CSE332: Data Abstractions

Section 2

Hyeln Kim

Winter 2013

Section Agenda

- Asymptotic Analysis
  - Example problem & HW1 Q/A
- Heaps
  - Property of Heap & Example problem
  - Build Heap
- Project 2
  - Introduction
  - Working in a team
  - Testing strategies

Asymptotic Analysis

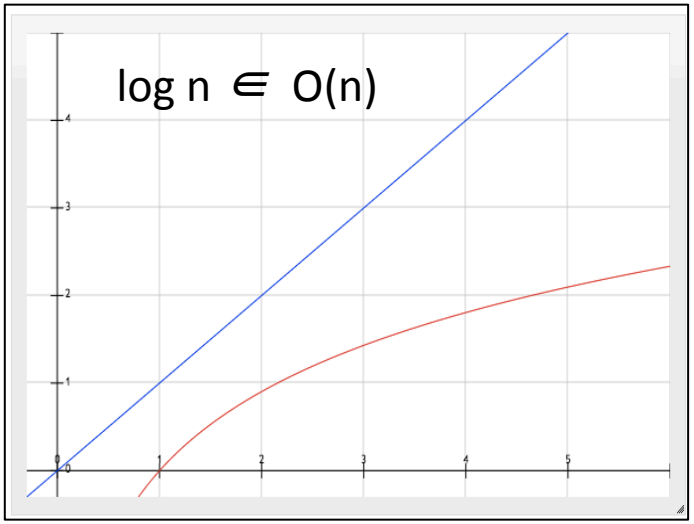
Asymptotic Analysis

- **Describe Limiting behavior of F(n)**

  - Characterize growth rate of F(n)
  - Use O(g(n)), Ω(g(n)), Θ(g(n)) for **set** of functions with asymptotic behavior ≤, ≥, ≤ & ≥ to g(n)

- **Upper Bound: O(n)**

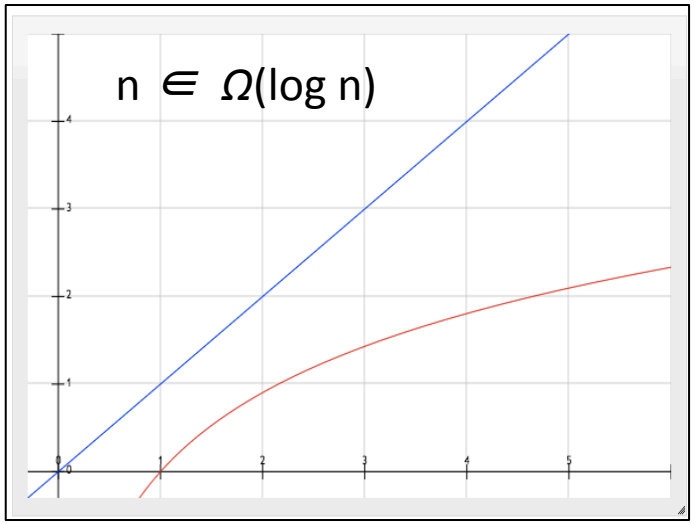
  f(n) ∈ O(g(n)) if and only if there exist positive constants c and n_0 such that
  f(n) ≤ c*g(n) for all n_0 ≤ n



Asymptotic Analysis

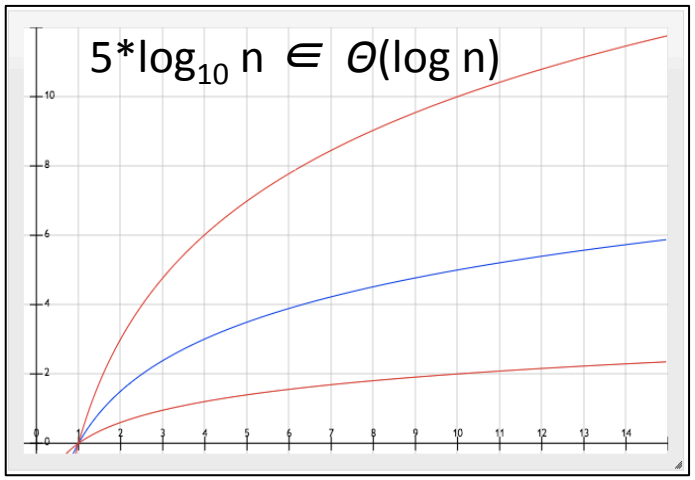
- **Lower Bound: Ω(n)**

  f(n) ∈ Ω(g(n)) if and only if there exist positive constants c and n_0 such that $c*g(n) \leq f(n)$ for all $n_0 \leq n$



