Apple

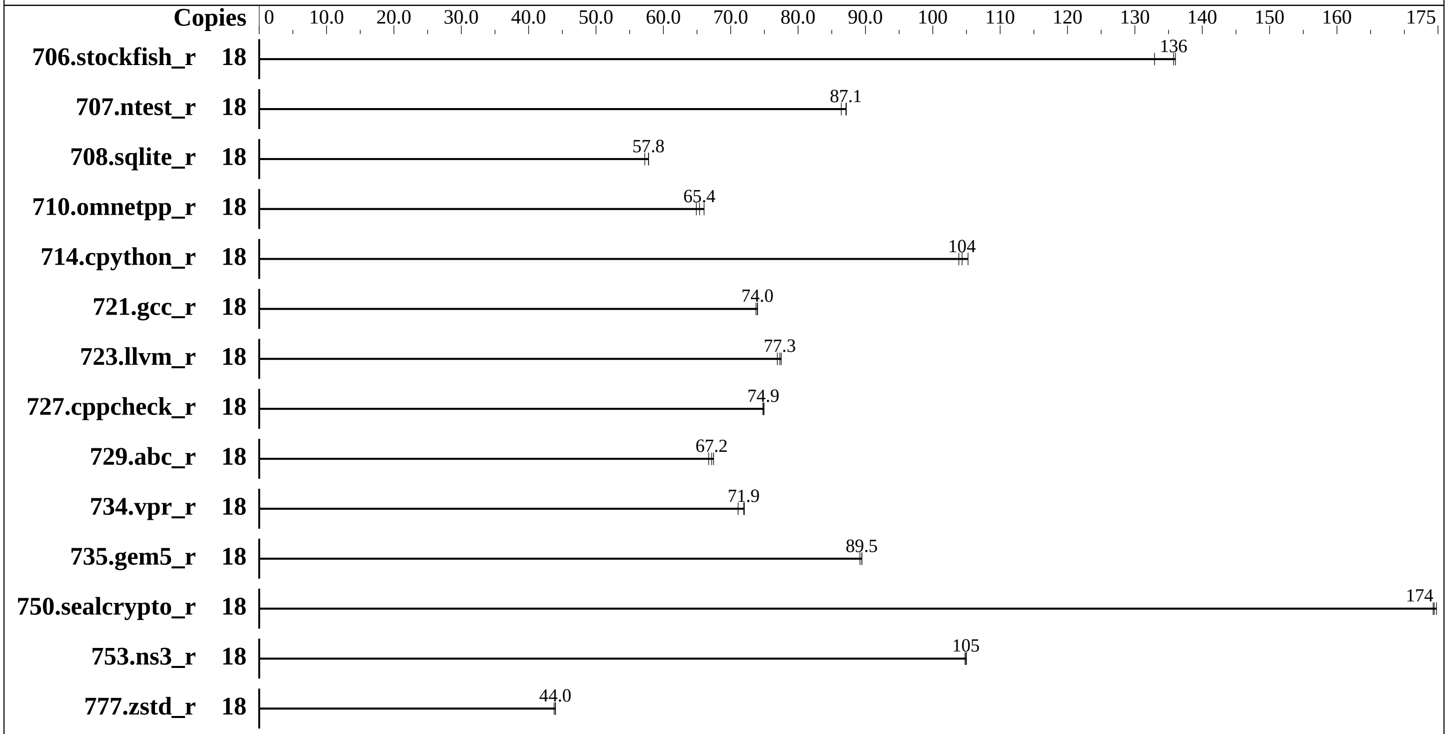
(Test Sponsor: SPEC CPU Supporting Contributor jhenning)

MacBook Pro (16-inch, M5 Pro) 18-copy SPECrate

SPECrate®2026_int_base = 82.7

SPECrate®2026_int_peak = Not Run

CPU2026 License:	7	**Test Date:**	Apr-2026
Test Sponsor:	SPEC CPU Supporting Contributor jhenning	**Hardware Availability:**	Mar-2026
Tested by:	SPEC CPU Supporting Contributor jhenning	**Software Availability:**	Apr-2026



