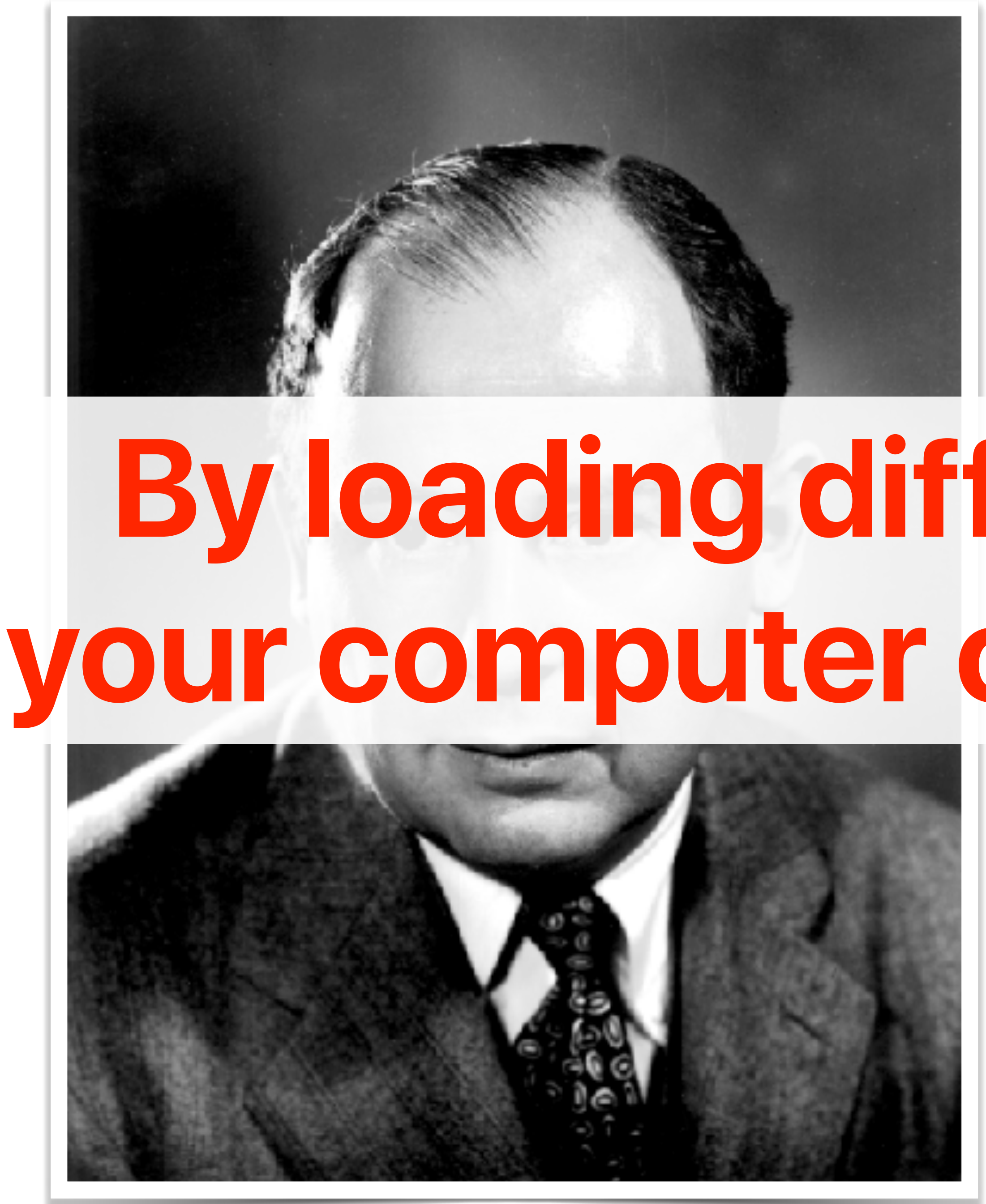
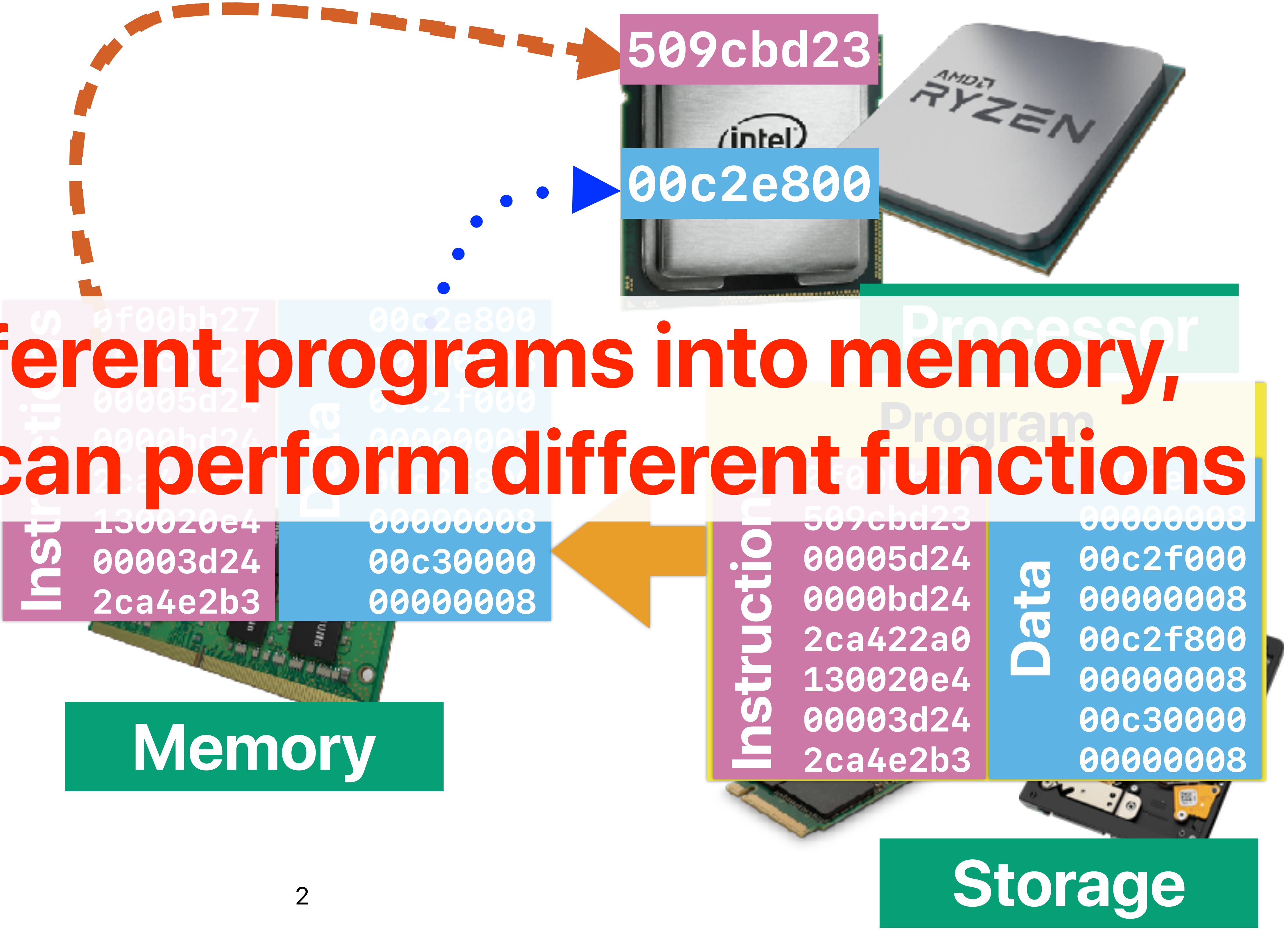
Performance (1): How Good Is “Good”?

