



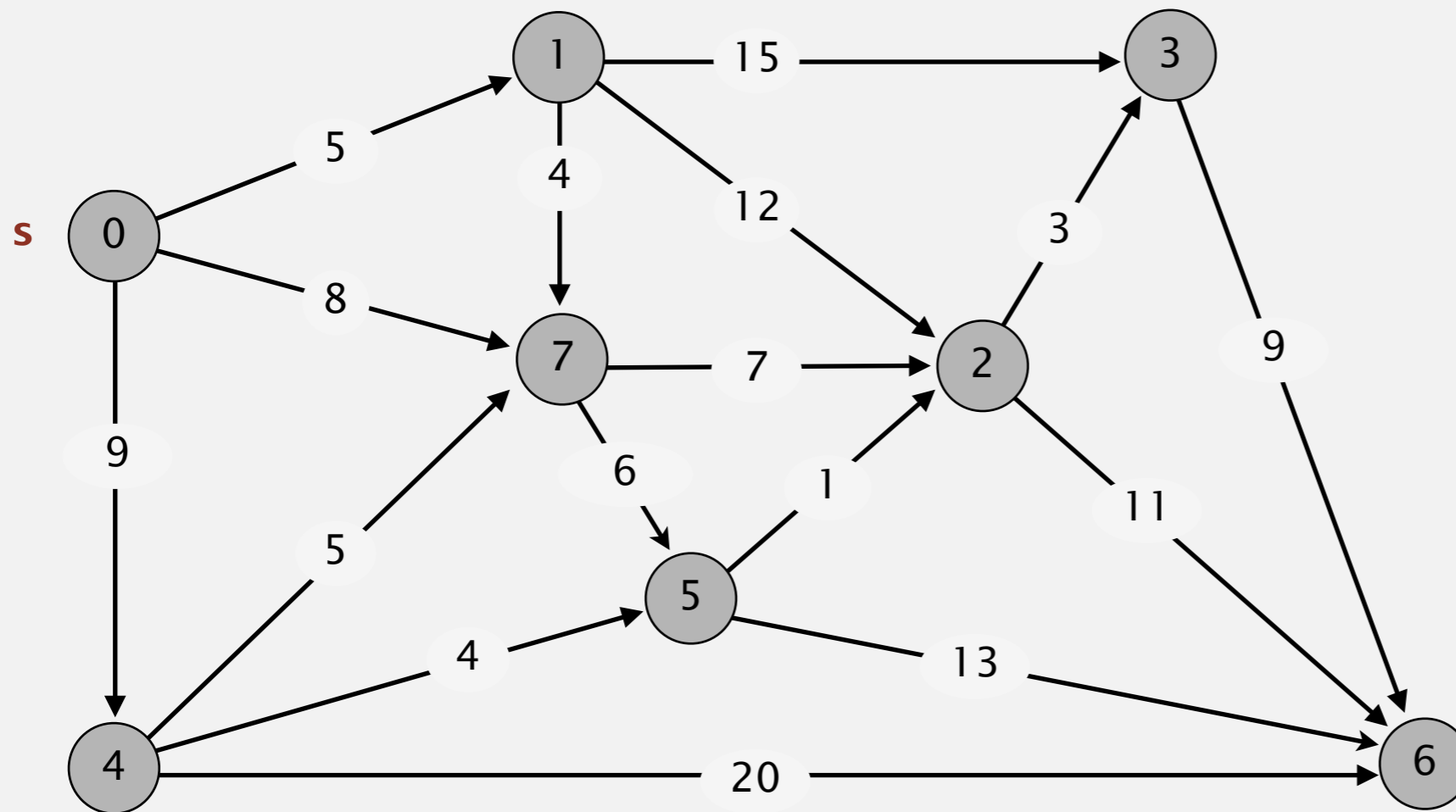
<http://algs4.cs.princeton.edu>

## DIJKSTRA'S ALGORITHM DEMO

---

# Dijkstra's algorithm demo

- Consider vertices in increasing order of distance from  $s$  (non-tree vertex with the lowest  $\text{distTo}[]$  value).
- Add vertex to tree and relax all edges adjacent from that vertex.

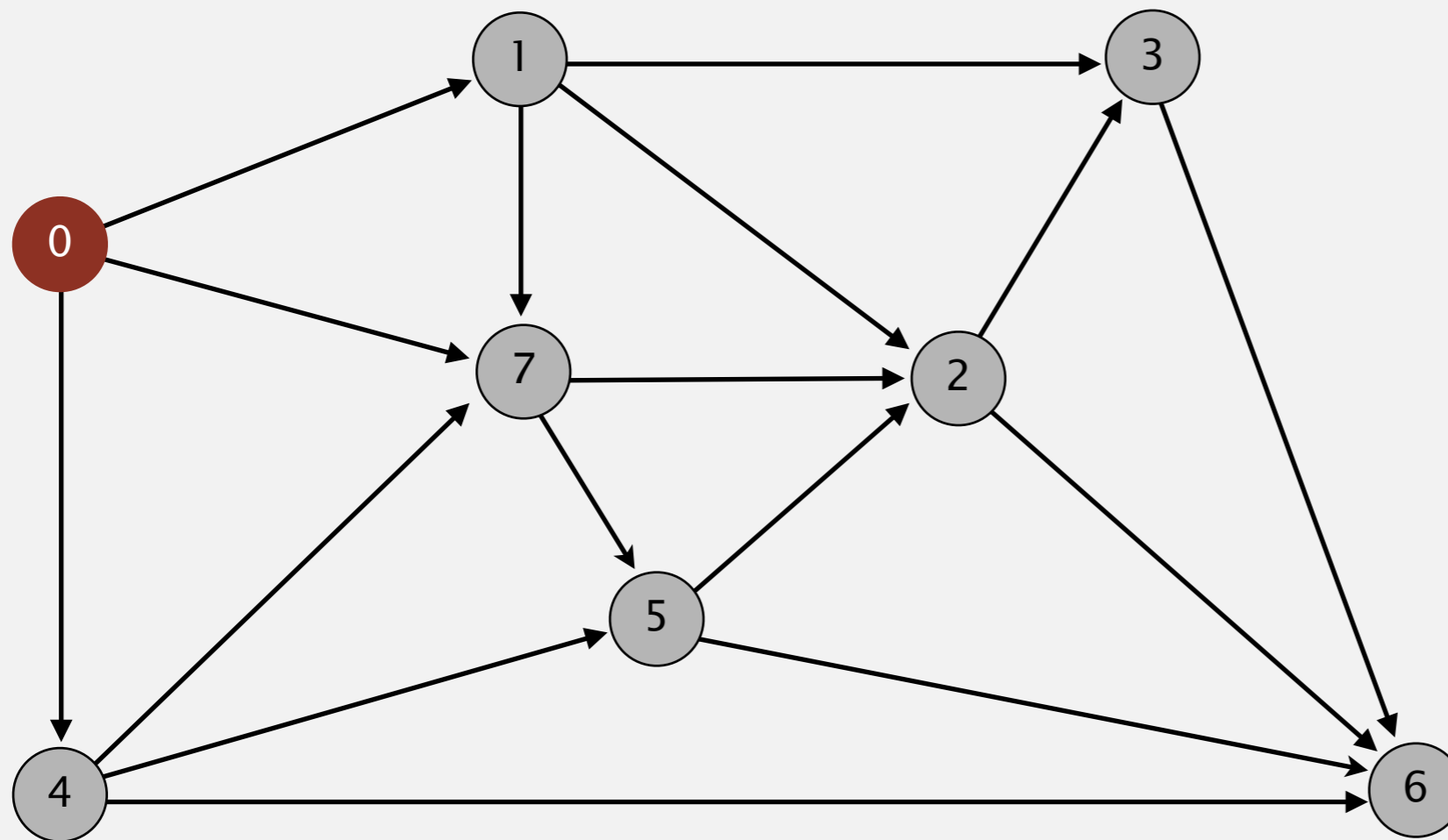


0→1	5.0
0→4	9.0
0→7	8.0
1→2	12.0
1→3	15.0
1→7	4.0
2→3	3.0
2→6	11.0
3→6	9.0
4→5	4.0
4→6	20.0
4→7	5.0
5→2	1.0
5→6	13.0
7→5	6.0
7→2	7.0

**an edge-weighted digraph**

# Dijkstra's algorithm demo

- Consider vertices in increasing order of distance from  $s$  (non-tree vertex with the lowest  $\text{distTo}[]$  value).
- Add vertex to tree and relax all edges adjacent from that vertex.

