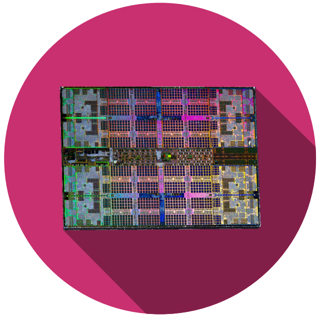
Computer Architecture

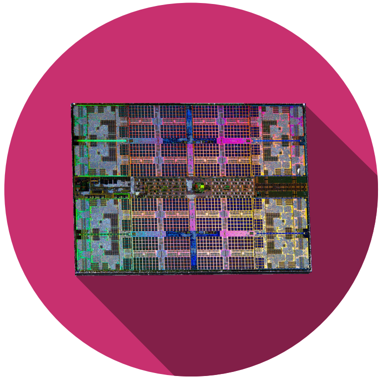
Performance

Per

Information and Communication Systems program

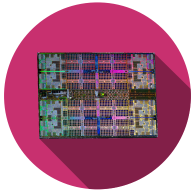


Silvan Zahno silvan.zahno@hevs.ch



Benchmarks

Benchmarks

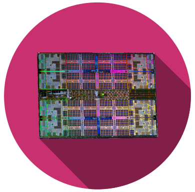


Judges the relative performance of a CPU

Key terms

- MIPS
- MFLOPS
- MHz
- FPS
- Render Time
- Dropped Frames

Benchmarks

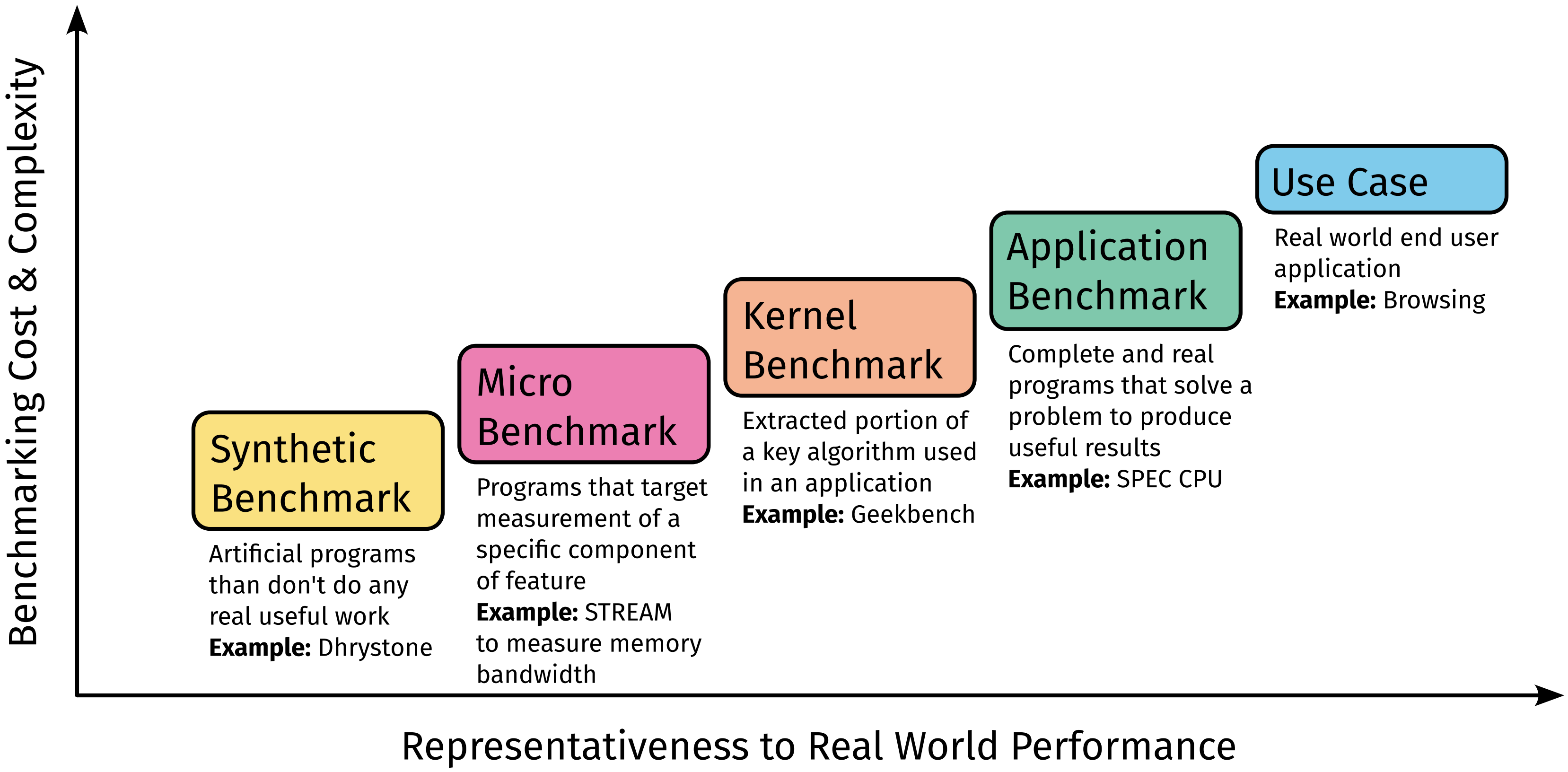


Judges the relative performance of a CPU

Key terms

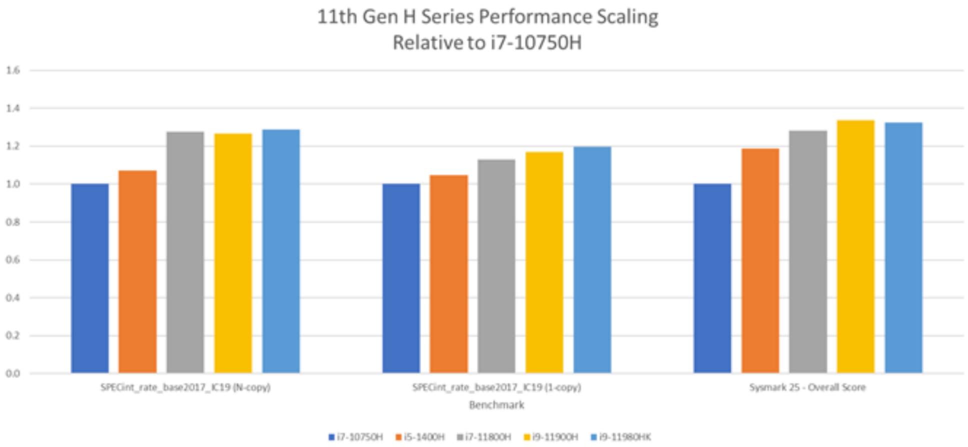
- MIPS – Million Instruction per Second
- MFLOPS – Million Floating Point Operation per Second
- MHz - Frequency
- FPS – Frame per second
- Render Time – Time to render 2D/3D scene
- Dropped Frames – number of frames lost in streaming