Hung-Wei Tseng

Recap: von Neumman Architecture





By loading different programs into memory, your computer can perform different functions

Recap: Demo

```
if(option)
    std::sort(data, data + arraySize);

for (unsigned c = 0; c < arraySize*1000; ++c) {
        if (data[c%arraySize] >= INT_MAX/2)
            sum ++;
    }
}
```

$O(nlog_2n)$

$O(n)$

if option is set to 1: $O(nlog_2n)$

otherwise, O(n): $O(n)$

What’s your favorite restaurant on campus?

- Speed of service
- Taste of food
- Friendliness
- Cleanness

Outline

- Definition of “Performance”
- What affects each factor in “Performance Equation”

Best Natio

Schools in the National Univer
a full range of undergraduate r
producing groundbreaking res

To unlock full rankings, SAT/A

SUMMARY

443

Scho

Loca

Rank

Best Compute

Best Computer Science Schools

Ranked in 2022, part of Best Science S

Earning a graduate degree in compute
technology companies and colleges a
reflects its average rating on a scale fr
institutions. Read the methodology »

UC San Diego Ranked No. 1 Public University by Washington Monthly

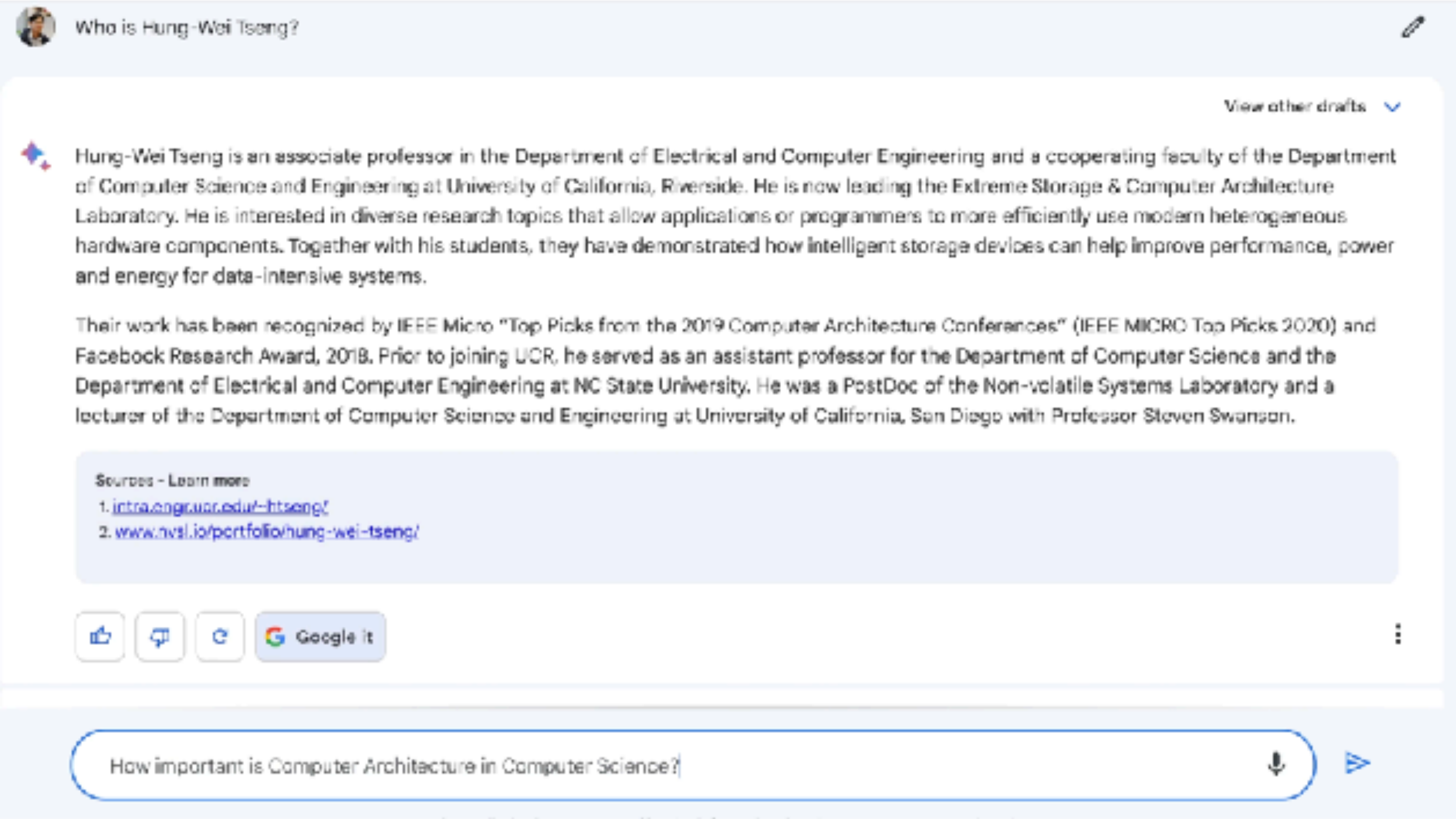
Campus celebrated as a leader in social mobility, research and public

U.S. News names UCR No. 1 public for social mobility

UC Riverside has been the top public university in the nation every year since the social mobility category's inception

What does it really mean by “better” performance

Bard



Who is Hung-Wei Tseng?

Hung-Wei Tseng is an associate professor in the Department of Electrical and Computer Engineering and a cooperating faculty of the Department of Computer Science and Engineering at University of California, Riverside. He is now leading the Extreme Storage & Computer Architecture Laboratory. He is interested in diverse research topics that allow applications or programmers to more efficiently use modern heterogeneous hardware components. Together with his students, they have demonstrated how intelligent storage devices can help improve performance, power and energy for data-intensive systems.

