Shortest Path - Continued
Dijkstra's Algorithm

\[
\text{prec} = \text{array indexed by vertices initialized to NIL}
\]
\[
\text{dist} = \text{array indexed by vertices initialized to } \infty
\]
\[
Q = \text{priority queue of vertices indexed by dist[\cdot]}
\]
\[
\text{dist}[s] = 0
\]

For each \( v \): \( Q.\text{insert}(v) \)

while \( Q \) is not empty

\[
v = Q.\text{deletemin}()
\]

for each \( w \) neighbor of \( v \):

if \( \text{dist}[w] > \text{dist}[v] + \ell(v, w) \):

\[
\text{dist}[w] = \text{dist}[v] + \ell(v, w)
\]

\( Q.\text{decreasekey}(w) \)

\( \text{prec}[w] = v \)
\[Q = \{1, \ldots, 4\}\]
\[\text{dist} = [0, 3, 1, 4, 4]\]
\[S = \{A, B, C, D\}\]
\[\text{prec} = [1, 1, 1, 1]\]
Properties:

At the end of each iteration:

A. nodes outside Q have \( \text{dist}[v] = \text{length of shortest path from } s \text{ to } v \)

B. every node has \( \text{dist}[v] = \text{length of shortest path from } s \text{ to } v \) that uses only nodes not in Q as intermediate steps

First step:

\[ \text{dist}[s] = 0 \]
\[ \text{dist}[v] = \ell(s,v) \quad \text{if exists} \]
\[ = \infty \quad \text{if no edge from } s \text{ to } v \]
Prop. A is true at step $t+1$

Take any path $s \rightarrow v$

1. If path has only nodes not in $Q$ as intermediate steps, length $\geq \text{dist}[v]$ [at step $t$]

2. If path passes through some vertex $w$ in $Q$, part of path $s \rightarrow w$ only has vertices in $Q$ as intermediate steps. Part of path $s \rightarrow w$ is of length $\geq \text{dist}[w]$

\[ \text{dist}[w] \geq \text{dist}[v] \]

Length of path from $s \rightarrow v$

$\geq$ part of path from $s$ to $w$

$\geq \text{dist}[w] \geq \text{dist}[v]$ [from B at step $t$]
Prop B is true at step $t+1$

take any path from $s$ to $w$ that has only nodes $\overline{Q}$ as intermediate steps

(1) if path does not contain $v$ length path $\geq \text{dist}[w]$

by inductive assumption

(2) if path uses $v$ not in 2nd-to-last step it is not shortest

(3) if path uses $v$ as the second-to-last step

$\text{dist}[w] \leq \text{dist}[v] + l(v,w)$

$\leq$ length of path
Negative weight edges

Suppose $G$ is a directed weighted graph with no negative cycle. Then for every $s, t$ if there is a path from $s$ to $t$ then there is a shortest path. Take any path $s \rightarrow \cdots \rightarrow t$. Suppose some vertex is repeated.

Remove $v \rightarrow v$ cycle. The path of length $\leq$ before every path $\geq$ shortest path with no repeating vertices.
Bellman-Ford

\[ \text{dist} = \text{array indexed by } V \text{ initialized to } \infty \]
\[ \text{prec} = \text{array indexed by } V \text{ initialized to } 1 \]

\[ \text{dist}[S] = 0 \]

for \( l = 1 \) to \( |V| - 1 \):

\[ \left( \|V\| - 2 \right) \sum_{v \in V} \text{indegree}(v) \]

for each \( v \) in \( V \) - \{S\}:

\[ \text{for each edge } (u,v) : \]

if \( \text{dist}[u] + \text{length}(u,v) < \text{dist}[v] \):

\[ \text{dist}[v] = \text{dist}[u] + \text{length}(u,v) \]

\[ \text{prec}[v] = u \]

Running time \( O(|V| \cdot |E|) \)

Correctness: At step \( l \) of outer for:

for every \( v \)

\[ \text{dist}[v] \leq \text{length of shortest path from } S \text{ to } v \text{ that uses } \leq l \text{ edges} \]