- **Tight Bound: Θ(n)**

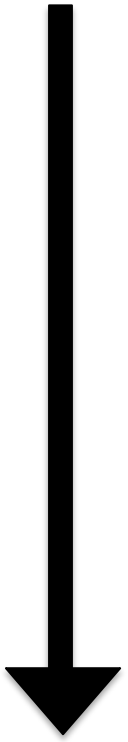
  f(n) ∈ Θ(g(n)) if and only if
  f(n) ∈ Ω(g(n)) and
  f(n) ∈ O(g(n))



Asymptotic Analysis

- **Ordering Growth rates (k = constant)**
  - Ignore Low-Order terms & Coefficients

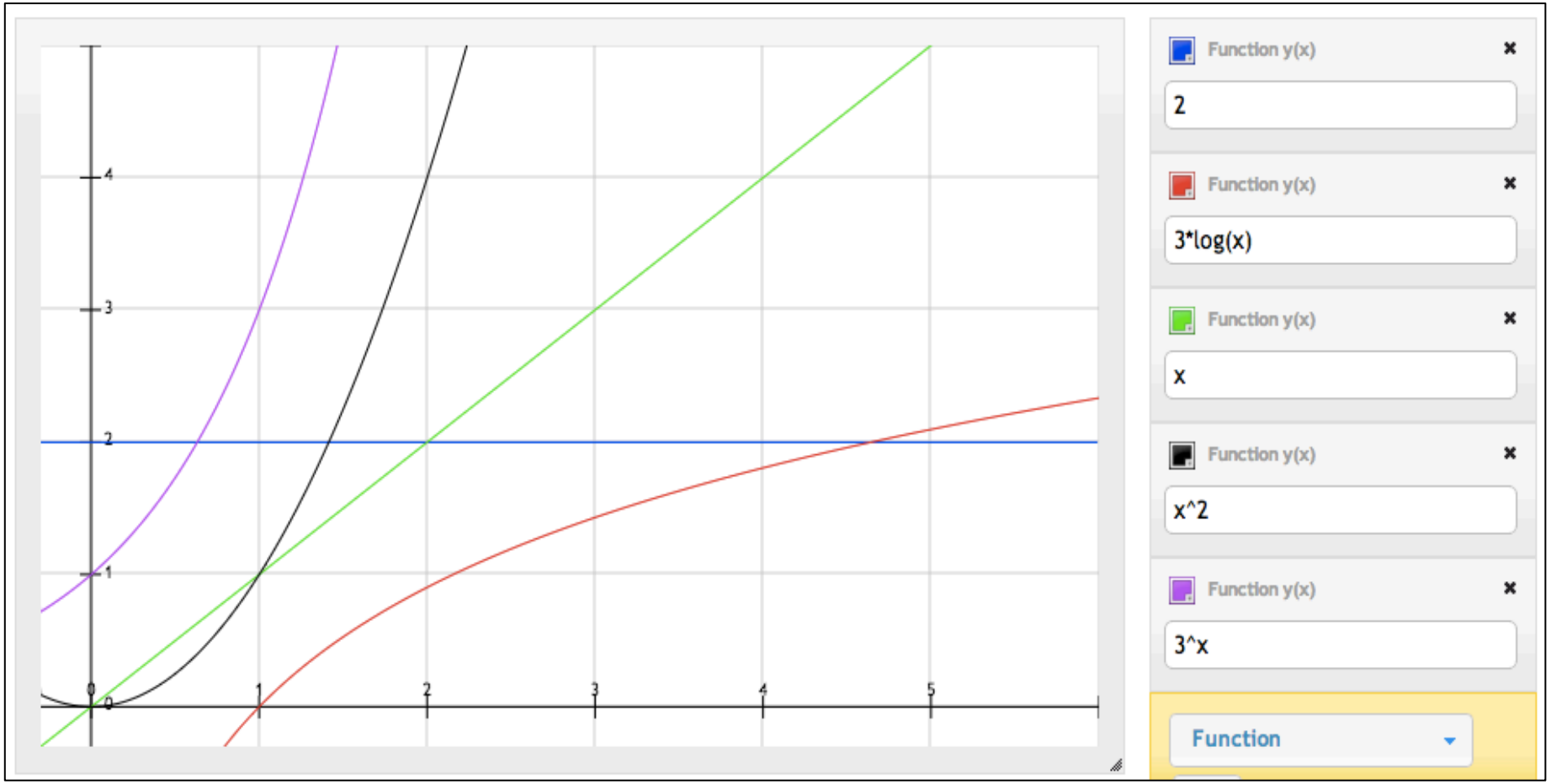
O(k)	constant
O(log n)	logarithmic
O(n)	linear
$O(n^k)$	polynomial
$O(k^n)$	exponential (k > 1)



