Data into Information and Knowledge

What can you do with Compsci?

Foursquare lets me check in from Shooters to the White House in under an hour. So many reasons why that should prohibited...

15 Nov via web
What can you do with Compsci?

http://bit.ly/1hUNmz

What can you do with Compsci?

Hackers Select a New Target: Other Hackers

by DOMINIC SINGAPOTRA and NICK BILTON

Published: JUly 4, 2011

What can you do with Compsci?

Annals of Communications

A Woman's Place

Can Sheryl Sandberg upend Silicon Valley's male-dominated culture?

BY KEN AILETTA

July 11, 2011

Apple, Google, Facebook turn N.C. into data center hub

Cloud computing growth is bringing big benefits to small towns in western North Carolina.

By Patrick Thibodeau

June 5, 2011 08:00 AM ET

Google Buys Motorola Mobility for $12.5 Billion

Facebook usa by Sarah Kocker

426
4,341 people like this.
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 will you experience?

Programming != Computer Science

- What is the nature of intelligence? How can one predict the performance of a complex system? What is the nature of human cognition? Does the natural world 'compute'? 

- It is the interplay between such fundamental challenges and the human condition that makes computer science so interesting. The results from even the most esoteric computer science research programs often have widespread practical impact. Computer security depends upon the innovations in mathematics. Your Google search for a friend depends on state-of-the-art distributed computing systems, algorithms, and artificial intelligence.

Efficient design, programs, code

Tradeoffs, Tradeoffs, Tradeoffs

Using the language: Java (or C++, or Python, or ...), its idioms, its idiosyncracies

Object-oriented design and patterns. Software design principles transcend language, but ...

Know data structures and algorithms. Trees, hashing, binary search, sorting, priority queues, greedy methods, graphs ...

Engineer, scientist: what toolkits do you bring to programming? Mathematics, design patterns, libraries --- standard and others...
Course Overview

- **There are details, see the course web page**
  - Midterms and final are open book, what does that mean?
  - AFTs: Algorithmic Problem-solving and Testing
    - Weekly small programming assignments, tested online
    - Programming assignments: solo, paired, ...

- **Why should you come to class?**
  - Meet people, learn things, participate in a community
  - Provide help, get help, wonder, dance, think
  - Participate in graded exercises that affect your final grade

- **Why is this course so great?**
  - Because you’re in it

What’s in Compsci 100?

- **Understanding tradeoffs: reasoning, analyzing, describing...**
  - Algorithms
  - Data Structures
  - Programming
  - Design

- **Programming using Java**
  - Tools: Eclipse, JDK, Libraries, ...
  - Ideas: Design Patterns, OOP, Agile programming, ...
  - Engineering and analyzing designs and programs
  - Using mathematical and scientific techniques
  - Scaling solutions

Questions

If you gotta ask, you’ll never know
Louis Armstrong: "What’s Jazz?"

If you gotta ask, you ain’t got it
Fats Waller: “What’s rhythm?”

What questions did you ask today?
Arno Penzias

Tradeoffs

Programming, design, algorithmic, data-structural

Simple, elegant, quick, efficient: what are our goals in programming?
What does XP say about simplicity? Einstein?

Fast programs, small programs, run anywhere-at-all programs. Runtime, space-time, your time, CPU time...

How do we decide what tradeoffs are important?
Tension between generality, simplicity, elegance, ...
Language

Java

www.java.com/en/about/

From Blog to Scientific Visualization

- **Text Cloud aka Tag Cloud?**
  - Number of occurrences/emphasis indicated by size of word
  - Great visual/statistic: [http://chir.ag/phernalia/preztags/](http://chir.ag/phernalia/preztags/)

- **What is involved with generating tag clouds?**
  - Steps? Issues?
  - We will look at code to generate the tag cloud, but we'll look at simpler code first

Problem Solving and Programming

- **How many words are in a file?**
  - What’s a word?
  - What’s a file?
  - How do we solve this: simply, quickly, ...?
    - What’s the best we can do? Constraints?

- **How many different/unique words are in a file?**
  - How is this related to previous task?

- **How many words do two files have in common?**
  - Spell-checking, Google did you mean ..?
Fast, cheap, out-of-control?