Hardware

CPU Name:	Apple M5 Pro
Max MHz:	4600 (approximate; see notes)
Nominal:	4600 (approximate; see notes)
Enabled:	18 (6 Super and 12 Performance) cores, 1 chip
Orderable:	1 chip
Cache L1:	192 KB I + 128 KB D on chip per core (Super); 128 KB I + 64 KB D on chip per core (Performance)
L2:	32 MB I+D on chip per chip, (16 MB per 6 Super cores, 8 MB per 6 Performance cores)
L3:	None disclosed
Other:	None disclosed
Memory:	64.0 GB
Storage:	4 TB APPLE SSD AP4096Z
Cooling:	Air
Other:	None

Software

OS:	macOS 26.4.1 (25E253)
Compiler:	C/C++: Version 22.1.4 of LLVM
Compiler Category:	Community
Firmware:	Version 18000.101.7 released Apr-2026
File System:	apfs
System State:	Default
Base Pointers:	64-bit
Peak Pointers:	Not Applicable
Other:	Xcode 26.4.1.0.1775747724
Power Management:	Default

Results Table

	Base							Peak						
Benchmark	Copies	Seconds	Ratio	Seconds	Ratio	Seconds	Ratio	Copies	Seconds	Ratio	Seconds	Ratio	Seconds	Ratio
706.stockfish_r	18	171	133	**167**	**136**	167	136							
707.ntest_r	18	123	86.4	122	87.2	**122**	**87.1**							
708.sqlite_r	18	166	57.2	**164**	**57.8**	164	57.8							
710.omnetpp_r	18	**134**	**65.4**	132	66.0	135	64.9							
714.cpython_r	18	83.0	104	81.9	105	**82.6**	**104**							
721.gcc_r	18	167	73.8	**167**	**74.0**	167	74.0							
723.llvm_r	18	119	76.9	118	77.5	**118**	**77.3**							
727.cppcheck_r	18	86.4	74.8	**86.3**	**74.9**	86.2	74.9							
729.abc_r	18	124	66.7	**123**	**67.2**	122	67.5							
734.vpr_r	18	117	71.1	115	72.0	**115**	**71.9**							
735.gem5_r	18	**98.0**	**89.5**	98.0	89.5	98.3	89.2							
750.sealcrypto_r	18	55.2	175	**55.3**	**174**	55.4	174							
753.ns3_r	18	105	105	**105**	**105**	105	105							
777.zstd_r	18	265	43.8	264	44.0	**264**	**44.0**							

SPECrate®2026_int_base = 82.7

SPECrate®2026_int_peak = Not Run

Results appear in the order in which they were run. Bold underlined text indicates a median measurement.

Compiler Notes

```
NOTE - MODERATE COMPILER TUNING

The results on this page intentionally use only moderate compiler tuning. This page differs from many other SPEC CPU
results which are published to demonstrate the best performance that can be achieved by a system. Such tests often
use compilers that are optimized for specific chip vendors and use advanced tuning options.

Moderate optimization may be of interest to users who prefer a simple build process, or who do not know whether
their application works correctly with advanced tuning options. Specifically, these tests:

   - Use an open source, community-supported compiler, rather than a vendor-specific compiler.
   - Specify only one optimization option, namely -O3.
   - Avoid options that disregard strict standards compliance, such as -Ofast, -ffinite-math-only, or
     -funsafe-math-optimizations.
   - Avoid options such as -march=native or -mcpu=native. Without those, the code generator cannot assume that
     the program will run only on systems that use the same chip as the current system.

There are many other kinds of testing can be done with SPEC CPU, such as:
  - Performance modeling prior to CPU chip development.
  - Hardware validation.
  - Compiler regression testing (both correctness and performance).
  - Academic and industrial research into new optimizations for CPUs, memory systems, and compilers.
```

Environment Variables Notes

```
Environment variables set by runcpu before the start of the run:
LD_LIBRARY_PATH =
     "/usr/local/LLVM-22.1.4-macOS-ARM64//lib64/:/usr/local/LLVM-22.1.4-macOS
     -ARM64//lib/:/lib64"
SDKROOT =
     "/Applications/Xcode.app/Contents/Developer/Platforms/MacOSX.platform/De
     veloper/SDKs/MacOSX.sdk"
```

