Staff Solutions to In-Class Problems Week 7, Wed.

STAFF NOTE: Trees, Ch. 12.9 ■

STAFF NOTE: First 3 problems should be re-ordered:

1. CP$_t ree_c haracterizations$
2. CP$_m in_w eight_e dge$
3. CP$_b uild_M STs$

■

STAFF NOTE: This problem can become a tedious time-sink. If it takes more than 20 minutes, abort it, and explain the black-white coloring rules that justify the first two parts. ■

Problem 1.
Let G be a 4×4 grid with vertical and horizontal edges between neighboring vertices. Formally,

$$V(G) = [0,3]^2 ::= \{(k, j) \mid 0 \le k, j \le 3\}.$$

Letting $h_{i,j}$ be the horizontal edge $\langle (i, j)—(i+1, j) \rangle$ and $v_{j,i}$ be the vertical edge $\langle (j, i)—(j, i+1) \rangle$ for $i \in [0,2]$, $j \in [0,3]$, the weights of these edges are

$$w(h_{i,j}) ::= \frac{4i + j}{100},$$
$$w(v_{j,i}) ::= 1 + \frac{i + 4j}{100}.$$

(A picture of G would help; you might like to draw one.)

(a) Construct a minimum weight spanning tree (MST) for G by initially selecting the minimum weight edge, and then successively selecting the minimum weight edge that does not create a cycle with the previously selected edges. Stop when the selected edges form a spanning tree of G. (This is Kruskal's MST algorithm.)

For any step in Kruskal'sprocedure, describe a black-white coloring of the graph components so that the edge Kruskal chooses is the minimum weight "gray edge" according to Lemma **??**.

Solution. The edges are in the order that they are constructed by the given algorithm.

Answer: $h_{0,0} h_{0,1} h_{0,2} h_{0,3} h_{1,0} h_{1,1} h_{1,2} h_{1,3} h_{2,0} h_{2,1} h_{2,2} h_{2,3} v_{0,0} v_{0,1} v_{0,2}$

From the text: An edge does not create a cycle iff it connects different components. The edge chosen by Kruskal's algorithm will be the minimum weight gray edge when the components it connects are assigned different colors.

■

(b) Grow an MST for G starting with the tree consisting of the single vertex $(1, 2)$ and successively adding the minimum weight edge with exactly one endpoint in the tree. Stop when the tree spans G. (This is Prim's MST algorithm.) For any step in Prim's procedure, describe a black-white coloring of the graph components so that the edge Prim chooses is the minimum weight "gray edge" according to Lemma **??**.

Solution. Answer: $h_{0,2}h_{1,2}h_{2,2}v_{0,1}h_{0,1}h_{1,1}h_{2,1}v_{0,0}h_{0,0}h_{1,0}h_{2,0}v_{0,2}h_{0,3}h_{1,3}h_{2,3}$

From the text: This is the algorithm that comes from coloring the growing tree white and all the vertices not in the tree black. Then the gray edges are the ones with exactly one endpoint in the tree. ■

(c) Grow an MST for G by treating the vertices (0, 0), (0, 3), (2, 3) as 1-vertex trees and then successively adding, for each tree in parallel, the minimum weight edge among the edges with one endpoint in the tree. Continue as long as there is no edge between two trees, then go back to applying the general gray edge method until the parallel trees merge to form a spanning tree of G. (This is 6.042's parallel MST algorithm.)

Solution. Done in parallel:

T1@(0,0): $h_{0,0}h_{1,0}h_{2,0}v_{0,0}h_{0,1}h_{1,1}h_{2,1}$

T2@(0,3): $h_{0,3}h_{2,3}v_{0,2}h_{0,2}h_{1,2}h_{2,2}v_{0,1}$ (merges with T1)

T3@(2,3): $h_{1,3}$ (merges with T2) ■

(d) Verify that you got the same MST each time.

Solution. They are the same—if no mistake was made. Problem **??** explains why there is a unique MST for any finite connected weighted graph where no two edges have the same weight. ■

STAFF NOTE: Propose a couple of figures that could be added to the text to help make the proof of the "gray edge" Lemma **??** clearer. ■

Problem 2.
Prove that a graph is a tree iff it has a unique path between every two vertices.

STAFF NOTE: Students should be told *not* to look up the proof in the text until they try this on their own. ■

Solution. Theorem **??.??** shows that in a tree there are unique paths between any two vertices.

