

# **CS-200**

# **Computer Architecture**

—

## **Part 0. Introduction**

Paolo Ienne  
<paolo.ienne@epfl.ch>

# Who's Who

**Lecturer:** Paolo lenne

## Teaching Assistants



Louis Coulon

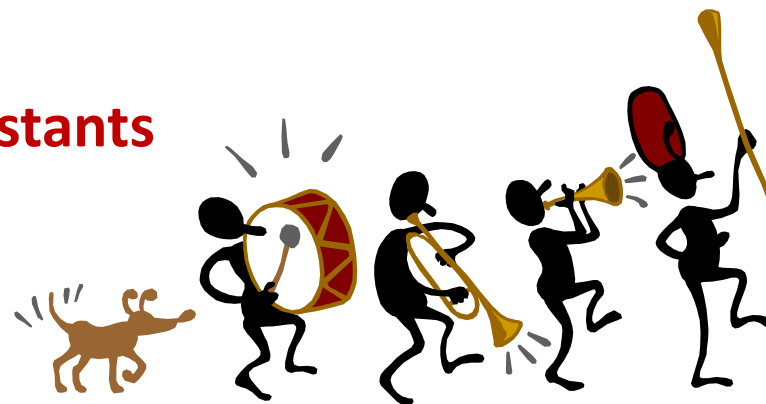


Buğra Eryılmaz



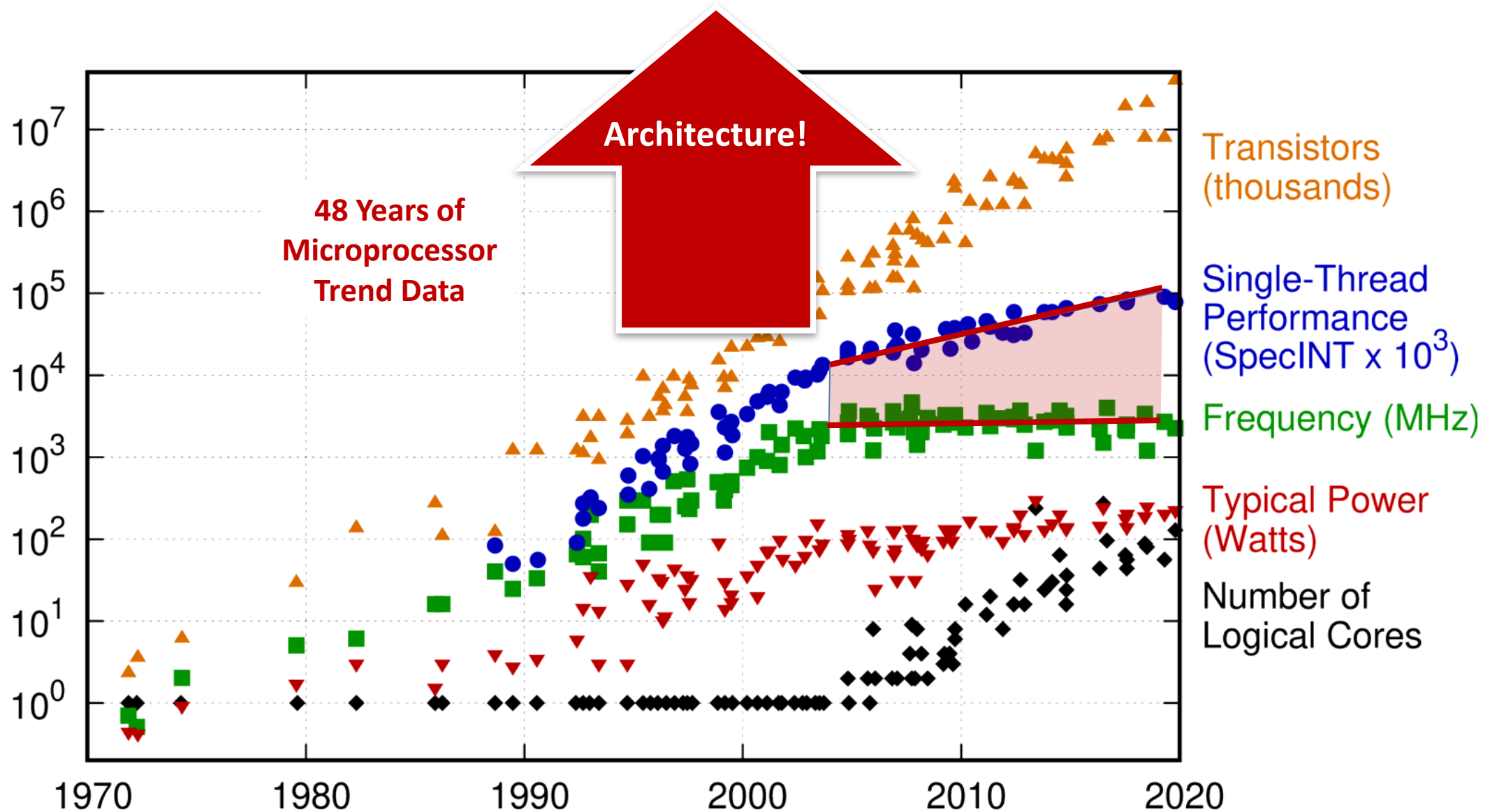
Tz-Ching Yu

## Student Assistants





# ...Especially Today!



# Content of CS-200 (1/2)

- **Part I: Processors and ISA**

- What is a processor? How can we design one? How do programs look like when they are executed?

- **Part II: I/Os and Exceptions**