Spectrum of Benchmarks



SPEC Benchmark SPEC CPU 2017

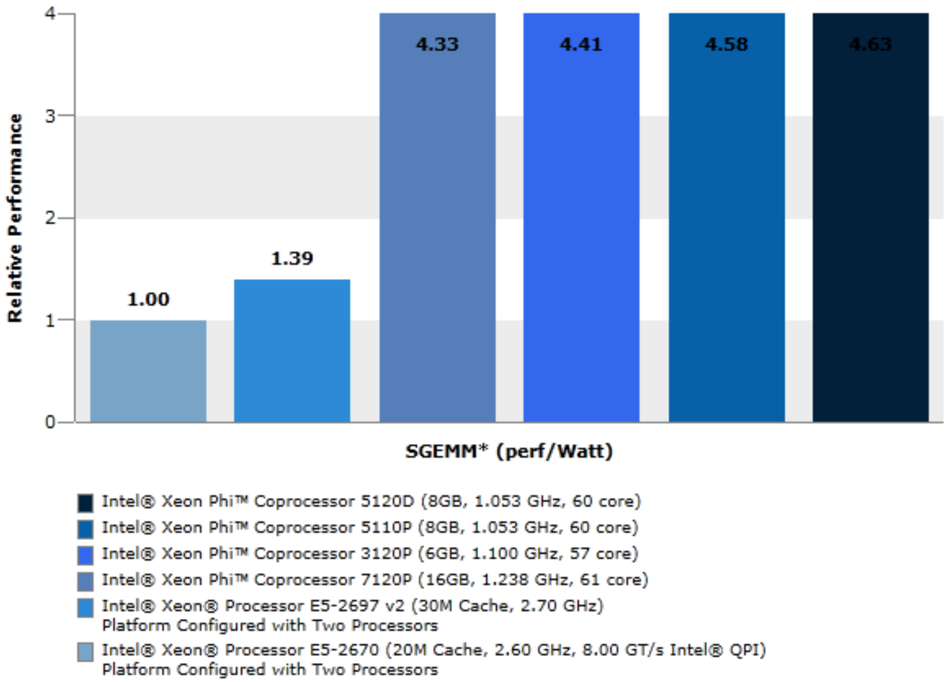
- **S**tandard **P**erformance **E**valuation **C**orporation
- Measures integer and floating point operation performance
- Contains 10 integer and 13 floating point operations
- Compute intensive, concentrates on CPU and memory



Linpack

Performance per Watt

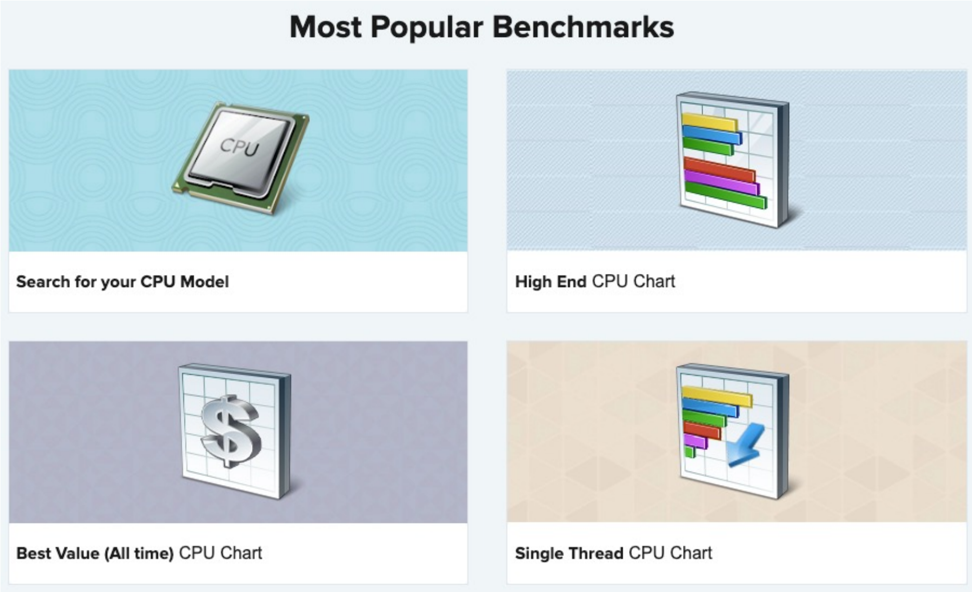
- Performances of double precision vector and matrix operations
- Ofen used by HPC communicate to measure FLOPS of a processor



PassMark

Multitude of Benchmarks

- CPU
  https://www.cpubenchmark.net
- Video Cards
  https://www.videocardbenchmark.net
- Hard Drive
  https://www.harddrivebenchmark.net
- Memory
  https://www.memorybenchmark.net
- PC
  https://www.pcbenchmarks.net
- Database
  https://www.databasebenchmarks.net



Aspects of Computer Performance

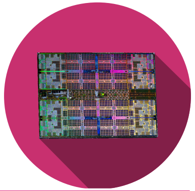
Time

- Response time / latency / execution time
  - Time between start and completion of event / task / program (n seconds)

Other Aspects

- Availability, scalability, performance per watt

Benchmark Suites



Benchmark	Organization	Description	Comment
Dhrystone	Open Source	Synthetic benchmark measure compute-intensice integer performance	No longer representative of modern workloads
CoreMark	EEMBC	Popular benchmark to measurecompute-intensive integer performance	Result published through EEMBS, continuation of Dhrystone
SPEC CPU	SPEC OSG CPU	Application benchmarks that comprise of open-source real world applications.	Most popular for measuring CPU performamce
STREAM	Open Source	Suite of microbenchmarks to measure sustained memory bandwidth	Measure of memory latency and bandwidth
Geekbench 5	Primate labs	Kernel benchmarks that measure CPU Integer, Floating Point and Memory performance	Used in industry, but with some limitation on the representation of real-world scenarios
PassMark	PassMark Software	Kernel benchmarks that measure CPU Integer, Floating Point and Memory performance Per	Used in industry, but with some limitation on the representation of real-world scenarios

