

Class 01: An Introduction to Algorithms

Held: Monday, January 25, 2010

Summary: We begin the class by exploring the definition of computer science and by trying to write some basic algorithms.

Related Pages:

- EBoard.

Notes:

- Welcome to 151.
- The Course Web site will be in a bit of flux for the first few days.
- I will take attendance for the first few days, but I will probably take an month to attach names to faces and perhaps longer to learn to pronounce names correctly.
- Two assignments for Tuesday: (1) Take the introductory survey; (2) Read Grinnell's Linux Environment.

Overview:

- Introduction: What is CS?
- Exercise: An everyday algorithm.

An Introductory Question

- Rather than telling you exactly what the class is about, I'm going to start the class with a question for you.
 - The question will help ground the class.
 - The question will begin to get you used to some aspects of my teaching style (particularly my reliance on recitation-style work).
 - The question will test your abilities as a liberal artist.
- *What is Computer Science?*
- I will give some of my own responses after I've heard some of yours.

What is CS?

- Computer scientists differ in how they define the discipline. However, most would agree that, at least in part,

Computer science is the study of algorithms and data structures.

- By “algorithms”, we mean sets of instructions that can be used to solve problems.
 - Some problems are mathematical. For example, you might write an algorithm to find the square root of a real number.
 - Other problems deal with textual information. For example, you might write an algorithm that tells how to find a name in the phone book.
 - You can write algorithms for a wide variety of problems.
- By “data structures”, we mean mechanisms for organizing information. For example, we organize some information in lists, other information in tables, and other information in more complex structures.
- By “study”, we mean specify, design, describe, evaluate mathematically, evaluate experimentally, implement in software, implement in hardware, prove properties, consider applications and implications, and much, much more.
- In our studies, we rely on the tools and techniques from a number of other disciplines.
 - From *mathematics*, we take proof techniques, formal language for describing problems and solutions, and even core ideas.
 - From *science*, we take experimental techniques.
 - From *engineering*, we take techniques for designing and constructing things.
 - From *psychology and the social sciences*, we take techniques for understanding the relationship of our work to human endeavors.
- These diverse perspectives and skill sets make CS interesting and challenging.
- Our European colleagues often refer to the discipline as “Informatics” (that is, implying the study of information), and I will admit that I have come to prefer the term because it distances us a bit from both computers (we are broader than the technology) and from science (since we do not necessarily emphasize the scientific method).

An Everyday Algorithm

- We’ll explore the problems of writing clear instructions through a simple exercise.
- Challenge: Write a clear, unambiguous, and detailed set of instructions for making a peanut butter and jelly sandwich.
- Format: Work in groups of about four.
- Each group will write its solution on one of the four boards.
- SamR will play the role of the sentient, but malicious and clueless “follower of instructions”.
- We will do some debriefing today, and additional debriefing in the subsequent class.

Copyright © 2007-10 Janet Davis, Matthew Kluber, Samuel A. Rebelsky, and Jerod Weinman. (Selected materials copyright by John David Stone and Henry Walker and used by permission.) This material is based upon work partially supported by the National Science Foundation under Grant No. CCLI-0633090. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation. This work is licensed under a Creative Commons Attribution-NonCommercial 2.5 License. To view a copy of this license, visit <http://creativecommons.org/licenses/by-nc/2.5/> or send a letter to Creative Commons, 543 Howard Street, 5th Floor, San Francisco, California, 94105, USA.