Increasing
Growth rate

Asymptotic Analysis

- **Ordering Growth rates**



Asymptotic Analysis

- **Ordering Growth rates (k, b = constant)**
  - $\log^k n \in O(n^b)$ if $1 < k$ & $0 < b$
  - $n^k \in O(b^n)$ if $0 < k$ & $1 < b$

- **Ordering Example**

$2n^{100} + 10n$	n^{100}	4
$2^{n/100} + 2^{n/270}$	$2^{n/100}$	5
$1000n + \log^8 n$	n	3
$23785n^{1/2}$	$n^{1/2}$	2
$1000 \log^{10} n + 1^{n/300}$	$\log^{10} n$	1

Asymptotic Analysis

- **Proof Example: $f(n) \in O(g(n))$**

  - Prove or disprove $n \log n \in O(3n)$

  $n \log n \in O(3n)$, then by definition of Big-O
  $n \log n \leq c*(3n)$, for $0 < c \;\&\&\; 0 < n_0 \leq n$
  $(1/3) \log n \leq c$

  but as $n \to \infty$, $\log n \to \infty$
  Finite constant c always greater than log n
  cannot exist, no matter what n_0 we choose

  $n \log n \notin O(3n)$

Homework 1 Q&A

- **Solutions are NOT posted**

  - Anything you want to go over?

Heaps

Project 2

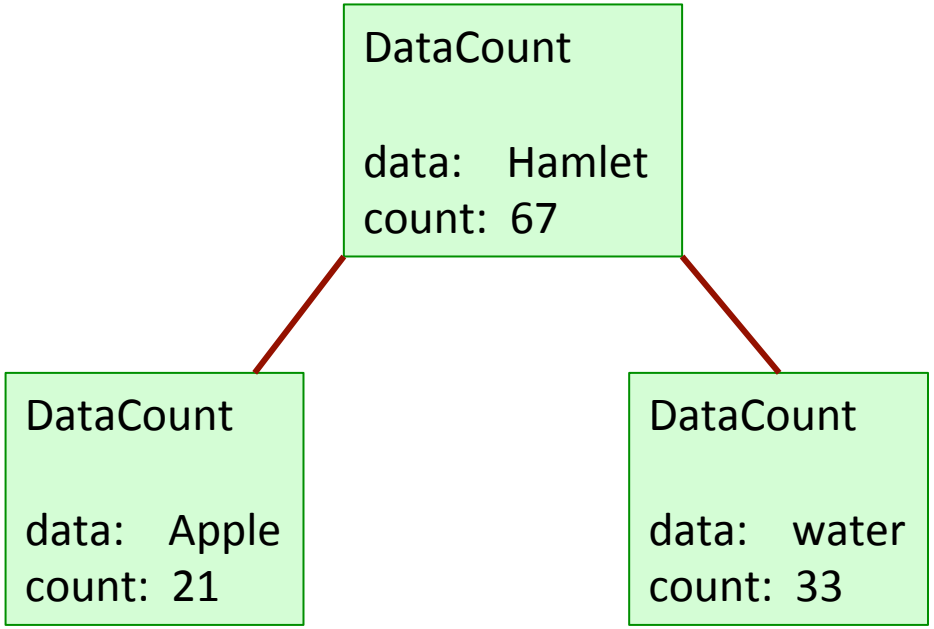
shake-n-bacon

Project 2

- **Word Frequency Analysis**

- **Phase A:** Implement 3 ADTs
  - Due about 2 weeks from now (Wednesday October 23rd)
  - Word frequency analysis using different DataCounters