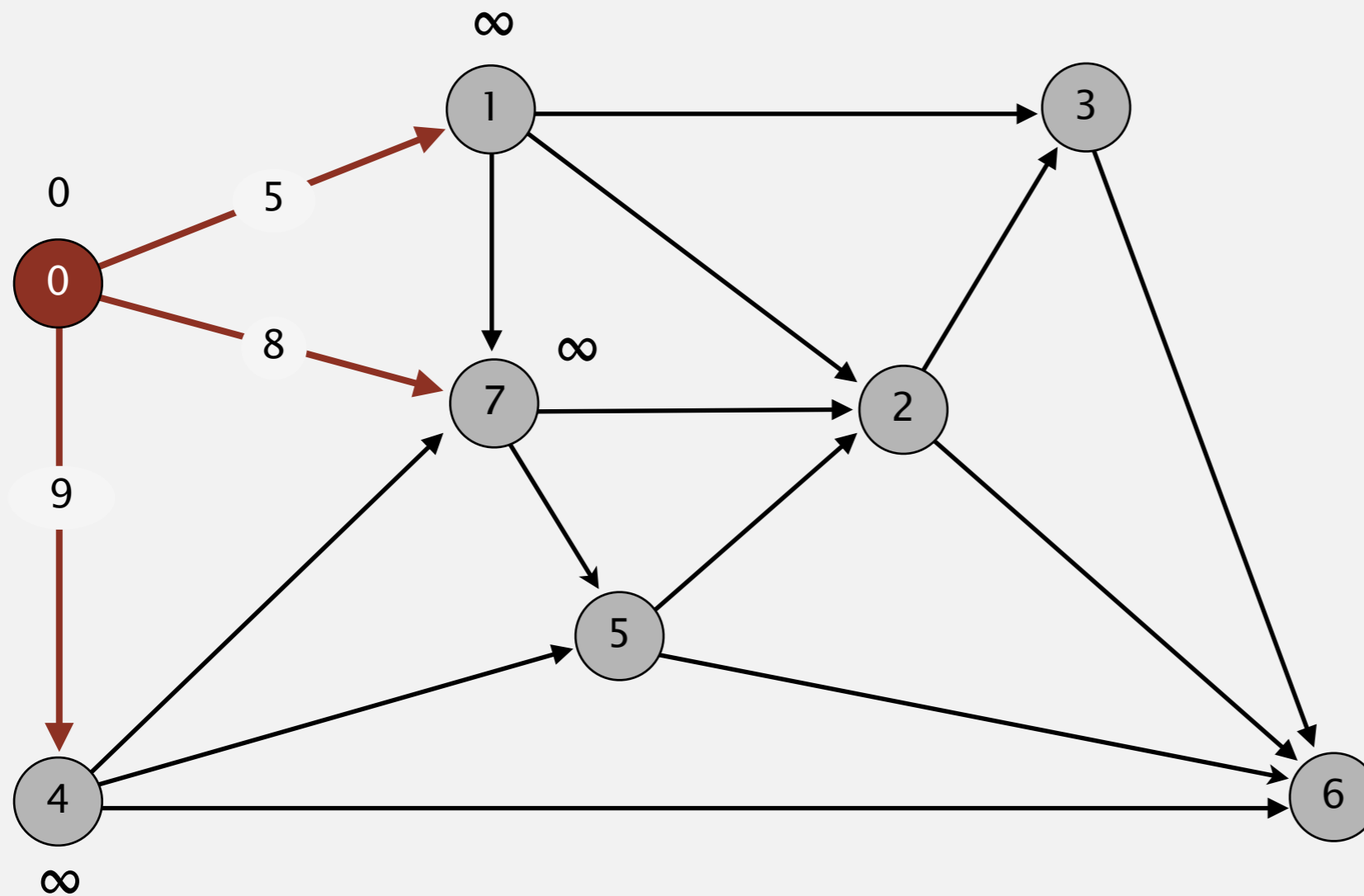


$v$	$\text{distTo}[]$	$\text{edgeTo}[]$
→ 0	0.0	-
1		
2		
3		
4		
5		
6		
7		

choose source vertex 0

# Dijkstra's algorithm demo

- Consider vertices in increasing order of distance from  $s$  (non-tree vertex with the lowest  $\text{distTo}[]$  value).
- Add vertex to tree and relax all edges adjacent from that vertex.

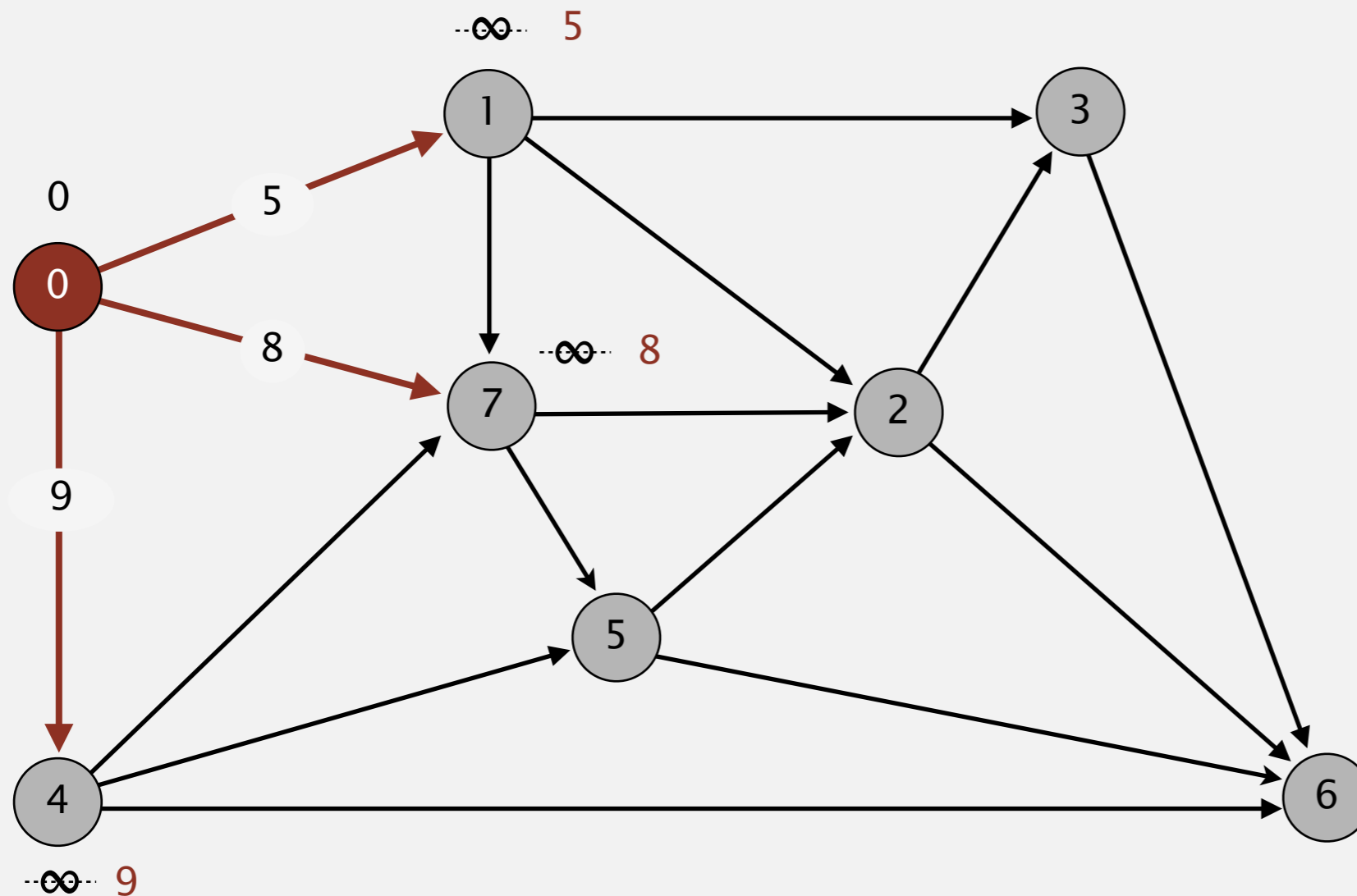


$v$	$\text{distTo}[]$	$\text{edgeTo}[]$
→ 0	0.0	-
1		
2		
3		
4		
5		
6		
7		

relax all edges adjacent from 0

# Dijkstra's algorithm demo

- Consider vertices in increasing order of distance from  $s$  (non-tree vertex with the lowest  $\text{distTo}[]$  value).
- Add vertex to tree and relax all edges adjacent from that vertex.



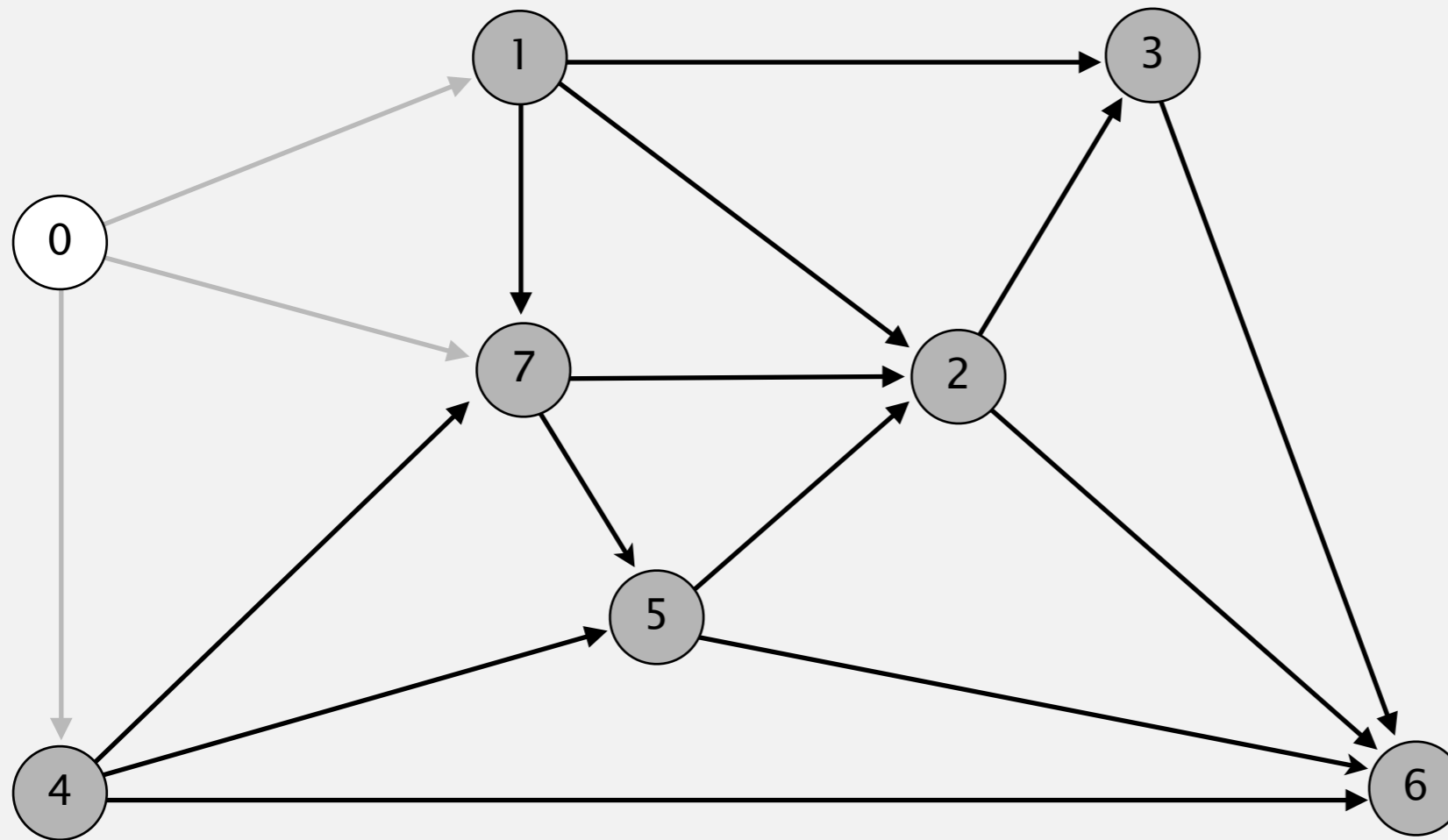
$v$	$\text{distTo}[]$	$\text{edgeTo}[]$
→ 0	0.0	-
1	5.0	0→1
2	$\infty$	
3	$\infty$	
4	9.0	0→4
5	$\infty$	
6	$\infty$	
7	8.0	0→7

