CS10: The Beauty and Joy of Computing

Lecture #6
Algorithms

UC Berkeley EECS
Summer Instructor
Ben Chun

2012-06-26

A VOCAL TEST FOR PARKINSON’S

Announced at TEDGlobal 2012 on Monday, the new system is already 86% accurate. They need more recordings of voices to improve, so give them a call.

What is the world record?

For the standard 3x3x3 cube…

a) 56.12 s
b) 14.76 s
c) 7.96 s
d) 5.66 s
e) 3.31 s
Feliks Zemdegs

16 years old, current world record holder

2x2 average
3x3 single
3x3 average
5x5 average
An algorithm is any well-defined computational procedure that takes an input and produces an output.

The concept is older than digital computers.
Algorithms Everywhere

- Dances
- Ceremonies
- Recipes
- Building techniques

All are conceptually similar to algorithms!
Algorithms You’ve Seen

- Subtract each digit from the one above, moving from least to greatest place value, “borrowing from the neighbor” if necessary to get a positive result.

\[
\begin{array}{c}
67128 \\
- 51 \\
\hline
677
\end{array}
\]
Algorithms You’ve Seen

- Length of word
- Does a word appear in a list?
- Is a list sorted?
- Pick a random word of length x from a list
Commonly-Used Algorithms

- **Luhn algorithm**
  - credit card validation

- **PageRank**
  - Google’s way of determining the relevance of a page to a search term

- **Damerau-Levenshtein distance**
  - Spell checking

- **EdgeRank**
  - Facebook’s way of determining what appears in your news feed
Choosing a Technique

- Most problems can be solved >1 way
  - give a solution
  - But all algorithms are not created equal!

- Trade-offs
  - Time
  - Space
Three Ways to Attack Problems

- **Brute Force**
  Keep trying stuff until something works

- **Top-down**
  Divide the problems up into smaller problems, solve them, combine the answers

- **Bottom-up**
  Start with a simple solution and elaborate it until the full problem is solved
Algorithms vs. Functions / Procedures

- An algorithm is a conceptual definition of how to accomplish a task
- Language agnostic

- A function or procedure is an implementation of an algorithm
- Written in a particular language, can be run
Example: Finding the Maximum

**Scratch**

```
set max to 0
set count to 0
repeat length of list
  change count by 1
  if item count of list > max
    set max to item count of list
```

**Scheme**

```
(define (find-max L)
  (if (null? (cdr L))
      (car L)
      (if (> (car L) (find-max (cdr L)))
        (car L)
        (find-max (cdr L))
      ))
)
```
Correctness

- **Total Correctness**
  Always reports, and is always correct

- **Partial Correctness**
  Sometimes reports, and the answer is correct when it reports

- **Probabilistic**
  A certain probability of returning the right answer
Summary

- Algorithms are an old idea, integral to CS
- Definition: **well-defined procedure** that takes inputs and produces output
- **Trade-offs** usually can’t be avoided
- Correctness is important, and **testing** is a practical strategy to ensure this