- What is around a processor to make a full computer? How the processor exchanges information with the rest of the world?

- **Part III: Memory Hierarchy**

- Processors are fast and memory is slow—how can one combine the two? How can one protect the data of users in memory?

# Content of CS-200 (2/2)

- **Part IV: Instruction-Level Parallelism**

- What makes a good processor? How real processors achieve ever increasing performances?

- **Part V: Multiprocessors**

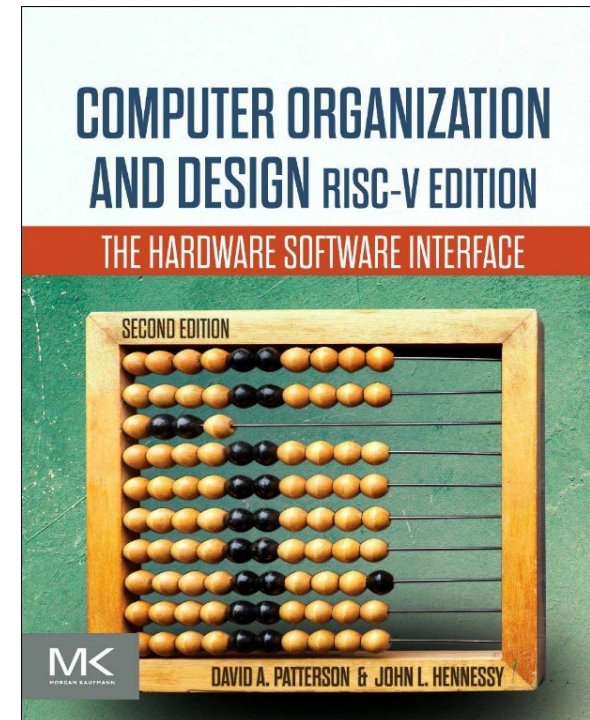
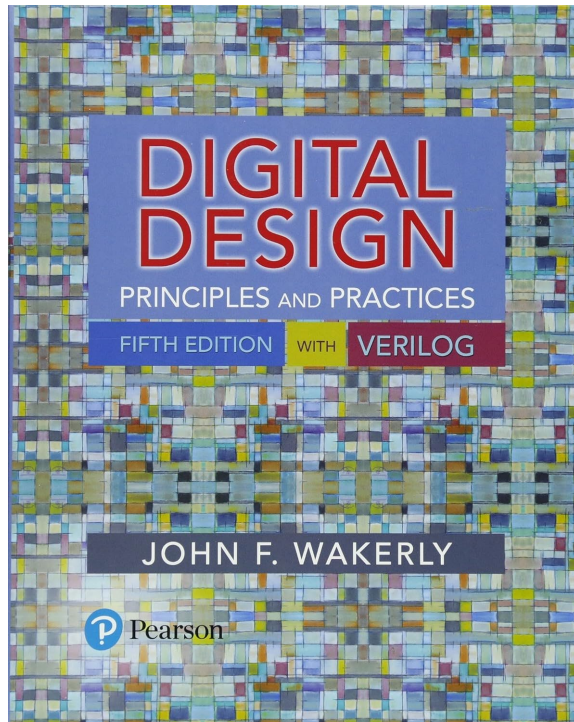
- What are the basic challenges of connecting many processors together? What changes from a single processor system?

- **Part VI: Rudiments of Hardware Security**

- How can a hacker exploit what we have built in the previous parts to attack a system? How physics helps jeopardizing security?

# Literature

- Everything recommended for **CS-173** applies here too
- In particular, **Wakerly** and **COD (RISC-V)**:



# Moodle

- Moodle of this course:

<https://moodle.epfl.ch/course/view.php?id=18287>

- Everything is or will be there or linked there:
  - Schedule, slides, videos, forum, exercise book, labs, instructions, etc.

# Slides & Video Recordings

- Slides are all already available on **Moodle**
- **Videos** are all already available on

[https://mediaspace.epfl.ch/playlist/dedicated/55022/0\\_78dtdaar/](https://mediaspace.epfl.ch/playlist/dedicated/55022/0_78dtdaar/)

linked on Moodle

- Lectures are **not streamed** this year

# Room GR C0 01

- **BCH 2201** may not have enough seats for everyone on **Fridays**
- The lecture will also be **streamed live in GR C0 01**
- If BCH 2201 proves to have sufficient capacity, we may **not continue using the overflow room**
- Therefore, please **come to BCH 2201 first** and use GR C0 01 only if there are no available seats
- After all, if you only want to see me on a screen, the **recordings are always there!** 😊

# CS-200 – Computer Architecture

## Fall 2025

Week	Date	Time	Room	Lecture	Lab	Deadline
	Mon 08.09.25	1pm - 3pm	INF1 - INF3			
	Wed 10.09.25	10am - noon	SG1	Introduction + 1a. Instruction Set Architecture (ISA Reminder, Assembly Language, Compilers)		
1	Wed 10.09.25	1pm - 3pm	INF1 - INF3		Introduction to the infrastructure	
	Fri 12.09.25	10am - noon	BCH 2201	1b. Instruction Set Architecture (Branches, Functions, and		Friday 12.09: Lab A (published)
	Mon 15.09.25	1pm - 3pm	INF1 - INF3		Lab A (Game of Life)	
2	Wed 17.09.25	10am - noon	SG1	1c. Instruction Set Architecture (Memory and Addressing		
	Wed 17.09.25	1pm - 3pm	INF1 - INF3		Lab A (Game of Life)	
	Fri 19.09.25	10am - noon	BCH 2201	1c. Instruction Set Architecture (Memory and Addressing		
	Mon 22.09.25					
	Wed 24.09.25	10am - noon	SG1	1e. Instruction Set Architecture (Arithmetic)		
3	Wed 24.09.25	1pm - 3pm	INF1 - INF3			
	Fri 26.09.25	10am - noon	BCH 2201	1e. Instruction Set Architecture (Arithmetic; cont'd) + 2a. Processor, I/Os, and Exceptions (Multicycle Processor)		
	Mon 29.09.25	1pm - 3pm	INF1 - INF3		Lab A (Game of Life)	
4	Wed 01.10.25	10am - noon	SG1	2a. Processor, I/Os, and Exceptions (Multicycle Processor; cont'd) + 2b. Processor, I/Os, and Exceptions (Inputs and Outputs)		
	Wed 01.10.25	1pm - 3pm	INF1 - INF3		Lab A (Game of Life)	
	Fri 03.10.25	10am - noon	BCH 2201	2b. Processor, I/Os, and Exceptions (Inputs and Outputs; cont'd) + 2c. Processor, I/Os, and Exceptions (Interrupts)		
	Mon 06.10.25	1pm - 3pm	INF1 - INF3		Lab A (Game of Life)	
5	Wed 08.10.25	10am - noon	SG1	2d. Processor, I/Os, and Exceptions (Exceptions)		
	Wed 08.10.25	1pm - 3pm	INF1 - INF3		Lab A (Game of Life)	
	Fri 10.10.25	10am - noon	BCH 2201	2d. Processor, I/Os, and Exceptions (Exceptions; cont'd)		Sunday 12.10 @ 23:59: Lab A (final)
	Mon 13.10.25	1pm - 3pm	INF1 - INF3		Exercises on I/Os and Exceptions	Monday 13.10: Lab B (published)
	Wed 15.10.25			3a. Memory Hierarchy (Caches)		
