1. Linked Lists, Object Oriented Approach
   i. List operations: insert nodes, delete nodes, traversal
   ii. List container, list iterators
   iii. Standard Template Library lists
2. The Stack and Queue Abstract Data Types (ADT)
   i. Definition as an ADT
   ii. Linked list and array implementations
   iii. Applications
3. The C++ Standard Template Library (STL)
   i. Lists, vectors, stacks, queues, dictionaries (maps), union-find
4. Measuring Algorithm Performance
   i. Definitions of Big-Oh, Big-Theta, Big-Omega
   ii. Basic operation, complexity function
   iii. Counting loop iterations and recursive calls
5. The Table ADT, Hash Tables
   i. Table ADT definition, applications
   ii. Definition of a hash table
   iii. Examples of Hash functions for integers and strings
   iv. Collision resolution schemes, implementation
6. Binary Trees
   i. Definition,
   ii. Node struct implementation
   iii. Tree operations, inserting and deleting nodes
   iv. Tree traversal, pre-order, post-order, in-order
7. Recursion, Tail Recursion
   i. Definition of a BST
   ii. Search, insert, and remove from a BST
   iii. AVL: Height-Balanced trees definition
   iv. Insert and remove from an AVL tree, rotations
   v. Definition of a Red-Black tree, insert and remove functions

   i. Node struct for a general purpose tree

10. The Priority Queue ADT
    i. Min-queues and Max-queues
    ii. Heap definition and array representation
    iii. Insert and percolate up
    iv. Delete and percolate down

11. Sorting Algorithms
    i. \(O(n^2)\) sorts: Selection Sort, Insertion Sort, Bubble Sort
    ii. \(O(n \log(n))\) sorts: Quicksort, Merge Sort, Heapsort
    iii. Other sort: Bucket sort, Radix sort

12. Finite Graphs
    i. Definition, applications
    ii. Directed-undirected, weighted, path, cycle, connectivity
    iii. Breadth-first search, depth-first search
    iv. Dijkstra's Shortest Path Algorithm,
    v. The Collection of Disjoint Sets (Union-Find) ADT
    vi. Minimum Spanning Tree, Kruskal’s and Prim’s algorithms