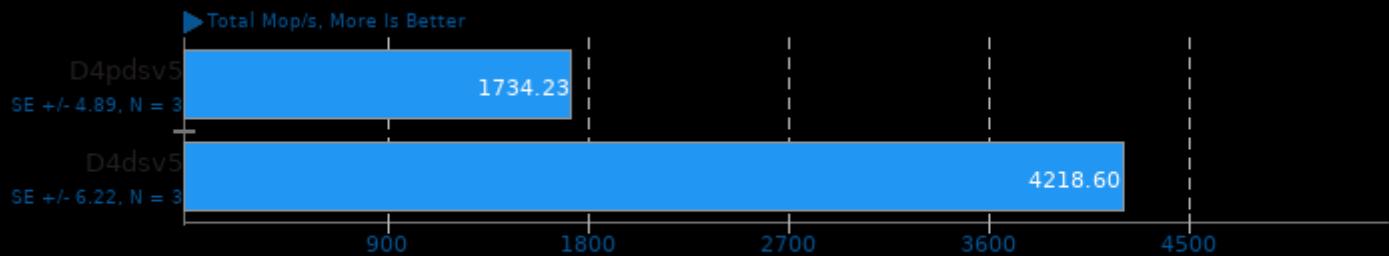
Complex Connect-4 Solving



1. (CC) gcc options: -O3

## NAS Parallel Benchmarks 3.4

Test / Class: BT.C

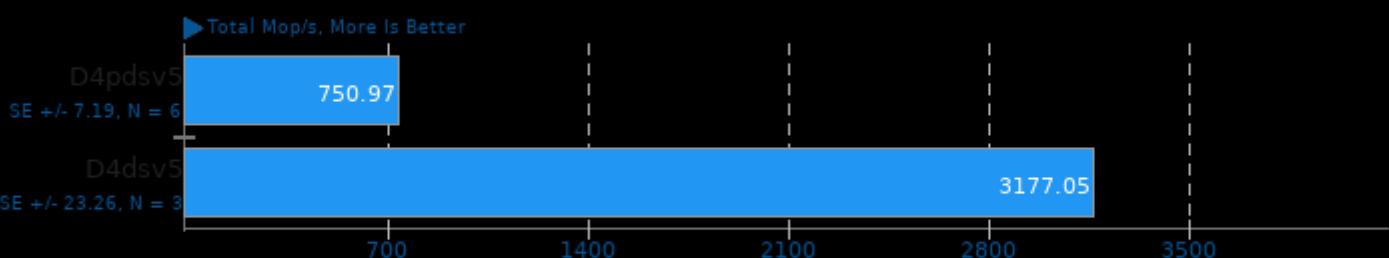


1. (F9X) gfortran options: -O3 -march=native -pthread -lmpi\_usempif08 -lmpi\_mpifh -lmpi

2. Open MPI 2.1.1

## NAS Parallel Benchmarks 3.4

Test / Class: CG.C

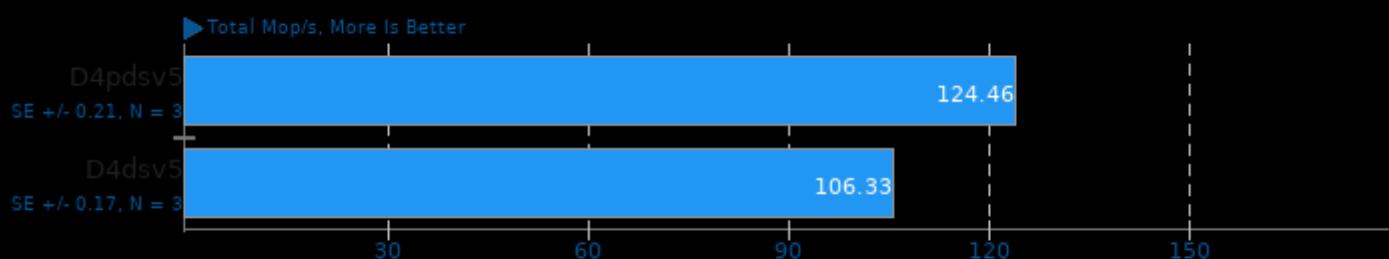


1. (F9X) gfortran options: -O3 -march=native -pthread -lmpi\_usempif08 -lmpi\_mpifh -lmpi

