CASPER

CS683: Advanced Computer Architecture

Introduction

https://www.cse.iitb.ac.in/~biswa/courses/CS683/main.html

https://www.cse.iitb.ac.in/~biswa/

Phones (smart/non-smart) on silence plz, Thanks

Who Am I?

Biswa

Research Group: CASPER (https://casper-iitb.github.io/)

Research interests: Architecture performance/security

Architecture-compiler/OS/network interactions ☺

Office hours (Join me over lunch or dinner with your doubts)

Where: CC 217

When: An email away

Email: [CS683] in the subject line

I will be happy

If you just call me Biswa ☺

However, I wont be unhappy if you wont ☺

TAs

- Sumon (MS)
- Harikrishna (Mtech)
- Kalind (Mtech)

Assessment Policies (All are group based except… quizzes)

Two Quizzes: The best (1 X 30) = 30 points

Programming assignments: (2 X 15) = 30 points

Group Projects: 30 points

Scribes (lecture summary)
10 points

Quizzes: Sept./Oct.

Project is your end-term

If you do well in Sept. Then no need of quiz in Oct.

Bonus points

- +5 best assignment
- +5 best project
- +5 best scribe
- +5 Memes ☺

Memes: Oh yes,
+5 for creative memes/videos
on computer architecture topics

Late Submission

-1 per day, after 10 days, zero points

Slow-learners, Take your time, submit only one assignment, write only one exam, Maximum grade: B

Coffee credits

- Create a group of three members:
- Answer/ask a pertinent question
- 10 coffee points/group = 1 day extension in programming assignments
- Real coffee with Biswa if you do not want to redeem your points.

Do not like coffee, Chai is there ☺

- Is it for real? Some pics from previous sem.

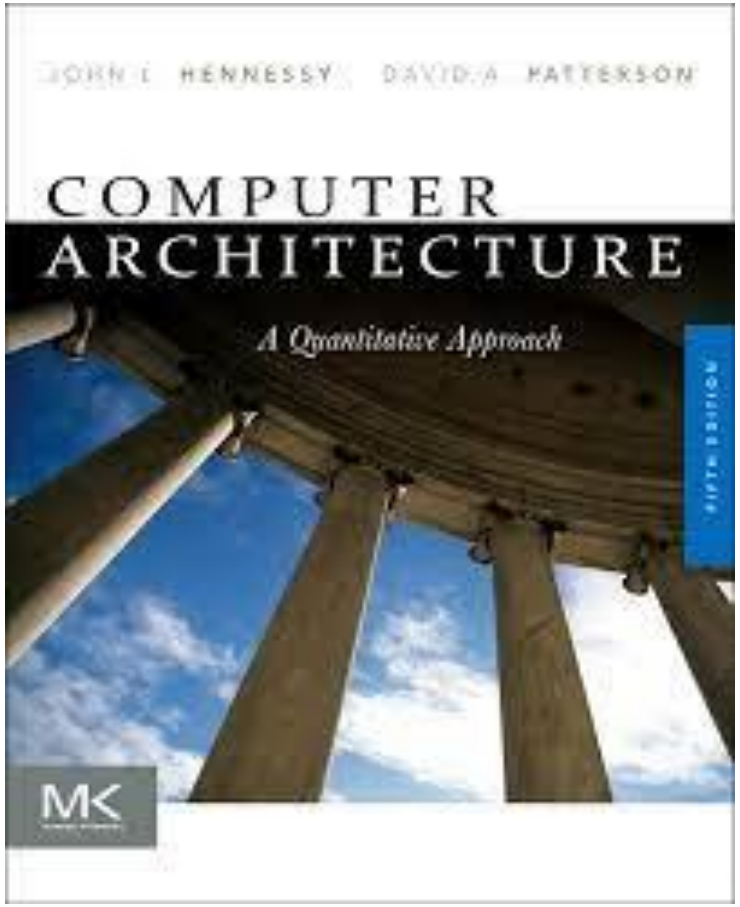
Projects

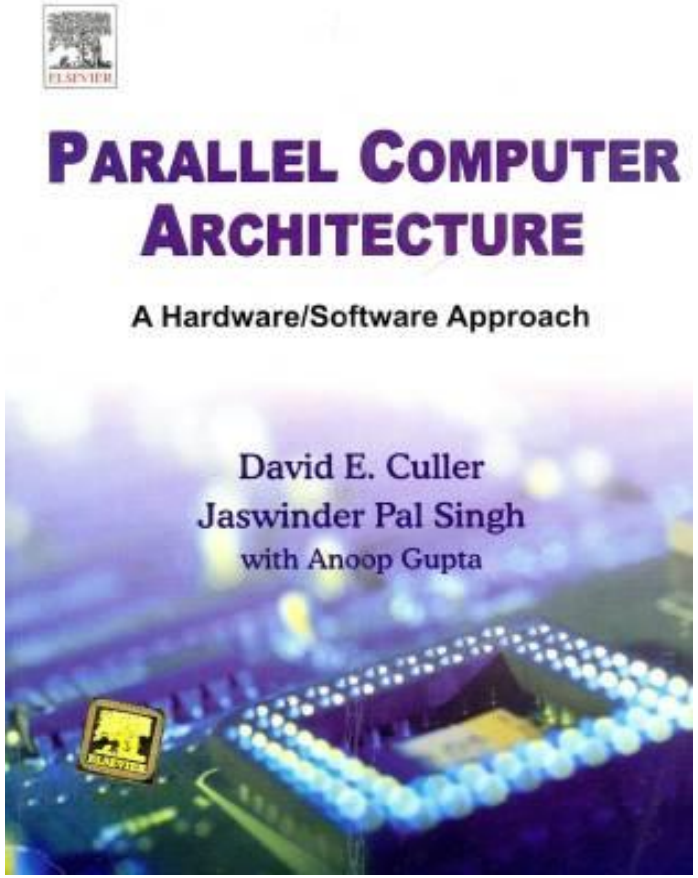
Post quizzes and assignments

Topics: you can choose, or Biswa can provide

Grading based on learning

Books


JOHN L. HENNESSY
DAVID A. PATTERSON
COMPUTER ARCHITECTURE
A Quantitative Approach
FIFTH EDITION
MK


PARALLEL COMPUTER ARCHITECTURE
A Hardware/Software Approach
David E. Culler
Jaswinder Pal Singh
with Anoop Gupta

Research Papers (Top-tier conferences only)

- ISCA
- MICRO
- HPCA
- ASPLOS
- PACT
- USENIX Security (For security)
- S&P (For security)

Attendance Policy (Either attend/!attend)

Do not make it fuzzy, do come on time else ditch it

CS683: NO

Feel free to come (!come), I would be happy if you attend all.

but I can understand, lectures can be boring ☹ too, do mention it whenever you feel like

Course means grades and grades are….
So we will take care, you take care of learning and earn your grades

Slides won't contain all

You wont be able to read and understand all ☹

Attend lectures, take notes, ask questions,

Feel free to pause me if I go fast/slow/boring/engaging

Lecture style

- Highly informal and interactive
- So, shout if you do not get any
- Do not assume you will get it later ☺
- Right time is NOW
- Get fully engaged 101% ☺
- My slides!= Me in class

CS683 (PG course)

Let’s have a dialogue and not monologue

Questions on Assessment or any…

Join Piazza ASAP (course page)

All notifications on Piazza ONLY

Moodle: For Assignment submissions

Academic dishonesty

Plz go through it:

https://www.iitb.ac.in/newacadhome/procedures201521July.pdf

https://www.iitb.ac.in/newacadhome/punishments201521July.pdf

Pre-req

- You should be aware of vanilla baby 5-stage pipeline and baby caches ☺
- This course will start from where you left your babies

