



RAČUNALNIŠKA ARHITEKTURA

7 Merjenje zmogljivosti CPE

7 Merjenje zmogljivosti CPE- cilji:

■ Osnovno razumevanje pojmov:

- Pogostost izvajanja ukazov
- Čas izvajanja ukazov
- CPI, MIPS
- CPEčas

■ Razumevanje meritev zmogljivosti:

- CPEčas
- Sintetični benchmark programi
- SPEC

7 Merjenje zmogljivosti CPE

- Pogostost izvajanja posameznih vrst ukazov
- Čas izvajanja posameznih vrst ukazov
- CPI – povprečno število urinih period za izvedbo ukaza v določenem programu . MIPS – povprečno število ukazov (v milijonih), ki se izvedejo v eni sekundi
- CPEčas – čas izvrševanja programa v sekundah
- Benchmark programi
- SPEC

- Zmogljivost računalnikov se tradicionalno meri predvsem z zmogljivostjo CPE.
- Pomnilnik in V/I naprave se pri tem običajno ignorira.
- Opravičilo za to je, da naj bi bila pomnilnik in V/I sistem dovolj zmogljiva, da ne povzročata čakanja CPE.
- V takem primeru lahko zmogljivost računalnika enačimo z zmogljivostjo CPE.

Pogostost izvajanja posameznih vrst ukazov

Z merjenjem (štetjem) lahko ugotovimo, kako pogosto (odstotek celotnega časa izvajanja programa) se v nekem programu izvajajo posamezne skupine ukazov.

- **Statična pogostost** - število posameznih vrst ukazov v programu.
- **Dinamična pogostost** - število izvrševanj posameznih vrst ukazov pri izvajanju programa.
 - V programu je npr. 20 ukazov, od tega je 6 ukazov v zanki, ki se izvede 10krat, vseh ukazov je 20; število ukazov, ki se izvedejo v programu, pa je 74.
 - Dinamično pogostost izvajanja ukazov vrste i označimo s p_i in izrazimo v odstotkih celotnega časa izvajanja programa.

- Pogostost izvajanja posameznih vrst ukazov se med programi običajno razlikuje.
- Primer dinamične pogostosti izvajanja posameznih vrst ukazov na določenem računalniku pri dveh različnih programih P1 in P2:

i	Vrsta ukazov	p_i % za program P1	p_i % za program P2
1	Load/store	40%	25%
2	ALE	45%	35%
3	Kontrolni ukazi	10%	10%
4	V/I ukazi	5%	8%
5	FP ukazi	0	22%
	Skupaj v programu	100%	100%

Čas izvajanja posameznih vrst ukazov

- Trajanje izvajanja ukaza se običajno podaja v urinih periodah. Tako podatek velja za vse (istovrstne) procesorje z različnimi frekvencami ure.
- Podatek o trajanju posamezne vrste ukaza ima oznako **CPI (Cycles per Instruction)** in je enak številu urinih period, ki so potrebne za izvedbo ukaza
- Za posamezen ukaz je CPI vedno celo število (ukaz se v CPE izvaja vedno celo število urinih period)
- Če CPI pomnožimo s časom trajanja urine periode t_{CPE} , dobimo čas izvajanja ukaza v sekundah.

- Primer časov izvajanja ukaza v sekundah pri različnih frekvencah ure, če se ukaz izvaja 5 urinih period $\Rightarrow CPI = 5$:

CPI [urine periode]	f_{CPE} [Hz]	t_{CPE} [s]	Trajanje ukaza [s]
5	100 MHz	10 ns	50 ns
5	800 MHz	1,25 ns	6,25 ns
5	2 GHz	0,5 ns	2,5 ns
5	3 GHz	0,33 ns	1,65 ns