6	Wed 15.10.25	1pm - 3pm	INF1 - INF3		Lab B.1 (RISC-V Multicycle Processor): ALU	
	Fri 17.10.25					
	Mon 27.10.25	1pm - 3pm	INF1 - INF3		Lab B.2 (RISC-V Multicycle Processor): Core	
7	Wed 29.10.25	10am - noon	SG1	3a. Memory Hierarchy (Caches; cont'd)		
	Wed 29.10.25	1pm - 3pm	INF1 - INF3		Lab B.2 (RISC-V Multicycle Processor): Core	
	Fri 31.10.25	10am - noon	BCH 2201	3b. Memory Hierarchy (Virtual Memory)		Sunday 02.11 @ 23:59: Lab B.1 (partial)
	Mon 03.11.25	1pm - 3pm	INF1 - INF3		Lab B.2 (RISC-V Multicycle Processor): Core	
8	Wed 05.11.25	10am - noon	SG1	3c. Memory Hierarchy (Simple Virtual Memory Examples)		
	Wed 05.11.25	1pm - 3pm	INF1 - INF3		Exercises on Memory Hierarchy	
	Fri 07.11.25	10am - noon	BCH 2201	3d. Memory Hierarchy (Parallelism (Basic Pipelining))		
	Mon 10.11.25	1pm - 3pm	INF1 - INF3		Lab B.3 (RISC-V Multicycle Processor): Memory, Peripherals, and Integration	
9	Wed 12.11.25					
	Wed 12.11.25	10am - 1:15pm	TBA			
	Fri 14.11.25	10am - noon	BCH 2201	4c. Instruction-Level Parallelism (Pipelining)		
	Mon 17.11.25	1pm - 3pm	INF1 - INF3		Lab B.3 (RISC-V Multicycle Processor): Memory, Peripherals, and Integration	
10	Wed 19.11.25	10am - noon	SG1	4c. Instruction-Level Parallelism (Pipelining; cont'd)		
	Wed 19.11.25	1pm - 3pm	INF1 - INF3		Lab B.3 (RISC-V Multicycle Processor): Memory, Peripherals, and Integration	
	Fri 21.11.25	10am - noon	BCH 2201	4d. Instruction-Level Parallelism (Examples of Scheduling)		Sunday 23.11 @ 23:59: Lab B.2 (partial)
	Mon 24.11.25	1pm - 3pm	INF1 - INF3		Lab B.3 (RISC-V Multicycle Processor): Memory, Peripherals, and Integration	Monday 24.11: Lab C (published)
	Wed 26.11.25	10am - noon	SG1	4d. Instruction-Level Parallelism (Examples of Scheduling; cont'd)		
11	Wed 26.11.25	1pm - 3pm	INF1 - INF3		Lab C.1 (RISC-V Interrupts): Adding Interrupt Support	
	Fri 28.11.25	10am - noon	BCH 2201	4e. Instruction-Level Parallelism (Examples of Scheduling; cont'd) + 4f. Instruction-Level Parallelism (Besides and Beyond Superscalars)		Sunday 30.11 @ 23:59: Lab B (final)
	Mon 01.12.25	1pm - 3pm	INF1 - INF3		Lab C.1 (RISC-V Interrupts): Adding Interrupt Support	
12	Wed 03.12.25	10am - noon	SG1	4f. Instruction-Level Parallelism (Besides and Beyond Superscalars; cont'd)		
	Wed 03.12.25	1pm - 3pm	INF1 - INF3		Exercises on Instruction-Level Parallelism	
	Fri 05.12.25	10am - noon	BCH 2201	4f. Instruction-Level Parallelism (Besides and Beyond Superscalars; cont'd) + 4g. Instruction-Level Parallelism (Intel x86 and ARM)		
	Mon 08.12.25	1pm - 3pm	INF1 - INF3		Lab C.2 (RISC-V Interrupts): Interrupt Controller and Integration	
13	Wed 10.12.25	10am - noon	SG1	5a. Multiprocessors (Cache Coherence)		
	Wed 10.12.25	1pm - 3pm	INF1 - INF3		Lab C.2 (RISC-V Interrupts): Interrupt Controller and Integration	
	Fri 12.12.25	10am - noon	BCH 2201	5a. Multiprocessors (Cache Coherence; cont'd) + 5b. Multiprocessors (Examples of Cache Coherence)		Sunday 14.12 @ 23:59: Lab C.1 (partial)
	Mon 15.12.25	1pm - 3pm	INF1 - INF3		Lab C.2 (RISC-V Interrupts): Interrupt Controller and Integration	
14	Wed 17.12.25	10am - noon	SG1	5c. Multiprocessors (Memory Consistency)		
	Wed 17.12.25	1pm - 3pm	INF1 - INF3		Exercises on Multiprocessors	
	Fri 19.12.25	10am - noon	BCH 2201	6. Hardware Security		Sunday 21.12 @ 23:59: Lab C (final)

