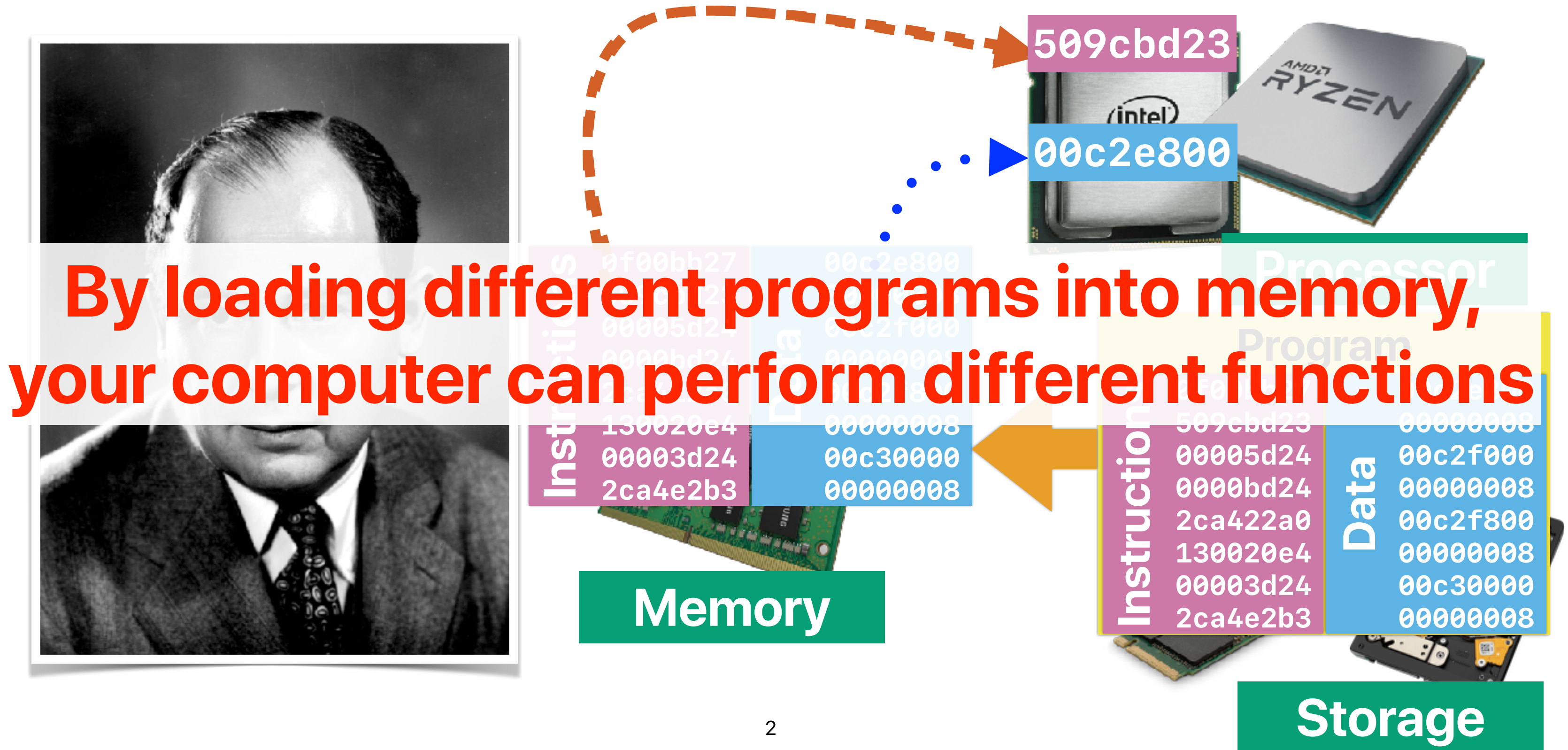


Performance (1): How Good Is "Good"?

