Introduction

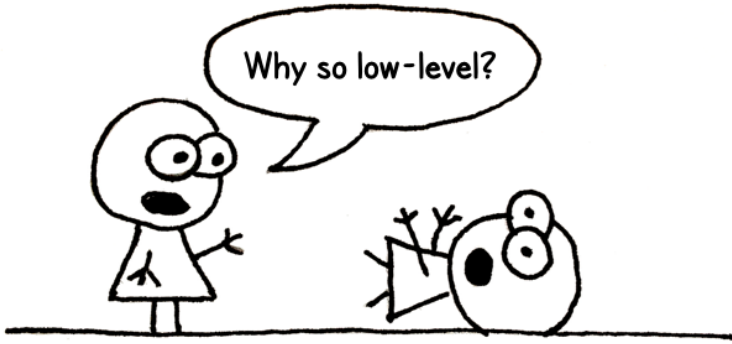
ICS312

Machine-Level and Systems Programming

Henri Casanova (henric@hawaii.edu)

Course Goal

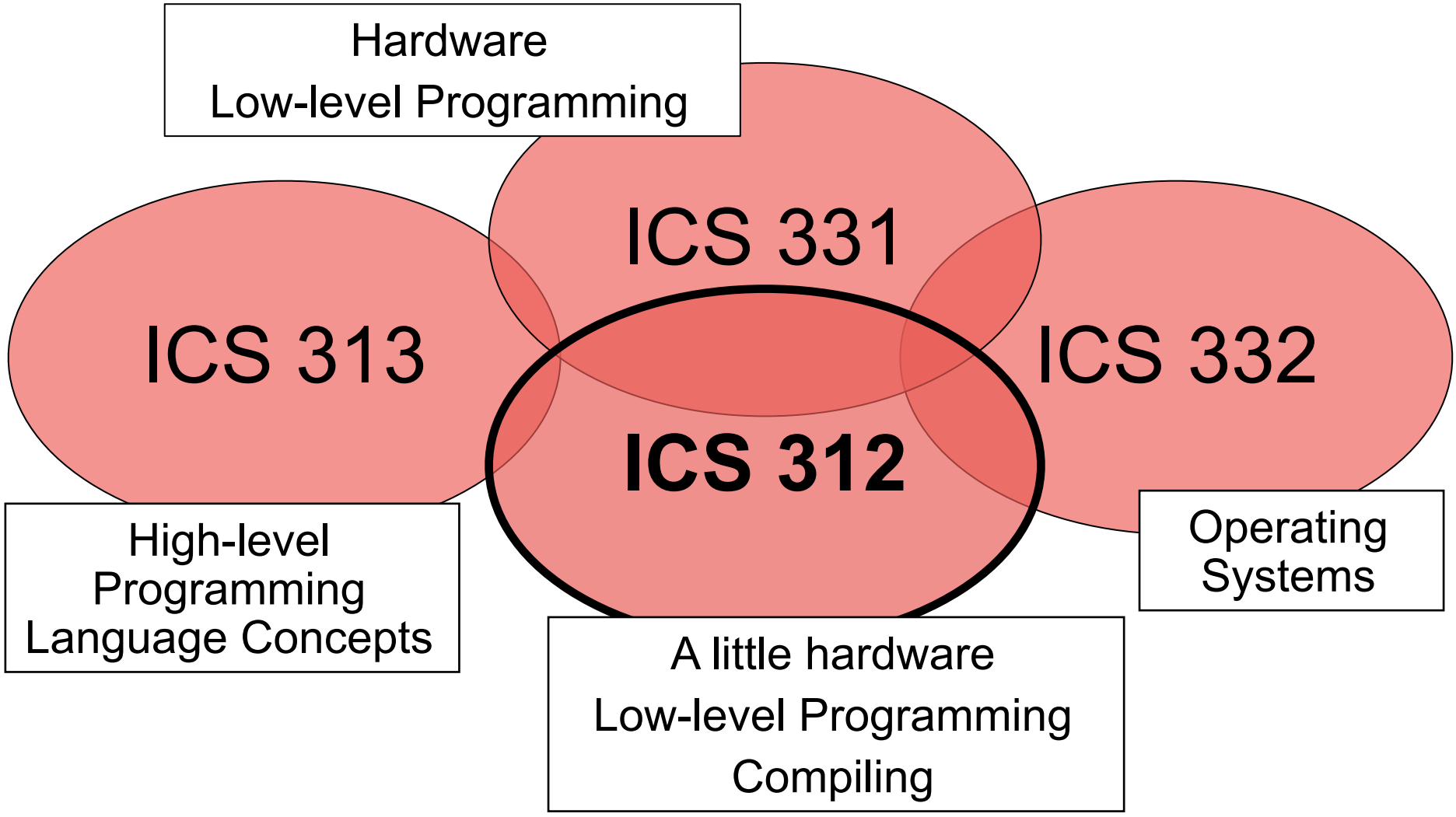
- At this point in your computer science education, most of you have only a very high-level understanding of how a computer run programs
- Your experience so far: you write code in some language (say a compiled language), you build it, and then you run it
  - Each of these steps involve hitting buttons in IDEs or typing some Shell commands
- You all know how to do this, but (most of) you have only very superficial understanding beyond “this magically creates an executable”, “this magically runs an executable, which my CPU magically knows how to do”
- A big part of being an effective (and employable) computer scientist is knowing what the magic is, which is the goal of this course



