Graph Theory
Shortest Paths II
Bellman-Ford algorithm

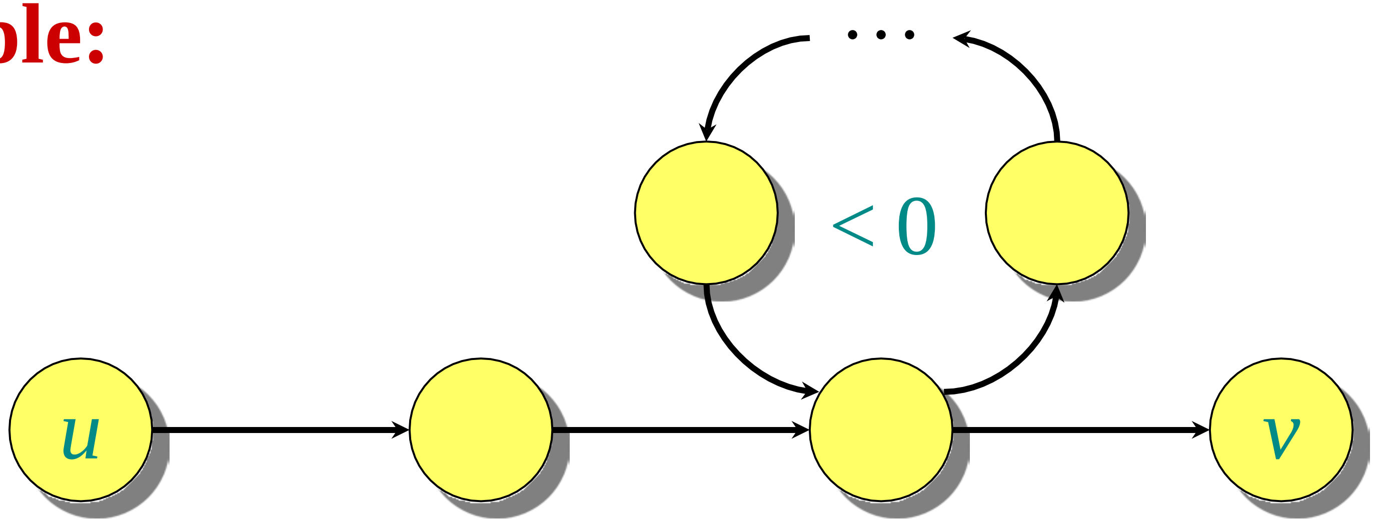
Design and Analysis of Algorithms

Fall 2021

NEGATIVE-WEIGHT CYCLES

Recall: If a graph $G = (V, E)$ contains a negative-weight cycle, then some shortest paths may not exist.

Example:

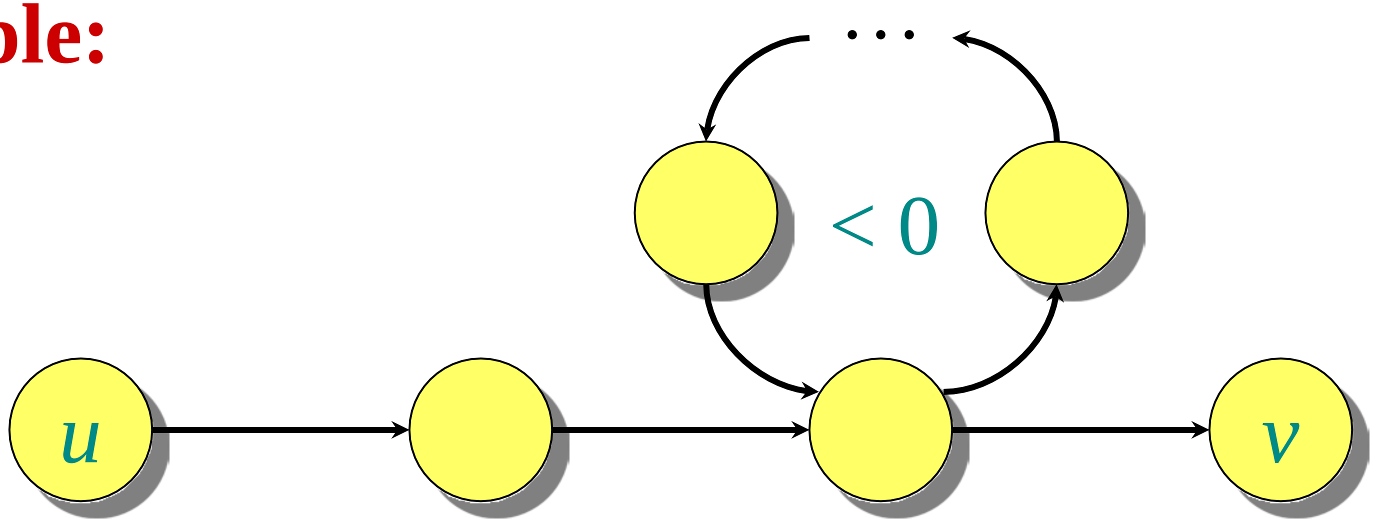


Copyright © 2001-5 by Erik D. Demaine and Charles E. Leiserson

NEGATIVE-WEIGHT CYCLES

Recall: If a graph $G = (V, E)$ contains a negative-weight cycle, then some shortest paths may not exist.

Example:



Bellman-Ford algorithm: Finds all shortest-path lengths from a ***source*** $s \in V$ to all $v \in V$ or determines that a negative-weight cycle exists.

Copyright © 2001-5 by Erik D. Demaine and Charles E. Leiserson

Bellman-Ford algorithm

$d[s] \leftarrow 0$
for each $v \in V - \{s\}$
  do $d[v] \leftarrow \infty$

initialization

for $i \leftarrow 1$ **to** $|V| - 1$
  do for each edge $(u, v) \in E$
    do if $d[v] > d[u] + w(u, v)$
      then $d[v] \leftarrow d[u] + w(u, v)$

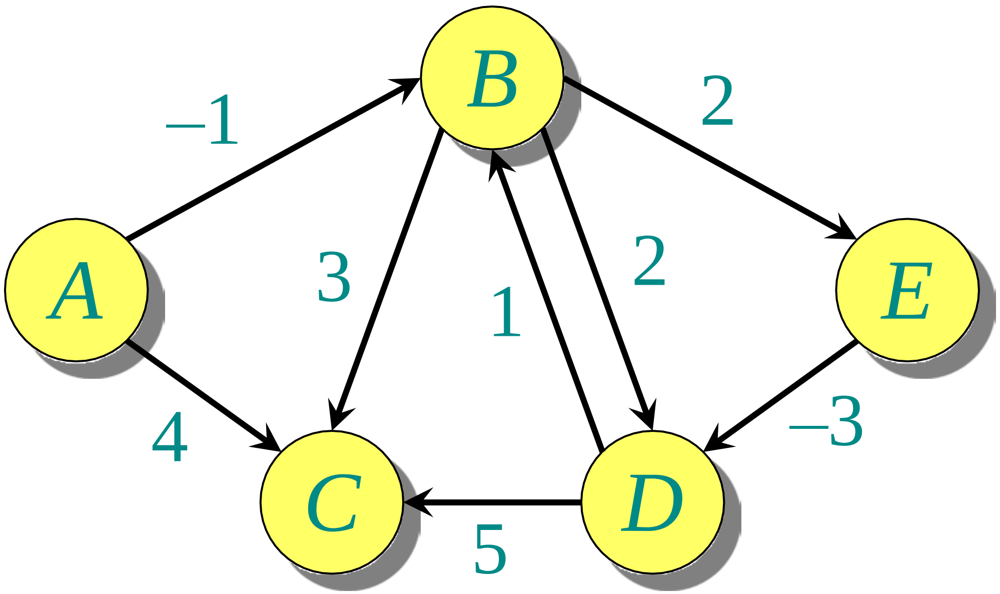
relaxation step

for each edge $(u, v) \in E$
  do if $d[v] > d[u] + w(u, v)$
    then report that a negative-weight cycle exists

At the end, $d[v] = \delta(s, v)$, if no negative-weight cycles.
Time $= O(VE)$.

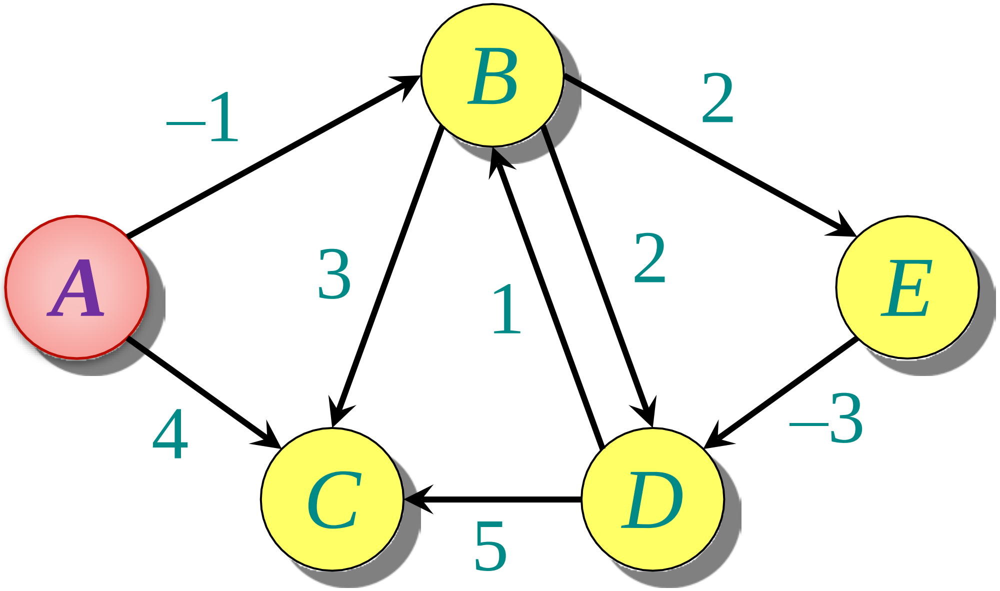
Copyright © 2001-5 by Erik D. Demaine and Charles E. Leiserson

