Home Work #12
DUE: See Canvas
(upload portrait-mode PDF on Canvas)

☞ Handwritten assignments will not be accepted.

Start your assignment with the following text provided you can honestly agree with it.

- I certify that every answer in this assignment is the result of my own work; that I have neither copied off the Internet nor from any one else's work; and I have not shared my answers or attempts at answers with anyone else.

1. A weighted graph G is defined by the following (weighted) adjacency list:

node	*<(neighbor, weight)>*
u	$< (v,15),(w,4) >$
v	$< (y,3) >$
w	$< (v,9),(x,6),(y,4) >$
x	$< (v,2) >$
y	$< (x,3) >$

Trace the execution of DIJKSTRA-SSSP(G,u) by filling out the following tables for d and Π. Each table has one row per node.

In each column, show the value of $d[\ ]$ and $\Pi[\ ]$ respectively for that node at the beginning of the while-loop.

Circle the d-value for the node about to be extracted.

Indicate a node that is no longer in Q with a — (dash).

Node	$d[\ ]$	$d[\ ]$	...
u			
v			
w			
x			
y			

Node	$\Pi[\ ]$	$\Pi[\ ]$	...
u			
v			
w			
x			
y			

From the above tables, how would you find the shortest path from u to v?

2. Prove that in Dijkstra's algorithm, once a vertex is returned by EXTRACTMIN (Line 7), no vertex will ever be returned in future iterations with a smaller $d[\,]$ value even though some $d[\,]$ values decrease in the relaxation step (Line 11).
   (Line numbers are from slide 25/46 in graphs3.pdf.)

3. Professor Flippy claims to have found an algorithm for the SSLSP (single-source longest simple path) problem. That algorithm modifies Dijkstra's algorithm (for SSSP) by flipping the sign in the initialization step (from positive infinity to negative infinity in Line 2), replacing $>$ by $<$ in the relaxation step (Line 10), and changing EXTRACTMIN to EXTRACTMAX (Line 7).
   (Implementation: replace min-heap by max-heap.)

   Show that the professor is incorrect by providing a counter-example: draw a graph and show the execution of the algorithm in each iteration using a table (as in the first question of this homework) and point out the error.

   Note: A *simple* path means is one in which no vertex is repeated, i.e., the path doesn't traverse a loop.

   (Line numbers are from slide 25/46 in graphs3.pdf.)