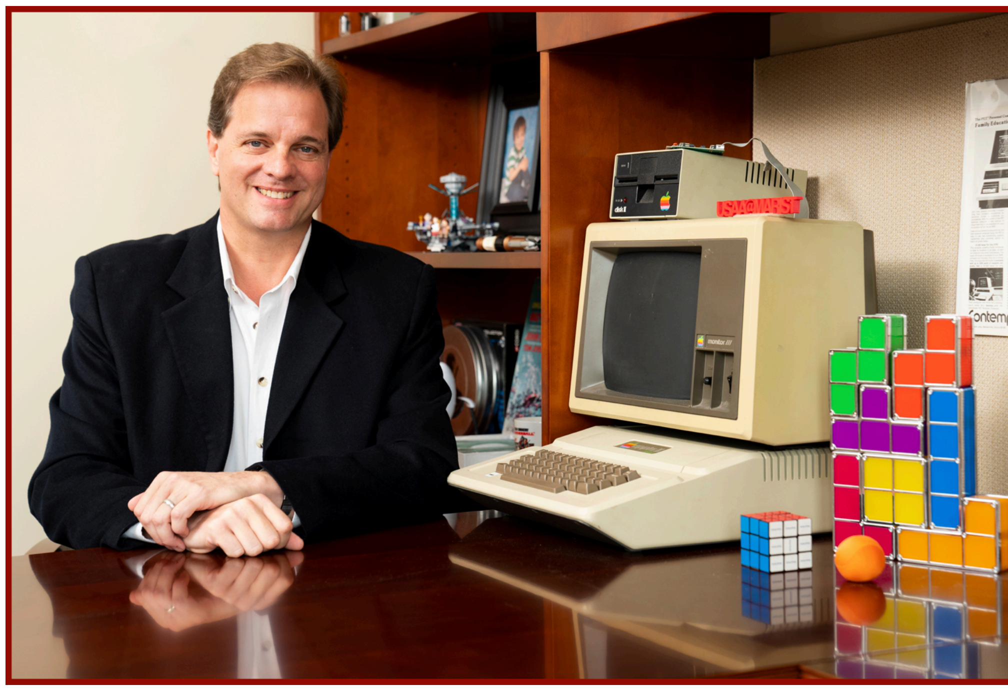
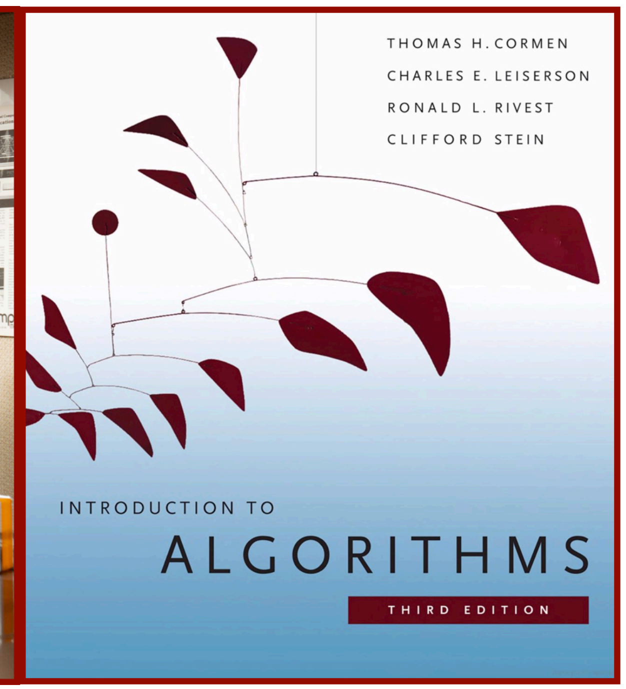
Bellman-Ford SSSP





Alan G. Labouseur, Ph.D.
Alan.Labouseur@Marist.edu

Bellman-Ford SSSP

The Bellman-Ford
Single-Source Shortest Path
Algorithm

CLRS chapter 24.1

Graph G

a weighted, directed graph with negative edge weights

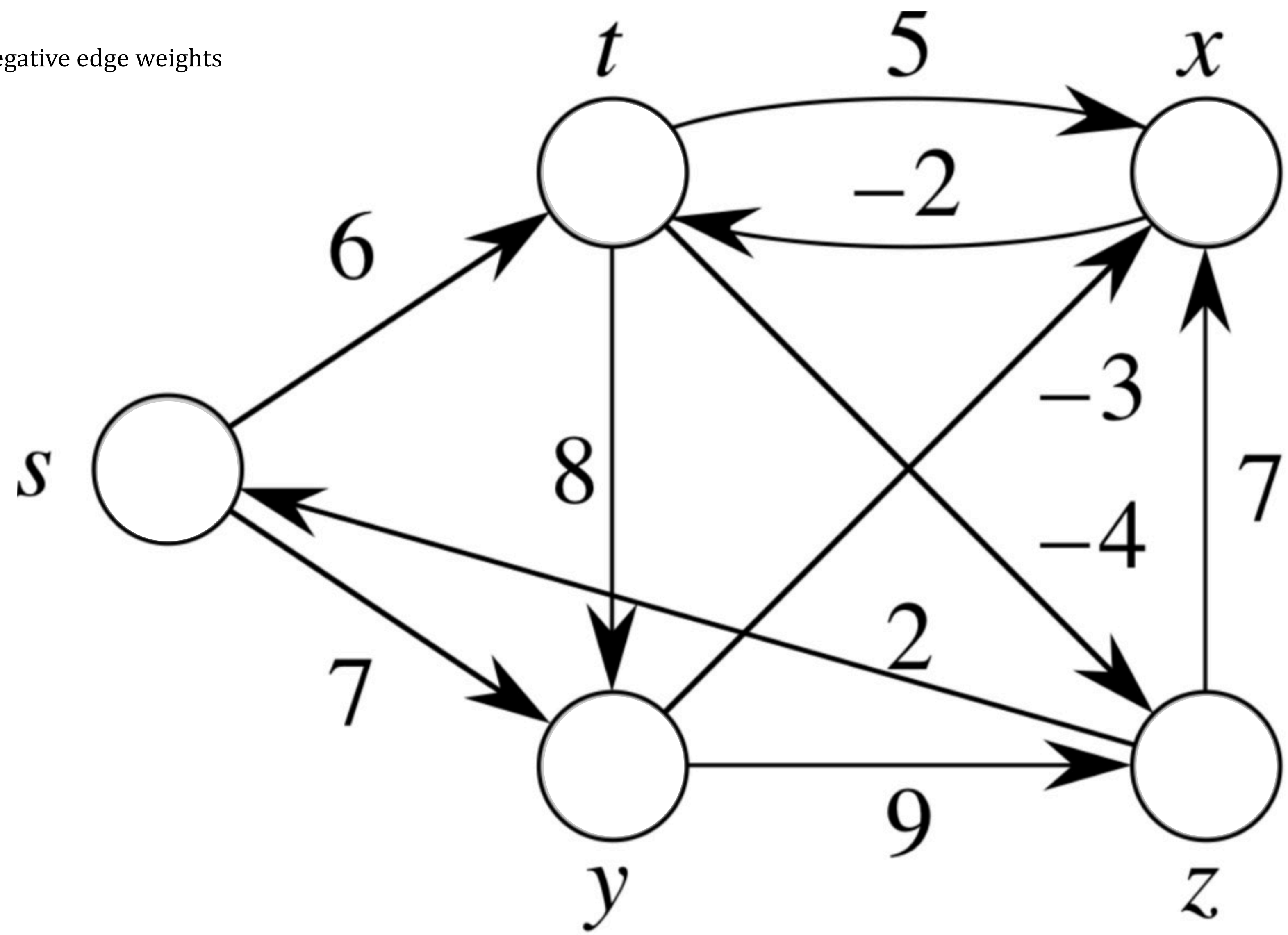
G.V = s, t, x, y, z
G.E = (t, x : *w* = 5)
(t, y : *w* = 8)
(t, z : *w* = -4)
(x, t : *w* = -2)
(y, x : *w* = -3)
(y, z : *w* = 9)
(z, x : *w* = 7)
(z, s : *w* = 2)
(s, t : *w* = 6)
(s, y : *w* = 7)

The Bellman-Ford Single-Source Shortest Path Algorithm

Graph G

a weighted, directed graph with negative edge weights

G.V = s, t, x, y, z
G.E = (t, x : *w* = 5)
(t, y : *w* = 8)
(t, z : *w* = -4)
(x, t : *w* = -2)
(y, x : *w* = -3)
(y, z : *w* = 9)
(z, x : *w* = 7)
(z, s : *w* = 2)
(s, t : *w* = 6)
(s, y : *w* = 7)



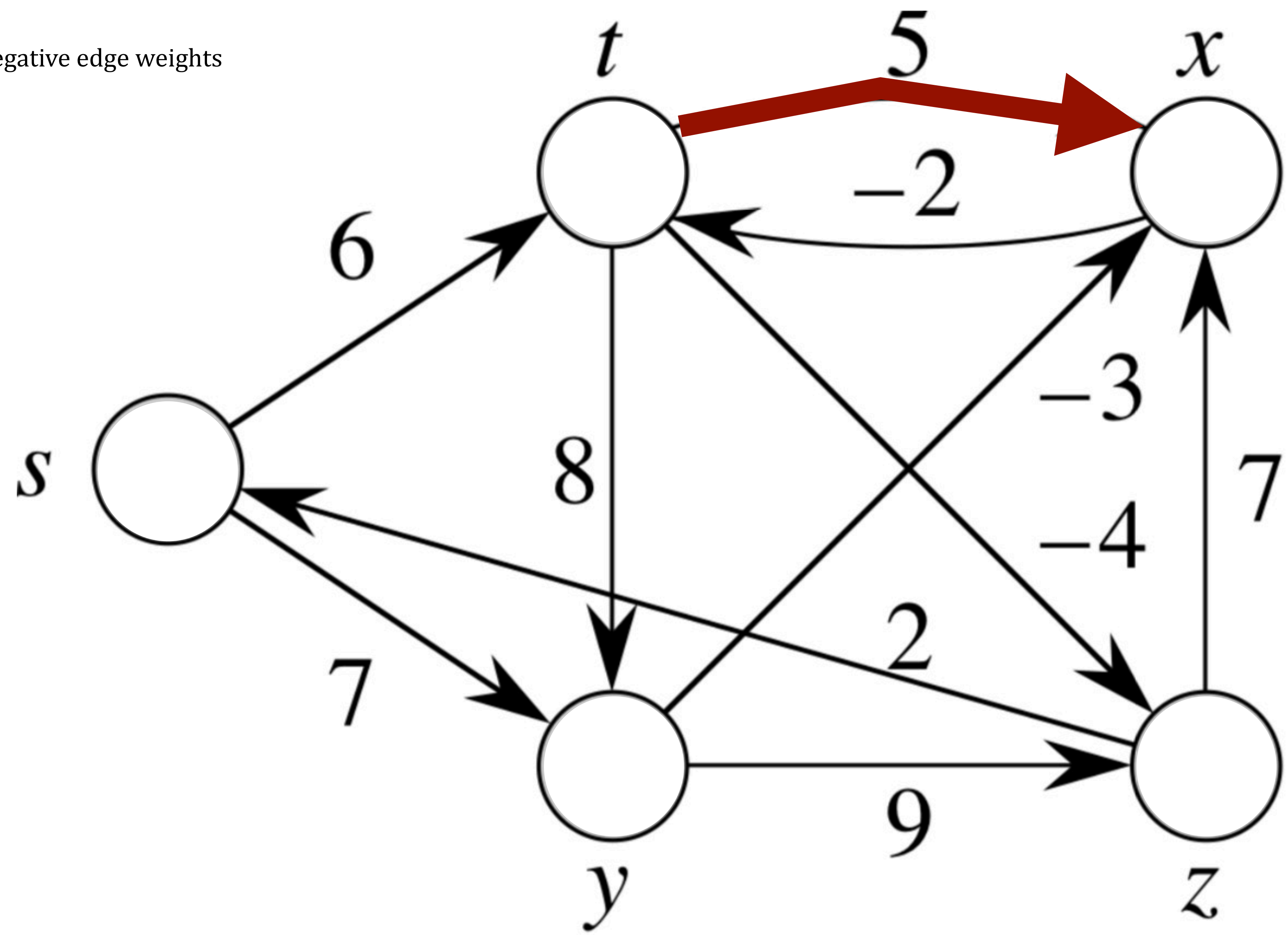
The Bellman-Ford Single-Source Shortest Path Algorithm

Graph G

a weighted, directed graph with negative edge weights

G.V = s, t, x, y, z

G.E = (t, x : *w* = 5)
(t, y : *w* = 8)
(t, z : *w* = -4)
(x, t : *w* = -2)
(y, x : *w* = -3)
(y, z : *w* = 9)
(z, x : *w* = 7)
(z, s : *w* = 2)
(s, t : *w* = 6)
(s, y : *w* = 7)



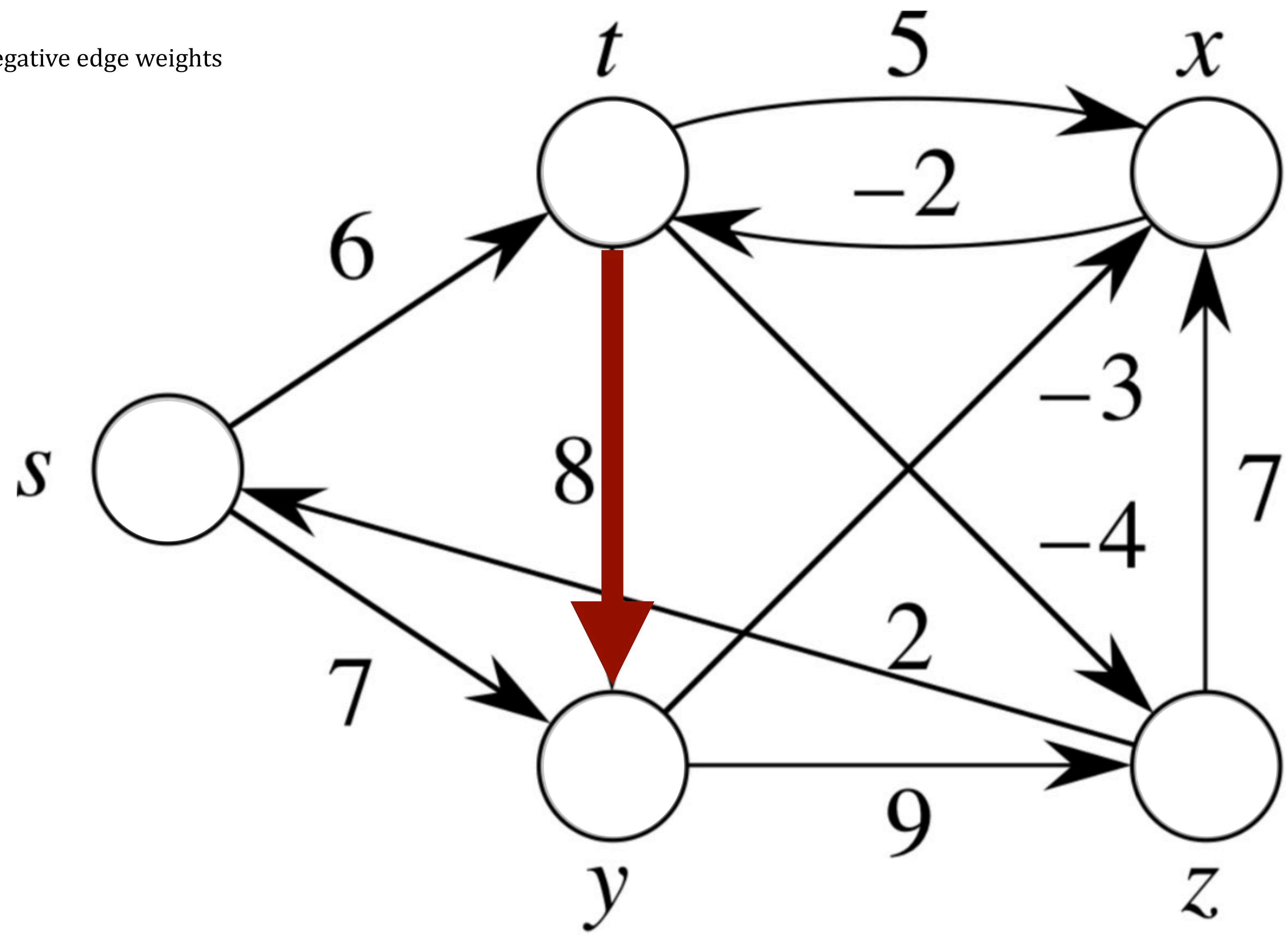
The Bellman-Ford Single-Source Shortest Path Algorithm

Graph G

a weighted, directed graph with negative edge weights

G.V = s, t, x, y, z

G.E = (t, x : w = 5)
(t, y : w = 8)
(t, z : w = -4)
(x, t : w = -2)
(y, x : w = -3)
(y, z : w = 9)
(z, x : w = 7)
(z, s : w = 2)
(s, t : w = 6)
(s, y : w = 7)

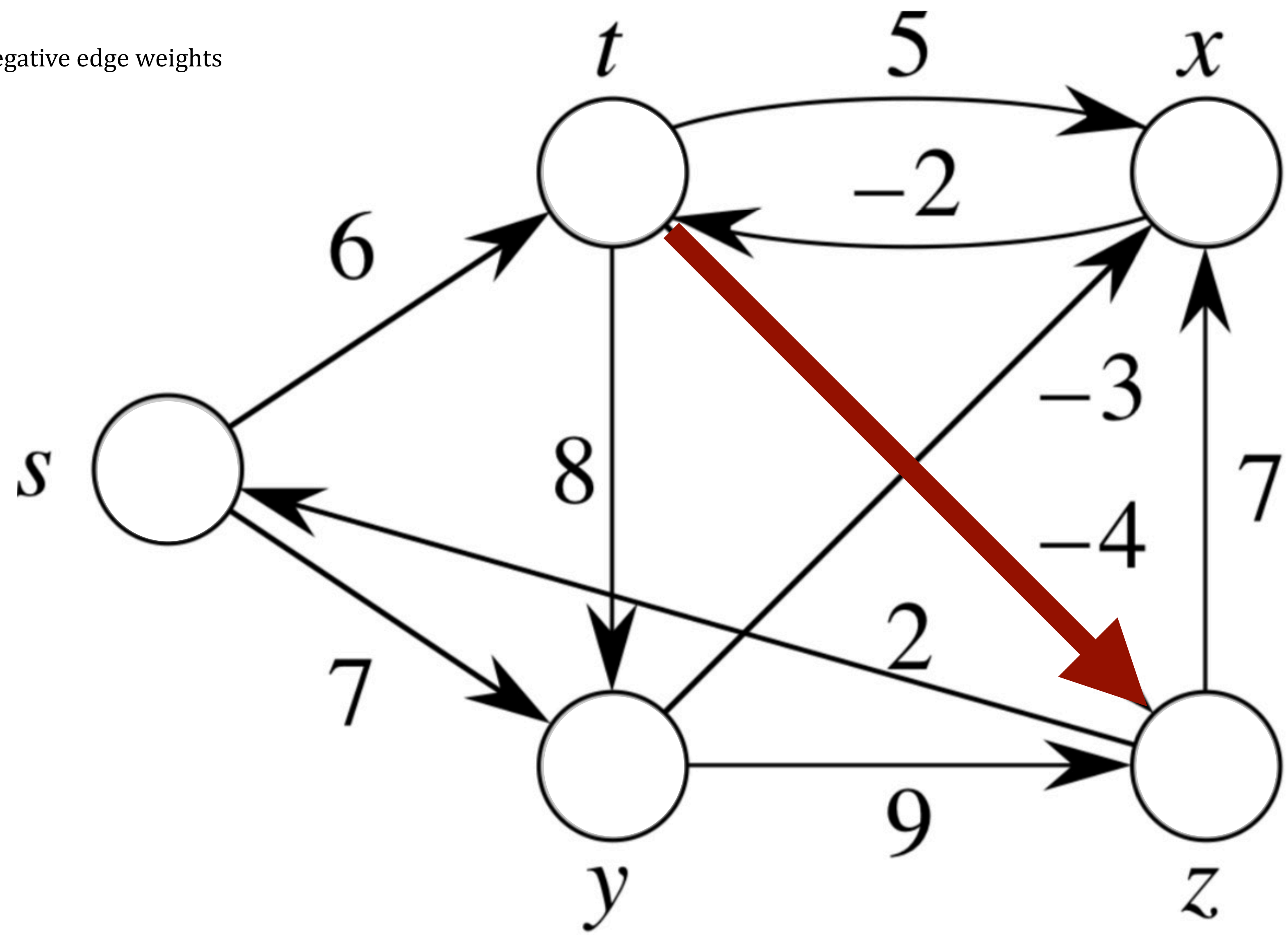


The Bellman-Ford Single-Source Shortest Path Algorithm

Graph G

a weighted, directed graph with negative edge weights

G.V = s, t, x, y, z
G.E = (t, x : *w* = 5)
(t, y : *w* = 8)
(t, z : *w* = -4)
(x, t : *w* = -2)
(y, x : *w* = -3)
(y, z : *w* = 9)
(z, x : *w* = 7)
(z, s : *w* = 2)
(s, t : *w* = 6)
(s, y : *w* = 7)



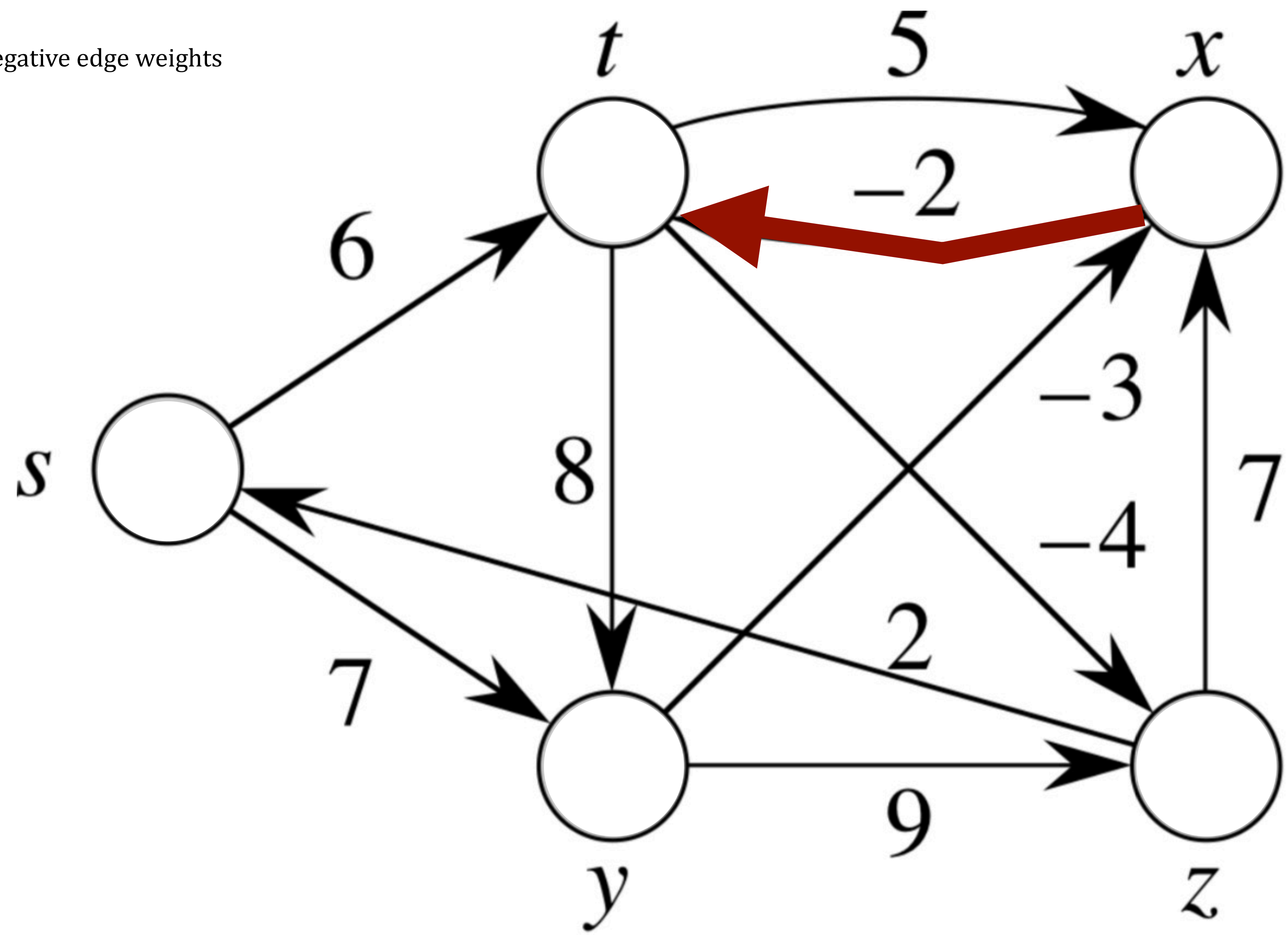
The Bellman-Ford Single-Source Shortest Path Algorithm

Graph G

a weighted, directed graph with negative edge weights

G.V = s, t, x, y, z

G.E = (t, x : w = 5)
(t, y : w = 8)
(t, z : w = -4)
(x, t : w = -2)
(y, x : w = -3)
(y, z : w = 9)
(z, x : w = 7)
(z, s : w = 2)
(s, t : w = 6)
(s, y : w = 7)



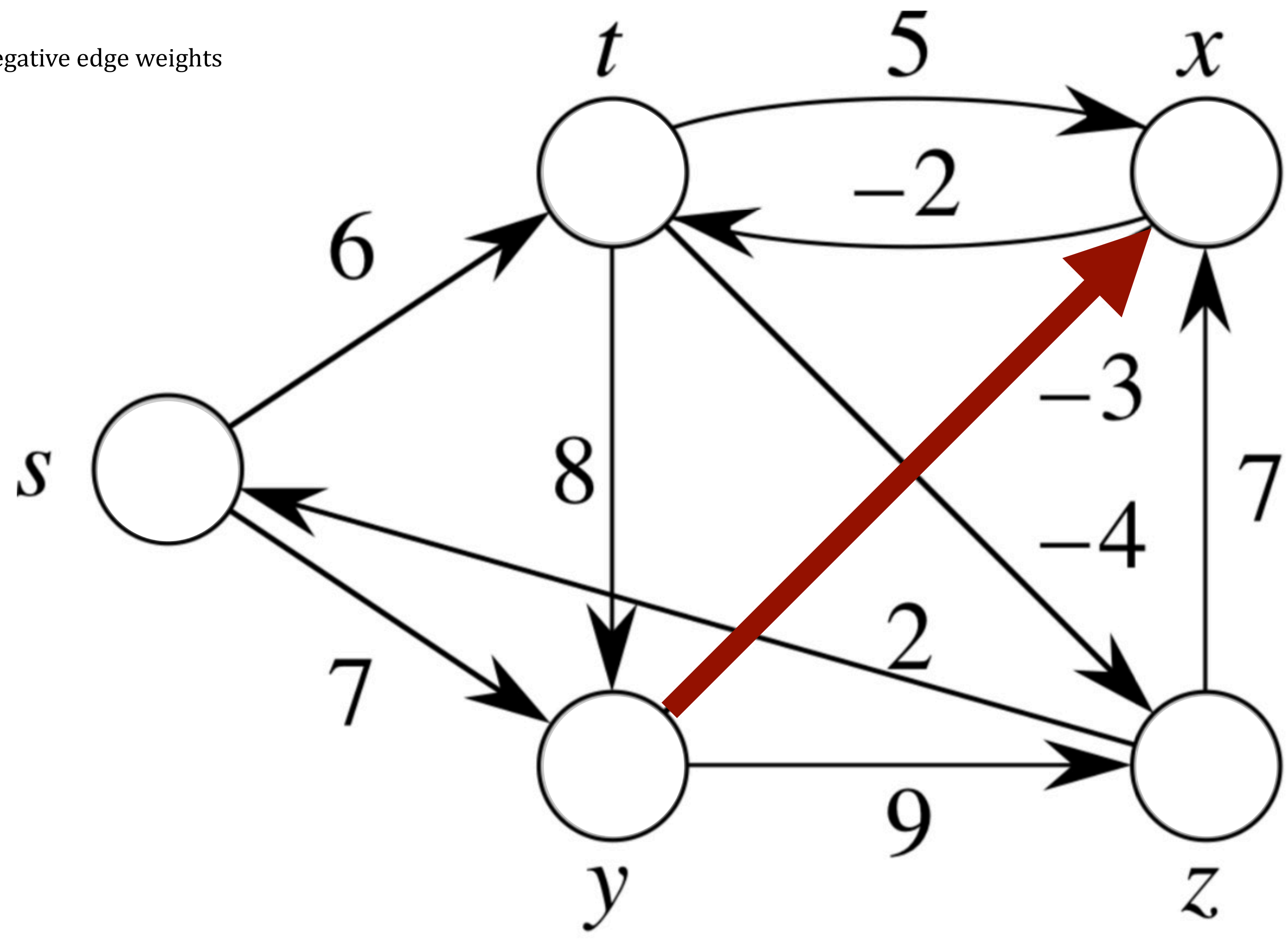
The Bellman-Ford Single-Source Shortest Path Algorithm

Graph G

a weighted, directed graph with negative edge weights

G.V = s, t, x, y, z

G.E = (t, x : w = 5)
(t, y : w = 8)
(t, z : w = -4)
(x, t : w = -2)
(y, x : w = -3)
(y, z : w = 9)
(z, x : w = 7)
(z, s : w = 2)
(s, t : w = 6)
(s, y : w = 7)

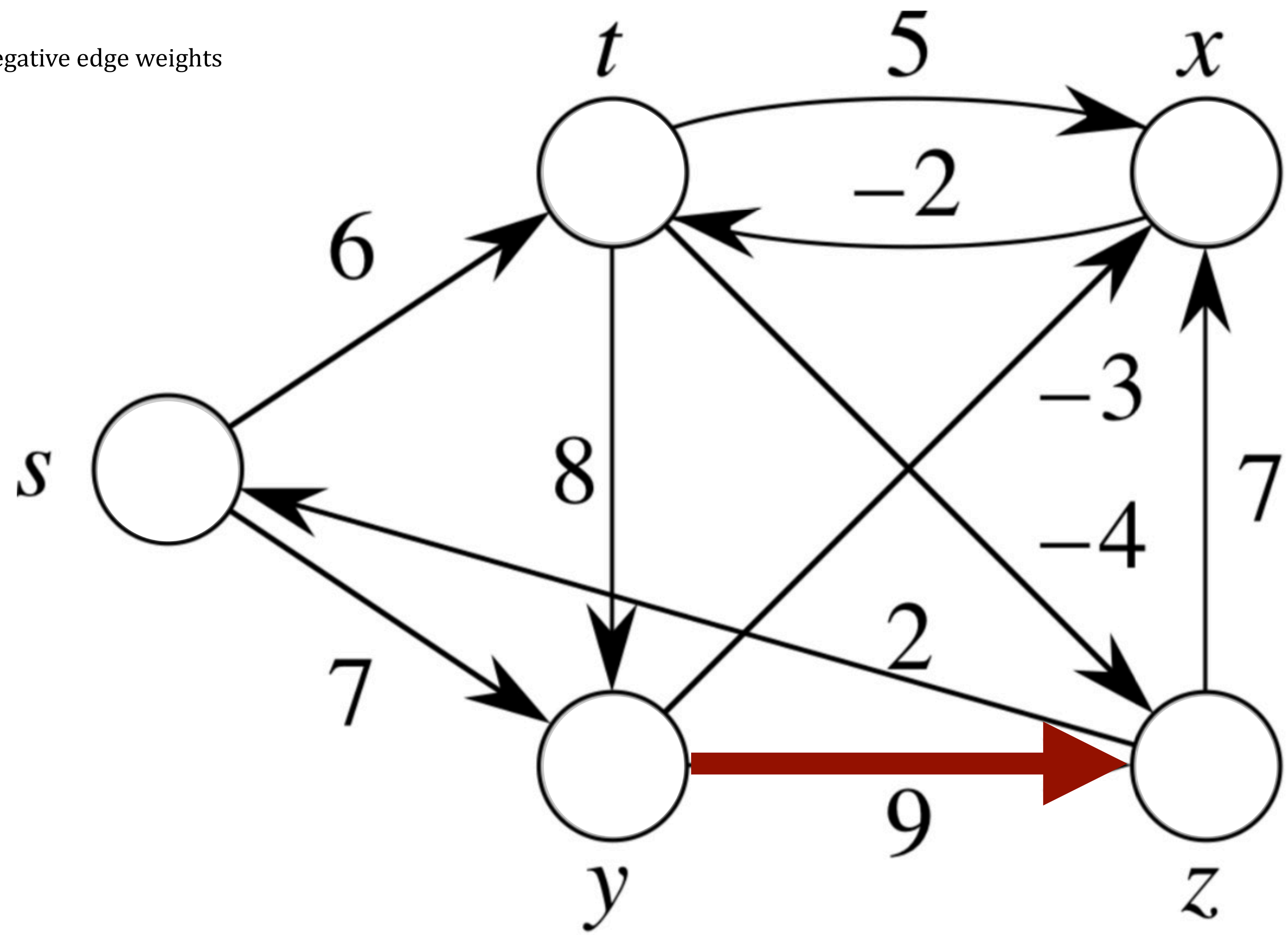


The Bellman-Ford Single-Source Shortest Path Algorithm

Graph G

a weighted, directed graph with negative edge weights

G.V = s, t, x, y, z
G.E = (t, x : *w* = 5)
(t, y : *w* = 8)
(t, z : *w* = -4)
(x, t : *w* = -2)
(y, x : *w* = -3)
(y, z : *w* = 9)
(z, x : *w* = 7)
(z, s : *w* = 2)
(s, t : *w* = 6)
(s, y : *w* = 7)



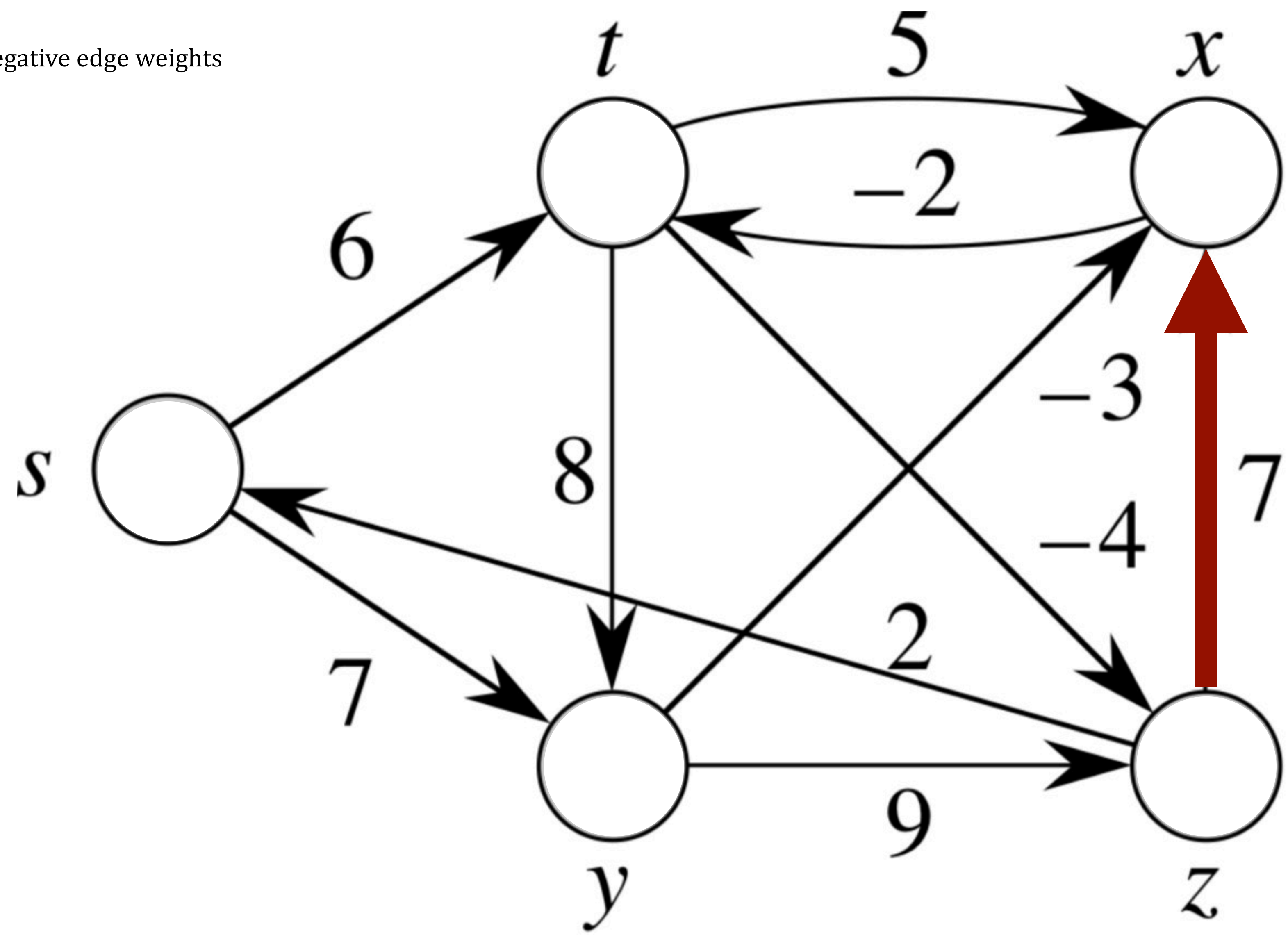
The Bellman-Ford Single-Source Shortest Path Algorithm

Graph G

a weighted, directed graph with negative edge weights

G.V = s, t, x, y, z

G.E = (t, x : w = 5)
(t, y : w = 8)
(t, z : w = -4)
(x, t : w = -2)
(y, x : w = -3)
(y, z : w = 9)
(z, x : w = 7)
(z, s : w = 2)
(s, t : w = 6)
(s, y : w = 7)



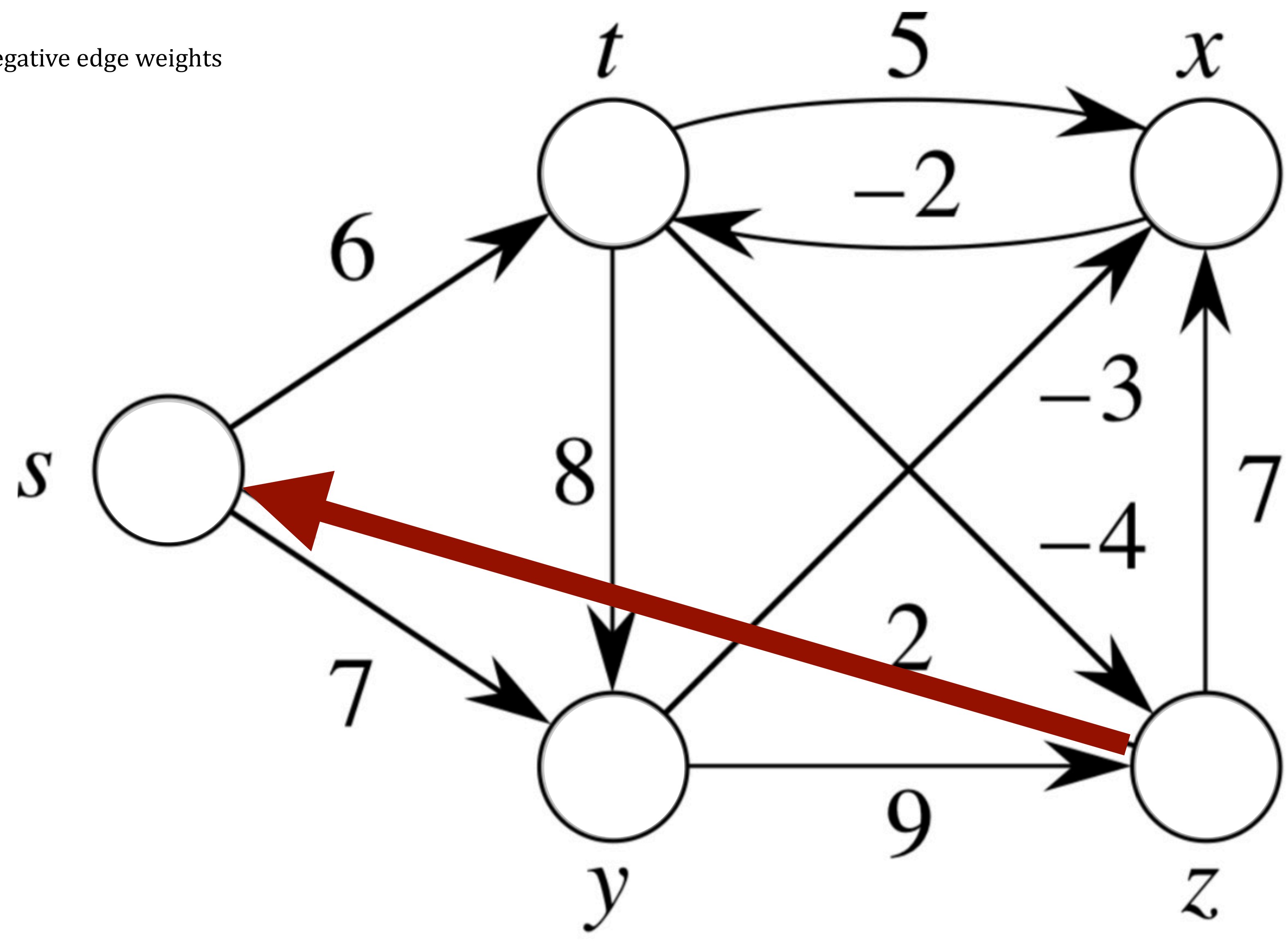
The Bellman-Ford Single-Source Shortest Path Algorithm

