1. Basic Data Structures
   • arrays, loops and invariants
   • lists, stacks, queues, sets and recursion
   • primitive operators
   • uses and algorithms

2. Complexity
   • space v. time
   • average case v. worst case
   • Big Oh notation
   • computation – exact and approximate

3. Trees – general, binary and quad trees
   • inductive definitions
   • primitive operators
   • uses and algorithms

4. Binary Search Trees
   • searching in general
   • definition of binary search trees
   • building, modifying and searching
   • tree rotations – why, what, how?

5. Heap trees and Priority queues
   • definition of heaps and priority queues
   • binary heap trees – insertions, deletions, building
   • bubbling up and bubbling down
   • Binominal trees and heaps

6. Sorting
   • general definitions and theoretical limits
   • $O(n^2)$ algorithms – bubble, selection, insertion
   • tree based algorithms – Treesort and Heapsort
   • divide and conquer algorithms – Quicksort and Mergesort
   • non-comparison algorithms – Radix sort
   • comparisons – average/worst speed, stability, only first $m << n$, …

7. Hash Tables
   • general definitions and implementations
   • load factors, efficiency, computational costs
   • dealing with collisions – buckets, direct chaining, open addressing
   • linear probing, secondary/double hashing
   • choosing good hash functions

8. Graphs
   • general definitions and implementations
   • planarity – definitions and theorems
   • traversals – depth first and breadth first
   • shortest paths – Dijkstra’s and Floyd’s algorithms
   • minimal spanning trees – Prim’s and Kruskal’s algorithm