**Schedule irregularities**

**Five exercise sessions**

**Midterm!**

**Important lab deadlines**

**Deadline**

**Exercises on Processors & ISA**

**Sunday 30.11 @ 23:59: Lab B (final)**

**Sunday 14.12 @ 23:59: Lab C.1 (partial)**

# Grading

- Midterm test (**3h**):
  - **Wednesday 12<sup>th</sup> November, 10:15am-13:15pm**, rooms **TBA** (the afternoon lab is cancelled)
  - On **Processors and ISA, I/Os & Exceptions**, and **Memory Hierarchy**
  - Books, notes, exercises, etc. are **not permitted**
  - Essential quick references will be **distributed** (e.g., ISA)
  - The midterm test counts for **35% of the final grade**
- Final exam in the exam session (**3h**):
  - Will be fixed by SAC in due time
  - On **Instruction-Level Parallelism** and **Multiprocessors**
  - Same rules as above for the Midterm
  - The final exam counts for **35% of the final grade**
- Labs:
  - **Three labs (A, B, C)** composed of various sessions each; each lab counts for **10% of the final grade**

# Labs

- **Graded** automatically with **anti-plagiarism check**
- Must be done **individually**
- **Deadlines** on the schedule
- **Verilog** code must follow our **guidelines** or (i) **no support** during the lab sessions and (ii) **50% grade lost!**
- They need a variety of tools, so **use the preconfigured VM**
  - **First lab is a tutorial** (later today)
  - Everything is open source but **you are on your own** if you want to install on your computer (rather easy on Linux, harder elsewhere)

# Lab Submissions

- Submission deadlines marked **“final”** are **graded**
  - **Three final submissions** for the three labs A, B, and C
  - You can submit **twice** and the **last one** counts for the grade
  - **Limited diagnostic information**—essentially, what passed and what not
- Before, there are submission deadlines marked **“partial”** (e.g., not the complete lab B but only part B.1)
  - You can submit **twice**
  - They are **not graded** but should give you a sense of what is expected
  - Reported **diagnostic information on the first error** and some **failure stats**
  - Do not expect full debugging info—**testing and debugging is your job!**

