15-122: Principles of Imperative Computation

Penny Anderson, Tom Cortina, Frank Pfenning

http://www.cs.cmu.edu/~fp/courses/15122-f15
http://c0.typesafety.net/
Overview

• Goals of this course
• Interactions
  – Lectures, labs, recitations, office hours
• Assessment
  – Quizzes, homework (written, prog.), exams
• A mysterious function!
Goals

Computational Thinking

Programming

Algorithms
Programming Skills

• Transforming algorithmic ideas to code
• Writing tests
• Imperative programming in C and C0
• Basic Unix survival
Algorithmic Ideas

• Asymptotic complexity
  – time/space/amortized
  – worst case/average case
  – important classes: \( O(1) \), \( O(\log n) \), \( O(n \log n) \), \( O(n^k) \), \( O(2^n) \)

• Important ideas like order and randomness

• Lots of fundamental data structures
  – (Psst... this is often what tech interviews test on!)
Computational Thinking

• “Thinking like a computer scientist” is important for lots of people, not just computer scientists!

• A computer science approach to thinking about the correctness of programs
The Big Picture

• Pre- or co-requisites
  – either 15-151 (Math Foundations for CS)
  – or 21-127 (Concepts of Mathematics)

• Counterpart
  – 15-150 (Principles of Functional Programming)

• Pre-requisite for
  – 15-213 (Introduction to Computer Systems)
  – 15-210 (Parallel and Sequential Data Structures and Algorithms)
  – 15-214 (Principles of Software System Construction)
Overview

• Goals of this course
• Interactions
  – Lectures, recitations, office hours
• Assessment
  – Quizzes, homework, exams
• A mysterious function!
Lectures

• Tuesday and Thursday

• Please be here, please be active
  – Ask and answer questions, pay attention
  – Lecture notes published after lecture

• Laptops for note-taking only
  – No surfing, email, games...
  – Work on your homework elsewhere
  – If you can see board from the back row, be there
  – Too distracting for other students
Labs and Recitations

- (Hello, TAs!)
- Labs Monday, recitations Friday
- Reinforce lecture material
- Problem solving (and working in groups!)
- How-to programming and tool support
- *Attend the lab/recitation you’re registered for*
Laptop Setup Office Hours

- Day and time TBA
- Set up using the C0 tools with Andrew Linux
- Format: drop-in for half an hour
- Or do it yourself:
  
  http://c0.typesafety.net/tutorial/C0-at-CMU.html
Online communication

• Autolab for homework and grades
• Piazza for announcements, questions, and communication with course staff. Get help, help each other!
• Cluster Linux machines and SSH to shared machines for assignments
Other Resources

• Course home page
  – http://www.cs.cmu.edu/~fp/courses/15122-f15
  – Schedule, lecture notes, calendar, contact info...
  – Office hours start soon, check Piazza

• C0 home page
  – http://c0.typesafety.net/
  – Tutorial, reference, examples, binaries
Overview

• Goals of this course
• Interactions
  – Lectures, recitations, office hours
• Assessment
  – Quizzes, homework, exams
• A mysterious function!
Assessment

• 50% - Exams (2 midterms and a final)
• 45% - Weekly Homework
  – Programming usually due Thursday 10pm through Autolab
    • 4 late days total, max 1 day per assignment
    • Download assignments and code from Autolab
    • Style grading
  – Written usually due Monday by 5:30pm in person
    • No late days, turn in Tuesday in lecture for a significant penalty
    • *Don’t hand in work for other people*
• 5% - Quizzes and Lab participation
  – Labs Monday, Quizzes Fridays (still under discussion)
  – Basically: make a good effort to get full credit

http://www.cs.cmu.edu/~fp/courses/15122-f15/schedule.html
Academic integrity

• Quizzes, exams, homework must be your own
• You must hand in your work
• OK: discussion of course material, practice problems, study sessions, going over handed-back homework in groups
• Not OK: copying or discussing answers, looking at or copying code (even parts)
• Not OK: talking through the assignment as you code with a classmate

• We use MOSS to catch code duplication
• If you make a mistake, come to us, don’t let us come to you
Overview

• Goals of this course

• Interactions
  – Lectures, recitations, office hours

• Assessment
  – Quizzes, homework, exams

• A mysterious function!
A Mysterious Function Approaches!