relax all edges adjacent from 0

# Dijkstra's algorithm demo

---

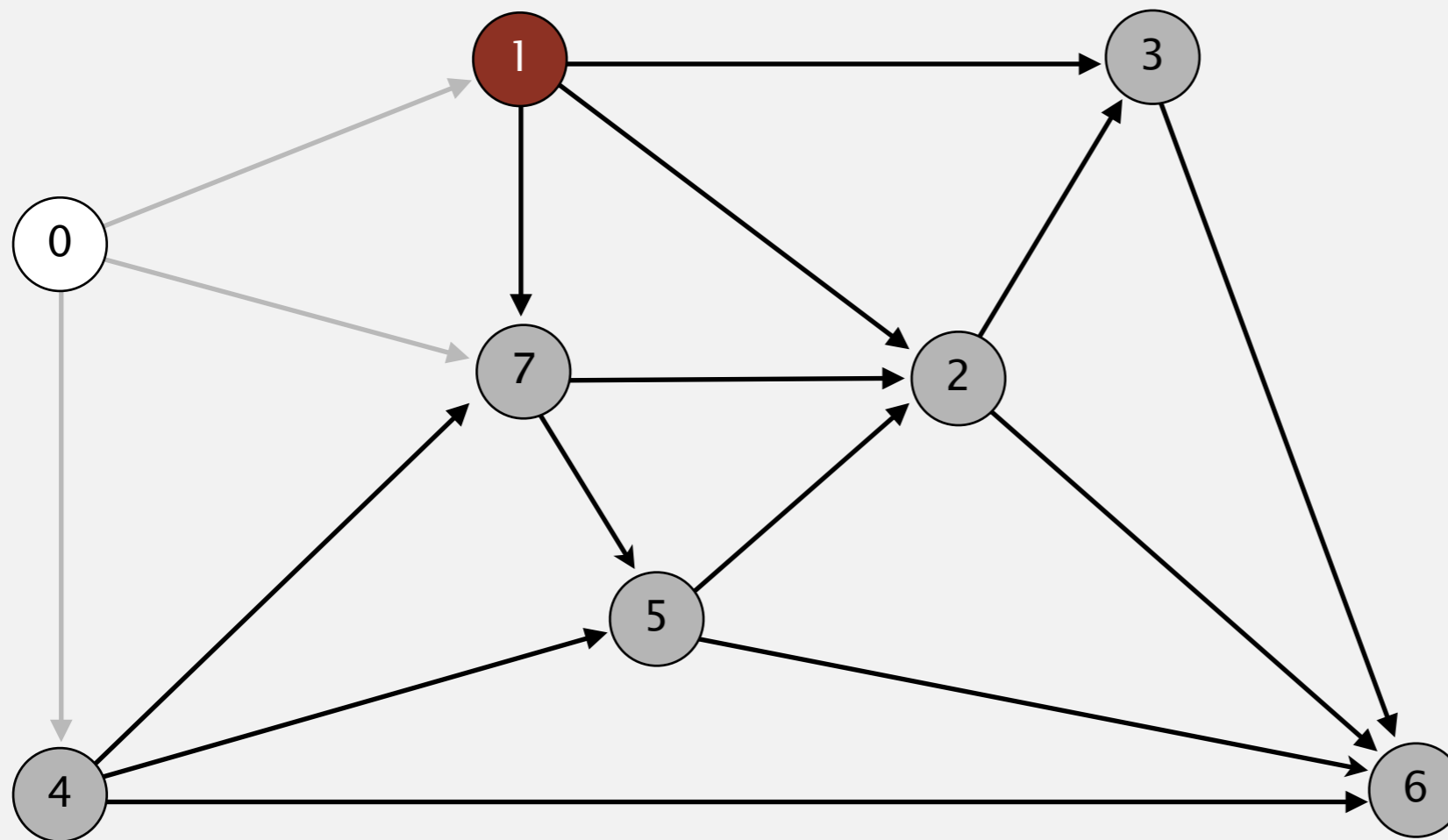
- Consider vertices in increasing order of distance from  $s$  (non-tree vertex with the lowest  $\text{distTo}[]$  value).
- Add vertex to tree and relax all edges adjacent from that vertex.



$v$	$\text{distTo}[]$	$\text{edgeTo}[]$
0	0.0	-
1	5.0	0→1
2		
3		
4	9.0	0→4
5		
6		
7	8.0	0→7

# Dijkstra's algorithm demo

- Consider vertices in increasing order of distance from  $s$  (non-tree vertex with the lowest  $\text{distTo}[]$  value).
- Add vertex to tree and relax all edges adjacent from that vertex.

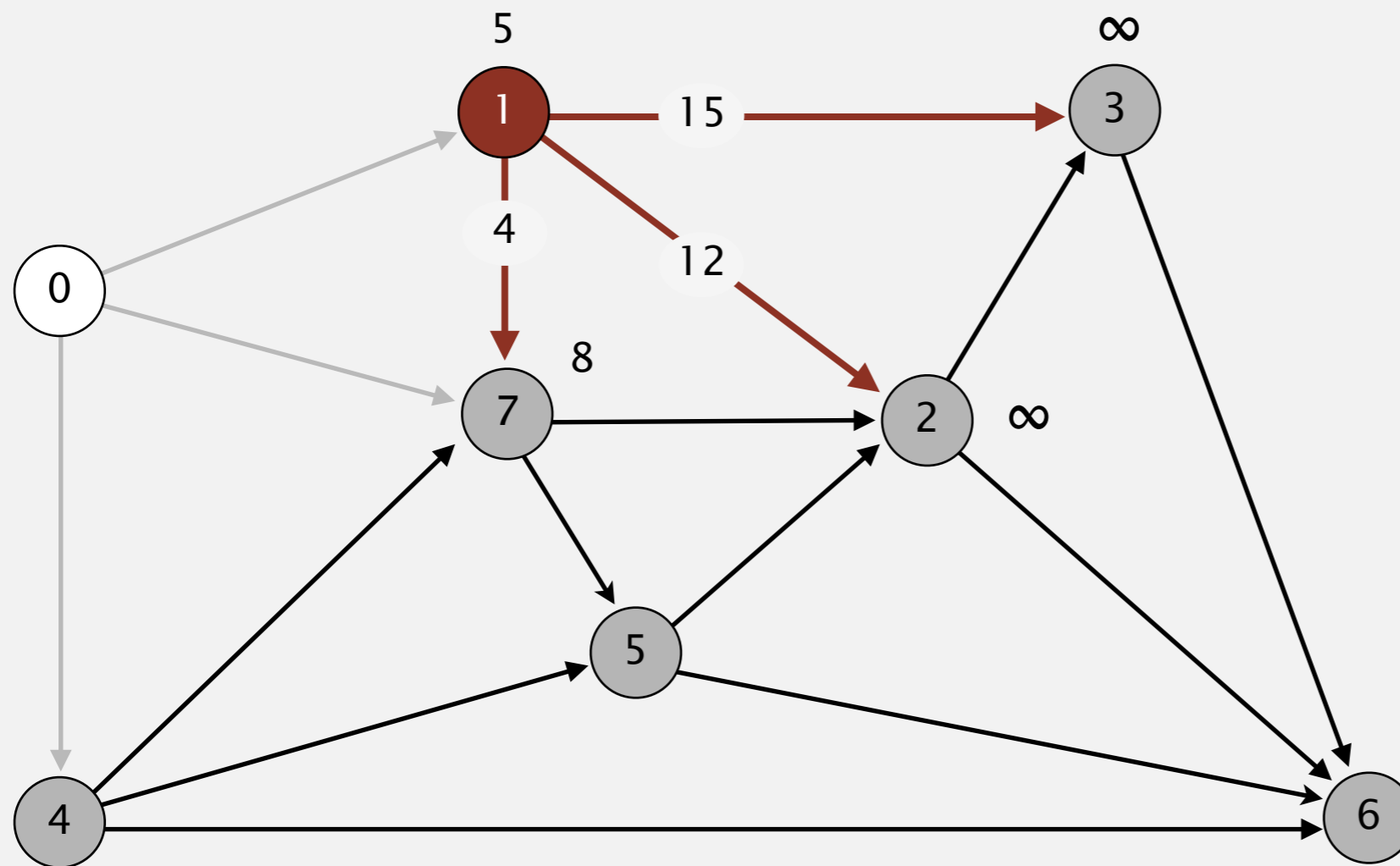


$v$	$\text{distTo}[]$	$\text{edgeTo}[]$
0	0.0	-
→ 1	5.0	0→1
2		
3		
4	9.0	0→4
5		
6		
7	8.0	0→7

choose vertex 1

# Dijkstra's algorithm demo

- Consider vertices in increasing order of distance from  $s$  (non-tree vertex with the lowest  $\text{distTo}[]$  value).
- Add vertex to tree and relax all edges adjacent from that vertex.



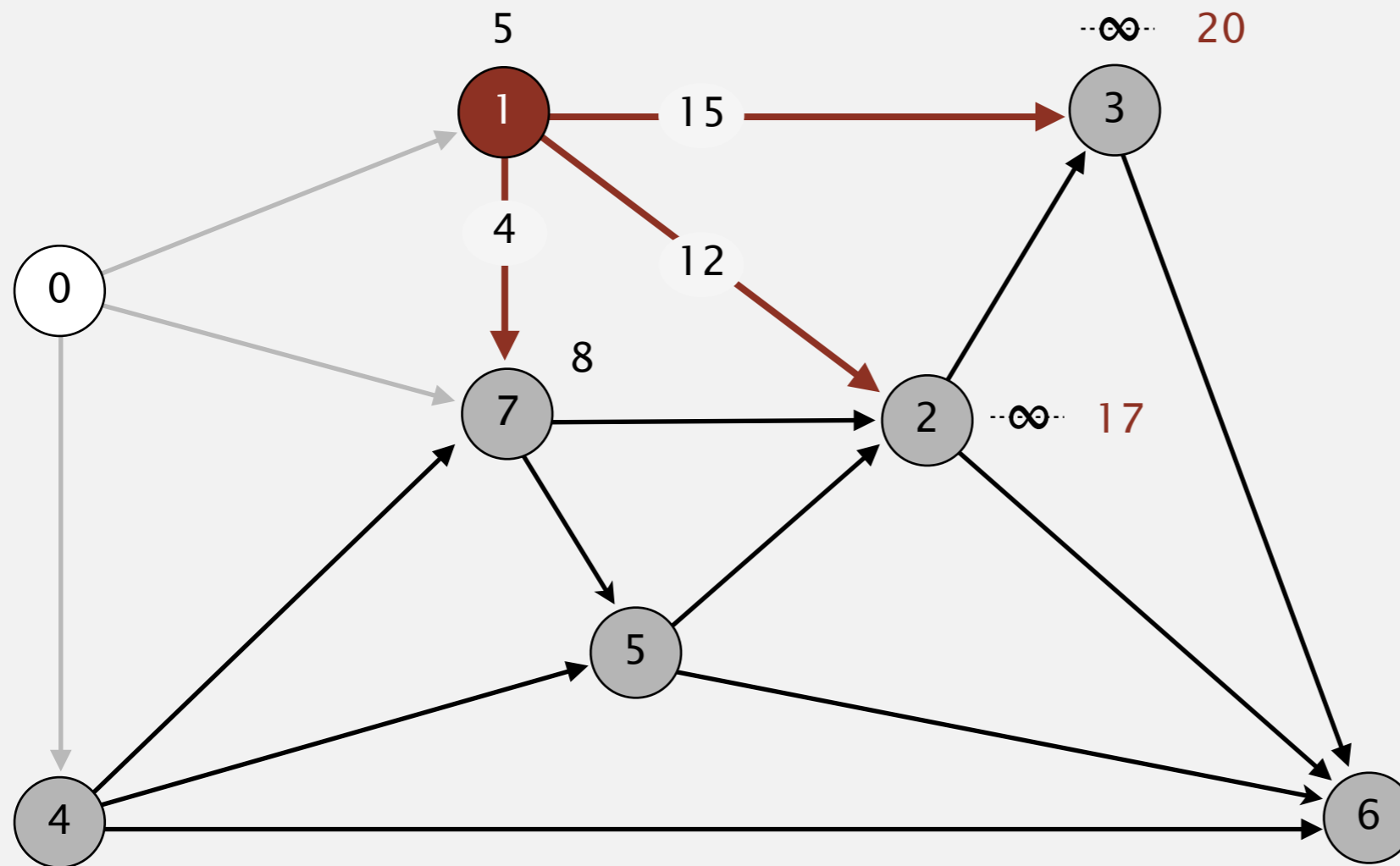
$v$	$\text{distTo}[]$	$\text{edgeTo}[]$
0	0.0	-
→ 1	5.0	0→1
2		
3		
4	9.0	0→4
5		
6		
7	8.0	0→7

relax all edges adjacent from 1



# Dijkstra's algorithm demo

- Consider vertices in increasing order of distance from  $s$  (non-tree vertex with the lowest  $\text{distTo}[]$  value).
- Add vertex to tree and relax all edges adjacent from that vertex.



