Bellman-Ford

Given a weighted graph $G(V, E)$ with a source vertex $s$, we can solve the single-source shortest paths problem using Bellman-Ford.

Previously, in 6.006 recitation:

- **$w(u, v)$** - Weight of the edge $(u, v)$
- **$\delta(s, v)$** - Weight of the shortest path from $s$ to $v$
- **$v.d$** - Estimate of the weight of the shortest path from $s$ to $v$. Goal is $v.d = \delta(s, v)$
- **$v.\pi$** - Pointer to the parent vertex of $v$ in the shortest path from $s$ to $v$
- **Relaxing** an edge $(u, v)$ updates $v.d$ if $u.d + w(u, v)$ is less than $v.d$.

The Bellman-Ford algorithm can be described in three steps:

1. **Initialize**: For all $v$, set $v.d = \infty$, $v.\pi = \text{NIL}$. Set $s.d = 0$

2. **Relax**: Relax every edge in $G$. Repeat for a total of $|V| - 1$ times

3. **Detect Negative Cycles**: Relax every edge in $G$ one more time. If no vertices were updated with a smaller $v.d$ value, then we are done and $v.d = \delta(s, v)$. If at least one vertex was updated, then a negative weight cycle must exist and the $v.d$ values are not necessarily correct. (Optional: Find the negative weight cycle and mark all the vertices on it and reachable from it to have $v.d = -\infty$)

Initialization takes $O(V)$ time, relaxation takes $O(E(V - 1)) = O(VE)$ time, and detecting negative cycles takes $O(E)$ time. Overall, the runtime of Bellman-Ford is $O(VE)$. There is a $O(VE)$ algorithm that corrects $v.d$ in the case of negative weight cycles (see lecture notes), so even with the optional step, the runtime remains $O(VE)$. 