2. Open MPI 2.1.1

## NAS Parallel Benchmarks 3.4

Test / Class: EP.C

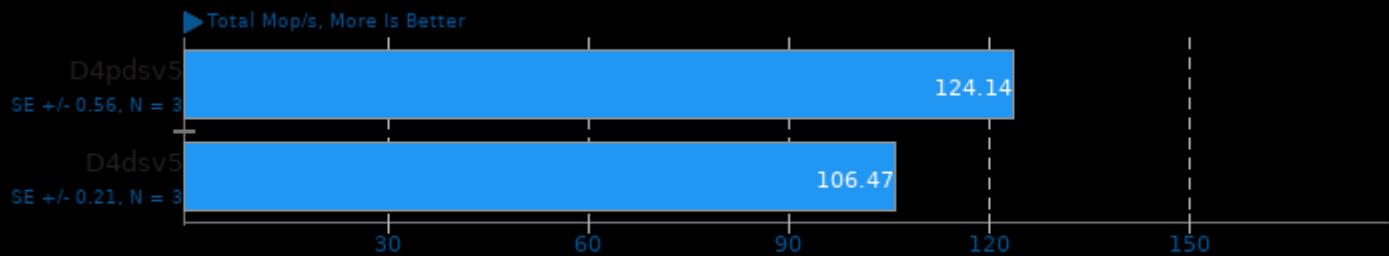


1. (F9X) gfortran options: -O3 -march=native -pthread -lmpi\_usempif08 -lmpi\_mpifh -lmpi

2. Open MPI 2.1.1

## NAS Parallel Benchmarks 3.4

Test / Class: EP.D

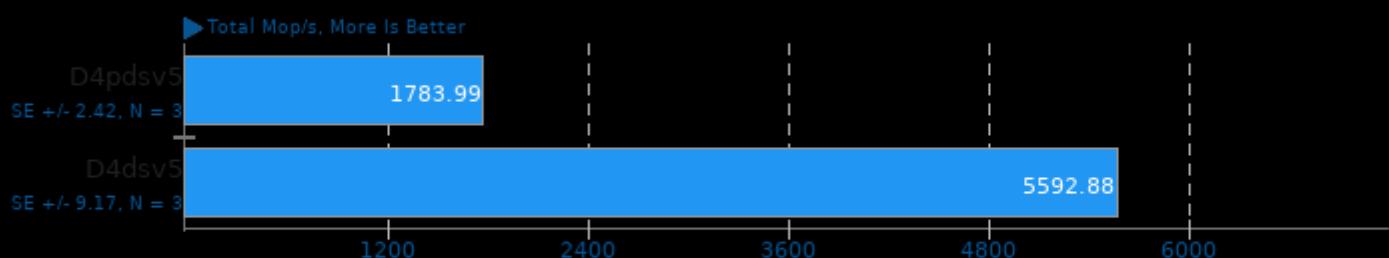


1. (F9X) gfortran options: -O3 -march=native -pthread -lmpi\_usempif08 -lmpi\_mpifh -lmpi

