Course Review

Graphs

Graph Representations

Adjacency List

Adjacency Matrix

Given a small graph, you should be able to illustrate these

Choose which is appropriate for an application
Graph Algs

Single Source Shortest Path Algs

Common Use:

Vertex $\leftrightarrow$ state in world

Edge $\leftrightarrow$ action that moves you between states

directed vs undirected

weighted vs unweighted
See when you can model a problem as a shortest path problem.

Unweighted - BFS starting $O(n+m)$ at source (if there's a goal you can stop when you discover a path)

Weighted -

Dijkstra's alg - No negative weight edges are allowed.
Minimum Spanning Tree
Prim's & Kruskal's Alg

Depth First Search

topological sort

in-place dfs & use in
garbage collection (Mark & Sweep)
Trie-based data structure
find all elements with common prefix

\( \mathcal{O}(\text{# digits in prefix} + \text{# elements returned}) \)

alphabet size
(close)
Taxonomy of ADTs (Secs 2.1-2.6)

- manually positioned
  - Positional Collection ADT
    - dynamic array, circular array, doubly-linked list...
  - membership
  - Set
    - hashing-based data structures
    - Priority Queue
    - Ordered Collection
    - Digitized Ordered Collection
    - Spatial Collection

- algorithmically positioned
  - untagged
    - compare values
    - unique order
    - multiple orders
    - Spatial Collection
  - tagged
    - tagged version for each
Spatial Collections

kd-tree & quad tree

Digitized Ordered Collection

trie, compact trie, compressed trie

at high-level understand adv & disadv to ordered collection data structures

\[ O(n^{\frac{k-1}{k}} + \text{element in box}) \]
Ordered Collection

Binary Search Tree

Balanced Search Tree (Rotation)

Red-Black Tree properties & insertion
I won't ask about deletion

B-trees (+ brief into to B⁺-tree)
I won't ask about deletion (except in really easy case like in homework)

only need to merge (reverse split)
Priority Queue
Binary Heap

Focus is up to here

Covered on Mid-term
Hashing-based data structures
Open Addressing + Separate chaining
Also Direct Addressing
Sorting Alg's

radix sort & counting sort
quick sort
merge sort
insertion

Know how these work, pros & cons, & which is the best choice
Adversary Lower Bound

Designing a strategy for an adversary to answer questions chosen by an algorithm to show how many questions any alg must ask so adv. only has one input test consistent with all past answers.
Asymptotic notation

Divide and conquer algs

master method