# In-Class Problems Week 10, Mon.

**Problem 1.**
Let $G$ be the $4 \times 4$ grid with vertical and horizontal edges between neighboring vertices and edge weights as shown in Figure 1.

In this problem you will practice some of the ways to build minimum-weight spanning trees. For each part, list the edge weights in the order in which the edges with those weights were chosen by the given rules.

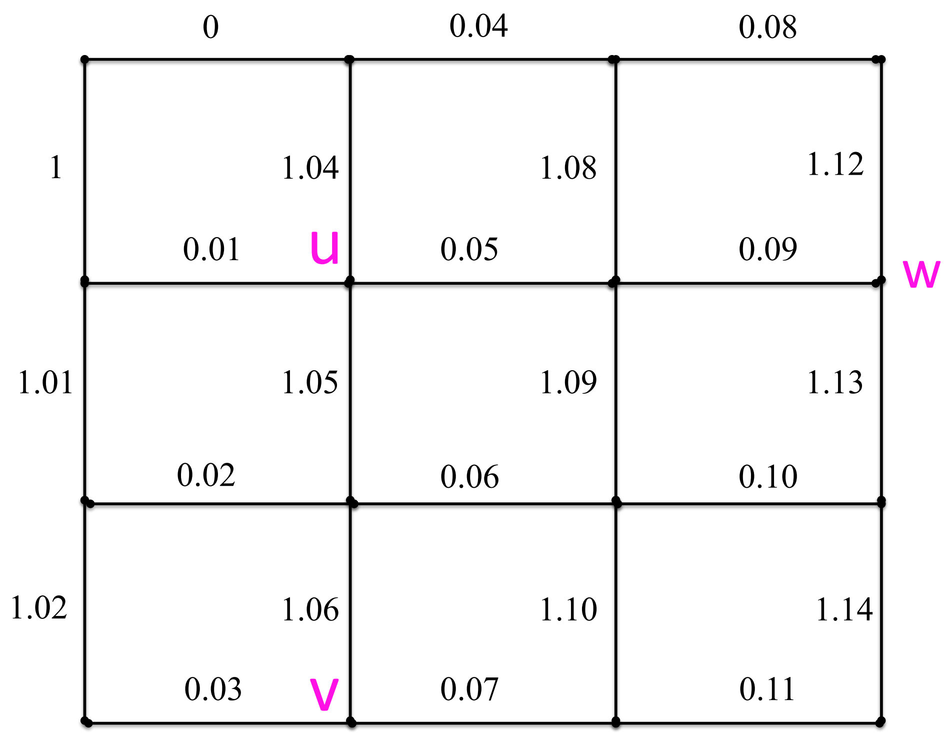


**Figure 1** The 4x4 array graph $G$

**(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's procedure, describe a black-white coloring of the graph components so that the edge Kruskal chooses is the minimum weight "gray edge" according to Lemma 12.11.10.

**(b)** Grow an MST for $G$ by starting with the tree consisting of the single vertex $u$ and then 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".

**(c)** The 6.042 "parallel" MST algorithm can grow an MST for $G$ by starting with the upper left corner vertex along with the vertices labelled $v$ and $w$. Regard each of the three vertices as one-vertex trees. Successively add, for each tree in parallel, the minimum-weight edge among the edges with exactly one endpoint in the tree. Stop working on a tree when there is an edge connecting it to another tree. Continue until there are no more eligible trees—that is, each tree is connected by an edge to another tree—then go back to applying the general gray-edge method until the parallel trees merge to form a spanning tree of $G$.

**(d)** Verify that you got the same MST each time as explained in the text.

**Problem 2.**
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 $e$.

**Problem 3.**
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.

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

**Problem 4.**
Starting with the definition of a tree as a simple graph that is connected and acyclic, prove that a graph is a tree iff it has a unique path between every two vertices.