2. Open MPI 2.1.1

## NAS Parallel Benchmarks 3.4

Test / Class: FT.C

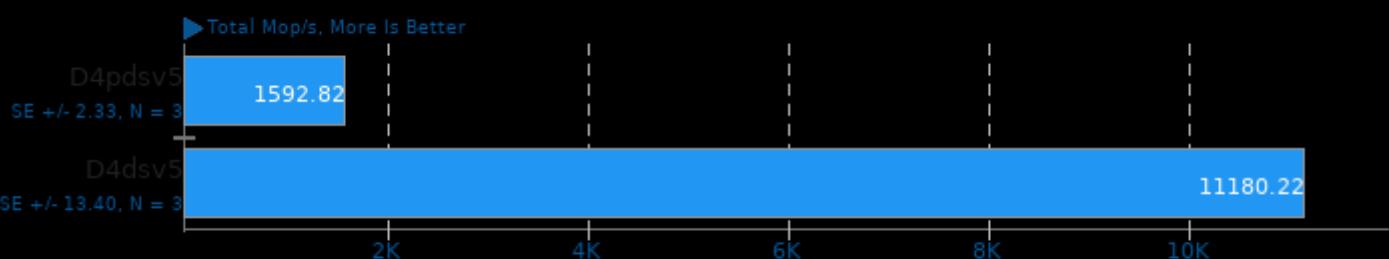


1. (F9X) gfortran options: -O3 -march=native -pthread -lmpi\_usempif08 -lmpi\_mpifh -lmpi

2. Open MPI 2.1.1

## NAS Parallel Benchmarks 3.4

Test / Class: LU.C

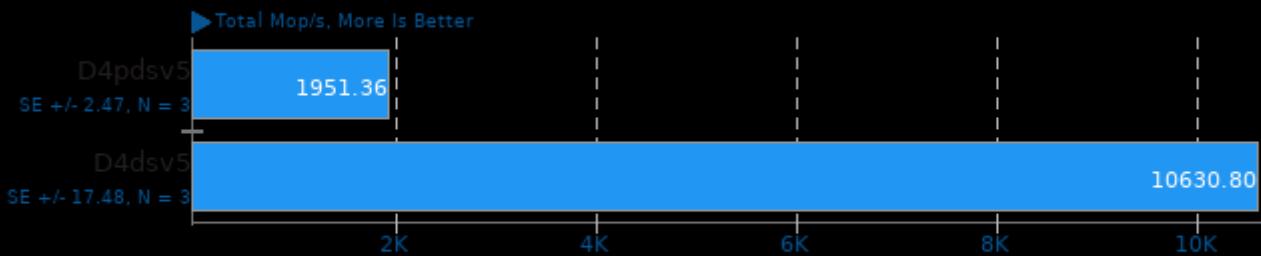


1. (F9X) gfortran options: -O3 -march=native -pthread -lmpi\_usempif08 -lmpi\_mpifh -lmpi

2. Open MPI 2.1.1

## NAS Parallel Benchmarks 3.4

Test / Class: MG.C

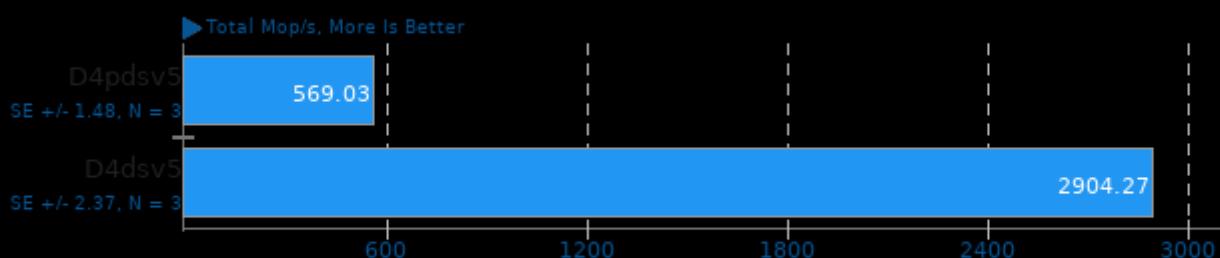


1. (F9X) gfortran options: -O3 -march=native -pthread -lmpi\_usempif08 -lmpi\_mpifh -lmpi