STAFF NOTE: Since a tree is connected, there is at least one path between every pair of vertices. To show paths are unique:

first proof: Suppose for the purposes of contradiction, that there are two different paths between some pair of vertices. Then there are two distinct paths $\mathbf{p} \neq \mathbf{q}$ between two vertices with minimum total length $|\mathbf{p}| + |\mathbf{q}|$. If these paths shared a vertex, w, other than at the start and end of the paths, then the parts of $\mathbf{p}$ and $\mathbf{q}$ from start to w, or the parts of $\mathbf{p}$ and $\mathbf{q}$ from w to the end, must be distinct paths between the same vertices with total length less than $|\mathbf{p}| + |\mathbf{q}|$, contradicting the minimality of this sum. Therefore, $\mathbf{p}$ and $\mathbf{q}$ are distinct paths with no vertices in common besides their endpoints, and so

$$\mathbf{p}\,\widehat{}\,\text{reverse}(\mathbf{q})$$

is a cycle.

second proof: Beginning at u, let x be the first vertex where the paths diverge, and let y be the next vertex they share. (For example, see Figure 1.) Then there are two paths from x to y with no common

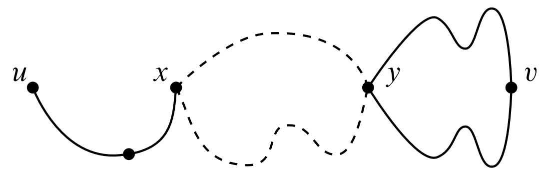

Figure 1 If there are two paths between u and v, the graph must contain a cycle.

edges, which defines a cycle. This is a contradiction, since trees are acyclic. Therefore, there is exactly one path between every pair of vertices.

■

Conversely, suppose we have a graph, G, with unique paths. Now G is connected since there is a path between any two vertices. So we need only show that G has no cycles. But if there was a cycle in G, there are two paths between any two vertices on the cycle (going one way around the cycle or the other way around), a violation of uniqueness. So G cannot have any cycles. ■

Problem 3.
Let G be a weighted graph and suppose there is a unique edge $e \in E(G)$ with smallest weight, that is, $w(e) < w(f)$ for all edges $f \in E(G) - \{e\}$. Prove that any minimum weight spanning tree (MST) of G must include e.

Solution. Suppose to the contrary that e is not included in some MST, T. Since T is a spanning tree, if we add e to T, everything stays connected, so the new graph, $T + e$, is a connected spanning subgraph. Moreover, $T + e$ now has too many edges to be a tree, so the edge e must be on a cycle in $T + e$.

Let f be another edge on this cycle. Now remove f to obtain the graph $(T + e) - f$. Then

1. $(T + e) - f$ is still a connected spanning subgraph of G, because removing an edge, f, on a cycle does not change connectedness.

2. $(T + e) - f$ is a tree, because it is a connected subgraph with the same number of vertices and edges as the tree T.

3. $w((T + e) - f) = w(T) - (w(f) - w(e)) < w(T)$.

Hence $(T + e) - f$ is a spanning tree of G with strictly smaller weight than the MST, T, contradicting the minimality of T.

■

Problem 4.
A simple graph, G, is said to have *width* 1 iff there is a way to list all its vertices so that each vertex is adjacent to at most one vertex that appears earlier in the list. All the graphs mentioned below are assumed to be finite.

(a) Prove that every graph with width one is a forest.

Hint: By induction, removing the last vertex.

Solution. *Proof.* By induction on the number of vertices, n. The induction hypothesis is

$$P(n) ::= \text{all } n\text{-vertex graphs } G \text{ with width one are forests.}$$

Base case: ($n = 1$). A graph with one vertex is acyclic and therefore is a forest.

Induction step. Assume that $P(n)$ is true for some $n \geq 1$ and let G be an $(n+1)$-vertex graph with width one. We need only show that G is acyclic.

The vertices of G can be listed with each vertex adjacent to at most one vertex earlier in the list. Let v be the last vertex in the list. Since *all* the vertices adjacent to v appear earlier in the list, it follows that $\deg(v) \leq 1$.

Now removing a vertex won't increase width, so $G - v$ still has width one, so it is acyclic by Induction Hypothesis. But no degree-one vertex is in a cycle, so adding v back to $G - v$ will not create a cycle. Hence G is acyclic, as claimed.

This proves $P(n+1)$ and completes the induction step.

■

(b) Prove that every finite tree has width one. Conclude that a graph is a forest iff it has width one.

Solution. By induction on the number of vertices, n. The induction hypothesis is

$$Q(n) ::= \text{all } n\text{-vertex trees } T \text{ have width one.}$$

Base case: ($n = 1$). Trivial.

Induction step. Assume that that $Q(n)$ is true for some $n \geq 1$ and let T be an $(n+1)$-vertex tree. We need only show that T has width one.

By Theorem **??**, every tree with at least two vertices has a leaf. Let v be a leaf of T. Then $T - v$ has width one by Induction Hypothesis, so its vertices can be listed with each vertex adjacent to at most one vertex earlier in the list. Since v has degree one, we can add it to the end of the list of vertices for $T - v$ to obtain the required list for T. Hence T has width one, as claimed.

This proves $Q(n+1)$ and completes the induction step.

Note that if all the connected components of a graph have width 1, then so does the whole graph: just append the lists of vertices for each successive component. In particular, since every tree has width one, so does every forest. Therefore by part (a), a graph is a forest iff it has width one. ■