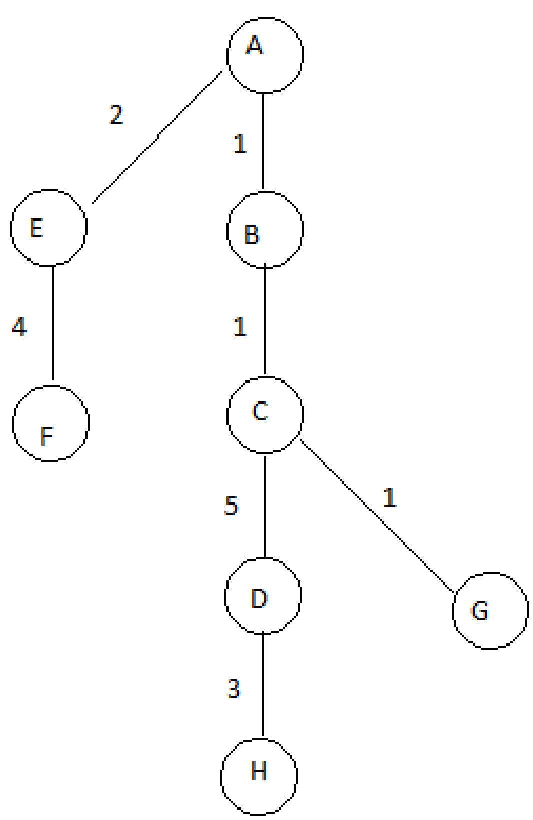
David Sun
CMPS 102
Homework 4

<u>Problem 1:</u>

(a) The chart below show's Dijkstra's algorithm on the graph. The letter next to each weight keeps track of the parent node of the shortest path tree for that iteration. Assume vertex A is the source.

	A	B	C	D	E	F	G	H
A	0	1 (A)	∞	∞	2 (A)	10 (A)	∞	∞
B	0	1 (A)	2 (B)	∞	2 (A)	9 (B)	4 (B)	∞
C	0	1 (A)	2 (B)	7 (C)	2 (A)	9 (B)	3 (C)	∞
E	0	1 (A)	2 (B)	7 (C)	2 (A)	6 (E)	3 (C)	∞
G	0	1 (A)	2 (B)	7 (C)	2 (A)	6(E)	3 (C)	13 (G)
F	0	1 (A)	2 (B)	7 (C)	2 (A)	6 (E)	3 (C)	13 (G)
D	0	1 (A)	2 (B)	7 (C)	2 (A)	6 (E)	3 (C)	10 (D)
H	0	1 (A)	2 (B)	7 (C)	2 (A)	6 (E)	3 (C)	10 (D)

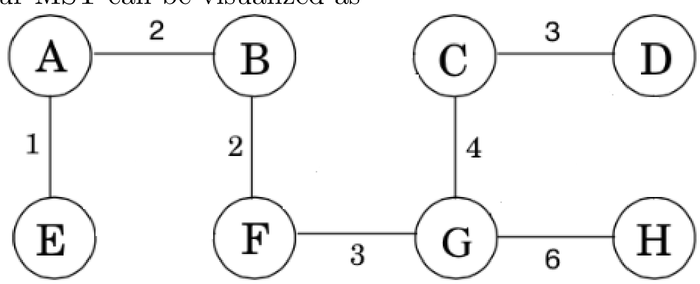
(b) Show the shortest path tree corresponding to running Dijkstra on this graph.
From the table, we can derive the shortest-path tree by looking at the top-most row and bottom-most row. In this case, the edge set for our shortest-path tree is $\{AB, BC, CD, AE, EF, CG, DH\}$. Thus the tree can be visualized as:



Problem 2: Trace Kruskal's algorithm on the graph.
First we sort the edges by weight and we construct a tree by taking the edges in order so long as it does not introduce a cycle.

Edge	Weight	Take?
A - E	1	Yes
A - B	2	Yes
B - F	2	Yes
C - D	3	Yes
F - G	3	Yes
E - F	4	No, introduces cycle ABEF.
C - G	4	Yes
B - E	5	No, introduces cycle ABE
C - F	5	No, introduces cycle CFG
D - G	5	No, introduces cycle CGD
G - H	6	Yes
B - C	7	No, the tree is spanning
D - H	7	No, the tree is spanning

Our MST can be visualized as



Problem 3:

1. Consider the edge (B, E) in the graph of Problem 2. Either show that there is a minimum cost spanning tree using this edge by applying the cut property, or prove that this edge is not used in any minimum cost spanning tree using the cycle property.

2. Ditto for edge (C, G).

Part 1: Consider the cycle $A - B$, $B - E$, $E - A$. Out of all these edges $B - E$ has maximal weight. By the cycle property, edge $B - E$ is not part of any MST of the graph in problem 2.

