Solutions to Homework Five

CSE 101

1. *Textbook problem 5.1.* The graph has two minimum spanning trees, of cost 19. There are several different orders in which Kruskal might potentially add the edges, for instance:

Edge	one side of cut
$A - E$	$\{A\}$
$E - F$	$\{A, E\}$
$B - F$	$\{A, E, F\}$
$F - G$	$\{A, B, E, F\}$
$G - H$	$\{H\}$
$C - G$	$\{C\}$
$D - G$	$\{D\}$

2. *Textbook problem 5.4.*

   Claim. A graph G with k connected components has at least $|V| - k$ edges.

   Proof. Take any such $G = (V, E)$ and run an iterative trimming procedure on it: "while there is a cycle, remove an edge from the cycle". The result is a forest of k trees; say they contain $n_1, n_2, \ldots, n_k$ nodes respectively (where $n_1 + \cdots + n_k = |V|$). Then they must contain $n_1 - 1, \ldots, n_k - 1$ edges respectively, for a total of $|V| - k$. Therefore G must originally have had at least this many edges. □

3. *Another characterization of trees.*

   Claim. Let G be any undirected graph with n nodes, $n - 1$ edges, and no cycles. Then G is a tree.

   Proof. Suppose G has k connected components, containing $n_1, n_2, \ldots, n_k$ nodes, respectively, where $n_1 + \cdots + n_k = n$. Each component is acyclic and connected and is therefore a tree; hence the component with n_i nodes must have $n_i - 1$ edges, implying that the total number of edges in G is exactly $(n_1 - 1) + (n_2 - 1) + \cdots + (n_k - 1) = n - k$. But we know there are $n - 1$ edges; thus $k = 1$, so G is connected and therefore a tree. □

4. *Another greedy approach to MST.*

   (a) **Claim.** Pick any cycle in a graph, and let e be the heaviest edge in that cycle. Then there is a minimum spanning tree that does not contain e.

   Proof. Call the cycle C. We'll show that for any spanning tree T that contains e, there is another spanning tree T' which doesn't contain e and whose weight is at most that of T.

   Remove e from T; this splits T into two subtrees, T_1 and T_2. Since e crosses the cut (T_1, T_2), the cycle C must contain at least one other edge e' across this cut. Let $T' = T - e + e'$. T' is connected and has $|V| - 1$ edges; therefore it is a spanning tree. And since $w(e) \geq w(e')$, the weight of T' is at most that of T. □

   (b) Let G_t be what remains of the graph after t iterations of the loop. Part (a) tells us that for any t, we have $\text{MST}(G_{t+1}) = \text{MST}(G_t)$, where $\text{MST}(\cdot)$ denotes the cost of the minimum spanning tree. By induction, we therefore have $\text{MST}(G_T) = \text{MST}(G)$, where G_T is the final graph. This G_T has no remaining cycles and is therefore a spanning tree, whereby it must be a minimum spanning tree of G.

   (c) G has a cycle containing $e = \{u, v\}$ if and only if there is a path from u to v in $G - e$.

   - explore$(G - e, u)$.
   - return `visited`$[v]$.

(d) Sorting takes $O(|E| \log |E|)$ and there are $|E|$ iterations of the loop, each of which takes time $O(|V| + |E|)$. Since we're assuming the graph is initially connected, we have $|V| \leq |E| + 1$, so the total time is $O((|V| + |E|)|E|) = O(|E|^2)$.

5. *Updating an MST when an edge weight is increased.* For graph $G = (V, E)$ with edge weights $w(\cdot)$, you already have a minimum spanning tree $T = (V, E')$. Then the weight of an edge e increases. How should T be updated?

   Case 1: $e \notin E'$. Nothing to do.

   Case 2: $e \in E'$. In this case, remove e from T; this divides the tree in two, with vertices V_1 on one side and $V_2 = V - V_1$ on the other. Find the lightest edge (in E) between V_1 and V_2 and add it in. The total time taken is $O(|V| + |E|)$.