2. Open MPI 2.1.1

## NAS Parallel Benchmarks 3.4

Test / Class: SP.B

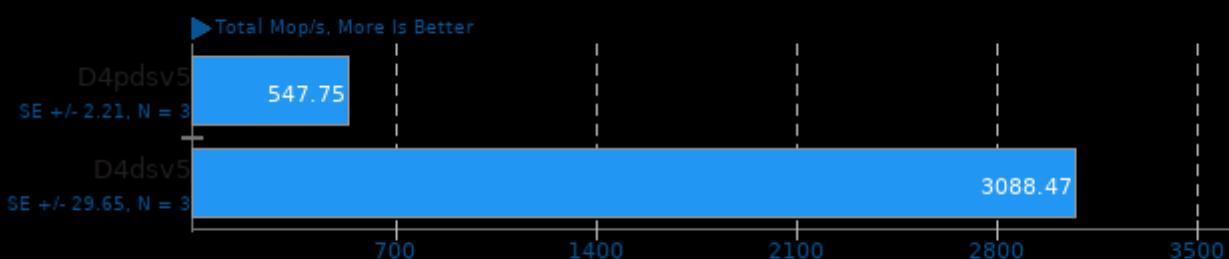


1. (F9X) gfortran options: -O3 -march=native -pthread -lmpi\_usempif08 -lmpi\_mpifh -lmpi

2. Open MPI 2.1.1

## NAS Parallel Benchmarks 3.4

Test / Class: SP.C

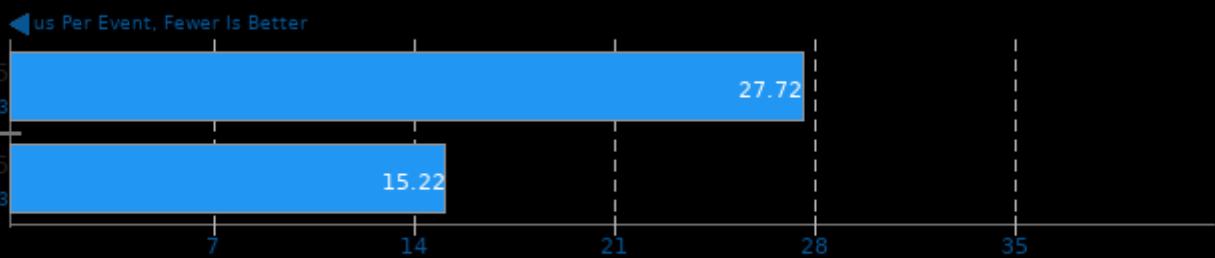


1. (F9X) gfortran options: -O3 -march=native -pthread -lmpi\_usempif08 -lmpi\_mpifh -lmpi