Graph G

a weighted, directed graph with negative edge weights

G.V = s, t, x, y, z

G.E = (t, x : w = 5)
(t, y : w = 8)
(t, z : w = -4)
(x, t : w = -2)
(y, x : w = -3)
(y, z : w = 9)
(z, x : w = 7)
(z, s : w = 2)
(s, t : w = 6)
(s, y : w = 7)



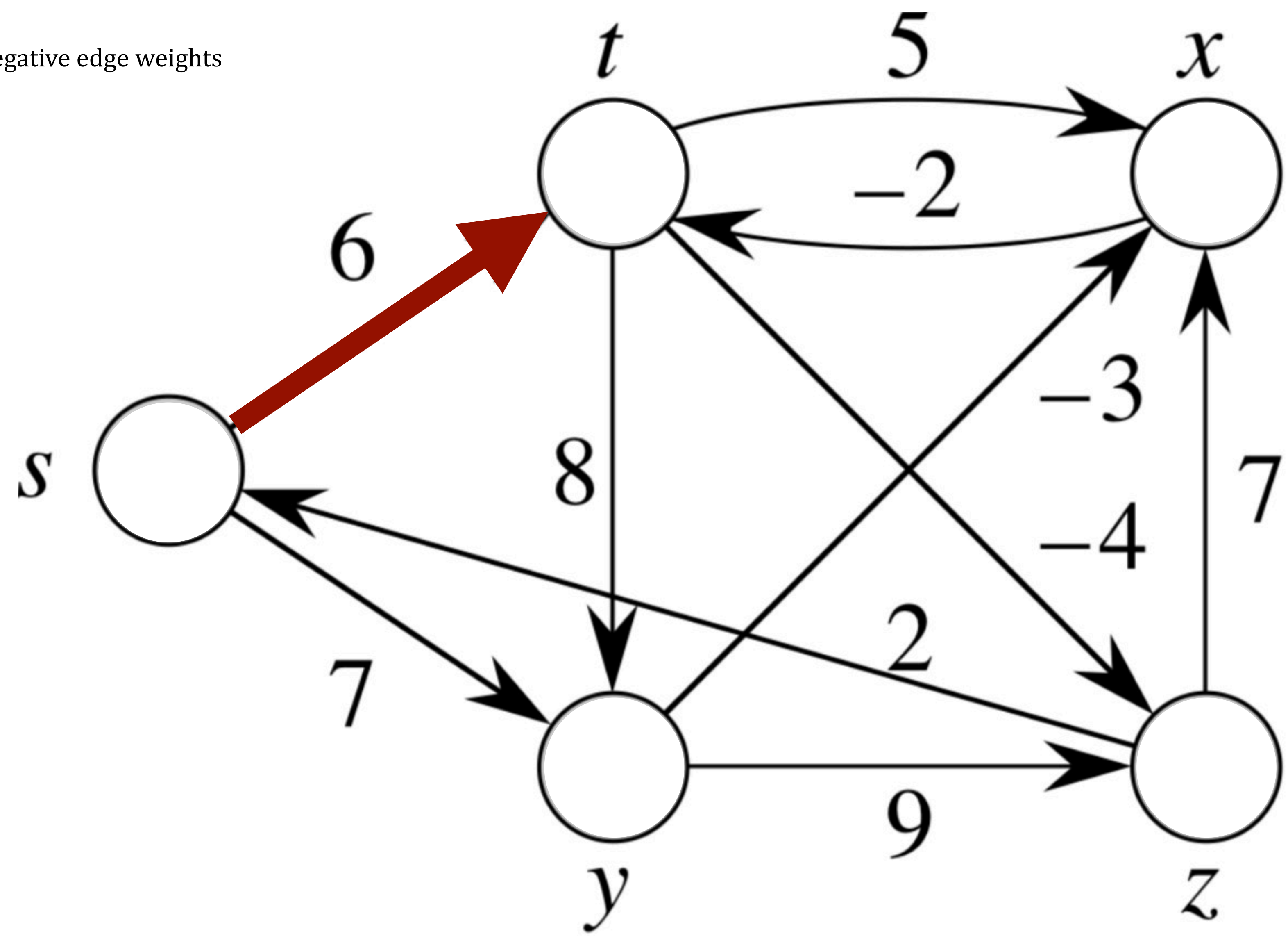
The Bellman-Ford Single-Source Shortest Path Algorithm

Graph G

a weighted, directed graph with negative edge weights

G.V = s, t, x, y, z

G.E = (t, x : *w* = 5)
(t, y : *w* = 8)
(t, z : *w* = -4)
(x, t : *w* = -2)
(y, x : *w* = -3)
(y, z : *w* = 9)
(z, x : *w* = 7)
(z, s : *w* = 2)
(s, t : *w* = 6)
(s, y : *w* = 7)

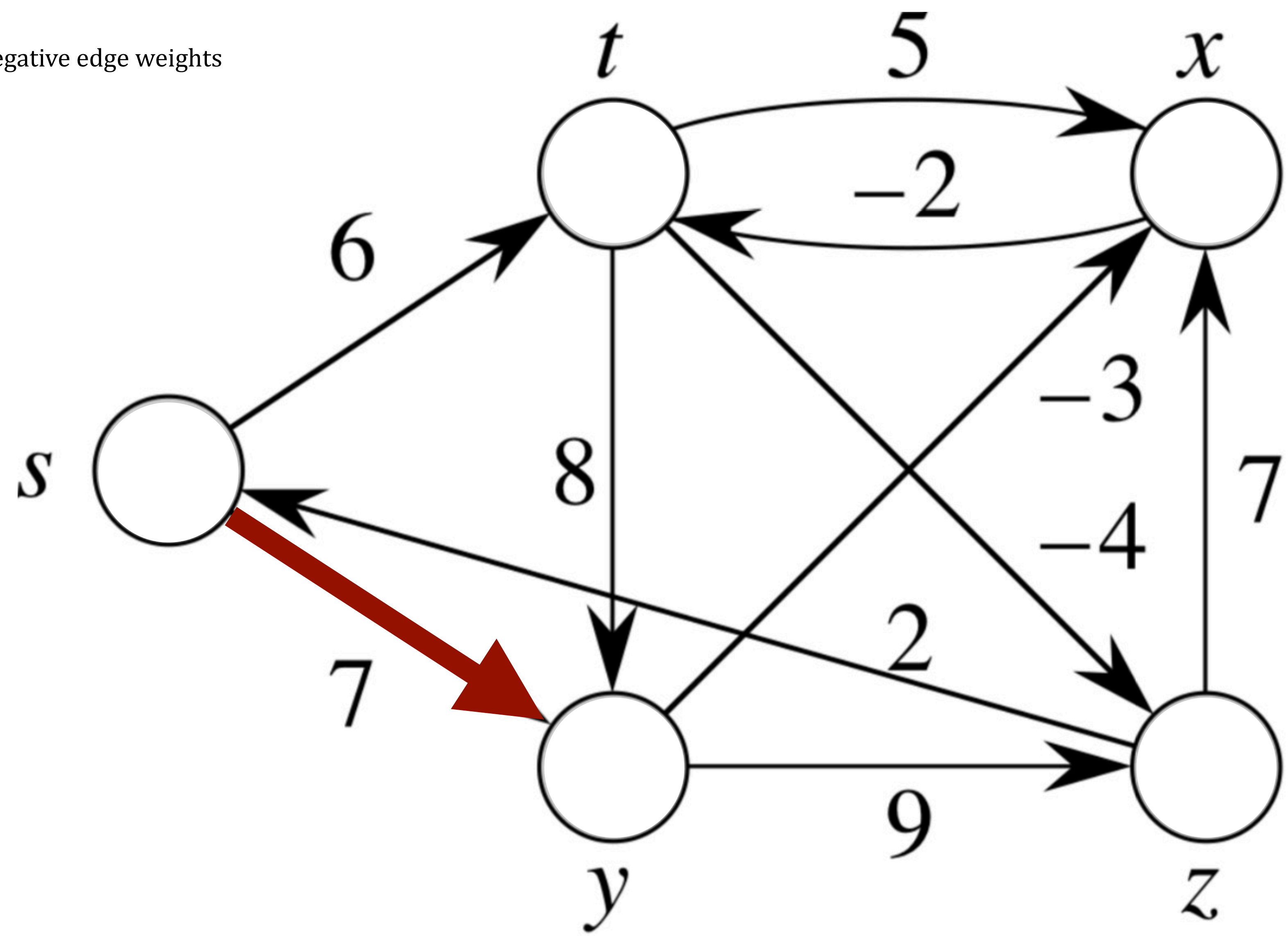


The Bellman-Ford Single-Source Shortest Path Algorithm

Graph G

a weighted, directed graph with negative edge weights

G.V = s, t, x, y, z
G.E = (t, x : w = 5)
(t, y : w = 8)
(t, z : w = -4)
(x, t : w = -2)
(y, x : w = -3)
(y, z : w = 9)
(z, x : w = 7)
(z, s : w = 2)
(s, t : w = 6)
(s, y : w = 7)

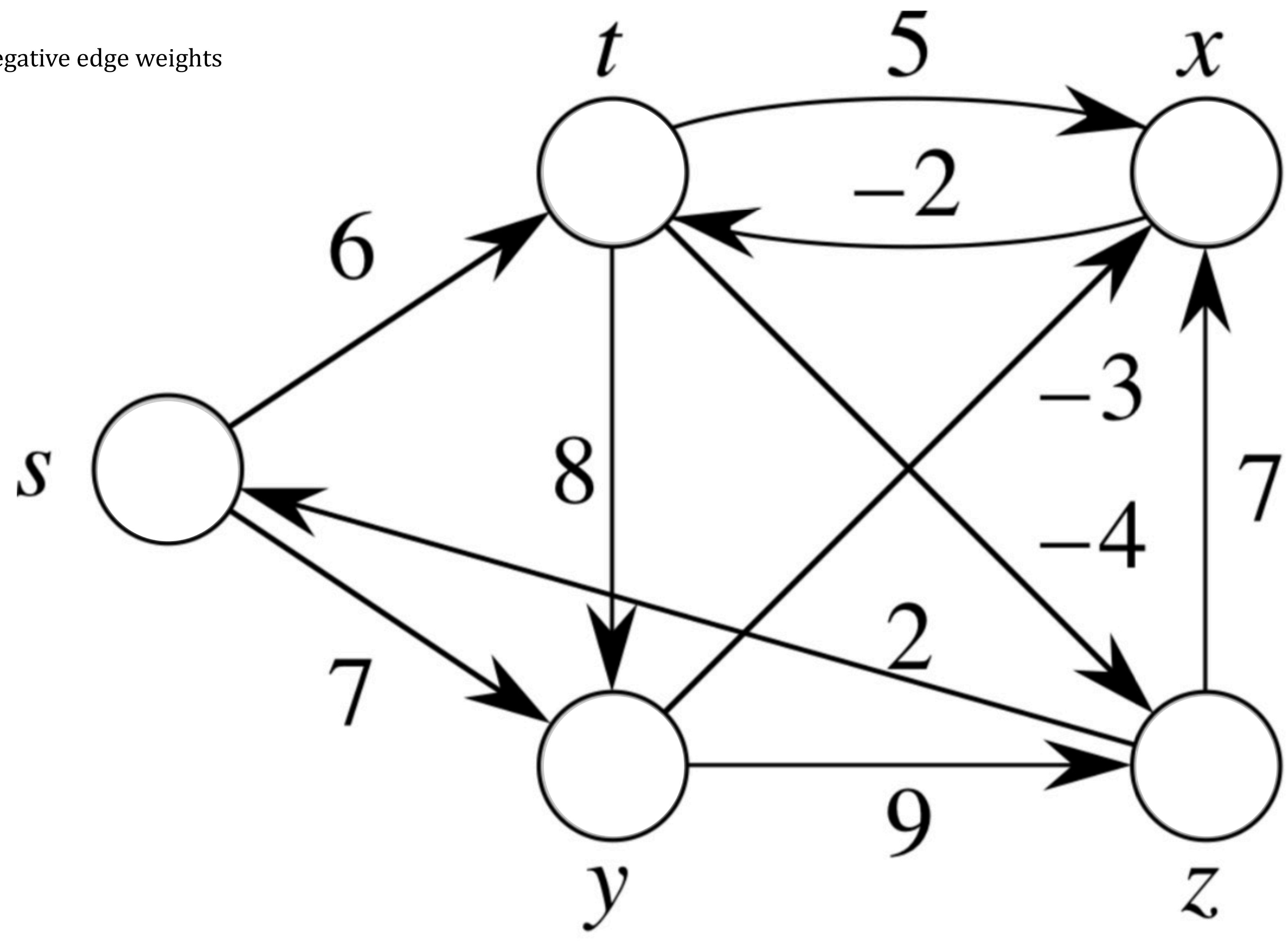


The Bellman-Ford Single-Source Shortest Path Algorithm

Graph G

a weighted, directed graph with negative edge weights

G.V = s, t, x, y, z
G.E = (t, x : *w* = 5)
(t, y : *w* = 8)
(t, z : *w* = -4)
(x, t : *w* = -2)
(y, x : *w* = -3)
(y, z : *w* = 9)
(z, x : *w* = 7)
(z, s : *w* = 2)
(s, t : *w* = 6)
(s, y : *w* = 7)



Ready?

Let's go!

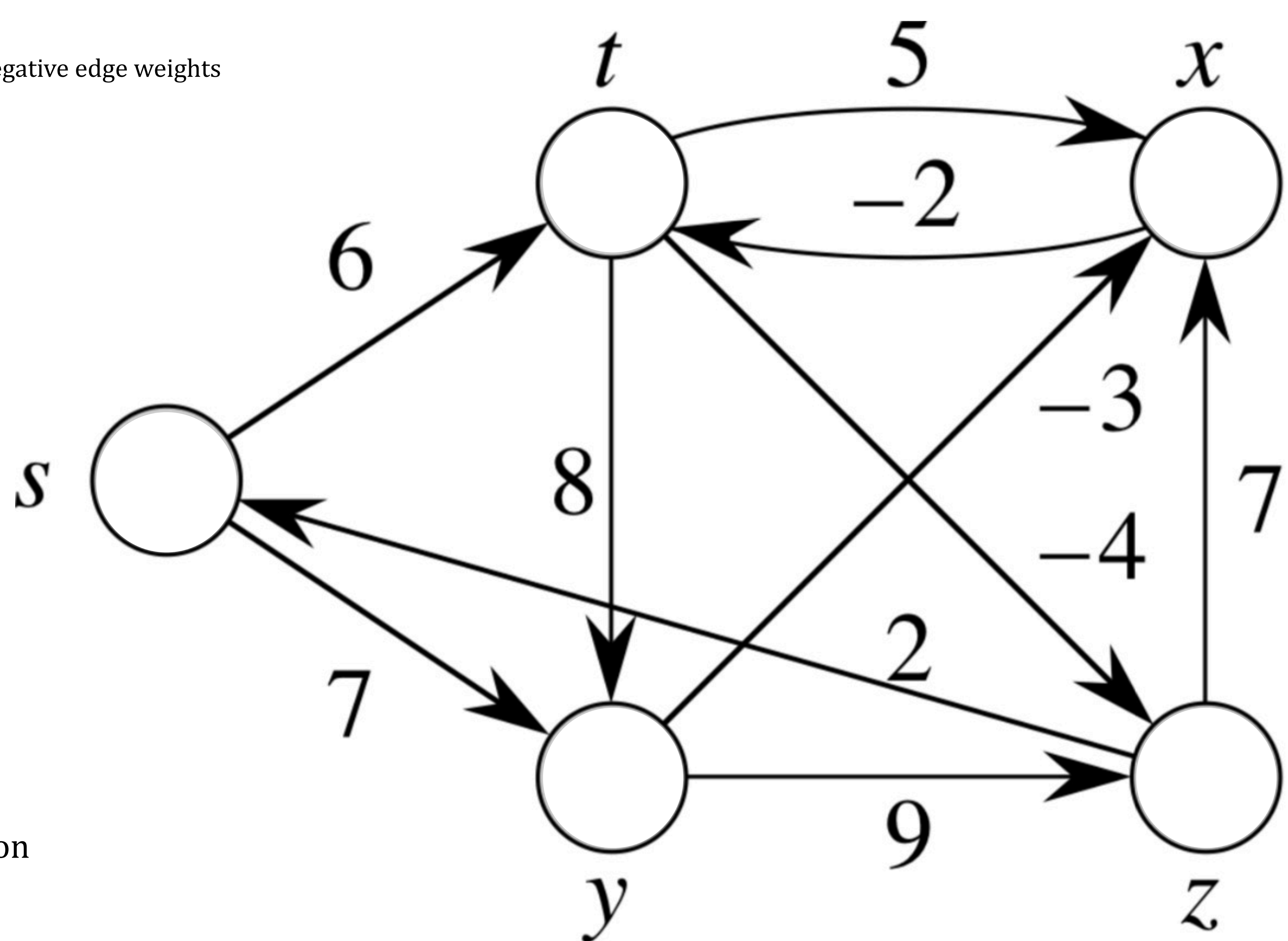
The Bellman-Ford Single-Source Shortest Path Algorithm

Graph G

a weighted, directed graph with negative edge weights

G.V = `s`, `t`, `x`, `y`, `z`

G.E = (`t`, `x` : w = 5)
(`t`, `y` : w = 8)
(`t`, `z` : w = -4)
(`x`, `t` : w = -2)
(`y`, `x` : w = -3)
(`y`, `z` : w = 9)
(`z`, `x` : w = 7)
(`z`, `s` : w = 2)
(`s`, `t` : w = 6)
(`s`, `y` : w = 7)



Weight function

Graph   Source vertex

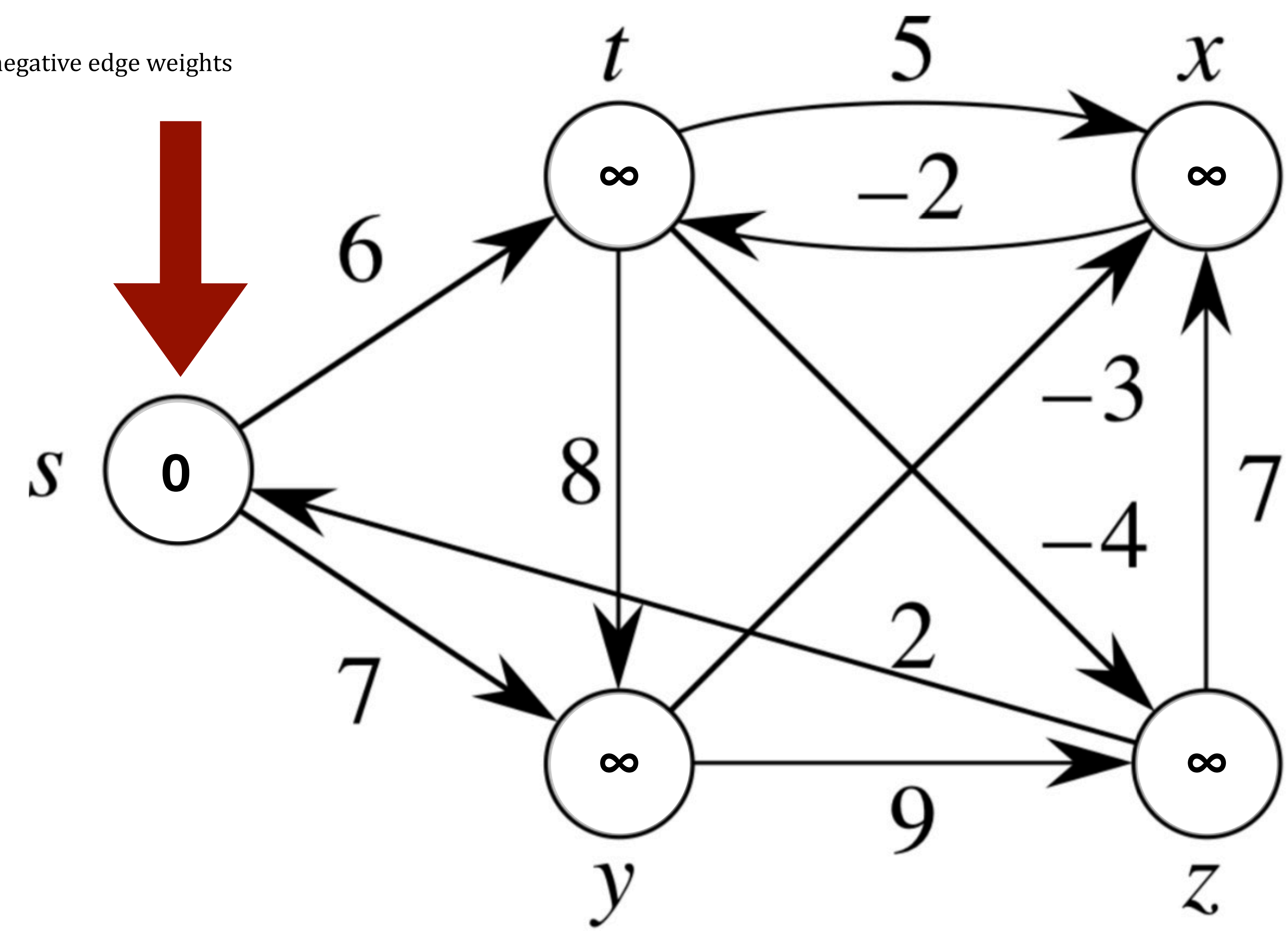
BELLMAN-FORD(G, w, s)
  INIT-SINGLE-SOURCE(G, s)
  for $i = 1$ to $|G.V| - 1$
    for each edge $(u, v) \in G.E$
      RELAX(u, v, w)
  for each edge $(u, v) \in G.E$
    if $v.d > u.d + w(u, v)$
      return FALSE
  return TRUE

The Bellman-Ford Single-Source Shortest Path Algorithm

Graph G

a weighted, directed graph with negative edge weights

G.V = s, t, x, y, z
G.E = (t, x : w = 5)
(t, y : w = 8)
(t, z : w = -4)
(x, t : w = -2)
(y, x : w = -3)
(y, z : w = 9)
(z, x : w = 7)
(z, s : w = 2)
(s, t : w = 6)
(s, y : w = 7)



BELLMAN-FORD(G, w, s)

```
INIT-SINGLE-SOURCE(G, s)
for i = 1 to |G.V| − 1
    for each edge (u, v) ∈ G.E
        RELAX(u, v, w)
for each edge (u, v) ∈ G.E
    if v.d > u.d + w(u, v)
        return FALSE
return TRUE
```

INIT-SINGLE-SOURCE(G, s)

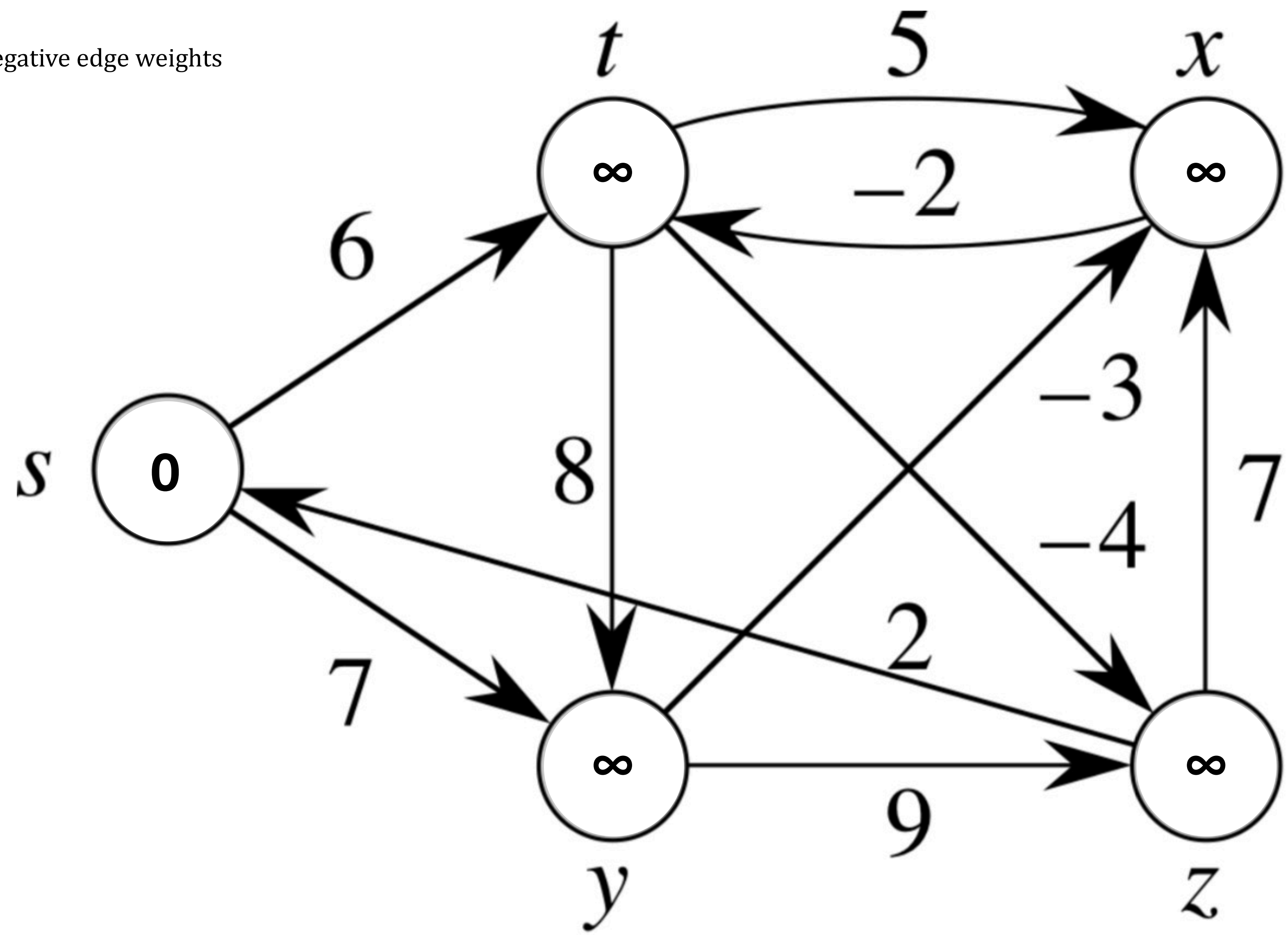
```
for each v ∈ G.V
    v.d = ∞      // estimate of shortest path distance
    v.π = NIL    // predecessor vertex
s.d = 0
```

The Bellman-Ford Single-Source Shortest Path Algorithm

Graph G

a weighted, directed graph with negative edge weights

G.V = s, t, x, y, z
G.E = (t, x : w = 5)
(t, y : w = 8)
(t, z : w = -4)
(x, t : w = -2)
(y, x : w = -3)
(y, z : w = 9)
(z, x : w = 7)
(z, s : w = 2)
(s, t : w = 6)
(s, y : w = 7)




```
BELLMAN-FORD(G, w, s)
  INIT-SINGLE-SOURCE(G, s)
  for i = 1 to |G.V| − 1
      for each edge (u, v) ∈ G.E
          RELAX(u, v, w)
  for each edge (u, v) ∈ G.E
      if v.d > u.d + w(u, v)
          return FALSE
  return TRUE
```

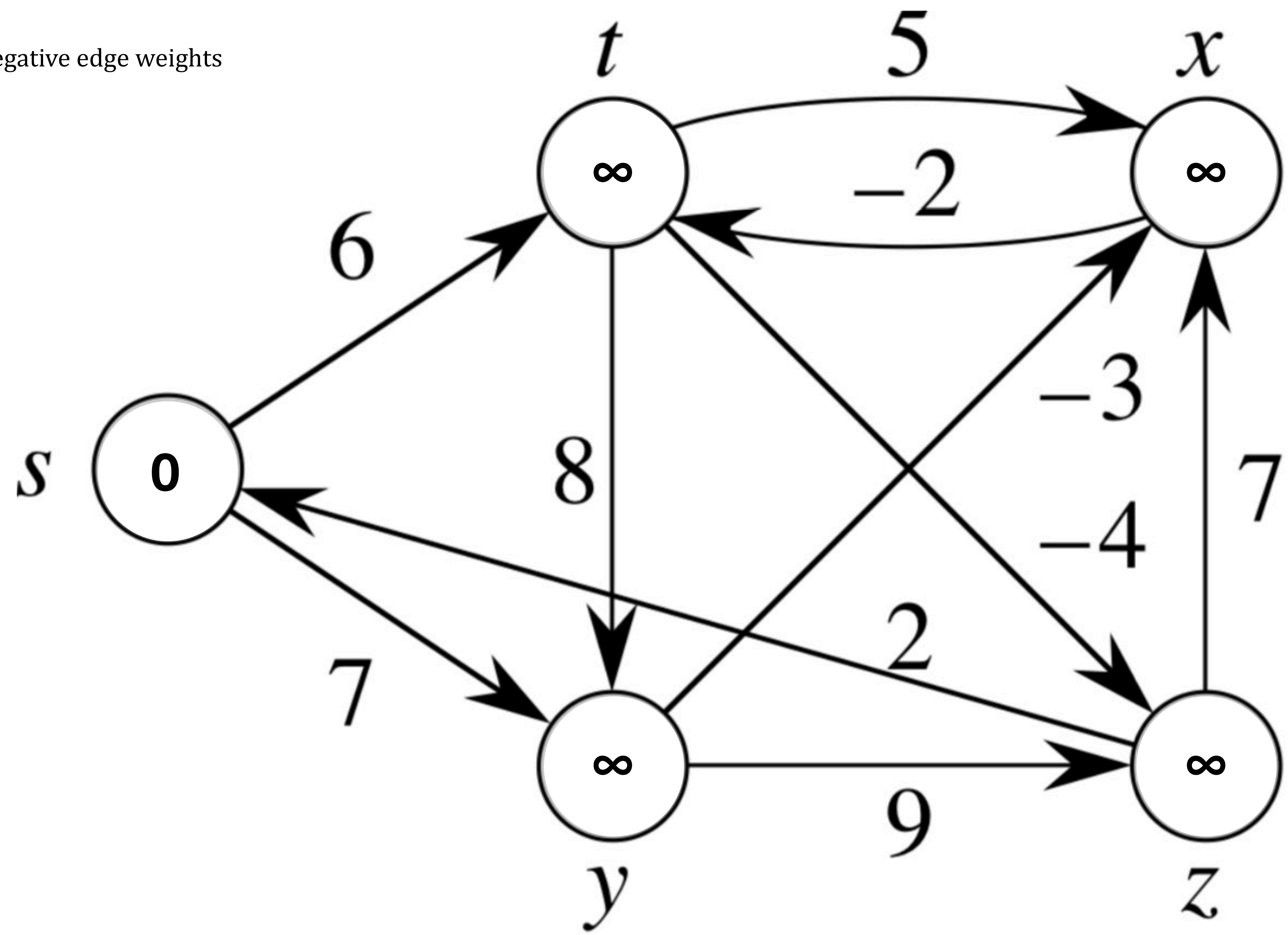
The Bellman-Ford Single-Source Shortest Path Algorithm

Graph G

a weighted, directed graph with negative edge weights

G.V = s, t, x, y, z

G.E = (t, x : w = 5)
(t, y : w = 8)
(t, z : w = -4)
(x, t : w = -2)
(y, x : w = -3)
(y, z : w = 9)
(z, x : w = 7)
(z, s : w = 2)
(s, t : w = 6)
(s, y : w = 7)




```
BELLMAN-FORD(G, w, s)
  INIT-SINGLE-SOURCE(G, s)
  for i = 1 to |G.V| − 1
      for each edge (u, v) ∈ G.E
          RELAX(u, v, w)
  for each edge (u, v) ∈ G.E
      if v.d > u.d + w(u, v)
          return FALSE
  return TRUE
```

```
RELAX(u, v, w)
  if v.d > u.d + w(u, v)
      v.d = u.d + w(u, v)
      v.π = u
```

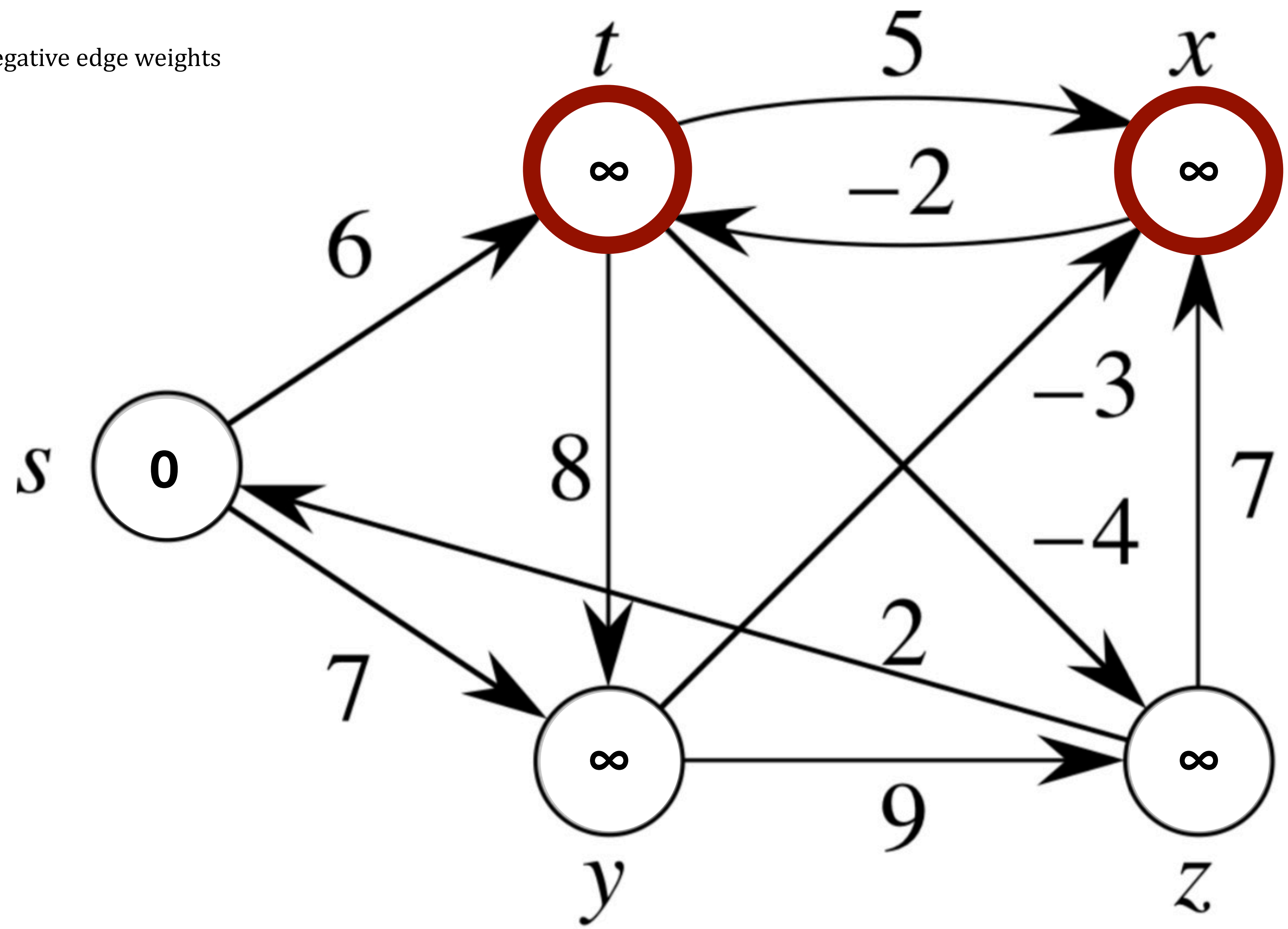
The Bellman-Ford Single-Source Shortest Path Algorithm

Graph G

a weighted, directed graph with negative edge weights

G.V = s, t, x, y, z

G.E = (t, x : w = 5)
(t, y : w = 8)
(t, z : w = -4)
(x, t : w = -2)
(y, x : w = -3)
(y, z : w = 9)
(z, x : w = 7)
(z, s : w = 2)
(s, t : w = 6)
(s, y : w = 7)



BELLMAN-FORD(G, w, s)
INIT-SINGLE-SOURCE(G, s)
for $i = 1$ to $|G.V| - 1$
  for each edge $(u, v) \in G.E$
    RELAX(u, v, w)
for each edge $(u, v) \in G.E$
  if $v.d > u.d + w(u, v)$
    return FALSE
return TRUE

RELAX(u, v, w)
if $v.d > u.d + w(u, v)$ // if **x**.d (∞) > **t**.d (∞) + w(t,x) (5) then
  $v.d = u.d + w(u, v)$ // set **x**.d = t.d + w(t,x)
  $v.\pi = u$ // set predecessor vertex to **t**

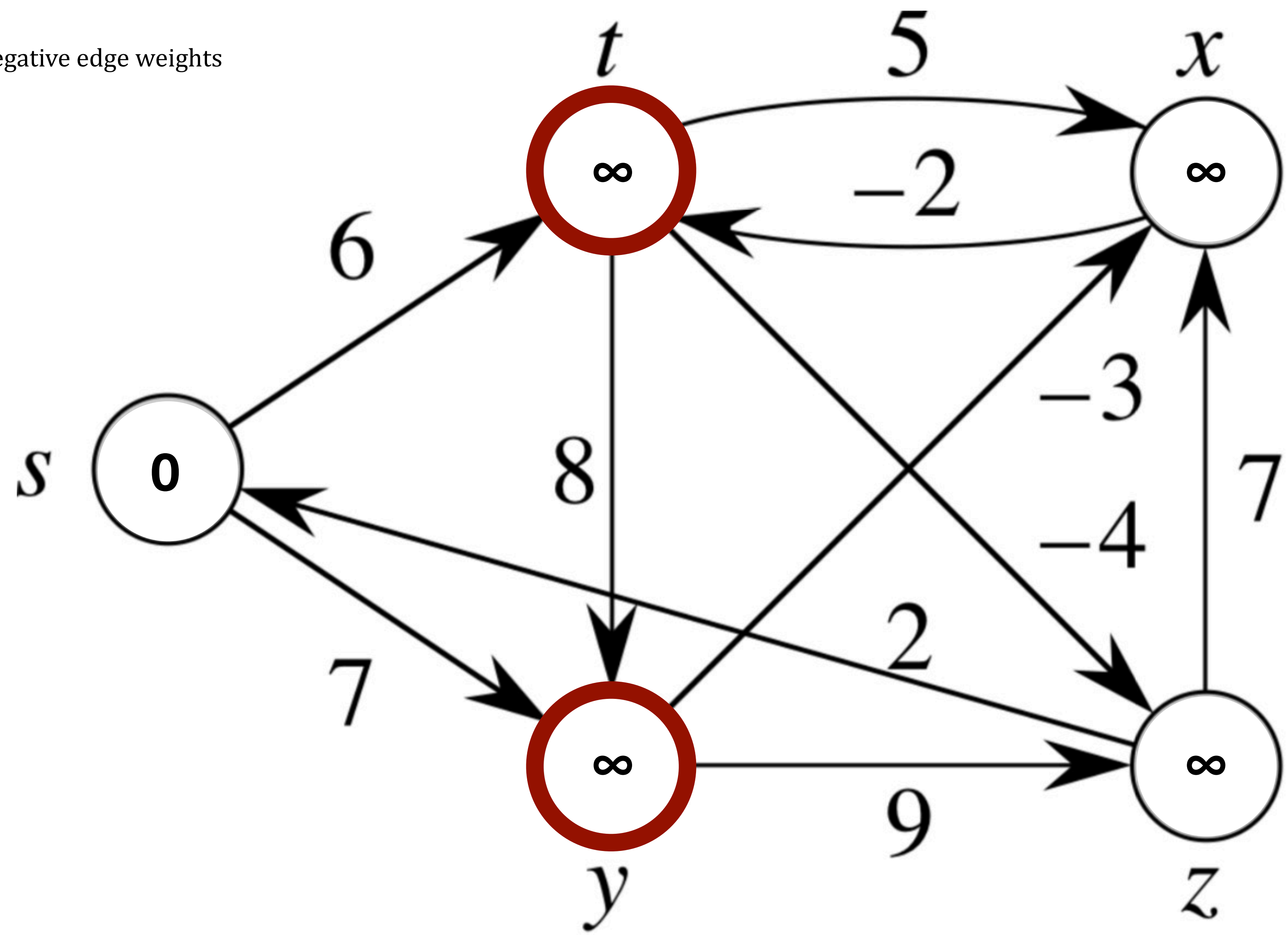
Assume for any real number a: a + ∞ = ∞ + a = ∞

The Bellman-Ford Single-Source Shortest Path Algorithm

Graph G

a weighted, directed graph with negative edge weights

G.V = s, t, x, y, z
G.E = (t, x : w = 5)
(t, y : w = 8)
(t, z : w = -4)
(x, t : w = -2)
(y, x : w = -3)
(y, z : w = 9)
(z, x : w = 7)
(z, s : w = 2)
(s, t : w = 6)
(s, y : w = 7)




```
BELLMAN-FORD(G, w, s)
  INIT-SINGLE-SOURCE(G, s)
  for i = 1 to |G.V| − 1
      for each edge (u, v) ∈ G.E
          RELAX(u, v, w)
  for each edge (u, v) ∈ G.E
      if v.d > u.d + w(u, v)
          return FALSE
  return TRUE
```