SPEC Benchmarks

SPECrate®2017 Integer	SPECspeed®2017 Integer	Language[1]	KLOC[2]	Application Area
500.perlbench_r	600.perlbench_s	C	362	Perl interpreter
502.gcc_r	602.gcc_s	C	1,304	GNU C compiler
505.mcf_r	605.mcf_s	C	3	Route planning
520.omnetpp_r	620.omnetpp_s	C++	134	Discrete Event simulation - computer network
523.xalancbmk_r	623.xalancbmk_s	C++	520	XML to HTML conversion via XSLT
525.x264_r	625.x264_s	C	96	Video compression
531.deepsjeng_r	631.deepsjeng_s	C++	10	Artificial Intelligence: alpha-beta tree search (Chess)
541.leela_r	641.leela_s	C++	21	Artificial Intelligence: Monte Carlo tree search (Go)
548.exchange2_r	648.exchange2_s	Fortran	1	Artificial Intelligence: recursive solution generator (Sudoku)
557.xz_r	657.xz_s	C	33	General data compression

SPECrate®2017 Floating Point	SPECspeed®2017 Floating Point	Language[1]	KLOC[2]	Application Area
503.bwaves_r	603.bwaves_s	Fortran	1	Explosion modeling
507.cactuBSSN_r	607.cactuBSSN_s	C++, C, Fortran	257	Physics: relativity
508.namd_r		C++	8	Molecular dynamics
510.parest_r		C++	427	Biomedical imaging: optical tomography with finite elements
511.povray_r		C++, C	170	Ray tracing
519.lbm_r	619.lbm_s	C	1	Fluid dynamics
521.wrf_r	621.wrf_s	Fortran, C	991	Weather forecasting
526.blender_r		C++, C	1,577	3D rendering and animation
527.cam4_r	627.cam4_s	Fortran, C	407	Atmosphere modeling
	628.pop2_s	Fortran, C	338	Wide-scale ocean modeling (climate level)
538.imagick_r	638.imagick_s	C	259	Image manipulation
544.nab_r	644.nab_s	C	24	Molecular dynamics
549.fotonik3d_r	649.fotonik3d_s	Fortran	14	Computational Electromagnetics
554.roms_r	654.roms_s	Fortran	210	Regional ocean modeling

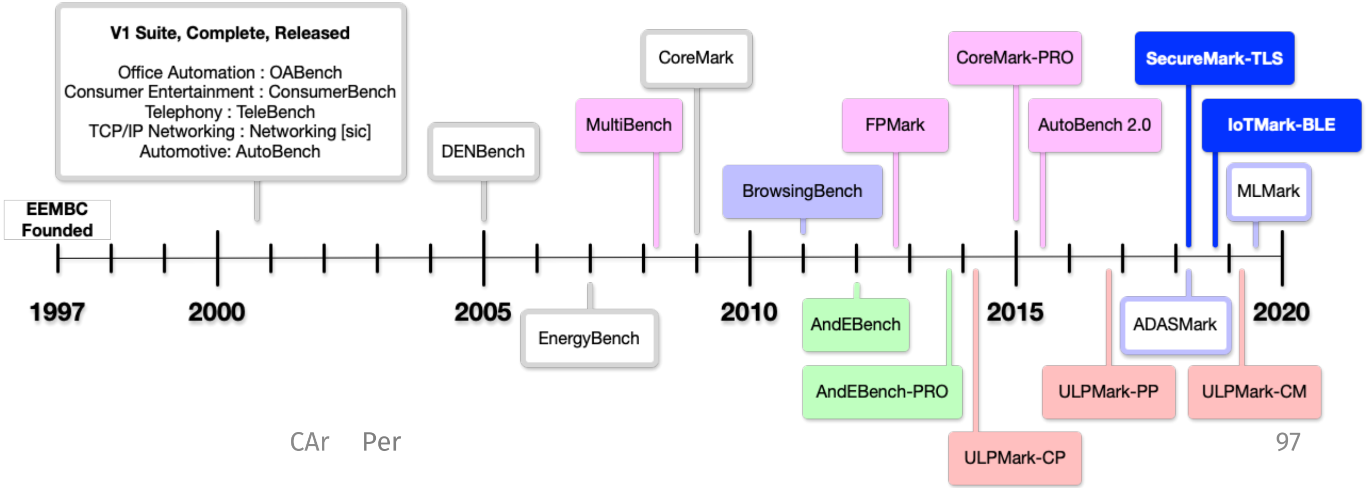
[1] For multi-language benchmarks, the first one listed determines library and link options (details)

[2] KLOC = line count (including comments/whitespace) for source files used in a build / 1000

EEMBC Benchmarks

Benchmarks can be grouped into the following categories:

- [Ultra-Low Power and Internet of Things]
- [Heterogeneous Compute]
- [Single-core Processor Performance]
- [Multi-core Processor Performance]
- [Phone and Tablet]



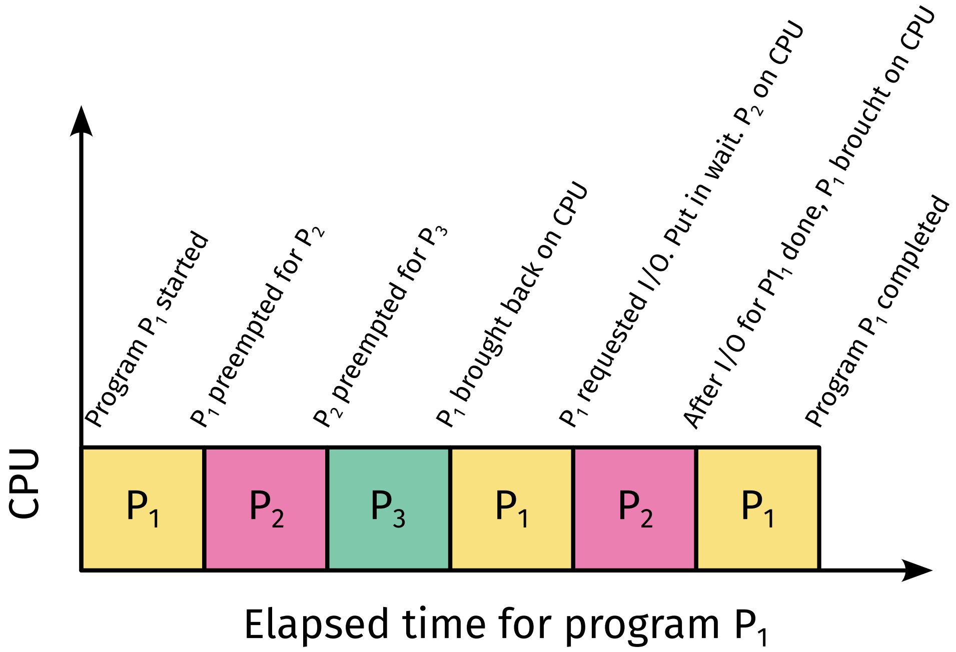
How to Measure Execution time?