Example of Bellman-Ford



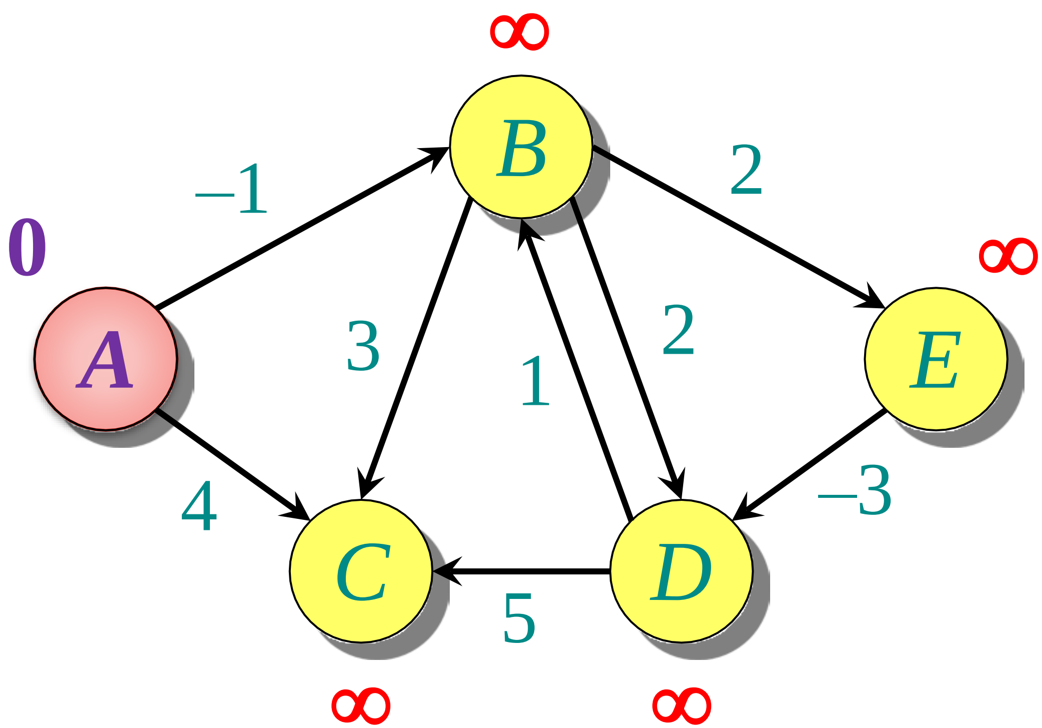
Copyright © 2001-5 by Erik D. Demaine and Charles E. Leiserson

Example of Bellman-Ford



Copyright © 2001-5 by Erik D. Demaine and Charles E. Leiserson

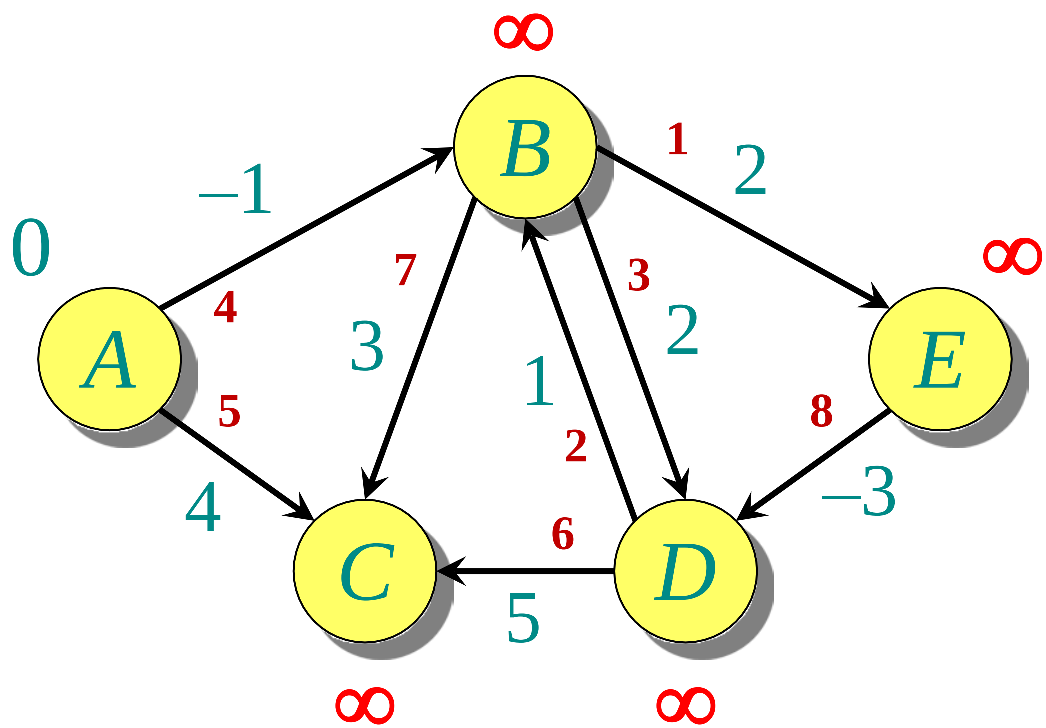
EXAMPLE OF BELLMAN-FORD



Initialization.

Copyright © 2001-5 by Erik D. Demaine and Charles E. Leiserson

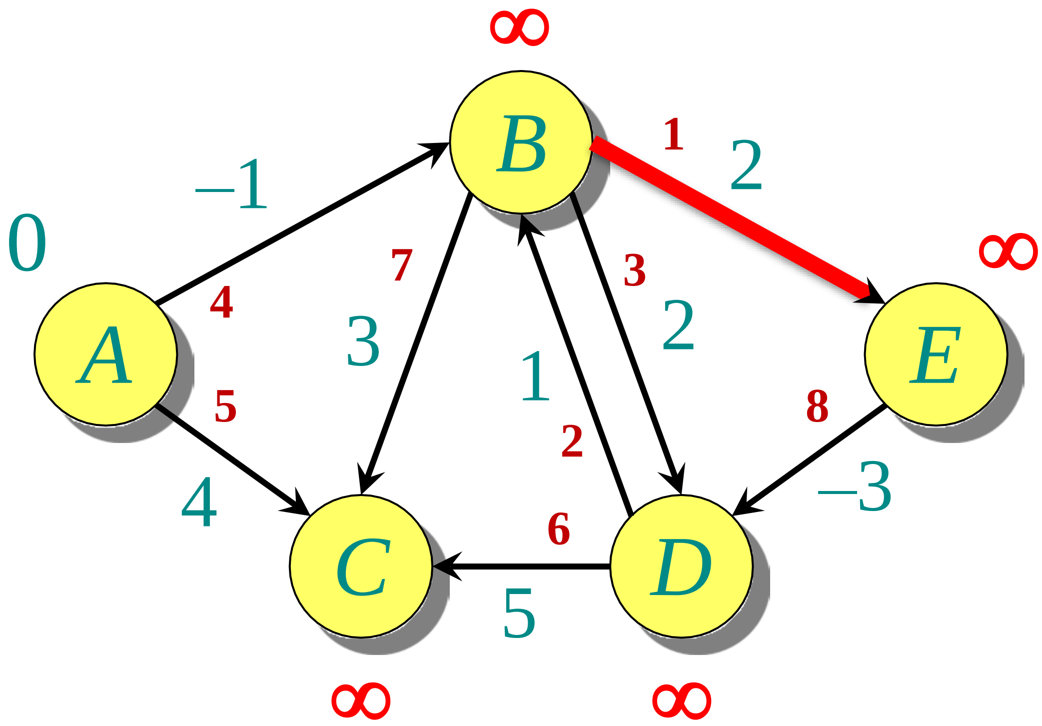
Example of Bellman-Ford



Note: Red number with edges is showing order of edge relation

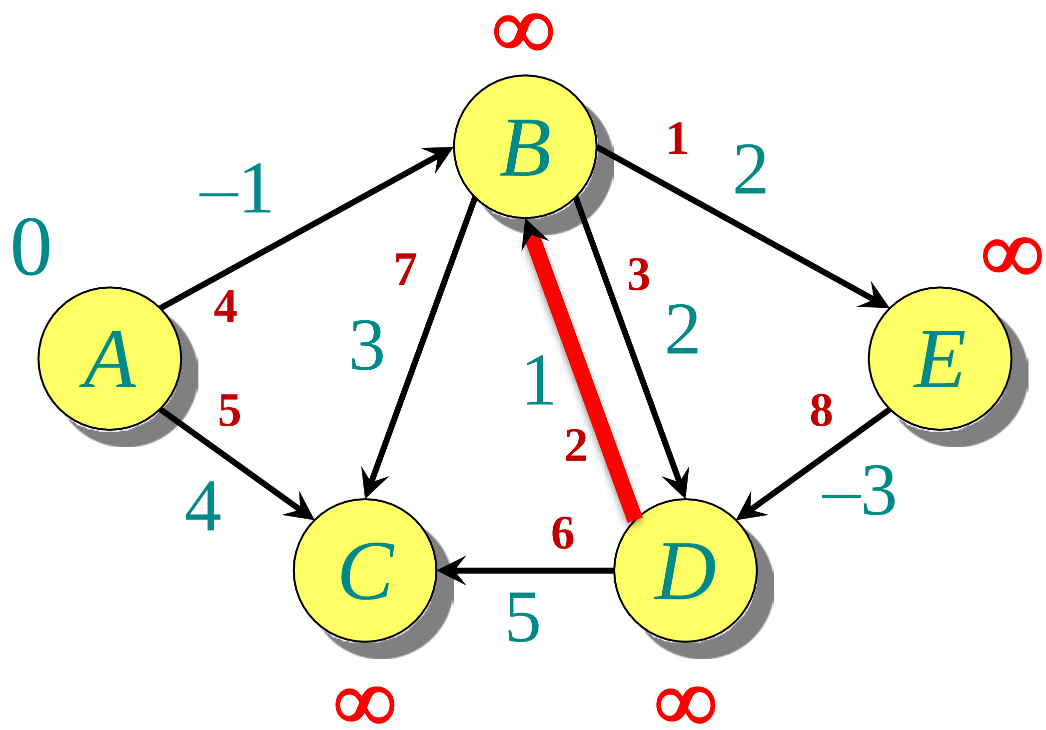
Copyright © 2001-5 by Erik D. Demaine and Charles E. Leiserson

