Name: ________________________________ J#: _______________________

CSC 323 Algorithm Design and Analysis, Fall 2016 Quiz 8 (Take Home Quiz)

Maximum Points: 50 **Due on: Dec. 1, 2016: 11.30 AM**

Print this page and staple it (as a cover sheet) along with your answer sheets.
Quiz submission without this coversheet will NOT be graded.

ANSWER ALL OF THE FOLLOWING THREE QUESTIONS (Each question is worth 10 points)

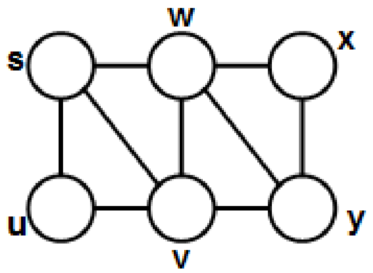
Q1) Prove that the Kruskal's algorithm when applied to determine the **maximum spanning tree** indeed finds one [i.e., Prove the correctness of the Kruskal's algorithm for **maximum spanning tree**]. Note: You will get ZERO points if you write the proof for a minimum spanning tree.

Q2) If the edges of a graph of have unique weights, prove that there exists only one **maximum spanning tree** for the graph. Note: You will get ZERO points if you write the proof for a minimum spanning tree.

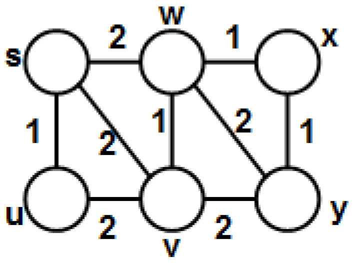
Q3) Analyze the run-time complexity of the Dijkstra's shortest path algorithm if the data structure used to store the estimates for the shortest path weights is an array (and not a heap: the data structure used in the lecture slides). Show your analysis in detail for each step of the pseudo code as shown in the slides for a heap.

ANSWER ANY ONE OF THE FOLLOWING TWO QUESTIONS (20 points)

Q4) Given a graph of all negative edge weights:
(a) Design an algorithm that could be used to determine the **largest weight paths** starting from a particular vertex (i.e., the source vertex) to each of the other vertices in the graph. The largest weight path from a source vertex *s* to a target vertex *d* is the path with the largest sum of the negative edge weights (i.e., smaller value for the absolute sum).
(b) What is the time complexity of your algorithm?
(c) For the following graph, assign negative weights of your choice (in the range -10 ... -1) to the edges and determine the largest weight path starting from a vertex of your choice. Clearly indicate the chosen edge weights and the chosen starting vertex (i.e., the source vertex). Show all the steps.



Q5) Given a graph G = (V, E) whose edge weights are either 1 or 2:
(a) Design a Θ(V + E) algorithm to determine the **minimum weight paths** starting from a source vertex to the rest of the vertices in the graph.
Note: You should not use Dijkstra algorithm as its time-complexity is Θ(ElogV)
(b) Analyze the space complexity of your algorithm.
(c) Show the working of your algorithm on the following graph. Pick a source vertex of your choice and indicate the same clearly. Show the minimum weight paths from the chosen vertex to the rest of the vertices.


w
s
2
1
x
1
2
1
2
1
u
2
2
y
v