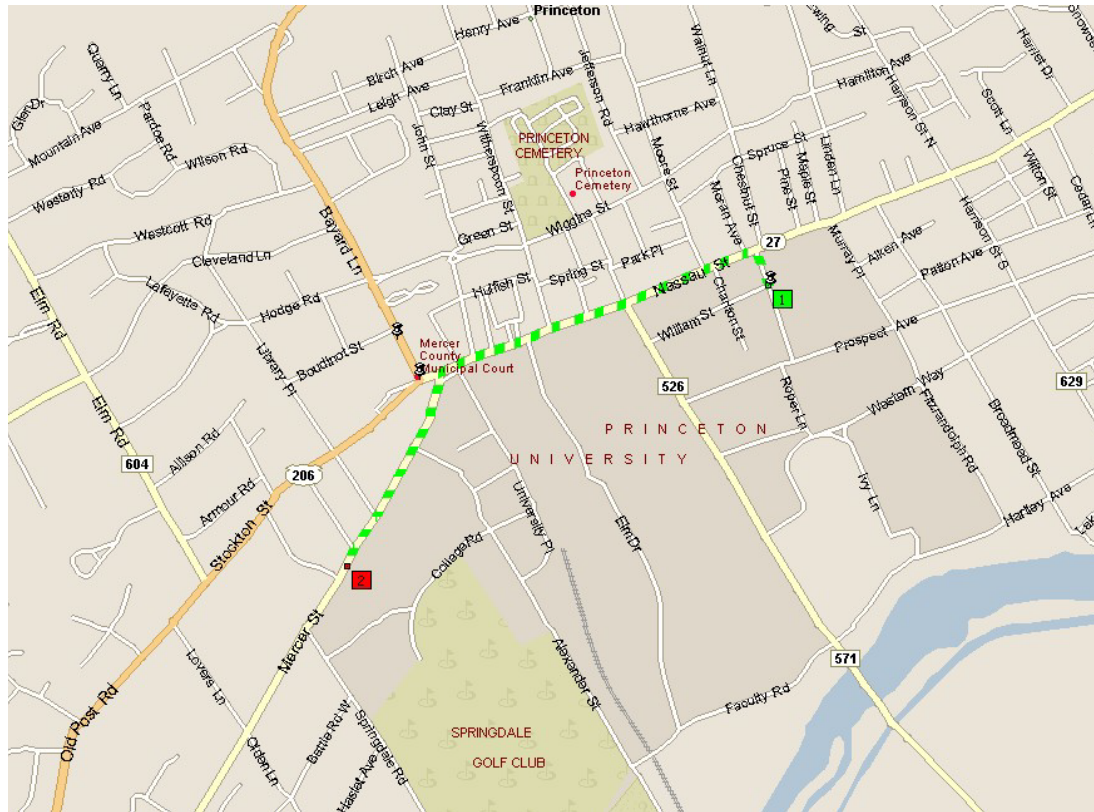


4.4 Shortest Paths in a Graph



shortest path from Princeton CS department to Einstein's house

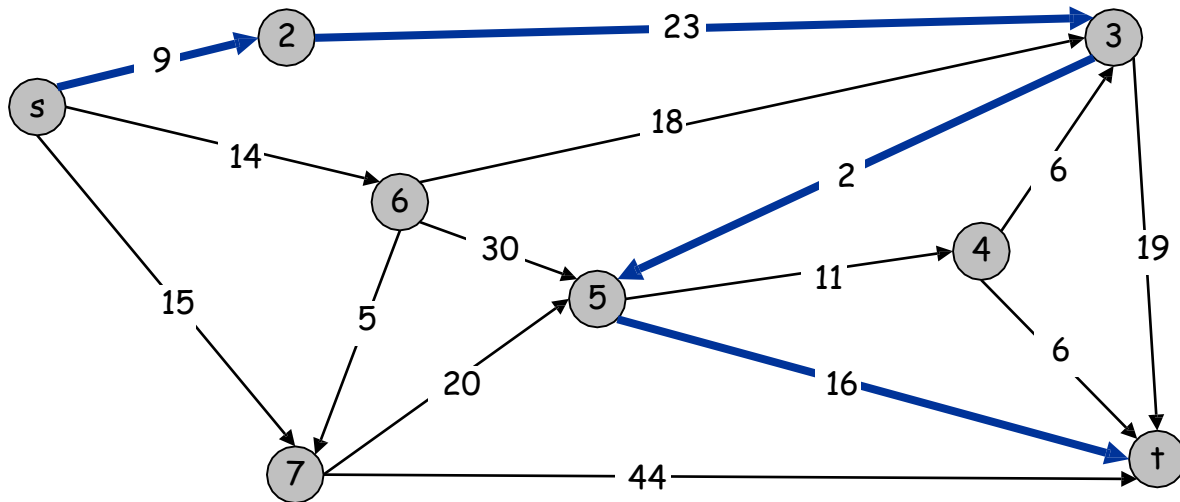
Shortest Path Problem

Shortest path network.

- Directed graph $G = (V, E)$.
- Source s , destination t .
- Length ℓ_e = length of edge e .

Shortest path problem: find shortest directed path from s to t .

↑
cost of path = sum of edge costs in path



Cost of path $s-2-3-5-t$
