Lecture #1 – Introduction

2006-08-28

There are two handouts today at the front and middle of the room!

Lecturer SOE Dan Garcia

www.cs.berkeley.edu/~ddgarcia

Great book ⇒

The Universal History of Numbers

by Georges Ifrah
“I stand on the shoulders of giants…”

Prof David Patterson
Prof John Wawrznek
TA Andy Carle
TA Kurt Meinz

Thanks to these talented folks (& many others) whose contributions have helped make 61C a really tremendous course!
Where does CS61C fit in?

We will not be enforcing the CS61B prerequisite this semester.

I will be teaching CS61C in Fall 2006 and Spring 2007 so no reason to swap, imho.

http://hkn.eecs.berkeley.edu/student/cs-prereq-chart1.gif
Are Computers Smart?

• To a programmer:
  • Very complex operations / functions:
    - `(map (lambda (x) (* x x)) '(1 2 3 4))`
  • Automatic memory management:
    - `List l = new List;`
  • “Basic” structures:
    - Integers, floats, characters, plus, minus, print commands

Computers are smart!
Are Computers Smart?

• In real life:
  • Only a handful of operations:
    - {and, or, not}
  • **No** memory management.
  • Only 2 values:
    - {0, 1} or {low, high} or {off, on}

Computers are dumb!
What are “Machine Structures”? 

*Coordination of many levels (layers) of abstraction
61C Levels of Representation

High Level Language Program (e.g., C)

Compiler

Assembly Language Program (e.g., MIPS)

Assembler

Machine Language Program (MIPS)

Machine Interpretation

Hardware Architecture Description (Logic, Logisim, Verilog, etc.)

Architecture Implementation

Logic Circuit Description (Logisim, etc.)

\[
\begin{align*}
temp &= v[k]; \\
v[k] &= v[k+1]; \\
v[k+1] &= temp; \\
lw &= \$t0, 0($2) \\
lw &= \$t1, 4($2) \\
sw &= \$t1, 0($2) \\
sw &= \$t0, 4($2) \\
\end{align*}
\]
Anatomy: 5 components of any Computer

Personal Computer

- Processor
  - Control (“brain”)
  - Datapath (“brawn”)
- Memory
  (where programs, data live when running)
- Devices
  - Input
  - Output
- Keyboards, Mouse
- Disk
  (where programs, data live when not running)
- Display, Printer
Overview of Physical Implementations

The hardware out of which we make systems.

• Integrated Circuits (ICs)
  • Combinational logic circuits, memory elements, analog interfaces.

• Printed Circuits (PC) boards
  • substrate for ICs and interconnection, distribution of CLK, Vdd, and GND signals, heat dissipation.

• Power Supplies
  • Converts line AC voltage to regulated DC low voltage levels.

• Chassis (rack, card case, ...)
  • holds boards, power supply, provides physical interface to user or other systems.

• Connectors and Cables.
Integrated Circuits (2005 state-of-the-art)

- Primarily Crystalline Silicon
- 1mm - 25mm on a side
- 2005 - feature size \( \sim 90 \, \text{nm} = 90 \times 10^{-9} \, \text{m} \)
- 100 - 1000M transistors
- (25 - 100M “logic gates”)
- 3 - 10 conductive layers
- “CMOS” (complementary metal oxide semiconductor) - most common.

- Package provides:
  - spreading of chip-level signal paths to board-level
  - heat dissipation.

- Ceramic or plastic with gold wires.

Chip in Package

Bare Die

• Package provides:
  • spreading of chip-level signal paths to board-level
  • heat dissipation.

• Ceramic or plastic with gold wires.
Printed Circuit Boards

- fiberglass or ceramic
- 1-20 conductive layers
- 1-20 in on a side
- IC packages are soldered down.

Provides:
- Mechanical support
- Distribution of power and heat.
Technology Trends: Microprocessor Complexity

Gordon Moore
Intel Cofounder
B.S. Cal 1950!