1 MHz = 10^6 Hz 1 ns = 10^{-9} s 1 GHz = 10^9 Hz
1Hz = 1/s = ena perioda na sekundo

- Za nek program lahko iz podatkov o:
 - dinamični pogostosti izvajanja posameznih vrst ukazov p_i
 - in številu urinih period CPI_i , potrebnih za izvedbo posamezne vrste ukazov
- izračunamo povprečno število urinih period za izvedbo enega ukaza.
- **Povprečno število** urinih period CPI , ki so potrebne za izvedbo enega ukaza v določenem programu z n različnimi vrstami ukazov:

$$CPI = \sum_{i=1}^n CPI_i \cdot p_i$$

- Primer izračuna *CPI* (povprečno število urinih period za izvedbo ukaza) za pogostost izvajanja ukazov v programu P2 (iz tabele na str. 4):

i	Vrsta ukazov	p_i % za program P1	p_i % za program P2	CPI_i
1	Load/store	40%	25%	6
2	ALE	45%	35%	4
3	Kontrolni ukazi	10%	10%	3
4	V/I ukazi	5%	8%	7
5	FP ukazi	0	22%	8

$$CPI(P2) = \sum_{i=1}^5 p_i \cdot CPI_i = 0,25 \cdot 6 + 0,35 \cdot 4 + 0,10 \cdot 3 + 0,08 \cdot 7 + 0,22 \cdot 8 = 5,52$$

- Primer izračuna *CPI* (povprečno število urinih period za izvedbo ukaza) za pogostost izvajanja ukazov v programu P1 (iz spodnje tabele):

i	Vrsta ukazov	p_i % za program P1	p_i % za program P2	CPI_i
1	Load/store	40%	25%	6
2	ALE	45%	35%	4
3	Kontrolni ukazi	10%	10%	3
4	V/I ukazi	5%	8%	7
5	FP ukazi	0	22%	8

$$CPI(P2) = \sum_{i=1}^5 p_i \cdot CPI_i = 0,25 \cdot 6 + 0,35 \cdot 4 + 0,10 \cdot 3 + 0,08 \cdot 7 + 0,22 \cdot 8 = 5,52$$

$$CPI(P1) = ? \quad CPI(P1) = 4,85$$

- Če poznamo frekvenco ure f_{CPE} , lahko za določen program izračunamo tudi, koliko ukazov povprečno izvede procesor v sekundi.
- Ker je rezultat velikostnega reda nekaj milijonov (ukazov/sek), ga običajno delimo z 10^6 in dobimo rezultat v milijonih ukazov, ki jih procesor izvede v sekundi.
- **MIPS (Million Instructions Per Second)**

$$MIPS = \frac{f_{CPE}}{CPI \cdot 10^6}$$

$$t_{CPE} = \frac{1}{f_{CPE}}$$



$$MIPS = \frac{1}{CPI \cdot t_{CPE} \cdot 10^6}$$

- Vzemimo da je:
 - frekvenca ure procesorja $f_{CPE} = 1,8 \text{ GHz}$,
 - povprečni $CPI = 5,52$ urine periode (pri izvajanju programa P2)
 - potem lahko izračunamo $MIPS$:

$$MIPS(P2) = \frac{1,8 \cdot 10^9}{5,52 \cdot 10^6} = \frac{1800}{5,52} = 326,08$$

- Za isti procesor lahko izračunamo, da je $MIPS$ pri izvajanju programa P1:

$$MIPS(P1) = \frac{1,8 \cdot 10^9}{4,85 \cdot 10^6} = \frac{1800}{4,85} = 371,13$$

- Če smatramo, da sta pomnilnik in V/I sistem dovolj zmogljiva, da ne povzročata čakanja CPE, lahko zmogljivost računalnika enačimo z zmogljivostjo CPE.
- Edino pravo merilo zmogljivosti CPE pa je **čas izvrševanja programa** merjen v sek/program.
- Če zanemarimo čas za V/I prenose, lahko čas izvajanja programa izenačimo s časom, ki ga potrebuje CPE.

$$CPE_{čas} = \text{število_ukazov} \cdot CPI \cdot t_{CPE}$$

- Zmogljivost CPE je odvisna od:
 - **števila ukazov**, v katere se prevede določeni program
 - **CPI** - povprečnega števila urinih period za izvedbo ukaza
 - **t_{CPE}** - urine periode oziroma frekvence ure
- Te tri lastnosti so med seboj odvisne in vplivajo ena na drugo. Kaj vpliva nanje?