Example of Bellman-Ford



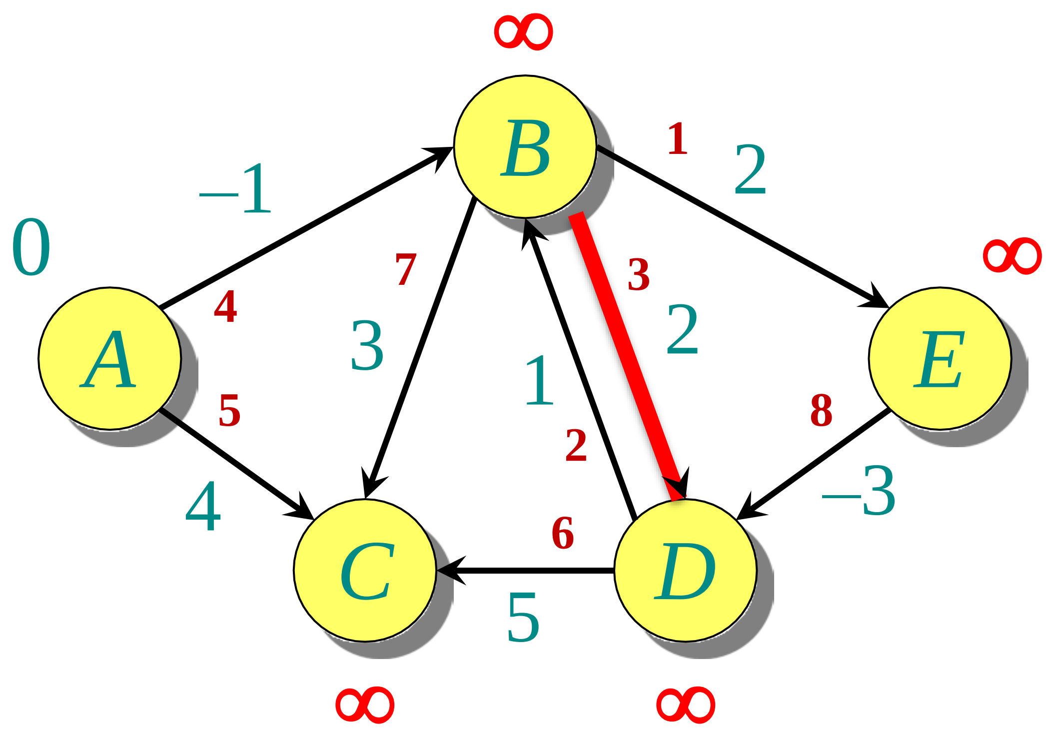
Copyright © 2001-5 by Erik D. Demaine and Charles E. Leiserson

Example of Bellman-Ford



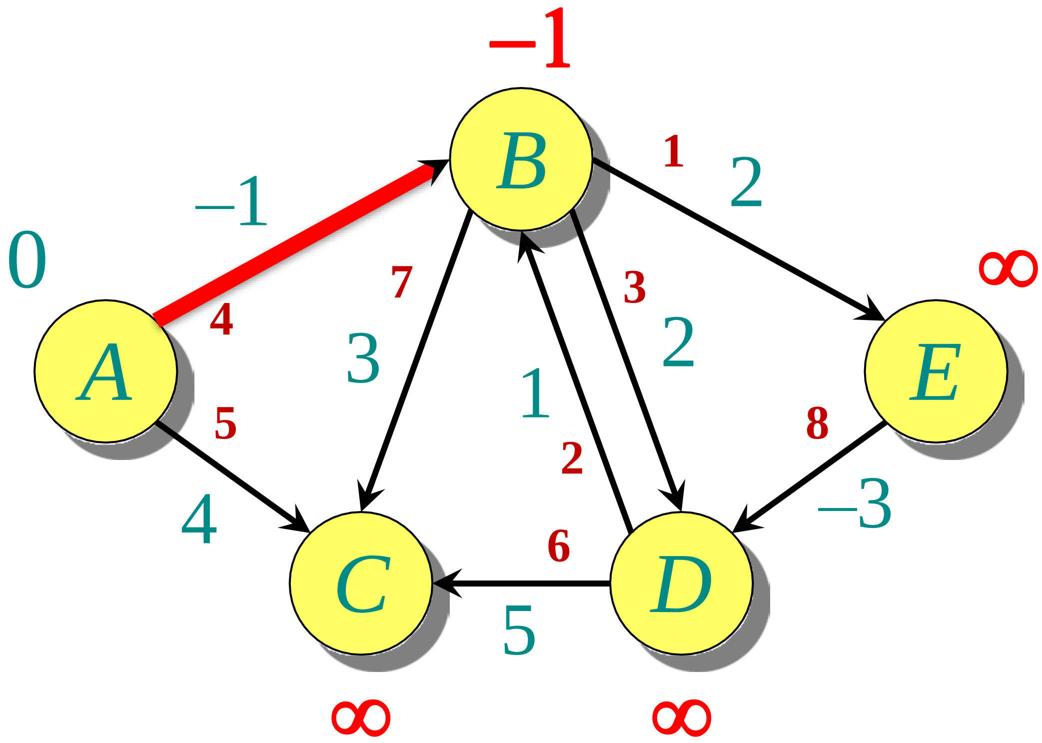
Copyright © 2001-5 by Erik D. Demaine and Charles E. Leiserson

Example of Bellman-Ford



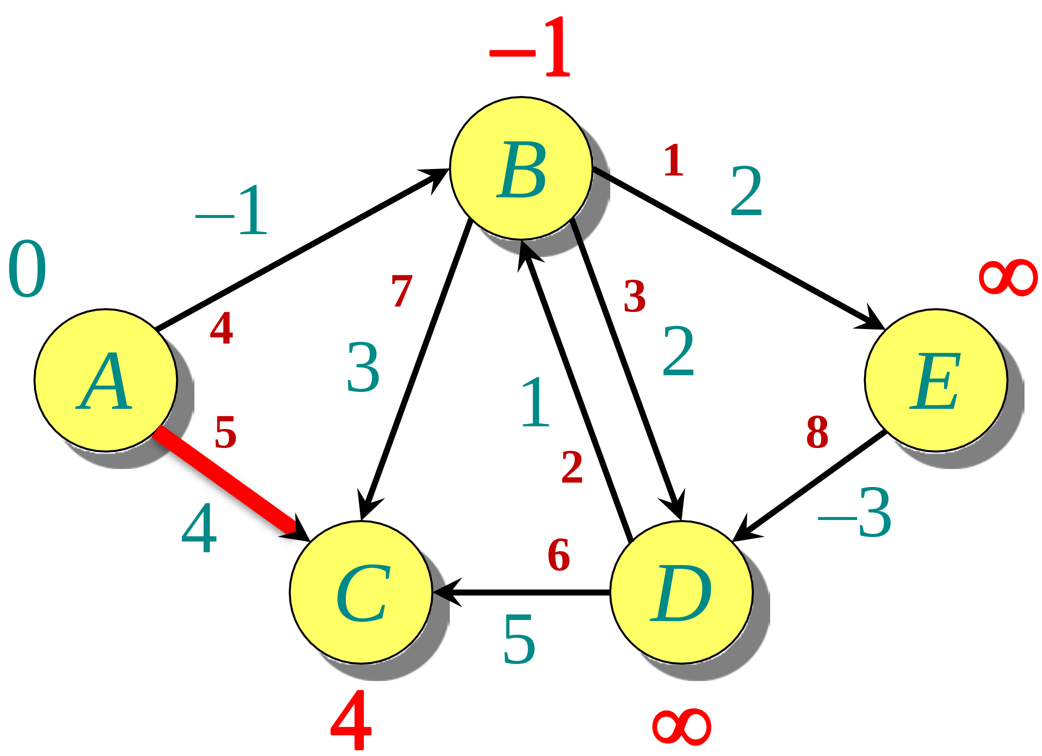
Copyright © 2001-5 by Erik D. Demaine and Charles E. Leiserson

