Programming with Data Structures

CMPSCI 187

Spring 2016

- **Please find a seat**
  - **Try not to leave empty seats (the room will be pretty full!)**
- **Turn off or silence your mobile phone**
- **Turn off your other internet-enabled devices**

Today's class

- Overview of course topics
- Course staff
- Course logistics, policies & grading
- Brief intro:
  - Program testing & JUnit
  - The Eclipse IDE

Getting into CS 187 (if you aren't enrolled)

- The class is full.
- The College has an override list and all enrollment decisions are made based on this list.
  - Online override form
- If you've submitted the override form but haven't been enrolled, come to me after class to sign in.

What are Data Structures?

Anyone?

What are Data Structures?

Ways to store (complex) data, and operate on it in specified ways

- Our life is surrounded by all sorts of data: files, photos, videos, 3D models, social network… We need efficient ways to represent and store data. Examples:
  - A collection of names -> Arrays
  - Sparse data sets -> Hash tables
  - Hierarchical / organizational data -> Trees
  - Social network -> Graphs

What are Data Structures?

Ways to store (complex) data, and operate on it in specified ways

- Algorithms that operate on the data:
  - Sorting
  - Searching
  - Add / delete / modify
- Think about Java classes:
  - Objects: where data are stored
  - Methods: algorithms

What we will learn in this class

- Learn several important data structures and implement them using Java
- Improve your general programming skills
- Learn about and use Java-provided data structure implementations
- Compare implementations of data structures
- Learn when to use which data structures

Syllabus (topics)

1. Java review
2. Abstract data types
3. Complexity of algorithms
4. Stacks
5. Recursion
6. Queues
7. Lists and binary search
8. Binary search trees
9. Heaps, priority queues
10. Graphs
11. Sorting and searching
12. Hashing

Class structure

Lectures (T/Th or M/W)

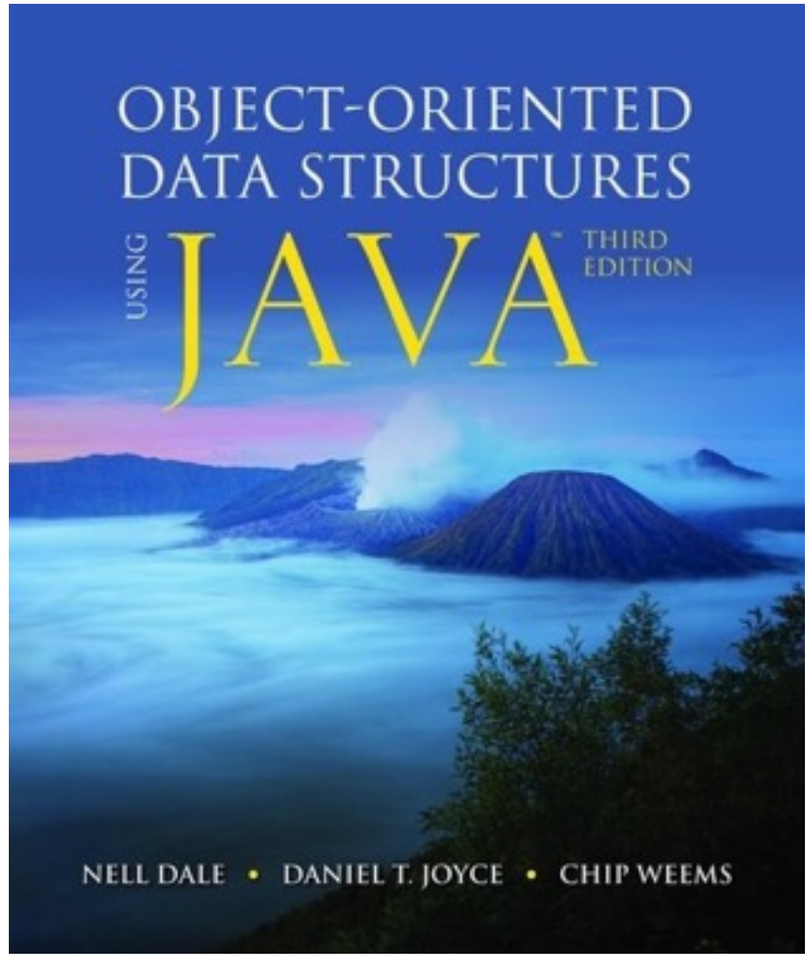
- Will hit the high points of material in the book
- Plus additional material not in the book
- Fun and games (and course credit) with i-clickers

