UNIVERSIDAD NACIONAL DE COLOMBIA

Algoritmia Avanzada

Sesión 6

Single-Source Shortest Path Algorithms

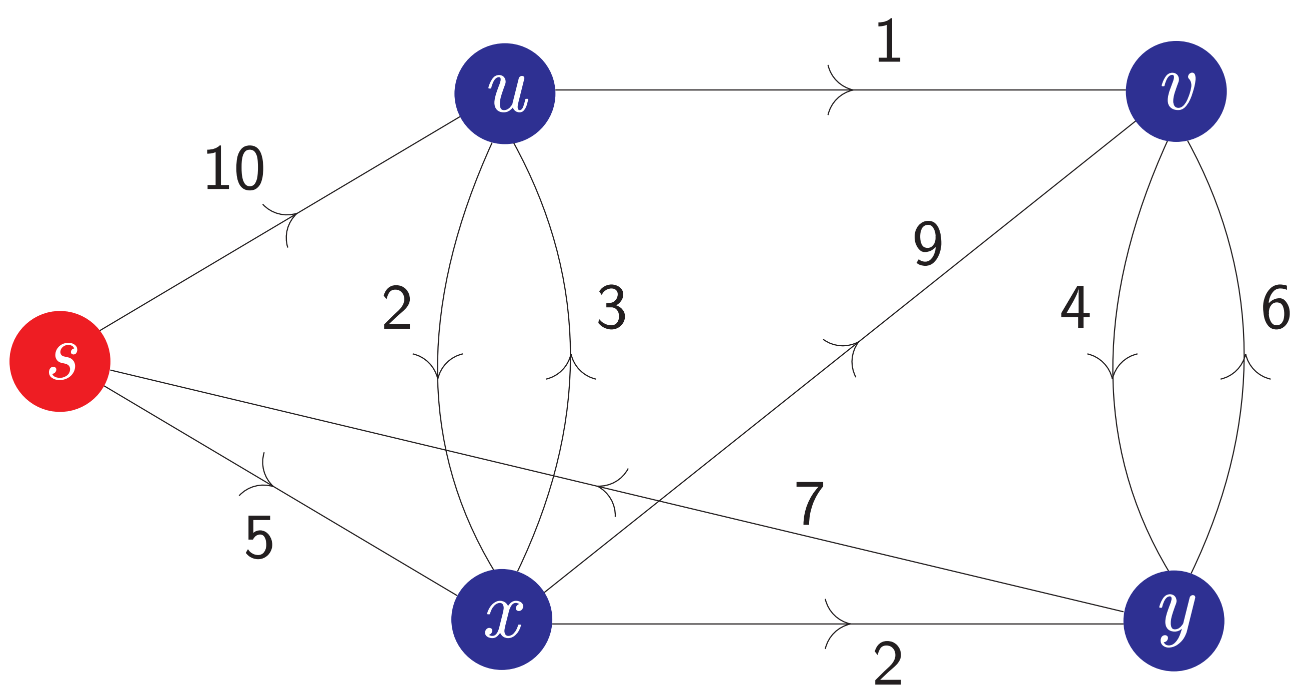
Yoan Pinzón, PhD
Universidad Nacional de Colombia
http://disi.unal.edu.co/~ypinzon/2019762/

Session 6

- **Shortest Paths**
  - ▷ SSSP (Single-Source Shortest Paths)
    - ◇ The Dijkstra's Algorithm
    - ◇ The Bellmand-Ford Algorithm

Single-Source Shortest Paths (SSSP)

Compute the shortest paths, that is, paths with the smallest total weight, from s to all other vertices.



- $G = (V, E)$: directed graph; $n = |V|, m = |E|$
- $w(u, v)$: the weight of edge (u, v)
- $s \in V$: the source vertex

The Dijkstra's Algorithm

by Edsger Wybe Dijkstra, 1959

Dijkstra's algorithm solves the SSSP problem for the case in which all edge weights are nonnegative.

- **Pseudo-code:**

```
DIJKSTRA(G, w, s)
    for each vertex v ∈ V[G]
        do d[v] ← ∞
           π[v] ← NIL
    d[s] ← 0
    S ← ∅
    Q ← V[G]
    while Q ≠ ∅
        do u ← EXTRACT-MIN(Q)
           S ← S ∪ {u}
           for each vertex v ∈ Adj[u]
               do RELAX(u, v, w)
```

```
RELAX(u, v, w)
    if d[v] > d[u] + w(u, v)
        then d[v] ← d[u] + w(u, v)
             π[v] ← u
```

- **Time Complexity:** $O(m \log n)$

lines 1-6 takes $\Theta(n)$. For each edge (u, v), $\textsc{Relax}(u, v)$ is performed exactly once. Note that $n - 1 \leq m \leq n(n-1)$, so $m = \Omega(n)$ and $m = O(n^2)$. We can implement Q in two ways:

1) Q implemented as unordered list:

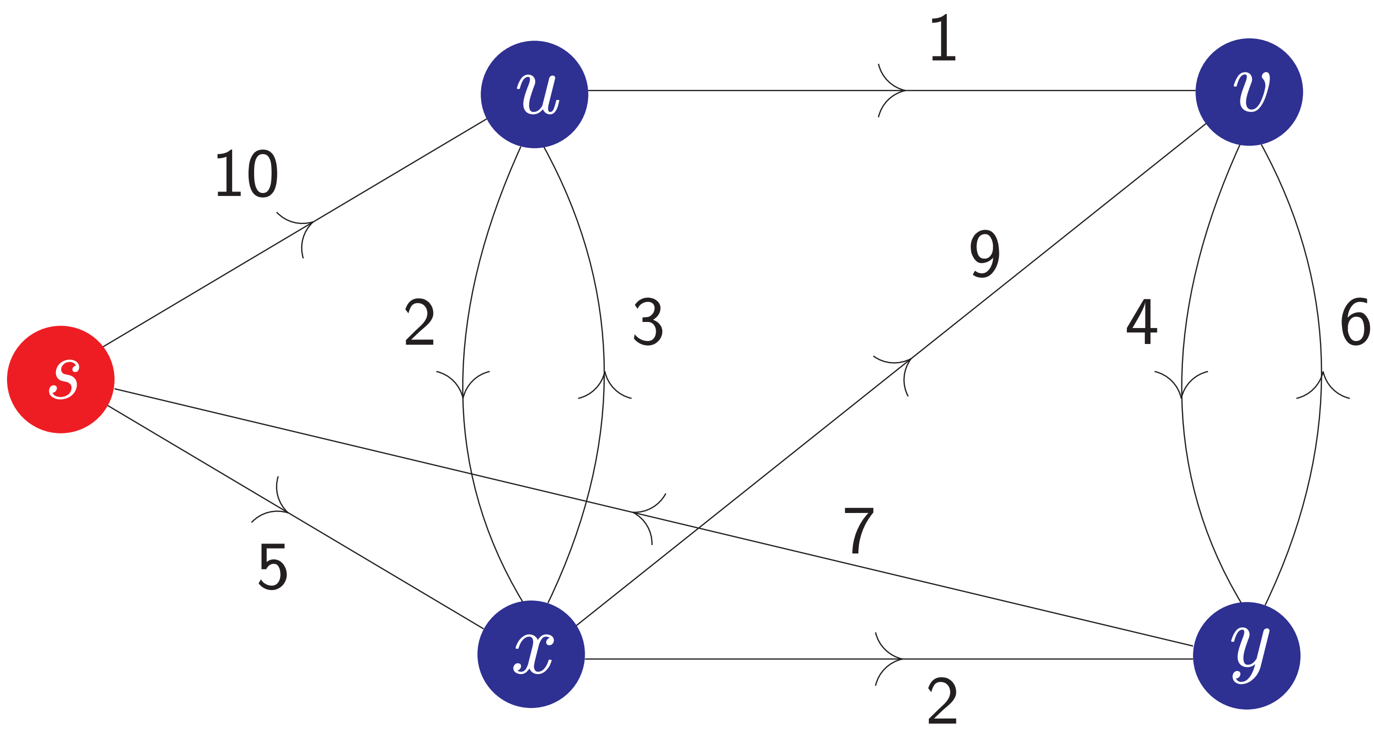
- One $\textsc{Extract-Min}(Q)$: $O(n)$, all of them $O(n^2)$
- One $\textsc{Relax}(u, v)$: $O(1)$, all of them $O(m)$
- The total running time: $\Theta(n) + O(n^2) + O(m) = \boxed{O(n^2)}$