2. Open MPI 2.1.1

## OSBench

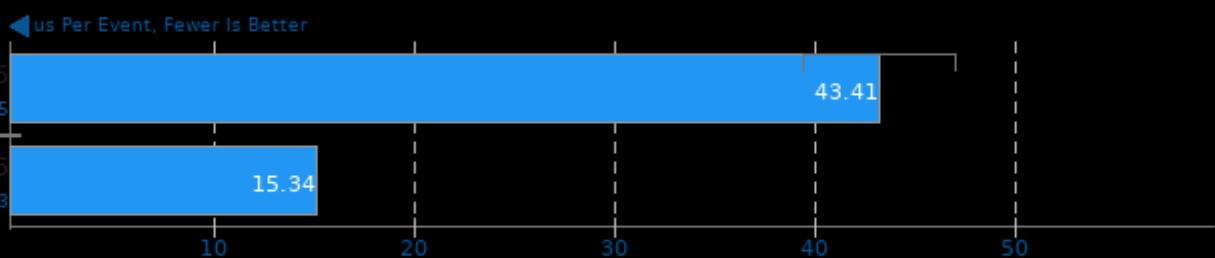
Test: Create Files



1. (CC) gcc options: -lm

## OSBench

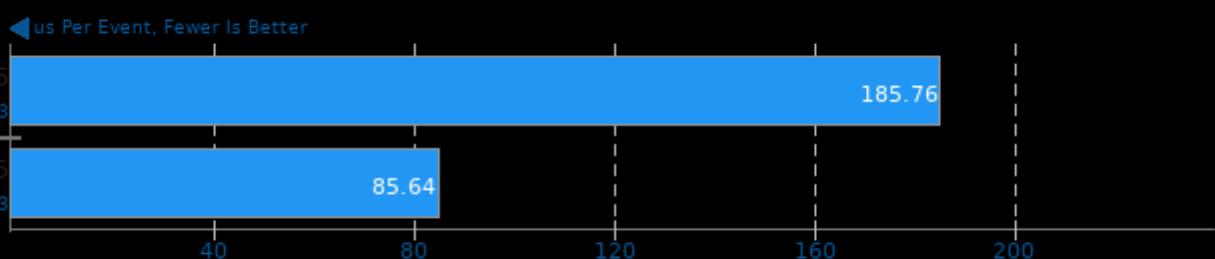
Test: Create Threads



1. (CC) gcc options: -lm

## OSBench

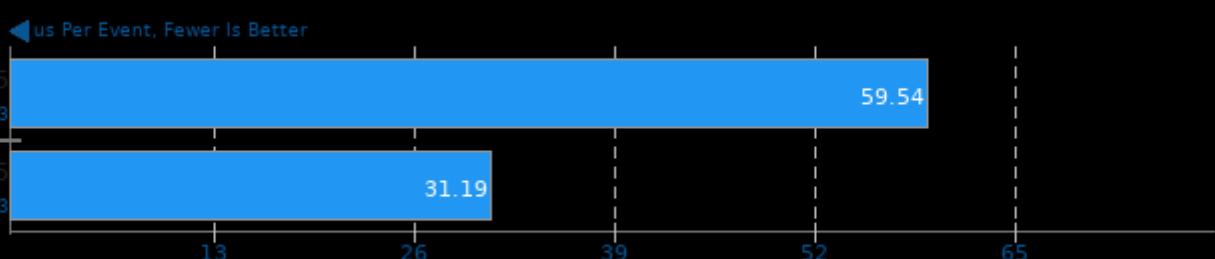