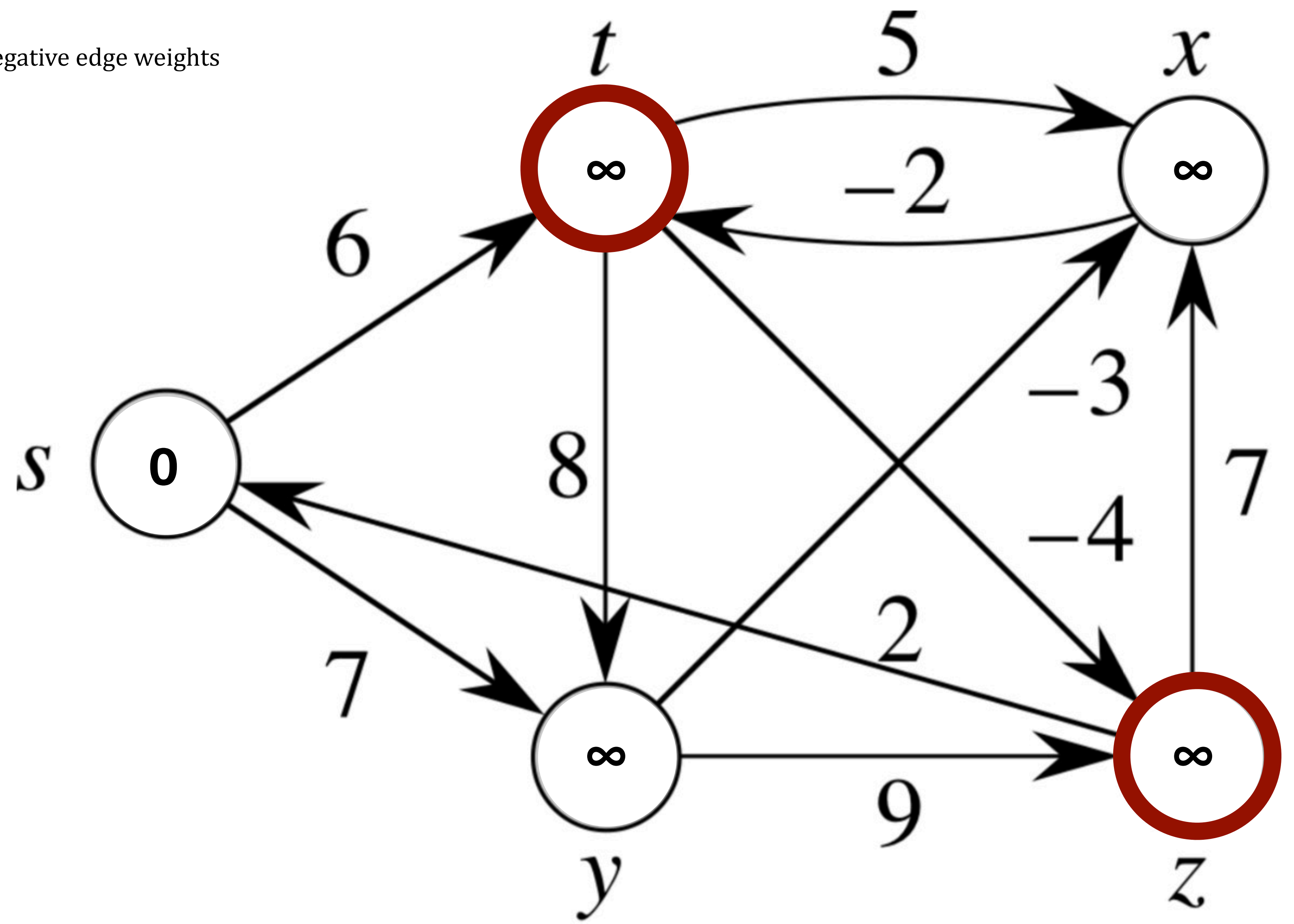
```
RELAX(u, v, w)
  if v.d > u.d + w(u, v)        // if y.d (∞) > t.d (∞) + w(t,y) (8) then
      v.d = u.d + w(u, v)       //   set y.d = t.d + w(t,y)
      v.π = u                   //   set predecessor vertex to t
```

The Bellman-Ford Single-Source Shortest Path Algorithm

Graph G

a weighted, directed graph with negative edge weights

G.V = s, t, x, y, z
G.E = (t, x : w = 5)
(t, y : w = 8)
(t, z : w = -4)
(x, t : w = -2)
(y, x : w = -3)
(y, z : w = 9)
(z, x : w = 7)
(z, s : w = 2)
(s, t : w = 6)
(s, y : w = 7)




```
BELLMAN-FORD(G, w, s)
  INIT-SINGLE-SOURCE(G, s)
  for i = 1 to |G.V| − 1
      for each edge (u, v) ∈ G.E
          RELAX(u, v, w)
  for each edge (u, v) ∈ G.E
      if v.d > u.d + w(u, v)
          return FALSE
  return TRUE
```

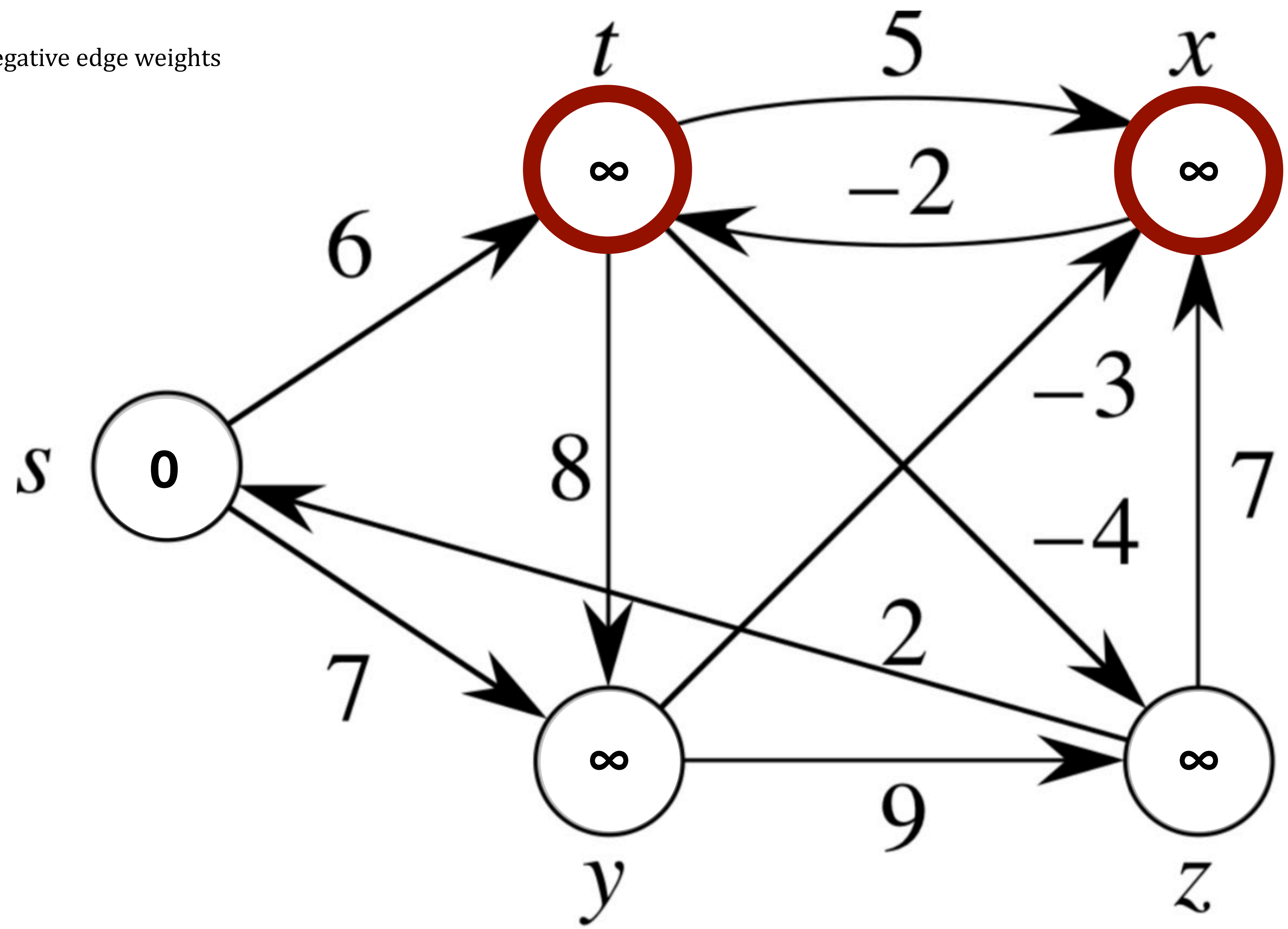
```
RELAX(u, v, w)
  if v.d > u.d + w(u, v)
      v.d = u.d + w(u, v)
      v.π = u
```

The Bellman-Ford Single-Source Shortest Path Algorithm

Graph G

a weighted, directed graph with negative edge weights

G.V = s, t, x, y, z
G.E = (t, x : w = 5)
(t, y : w = 8)
(t, z : w = -4)
(x, t : w = -2)
(y, x : w = -3)
(y, z : w = 9)
(z, x : w = 7)
(z, s : w = 2)
(s, t : w = 6)
(s, y : w = 7)



BELLMAN-FORD(G, w, s)
INIT-SINGLE-SOURCE(G, s)
for $i = 1$ to $|G.V| - 1$
for each edge $(u, v) \in G.E$
RELAX(u, v, w)
for each edge $(u, v) \in G.E$
if $v.d > u.d + w(u, v)$
return FALSE
return TRUE

RELAX(u, v, w)
if $v.d > u.d + w(u, v)$
$v.d = u.d + w(u, v)$
$v.\pi = u$

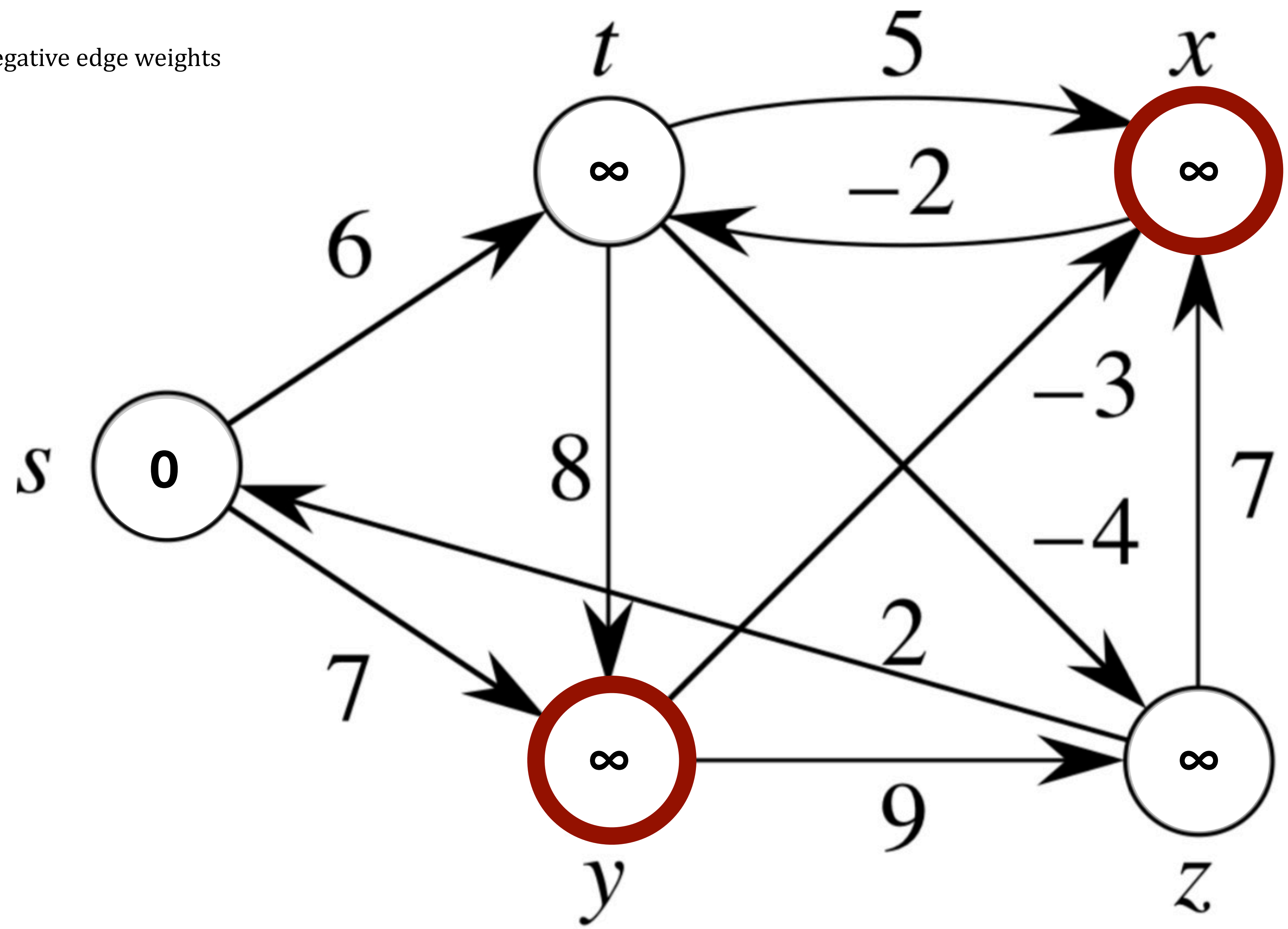
The Bellman-Ford Single-Source Shortest Path Algorithm

Graph G

a weighted, directed graph with negative edge weights

G.V = s, t, x, y, z

G.E = (t, x : w = 5)
(t, y : w = 8)
(t, z : w = -4)
(x, t : w = -2)
(y, x : w = -3)
(y, z : w = 9)
(z, x : w = 7)
(z, s : w = 2)
(s, t : w = 6)
(s, y : w = 7)




```
BELLMAN-FORD(G, w, s)
  INIT-SINGLE-SOURCE(G, s)
  for i = 1 to |G.V| − 1
      for each edge (u, v) ∈ G.E
          RELAX(u, v, w)
  for each edge (u, v) ∈ G.E
      if v.d > u.d + w(u, v)
          return FALSE
  return TRUE
```

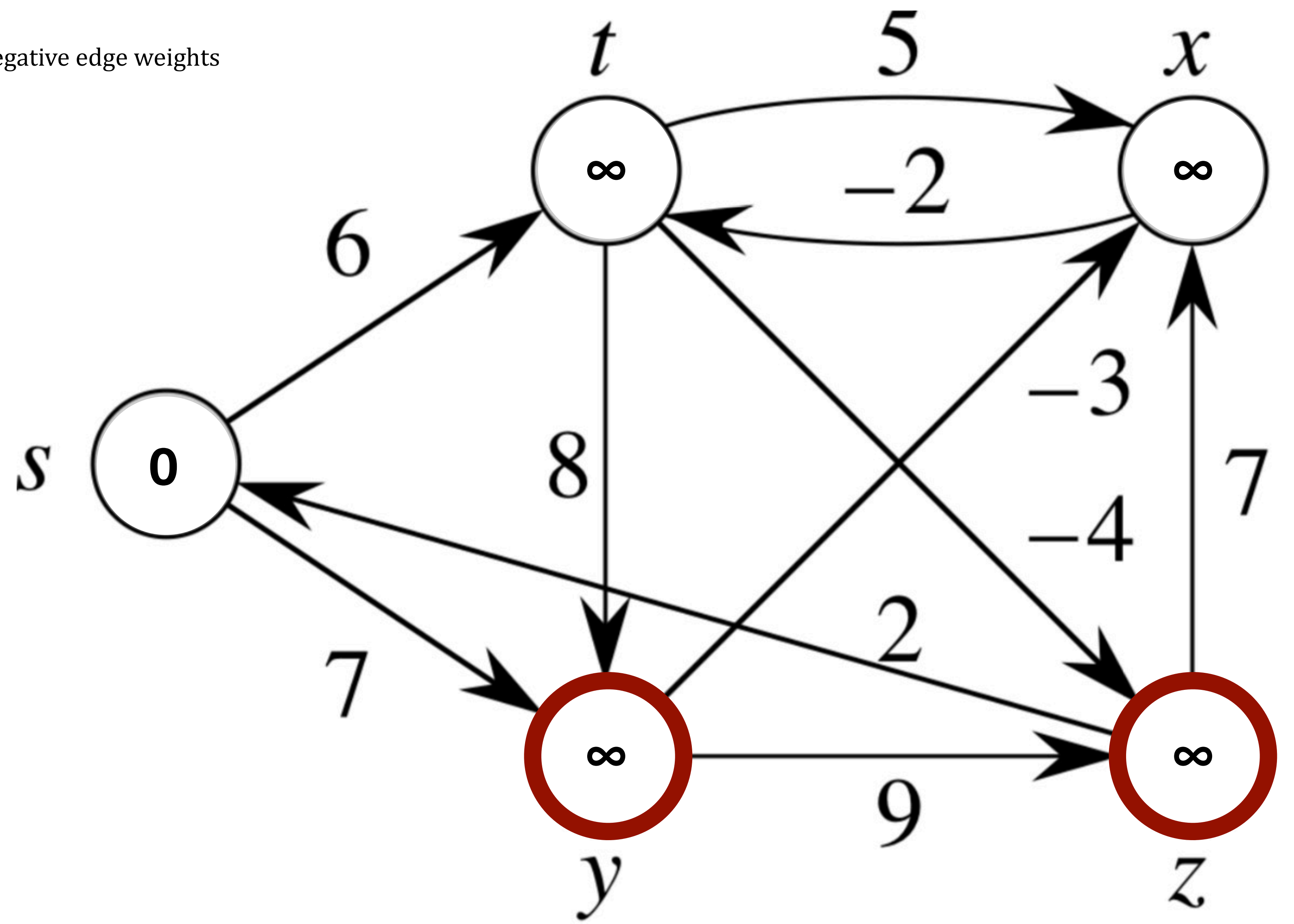
```
RELAX(u, v, w)
  if v.d > u.d + w(u, v)
      v.d = u.d + w(u, v)
      v.π = u
```

The Bellman-Ford Single-Source Shortest Path Algorithm

Graph G

a weighted, directed graph with negative edge weights

G.V = s, t, x, y, z
G.E = (t, x : w = 5)
(t, y : w = 8)
(t, z : w = -4)
(x, t : w = -2)
(y, x : w = -3)
(y, z : w = 9)
(z, x : w = 7)
(z, s : w = 2)
(s, t : w = 6)
(s, y : w = 7)




```
BELLMAN-FORD(G, w, s)
  INIT-SINGLE-SOURCE(G, s)
  for i = 1 to |G.V| − 1
      for each edge (u, v) ∈ G.E
          RELAX(u, v, w)
  for each edge (u, v) ∈ G.E
      if v.d > u.d + w(u, v)
          return FALSE
  return TRUE
```

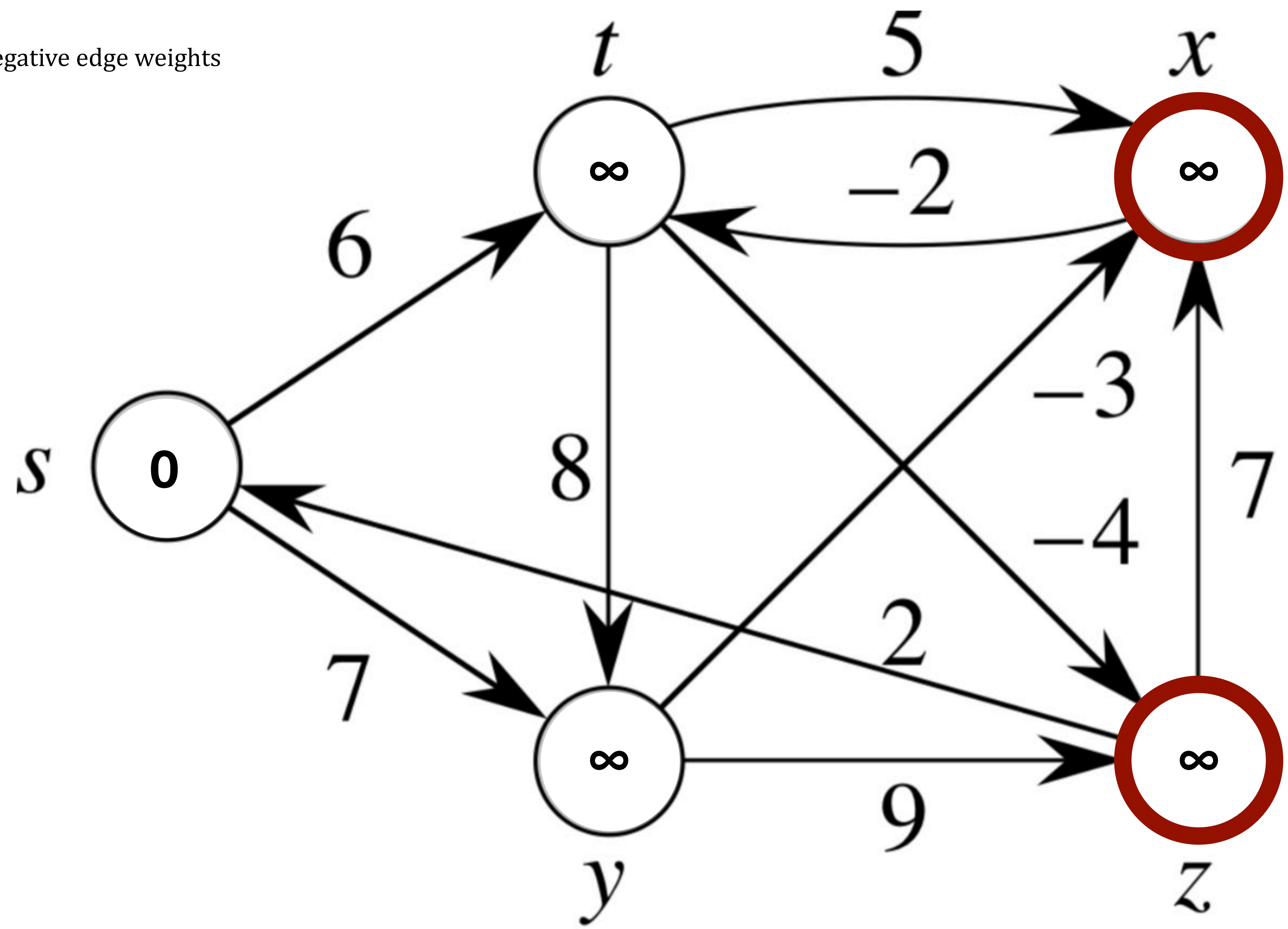
```
RELAX(u, v, w)
  if v.d > u.d + w(u, v)
      v.d = u.d + w(u, v)
      v.π = u
```

The Bellman-Ford Single-Source Shortest Path Algorithm

Graph G

a weighted, directed graph with negative edge weights

G.V = s, t, x, y, z
G.E = (t, x : w = 5)
(t, y : w = 8)
(t, z : w = -4)
(x, t : w = -2)
(y, x : w = -3)
(y, z : w = 9)
(z, x : w = 7)
(z, s : w = 2)
(s, t : w = 6)
(s, y : w = 7)



BELLMAN-FORD(G, w, s)
INIT-SINGLE-SOURCE(G, s)
for $i = 1$ to $|G.V| - 1$
for each edge $(u, v) \in G.E$
RELAX(u, v, w)
for each edge $(u, v) \in G.E$
if $v.d > u.d + w(u, v)$
return FALSE
return TRUE

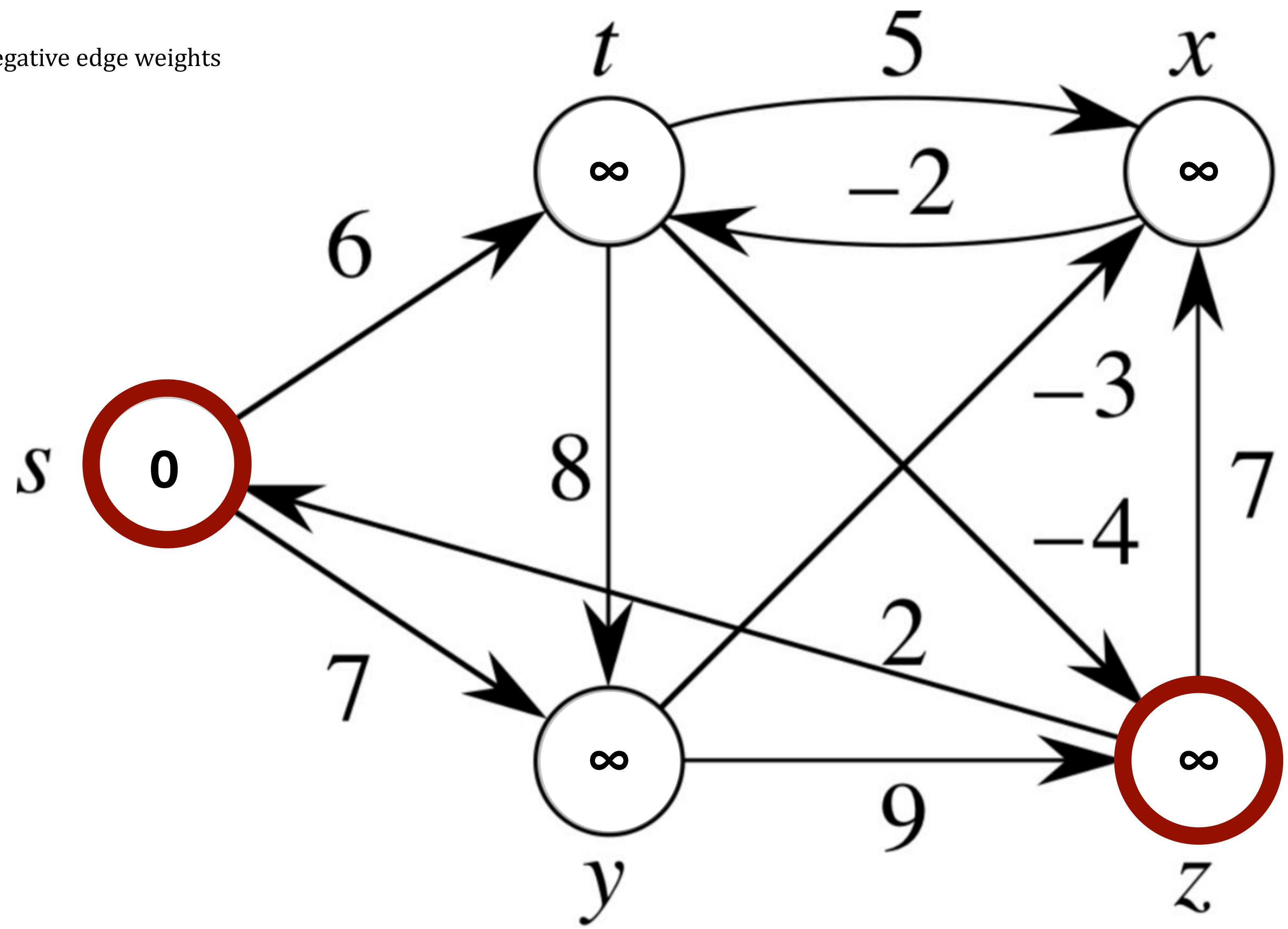
RELAX(u, v, w)
if $v.d > u.d + w(u, v)$
$v.d = u.d + w(u, v)$
$v.\pi = u$

The Bellman-Ford Single-Source Shortest Path Algorithm

Graph G

a weighted, directed graph with negative edge weights

G.V = s, t, x, y, z
G.E = (t, x : w = 5)
(t, y : w = 8)
(t, z : w = -4)
(x, t : w = -2)
(y, x : w = -3)
(y, z : w = 9)
(z, x : w = 7)
(z, s : w = 2)
(s, t : w = 6)
(s, y : w = 7)



BELLMAN-FORD(G, w, s)
INIT-SINGLE-SOURCE(G, s)
for $i = 1$ to $|G.V| - 1$
for each edge $(u, v) \in G.E$
RELAX(u, v, w)
for each edge $(u, v) \in G.E$
if $v.d > u.d + w(u, v)$
return FALSE
return TRUE

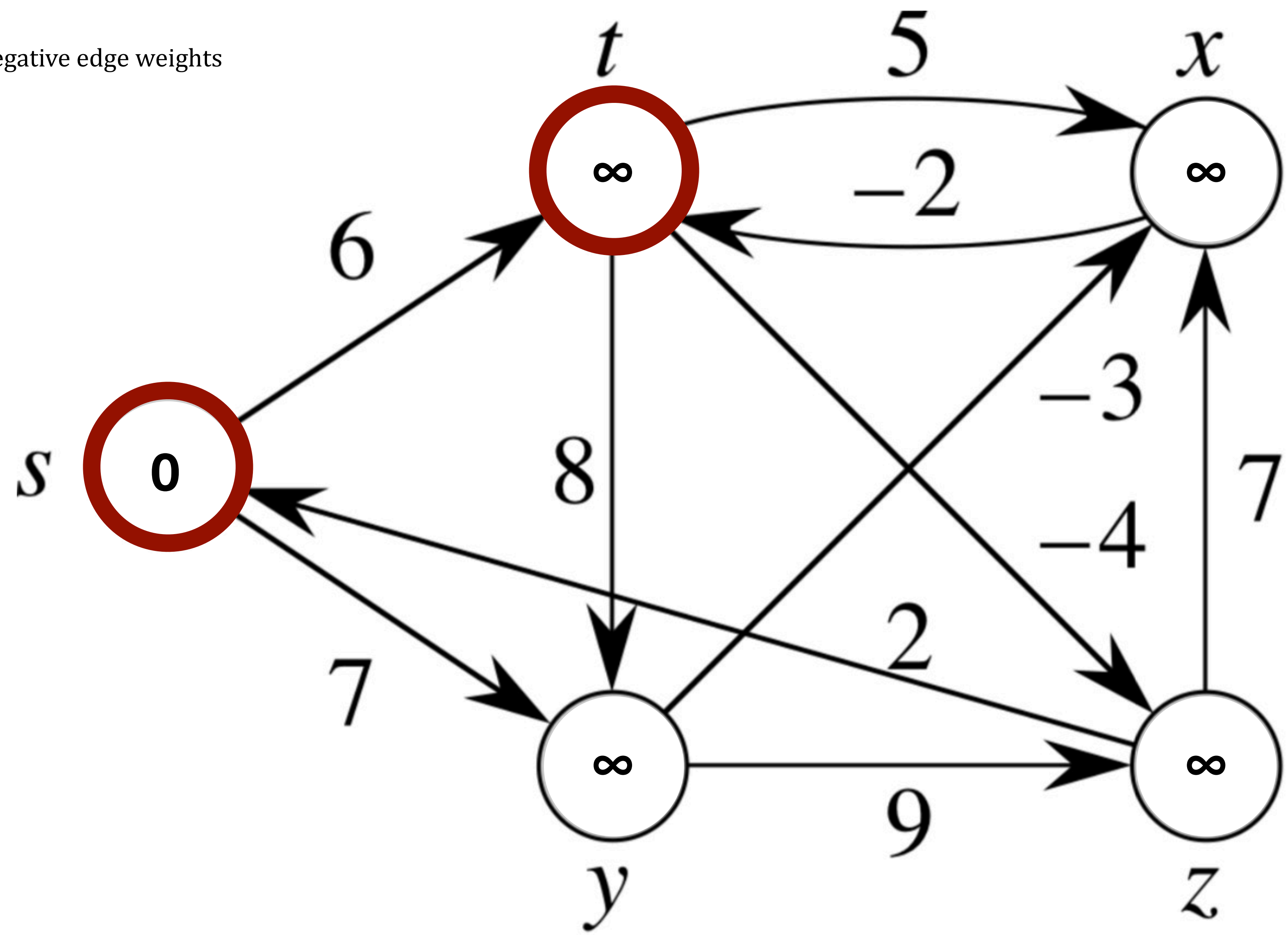
RELAX(u, v, w)
if $v.d > u.d + w(u, v)$
$v.d = u.d + w(u, v)$
$v.\pi = u$

The Bellman-Ford Single-Source Shortest Path Algorithm

Graph G

a weighted, directed graph with negative edge weights

G.V = s, t, x, y, z
G.E = (t, x : w = 5)
(t, y : w = 8)
(t, z : w = -4)
(x, t : w = -2)
(y, x : w = -3)
(y, z : w = 9)
(z, x : w = 7)
(z, s : w = 2)
(s, t : w = 6)
(s, y : w = 7)



BELLMAN-FORD(G, w, s)
INIT-SINGLE-SOURCE(G, s)
for $i = 1$ to $|G.V| - 1$
for each edge $(u, v) \in G.E$
RELAX(u, v, w)
for each edge $(u, v) \in G.E$
if $v.d > u.d + w(u, v)$
return FALSE
return TRUE

RELAX(u, v, w)
if $v.d > u.d + w(u, v)$ // if **t**.d (∞) > **s**.d (0) + w(s,t) (6) then
$v.d = u.d + w(u, v)$ // set **t**.d = s.d + w(s,t) = 6
$v.\pi = u$ // set predecessor vertex to **s**

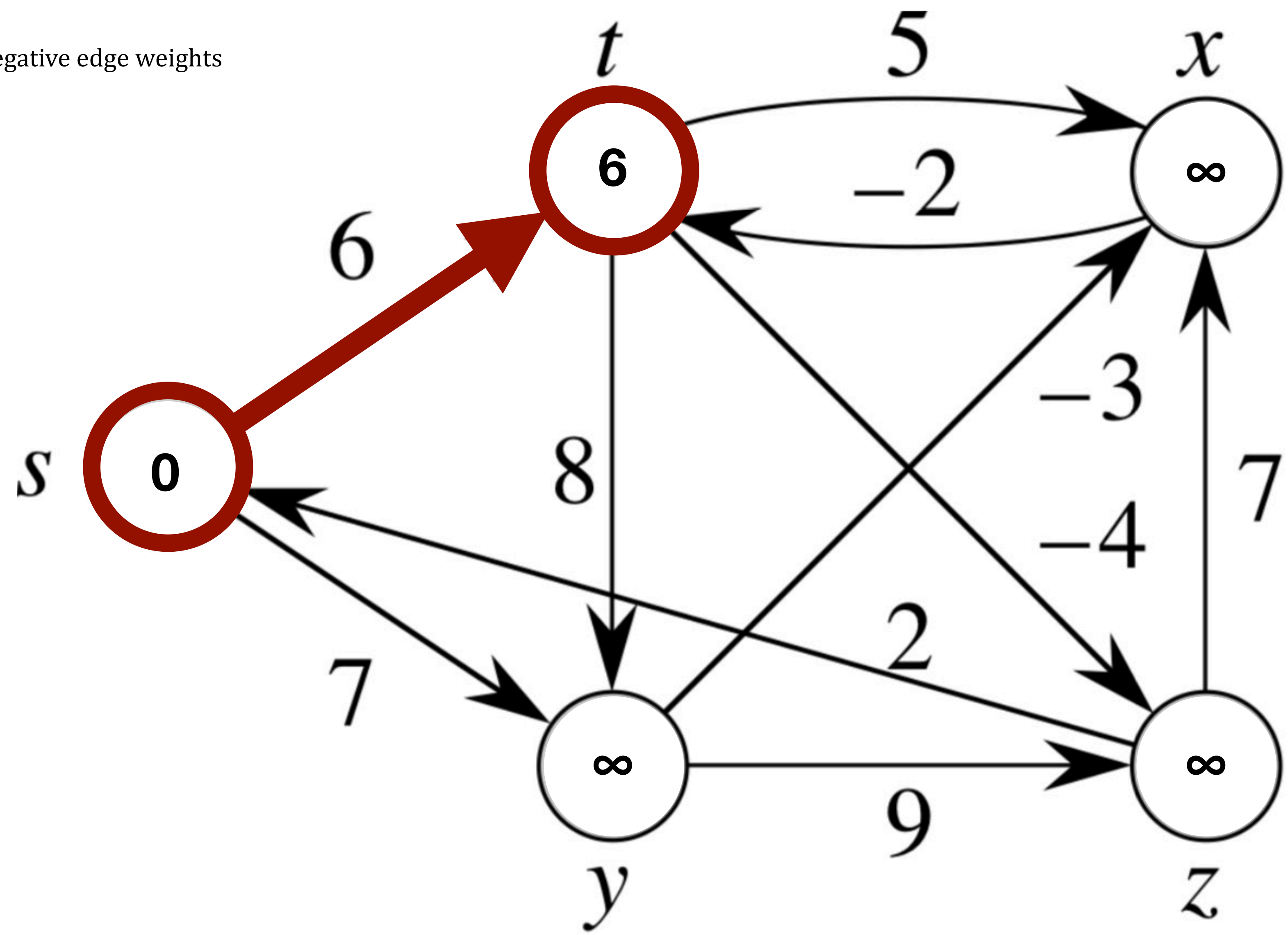
Finally!

The Bellman-Ford Single-Source Shortest Path Algorithm

Graph G

a weighted, directed graph with negative edge weights

G.V = s, t, x, y, z
G.E = (t, x : w = 5)
(t, y : w = 8)
(t, z : w = -4)
(x, t : w = -2)
(y, x : w = -3)
(y, z : w = 9)
(z, x : w = 7)
(z, s : w = 2)
(s, t : w = 6)
(s, y : w = 7)




```
BELLMAN-FORD(G, w, s)
  INIT-SINGLE-SOURCE(G, s)
  for i = 1 to |G.V| − 1
      for each edge (u, v) ∈ G.E
          RELAX(u, v, w)
  for each edge (u, v) ∈ G.E
      if v.d > u.d + w(u, v)
          return FALSE
  return TRUE
```

```
RELAX(u, v, w)
  if v.d > u.d + w(u, v)      // if t.d (∞) > s.d (0) + w(s,t) (6) then
      v.d = u.d + w(u, v)     //   set t.d = s.d + w(s,t) = 6
      v.π = u                 //   set predecessor vertex to s
```

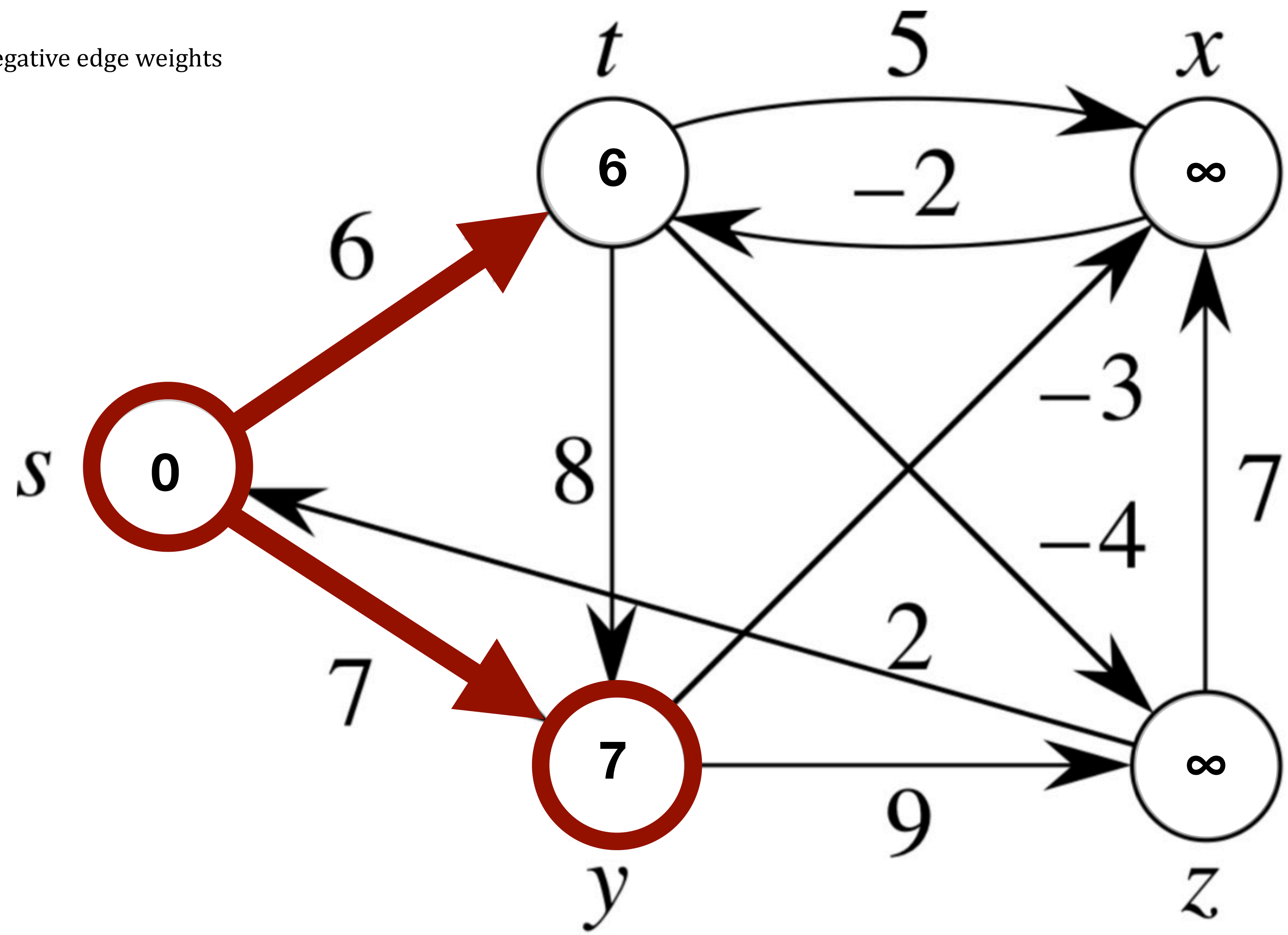
Finally!

