CSE 373: Section 9

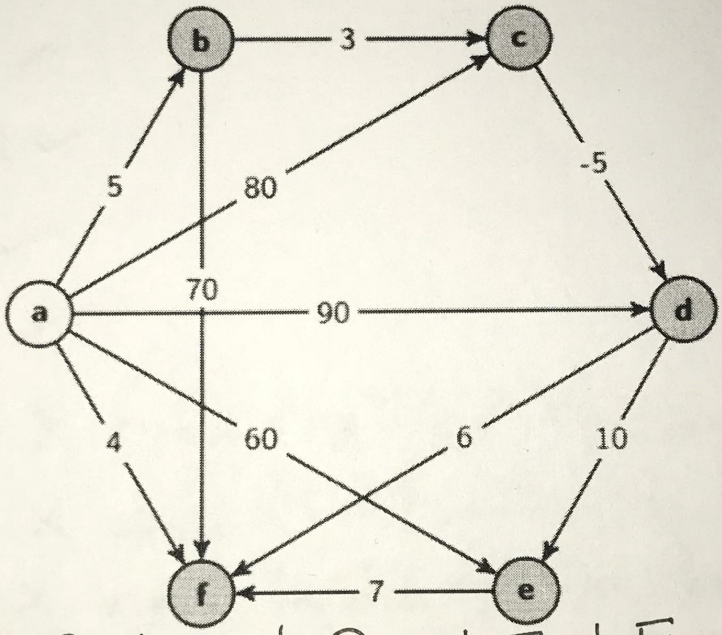
Graphs

November 30th

Dijkstra's Algorithm

Use Dijkstra's algorithm to find the costs of the shortest paths from vertex a to the other vertices. Show your work at each step and indicate the order which vertices are added to the known set. In the case of a tie, add the vertex that comes first alphabetically.

Note: the edge (c,d) has negative weight. Run Dijkstra's and observe if this has an effect on the correctness of the output



order

	A	F	B	C	D	F	E
a	0, 0						
b	∞, 5, A						
c	∞, 80, A		8, B				
d	∞, 90, A			3, C			
e	∞, 60, A				13, D		
f	∞, 4, A				9, D		

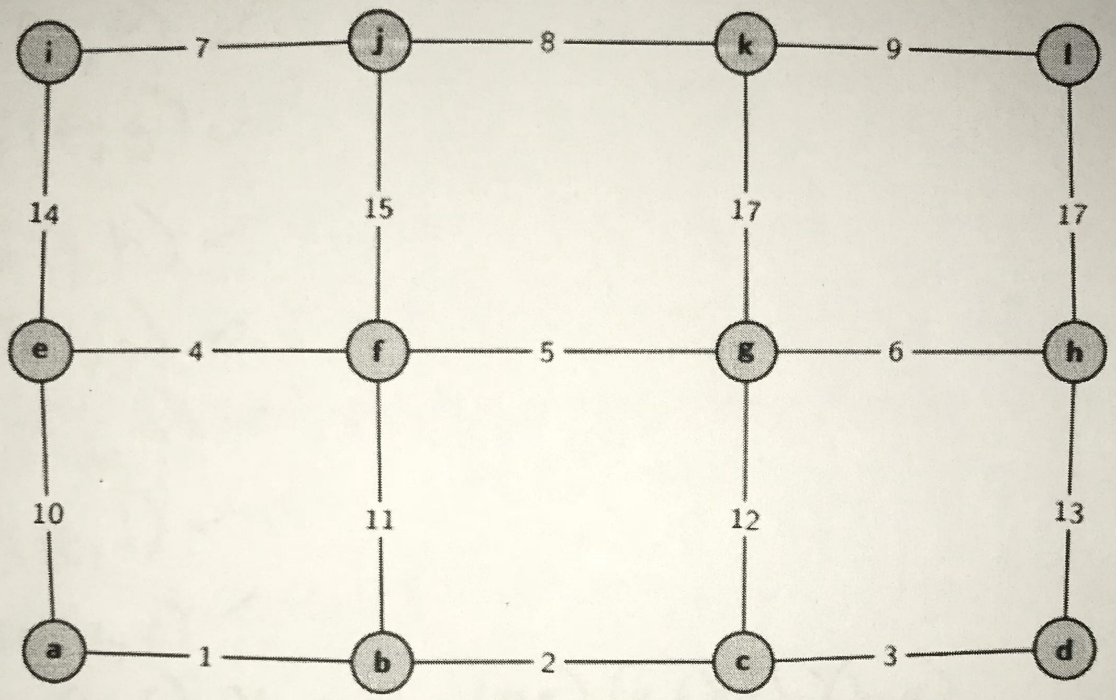
Even though (c, d) is negative, our solution is unaffected

Minimum Spanning Trees

For the following graph, find the minimum spanning tree using (a) Prim's algorithm and (b) Kruskal's Algorithm.

For Prim's, break ties by choosing the vertex that comes first alphabetically.

For Kruskal's, provide *an* ordering of the edges before running the algorithm. Notice there are multiple possible orderings. Do these orderings create different MSTs?



Prims

Start at vertex a.