Test: Launch Programs



1. (CC) gcc options: -lm

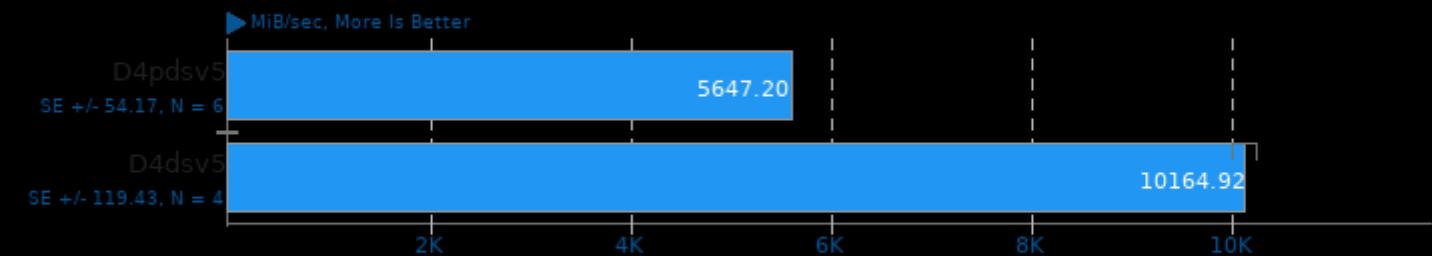
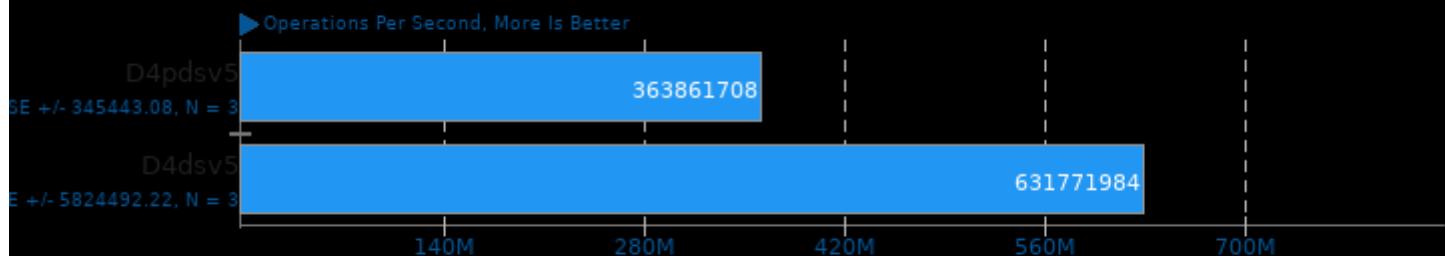
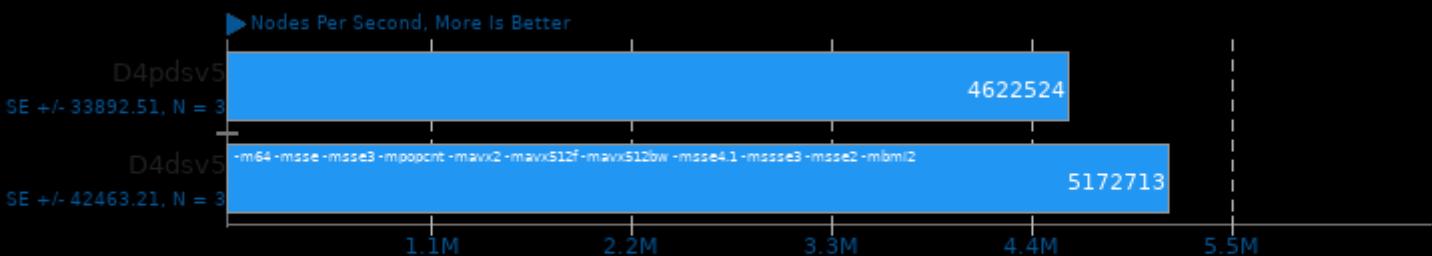
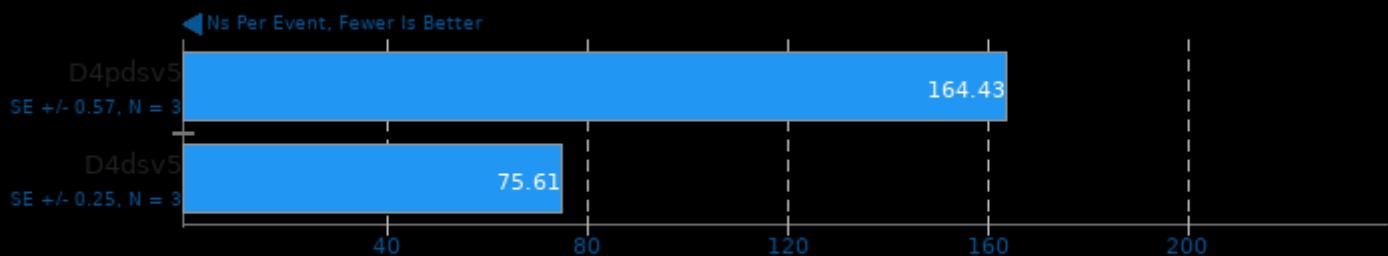
## OSBench

