

Introduction

ICS312 - Spring 2009 Machine-Level and Systems Programming

Henri Casanova (henric@hawaii.edu)

Course Goal

- Overall goal: Understand many of the things that happen “under the cover” when running a program on a computer
- By the end of the class you will
 - Have a basic understanding of how a CPU works and can be programmed
 - Be able to write programs (or pieces of programs) in x86 assembly
 - Have a basic understanding of compiler technology and (hopefully) experience with building a tiny compiler

Course Website

- Located at:
 - http://navet.ics.hawaii.edu/~casanova/courses/ics312_spring09
 - Linked from my homepage
 - Accessible from the ICS homepage
- Contains
 - All lecture notes (posted 1-2 weeks before the corresponding lecture)
 - Pointers to useful on-line material
 - All assignments
 - Announcements
 - A link to the PDF Syllabus

Textbook

- 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
- Another interesting free resource available for download is:
 - *The art of assembly programming*, John W. Lockwood
- Do not print the above on ICS printers, print them at your own expenses (e.g., using ITS)
- Finally, an interesting book that could come in handy for more details is:
 - *Assembly Language for Intel-based Computers* (5th edition), Kip Irvine
 - Available from Amazon for ~\$100 (new) or ~\$50 (used)

Lectures and Office Hours

- Lectures: Tue & Thu, 4:30PM-5:45PM
- Office Hours: Wed, 1PM-3PM
 - POST 310C
 - phone: 956-2649
 - e-mail: henric@hawaii.edu
- Teaching Assistant: Andrew Wong
 - POST 303-5
 - e-mail: wongandr@hawaii.edu

Homework Assignments

- Assignments
 - Most of them will be about writing code
 - To be turned in by e-mail by 11:59PM on the day the assignment is due
 - Also a few “pencil and paper” assignments
 - Time for doing assignments will be somewhere between 1 week and 2 weeks
- **Assignments will not be accepted late**
 - Solutions will be posted on the course’s Web site the morning after the assignment is due
- Read the syllabus’ statement about “academic dishonesty”

Exams, Grading

- Exams
 - Several in-class Quizzes (~7 in the semester)
 - Announced beforehand
 - Two midterm exams
 - One comprehensive final exam
- Grading
 - Homework Assignments 40%
 - Quizzes 15%
 - Midterms 20%
 - Final 25%

Questions about the Syllabus?

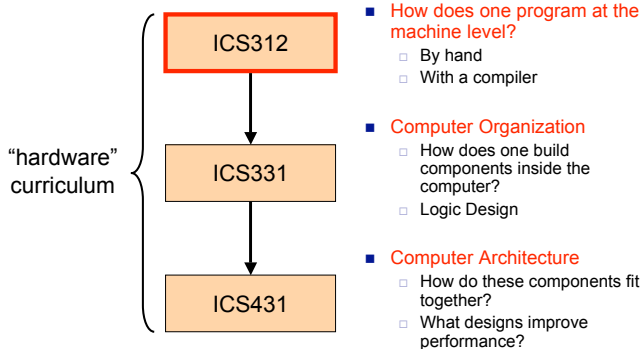
What is this course about?

- At this point in your computer science education, most of you have only a very high-level understanding of how a computer runs programs:
 - You write code in some language and compile it
 - compilation is probably this “magical” step at this point
 - You run the code
 - somehow the computer “magically” runs stuff
 - You debug with print statements and the likes
 - You repeat
- A lot of **magic** happens here under the cover
- A big part of being a good computer scientist is knowing what the magic is, which is what we’ll learn in this class

Topics in this class

- What is inside a computer?
 - Elements of computer organization
 - Elements of computer architecture
- Assembly programming
 - Intel Assembly
- What are compilers, linkers and debuggers and how do they work?
- How does one build a compiler?
- A hands-on course
 - Writing (pieces of) programs in assembly code and C
 - Implementing a small (piece of a) compiler

ICS312 and the ICS Curriculum



ICS312 and the ICS Curriculum

- There is clear overlap between these courses
 - In 312 we have to talk a little bit about computer architecture and computer organization
 - In 331 and 431 we write some assembly code
- In this class I’ll mention the topics that we’re barely exploring here and will point whether you’ll see them in 331 or in 431

Software/Hardware for ICS312

- We'll use the following software packages:
 - NASM, Lex, Yacc
- All these are available for both Linux and Windows
- You'll have to use an Intel-based machine that speaks the x86 32-bit Instruction Set Architecture (IA-32, ...)
- You can install all needed software on your own machine
- I'll also provide you with an account on my own Linux machine with all the software installed
 - This is where I'll test your assignments
- My explanations in the course will have a Linux bias
 - I actually don't have a Windows machine
 - I won't be able to provide much help for Windows issues
- Note that Linux development is something that you'll most likely have to do in your professional life, so you may as well be exposed to it early on
 - Linux is not a "professor thing"! :)

First Quiz

- Who has never written programs in C before?
- Who has never programmed in a Linux environment before?
- Who has never written a Makefile?
- Who has written programs in Assembly before?
- Who has heard of "Lex" and "Yacc" before?

Any Questions?