Course Information Online


- Sakai

A is for …

- **Algorithm**
  - This course is: Data Structures + Algorithms

- **API**
  - Using Libraries is the future

PFTD: Plan For the Day

- Be able to articulate why 201 is the right course for you, in terms of being able to complete it with understanding
  - What are pre-requisites?

- Be able to explain what work is expected, collaboration policies, exams, discussions, assignments, APTs
  - Why are you taking Compsci 201?
PFTD: Part II

- Be able to read some Java programs and to analyze them by applying your knowledge of programming to Java programs
  - Analyze for understanding and prediction

- Know what work you should complete before January 13, 15, and 16
  - What is due, when is it due

What is Computer Science?

“Our species needs, and deserves, a citizenry with minds wide awake and a basic understanding of how the world works.”

-Carl Sagan

Algorithms and Data Structures

<table>
<thead>
<tr>
<th>keys</th>
<th>hash function</th>
<th>buckets</th>
</tr>
</thead>
<tbody>
<tr>
<td>John Smith</td>
<td></td>
<td>00: 521-8976</td>
</tr>
<tr>
<td>Lisa Smith</td>
<td></td>
<td>01: 521-1234</td>
</tr>
<tr>
<td>Sandra Dee</td>
<td></td>
<td>02: 521-1234</td>
</tr>
<tr>
<td></td>
<td></td>
<td>03: 521-9655</td>
</tr>
<tr>
<td></td>
<td></td>
<td>13: 521-9655</td>
</tr>
<tr>
<td></td>
<td></td>
<td>14: 521-9655</td>
</tr>
<tr>
<td></td>
<td></td>
<td>15: 521-9655</td>
</tr>
</tbody>
</table>

Course Staff

- Teaching Associate: Kate O’Hanlon
- Teaching Assistants: Yongxin Tan, Carol Yang
- Head UTAs: Belanie Nagiel, Megan Phibbons, Charles Lyu, Daniel Hwang
- 26 UTAs: see course website
What is Computer Science?

- **Groups of** 2-4, Do NOT use a search engine

- How is this relevant to Compsci 201?
  - Tradeoffs in scaling Data Structures/Algorithms
  - WOTO

---

What is Computer Science?

What is it that distinguishes it from the separate subjects with which it is related? What is the linking thread which gathers these disparate branches into a single discipline? My answer to these questions is simple --- *it is the art of programming a computer*. It is the art of designing efficient and elegant methods of getting a computer to solve problems, theoretical or practical, small or large, simple or complex.

C.A.R. (Tony) Hoare

---

What is Computer Science?

- **It is the study of automating algorithmic processes that scale**
  - Long Division?
  - Updating Search Engine information?
  - Adding up a column of numbers?
  - Sending $$ via Venmo?
  - Updating your LinkedIn page?
Some Goals for 201

• Given a problem statement & a real data source, design, develop, debug, and test a Java program that uses appropriate standard libraries to efficiently solve the problem.

• Write programs that effectively implement and use data structures such as: arrays, maps, linked lists, stacks, queues, trees, and graphs.

More Goals for 201

• Evaluate the time and space complexity of algorithms, especially algorithms that scale, using empirical and mathematical analysis.

• Apply basic object-oriented design and programming principles in developing software

Who are you?

Latanya Sweeney

I am a computer scientist with a long history of weaving technology and policy together to remove stakeholder barriers to technology adoption. My focus is on "computational policy" and I term myself a "computer (cross) policy" scientist. I have enjoyed success at creating technology that weaves with policy to resolve real-world technology-privacy clashes.

http://latanyasweeney.org/

Identify 87% of US population using (dob,zip,gender). Prof. Government and Technology @ Harvard, instrumental in HIPAA because if de-identification work. Former CTO of the FTC
**Code Interlude**

- **First 201 Java program – maybe first ever!**
  - Use what you know
  - Make informed assumptions
  - When more time? Book, Internet, Friends, Brain

- **All code online: website and GitLab**
  - Browse, fork, clone, critique, …

---

**Understanding Repetition**

- When does loop terminate?
- What takes time when this code executes?

```java
HashSet<String> set = new HashSet<>();
int wcount = 0;
double start = System.nanoTime();

while (s.hasNext()) {
    wcount += 1;
    String word = s.next();
    if (!set.contains(word)) {
        set.add(word);
    }
}
double end = System.nanoTime();
```

---

**Java Variables and Types**

- **Variables:** name, type, value
  - Primitives: int, double
    - `wcount`, `start`, `end`
  - Object: String, Scanner, HashSet
    - `s`, `set`, `word`

- **For each variable/object:** name, type, value
  - What are the operations on these?
Understand and Explain

• "kjv10.txt" with 823,135 words, 34,027 unique
  • About one second to run on rodger laptop

• Double, Triple, Quadruple original file
  • Total # words different, # unique same
  • N words in "kjv10", 2N, 3N, 4N respectively

• Runtimes: 1, 1.5, 2.0, 2.5 -- pattern similar
  • How to justify empirical with analytical analysis?

From Analysis to Code

• How to understand (read) and create (write)
  • Read book, use Google? read book, ask
    • What book? See syllabus
  • Practice, practice, practice

• We will look at a high-level today, more details in Discussion Monday and Lecture Wednesday
  • Pre-discussion work

Java Types and Operations

• What can we do with int and double?
  • +,-,/, seen in code, many more
  • Initialize and update

• What can we do with HashSet?
  • .add(..), .contains(..), .size()

• What can we do with Scanner?
  • .hasNext(), .next(), .close()

Java Concepts and Conventions

• Classes and Objects
  • You invoke methods on ... call functions on ..
  • Object dot method: set.add(..) or s.hasNext() or set.size() or ...

• Variable identifiers begin with lowercase letter
• Class identifiers begin with uppercase letter
• Statements end in semi-colon, statement blocks created by braces: {}
Understanding Java and Scale

- Java is an object-oriented language
  - More on classes and objects later
- Primitive types: int, double, char, boolean, ...
  - Fast, small, values stored in memory
- Arrays are homogeneous collections
  - Like Python lists, Matlab array/vector
  - Once created, don’t grow, can hold primitives
- ArrayList<>, Set<>, Map<> are collections
  - Dynamic, powerful, scale, no primitives

Analyzing Code, Algorithm

- What file of 1,000,000 strings will result in this code executing most quickly?
  - Characteristics of file? Bottleneck of method?
- What file of 1,000,000 strings will result in this code executing most slowly?

Tradeoffs

- What line is “bottleneck” in this code/algorithm?
  - How can we make this faster?
  - Why is Google so fast when searching?
- Replace HashSet<> with ArrayList<>?
  - Why does code still run?
  - Common API, valuable for programmer!!
Course Logistics

- Please see course website for most material/dates

- For grades
  - Sakai:

- Exams:
  - February 14, April 3, April 30 (final)

Succeeding in 201

- Come to class and participate: WOTO
  - Only one in the group fills out the WOTO
  - Put in all netids from the group

- Start work early, get help when needed

- Initial late penalties aren't harsh, on-time doesn't matter, but not doing assignments is a bad idea
  - Ask if you have any reason to ask, any reason

Course Logistics

- Please see course website for due dates/grading

- APT Quizzes, Midterm and Final Exams are assessments

- Assignments and APTs are where you will practice and learn the material on your own

- Discussion – practice and think together

- Please note collaboration policies, APT quizzes, late policies, why you should come to class

Success in 201 is …