- Na urino periodo t_{CPE} ali frekvenco ure vpliva:
 - Hitrost in število digitalnih vezij, s katerimi je zgrajena CPE
 - Zgradba kontrolne in podatkovne enote

- Na *CPI* - povprečno število urinih period za izvedbo ukaza vpliva:
 - Zgradba kontrolne in podatkovne enote
 - Vrsta in število ukazov procesorja
 - Program

- Na število ukazov, v katere se prevede program, vpliva:
 - Število in vrsta ukazov procesorja
 - Lastnosti prevajalnika

- Pri primerjavi zmogljivosti računalnikov je treba zato primerjati vse tri lastnosti (oziroma njihov produkt - $CPE_{čas}$), ne pa samo eno od njih.

- Kljub tej ugotovitvi pa čas za praktično uporabo ni najbolj primeren.
 - Odvisen je od programa
 - Pri istem programu odvisen tudi od vhodnih podatkov
- Vendar so se vsi drugi načini, ki naj bi bili neodvisni od programov in bi služili za primerjavo različnih računalnikov, izkazali za slabše ali celo zavajajoče.

- Primer: Za določeni program, ki se je prevedel in izvedel na dveh različnih računalnikih R1 in R2, so bili izmerjeni naslednji podatki:

Meritev	Računalnik R1	Računalnik R2
Število strojnih ukazov	10×10^9	8×10^9
Frekvenca ure	4 GHz	4 GHz
CPI	1,0	1,1

- Kateri računalnik ima večji MIPS?
- Kateri računalnik je hitrejši pri izvajanju tega programa?

- **Frekvenca ure f_{CPE}** je neustrezna mera za primerjavo hitrosti (zmogljivosti) različnih računalnikov.
- Tudi **parameter MIPS** ni najboljši za primerjavo zmogljivosti različnih računalnikov:
 - Odvisen je od števila in vrste ukazov, zato ni primeren za primerjavo računalnikov z različnim naborom ukazov
 - Odvisen je od programa (od vrste ukazov, ki se izvajajo v programu)
 - Večji MIPS ne pomeni vedno bolj zmogljiv računalnik (primerjava računalnika s programsko realizacijo FP ukazov in računalnika s FP enoto ali primerjava RISC in CISC računalnikov)

- MIPS – hudomušna oznaka „Meaningless Indication of Processor Speed“
- Da bi odpravili te pomanjkljivosti, se je začel **uporabljati relativni MIPS**, kjer se čas izvajanja programa primerja s časom, izmerjenim na referenčnem računalniku.
 - Težave pri izbiri referenčnega računalnika
 - Potrebno meriti čas, kar pri MIPS ni potrebno

- **M(GT)FLOPS** (.. Floating Point Operations Per Second)
 - Tudi odvisen od programa
 - Smisel ima samo pri procesorjih, ki imajo operacije v plavajoči vejici
 - Resnična zmogljivost v MFLOPS je pogosto samo 10 % teoretične, ki jo navajajo proizvajalci

Izboljšava: Uporaba posebej za merjenje zmogljivosti napisanih programov – **sintetični benchmarki.**

Merjenje zmogljivosti CPE

■ **M(GT)FLOPS** (.. Floating Point Operations Per Second) - Primer

TOP500 List - November 2019

R_{max} and R_{peak} values are in TFlops. For more details about other fields, check the [TOP500 description](#).

R_{peak} values are calculated using the advertised clock rate of the CPU. For the efficiency of the systems you should take into account the Turbo CPU clock rate where it applies.

- R_{max} - Maximal LINPACK performance achieved
- R_{peak} - Theoretical peak performance

previous 1 2 3 4 5 next

Rank	Site	System	Cores	R_{max} (TFlop/s)	R_{peak} (TFlop/s)	Power (kW)
1	DOE/SC/Oak Ridge National Laboratory United States	Summit - IBM Power System AC922, IBM POWER9 22C 3.07GHz, NVIDIA Volta GV100, Dual-rail Mellanox EDR Infiniband IBM	2,414,592	148,600.0	200,794.9	10,096
2	DOE/NNSA/LLNL United States	Sierra - IBM Power System AC922, IBM POWER9 22C 3.1GHz, NVIDIA Volta GV100, Dual-rail Mellanox EDR Infiniband IBM / NVIDIA / Mellanox	1,572,480	94,640.0	125,712.0	7,438
3	National Supercomputing Center in Wuxi China	Sunway TaihuLight - Sunway MPP, Sunway SW26010 260C 1.45GHz, Sunway NRCP	10,649,600	93,014.6	125,435.9	15,371