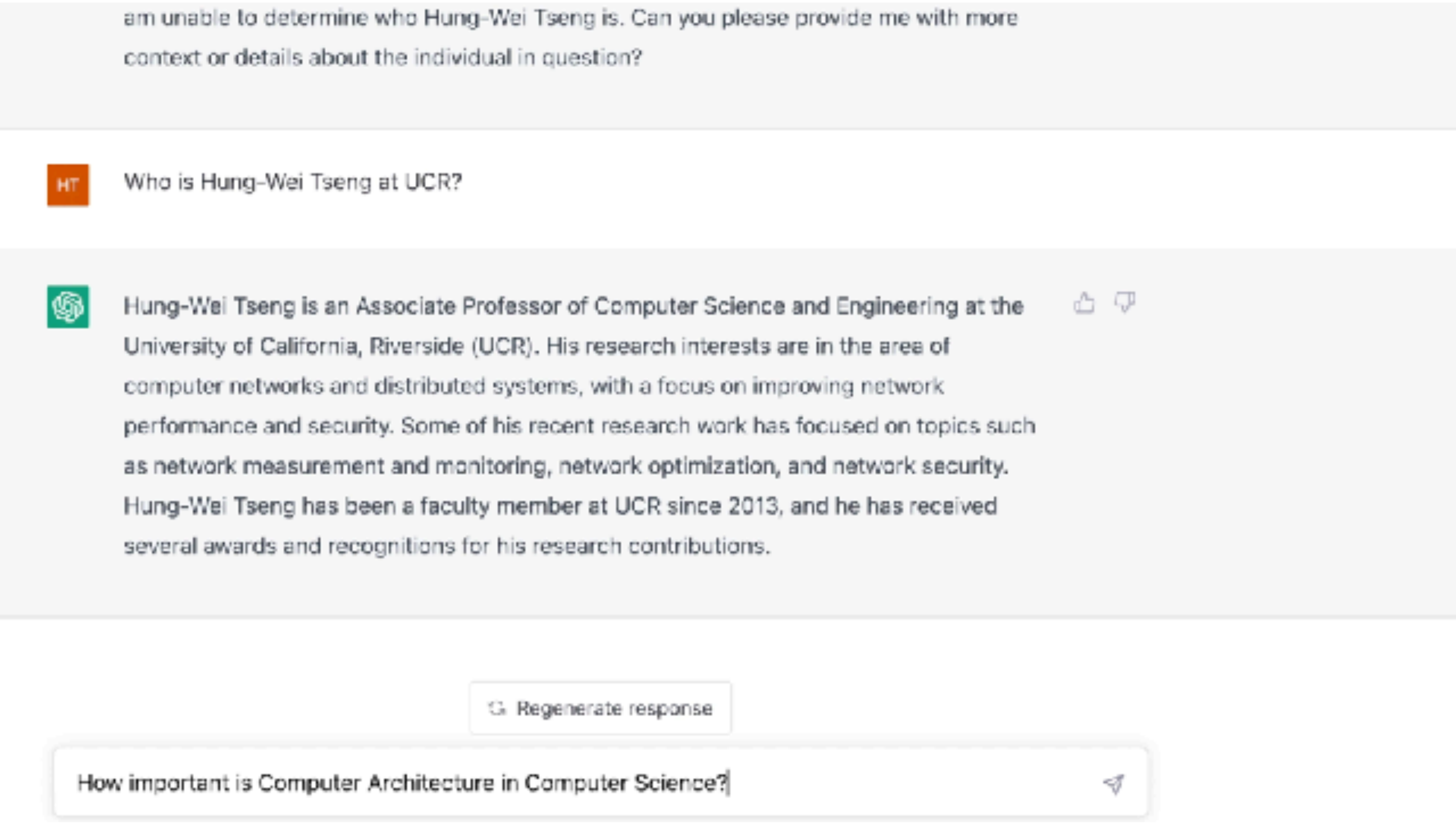
Their work has been recognized by IEEE Micro "Top Picks from the 2019 Computer Architecture Conferences" (IEEE MICRO Top Picks 2020) and Facebook Research Award, 2018. Prior to joining UCR, he served as an assistant professor for the Department of Computer Science and the Department of Electrical and Computer Engineering at NC State University. He was a PostDoc of the Non-volatile Systems Laboratory and a lecturer of the Department of Computer Science and Engineering at University of California, San Diego with Professor Steven Swanson.

Sources - Learn more

1. intra.engr.ucr.edu/~htseng/
2. www.nvsl.io/portfolio/hung-wei-tseng/

How important is Computer Architecture in Computer Science?

182 words/6 secs



am unable to determine who Hung-Wei Tseng is. Can you please provide me with more context or details about the individual in question?

Who is Hung-Wei Tseng at UCR?

Hung-Wei Tseng is an Associate Professor of Computer Science and Engineering at the University of California, Riverside (UCR). His research interests are in the area of computer networks and distributed systems, with a focus on improving network performance and security. Some of his recent research work has focused on topics such as network measurement and monitoring, network optimization, and network security. Hung-Wei Tseng has been a faculty member at UCR since 2013, and he has received several awards and recognitions for his research contributions.

How important is Computer Architecture in Computer Science?

170 words/18 secs

Peer instruction

- Before the lecture — You need to complete the required **reading**
- During the lecture — I'll bring in activities to ENGAGE you in exploring your understanding of the material
  - Popup questions
  - Individual **thinking** — use polls in Zoom to express your opinion
  - Group **discussion**
    - Breakout rooms based on your residential colleges!
    - Use polls in Zoom to express your group's opinion
  - Whole-classroom **discussion** — we would like to hear from you

Read **Think** **Discuss**

Now, make sure you login to Poll Everywhere (through the App or the website) with UCRNetID

Now, you have at least 90 seconds to answer the question!

https://www.pollev.com/hungweitseng close in 1:30

Bard v.s. ChatGPT

- Comparing the experiments we have done with Bard and ChatGPT, how many of the following metrics does Bard outperforms ChatGPT?
  ① Response time
  ② Throughput
  ③ End-to-end latency (i.e., total execution time)
  ④ Quality of results
  A. 0
  B. 1
  C. 2
  D. 3
  E. 4

A

B

C

D

E

Total Results: 0

Powered by Poll Everywhere

[illegible]

Now, it’s time to discuss with your surroundings — and make sure you vote again after the discussion!

What do you care the most when?

- Consider the following performance metrics
  1. Network Bandwidth (data/sec)
  2. End-to-end Latency (ms)
  3. Frame Rate (frames/sec)
  4. Throughput (ops/sec)

  Which option contains the best match of the most important performance metric for each application?

	Fortnite (Online gaming)	YouTube/Netflix	Download ISO images	Training an ML model
A	4	3	1	2
B	4	1	3	2
C	2	1	3	4
D	2	3	1	4
E	None of the above			

“Better” Performance

- The right metric — latency? throughput? quality of results?
- The quantitative comparison — A is better than B by “how much”

Let’s start with “end-to-end latency” as the default metric — how long it takes to execute a program?

https://www.pollev.com/hungweitseng close in 1:30

CPU Performance Equation (X)

- Assume that we have an application composed with a total of **5000000000** instructions, in which **20%** of them are “Type-A” instructions with an average **CPI of 8** cycles, **20%** of them are “Type-B” instructions with an average **CPI of 4** cycles and **the rest** instructions are “Type-C” instructions with average **CPI of 1** cycle. If the processor runs at **3 GHz**, how long is the execution time?
  A. 3.67 sec
  B. 5 sec
  C. 6.67 sec
  D. 15 sec
  E. 45 sec

