15-122: Principles of Imperative Computation

Iliano Cervesato, Pat Virtue

http://cs.cmu.edu/~15122

http://c0.typesafety.net/

Overview

- Goals of this course
- Interactions
  - Lectures, labs, recitations, office hours
- Assessment
  - Quizzes, homework (written, prog.), exams

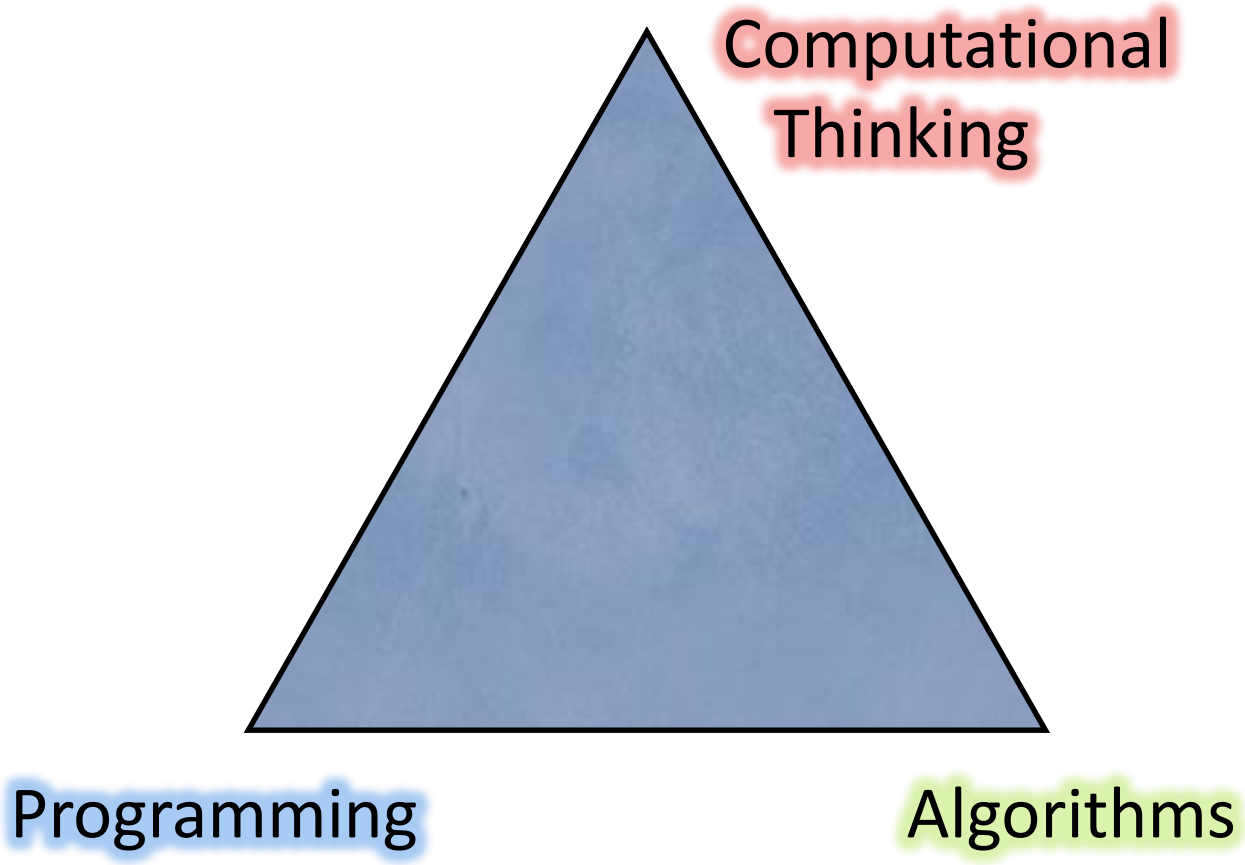
- The course begins …

Activity

T A B L E	B L _ _ _
R E P L Y	P I A _ _
H A N D L E	C L _ _ _
B E L O W	S T U _ _
D E F I N E	D A _ _ _

- *Copy these words on a sheet of paper*
- *Complete the blanks to create new words (in any language)*
- *Turn the sheet of paper over and leave it that way*

Goals



Programming Skills

- Transforming algorithmic ideas to code
  - Code that works the first time around
    - *Deliberate programming*
  - Well, *nearly* the first time around
    - Writing tests
- Imperative programming in C and C0
- Basic Unix survival

Algorithmic Ideas

- Asymptotic complexity
  - time/space
  - worst case/average case/amortized analysis
  - 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!
  - Systematic approach to solving a problem
  - Finding solutions that are *correct*
  - Finding solutions that are *efficient*
- Develop vocabulary and tool kit

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
- The course begins …

Online Resources

- Course home page
  - http://cs.cmu.edu/~15122
  - Schedule, lecture notes, calendar, contact info…
  - Office hours start soon
- C0 home page
  - http://c0.typesafety.net/
  - Tutorial, reference, examples, binaries

Lectures

- Tuesday and Thursday
- Please be here, please be active
  - Ask and answer questions, pay attention
  - Lecture notes for review
  - **NEW!** *a few online modules* (optional)
- 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

- Labs Monday (programming exercises)
- Recitations Friday (review & written exercises)

- **Collaborative** problem solving
  - Help others if you are done early!
- How-to programming and tool support
- *Attend the lab/recitation you're registered for*

Getting-started Help

- Laptop setup office hours
  - Wednesday 4:30 to 6:30pm, Porter Hall 100
  - 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)
- Linux workshops
  - Thursday and Tuesday 7 to 9pm, Doherty Hall 2315
  - Learn useful Linux commands