(a,b)(b,c)(c,d)(a,e)(e,f)(f,g)
(g,h)(e,i)(i,j)(j,k)(k,l)

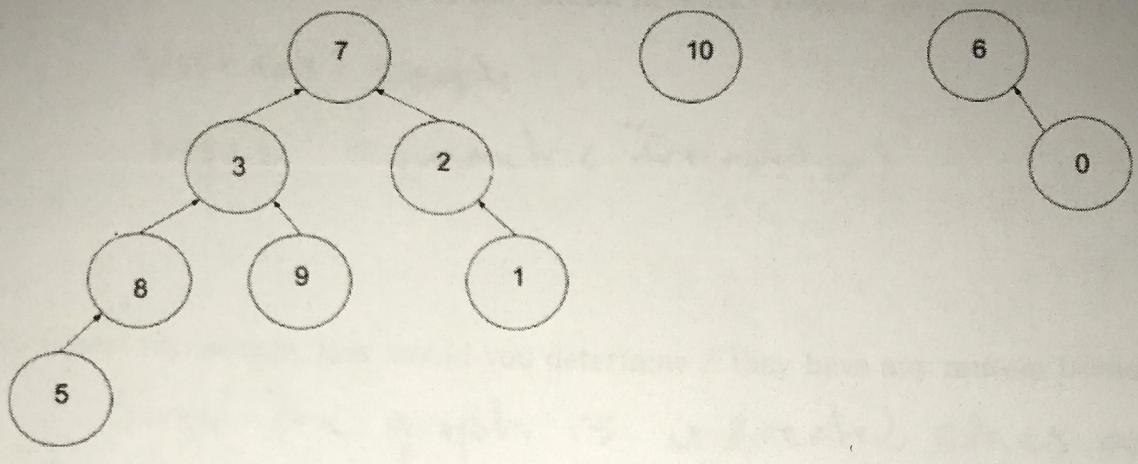
	A	B	C	D	E	F	G	H	I	J	K	L
a	0											
b	∞	1,a										
c	∞		2,b									
d	∞			3,c								
e	∞	10,a										
f	∞		11,b		4,E	5,f						
g	∞			12,c			6,g					
h	∞			13,d								
i	∞				14,E	15,f			7,I	8,J	9,K	
j	∞						17,g	17,h				
k	∞											
L	∞											

Continue your work here

1 (a,b) ✓
2 (b,c) ✓
3 (c,d) ✓
4 (e,f) ✓
5 (f,g) ✓
6 (g,h) ✓
7 (i,j) ✓
8 (j,k) ✓
9 (k,l) ✓
10 (a,e) ✓
11 (b,f) ✗ cycle (a,e)(e,f)(f,b)(b,a)
12 (c,g) ✗ cycle (a,b)(b,c)(c,g)(g,f)(f,e)(e,a
13 (d,h) ✗ cycle (a,b)(b,c)(c,d)(d,h)(h,g)(g,f)(f,e)(e,a)
14 (e,i) ✓ → terminate here
15 (f,j)
17 (g,k)(h,l)

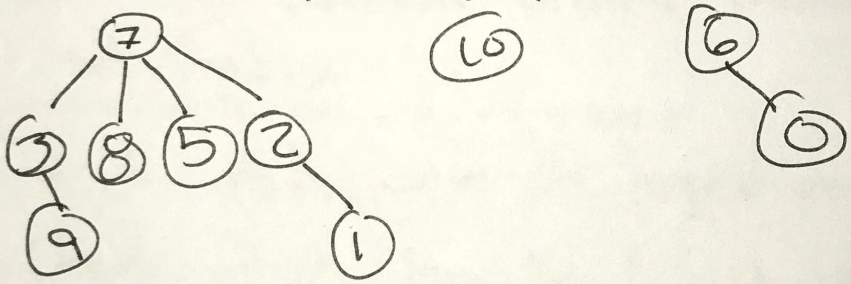
Uptrees and Union Find

Given the following up-tree, answer the following questions. Assume that the up-tree implements both weighted unions and path compression.



1. Draw the resulting trees after a call to `find(5)`. Also, indicate what will be returned by the call.

returns: 7



2. Give the array representation of the up-tree data structure from part 1 after a call of `union(1,5)`

no union occurs since same representative

0	1	2	3	4	5	6	7	8	9	10
6		7	7	-1	7	-2	-7	7	3	-1

assumed to be present and not-joined

Short Answer

1. Suppose you have data from an undirected social network for some people.

   (a) How would you represent this social network in a graph? Consider all graph properties? What factors of the "social network" impact your choices?

   Vertices: people
   edges: symmetric "friendships"

   (b) Given two people, how would you determine if they have any mutual friends?

   Since the graph is undirected, edges are symmetric. Check to see if there are any common people in their adjacency list/matrix

   (c) How would you find the person with the most friends?

   Find the vertex with the largest degree
   Check which has the largest adjacency list O(E)
   or which row of the adjacency matrix is most full
   $O(|V|^2)$

   (d) Propose a method to find an individual's degree-of-separation from Paul Erdős.

   Iterative deepening or modified BFS.
   Since the edges don't have weight, dijkstra's algorithm is unnecessary.

2. Given a graph G propose a method to find if it has a cycle if it is:

(a) Undirected

Any traversal. If you revisit a vertex that has already been visited (provided you did not just come from that node) then a cycle exists. (A)—(B) is a cycle

(b) Directed

Topological sort

(c) How might the Union-find ADT be helpful for finding cycles in a graph?

Like kruskals, add edges until a cycle occurs. (Only works for undirected graphs)