Final Review

Final Info

- Final will be from 11:45am to 1:30pm on Thursday 12/14 in class.
- There will be 4 problems.
- Same cheat-sheet rules as in the midterm.

Final Topics

- Everything other than NP-hardness and Randomized Algorithms.
- You should be prepared for everything (Asymptotics, D&C, DP).
- More emphasis will be on the topics taught after midterm:
  - Greedy Algorithms
  - Graph Traversals (BFS, DFS, topological sorting)
  - Shortest Paths (Dijkstra's, Bellman-Ford's)
  - MST (Kruskal, Prim, Boruvka)

Final Topics

- You need to be able to do the following on small graphs:
- Draw the adjacency matrix or adjacency list of a graph.
- Run BFS on a graph.
- Label the connected or strongly connected components of a graph.
- Run DFS on a graph and label the edges.
- Find a topological ordering of a DAG.
- Compute shortest paths lengths and the shortest path tree via Dijkstra or Bellman-Ford.
- Compute a minimum spanning tree of a graph.

Problem 1 (Heap it On – 25 Points). You are given the following minimum heap H containing the keys $[1, 2, 3, 4, 3, 5]$ (see Figure 1). Perform the following operations in sequence and show the updated heap at the end of each operation.

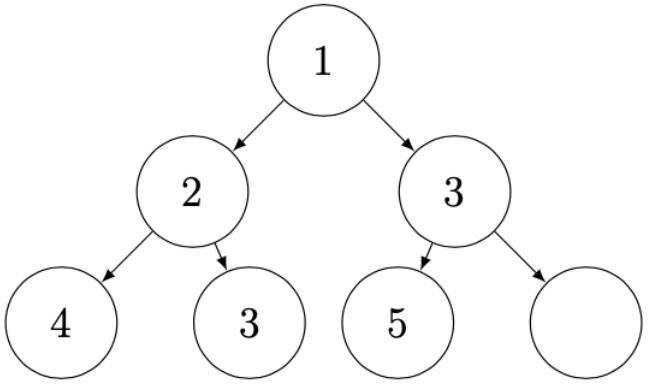
a. $DeleteMin()$.

b. $Insert(2)$.

c. $Insert(1)$.

d. $DeleteMin()$.

e. $DeleteMin()$.



Problem 2 (Allomantic Alloys – 25 Points). There is a collection of n bags each with a different metallic powder (copper, iron, zinc, etc.). Bag i contains w_i grams of total value v_i dollars of metal i.

1. Describe an algorithm to create W grams of an alloy of lowest possible total value by combining suitable quantities of the different metallic powders. Formally, your algorithm must find fractions $\alpha_i \in [0, 1]$ for each metal i such that $\sum_i \alpha_i v_i$ is minimized subject to the weight target W being achieved, that is, subject to $\sum_i \alpha_i w_i = W$.

2. Prove the correctness of your algorithm.

3. Briefly analyze its running time.

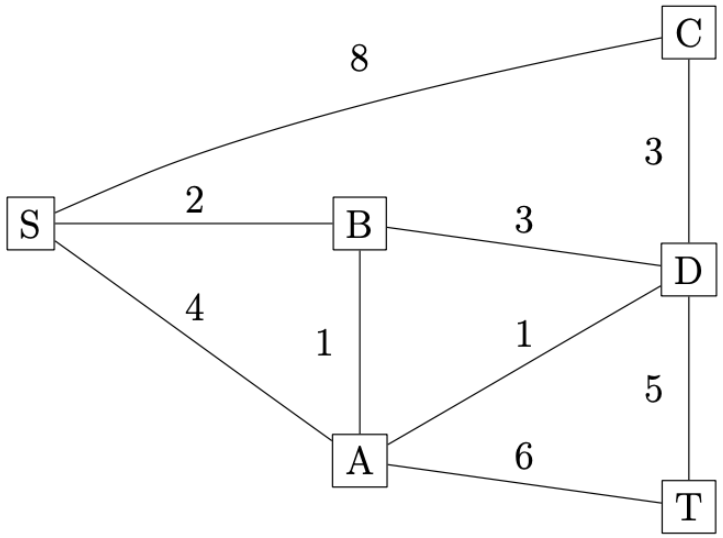
Problem 3 (Smallest Dominating Subset – 25 Points).

a Given a set of positive integers S, describe a greedy algorithm that runs in $O(n \log n)$ to find the smallest subset A of S such that the sum of elements in A is more than the sum of the remaining elements.

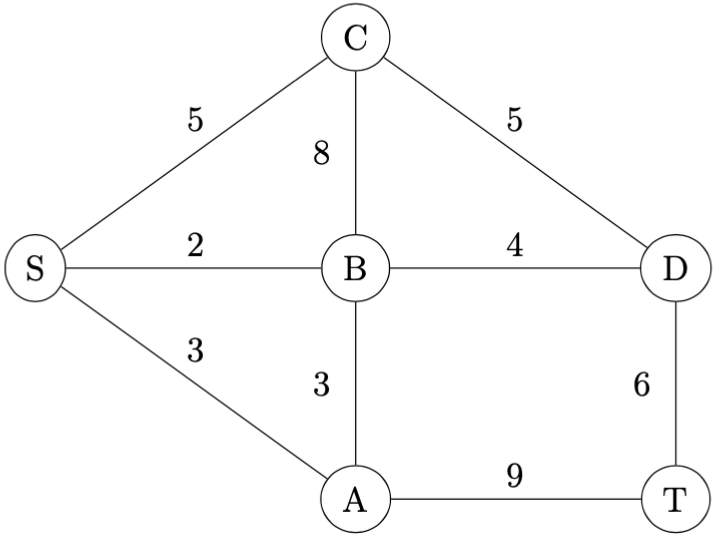
b Prove that your algorithm is optimal using the exchange argument and show that it runs in $O(n \log n)$ as required.

c Describe how you can modify your algorithm to run in $O(n + k \log n)$ where k is the size of the optimal subset A. For this, treat k as a parameter, not a constant.

Problem 4 (Execute Dijkstra's Algorithm – 25 Points). Execute Dijkstra's algorithm to find the shortest path from S to every other vertex on the following graph. Give the final output graph with the shortest distance to each node from s and the order in which edges are added to the output graph.



Execute Kruskal's algorithm to find the minimum spanning tree on the following graph. Then execute Prim's algorithm. In both cases, give the final output tree and the order in which edges are added to the output tree.



Given a directed graph $G = (V, E)$ with positive integer weights on the edges, a source $s \in V$, a destination $t \in V$, and an integer k, design an efficient algorithm to determine the shortest-weight path from s to t containing at most k edges; if there is no path from s to t using at most k edges, your algorithm must indicate so.

You are given a directed graph G and an integer weight $w(v)$ for each vertex v. For each vertex u, define $R(u)$ to be the set of all vertices that are reachable from u (by a directed path) in G.

For each vertex u, define the *reach* of u to be the largest weight of a vertex in $R(u)$. For instance, if the set of vertices that u can reach in G is $\{x, y, z\}$ and the weights of x, y, and z are 3, 9, and 4, respectively, then the reach of u is 9.

Design an algorithm that computes the reach of v, for each v in G. Analyze the worst-case running time of your algorithm. The more efficient your algorithm is in terms of its worst-case running time, the more credit you will get.