Michael Lee

- lee@iit.edu
- http://moss.cs.iit.edu
- Office hours: Tue/Thu 9:30-12:30 (by online appointment)
Agenda

- Course overview & Administrivia
  - Prerequisites
  - Topics & Resources
  - Grading
  - Dev environment & Class procedures
Data Structures
- How do we store, organize, and retrieve data on a computer?

& Algorithms
- How can we efficiently (in space/time) carry out some typical data processing operations?
- How do we analyze and describe their performance?
Prerequisites

- I assume you are …
  - fluent in some programming language
  - familiar with procedural & OO paradigms
- comfortable with development processes:
  - compilation, debugging, testing
Python

- We’ll use the Python programming language to explore data structures & algorithms
- Easy-to-learn, clean (“one obvious way to do” things), and popular language
- Ton of useful, powerful libraries
Topics

- Python crash course
- Algorithmic analysis
- Linear data structures (Lists, Stacks, Queues)
- Hashing and Hashtables (aka Maps)
- Recursion and Trees
Class Resources

1. Course website: moss.cs.iit.edu/cs331
   - static information
   - lecture calendar, slides, links to external resources, etc.
2. Learning platform: **Mimir**

- lab notebooks with built-in tests
- quizzes & exams
Class Resources

3. Google Colaboratory
   - interactive lecture notebooks
Class Resources

4. Blackboard
   - Final gradebook
5. Discord: discussion forum
   - text/voice chat + screen share
Supplements

- The Python Tutorial (docs.python.org/3/)

- Problem Solving with Algorithms and Data Structures Using Python
Grading

- 35% Machine Problems
- 60% Exams (3 total: 2 midterms + final)
- 5% Quizzes / Self-evaluation
Machine Problems

- New programming assignment most weeks
- All assignments are retrieved and submitted on Mimir
  - Provided codebase typically covered in preceding lectures
- Late policy: 7 total late days for the entire semester, distributed as you like — once out, no late work accepted!
On Exams

- Tentative dates are on course website (Mar 4, Apr 8)
- All exams are cumulative (but will focus on recent material)
- Will be administered online, to be taken on your own computer (or in lab) — there will be no live lecture on exam days!
- Later exam scores, if higher, will improve earlier ones
```python
gap scores = [60, 80, 75]
gap [max(scores[i:]) for i in range(3)]
[80, 80, 75]

>> scores = [75, 80, 100]
>> [max(scores[i:]) for i in range(3)]
[100, 100, 100]
```
Quizzes

- Typically involve modifications/applications of data structures implemented in lab
- Released periodically on Mimir
  - Complete and submit by due date; late days not applicable
Jupyter Notebook

- In-browser Python development platform
  - “Cells” can contain plain text, code, output (and more)
  - All lecture notes, demos, and assignments will be distributed as notebook files
Jupyter Notebook

- You can optionally install a notebook server on your own computer (we recommend doing this!)

- See http://jupyter.org/install.html — either JupyterLab & “Classic” Jupyter Notebook are fine (with Python3)

- You may instead use Google Colab, a free, in-cloud Jupyter-based platform
Interactive Lectures

- Lecture notebooks available in course repository
- Open on Google Colab or local Jupyter installation
  - Class is usually one long interactive demo
- Starter and completed lecture notebooks are available in the course GitHub repository
§ Demo