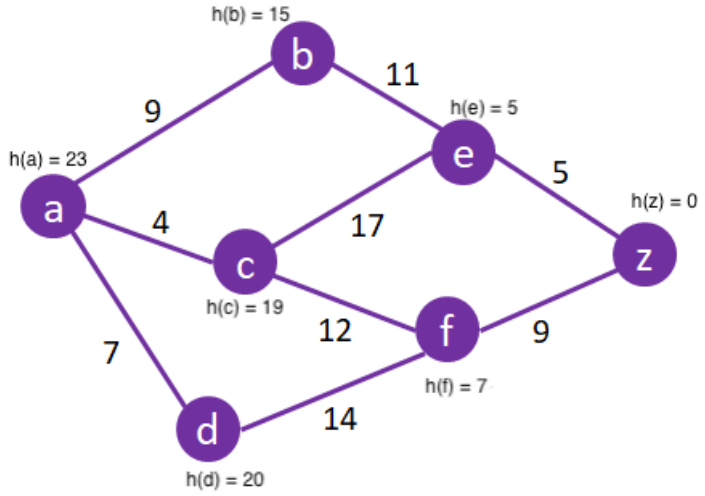
CS 61B Spring 2020 | Small Group Tutoring
Section 11: Graphs, Shortest Path, UF | Worksheet 14

1 Dijkstra's and A*



(a) Given the graph above, run Dijkstras algorithm starting at node a. At each step, write down the entire state of the algorithm. This includes the value dist(v) for all vertices v for that iteration as well as what node was popped off of the fringe for that iteration. List the final shortest distances to every vertex and state the runtime.

v	init	Pop __	Pop __	Pop __	Pop __	Pop __	Pop __	Pop __
dist(a)	**0**							
dist(b)	∞							
dist(c)	∞							
dist(d)	∞							
dist(e)	∞							
dist(f)	∞							
dist(z)	∞							

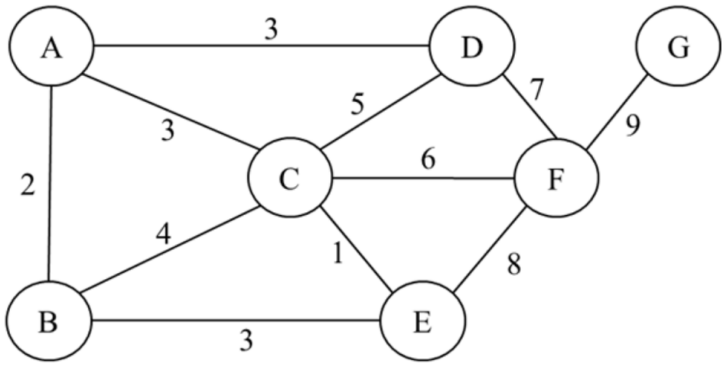
(b) Now run A* search on the graph above to find the shortest distance from node a to node z. If there is a tie, choose the node that was inserted first. Write down the order in which vertices are dequeued.

v	init	Pop __	Pop __	Pop __	Pop __	Pop __
(dist(a), h(a))	(0, 23)					
(dist(b), h(b))	(∞, -)					
(dist(c), h(c))	(∞, -)					
(dist(d), h(d))	(∞, -)					
(dist(e), h(e))	(∞, -)					
(dist(f), h(f))	(∞, -)					
(dist(z), h(z))	(∞, -)					

(c) What must be true about our graph in order to guarantee Dijkstras will return the shortest paths tree to every vertex?

(d) Recall that the Dijkstra's uses a min priority queue ordered by distance from the start node. How would we modify the priority queue for A* search?

2 Minimum Spanning Tree



(a) Given the graph above, run Kruskal's and Prim's algorithm to determine the minimum spanning tree of this graph. For Prim's algorithm, assume we start at node A and fill in the following chart including the value cost(v) for all vertices v for that iteration as well as which node was popped off of the fringe for that iteration. (Note: Ties are broken in alphabetical order)

v	init	Pop __	Pop __	Pop __	Pop __	Pop __	Pop __	Pop __
cost(a)	**0**							
cost(b)	∞							
cost(c)	∞							
cost(d)	∞							
cost(e)	∞							
cost(f)	∞							
cost(g)	∞							

(b) Does Kruskal's algorithm for finding the minimum spanning tree work on graphs with negative edge weights? Does Prim's?

(c) True or False: A graph with unique edge weights has a unique minimum spanning tree.

3 Union Find Quick Checks

1. Consider the naive implementation of union find, where for each element, we store the set number that an element belongs in. All elements belonging to the same set has the same set number. For connect(), we change the set number of all elements belonging to the smaller set to be the set number of the element that belongs to the larger set. What is the worst runtime for find and union individually?

$$\theta(\text{find}) =$$

$$\theta(\text{union}) =$$

2. Now consider an implementation of union find where each set is represented by a tree with a single root. When we call connect() on two sets, the root of the smaller set becomes a child of the root of the larger set. What is the worst case runtime for find and union individually?

$$\theta(\text{find}) =$$

$$\theta(\text{union}) =$$

3. Now, let's consider the same union find implementation as above, but with path compression. When we call connect(), we reassign the parent of every element we pass to be the root of the set. What is the (approximate) amortized runtime for find and union individually?

4 Optional: Fiat Lux

After graduating from Berkeley with solid understanding of CS61B topics, Josh became a billionaire and wants to build power stations across Berkeley campus to help students survive from PG&E power outages. Josh want to minimize his cost, but due to the numerous power outages when he took CS61B, he did not learn anything about Prim's or Kruskal's algorithm and he is asking for your help! We must meet the following constrains to power the whole campus:

- There are V locations where Josh can build power stations, and it costs v_i dollars to build a power station at the i^{th} position.
- There are E positions we can build wires and it cost e_{ij} to build a wire between location i and j.
- All locations must have a power station itself or be connected to another position with power station.
- $e_{ij} << v_i, \forall i, j$

Modify the Prim's or Kruskal's algorithm taught in class that will minimize the cost Josh will spend while still fulfilling the constrains above.