Quiz 4 Solutions

February 3, 2014

1. True or False: The Bellman-Ford algorithm can be used to detect the existence of a negative-weight cycle in a graph.
   Answer. True. The Bellman-Ford algorithm can find if a negative-weight cycle exists by executing $|V|$ iterations and checking whether any of the $\text{dist}$ values decreases on the final iteration. If any any of the $\text{dist}$ values decrease, the graph contains a negative cycle.

2. Suppose that a new priority queue (PQ) implementation is discovered that takes $O(\sqrt{n})$ time for a single $\text{deleteMin}$ operation and $O(1)$ time for a single $\text{insert}$ operation. For which of the following classes of graphs is Dijkstra’s algorithm implemented with the new PQ implementation NOT asymptotically faster than with an array implementation? [Note: For time complexity of Dijkstra with an array implementation of a PQ, see the “Which heap is best?” table in Section 4.5, Page 114, of the Dasgupta et al. textbook.]
   • A tree graph $G = (V, E)$ with $|E| = |V| - 1$.
   • A complete graph $G = (V, E)$ with $|E| = |V|(|V| - 1)/2$.
   • A graph $G = (V, E)$ with $|E| = \sqrt{|V|}$.
   • A graph $G = (V, E)$ with $|E| = \frac{|V|}{4}$.

   Answer. Complete graphs. The complexity of Dijkstra’s can be computed as $O(|V| \cdot \text{deleteMin} + (|V| + |E|) \cdot \text{insert})$. For an array, $\text{deleteMin}$ requires $O(|V|)$ and this dominates the running time for an overall complexity of $O(|V|^2)$ (see page 125, Dasgupta et al. textbook). So, this new implementation is asymptotically faster than an array except when $|E| = \Omega(|V|^2)$, as in complete graphs.

3. Dijkstra’s algorithm is executed on an arbitrary, directed, connected graph $G = (V, E)$, starting from a source vertex $s \in V$.
   True or False: If $G$ contains a negative-weight cycle, Dijkstra’s algorithm will still correctly find the weights of the shortest paths from $s$ to all other vertices $v \in V$.
   Answer. False. Dijkstra’s algorithm will terminate, but some of the calculated shortest path distances from $s$ to all other vertices may be incorrect.

4. Dijkstra’s algorithm is executed on the following directed graph, starting from vertex $A$ and breaking all ties in lexicographic order. What is the
weight of the first finite path calculated by the algorithm from vertex $A$ to vertex $H$? (i.e., when Dijkstra’s algorithm first updates the value of the estimated shortest $A - H$ path distance from $\infty$ to a new finite value, what is the new value?)

(Credit: Dasgupta et al. textbook.)

Answer. 8, the first path from $A$ to $H$ found is $A \rightarrow B \rightarrow C \rightarrow D \rightarrow H$.

5. Let $G = (V, E)$ be an undirected graph with positive edge weights. Consider a shortest path $P$ from $s \in V$ to $t \in V$ in $G$.

True or False: If we double the weight of every edge in $E$, then it is possible for $P$ to no longer be a shortest path from $s \in V$ to $t \in V$ in $G$.

Answer. False. The relative distances of the paths in the graph do not change if the edge weights of the graph are scaled, i.e., multiplied, by any positive value.

6. In an arbitrary, connected, directed graph $G = (V, E)$ with no negative-weight cycles, assume that there is a shortest path of exactly $k$ edges from $s \in V$ to $t \in V$.

True or False: After the $k^{th}$ round of update operations of the Bellman-Ford algorithm, starting from $s$, the current estimate of the $s$-$t$ shortest path distance as evaluated by the algorithm is correct. [Note: Use the update procedure of the Bellman-Ford algorithm as described in Figure 4.13 of the Dasgupta et al. textbook.]

Answer. True. Bellman-Ford algorithm calculates estimates for paths of length $i$ in the $i^{th}$ iteration. Therefore, it calculates an estimate for the $s$-$t$ path containing $k$ edges in the $k^{th}$ iteration, and since this is the shortest path, any other $s$-$t$ path found during the execution of the algorithm will not change this estimate.

7. The Bellman-Ford algorithm is executed on the given directed graph starting from vertex $A$. What is the estimate of the shortest distance from $A$ to $D$, as evaluated after performing two rounds of the update operation on all edges? [Note: Use the update procedure of the Bellman-Ford algorithm as described in Figure 4.13 of the Dasgupta et al. textbook.]
2. After the first round, the estimated distance to $D$ is $\infty$. After the second round, the estimated distance to $D$ is 2. If the algorithm is allowed to continue for a further round, the estimated distance to $D$ is updated to $-7$, which is the distance through the shortest path from $A$ to $D$.

8. The frequencies of characters used in an arbitrary message are as follows: $A : 4$, $B : 3$, $C : 7$, $D : 2$, $E : 5$, $SPACE : 9$. What are the values, respectively, of the parent node of $A$, the parent node of $C$, and the root node of the Huffman tree generated when encoding the above six characters using Huffman coding?

A. 5, 9, 30
B. 9, 12, 36
C. 9, 12, 30
D. 5, 12, 36

Answer. C. By constructing the Huffman coding tree, it is found that the parent of $A$ has value 9, the parent of $C$ has value 12, and the root of the tree has a value equivalent to the sum of all the frequencies, 30.

9. True or False: It is possible to find all longest paths in a DAG in $O(|V| + |E|)$ time even when the DAG contains edges with both positive and negative weights.

Answer. True. The $\text{dag-shortest-paths}$ algorithm works with both positive and negative edges and the complexity of the algorithm is $O(|V| + |E|)$.

10. Please enter the number of questions in the course survey rolled out in Week 4. (2 points)
A. 23
B. 22
C. 25
D. 20

Answer: B