Hung-Wei Tseng

Recap: von Neumann Architecture



Recap: Start with this simple program in C

```
int A[] =  
{1,2,3,4,5,6,7,8,9,10,1,2,3,4  
,5,6,7,8,9,10};
```

Compiler

Contents of section .data:

0000	01000000	02000000	03000000	04000000
0010	05000000	06000000	07000000	08000000
0020	09000000	0a000000	01000000	02000000
0030	03000000	04000000	05000000	06000000
0040	07000000	08000000	09000000	0a000000

control flow

operations

logical operations

```
int main()  
{  
    int i=0, sum=0;  
    for(i = 0; i < 20; i++)  
    {  
        sum += A[i];  
    }  
    return 0;  
}
```

memory access

arithmetic operations

main:

.LFB0:

endbr64

pushq %rbp

movq %rsp, %rbp

movl \$0, -8(%rbp)

movl \$0, -4(%rbp)

movl \$0, -8(%rbp)

jmp .L2

.L3:

movl -8(%rbp), %eax

cltq

leaq 0(,%rax,4), %rdx

leaq A(%rip), %rax

movl (%rdx,%rax),

%eax

addl %eax, -4(%rbp)

addl \$1, -8(%rbp)

.L2:

cmpl \$19, -8(%rbp)

jle .L3

movl \$0, %eax

popq %rbp

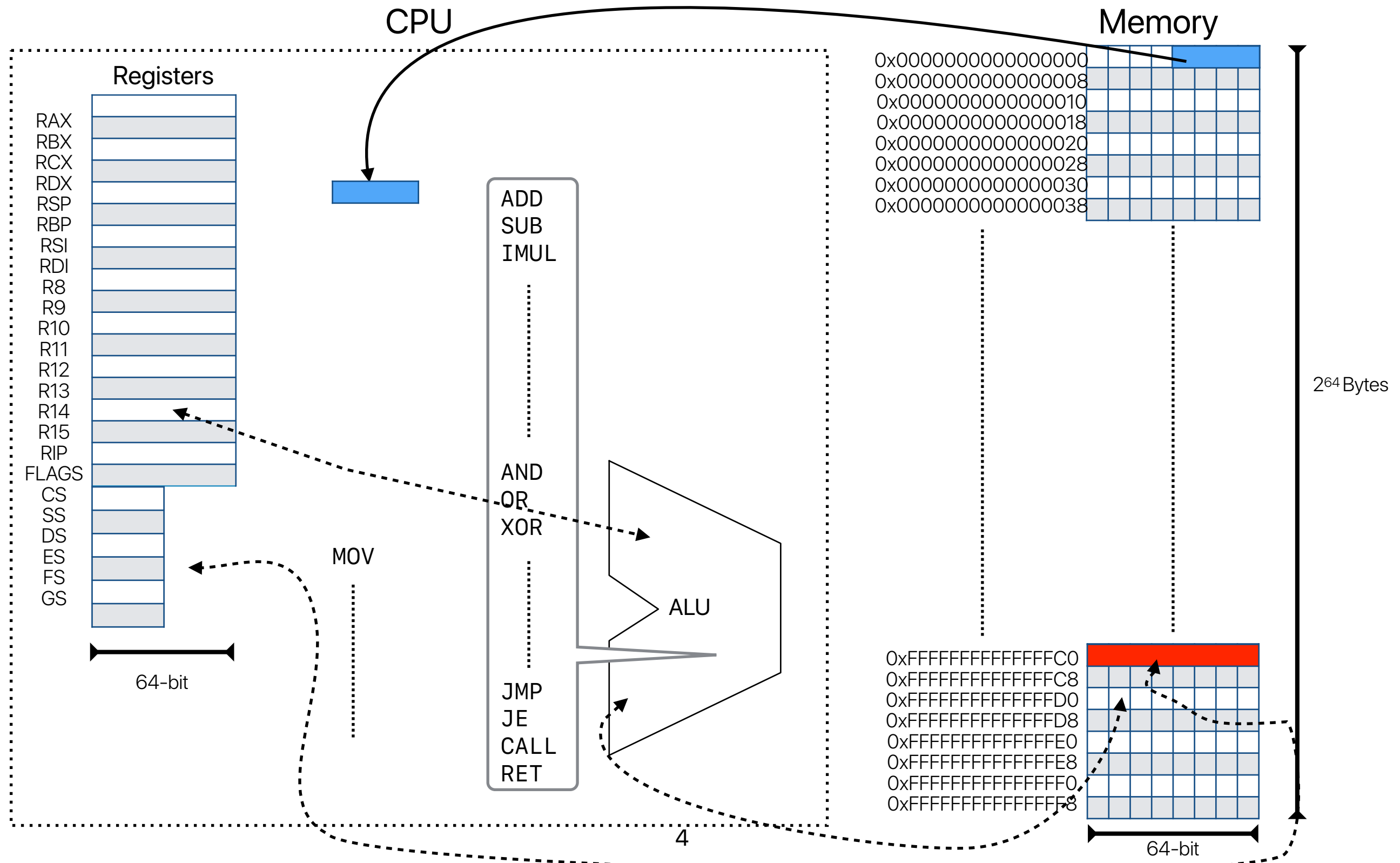
ret

Compiler

Contents of section .text:

0000	f30f1efa	554889e5	c745f800	000000c7
0010	45fc0000	0000c745	f8000000	00eb1e8b
0020	45f84898	488d1148	00000000	488d0500
0030	0000008b	04020145	fc8345f8	01837df8
0040	137edcb8	00000000	5dc3	

x86 ISA: the abstracted machine



Recap: Demo

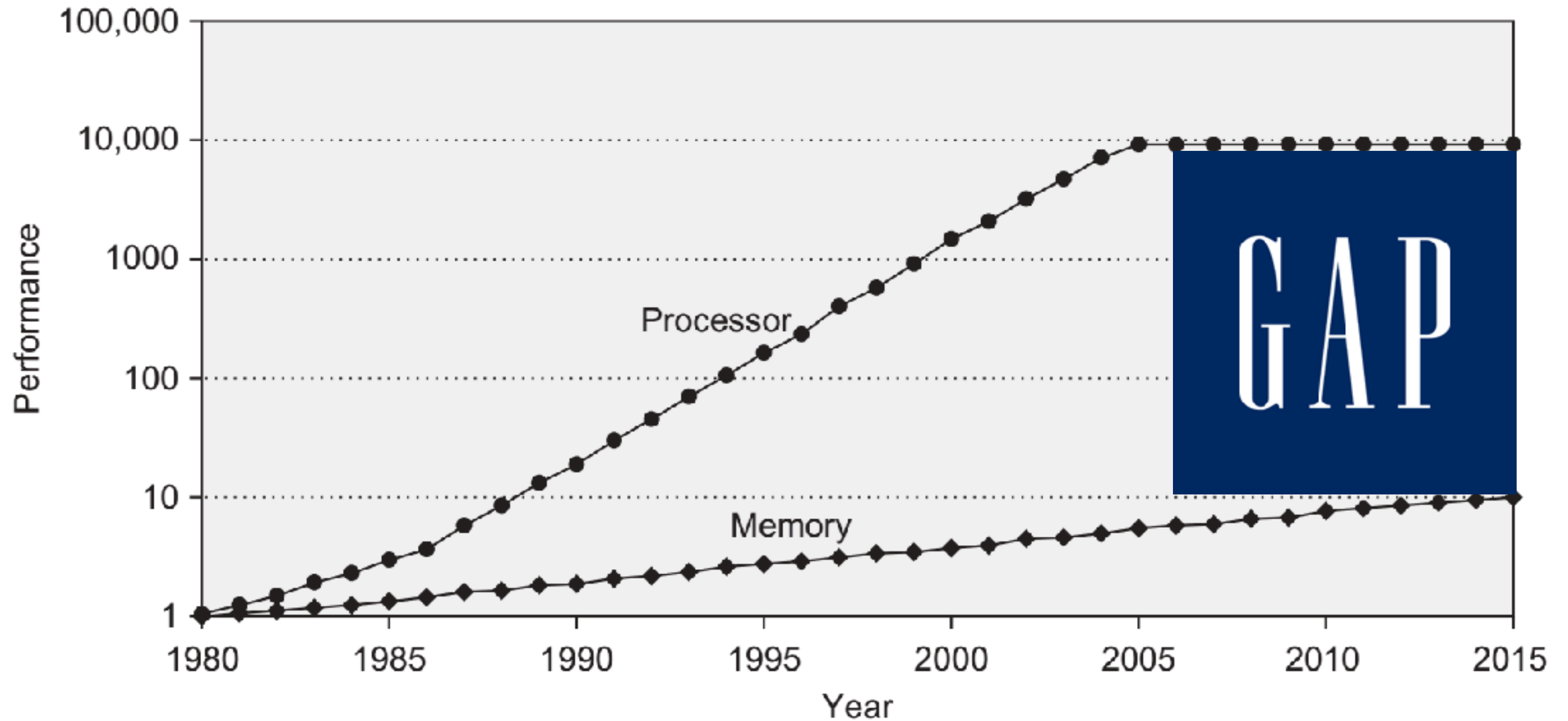
```
if(option)
    std::sort(data, data + arraySize);  $O(n \log_2 n)$ 

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

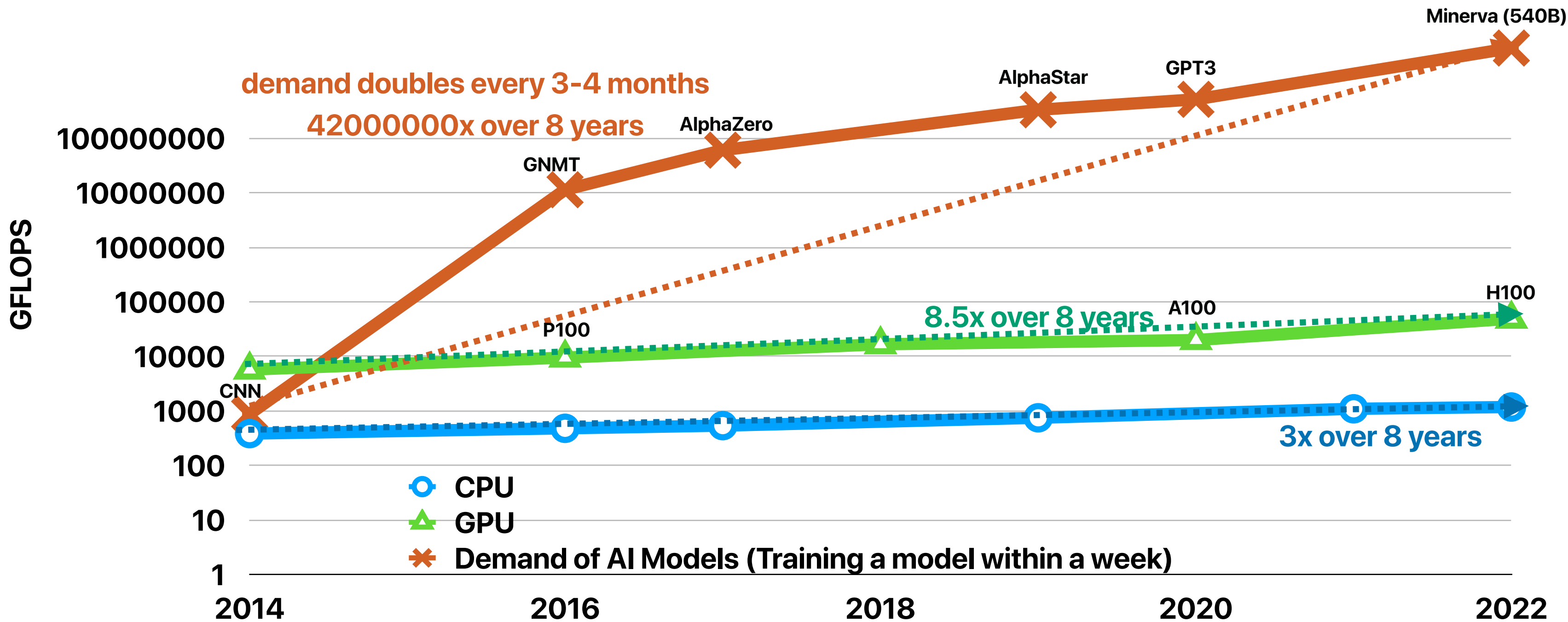
if option is set to 1: *$O(n \log_2 n)$*

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

Recap: Performance gap between Processor/Memory



Recap: Mis-matching AI/ML demand and general-purpose processing



<https://ourworldindata.org/grapher/artificial-intelligence-training-computation>

Where do you typically go for lunch on campus? Why?

Outline

- Definition of "Performance"
- The performance equation
- What affects each factor in "Performance Equation"

Best Nation

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

To unlock full rankings, SAT/A

SUMMARY ▾



Best Computer

Best Computer Science Schools

Ranked in 2022, part of [Best Science S](#)

Earning a graduate degree in computer
technology companies and colleges ar
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

443

Scho

Sc

Loca

Cit

All

Rank

Nat

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?

View other drafts

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.

Regenerate response

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 your UCSD email

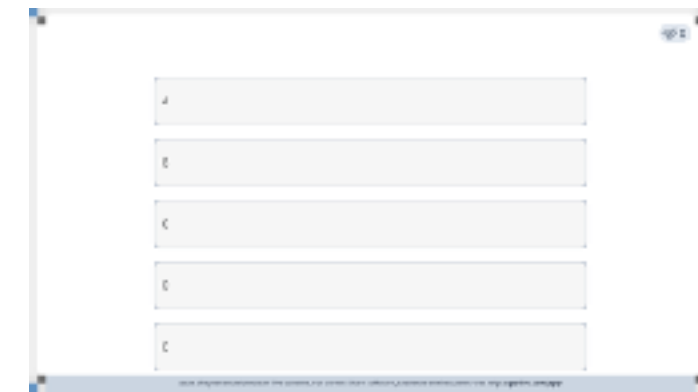
