

Welcome!

CSI3102-02: Architecture of Computers
(컴퓨터아키텍처)

Youngsok Kim (김영석)



Teaching Staff

- **Lecturer:** Youngsok Kim (김영석)
 - D703, Eng. Hall #4
- **TAs:** Seil Lee (이세일), Sungchul Lee (이성철)
- Please ask questions through **LearnUs Q&A board**.
 - <https://ys.learnus.org/mod/ubboard/view.php?id=3534021>
 - The TAs and I will only respond to the questions uploaded to the Q&A board.
 - If you believe your post should be kept private, use the private post feature of the Q&A board.



Computer Architecture



Have You Ever Wondered...

- Why your desktop software doesn't run on phones?
 - x86/AMD64 vs. ARM
- Why different CPUs achieve different performance?
 - Intel CPUs used to be faster than those of AMD.
- Why GPUs & NPUs exist despite having CPUs?
 - Target different types of applications
- ... and many more!



What's Under the Hood? (1/3)

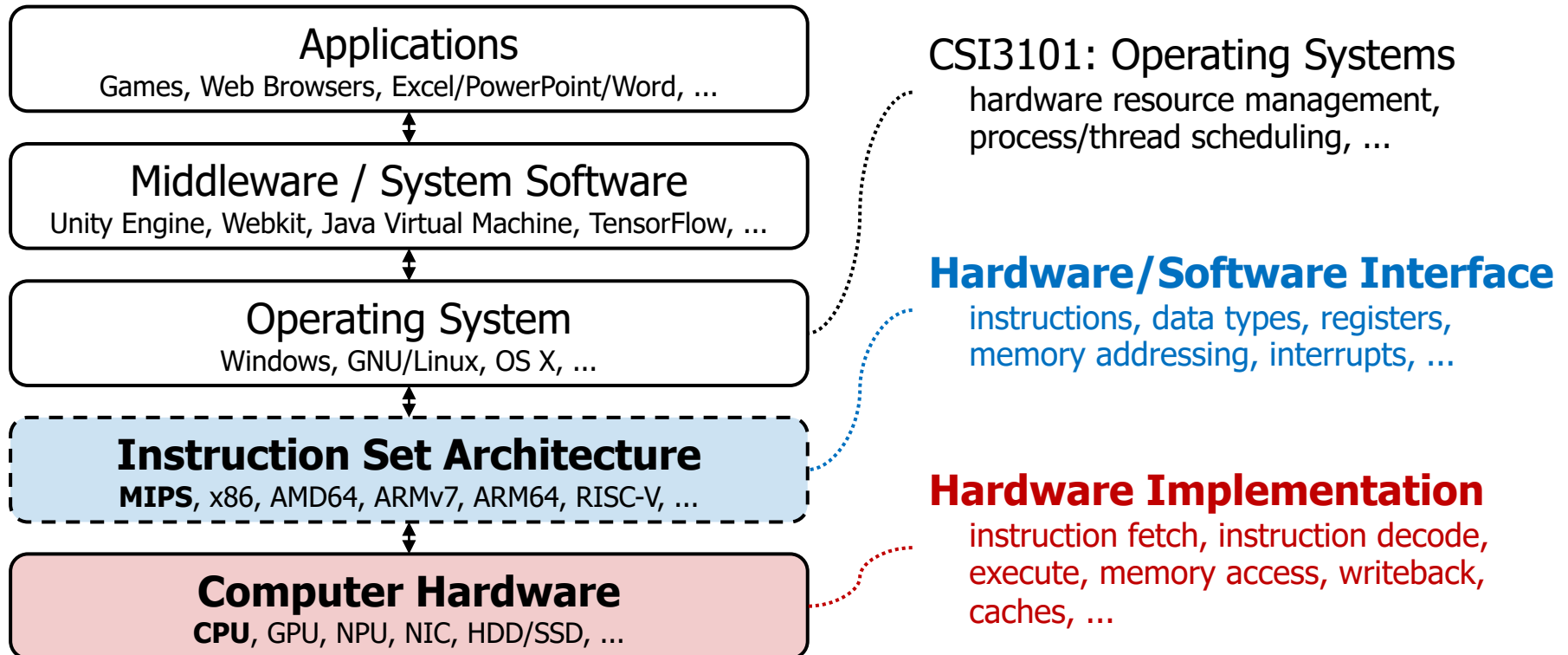
- Let's suppose you run C/C++ code on a CPU.
 1. You compile your code with a compiler (e.g., gcc/g++).
 2. You execute the executable binary generated by the compiler on an Operating System (OS).
 3. The OS schedules a piece of machine code of the executable binary on an underlying CPU core.
- Q: What are the **requirements** of hardware to execute the software code scheduled by the OS?
- A: We define a hardware-software interface called an **Instruction Set Architecture (ISA)**!