CPU Performance Equation

$$Performance = \frac{1}{Execution\ Time}$$

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

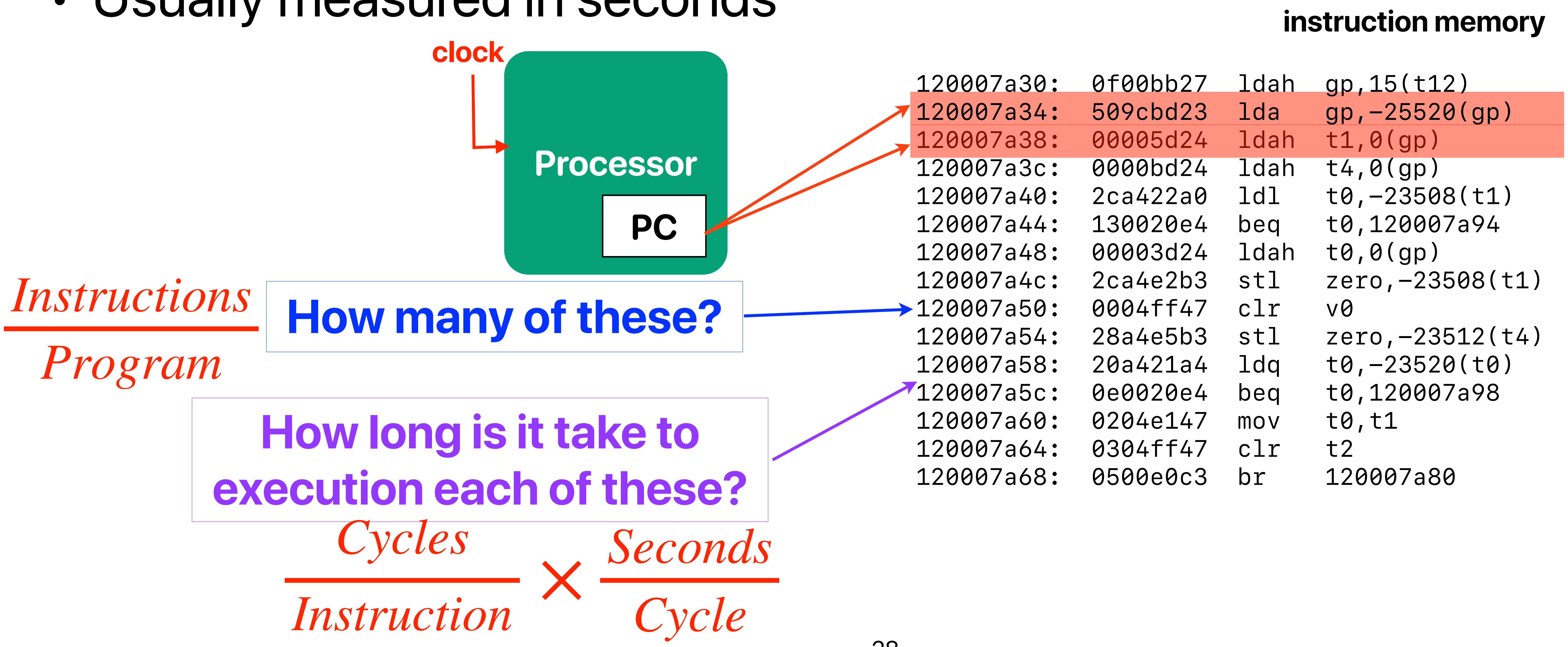
$$ET = IC \times CPI \times CT$$

$1GHz = 10^9 Hz = \frac{1}{10^9} sec\ per\ cycle = 1\ ns\ per\ cycle$

$$\frac{1}{Frequency(i.e., clock\ rate)}$$

Execution Time

- The simplest kind of performance
- Shorter execution time means better performance
- Usually measured in seconds



instruction memory

```
120007a30:  0f00bb27  ldah  gp,15(t12)
120007a34:  509cbd23  lda   gp,-25520(gp)
120007a38:  00005d24  ldah  t1,0(gp)
120007a3c:  0000bd24  ldah  t4,0(gp)
120007a40:  2ca422a0  ldl   t0,-23508(t1)
120007a44:  130020e4  beq   t0,120007a94
120007a48:  00003d24  ldah  t0,0(gp)
120007a4c:  2ca4e2b3  stl   zero,-23508(t1)
120007a50:  0004ff47  clr   v0
120007a54:  28a4e5b3  stl   zero,-23512(t4)
120007a58:  20a421a4  ldq   t0,-23520(t0)
120007a5c:  0e0020e4  beq   t0,120007a98
120007a60:  0204e147  mov   t0,t1
120007a64:  0304ff47  clr   t2
120007a68:  0500e0c3  br    120007a80
```

$$\frac{\textit{Instructions}}{\textit{Program}}$$

How many of these?

How long is it take to execution each of these?

$$\frac{\textit{Cycles}}{\textit{Instruction}} \times \frac{\textit{Seconds}}{\textit{Cycle}}$$

Performance Equation (X)

- Assume that we have an application composed with a total of **5000000000** instructions, in which **20%** of them are “Type-A” instructions with an average **CPI of 8** cycles, **20%** of them are “Type-B” instructions with an average **CPI of 4** cycles and the rest instructions are “Type-C” instructions with average **CPI of 1** cycle. If the processor runs at **3 GHz**, how long is the execution time?

A. 3.67 sec

[✓ B. 5 sec]

C. 6.67 sec

D. 15 sec

E. 45 sec

$$ET = (5 \times 10^9) \times \underbrace{(20\% \times 8 + 20\% \times 4 + 60\% \times 1)}_{\textbf{average CPI}} \times \frac{1}{3 \times 10^9} sec = 5$$

$$ET = IC \times CPI \times CT$$

https://www.pollev.com/hungweitseng close in 1:30

Speedup of Y over X

- Consider the same program on the following two machines, X and Y. By how much Y is faster than X?

	Clock Rate	Instructions	Percentage of Type-A	CPI of Type-A	Percentage of Type-B	CPI of Type-B	Percentage of Type-C	CPI of Type-C
Machine X	3 GHz	5000000000	20%	8	20%	4	60%	1
Machine Y	5 GHz	5000000000	20%	13	20%	4	60%	1

A. 0.2
B. 0.25
C. 0.8
D. 1.25
E. No changes

Speedup

- The relative performance between two machines, X and Y. Y is n times faster than X

$$n = \frac{Execution\ Time_X}{Execution\ Time_Y}$$

- The speedup of Y over X

$$Speedup = \frac{Execution\ Time_X}{Execution\ Time_Y}$$

Speedup of Y over X

- Consider the same program on the following two machines, X and Y. By how much Y is faster than X?

	Clock Rate	Instructions	Percentage of Type-A	CPI of Type-A	Percentage of Type-B	CPI of Type-B	Percentage of Type-C	CPI of Type-C
Machine X	3 GHz	5000000000	20%	8	20%	4	60%	1
Machine Y	5 GHz	5000000000	20%	13	20%	4	60%	1

$$ET_Y = (5 \times 10^9) \times (20\% \times 13 + 20\% \times 4 + 60\% \times 1) \times \frac{1}{5 \times 10^9} sec = 4$$

A. 0.2

B. 0.25

C. 0.8

[✓ D. 1.25]

E. No changes

$$Speedup = \frac{Execution\ Time_X}{Execution\ Time_Y}$$

$$= \frac{5}{4} = 1.25$$

What Affects Each Factor in Performance Equation

https://www.pollev.com/hungweitseng close in 1:30

What can programmers affect?

- Performance equation consists of the following three factors

  ① IC

  ② CPI

  ③ CT

How many can a **programmer** affect?