Example of Bellman-Ford



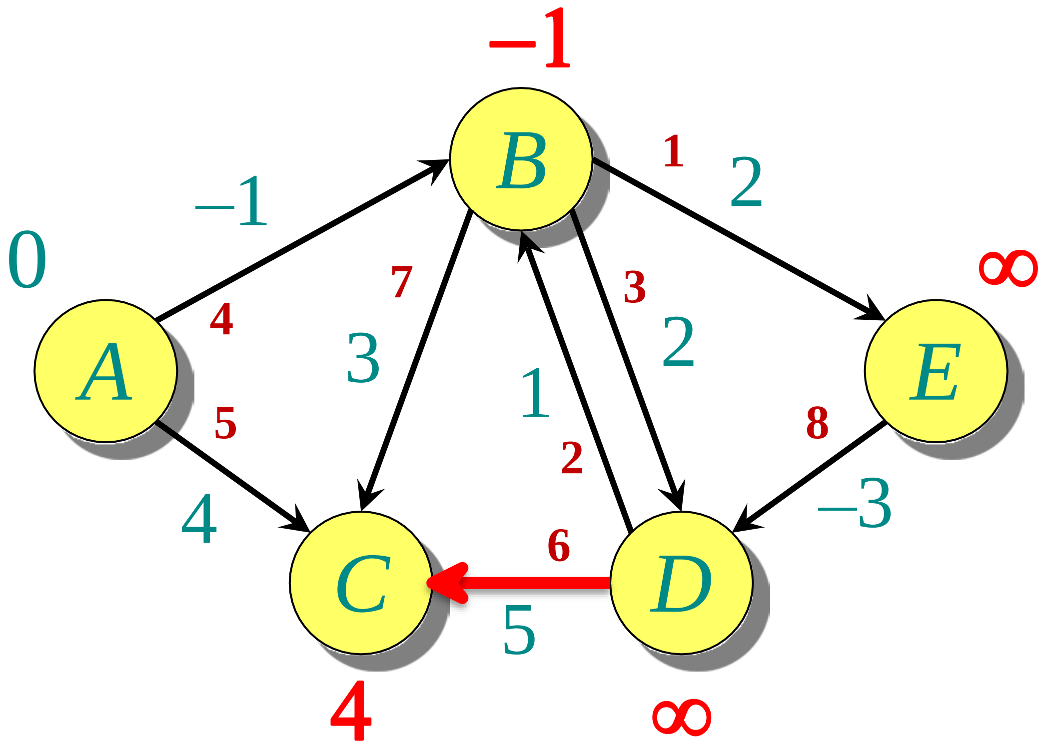
Copyright © 2001-5 by Erik D. Demaine and Charles E. Leiserson

Example of Bellman-Ford



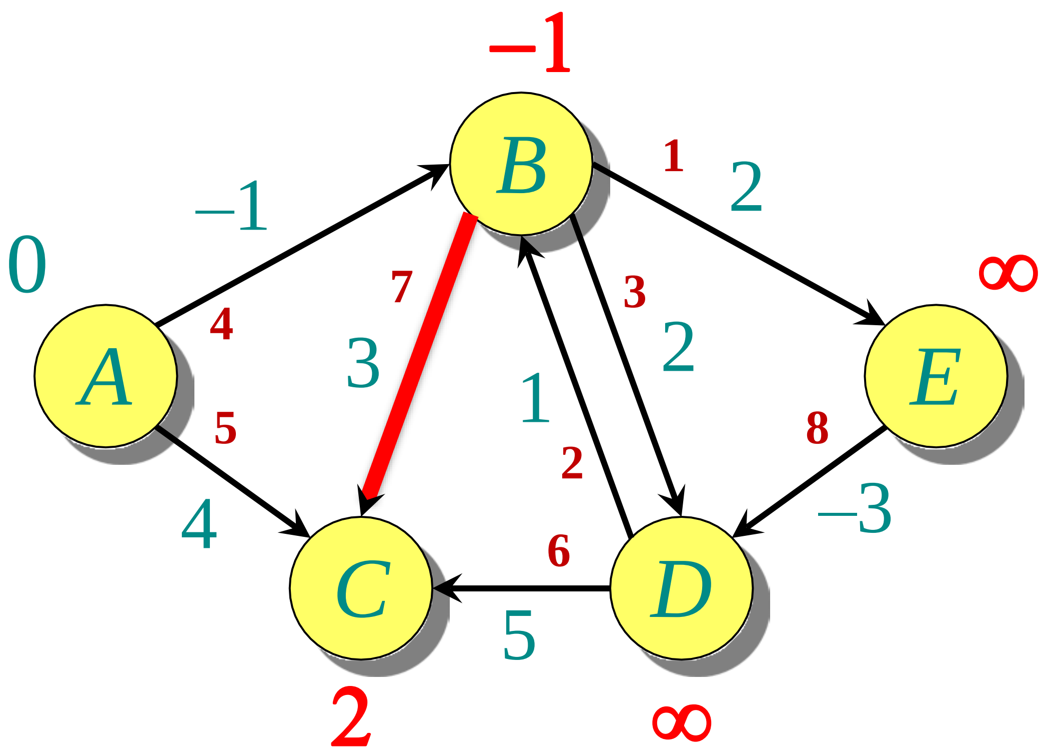
Copyright © 2001-5 by Erik D. Demaine and Charles E. Leiserson

Example of Bellman-Ford



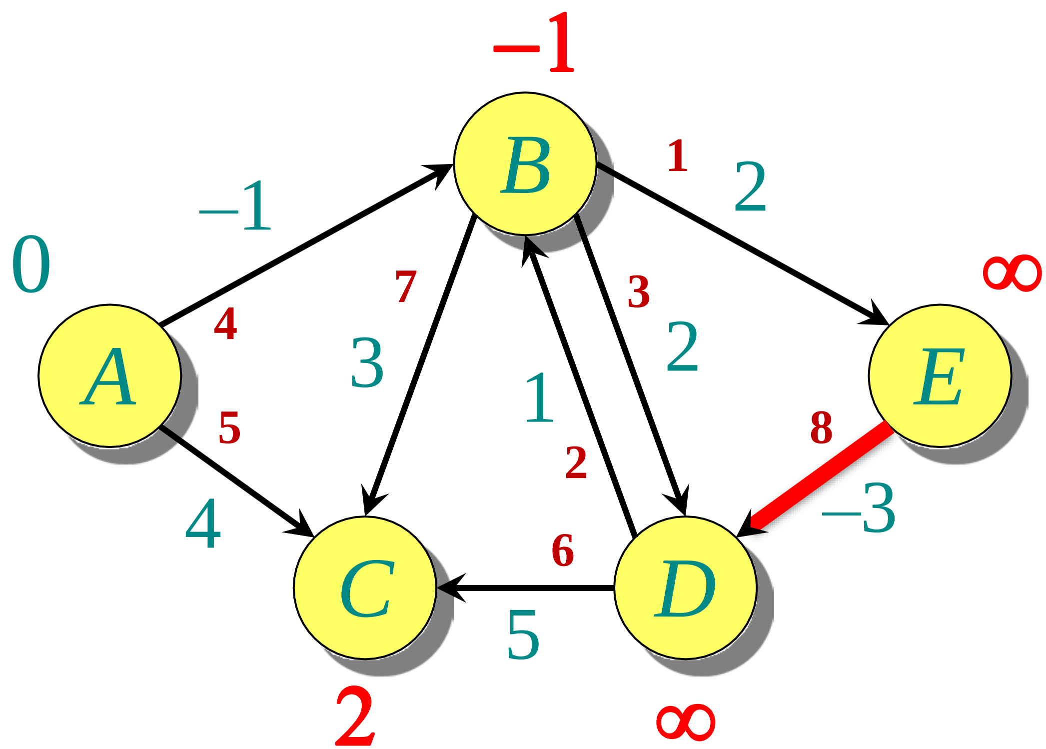
Copyright © 2001-5 by Erik D. Demaine and Charles E. Leiserson

EXAMPLE OF BELLMAN-FORD



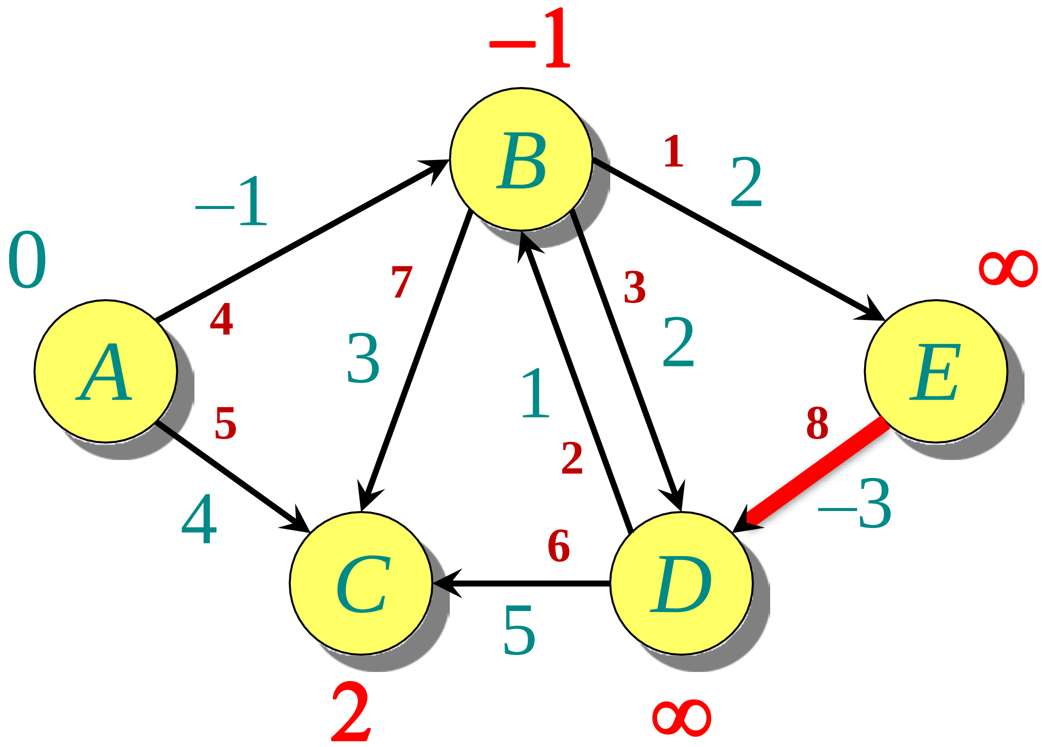
Copyright © 2001-5 by Erik D. Demaine and Charles E. Leiserson