The Bellman-Ford Single-Source Shortest Path Algorithm

Graph G

a weighted, directed graph with negative edge weights

G.V = s, t, x, y, z
G.E = (t, x : w = 5)
(t, y : w = 8)
(t, z : w = -4)
(x, t : w = -2)
(y, x : w = -3)
(y, z : w = 9)
(z, x : w = 7)
(z, s : w = 2)
(s, t : w = 6)
(s, y : w = 7)




```
BELLMAN-FORD(G, w, s)
  INIT-SINGLE-SOURCE(G, s)
  for i = 1 to |G.V| − 1
      for each edge (u, v) ∈ G.E
          RELAX(u, v, w)
  for each edge (u, v) ∈ G.E
      if v.d > u.d + w(u, v)
          return FALSE
  return TRUE
```

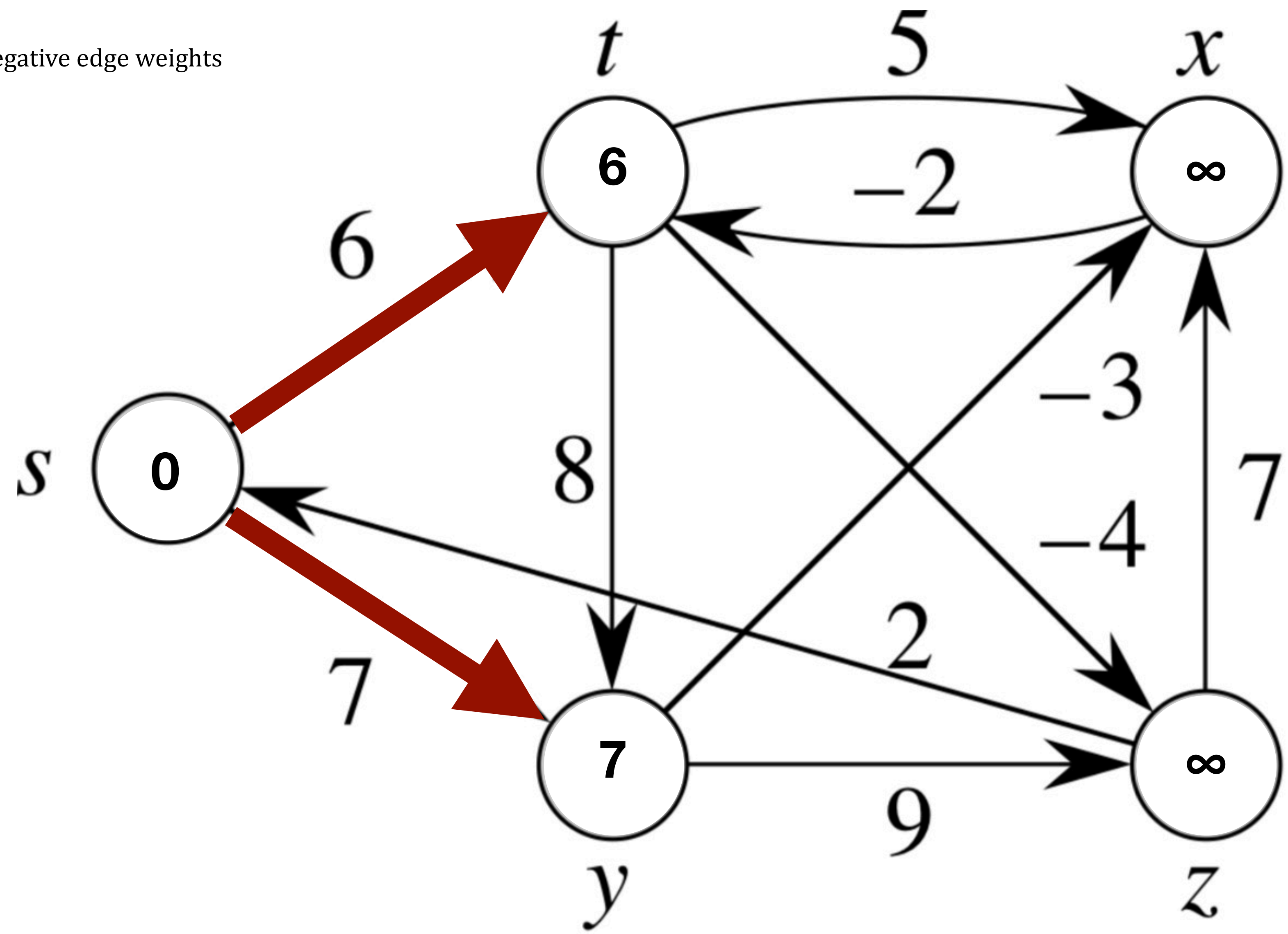
```
RELAX(u, v, w)
  if v.d > u.d + w(u, v)      // if y.d (∞) > s.d (0) + w(s,t) (7) then
      v.d = u.d + w(u, v)     //   set y.d = s.d + w(s,y) = 7
      v.π = u                 //   set predecessor vertex to s
```

The Bellman-Ford Single-Source Shortest Path Algorithm

Graph G

a weighted, directed graph with negative edge weights

G.V = s, t, x, y, z
G.E = (t, x : w = 5)
(t, y : w = 8)
(t, z : w = -4)
(x, t : w = -2)
(y, x : w = -3)
(y, z : w = 9)
(z, x : w = 7)
(z, s : w = 2)
(s, t : w = 6)
(s, y : w = 7)




```
BELLMAN-FORD(G, w, s)
  INIT-SINGLE-SOURCE(G, s)
  for i = 2 to |G.V| − 1
      for each edge (u, v) ∈ G.E
          RELAX(u, v, w)
  for each edge (u, v) ∈ G.E
      if v.d > u.d + w(u, v)
          return FALSE
  return TRUE
```

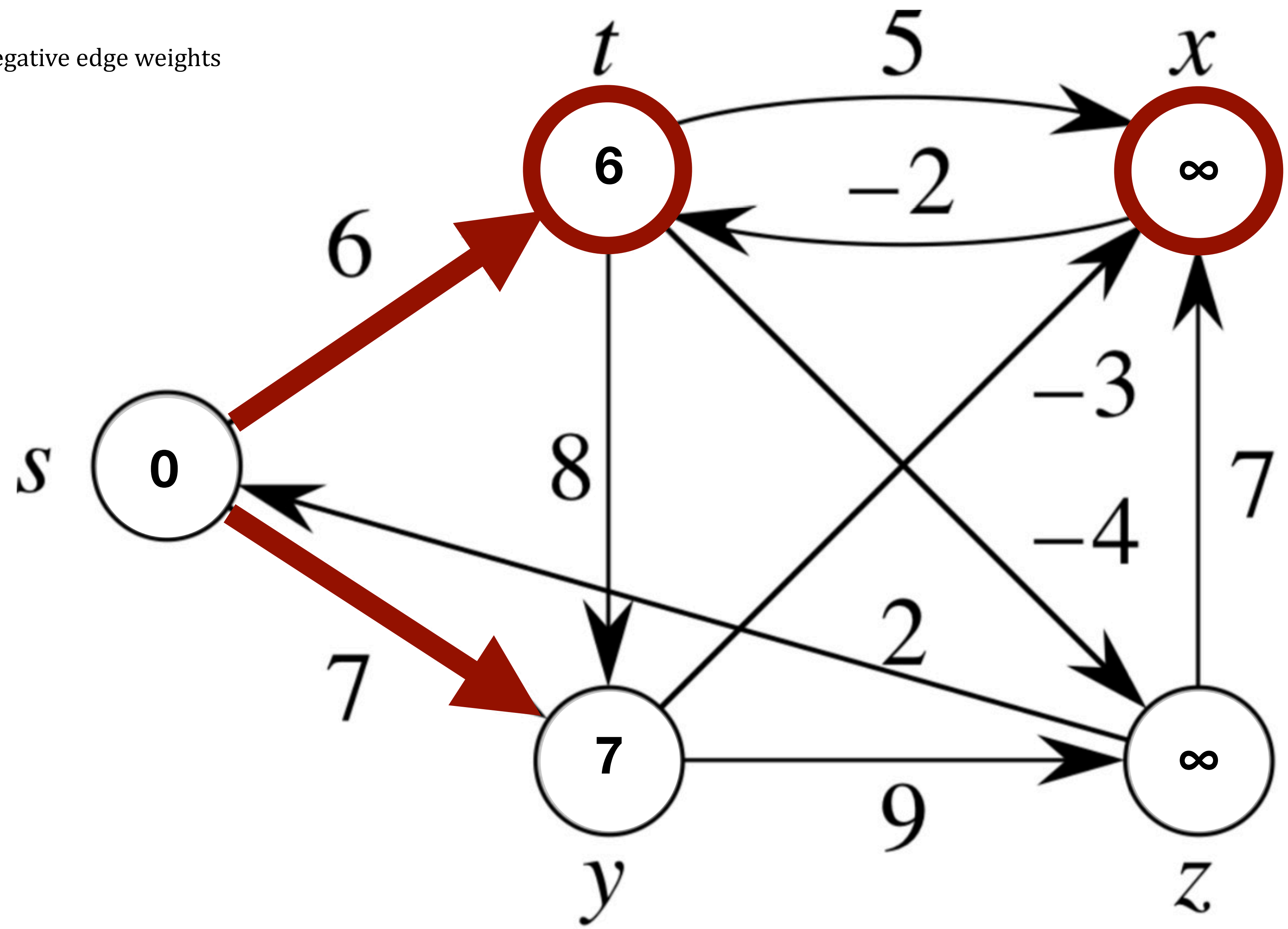
```
RELAX(u, v, w)
  if v.d > u.d + w(u, v)
      v.d = u.d + w(u, v)
      v.π = u
```

The Bellman-Ford Single-Source Shortest Path Algorithm

Graph G

a weighted, directed graph with negative edge weights

G.V = s, t, x, y, z
G.E = (t, x : w = 5)
(t, y : w = 8)
(t, z : w = -4)
(x, t : w = -2)
(y, x : w = -3)
(y, z : w = 9)
(z, x : w = 7)
(z, s : w = 2)
(s, t : w = 6)
(s, y : w = 7)



BELLMAN-FORD(G, w, s)
INIT-SINGLE-SOURCE(G, s)
for $i = \mathbf{2}$ to $|G.V| - 1$
for each edge $(u, v) \in G.E$
RELAX(u, v, w)
for each edge $(u, v) \in G.E$
if $v.d > u.d + w(u, v)$
return FALSE
return TRUE

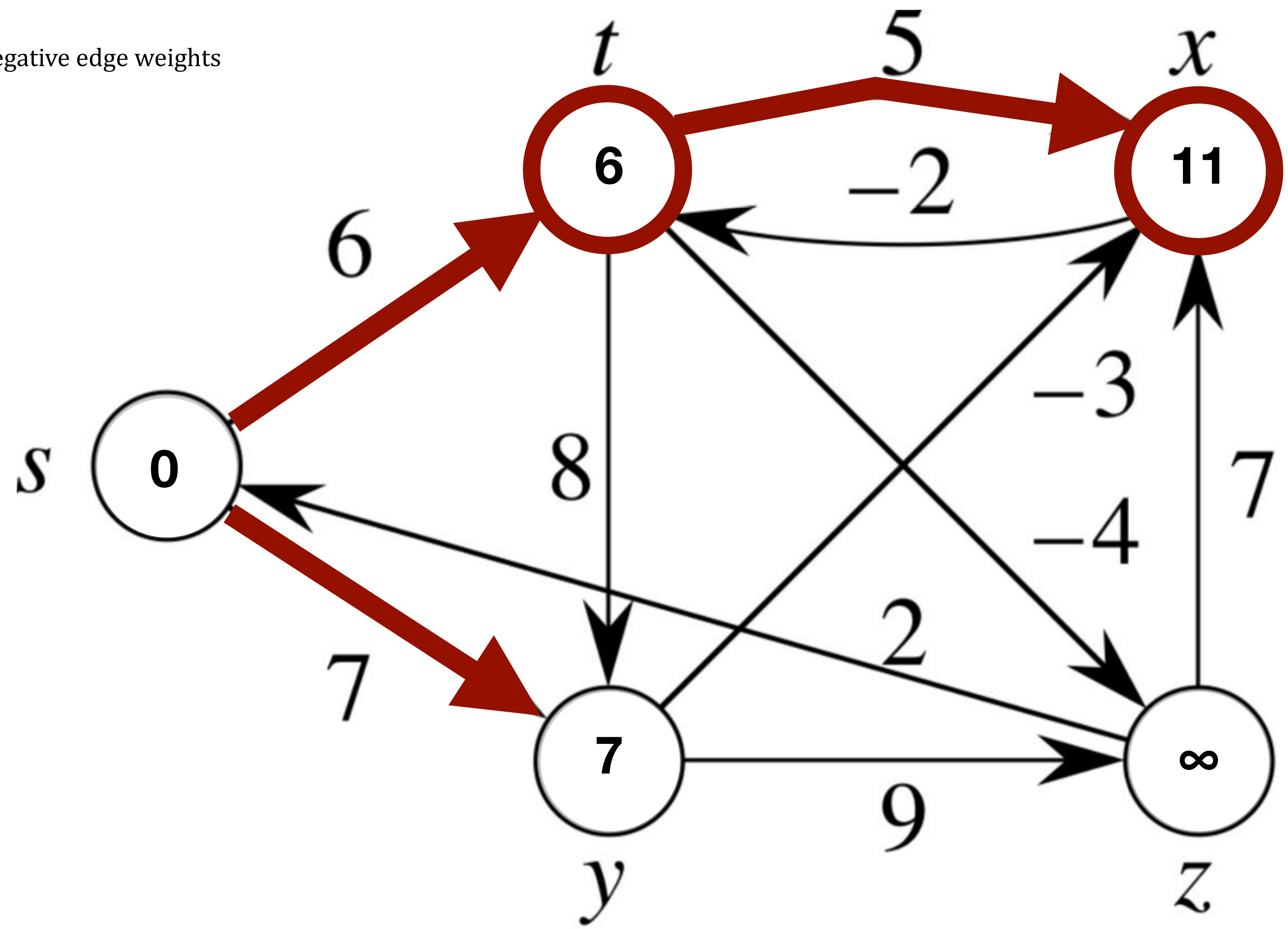
RELAX(u, v, w)
if $v.d > u.d + w(u, v)$
$v.d = u.d + w(u, v)$
$v.\pi = u$

The Bellman-Ford Single-Source Shortest Path Algorithm

Graph G

a weighted, directed graph with negative edge weights

G.V = s, t, x, y, z
G.E = (t, x : w = 5)
(t, y : w = 8)
(t, z : w = -4)
(x, t : w = -2)
(y, x : w = -3)
(y, z : w = 9)
(z, x : w = 7)
(z, s : w = 2)
(s, t : w = 6)
(s, y : w = 7)



Remember,
these are estimates
until we finish.
Can we do better?

BELLMAN-FORD(G, w, s)
  INIT-SINGLE-SOURCE(G, s)
  for $i = \mathbf{2}$ to $|G.V| - 1$
    for each edge $(u, v) \in G.E$
      RELAX(u, v, w)
  for each edge $(u, v) \in G.E$
    if $v.d > u.d + w(u, v)$
      return FALSE
  return TRUE

RELAX(u, v, w)
  if $v.d > u.d + w(u, v)$
    $v.d = u.d + w(u, v)$
    $v.\pi = u$

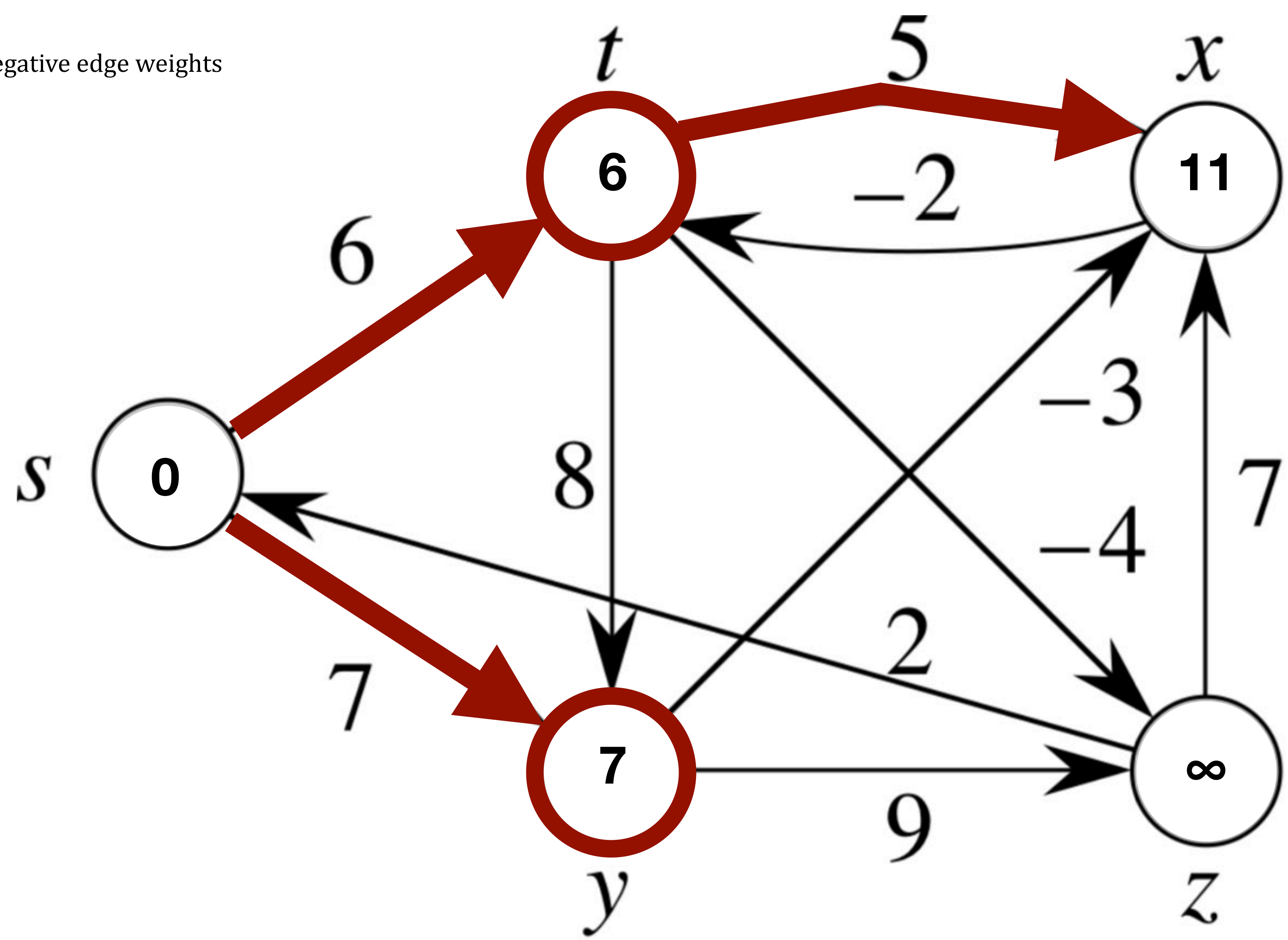
The Bellman-Ford Single-Source Shortest Path Algorithm

Graph G

a weighted, directed graph with negative edge weights

G.V = s, t, x, y, z

G.E = (t, x : w = 5)
(t, y : w = 8)
(t, z : w = -4)
(x, t : w = -2)
(y, x : w = -3)
(y, z : w = 9)
(z, x : w = 7)
(z, s : w = 2)
(s, t : w = 6)
(s, y : w = 7)




```
BELLMAN-FORD(G, w, s)
  INIT-SINGLE-SOURCE(G, s)
  for i = 2 to |G.V| − 1
        for each edge (u, v) ∈ G.E
              RELAX(u, v, w)
  for each edge (u, v) ∈ G.E
        if v.d > u.d + w(u, v)
              return FALSE
  return TRUE
```

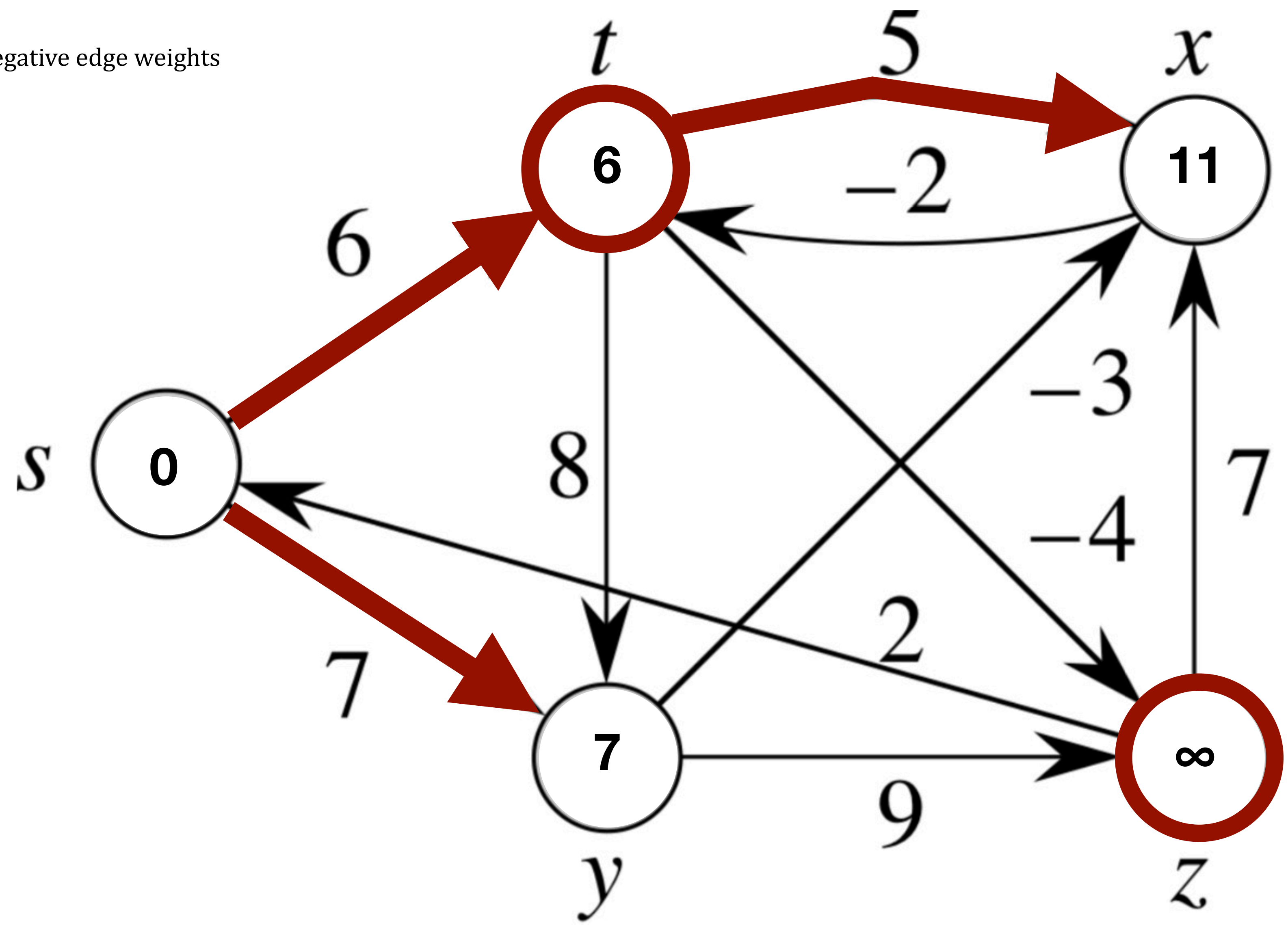
```
RELAX(u, v, w)
  if v.d > u.d + w(u, v)
        v.d = u.d + w(u, v)
        v.π = u
```

The Bellman-Ford Single-Source Shortest Path Algorithm

Graph G

a weighted, directed graph with negative edge weights

G.V = s, t, x, y, z
G.E = (t, x : w = 5)
(t, y : w = 8)
(t, z : w = -4)
(x, t : w = -2)
(y, x : w = -3)
(y, z : w = 9)
(z, x : w = 7)
(z, s : w = 2)
(s, t : w = 6)
(s, y : w = 7)




```
BELLMAN-FORD(G, w, s)
  INIT-SINGLE-SOURCE(G, s)
  for i = 2 to |G.V| − 1
      for each edge (u, v) ∈ G.E
          RELAX(u, v, w)
  for each edge (u, v) ∈ G.E
      if v.d > u.d + w(u, v)
          return FALSE
  return TRUE
```

```
RELAX(u, v, w)
  if v.d > u.d + w(u, v)
      v.d = u.d + w(u, v)
      v.π = u
```

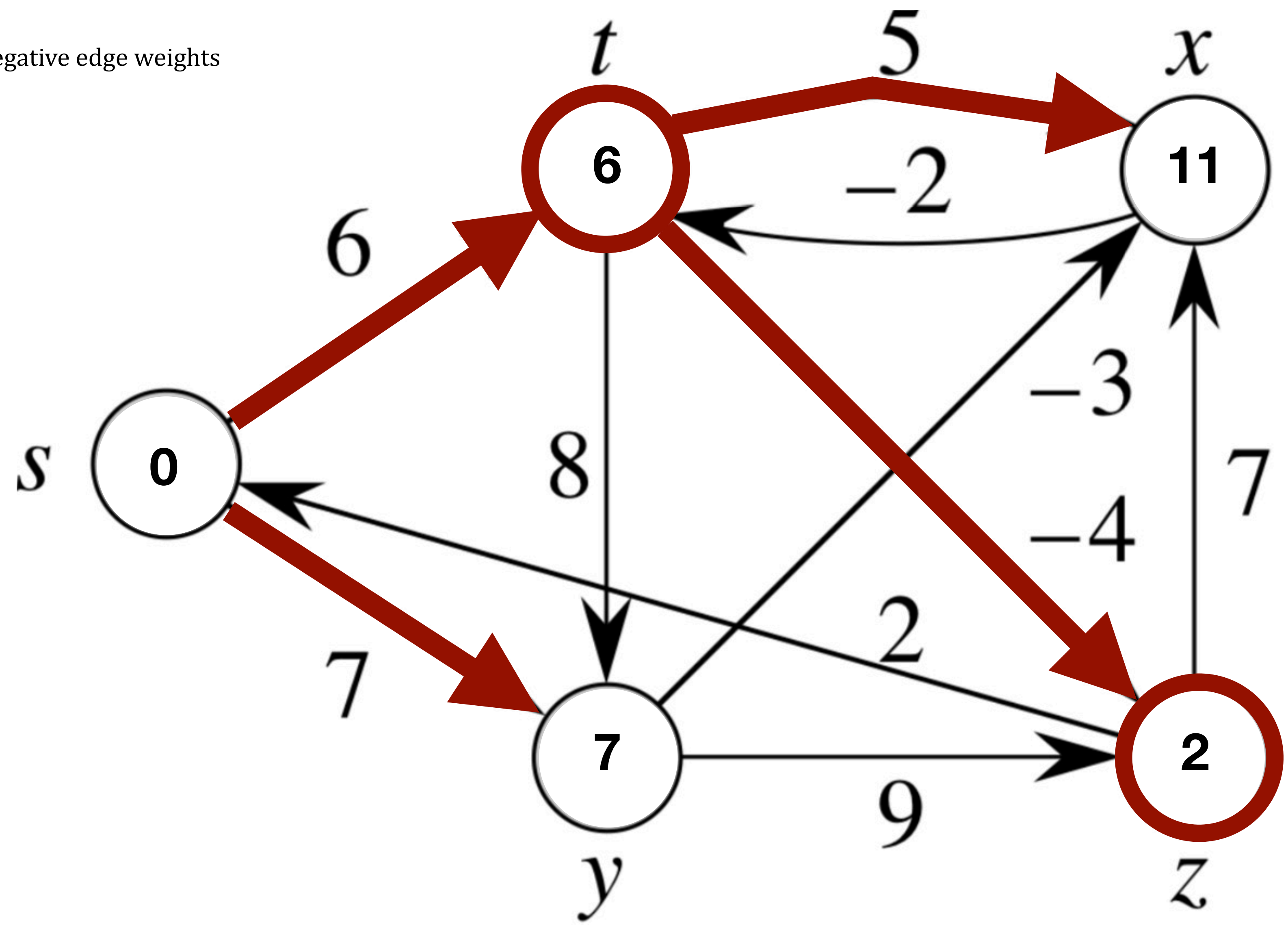
The Bellman-Ford Single-Source Shortest Path Algorithm

Graph G

a weighted, directed graph with negative edge weights

G.V = s, t, x, y, z

G.E = (t, x : w = 5)
(t, y : w = 8)
(t, z : w = -4)
(x, t : w = -2)
(y, x : w = -3)
(y, z : w = 9)
(z, x : w = 7)
(z, s : w = 2)
(s, t : w = 6)
(s, y : w = 7)




```
BELLMAN-FORD(G, w, s)
  INIT-SINGLE-SOURCE(G, s)
  for i = 2 to |G.V| - 1
      for each edge (u, v) ∈ G.E
          RELAX(u, v, w)
  for each edge (u, v) ∈ G.E
      if v.d > u.d + w(u, v)
          return FALSE
  return TRUE
```

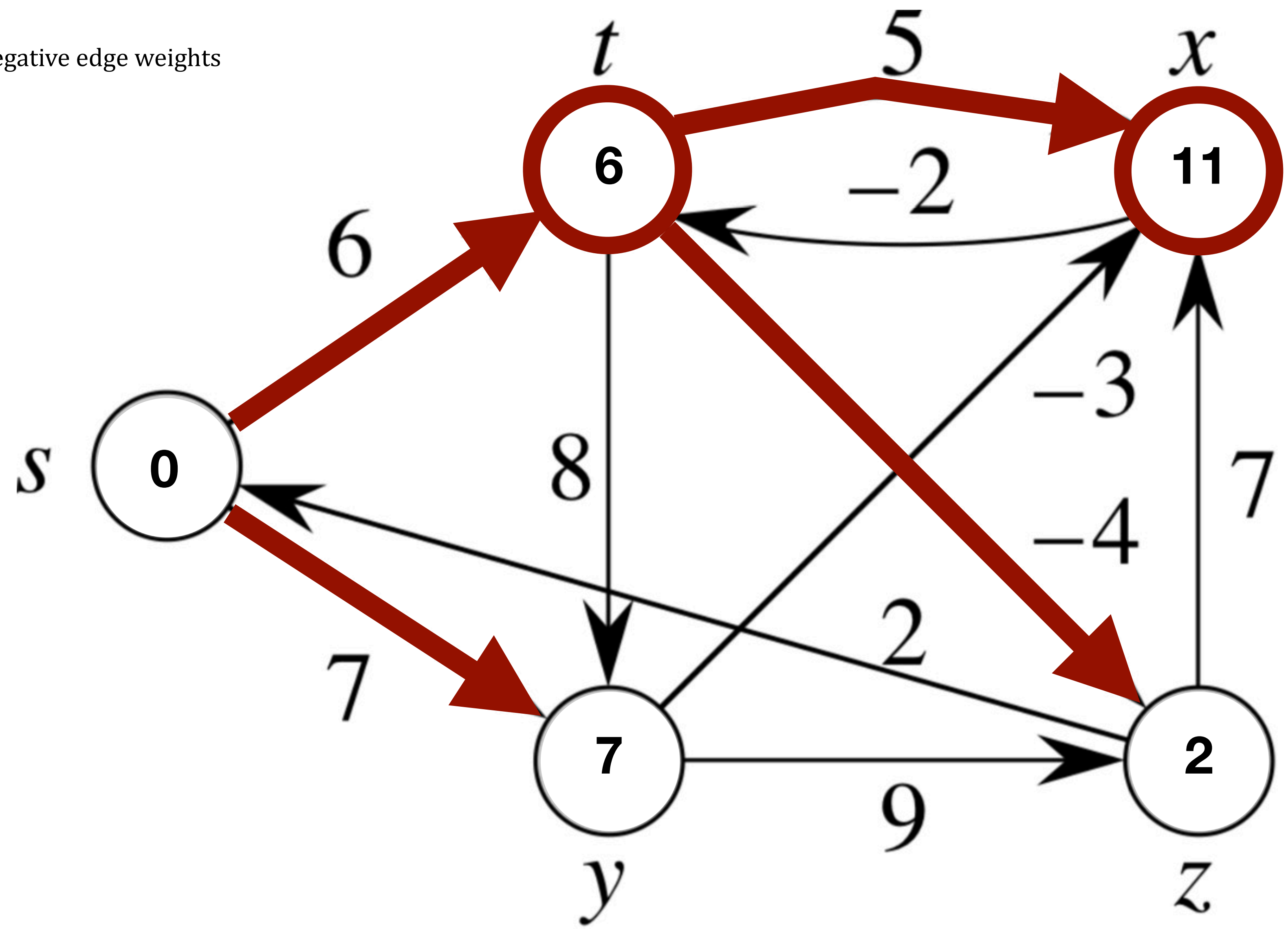
```
RELAX(u, v, w)
  if v.d > u.d + w(u, v)
      v.d = u.d + w(u, v)
      v.π = u
```

The Bellman-Ford Single-Source Shortest Path Algorithm

Graph G

a weighted, directed graph with negative edge weights

G.V = s, t, x, y, z
G.E = (t, x : w = 5)
(t, y : w = 8)
(t, z : w = -4)
(x, t : w = -2)
(y, x : w = -3)
(y, z : w = 9)
(z, x : w = 7)
(z, s : w = 2)
(s, t : w = 6)
(s, y : w = 7)



BELLMAN-FORD(G, w, s)
INIT-SINGLE-SOURCE(G, s)
for i = **2** to $|G.V| - 1$
for each edge $(u, v) \in G.E$
RELAX(u, v, w)
for each edge $(u, v) \in G.E$
if $v.d > u.d + w(u, v)$
return FALSE
return TRUE

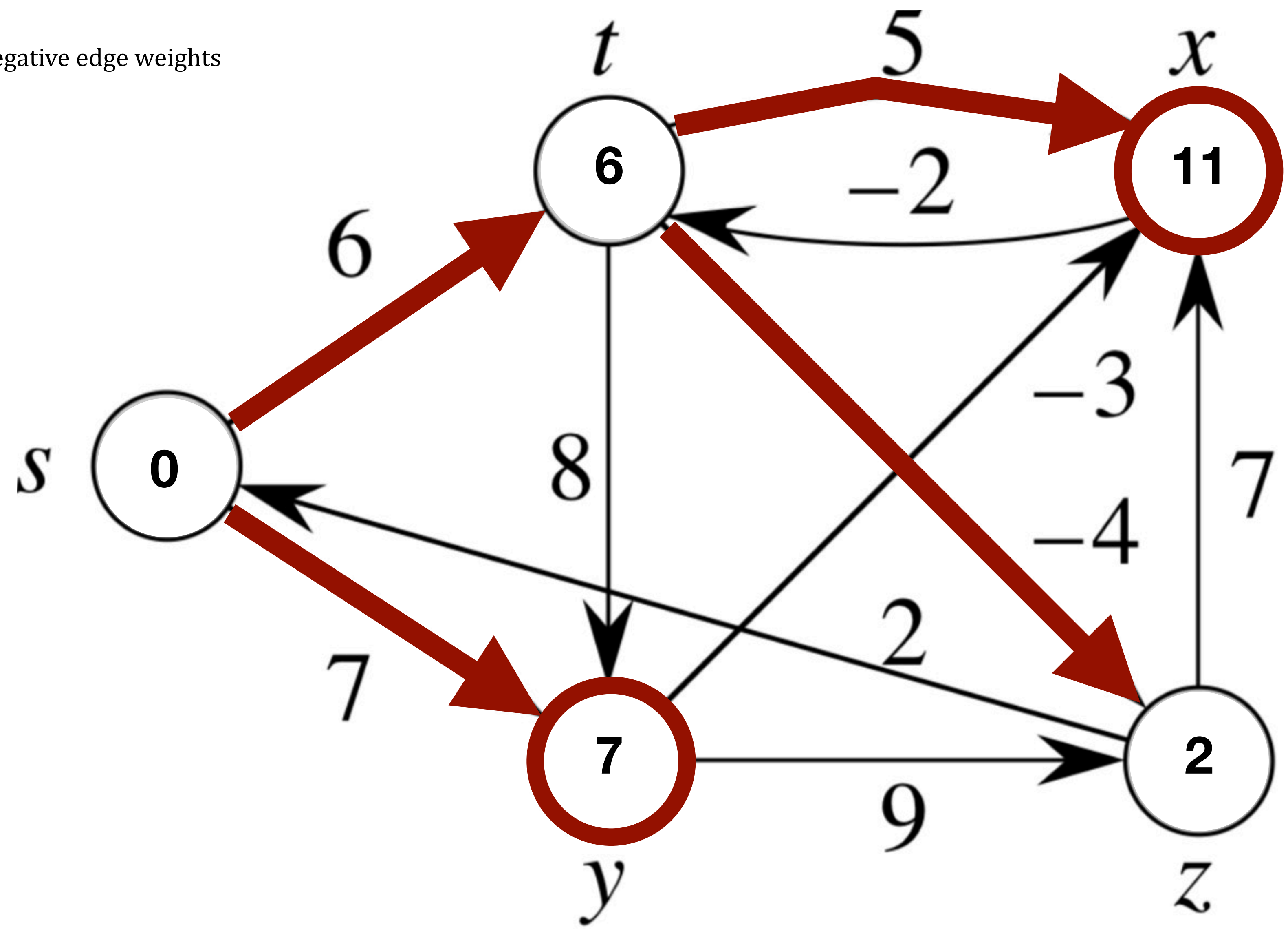
RELAX(u, v, w)
if $v.d > u.d + w(u, v)$
$v.d = u.d + w(u, v)$
$v.\pi = u$

The Bellman-Ford Single-Source Shortest Path Algorithm

Graph G

a weighted, directed graph with negative edge weights

G.V = s, t, x, y, z
G.E = (t, x : w = 5)
(t, y : w = 8)
(t, z : w = -4)
(x, t : w = -2)
(y, x : w = -3)
(y, z : w = 9)
(z, x : w = 7)
(z, s : w = 2)
(s, t : w = 6)
(s, y : w = 7)



Remember,
these are estimates
until we finish.
We CAN do better!

```
BELLMAN-FORD(G, w, s)
  INIT-SINGLE-SOURCE(G, s)
  for i = 2 to |G.V| − 1
      for each edge (u, v) ∈ G.E
          RELAX(u, v, w)
  for each edge (u, v) ∈ G.E
      if v.d > u.d + w(u, v)
          return FALSE
  return TRUE
```

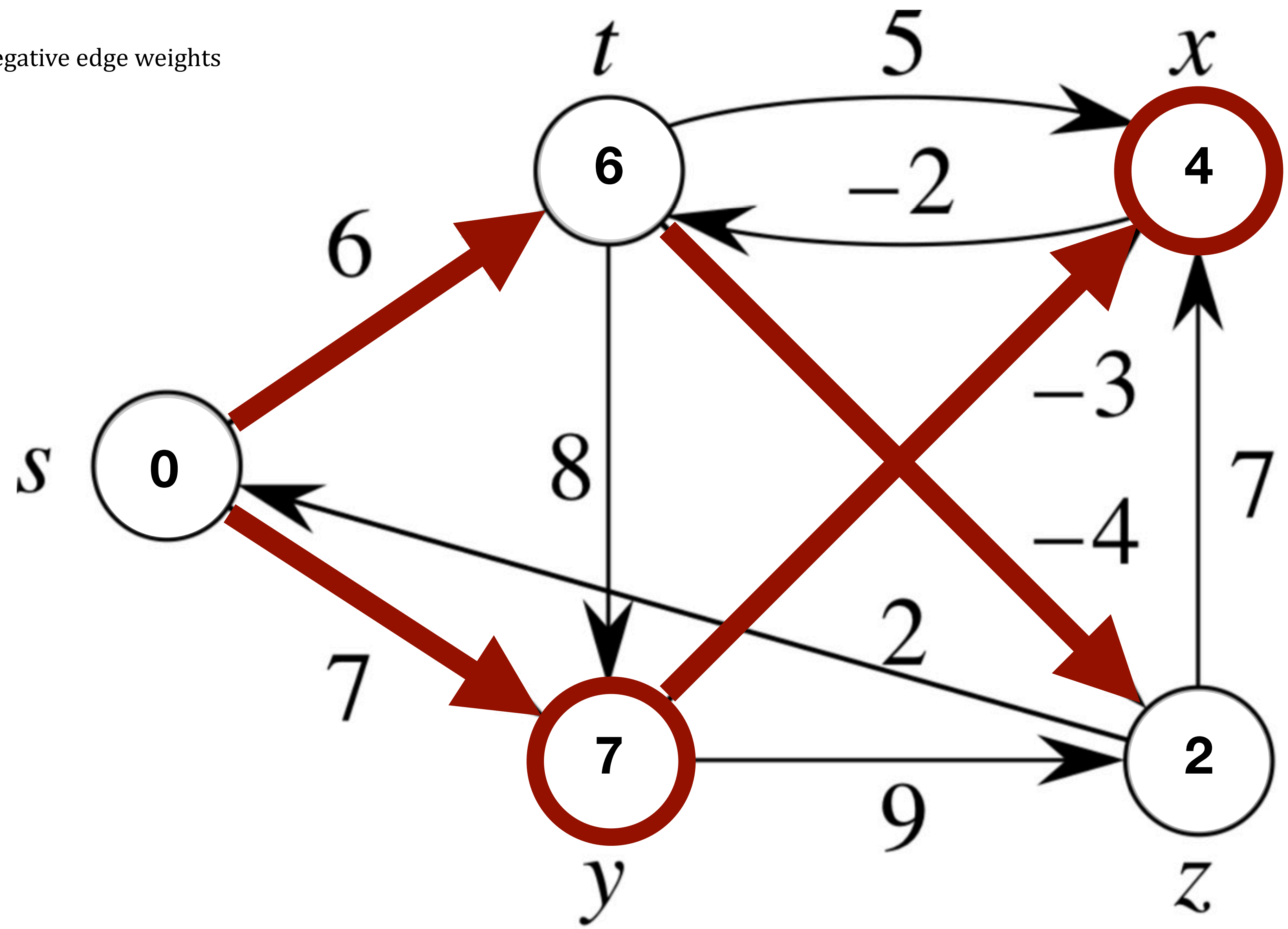
```
RELAX(u, v, w)
  if v.d > u.d + w(u, v)
      v.d = u.d + w(u, v)
      v.π = u
```

