

CSC151.02 2013F, Class 45: Binary Search

Overview

- Preliminaries.
 - Admin.
 - Questions on the project.
- The problem of searching.
- Making searching more efficient using divide-and-conquer.
- A demo: Destructive binary search.
- Considering the parameters to binary search.

Preliminaries

Admin

- Today's lab writeup (#27): Turn in 5
- There is no reading for Monday.
- Dr. Davis moved the quiz to next week, and so I will give it to you on Monday. We'll use the extra time today to talk a bit more about binary search.
- Note: If you're using turtles for part of your drawing, we'll understand if they don't take aspect ratio into account. (Everything else should take aspect ratio into account.)
- Why are there four+ concerts on the same weekend as Hamlet?
- Upcoming extra credit opportunities:
 - Hamlet, Friday (7:30pm), Saturday (7:30pm), Sunday (2pm).
 - Swimming, Saturday, 10am.
 - Chamber Ensemble, Saturday, 7:30 pm.
 - Typhoon Halyan Relief benefit show, Sunday, November 24th from 7-9pm in Harris. (If the entry fee is a burden, let me know and I'll give you the money.)
 - "Data Sovereignty: The Challenge of Geolocating Data in the Cloud", November 25, 4:15 JRC 101
 - Showing of "Gold Fever" by Andrew Shurburne '01 or so, 7:00 p.m. Monday, November 25, ARH 302
 - Tuesday, November 26, 4:15 p.m., JRC 209 a gaming event with the game [d0x3d!] Popcorn will be served.

Questions on the Project

- Is it all Alex's fault? Yes.
- Can I put everything into one procedure? I expect that such procedures will be hard to read unless you segment it into smaller procedures, but those can be internal.

The problem of searching

Context: Collection of structured values

```
(define people
  (vector
    ("Anderson" "Aan" 2017 4114 "x4410")
    ("Brown" "Bruin" 2016 8123 "x9000")
    ("Doe" "J" 2018 9999 "none")
    ("Mitchell" "Alexander" 2000 1234 "X1111")
    ("Smith" "Kieran" 2015 4112 "x9231")
    ("Tailor" "Mic" 2017 1234 "x0001")
  ))
```

e Given that we've put things in a vector, and each thing is a list of the given form, how do we find something?

- Recursively go through until we find something matching.
- Change the vector to a list and use assoc if we're looking for last names.

Suppose we wanted to write a generalized procedure?

```
(define search-vector (lambda (vector key get-key)
```

```
  (search-vector people 4112 caddr)
```

```
  (search-vector people "Mitchell, Alexander" (lambda (entry) (string-append (car entry) ", " (cadr entry))))
```

```
  (search-vector people "Alex" (lambda (entry) (substring (cadr entry) 0 4)))
```

How many items do we typically have to look through to decide someone is not there?

Making searching more efficient using divide-and-conquer

- Can we do better?
- Repeatedly: Look in the middle and throw away half

A demo: Destructive binary search

- 8000 -> 4000 -> 2000 -> 1000 -> 500 -> 250 -> 125 -> 62 -> 31 -> 15 -> 7 -> 3 -> 1
- $\log_2(n)$

Considering the parameters to binary search

Lab

- Do the lab
 - Be prepared to reflect.
-

Samuel A. Rebelsky, rebelsky@grinnell.edu

Copyright (c) 2007-2013 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.