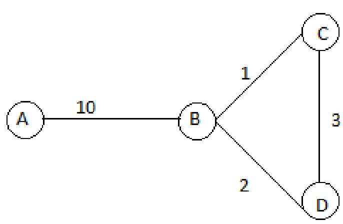
Part 2: Consider the cut $\{C, D\}$. The corresponding cut-set is the set of edges $\{C - B,\ C - F,\ C - G,\ D - G,\ H - D\}$. Out of these set of edges, edge $C - G$ has the minimal weight of 4, so by the cut property, the MST of problem 2 must contain edge $C - G$.

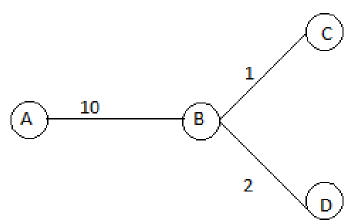
<u>Problem 4:</u> Prove or disprove the following statements about an arbitrary undirected graph $G = (V, E)$:

1. If e is part of some MST of G, then it must be a lightest edge in some cutset of G.
2. If graph G has more than $|V| - 1$ edges, and there is a unique heaviest edge, then this edge cannot be part of a minimum spanning tree.
3. The shortest path between two nodes is necessarily part of some MST.

Part 1: Assume towards a contradiction that if e is part of some MST of G, then it is not necessarily the lightest edge in some cutset of G. Let CS represent the cutset that connects the cut with the rest of G. It is given that $e \in CS$ and suppose $\exists e' \in CS$ such that $weight(e) > weight(e')$. Our claim is that the MST of G currently contains contains e instead of e'. The cut property states that if an edge x is the minimum cost edge with exactly one endpoint in a particular cut, then the MST of that graph must contain x. However, our MST of G contained e, rather than the lower weighted edge, e'. Thus by assuming that e is not the lightest weight edge in some cutset of G, we have violated the cut property, so e is indeed the lightest edge in some cutset of G when e is in the MST.

Part 2: False, consider the graph G below.





and its MST

The graph contains 4 vertices and 4 edges, so G has more than $|V| - 1$ edges. The unique heaviest edge is $A - B$. The MST of G still contains $A - B$ because the tree cannot span G without it.

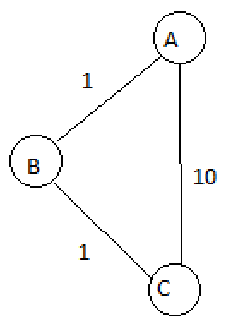
Part 3: False, consider the graph in problem 2. The shortest distance between the nodes F and C is has weight 5 and is the edge $F - C$. However, the MST of the graph in problem 2 does not contain the edge $F - C$.

<u>Problem 5:</u> Consider an undirected graph $G = (V, E)$ with nonnegative edge weights $w_e \geq 0$. Suppose that you have computed a minimum spanning tree of G, and that you have also computed shortest paths to all nodes from a particular $s \in V$.
Now suppose each edge weight is replaced by $w'_e = f(w_e)$, where f is a strictly increasing function.

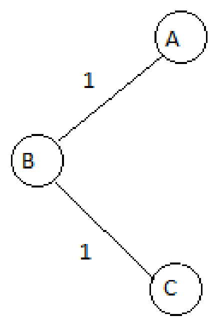
1. Does the minimum spanning tree change? Give an example where it changes or prove that it cannot change.

2. Do the shortest paths change? Give an example where they change or prove they cannot change.

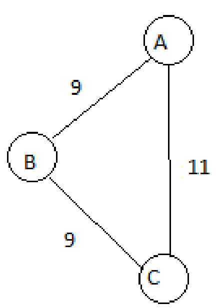
Part 1: The minimum spanning tree does not change. Assume towards a contradiction that when the function is applied to every $e \in E$, that a different MST M' can be constructed where M' contains a different edge from the original MST of G. By definition of an MST, the sum of the weights in the original MST is minimized. Note that $f(w_e)$ is strictly increasing. Thus for every $e_1, e_2 \in E$ and $e_1 < e_2$, then $f(e_1) < f(e_2)$. Then if M' contained a different edge from the original MST, M' must have taken an edge whose weight was greater than some other edge taken by the original MST e_x, where $f(e_x)$ has lower weight than the edge taken and would have still spanned the graph. In this case weights of M' is no longer minimal, thus M' must be the exact same MST of G.
Part 2: The shortest path tree can potentially change. Consider the following graph



Shortest Path tree:



We know that f is a strictly increasing function, so let $f(1) = 9$ and $f(10) = 11$. That means edges $A - B$ and $B - C$ will have weight 9 and edge $A - C$ will have weight 11. Our new shortest path tree changes as such



Shortest path tree: