Course summary: Data Structures and Algorithms (FS 2015)

Below you find the exam relevant topics discussed in the Data Structures and Algorithms lecture held in spring 2015, 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

19.2.2015 Introduction and Algorithm Design. Role of algorithms in computer science. An example for the design of algorithms: fast multiplication of integers, \textsuperscript{Karatsuba/Ofman 1962} Finding a star by asking questions. Uniform cost model. Growth of functions: big $O$ notation, Omega, Theta. [Cormen, Ch. 1–3]

20.2.2015 Algorithm Design by Induction. Maximum subarray problem: naïve algorithm, precomputation of prefix sums, divide and conquer, linear scan. [Cormen, Ch. 4.1]

26.2.2015 Searching I: Searching. Computing the Median. Linear search, binary search, \textsuperscript{interpolation search} upper and lower bounds. Randomized median computation. Blum’s algorithm for computing the median. [Cormen, Problem 2.1-3, 2.2-3, 2.3-5, Ch. 9]

27.2.2015 Searching II: Hashing. Hash tables, hash functions, universal hashing. Collision resolution by chaining, open hashing, probing. [Cormen, Ch. 11]

5.3.2015 Sorting I: Elementary Sorting Methods. Mergesort. Heapsort. Bubblesort, \textsuperscript{odd-even transposition sort} selection sort, insertion sort. Recursive, iterative and natural mergesort. Heapsort, implicit representation of the heap, creating a heap in linear time. [Cormen, Ch. 2, Problem 2.2-2, 2-2, Ch. 6]

6.3.2015 Sorting II: Quicksort. Quicksort, key comparisons: best case, worst case. Movements: worst case. Additionally required memory. [Cormen, Ch. 7]
12.3.2015 Sorting III: Lower bounds. Radix sort. Dictionaries I: Basic concepts. Lower bound for decision-based sorting: decision tree, average height of a leaf in a binary tree. Radix Exchange Sort. Dictionaries: Array, linear list, skip list. [Cormen, Ch. 6, 8.1, 8.3]

13.3.2015 Dictionaries II: Binary Search Trees, AVL Trees. Binary search tree: traversal, searching, insertion, deletion. AVL trees: height of the tree, insertion. [Cormen, Ch. 12.1–12.3; Goodrich, Ch. 3.2]

19.3.2015 Dictionaries III: AVL trees, Amortized Analysis. AVL trees: deletion. Amortized analysis of the insert operation. [Goodrich, Ch. 3.2]

20.3.2015 Self-Organizing Linear Lists and Search Trees. Self-organizing linear lists, move-to-front rule (with analysis). Self-organizing search trees: Splay trees. [Cormen, Ch. 17, esp. Problem 17-5; Goodrich, Ch. 3.4]


27.3.2015 Dynamic Programming II. Subset Sum. Knapsack problem, FPTAS. [Cormen, Ch. 35.5; Vazirani, Ch. 8.1–8.2]

2.4.2015 Optimal Search Trees. Huffman Coding. Definition of optimal search trees, construction with dynamic programming. [Cormen, Ch. 15.5, 16.3]

16.4.2015 Backtracking, Branch and Bound. Backtracking, Examples: 4 Queens, SAT. Branch and Bound. Examples: MAX SAT, Knapsack, TSP. [Skiena, Ch. 7.1; Wolsey, Ch. 7.1–7.2]

17.4.2015 Graph Algorithms I. Reflexive and transitive closure. Graph traversal: BFS, DFS. Connected components. Topological sorting. [Cormen, Ch. 22]

23.4.2015 Graph Algorithms II: Minimum Spanning Trees. Minimum spanning trees: introduction, greedy algorithms, Kruskal’s algorithm with union find structure. [Cormen, Ch. 23]

24.4.2015 Graph Algorithms III: Minimum Spanning Trees. Minimum spanning trees with Prim/Dijkstra. [Cormen, Ch. 23]

30.4.2015 Graph Algorithms IV: Minimum Spanning Trees. Fibonacci Heaps and Minimum Spanning Trees. [Cormen, Ch. 19, 23]


9.5.2015 Graph Algorithms VII: Flows in Networks II. Shortest augmenting path algorithms, $O(m^2n)$ algorithm by Edmonds-Karp, $O(nm^2)$ algorithm by Dinic. Matchings in bipartite graphs. Hall’s theorem. [Ahuja, Ch. 7.5; Cormen, Ch. 26.2 – 26.3, Problem 26.3-4]
21.5.2015 Geometric Algorithms I. Convex hull of points in the plane: Jarvis, Graham, linear scan. Intersection of orthogonal line segments: check for intersection. [Cormen, Ch. 33.1 – 33.3; deBerg, Ch. 1.1 – 1.2]

22.5.2015 Geometric Algorithms II. Intersection of orthogonal line segments: report all intersection points, count all intersection points. Intersection of arbitrarily oriented line segments: checking. [Cormen, Ch. 33.1 – 33.2; deBerg, Ch. 5.3]

28.5.2015 Geometric Algorithms III. Intersection of axis-parallel rectangles. 2-dimensional range trees. Intersection of axis-parallel rectangles using segment trees, tile trees/interval trees, and priority trees. [deBerg, Ch. 5.3, Ch. 10.1 – 10.3]

29.5.2015 External Memory Structures. External storage model, idea of locality. B-Trees. [Cormen, Ch. 18]