2) Q implemented as a heap:

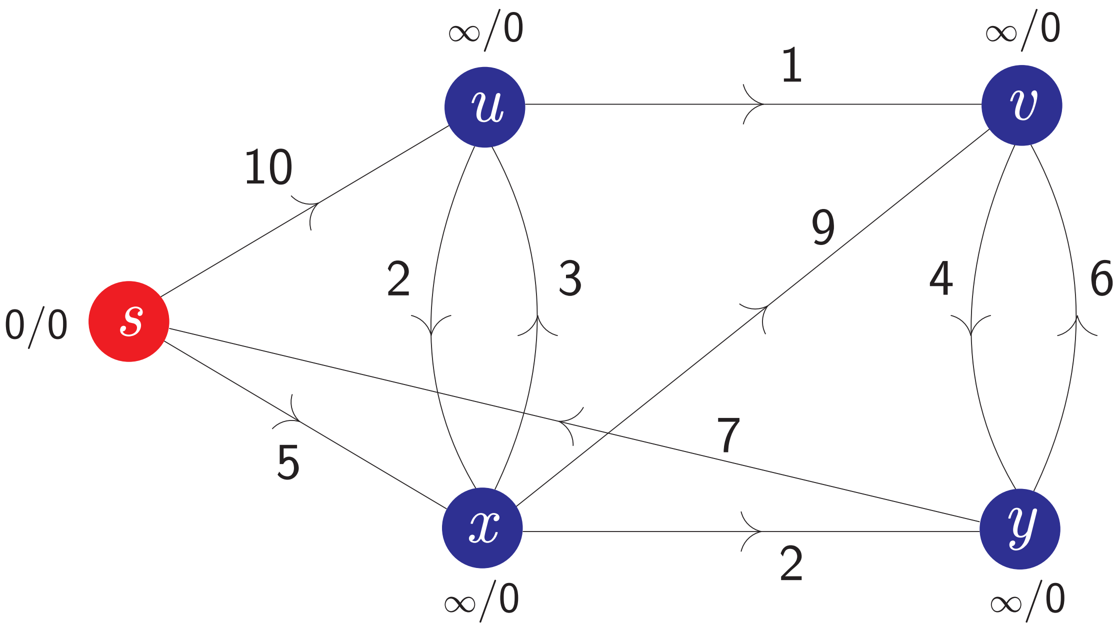
- One $\textsc{Extract-Min}(Q)$: $O(\log n)$, all of them $O(n \log n)$
- One $\textsc{Relax}(u, v)$: $O(\log n)$ (since may involve one $\textsc{Decrease-Key}$ operation), all of them $O(m \log n)$
- The total running time: $\Theta(n) + O(n \log n) + O(m \log n) = \boxed{O(m \log n)}$

The Dijkstra's Algorithm
Example

Find the SSSP for the following graph.



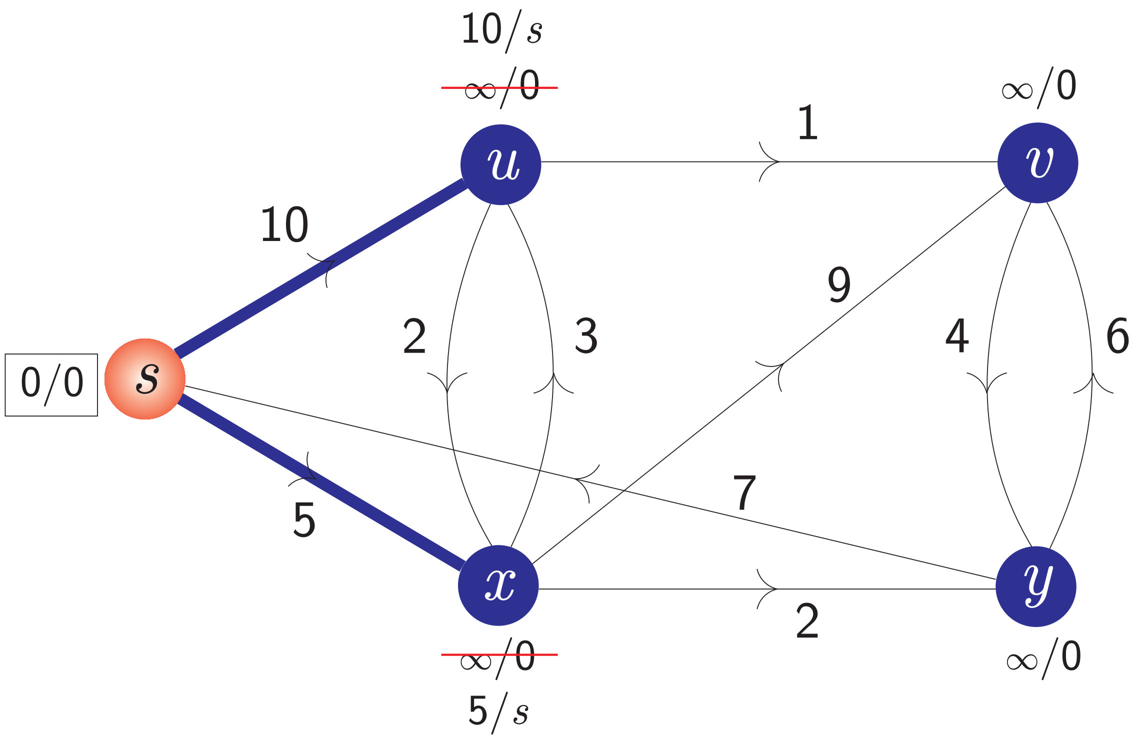
The Dijkstra's Algorithm
Solution - Iteration 0



0) $Q=\{s,u,v,x,y\},\ S=\{\}$

	0
s	$0/0$
u	$\infty/0$
v	$\infty/0$
x	$\infty/0$
y	$\infty/0$

The Dijkstra's Algorithm
Solution - Iteration 1

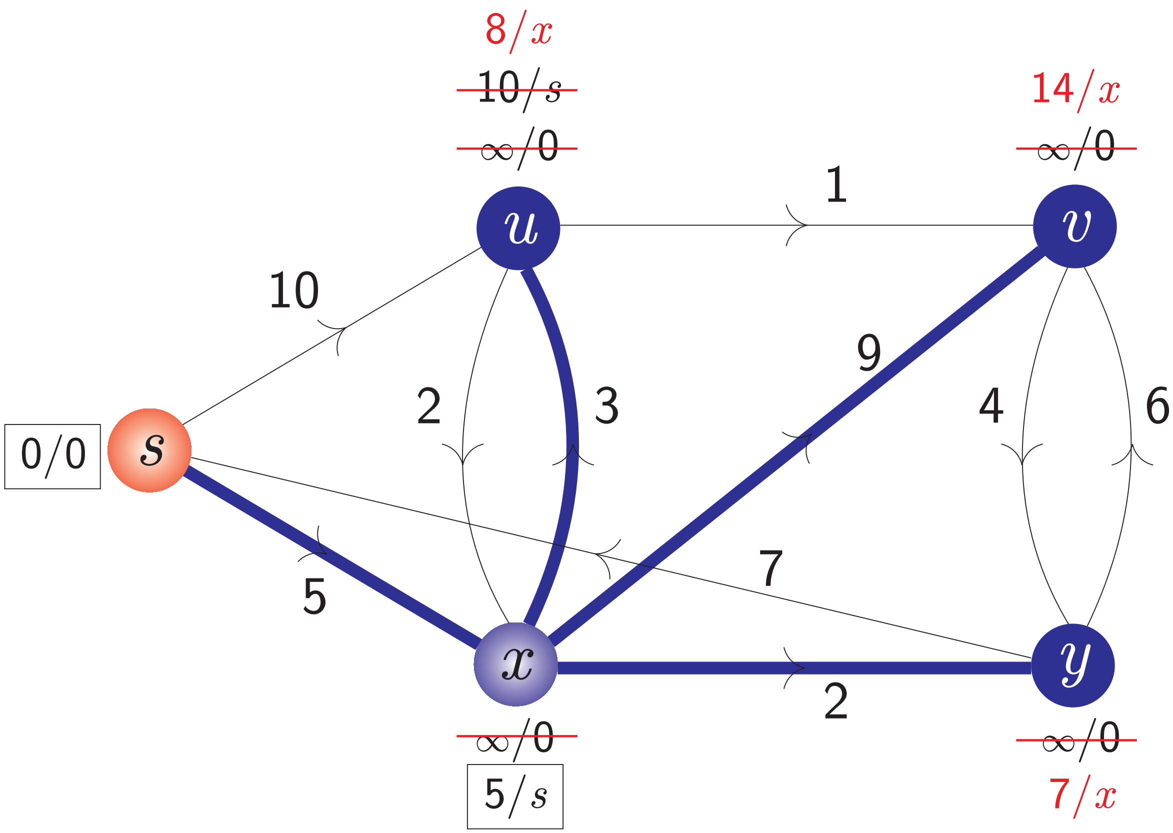


0) $Q=\{s,u,v,x,y\}$, $S=\{\}$
pickup s
1) $Q=\{u,v,x,y\}$, $S=\{s\}$

	0	s **1**
~~s~~	0/0	0/0
u	∞/0	10/s
v	∞/0	∞/0
x	∞/0	5/s
y	∞/0	∞/0

The Dijkstra's Algorithm

Solution - Iteration 2



0) $Q = \{s, u, v, x, y\}$, $S = \{\}$
pickup s

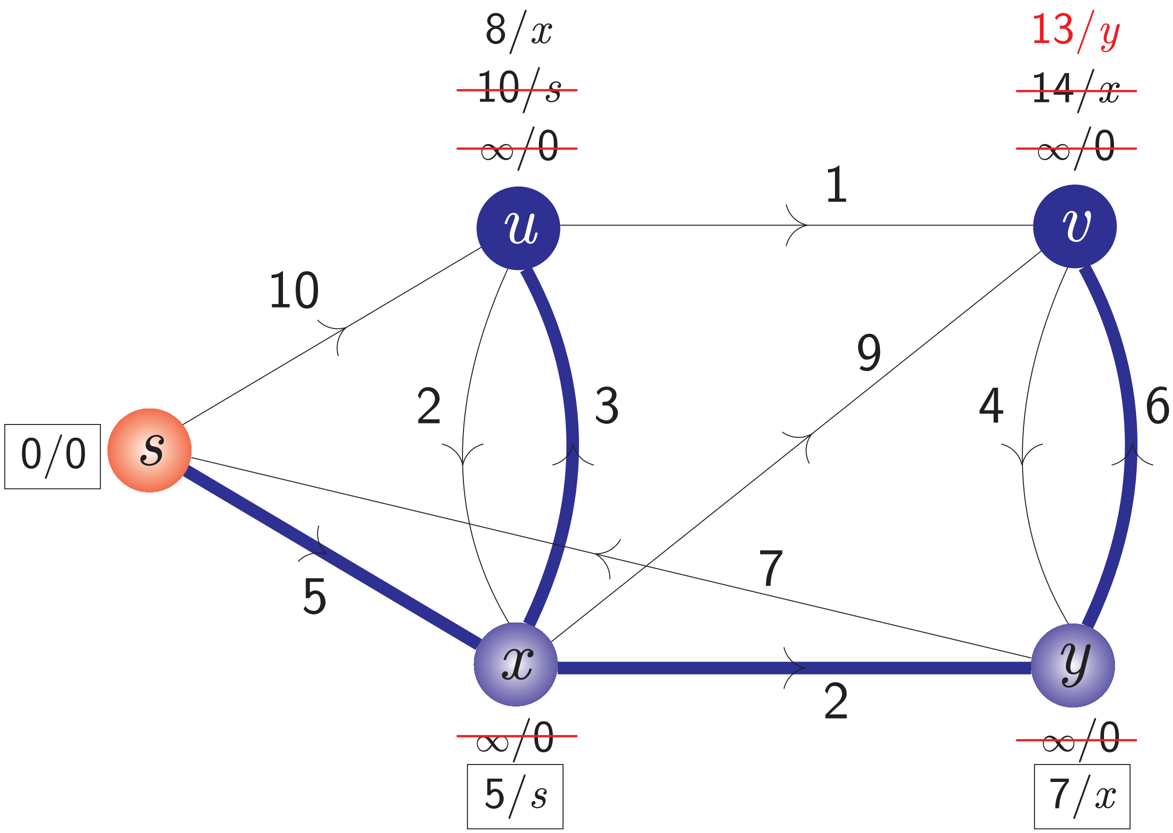
1) $Q = \{u, v, x, y\}$, $S = \{s\}$
pickup x

2) $Q = \{u, v, y\}$, $S = \{s, x\}$

	0	**1** (s)	**2** (x)
~~s~~	0/0	0/0	0/0
u	∞/0	10/s	8/x
v	∞/0	∞/0	14/x
~~x~~	∞/0	5/s	5/s
y	∞/0	∞/0	7/x

The Dijkstra's Algorithm

Solution - Iteration 3



0) $Q=\{s,u,v,x,y\}$, $S=\{\}$
pickup s

1) $Q=\{u,v,x,y\}$, $S=\{s\}$
pickup x

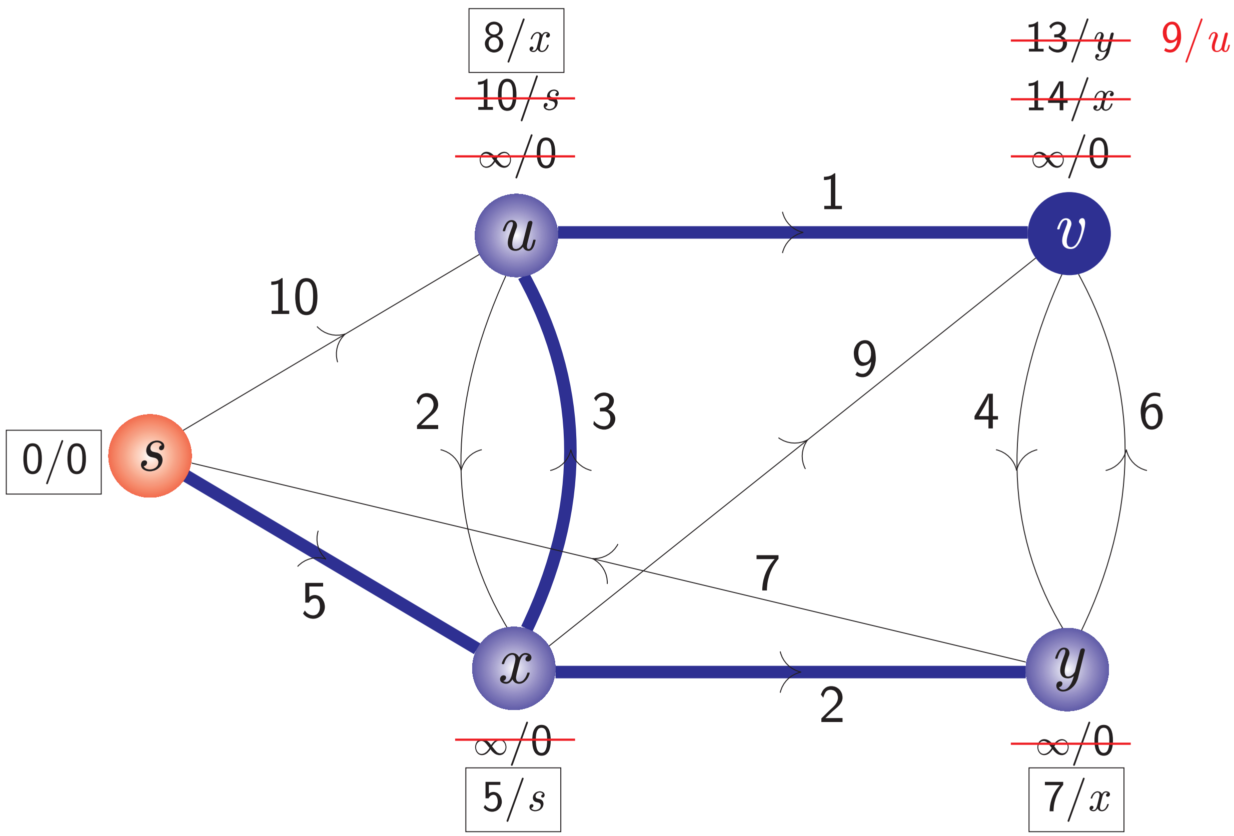
2) $Q=\{u,v,y\}$, $S=\{s,x\}$
pickup y

3) $Q=\{u,v\}$, $S=\{s,x,y\}$

		s	x	y
	0	**1**	**2**	**3**
$\cancel{s}$	0/0	0/0	0/0	0/0
u	$\infty/0$	$10/s$	$8/x$	$8/x$
v	$\infty/0$	$\infty/0$	$14/x$	$13/y$
$\cancel{x}$	$\infty/0$	$5/s$	$5/s$	$5/s$
$\cancel{y}$	$\infty/0$	$\infty/0$	$7/x$	$7/x$