A. 0

B. 1

C. 2

D. 3

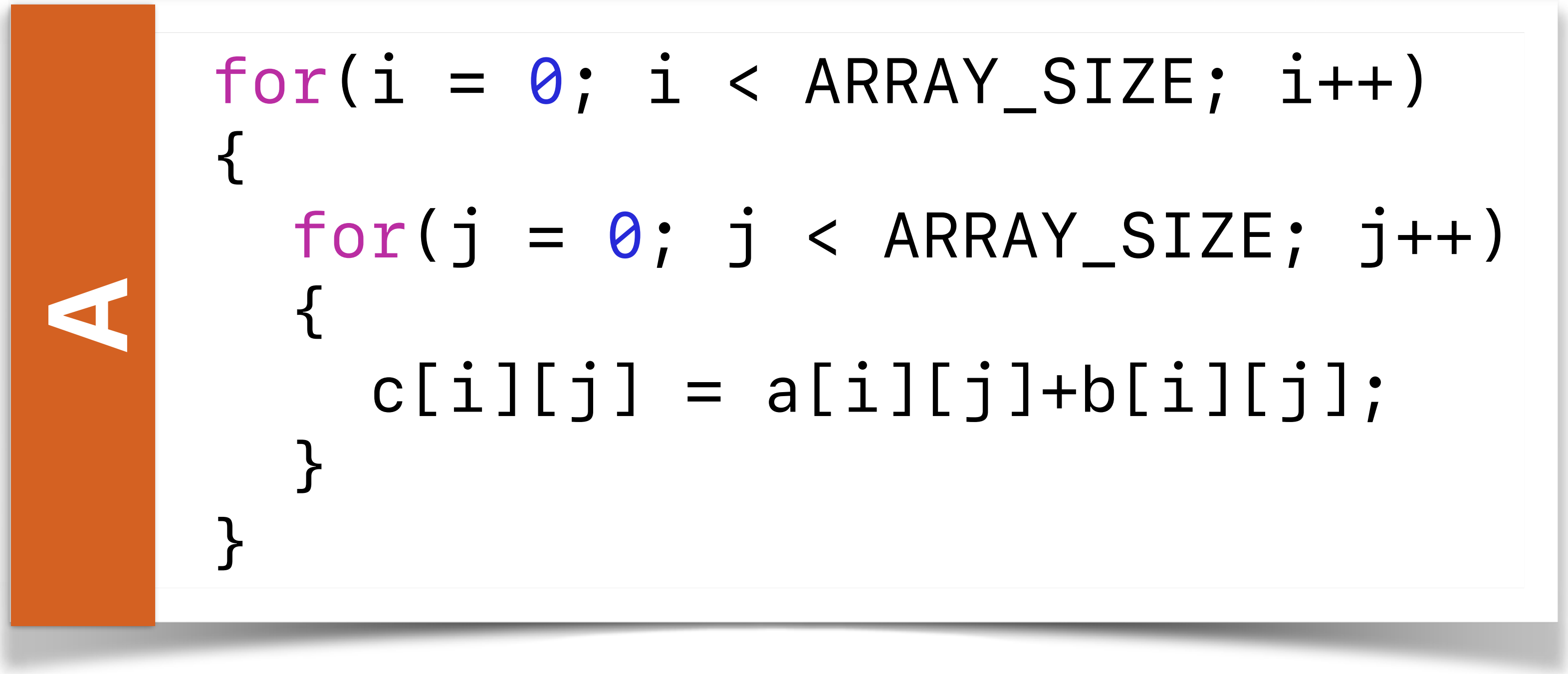
A

B

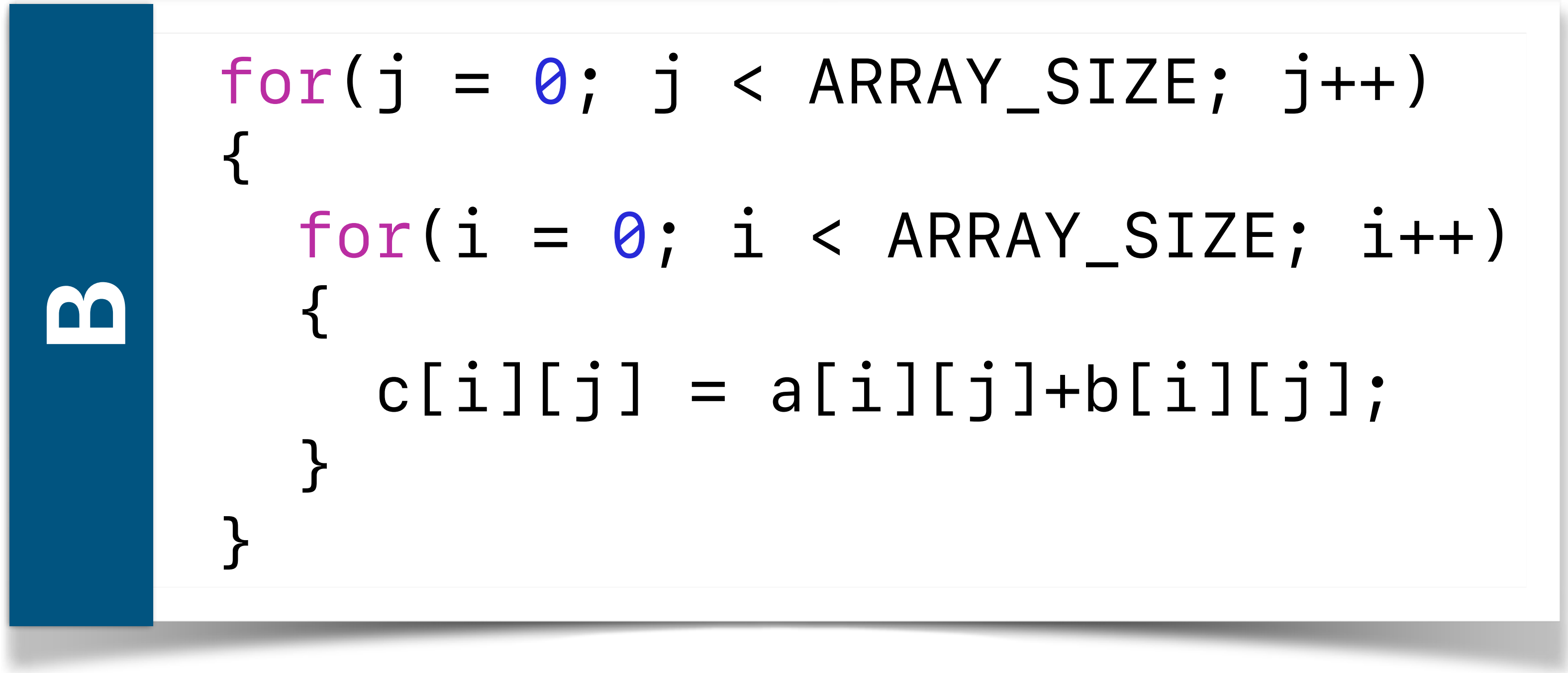
C

D

Demo — programmer & performance



```
for(i = 0; i < ARRAY_SIZE; i++)
{
  for(j = 0; j < ARRAY_SIZE; j++)
  {
    c[i][j] = a[i][j]+b[i][j];
  }
}
```



```
for(j = 0; j < ARRAY_SIZE; j++)
{
  for(i = 0; i < ARRAY_SIZE; i++)
  {
    c[i][j] = a[i][j]+b[i][j];
  }
}
```

