CS3000: Algorithms & Data — Summer ’23 — Laney Strange

Floyd-Warshall Algorithm

The Floyd-Warshall algorithm is a bottom-up dynamic programming solution to finding the shortest path between every pair of vertices in a directed, weighted graph. It assumes that negative-weight edges may exist, but there are no negative-weight cycles. It accepts as input a an $n \times n$ matrix W. An entry $W[i, j]$ will be $w(i, j)$ if there is an edge from i to j, 0 if $i == j$, and ∞ if $i \neq j$ and there is no edge from i to j

FLOYD-WARSHALL(G, w)

let $D^{(0)}$ be a new $V \times V$ matrix
for $i = 1$ **to** V
    for $j = 1$ **to** V
        if $i == j$
            $d_{ij}^{(0)} = 0$
        elseif $(i, j) \in G.E$
            $d_{ij}^{(0)} = w(i, j)$
        else
            $d_{ij}^{(0)} = \infty$
for $k = 1$ **to** V
    let $D^{(k)}$ be a new $V \times V$ matrix
    for $i = 1$ **to** V
        for $j = 1$ **to** V
            $d_{ij}^{(k)} = \min(d_{ij}^{(k-1)}, d_{ik}^{(k-1)} + d_{kj}^{(k-1)})$
return $D^{(V)}$

We typeset the Floyd-Warshall procedure above with the following LaTeX:

```
\begin{codebox}
\Procname{$\proc{Floyd-Warshall}(G, w)$}
\li let $D^{(0)}$ be a new $V \times V$ matrix
\li \For $i \gets 1 \To V$
\Do
\li \For $j \gets 1 \To V$
\Do
\li \If $i == j$
\Then
\li $d_{ij}^{(0)} = 0$
\li \ElseIf $(i,j) \in G.E$
\Then
\li $d_{ij}^{(0)} = w(i,j)$
\li \Else
\li $d_{ij}^{(0)} = \infty$
\End
```

```
\End
\End
\li \For $k \gets 1 \To V$
\Do
\li let $D^{(k)}$ be a new $V \times V$ matrix
\li \For $i\gets 1 \To V$
\Do
\li \For $j \gets 1 \To V$
\Do
\li $d_{ij}^{(k)} \gets$ min$(d_{ij}^{(k-1)}, d_{ik}^{(k-1)} + d_{kj}^{(k-1)})$
\End
\End
\End
\li \Return $D^{(V)}$
\end{codebox}
```