CS683: Welcome

Digital Computers everywhere

All THE major Software Companies are now …

https://www.ai-startups.org/top/hardware/

NEWS > COMPANY NEWS

Facebook Is Reportedly Building its Own Chip

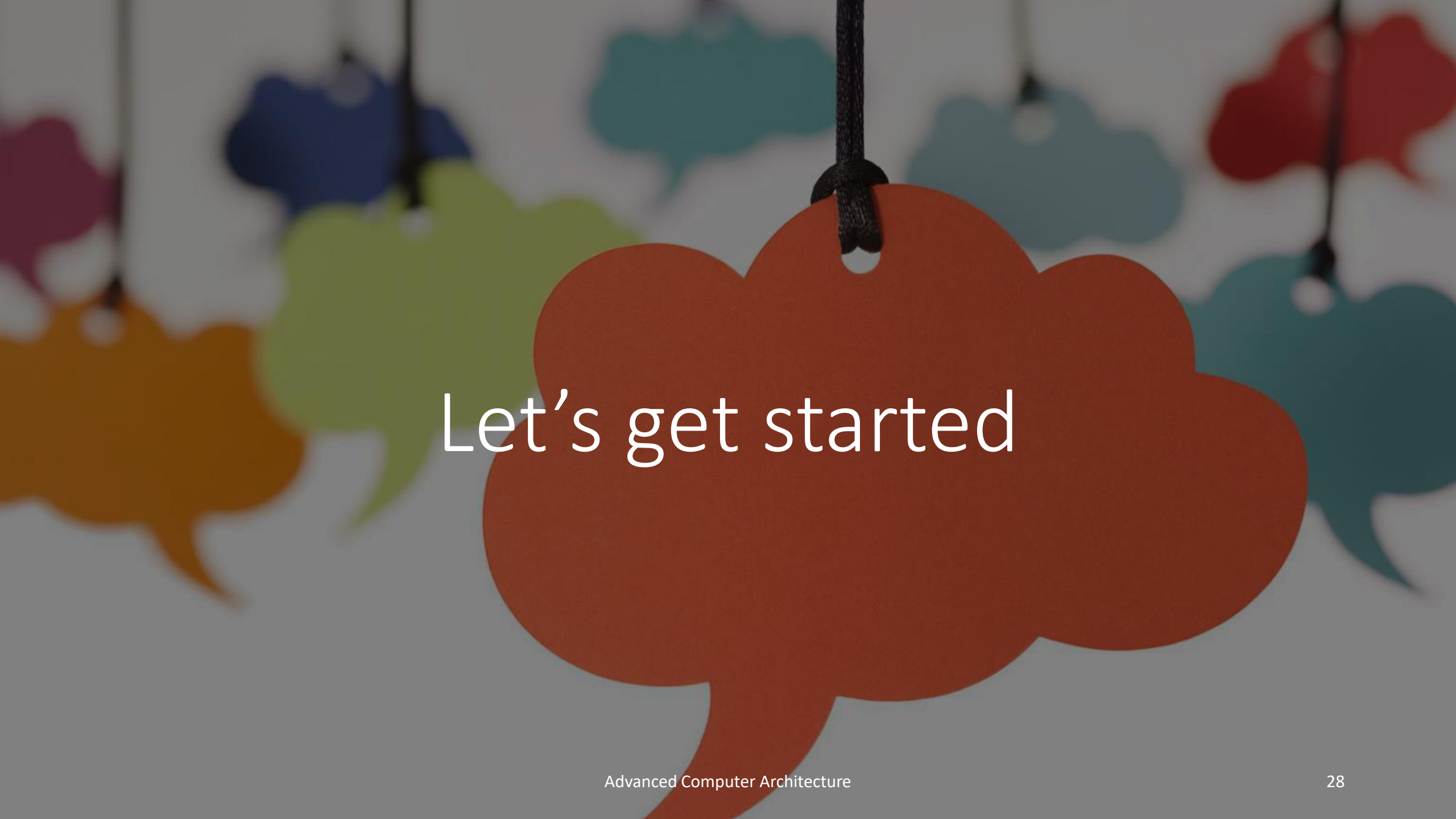
November 17, 2020

Meet the Microsoft Pluton processor – The security chip designed for the future of Windows PCs

GOOGLE MOBILE TECH

Google is reportedly building its own processor for Pixels and Chromebooks

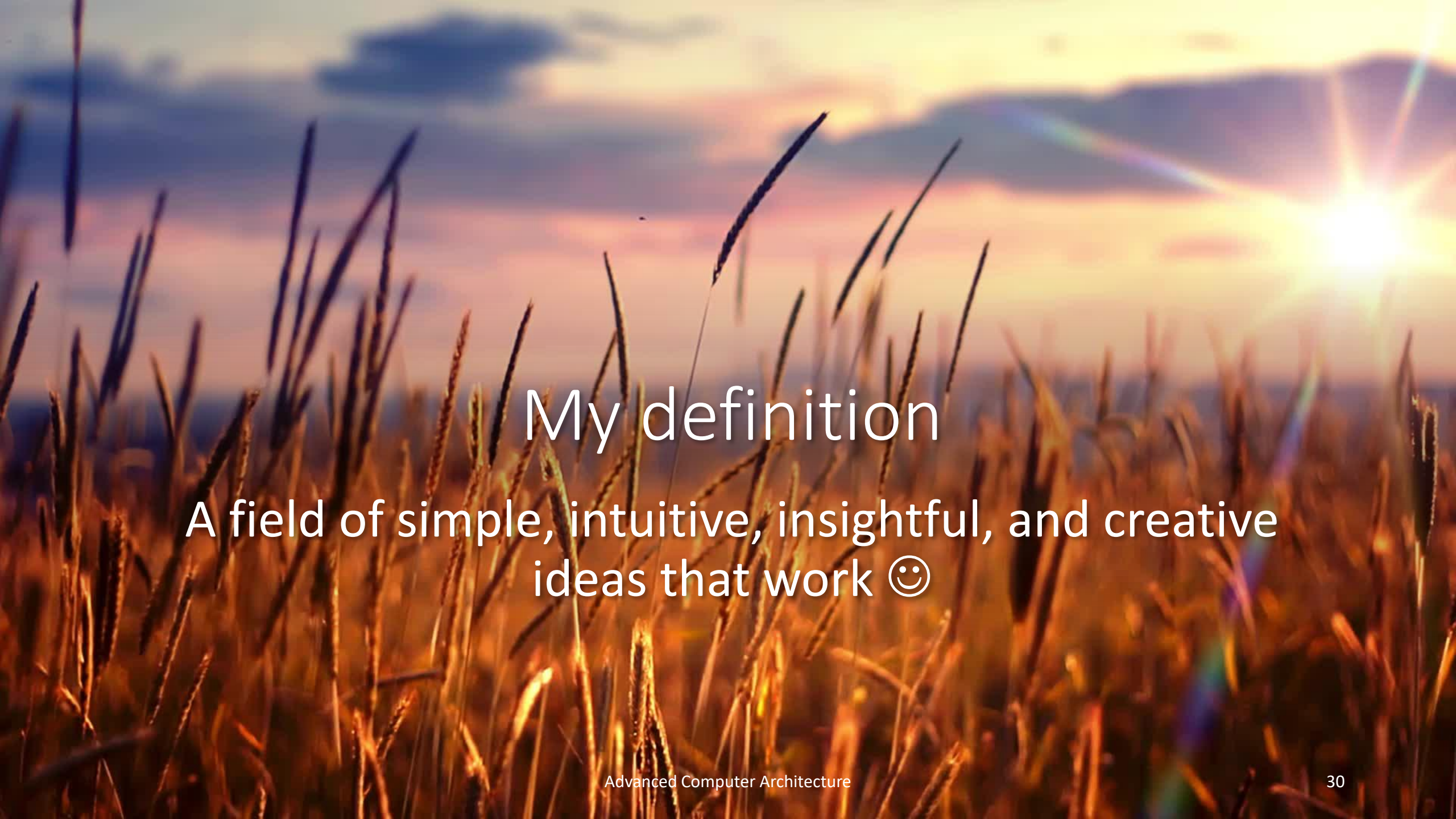
It could be used in Pixels as early as next year



