1 Dijkstra’s Algorithm (DA)

Given or Input:
(i) A directed graph $G = (V, A)$; a distinguished vertex, $s \in V$,
(ii) non-negative real valued costs $c_{ij} \geq 0$ for all $(i, j) \in A$.

Want or Output:
a shortest directed path from $s$ to $t$ and its length.

NOTATIONS:

Step 0 (Initialization)
(i) Set

\[
\nu(i) = \begin{cases} 
    c_{s,i} & \text{if arc } (s, i) \in A \\
    0 & \text{if } i = s \\
    +\infty & \text{otherwise}
\end{cases}
\]

(ii) Make source vertex, $s$, as permanently labelled, while others are temporarily labelled.

Main Step If all vertices are permanently labelled, STOP. Otherwise, choose vertex $\hat{i}$, a temporarily labelled vertex whose $\nu(\hat{i})$ value is the smallest among all temporarily labelled vertices, i.e. vertex $\hat{i}$ is a temporarily labelled vertex such that:

\[
\nu(\hat{i}) = \min\{\nu(i) : \text{vertex } i \text{ is temporarily labelled vertex}\}
\]

Make $\hat{i}$ permanently labelled
Go to Update Step

Update Step If all vertices are permanently labelled, STOP. Otherwise, update only the temporarily labelled vertices, $i$, in the following way:

\[
\nu(i) \leftarrow \min\{\nu(i), \nu(\hat{i}) + c_{\hat{i}, i}\}
\]

Repeat Main Step
2 Floyd-Warshall’s Algorithm (FW)

Given or Input:
(i) A directed graph $G = (V, A)$;
(ii) real valued costs $c_{ij}$ for all $(i, j) \in A$,
(iii) with NO negative weight directed cycles

Want or Output:
shortest directed paths from all vertices to all vertices, and their lengths.

NOTATIONS:

Step 0 (Initialization)
(i) Set

$$
\nu^0(i, j) \leftarrow \begin{cases} 
0 & \text{if } i = j \\
\min \{ c_{ij}, \infty \} & \text{if arc } (i, j) \notin A
\end{cases}
$$

Main Step (Triple “For” Loops)
For $m = 1, 2, 3, \ldots, |V|$, do:

$$
\begin{align*}
\nu^m(i, j) & \leftarrow \min \{ \nu^{m-1}(i, j), \nu^{m-1}(i, m) + \nu^{m-1}(m, j) \} \\
\text{If } \nu^m(i, i) & < 0, \text{ then STOP; } \exists \text{ a negative weight directed cycle}
\end{align*}
$$