The Bellman-Ford Single-Source Shortest Path Algorithm

Graph G

a weighted, directed graph with negative edge weights

G.V = s, t, x, y, z
G.E = (t, x : w = 5)
(t, y : w = 8)
(t, z : w = -4)
(x, t : w = -2)
(y, x : w = -3)
(y, z : w = 9)
(z, x : w = 7)
(z, s : w = 2)
(s, t : w = 6)
(s, y : w = 7)



Remember,
these are estimates
until we finish.
We CAN do better!

```
BELLMAN-FORD(G, w, s)
  INIT-SINGLE-SOURCE(G, s)
  for i = 2 to |G.V| − 1
      for each edge (u, v) ∈ G.E
          RELAX(u, v, w)
  for each edge (u, v) ∈ G.E
      if v.d > u.d + w(u, v)
          return FALSE
  return TRUE
```

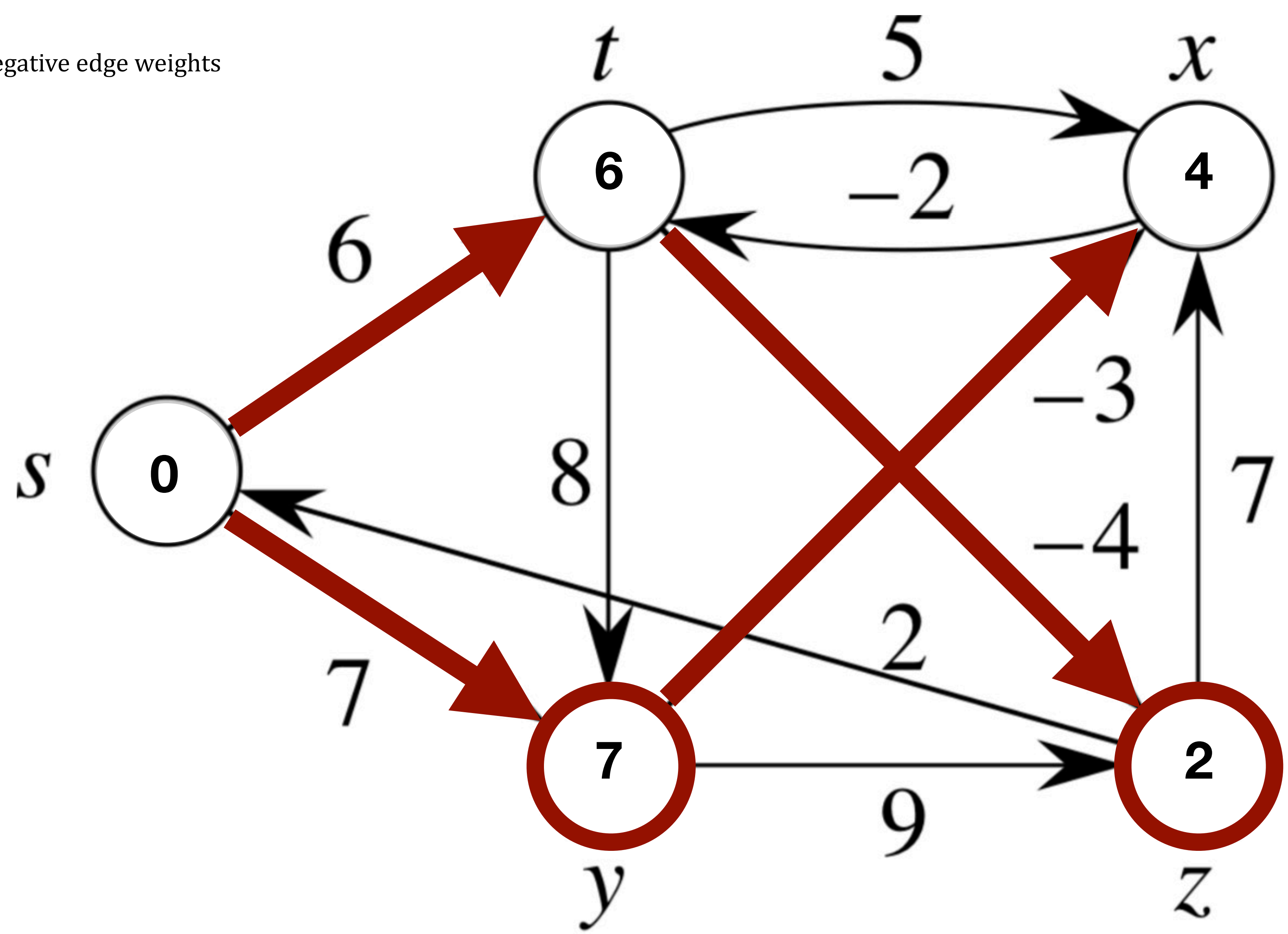
```
RELAX(u, v, w)
  if v.d > u.d + w(u, v)
      v.d = u.d + w(u, v)
      v.π = u
```

The Bellman-Ford Single-Source Shortest Path Algorithm

Graph G

a weighted, directed graph with negative edge weights

G.V = s, t, x, y, z
G.E = (t, x : w = 5)
(t, y : w = 8)
(t, z : w = -4)
(x, t : w = -2)
(y, x : w = -3)
(y, z : w = 9)
(z, x : w = 7)
(z, s : w = 2)
(s, t : w = 6)
(s, y : w = 7)




```
BELLMAN-FORD(G, w, s)
  INIT-SINGLE-SOURCE(G, s)
  for i = 2 to |G.V| − 1
      for each edge (u, v) ∈ G.E
          RELAX(u, v, w)
  for each edge (u, v) ∈ G.E
      if v.d > u.d + w(u, v)
          return FALSE
  return TRUE
```

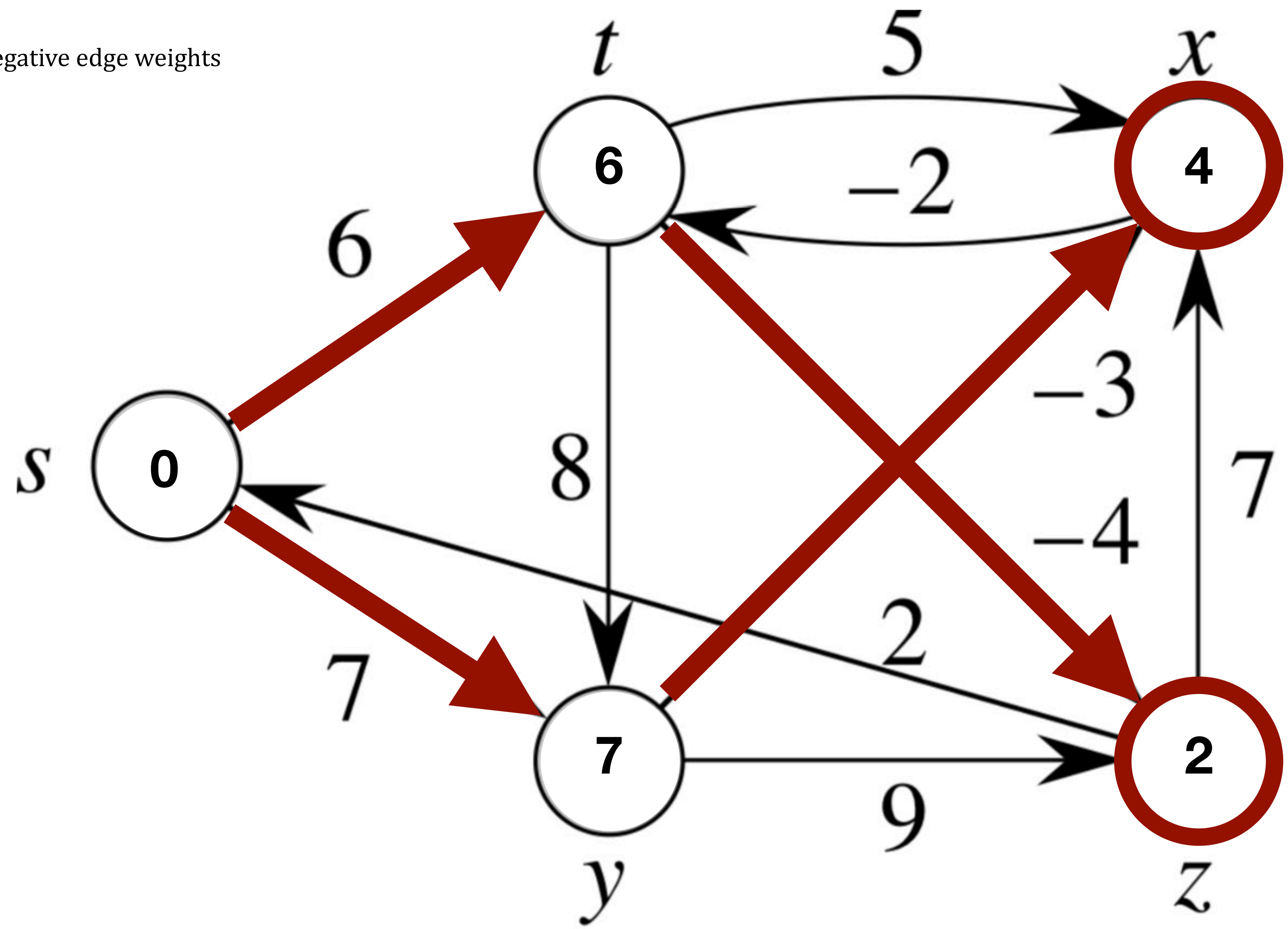
```
RELAX(u, v, w)
  if v.d > u.d + w(u, v)
      v.d = u.d + w(u, v)
      v.π = u
```

The Bellman-Ford Single-Source Shortest Path Algorithm

Graph G

a weighted, directed graph with negative edge weights

G.V = s, t, x, y, z
G.E = (t, x : w = 5)
(t, y : w = 8)
(t, z : w = -4)
(x, t : w = -2)
(y, x : w = -3)
(y, z : w = 9)
(z, x : w = 7)
(z, s : w = 2)
(s, t : w = 6)
(s, y : w = 7)




```
BELLMAN-FORD(G, w, s)
  INIT-SINGLE-SOURCE(G, s)
  for i = 2 to |G.V| − 1
      for each edge (u, v) ∈ G.E
          RELAX(u, v, w)
  for each edge (u, v) ∈ G.E
      if v.d > u.d + w(u, v)
          return FALSE
  return TRUE
```

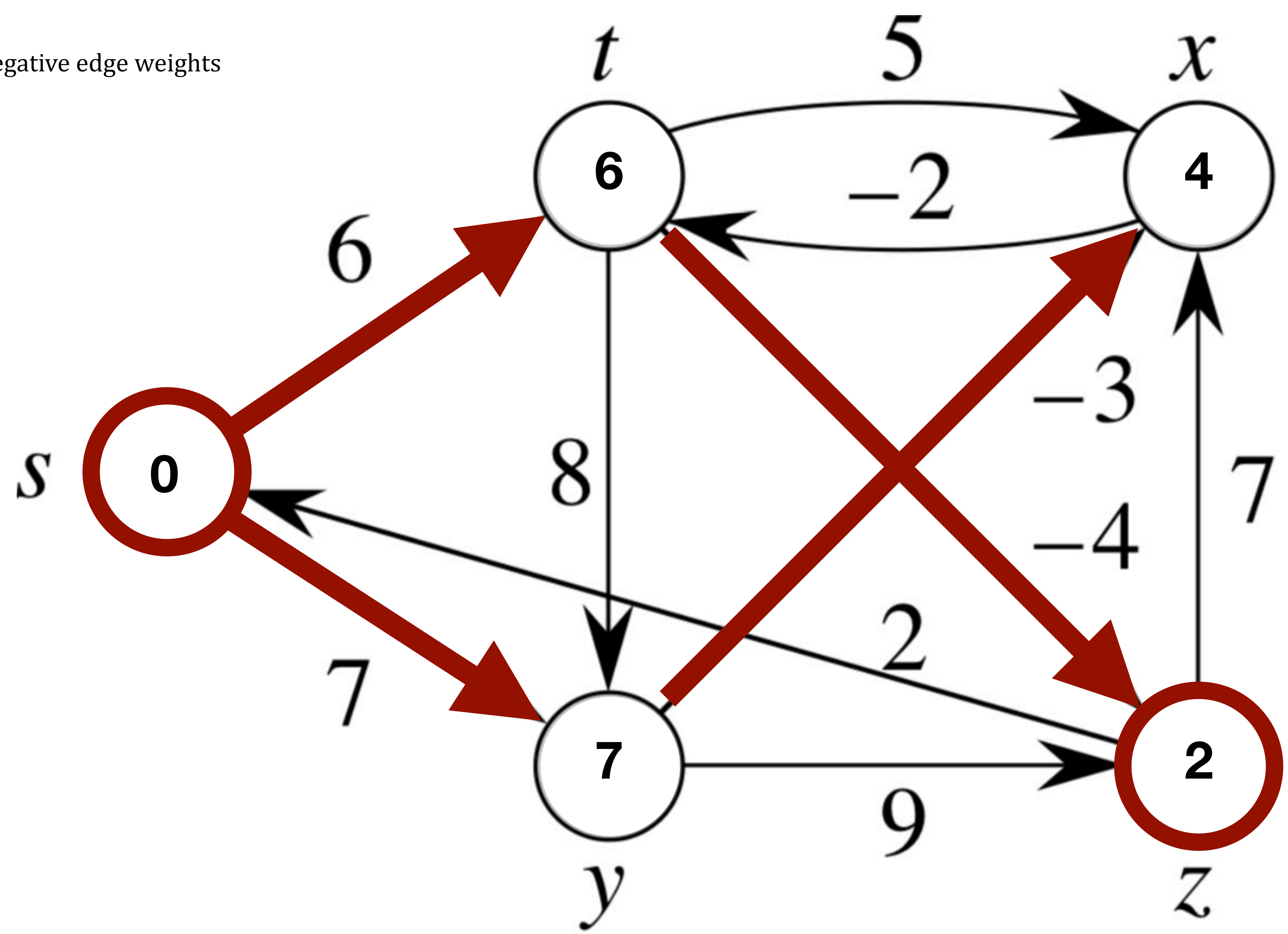
```
RELAX(u, v, w)
  if v.d > u.d + w(u, v)
      v.d = u.d + w(u, v)
      v.π = u
```

The Bellman-Ford Single-Source Shortest Path Algorithm

Graph G

a weighted, directed graph with negative edge weights

G.V = s, t, x, y, z
G.E = (t, x : w = 5)
(t, y : w = 8)
(t, z : w = -4)
(x, t : w = -2)
(y, x : w = -3)
(y, z : w = 9)
(z, x : w = 7)
(z, s : w = 2)
(s, t : w = 6)
(s, y : w = 7)



BELLMAN-FORD(G, w, s)
INIT-SINGLE-SOURCE(G, s)
for $i = 2$ to $|G.V| - 1$
for each edge $(u, v) \in G.E$
RELAX(u, v, w)
for each edge $(u, v) \in G.E$
if $v.d > u.d + w(u, v)$
return FALSE
return TRUE

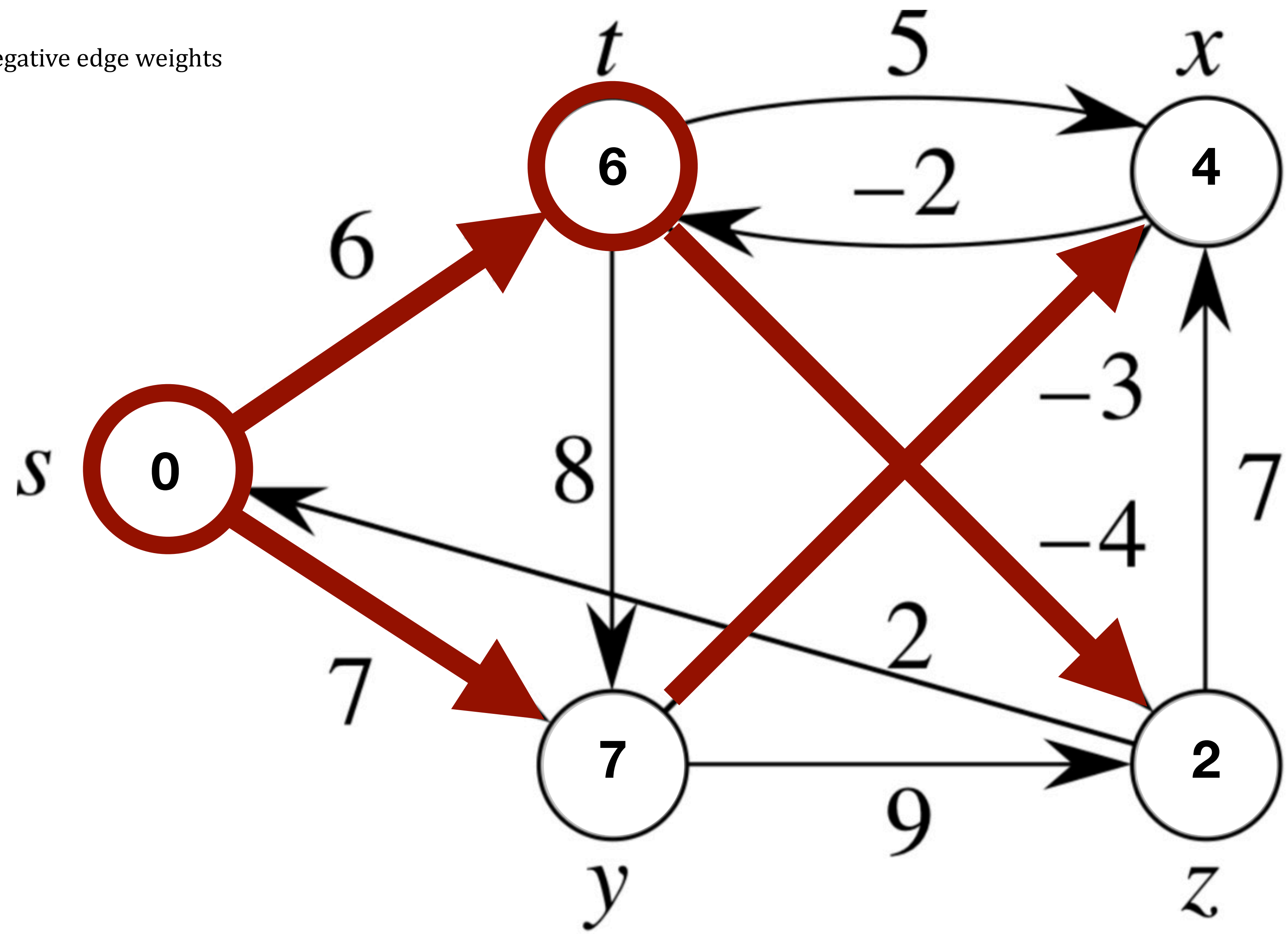
RELAX(u, v, w)
if $v.d > u.d + w(u, v)$
$v.d = u.d + w(u, v)$
$v.\pi = u$

The Bellman-Ford Single-Source Shortest Path Algorithm

Graph G

a weighted, directed graph with negative edge weights

G.V = s, t, x, y, z
G.E = (t, x : w = 5)
(t, y : w = 8)
(t, z : w = -4)
(x, t : w = -2)
(y, x : w = -3)
(y, z : w = 9)
(z, x : w = 7)
(z, s : w = 2)
(s, t : w = 6)
(s, y : w = 7)




```
BELLMAN-FORD(G, w, s)
  INIT-SINGLE-SOURCE(G, s)
  for i = 2 to |G.V| − 1
      for each edge (u, v) ∈ G.E
          RELAX(u, v, w)
  for each edge (u, v) ∈ G.E
      if v.d > u.d + w(u, v)
          return FALSE
  return TRUE
```

```
RELAX(u, v, w)
  if v.d > u.d + w(u, v)
      v.d = u.d + w(u, v)
      v.π = u
```

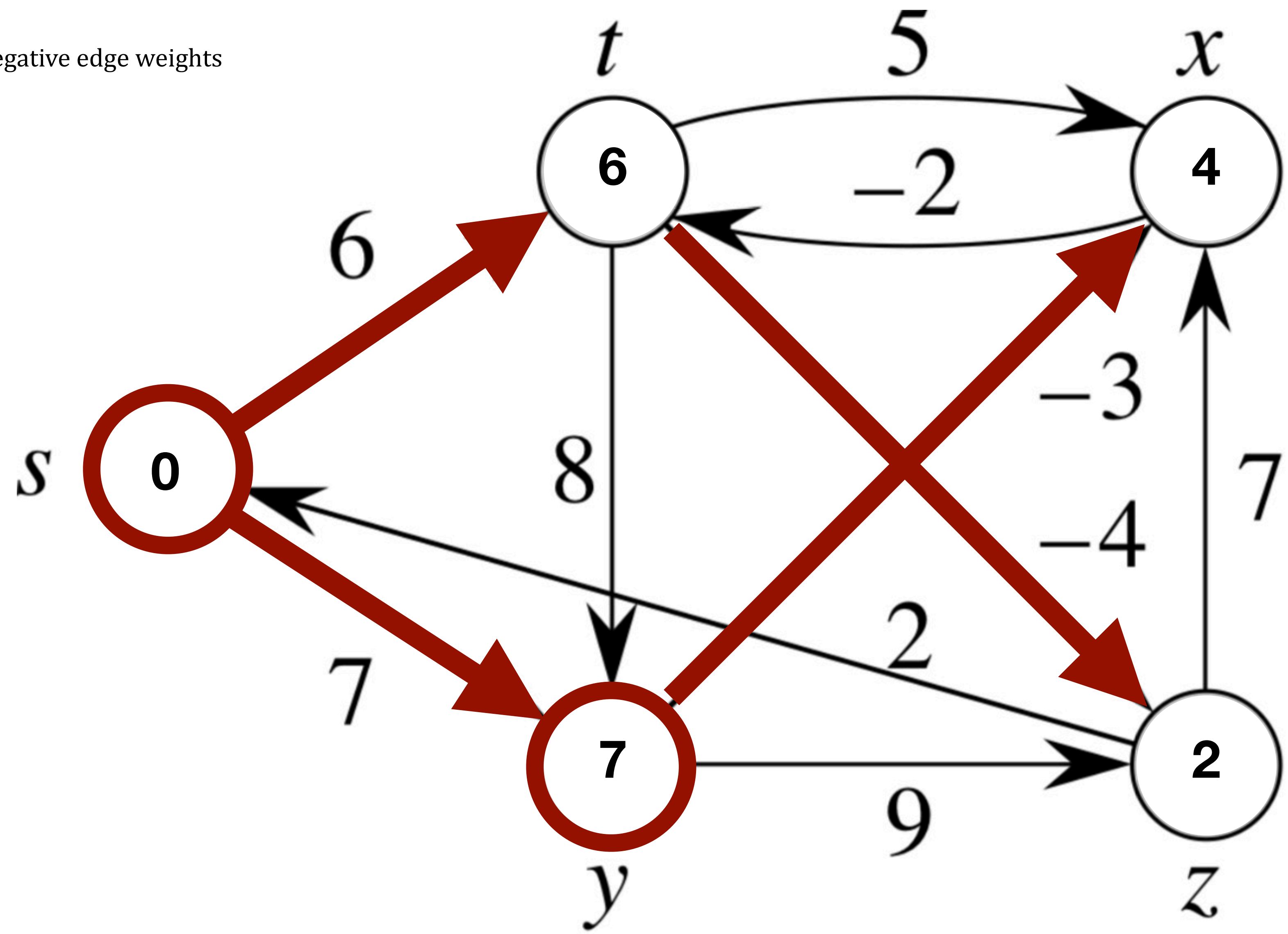
The Bellman-Ford Single-Source Shortest Path Algorithm

Graph G

a weighted, directed graph with negative edge weights

G.V = s, t, x, y, z

G.E = (t, x : w = 5)
(t, y : w = 8)
(t, z : w = -4)
(x, t : w = -2)
(y, x : w = -3)
(y, z : w = 9)
(z, x : w = 7)
(z, s : w = 2)
(s, t : w = 6)
(s, y : w = 7)



BELLMAN-FORD(G, w, s)
  INIT-SINGLE-SOURCE(G, s)
  for i = **2** to $|G.V| - 1$
    for each edge $(u, v) \in G.E$
      RELAX(u, v, w)
  for each edge $(u, v) \in G.E$
    if $v.d > u.d + w(u, v)$
      return FALSE
  return TRUE

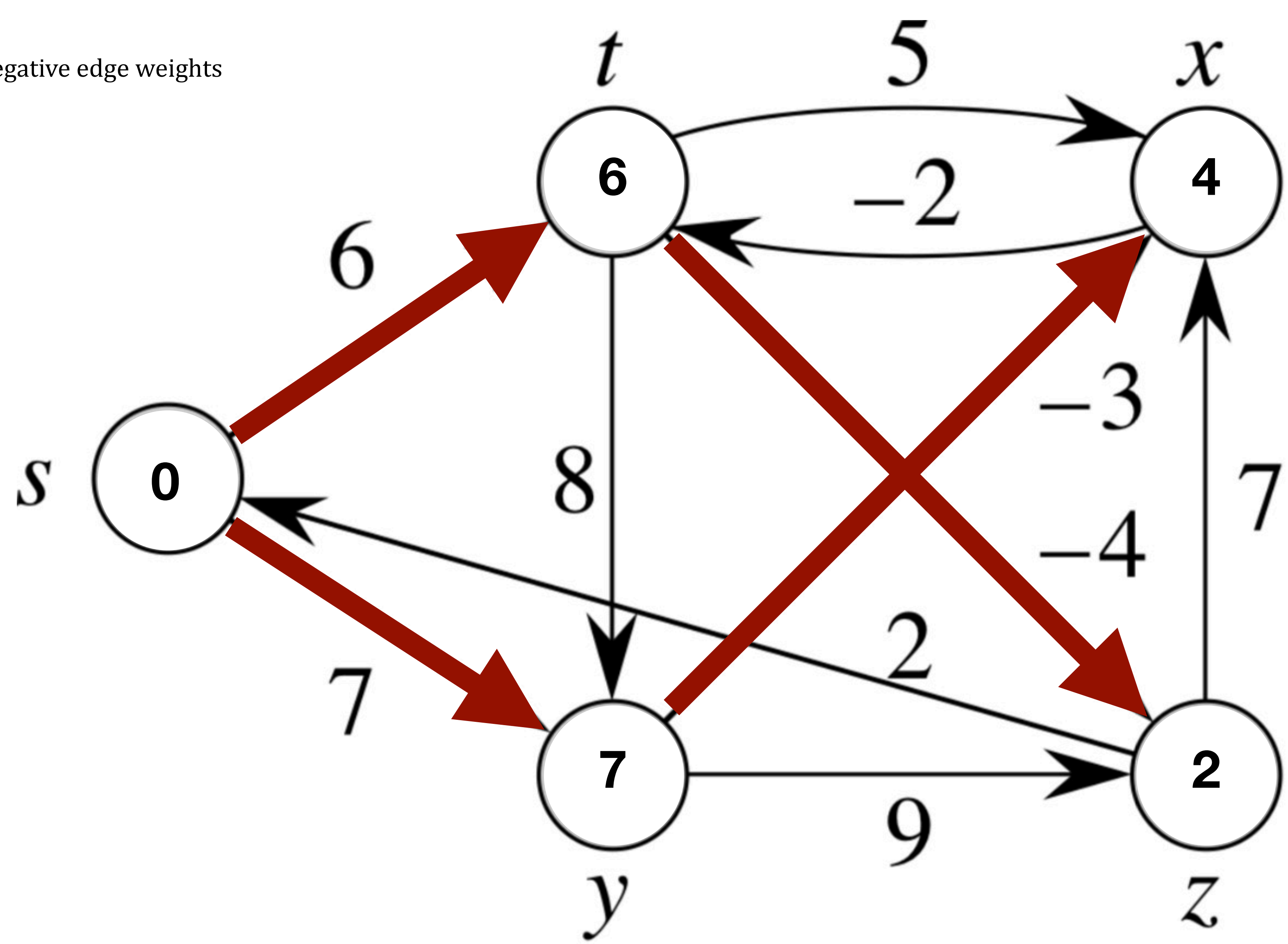
RELAX(u, v, w)
  if $v.d > u.d + w(u, v)$
    $v.d = u.d + w(u, v)$
    $v.\pi = u$

The Bellman-Ford Single-Source Shortest Path Algorithm

Graph G

a weighted, directed graph with negative edge weights

G.V = s, t, x, y, z
G.E = (t, x : *w* = 5)
(t, y : *w* = 8)
(t, z : *w* = -4)
(x, t : *w* = -2)
(y, x : *w* = -3)
(y, z : *w* = 9)
(z, x : *w* = 7)
(z, s : *w* = 2)
(s, t : *w* = 6)
(s, y : *w* = 7)




```
BELLMAN-FORD(G, w, s)
  INIT-SINGLE-SOURCE(G, s)
  for i = 3 to |G.V| − 1
      for each edge (u, v) ∈ G.E
          RELAX(u, v, w)
  for each edge (u, v) ∈ G.E
      if v.d > u.d + w(u, v)
          return FALSE
  return TRUE
```

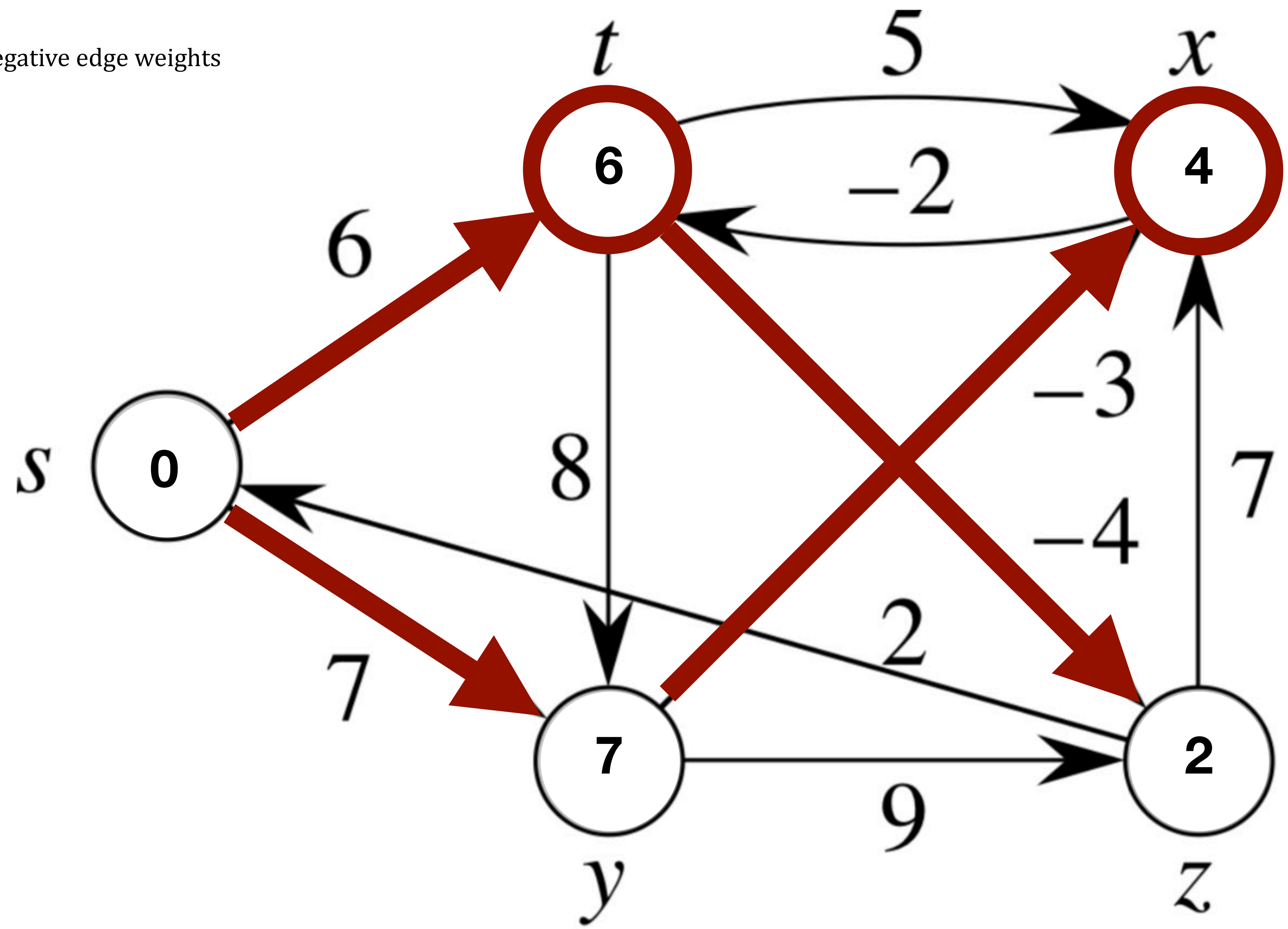
```
RELAX(u, v, w)
  if v.d > u.d + w(u, v)
      v.d = u.d + w(u, v)
      v.π = u
```

The Bellman-Ford Single-Source Shortest Path Algorithm

Graph G

a weighted, directed graph with negative edge weights

G.V = s, t, x, y, z
G.E = (t, x : w = 5)
(t, y : w = 8)
(t, z : w = -4)
(x, t : w = -2)
(y, x : w = -3)
(y, z : w = 9)
(z, x : w = 7)
(z, s : w = 2)
(s, t : w = 6)
(s, y : w = 7)




```
BELLMAN-FORD(G, w, s)
  INIT-SINGLE-SOURCE(G, s)
  for i = 3 to |G.V| − 1
      for each edge (u, v) ∈ G.E
          RELAX(u, v, w)
  for each edge (u, v) ∈ G.E
      if v.d > u.d + w(u, v)
          return FALSE
  return TRUE
```

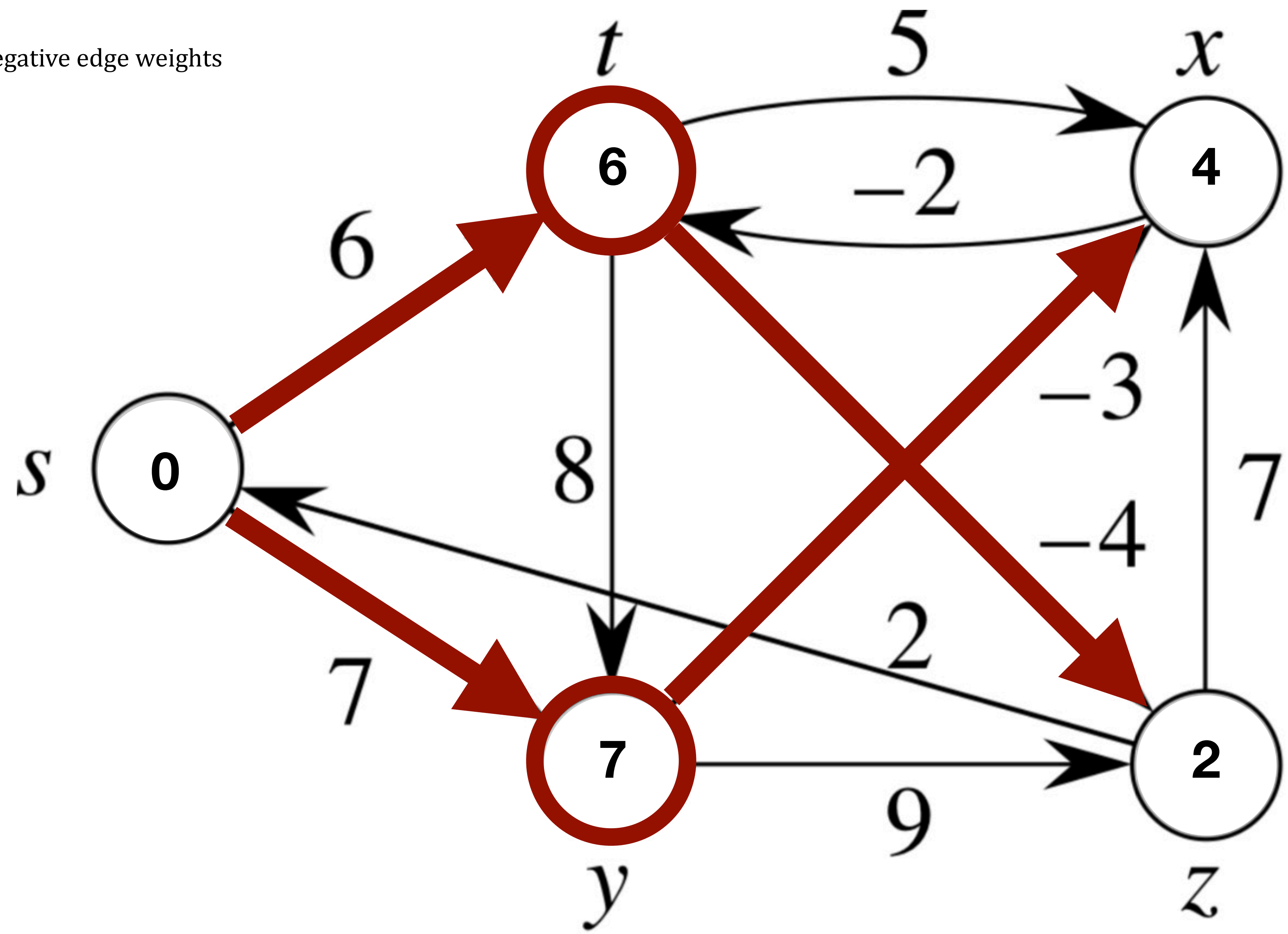
```
RELAX(u, v, w)
  if v.d > u.d + w(u, v)
      v.d = u.d + w(u, v)
      v.π = u
```

The Bellman-Ford Single-Source Shortest Path Algorithm

Graph G

a weighted, directed graph with negative edge weights

G.V = s, t, x, y, z
G.E = (t, x : w = 5)
(t, y : w = 8)
(t, z : w = -4)
(x, t : w = -2)
(y, x : w = -3)
(y, z : w = 9)
(z, x : w = 7)
(z, s : w = 2)
(s, t : w = 6)
(s, y : w = 7)




```
BELLMAN-FORD(G, w, s)
  INIT-SINGLE-SOURCE(G, s)
  for i = 3 to |G.V| − 1
      for each edge (u, v) ∈ G.E
          RELAX(u, v, w)
  for each edge (u, v) ∈ G.E
      if v.d > u.d + w(u, v)
          return FALSE
  return TRUE
```

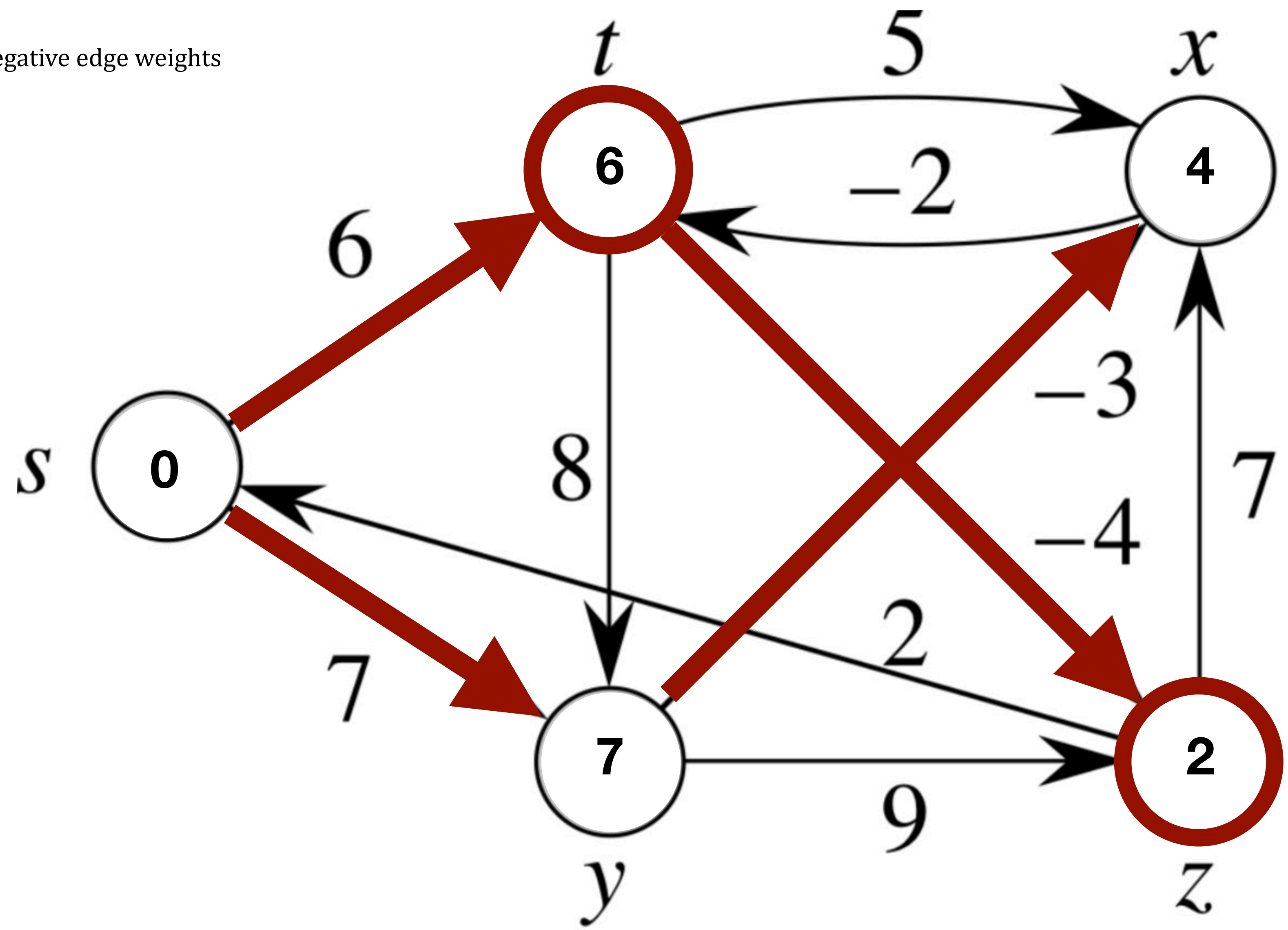
```
RELAX(u, v, w)
  if v.d > u.d + w(u, v)
      v.d = u.d + w(u, v)
      v.π = u
```

The Bellman-Ford Single-Source Shortest Path Algorithm

Graph G

a weighted, directed graph with negative edge weights

G.V = s, t, x, y, z
G.E = (t, x : w = 5)
(t, y : w = 8)
(t, z : w = -4)
(x, t : w = -2)
(y, x : w = -3)
(y, z : w = 9)
(z, x : w = 7)
(z, s : w = 2)
(s, t : w = 6)
(s, y : w = 7)




```
BELLMAN-FORD(G, w, s)
  INIT-SINGLE-SOURCE(G, s)
  for i = 3 to |G.V| − 1
      for each edge (u, v) ∈ G.E
          RELAX(u, v, w)
  for each edge (u, v) ∈ G.E
      if v.d > u.d + w(u, v)
          return FALSE
  return TRUE
```

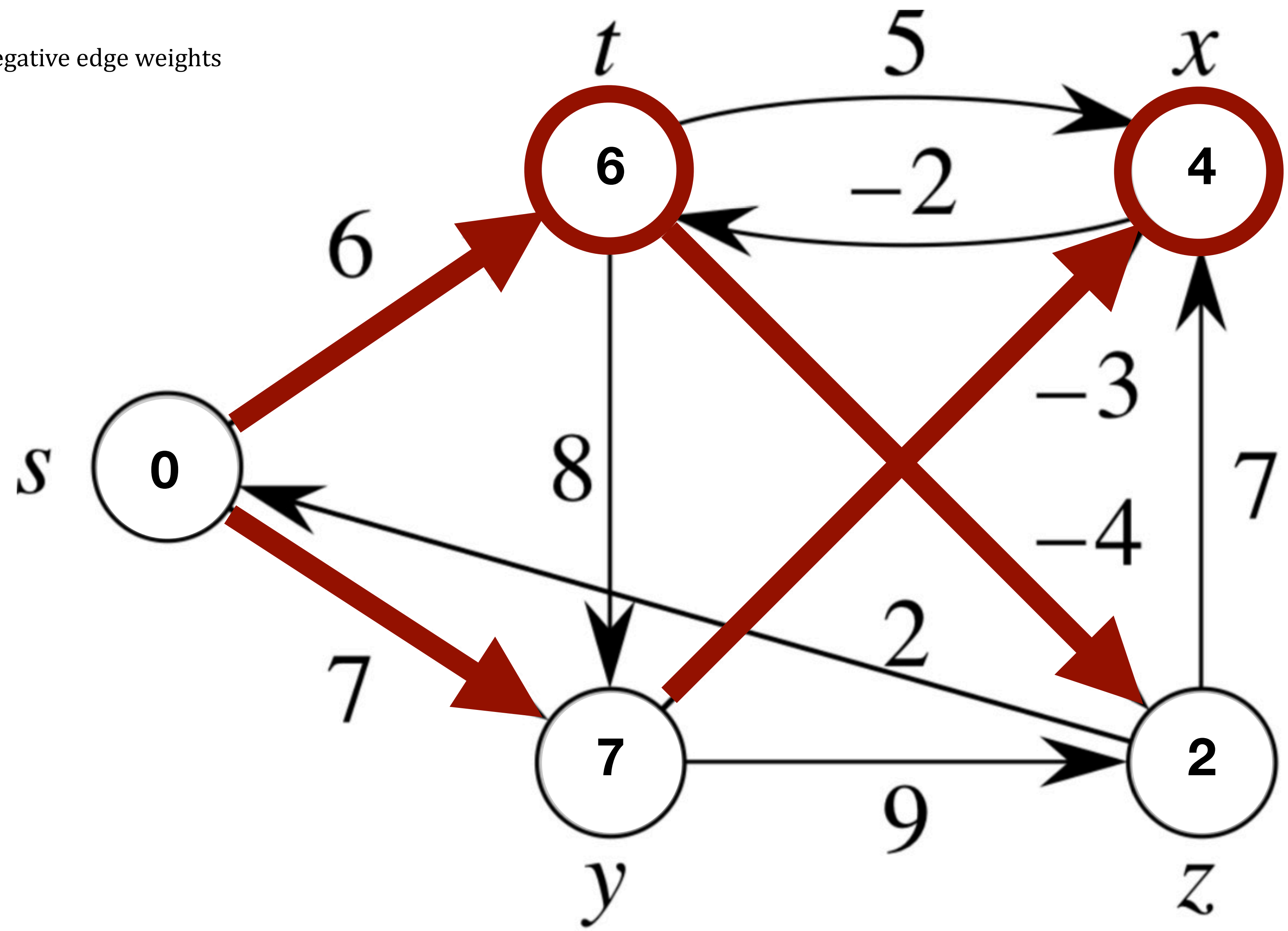
```
RELAX(u, v, w)
  if v.d > u.d + w(u, v)
      v.d = u.d + w(u, v)
      v.π = u
```

The Bellman-Ford Single-Source Shortest Path Algorithm

Graph G

a weighted, directed graph with negative edge weights

G.V = s, t, x, y, z
G.E = (t, x : w = 5)
(t, y : w = 8)
(t, z : w = -4)
(x, t : w = -2)
(y, x : w = -3)
(y, z : w = 9)
(z, x : w = 7)
(z, s : w = 2)
(s, t : w = 6)
(s, y : w = 7)



Remember,
these are estimates
until we finish.
We CAN do better!