Wall-clock time

- Elapsed time
    - Disk Access, I/O, OS overhead, …

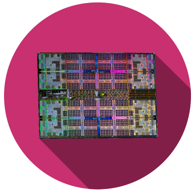
Multiprogrammed CPI works on other process when current process stalled for I/O

CPU-time = time CPU is computing



Benchmark

Exercise

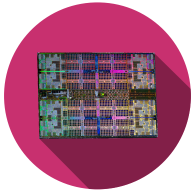


Which of the following statements are correct?

1. The wall clock time is the total elapsed time , including I/O, Operating system overhead etc.?
2. Multi-threading improves the throughput of a process
3. The CPU time does not include the I/O time
4. Multi-threading improves the execution time of a process

Benchmark

Exercise



Which of the following statements are correct?

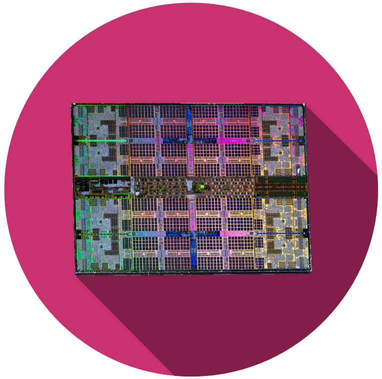
1. The wall clock time is the total elapsed time , including I/O, Operating system overhead etc.?
2. Multi-threading improves the throughput of a process
3. The CPU time does not include the I/O time
4. Multi-threading improves the execution time of a process

Benchmark

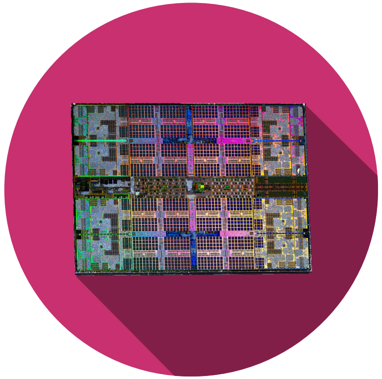
Quick and Dirty

For quickly benchmark your application or functions you can use hyperfine. A rust base CLI Program

```
▶ hyperfine --warmup 3 'fd -e jpg -uu' 'find -iname "*.jpg"'
```



Mini-Labo
“Debug”



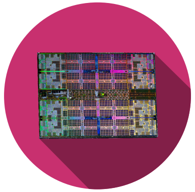
Performance

Performance

- With 1 program it is clear which computer is faster
- With >1 programs, it depends

How do you aggregate their performance?

Performance



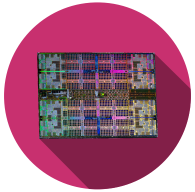
- With 1 program it is clear which computer is faster
- With >1 programs, it depends

How do you aggregate their performance?

$$ArithmeticMean = AM = \frac{1}{n}\sum_{i=1}^{n} Time_i$$

Performance

Exercise

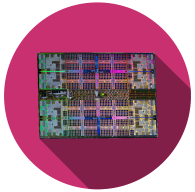


	CPU A	CPU B	CPU C
Program P_1 (sec)	1	10	20
Program P_2 (sec)	1000	100	20

1. Which CPU is the fastest for P_1 ?
2. Which CPU is the fastest for P_2 ?
3. Which CPU is the fastest?

Performance

Exercise

	CPU A	CPU B	CPU C
Program P_1 (sec)	1	10	20
Program P_2 (sec)	1000	100	20
Total (sec)			
Arithmetic Mean			

1. Which CPU is the fastest for P_1 ?
2. Which CPU is the fastest for P_2 ?
3. Which CPU is the fastest ?

Performance

- What if the program not run equally often?
- Two approaches:
    - Weighted execution time
    - Normalize execution times to a reference machine and take the average

Performance

Weighted Execution Time

- Program P_i has a weight $Weight_i$
  - Indicates execution frequency
  - weights add up to 1
- Program P_i takes time $Time_i$
- Weighted arithmetic mean:

$$WeightedArithmeticMean = WAM = \sum_{i=1}^{n} Weight_i * Time_i$$

Performance

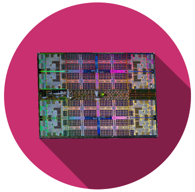
Weighted Execution Time - Example

	CPU A	CPU B	CPU C
Program P_1 (sec)	1	10	20
Program P_2 (sec)	1000	100	20
Arithmetic mean	500.5	55	20

1. Which CPU is the fastest with w_1=0.8 and w_2 = 0.2
2. Which CPU is the fastest with w_1=0.9 and w_2 = 0.1

Performance

Weighted Execution Time - Example

	CPU A	CPU B	CPU C
Program P_1 (sec)	1	10	20
Program P_2 (sec)	1000	100	20
Arithmetic mean	500.5	55	20
Weighted mean (w_1=0.8, w_2=0.2)			
Weighted mean (w_1=0.9, w_2=0.1)			

1. Which CPU is the fastest with w_1=0.8 and w_2 = 0.2

2. Which CPU is the fastest with w_1=0.9 and w_2 = 0.1

Performance

Weighted Execution Time - Exercise

We want to buy a new computer. It will mostly run programs P_1 and P_2.

1. When is CPU A the best buy?
2. When is CPU B the best buy?
3. When is CPU C the best buy?

	CPU A	CPU B	CPU C
Program P_1 (sec)	1	10	100
Program P_2 (sec)	100	10	1

Performance

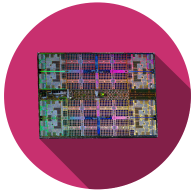
Normalized Execution Time

- Normalize the execution time to a reference computer

$$ExecutionTimeRatio = \frac{ExecutionTime_{reference}}{ExecutionTime_{new}}$$

- Used by SPEC Benchmarks, called SPECRatio

Performance



- Normalized Execution Time - Example

	CPU A	CPU B	CPU C
Program P_1 (sec)	1	10	20
Program P_2 (sec)	1000	100	20

	Normalized to A			Normalized to B		
	CPU A	CPU B	CPU C	CPU A	CPU B	CPU C
ETR P_1						
ETR P_2						

Performance

Normalized Execution Time - Example

- Do NOT use arithmetic mean to average normalized execution times!!