The Dijkstra's Algorithm

Solution - Iteration 4



0) $Q=\{s,u,v,x,y\}$, $S=\{\}$
pickup s

1) $Q=\{u,v,x,y\}$, $S=\{s\}$
pickup x

2) $Q=\{u,v,y\}$, $S=\{s,x\}$
pickup y

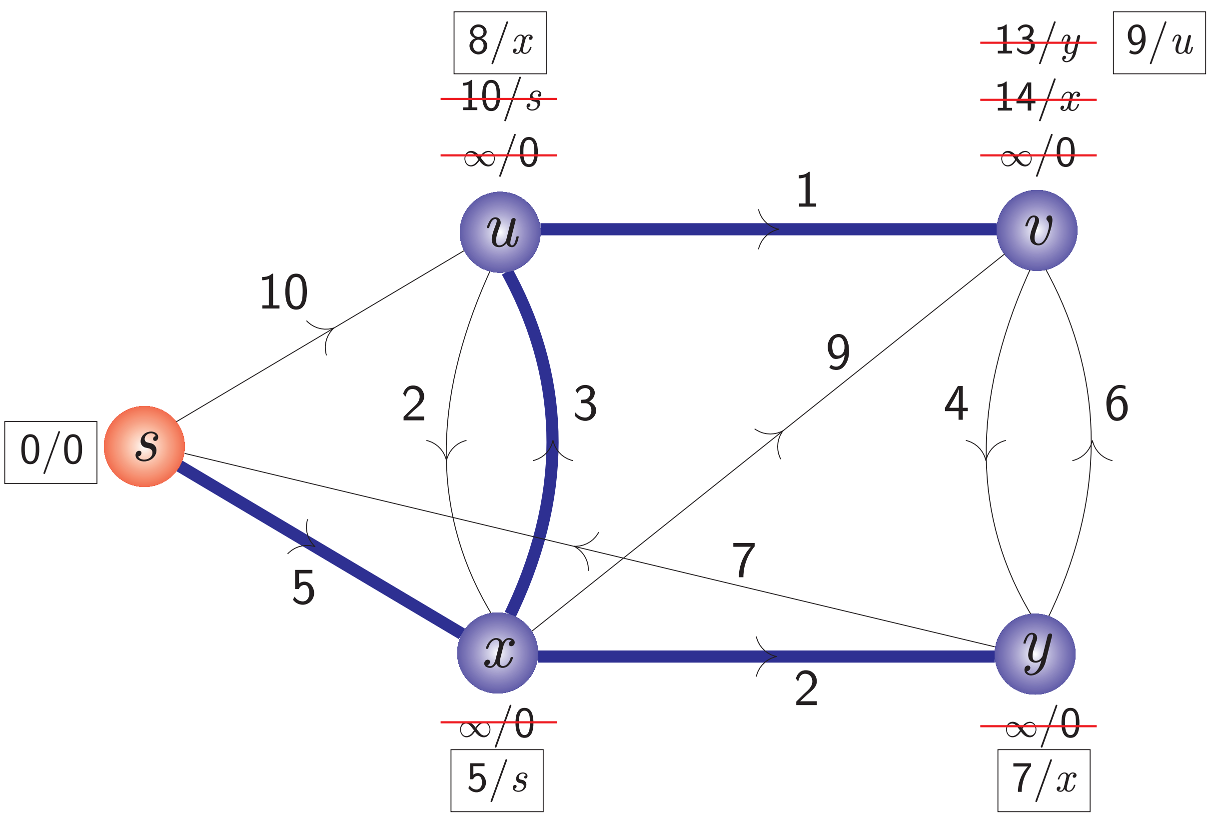
3) $Q=\{u,v\}$, $S=\{s,x,y\}$
pickup u

4) $Q=\{v\}$, $S=\{s,x,y,u\}$

		s	x	y	u
	0	**1**	**2**	**3**	**4**
$\cancel{s}$	$0/0$	$0/0$	$0/0$	$0/0$	$0/0$
$\cancel{u}$	$\infty/0$	$10/s$	$8/x$	$8/x$	$8/x$
v	$\infty/0$	$\infty/0$	$14/x$	$13/y$	$9/u$
$\cancel{x}$	$\infty/0$	$5/s$	$5/s$	$5/s$	$5/s$
$\cancel{y}$	$\infty/0$	$\infty/0$	$7/x$	$7/x$	$7/x$

