Course Intro & Objectives

CS 2110
Computer Organization

An introduction to basic computer hardware, machine language, assembly language, and C programming.

Instructor

- Caleb Southern (sections A,B)
  - CCB 220
  - caleb.southern@gatech.edu
  - Include “2110” in the subject line
- Dan Forsyth (section C)
  - CCB 242
  - dan.forsyth@cc.gatech.edu
  - Include “2110” in the subject line

TA's

- Shawn Wahi (Head TA)

Office Hours and Contact Information on the Canvas CS2110 Home Page

Objectives

- To understand the structure and operation of a modern computer from the ground up.
- Understand basic hardware concepts: digital circuits, gates, bits, bytes, number representation
- Understand the Von Neumann model and the structure and operation of a basic datapath

Objectives

- Structure and function of machine language instructions
- Structure and function of a symbolic assembly language
- Basic concepts of computer systems such as the runtime stack, simple I/O devices
- Introduce the C language with particular emphasis on the underlying assembly and machine language as well as interaction with hardware.

From the point of continuity
(Kishore Ramachandran)

This sounds like a lot of work!

"Can't we make this fun, too?"

Classes

- **Lecture**
  - **Tue/Thu**
    - **Section A – 12:30pm – 1:45pm**
    - **Section B – 2:00pm – 3:15pm**
    - **Section C – 3:30pm – 4:45pm**
  - **There is an Attendance Quiz due for most lectures.**
- **Lab *(required)***
  - **Mon/Wed 3:30pm, 5:00pm, or 6:30pm**
  - **Mix of tutorial, practice, and evaluation**
  - **You are required to attend, attendance is taken.**
  - **This is not a supplemental help session**

Textbooks

- Required
  - *Introduction to Computing Systems, 3rd edition*: Patt & Patel
  - *The C Programming Language*: Kernighan & Ritchie
- Recommended (If you want a Linux book)
  - Mastering Linux: Paul Wang

Canvas

- We will be using the Canvas LMS*
- http://canvas.gatech.edu
- Used for
  - Assignment distribution,
  - Assignment turn-in (along with Gradescope),
  - Grade display

*LMS - Learning Management System

Assignment Values

Item	Number (approx.)	Totals
Homework	10	30%
Quizzes	4	20%
Timed Labs	4	20%
Lecture Attendance		2%
Lab Attendance		3%
Final Exam	1	25%
TOTAL		**100%**

Homework

- Usually every week
- Types of assignments
  - Logic Simulation
  - Machine Language programming
  - Assembly programming
  - C programming
- High-Level Collaboration is allowed on Homework
  - You can share ideas ***but not source code!***

Homework

- Even though it looks like each homework doesn't count for many points nothing could be further from the truth!
  - You cannot and will not do well in lab and on tests if you do not have a deep understanding of how the homework works and is coded.
  - Questions will be taken directly from things covered in homework.

Late Policy

You are responsible for turning in assignments on time. This includes allowing for unforeseen circumstances. You are also responsible for ensuring that what you turned in is what you meant to turn in. Each assignment will have an official due date. Homeworks, only, will be allowed a 24-hour grace period for a 25% penalty. After the grace period absolutely no credit will be given. Therefore, it is your responsibility to plan and ensure that you have backups, early safety submissions, etc.

Academic Misconduct

- Academic misconduct is taken very seriously in this class.
- Quizzes, timed labs and the final examination are individual work.
- Homework assignments may be collaborative, but only at a high level. In addition many homework assignments will be evaluated via demo or code review. During this evaluation, you will be expected to be able to explain every aspect of your submission. Homework assignments will also be examined using electronic computer programs to find evidence of unauthorized collaboration.

Academic Misconduct

- What is unauthorized collaboration? Each individual programming assignment should be coded by you. You may work with others, sharing ideas and even high-level pseudo-code, but each student must turn in their own version of the assignment.
- Submissions that are substantially identical will receive a zero and will be forwarded to the Dean of Students' Office of Academic Integrity. Submissions which are copies that have been superficially modified to conceal that they are copies will also be considered unauthorized collaboration.

Academic Misconduct

- You are expressly forbidden to supply a copy of your homework to another student via electronic means. If you supply an electronic copy of your homework to another student and they are charged with copying you will also be charged. This includes storing your code on any site which would allow other parties to obtain your code such as but not limited to public repositories, etc.

Final Exam

IF YOU ARE LATE OR MISS THE FINAL EXAM YOU RECEIVE A ZERO

Final

- The final exam is comprehensive

Need help?