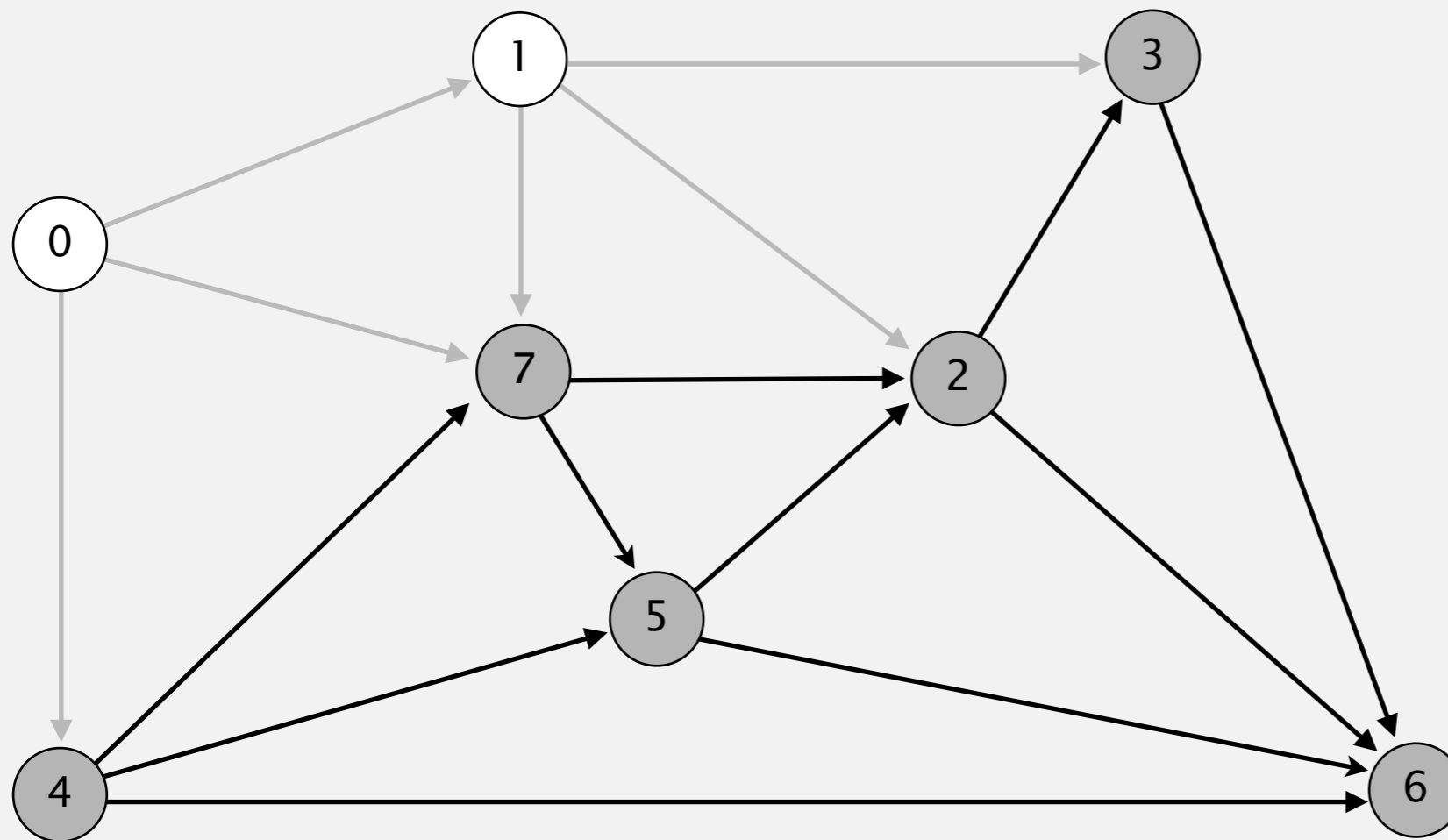
$v$	$\text{distTo}[]$	$\text{edgeTo}[]$
0	0.0	-
→ 1	5.0	0→1
2	17.0	1→2
3	20.0	1→3
4	9.0	0→4
5		
6		
7	8.0 ✓	0→7

relax all edges adjacent from 1

# Dijkstra's algorithm demo

---

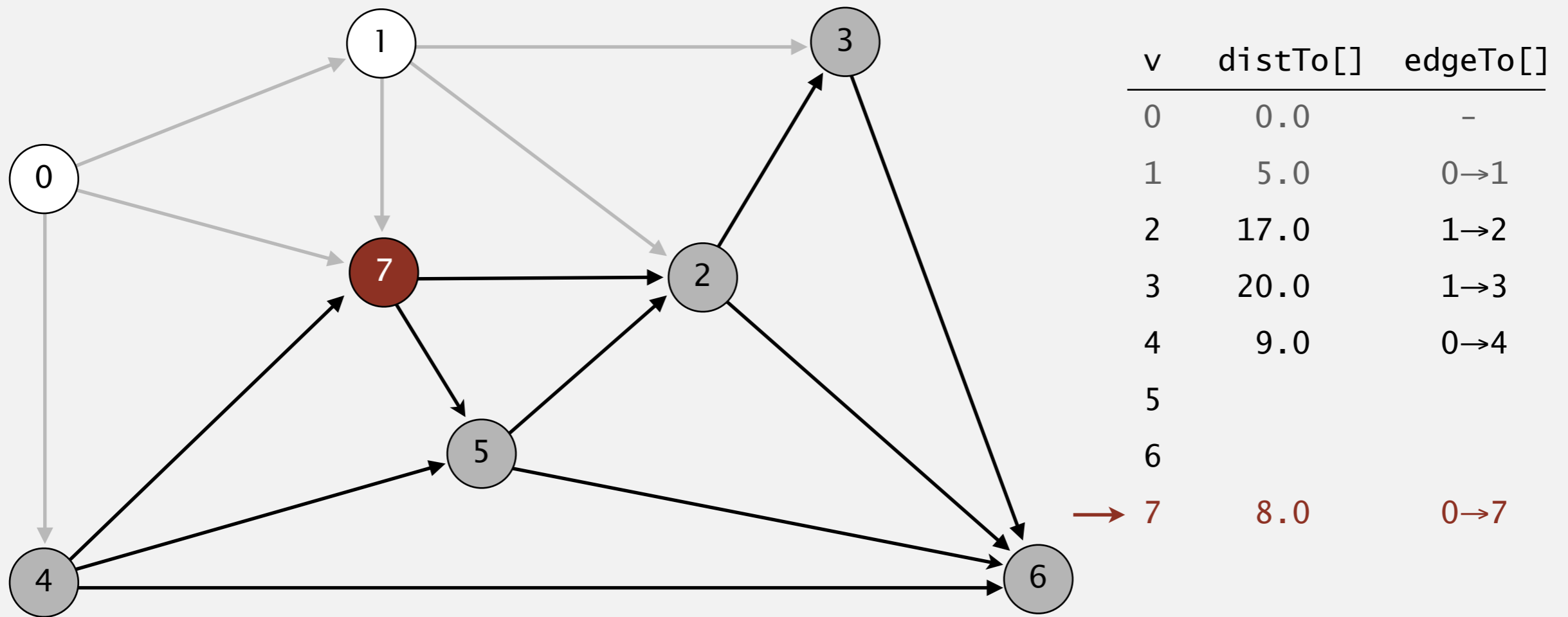
- Consider vertices in increasing order of distance from  $s$  (non-tree vertex with the lowest  $\text{distTo}[]$  value).
- Add vertex to tree and relax all edges adjacent from that vertex.



$v$	$\text{distTo}[]$	$\text{edgeTo}[]$
0	0.0	-
1	5.0	0→1
2	17.0	1→2
3	20.0	1→3
4	9.0	0→4
5		
6		
7	8.0	0→7

# Dijkstra's algorithm demo

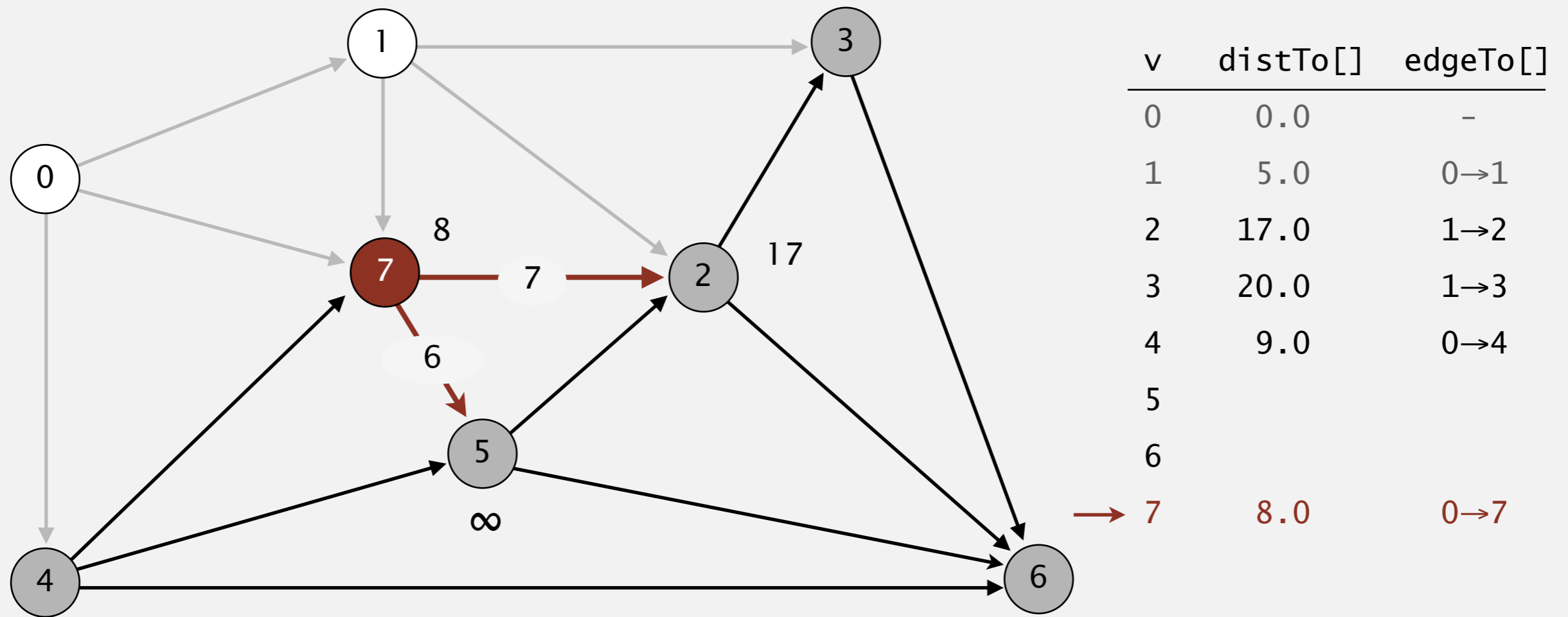
- Consider vertices in increasing order of distance from  $s$  (non-tree vertex with the lowest  $\text{distTo}[]$  value).
- Add vertex to tree and relax all edges adjacent from that vertex.