The Dijkstra's Algorithm

Solution - Iteration 5



0) $Q=\{s,u,v,x,y\}$, $S=\{\}$
pickup s

1) $Q=\{u,v,x,y\}$, $S=\{s\}$
pickup x

2) $Q=\{u,v,y\}$, $S=\{s,x\}$
pickup y

3) $Q=\{u,v\}$, $S=\{s,x,y\}$
pickup u

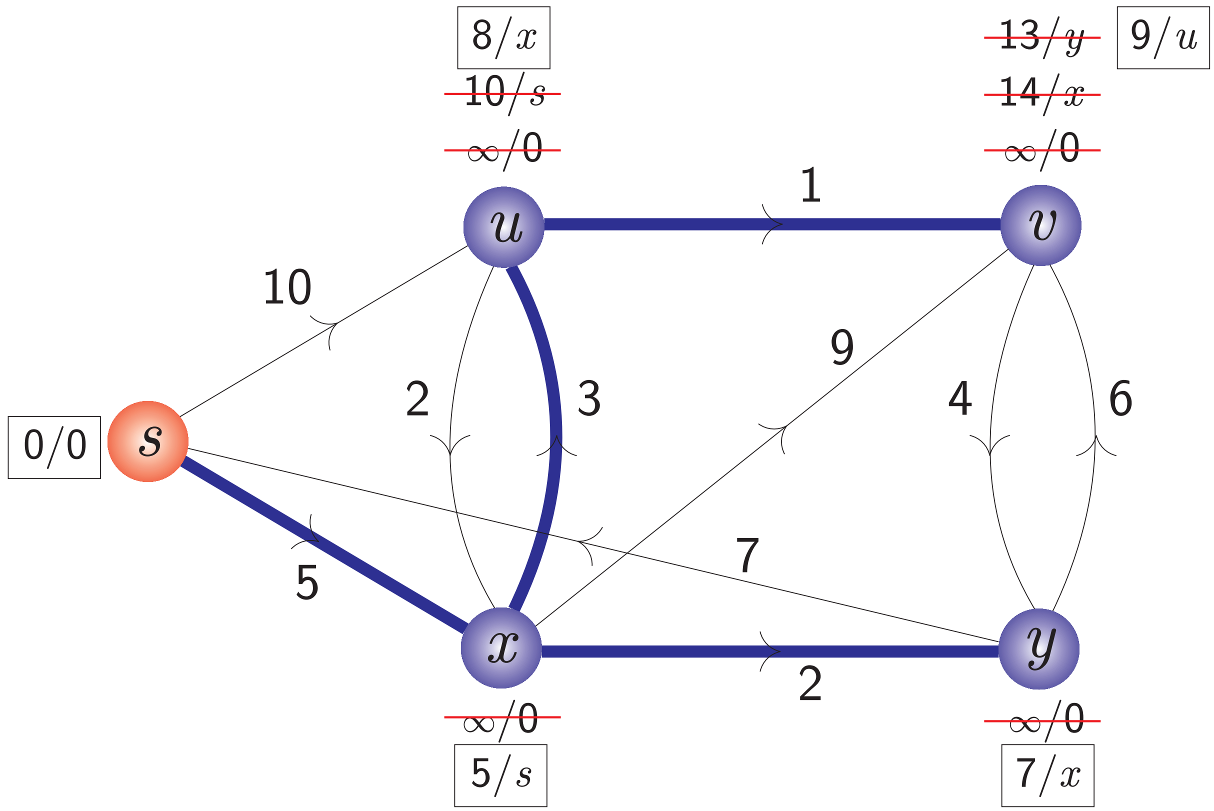
4) $Q=\{v\}$, $S=\{s,x,y,u\}$
pickup v

5) $Q=\{\}$, $S=\{s,x,y,u,v\}$

		s	x	y	u	v
	0	**1**	**2**	**3**	**4**	**5**
$\cancel{s}$	$0/0$	$0/0$	$0/0$	$0/0$	$0/0$	$0/0$
$\cancel{u}$	$\infty/0$	$10/s$	$8/x$	$8/x$	$8/x$	$8/x$
$\cancel{v}$	$\infty/0$	$\infty/0$	$14/x$	$13/y$	$9/u$	$9/u$
$\cancel{x}$	$\infty/0$	$5/s$	$5/s$	$5/s$	$5/s$	$5/s$
$\cancel{y}$	$\infty/0$	$\infty/0$	$7/x$	$7/x$	$7/x$	$7/x$

The Dijkstra's Algorithm

Solution - Iteration 5



0) $Q{=}\{s,u,v,x,y\}$, $S{=}\{\}$
pickup s

1) $Q{=}\{u,v,x,y\}$, $S{=}\{s\}$
pickup x

2) $Q{=}\{u,v,y\}$, $S{=}\{s,x\}$
pickup y

3) $Q{=}\{u,v\}$, $S{=}\{s,x,y\}$
pickup u

4) $Q{=}\{v\}$, $S{=}\{s,x,y,u\}$
pickup v

5) $Q{=}\{\}$, $S{=}\{s,x,y,u,v\}$

	0	1 (s)	2 (x)	3 (y)	4 (u)	5 (v)
$\cancel{s}$	$0/0$	$0/0$	$0/0$	$0/0$	$0/0$	$0/0$
$\cancel{u}$	$\infty/0$	$10/s$	$8/x$	$8/x$	$8/x$	$8/x$
$\cancel{v}$	$\infty/0$	$\infty/0$	$14/x$	$13/y$	$9/u$	$9/u$
$\cancel{x}$	$\infty/0$	$5/s$	$5/s$	$5/s$	$5/s$	$5/s$
$\cancel{y}$	$\infty/0$	$\infty/0$	$7/x$	$7/x$	$7/x$	$7/x$

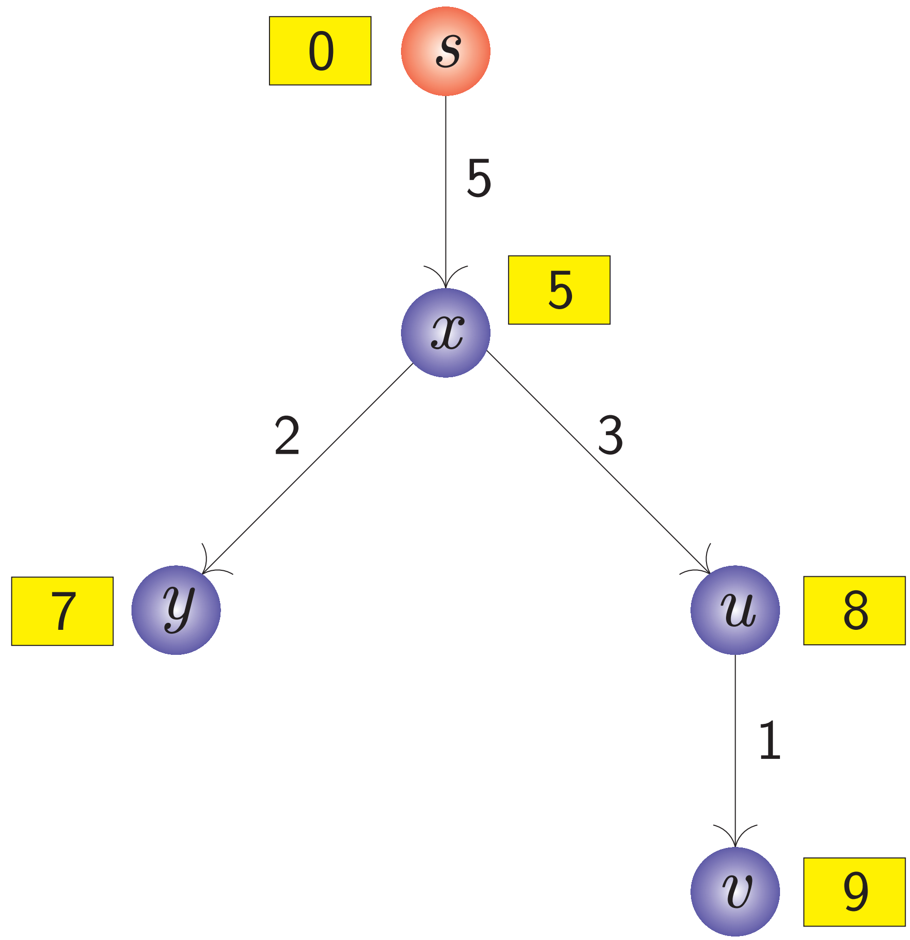
$$d = \begin{array}{ccccc} s & u & v & x & y \\ \hline 0 & 8 & 9 & 5 & 7 \end{array} \quad \delta(s,u)$$

$$\pi = \begin{array}{ccccc} s & u & v & x & y \\ \hline 0 & x & u & s & x \end{array} \quad PARENT[u]$$

The Dijkstra's Algorithm
Solution

$$d = \begin{array}{ccccc} s & u & v & x & y \\ \hline \boxed{0} & \boxed{8} & \boxed{9} & \boxed{5} & \boxed{7} \end{array} \quad \delta(s,u)$$

$$\pi = \begin{array}{ccccc} s & u & v & x & y \\ \hline \boxed{0} & \boxed{x} & \boxed{u} & \boxed{s} & \boxed{x} \end{array} \quad PARENT[u]$$



The Bellman-Ford Algorithm

by Richard Ernest Bellman and Lester Randolph Ford, 1958

Bellman-Ford algorithm solves the SSSP problem for the general case in which edge weights may be negative.

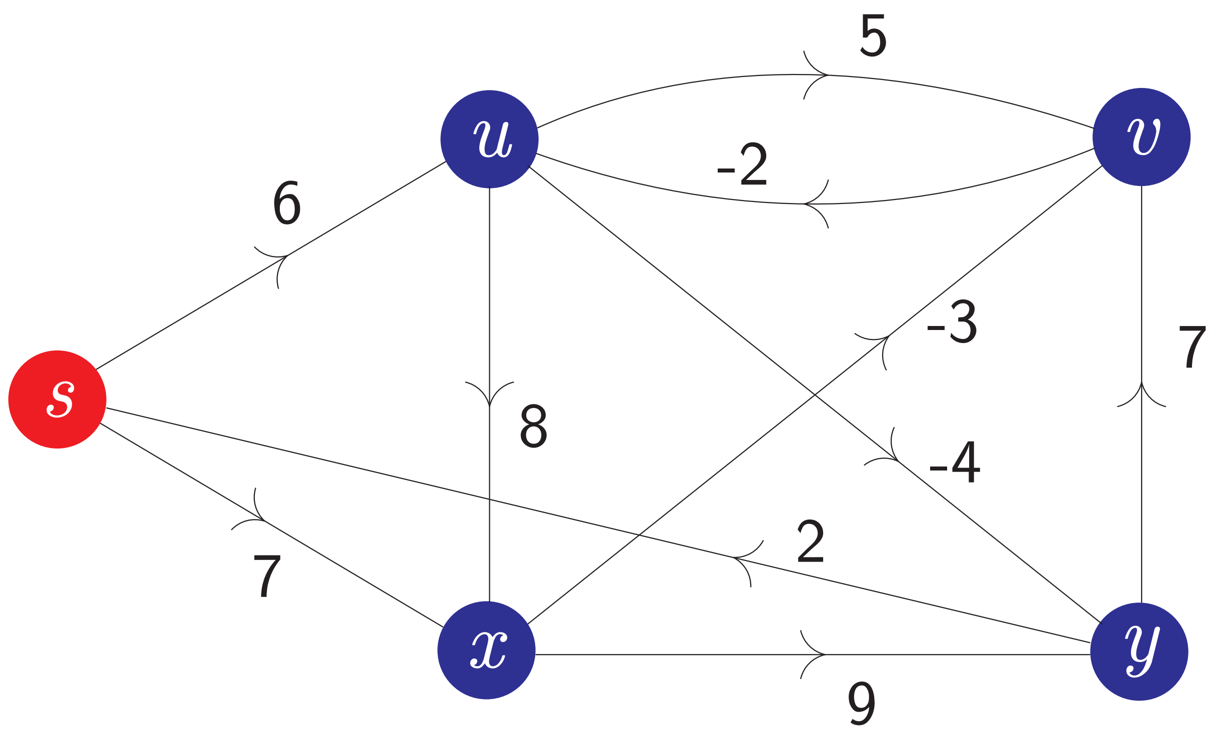
- **Pseudo-code:**

```
BELLMAND-FORD(G, w, s)
    for each vertex v ∈ V[G]
        do d[v] ← ∞
           π[v] ← NIL
    d[s] ← 0
    for i ← 1 to n − 1 do
        for each edge (u, v) ∈ E do
            RELAX(u, v, w)
    for each edge (u, v) ∈ E do
        if d[v] > d[u] + w(u, v) then return FALSE
    return TRUE
```