Example of Bellman-Ford



Copyright © 2001-5 by Erik D. Demaine and Charles E. Leiserson

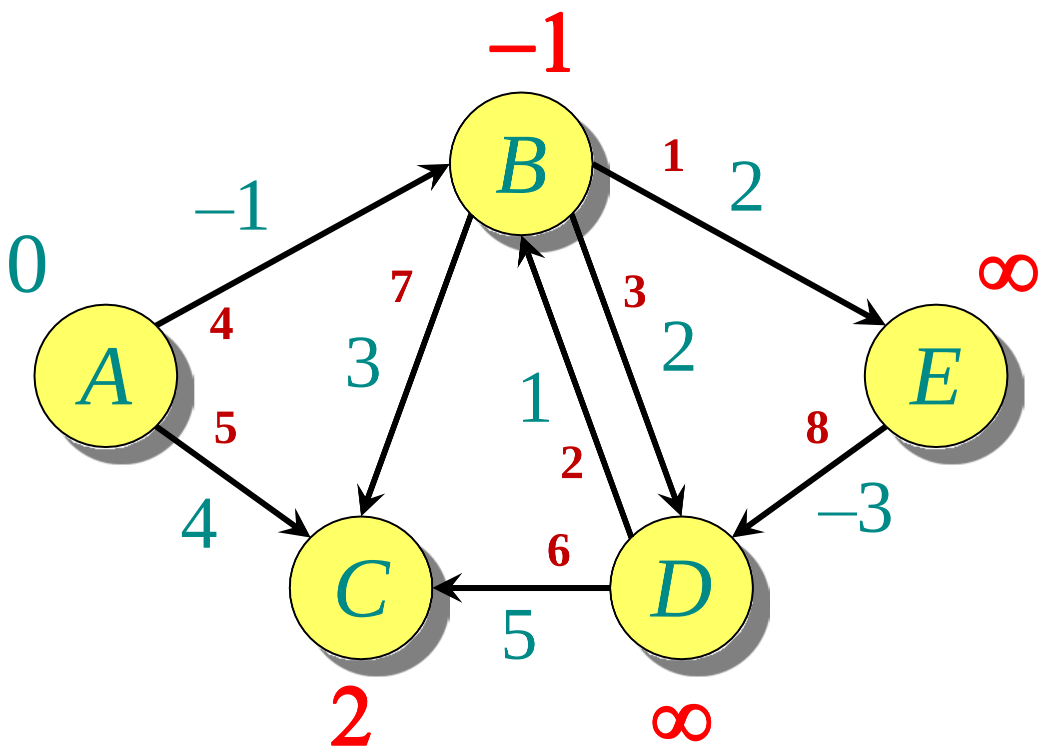
Example of Bellman-Ford



End of Pass 1

Copyright © 2001-5 by Erik D. Demaine and Charles E. Leiserson

Example of Bellman-Ford

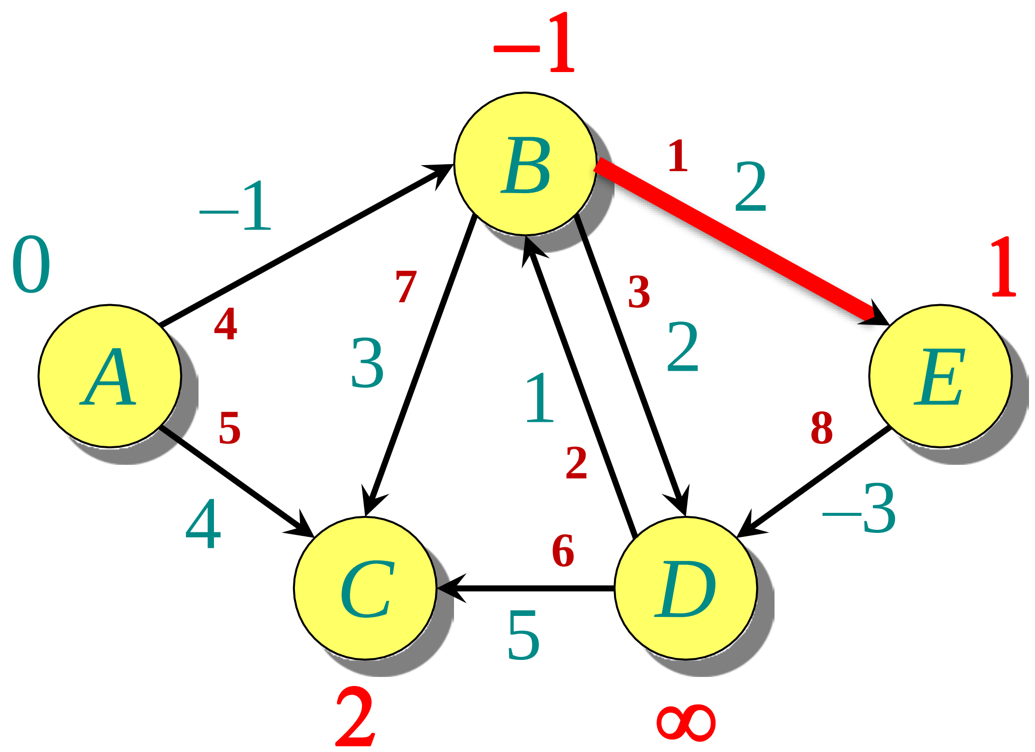


End of Pass 1

•

Copyright © 2001-5 by Erik D. Demaine and Charles E. Leiserson

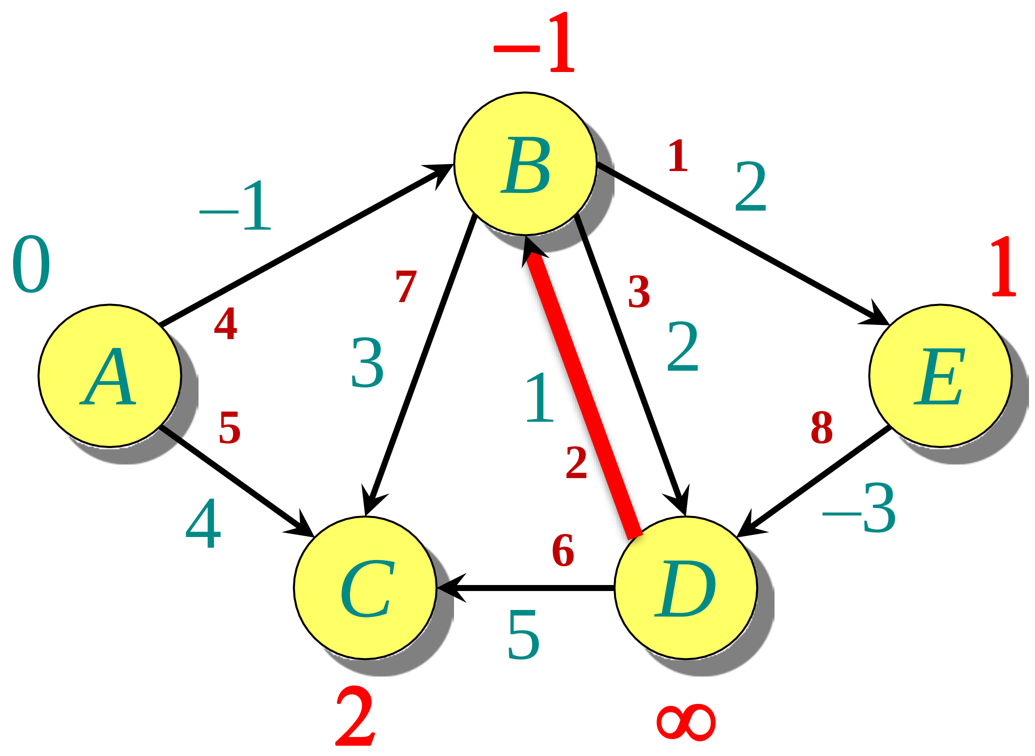
EXAMPLE OF BELLMAN-FORD-PASS2



Note: Red number with edges is showing order of edge relation

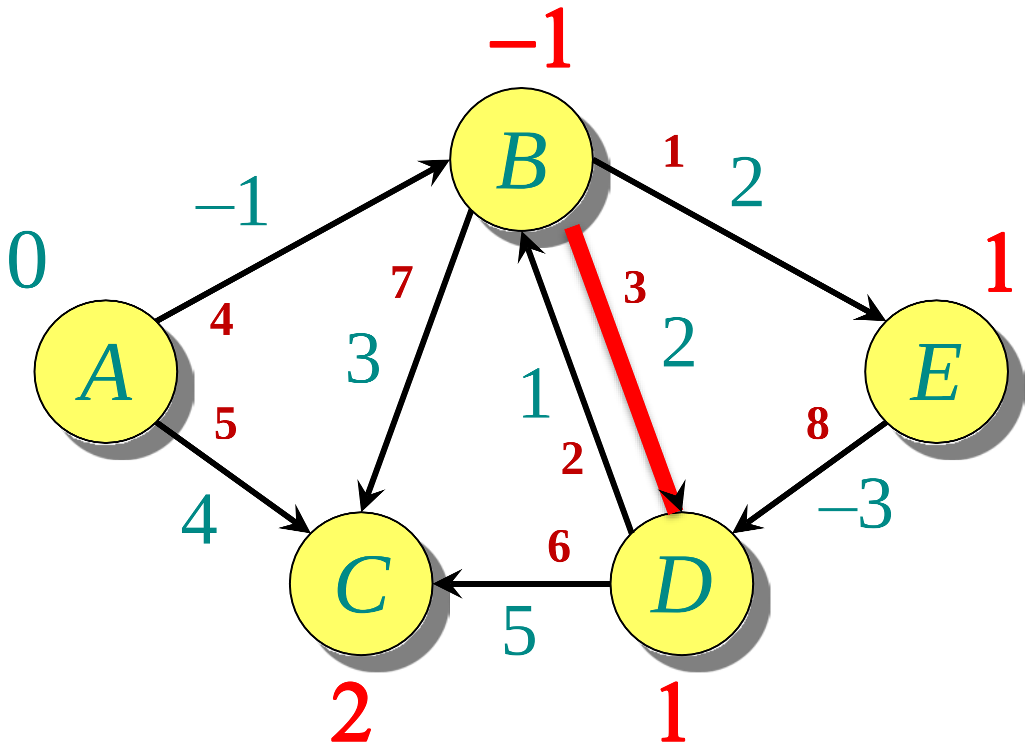
Copyright © 2001-5 by Erik D. Demaine and Charles E. Leiserson