Discussion (Mondays)

- Focused sessions
- Typically an in-depth problem related to material
- Usually more informal
- In the Edlab / LGRC A310

Textbook

- Object-Oriented Data Structures Using Java by Dale, Joyce and Weems (Jones & Bartlett, ISBN 978-1-4496-1354-9)



- Be sure you have the **3rd edition.**
- You can rent it inexpensively (you don't need to buy it)
- Get the book quickly
  - You need to be reading it and using it for homework.

Course staff

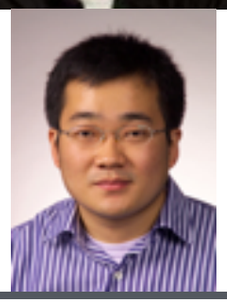
We're here to help you succeed in this course.

- **Come see either of us in office hours.**

- **One TA will lead your discussion section.**
- **You can see any TA for office hours.**
- **Any TA may answer your questions online.**

Mark Corner
instructor, sec 01

Rui Wang
instructor, sec 02



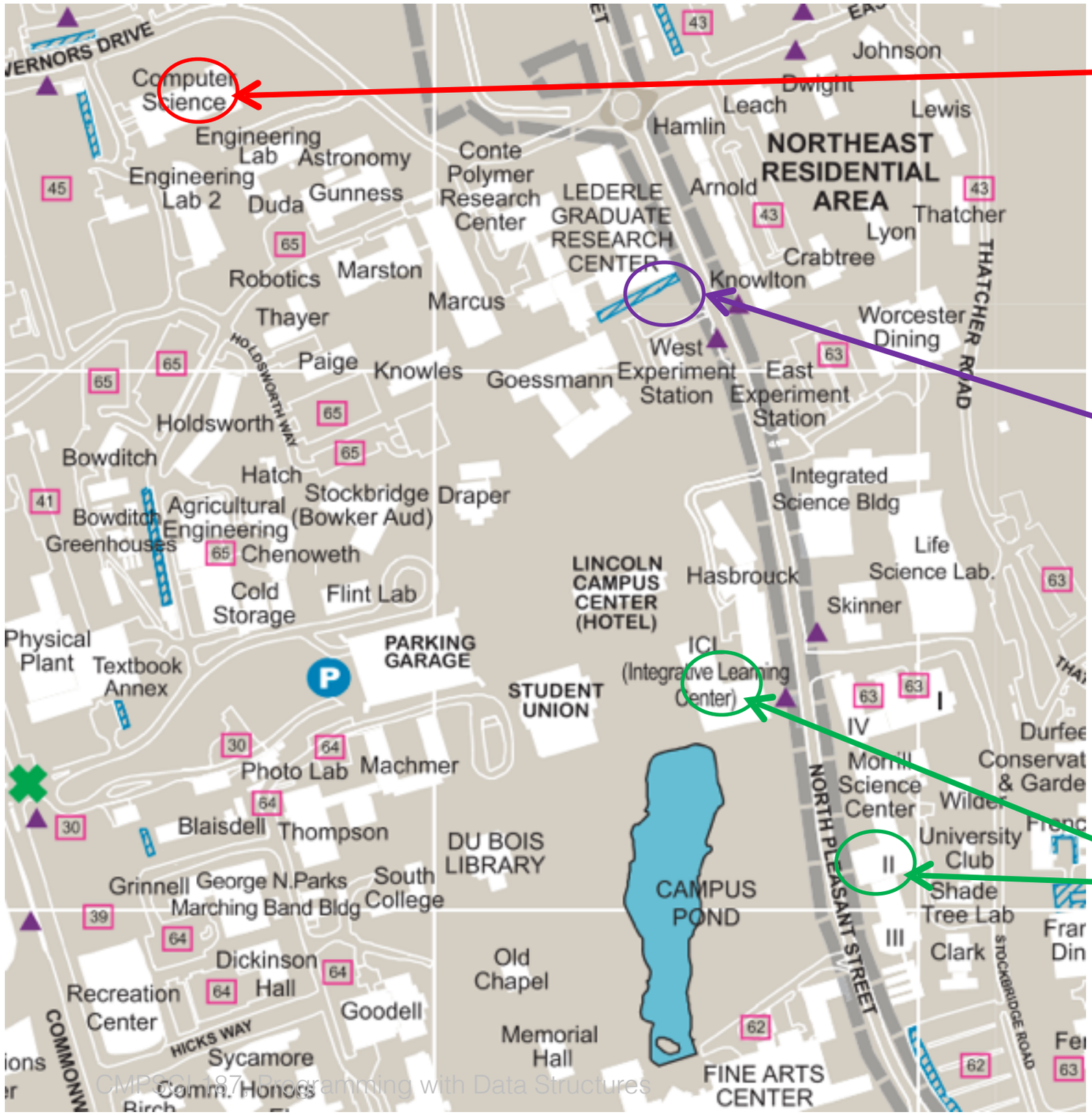
Venky Murthy
TA

Daniel Masi
TA

Kristina Fedorenko
TA

Yamin Tun
TA

Orientation



Office hours

Discussion sections

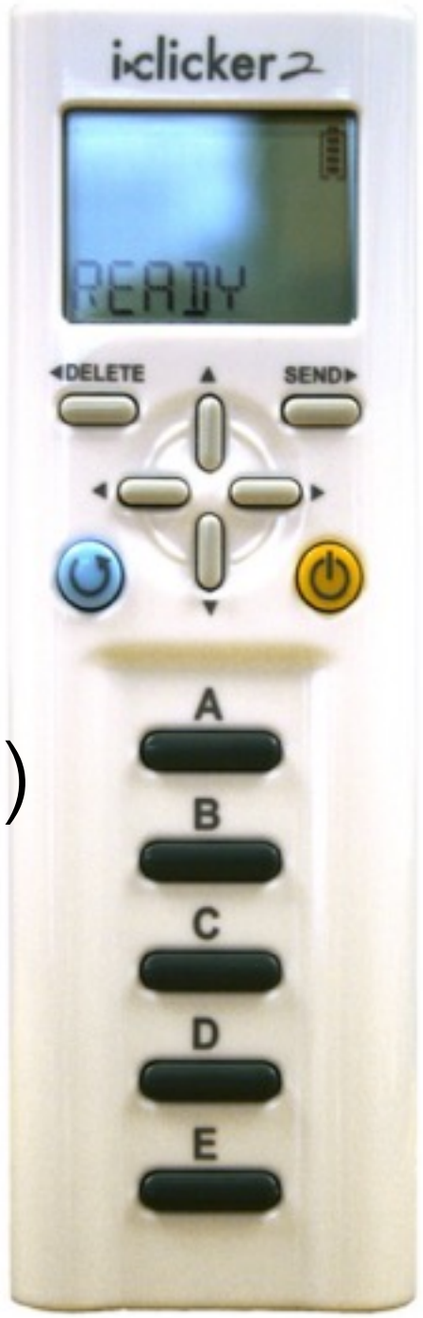
Lectures

Discussions

- LGRT 223 and 225
- LGRC (lowrise) A310
  - A310 has no computers
- Go to the right one!

i-clicker2

- **Required** for this class
  - no excuses for not having it
- Bring to every lecture (not discussions)
  - Questions start next class
- Set to correct frequency (look at room poster)
  - Hold power button for 3 seconds
  - When it blinks, select correct 2 letters
  - You should get a check mark



i-clicker – Let's try it out

What O/S will you use for programming assignments in this class?

