
2. Problem 23-4 on Page 641: Alternative minimum-spanning-tree algorithms. For each of the three algorithms, either give a counterexample or prove that it always outputs a minimum spanning tree. Make sure your proof is written clearly and concisely. Also there is no need to describe efficient implementations of these algorithms.


4. Problem 25-2 on Page 706: Shortest paths in $\epsilon$-dense graphs. Skip a). For a $d$-ary min-heap, INSERT takes time $O(\log_d n)$; EXTRACT-MIN takes time $O(\epsilon d \cdot \log_d n)$; and DECREASE-KEY takes time $O(\log_d n)$. Check Chapter 6 and Problem 6-2 if you are interested in $d$-ary min-heaps. But for this problem you may use these facts for free.

5. Show that if CLIQUE (the decision problem, where a pair $(G, k)$ is in the language iff the undirected graph $G$ has a clique of size at least $k$) is in P, then there is a polynomial-time algorithm that, given any undirected graph $G$, finds a clique of $G$ of maximum size.

6. In the Dominating Set (decision) problem we are given a directed graph $G = (V, E)$ and an integer $k$. We are asking whether there is a set $D$ of $k$ or fewer vertices such that for each $v \notin D$ there is a $u \in D$ with $(u, v) \in E$. Show that Dominating Set is NP-complete. (Start from Vertex Cover, obviously a very similar problem, and make a simple local replacement.)