Chapter I.2

- 2.4 Summations and Recurrences
  - e.g., recurrence relation and function
- 2.5 Recursion
  - e.g., execution results for a given recursion program
- No proof in the exam.

Chapter I.3

- Concepts of $O(\cdot)$, $\Omega(\cdot)$, and $\Theta(\cdot)$
- Asymptotic analysis
  - e.g., big-O analysis for a given algorithm or function

Chapter II.4

- 4.1 List: array-based list, linked list (singly linked list), and some basic operations on list
- 4.2 Stacks and 4.3 Queues
  - the usages of stacks and queues - *programming* (e.g., bracket matching)
  - the properties of stacks and queues

Chapter II.5

- 5.1 Definitions and Properties
  - binary tree, binary search tree, level, depth, height, interal node/leaf, full binary tree/complete binary tree
- 5.2 Tree traversals
  - Pre-order, In-order, Post-order, and Level-order
  - Find the ordering of nodes given a tree
- 5.4 Definition of Binary Search Tree (BST)
  - the relationship between the depth and the number of ndoes in a BST
- 5.5 Heaps and Priority Queue, and 5.6 Huffman coding tree
  - Concepts, and examples (toy problems)
  - Algorithm to build a Huffman tree (greedy algorithm), Assigning and using Huffman codes
  - No programming

~~**Chapter II.6**~~

Chapter III.7

- Three $\Theta(n^2)$ sorting algorithms
- *Programming*
  - Insertion sort, Bubble sort, Selection sort
  - Mergesort, Quick sort

Chapter III.8

- Concepts about disks, I/O

Chapter III.9

- 9.1 Concepts about unsorted and sorted arrays
- 9.4 Hashing
  - Hash functions, Collision issues

~~**Chapter III.10**~~

Chapter IV.11

- Basic terminology
  - Directed/undirected graph, weighted/unweighted graph, path, length
- Two representation forms of a graph
  - Adjacency Matrix, e.g., Figure 11.3
  - Adjacency Lists, e.g., Figure 11.4
- 11.3 Graph Traversals
  - Depth-first search, Breadth-first search, Topological sort (e.g., write the topological sorting result of a graph)
- 11.4 Shortest-Paths problems, 11.5 MST
  - Dijkstra's algorithm, Prim's algorithm, Kruskal's algorithm
  - No programmming, know how to describe the algorithms in words

~~**Chapter IV.12**~~

~~**Chapter IV.13**~~