Let’s get started

What is Computer Architecture

- No text-book definitions please ☺

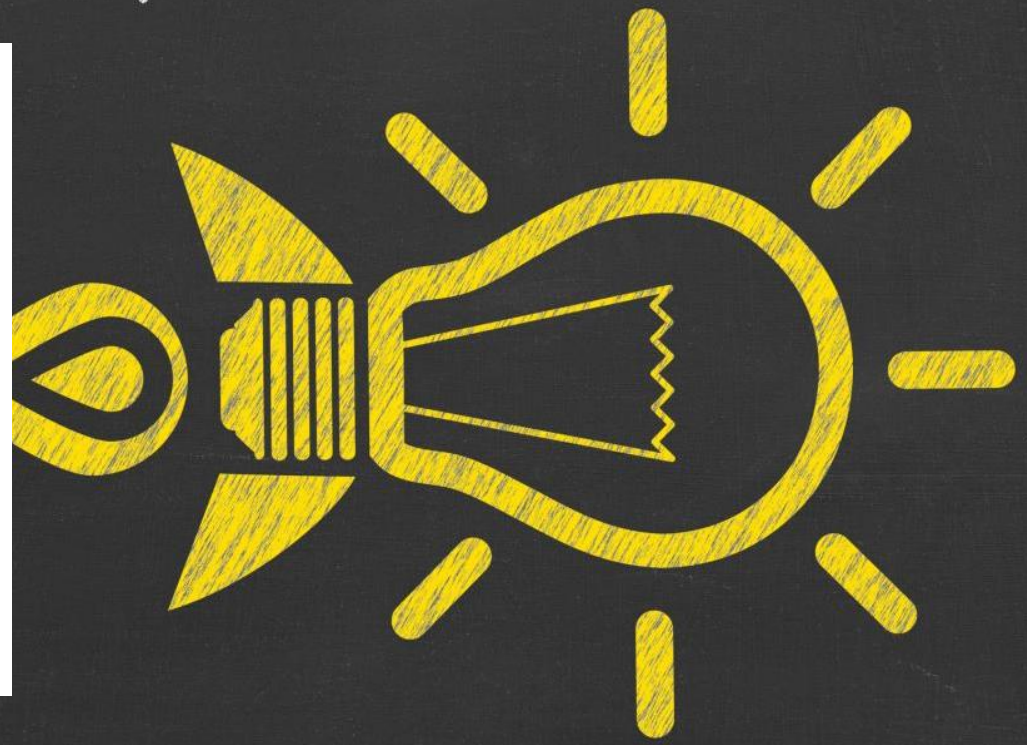


My definition

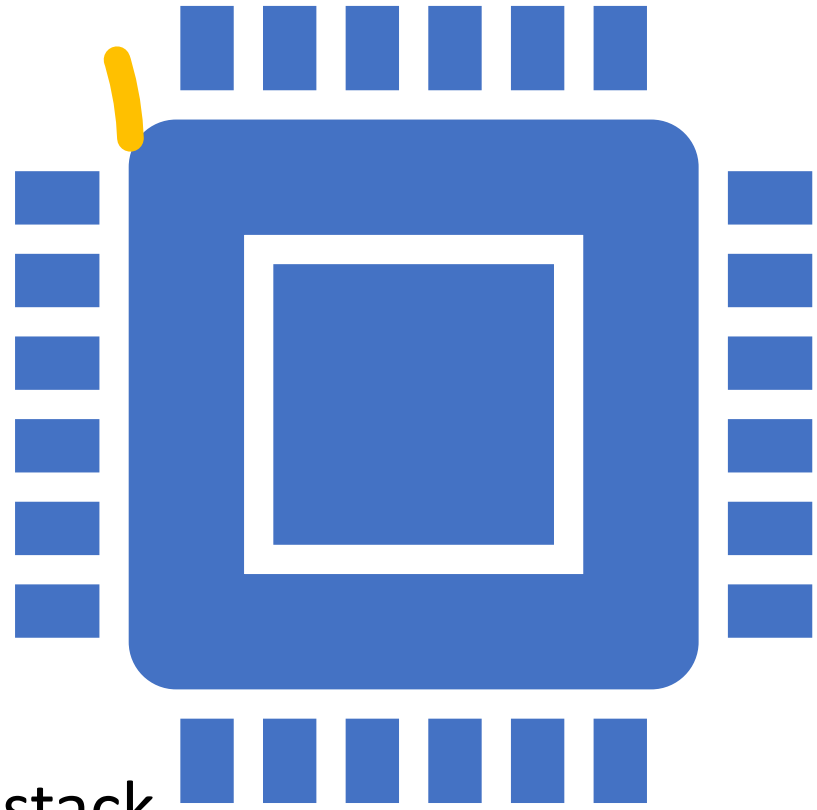
A field of simple, intuitive, insightful, and creative ideas that work ☺

Ideas for?

PAUSE



More precise ones



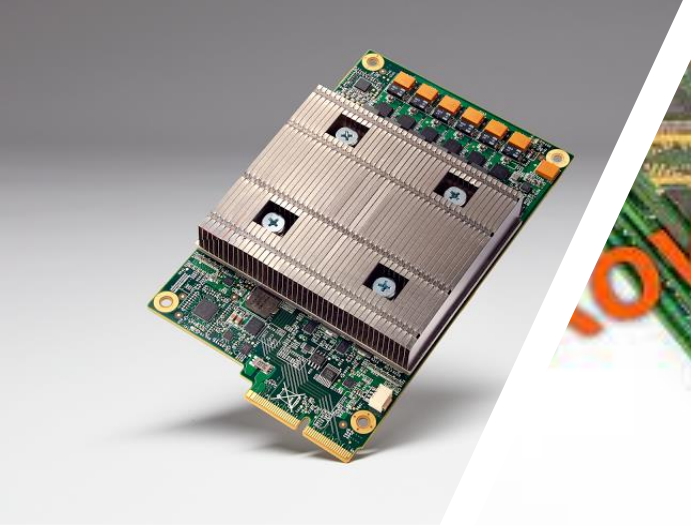
Architecture:

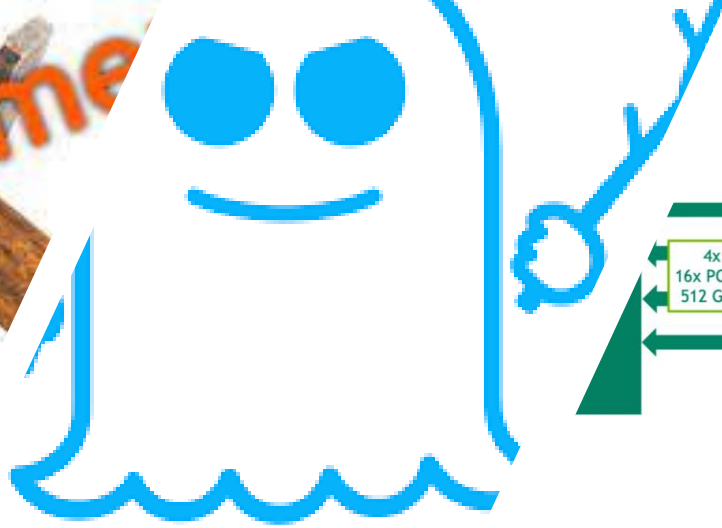
It is not circuits, that is EE part of hardware

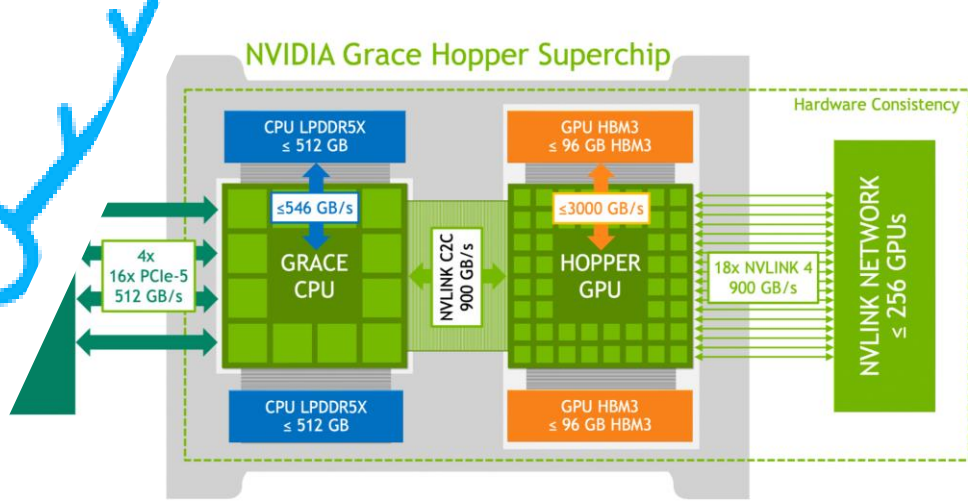
Architecture is the CS part of hardware

It talks to programmer/compiler/OS/…/system stack

CS engineer = compiler engine + OS engine + architecture engine







Computer Architecture in 2023 (Application-oriented)

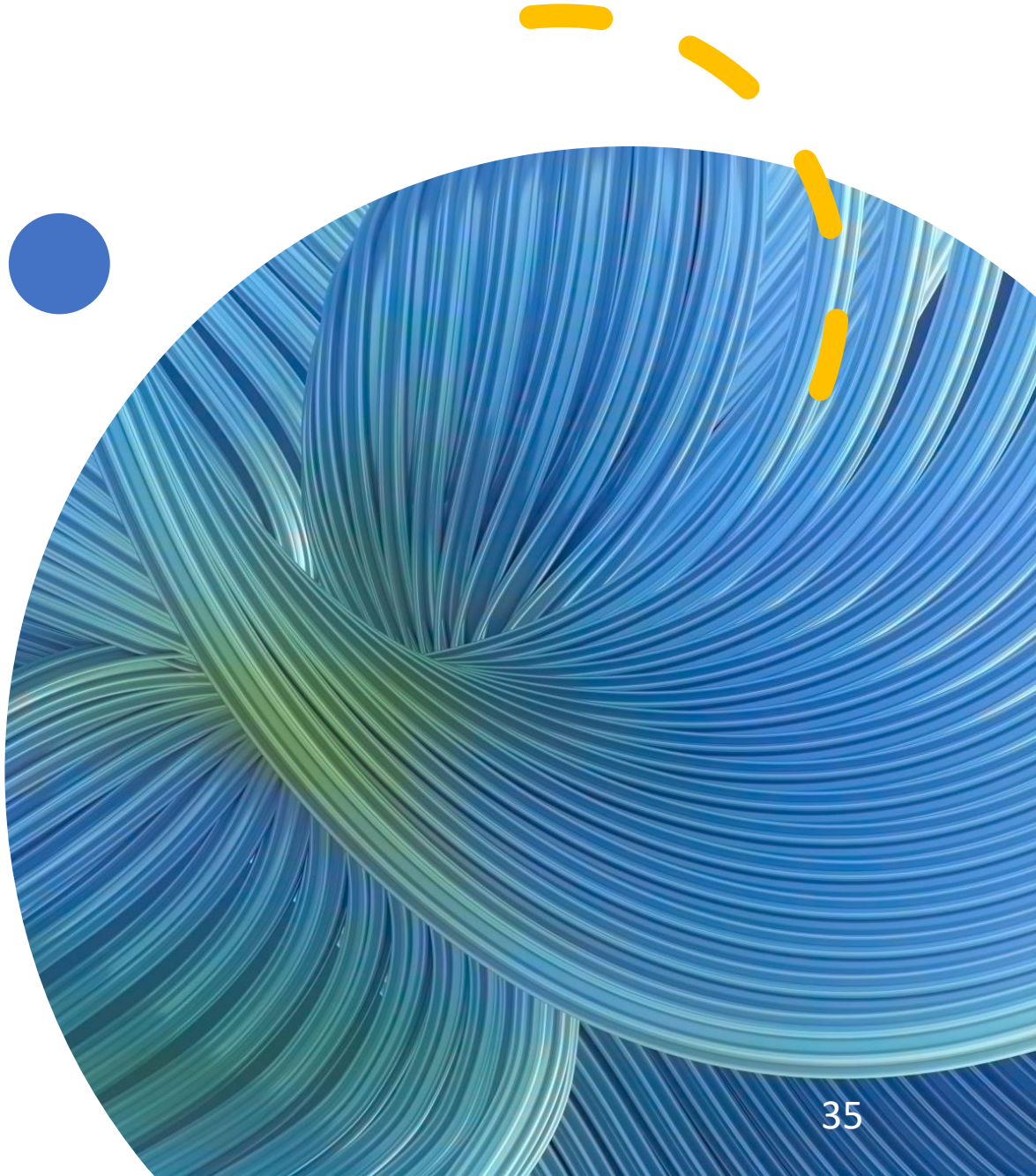
Application/domain oriented ?