General Notes

```
 Yes: The test sponsor attests, as of date of publication, that CVE-2017-5715
      (Spectre variant 2) is mitigated in the system as tested and documented.
 Yes: The test sponsor attests, as of date of publication, that CVE-2017-5753
      (Spectre variant 1) is mitigated in the system as tested and documented.
 Yes: The test sponsor attests, as of date of publication, that CVE-2017-5754
      (Meltdown) is mitigated in the system as tested and documented.
 The above 3 statements are based on https://support.apple.com/en-us/101886
```

Platform Notes

```
 Sysinfo program /Volumes/ScratchVolume/spec/cpu2026/1.0.0/bin/sysinfo
 Rev: 779ab21020787073335a329f3a45e2cd
 running on i4gots-MacBook-Pro Tue Apr 21 20:20:19 2026

 SUT (System Under Test) info as seen by some common utilities.

 uname -srvm
   Darwin 25.4.0 Darwin Kernel Version 25.4.0: Thu Mar 19 19:33:50 PDT 2026;
   root:xnu-12377.101.15~1/RELEASE_ARM64_T6050 arm64

 w
   20:20  up  1:10, 3 users, load averages: 2.26 2.16 2.16
   USER       TTY      FROM      LOGIN@  IDLE WHAT
   i4got      console  -        19:10    1:09 -
   i4got      s000     -        19:12      59 -bash
   i4got      s001     -        19:20      58 -bash

 Username
   From environment variable $USER:  i4got

 ulimit -a
   core file size          (blocks, -c) 0
```

(Continued on next page)

Platform Notes (Continued)

```
    data seg size            (kbytes, -d) unlimited
    file size                (blocks, -f) unlimited
    max locked memory        (kbytes, -l) unlimited
    max memory size          (kbytes, -m) unlimited
    open files                       (-n) 256
    pipe size             (512 bytes, -p) 1
    stack size               (kbytes, -s) 65000
    cpu time                (seconds, -t) unlimited
    max user processes               (-u) 10666
    virtual memory           (kbytes, -v) unlimited

  sysinfo process ancestry
   /sbin/launchd
   /System/Applications/Utilities/Terminal.app/Contents/MacOS/Terminal
   login -pf i4got
   -bash
   /bin/bash ./run.4.sh
   runcpu -c clang4 -n 3 --reportable --nobuild --copies=18 intrate
   runcpu --configfile clang4 --iterations 3 --reportable --nobuild --copies 18 --nopower --runmode rate --tune
     base --size refrate intrate --nopreenv --note-preenv --logfile
     $GOtmp/CPU2026.123/templogs/preenv.intrate.123.0.log --lognum 123.0 --from_runcpu 2
   specperl $SPEC/bin/sysinfo
  $SPEC = /Volumes/ScratchVolume/spec/cpu2026/1.0.0
  $GO   = /Volumes/ScratchVolume/spec/cpu2026/out/1.0.0/

  Hardware:

      Hardware Overview:

        Model Name: MacBook Pro
        Model Identifier: Mac17,8
        Model Number: Z1N0001A3LL/A
        Chip: Apple M5 Pro
        Total Number of Cores: 18 (6 Super and 12 Performance)
        Memory: 64 GB
        System Firmware Version: 18000.101.7
        OS Loader Version: 18000.101.7

  Software:

      System Software Overview:

        System Version: macOS 26.4.1 (25E253)
        Kernel Version: Darwin 25.4.0
        Boot Volume: Macintosh HD
        Boot Mode: Normal
        Computer Name: i4got’s MacBook Pro
```

(Continued on next page)

