CSE 373
Winter 2012

Looking Forward,
Looking Back

Today's Outline

- **Announcements**
  - Final Exam – next Tues March 13th , 2:30-4:20pm
  - Office Hours Next week:
    - Mon 3/12 12-1:30pm, Ruth, CSE 360
    - Tues 3/13 12-2pm, Ruth, CSE 360
- **Review/Overview**
- **Course Evaluations**

Final Exam

- **Final Exam,** Tuesday, March 13th , 2012
- 2:30 - 4:20pm in our regular lecture room
- Exam policies
  - Closed book, closed notes. No Calculators allowed.
  - The exam begins promptly at 2:30pm and ends at 4:20pm.
- The Final exam is cumulative, although more weight will be given to topics covered since the second midterm.

Overview and Goals

(From first day handout)

Achieve an understanding of fundamental data structures and algorithms and the tradeoffs between different implementations of these abstractions. Theoretical analysis, implementation, and application. Lists, stacks, queues, heaps, dictionaries, maps, hashing, trees and balanced trees, sets, and graphs. Searching and sorting algorithms.

Topics

- Stacks and Queues, array and list implementations.
- Asymptotic analysis, Big-O. Worst case, upper bound, lower bound, analyzing loops, recurrences, amortized complexity.
- Trees – definitions
- Dictionary ADT
- Binary search trees – Inorder, preorder, postorder traversals, insert, delete, find.
- AVL trees - Single and double rotations, insert, find.

Topics (cont.)

- Binary Heaps - Findmin, Deletemin, Insert. Additional operations of increase, decrease, buildheap.
- D-heaps - Findmin, Deletemin, Insert. Additional operations of increase, decrease, buildheap.
- Disjoint Union/Find. Up-trees. Weighted union (union by size) and path compression.
- Hashing. Properties of good hash functions. Selecting hash table size. Separate chaining and open addressing. Linear Probing, Quadratic Probing, & Double Hashing to resolve collisions. Rehashing.
- The memory hierarchy. Temporal and spatial locality. Data structure choice and the memory hierarchy.

Topics (cont.)

- Graphs. Directed and undirected. Adjacency list and adjacency matrix representations.
  - Topological sorting.
  - Graph searching. Depth-first, breadth-first search.
  - Shortest paths. Dijkstra's algorithm. Greedy Algorithms.
- Sorting. Insertion sort, Selection sort, Heap sort, Merge sort, Quicksort. Lower bound on comparison sorting. In-place sorting. Stable sorting. Bucket sort, Radix sort.
- B-trees. Motivation, choice of M and L, Insert & delete.

Concepts

- ADT – what it is, why we have them, how to compare implementations
- Comparisons – Running time, Space, Big-O, Data Locality
- Tradeoffs – Pointers vs. Arrays, Space vs. Time
- Algorithm Design – Iteration, Recursion, Greedy Algorithms, Divide and Conquer

Skills

- Use Big-O to help you select the best data structure
- Time your code
- Java programming

Programming Projects

- Implement Stack using Linked list & array
- Implement Heaps (2,3, other heap)
- Solve Problems using data structures:
  - Maze Generation (Disjoint Sets)
  - Dijkstra's for shortest path (Graphs + Dictionary + Heap? + ???), using Java Collections

More Computer Science Courses!!

- CSE 374 Intermediate Programming (12wi)
  Concepts & Tools
- CSE 410 Computer Systems (12sp)
  (Operating Systems & Architecture)
- CSE 413 Programming Languages
  and their Implementation
- CSE 415 Artificial Intelligence (autumn)
- CSE 417 Algorithms and Complexity (12wi)

Thanks!