Compute units

Generic

Application-specific

Compute units for mobile phones, desktops, and servers



Why study or credit CS683

No computer architecture – No computers/smart-phones ☺

Understand the past to understand the future technologies

Understand the systems stack

Finally get a job at top software/hardware companies.

Content

Memory Hierarchy (including background)

Memory Hierarchy:
Programmer's/Microarchitect's view

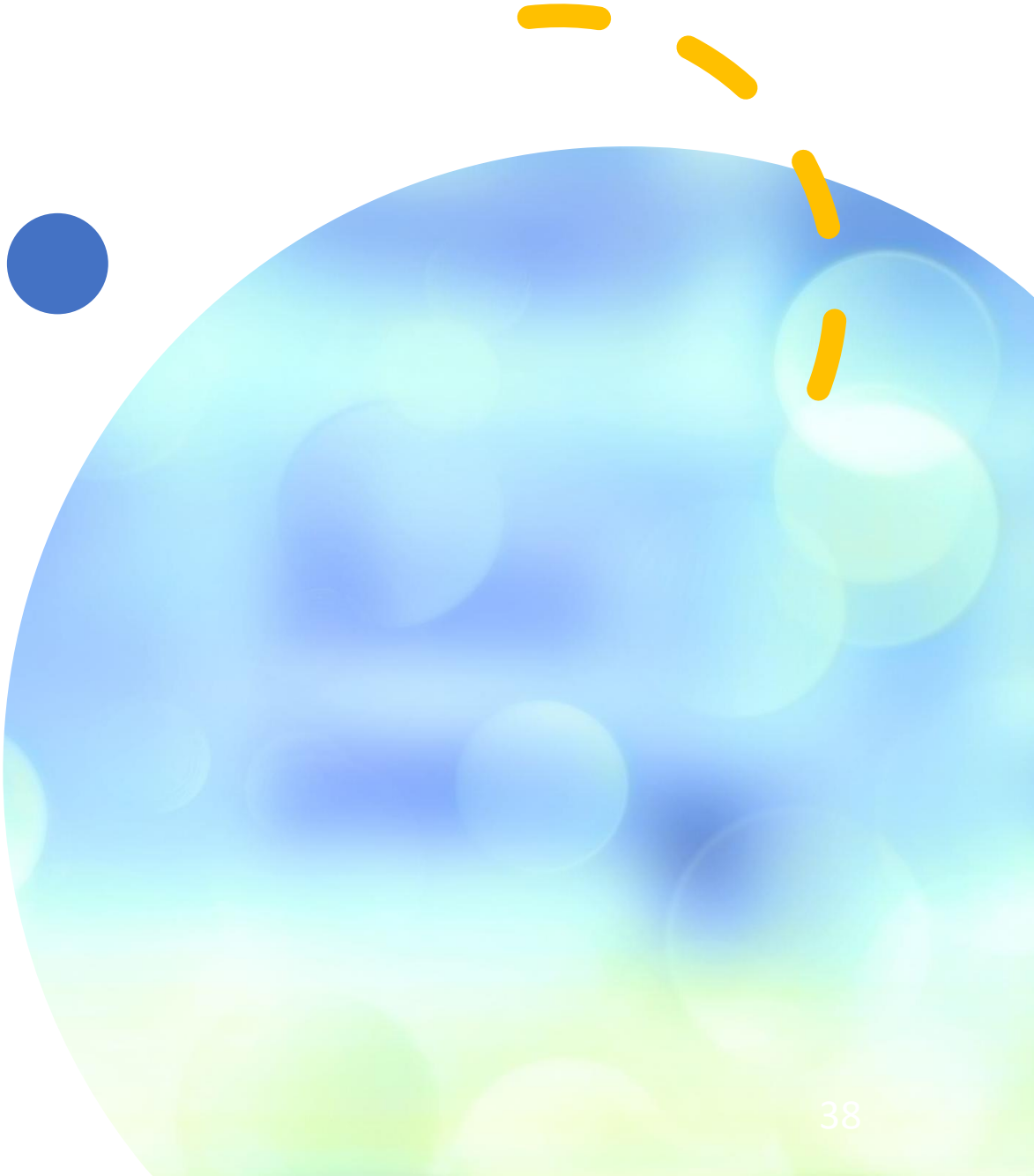
Application specific optimizations: L1I, L1D, LLC

Software/hardware prefetching

Cache optimizations: Improving miss rate and miss latency

Virtual caches: TLB-cache interactions, Page-table walkers

Cache for many-core systems: cache coherence, memory consistency



Modern Processors for servers/desktops/mobiles (including background)

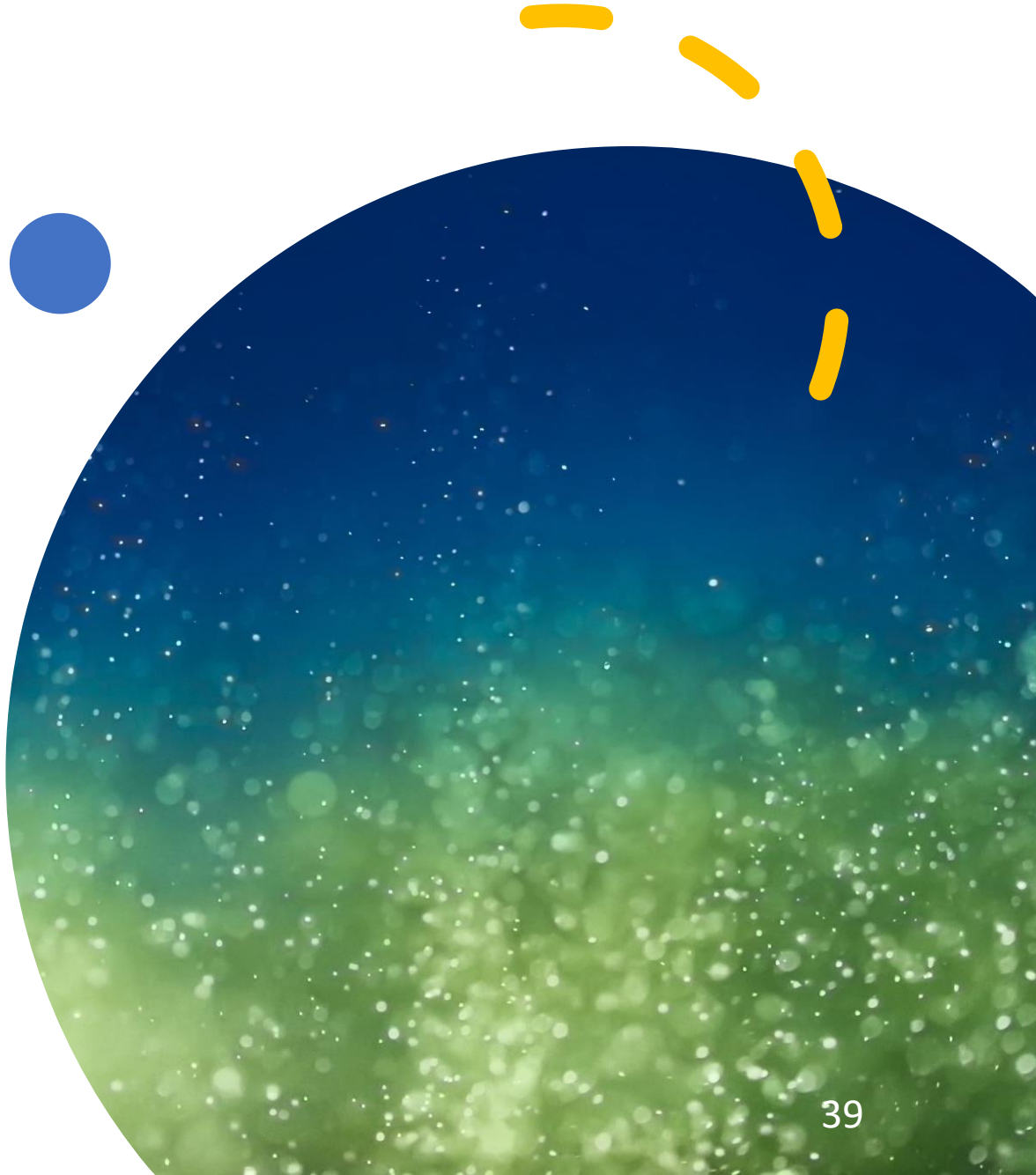
Processors: O3, SMT, vector, SIMD

Designs: Decoupled front-end for servers and mobile phones

Subtle issues while designing high-end processors and application specific processors