- Ed Discussion
- TAs – You may attend any of the TA's office hours
- Instructor
- Dean of Students' Office

End of Semester

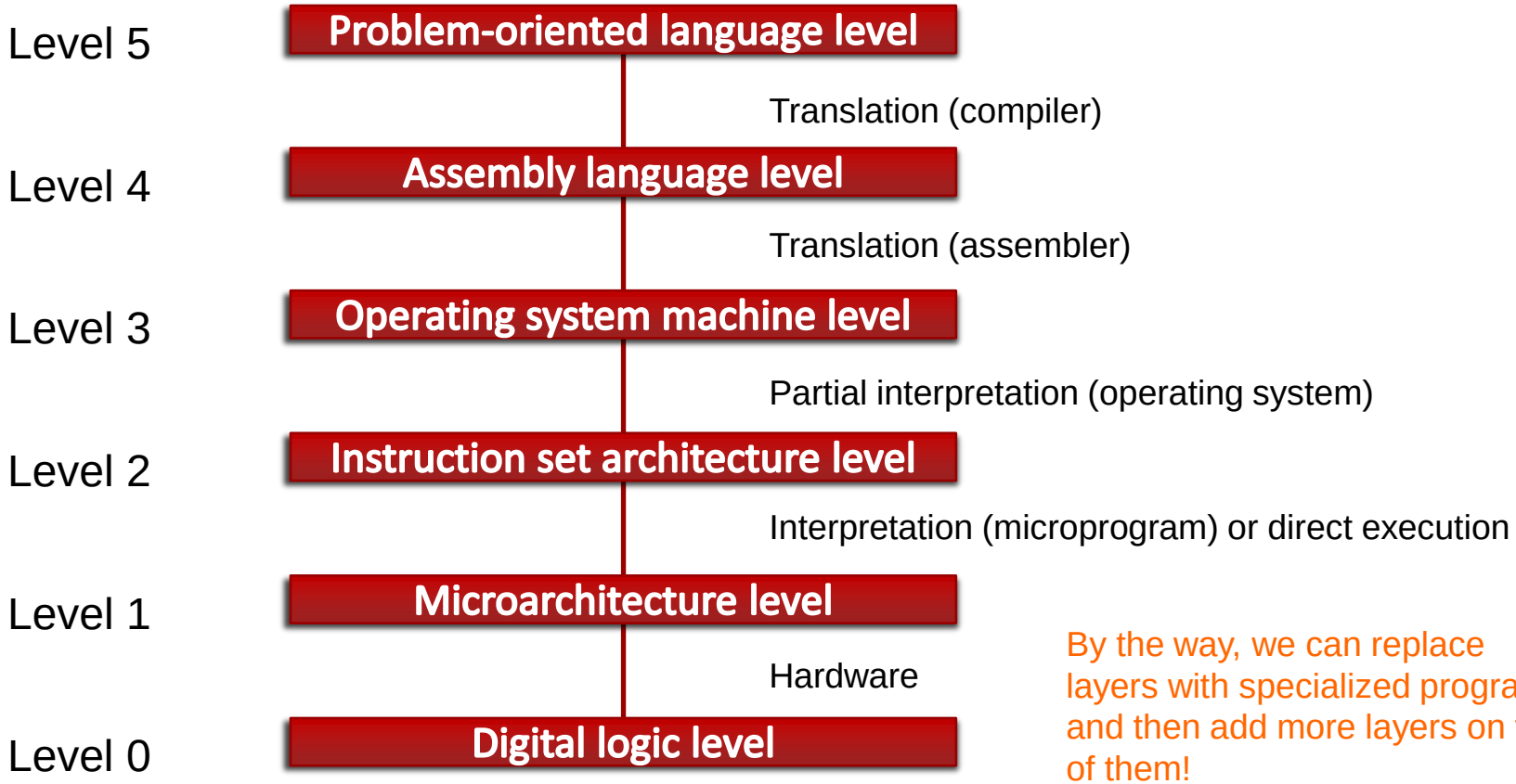
- There is no time available to review your final at the end of the semester.
- We do not review finals or discuss grades over break.
- You have the entire next semester you are on campus to review your final and all grades and have any problem fixed.

Coming Up: Big Ideas

Big Idea #1: All computers can compute the same kinds of things

- We call this Turing-equivalence
- Just about everything that we use for computation can be proved capable of solving the same set of problems.
- That includes Turing machines, stored program computers (and their programming languages), regular expressions, automata theory, formal grammars, etc.