A. Windows
B. Mac
C. Linux
D. Any of the above
E. I haven't the foggiest idea

Another question

What experience do you have with the Eclipse IDE?

A. None
B. I tried it and gave up.
C. I've used it a little.
D. I've used it extensively.
E. I write Eclipse plug-ins.

Course website

- Most course materials are on the public website:
  - http://umass-cs-187.github.io
  - syllabus, policies, schedule, assignments…
  - Please check this site regularly

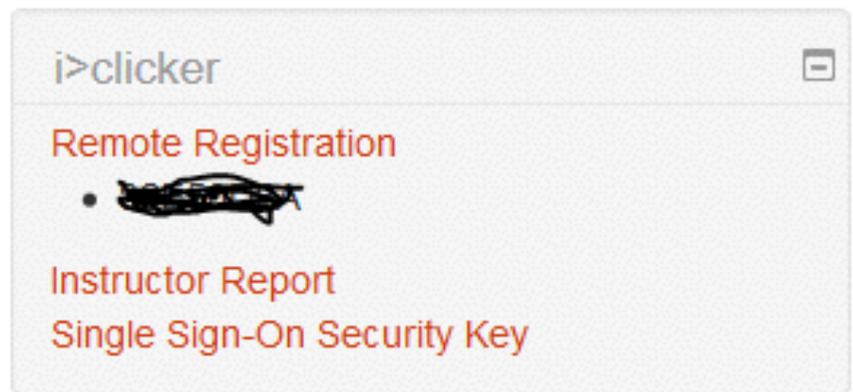
Moodle

http://moodle.umass.edu

- Log in with your UMass username and password.
- If you're registered (in SPIRE) you should see this course listed.
- Only used for grade management and posting solutions to projects/exams.

Log in soon and register your i>clicker2 in Moodle

- Bottom left of Moodle's screen
  - Mine looks like the picture there:
  - (Not at the iclicker site)
- You won't get credit until you do this!



Piazza

Almost all course communication will go through Piazza

- Post your questions.
- Contact instructors & TAs.
- Answer other students' questions.
  - TAs and instructors will be monitoring closely.
- You will receive **email with an invitation** soon. If you don't, contact us.

Office hours

- Instructor office hours
  - Rui Wang: CS 270
  - Mark Corner: CS 338
- TA office hours held in the LGRT Edlab 223
- Check website for days and times.

Asking for help

Try to use Piazza whenever possible.

Public posts to Piazza www.piazza.com	Any general questions that others could benefit from.
Private posts to Piazza	Any questions/issues not appropriate for Piazza. You can choose to send to only TAs and instructors.
Email your TA	Specific questions pertaining to discussion section.
Email instructor	Personal or serious issues that need the instructor's attention.

Grading

Percentage	Component
7.5%	iClicker
7.5%	Discussions
25%	Projects
20%	Midterm 1
20%	Midterm 2
20%	Final exam

Evening, TBD

Evening, TBD

Exam period, TBD

Programming assignments

- Substantial programming problems, typically extending starter code provided to you.
  - Assignments come with **automated tests** your code should pass (these are the *public tests*).
  - When done, you submit your code to the online autograder, which will check both the public tests and additional *private tests* and give feedback.
    - You can submit prior to the deadline, once per hour. Details on the course webpage.
- Due on (most) **Fridays at 4pm**.
- **First programming assignment is due Friday, 4pm**

Late policy

- **Deadlines are hard and non-negotiable**
- **Only exception: 3 late days available total**
  - Use them for exceptional circumstances
  - These are integral days (even if you are 1 minute late, it's counted as 1 day late)
  - These are calendar days (holidays all count)

Why people fail this class

- Phrasebook Java: Know Java really well, not just what it looks like
- Not knowing what they don't know: Come to class and discussion sections
- Trying to learn material by osmosis: Passive learning doesn't work
- Not practicing enough: Knowing syntax is important but not enough
- Poor preparation: <B in 121 probably mean a failing grade in 187
- Academic Dishonesty

Academic dishonesty

We take this very seriously. It can have a negative impact on your course grade, your GPA, and your overall record at UMass and beyond.

