

# Algorithm Design and Analysis

## Exam Topics

1. Insertion Sort algorithm and its analysis.  
C2.1(18-28) + Ex. C2.1-1(22) + file attached. *Explanation:* Section 2.1 from Cormen's book on my site, pages 18-28; exercise on page 22. Section numbers in the file coincide with section numbers in the hardcopy from MIF library.
2. Merge-Sort algorithm and upper bound of its complexity.  
C2.3(29-35) + Ex. C2.3-1(35).
3. Lower bound for SORT problem complexity.  
C8.1(145-147) + file attached.
4. Growth of functions and asymptotical notation.  
C3.1(40-47) + Problem C3-2a,b,d,e(56) + file attached.
5. Dynamic programming. Knapsack problem.  
Chapter 42 of Sedgewick (you have a copy)595-598
6. Dynamic programming. Matrix chain product.  
Chapter 42 of Sedgewick (you have a copy)599-604 + C15.2(284-289) + Ex.15.2-1(289).
7. Method "Divide-and-conquer". Master's theorem. Proof of someone from 3 cases.  
C4.3-4.4(67-78) + Problem C4.1a-f(78) + file attached
8. Strassen's algorithm for matrix multiplication.  
C28.2(642-647).
9. Backtracking. 8 queens problem.  
File attached.
10. Branch-and-bound method. Its application to TSP.  
Literature from internet + Ex.: find shortest Hamilton cycle in a given weighted graph.
11. Representation of graphs.  
C22.1(445-447) + Ex. 22.1-2(447)
12. Breadth-first search and depth-first search (ideas).  
C22.2(449-455) + C22.3(457-464) + files attached + Ex. C22.3-2(463) (to solve both for BFS and DFS).
13. Minimum spanning trees. Kruskal's algorithm.  
C23.2(476-486) + Ex.: find MST for a given weighted graph.
14. Minimum spanning trees. Prim's algorithm.  
C23.2(476-486) + Ex.: find MST for a given weighted graph.
15. Shortest-path problem. Floyd-Warshall algorithm.  
C25.2(535-539) + Ex. C25-2-1(539).
16. Relation between problems and languages (sketch). Class P.  
C34.1(845-850).
17. Polynomial time verification. Class NP.  
C34.2(851-854).
18. Polynomial reduction. NP-completeness.  
C34.3(856-859)