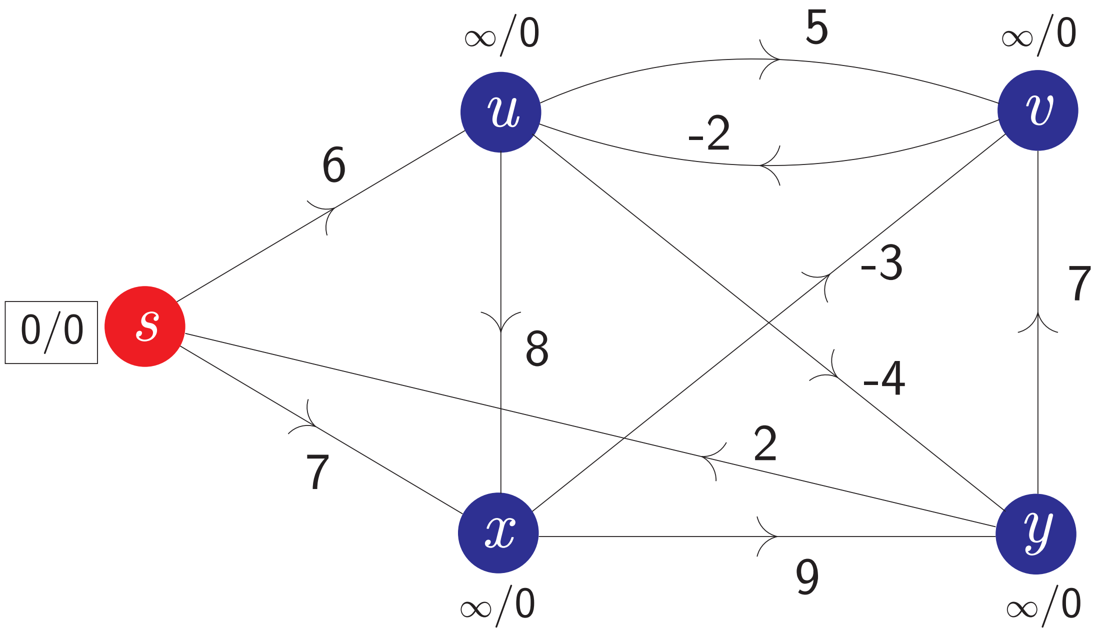
- **Time Complexity:** $\Theta(mn)$

The Bellmand-Ford Algorithm
Example

Find the SSSP for the following graph.



The Bellmand-Ford Algorithm
Solution - Iteration 0

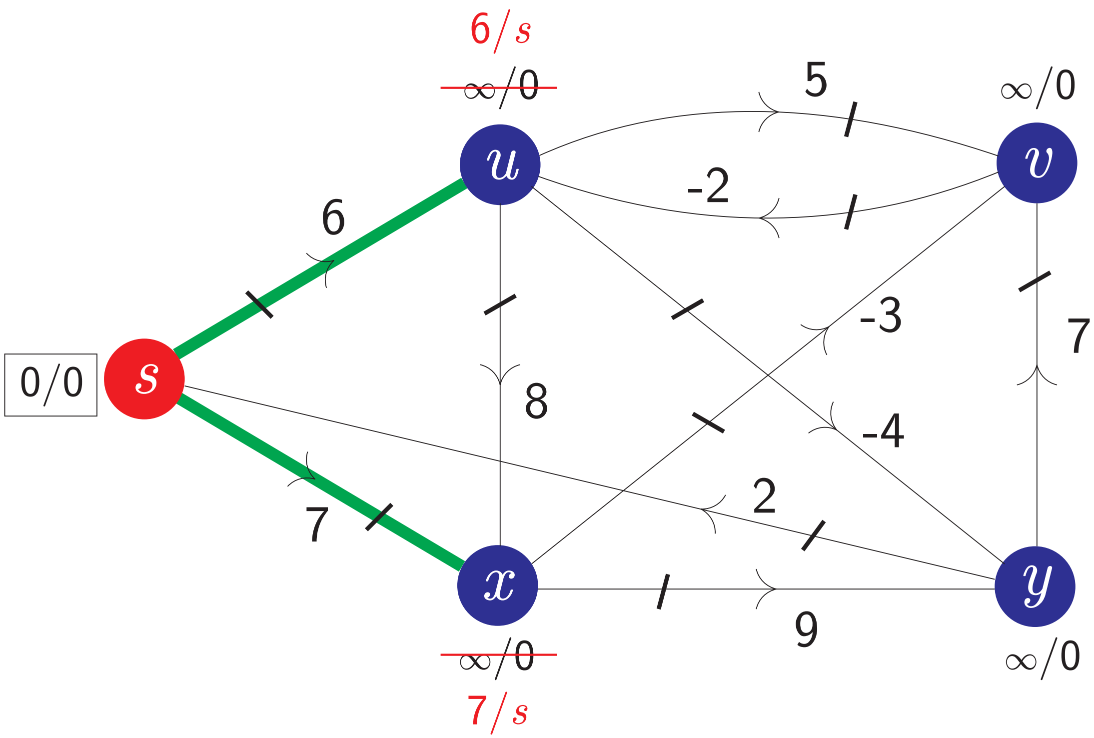


edge order

(u,v)
(u,x)
(u,y)
(v,u)
(x,v)
(x,y)
(y,v)
(y,s)
(s,u)
(s,x)