Platform Notes (Continued)

```
      User Name: i4got (i4got)
      Secure Virtual Memory: Enabled
      System Integrity Protection: Enabled
      Time since boot: 1 hour, 10 minutes

Memory:

      Memory: 64 GB
      Type: LPDDR5
      Manufacturer: Micron

Developer:

    Developer Tools:

      Version: 26.4.1 (17E202)
      Location: /Applications/Xcode.app
      Applications:
          Xcode: 26.4.1 (24909.0.3)
          Instruments: 26.4.1 (64575.156)
      SDKs:
          DriverKit:
              25.4:
          iOS:
              26.4: (23E252)
          iOS Simulator:
              26.4: (23E252)
          macOS:
              26.4: (25E251)
          tvOS:
              26.4: (23L236)
          tvOS Simulator:
              26.4: (23L236)
          visionOS:
              26.4: (23O248)
          visionOS Simulator:
              26.4: (23O248)
          watchOS:
              26.4: (23T238)
          watchOS Simulator:
              26.4: (23T238)


From sysctl:
   machdep.cpu.core_count: 18
   machdep.cpu.thread_count: 18
   machdep.cpu.brand_string: Apple M5 Pro
```

(Continued on next page)

Platform Notes (Continued)

```
    hw.memsize: 68719476736
    hw.activecpu: 18
    hw.perflevel0.l1icachesize: 196608
    hw.perflevel0.l1dcachesize: 131072
    hw.perflevel0.l2cachesize: 16777216
    hw.perflevel1.l1icachesize: 131072
    hw.perflevel1.l1dcachesize: 65536
    hw.perflevel1.l2cachesize: 8388608
    hw.physicalcpu: 18
    hw.physicalcpu_max: 18
    hw.logicalcpu: 18
    hw.logicalcpu_max: 18
    hw.cacheconfig: 18 1 6 0 0 0 0 0 0 0
    hw.cachesize: 2995912704 65536 8388608 0 0 0 0 0 0 0
    hw.pagesize: 16384
    hw.pagesize32: 16384
    hw.cachelinesize: 128
    hw.l1icachesize: 131072
    hw.l1dcachesize: 65536
    hw.l2cachesize: 8388608
    hw.memsize_usable: 67420422144

 SPEC is set to: /Volumes/ScratchVolume/spec/cpu2026/1.0.0
    Filesystem      Size    Used   Avail Capacity iused ifree %iused  Mounted on
    /dev/disk3s1   1998G    558G   1440G    28%    1.3M   14G    0%   /Volumes/ScratchVolume

 (End of data from sysinfo program)


 ADDITIONAL INFORMATION FROM SYSCTL

  $ sysctl hw | grep  perflevel0
  hw.perflevel0.physicalcpu: 6
  hw.perflevel0.physicalcpu_max: 6
  hw.perflevel0.logicalcpu: 6
  hw.perflevel0.logicalcpu_max: 6
  hw.perflevel0.l1icachesize: 196608
  hw.perflevel0.l1dcachesize: 131072
  hw.perflevel0.l2cachesize: 16777216
  hw.perflevel0.cpusperl2: 6
  hw.perflevel0.name: Super
  $ sysctl hw | grep  perflevel1
  hw.perflevel1.physicalcpu: 12
  hw.perflevel1.physicalcpu_max: 12
  hw.perflevel1.logicalcpu: 12
  hw.perflevel1.logicalcpu_max: 12
  hw.perflevel1.l1icachesize: 131072
```

(Continued on next page)

Platform Notes (Continued)

```
  hw.perflevel1.l1dcachesize: 65536
  hw.perflevel1.l2cachesize: 8388608
  hw.perflevel1.cpusperl2: 6
  hw.perflevel1.name: Performance
  $
```

```
The above shows cache sizes for both core types, and "cpusperl2" indicates sharing among cores. See
https://developer.apple.com/documentation/kernel/1387446-sysctlbyname/determining_system_capabilities
Note: it is possible that the sysctl cache information may be incomplete.  There might be other cache
types or other cache characteristics that are not disclosed by the above 'sysctl' command.


ADDITIONAL INFORMATION FROM POWERMETRICS

Regarding the CPU MHz fields above, note that:
 - The vendor does not state CPU frequencies in their marketing materials.
 - Frequency levels are printed by the vendor-provided 'powermetrics' utility; these might or might
   not correspond to traditional "nominal" or "maximum" MHz.  Therefore, the values on this page for
   Nominal and Maximum MHz must be considered approximate.
 - The table below lists all of the MHz values mentioned in the dvfm_states section of powermetrics.
 - On the tested system, the maximum frequencies mentioned are 4608 for the S-cluster and 4380 for the
   P-Clusters

----------------------------------------------------------------------------------------------------
CPU Dynamic Frequencies                                          From 'powermetrics --format=plist'
----------------------------------------------------------------------------------------------------
processor
clusters    dvfm_states
name        freq
----------  ----------------------------------------------------------------------------------------
P0-Cluster  1344 1644 1992 2304 2652 2964 3240 3504 3696 3876 4044 4176 4284 4308 4380
P1-Cluster  1344 1644 1992 2304 2652 2964 3240 3504 3696 3876 4044 4176 4284 4308 4380
S-Cluster   1308 1620 1980 2292 2580 2880 3180 3432 3648 3828 3984 4104 4188 4236 4284 4308 4332 4428 4512 4608

The table above was extracted from powermetrics via Perl code:

open P, "powermetrics --sample-count=0  --show-initial-usage --samplers=cpu_power --format=plist|"
   or die "cannot run powermetrics";
while (<P>) {
   if (m{<key>name</key><string>(\S+Cluster)</string>}) {
      printf "%-10s  ", $1;
      $printing = 1;
      @freqs    = ();
   }
   next if ! $printing;
   if (m{^</array>}) {
      print join(" ", @freqs), "\n";
```