choose vertex 7

# Dijkstra's algorithm demo

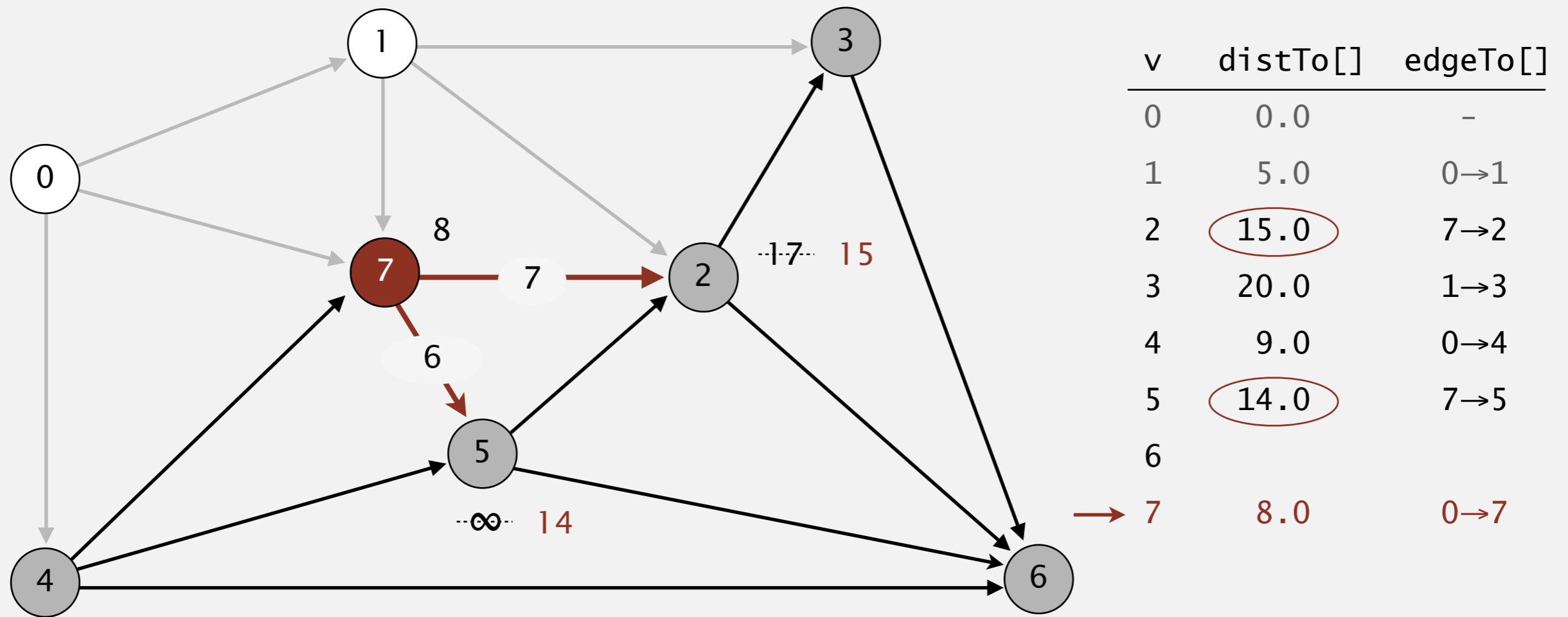
- Consider vertices in increasing order of distance from  $s$  (non-tree vertex with the lowest  $\text{distTo}[]$  value).
- Add vertex to tree and relax all edges adjacent from that vertex.



relax all edges adjacent from 7

# Dijkstra's algorithm demo

- Consider vertices in increasing order of distance from  $s$  (non-tree vertex with the lowest  $\text{distTo}[]$  value).
- Add vertex to tree and relax all edges adjacent from that vertex.

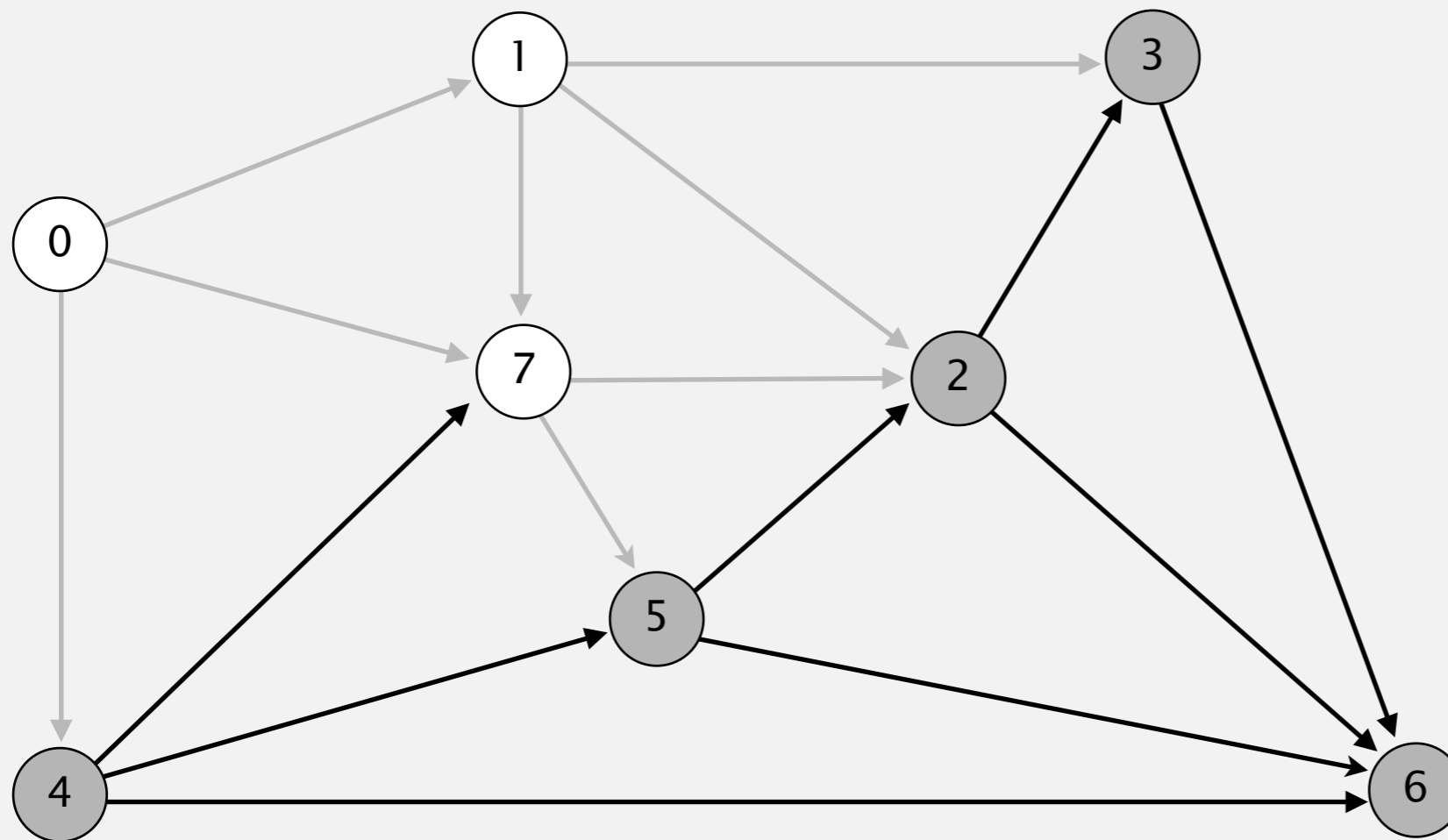


relax all edges adjacent from 7

# Dijkstra's algorithm demo

---

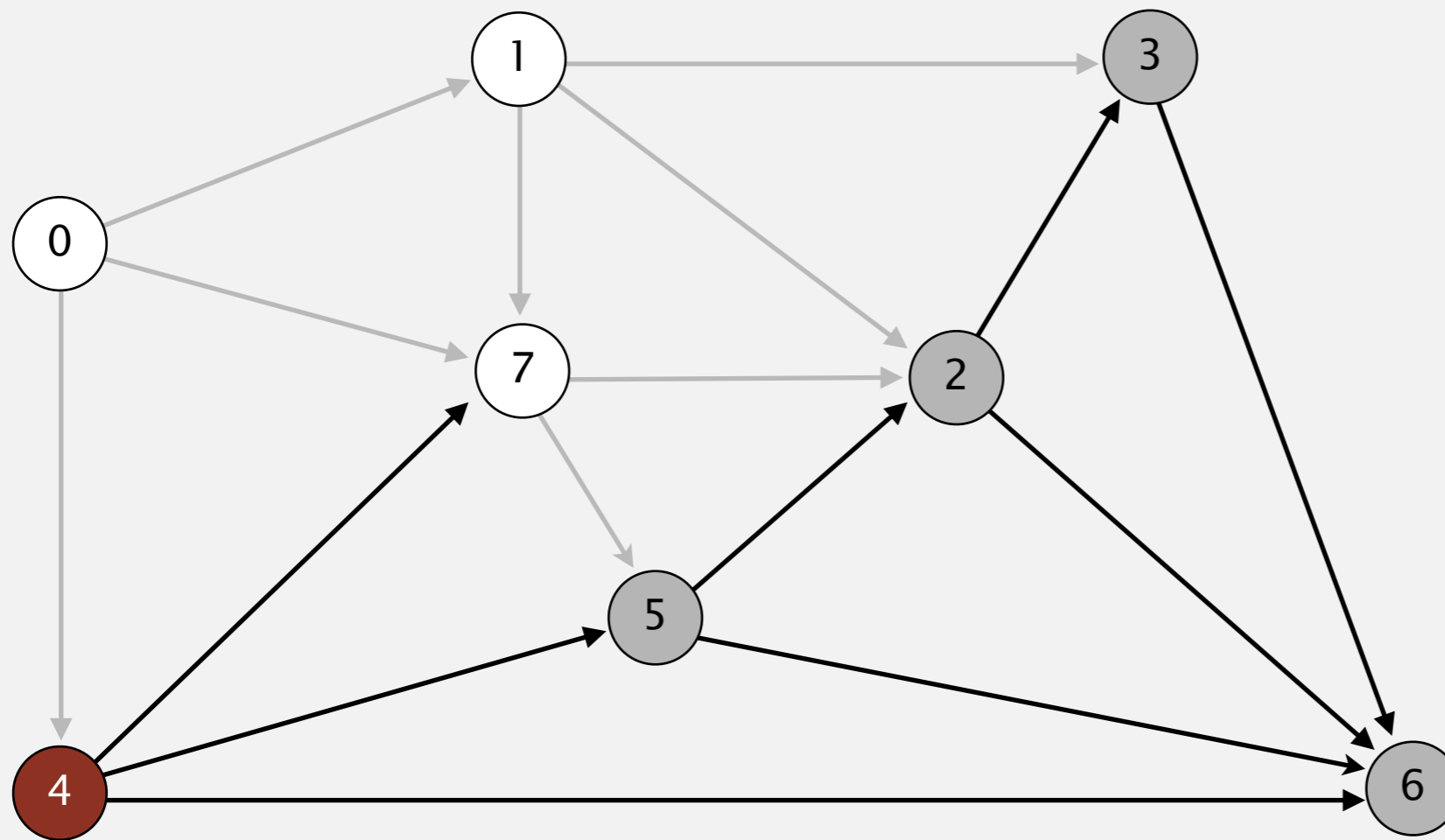
- Consider vertices in increasing order of distance from  $s$  (non-tree vertex with the lowest  $\text{distTo}[]$  value).
- Add vertex to tree and relax all edges adjacent from that vertex.



