Comp 250 Review

Game Plan

First Hour-ish: Review Class Material

Until 8:30: Go over old exam questions

Ask Questions!!! Homework questions?

Java Primitives vs. Objects

-Objects box up data!

-Objects let you couple data and functionality

-equality testing: ‘==’ vs. ‘.equals()’

Java Basic Data Types

-String: not a primitive! String is an object

-Array: contiguous space in memory, constant access time

-LinkedList - made of Nodes that maintain references to other Nodes (example)

Abstract Data Types

-Data types often represent real world concepts

1. Stack … push(), pop(), peek()
2. Queue … enqueue(), dequeue()

-Most data types are implemented using a linked list or an array

- how to implement Queue with Stacks (example)

Interfaces and Polymorphism

-We defined a Queue with two stacks. It implemented the Queue interface! We could provide several Queue implementations, each with its own pros and cons (we'll do it in a bit)

-Person/Student/Singer (example)

Intro to Algorithms

-iterative vs. recursive

-binarySearch with exceptions (example)

-iterativeFibonnaci (example)

More on Algorithms

-Profs love arrays, lists, and trees

-know how to traverse a list

-know how to traverse a tree

-remember that only Arrays have constant access!

Proof by Induction

-Show a simple base case

-Assume it works for all inputs up to size n

-Prove that it must work for an input of size n+1

Induction

Mathematical Induciton:

- Sum from 1 to n of
  - 1/(i*(i+1)) = n/(n+1)

Strong Induction:

- Prove MergeSort (we'll do the recurrence, too)

Big Oh

-a way of reasoning about the runtime of algorithms

-Best/Average/Worst case

-standard sort running times

-limit proof that f(n) is O(g(n))

1. Selection Sort/ Insertion Sort - O(n^2)
2. Merge Sort - O(n log(n))

Master Method

- use for divide and conquer algorithms

$T(n) = a * T(n/b) + n^c$

1. if $\log_b a < c$, $T(n) = \Theta(n^c)$.
2. if $\log_b a = c$, $T(n) = \Theta((n^c) * (\log n))$
3. if $\log_b a > c$, $T(n) = \Theta(n^{(\log b\ a)})$

Master Method Examples

$T(n) = a * T(n/b) + n^c$

1. MergeSort

- $T(n) = 2*T(n/2) + O(n)$ … $a = 2, b = 2, c = 1.$
- $\log_b a = \log_2 2 = 1.$
- Case 2: $T(n) = O(n \log n)$

2. $T(n) = 4T(n/2) + O(n^3)$ … $a = 4, b = 2, c = 1.$
   - $\log_b a = \log_2 4 < 3$
   - Case 1: $T(n) = O(n^3)$

Some More Code Examples

-Lets implement a stack with array and linked list

-Suggestions on something to write?

Any questions?

- Come after if you want me to help you with a problem

-github.com/itscharlieb/comp-250-review

-Facebook me with any other questions, I'll happily solve them and put them on my Github.

Testing your Code

- the point of testing is to break your code!
- three parts of writing test code:

1. unit under test
2. input data
3. test answer

- Java @Test and assert