**Now, you have at least 90 seconds
to answer the question!**





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



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

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?**



CPU Performance Equation (X)

- Assume that we have an application composed with a total of **5,000,000,000** instructions, in which **20%** of them are "Type-A" instructions with an average **CPI of 4** cycles, **20%** of them are "Type-B" instructions with an average **CPI of 3** cycles and **the rest** instructions are "Type-C" instructions with average **CPI of 1** cycle. If the processor runs at **4 GHz**, how long is the execution time?

- A. 1.25 sec
- B. 2.5 sec
- C. 3.75 sec
- D. 7.5 sec
- E. 40 sec

A screenshot of a poll interface. It features five horizontal input boxes stacked vertically, each preceded by a small letter (A, B, C, D, E) in a light blue font. The boxes are currently empty, suggesting a multiple-choice or short-answer format for the poll.

CPU Performance Equation

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

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

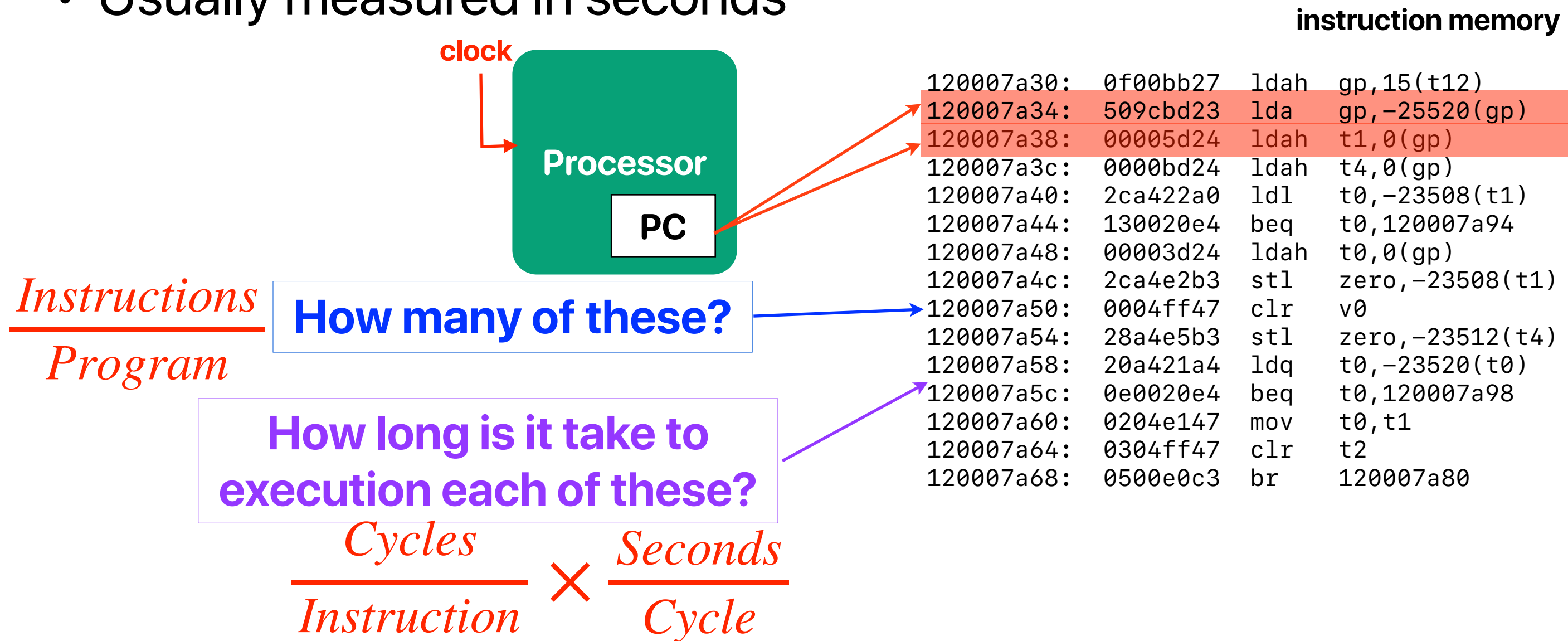
$$ET = IC \times CPI \times CT$$

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

Frequency(i.e., clock rate)

Execution Time

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



Performance Equation (X)