 $= 9 + 23 + 2 + 16$
 $= 48.$

Dijkstra's Algorithm

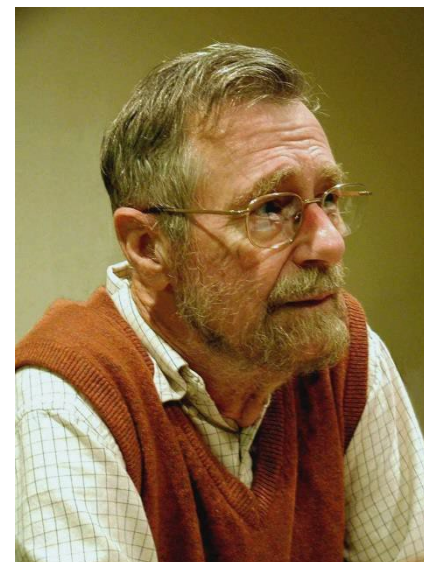
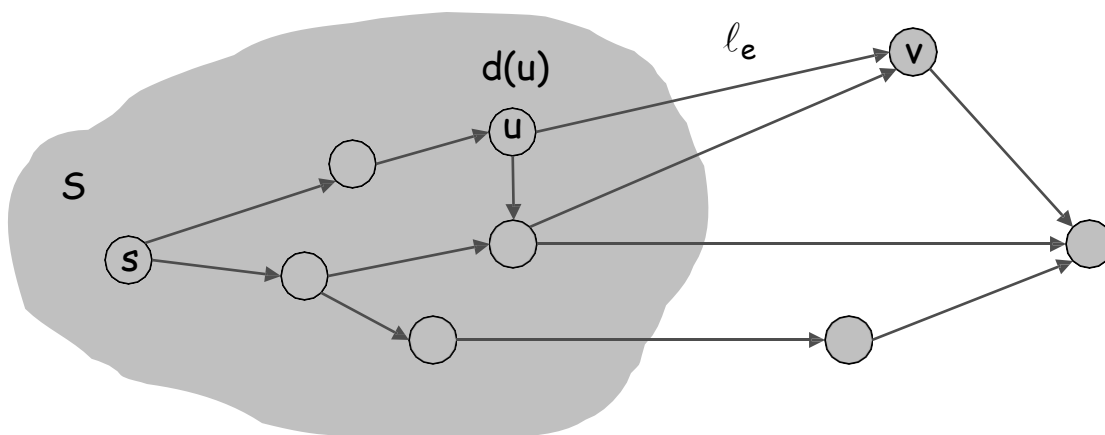
Dijkstra's algorithm.

- Maintain a set of **explored nodes** S for which we have determined the shortest path distance $d(u)$ from s to u .
- Initialize $S = \{s\}$, $d(s) = 0$.
- Repeatedly choose unexplored node v which minimizes

$$\pi(v) = \min_{e=(u,v) : u \in S} d(u) + \ell_e,$$

add v to S , and set $d(v) = \pi(v)$.

