3343 Analysis of Algorithms – Spring 09

3/25/09

Schedule

(subject to change)

Date	Material
Tu 1/13	**Analyzing algorithms (Ch. 2.2)** Best case and worst case runtimes; insertion sort, incremental algorithm
Th 1/15	**Asymptotic notation (Ch. 3, Ch. A)** O, Ω, Θ, o, limit-theorem; runtime for code-snippets
Tu 1/20	**Asymptotic notation (Ch. 3, Ch. A)** O, Ω, Θ, o, limit-theorem; runtime for code-snippets *Homework 1 assigned*
Th 1/22	**Heapsort (Ch. 6)** Abstract data types (ADT), priority queue, heap, heapsort, linear-time buildheap
Tu 1/27	**Recursion trees and induction (+)** Recursive algorithms. Guess solution of recurrence using recursion trees and prove the correctness of the solution using induction. *Homework 1 due; homework 2 assigned*
Th 1/29	**Divide-and-conquer (Ch. 2.3) and recurrences (Ch. 4.1, 4.2)** Divide-and-conquer, merge sort, binary search; Runtime recurrences. Big-Oh induction (substitution method)
Tu 2/3	**Master theorem (Ch. 4.3)** Use of master theorem to solve recurrences. *Homework 2 due; homework 3 assigned*
Th 2/5	**More divide-and-conquer (Ch. 31.6 pages 879–880; 28.2)** Repeated squaring for exponentiation, Strassen's matrix multiplication. *Programming project 1 assigned*
Tu 2/10	**Probability, random variables and expected values (Ch. C.2, C.3)** Probability, random variables, expected values. *Homework 3 due; homework 4 assigned*
Th 2/12	**Randomized algorithms (Ch. 5.1–5.3)** Hiring problem; Expected runtime analysis.
Tu 2/17	**Quicksort (Ch. 7.1–7.4)** Quicksort, best-case and worst-case runtimes, randomized quicksort. *Homework 4 due*
Th 2/19	**Test 1** Material until 2/10 (inclusive)
Tu 2/24	**Sorting (Ch. 8.1, 8.2, 8.3)** Decision trees, lower $\Omega(n \log n)$ bound for comparison sorts, counting sort, radix sort *Homework 5 assigned*
Th 2/26	**Sorting (Ch. 8.1, 8.2, 8.3)** Decision trees, lower $\Omega(n \log n)$ bound for comparison sorts, counting sort, radix sort
Tu 3/3	**Order statistics (Ch. 9)** Order statistics (find i-th smallest element); Randomized selection, deterministic selection in linear time *Homework 5 due*
Th 3/5	**Red-black trees (Ch. 13.1, 13.2, 13.3)** Red-black tree property, rotations, insertion; abstract data types, ADT dictionary

Date	Material
Tu 3/10	**SPRING BREAK**
Th 3/12	**SPRING BREAK**
Tu 3/17	**Dynamic programming (Ch. 15.4, +)** Fibonacci, binomial coefficient, LCS: fill table, then construct solution from the table. *Programming project 1 due* *Homework 6 assigned*
Th 3/19	**Dynamic programming (Ch. 15.3, 15.4. 16.2, +)** 0-1 Knapsack; general outline of dynamic programming: Optimal substructure (recurrence), overlapping subproblems, fill table bottom-up or by memoization.
Mo 3/23	**Drop deadline to drop with a 'W'**
Tu 3/24	**Greedy algorithms (Ch. 16.2, problem 16-1 on page 402)** Greedy algorithms (greedy-choice property, optimal substructure). Making change, fractional knapsack. *Homework 6 due; homework 7 assigned*
Th 3/26	**Elementary Graph Algorithms (Ch. 22.1–22.2)** Representations of graphs, breadth-first search (BFS) *Programming project 2 assigned*
Tu 3/31	**Elementary Graph Algorithms (Ch. 22.3–22.4)** Depth-first search (DFS), topological sort *Homework 7 due*
Th 4/2	**Test 2** Material from 2/12 until 3/24 (inclusive)
Tu 4/7	**Minimum Spanning Trees (Ch. 23)** Prim (grows single tree), Kruskal (grows forest; uses union/find data structure) *Homework 8 assigned*
Th 4/9	**Single-source shortest paths (Ch. 24 without 24.4)** Optimal substructure, triangle inequality, relaxation step; Dijkstra (only for non-negative edge weights), predecessor tree (shortest path tree); Bellman-Ford, detection of negative-weight cycles; Shortest paths in a DAG
Tu 4/14	**All-Pairs Shortest Paths (Ch. 25.2)** Dynamic programming: Floyd-Warshall *Homework 8 due; homework 9 assigned*
Th 4/16	**P and NP (Ch. 34)** Decision problems, definition of classes P and NP, polynomial-time reductions
Tu 4/21	**P and NP (Ch. 34)** NP-hardness, NP-completeness; Show that problems are NP-complete by reducing from other problems *Homework 9 due; homework 10 assigned*
Th 4/23	**P and NP (Ch. 34)** TSP, Clique, Independent Set, Vertex Cover, Hamilton Path, Hamilton Circuit *Programming project 2 due*
Tu 4/28	**Review for Final Exam** Review for final exam *Homework 10 due*

Chapter numbers refer to the CLRS book. "+" indicates additional material.

The comprehensive final exam will be on Monday May 4th, 10:30am – 1pm.