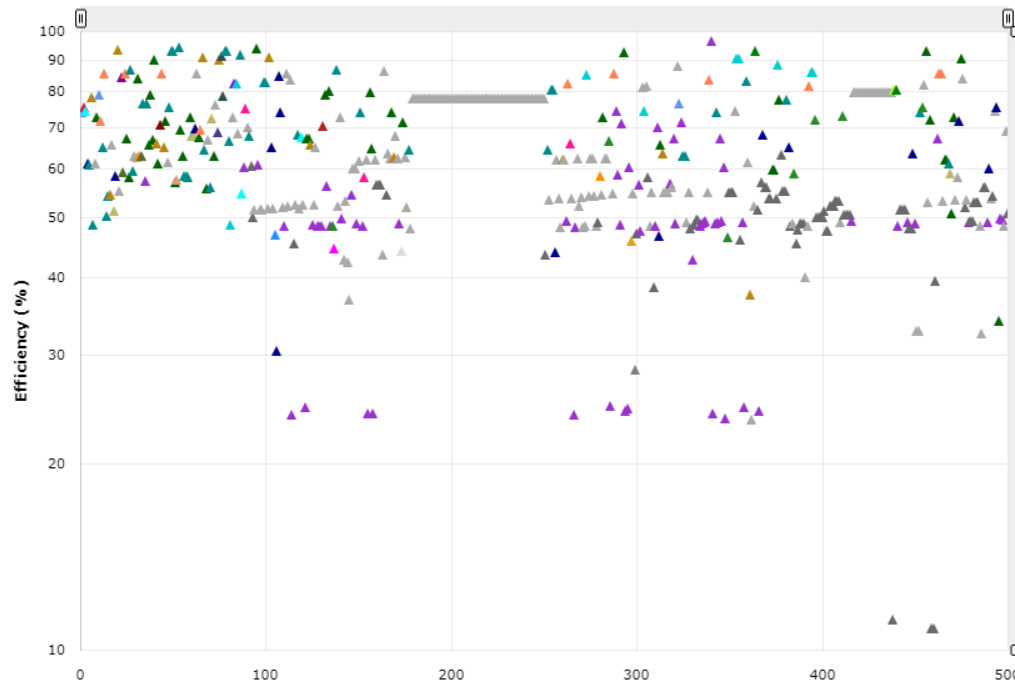
<https://www.top500.org/list/2019/11/>

- Rmax - Maximal LINPACK performance achieved
- Rpeak - Theoretical peak performance

Merjenje zmogljivosti CPE

■ M(GT)FLOPS (.. Floating Point Operations Per Second)

- Resnična zmogljivost R_{max} v M(GT)FLOPS pogosto nižja (včasih tudi samo 10 %) teoretične R_{peak} , ki jo navajajo proizvajalci



Legend:

IBM, Cray/HPE, Nvidia, NEC/HPE, Cray Inc./Hitachi, Huawei, T-Platforms, Dell EMC, Sugon, T-Platforms, Intel, Dell,
 NEC/MEGWARE, Fujitsu / Lenovo / Xenon, Atos, Cray Inc./T-Platforms, NTT Comm. / NTT PC Comm., NUDT, Dell EMC / IBM-
 GBS, Lenovo, NEC, Intel, IBM / NVIDIA / Mellanox, ClusterVision / Hammer, Self-made, IBM/Lenovo, Fujitsu, Penguin

<https://www.top500.org/statistics/efficiency-power-cores/>

- **Sintetični benchmark programi** so programi, s katerimi želimo primerjati zmogljivosti računalnikov:
 - Whetstone (Algol 60, FORTRAN, Pascal,...)
 - Linpack (FORTRAN)
 - Livermore loops (FORTRAN)
 - Dhrystone (ne vsebuje FP operacij, Ada, Pascal, C)
 - Quicksort (sortiranje)
 - Sieve (iskanje praštevil)
 - Puzzle

- Benchmark programi so bili znani, zato so proizvajalci računalnikov optimizirali delovanje (prevajalniki) samo za te programe!
- Do sedaj najboljša rešitev je uporaba večjega števila benchmark programov in izračun aritmetične ali geometrijske srednje vrednosti izmerjenih rezultatov.
- **SPEC:** Leta 1988 več firm ustanovi neprofitno organizacijo **SPEC** (Standard Performance Evaluation Corporation).

