Fall 2013 Farach-Colton

CS 513
Solutions to #5

1. The minimum spanning tree is defined a tree which spans the points of a weighted graph and whose weight is minimum. The weight of a tree is defined to be the sum of the weights of the edges. We can redefine the weight of a tree to be the *product* of the weight of the edges. Call the minimum weight spanning tree under this new definition the *Minimum Product Spanning Tree*. Show that if the minimum spanning tree of an n node, m edge graph can be computed in $T(n, m)$ time, then the minimum product spanning tree of the graph can be computed in $O(T(n, m))$.

   Answer: The weight of a tree is $\prod w(e)$. Minimizing $\prod w(e)$ is the same as minimizing $\log \prod w(e)$, since the log function is monotone. But $\log \prod w(e) = \sum \log w(e)$, which means that if we take log of all the edge weights and find a minimum spanning tree with the new weights, we get a minimum product spanning tree under the original weights. This suffices for the algorithm desired, but notice finally that in fact we can find a minimum spanning tree with the original weights and this will always give us a minimum product spanning tree.

2. Consider a positively weighted tree. Such a tree defines a matrix of pairwise distances between all leaves given by their path lengths. If a matrix can be defined by a tree in this manner, then we call it *additive*.

   (a) Show or give a counterexample to the claim that all matrices with non-negative entries are additive.

   Answer: Pick any matrix which is not symmetric or where the diagonal is not all zero. An alternative is to set $D[1, 2] = D[3, 4] = 2$, $D[1, 3] = D[2, 4] = 3$ and $D[1, 4] = D[2, 3] = 5$. Verify for yourself that this combination of distance, even if symmetric, does not fit a tree.

   (b) Show that if a matrix is additive, then it is defined by exactly one tree (note that this claim is clearly false if we allow degree 2 nodes in the tree, so assume that there are no such nodes).

   Proof: By induction. If $n = 2$, then it's trivial. Suppose it's true of all metric with less than n points. Now consider an additive metric on n points. If we remove any point p, the remaining $n - 1$ points have a unique representation as a tree. If all n original points don't have a unique representation, it must be because p can fit into different places in the tree. Now consider the original n points and remove some other point $q \neq p$. The remaining $n - 1$ points have one tree representation, so p can only fit into one spot on the tree. Thus, all n points have a unique tree representation.

   (c) Give an algorithm for constructing the generating tree from an additive matrix.

Answer: Suppose we have three nodes. Then the tree is a star with one internal node and three edges to the three leaves. The three inter-leaf distances give us three (linear) equations with three unknowns, which we can solve in constant time.

Suppose we have already added k nodes into the tree. We want to add the $k+1$st leaf, in time $O(k)$. If we do, we get an $O(n^2)$ algorithm (which is linear in the input size). We will do so by giving a constant time algorithm for determining if the new leaf to be added should split an edge e. Then we just test for each e if we should split e, thus giving $O(k)$ overall.

Suppose we know, for each $e = \{x, y\}$, e_a and e_b, which are two leaves which have x and y on their path (that is, removing e would leave e_a and e_b in different trees). Suppose w.l.o.g. that x is closer to e_a and y is closer to e_b. Finally, suppose know the distant from x to e_a and y to e_b.

Now we can compute the tree induced by e_a, e_b and $k+1$. Let α be the edge length on the edge incident on e_a, and β the edge length incident on e_b. $k+1$ hangs off of e iff $\alpha > D(x, e_a)$ and $\beta > D(y, e_b)$.

If we find the edge e which must be split, we have to introduce two new edges, and update their e_a and e_b and the distance to these leaves. But we can do this in constant time from α, β, $D(x, e_a)$ and $D(y, e_b)$. This completes the algorithm.

3. Consider a positively weighted tree. Such a tree defines a matrix of pairwise distances between all leaves given by the maximum edge in the path betwen them. If a matrix can be defined by a tree in this manner, the we call it *ultrametric*.

   (a) Show or give a counterexample to the claim that all symmetric matrices with non-negative entries and zeros of the diagonal are ultrametric.

   Answer: Pick a 3 by 3 matrix with $D[1,2] = 1, D[1,3] = 2$, and $D[2,3] = 3$. By the next problem, this matrix cannot define an ultrametric.

   (b) Show that if an matrix U is ultrametric, then $\forall i, j, k, U[i,j] \leq \max\{U[i,k], U[j,k]\}$.