← shortest path to some u in explored part, followed by a single edge (u, v)



Dijkstra's Algorithm: Proof of Correctness



Dijkstra's Algorithm: Proof of Correctness

Invariant. For each node $u \in S$, $d(u)$ is the length of the shortest s - u path.

Dijkstra's algorithm.

Recall:

- Maintain a set of **explored nodes** S for which we have determined the shortest path distance $d(u)$ from s to u .
- Initialize $S = \{s\}$, $d(s) = 0$.
- Repeatedly choose unexplored node v which minimizes

$$\pi(v) = \min_{e = (u,v) : u \in S} d(u) + \ell_e,$$

add v to S , and set $d(v) = \pi(v)$.

← shortest path to some u in explored part, followed by a single edge (u, v)

Dijkstra's Algorithm: Proof of Correctness

Invariant. For each node $u \in S$, $d(u)$ is the length of the shortest s - u path.

Pf. (by induction on $|S|$)

Base case: $|S| = 1$?

Dijkstra's algorithm.

Recall:

- Maintain a set of **explored nodes** S for which we have determined the shortest path distance $d(u)$ from s to u .
- Initialize $S = \{s\}$, $d(s) = 0$.
- Repeatedly choose unexplored node v which minimizes

$$\pi(v) = \min_{e = (u,v) : u \in S} d(u) + \ell_e,$$

add v to S , and set $d(v) = \pi(v)$.

← shortest path to some u in explored part, followed by a single edge (u, v)

Dijkstra's Algorithm: Proof of Correctness

Invariant. For each node $u \in S$, $d(u)$ is the length of the shortest s - u path.

Pf. (by induction on $|S|$)

Base case: $|S| = 1$ holds by definition.

Inductive hypothesis:

Dijkstra's algorithm.

Recall:

- Maintain a set of **explored nodes** S for which we have determined the shortest path distance $d(u)$ from s to u .
- Initialize $S = \{s\}$, $d(s) = 0$.
- Repeatedly choose unexplored node v which minimizes

$$\pi(v) = \min_{e = (u,v) : u \in S} d(u) + \ell_e,$$

add v to S , and set $d(v) = \pi(v)$.

← shortest path to some u in explored part, followed by a single edge (u, v)

Dijkstra's Algorithm: Proof of Correctness

Invariant. For each node $u \in S$, $d(u)$ is the length of the shortest s - u path.

Pf. (by induction on $|S|$)

Base case: $|S| = 1$ holds by definition.

Inductive hypothesis: Assume invariant true when $|S| = k \geq 1$.

Dijkstra's algorithm.

Recall:

- Maintain a set of **explored nodes** S for which we have determined the shortest path distance $d(u)$ from s to u .
- Initialize $S = \{s\}$, $d(s) = 0$.
- Repeatedly choose unexplored node v which minimizes

$$\pi(v) = \min_{e = (u,v) : u \in S} d(u) + \ell_e,$$

add v to S , and set $d(v) = \pi(v)$.

← shortest path to some u in explored part, followed by a single edge (u, v)

Dijkstra's Algorithm: Proof of Correctness

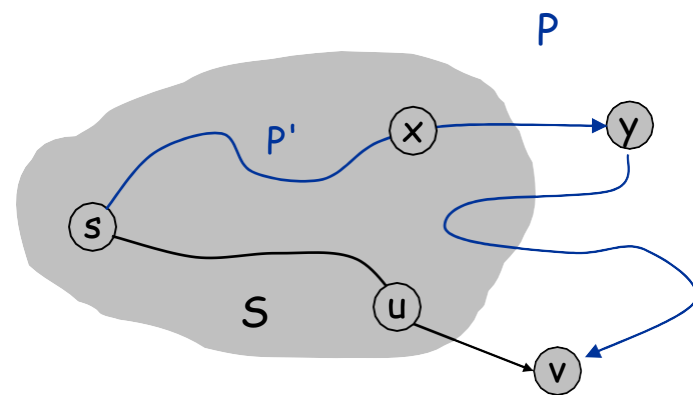
Invariant. For each node $u \in S$, $d(u)$ is the length of the shortest s - u path.

