Final Algorithms Credit

Q1: T/F no justification

(g) T F [4 points] Negating all the edge weights in a weighted undirected graph G and then finding the minimum spanning tree gives us the *maximum*-weight spanning tree of the original graph G.

Solution: True.

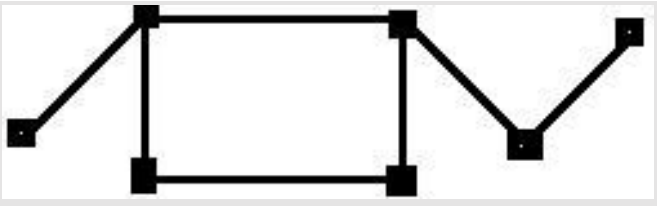
(b) For any pair of distinct vertices $s, t \in V$, the cost of a path between s and t in T is minimal among all paths from s to t in G.

True False

3. If you have G=(V,E), and S subset from V, then the least weight edge connecting any vertex from S to V-S is in the MST of G

4. . If P=NP then P is subset of NP-hard

Q2: MCQ

1- law 3ndy 10 vertices w bst5dm all-pairs shortest path ba7seb el W^kam (w^9 , W^10) ?
2- From which problem did we prove that vertex cover is np complete ?
3- In the graph given, what is the number of vertices in the min vertex cover set
4-  =3
5- which algorithm did we perform DFS on graph then perform DFS on graph transpose and produce a set? (strongly connected)

Q3: T/F with justification

1. In any MST algorithm, the addition of a constant to all weights results in a different MST?

Q4:

a) You are given an array of fuel stations D[n] where d1<d2<..dn you want to travel from A to B (on a straight line) where all distances in the array are given relevant to point A (ie. D1 is 15 m from A , d2 is 20 m from A and so on) . when ur car is supplied with fuel it can only move m km away and u may assume that the distance between two stations is at most m. Design an algorithm to choose the min number of fuel stations to stop by (so that u don't run out of fuel) on ur way from A to B.
b) Mention its complexity

Q5:

a) Write a recursive fn to find Longest Increasing Subsequence in a given array
b) Modify ur fn to use top down dynamic programming
c) Can u use greedy algorithm to give u optimal soln? and why?

Q6: Find an optimized algorithm to Count cycles in directed graph.

Q7: Short questions

a) Can prim's algorithm be used in a directed graph? Why?
b) Exact condition emta tst3ml adjacency matrix we emta tst3ml adjacency list l Graph given en entry adjacency list fe 1 word pointer to list we list el entry ta5od 2 words (1 word data we 1 word pointer) while Adjacency matrix kol entry 1 word
c) For a tree what's the fastest variation from all pair shortest path we idea bta3to w complexity w leh est5dmto bdal el tnyeen ?
d) Why can't we just add constant values to the edges in the re-weighting technique in Johnson's algorithm?
e) why topological sorting can't work for cyclic graph?

Other Questions:

1. Determine in each of the following which algorithm used greedy and which used DP:
    a. Prim
    b. Floyd
    c. Dijkstra
    d. Kruskal