CIS 211: Computer Science II

Instructor: Daniel Lowd
Teaching Assistants: Mojtaba Torkjazi Kofi Appiah
Winter 2011

What’s this class all about?

CIS 210: Variables, loops, conditionals, arrays, functions, recursion, (simple) objects, pointers
“Programming is about specifying a series of steps to accomplish some goal.”

CIS 211: Objects, inheritance, data structures, (basic) computational complexity
“Programming is also about building abstractions to manage complexity.”

Primary Resources

Lecture:
MWF 1pm in Gerlinger 302
Textbook:
Building Java Programs, 2nd Edition by Reges and Stepp
Web page: http://www.cs.uoregon.edu/classes/11W/cis211/index.php
Announcement blog: http://uo-cis211.blogspot.com/

Supplementary Resources

• Practice-It! tool
• Online video lectures (with book code)
• PowerPoint slides (sometimes different from mine)
http://www.buildingjavaprograms.com/supplements.shtml
• UW CSE 143 web site
http://www.cs.washington.edu/education/courses/cse143/
• Java API specification
• Other books
Office Hours

- **Kofi:**
  - Monday 11am-Noon
  - Thursday 10:30am-11:30am
  - Friday 11am-Noon
- **Moji:**
  - Tuesday 4-5pm
  - Wednesday 4-5pm
  - Thursday 4-5pm
- **Daniel**
  - Tuesday 2-3:30pm
  - Wednesday 2-3:30pm

Grading

- Similar to CIS 210
- Assignments: 35%
  - 8 assignments total, drop the lowest score
  - Each of the 7 is worth 5% of your grade
  - Due Thursday at 5:00pm submitted through Blackboard
  - No collaboration allowed
- Lab attendance and quizzes: 10%
  - New option: Skip lab for the week if you complete and turn-in a “challenge problem” by 1:00pm on Monday.
- Midterm: 20%
- Final: 35%

Collaboration

- All assignments are to be done independently.
- However... collaborative studying and practice are encouraged
- Excellent tool: Practice-It

Tips for Success

- Read the book
- Come to lecture and lab
- Start on the assignments early
- Try extra practice problems in the book, or using the Practice-It! tool
- Come to office hours for help when you need it
Teach Yourself Programming in Ten Years (ideas by Peter Norvig)

• Many books promise quick success:

• Mastering a skill takes about 10 years or 10,000 hours; programming is no different.

• More: http://norvig.com/21-days.html

Peter Norvig’s recipe for success:

• Get interested in programming, and do some because it is fun.
• Talk to other programmers; read other programs.
• Program.
• Get a degree in computer science. (Optional)
• Work on projects with other programmers.
• Work on projects after other programmers.
• Learn 6+ languages.
• Understand the computer running the programs.
• Get involved in language standardization.
• Get out of language standardization.

What if you don’t want to be an expert?

• Computer science is prevalent in more and more fields:
  – Biology: Computational biology, bioinformatics
  – Physics: Simulations
  – Math: Theoretical computer science
  – Linguistics: Computational linguistics
  – Sociology: Social network analysis
• Any knowledge of programming is handy.
• CIS 200-series can help with this, too.

Tentative Course Outline

• Week 1: ArrayIntLists (Ch.15)
• Week 2: ArrayLists (Ch.10)
• Week 3: Linked lists (Ch.16)
• Week 4: Maps (Ch. 11)
• Week 5: Inheritance (Ch. 9)
• Week 6: Recursion (Ch. 12)
• Week 7: Searching and sorting (Ch. 13)
• Week 8-9: Binary search trees (Ch.17)
• Week 10: GUIs (Ch. 14)