BELLMAN-FORD(G, w, s)
INIT-SINGLE-SOURCE(G, s)
for i = **3** to $|G.V| - 1$
for each edge $(u, v) \in G.E$
RELAX(u, v, w)
for each edge $(u, v) \in G.E$
if $v.d > u.d + w(u, v)$
return FALSE
return TRUE

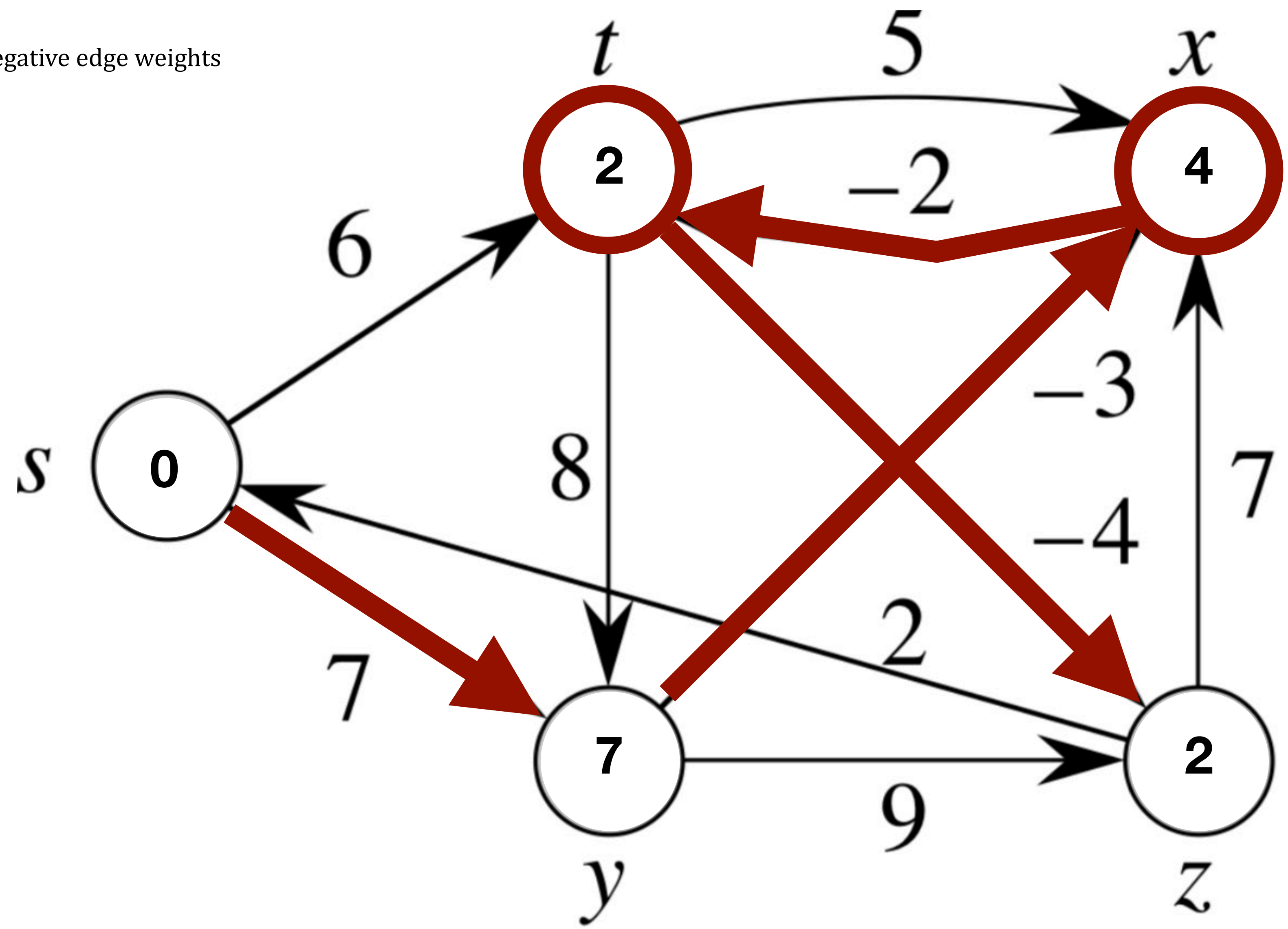
RELAX(u, v, w)
if $v.d > u.d + w(u, v)$
$v.d = u.d + w(u, v)$
$v.\pi = u$

The Bellman-Ford Single-Source Shortest Path Algorithm

Graph G

a weighted, directed graph with negative edge weights

G.V = s, t, x, y, z
G.E = (t, x : w = 5)
(t, y : w = 8)
(t, z : w = -4)
(x, t : w = -2)
(y, x : w = -3)
(y, z : w = 9)
(z, x : w = 7)
(z, s : w = 2)
(s, t : w = 6)
(s, y : w = 7)



Remember,
these are estimates
until we finish.
We CAN do better!

BELLMAN-FORD(G, w, s)
INIT-SINGLE-SOURCE(G, s)
for i = **3** to $|G.V| - 1$
  for each edge $(u, v) \in G.E$
    RELAX(u, v, w)
for each edge $(u, v) \in G.E$
  if $v.d > u.d + w(u, v)$
    return FALSE
return TRUE

RELAX(u, v, w)
if $v.d > u.d + w(u, v)$
  $v.d = u.d + w(u, v)$
  $v.\pi = u$

The Bellman-Ford Single-Source Shortest Path Algorithm

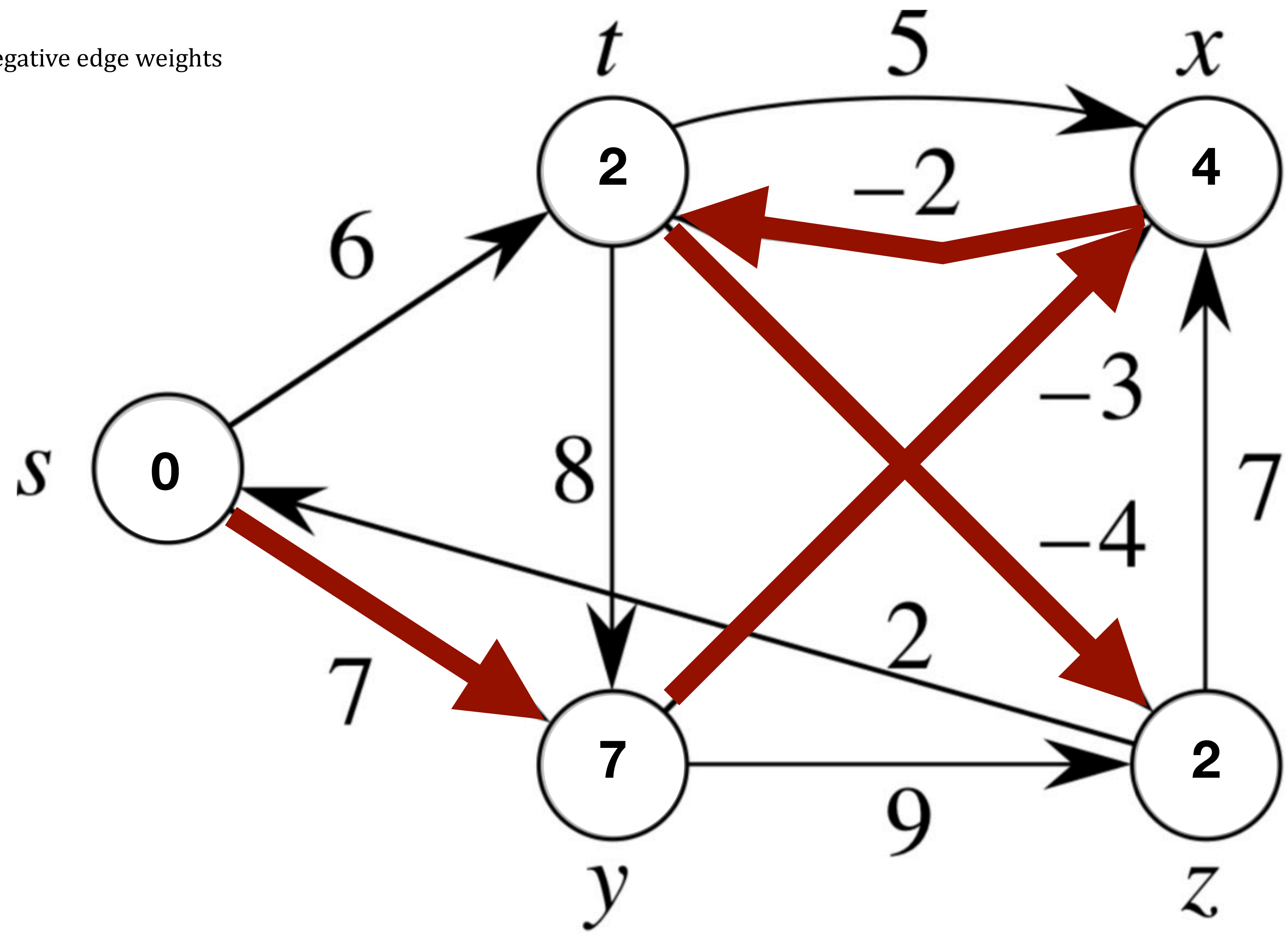
Graph G

a weighted, directed graph with negative edge weights

G.V = s, t, x, y, z

G.E = (t, x : w = 5)
(t, y : w = 8)
(t, z : w = -4)
(x, t : w = -2)
(y, x : w = -3)
(y, z : w = 9)
(z, x : w = 7)
(z, s : w = 2)
(s, t : w = 6)
(s, y : w = 7)

There are no changes after this point for the rest of this pass.
But we're not yet done.




```
BELLMAN-FORD(G, w, s)
  INIT-SINGLE-SOURCE(G, s)
  for i = 3 to |G.V| − 1
      for each edge (u, v) ∈ G.E
          RELAX(u, v, w)
  for each edge (u, v) ∈ G.E
      if v.d > u.d + w(u, v)
          return FALSE
  return TRUE
```

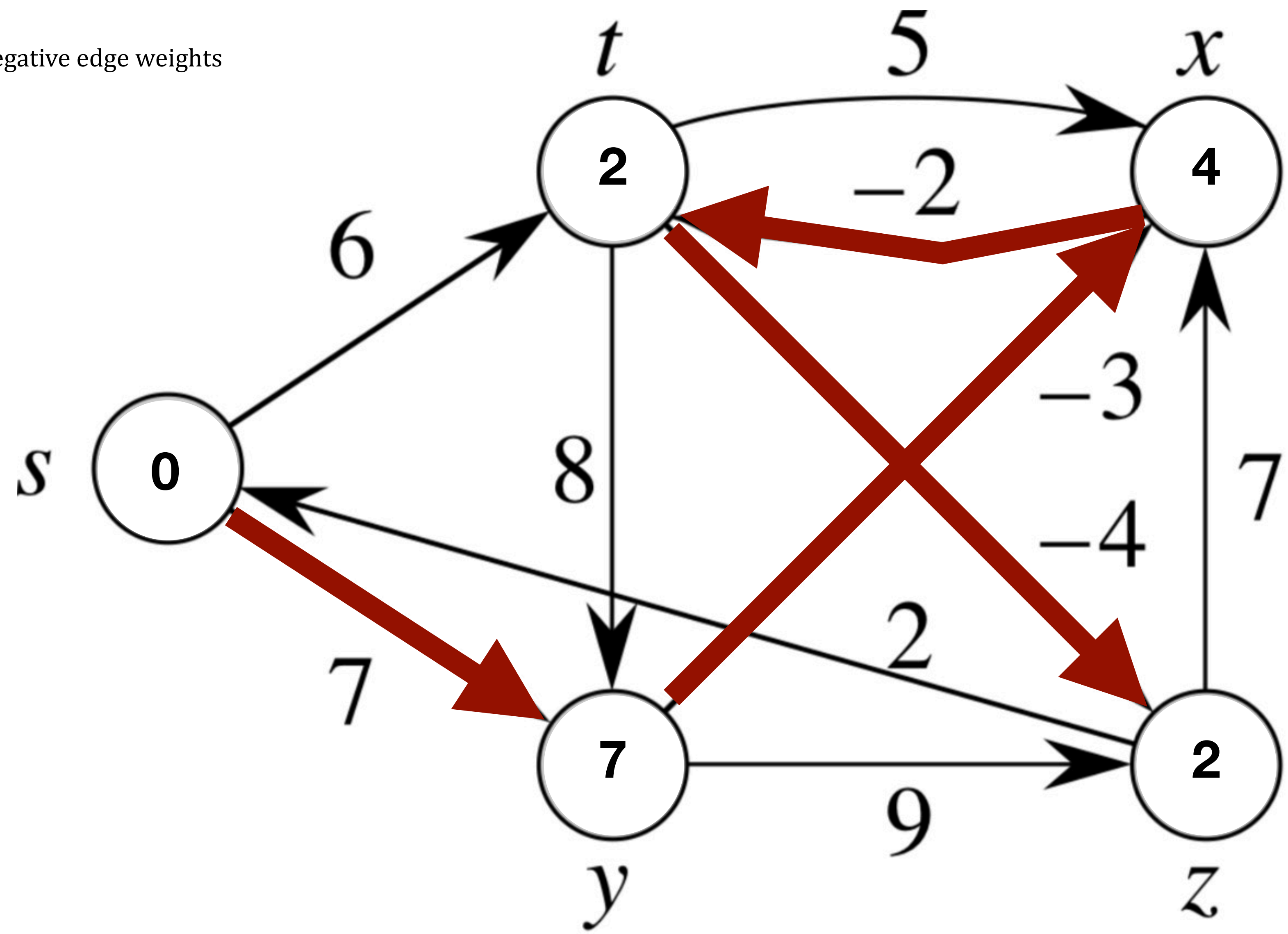
```
RELAX(u, v, w)
  if v.d > u.d + w(u, v)
      v.d = u.d + w(u, v)
      v.π = u
```

The Bellman-Ford Single-Source Shortest Path Algorithm

Graph G

a weighted, directed graph with negative edge weights

G.V = s, t, x, y, z
G.E = (t, x : w = 5)
(t, y : w = 8)
(t, z : w = -4)
(x, t : w = -2)
(y, x : w = -3)
(y, z : w = 9)
(z, x : w = 7)
(z, s : w = 2)
(s, t : w = 6)
(s, y : w = 7)




```
BELLMAN-FORD(G, w, s)
  INIT-SINGLE-SOURCE(G, s)
  for i = 4 to |G.V| − 1
      for each edge (u, v) ∈ G.E
          RELAX(u, v, w)
  for each edge (u, v) ∈ G.E
      if v.d > u.d + w(u, v)
          return FALSE
  return TRUE
```

```
RELAX(u, v, w)
  if v.d > u.d + w(u, v)
      v.d = u.d + w(u, v)
      v.π = u
```

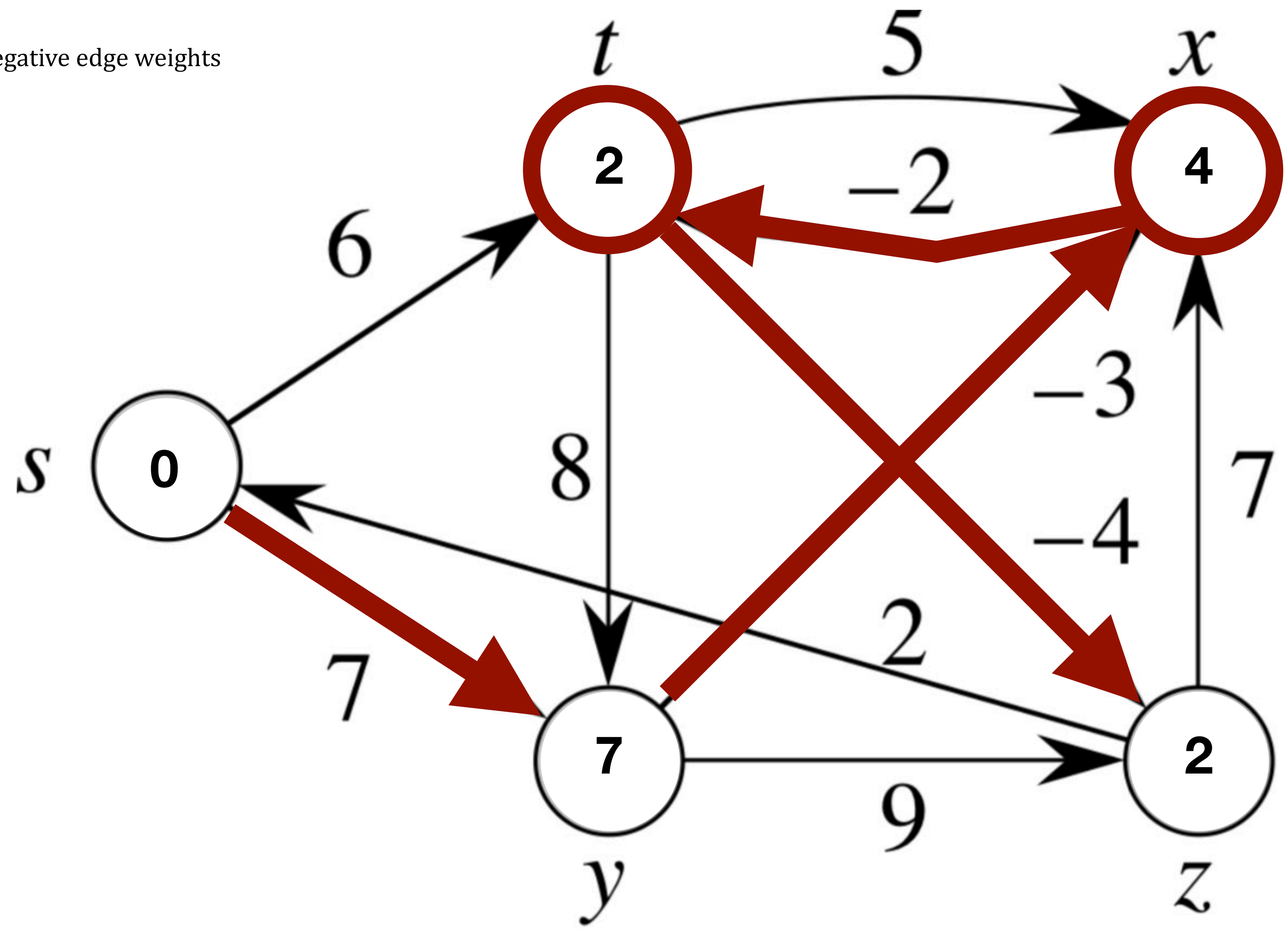
The Bellman-Ford Single-Source Shortest Path Algorithm

Graph G

a weighted, directed graph with negative edge weights

G.V = s, t, x, y, z

G.E = (t, x : w = 5)
(t, y : w = 8)
(t, z : w = -4)
(x, t : w = -2)
(y, x : w = -3)
(y, z : w = 9)
(z, x : w = 7)
(z, s : w = 2)
(s, t : w = 6)
(s, y : w = 7)




```
BELLMAN-FORD(G, w, s)
  INIT-SINGLE-SOURCE(G, s)
  for i = 4 to |G.V| − 1
      for each edge (u, v) ∈ G.E
          RELAX(u, v, w)
  for each edge (u, v) ∈ G.E
      if v.d > u.d + w(u, v)
          return FALSE
  return TRUE
```

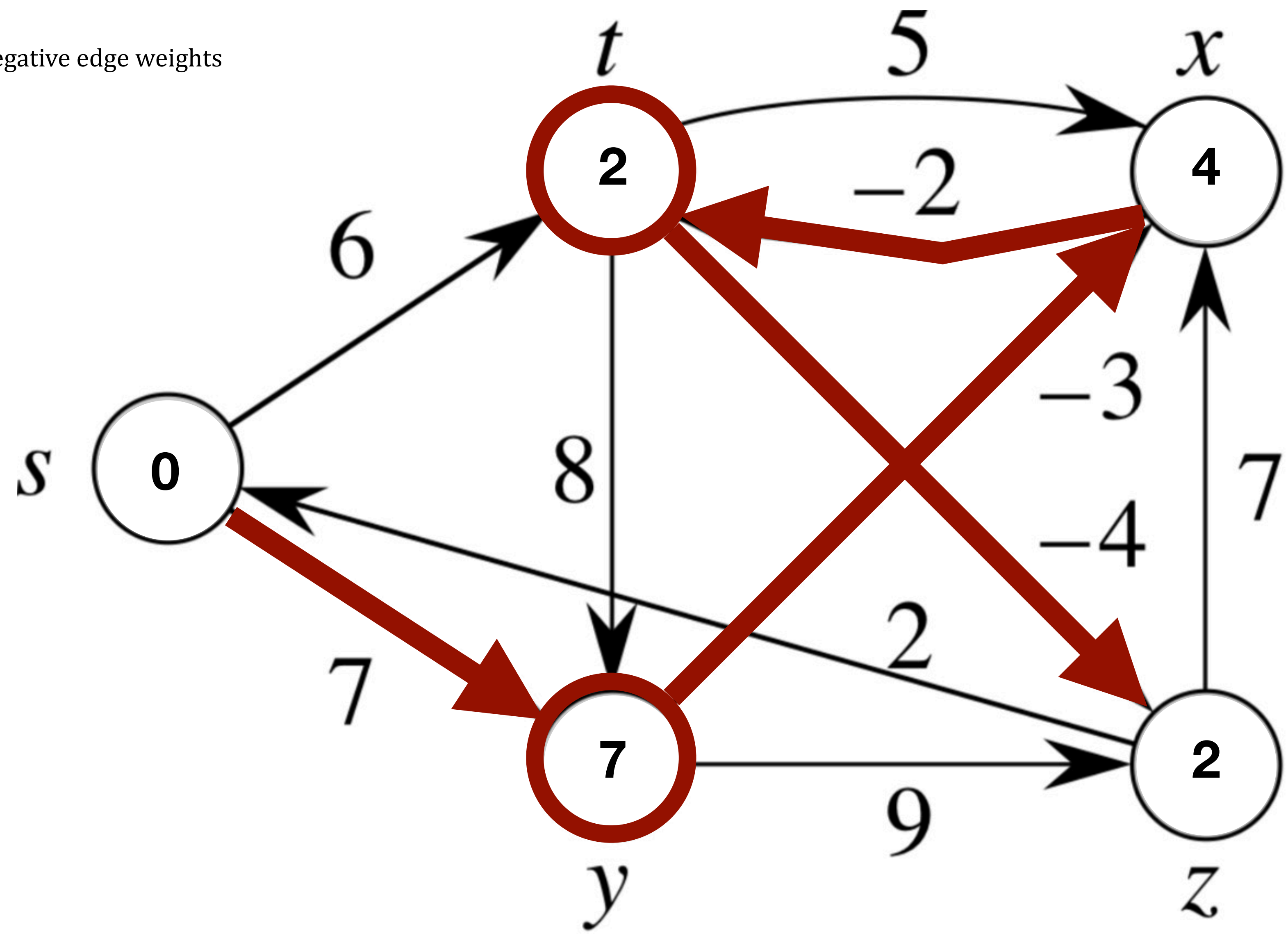
```
RELAX(u, v, w)
  if v.d > u.d + w(u, v)
      v.d = u.d + w(u, v)
      v.π = u
```

The Bellman-Ford Single-Source Shortest Path Algorithm

Graph G

a weighted, directed graph with negative edge weights

G.V = s, t, x, y, z
G.E = (t, x : w = 5)
(t, y : w = 8)
(t, z : w = -4)
(x, t : w = -2)
(y, x : w = -3)
(y, z : w = 9)
(z, x : w = 7)
(z, s : w = 2)
(s, t : w = 6)
(s, y : w = 7)



BELLMAN-FORD(G, w, s)
INIT-SINGLE-SOURCE(G, s)
for i = **4** to $|G.V| - 1$
for each edge $(u, v) \in G.E$
RELAX(u, v, w)
for each edge $(u, v) \in G.E$
if $v.d > u.d + w(u, v)$
return FALSE
return TRUE

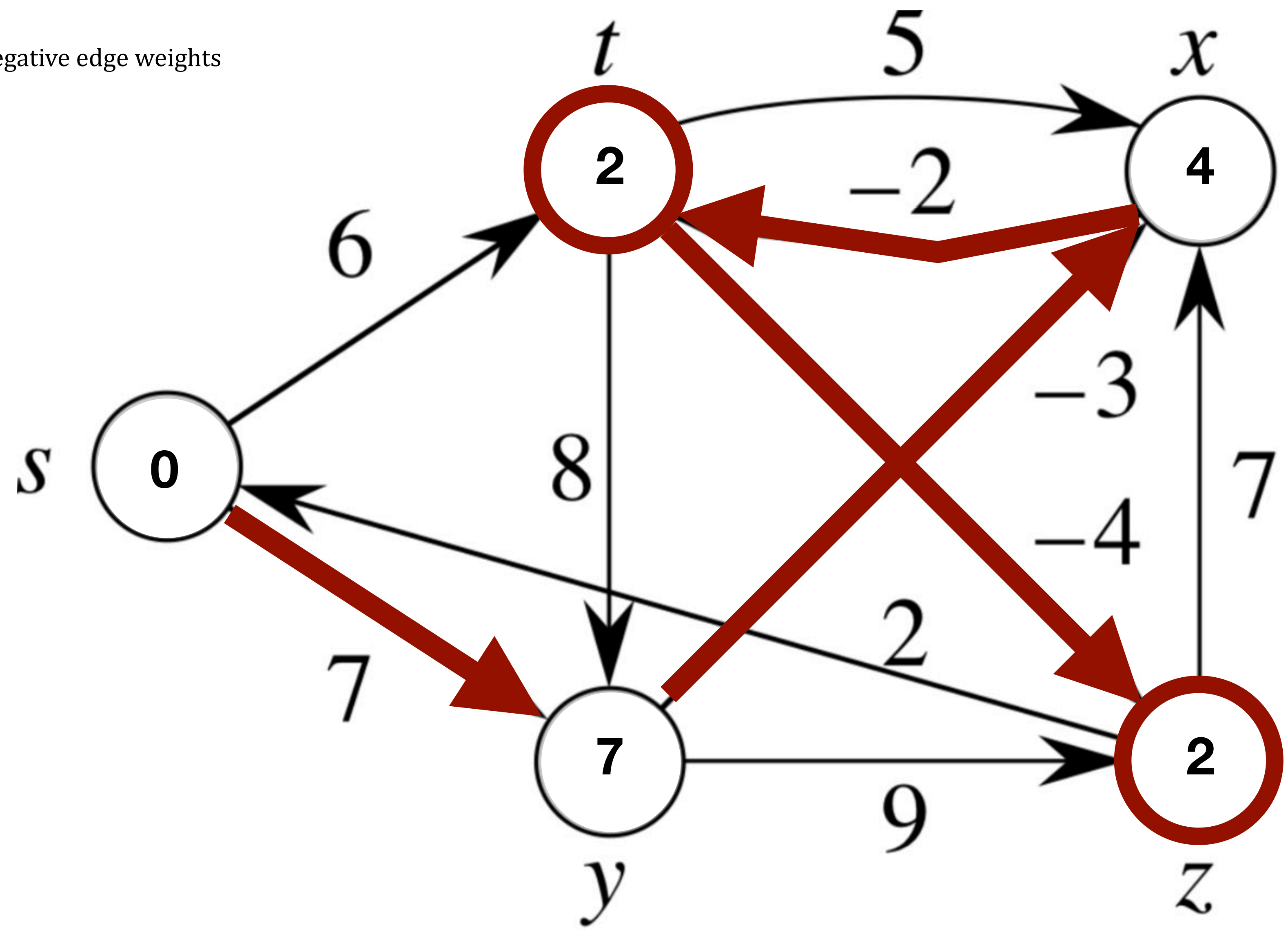
RELAX(u, v, w)
if $v.d > u.d + w(u, v)$
$v.d = u.d + w(u, v)$
$v.\pi = u$

The Bellman-Ford Single-Source Shortest Path Algorithm

Graph G

a weighted, directed graph with negative edge weights

G.V = s, t, x, y, z
G.E = (t, x : *w* = 5)
(t, y : *w* = 8)
(t, z : *w* = -4)
(x, t : *w* = -2)
(y, x : *w* = -3)
(y, z : *w* = 9)
(z, x : *w* = 7)
(z, s : *w* = 2)
(s, t : *w* = 6)
(s, y : *w* = 7)



Ooohhh . . . .

BELLMAN-FORD(G, w, s)
INIT-SINGLE-SOURCE(G, s)
for $i = \mathbf{4}$ to $|G.V| - 1$
for each edge $(u, v) \in G.E$
RELAX(u, v, w)
for each edge $(u, v) \in G.E$
if $v.d > u.d + w(u, v)$
return FALSE
return TRUE

RELAX(u, v, w)
if $v.d > u.d + w(u, v)$
$v.d = u.d + w(u, v)$
$v.\pi = u$

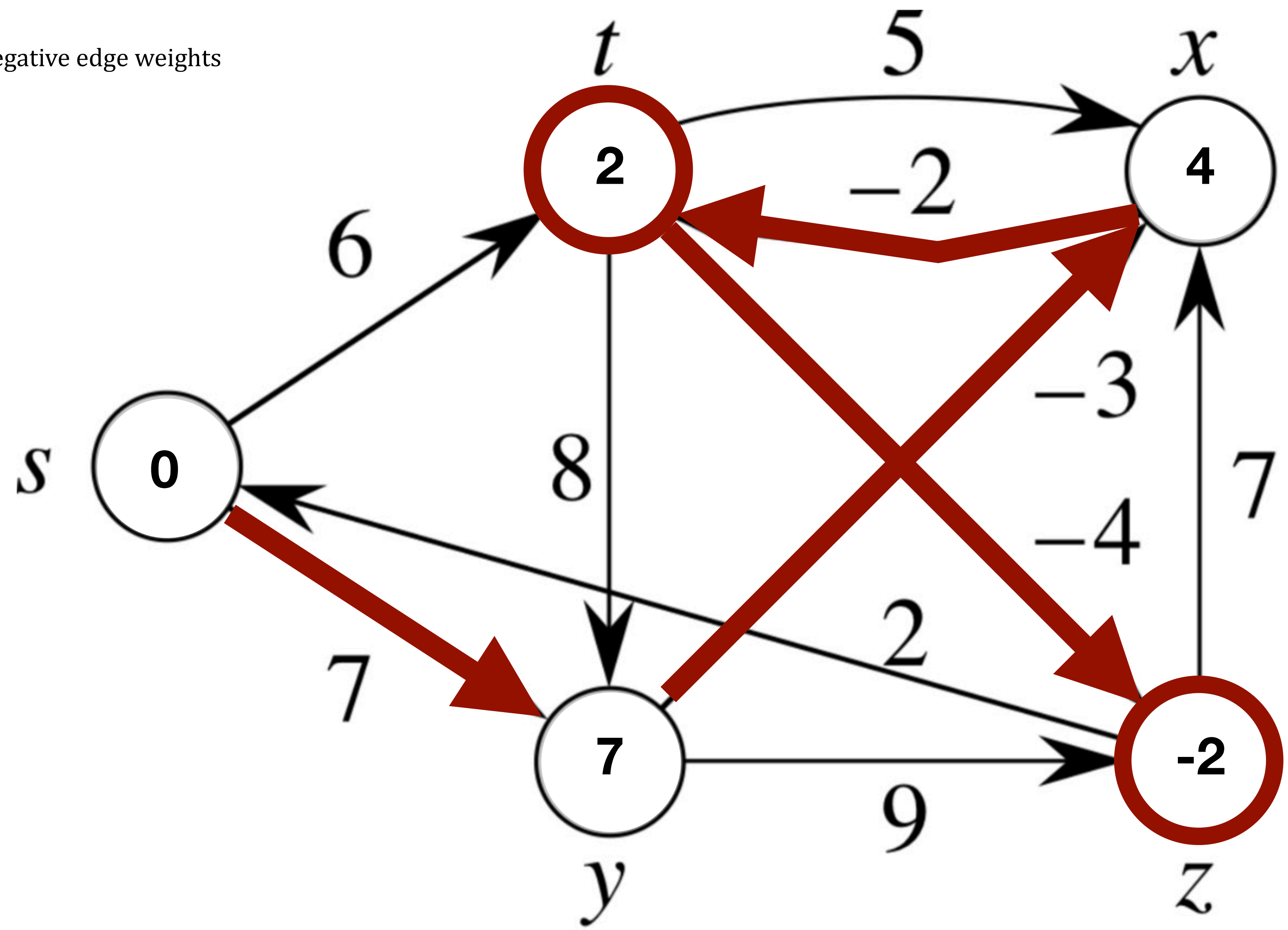
The Bellman-Ford Single-Source Shortest Path Algorithm

Graph G

a weighted, directed graph with negative edge weights

G.V = s, t, x, y, z

G.E = (t, x : w = 5)
(t, y : w = 8)
(t, z : w = -4)
(x, t : w = -2)
(y, x : w = -3)
(y, z : w = 9)
(z, x : w = 7)
(z, s : w = 2)
(s, t : w = 6)
(s, y : w = 7)



Oooohhh . . . .

BELLMAN-FORD(G, w, s)
INIT-SINGLE-SOURCE(G, s)
for i = **4** to $|G.V| - 1$
for each edge $(u, v) \in G.E$
RELAX(u, v, w)
for each edge $(u, v) \in G.E$
if $v.d > u.d + w(u, v)$
return FALSE
return TRUE

RELAX(u, v, w)
if $v.d > u.d + w(u, v)$
$v.d = u.d + w(u, v)$
$v.\pi = u$

The Bellman-Ford Single-Source Shortest Path Algorithm

Graph G

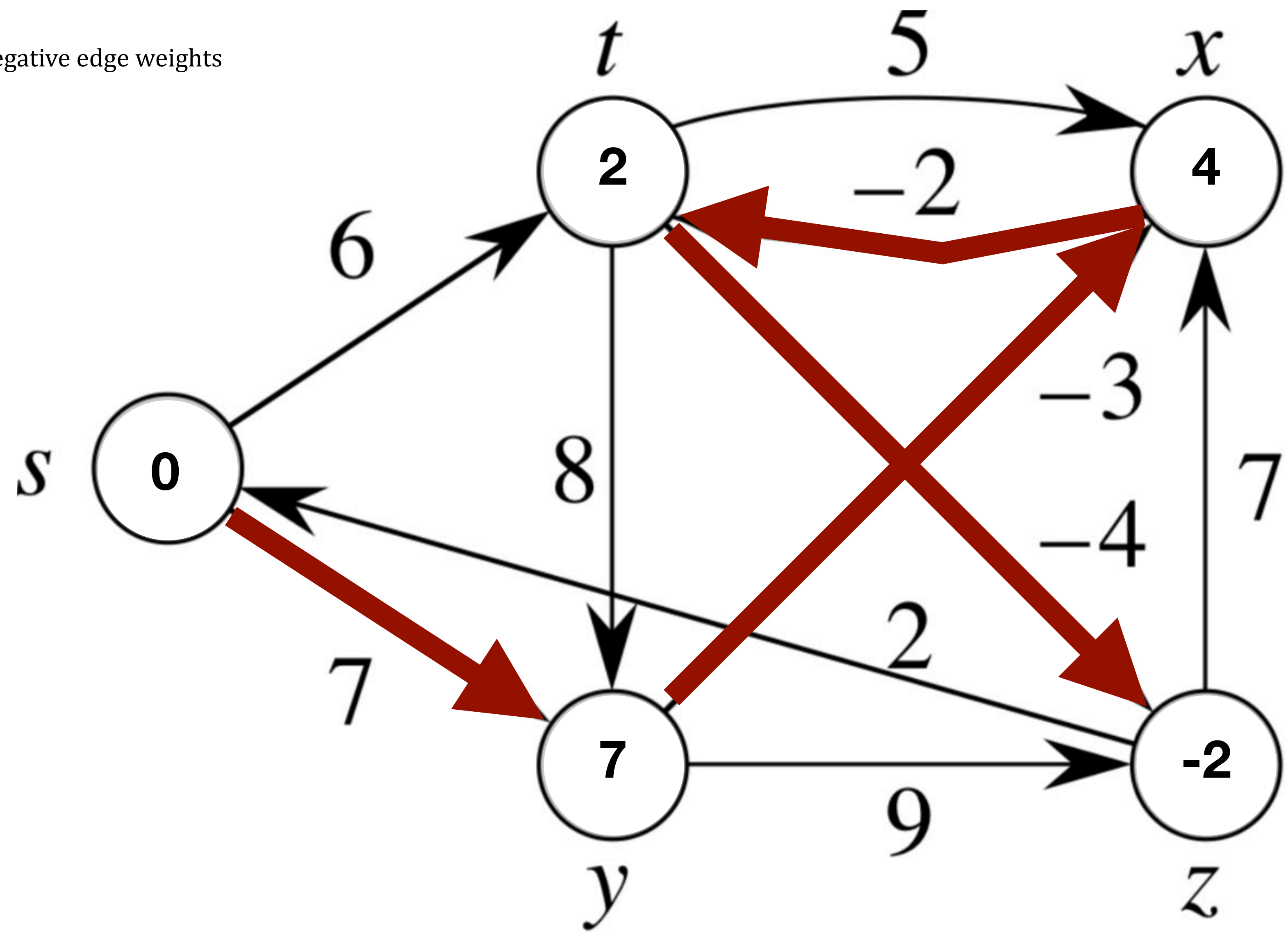
a weighted, directed graph with negative edge weights

G.V = s, t, x, y, z

G.E = (t, x : w = 5)
(t, y : w = 8)
(t, z : w = -4)
(x, t : w = -2)
(y, x : w = -3)
(y, z : w = 9)
(z, x : w = 7)
(z, s : w = 2)
(s, t : w = 6)
(s, y : w = 7)

There are no changes after this point for the rest of this pass.
And now we're done.