GPUs and TPUs

Processor-memory hierarchy interaction



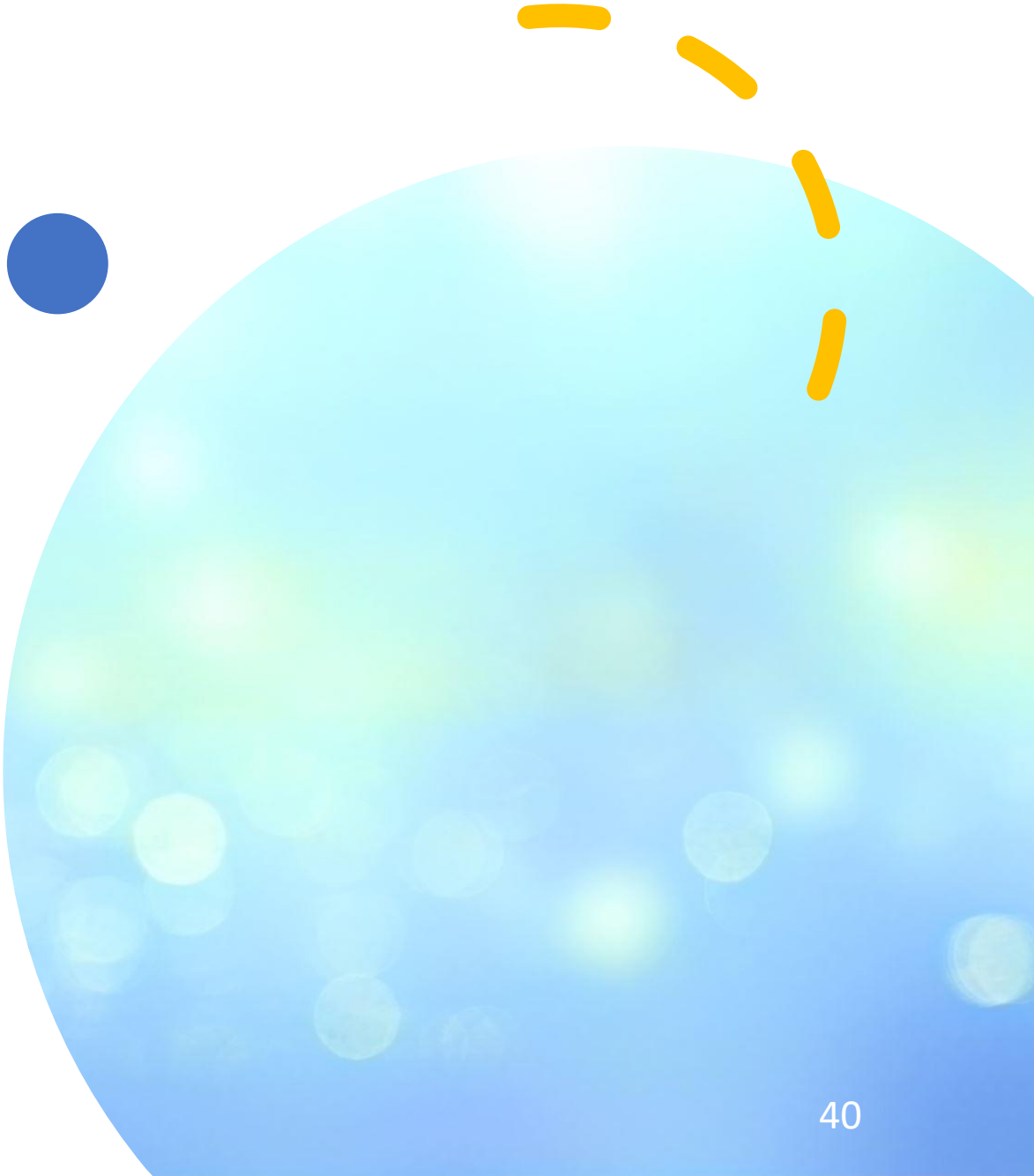
Security (including background)

Timing channel attacks: Caches and other microarchitecture units

Speculative attacks: Spectre and Meltdown

Integrity attacks: Row-hammer attacks

GPU attacks, Attacks on ML models, Trusted Execution environments



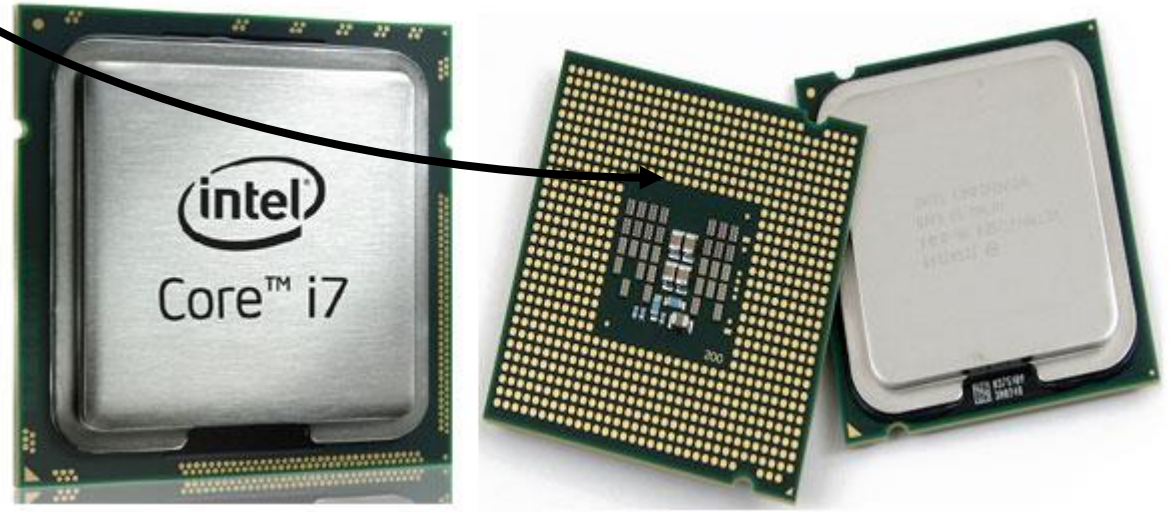
BTW Drop the course if you can ☺ ☺

- Registration count : 99 (good or bad, no idea)
- Ideally, 25 to 50 would be nice ☺
- See if you can drop the course
- If not, it is fine too

Programmer’s view

Let’s start with programming for Performance (know your architecture)

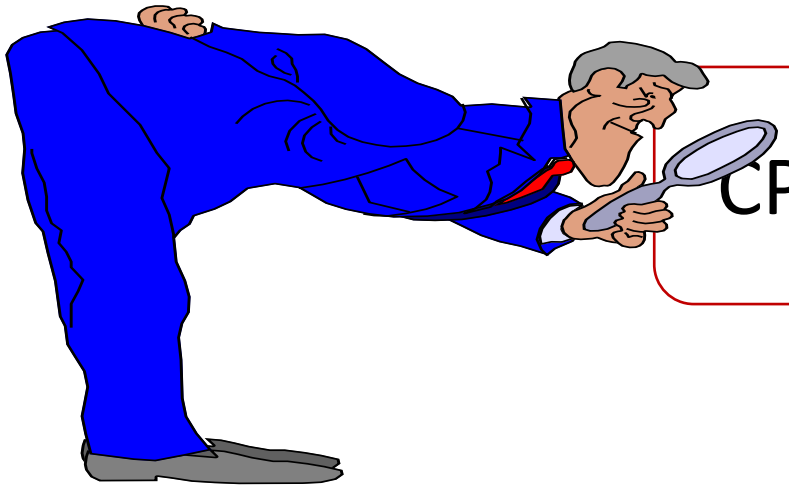
```
void CS683(int *arr, int size) {
   for (int i = 0; i < size-1; ++i)
      arr[i] = arr[i+1];
}
```



Performance

Execution time: CPU time + IO waiting

CPU time: User time + System (OS) time



CPU time: Time spent running a program

Spell it out

$$= \frac{\text{Instructions}}{\text{Program}} \times \frac{\text{Cycles}}{\text{Instruction}} \times \frac{\text{Time}}{\text{Cycle}}$$

(code size) (CPI) (cycle time)