- Assume that we have an application composed with a total of **5,000,000,000** instructions, in which **20%** of them are "Type-A" instructions with an average **CPI of 4** cycles, **20%** of them are "Type-B" instructions with an average **CPI of 3** cycles and **the rest** instructions are "Type-C" instructions with average **CPI of 1** cycle. If the processor runs at **4 GHz**, how long is the execution time?

A. 1.25 sec

B. 2.5 sec

C. 3.75 sec

D. 7.5 sec

E. 40 sec

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

$$ET = (5 \times 10^9) \times (20\% \times 4 + 20\% \times 3 + 60\% \times 1) \times \frac{1}{4 \times 10^9} \text{ sec} = 2.5 \text{ sec}$$

average CPI



Performance equation (round 2)

- Consider the following c code snippet and x86 instructions implement the code snippet

C	x86
<pre>for(i = 0; i < count; i++) { s += a[i]; }</pre>	<pre>.L3: movslq (%rdi), %rdx addq \$4, %rdi addq %rdx, %tax cmpq %rcx, %rdi jne .L3</pre>

If (1) count is set to 1,000,000,000, (2) a memory instruction takes 4 cycles, (3) a branch/jump instruction takes 3 cycles, (4) other instructions takes 1 cycle on average, and (5) the processor runs at 4 GHz, how much time is it take to finish executing the code snippet?

- A. 0.5 sec
- B. 1 sec
- C. 2.5 sec
- D. 3.75 sec
- E. 4 sec

Performance equation (round 2)

- Consider the following c code snippet and x86 instructions implement the code snippet

C	x86
<pre>for(i = 0; i < count; i++) { s += a[i]; }</pre>	<pre>.L3: movslq (%rdi), %rdx addq \$4, %rdi addq %rdx, %tax cmpq %rcx, %rdi jne .L3</pre>

If (1) count is set to 1,000,000,000, (2) a memory instruction takes 4 cycles, (3) a branch/jump instruction takes 3 cycles, (4) other instructions takes 1 cycle on average, and (5) the processor runs at 4 GHz, how much time is it take to finish executing the code snippet?

