Due: December 13, 2016, 3:30PM (sharp) (in box)

Unless otherwise stated, for each algorithm you design you should give a detailed description of the idea, proof of correctness, termination, analysis and proof of time and space complexity. If not, your answer will be incomplete and you will miss credit!

1. **[NP Completeness, 15 Points]** Problem 34.5-2, (CLRS 2nd edition page 1017) (CLRS 3rd edition page 1100)

2. **[NP Completeness, 10 Points]** The low degree spanning tree (LDST) problem is as follows. Given a graph $G$ and an integer $k$, does $G$ contain a spanning tree such that all vertices in the tree have degree at most $k$? Prove that the LDST is NP-hard with a reduction from HAM-PATH.\(^1\)

3. **[Approximation Algorithms, 15 Points]** Given a connected, weighted, undirected graph $G = (V, E)$ and a subset $R \subseteq V$ (of "required" vertices), a minimum Steiner tree of $G$ is a tree of minimum weight that contains all the vertices in $R$. (It may or may not contain the remaining vertices).

Finding a minimum Steiner tree is NP-hard in general. Consider a class of approximation algorithms that uses the following heuristic strategy:

(a) Compute the complete distance graph $G_1 = (R, R \times R)$ between vertices in $R$; each edge $(u, v)$ in $G_1$ is weighted with the length of the shortest path from $u$ to $v$ in $G$.

(b) Compute a minimum spanning tree $G_2$ of $G_1$.

(c) Map the graph $G_2$ back into $G$ by substituting for each edge of $G_2$ a corresponding shortest path in $G$. Call the resulting graph $G_3$.

(d) Compute a minimum spanning tree $G_4$ of $G_3$.

(e) Iteratively delete all leaves in $G_4$ that are not vertices in $R$.

Of all the minimum Steiner trees for $G$ and $R$, let $T_{opt}$ be the one with the minimum number of leaves. Let $T_{approx}$ be the Steiner tree obtained using the strategy outlined above. If $w(T)$ denotes the total cost of a tree $T$ (i.e. the sum of all its edge weights), prove that $w(T_{approx}) \leq 2(1 - \frac{1}{l})w(T_{opt})$, where $l$ is the number of leaves in $T_{approx}$.

*Hint:* Consider a clockwise circular traversal of $T_{opt}$; such a traversal will start and end at the same vertex and will go along each edge in $T_{opt}$ exactly twice.

\(^1\)For the purposes of this problem you may assume that HAM-PATH, defined analogously to HAM-CYCLE, is NP-complete.