Example of Bellman-Ford



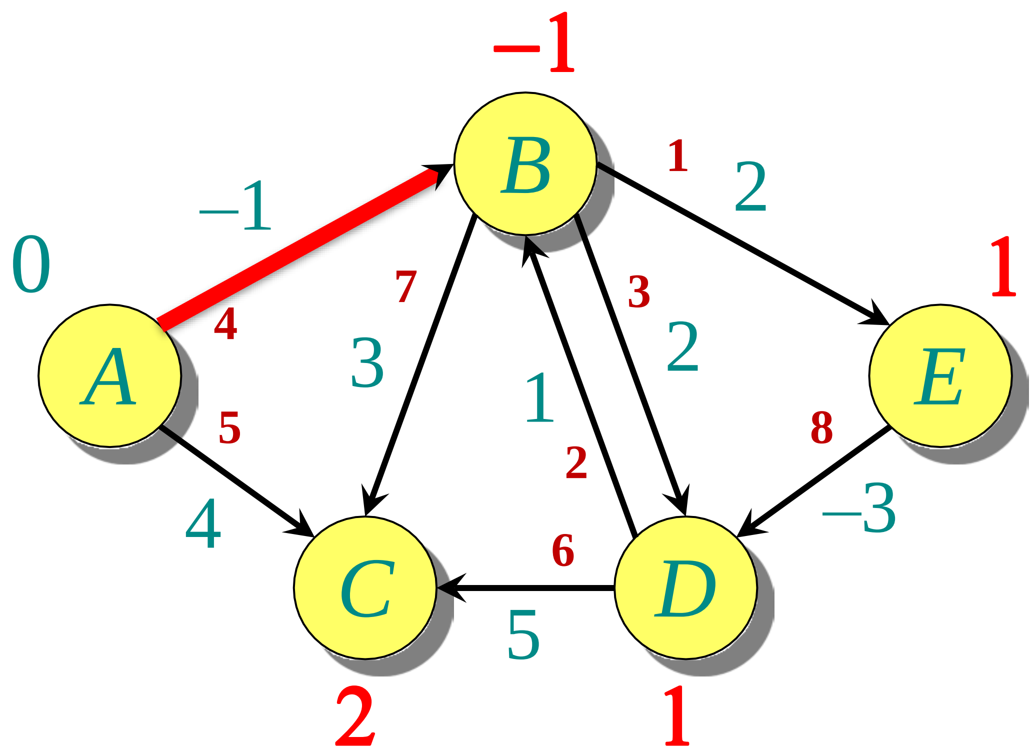
Copyright © 2001-5 by Erik D. Demaine and Charles E. Leiserson

EXAMPLE OF BELLMAN-FORD



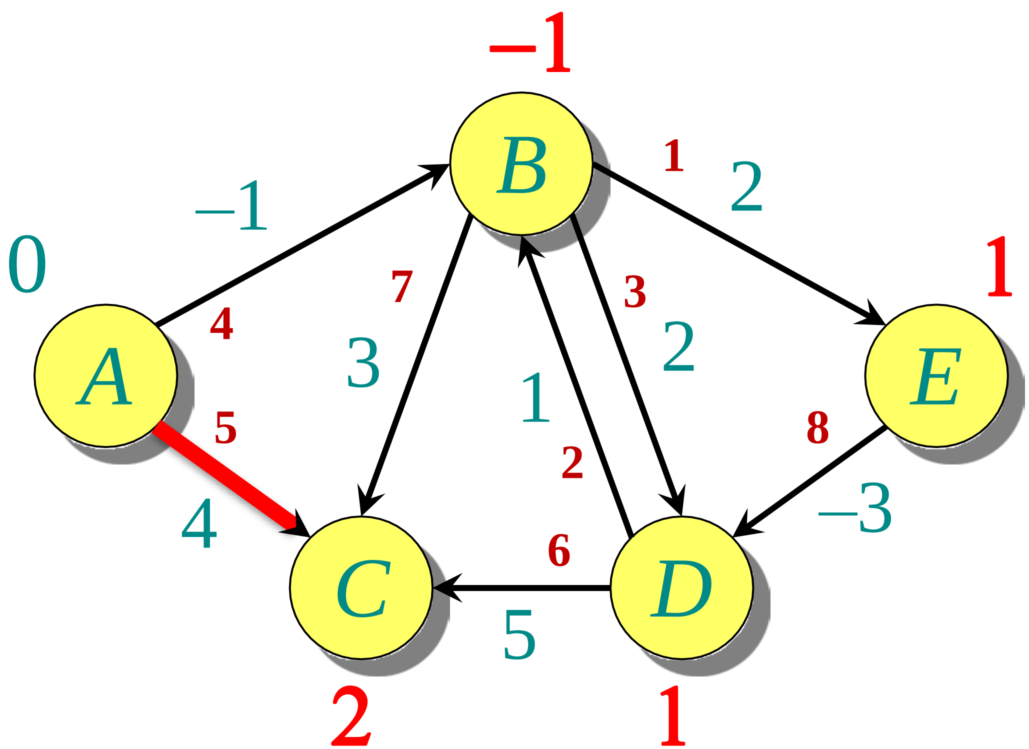
Copyright © 2001-5 by Erik D. Demaine and Charles E. Leiserson

Example of Bellman-Ford



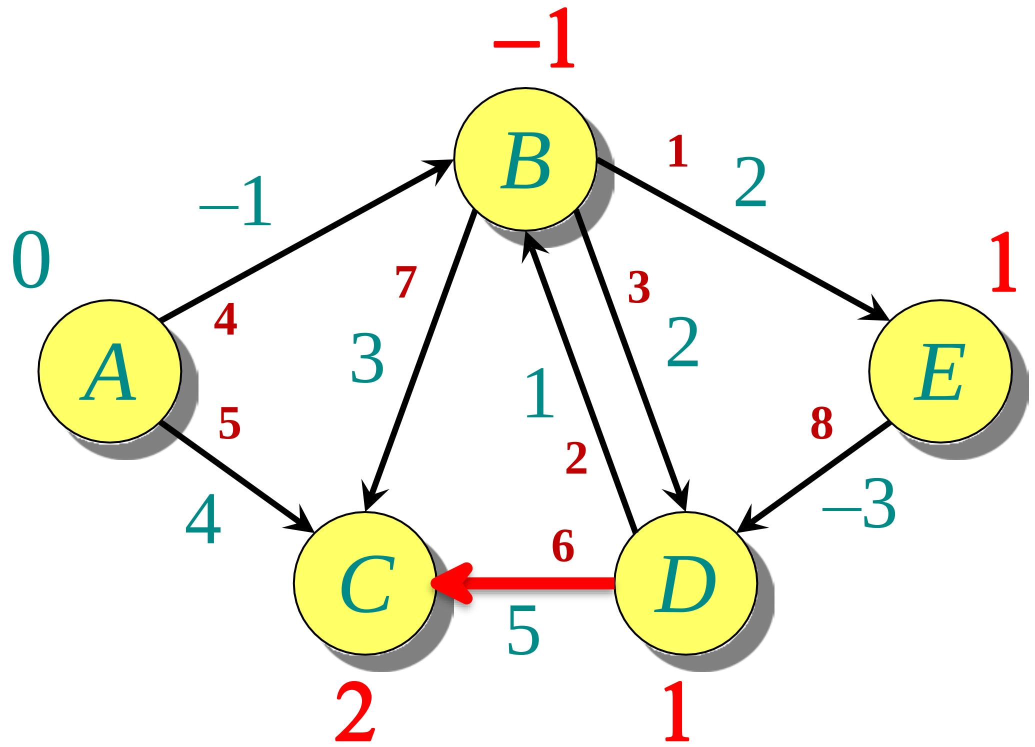
Copyright © 2001-5 by Erik D. Demaine and Charles E. Leiserson