	0
s	0/0
u	$\infty/0$
v	$\infty/0$
x	$\infty/0$
y	$\infty/0$

The Bellmand-Ford Algorithm
Solution - Iteration 1



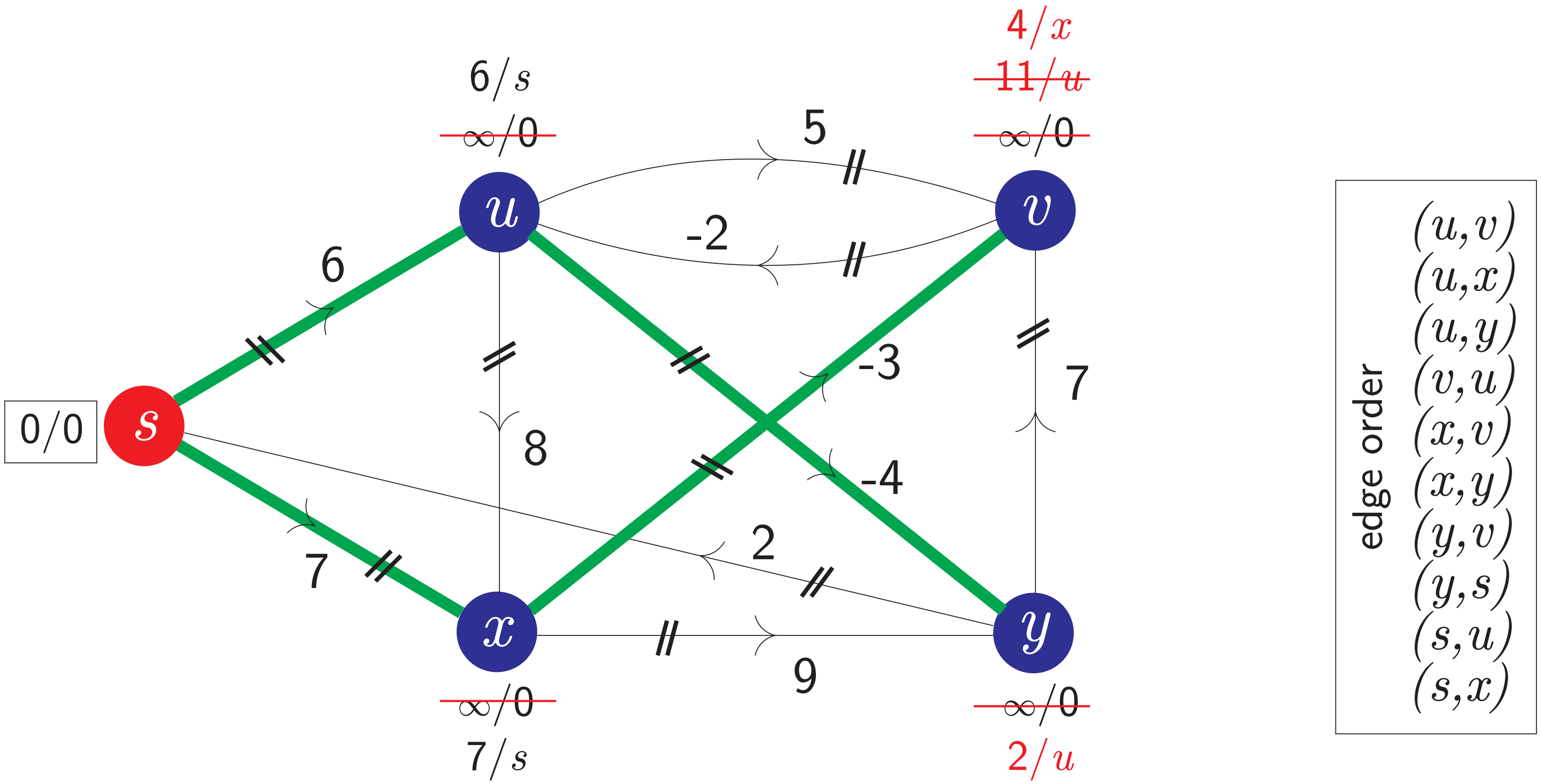
edge order

(u,v)
(u,x)
(u,y)
(v,u)
(x,v)
(x,y)
(y,v)
(y,s)
(s,u)
(s,x)

	0	**1**
s	0/0	0/0
u	∞/0	6/s
v	∞/0	∞/0
x	∞/0	7/s
y	∞/0	∞/0

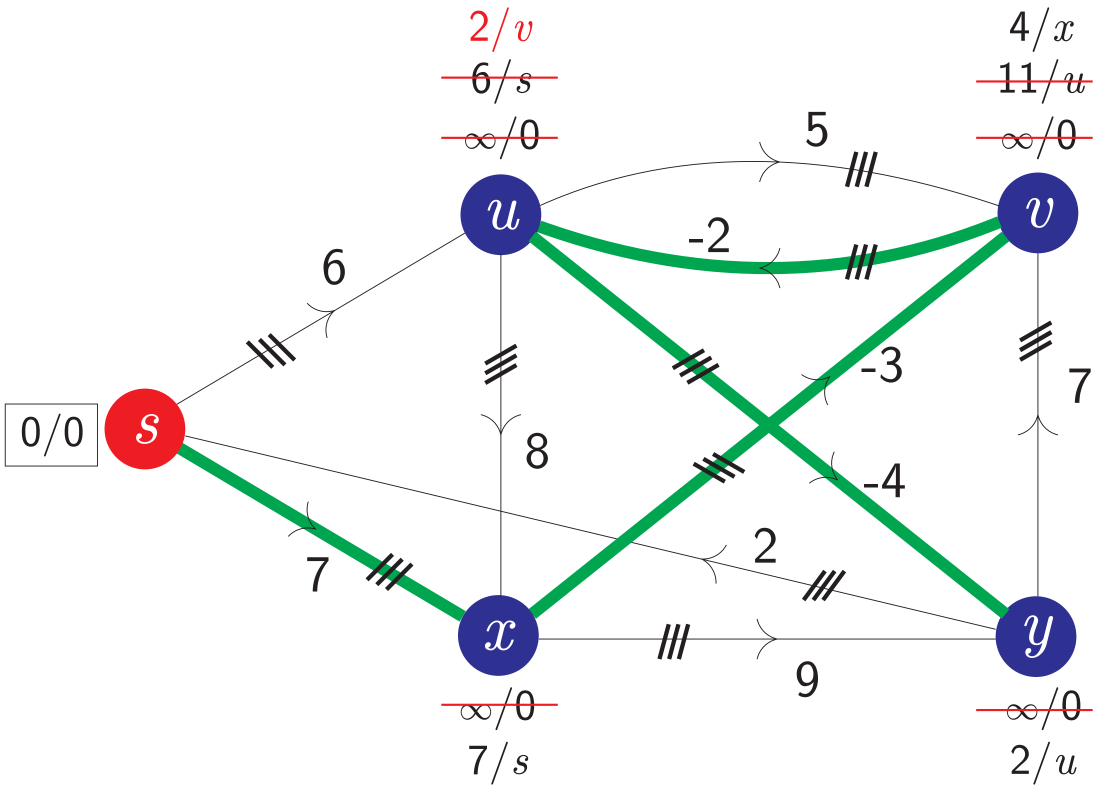
The Bellmand-Ford Algorithm

Solution - Iteration 2



	0	**1**	**2**
s	0/0	0/0	0/0
u	∞/0	6/s	6/s
v	∞/0	∞/0	4/x
x	∞/0	7/s	7/s
y	∞/0	∞/0	2/u

The Bellmand-Ford Algorithm
Solution - Iteration 3



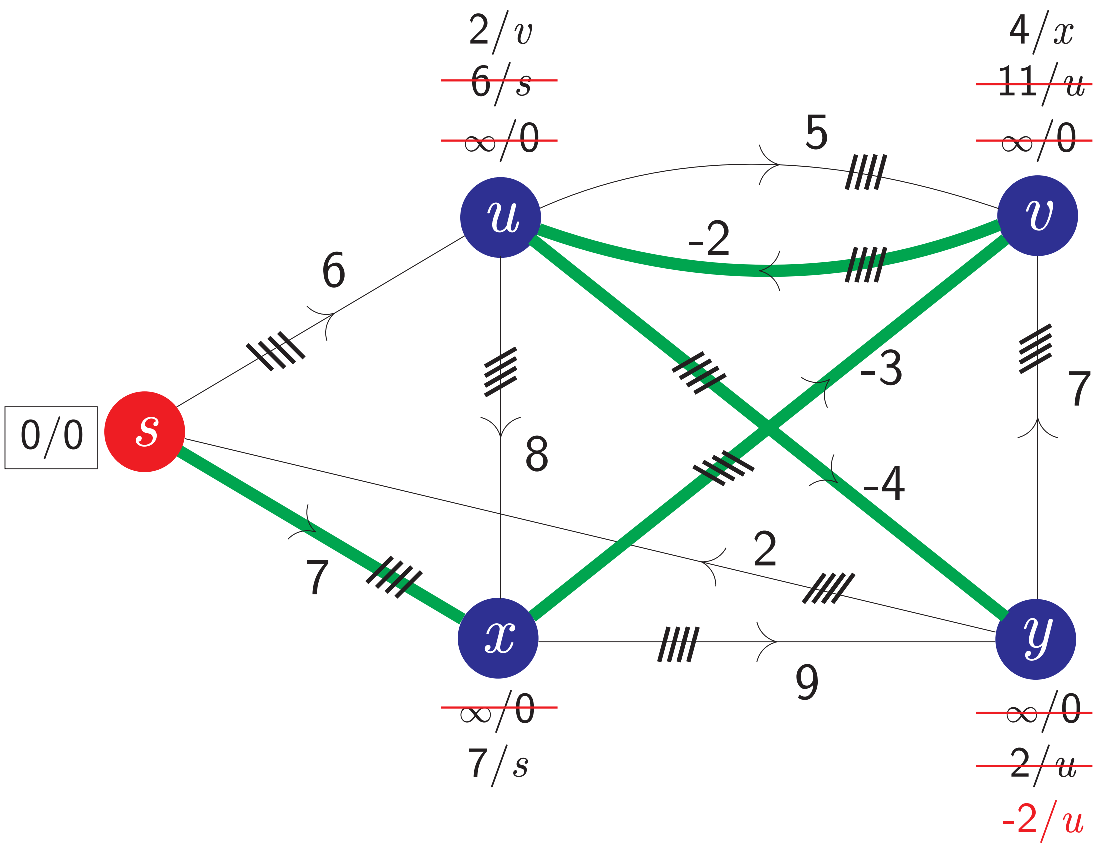
edge order

(u,v)
(u,x)
(u,y)
(v,u)
(x,v)
(x,y)
(y,v)
(y,s)
(s,u)
(s,x)