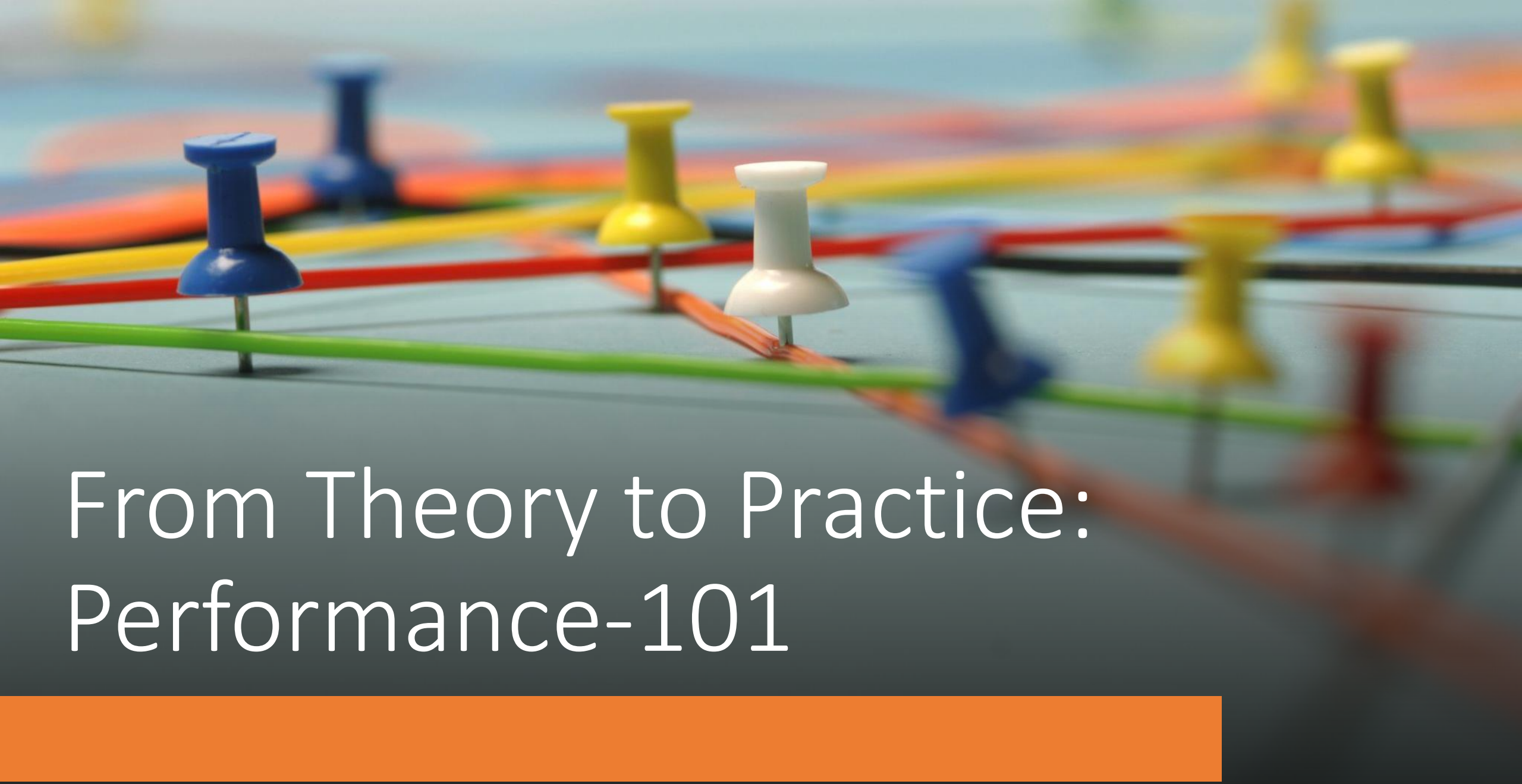
Cycles per Instruction

Instructions: *I mean the DYANMIC (ADD, MUL, JUMP, LOAD/STORE) ones*

```
void bootcamp(int *arr, int size) {
   for (int i = 0; i < size-1; ++i)
      arr[i] = arr[i+1];
}
```