Example of Bellman-Ford



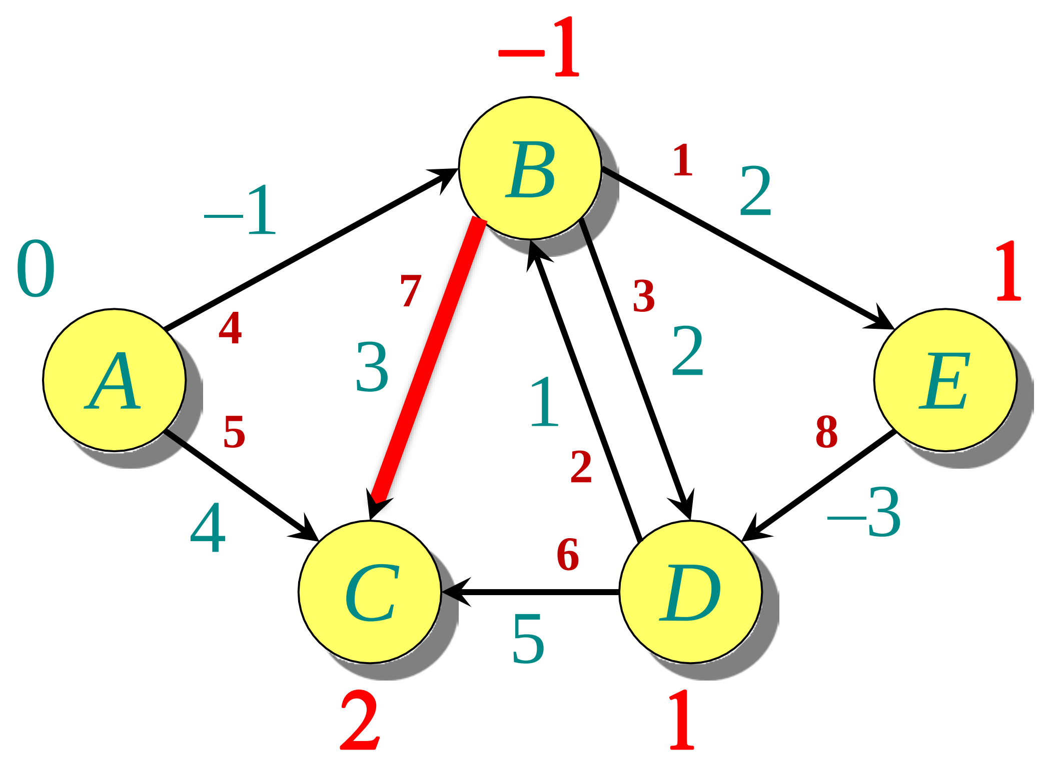
Copyright © 2001-5 by Erik D. Demaine and Charles E. Leiserson

Example of Bellman-Ford



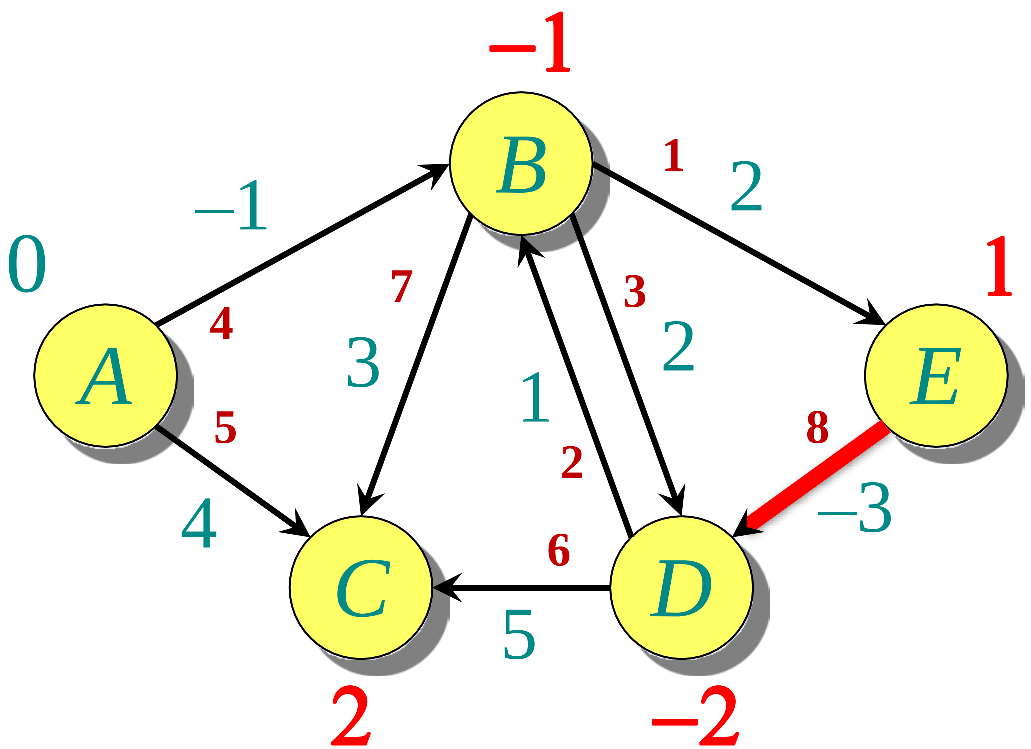
Copyright © 2001-5 by Erik D. Demaine and Charles E. Leiserson

Example of Bellman-Ford



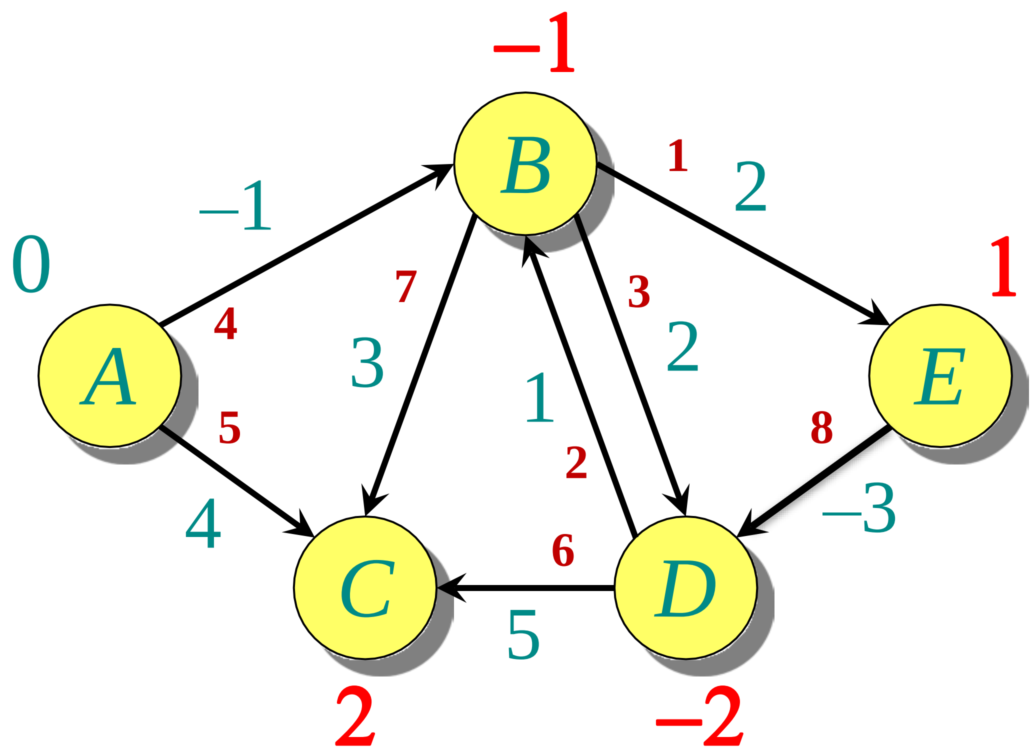
Copyright © 2001-5 by Erik D. Demaine and Charles E. Leiserson

EXAMPLE OF BELLMAN-FORD



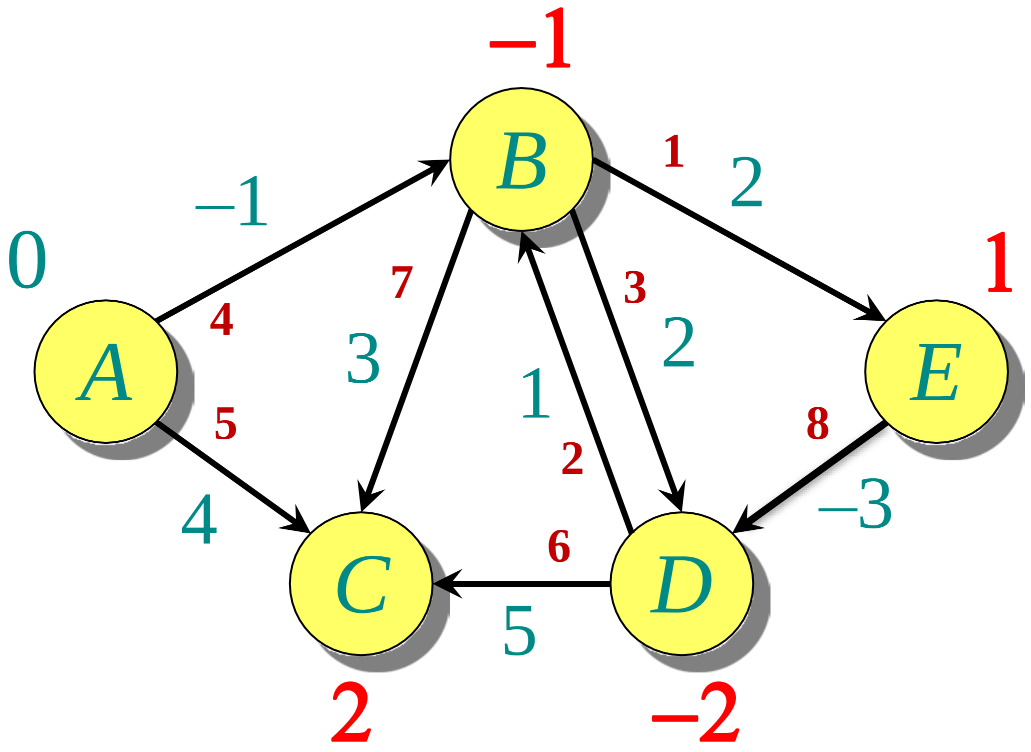
Copyright © 2001-5 by Erik D. Demaine and Charles E. Leiserson

Example of Bellman-Ford



Copyright © 2001-5 by Erik D. Demaine and Charles E. Leiserson

Example of Bellman-Ford



End of pass 2 (and 3 and 4).