- Izberejo standardno množico benchmark programov za merjenje zmogljivosti CPE.
 - Prvi nabor programov objavljen leta 1989: SPECmark 89 (aritmetična srednja vrednost razmerja časov 10 programov glede na čas izmerjen na VAX-11/780)
 - Peta generacija objavljena leta 2006: SPEC CPU2006
 - CINT2006 za celoštevilčne operacije v fiksni vejici – 12 programov,
 - CFP2006 za operacije v plavajoči vejici – 17 programov
 - Referenčni čas za posamezen program se deli z izmerjenim časom programa, rezultat je »SPEC ratio« posameznega programa ⇒ večji rezultat pomeni boljšo zmogljivost
 - Končni rezultat je geometrična srednja vrednost posameznih zmogljivosti

2017: zadnja, šesta generacija:

Benchmark Description:

The **SPEC CPU® 2017** benchmark package contains SPEC's next-generation, industry-standardized, CPU intensive suites for measuring and comparing compute intensive performance, stressing a system's processor, memory subsystem and compiler.

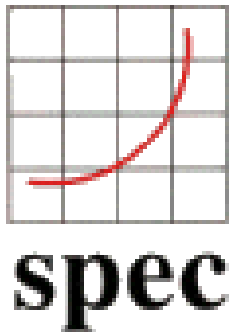
SPEC designed these suites to provide a comparative measure of compute-intensive performance across the widest practical range of hardware using workloads developed from real user applications. The benchmarks are provided as source code and require the use of compiler commands as well as other commands via a shell or command prompt window.

<http://www.intel.com/performance>

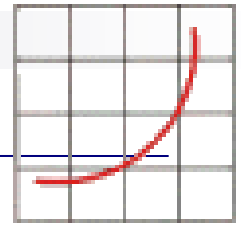
<https://www.spec.org/cpu2017/>

[Search across all the SPEC CPU® 2017 benchmark results in SPEC's online result database.](#)

- To niso več sintetični programi (narejeni posebej za merjenje zmogljivosti), temveč realne aplikacije, ki jih obnavljajo in dopolnjujejo vsakih nekaj let.
- Ti programi omogočajo ovrednotenje zmogljivosti CPE, pomnilniške arhitekture in prevajalnika.



<http://www.spec.org>



spec

SPEC2017 – praktični vidiki :

Q16. What are "base" and "peak" metrics?

- The **base metrics** (such as SPECspeed2017_int_base) require that all modules of a given language in a suite **must be compiled using the same flags, in the same order.**
- The optional **peak metrics** (such as SPECspeed2017_int_peak) allow greater flexibility. **Different compiler options may be used for each benchmark, and feedback-directed optimization is allowed.**

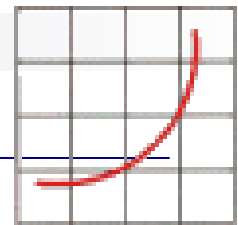
Q17. Which SPEC CPU 2017 metric should I use?

Examples:

- A **single user running a variety of generic desktop programs** may, perhaps, be interested in
 - **SPECspeed2017_int_base.**
- A **group of scientists** running customized modeling programs may, perhaps, be interested in
 - **SPECrate2017_fp_peak.**

Z naslova <<https://www.spec.org/cpu2017/Docs/overview.html#Q16>>

SPEC2017 – primer iskanja po rezultatih :



spec

SPEC® CPU2017 Floating Point Rate Result			
Copyright 2017-2019 Standard Performance Evaluation Corporation			
Supermicro SuperWorkstation 5039C-T (X11SCA , Intel Core i7-9700K)		SPECrate2017_fp_base =	42.6
		SPECrate2017_fp_peak =	43.3
CPU2017 License:	001176	Test Date:	Jan-2019
Test Sponsor:	Supermicro	Hardware Availability:	Oct-2018
Tested by:	Supermicro	Software Availability:	Mar-2018

Benchmark result graphs are available in the PDF report.