Test: Create Processes



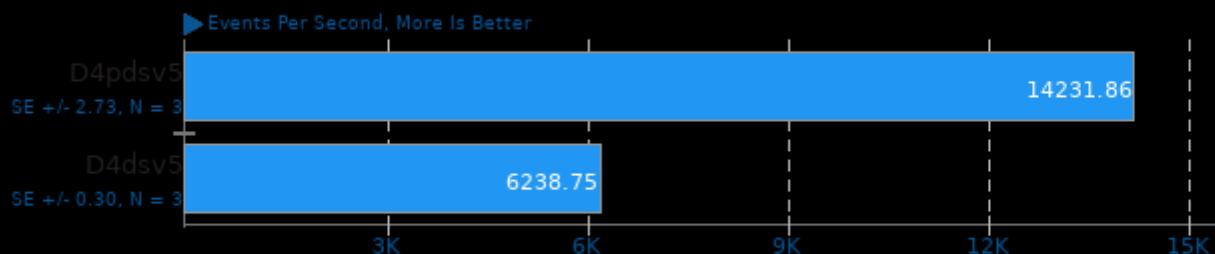
1. (CC) gcc options: -lm

OSBench



## Sysbench 1.0.20

Test: CPU

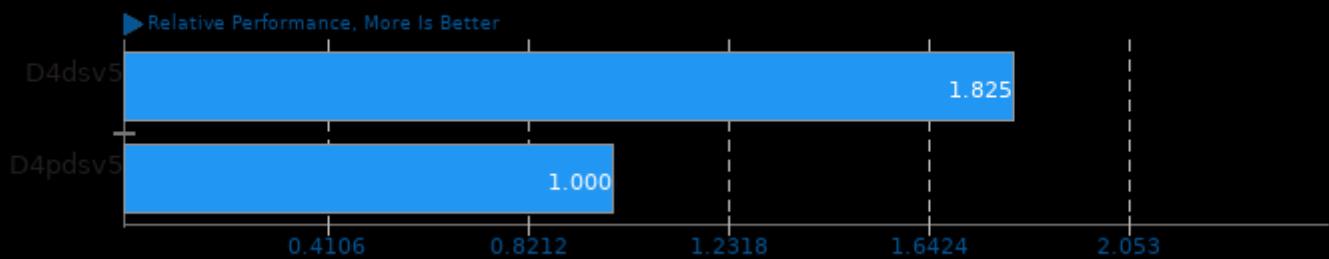


1. (CC) gcc options: -pthread -O2 -funroll-loops -rdynamic -ldl -lao -lm

These geometric means are based upon test groupings / test suites for this result file.

### Geometric Mean Of CPU Massive Tests

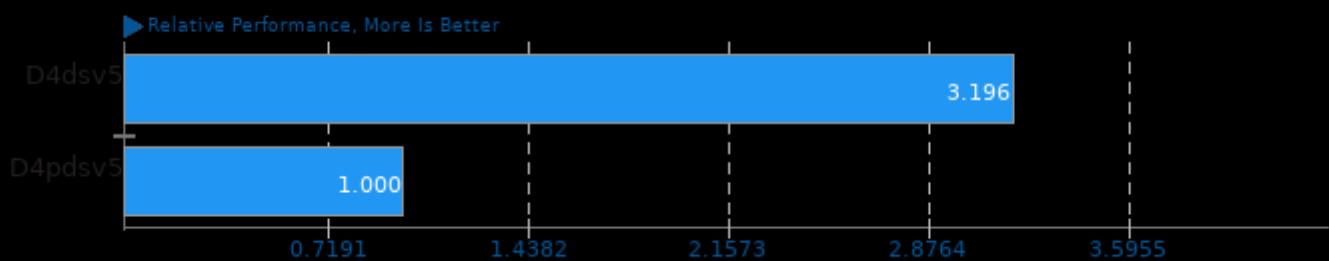
Result Composite - 2022-06-17-0443



Geometric mean based upon tests: pts/npb, pts/stockfish, pts/swet, pts/sysbench and pts/botan