Almost.



BELLMAN-FORD(G, w, s)
INIT-SINGLE-SOURCE(G, s)
for $i = \mathbf{4}$ to $|G.V| - 1$
  for each edge $(u, v) \in G.E$
    RELAX(u, v, w)
for each edge $(u, v) \in G.E$
  if $v.d > u.d + w(u, v)$
    return FALSE
return TRUE

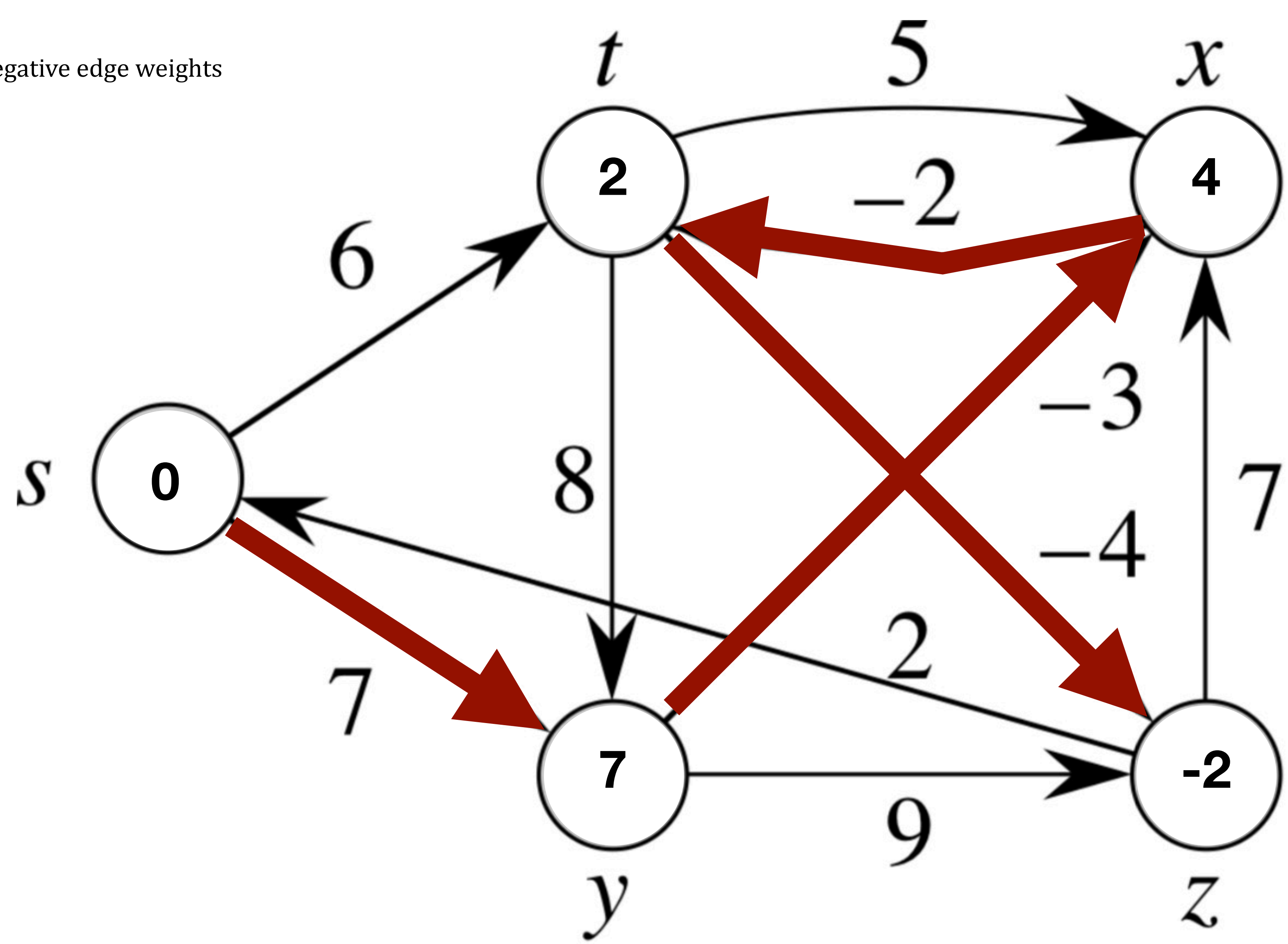
RELAX(u, v, w)
if $v.d > u.d + w(u, v)$
  $v.d = u.d + w(u, v)$
  $v.\pi = u$

The Bellman-Ford Single-Source Shortest Path Algorithm

Graph G

a weighted, directed graph with negative edge weights

G.V = s, t, x, y, z
G.E = (t, x : w = 5)
(t, y : w = 8)
(t, z : w = -4)
(x, t : w = -2)
(y, x : w = -3)
(y, z : w = 9)
(z, x : w = 7)
(z, s : w = 2)
(s, t : w = 6)
(s, y : w = 7)



BELLMAN-FORD(G, w, s)
INIT-SINGLE-SOURCE(G, s)
for i = **4** to $|G.V| - 1$
for each edge $(u, v) \in G.E$
RELAX(u, v, w)
for each edge $(u, v) \in G.E$
if $v.d > u.d + w(u, v)$
return FALSE
return TRUE

What does this do?

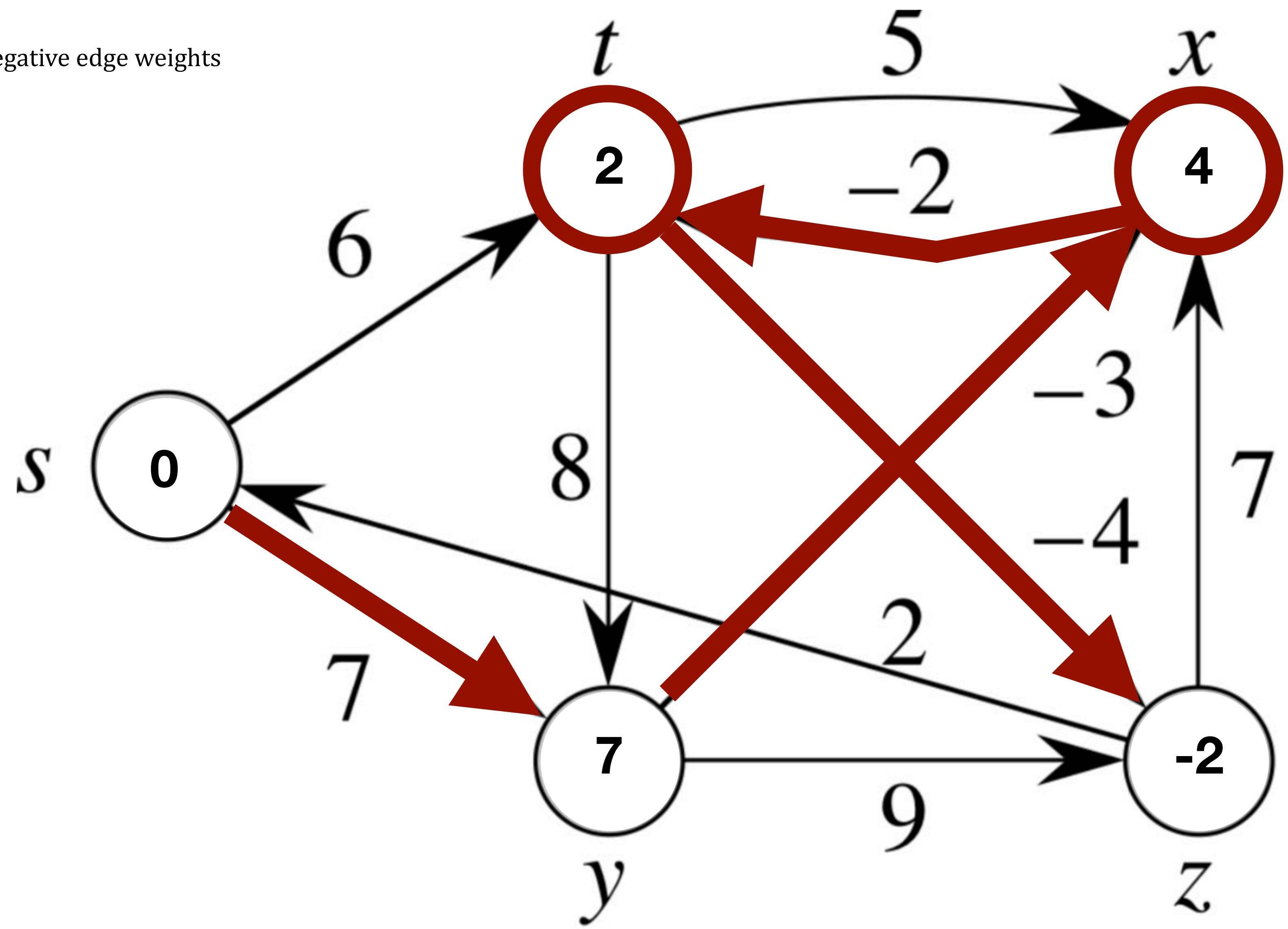
The Bellman-Ford Single-Source Shortest Path Algorithm

Graph G

a weighted, directed graph with negative edge weights

G.V = s, t, x, y, z

G.E = (t, x : w = 5)
(t, y : w = 8)
(t, z : w = -4)
(x, t : w = -2)
(y, x : w = -3)
(y, z : w = 9)
(z, x : w = 7)
(z, s : w = 2)
(s, t : w = 6)
(s, y : w = 7)



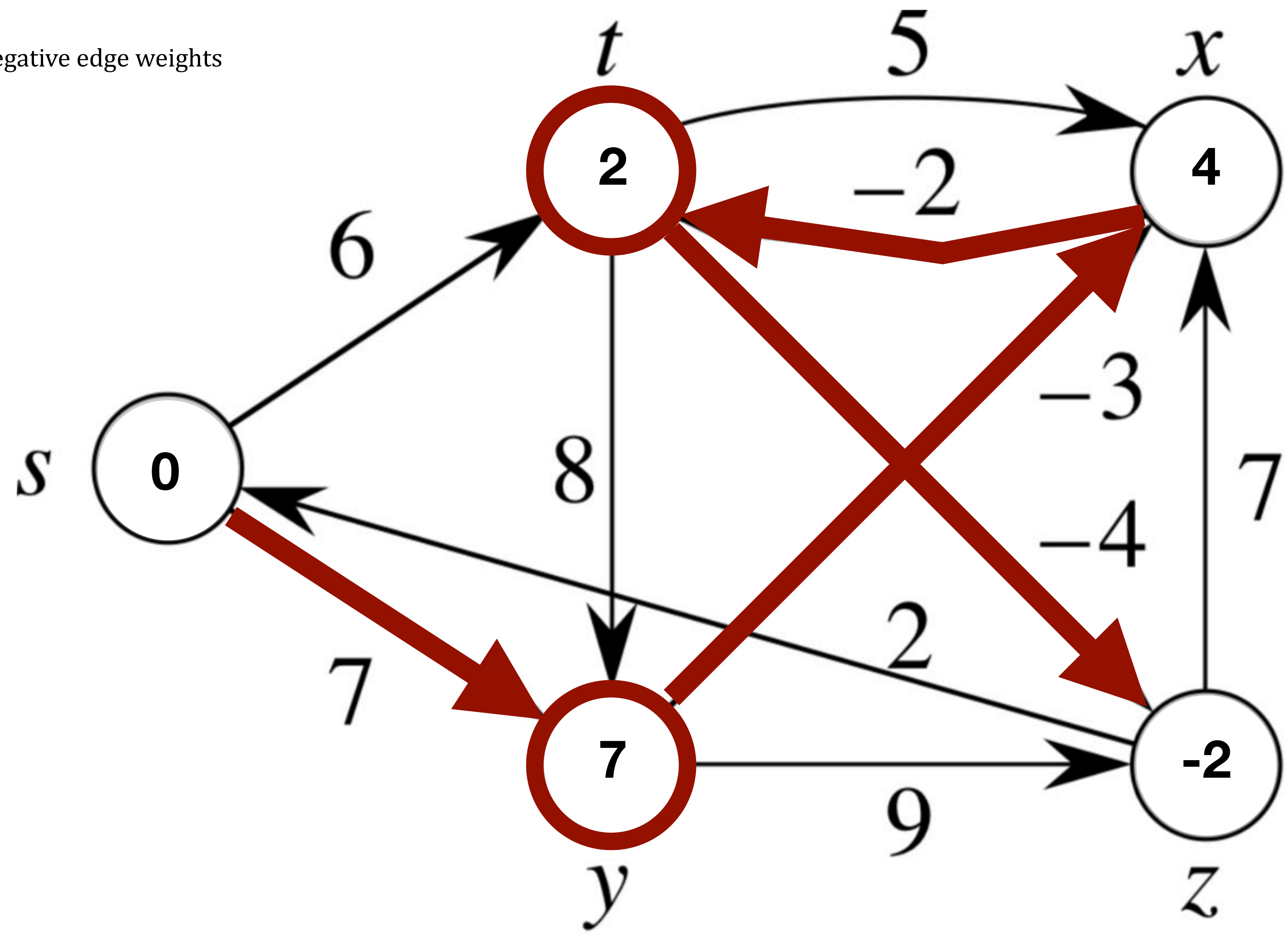

```
BELLMAN-FORD(G, w, s)
  INIT-SINGLE-SOURCE(G, s)
  for i = 4 to |G.V| − 1
      for each edge (u, v) ∈ G.E
          RELAX(u, v, w)
  for each edge (u, v) ∈ G.E
      if v.d > u.d + w(u, v)            no
          return FALSE
  return TRUE
```

The Bellman-Ford Single-Source Shortest Path Algorithm

Graph G

a weighted, directed graph with negative edge weights

G.V = s, t, x, y, z
G.E = (t, x : w = 5)
(t, y : w = 8)
(t, z : w = -4)
(x, t : w = -2)
(y, x : w = -3)
(y, z : w = 9)
(z, x : w = 7)
(z, s : w = 2)
(s, t : w = 6)
(s, y : w = 7)



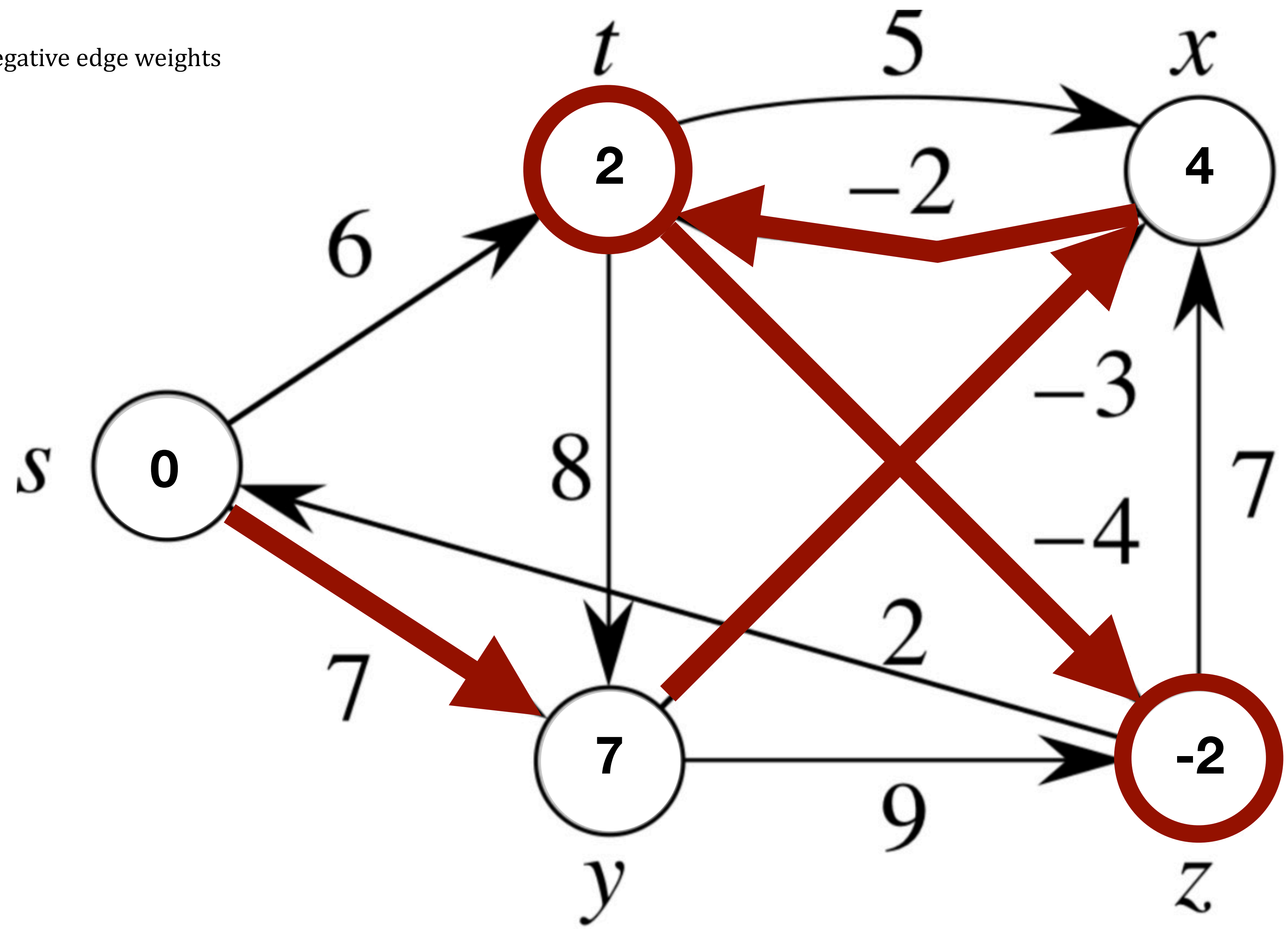

```
BELLMAN-FORD(G, w, s)
  INIT-SINGLE-SOURCE(G, s)
  for i = 4 to |G.V| − 1
      for each edge (u, v) ∈ G.E
          RELAX(u, v, w)
  for each edge (u, v) ∈ G.E
      if v.d > u.d + w(u, v)          no
          return FALSE
  return TRUE
```

The Bellman-Ford Single-Source Shortest Path Algorithm

Graph G

a weighted, directed graph with negative edge weights

G.V = s, t, x, y, z
G.E = (t, x : w = 5)
(t, y : w = 8)
(t, z : w = -4)
(x, t : w = -2)
(y, x : w = -3)
(y, z : w = 9)
(z, x : w = 7)
(z, s : w = 2)
(s, t : w = 6)
(s, y : w = 7)



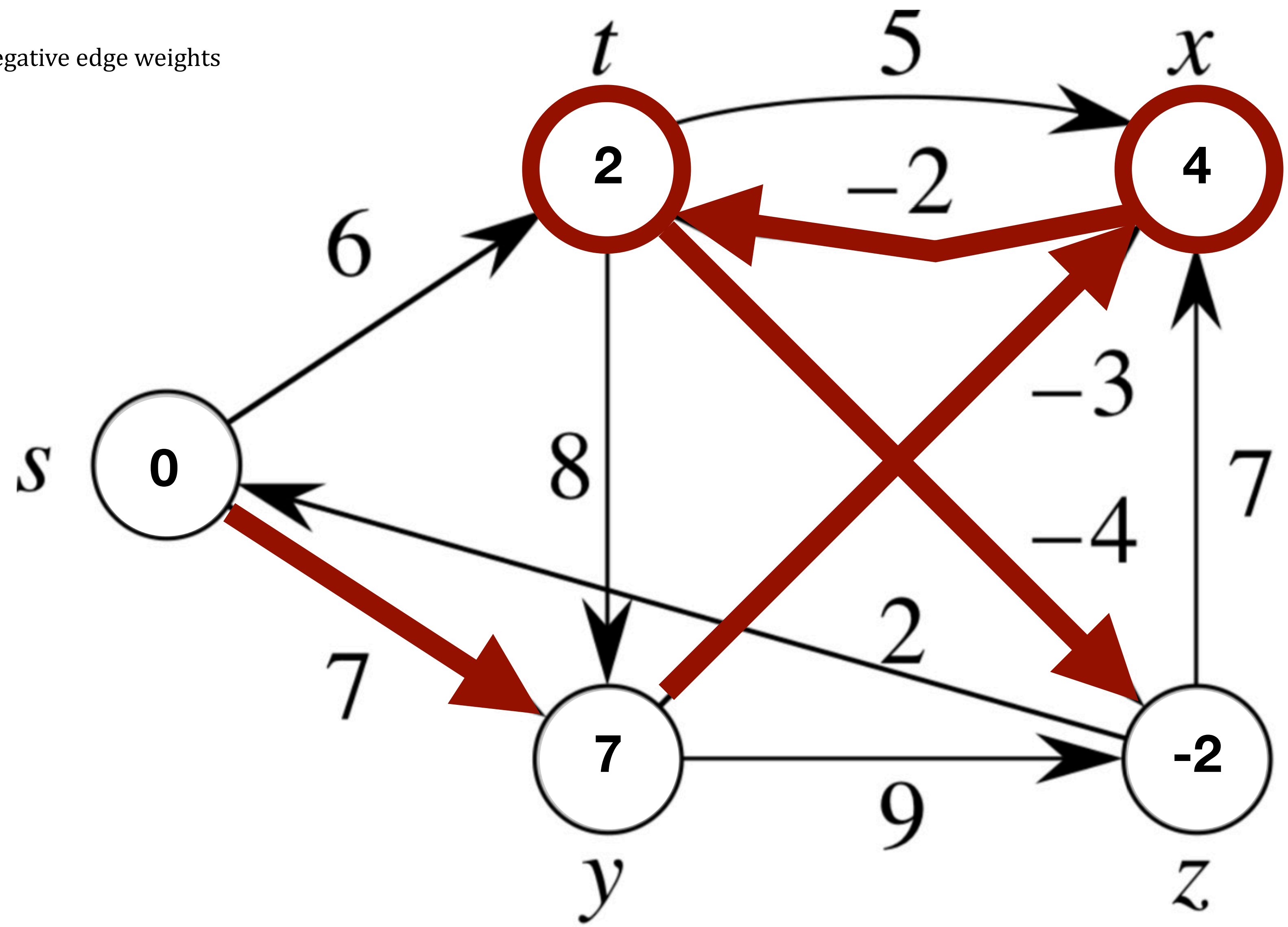

```
BELLMAN-FORD(G, w, s)
  INIT-SINGLE-SOURCE(G, s)
  for i = 4 to |G.V| − 1
      for each edge (u, v) ∈ G.E
          RELAX(u, v, w)
  for each edge (u, v) ∈ G.E
      if v.d > u.d + w(u, v)          no
          return FALSE
  return TRUE
```

The Bellman-Ford Single-Source Shortest Path Algorithm

Graph G

a weighted, directed graph with negative edge weights

G.V = s, t, x, y, z
G.E = (t, x : w = 5)
(t, y : w = 8)
(t, z : w = -4)
(x, t : w = -2)
(y, x : w = -3)
(y, z : w = 9)
(z, x : w = 7)
(z, s : w = 2)
(s, t : w = 6)
(s, y : w = 7)



BELLMAN-FORD(G, w, s)
INIT-SINGLE-SOURCE(G, s)
for i = **4** to $|G.V| - 1$
for each edge $(u, v) \in G.E$
RELAX(u, v, w)
for each edge $(u, v) \in G.E$
if $v.d > u.d + w(u, v)$ no
return FALSE
return TRUE

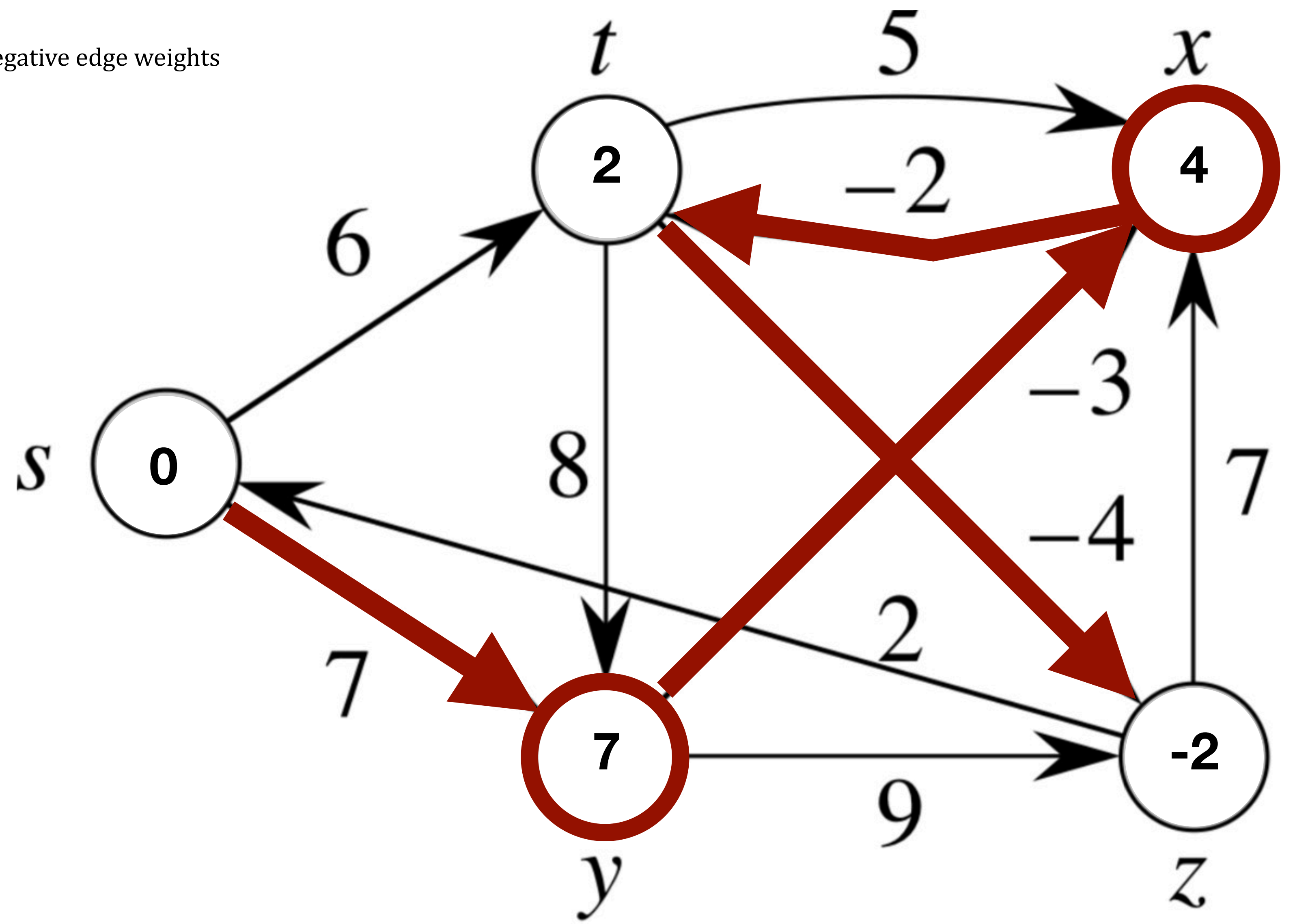
The Bellman-Ford Single-Source Shortest Path Algorithm

Graph G

a weighted, directed graph with negative edge weights

G.V = s, t, x, y, z

G.E = (t, x : w = 5)
(t, y : w = 8)
(t, z : w = -4)
(x, t : w = -2)
(y, x : w = -3)
(y, z : w = 9)
(z, x : w = 7)
(z, s : w = 2)
(s, t : w = 6)
(s, y : w = 7)



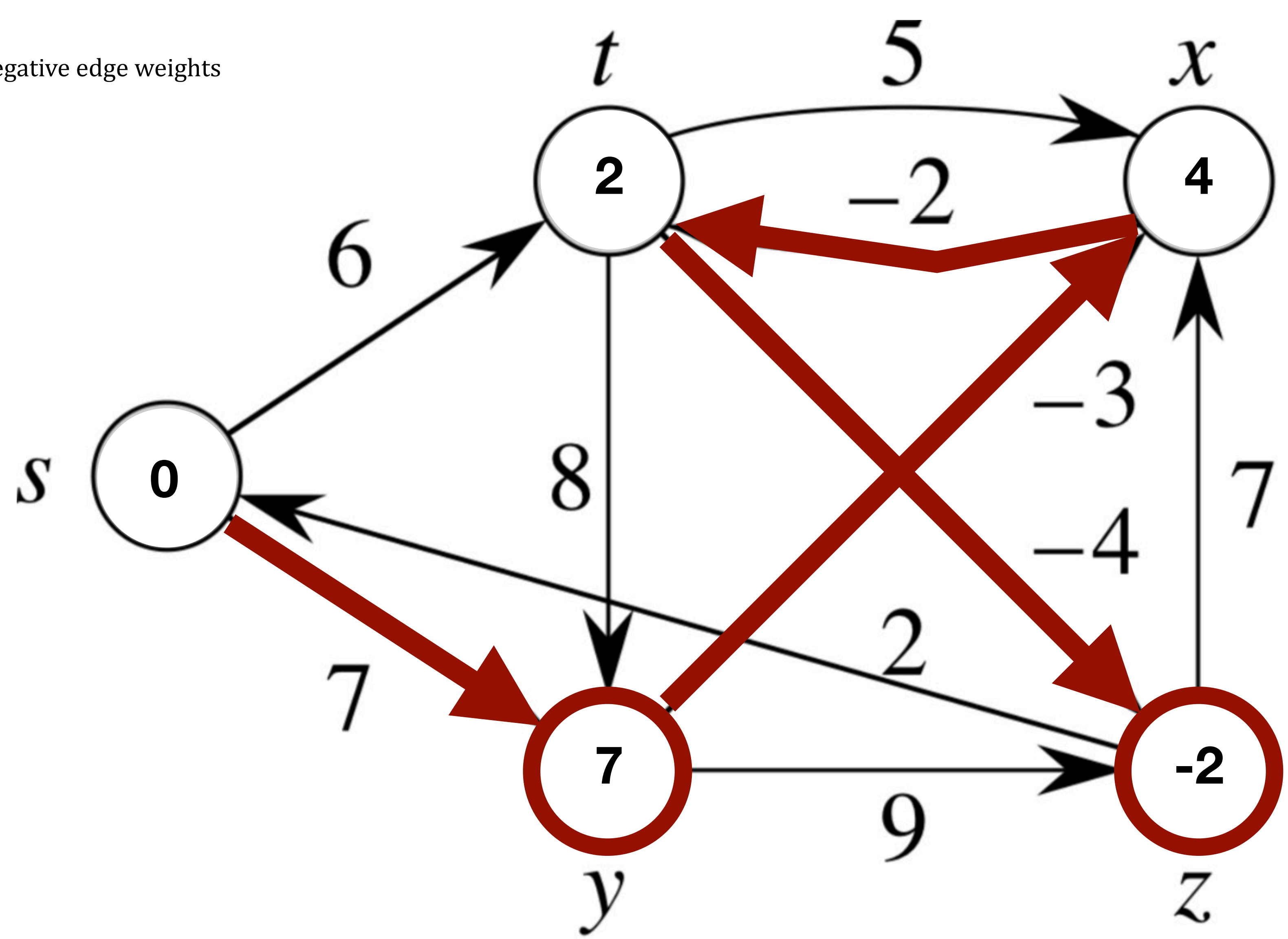

```
BELLMAN-FORD(G, w, s)
  INIT-SINGLE-SOURCE(G, s)
  for i = 4 to |G.V| − 1
      for each edge (u, v) ∈ G.E
          RELAX(u, v, w)
  for each edge (u, v) ∈ G.E
      if v.d > u.d + w(u, v)          no
          return FALSE
  return TRUE
```

The Bellman-Ford Single-Source Shortest Path Algorithm

Graph G

a weighted, directed graph with negative edge weights

G.V = s, t, x, y, z
G.E = (t, x : w = 5)
(t, y : w = 8)
(t, z : w = -4)
(x, t : w = -2)
(y, x : w = -3)
(y, z : w = 9)
(z, x : w = 7)
(z, s : w = 2)
(s, t : w = 6)
(s, y : w = 7)



BELLMAN-FORD(G, w, s)
INIT-SINGLE-SOURCE(G, s)
for i = **4** to $|G.V| - 1$
for each edge $(u, v) \in G.E$
RELAX(u, v, w)
for each edge $(u, v) \in G.E$
if $v.d > u.d + w(u, v)$ no
return FALSE
return TRUE

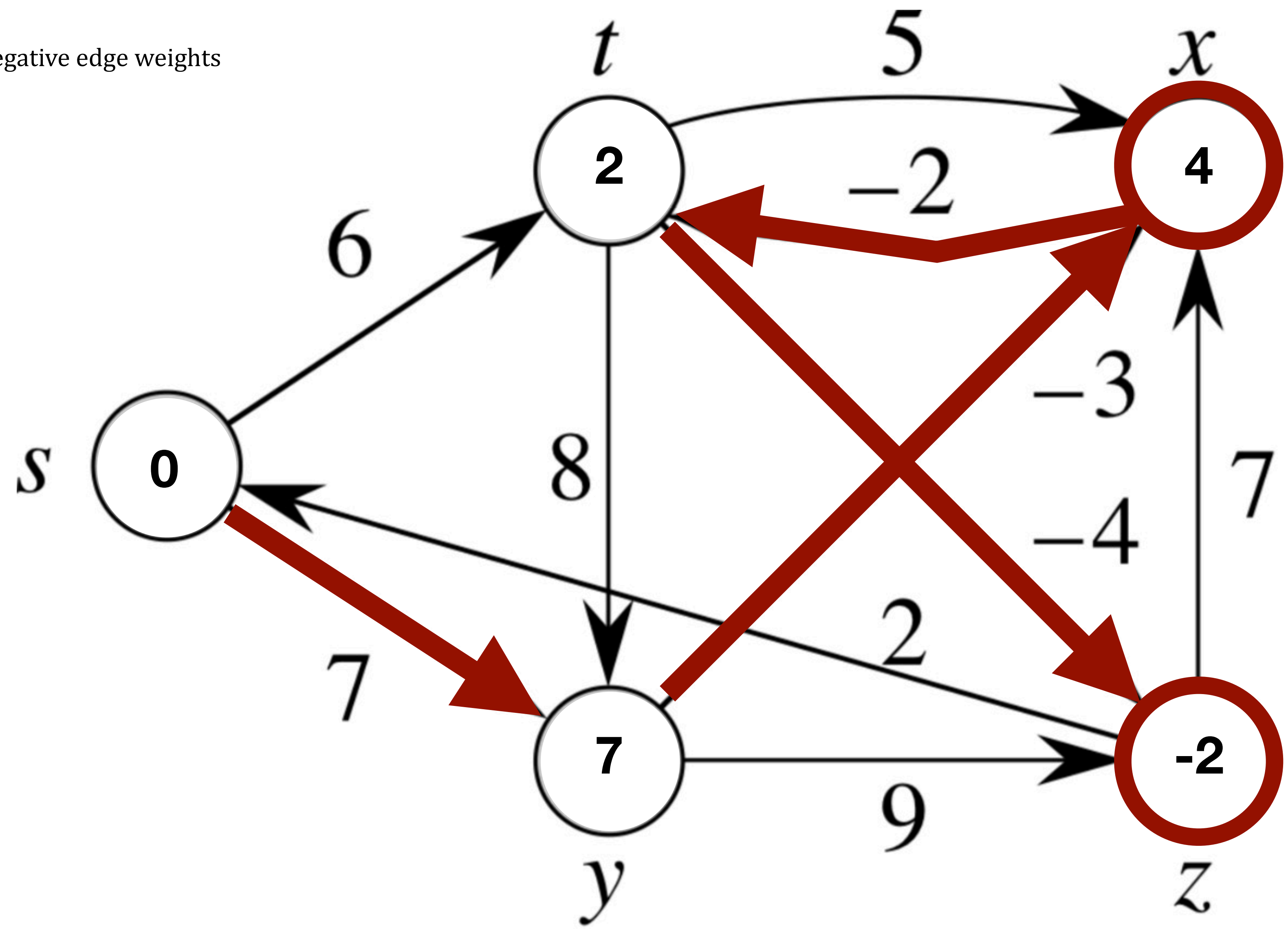
The Bellman-Ford Single-Source Shortest Path Algorithm

Graph G

a weighted, directed graph with negative edge weights

G.V = s, t, x, y, z

G.E = (t, x : w = 5)
(t, y : w = 8)
(t, z : w = -4)
(x, t : w = -2)
(y, x : w = -3)
(y, z : w = 9)
(z, x : w = 7)
(z, s : w = 2)
(s, t : w = 6)
(s, y : w = 7)



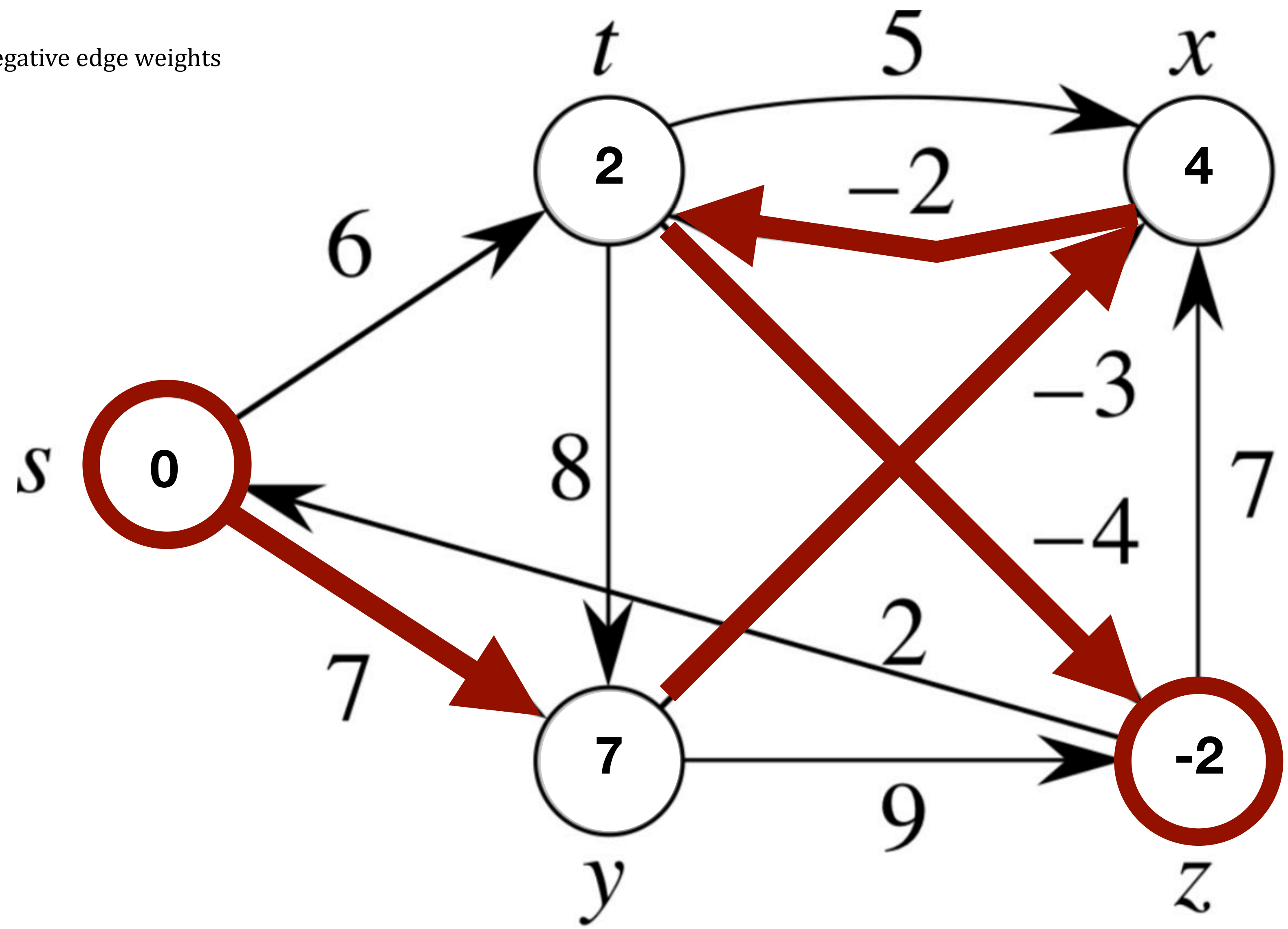
BELLMAN-FORD(G, w, s)
INIT-SINGLE-SOURCE(G, s)
for i = **4** to $|G.V| - 1$
for each edge $(u, v) \in G.E$
RELAX(u, v, w)
for each edge $(u, v) \in G.E$
if $v.d > u.d + w(u, v)$ no
return FALSE
return TRUE

