Massachusetts Institute of Technology
6.046J/18.410J: Introduction to Algorithms
Professors Piotr Indyk and Madhu Sudan

Handout 4
February 6, 2001

Course Outline

February

Tue	6	**Lecture 1** Administrivia. Introduction: analysis of algorithms, insertion sort, mergesort, O-notation, recurrences. *Reading*: Chapters 1–2.	Sign Up! Survey Out.
Thu	8	**Lecture 2** Asymptotic notation. Recurrences: substitution, iteration, master method. *Reading*: Chapters 3–4, excluding Section 4.4.	Survey Due. PS 1 out.
Fri	9	**Recitation 1** Correctness of algorithms. *Reading*: Kingston chapter.	
Tue	13	**Lecture 3** Divide and conquer: merge sort, Strassen's algorithm, polynomial multiplication, exponentiation. *Reading*: Chapter 5; Sections 31.2 and 32.1–2.	
Thu	15	**Lecture 4** Randomized algorithms: quicksort. *Reading*: Chapter 8.	PS 1 due. PS 2 out.
Fri	16	**Recitation 2** Divide and conquer + Randomization examples.	
Tue	20	**Virtual Monday - no lecture!**	
Thu	22	**Lecture 5** Heaps, Heapsort, Priority Queues *Reading*: Chapter 7.	
Fri	23	**Recitation 3** Randomization examples + Heap usage *Reading*: Chapters 7 & 8.	
Tue	27	**Lecture 6** Sorting: lower bounds in the comparison model. Linear time sorting: counting sort, radix sort. *Reading*: Sections 9.1-3	PS 2 due. PS 3 out.

March

Thu	1	**Lecture 7** Sorting, median, order statistics *Reading*: Chapter 10	
Fri	2	**Recitation 4** Review of sorting lower bounds. *Reading*: Sections 9.1-3	

Tue	6	**Lecture 8** Data structures: Hashing, dynamic dictionaries. *Reading*: Chapter 12.	PS 3 due. PS 4 out.
Thu	8	**Lecture 9** Data structures: binary search trees, tree walks, relation to quicksort. *Reading*: Sections 13.1–3	
Fri	9	**Recitation 5** Quiz Review.	
Tue	13	**Lecture 10** Data structures: 2-3 trees, insertion, deletion. *Reading*: Chapter 19.	PS 4 due.
Thu	15	QUIZ 1: in-class test.	
Fri	16	**No recitation.**	
Tue	20	**Lecture 11** Augmenting data structures: dynamic order statistics, interval trees. *Reading*: Chapter 15.	
Thu	22	**Lecture 12** Amortized algorithms: table doubling, potential method, self-organizing lists, competitive analysis. *Reading*: Chapter 18.	
Fri	23	**Recitation 6** Amortized algorithms and dynamic programming.	
Tue	27	SPRING BREAK	
Thu	29	SPRING BREAK	
Fri	30	SPRING BREAK	

April

Tue	3	**Lecture 13** Dynamic programming: optimal binary search trees, longest common subsequence. *Reading*: Chapter 16.	
Thu	5	**Lecture 14** Greedy algorithms: minimum-spanning trees, Prim's and Kruskal's algorithms. *Reading*: Chapters 22 and 24.	PS 5 out.
Fri	6	**Recitation 7** Examples of greedy algorithms and dynamic programming. *Reading*: Chapters 16, 22, and 24.	

Tue	10	**Lecture 15** Graph algorithms: depth-first search, topological sort, strongly-connected components. *Reading*: Sections 23.3–5.	
Thu	12	**Lecture 16** Graph algorithms: single-source shortest paths, Dijkstra's algorithm, Bellman-Ford algorithm. *Reading*: Sections 25.1–3.	PS 5 due.
Fri	13	**Recitation 8** Graph algorithms. *Reading*: Chapters 23–26	
Tue	17	**Patriots Day Holiday**	
Thu	19	**Lecture 17** Network flow: max-flow min-cut, Ford-Fulkerson algorithm, augmenting paths. *Reading*: Sections 27.1–3.	
Fri	20	**Recitation 9** Bipartite matching *Reading*: Section 27.3.	
Mon	23	**Quiz 2 (Take Home) Out**	Quiz 2 Out
Tue	24	No lecture!	
Thu	26	**Lecture 18** Randomized Algorithms.	Quiz 2 Due
Fri	27	**No recitation.**	
May			
Tue	1	**Lecture 19** Computational Geometry I.	
Thu	3	**Lecture 20** Computational Geometry II.	PS 6 out.
Fri	4	**Recitation 10** Computational geometry examples.	
Tue	8	**Lecture 21** Complexity: P vs. NP, efficient verification. *Reading*: Sections 36.1–2.	
Thu	10	**Lecture 22** NP-completeness. *Reading*: Sections 36.3–5.	PS 6 due.
Fri	11	**Recitation 11** NP-completeness. *Reading*: Chapter 36.	

Tue	15	**Lecture 23** Approximation algorithms: vertex cover, traveling salesman problem, partition. *Reading*: Chapter 37.
Thu	17	**Lecture 24** Advanced topics.
??	??	FINAL EXAM: Week of Final exam.