	CPU A	CPU B	CPU C
Program P_1 (sec)	1	10	20
Program P_2 (sec)	1000	100	20

	Normalized to A			Normalized to B		
	CPU A	CPU B	CPU C	CPU A	CPU B	CPU C
ETR P_1	1	0.1	0.05	10	1	0.5
ETR P_2	1	10	50	0.1	1	5.0
~~Arithmetic mean~~	~~1~~	~~5.05~~	~~**25.025**~~	~~**5.05**~~	~~1~~	~~2.75~~

Performance

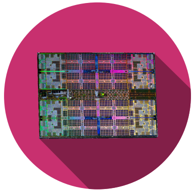
Geometric Mean

- Independent of reference
- No physical meaning

$$Geometric\ Mean = GM = \sqrt[n]{\prod_{i=1}^{n} ExecutionTimeRatio_i}$$

Performance

Geometric Mean - Example



$$GM = \sqrt[n]{\prod_{i=1}^{n} ExecutionTimeRatio_i}$$

	CPU A	CPU B	CPU C
Program P_1 (sec)	1	10	20
Program P_2 (sec)	1000	100	20

	Normalized to A			Normalized to B		
	CPU A	CPU B	CPU C	CPU A	CPU B	CPU C
ETR P_1	1	0.1	0.05	10	1	0.5
ETR P_2	1	10	50	0.1	1	5.0
Geometric mean						

CPU Performance Equation

$$T = IC * CPI * CT = \frac{IC * CPI}{f}$$

- T = *Execution* ***t****ime*
- $IC = N_{instr}$ = # Instructions executed (**I**nstruction **C**ount)
- CPI = **C**ycles **P**er **I**nstruction
- $CT = t_{cycle}$ = **C**ycle **T**ime = duration of clock cycle
- f = clock frequency = $\frac{1}{t_{cycle}}$

CPU Performance Equation

Improving Performance

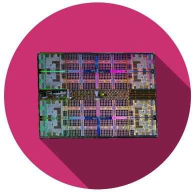
$$T = IC * CPI * CT = \frac{IC * CPI}{f}$$

- Factors are interdependent => Trade-offs
  - Cost
  - Power
  - Performance

Reduce	How	Side effect
IC / N_{instr}	Increase capability / complexity of instructions	CPI / t_{cycle} increases
CPI	Simple instructions required fewer cycles	IC / N_{instr} increases
CT / t_{cycle}	Less work per cycle	CPI increases

CPU Performance Equation

Calculating Execution Time



A program executes 5 Million instruction with a given instruction mix:

Instruction	Frequency	CPI_{instr}
ALU	50%	3
Load	20%	5
Store	10%	4
Branch	20%	3

- The CPU has a frequency of 2GHz

1. What is the execution time?

CPU Performance Equation

Calculating Execution Time

A program executes 5 Million instruction with a given instruction mix:

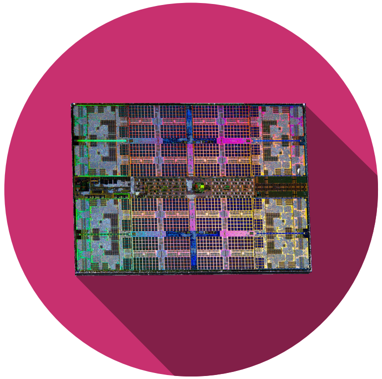
Instruction	Frequency	CPI_{instr}
ALU	50%	3
Load	20%	5
Store	10%	4
Branch	20%	3