CPU2017 Results -- Query

This configuration offers access to summary information across all CPU2017 results.

Simple Request

Fetch just the summary information for all results.

- Optional: Return only those results where
 - Processor: Matches Case

Execute Simple Fetch

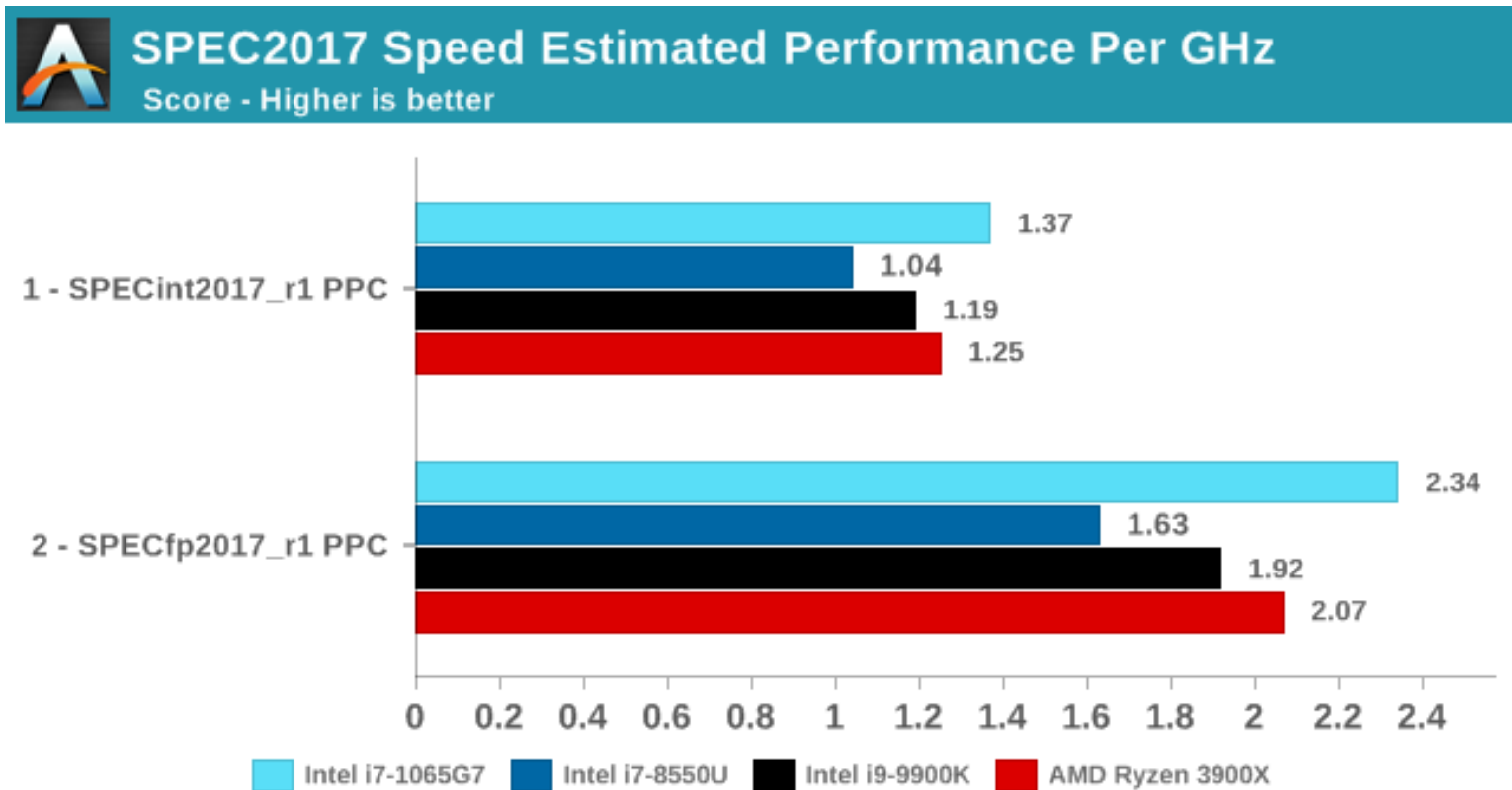
Z naslova <<https://www.spec.org/cgi-bin/osgresults?conf=cpu2017>>