$v$	$\text{distTo}[]$	$\text{edgeTo}[]$
0	0.0	-
1	5.0	0→1
2	15.0	7→2
3	20.0	1→3
4	9.0	0→4
5	14.0	7→5
6		
7	8.0	0→7

# Dijkstra's algorithm demo

- Consider vertices in increasing order of distance from  $s$  (non-tree vertex with the lowest `distTo[]` value).
- Add vertex to tree and relax all edges adjacent from that vertex.

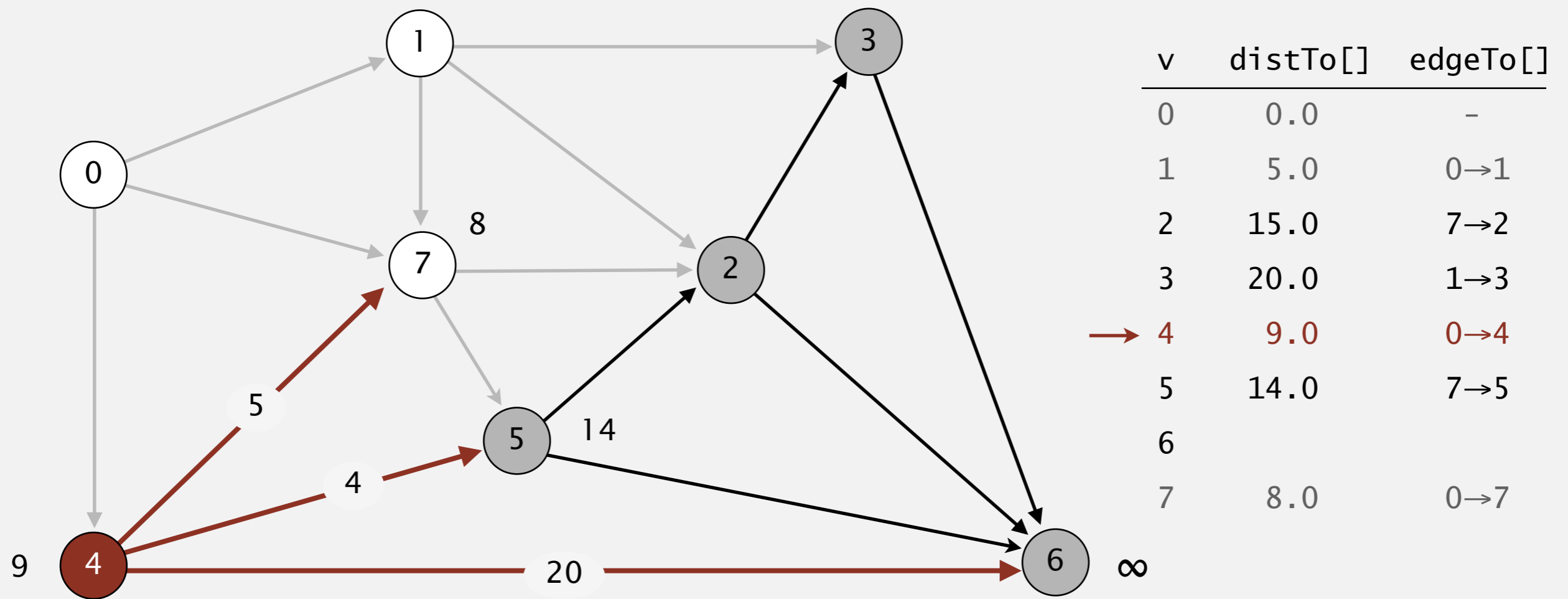


<u>v</u>	<u>distTo[]</u>	<u>edgeTo[]</u>
0	0.0	-
1	5.0	0→1
2	15.0	7→2
3	20.0	1→3
→ 4	9.0	0→4
5	14.0	7→5
6		
7	8.0	0→7

**select vertex 4**

# Dijkstra's algorithm demo

- Consider vertices in increasing order of distance from  $s$  (non-tree vertex with the lowest `distTo[]` value).
- Add vertex to tree and relax all edges adjacent from that vertex.

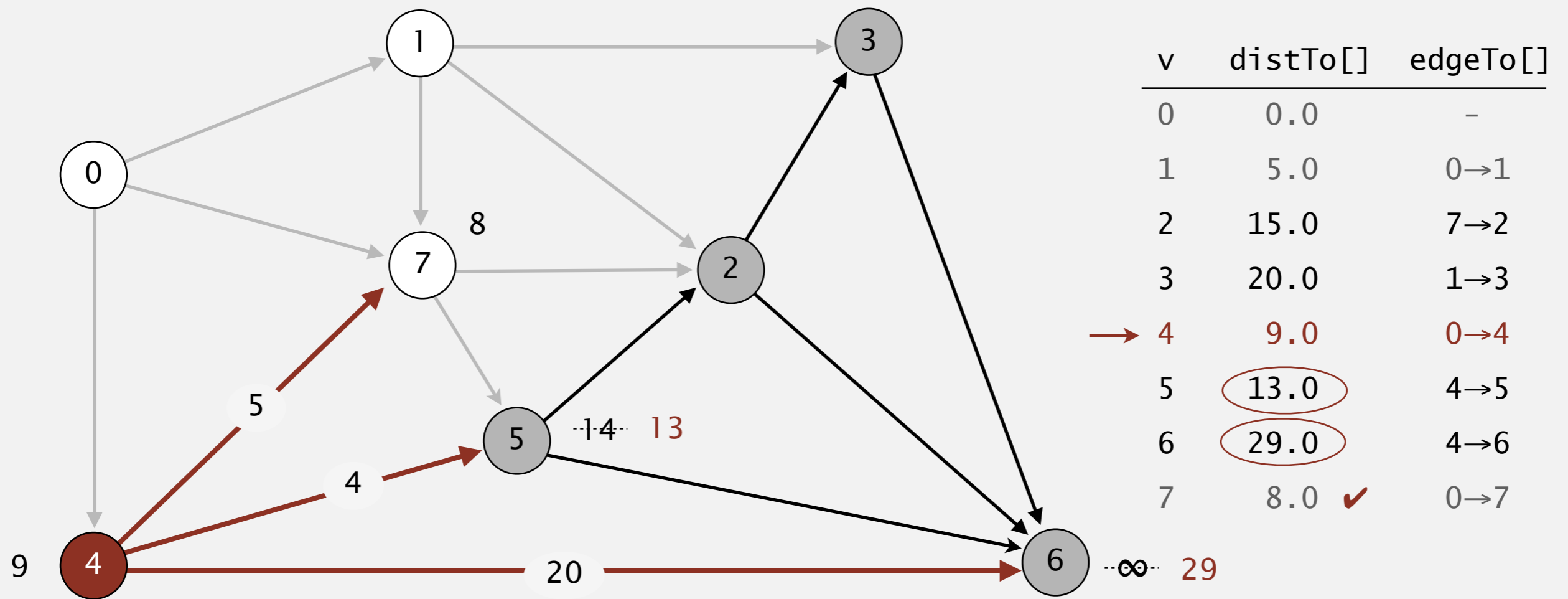


relax all edges adjacent from 4



# Dijkstra's algorithm demo

- Consider vertices in increasing order of distance from  $s$  (non-tree vertex with the lowest  $\text{distTo}[]$  value).
- Add vertex to tree and relax all edges adjacent from that vertex.