Big Idea #2: Abstraction: Layers Making the Electrons Work

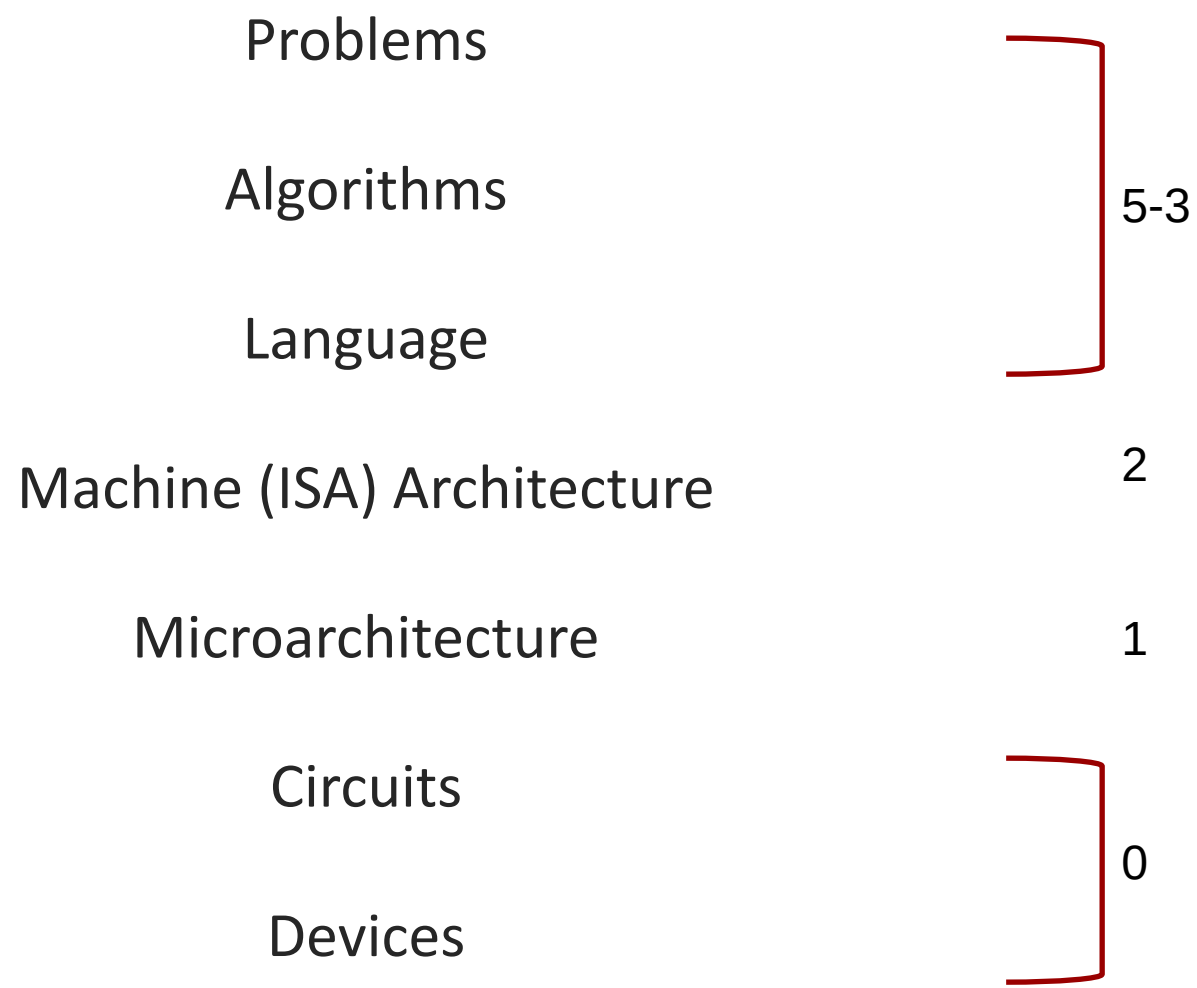


By the way, we can replace layers with specialized programs and then add more layers on top of them!

That's how we will run code for an imaginary machine, among many other things.

From *Structured Computer Organization,* Tanenbaum & Austin, 6th ed

Big Idea #2: Abstraction: Layers Making the Electrons Work



From *Introduction to Computing Systems*, Patt and Patel, 3ed.

Big Idea #3 Binary

- Binary is “better” than decimal for electronic computing.
- Why?
- Lots of small physical and economic reasons:
  - It’s easier to determine presence/absence of current rather than magnitude.
  - Can use lower voltages to distinguish only 0/1 instead of 0/1/2/3/4/5/6/7/8/9, so less power.
  - Binary-coded decimal math takes more circuitry than pure binary.

Big Idea #4: Computers Store Representations of Something Outside

- Computers can't store the mathematical abstraction we call a "number". Why?
- How many digits can a "number" have? How would you build that?
- So everything in a computer is a finite-sized **representation** of something outside.
- A bunch of binary digits (bits) is always interpretable as an unsigned whole number. We use that representation often.
- So we can always claim the bits stored in a computer represent a positive whole number.
- Is that the end of the story? Definitely not. Stay tuned.

Things to do

- Get the textbooks.
- Start reading!
  - Patt, Chapters 1, 2

Questions?