A. 0.5 sec

B. 1 sec

C. 2.5 sec

D. 3.75 sec

E. 4 sec

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

$$ET = (5 \times 10^9) \times (20\% \times 4 + 20\% \times 3 + 60\% \times 1) \times \frac{1}{4 \times 10^9} \text{ sec} = 2.5 \text{ sec}$$

**total # of dynamic
instructions**

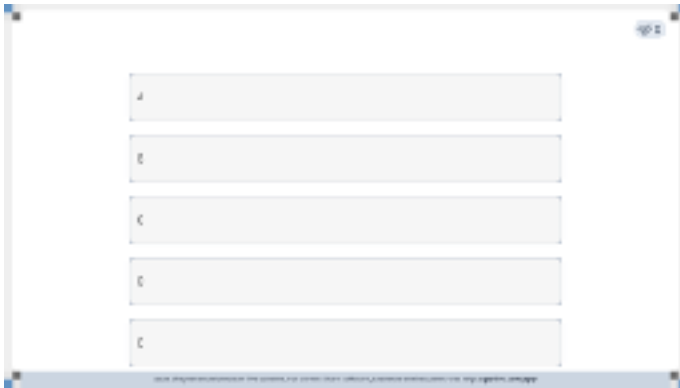
average CPI

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	Dynamic Instruction Count	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	4 GHz	5000000000	20%	4	20%	3	60%	1
Machine Y	6 GHz	5000000000	20%	6	20%	3	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

$$n = \frac{\textit{Execution Time}_X}{\textit{Execution Time}_Y}$$

- The speedup of Y over X

$$\textit{Speedup} = \frac{\textit{Execution Time}_X}{\textit{Execution Time}_Y}$$

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	4 GHz	5000000000	20%	4	20%	3	60%	1
Machine Y	6 GHz	5000000000	20%	6	20%	3	60%	1

A. 0.2

B. 0.25

C. 0.8

D. 1.25

E. No changes

$$ET_X = (5 \times 10^9) \times (20\% \times 4 + 20\% \times 3 + 60\% \times 1) \times \frac{1}{4 \times 10^9} \text{ sec} = 2.5 \text{ sec}$$

$$ET_Y = (5 \times 10^9) \times (20\% \times 6 + 20\% \times 3 + 60\% \times 1) \times \frac{1}{6 \times 10^9} \text{ secs} = 2 \text{ secs}$$

$$\text{Speedup} = \frac{\text{Execution Time}_X}{\text{Execution Time}_Y} = \frac{2.5}{2} = 1.25$$

What Affects Each Factor in Performance Equation



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

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];
    }
}
```