	0	1	2	3
s	0/0	0/0	0/0	0/0
u	$\infty/0$	$6/s$	$6/s$	$2/v$
v	$\infty/0$	$\infty/0$	$4/x$	$4/x$
x	$\infty/0$	$7/s$	$7/s$	$7/s$
y	$\infty/0$	$\infty/0$	$2/u$	$2/u$

The Bellmand-Ford Algorithm

Solution - Iteration 4

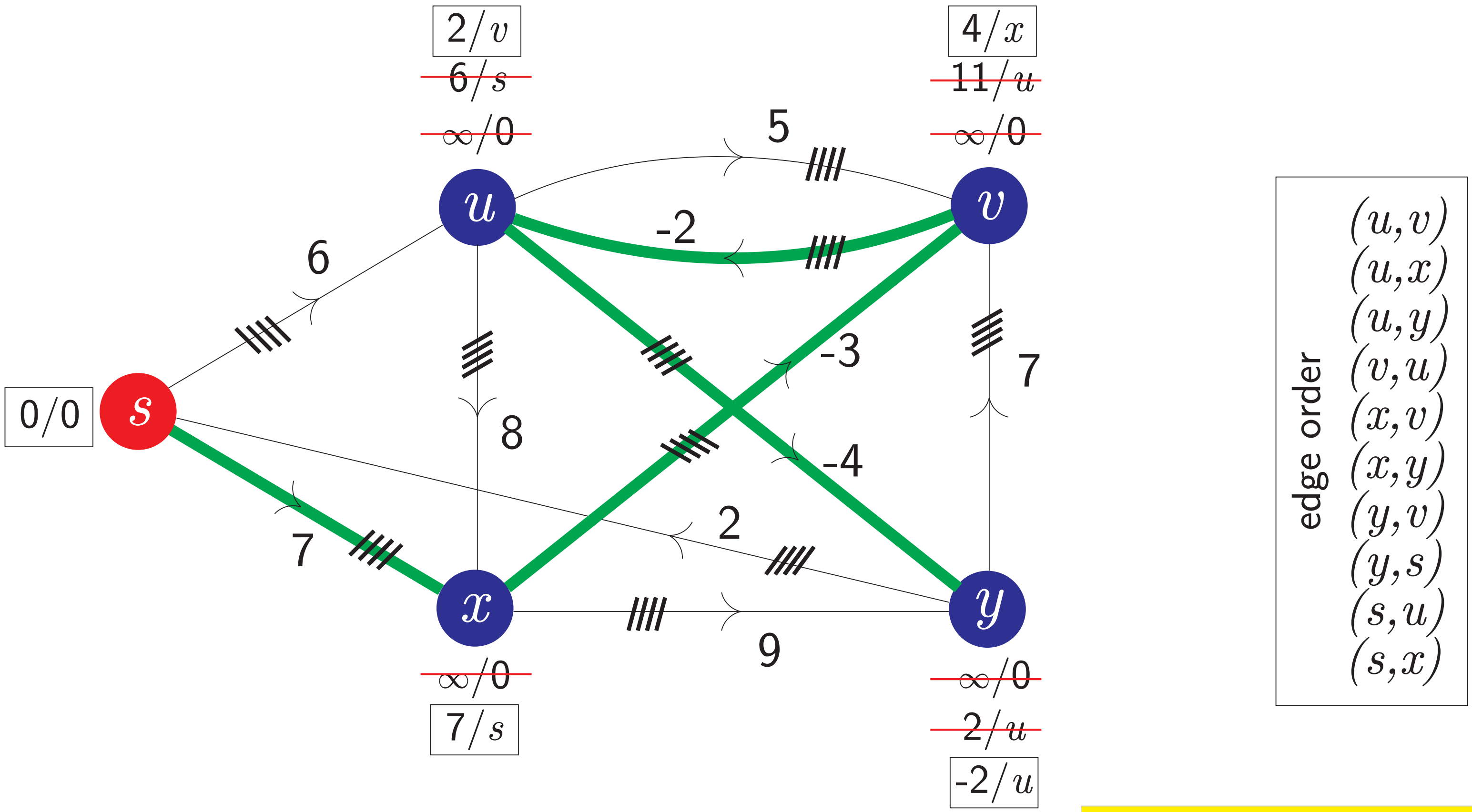


edge order

(u,v)
(u,x)
(u,y)
(v,u)
(x,v)
(x,y)
(y,v)
(y,s)
(s,u)
(s,x)

	0	**1**	**2**	**3**	**4**
s	0/0	0/0	0/0	0/0	0/0
u	∞/0	6/s	6/s	2/v	2/v
v	∞/0	∞/0	4/x	4/x	4/x
x	∞/0	7/s	7/s	7/s	7/s
y	∞/0	∞/0	2/u	2/u	-2/u

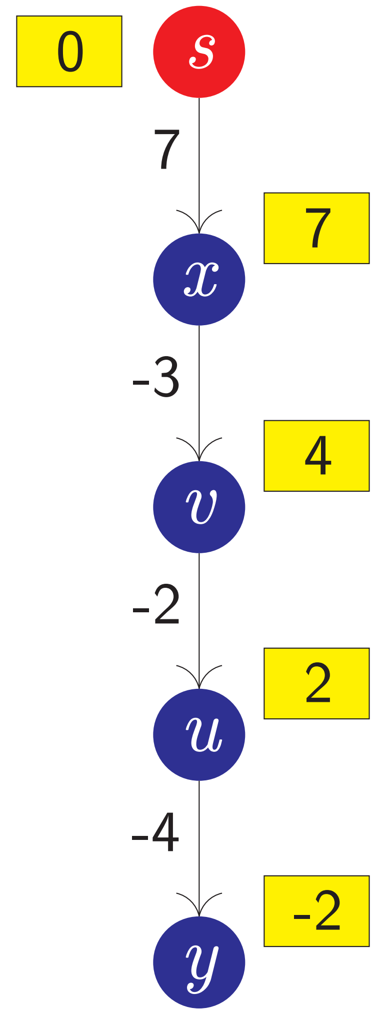
The Bellmand-Ford Algorithm
Solution - Iteration 4



	0	**1**	**2**	**3**	**4**
s	$0/0$	$0/0$	$0/0$	$0/0$	$0/0$
u	$\infty/0$	$6/s$	$6/s$	$2/v$	$2/v$
v	$\infty/0$	$\infty/0$	$4/x$	$4/x$	$4/x$
x	$\infty/0$	$7/s$	$7/s$	$7/s$	$7/s$
y	$\infty/0$	$\infty/0$	$2/u$	$2/u$	$-2/u$

	s	u	v	x	y	
$d=$	0	2	4	7	-2	$\delta(s,u)$
$\pi=$	0	v	x	s	u	$PARENT[u]$

The Bellmand-Ford Algorithm
Solution



	s	u	v	x	y	
$d=$	0	2	4	7	-2	$\delta(s,u)$

	s	u	v	x	y	
$\pi=$	0	v	x	s	u	$PARENT[u]$

The Bellman-Ford Algorithm returns TRUE in this example.