Pf. (by induction on $|S|$)

Base case: $|S| = 1$ holds by definition.

Inductive hypothesis: Assume invariant true when $|S| = k \geq 1$.

- Let v be next node added to S , and let u - v be the chosen edge.



Dijkstra's algorithm.

- Maintain a set of **explored nodes** S for which we have determined the shortest path distance $d(u)$ from s to u .
- Initialize $S = \{s\}$, $d(s) = 0$.
- Repeatedly choose unexplored node v which minimizes

$$\pi(v) = \min_{e = (u, v) : u \in S} d(u) + \ell_e,$$

add v to S , and set $d(v) = \pi(v)$.

shortest path to some u in explored part, followed by a single edge (u, v)

Recall:

Dijkstra's Algorithm: Proof of Correctness

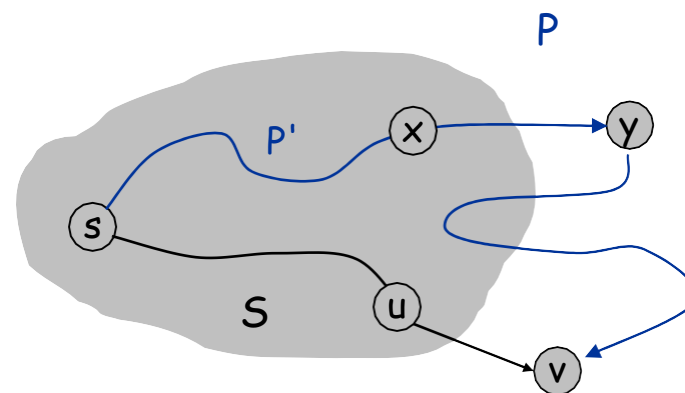
Invariant. For each node $u \in S$, $d(u)$ is the length of the shortest s - u path.

Pf. (by induction on $|S|$)

Base case: $|S| = 1$ holds by definition.

Inductive hypothesis: Assume invariant true when $|S| = k \geq 1$.

- Let v be next node added to S , and let u - v be the chosen edge.
- The shortest s - u path plus (u, v) is an s - v path of length $\pi(v)$.



Dijkstra's algorithm.

- Maintain a set of **explored nodes** S for which we have determined the shortest path distance $d(u)$ from s to u .
- Initialize $S = \{s\}$, $d(s) = 0$.
- Repeatedly choose unexplored node v which minimizes

$$\pi(v) = \min_{e = (u, v) : u \in S} d(u) + \ell_e,$$

add v to S , and set $d(v) = \pi(v)$.

shortest path to some u in explored part, followed by a single edge (u, v)

Recall:

Dijkstra's Algorithm: Proof of Correctness

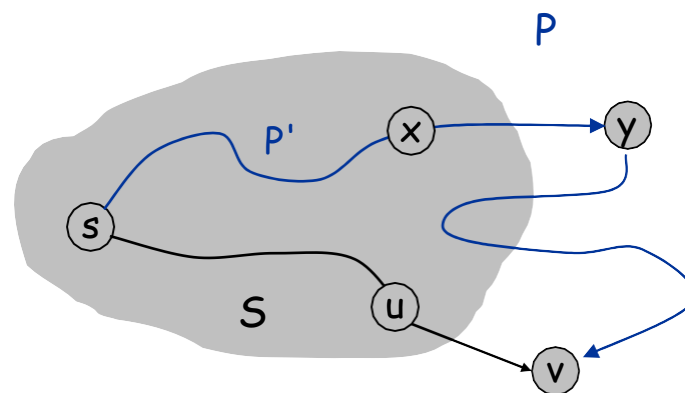
Invariant. For each node $u \in S$, $d(u)$ is the length of the shortest s - u path.

Pf. (by induction on $|S|$)

Base case: $|S| = 1$ holds by definition.

Inductive hypothesis: Assume invariant true when $|S| = k \geq 1$.