$CPI_{avg} =$

$T =$

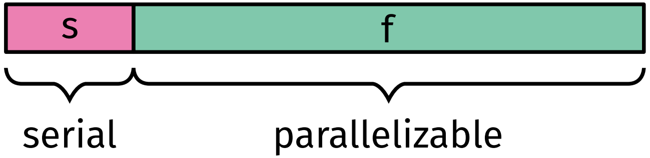
- The CPU has a frequency of 2GHz

1. What is the execution time?



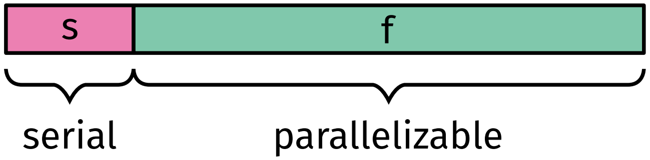
Amdahls Law

Amdahl’s Law



1 core

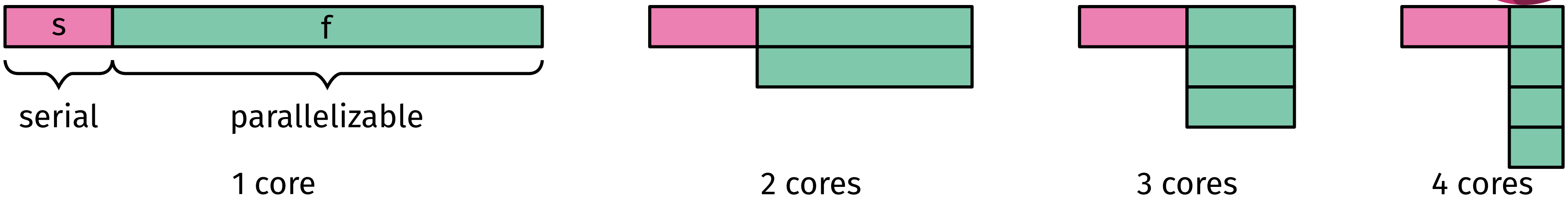
Amdahl’s Law



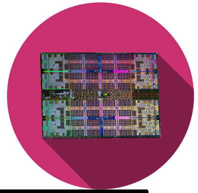
1 core

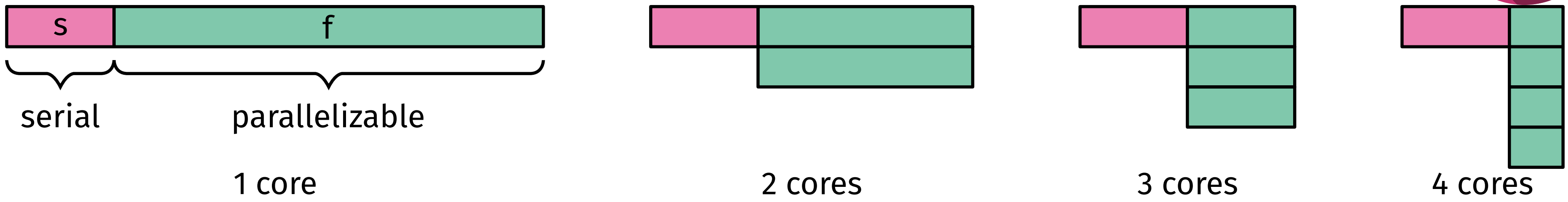
2 cores

Amdahl’s Law



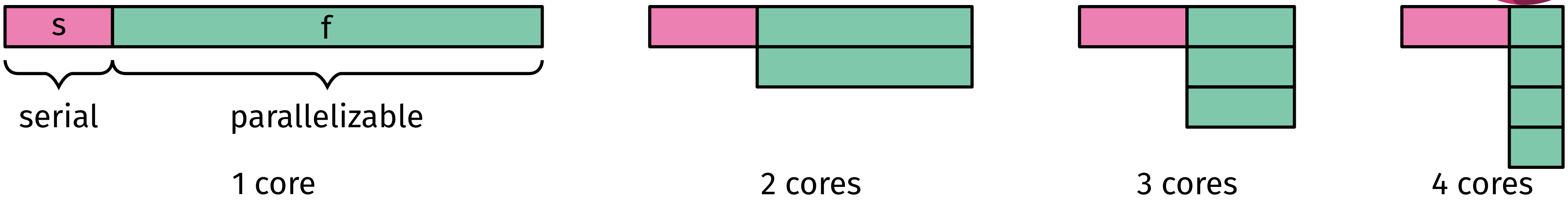
Amdahl’s Law

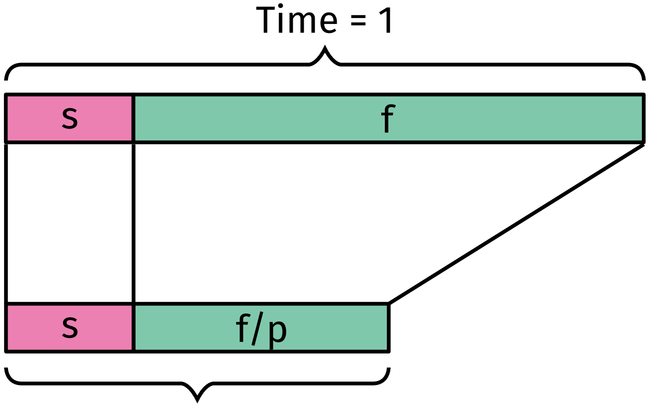




If 50% of the execution time is sequential, the maximum speedup is 2, no matter how many cores you use.
- Gene Amdahl -

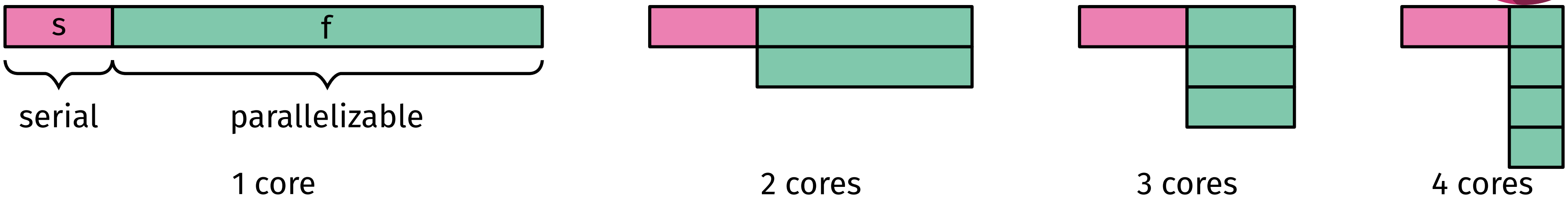
Amdahl’s Law

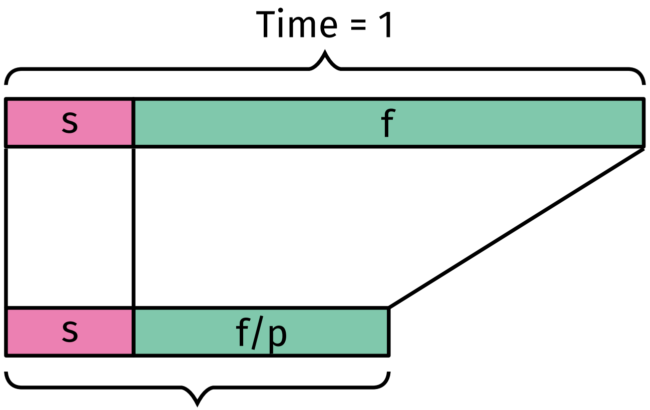




$$Time = s + \frac{f}{p}$$

Amdahl's Law





$$Time = s + \frac{f}{p}$$

$$S = \frac{1}{s + \frac{f}{p}} = \frac{1}{1 - f + \frac{f}{p}}$$

S = Speedup
$s = 1 - f = \mathit{serial\ fraction}$
$f = 1 - s =$ parallel fraction
$p = \#\ \mathit{processor\ cores}$

Amdahl’s Law

Speedup for Different Parallel Fractions

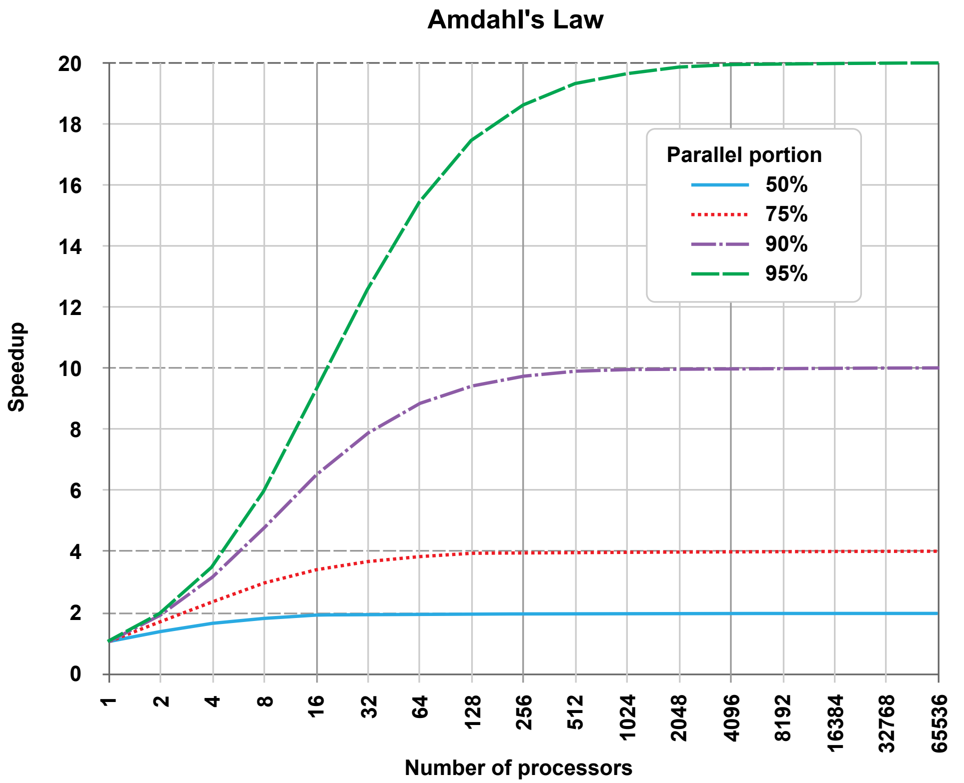
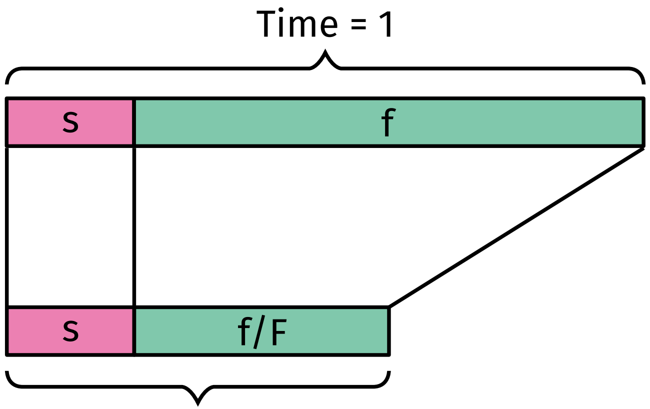