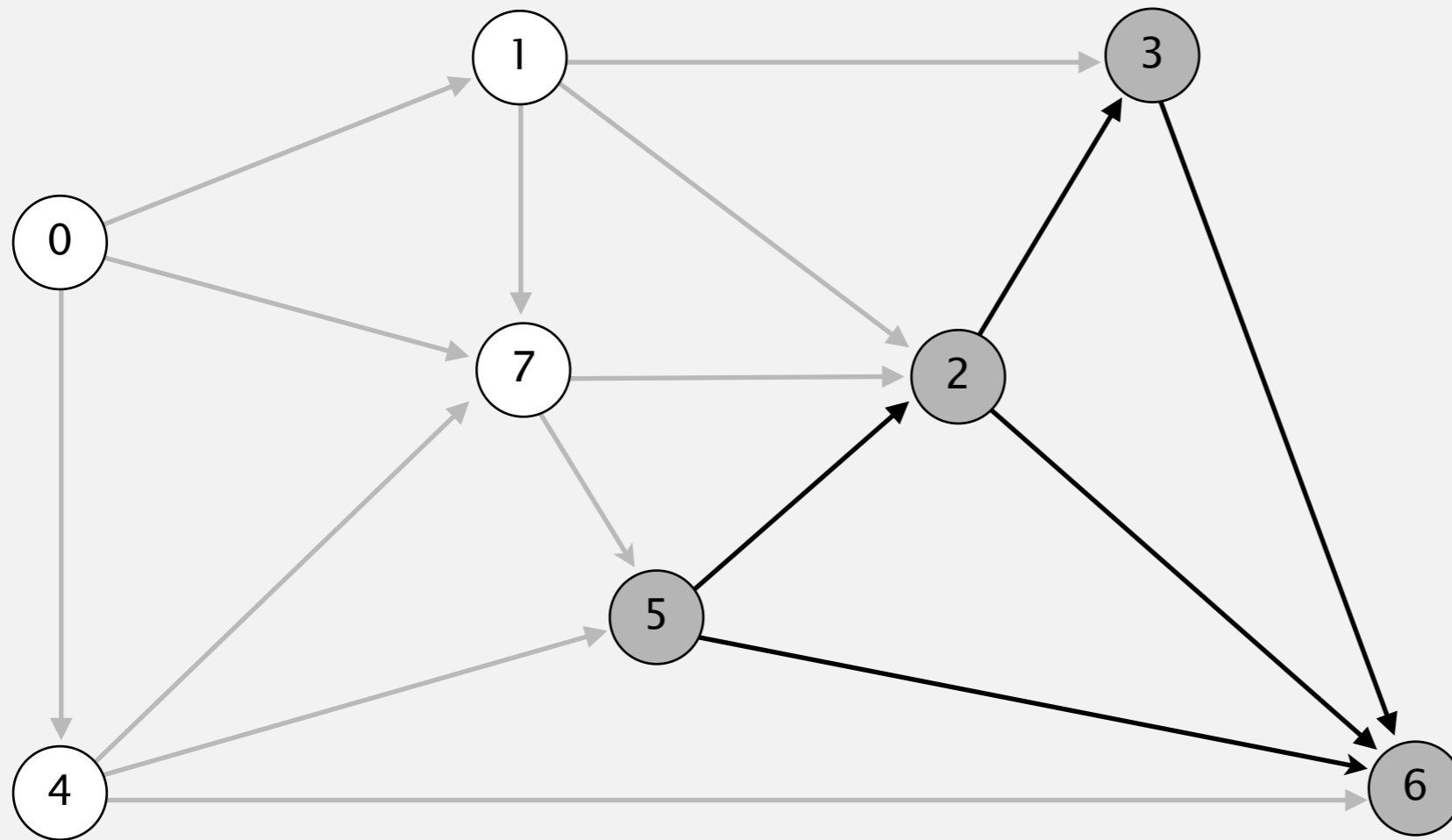


relax all edges adjacent from 4

# Dijkstra's algorithm demo

---

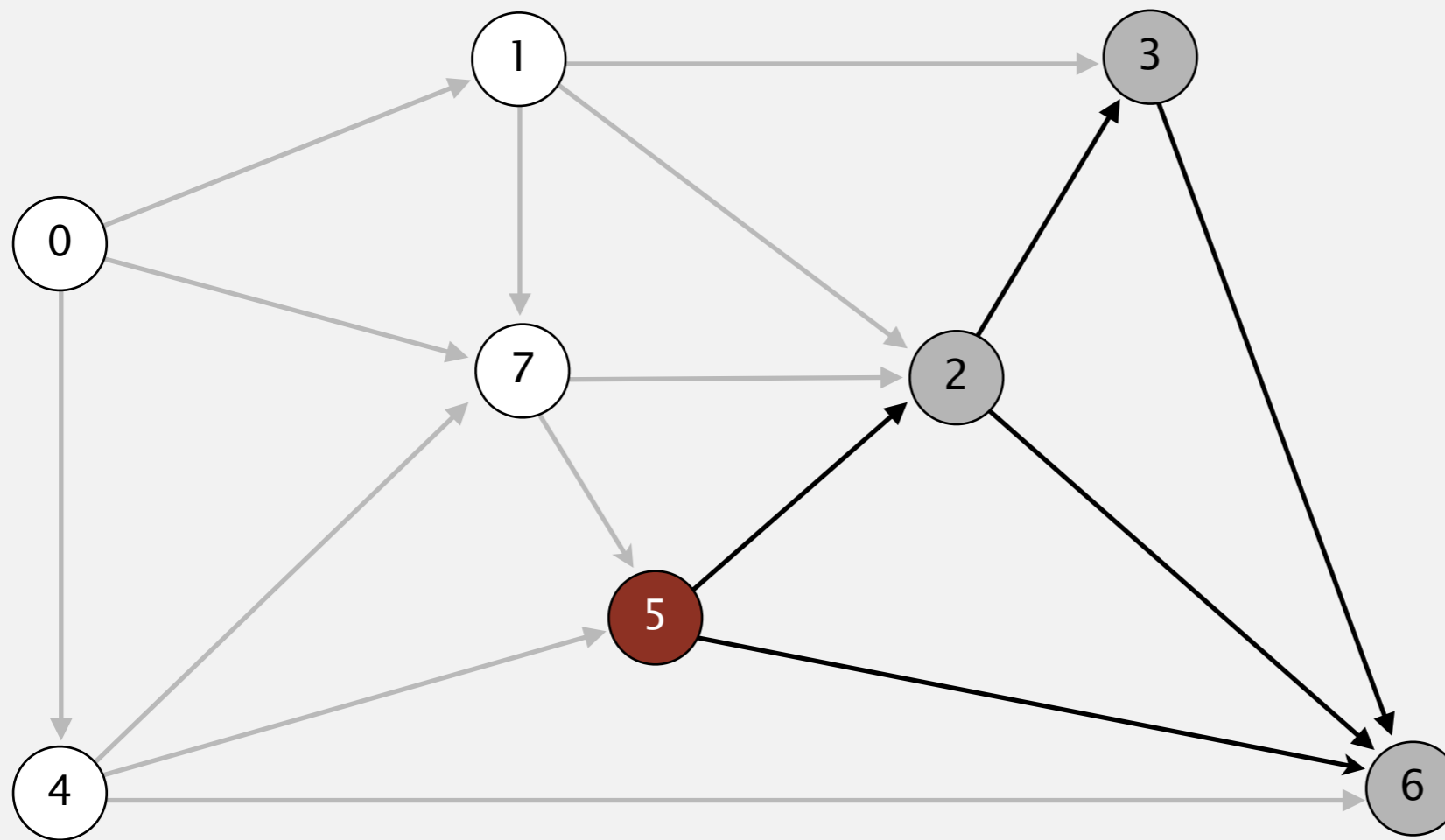
- Consider vertices in increasing order of distance from  $s$  (non-tree vertex with the lowest  $\text{distTo}[]$  value).
- Add vertex to tree and relax all edges adjacent from that vertex.



$v$	$\text{distTo}[]$	$\text{edgeTo}[]$
0	0.0	-
1	5.0	0→1
2	15.0	7→2
3	20.0	1→3
4	9.0	0→4
5	13.0	4→5
6	29.0	4→6
7	8.0	0→7

# Dijkstra's algorithm demo

- Consider vertices in increasing order of distance from  $s$  (non-tree vertex with the lowest  $\text{distTo}[]$  value).
- Add vertex to tree and relax all edges adjacent from that vertex.

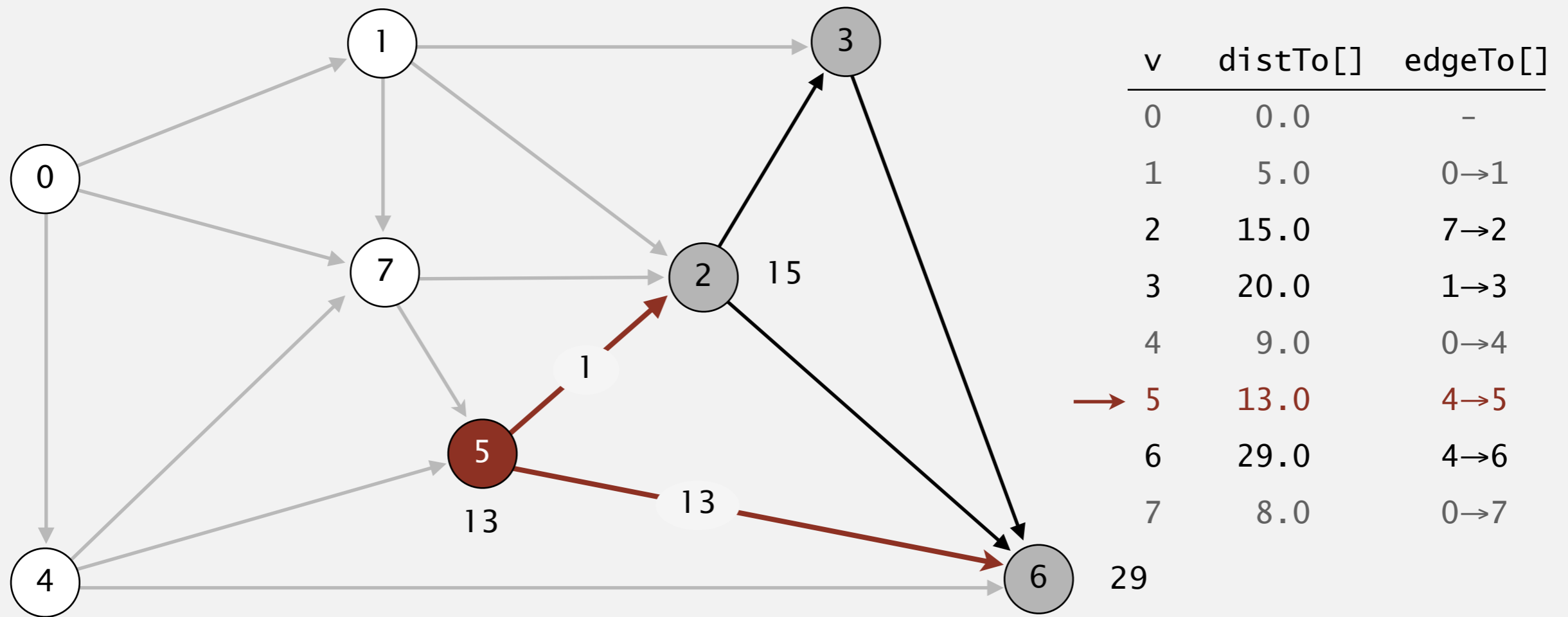


$v$	$\text{distTo}[]$	$\text{edgeTo}[]$
0	0.0	-
1	5.0	0→1
2	15.0	7→2
3	20.0	1→3
4	9.0	0→4
→ 5	13.0	4→5
6	29.0	4→6
7	8.0	0→7