- Let v be next node added to S , and let u - v be the chosen edge.
- The shortest s - u path plus (u, v) is an s - v path of length $\pi(v)$.
- Consider any s - v path P . We'll see that it's no shorter than $\pi(v)$.



Dijkstra's algorithm.

- Maintain a set of **explored nodes** S for which we have determined the shortest path distance $d(u)$ from s to u .
- Initialize $S = \{s\}$, $d(s) = 0$.
- Repeatedly choose unexplored node v which minimizes

$$\pi(v) = \min_{e = (u, v) : u \in S} d(u) + \ell_e,$$

add v to S , and set $d(v) = \pi(v)$.

shortest path to some u in explored part, followed by a single edge (u, v)

Recall:

Dijkstra's Algorithm: Proof of Correctness

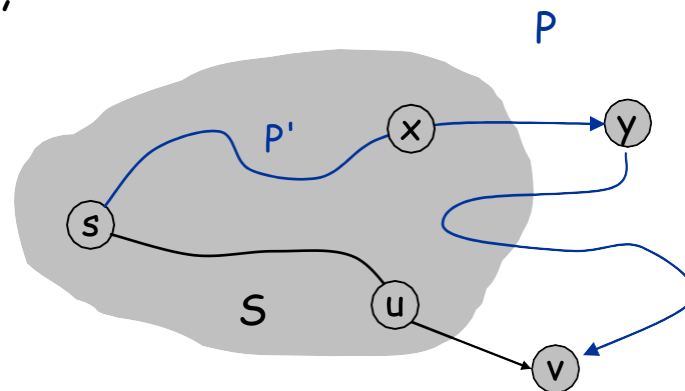
Invariant. For each node $u \in S$, $d(u)$ is the length of the shortest s - u path.

Pf. (by induction on $|S|$)

Base case: $|S| = 1$ holds by definition.

Inductive hypothesis: Assume invariant true when $|S| = k \geq 1$.

- Let v be next node added to S , and let u - v be the chosen edge.
- The shortest s - u path plus (u, v) is an s - v path of length $\pi(v)$.
- Consider any s - v path P . We'll see that it's no shorter than $\pi(v)$.
- Let x - y be the first edge in P that leaves S , and let P' be the subpath to x .



Dijkstra's algorithm.

- Maintain a set of **explored nodes** S for which we have determined the shortest path distance $d(u)$ from s to u .
- Initialize $S = \{s\}$, $d(s) = 0$.
- Repeatedly choose unexplored node v which minimizes

$$\pi(v) = \min_{e = (u, v) : u \in S} d(u) + \ell_e,$$

add v to S , and set $d(v) = \pi(v)$.

shortest path to some u in explored part, followed by a single edge (u, v)

Recall:

Dijkstra's Algorithm: Proof of Correctness

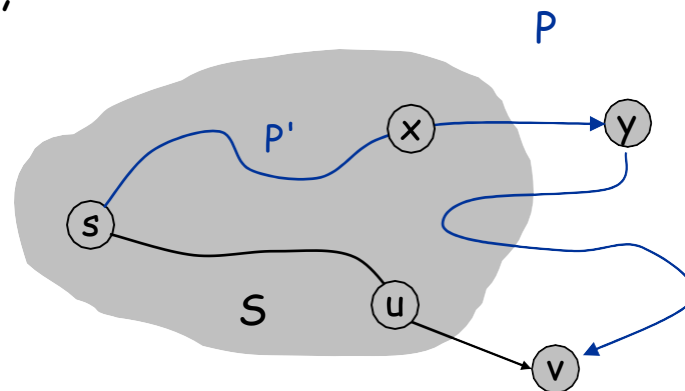
Invariant. For each node $u \in S$, $d(u)$ is the length of the shortest s - u path.

Pf. (by induction on $|S|$)

Base case: $|S| = 1$ holds by definition.

Inductive hypothesis: Assume invariant true when $|S| = k \geq 1$.

- Let v be next node added to S , and let u - v be the chosen edge.
- The shortest s - u path plus (u, v) is an s - v path of length $\pi(v)$.
- Consider any s - v path P . We'll see that it's no shorter than $\pi(v)$.
- Let x - y be the first edge in P that leaves S , and let P' be the subpath to x .
- P will turn out to be too long:



Dijkstra's algorithm.