# Lab A Submission

```
===== Testing summary =====
clear_leds:
    clear_leds on a random screen: passed
Test clear_leds: 1 out of 1 passed -> Score: 2.0 / 2
set_pixel:
    set_pixel setting some pixel on empty screen: passed
    set_pixel setting some pixel on non empty screen: passed
.
.
    Changes the number of steps in the INIT state with button 1 set: passed
    Changes the number of steps in the INIT state with button 0 set: passed
    Changes the number of steps in the INIT state with no button set: passed
Test change_steps: 8 out of 8 passed -> Score: 5.0 / 5
increment_seed:
    Increments the seed from a random number: failed
    Increments the seed from a random number: failed
Test increment_seed: 0 out of 2 passed -> Score: 0.0 / 5
update_state:
    INIT, button 2: passed
    INIT, button 1: passed
    INIT, button 0: passed
    INIT, button jr: passed
.
.
get_input:
    random button pressed: passed
Test get_input: 1 out of 1 passed -> Score: 5.0 / 5
decrement_step:
    decrement step, PAUSED RUN: passed
    decrement step, RUNNING RUN: passed
    decrement step, PAUSED INIT: passed
    decrement step, PAUSED RAND: passed
Test decrement_step: 4 out of 4 passed -> Score: 5.0 / 5
reset_game:
    reset_game from a random memory state: failed
Test reset_game: 0 out of 1 passed -> Score: 0.0 / 5
===== Total score: 63.05 / 100 =====
Note that 100% score corresponds to 80% of the maximum possible score on the
project. For the remaining 20% you will need to make a successful live
demonstration of the game to the Teaching Assistants.
```

Several checks  
in each unit test

Details on the failing checks

20% of the grade for a  
live demonstration

# Labs B & C

## “Partial” Submission

```
Running lint on Verilog files...
[ERROR] Lint failed for verilog/buttons.v
[ERROR] Lint failed for verilog/csr.v
Lint failed for one or more Verilog files

=====

Compiling testbench: testbench/tb_buttons.v
Compilation successful for testbench/tb_buttons.v
Running testbench: testbench/tb_buttons.v

First error encountered:
Error in test case 3 at time 160000:
  Expected: rdata_o = 0x00ff03ff, irq_o = 1
  Got:      rdata_o = 0x000003ff, irq_o = 1
3 tests failed

=====

Compiling testbench: testbench/tb_csr.v
Compilation successful for testbench/tb_csr.v
Running testbench: testbench/tb_csr.v

All tests passed

=====
```

Lint fails: no details because you can run it yourself

Details on the first error of each testbench

More errors only counted

Same for all testbenches included  
in the “partial” submission

# Labs B & C

## “Final” Submission

```
Running lint on Verilog files...  
[ERROR] Lint failed for verilog/buttons.v  
[ERROR] Lint failed for verilog/csr.v  
Lint failed for one or more Verilog files
```

```
=====  
Compiling testbench: testbench/tb_buttons.v  
Running testbench: testbench/tb_buttons.v
```

```
3 tests failed
```

```
=====  
Compiling testbench: testbench/tb_csr.v  
Running testbench: testbench/tb_csr.v
```

```
All tests passed
```

```
=====  
=====  
Final grade = 25.0 / 100  
=====
```

**Linter fails: no details because you can run it yourself**

**Only fail/pass info per testbench**

**Final grade (halved because linter failed)**

# Labs

(if you are taking CS-200 again)

- Must do the labs **again!**



# Q&A Forum

- Q&A Forum on **Ed Discussion**
- Provides **answers to FAQs** and a way to get help efficiently, from both TAs and colleagues
- Please **read the rules** before posting (no posting of solutions, “netiquette”, etc.)

# Exercise Book

- Available soon on **Moodle**
- A collection of **old exam questions (translated to RISC-V and Verilog)** with full solutions
- **Best way to assess if you are prepared for the tests!**
- Some of the in-class exercises and lab-session exercises are also (modified) exam questions

# On Your Mark!... Get Set!... Go!

- Any info missing? Ask now...