$O(n^2)$

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

Instruction Count?

Clock Rate

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];  
    }  
}
```

$O(n^2)$

Same

Same

???

Complexity

Instruction Count?

Clock Rate

CPI

$O(n^2)$

Same

Same

???

Use “performance counters” to figure out!

- Modern processors provides performance counters
 - instruction counts
 - cache accesses/misses
 - branch instructions/mis-predictions
- How to get their values?
 - You may use “perf stat” in linux
 - You may use Instruments —> Time Profiler on a Mac
 - Intel’s vtune — only works on Windows w/ intel processors
 - You can also create your own functions to obtain counter values

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];
    }
}
```

$O(n^2)$

Same

Same

Better

B

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

Complexity

Instruction Count?

Clock Rate

CPI

$O(n^2)$

Same

Same

Worse

Programmers can also set the cycle time

<https://software.intel.com/sites/default/files/comment/1716807/how-to-change-frequency-on-linux-pub.txt>

```
=====
Subject: setting CPU speed on running linux system
```

If the OS is Linux, you can manually control the CPU speed by reading and writing some virtual files in the "/proc"

1.) Is the system capable of software CPU speed control?

If the "directory" /sys/devices/system/cpu/cpu0/cpufreq exists, speed is controllable.

-- If it does not exist, you may need to go to the BIOS and turn on EIST and any other C and P state control and vi

2.) What speed is the box set to now?

Do the following:

```
$ cd /sys/devices/system/cpu
```

```
$ cat ./cpu0/cpufreq/cpuinfo_max_freq
```

```
3193000
```

```
$ cat ./cpu0/cpufreq/cpuinfo_min_freq
```

```
1596000
```

3.) What speeds can I set to?

Do

```
$ cat /sys/devices/system/cpu/cpu0/cpufreq/scaling_available_frequencies
```

It will list highest settable to lowest; example from my NHM "Smackover" DX58SO HEDT board, I see:

```
3193000 3192000 3059000 2926000 2793000 2660000 2527000 2394000 2261000 2128000 1995000 1862000 1729000 159600
```

You can choose from among those numbers to set the "high water" mark and "low water" mark for speed. If you set "h:

4.) Show me how to set all to highest settable speed!

Use the following little sh/ksh/bash script:

```
$ cd /sys/devices/system/cpu # a virtual directory made visible by device drivers
```

```
$ newSpeedTop=`awk '{print $1}' ./cpu0/cpufreq/scaling_available_frequencies`
```

```
$ newSpeedLow=$newSpeedTop # make them the same in this example
```

```
$ for c in ./cpu[0-9]* ; do
```

```
> echo $newSpeedTop >${c}/cpufreq/scaling_max_freq
```

```
> echo $newSpeedLow >${c}/cpufreq/scaling_min_freq
```

```
> done
```

```
$
```

5.) How do I return to the default - i.e. allow machine to vary from highest to lowest?

Edit line # 3 of the script above, and re-run it. Change the line:

```
$ newSpeedLow=$newSpeedTop # make them the same in this example
```

```
To read
```

Announcement

- Reading quiz due next Monday before the lecture
 - We will drop two of your least performing reading quizzes
 - You have two shots, both unlimited time
- Check our website for slides, Canvas for quizzes/assignments, piazza for discussions
- Youtube channel for lecture recordings:
<https://www.youtube.com/c/ProfUsagi/playlists>

Computer Science & Engineering

142

つづく

