Course summary: Data Structures and Algorithms (FS 2013)

Below you find the exam relevant topics discussed in the Data Structures and Algorithms lecture held in spring 2013, along with suggested readings in English. Please note that not every topic is discussed at the same level as in the german book accompanying the lecture.

Literature


Topics relevant for the exam


[Cormen, Ch. 1 – 3]


[Cormen, Ch. 4.1]


[Cormen, Ch. 2, Problem 2.2-2, 2.3-5, 2-2]


[Cormen, Ch. 7]


[Cormen, Ch. 6, 8.1]

Searching II: Data structures for dictionaries. Binary search trees. AVL trees. [Cormen, Ch. 12.1 – 12.3 & Goodrich, Ch. 3.2]

Searching II (continuation): AVL Trees and amortized analysis. AVL-Trees: insertion, deletion. Amortized analysis of the insert operation. [Goodrich, Ch. 3.2]


Splay trees. Optimal search trees. Self-organizing search trees: move-to-root rule, splay trees. Optimal search trees, construction with dynamic programming. [Cormen, Ch. 15.5 & Goodrich, Ch. 3.4]


Dynamic programming II. Knapsack problem, FPTAS. [Vazirani, Ch. 8.1 – 8.2]

Graph algorithms I. Data structures for graphs. Reflexive and transitive closure. Graph traversal: BFS and DFS. Connected components. Topological sorting. [Cormen, Ch. 22.1 – 22.5]

Graph algorithms II. Minimal spanning trees: introduction, greedy algorithms, Kruskal’s algorithm with union find structure. [Cormen, Ch. 23]

Graph algorithms III. Minimal spanning trees with Prim/Dijkstra. Fibonacci Heaps. [Cormen, Ch. 19, 23]

Graph algorithms IV. Shortest paths from a single source. Algorithms of Dijkstra, Bellman-Ford. [Cormen, Ch. 24.1 – 24.3]


Graph algorithms VI: Flows in networks II. Shortest augmenting path algorithms, $O(m^2n)$ algorithm by Edmonds-Karp, $O(mn^2)$ algorithm by Dinic. Matchings in bipartite graphs. Hall’s theorem. [Ahuja, Ch. 7.5 & Cormen, Ch. 26.2 – 26.3, Problem 26.3-4]

Geometric algorithms I. Convex hull of points in the plane: Jarvis, Graham, linear scan. Intersection of orthogonal line segments: Check for intersection, report all intersection points, count all intersection points. [Cormen, Ch. 33.1 – 33.3 & deBerg, Ch. 1.1 – 1.2]

Geometric algorithms II. Intersection of arbitrarily oriented line segments: checking and reporting. Intersection of axis-parallel rectangles, 2-dimensional range trees. [deBerg, Ch. 5.3]
17.5.2013  *Geometric algorithms III.* Intersection of axis-parallel rectangles using segment trees, tile trees, and interval trees.  
[deBerg, Ch. 10.1 – 10.3]

[Cormen, Ch. 33.4]

[Cormen, Ch. 18]