# Selected Solutions for Chapter 25: All-Pairs Shortest Paths

## Solution to Exercise 25.1-3

The matrix $L^{(0)}$ corresponds to the identity matrix

$$I = \begin{pmatrix} 1 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \end{pmatrix}$$

of regular matrix multiplication. Substitute 0 (the identity for $+$) for $\infty$ (the identity for min), and 1 (the identity for $\cdot$) for 0 (the identity for $+$).

## Solution to Exercise 25.1-5

The all-pairs shortest-paths algorithm in Section 25.1 computes

$$L^{(n-1)} = W^{n-1} = L^{(0)} \cdot W^{n-1} ,$$

where $l_{ij}^{(n-1)} = \delta(i, j)$ and $L^{(0)}$ is the identity matrix. That is, the entry in the $i$th row and $j$th column of the matrix "product" is the shortest-path distance from vertex $i$ to vertex $j$, and row $i$ of the product is the solution to the single-source shortest-paths problem for vertex $i$.

Notice that in a matrix "product" $C = A \cdot B$, the $i$th row of $C$ is the $i$th row of $A$ "multiplied" by $B$. Since all we want is the $i$th row of $C$, we never need more than the $i$th row of $A$.

Thus the solution to the single-source shortest-paths from vertex $i$ is $L_i^{(0)} \cdot W^{n-1}$, where $L_i^{(0)}$ is the $i$th row of $L^{(0)}$—a vector whose $i$th entry is 0 and whose other entries are $\infty$.

Doing the above "multiplications" starting from the left is essentially the same as the BELLMAN-FORD algorithm. The vector corresponds to the $d$ values in BELLMAN-FORD—the shortest-path estimates from the source to each vertex.

- The vector is initially 0 for the source and $\infty$ for all other vertices, the same as the values set up for $d$ by INITIALIZE-SINGLE-SOURCE.

- Each "multiplication" of the current vector by $W$ relaxes all edges just as BELLMAN-FORD does. That is, a distance estimate in the row, say the distance to $v$, is updated to a smaller estimate, if any, formed by adding some $w(u, v)$ to the current estimate of the distance to $u$.
- The relaxation/multiplication is done $n - 1$ times.

---

## Solution to Exercise 25.2-4

With the superscripts, the computation is $d_{ij}^{(k)} = \min\left(d_{ij}^{(k-1)}, d_{ik}^{(k-1)} + d_{kj}^{(k-1)}\right)$. If, having dropped the superscripts, we were to compute and store $d_{ik}$ or $d_{kj}$ before using these values to compute $d_{ij}$, we might be computing one of the following:

$$
\begin{aligned}
d_{ij}^{(k)} &= \min\left(d_{ij}^{(k-1)}, d_{ik}^{(k)} + d_{kj}^{(k-1)}\right) , \\
d_{ij}^{(k)} &= \min\left(d_{ij}^{(k-1)}, d_{ik}^{(k-1)} + d_{kj}^{(k)}\right) , \\
d_{ij}^{(k)} &= \min\left(d_{ij}^{(k-1)}, d_{ik}^{(k)} + d_{kj}^{(k)}\right) .
\end{aligned}
$$

In any of these scenarios, we're computing the weight of a shortest path from $i$ to $j$ with all intermediate vertices in $\{1, 2, \ldots, k\}$. If we use $d_{ik}^{(k)}$, rather than $d_{ik}^{(k-1)}$, in the computation, then we're using a subpath from $i$ to $k$ with all intermediate vertices in $\{1, 2, \ldots, k\}$. But $k$ cannot be an *intermediate* vertex on a shortest path from $i$ to $k$, since otherwise there would be a cycle on this shortest path. Thus, $d_{ik}^{(k)} = d_{ik}^{(k-1)}$. A similar argument applies to show that $d_{kj}^{(k)} = d_{kj}^{(k-1)}$. Hence, we can drop the superscripts in the computation.

---

## Solution to Exercise 25.3-4

It changes shortest paths. Consider the following graph. $V = \{s, x, y, z\}$, and there are 4 edges: $w(s, x) = 2$, $w(x, y) = 2$, $w(s, y) = 5$, and $w(s, z) = -10$. So we'd add 10 to every weight to make $\hat{w}$. With $w$, the shortest path from $s$ to $y$ is $s \to x \to y$, with weight 4. With $\hat{w}$, the shortest path from $s$ to $y$ is $s \to y$, with weight 15. (The path $s \to x \to y$ has weight 24.) The problem is that by just adding the same amount to every edge, you penalize paths with more edges, even if their weights are low.