   Proof: Let a b and c be any three leaves in a tree. Consider the union of all the edges on all three pairwise paths. Let e be the edge in this set with biggets weight. Removing e from the tree leaves two leaves, say a and b in one subtree, and the other leaf, c in the other subtree. Then $U[a,c] = U[b,c] = W(e) \geq U[a,b]$. Thus the inequality above holds (the above inequality is the same as saying that the maximum of the tree quantities occurs at least twice in the set of three distance, but we have shown that the maximum $W(e)$ does occur at least twice).

   (c) (Extra Credit) Show that if $\forall i, j, k$, $U[i,j] \leq \max\{U[i,k], U[j,k]\}$, then U is ultrametric.

   Proof: We give a constructive proof of this claim showing an algorithm to build the generating tree. First, we extend the notion of an ultrametric matrix to pairwise distances between all nodes of the generating tree. Later, by a simple reduction, we obtain the generating tree under the previous notion. We start with the following useful observation: if $\forall i,j,k \in [n]$, $U[i,j] \leq \max\{U[i,k], U[j,k]\}$, then $\forall i,j,k \in [n]$, at least two of $\{U[i,j], U[i,k], U[j,k]\}$ are equal to $\max\{U[i,j], U[i,k], U[j,k]\}$. Then, it is enough to show that if $\forall i,j,k \in [n]$, at least two of $\{U[i,j], U[i,k], U[j,k]\}$. are equal to $\max\{U[i,j], U[i,k], U[j,k]\}$, then U is ultrametric. We prove this claim by induction in n, the number of rows.

Base case: $n = 3$. First, add the edge e_{21} of weigth $U[2,1]$. Let $U[3,i]$ be the smallest entry of the 3rd row such that $i \neq 3$. Add the edge e_{3i} of weight $U[3,i]$. In order to prove that U is ultrametric it is enough to show that the entries $U[3,j], j \in [3]$ are correct. For $j = i$ and $j = 3$ the entries are correct with respect to the tree built. Let $j \neq 3$ and $j \neq i$. Now, we have two cases, if $U[3,j] = U[3,i]$ then by the hypothesis $U[3,j] = U[3,i] = \max\{U[3,j], U[3,i], U[i,j]\} \geq U[i,j]$ which means that the weight of the edge e_{ij} is at most $U[3,i]$. On the other hand, if $U[3,j] > U[3,i]$, by the hypothesis $U[3,j] = U[i,j] = \max\{U[3,j], U[3,i], U[i,j]\} \geq U[3,i]$ which means that the weight of the edge e_{ij} is bigger than $U[3,i]$, but it is also equal to $U[3,j]$.

Inductive step: Assume that the statement is true for any matrix of size $n-1 \times n-1$. Let U be an $n \times n$ matrix, satisfying the given assumptions. If we delete the nth rows and columns, we obtain an $n-1 \times n-1$ matrix which by induction is ultrametric. Let T_{n-1} be its generating tree. We extend T_{n-1} to a new tree T_n as in the base case as follows. Let $U[n,i]$ be the smallest entry of the n-th row such that $i \neq n$. Obtain T_n by adding the edge e_{ni} of weight $U[n,i]$ to T_{n-1}. In order to prove that U is ultrametric it is enough to show that the entries $U[n,j], j \in [n]$ are correct with respect to T_n. For $j = i$ and $j = n$ the entries are correct. Let $j \neq n$ and $j \neq i$. Now, we have two cases, if $U[n,j] = U[n,i]$ then by the hypothesis $U[n,j] = U[n,i] = \max\{U[n,j], U[n,i], U[i,j]\} \geq U[i,j]$ which means that heaviest weight in the path ij is at most $U[n,i]$. On the other hand, if $U[n,j] > U[n,i]$, by the hypothesis $U[n,j] = U[i,j] = \max\{U[n,j], U[n,i], U[i,j]\} \geq U[n,i]$ which means that the weight of some edge in the path ij is bigger than $U[n,i]$, but it is also equal to $U[n,j]$.

It remains to show that the generating tree under the original ultrametric notion can be obtained from this one. In order to do that, for each internal node v add a new node v' connected by an edge vv' and assign to these edges the minimum edge weight in the tree. Thus, we obtain an n leaves tree which is the generating tree of U.

(d) (Extra Credit) Show that if U is ultrametric, then the minimum spanning tree of the weighted undirected graph defined by U is a generating tree of U.

Answer: The algorithm given in 3c is nothing but Prim's algorithm. Therefore, the tree obtained is an MST of U.