The fourth assignment covers depth-first search and minimum spanning trees. As usual, check blackboard for any updates and corrections.

1. Rewrite the pseudo-code for depth-first search so as to eliminate recursion, and analyze its running time.

2. Explain how a vertex $u$ of a directed graph $G$ can end up in a depth-first tree containing only $u$ even though $u$ has both incoming and outgoing edges in $G$.

3. Describe a dynamic programming for the weighted interval scheduling problem in which there are $k$ identical resources that can be assigned to interval requests. In your answer, give appropriate pseudo-code, a proof of correctness, and a running time analysis.

4. Let $G=(V, E)$ be an undirected graph with real weights $w: V \rightarrow \mathbb{R}$ associated to each vertex. Define $n=|V|$, $m=|E|$. Let $S \subseteq V$ be a subset of vertices called the source vertices and $T \subseteq V$ be a subset of vertices called the target vertices. A set of target vertices $A \subseteq T$ is said to be admissible if there are $|A|$ vertex-disjoint paths from $S$ onto $A$. For example, a single target vertex $u$ is admissible if there is a path from $S$ to $u$. As another example, two target vertices $u$ and $v$ are admissible if there is a path $P_u$ from $S$ to $u$, a path $P_v$ from $S$ to $v$, and the two paths $P_u$ and $P_v$ contain different vertices. Assume that you have a black box function that returns the $|A|$ disjoint paths if they exist, or false otherwise. The disjoint path function takes $O(m^{3/2})$ time. Describe a greedy algorithm that finds the admissible set of maximum weight. Give appropriate pseudo-code, a running time analysis, and a proof of correctness that uses an exchange argument.

5. Give a linear-time algorithm that takes as input a DAG $G=(V, E)$ and two vertices $s$ and $t$, and returns the number of paths from $s$ to $t$ in $G$. Your algorithm only needs to count the paths, not list them. In your answer, give appropriate pseudo-code, a proof of correctness, and a running time analysis. (Hint: use topological sort.)

6. Let $G=(V, E)$ be a connected undirected graph. Each edge $\{i, j\}$ has a weight $w(i, j)$. An Euclidean minimum spanning tree (EMST) is a spanning tree $T$ of $G$ that minimizes

$$\sqrt{\sum_{\{i, j\} \in T} w^2(i, j)}$$
Describe an algorithm to find the EMST. In your answer, give appropriate pseudo-code, a proof of correctness, and a running time analysis.