What's Under the Hood? (2/3)

- Q: Given an ISA (e.g., x86, ARMv8), how should the underlying CPU core **implement** the ISA?
- A: The CPU core should fetch an instruction from memory, decode the instruction, and execute the instruction!
- Q: How to **optimize** an implementation of an ISA?
- A: Pipelining for throughput,
caches for memory access latency,
...

What's Under the Hood? (3/3)

- Your computer is more complex than you think :)



Computer Architecture

- A set of rules/methods which describe the functionality, organization, and implementation of computer systems
- A **common** interface between hardware & software
 - e.g., data types (e.g., two's complement, big/little endian),
of registers and their purposes,
32/64/128 bits for memory addressing
- The interface is **agreed** btw. hardware & software!
 - e.g., Windows & x86-64, Linux & ARMv8



About This Course



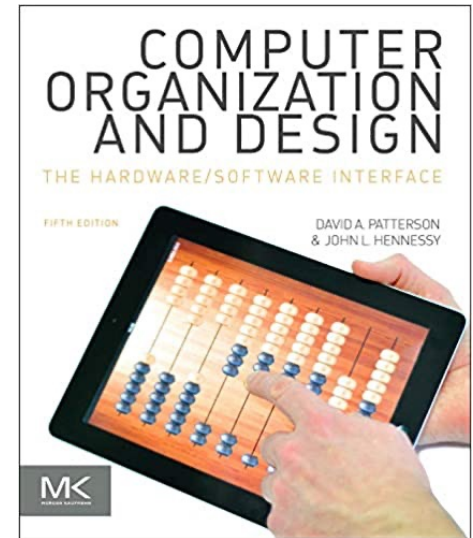
Course Organization

- Two tracks: principles & practice
- **Track 1: Principles (Textbook Contents)**
 - How to implement a processor?
 - What aspects of the processor you should consider for your target applications.
- **Track 2: Practice (Assembly/C++ Programming)**
 - You will learn how to write code in MIPS assembly.
 - You will implement a soft CPU implementing the assembly using C++.



Track 1: Principles

- **Computer Organization and Design:**
The Hardware/Software Interface
(5th Edition)



- **Keyword #1:** **Computer Organization & Design**
 - What forms a computer and how do we design it?
- **Keyword #2:** **The Hardware/Software Interface**
 - How do we define an ISA, and how do we implement it?
- **Keyword #3:** MIPS
 - The ISA we will focus on.

Track 2: Practice

- Write programs in MIPS assembly
 - Much lower level than C/C++
 - Start with pseudo-assembly, followed by machine code
- Implement the classic 5-stage CPU pipeline
 - You will implement a soft CPU using C/C++.
 - In 2020, we used VHDL, a hardware description language.
 - The students suffered a lot, so we fell back to Python in 2021.
 - However, Python is too high-level to implement CPUs :(
 - You will later optimize the performance of the soft CPU.
 - Implement caches, branch prediction, out-of-order execution, etc.



Reminders



This Course Will Be Hard!

- The textbook contains more detailed contents.
 - 6 chapters + 1 appendix
 - MIPS assembly programming, CPU pipeline, ...
- 5+ MIPS/C++ programming assignments
 - Expect one assignment per two weeks
- Start studying the textbook and programming assignments **early!**

Grading Policy

- You will get an F if
 - You miss more than 1/3 of the semester.
 - You skip an exam.
 - You copy other students' assignments.
- No late submissions will be accepted.
 - Start your assignments early!
- Check your (submitted) submissions in advance.
 - The non-working submissions will get 0 point.



Questions?

- Post any question to the LearnUs Q&A board
 - We will not respond to questions from the other sources.
 - e.g., e-mails, offline visits, ...
 - This allows us to share your questions to all the students.

& welcome to the world of computer architecture :)