Hardware				Software			
CPU Name:	Intel Core i7-9700K			OS:	SUSE Linux Enterprise Server 12 SP3 (x86_64)		
Max MHz.:	4900			Kernel:	4.4.114-94.11-default		
Nominal:	3600			Compiler:	C/C++: Version 18.0.2.199 of Intel C/C++ Compiler for Linux;		
Enabled:	8 cores, 1 chip			Fortran:	Version 18.0.2.199 of Intel Fortran Compiler for Linux		
Orderable:	1 chip			Parallel:	No		
Cache L1:	32 KB I + 32 KB D on chip per core			Firmware:	Version 1.0a released Sep-2018		
L2:	256 KB I+D on chip per core			File System:	xfs		
L3:	12 MB I+D on chip per chip			System State:	Run level 3 (multi-user)		
Other:	None			Base Pointers:	64-bit		
Memory:	64 GB (4 x 16 GB 2Rx8 PC4-2666V-E)			Peak Pointers:	64-bit		
Storage:	1 x 200 GB SATA III SSD			Other:	None		
Other:	None						

Benchmark	Base								Peak							
	Copies		Seconds		Ratio		Seconds		Ratio		Seconds		Ratio			
	Copies	Seconds	Ratio	Seconds	Ratio	Seconds	Ratio	Copies	Seconds	Ratio	Seconds	Ratio	Seconds	Ratio		
503.bwaves_r	8	1062	75.5	1069	75.1	1070	75.0	8	1062	75.5	1069	75.1	1070	75.0		
507.cactuBSSN_r	8	229	44.2	233	43.4	233	43.5	8	232	43.6	232	43.6	232	43.7		
508.namd_r	8	178	42.7	178	42.8	189	40.3	8	177	43.0	176	43.2	176	43.2		
510.parest_r	8	1012	20.7	1007	20.8	1026	20.4	8	1012	20.7	1007	20.8	1026	20.4		
511.povray_r	8	284	65.9	279	67.0	275	68.0	8	238	78.5	241	77.4	244	76.5		
519.ibm_r	8	468	18.0	470	17.9	470	18.0	8	469	18.0	469	18.0	470	18.0		
521.wrf_r	8	491	36.5	490	36.5	491	36.5	8	491	36.5	491	36.5	492	36.4		
526.blender_r	8	223	54.7	222	54.8	223	54.7	8	222	54.9	222	54.8	223	54.7		
527.cam4_r	8	263	53.1	262	53.4	261	53.7	8	263	53.1	262	53.4	261	53.7		
538.imagick_r	8	143	139	140	142	140	142	8	140	143	141	142	141	142		
544.nab_r	8	151	89.4	151	89.0	151	89.3	8	151	89.4	151	89.0	151	89.3		
549.fotonik3d_r	8	1361	22.9	1363	22.9	1363	22.9	8	1361	22.9	1363	22.9	1363	22.9		
554.roms_r	8	896	14.2	897	14.2	899	14.1	8	855	14.9	857	14.8	867	14.7		
SPECrate2017_fp_base																
SPECrate2017_fp_peak																