- **AVLTree**
- **MoveToFrontList**
- **FourHeap**
- **Heap Sort**



Project 2 – Find Partner

- **Form a 2 person team**
- **Not required, but greatly encouraged**
  - Learn from each other
  - Check each other's style (large fraction of grade!)
  - Collaborate working experience
    (You can put it on your resume!)
- **Many had hard time meeting deadline**
  - Don't be spoiled by project 1!

Project 2 – Find Partner

- **Form a 2 person team**

  - Use discussion board to find partner

    https://catalyst.uw.edu/gopost/area/swansond/125971

  - Complete catalyst survey to form team by next Friday January 24th (Only one survey for a team!)

    https://catalyst.uw.edu/webq/survey/kainby87/223303

  - Anyone wants a partner but don't have one yet?

Project 2 – Sharing Code

- **Version Control System**

  - Git, Mercurial, SVN, CVS, Perforce...

  - Lots of choices (Use whatever you want):
    http://en.wikipedia.org/wiki/Comparison_of_revision_control_software

- **Some posts to help choosing one**

  - Choosing a Version Control: Beginners Tour
    http://www.codeproject.com/Articles/431125/Choosing-a-Version-Control-System-A-Beginners-Tour

  - Review of 7 Version Control Systems
    http://www.smashingmagazine.com/2008/09/18/the-top-7-open-source-version-control-systems/

  - What is your favorite version control system?
    http://programmers.stackexchange.com/questions/940/what-are-your-favorite-version-control-systems

Project 2 – Sharing Code

- **Repository:** Where you store your code

  - Your machine
  - Shared directory in attu (you need to request one)
  - Web based repository (should be private!)
    Bitbucket: https://bitbucket.org/
    GitHub: https://github.com/

Project 2 – Sharing Code

- **Useful tutorials**

  - Bitbucket 101
    https://confluence.atlassian.com/display/BITBUCKET/Bitbucket+101

  - Git
    https://www.atlassian.com/git?utm_source=cac-bitbucket-1&utm_medium=banner&utm_content=visual-git-guides&utm_campaign=git-tutorial

  - EGit (Git with Eclipse)
    http://www.vogella.com/tutorials/EclipseGit/article.html
    http://wiki.eclipse.org/EGit/User_Guide

- **Easy & Quick Code Sharing**

  - If you don't want to bother having version control
    http://www.smashingapps.com/2011/05/25/ten-best-collaborative-sites-for-quick-code-sharing.html

Bugs &Testing

Bugs &Testing

- **Why Testing?**

  Bugs can be costly

  - Cost points in homework
  - Can cost $$$ and even life (Therac-25)

  Interesting Bug References

  - List of bugs http://en.wikipedia.org/wiki/List_of_software_bugs
  - History's worst http://www.wired.com/software/coolapps/news/2005/11/69355?currentPage=all
  - Bugs of the month http://www.gimpel.com/html/bugs.htm

Bugs & Testing

- **Reverse.java**
  does not test your stack!!

  - Stack can still have lots of bugs
    when working perfectly with Reverse.java

  - Some extreme case in past quarter:
    it only worked with Reverse.java
    (Not a good stack!)

Bugs & Testing

- **Tips for Testing**

  - Make sure program meets the spec

  - Test if each method works independently

  - Test if methods work together

  - Test for edge cases

Bugs & Testing

- **Make sure program meets the spec**

  - What is wrong with this implementation?

```
public class ListStack implements DStack {
        private LinkedList<Double> myStack;

        public ListStack() {
                myStack = new LinkedList<Double>();
        }
        public void push(double d) {
                myStack.add(d);
        }
        …
}
```

Bugs & Testing

- **Test for edge cases**

  - Empty stack

  - Push after resizing

  - Anything else?

Bugs & Testing

- **Testing tools:** JUnit Testing

  - Not required for Project 1

  - Required for Project 2

  - Covered in section later