- This is valid and correct Java code, questions?

```java
import java.util.*;
import java.io.*;
public class SimpleUnique {
    public static void main(String[] args) throws FileNotFoundException {
        Scanner s = new Scanner(new File("/data/kjv10.txt"));
        s.useDelimiter("\Z");
        String all = s.next();
        String[] words = all.split("\s+" );
        TreeSet<String> set = new TreeSet<String> ();
        set.addAll(Arrays.asList(words));
        System.out.printf("total #: %d, unique #: %d\n",
                words.length, set.size());
    }
}
```

total #: 823135, unique #: 34027

Pythonic WordCounting

- This is valid and correct Python code, questions?

```python
def countWords(filename):
    file = open(filename)
    str = file.read()
    words = str.split()
    unique = set(words)
    print "filename: ", filename
    print "total # words = ", len(words)
    print "unique # words = ", len(unique)
    countWords('/data/kjv10.txt')
```

filename: /data/kjv10.txt
total # words = 823135
unique # words = 34027

How fast is fast? How cheap is cheap?

- How do we measure how fast the code/design is?
  - Can we implement this design in C++?
  - Can we implement this in Ruby?

- We want a measure that’s independent of language?
  - What are we measuring?
  - How do we express answer?
  - Units? Best case? Average? Worst?

- What is answer using recognized terminology?

What is Computer Science?

- Computer science is no more about computers than astronomy is about telescopes.
  - Edsger Dijkstra

- Computer science is not as old as physics; it lags by a couple of hundred years. However, this does not mean that there is significantly less on the computer scientist’s plate than on the physicist’s: younger it may be, but it has had a far more intense upbringing!
  - Richard Feyneman

http://www.wordiq.com
Some Java Vocabulary and Concepts

- Java is a (statically) typed language
  - Every variable/object has a type that's defined at compile time (compare Python)
  - More typing, but more errors caught before runtime
- Java is an object-oriented language
  - All code is in a class and all code inside function/method
  - Classes are also types, so variables/objects extendible
  - We will see interfaces and inheritance later
  - Classes have instance variables, methods local variables
- Java has different 'types' primitive and Object/class. This is for efficiency, but it's sometimes confusing

More Java Vocabulary and Concepts

- Java has a huge standard library
  - Organized in packages: java.lang, java.util, javax.swing, ...
  - API browseable online, but Eclipse IDE helps a lot
- Java methods have different kinds of access inter/intra class
  - Public methods ...
  - Private methods ...
  - Protected and Package methods ...
- Primitive types (int, char, double, boolean) are not objects but everything else is literally an instance of class Object
  - foo.callMe();

Solving problems, writing code

- APT: AccessLevel
  - Understand the problem, know how to solve an instance
  - Ideas? Caveats?
- Writing code to implement proposed solution
  - Will it run? In time? Constraints? Look before you code
  - How will we test the solution? When to start testing?
- What's the green dance and when do we do it?
  - Satisfaction of finishing something
  - Knowing when to stop when you're not making progress
  - Leveraging community wisdom

Structuring Information: ideas & code

- Is an element in an array, Where is an element in an array?
  - DIY: use a loop
  - Use Collections, Tradeoffs?
  - public boolean contains(String[] list, String target){
    for(String s : list){
      if(s.equals(target)) return true;
    }
    return false;
  }
  - public boolean contains(String[] list, String target){
    return Arrays.asList(list).contains(target);  
  }
  - public boolean contains(String[] list, String target){
    new HashSet<String>(Arrays.asList(list)).contains(target);
  }
Basic data structures and algorithms

- Arrays are typed and fixed in size when created
  - Don’t have to fill the array, but cannot expand it
  - Can store int, double, String, Object, ...

- ArrayList (and related interface List) grows s needed
  - Stores objects, not primitives
    - Autoboxing in Java 5 facilitates int to/from Integer conversion
    - Store an int in an ArrayList, get one out
    - Can’t convert entire ArrayList to array for int, can for String, why?
  - ArrayList objects grow themselves intelligently

- java.util package has lots of data structures and algorithms
  - Use rather than re-implement, but know how do to do both

Who is Alan Perlis?

- It is easier to write an incorrect program than to understand a correct one
- Simplicity does not precede complexity, but follows it
- If you have a procedure with ten parameters you probably missed some
- If a listener nods his head when you’re explaining your program, wake him up
- Programming is an unnatural act
- Won first Turing award

http://www.cs.yale.edu/homes/perlis-alan/quotes.html

Analysis of StaticSimpleWords.java

- How are variables used?
  - What types are there, definitions, re-used variables,
  - What is a File, a Scanner, a String[], a HashSet, a ...

- How are strings and values printed?
  - What is System.out? How is it used?

- Why is there a FileNotFoundException thrown from main?
  - What is an exception, how do we handle them?

- How are statements organized, how is whitespace used?