



The Beauty and Joy of Computing

Lecture #6 Algorithms I

UC Berkeley
EECS Lecturer
Pierce Vullucci

BIG DATA AND MEDICINE

Health care providers have always used basic data about you to estimate treatment costs and evaluate your health but now other seemingly unrelated activities such as shopping and internet usage are being considered to act as proxy signals for relevant but harder to check factors of the patient.

<http://www.nytimes.com/2014/06/29/technology/when-a-health-plan-knows-how-you-shop.html?ref=technology>



What is an algorithm?

- An **algorithm** is any well-defined computational procedure that takes some value or set of values as input and produces some value or set of values as output.
- The concept of algorithms, however, is far older than computers.



UC Berkeley "The Beauty and Joy of Computing" : Algorithms I (2)



Early Algorithms

- Dances, ceremonies, recipes, and building instructions are all conceptually similar to algorithms.
- Babylonians defined some fundamental mathematical procedures ~3,600 years ago.
- Genes contain algorithms!



Photo credit: Daniel Niles



UC Berkeley "The Beauty and Joy of Computing" : Algorithms I (3)



Algorithms You've Seen in CS10

- Length of word
- Whether a word appears in a list
- Interact with the user (ask)
- Word Comparisons (You'll write one for HW1!)
- Sort a List (see lab!)



UC Berkeley "The Beauty and Joy of Computing" : Algorithms I (4)



Algorithms You Might Have Heard Of

Luhn algorithm

Credit card number validation

Deflate

Lossless data compression

PageRank

Google's way of measuring "reputation" of web pages

EdgeRank

Facebook's method for determining what is highest up on your news feed



UC Berkeley "The Beauty and Joy of Computing" : Algorithms I (5)



Important Terms

Sequencing

Application of each step of an algorithm in order (sometimes: find order)

Selection

Use of Boolean condition to select execution path

Iteration

Repetition of part of an algorithm until a condition is met

Recursion

Repeated application of the same part of algorithm on smaller problems



UC Berkeley "The Beauty and Joy of Computing" : Algorithms I (6)





Properties of Algorithms

- Algorithm + Algorithm = Algorithm
- Part of Algorithm = Algorithm
- Algorithms can be efficient or inefficient given a comparison algorithm
- Several algorithms may solve the same problem



UC Berkeley "The Beauty and Joy of Computing" : Algorithms I (7)



Algorithm Correctness

We don't only want algorithms to be fast and efficient; we also want them to be **correct!**

TOTAL Correctness

Always reports, and the answer is always correct.

PARTIAL Correctness

Sometimes reports, and the answer is always correct *when it reports*.

We also have *probabilistic* algorithms that have a certain *probability* of returning the right answer.



UC Berkeley "The Beauty and Joy of Computing" : Algorithms I (8)



How to Express Algorithms...

A programmer's spouse tells him: "Run to the store and pick up a loaf of bread. If they have eggs, get a dozen." The programmer comes home with 12 loaves of bread.

Algorithms need to be expressed in a context-free, unambiguous way for all participants

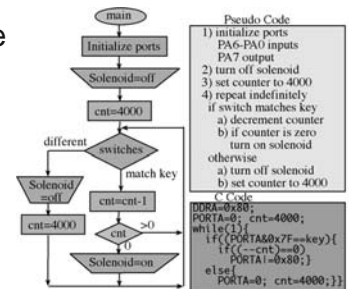


UC Berkeley "The Beauty and Joy of Computing" : Algorithms I (9)



Ways to Express Algorithms

- Natural Language
- Pseudo Code
- Programming Language
- ...or in any other information conveying way!



UC Berkeley "The Beauty and Joy of Computing" : Algorithms I (10)



Programming Languages

C/C++

Good for programming that is close to hardware

Java/C#

Portable code

Python/Perl/Tcl/TK

Fast to write and portable

BASIC/BYOB/SNAP

Good for teaching programming concepts

All programming languages can be used to implement (almost) any algorithm!



UC Berkeley "The Beauty and Joy of Computing" : Algorithms I (11)



Choosing a Technique

- Most problems can be solved in more than one way, i.e., multiple algorithms exist to describe how to find the solution.
- The right language makes formulating algorithms easier and clearer
- Not all of these algorithms are created equal. Very often we have to make some trade-offs when we select a particular one.
- There are unsolvable problems!



UC Berkeley "The Beauty and Joy of Computing" : Algorithms I (12)





Algorithms vs. Functions & Procedures

- Algorithms are conceptual definitions of how to accomplish a task and are language agnostic, usually written in pseudo-code.
- Find max value in list
 - A function or procedure is an implementation of an algorithm, in a particular language.

Find max value in list [1, 2, 3, 4, 5]



Garcia + Vollucci



UC Berkeley "The Beauty and Joy of Computing": Algorithms I (13)



Summary

- The concept of an algorithm has been around forever, and is an integral topic in CS.
- Algorithms are well-defined procedures that can take inputs and produce output. Programming languages help us express them.
- We're constantly dealing with trade-offs when selecting / building algorithms.
- Correctness is particularly important and testing is the most practical strategy to ensure it.
 - Edge Cases
 - Many write tests first!
 - TDD



UC Berkeley "The Beauty and Joy of Computing": Algorithms I (14)

Garcia + Vollucci