- Maintain a set of **explored nodes** S for which we have determined the shortest path distance $d(u)$ from s to u .
- Initialize $S = \{s\}$, $d(s) = 0$.
- Repeatedly choose unexplored node v which minimizes

$$\pi(v) = \min_{e = (u, v) : u \in S} d(u) + \ell_e,$$

add v to S , and set $d(v) = \pi(v)$.

← shortest path to some u in explored part, followed by a single edge (u, v)

Recall:

Dijkstra's Algorithm: Proof of Correctness

Invariant. For each node $u \in S$, $d(u)$ is the length of the shortest s - u path.

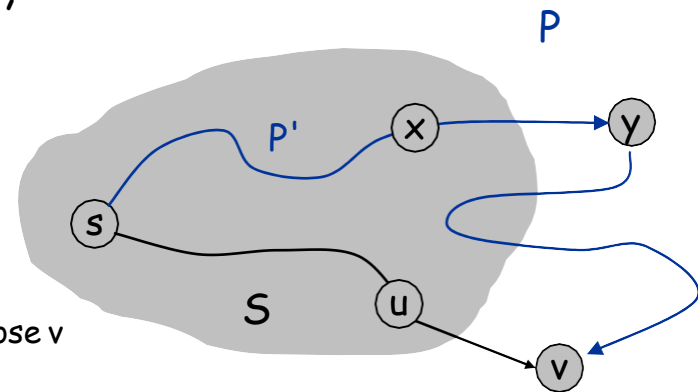
Pf. (by induction on $|S|$)

Base case: $|S| = 1$ holds by definition.

Inductive hypothesis: Assume invariant true when $|S| = k \geq 1$.

- Let v be next node added to S , and let u - v be the chosen edge.
- The shortest s - u path plus (u, v) is an s - v path of length $\pi(v)$.
- Consider any s - v path P . We'll see that it's no shorter than $\pi(v)$.
- Let x - y be the first edge in P that leaves S , and let P' be the subpath to x .
- P will turn out to be too long:

$$\begin{array}{ccccccc} \ell(P) \geq \ell(P') + \ell(x, y) \geq d(x) + \ell(x, y) \geq \pi(y) \geq \pi(v) \\ \uparrow & & \uparrow & & \uparrow & & \uparrow \\ \text{nonnegative} & & \text{inductive} & & \text{defn of } \pi(y) & & \text{Dijkstra chose } v \\ \text{weights} & & \text{hypothesis} & & & & \text{instead of } y \end{array}$$



Dijkstra's algorithm.

- Maintain a set of **explored nodes** S for which we have determined the shortest path distance $d(u)$ from s to u .
- Initialize $S = \{s\}$, $d(s) = 0$.
- Repeatedly choose unexplored node v which minimizes

Recall:

$$\pi(v) = \min_{e = (u, v) : u \in S} d(u) + \ell_e,$$

add v to S , and set $d(v) = \pi(v)$.

shortest path to some u in explored part, followed by a single edge (u, v)

Dijkstra's Algorithm: Implementation

For each unexplored node, explicitly maintain $\pi(v) = \min_{e=(u,v): u \in S} d(u) + \ell_e$.

- Next node to explore = node with minimum $\pi(v)$.
- When exploring v , for each incident edge $e = (v, w)$, update

$$\pi(w) = \min \{ \pi(w), \pi(v) + \ell_e \}.$$

Efficient implementation. Maintain a priority queue of unexplored nodes, prioritized by $\pi(v)$.

PQ Operation	Dijkstra	Array	Binary heap	d-way Heap	Fib heap [†]
Insert	n	n	$\log n$	$d \log_d n$	1
ExtractMin	n	n	$\log n$	$d \log_d n$	$\log n$
ChangeKey	m	1	$\log n$	$\log_d n$	1
IsEmpty	n	1	1	1	1
Total		n^2	$m \log n$	$m \log_{m/n} n$	$m + n \log n$

[†] Individual ops are amortized bounds