- You may discuss assignment problems with others in this course, however, your writing (including code) of solutions must be your own.
- Copying any material directly from the web is considered dishonest.
- Copying or using sections of someone else's program or assignment, even if it has been modified by you, **is not acceptable**.

UMass Code of student conduct

Academic dishonesty

We take this very seriously. It can have a negative impact on your course grade, your GPA, and your overall record at UMass and beyond.

- We will be using automated and manual means for detecting software plagiarism.
- When in doubt, contact the instructor about whether a potential action would be considered plagiarism.
- **Last Fall failed >10 students for academic dishonesty, prevented graduations, AOCs, etc.**

UMass Code of student conduct

Clicker Question #2

- I worked with another student on a programming project. We wrote a few lines of code that are the same. Is this academic dishonesty?

a) Yes

b) No

Answer on next slide

Clicker Question #2

- I worked with another student on a programming project. We wrote a few lines of code that are the same. Is this academic dishonesty?

a) Yes

b) No

Programming environment

- Eclipse IDE 

- Required. *You must use Eclipse to complete projects for this course.*
- You will download and install the latest version of Eclipse (Mars) as part of Project 1.

First discussion

- Next Monday
- Meet in room scheduled on SPIRE
- Purpose is to learn about Eclipse, Java testing, and prepare for the second assignment.
- Bring a laptop if you are in LGRC A310 (i.e. not in the Edalb LGRT 223/225), because A310 does not have computers.

First programming project

- Install Eclipse, import code, run JUnit tests, fix code to pass tests
- submit project through the autograder.
- Up on the website now! Get started!
  - Download PDF
  - Download starter code (hamspam.zip)
- Due this Friday, 4pm

Introduction to software testing

- What is testing?
- Why is it important?
- Why should you care?
- Your assignments depend on it!

Program testing example

- Programs often fail on special cases of inputs; testing should identify these and check them.
- When testing a method that sorts an array of integers, we may want to test the method on these special cases:
  - array is empty
  - array has one element
  - array is already sorted
  - array is reverse sorted
  - all array elements are the same

Unit testing

- Checks an individual unit of source code (e.g. a method, class, interface) for correctness.
  - Specify inputs, check that output meets requirements
- Incremental testing: when writing a complex program, test incrementally. Do NOT wait till you finish the whole program to start testing.
- If our tests are well-designed, and our program passes all tests, we have strong evidence program is correct.

Testing in this course

- To facilitate auto-grading, this course uses testing to evaluate your assignment submissions.
- Test cases are used to examine many aspects of your submission. Each test case represents an important testable piece of your program.
- We will provide ***public*** test cases that you can use to evaluate your implementation before submission. After submission, additional ***private*** test cases will be applied during the grading process.
    - Autograder will provide feedback and a score.

JUnit

- JUnit is a testing framework for testing Java code.
- JUnit test cases focus on individual classes, however, they can be used to test the interaction between groups of classes.
- Eclipse supports JUnit directly and provides a visual report of tests that pass/fail.

Eclipse & JUnit

- Next, we will show you how to import the first assignment into Eclipse, run the code, and run the JUnit tests to test your code.

[In class demo]
import hamspam project
run JUnit tests

Things for you to do

- On course webpage, read:
  - syllabus, course policies, course management, and academic honesty.
- Make sure you've received a Piazza invitation by email and that you've logged in.
- Login to Moodle:
  - Register your iClicker
- Start on Assignment 1.