Online communication

- **Autolab** and **Gradescope** for homework
- Grades from web page
- **Piazza** for announcements, questions, and communication with course staff
  - Get help, help each other!
- Cluster Linux machines and SSH to shared machines for assignments

Overview

- Goals of this course
- Interactions
  - Lectures, recitations, office hours
- Assessment
  - Quizzes, homework, exams
- The course begins …

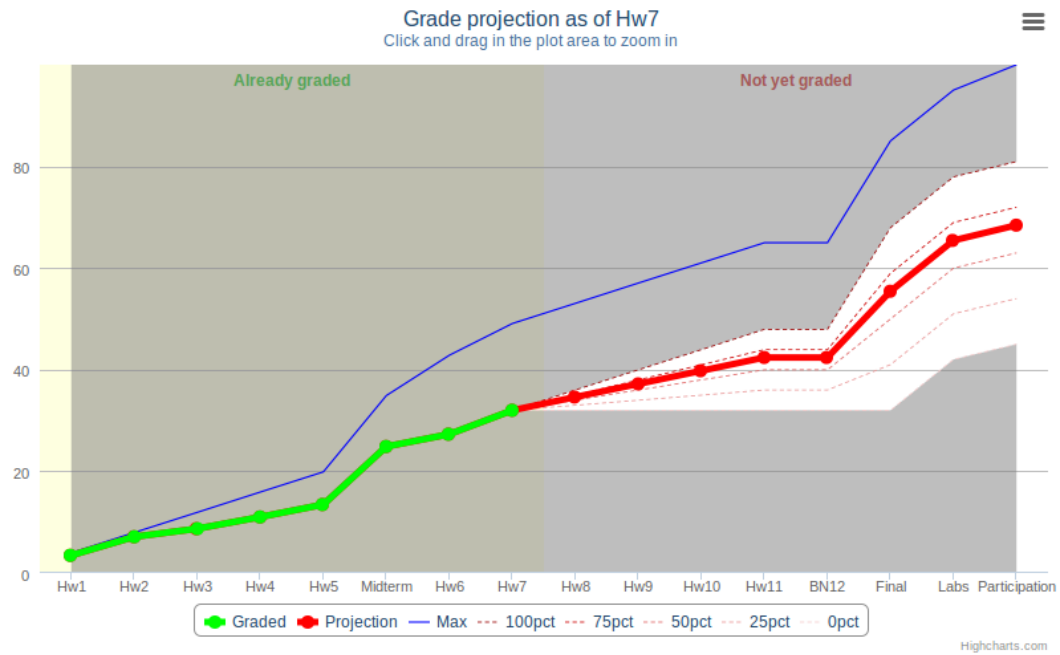
Assessment

- 50% - Exams (2 midterms and a final)
- 45% - Weekly Homework
  - *Written* due Monday by 9pm through Gradescope
    - No late days: 50% penalty if handed in within 12 hours
    - ∞ submissions
  - *Programming* due Thursday 9pm through Autolab
    - Download assignments and code from Autolab
    - 3 late days, at most 1 per homework
    - No extensions
- 5% - Quizzes and lab/recitation participation
  - Quizzes in lecture and recitation
  - Basically: attend, make a good effort, get full credit

Grades

Grade Forecaster

david49's Past Performance and Future Projections



... what if ...

Hw8: NA % Hw9: NA % Hw10: NA % Hw11: NA % BN12: NA % Final: NA %

% /max Reset

david49's Grades

	Hw1	Hw2	Hw3	Hw4	Hw5	Midterm	Hw6	Hw7	Hw8	Hw9	Hw10	Hw11	BN12	Final	Labs	Participation
Weight	3.98%	3.98%	3.98%	3.98%	3.98%	15%	7.95%	6.23%	3.98%	3.98%	3.98%	3.98%	0%	20%	10%	5%
Max	95	74	105	97	86	100	105	115	70	90	80	75	65	100	8	10
Grade	83	68	42	56	53	76	32	87	*45.66*	*58.7*	*52.18*	*48.92*	*42.4*	*65.23*	8	6
Grade (%)	87.37	91.89	40	57.73	61.63	76	30.48	75.65	*65.23*	*65.23*	*65.23*	*65.23*	*65.23*	*65.23*	100	60
C.W. Grade	3.48	7.13	8.73	11.02	13.48	24.88	27.3	32.01	*34.61*	*37.21*	*39.8*	*42.4*	*42.4*	*55.44*	65.44	68.44

Trend

Academic integrity

- Quizzes, exams, homework *must be <u>your own</u>*
- *You* must hand in your work
  - **OK**: discussing course material, practice problems, study sessions, going over handed-back homework in groups
  - **Not OK**: copying or discussing answers, looking at or copying code or tests (even parts)
  - **Not OK**: talking through the assignment as you code with a classmate
- **Whiteboard policy**
  - **OK**: discussing *approaches* to solving a problem
  - Wait **at least 4 hours**, write solutions **individually**
  - **Not OK**: taking notes or pictures, memorizing answers

- **Never OK**: sharing/writing code together (even pseudocode)
  - We use MOSS to catch code duplication across semesters

If you make a mistake, come to us, don't let us come to you

How to do Well in this Course

- Do not stress over grades
- Participate
- Manage your time wisely
  - *Don't use late days in 1st half of course*
- Start homework early
- Get all the help you need
- Make time for fun

Activity debrief

- *Without looking at your paper, write down as many of those words as you can recall*

- *How many people got more from the left column?*
- *From the right column?*

- **It's going to be a lot easier if you take good handwritten notes**

Overview

- Goals of this course
- Interactions
  - Lectures, recitations, office hours
- Assessment
  - Quizzes, homework, exams
- The course begins …

A Mysterious Function Approaches …