What we will learn

- What is inside a computer?
  - Elements of computer organization
  - Elements of computer architecture
- Assembly programming
  - Intel x86 Assembly
  - Important concepts that apply to all low-level programming and how they relate to high-level programming
- What is compiling, linking and debugging?
- How does one build a compiler?
  - The “theory” behind it
  - We’ll build a small one in class

ICS312 and the ICS Curriculum



Course Website

- Located at:
  - http://courses.ics.hawaii.edu/ics312_spring2026/
  - Linked from my personal homepage
    - Google for “Henri Casanova”
- Organized as Modules
  - All lecture notes as PDF files
  - Pointers to useful on-line material
  - All assignments
  - Announcements
  - A link to the Syllabus
    - Which we’re going over now in these slides
- Let’s look at the Web site...

Textbook(s)

- The main text is a free book:
  - *PC Assembly Language*, Paul A. Carter
  - Available for download on the course's Web site
- The user's manual for our assembler, NASM, is also available on the course's Web site, and other manuals will be made available throughout the semester

- Other interesting free resource available for download:
  - *The art of assembly programming*, John W. Lockwood
  - *Assembly Language for x86 Processors*, Kip Irvine
- See links from the Syllabus on the Course Web site

Lectures and Office Hours

- Lectures in MSB 114, Tue/Thu 10:30AM-11:45AM
- Lecture notes are posted on the course's Web site regularly
  - You can read them before or after the lecture, up to you really
  - I am notorious for spacing out on putting the notes up on the site, so DM me (Discord, E-mail)
- Office hours, TA, etc.
  - All information on the Web site

Inverted Lectures

- *A few* lectures will be "inverted"
  - You watch a screencast at your own pace
  - The lecture period is for questions and practice exercises
- I do this for a few topics in the course that are more "mechanical" or "difficult"
- You must watch the screencast ahead of time!
  - E-mails reminders will be sent out
- Scheduling may be imperfect
  - Out-of-order and/or overlapping modules
  - We might end a few lecture periods early

Screencast Lectures

- A few lectures might be screencast
  - This is because I am often required to travel for research purposes during the semester
- More information later if necessary…

Web site Content

- In spite of my best efforts it happens that the course Web site could have small problems (typos, missing link, etc.)

- Anytime you see anything strange/broken on the Web site, please let me know right away!
  - A one-line e-mail, a DM on Discord, etc.

Grading on 1000 points

- Sample and optional homework assignments for 0 points
- Four exams
  - Midterm #1 (220 points)
  - Midterm #2 (220 points)
  - Midterm #3 (220 points)
  - Final exam (270 points)
- Each exam assumes knowledge of the material up to that point in the semester (i.e., “cumulative”)
- Quizzes (70 points)
  - 8 10-point quizzes, worst grade is discarded

Homework Assignments

- **All homework assignments in this course are either "sample homework" or "optional homework", worth zero points**
- **Sample homework assignments:**
  - Posted as regular assignments would be
  - Solutions are provided on the assignment's page
  - You cannot turn them in
- **Optional homework assignments:**
  - Posted as regular assignments would be
  - You can turn them in and there is a due date
  - You will receive feedback
  - Solutions are available upon request after the due date
- Why? …

Homework Assignments

- Rationale for 0 points on homework assignments:
  - The use of LLMs has rendered homework unfair across students (blatantly seen last semester)
  - Students in this course used to “write a lot of code and struggle somewhat on assignments”, but this is no longer a thing
- The whole point of the course has always been to teach key concepts, not to attempt to make you assembly programming pros
  - Writing code was only a means-to-an-end for learning in this course anyway
  - Graded homework assignments used to be a way to force students to be prepared for exams

Homework Assignments

- **Can students learn the important concepts without going through the work of doing homework assignments?**
- Nobody really knows, some people think “absolutely” and some think “absolutely not”
- I think it completely depends on the student:
  - For some of you, not attempting the homework assignments or practice problems on your own will lead to catastrophic results
  - Some of you will ace all exams regardless
- The assumption in this “new world” is now that students are adults and know what they need to do to pass exams 😬

