Welcome to CS107!

Computer Organization and Systems

Agenda

- CS107, the course
  - What, who, and why
- Admin and logistics
- Let's code!

Learning goals

- **Mastery**
  - Can write/debug C code with complex use of memory/pointers
  - Have accurate model of address space and runtime behavior of program
- **Competency**
  - Can translate C code to/from assembly language equivalent
  - Can write C code that respects the limitations of computer arithmetic
  - Can identify bottlenecks and improve runtime performance of C code
  - Can write code that correctly ports to other architectures
  - Can work effectively in Unix development environment
- **Exposure**
  - Have working understanding of computer architecture

Philosophy

- **Importance of tools**
  - What they do, how to use them effectively, where to learn more
- **Hands-on exploration**
  - Observe, examine, measure, trace, experiment
  - Answer questions by doing
- **Followthrough**
  - Drill down to make connections, map out cause & effect
  - Leave no stone unturned
- **= Empowerment & Enlightenment**
  - You can do it, we can help!

Course logistics

- Lectures Monday & Friday
- Hands-on weekly lab
- Readings
  - Computer Systems, C language reference
- Challenging programming assignments
  - C, x86 assembly, Unix development tools
- Midterm & final
- Website http://cs107.stanford.edu
- Student skills for success
  - CS106/C++ experience, curiosity, perseverance, hard work, when to get help

Getting help

- **Website materials**
  - Good for: topic resources, course policy info, general advice
- **Discussion forum**
  - Questions and answers very welcome! Staff also participates
  - Good for: discussions about course content, tool use, tactics
- **Email to cs107@cs.stanford.edu**
  - Good for: questions about your specific code, private issues
- **Office hours**
  - Good for: in-person debugging advice, conceptual help
- **Peers**
  - Good for: conceptual help, topic review, shared joy/commiseration

Honor code

- You are expected to turn in original, independent work
- Allowed and encouraged:
  - Helping each other with general knowledge: course concepts, assignment specifications, language features, tool use
- Not allowed:
  - Sharing/copying code: neither to give nor to receive is divine
  - Using code from previous quarters/others/web
  - Joint design/coding/debugging
- Plagiarism detection tools in use
  - Vigilant followup

Celebrate the programmer!

- **Most systems courses are implementation-centric**
  - Building a compiler, operating system, database, microprocessor, etc.
- **CS107 is programmer-centric**
  - We are building YOU into a master programmer
- **Your code will be more robust, efficient, portable, reliable**
- **Not just for dedicated hackers**
  - Your backstage pass to the inner workings of your computer
  - Find the hidden hacker within…!

Ready, set, go!

- Things to do in first week
- Sign up for lab
  - Online at course website, signups open Wed morning
- Acclimate yourself to unix
  - Read/view unix reference topics as needed
  - Practice with tools
  - Configure your environment
- First assignment
  - Will be posted tomorrow and due next Monday
  - Warmup exercise with unix and C
- Need help?
  - Piazza open now, office hours starting tomorrow