
You're expected to work on the discussion problems before coming to the lab. Discussion session is not meant to be a lecture. TA will guide the discussion and correct your solutions if needed. We will not release 'official' solutions. If you're better prepared for discussion, you will learn more. TAs will record names of the students who actively engage in discussion and report them to the instructor; they are also allowed to give some extra points to those students at their discretion. The instructor will factor in participation in final grade.

1. (Basic) Explain the "optimality of subpaths" lemma.
2. (Intermediate) Suppose every edge has weight 1. To compute a shortest path from s to another vertex u , what algorithm would you like to use?
3. (Basic) Consider the graph in Fig 24.4. We would like to run the Bellman-Ford algorithm with s as the starting vertex. Suppose that in each iteration we relax edges in a lexicographic order – (u, v) is considered before (u', v') if u is ahead of u' in alphabetical order, or $u = u'$ and v is ahead of v' in alphabetical order. Show how $v.d$ and $v.\pi$ are updated during the execution of the algorithm.
4. (Basic) Consider the graph in Fig 24.4. Run the Dijkstra algorithm with s as the starting point. Does it correctly compute a shortest path from s to each vertex?
5. (Basic) Consider the graph in Fig 24.6. Run the Dijkstra algorithm with z as the starting point.
6. (Basic) What is the limitation of the Dijkstra algorithm compared to the Bellman-Ford?
7. (Advanced) We are given a directed graph with positive weight on edges. We're also given two disjoint subsets S and T of vertices. We would like to find a shortest path from any vertex in S to any vertex in T . What is your algorithm? The faster, the better.
8. (Advanced) By running the Dijkstra algorithm we obtain a shortest paths tree T with the starting point s as the root. Suppose we deleted an edge (u, v) of the tree from the input graph. To compute a shortest paths tree for this new graph, do we need to run the Dijkstra algorithm from the beginning? Can we partly recycle T ?
9. (Advanced) We're given a directed graph $G = (V, E)$ on which each edge $(u, v) \in E$ has an associated value $0 \leq r(u, v) \leq 1$, which is the probability the communication channel from u to v will not fail. Give an algorithm to find the most reliable path between two given vertices. Assume that the events that distinct edges fail are independent.