CS 317, Assignment 4

Due on 12 November in class

1. Let $G = (V, E)$ be an undirected, connected graph with weight function $w : E \longrightarrow R$. Suppose that $|E| \geq |V|$ and that all the edge weights are distinct.

   Let $\mathcal{T}$ be the set of all spanning trees of G, and let T be the minimum spanning tree of G. Then a *second-best minimum spanning tree* is a spanning tree T' such that $w(T') = \min\{w(T'') : T'' \in \mathcal{T} \setminus \{T\}\}$.

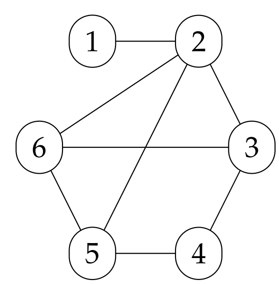
   (a) Show that the minimum spanning tree is unique, but that the second-best minimum spanning tree is not necessarily unique.

   (b) Let T be the minimum spanning tree of G. Prove that G contains some edge $(u, v) \in T$ and some edge $(x, y) \notin T$ such that $T \setminus \{(u, v)\} \cup \{(x, y)\}$ is a second-best minimum spanning tree of G.

2. Let $G = (V, E)$ be an undirected graph. A vertex cover of G is a subset U of V such that every edge in E is incident to at least one vertex in U. A minimum vertex cover is one with the lowest number of vertices. For example $\{1, 3, 5, 6\}$ and $\{2, 3, 5\}$ are both vertex covers for the graph below, with the latter being a minimum vertex cover.

   Consider the following greedy algorithm for the minimum vertex cover problem:

   algorithm VERTEXCOVER($G = (V, E)$)**:**
   $U \leftarrow \emptyset$
   while $E \neq \emptyset$ **do**
   Find a vertex v of maximum degree
   $U \leftarrow U \cup \{v\}$
   $V \leftarrow V \setminus \{v\}$
   $E \leftarrow E \setminus \{(x, y) \in E : x = v \lor y = v\}$
   return U



   (a) Refine the algorithm above by providing detailed descriptions of all the steps using suitable data structures. Justify your choice of data structure and determine the overall running time.

   (b) Give an example (other than the graph above) where this algorithm returns a minimum vertex cover.

   (c) Give an example (other than the graph above) where this algorithm does not return a minimum vertex cover.

   (d) How far from the minimum vertex cover do you think the vertex cover produced by this algorithm is in the worst case? You do not have to provide a proof or tight argument, but you need to offer some justification.

3. You are an advertising company that wants to advertise something to all n people in the city. You know that each of these n people will come downtown on Sunday for some interval of time. You have acquired these time intervals for all people through some unethical means. You do not afford to put ads downtown, but you can pay people to carry your ad on their t-shirts. Assume that if X is carrying the ad, then anyone whose time interval intersects with the time interval of X will see the ad (of course, X will also see the ad). You want to choose the minimum number of people you should pay so that everyone sees the ad. Design an algorithm for this problem. Establish the correctness of the algorithm and analyze its running time.

4. Consider the problem of making change for n cents using the smallest number of coins. Assume that the value of each coin is an integer.

   (a) Describe a greedy algorithm to make change consisting of quarters, dimes, nickels, and pennies. Analyze the running time and prove that your algorithm yields an optimal solution.

   (b) Suppose that the available coins are in denominations that are powers of c that is, the denominations are $c^0, c^1, \ldots, c^k$ for some integers $c > 1$ and $k \geq 1$. Show that the greedy algorithm always yields an optimal solution.

   (c) Give a set of coin denominations for which the greedy algorithm does not yield an optimal solution. Your set should include a penny so that there is a solution for every value of n.

   (d) Give an $O(nk)$-time algorithm that makes change for any set of k different coin denominations using the smallest number of coins, assuming that one of the coins is a penny.

   Hint. Such an algorithm is obviously not going to be a greedy algorithm, so you may want to look at the next algorithm design technique.

Make sure you review the submission guidelines posted on the course's Web site before submitting.