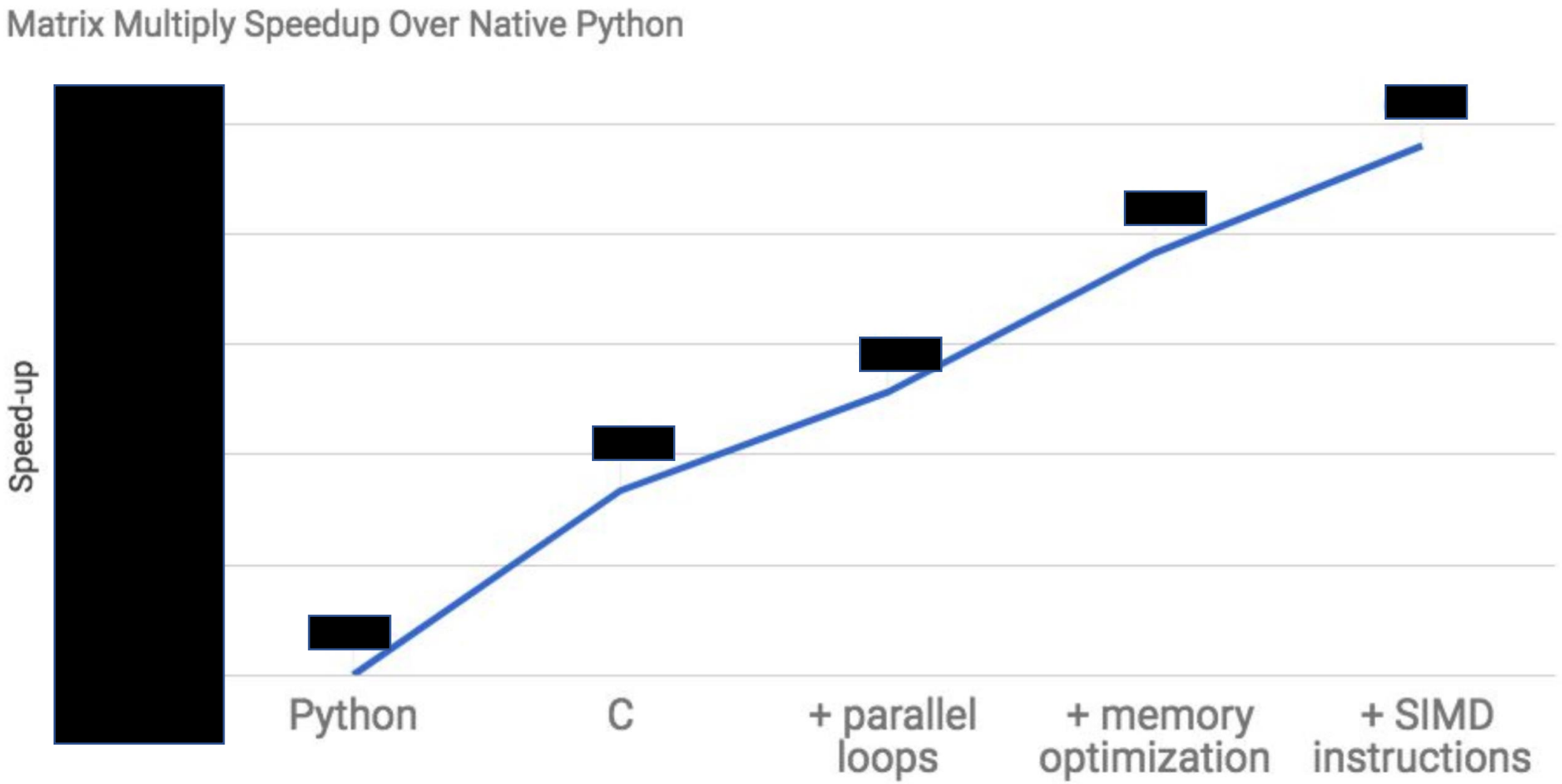
All instructions are not the SAME

So, cycles per instruction is not a constant

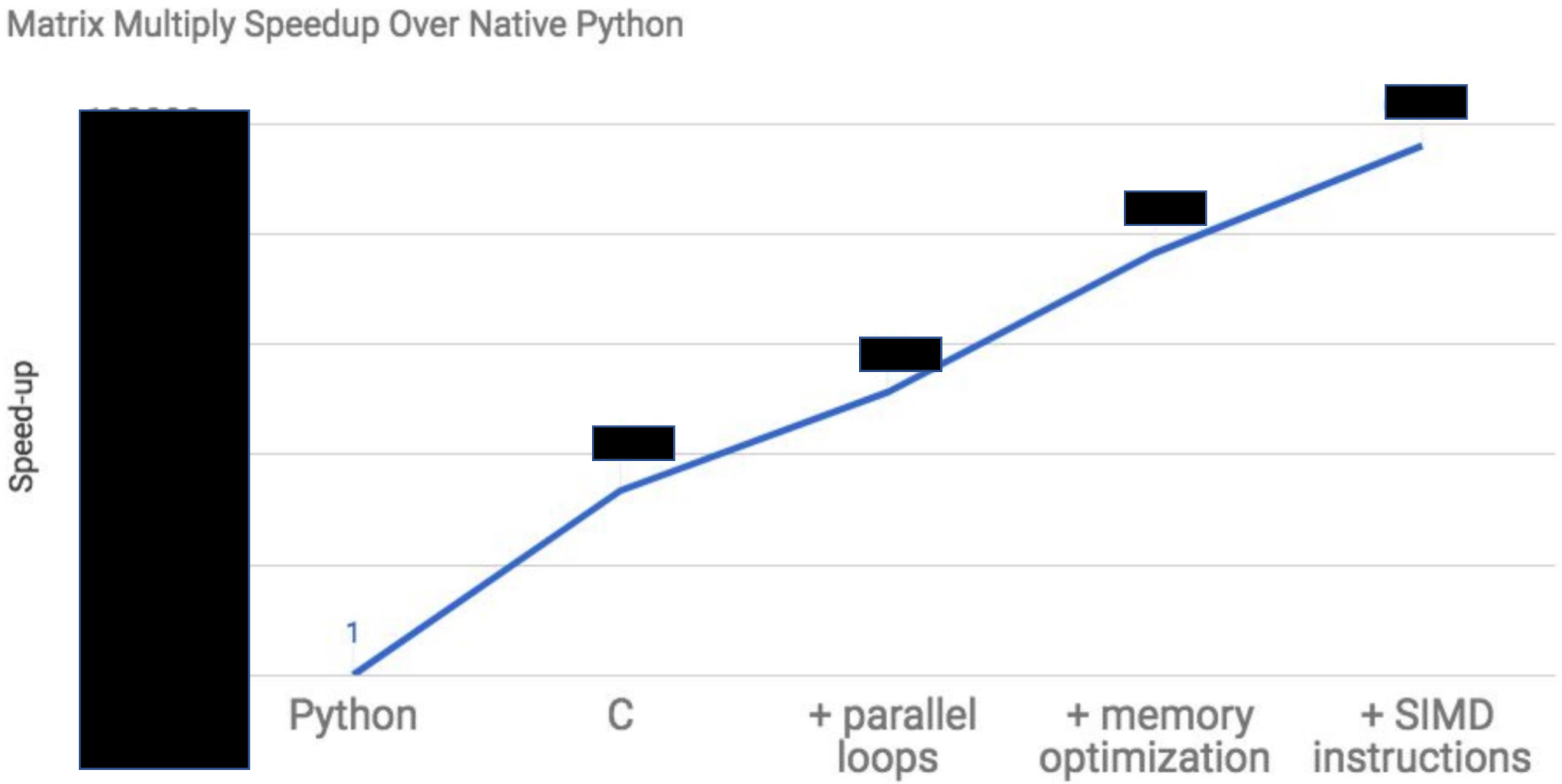


From Theory to Practice: Performance-101

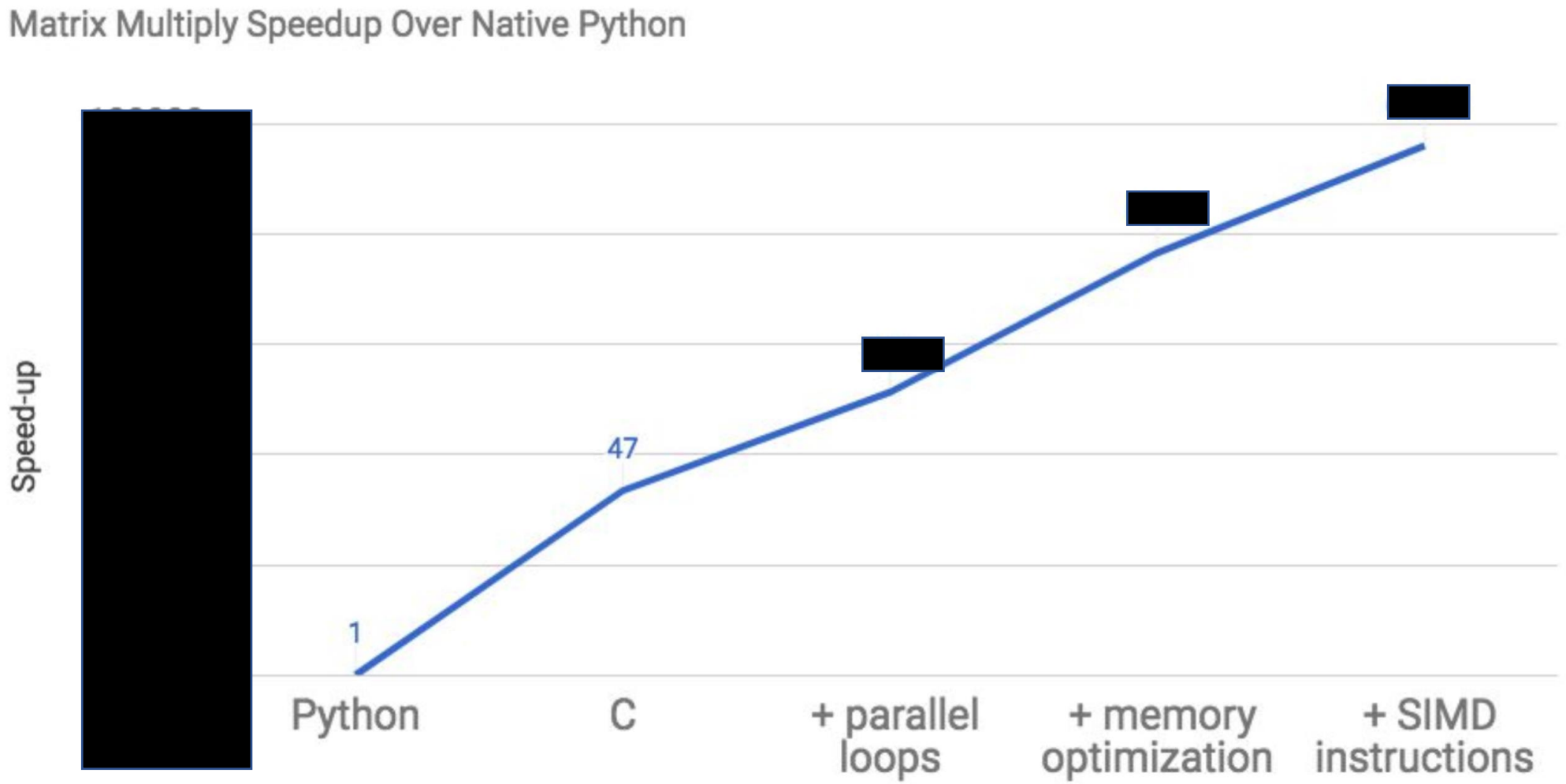
Language: No Big Deal!!



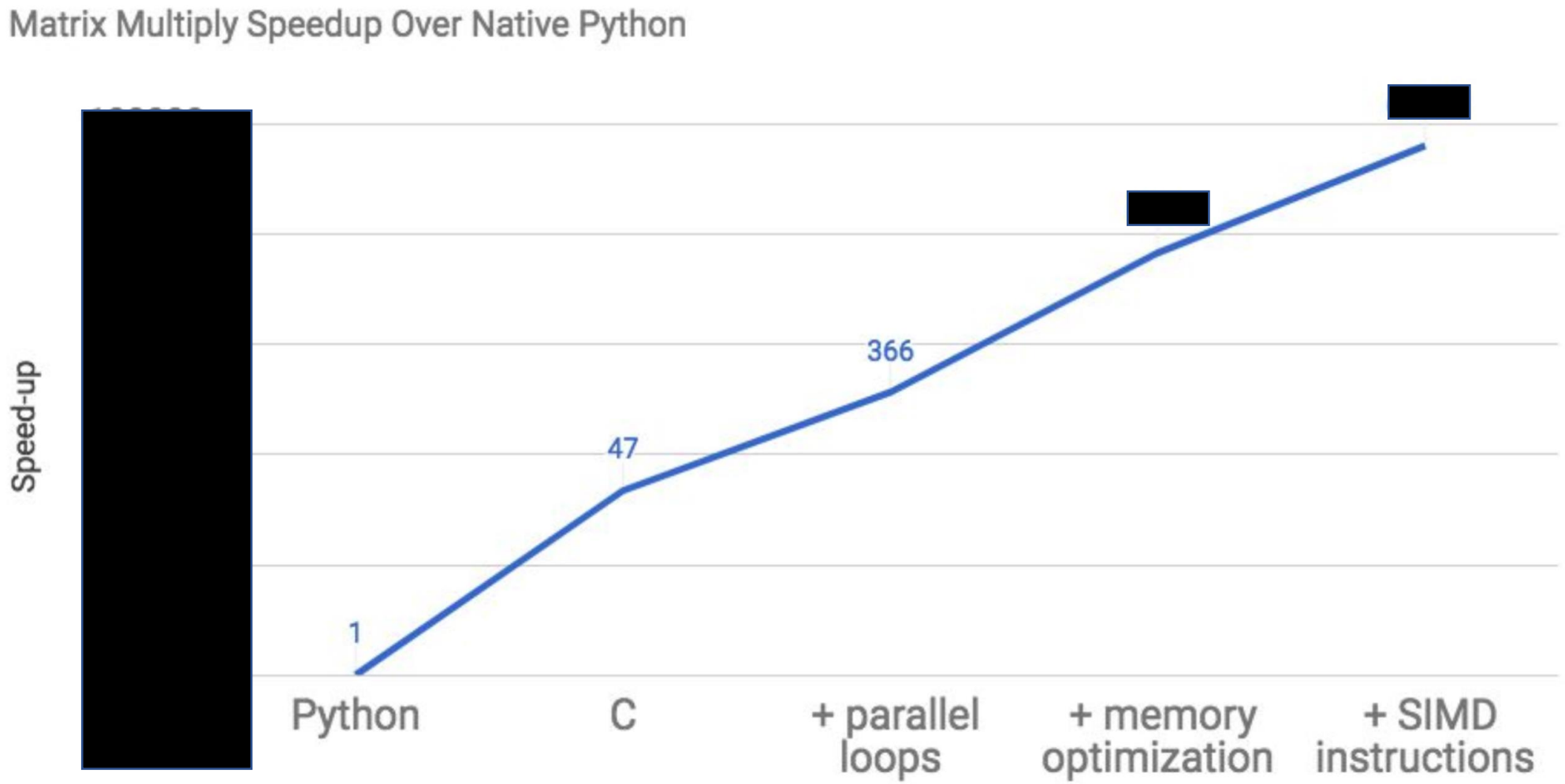
OK. So??