Image Credit: Wikipedia

Amdahl's Law

Generalization of Amdahl's Law

- Suppose we can improve fraction f by factor F
- What will be overall improvement?



$$Time = s + \frac{f}{F}$$

$$S = \frac{1}{s + \frac{f}{F}} = \frac{1}{1 - f + \frac{f}{F}}$$

S = Speedup
$s = 1 - f$ = fraction we cannot improve
$f = 1 - s$ = fraction we can improve
$F = improvement\ factor$

Amdahl’s Law

Example

FP instruction improved to run 2x as fast, but only 10% of all executed instructions are FP.

- *What will be the overall speedup?*

References

[1]
“11th Gen H Series: Benchmarks Across the Stack,” *Intel*. https://www.intel.com/content/www/us/en/support/articles/000059752/processors.html (accessed Jul. 21, 2022).

[2]
“Amdahl’s law,” *Wikipedia*. Jun. 29, 2022. Accessed: Jul. 18, 2022. [Online]. Available: https://en.wikipedia.org/w/index.php?title=Amdahl%27s_law&oldid=1095605183

[3]
ARM Ltd., “An Introduction to CPU Performance Benchmarks and How This Applies to the Home Market.” 2021.

[4]
S. H. Fuller and L. I. Millett, “Computing Performance: Game Over or Next Level?,” *Computer*, vol. 44, no. 1, pp. 31–38, Jan. 2011, doi: 10.1109/MC.2011.15.

[5]
“EEMBC.” https://www.eembc.org/products/ (accessed Feb. 23, 2023).

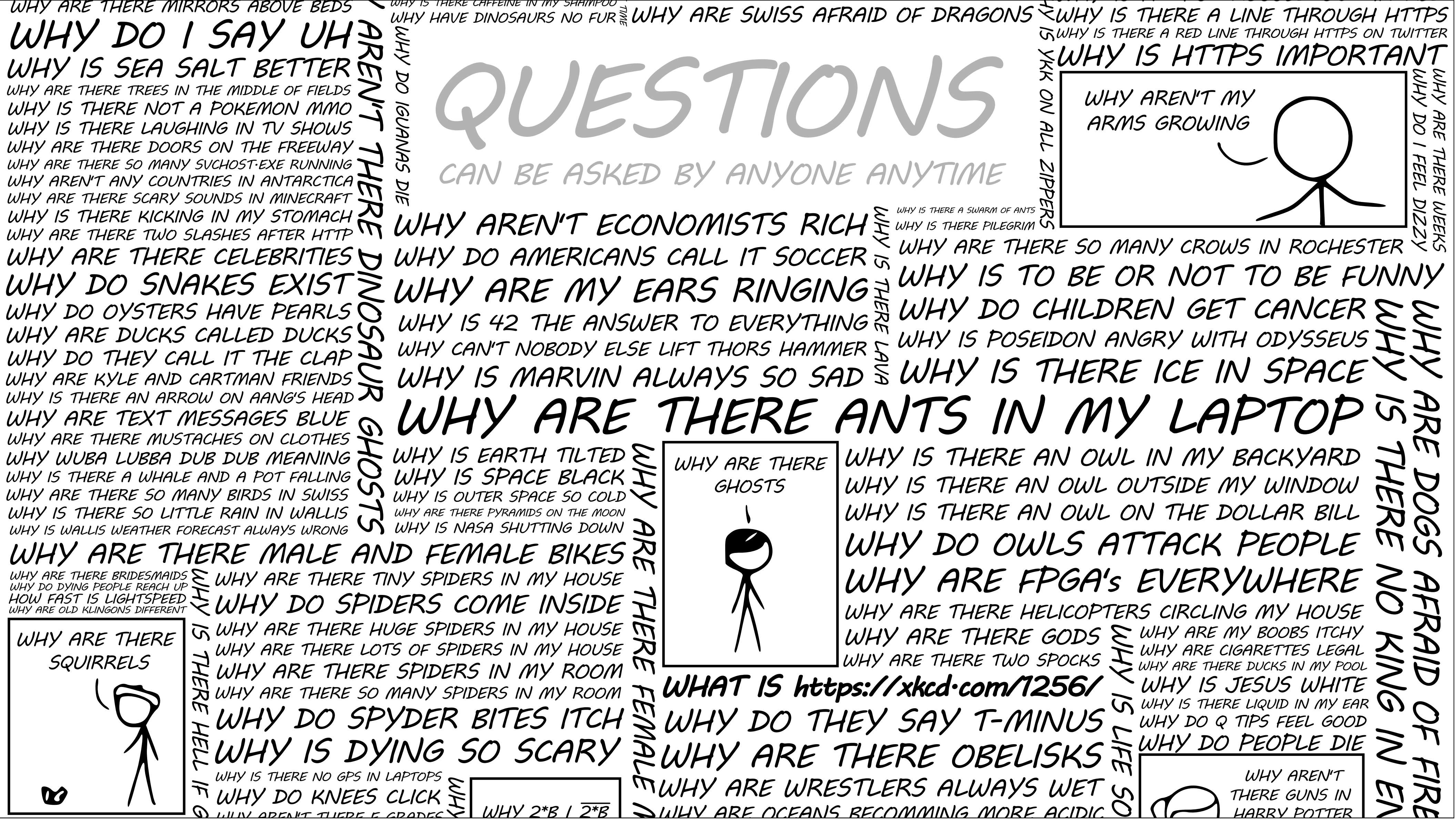
[6]
C. benchmarks are crucial in determining whether your processor can run the games and applications you like H. what you need to get started.1, “How to Read and Understand CPU Benchmarks,” *Intel*. https://www.intel.com/content/www/us/en/gaming/resources/read-cpu-benchmarks.html (accessed Jul. 21, 2022).

[7]
“linpack performance per wat at DuckDuckGo.” https://duckduckgo.com/?q=linpack+performance+per+wat&t=osx&iar=images&iax=images&ia=images&iai=http%3A%2F%2Fwww.intel.com%2Fcontent%2Fdam%2Fwww%2Fpublic%2Fus%2Fen%2Fimages%2Fcharts%2Fchart-id-642.png (accessed Jul. 21, 2022).

