Me

- Instructor: Professor Shlomo Hershkop (shlomo@cs.columbia.edu)
- My Research
- Office hours:
  - Mondays 11-12:30
  - AIM: Prof Hershkop
- Class website:
  - cs.columbia.edu/~sh553/teaching/w3134-s07/
  - Check it regularly (at least twice a week).
- Meet twice a week
  - Please show up on time
Welcome! Any idea what this is?
Overview

Today:
- Basic overview of the course and objectives
- Programming Environment Basics
- Java Intro

Goal:
- Thing are much easier if everyone knows why they are here, and what we are trying to accomplish.
- Interactive course
- We will learn about programming ideas while trying to have fun.
What is 3134?

- **CS3134:**
  - Basic DS for non CS majors

- **Prerequisites:**
  - Basic computer science background
  - Basic idea of Java

- **Goals:**
  - Not enough to know how to just write a program, need to know how to analyze and choose which structures work best for a specific task
Example

- How long do you expect to wait for a result from searches online?
So what are we going to be doing?

- Learn about using and analyzing basic and advanced programming data structures
- Learn basic algorithm analysis
- Learn Java programming
- Learn to solve practical programming problems correctly
  - Anyone can hack into the your machine
  - Learn to do it efficiently
Resources

- TA:
  - Jonathan DePeri
  - Matt Chu
  - Sara Stolbach

- Since we are a small class 😊
  - Please take advantage of the web board
    - Ok to post general type questions
  - Seek help before getting overwhelmed
    - Especially in the beginning
  - Detailed questions or code snippets please aim/email to TA or Instructor

- Open office policy
Requirements

- Interest to learn about Computer Science
- Learn to use cool DS
- Learn to make your own program work better
Textbook

Textbook can be acquired online or at the Columbia Bookstore.
- Else: borrow, threaten, or ‘acquire’ a book

Required:
- Mark Allen Weiss
  Data Structures and Algorithm Analysis in Java
  2nd edition

Recommended:
- Any java background book or internet connection
Reading

- I will be posting reading on the website and in class notes

- Please try to keep up with the reading
  - I will try to make up examples for class, but there are random stuff which the book covers which is good to see in print
  - Feel free to ask questions from anything you read/see/imagine in the book
Course Structure

- 6 Homeworks – 120 points
  - Will have about 2 weeks per homework
- Midterm – 30 Points
- Final (90 points)
  - open book, in class

Homework is important:
- Firm believer in hands on learning
- Start early
- Come to office hours, and ask questions
  - We are here for YOU!
Class participation and Attendance

- Attendance and participation is expected
  - Very interactive lectures & Labs
  - More class time will make your life easier
  - Class anonymous feedback system

- If you have to miss class, I expect you to catch up.
  - There will be class notes posted to the website
  - There will be many examples in class only, so make sure to get someone’s notes
  - Jokes are only funny first time around
Homework & Projects

- Homework will consist of
  - Written:
    - Theory, will be collected at first class after HW deadline.
  - Programming:
    - Online submission
    - Must be able to run on cunix system (this is important).

- Late policy:
  - Theory is required to be on time
  - Programming, will grant extensions for reasonable requests....but you have to start early
  - Limit how late, I can’t discuss answers in class if your going to be late.
Cheating Policy

- Plagiarism and cheating:
  - I’m all against it. It is unacceptable.
  - No patience for it
- You’re expected to do homeworks by yourself
  - This is a learning experience.
  - You will only cheat yourself.
  - My job is to help you learn, not catch you cheating, but....
- Automated tools to catch plagiarizers
  - [http://www.cs.berkeley.edu/~aiken/moss.html](http://www.cs.berkeley.edu/~aiken/moss.html)
  - Moving stuff around, renaming, etc. doesn’t help
  - The only way to fool the system is to spend about double the amount of time it takes to do the assignment itself 😊
- Cheating: instant zero on assignment, referral to academic committee
  - Columbia takes dishonesty very seriously
- I’d much rather you come to me or the TAs for help rather than resort to other stuff
Feedback System

- Last minute of class will be set aside for feedback:
  - Please bring some sort of scrap paper to class to provide feedback.
  - Feel free to leave it anonymous.
  - Content: Questions, comments, ideas, random thoughts.
- I will address any relevant comments at the beginning of each class
- Please provide feedback:
  - Going too fast
  - Going too slow
  - Missed something
  - Like something
- Please feel free to show up to office hours or make an appointment at any time
Shopping List

- You need a columbia cunix account
  - log into cunix.cc.columbia.edu
- Check out the class page
- Make textbook plans
  - try keep up with the reading
Hope you have fun learning the material

Any Questions?
Survey 1

- Please take a survey and take a minute or two to fill it out
Question:

- What is an algorithm
- What is a Heuristic?
Definitions

- **Algorithm:**
  - Problem solving method to be used to solve a problem independent of particular computer or program
  - Central objects of study in computer science

- **Heuristic**
  - In CS it is an Algorithm which is not guaranteed to find a solution
  - we will be studying algorithms which guarantee a solution with some constraints
most algorithms involve organizing data in a specific way and supporting a specific set of operations

These are called Data Structures

- will start with simple ones
- study analysis techniques
- combination of structures
solving a problem

- once you outline a problem to be solved by a computer
  - choice of language
  - choice of approach

- for small problems exact solution might not make a big difference

- for huge problems, sometimes a specific solution might take too long, and we are trying to get it solved faster
simple approach

- Throw money and buy faster computer
  - might give you 10 – 100 times speedup

- Study the algorithms
  - might give you a million times speedup
Connectivity Example

Given a pair of relationships between items, we want to know if a relationship can be inferred for a new pair a,b

- 3-4
- 4-9
- 8-0
- 2-3
- 5-6

? 2-9 ?
Graphical Example
Another Example

- Here is a picture of my car this morning

- Yes, I do live in brooklyn
Applications

- Crime statistics – learning patterns
- network communications
- circuitry
- mapping software
- variable name equivalence
- telephone network
- computer chip design
basic idea

- outline the problem
- understand clearly what kind of questions you are answering
  - don’t do all the work only to discover you can’t answer the question at the end
- understand the resource requirements
Sample Problem

- Have a collection of index cards with everyone’s names on it

- I want to organize it in alphabetical order

- Any ideas ??
Straightforward

- Find first name in list by going through it
- Find next
- etc

- Feels slow, how?
Creative Approach

- Throw list in the air and make a new pile

- Will this ever find a solution ??

- any better ?
Fastest Solution

- Take a random name
  - can throw in air if you wish

- Sort into two piles

- Redo from start

- known as quicksort, will cover when we cover sorting routines
Measurements

- **Time**
  - When designing an algorithm, think how fast it will run....then prove it

- **Space**
  - how much memory will it take up ?
  - important since we tend to treat memory as infinite

- **Complexity**
  - how easy it is to understand
    - given two algorithms, one complicated and one clear, tend to prefer the clear one
Java Review

- Would like to review relevant Java

- Even if you have some java experience will be helpful to stay awake
Programming Environment

- Java is an object oriented language
- You type code as text file
- Compile into machine independent byte code
  - Java byte code
- Java virtual machine interprets the byte code into local machine code
- “Write once, run anywhere”
Life easier – Step 1

- We will be using eclipse as an IDE

- This will automate many tedious java related stuff
  - Very important to understand what it is doing
  - Feel free to mess around and explore its capabilities