

CSC151.02 2014S, Class 01: An Introduction to Algorithms

Overview

- Preliminaries.
 - Admin.
 - Homework.
 - Questions.
- Introduction: What is CS?
- Exercise: An everyday algorithm.
- Debriefing on exercise.

Preliminaries

Admin

- Welcome to CSC 151. I hope you have a great time.
 - I'm Sam
 - Kim is our class mentor
 - We should be adding another class mentor
 - And we'll have some guest mentors for the first week or two (starting tomorrow)
- I anticipate this being a large class. (I agreed to go over the cap.) I will sign the add form of anyone who is here today.
- I will take attendance for the first few weeks of class.
 - I have 120 students this semester, so it's going to take me a little while to learn everyone's name.
 - Please respond with your preferred gender pronoun (he, she, or zhi). I'm a little hard of hearing, so please forgive me if I ask you to repeat yourself.
- I don't believe in going over class policies on the first day of class. Please read the course web and ask me questions via email or in person.
- Please try to participate actively in class. I know it's hard when you're first trying to get used to a professor, but please try.
- I'm trying to set up a Google calendar for this course. Please let me know what else you'd like to see on that calendar.
- Extra credit:
 - MLK Day Talk, tonight @ 7:30 in JRC 101.
 - Thursday extra on summer research in CS, Thursday @ 4:30 in Noyce 3821.
 - CS Table Friday at noon in the Day PDR in the Marketplace.
- Hi Kim!

Homework

- Your first assignment is due Tuesday night at 10:30 p.m. I have distributed it in class today.
- You also have readings for class tomorrow. You can find the described in today's assignment.
 - One reading is on algorithmic thought.
 - The other is on Linux, the operating system we use in this course.

Questions

What do you want for the EC?

Send me a short paragraph.

Introduction: What is CS?

- Computer science is the study of computers and the stuff inside them, such as digital information.
- Computer science is the study of computer language
- Computer science is the study of programming and otherwise manipulating computers
- Using programming languages and algorithms to solve problems
- Learn how to make computers work, because normally they don't
- Using computer languages to manipulate computers
- Some commonalities
 - "Study" - But how do we study?
 - Computer language
 - Manipulate computers
- To a computer scientist, computer science is the *study of algorithms and data structures*
 - Algorithm: Instructions for accomplishing a task
 - Data structures: Ways to organize information
 - Study -
- The scientific method - Observe the world, Form a hypothesis, Test (Design an Experiment and carry out the experiment), Gain some understanding of the hypothesis
 - Characteristics: Repeatable and reliable
- While we have the name, it's not generally how we study algorithms and data structures
 - Build them
 - Analyze for correctness
 - Scientific method: Try a lot of examples
 - Like mathematicians
 - Consider usability and social impact

Exercise: An everyday algorithm

Please give me instructions for making a nut butter and jelly sandwich.

- You should assume that I'm as clueless as computer science profs are reputed to be. I'm also lazy. So if you tell me to open something, I'll probably just throw it against the wall.
- I like careful details.

Debriefing on exercise

Note: We will continue this debriefing in the next class period.

What are some things you learned?

- When writing instructions, know what your audience knows.
 - Sometimes you have to be very detailed.
 - Be specific. If your audience can misinterpret you, it will.
 - When things go wrong, it's funny, not painful (well, except for me)
 - You write better algorithms if you can observe them working step by step
 - You write better algorithms when you write them with other people
 - Know your data - (Screw-top vs pop top)
 - Watch for infinite loops
 - Sam intentionally misinterprets directions
 - In my experience, computers do, too.
 - Keep instructions simple, if possible
 - Testing should be rigorous and thorough
-

Samuel A. Rebelsky, rebelsky@grinnell.edu

Copyright (c) 2007-2014 Janet Davis, Samuel A. Rebelsky, and Jerod Weinman. (Selected materials are copyright by John David Stone or Henry Walker and are used with permission.)



This work is licensed under a Creative Commons Attribution 3.0 Unported License. To view a copy of this license, visit <http://creativecommons.org/licenses/by-nc/3.0/> or send a letter to Creative Commons, 543 Howard Street, 5th Floor, San Francisco, California, 94105, USA.