2X Transistors / Chip
Every 1.5 years

Called “Moore’s Law”
Technology Trends:
Memory Capacity (Single-Chip DRAM)

- Now 1.4X/yr, or 2X every 2 years.

- 8000X since 1980!
Technology Trends:
Uniprocessor Performance (SPECint)

- **VAX**: 1.25x/year 1978 to 1986
- **RISC + x86**: 1.52x/year 1986 to 2002
- **RISC + x86**: 1.20x/year 2002 to present
Computer Technology - Dramatic Change!

• Memory
  - DRAM capacity: 2x / 2 years (since ‘96); 64x size improvement in last decade.

• Processor
  - Speed 2x / 1.5 years (since ‘85); [slowing!] 100X performance in last decade.

• Disk
  - Capacity: 2x / 1 year (since ‘97) 250X size in last decade.
Computer Technology - Dramatic Change!

We’ll see that Kilo, Mega, etc. are incorrect later!

- State-of-the-art PC when you graduate: (at least…)
  - Processor clock speed: 5000 MegaHertz (5.0 GigaHertz)
  - Memory capacity: 8000 MegaBytes (8.0 GigaBytes)
  - Disk capacity: 2000 GigaBytes (2.0 TeraBytes)
  - New units! Mega ⇒ Giga, Giga ⇒ Tera

  (Tera ⇒ Peta, Peta ⇒ Exa, Exa ⇒ Zetta, Zetta ⇒ Yotta = 10^{24})
CS61C: So what's in it for me?

• Learn some of the big ideas in CS & engineering:
  • Principle of abstraction, used to build systems as layers
  • 5 Classic components of a Computer
  • Data can be anything (integers, floating point, characters): a program determines what it is
  • Stored program concept: instructions just data
  • Principle of Locality, exploited via a memory hierarchy (cache)
  • Greater performance by exploiting parallelism
  • Compilation v. interpretation thru system layers
  • Principles/Pitfalls of Performance Measurement
Others Skills learned in 61C

• Learning C
  • If you know one, you should be able to learn another programming language largely on your own
  • Given that you know C++ or Java, should be easy to pick up their ancestor, C

• Assembly Language Programming
  • This is a skill you will pick up, as a side effect of understanding the Big Ideas

• Hardware design
  • We’ll learn just the basics of hardware design
  • CS 150, 152 teach this in more detail
Course Lecture Outline

- Number representations
- C-Language (basics + pointers)
- Storage management
- Assembly Programming
- Floating Point
- make-ing an Executable (compilation, assembly)
- Logic Circuit Design
- CPU organization
- Pipelining
- Caches
- Virtual Memory
- Performance
- I/O Interrupts
- Disks, Networks
- Advanced Topics
Yoda says…

“Always in motion is the future…”

Our schedule may change slightly depending on some factors. This includes lectures, assignments & labs…
Texts

• Required: *Computer Organization and Design: The Hardware/Software Interface, Third Edition*, Patterson and Hennessy (COD). *The second edition is far inferior, and is not suggested.*

• Required: *The C Programming Language*, Kernighan and Ritchie (K&R), 2nd edition

• Reading assignments on web page
What is this?

Attention over time!
What is this?!

Attention over time!

~5 min
Tried-and-True Technique: Peer Instruction

• Increase real-time learning in lecture, test understanding of concepts vs. details

• As complete a “segment” ask multiple choice question
  • 1-2 minutes to decide yourself
  • 3 minutes in pairs/triples to reach consensus. Teach others!
  • 5-7 minute discussion of answers, questions, clarifications

• You’ll get transmitters from ASUC bookstore (or Neds) (hopefully they’re in!)
Peer Instruction

• Read textbook
  • Reduces examples have to do in class
  • Get more from lecture (also good advice)

• Fill out 3-question Web Form on reading (released mondays, due every friday before lecture)
  • Graded for effort, not correctness…
  • This counts toward “E”ffort in EPA score
### Weekly Schedule

We are having discussion, lab and office hours this week...