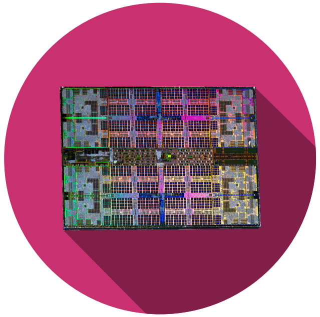
[8]
“Overview - CPU 2017.” https://www.spec.org/cpu2017/Docs/overview.html#benchmarks (accessed Feb. 23, 2023).

[9]
“SPEC CPU2017 Results.” https://www.spec.org/cpu2017/results/ (accessed Jul. 21, 2022).

[10]
TU Berlin: Course AES, (Oct. 22, 2018). Accessed: Jul. 18, 2022. [Online Video]. Available: https://www.youtube.com/channel/UCPSsA8oxlSBjidJsSPdpjsQ


QUESTIONS
CAN BE ASKED BY ANYONE ANYTIME
WHY ARE THERE MIRRORS ABOVE BEDS
WHY DO I SAY UH
WHY IS SEA SALT BETTER
WHY ARE THERE TREES IN THE MIDDLE OF FIELDS
WHY IS THERE NOT A POKEMON MMO
WHY IS THERE LAUGHING IN TV SHOWS
WHY ARE THERE DOORS ON THE FREEWAY
WHY ARE THERE SO MANY SVCHOST.EXE RUNNING
WHY AREN'T ANY COUNTRIES IN ANTARCTICA
WHY ARE THERE SCARY SOUNDS IN MINECRAFT
WHY IS THERE KICKING IN MY STOMACH
WHY ARE THERE TWO SLASHES AFTER HTTP
WHY ARE THERE CELEBRITIES
WHY DO SNAKES EXIST
WHY DO OYSTERS HAVE PEARLS
WHY ARE DUCKS CALLED DUCKS
WHY DO THEY CALL IT THE CLAP
WHY ARE KYLE AND CARTMAN FRIENDS
WHY IS THERE AN ARROW ON AANG'S HEAD
WHY ARE TEXT MESSAGES BLUE
WHY ARE THERE MUSTACHES ON CLOTHES
WHY WUBA LUBBA DUB DUB MEANING
WHY IS THERE A WHALE AND A POT FALLING
WHY ARE THERE SO MANY BIRDS IN SWISS
WHY IS THERE SO LITTLE RAIN IN WALLIS
WHY IS WALLIS WEATHER FORECAST ALWAYS WRONG
WHY AREN'T THERE DINOSAUR GHOSTS
WHY IS THERE CAFFEINE IN MY SHAMPOO
WHY HAVE DINOSAURS NO FUR
TIME
WHY DO IGUANAS DIE
WHY ARE SWISS AFRAID OF DRAGONS
WHY IS YKK ON ALL ZIPPERS
WHY IS THERE A LINE THROUGH HTTPS
WHY IS THERE A RED LINE THROUGH HTTPS ON TWITTER
WHY IS HTTPS IMPORTANT
WHY AREN'T MY ARMS GROWING
WHY ARE THERE WEEKS
WHY DO I FEEL DIZZY
WHY AREN'T ECONOMISTS RICH
WHY DO AMERICANS CALL IT SOCCER
WHY ARE MY EARS RINGING
WHY IS 42 THE ANSWER TO EVERYTHING
WHY CAN'T NOBODY ELSE LIFT THORS HAMMER
WHY IS MARVIN ALWAYS SO SAD
WHY IS THERE LAVA
WHY IS THERE A SWARM OF ANTS
WHY IS THERE PILEGRIM
WHY ARE THERE SO MANY CROWS IN ROCHESTER
WHY IS TO BE OR NOT TO BE FUNNY
WHY DO CHILDREN GET CANCER
WHY IS POSEIDON ANGRY WITH ODYSSEUS
WHY IS THERE ICE IN SPACE
WHY ARE THERE ANTS IN MY LAPTOP
WHY IS EARTH TILTED
WHY IS SPACE BLACK
WHY IS OUTER SPACE SO COLD
WHY ARE THERE PYRAMIDS ON THE MOON
WHY IS NASA SHUTTING DOWN
WHY ARE THERE MALE AND FEMALE BIKES
WHY ARE THERE FEMALE
WHY ARE THERE GHOSTS
WHY IS THERE AN OWL IN MY BACKYARD
WHY IS THERE AN OWL OUTSIDE MY WINDOW
WHY IS THERE AN OWL ON THE DOLLAR BILL
WHY DO OWLS ATTACK PEOPLE
WHY ARE FPGA's EVERYWHERE
WHY ARE THERE HELICOPTERS CIRCLING MY HOUSE
WHY ARE THERE GODS
WHY ARE THERE TWO SPOCKS
WHY IS THERE NO KING IN EN
WHY ARE DOGS AFRAID OF FIRE
WHY ARE THERE BRIDESMAIDS
WHY DO DYING PEOPLE REACH UP
HOW FAST IS LIGHTSPEED
WHY ARE OLD KLINGONS DIFFERENT
WHY IS THERE HELL IF G
WHY ARE THERE TINY SPIDERS IN MY HOUSE
WHY DO SPIDERS COME INSIDE
WHY ARE THERE HUGE SPIDERS IN MY HOUSE
WHY ARE THERE LOTS OF SPIDERS IN MY HOUSE
WHY ARE THERE SPIDERS IN MY ROOM
WHY ARE THERE SO MANY SPIDERS IN MY ROOM
WHY DO SPYDER BITES ITCH
WHY IS DYING SO SCARY
WHY IS THERE NO GPS IN LAPTOPS
WHY DO KNEES CLICK
WHY
WHY ARE THERE SQUIRRELS
WHY 2*B | 2*B
WHAT IS https://xkcd.com/1256/
WHY DO THEY SAY T-MINUS
WHY ARE THERE OBELISKS
WHY ARE WRESTLERS ALWAYS WET
WHY ARE OCEANS BECOMMING MORE ACIDIC
WHY IS LIFE SO
WHY ARE MY BOOBS ITCHY
WHY ARE CIGARETTES LEGAL
WHY ARE THERE DUCKS IN MY POOL
WHY IS JESUS WHITE
WHY IS THERE LIQUID IN MY EAR
WHY DO Q TIPS FEEL GOOD
WHY DO PEOPLE DIE
WHY AREN'T THERE GUNS IN HARRY POTTER

Haute Ecole d'Ingénierie
Hochschule für Ingenieurwissenschaften



Silvan Zahno silvan.zahno@hevs.ch