$O(n^2)$ **Complexity** $O(n^2)$

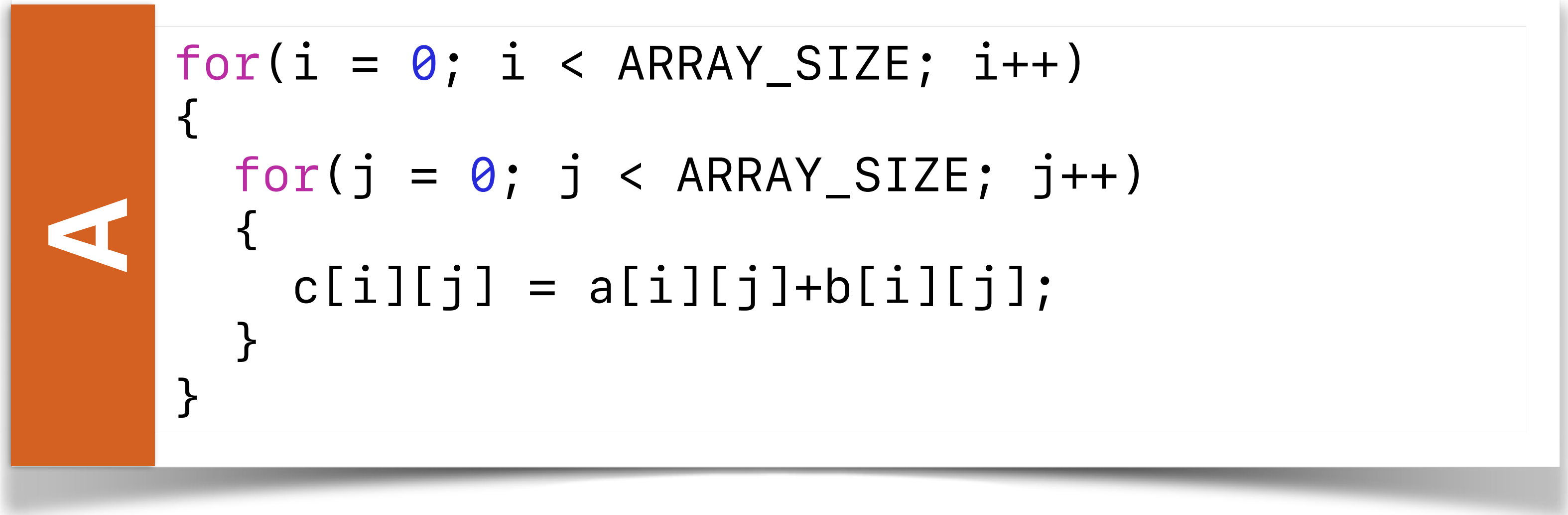
Instruction Count?

Clock Rate

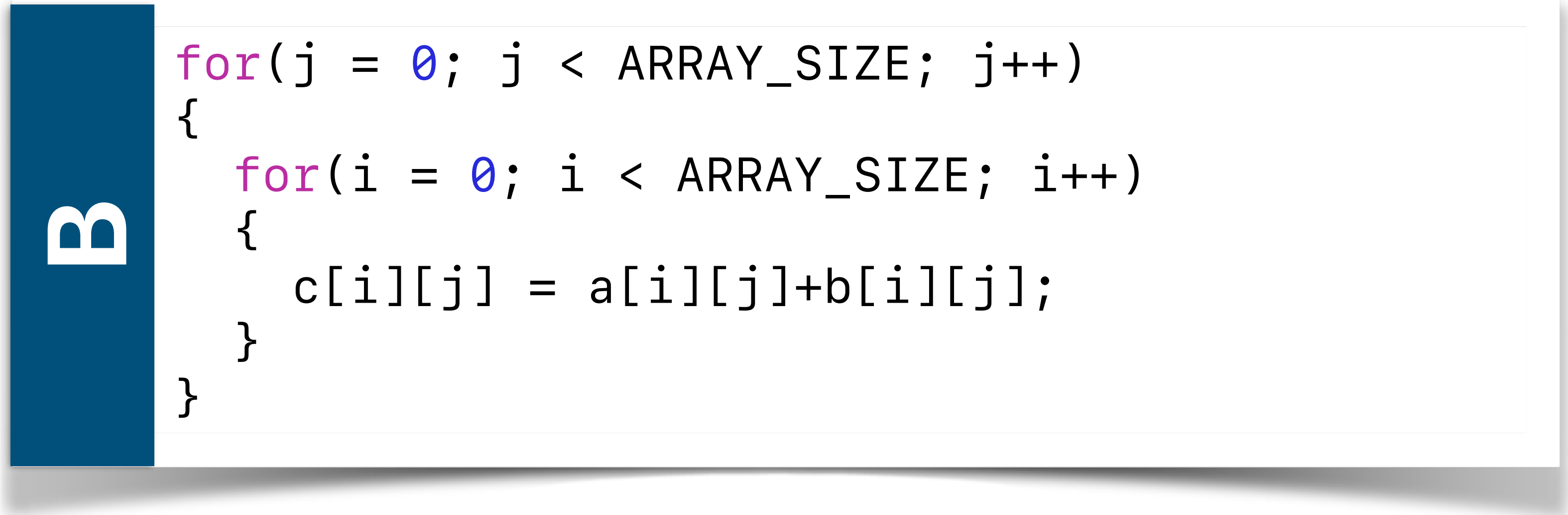
CPI

https://www.pollev.com/hungweitseng close in 1:30

Demo — programmer & performance



```
for(i = 0; i < ARRAY_SIZE; i++)
{
  for(j = 0; j < ARRAY_SIZE; j++)
  {
    c[i][j] = a[i][j]+b[i][j];
  }
}
```