<table>
<thead>
<tr>
<th></th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Mon</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>COMPSCI 61C S 116 DIS Machine Structures 320 Soda</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>COMPSCI 61C S 016 LAB Machine Structures 271 Soda</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

---

Cal

CS61C L01 Introduction (26)

Garcia, Fall 2006 © UCB
Homeworks, Labs and Projects

- **Lab exercises** (every wk; due in that lab session unless extension given by TA) – extra point if you finish in 1st hour!

- **Homework exercises** (~ every week; (HW 0) out now, due in section *next week*)

- **Projects** (every 2 to 3 weeks)

  - All exercises, reading, homeworks, projects on course web page

  - We will DROP your lowest HW, Lab!

- Only one {HW, Project, Midterm} / week
2 Course Exams

• **Midterm:** Monday 2006-10-16 @ 7-10pm
  - Give 3 hours for 2 hour exam
  - One “review sheet” allowed
  - Review session Sun beforehand, time/place TBA

• **Final:** Thu 2006-12-14 @ 12:30-3:30pm (grp 8)
  - You can *clobber* your midterm grade!
  - (students last semester LOVED this…)

![Graph showing midterm clobber results]
Your final grade

- Grading (could change before 1st midterm)
  - 15pts = 5% Labs
  - 30pts = 10% Homework
  - 60pts = 20% Projects
  - 75pts = 25% Midterm* [can be clobbered by Final]
  - 120pts = 40% Final
  - Extra credit for EPA. What’s EPA?

- Grade distributions
  - Similar to CS61[AB], in the absolute scale.
  - Perfect score is 300 points. 10-20-10 for A+, A, A-
  - Similar for Bs and Cs (40 pts per letter-grade)
    … C+, C, C-, D, F (No D+ or D- distinction)
  - Differs: No F will be given if all-but-one {hw, lab}
    all projects submitted and all exams taken
  - We’ll “ooch” grades up but never down
Extra Credit: EPA!

• **Effort**
  - Attending Dan’s and TA’s office hours, completing all assignments, turning in HW0, doing reading quizzes

• **Participation**
  - Attending lecture and voting using the PRS system
  - Asking great questions in discussion and lecture and making it more interactive

• **Altruism**
  - Helping others in lab or on the newsgroup

• **EPA! extra credit points have the potential to bump students up to the next grade level!** (but actual EPA! scores are internal)
Course Problems…Cheating

• What is cheating?
  • Studying together in groups is encouraged.
  • Turned-in work must be completely your own.
  • Common examples of cheating: running out of time on an assignment and then pick up output, take homework from box and copy, person asks to borrow solution “just to take a look”, copying an exam question, ...
  • You’re not allowed to work on homework/projects/exams with anyone (other than ask Qs walking out of lecture)
  • Both “giver” and “receiver” are equally culpable

• Cheating points: negative points for that assignment / project / exam (e.g., if it’s worth 10 pts, you get -10) In most cases, F in the course.

• Every offense will be referred to the Office of Student Judicial Affairs.

www.eecs.berkeley.edu/Policies/acad.dis.shtml
My goal as an instructor

• To make your experience in CS61C as enjoyable & informative as possible
  • Humor, enthusiasm, graphics & technology-in-the-news in lecture
  • Fun, challenging projects & HW
  • Pro-student policies (exam clobbering)

• To maintain Cal & EECS standards of excellence
  • Your projects & exams will be just as rigorous as every year. Overall : B- avg

• To be an HKN “7.0” man
  • I know I speak fast when I get excited about material. I’m told every semester. Help me slow down when I go toooo fast.
  • Please give me feedback so I improve! Why am I not 7.0 for you? I will listen!!
Teaching Assistants

• Scott Beamer (also Head TA)
• Sameer Iyengar
• David Jacobs
• David Poll
• Aaron Staley
Summary

• Continued rapid improvement in computing
  • 2X every 2.0 years in memory size;
    every 1.5 years in processor speed;
    every 1.0 year in disk capacity;
  • Moore’s Law enables processor
    (2X transistors/chip ~1.5-2 yrs)

• 5 classic components of all computers
  Control  Datapath  Memory  Input  Output
  
  Processor