Session Plan

- Administrative overview
- Introduction to course content
About the Course: Description

- Lectures: Monday/Wednesday 2:40-3:55 PM Mudd 633
- We will study:
  - commonly used data structures and algorithms,
  - how to analyze these data structures and algorithms.
About the Course: Staff

*Bert Huang*
Office hours: Monday 4-6 PM (after class)
CEPSR/Schapiro Building 624
bert@cs.columbia.edu

*TA’s:*
Priyamvad Deshmukh, prd2112@columbia.edu
Manu N, mn2370@columbia.edu
Nikhil Ramesh, nf2241@columbia.edu
About the Course:
Reading

*Data Structures and Algorithm Analysis in Java, 2nd Edition* by Mark Allen Weiss.
ISBN-10: 0321370139
About the Course: Resources

- Course homepage: http://www.cs.columbia.edu/~bert/courses/3137
- Courseworks: http://courseworks.columbia.edu
- Textbook Source Code: http://users.cs.fiu.edu/~weiss/dsaajava2/code/
About the Course: Prerequisites etc.

- COMS W1007: Object-Oriented Programming and Design in Java (or equivalent)
- Co-requisite COMS W3203: Discrete Mathematics
- COMS W3134: Data Structures and Algorithms (less intense but covers similar topics for non-majors)
About the Course: Grading

- 50% Homework Assignments (six)
- 20% Midterm Exam (closed book, closed notes)
- 30% Final Exam (closed book, closed notes)
About the Course: Academic Honesty

- You **must** read the Computer Science department’s academic honesty policy listed at http://www.cs.columbia.edu/education/honesty/

- Additional Comments:
  - Plagiarism is easy to catch.
  - All homework and exams in *this class* are individual assignments. **No collaboration.**
About the Course: Expectations

- Attend class
- Ask questions
- Read assigned text
- Start homework early
- Write well and clearly
- Get help when you need it
About the Course: Grievances

- Write reports of grading disputes on paper
- Provide clear explanation of the disagreement
- Give report to TA, TA will decide if correction is warranted
- If there is still disagreement, submit grading dispute report to me
Definitions

- Data Structure - abstract way to organize information
- Algorithm - abstract way to perform computation tasks
Data Structures

- Variables: boolean, int/byte/short/long, float/double, char
- Arrays, Strings
- We’ll go over more advanced structures: linked lists, trees, heaps, graphs, hash tables, etc.
- Smarter data structures can be abstracted
Benefits of Abstraction

- Consider Java Strings
  - We use them all the time
  - How is the text in a String object stored?
  - When we call the length() method, how does it find the length?
  - How does it concatenate strings?
Course Goals

- A series of case studies on common data structures and algorithms
- Gain intuition about how to design useful and efficient data structures
- Understand how to analyze any data structure or algorithm
Algorithm Analysis

- We must analyze algorithms’ and data structures’ running times and memory requirements.

- Input data nowadays are huge. Need efficient algorithms.

  - Over 100 million facebook.com users with profiles, photos

  - Google’s system indexes over 1 trillion (1,000,000,000,000,000) URLs
Next Class

- We will discuss how to formally analyze algorithms
- Big-Oh notation
Reading

- Course Website:
  http://www.cs.columbia.edu/~bert/courses/3137

- Academic Honesty policy
  http://www.cs.columbia.edu/education/honesty

- Weiss Chapters 1 and 2

- Ch. 1 should be about 75% review