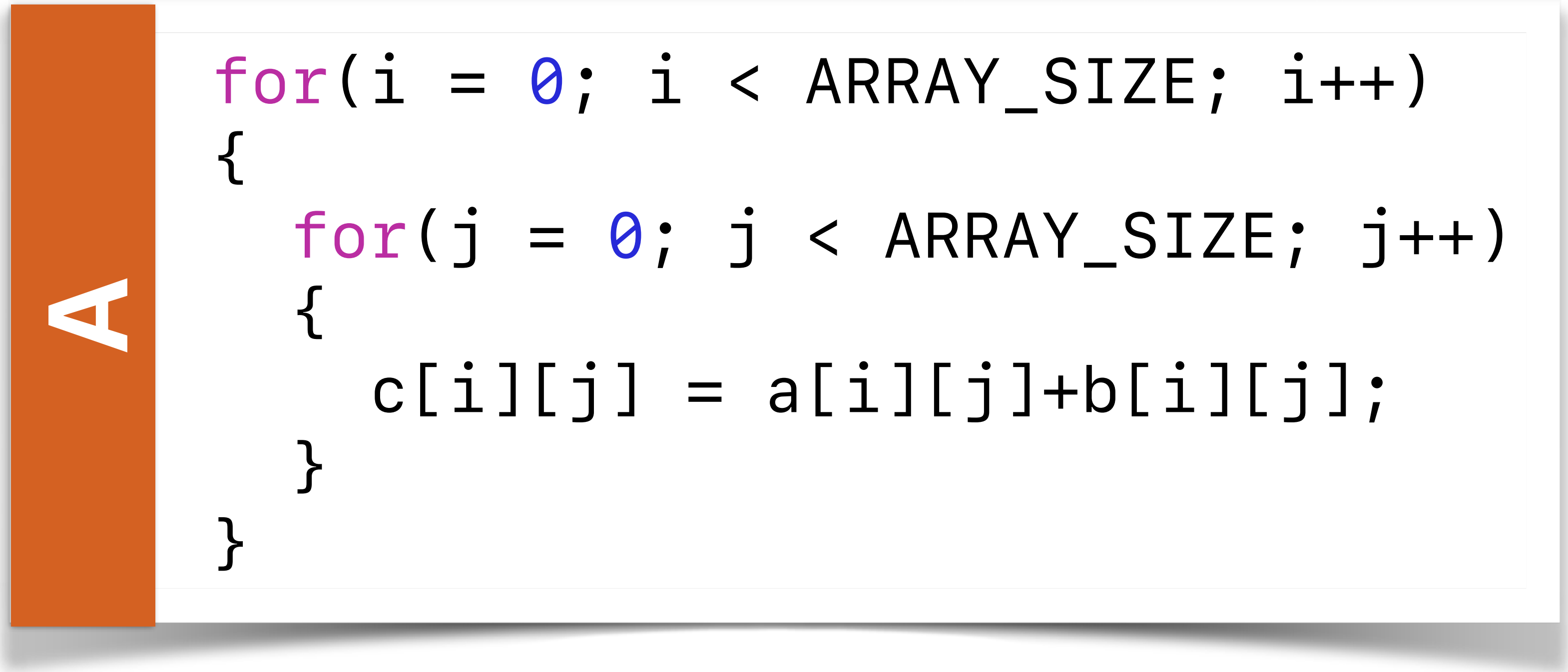
```
for(j = 0; j < ARRAY_SIZE; j++)
{
  for(i = 0; i < ARRAY_SIZE; i++)
  {
    c[i][j] = a[i][j]+b[i][j];
  }
}
```

How many of the following make(s) the performance different between version A & version B?