(Continued on next page)

Platform Notes (Continued)

```
        $printing = 0;
    }
    if (m{<key>freq</key><integer>(\d+)</integer>}) {
        push @freqs, $1;
    }
}
```

Compiler Version Notes

```
==============================================================================
C      | 708.sqlite_r(base) 714.cpython_r(base) 777.zstd_r(base)
------------------------------------------------------------------------------
clang version 22.1.4 (https://github.com/llvm/llvm-project
  35990504507d79e0b9deb809c8ee5e1b34ceef20)
Target: arm64-apple-darwin25.4.0
Thread model: posix
InstalledDir: /usr/local/LLVM-22.1.4-macOS-ARM64/bin
------------------------------------------------------------------------------

==============================================================================
C++    | 706.stockfish_r(base) 707.ntest_r(base) 727.cppcheck_r(base)
       | 753.ns3_r(base)
------------------------------------------------------------------------------
clang version 22.1.4 (https://github.com/llvm/llvm-project
  35990504507d79e0b9deb809c8ee5e1b34ceef20)
Target: arm64-apple-darwin25.4.0
Thread model: posix
InstalledDir: /usr/local/LLVM-22.1.4-macOS-ARM64/bin
------------------------------------------------------------------------------

==============================================================================
C++, C | 710.omnetpp_r(base) 721.gcc_r(base) 723.llvm_r(base) 729.abc_r(base)
       | 734.vpr_r(base) 735.gem5_r(base) 750.sealcrypto_r(base)
------------------------------------------------------------------------------
clang version 22.1.4 (https://github.com/llvm/llvm-project
  35990504507d79e0b9deb809c8ee5e1b34ceef20)
Target: arm64-apple-darwin25.4.0
Thread model: posix
InstalledDir: /usr/local/LLVM-22.1.4-macOS-ARM64/bin
------------------------------------------------------------------------------
```

Base Compiler Invocation

C benchmarks:
`clang`

C++ benchmarks:
`clang++`

Benchmarks using both C and C++:
`clang++ clang`

Base Portability Flags

714.cpython_r: `-UHAVE_PTHREAD_STUBS`

Base Optimization Flags

C benchmarks:
`-std=c18 -g -O3 -Wno-implicit-function-declaration`

C++ benchmarks:
`-std=c++17 -g -O3 -Wno-implicit-function-declaration`

Benchmarks using both C and C++:
`-std=c++17 -std=c18 -g -O3 -Wno-implicit-function-declaration`

The flags file that was used to format this result can be browsed at
http://www.spec.org/cpu2026/results/flags/llvm-rev-A3.html

You can also download the XML flags source by saving the following link:
http://www.spec.org/cpu2026/results/flags/llvm-rev-A3.xml

For questions about this result, please contact the tester. For other inquiries, please contact info@spec.org.

Tested with SPEC CPU®2026 v1.0.0 on 2026-04-21 20:20:18-0400.
Report generated on 2026-05-04 23:34:44 by CPU2026 PS formatter (86729d3b).
Originally published on 2026-05-05.