Algorithms and Data Structures DV2 Reading Instructions

Arne Andersson

Basics

- Do all excersices
- Finish the assignments
- Read the book chapters
- At the exam: relax and THINK

You must know from previous courses

- Sorting: chapters 6,7,8
- Searching: chapters 10, 11, 12, 13

Chapter 3

- You must know the three (five) basic notations of asymptotic growth
- You should have a strong intuitive feeling for the log function (base 2)
- You should be able to rank functions according to asymptotic growth (in particular, you need a good feeling for the differences between exponential functions, polynomial functions and polylogarithmic functions)

Chapter 4

- You must know the different methods to solve reucrrence equations (you don't need to be able to prove the master theorem)
- You must be able to set up a reucrrence equation for a computational problem

Chapter 15, Dynamic programing

- Know the basic ideas, be able to explain them
- Understand the typical examples: Fibonacci numbers, matrix multiplication
- Make it a part of your toolbox!

Chapter 16, Greedy algorithms

- Know the basics ideas of greedy algorithms
- Understand the Huffman coding algorithm, and why it is a greedy method
- Make it a part of your toolbox!

Chapter 21, Data structures for disjoint sets

- You should know the problem definition
- You should understand how dijoints sets are used later on in Kruskal's algorithm for minimum spanning trees

Chapters 22-24, 26

- Graph representation
- Directed and undirected graphs
- Depth-first search, bredth-first search
- Understand the algorithm for computing connected components
- Kruskal's algorithm for minimum spanning trees, including a simple algorithm for disjoint sets
- Dijkstraa's algorithm for single-source-shortest-paths
  - Remember the observation: if the distances are relatively small integers, we can run the algorithm in linear time, the time is O(|E| + longest distance).
    This is because we can use a simple bucketing structure as priority queue. The front of the queue will always be increasing, so we never have to go backwards in the bucketing structure.
- Edmunds-Karp algorithm for maximum flow
- Understand the idea of adding extra source and sink to a graph, in order to use maximum flow to solve other problems

Chapter 28, Matrix multiplication

- Understand the basic idea behind Strassen's algorithm: by reducing the required number of multiplications in each recursive step from 8 to 7, we get a better complexity.
- Know the recursion equation and its solution

Chapter 32 String Matching

- Rabin-Karp algorithm (including how to use modular arithmetric to get a "fingerprint" for large patterns)
- Understand the basic priciples on how to use a graph over the parttern (but no the details on how to construct the graph)

Chapter 33, Computational Geometry

- Understand the plane-sweep paradigm
- Finding convex hull:
  - Graham's scan
  - a general understanding on how to construct a convex hull by divide-and-conquer
  - Jarvi's march
- Finding the closest pair of points in a two-dimensional point set