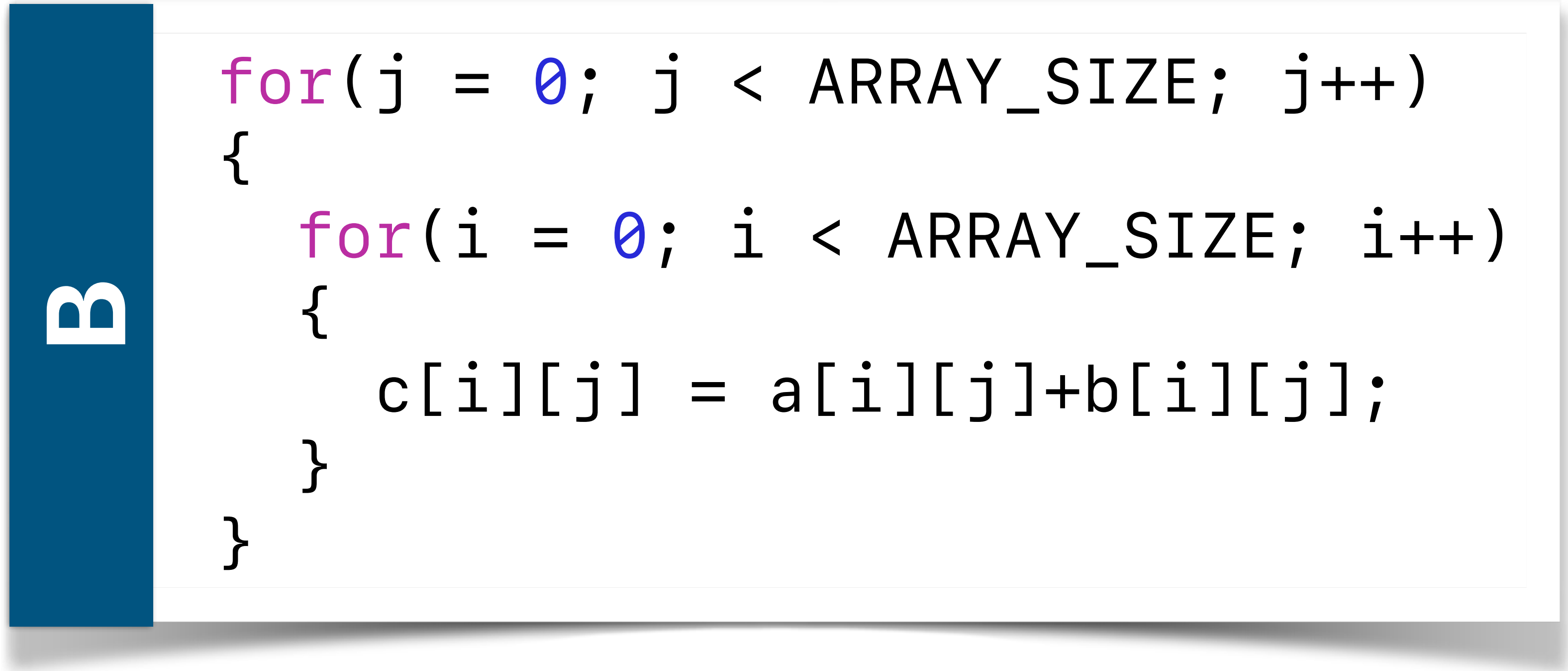
① IC
② CPI
③ CT

A. 0
B. 1
C. 2
D. 3

Demo — programmer & performance



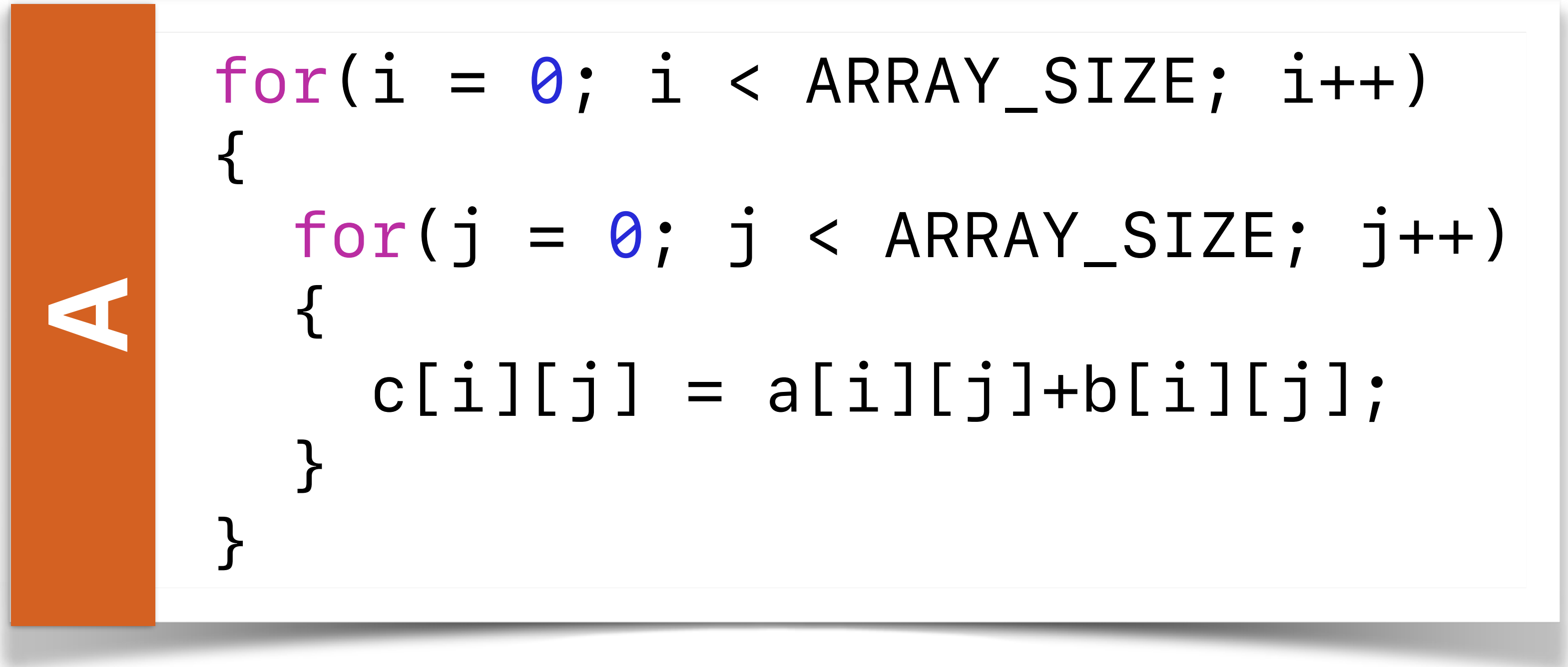
```
for(i = 0; i < ARRAY_SIZE; i++)
{
  for(j = 0; j < ARRAY_SIZE; j++)
  {
    c[i][j] = a[i][j]+b[i][j];
  }
}
```



```
for(j = 0; j < ARRAY_SIZE; j++)
{
  for(i = 0; i < ARRAY_SIZE; i++)
  {
    c[i][j] = a[i][j]+b[i][j];
  }
}
```

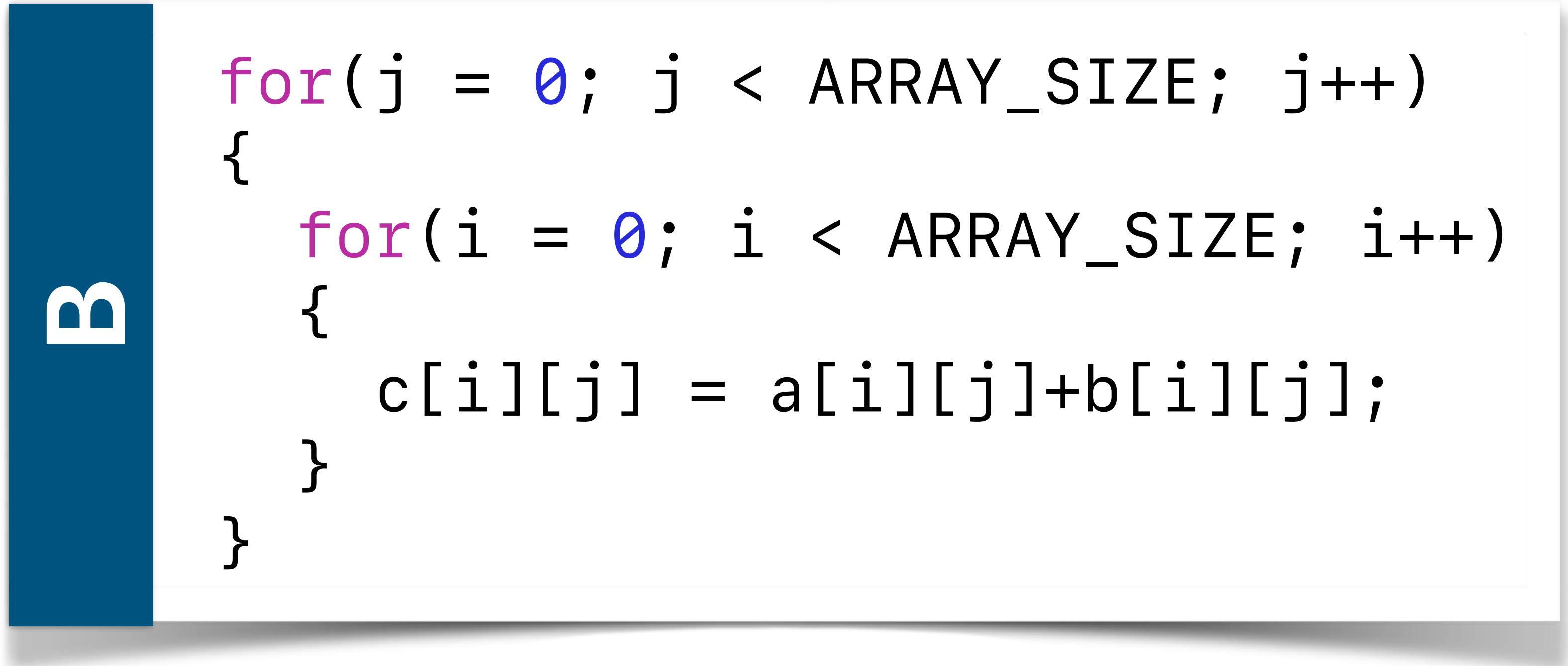
$O(n^2)$	Complexity	$O(n^2)$
Same	Instruction Count?	Same
Same	Clock Rate	Same
???	CPI	???

Demo — programmer & performance



A

```
for(i = 0; i < ARRAY_SIZE; i++)
{
  for(j = 0; j < ARRAY_SIZE; j++)
  {
    c[i][j] = a[i][j]+b[i][j];
  }
}
```



B

```
for(j = 0; j < ARRAY_SIZE; j++)
{
  for(i = 0; i < ARRAY_SIZE; i++)
  {
    c[i][j] = a[i][j]+b[i][j];
  }
}
```

$O(n^2)$	**Complexity**	$O(n^2)$
Same	**Instruction Count?**	**Same**
Same	**Clock Rate**	**Same**
Better	**CPI**	**Worse**

Announcement

- Reading quiz due next Tuesday before the lecture
  - We will drop two of your least performing reading quizzes
  - You have two shots, both unlimited time
- Assignment #1 released
  - We typically give you two weeks to work on an assignment
  - We never allow late submission and we will never have deadline extension
  - Due on 4/20
- Assignment #0 due on 4/13
- Check our website for slides, eLearn for quizzes/assignments, piazza for discussions
- Youtube channel for lecture recordings: https://www.youtube.com/c/ProfUsagi/playlists

Computer
Science &
Engineering

203

つづく