The Bellman-Ford Single-Source Shortest Path Algorithm

Graph G

a weighted, directed graph with negative edge weights

G.V = s, t, x, y, z
G.E = (t, x : w = 5)
(t, y : w = 8)
(t, z : w = -4)
(x, t : w = -2)
(y, x : w = -3)
(y, z : w = 9)
(z, x : w = 7)
(z, s : w = 2)
(s, t : w = 6)
(s, y : w = 7)



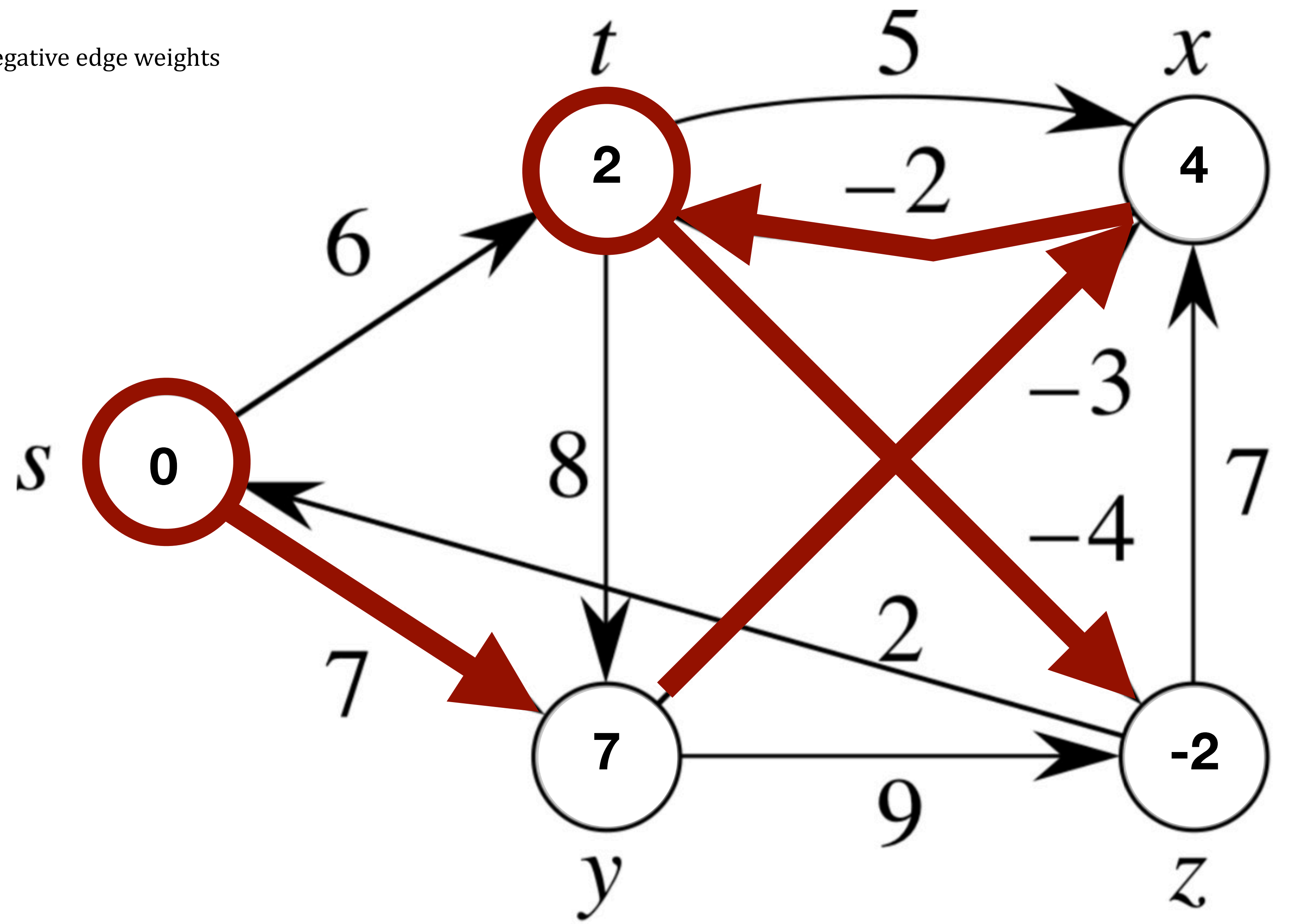

```
BELLMAN-FORD(G, w, s)
  INIT-SINGLE-SOURCE(G, s)
  for i = 4 to |G.V| − 1
      for each edge (u, v) ∈ G.E
          RELAX(u, v, w)
  for each edge (u, v) ∈ G.E
      if v.d > u.d + w(u, v)          no
          return FALSE
  return TRUE
```

The Bellman-Ford Single-Source Shortest Path Algorithm

Graph G

a weighted, directed graph with negative edge weights

G.V = s, t, x, y, z
G.E = (t, x : w = 5)
(t, y : w = 8)
(t, z : w = -4)
(x, t : w = -2)
(y, x : w = -3)
(y, z : w = 9)
(z, x : w = 7)
(z, s : w = 2)
(s, t : w = 6)
(s, y : w = 7)



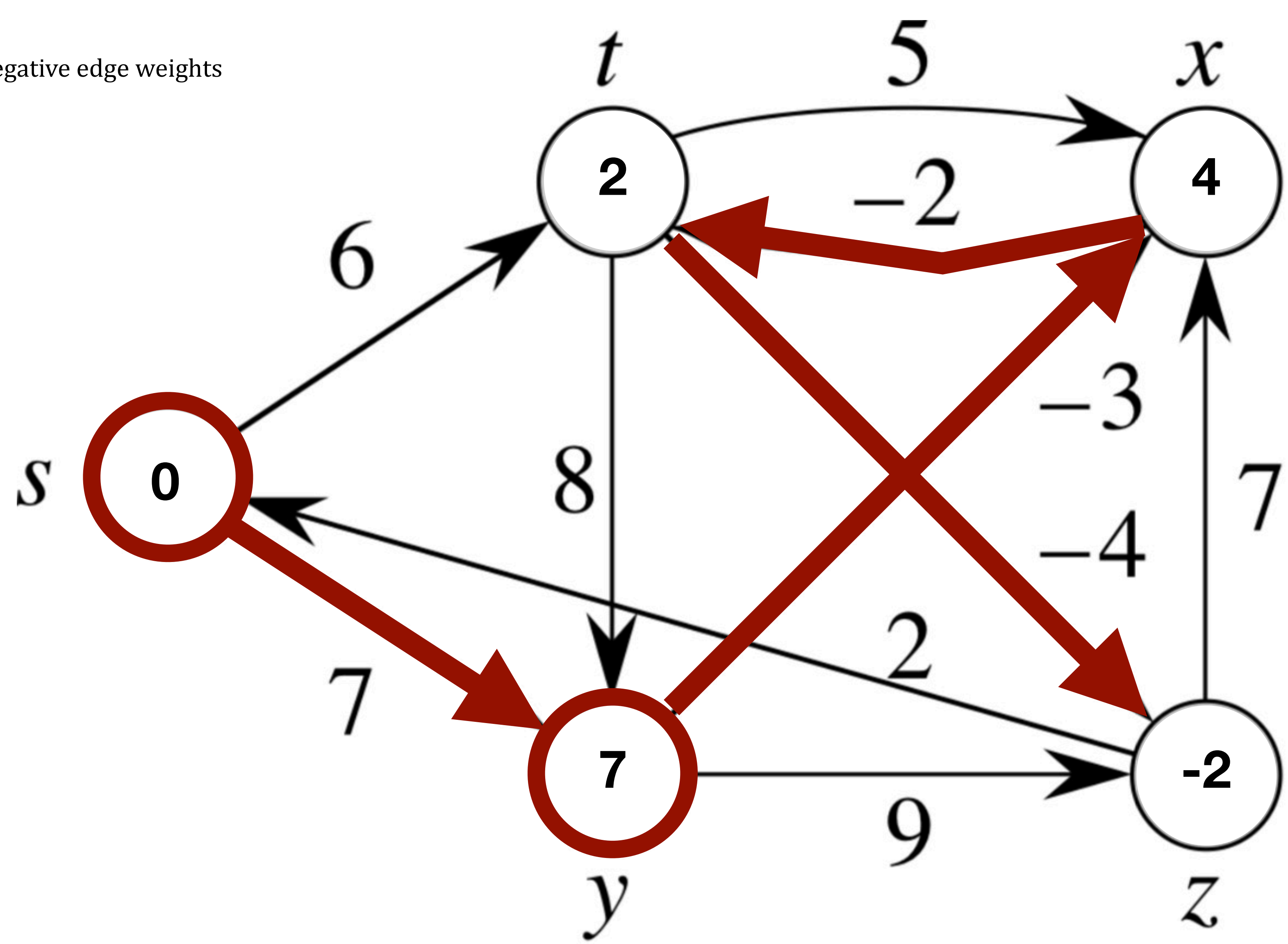
BELLMAN-FORD(G, w, s)
INIT-SINGLE-SOURCE(G, s)
for i = **4** to $|G.V| - 1$
for each edge $(u, v) \in G.E$
RELAX(u, v, w)
for each edge $(u, v) \in G.E$
if $v.d > u.d + w(u, v)$ no
return FALSE
return TRUE

The Bellman-Ford Single-Source Shortest Path Algorithm

Graph G

a weighted, directed graph with negative edge weights

G.V = s, t, x, y, z
G.E = (t, x : w = 5)
(t, y : w = 8)
(t, z : w = -4)
(x, t : w = -2)
(y, x : w = -3)
(y, z : w = 9)
(z, x : w = 7)
(z, s : w = 2)
(s, t : w = 6)
(s, y : w = 7)



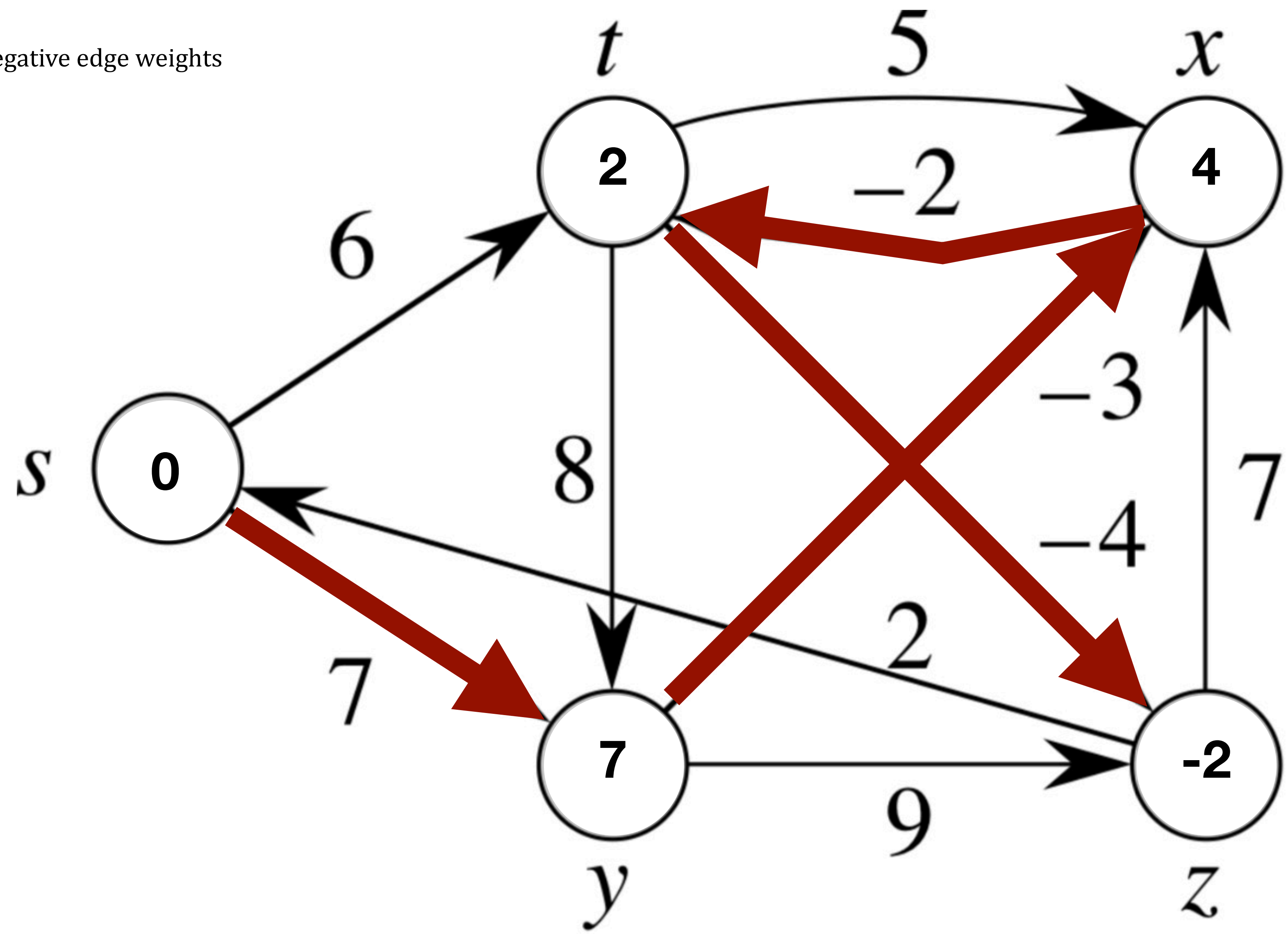
BELLMAN-FORD(G, w, s)
INIT-SINGLE-SOURCE(G, s)
for i = **4** to $|G.V| - 1$
for each edge $(u, v) \in G.E$
RELAX(u, v, w)
for each edge $(u, v) \in G.E$
if $v.d > u.d + w(u, v)$ no
return FALSE
return TRUE

The Bellman-Ford Single-Source Shortest Path Algorithm

Graph G

a weighted, directed graph with negative edge weights

G.V = s, t, x, y, z
G.E = (t, x : w = 5)
(t, y : w = 8)
(t, z : w = -4)
(x, t : w = -2)
(y, x : w = -3)
(y, z : w = 9)
(z, x : w = 7)
(z, s : w = 2)
(s, t : w = 6)
(s, y : w = 7)



BELLMAN-FORD(G, w, s)
  INIT-SINGLE-SOURCE(G, s)
  for i = **4** to $|G.V| - 1$
    for each edge $(u, v) \in G.E$
      RELAX(u, v, w)
  for each edge $(u, v) \in G.E$
    if $v.d > u.d + w(u, v)$
      return FALSE
  return TRUE

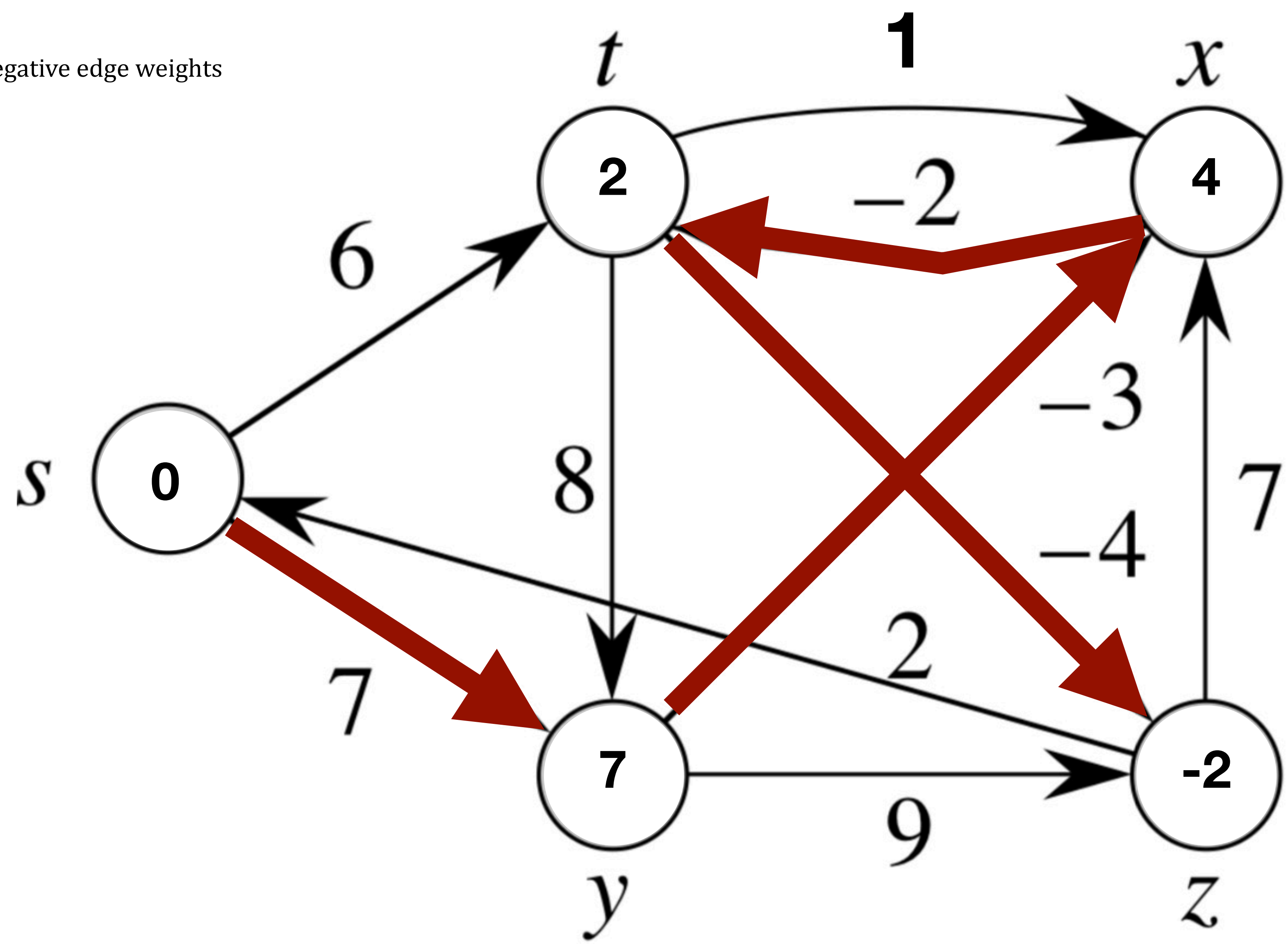
What does this do?

The Bellman-Ford Single-Source Shortest Path Algorithm

Graph G

a weighted, directed graph with negative edge weights

G.V = s, t, x, y, z
G.E = (t, x : w = 1)
(t, y : w = 8)
(t, z : w = -4)
(x, t : w = -2)
(y, x : w = -3)
(y, z : w = 9)
(z, x : w = 7)
(z, s : w = 2)
(s, t : w = 6)
(s, y : w = 7)




```
BELLMAN-FORD(G, w, s)
  INIT-SINGLE-SOURCE(G, s)
  for i = 4 to |G.V| − 1
      for each edge (u, v) ∈ G.E
          RELAX(u, v, w)
  for each edge (u, v) ∈ G.E
      if v.d > u.d + w(u, v)
          return FALSE
  return TRUE
```

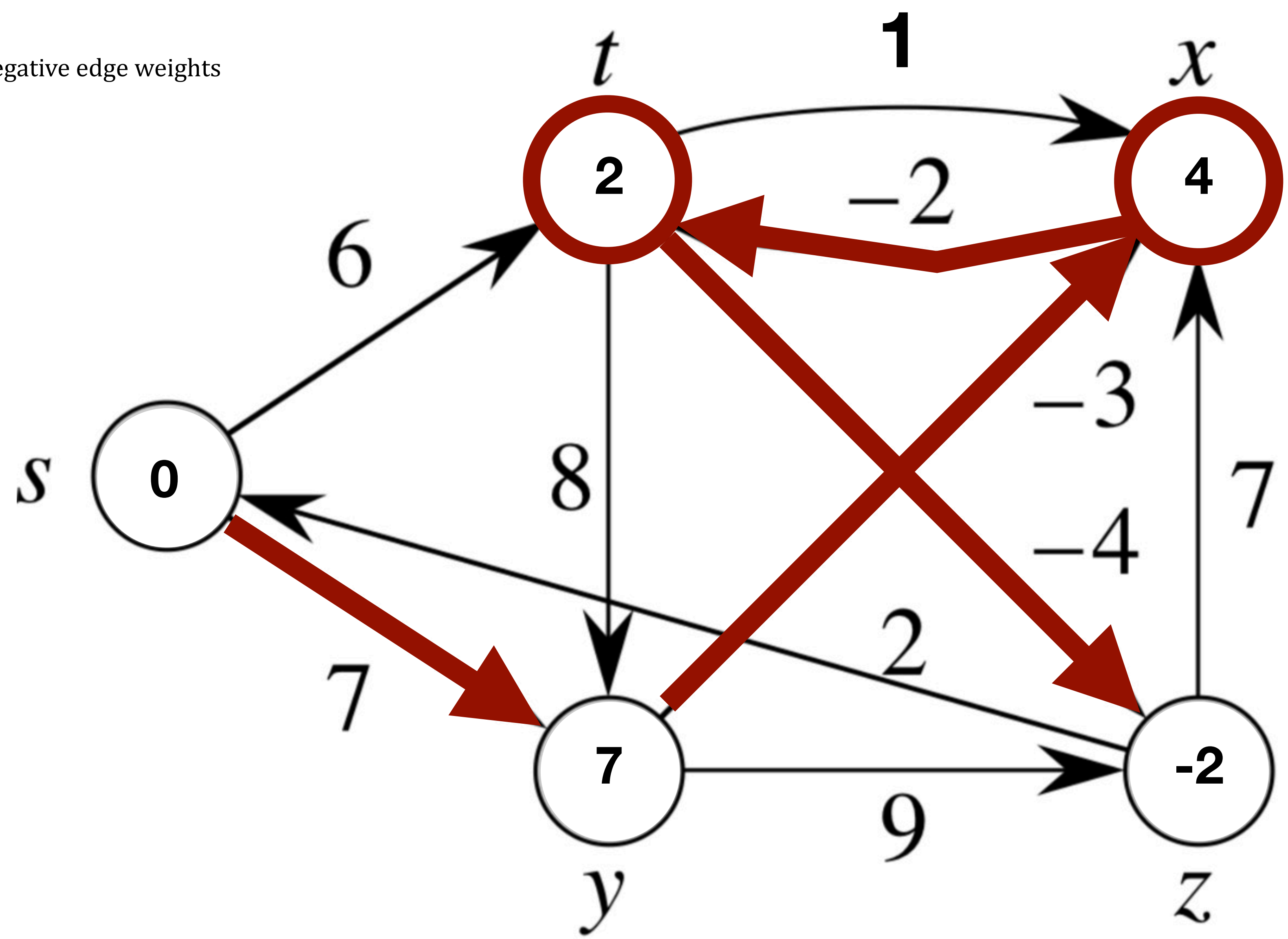
What does this do?
What if w(t,x)=1 ?

The Bellman-Ford Single-Source Shortest Path Algorithm

Graph G

a weighted, directed graph with negative edge weights

G.V = s, t, x, y, z
G.E = (t, x : w = 1)
(t, y : w = 8)
(t, z : w = -4)
(x, t : w = -2)
(y, x : w = -3)
(y, z : w = 9)
(z, x : w = 7)
(z, s : w = 2)
(s, t : w = 6)
(s, y : w = 7)




```
BELLMAN-FORD(G, w, s)
  INIT-SINGLE-SOURCE(G, s)
  for i  = 4  to |G.V| - 1
      for each edge (u, v) ∈ G.E
          RELAX(u, v, w)
  for each edge (u, v) ∈ G.E
      if v.d > u.d + w(u, v)          yes
          return FALSE
  return TRUE
```

What does this do?
What if w(t,x)=1 ?

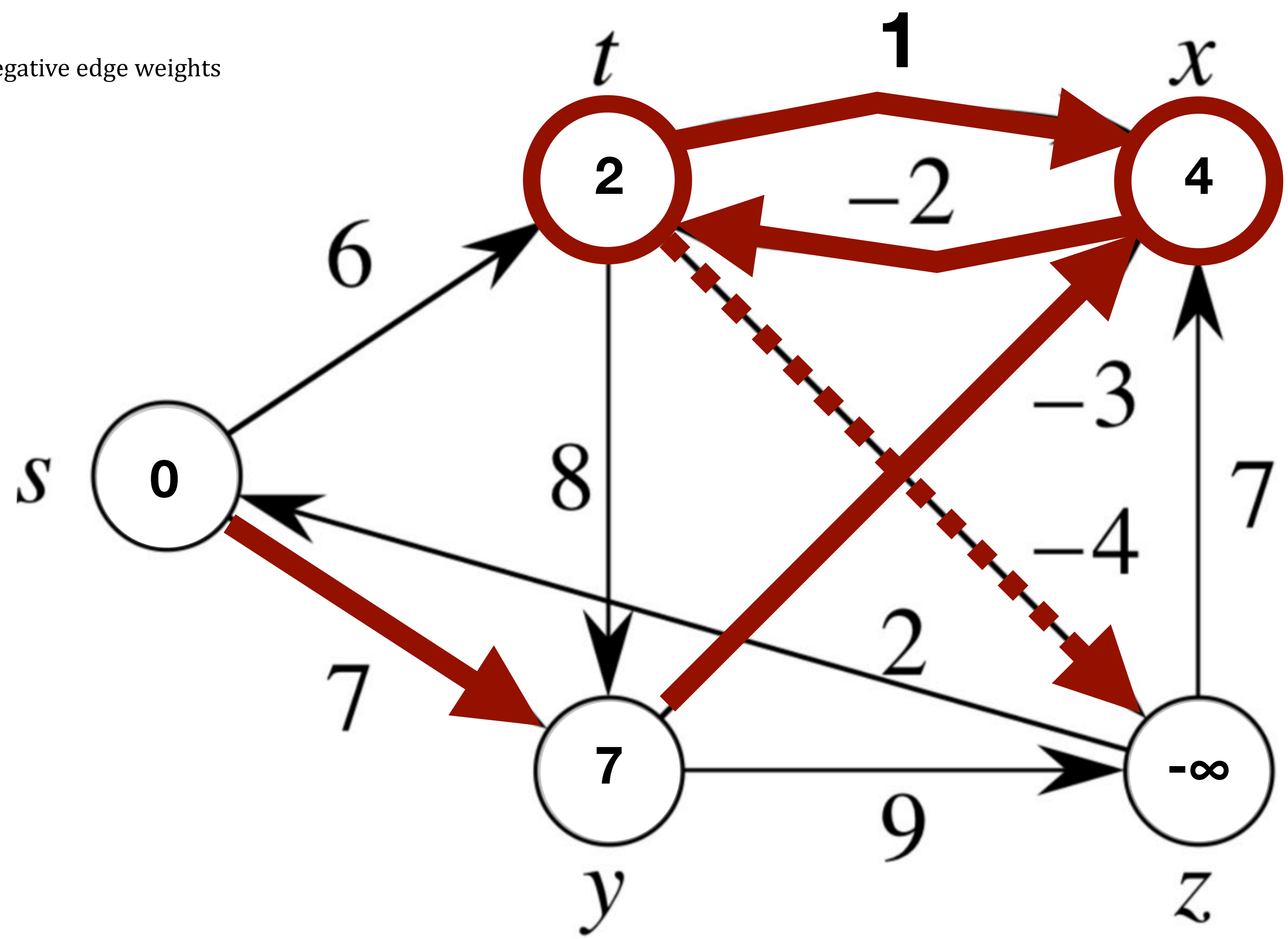
The Bellman-Ford Single-Source Shortest Path Algorithm

Graph G

a weighted, directed graph with negative edge weights

G.V = s, t, x, y, z

G.E = (t, x : w = 1)
(t, y : w = 8)
(t, z : w = -4)
(x, t : w = -2)
(y, x : w = -3)
(y, z : w = 9)
(z, x : w = 7)
(z, s : w = 2)
(s, t : w = 6)
(s, y : w = 7)



BELLMAN-FORD(G, w, s)
  INIT-SINGLE-SOURCE(G, s)
  for i = **4** to $|G.V| - 1$
    for each edge $(u, v) \in G.E$
      RELAX(u, v, w)
  for each edge $(u, v) \in G.E$
    if $v.d > u.d + w(u, v)$ yes
      return FALSE
  return TRUE

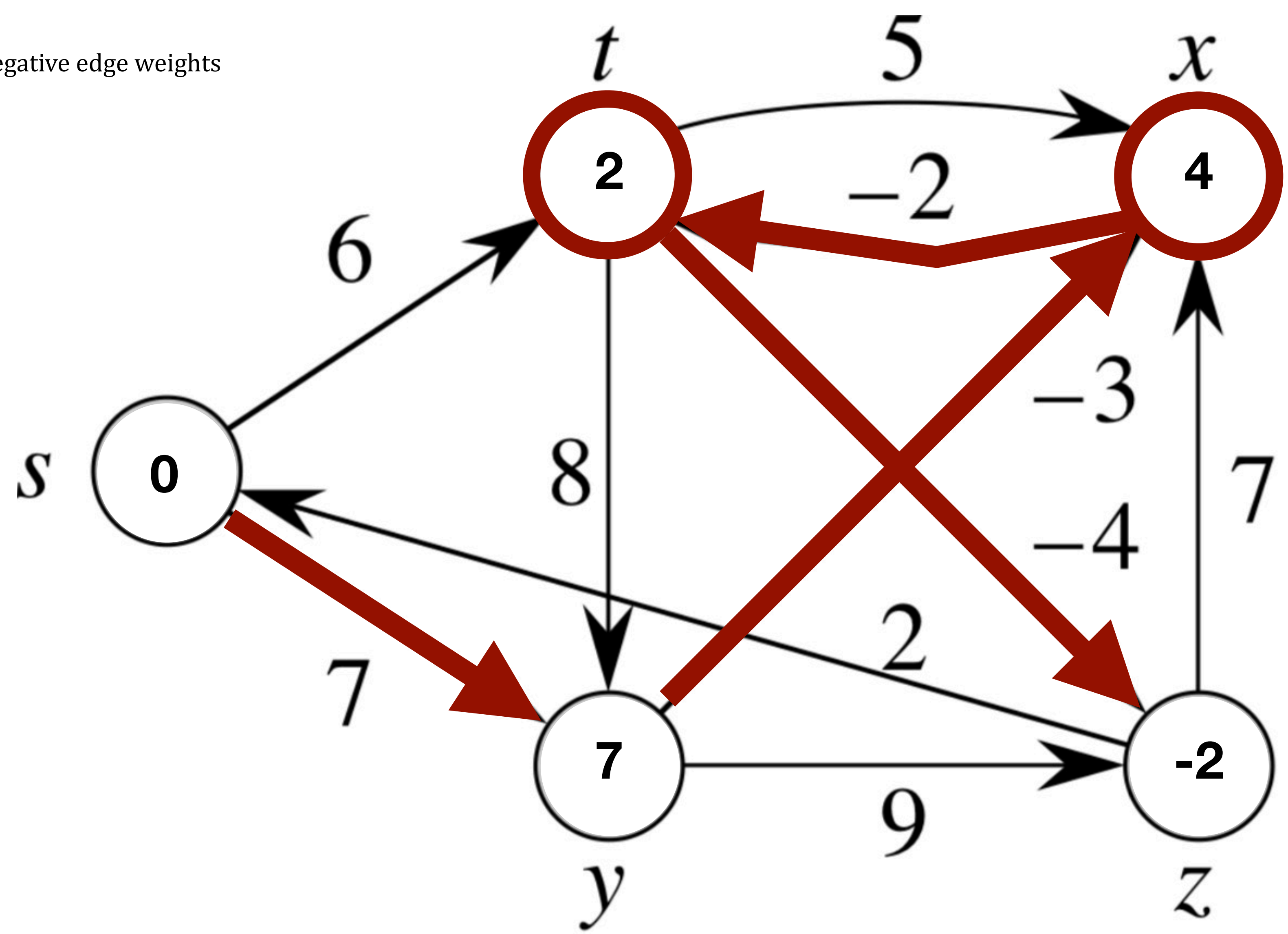
What does this do?
What if w(t,x)=1 ?
We'd have a negative weight cycle. That's… interesting. Now w(s,z) = -∞ .

The Bellman-Ford Single-Source Shortest Path Algorithm

Graph G

a weighted, directed graph with negative edge weights

G.V = s, t, x, y, z
G.E = (t, x : *w* = 5)
(t, y : *w* = 8)
(t, z : *w* = -4)
(x, t : *w* = -2)
(y, x : *w* = -3)
(y, z : *w* = 9)
(z, x : *w* = 7)
(z, s : *w* = 2)
(s, t : *w* = 6)
(s, y : *w* = 7)




```
BELLMAN-FORD(G, w, s)
  INIT-SINGLE-SOURCE(G, s)
  for i = 4 to |G.V| − 1
      for each edge (u, v) ∈ G.E
          RELAX(u, v, w)
  for each edge (u, v) ∈ G.E
      if v.d > u.d + w(u, v)
          return FALSE
  return TRUE
```

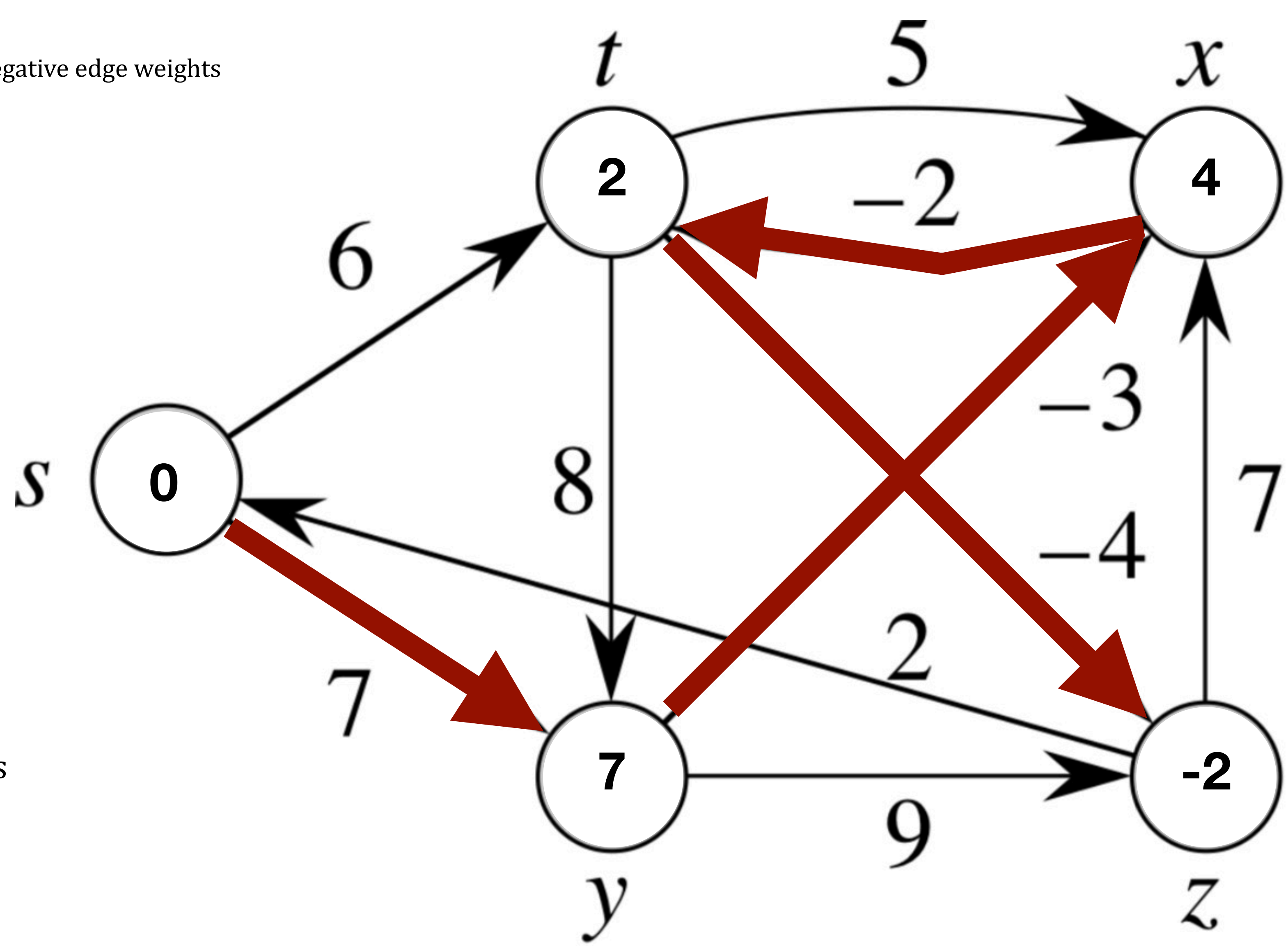
What does this do?
It detects negative-weight cycles . . .
and returns **false** is any are found.

The Bellman-Ford Single-Source Shortest Path Algorithm

Graph G

a weighted, directed graph with negative edge weights

G.V = s, t, x, y, z
G.E = (t, x : *w* = 5)
(t, y : *w* = 8)
(t, z : *w* = -4)
(x, t : *w* = -2)
(y, x : *w* = -3)
(y, z : *w* = 9)
(z, x : *w* = 7)
(z, s : *w* = 2)
(s, t : *w* = 6)
(s, y : *w* = 7)



Now we're done.

The value in each vertex is the length of the shortest path to that vertex from starting vertex **s**.

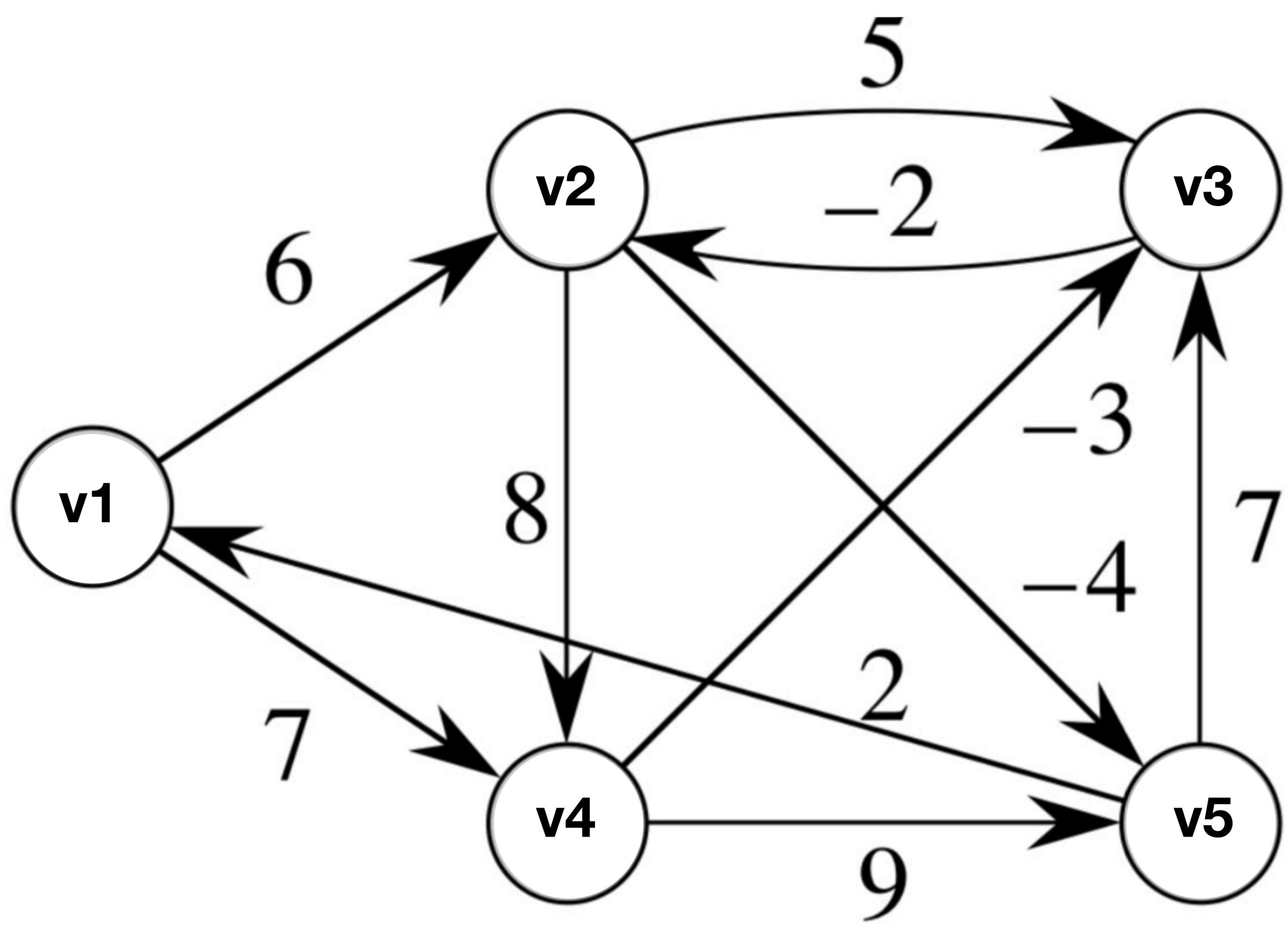
s → s = 0 shortest path: s
s → t = 2 shortest path: s → y → x → t
s → x = 4 shortest path: s → y → x
s → y = 7 shortest path: s → y
s → z = -2 shortest path: s → y → x → t → z

The Bellman-Ford Single-Source Shortest Path Algorithm

Graph G

Assignment 5 version
with integer vertex IDs

```
new graph
add vertex 1
add vertex 2
add vertex 3
add vertex 4
add vertex 5
add edge 2 - 3  5
add edge 2 - 4  8
add edge 2 - 5 -4
add edge 3 - 2 -2
add edge 4 - 3 -3
add edge 4 - 5  9
add edge 5 - 3  7
add edge 5 - 1  2
add edge 1 - 2  6
add edge 1 - 4  7
```



The Bellman-Ford Single-Source Shortest Path Algorithm