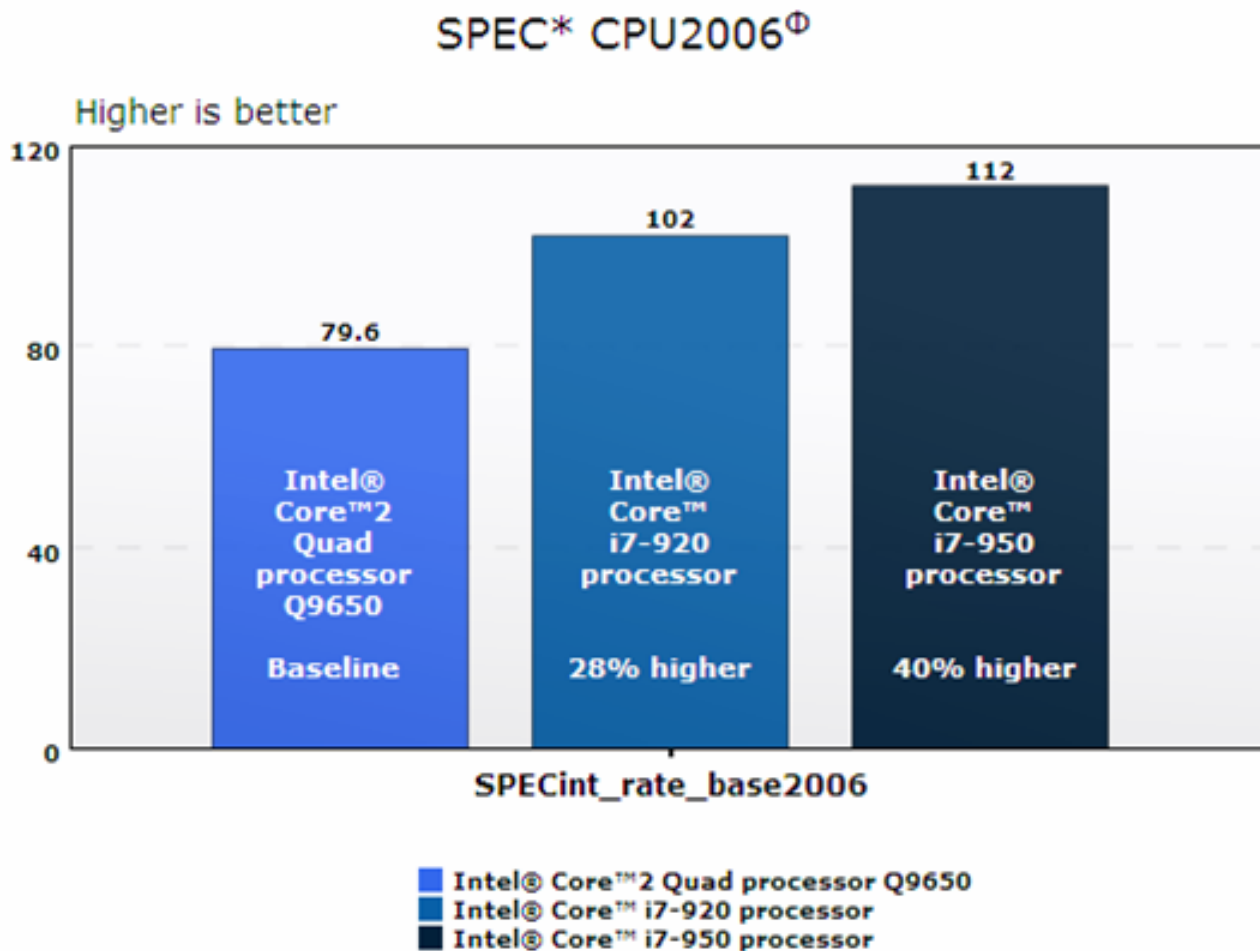
Results appear in the order in which they were run. Bold underlined text indicates a median measurement.

Primer primerjave zmogljivosti na osnovi SPEC2017:



<https://www.anandtech.com/show/14664/testing-intel-ice-lake-10nm/4>

Primer: Primerjava zmogljivosti Intelovih procesorjev



- Organizacija SPEC objavlja benchmark programe in rezultate testiranja ne samo za CPE, temveč tudi za:
 - Grafične in delovne postaje
 - Javanske odjemalce in strežnike
 - Visoko zmogljivo računanje (OpenMP, MPI)
 - Poštne strežnike
 - Mrežne datotečne sisteme
 - Merjenje porabe (pri velikem številu strežnikov)
 - SOA (Service Oriented Architectures)
 - SIP strežnike (Session Initiation Protocol)
 - Virtualizacijo
 - Web strežnike