Solutions for Chapter 22: Elementary Graph Algorithms

Solution to Exercise 22.1-6

We start by observing that if $a_{ij} = 1$, so that $(i, j) \in E$, then vertex i cannot be a universal sink, for it has an outgoing edge. Thus, if row i contains a 1, then vertex i cannot be a universal sink. This observation also means that if there is a self-loop (i, i), then vertex i is not a universal sink. Now suppose that $a_{ij} = 0$, so that $(i, j) \notin E$, and also that $i \neq j$. Then vertex j cannot be a universal sink, for either its in-degree must be strictly less than $|V| - 1$ or it has a self-loop. Thus if column j contains a 0 in any position other than the diagonal entry (j, j), then vertex j cannot be a universal sink.

Using the above observations, the following procedure returns TRUE if vertex k is a universal sink, and FALSE otherwise. It takes as input a $|V| \times |V|$ adjacency matrix $A = (a_{ij})$.

```
IS-SINK(A, k)
let A be |V| × |V|
for j ← 1 to |V|                ▷ Check for a 1 in row k
    do if a_kj = 1
          then return FALSE
for i ← 1 to |V|                ▷ Check for an off-diagonal 0 in column k
    do if a_ik = 0 and i ≠ k
          then return FALSE
return TRUE
```

Because this procedure runs in $O(V)$ time, we may call it only $O(1)$ times in order to achieve our $O(V)$-time bound for determining whether directed graph G contains a universal sink.

Observe also that a directed graph can have at most one universal sink. This property holds because if vertex j is a universal sink, then we would have $(i, j) \in E$ for all $i \neq j$ and so no other vertex i could be a universal sink.

The following procedure takes an adjacency matrix A as input and returns either a message that there is no universal sink or a message containing the identity of the universal sink. It works by eliminating all but one vertex as a potential universal sink and then checking the remaining candidate vertex by a single call to IS-SINK.

```
UNIVERSAL-SINK(A)
let A be |V| × |V|
i ← j ← 1
while i ≤ |V| and j ≤ |V|
    do if a_ij = 1
          then i ← i + 1
          else j ← j + 1
s ← 0
if i > |V|
  then return "there is no universal sink"
elseif IS-SINK(A, i) = FALSE
  then return "there is no universal sink"
else return i "is a universal sink"
```

UNIVERSAL-SINK walks through the adjacency matrix, starting at the upper left corner and always moving either right or down by one position, depending on whether the current entry a_{ij} it is examining is 0 or 1. It stops once either i or j exceeds $|V|$.

To understand why UNIVERSAL-SINK works, we need to show that after the **while** loop terminates, the only vertex that might be a universal sink is vertex i. The call to IS-SINK then determines whether vertex i is indeed a universal sink.

Let us fix i and j to be values of these variables at the termination of the **while** loop. We claim that every vertex k such that $1 \le k < i$ cannot be a universal sink. That is because the way that i achieved its final value at loop termination was by finding a 1 in each row k for which $1 \le k < i$. As we observed above, any vertex k whose row contains a 1 cannot be a universal sink.

If $i > |V|$ at loop termination, then we have eliminated all vertices from consideration, and so there is no universal sink. If, on the other hand, $i \le |V|$ at loop termination, we need to show that every vertex k such that $i < k \le |V|$ cannot be a universal sink. If $i \le |V|$ at loop termination, then the **while** loop terminated because $j > |V|$. That means that we found a 0 in every column. Recall our earlier observation that if column k contains a 0 in an off-diagonal position, then vertex k cannot be a universal sink. Since we found a 0 in every column, we found a 0 in every column k such that $i < k \le |V|$. Moreover, we never examined any matrix entries in rows greater than i, and so we never examined the diagonal entry in any column k such that $i < k \le |V|$. Therefore, all the 0s that we found in columns k such that $i < k \le |V|$ were off-diagonal. We conclude that every vertex k such that $i < k \le |V|$ cannot be a universal sink.

Thus, we have shown that every vertex less than i and every vertex greater than i cannot be a universal sink. The only remaining possibility is that vertex i might be a universal sink, and the call to IS-SINK checks whether it is.

To see that UNIVERSAL-SINK runs in $O(V)$ time, observe that either i or j is incremented in each iteration of the **while** loop. Thus, the **while** loop makes at most $2|V| - 1$ iterations. Each iteration takes $O(1)$ time, for a total **while** loop time of $O(V)$ and, combined with the $O(V)$-time call to IS-SINK, we get a total running time of $O(V)$.