CS390 Week 3 2.8.2025

Problem 6.5 #1

First, compute the strongly connected components in $O(m+n)$ using the algorithim discussed in class. Within a strongly connected component, all pairs of pairs are by definition half-connected as well, so we may collapse each strongly connected component into a single node.

The resulting graph is directed and acyclic. We claim that a DAG is half connected iff when we run topological sort (i.e. keep deleting source nodes), we never have two or more sources to choose from.

- *half-connected* $\implies$ *always* 1 *source node*
  Suppose by contradiction we have a half-connected graph, but at some point we could choose either s_1 or s_2 as source nodes. Since the topological sort algorithm only deletes in-edges to s_1, none of the previous (deleted) vertices were reachable from s_1. Since s_2 is now a source, any path into s_2 must go through some previous vertex. Therefore, s_1 cannot reach s_2, and by symmetry, s_2 also cannot reach s_1, so the graph is not half-connected.

- *always* 1 *source node* $\implies$ *half-connected*
  Let the vertices in order be $v_1, v_2, \ldots, v_n$. Then for $2 \leqslant i \leqslant n$, v_i only becomes a source node after v_{i-1} is removed, so there must be an edge $v_{i-1} \to v_i$.

  This means we can find a path from any v_i to v_j with $i < j$ by just taking $v_i \to v_{i+1} \to \cdots \to v_{j-1} \to v_j$, which is sufficient for the graph to be half connected.

Therefore, a DAG being half connected is equivalent to having only 1 source node at every iteration of the topological sort, so all we need to do to get the answer is to run topological sort in $O(m+n)$.

Problem 6.5 #2

Note that if G has any cycle, by definition it is not strictly half connected. To determine whether there is a cycle, we can run topological sort—if there is a cycle, the algorithm will not be able to delete those vertices.

On the other hand, if G has no cycle, then whenever vertex s can reach vertex t, then t *cannot* reach s. Therefore, in this case, G being strictly half connected is the same as G being regular half connected. By the analysis on the previous problem, we need only run topological sort in $O(m+n)$.

In summary, the exact condition for G to be strictly half connected is that after running topological sort once, (a) there is only 1 source at any point, and (b) the algorithm successfully deletes all vertices.