- We will do quite a bit of practice and live-coding in class!
  - And you should never hesitate asking “can we live-code this?” during lecture

Exams

- Exams are taken in class, closed-note
- Pocket calculators, not programmable calculators or phones, allowed (but not needed)
- Exams are randomly generated and students have different exams
  - So don’t cheat with your neighbors, it’s super obvious if you do (and it still sometimes happens!)
- Each exam will have mostly exercises that match exactly homework assignments and practice problems
- The final exam will include, among others, one exercise about material from each of the previous midterm exams

Quizzes

- 8 Quizzes in the semester
- Taken on the first lecture day of the week
  - Always on a Monday, unless that Monday is a holiday, in which case it will be on a Wednesday
- Always **announced the previous week ON THE COURSE'S WEB SITE**
- Taken at the **beginning of the lecture period, in the first 10 minutes**
  - You **cannot take the quiz if you show up more than 5 minutes late** to the lecture
- No make-up quizzes, unless a documented reason
- But **the worst quiz grade is discarded**

CES Evaluation

- Extra credit given to all students:
  - 0 points if CES completion rate is < 80%
  - 5 points if CES completion rate is >= 80%
  - 10 points if CES completion rate is 90%
- Why?
  - I do look at the evaluation every semester and evolve the course accordingly
    - Even if you love the course, it's important for me to hear what things didn't work
  - These evaluations have more impact than you may think and are taken seriously
    - Impact for individual faculty, for the whole department, for future students, etc.

How to not do well in this course?

- **Don't come to class ("the slides are nice")**
  - We do a LOT of stuff in class, including live coding, and I give a lot of explanations, examples
- **Don't attempt the homework assignments or practice problems ("I followed along in class, I am fine")**
  - Assembly programming and low-level concepts are often confusing to students
  - Putting in some time and struggling a bit on practice problems and assignments is the way to go
- **Don't come to office hours ("I am too busy and the instructor is too scary")**
  - After you struggle for a while on something, drop by
  - Instructor and TA office hours are an amazing service provided to you
- **Cheat**
  - Cheating is bad for many reasons, including hurting the reputation of ICS graduates!
  - If you are caught cheating or enabling cheating:
    - zero on the exam
    - overall grade lowered by a step (i.e., a "B" becomes a "C")
    - reported to UH's Office of Judicial Affairs (as required)
- **Expect that "what can I do for extra credit to pass this class?" at the end of the semester will be met with a positive response**
- **Don't study for the quizzes**
  - "It's only a small fraction of the grade"
  - But studying for quizzes is a HUGE help to prepare for exams
  - When I don't do quizzes, exam scores drops!

Show of hands

- To get an idea of your backgrounds here are a few “show-of-hands” questions
  - It’s totally OK if all the answers are “No”, don’t panic
- Have you taken / are taking ICS331?
- Have you taken / are taking ICS313?
- Have you written assembly code before as part of a course?
- Have you written assembly code before not as part of a course?
- If yes to above was it x86, MIPS, other?
- Do you feel comfortable you using the (UNIX/Linux/MacOS) Shell (not implying you are a Shell wizard)?
  - We don’t need much in this course
  - Note that this is something that you will have to do way more than you think beyond graduation
  - The “Getting Started” module has some pointers (let’s look at them quickly)

Software/Hardware for ICS312

- You'll have to use an Intel-based machine that speaks the x86 32-bit Instruction Set Architecture (IA-32)
  - Luckily, we all have that (or can emulate that)
  - In fact we have 64-bit machines!
  - But in this course I use 32-bit assembly because:
    - It has less "stuff" and is better for a first experience
    - We learn exactly the same set of things
    - Our free textbook is about 32-bit assembly
    - If you need to go to 64-bit assembly, the transition is easy
- We'll use the NASM software package, which is also free

- Let's look at Homework Assignment #0, which is ungraded but which you should do as soon as possible in the semester
  - If you intend to do any programming at all (which you really should!)

More Questions

- Any questions on the syllabus?
- Any questions on the course in general?

Participation Verification

- As you know, each instructor has to report on “Student Participation” and certify the class roster
  - If you have not “participated”, you could be dropped from the course

- **IMPORTANT:** Do the ungraded “Participation Verification” Assignment **posted on Lamakū**

What's Next?

- There is already an announcement on the course Web site regarding two short screencasts to watch **BEFORE THE NEXT LECTURE**
- These screencasts are about "numbers and computers"
  - Many of you will find them easy and can fast forward a lot of the material (or just look at the lecture notes quickly)
  - Make sure you watch them before this coming Thursday as we'll do in-class practice exercises to make sure we're all up to speed
- This is all in the Integers and Computers module…
  - Let's look at it now…
- Don't forget to get started on Homework #0…
- Let's do an ungraded quiz!