### Geometric Mean Of HPC - High Performance Computing Tests

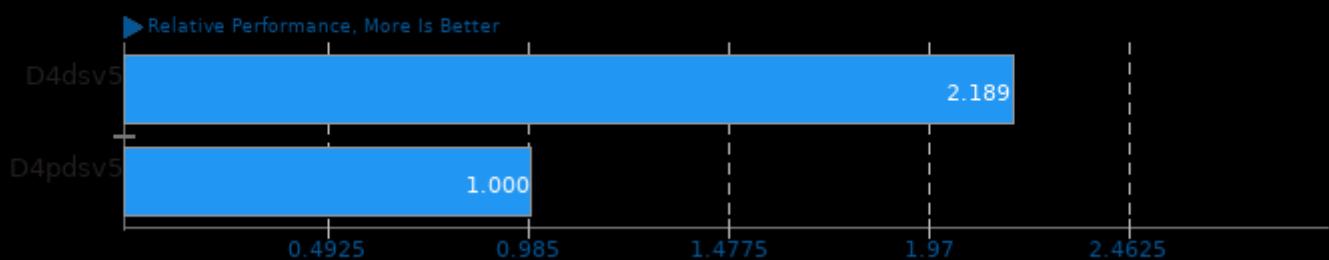
Result Composite - 2022-06-17-0443



Geometric mean based upon tests: pts/npb and pts/mt-dgemm

### Geometric Mean Of Multi-Core Tests

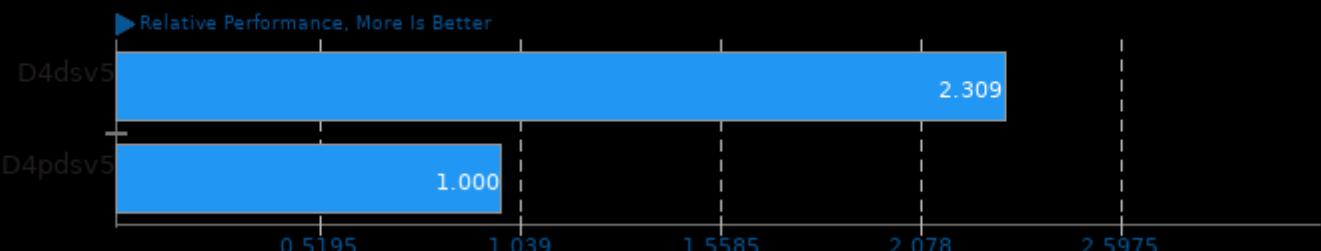
Result Composite - 2022-06-17-0443



Geometric mean based upon tests: pts/sysbench, pts/stockfish, pts/coremark, pts/npb, pts/swet and pts/mt-dgemm

## Geometric Mean Of Server CPU Tests

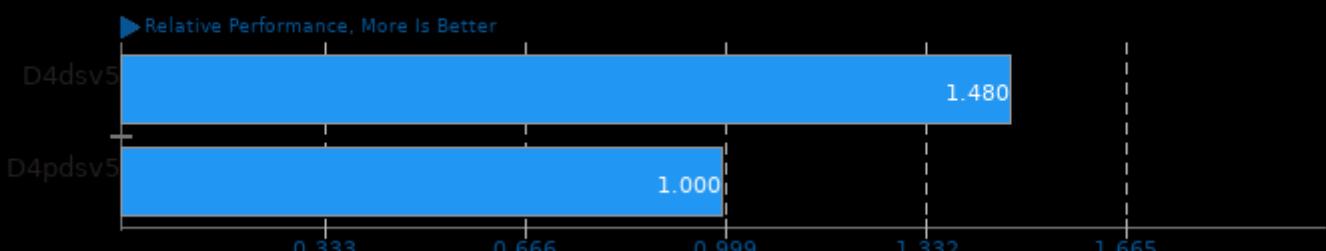
Result Composite - 2022-06-17-0443



Geometric mean based upon tests: pts/npb, pts/stockfish and pts/sysbench

## Geometric Mean Of Single-Threaded Tests

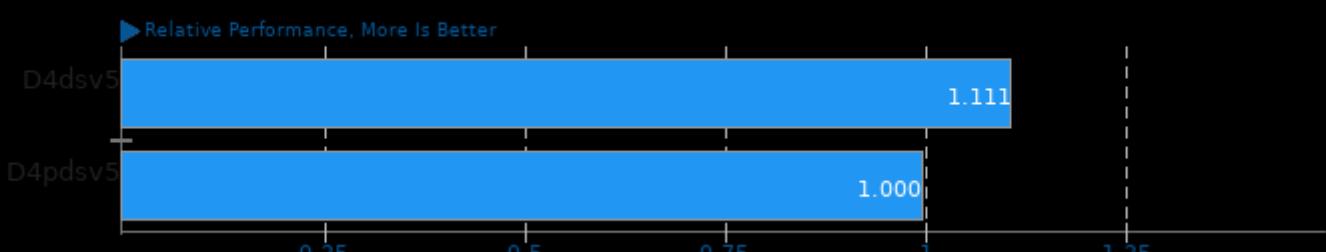
Result Composite - 2022-06-17-0443



Geometric mean based upon tests: pts/fhourstones, pts/byte, pts/botan and pts/swet

## Geometric Mean Of Common Workstation Benchmarks Tests

Result Composite - 2022-06-17-0443



Geometric mean based upon tests: pts/swet and pts/sysbench

*This file was automatically generated via the Phoronix Test Suite benchmarking software on Thursday, 28 March 2024 04:49.*