Matrix Multiply Speedup Over Native Python
Speed-up
1
Python
C
+ parallel loops
+ memory optimization
+ SIMD instructions

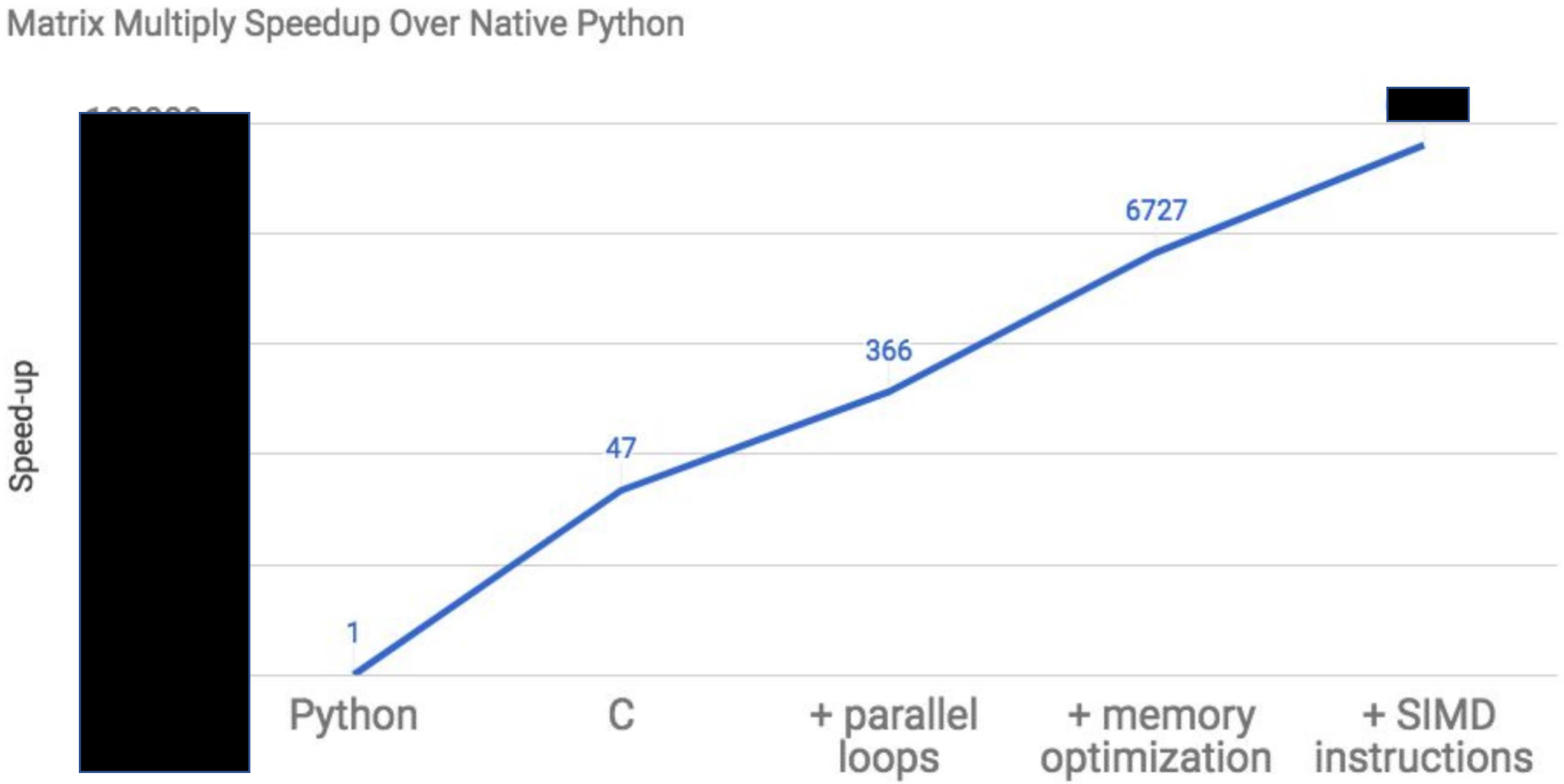
Seriously??


Matrix Multiply Speedup Over Native Python
Speed-up
1
47
Python
C
+ parallel loops
+ memory optimization
+ SIMD instructions

That’s It!! You can’t do better ☹

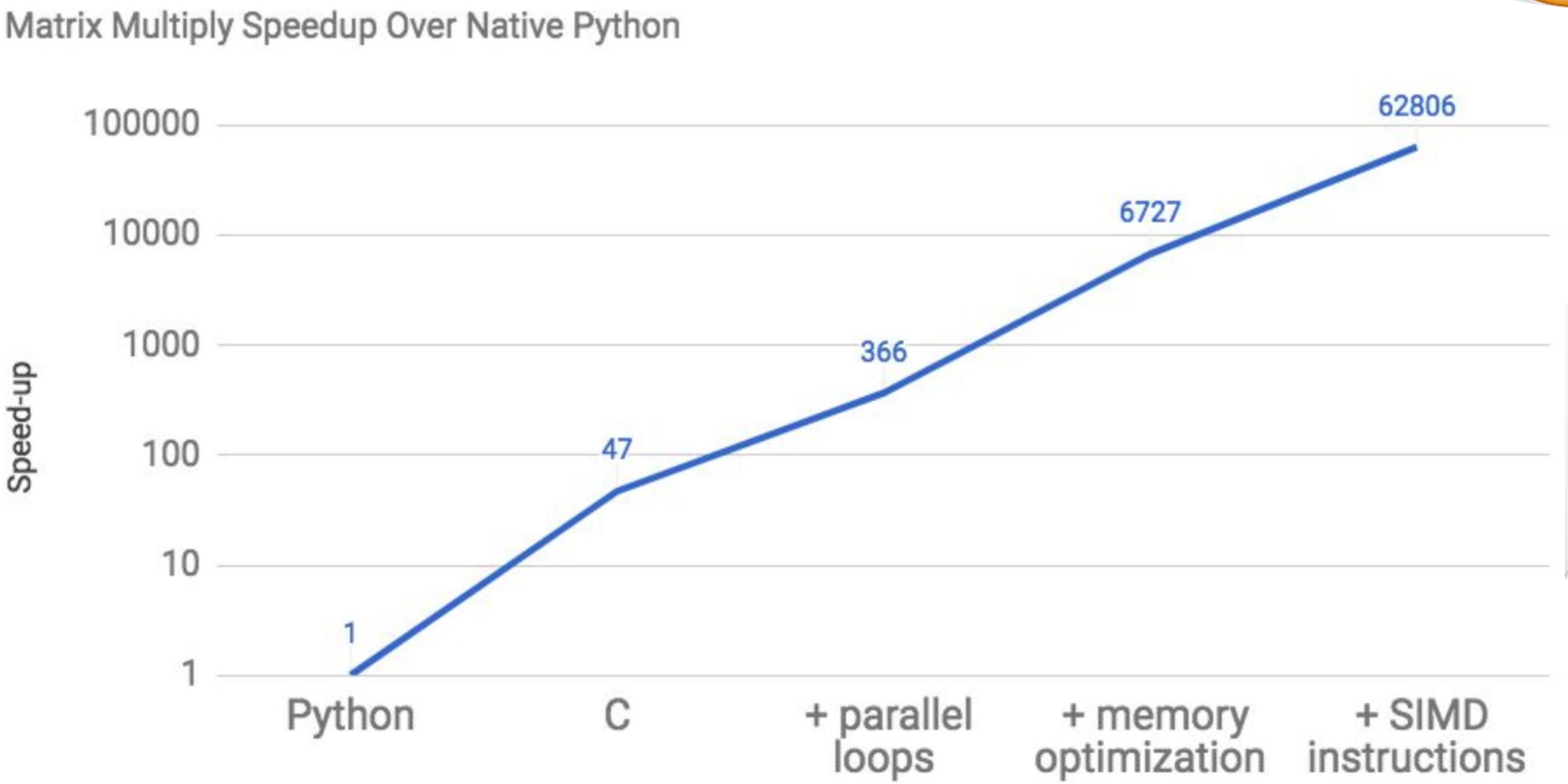


What the hell? This is insane !!




62,806X

Welcome to CS683 !!



First reading assignment (un-graded)

https://www.youtube.com/watch?v=3LVeEjsn8Ts

Turing Award Lecture by H&P, 2018

Exam mein aayega kya ☺

Carpe Diem