Copyright © 2001-5 by Erik D. Demaine and Charles E. Leiserson

Correctness

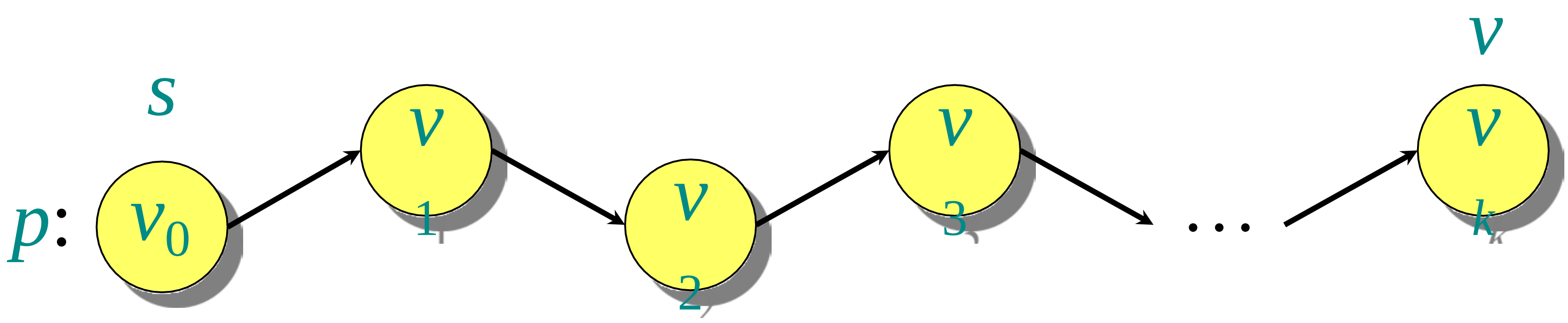
Theorem. If $G = (V, E)$ contains no negative-weight cycles, then after the Bellman-Ford algorithm executes, $d[v] = \delta(s, v)$ for all $v \in V$.

Copyright © 2001-5 by Erik D. Demaine and Charles E. Leiserson

CORRECTNESS

Theorem. If $G = (V, E)$ contains no negative-weight cycles, then after the Bellman-Ford algorithm executes, $d[v] = \delta(s, v)$ for all $v \in V$.

Proof. Let $v \in V$ be any vertex, and consider a shortest path p from s to v with the minimum number of edges.

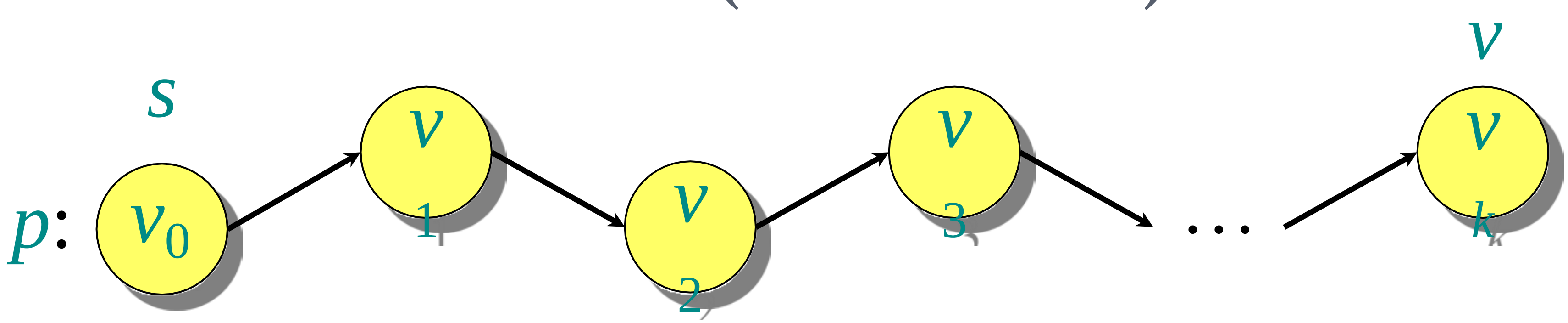


Since p is a shortest path, we have

$$\delta(s, v_i) = \delta(s, v_{i-1}) + w(v_{i-1}, v_i) .$$

Copyright © 2001-5 by Erik D. Demaine and Charles E. Leiserson

CORRECTNESS (CONTINUED)



Initially, $d[v_0] = 0 = \delta(s, v_0)$, and $d[v_0]$ is unchanged by subsequent relaxations (because of the lemma from *Shortest Paths I* that $d[v] \geq \delta(s, v)$).

- After 1 pass through E, we have $d[v_1] = \delta(s, v_1)$.
- After 2 passes through E, we have $d[v_2] = \delta(s, v_2)$.

- After k passes through E, we have $d[v_k] = \delta(s, v_k)$.

Since G contains no negative-weight cycles, p is simple. Longest simple path has $\leq |V| - 1$ edges. □

Copyright © 2001-5 by Erik D. Demaine and Charles E. Leiserson

DETECTION OF NEGATIVE-WEIGHT CYCLES

Corollary. If a value $d[v]$ fails to converge after $|V| - 1$ passes, there exists a negative-weight cycle in G reachable from s. $\square$

Copyright © 2001-5 by Erik D. Demaine and Charles E. Leiserson

Shortest paths

Single-source shortest paths

- Nonnegative edge weights
  - Dijkstra's algorithm: $O(E + V \lg V)$
- General
  - Bellman-Ford algorithm: $O(VE)$
- DAG
  - One pass of Bellman-Ford: $O(V + E)$

Copyright © 2001-5 by Erik D. Demaine and Charles E. Leiserson

Shortest paths

Single-source shortest paths

- Nonnegative edge weights
  - Dijkstra's algorithm: $O(E + V \lg V)$
- General
  - Bellman-Ford algorithm: $O(VE)$
- DAG
  - One pass of Bellman-Ford: $O(V + E)$

All-pairs shortest paths

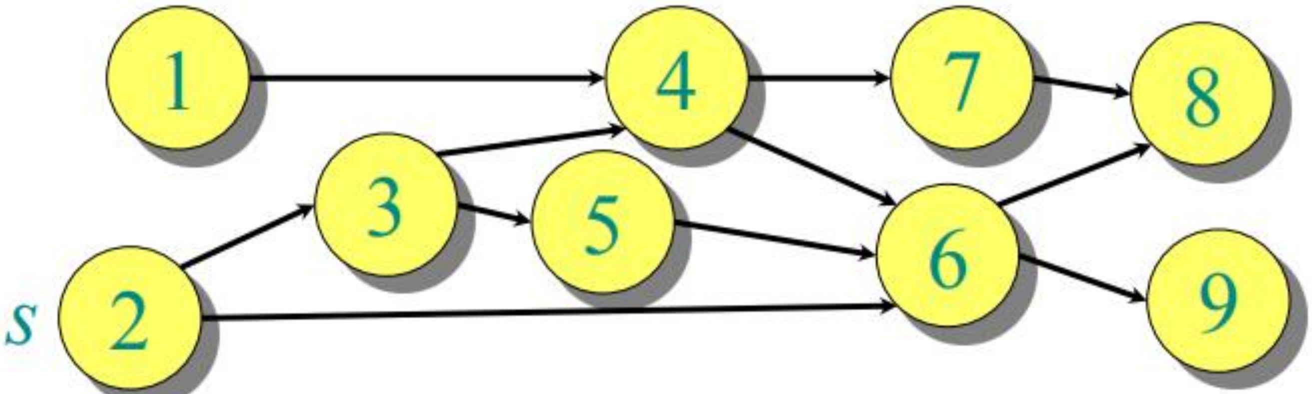
- Nonnegative edge weights
  - Dijkstra's algorithm $|V|$ times: $O(VE + V^2 \lg V)$

Copyright © 2001-5 by Erik D. Demaine and Charles E. Leiserson

Directed Acyclic Graph (DAG)

If the graph is a ***directed acyclic graph*** (***DAG***), we first ***topologically sort*** the vertices.

- Determine $f : V \to \{1, 2, \ldots, |V|\}$ such that $(u, v) \in E \Rightarrow f(u) < f(v)$.
- $O(V + E)$ time using depth-first search.



Walk through the vertices $u \in V$ in this order, relaxing the edges in $Adj[u]$, thereby obtaining the shortest paths from s in a total of $O(V + E)$ time.

All-pairs shortest paths

Input: Digraph $G = (V, E)$, where $V = \{1, 2, \ldots, n\}$, with edge-weight function $w : E \rightarrow \mathbb{R}$.
Output: $n \times n$ matrix of shortest-path lengths $\delta(i, j)$ for all $i, j \in V$.

Copyright © 2001-5 by Erik D. Demaine and Charles E. Leiserson

All-pairs shortest paths

Input: Digraph $G = (V, E)$, where $V = \{1, 2, \ldots, n\}$, with edge-weight function $w : E \to \mathbb{R}$.

Output: $n \times n$ matrix of shortest-path lengths $\delta(i, j)$ for all $i, j \in V$.

Idea:

- Run Bellman-Ford once from each vertex.
- Time $= O(V^2E)$.
- Dense graph ($\Theta(n^2)$ edges) $\Rightarrow$ $\Theta(n^4)$ time in the worst case.

Good first try!

Copyright © 2001-5 by Erik D. Demaine and Charles E. Leiserson

Reference

- **Introduction to Algorithms**
  - Single Source Shortest Path
  - Chapter # 24
  - Thomas H. Cormen