Quiz 3 Practice Questions

Shortest Path:
1. Design an example of a graph where the shortest path tree is longer than the minimum spanning tree. In the worst case, how much longer can the shortest path tree be than the minimum spanning tree?
2. A “geometric graph” is a special type of graph where the nodes are points on a 2-dimensional surface and edges are straight lines joining pairs of nodes. Show that the shortest path tree of such graphs cannot have edges that cross each other (other than at their endpoints).
3. Can you modify Dijkstra’s shortest path finding algorithm so that it can be used to find longest path in a graph?

Network Flow:
4. Run the Edmonds-Karp algorithm on the following network.

5. In the above problem, if you ran the Ford-Fulkerson algorithm instead of the Edmonds-Karp algorithm, give an upper bound on the number of iterations your algorithm will make in the worst case.
7. Given two vertices s and t of an undirected graph, show how to use network flow to determine whether there are at least two paths between s and t that do not share any common intermediate edge (the paths are allowed to share common vertices).
8. Imagine you had not been taught Network Flows. Try to design an algorithm for computing a bipartite matching. Prove/disprove that it will compute the maximum bipartite matching.
String Algorithms:

9. Show the execution of the Longest Common Subsequence algorithm on the following two strings:
   
   ABCABDCDBA
   BACDDCAB

10. Given a single sequence of numbers, design an algorithm to find the longest monotonically increasing subsequence. For example, in the sequence 23143758 the longest monotonically increasing subsequence is 23 4 58. What is the running time of your algorithm?

11. Given two sequences of length m and n each, we studied in class how we can implement LCSS in O(m*n) time using O(min{m, n}) space. Can you design an LCSS algorithm that only uses O(1) space?

12. Given two sequences S1 and S2, design and analyze an algorithm to find the longest common substring between S1 and S2 (a substring is a subsequence which cannot skip intermediate characters, e.g. CABD is a substring of ABCABDCDBA).

13. Compute the Prefix function by the KMP algorithm for the pattern:
   
   A B C A B D C D B A

14. Give an example of a string and a pattern which represents the “best case” for the KMP algorithm, i.e. the algorithm will run in the fastest possible manner for such a problem instance.

15. Give an example of a string and a pattern which represents the “worst case” for the KMP algorithm, i.e. the algorithm will run in the slowest possible manner for such a problem instance.

16. The input is two strings of characters A = a₁ a₂ . . . aₙ and B = b₁ b₂ . . . bₙ. Design an O(n) algorithm to determine whether B is a cyclic shift of A. In other words, the algorithm should determine whether there exists an index k, 1 <= k <= n such that aᵢ = b((k+i) mod n), for all i, 1 <= i <= n.