**select vertex 5**

# Dijkstra's algorithm demo

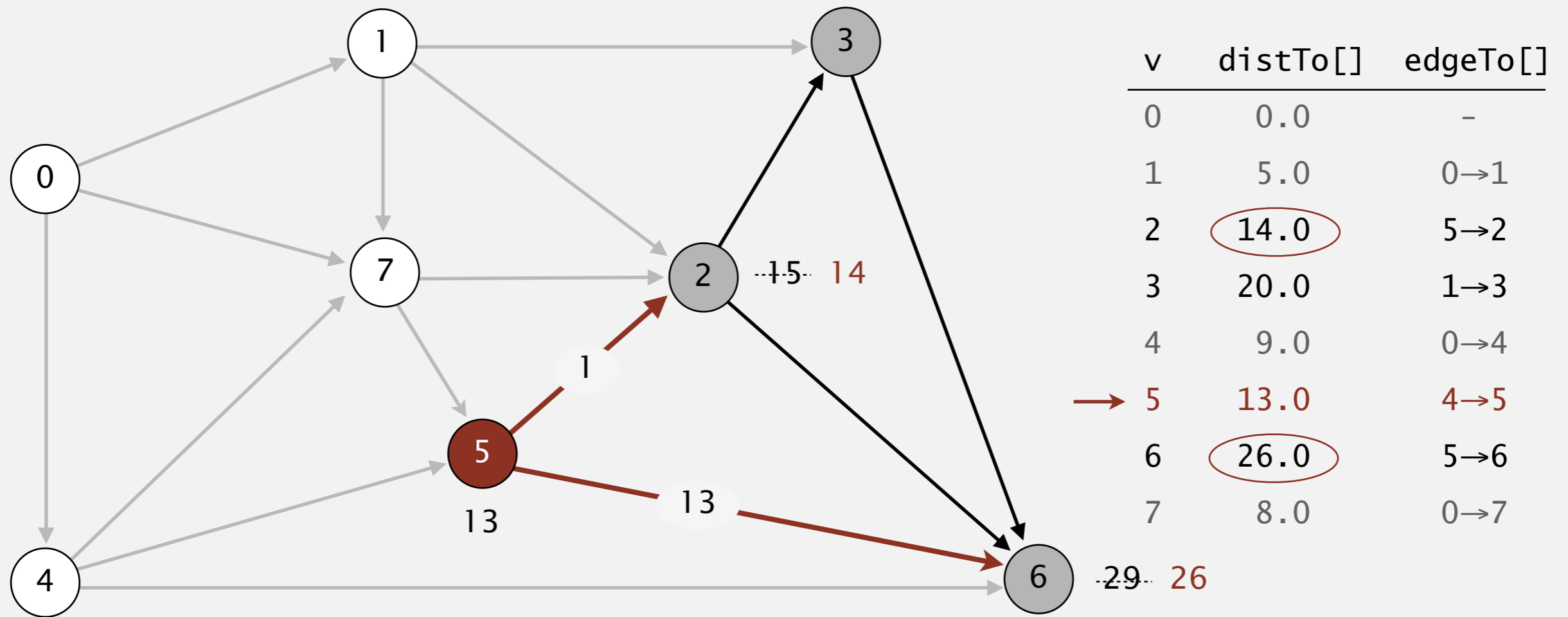
- Consider vertices in increasing order of distance from  $s$  (non-tree vertex with the lowest  $\text{distTo}[]$  value).
- Add vertex to tree and relax all edges adjacent from that vertex.



relax all edges adjacent from 5

# Dijkstra's algorithm demo

- Consider vertices in increasing order of distance from  $s$  (non-tree vertex with the lowest `distTo[]` value).
- Add vertex to tree and relax all edges adjacent from that vertex.

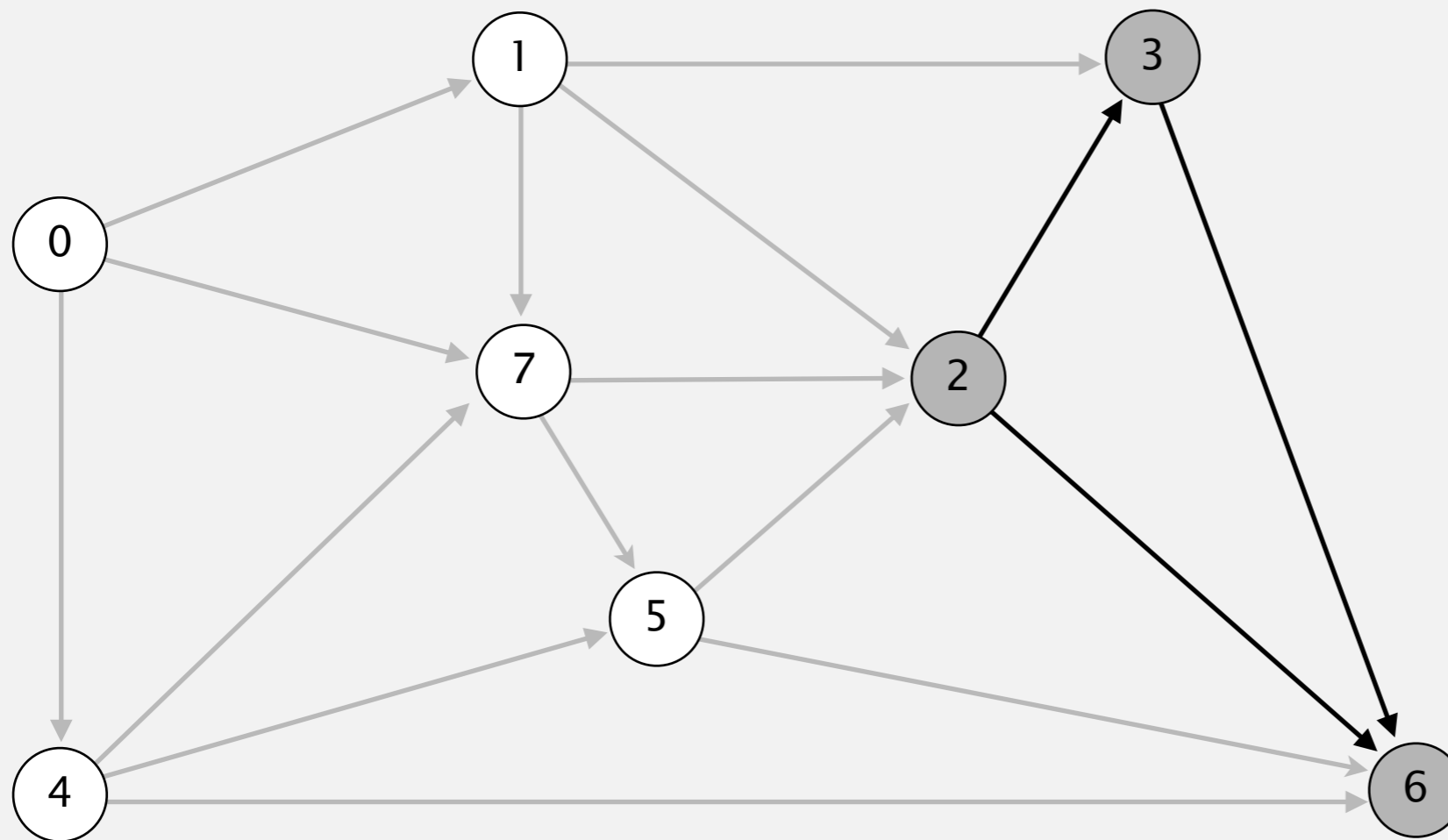


relax all edges adjacent from 5

# Dijkstra's algorithm demo

---

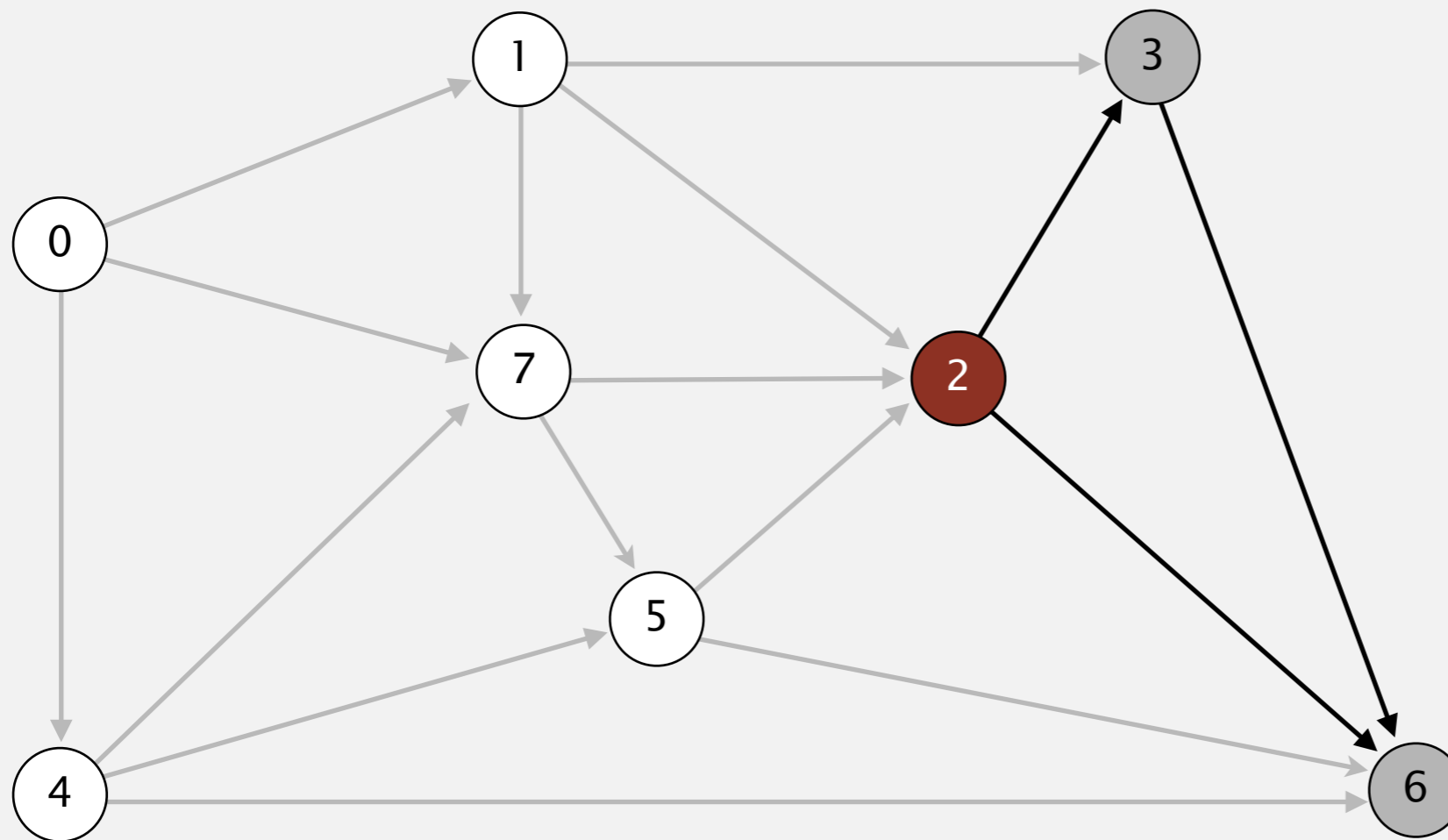
- Consider vertices in increasing order of distance from  $s$  (non-tree vertex with the lowest  $\text{distTo}[]$  value).
- Add vertex to tree and relax all edges adjacent from that vertex.



$v$	$\text{distTo}[]$	$\text{edgeTo}[]$
0	0.0	-
1	5.0	0→1
2	14.0	5→2
3	20.0	1→3
4	9.0	0→4
5	13.0	4→5
6	26.0	5→6
7	8.0	0→7

# Dijkstra's algorithm demo

- Consider vertices in increasing order of distance from  $s$  (non-tree vertex with the lowest  $\text{distTo}[]$  value).
- Add vertex to tree and relax all edges adjacent from that vertex.

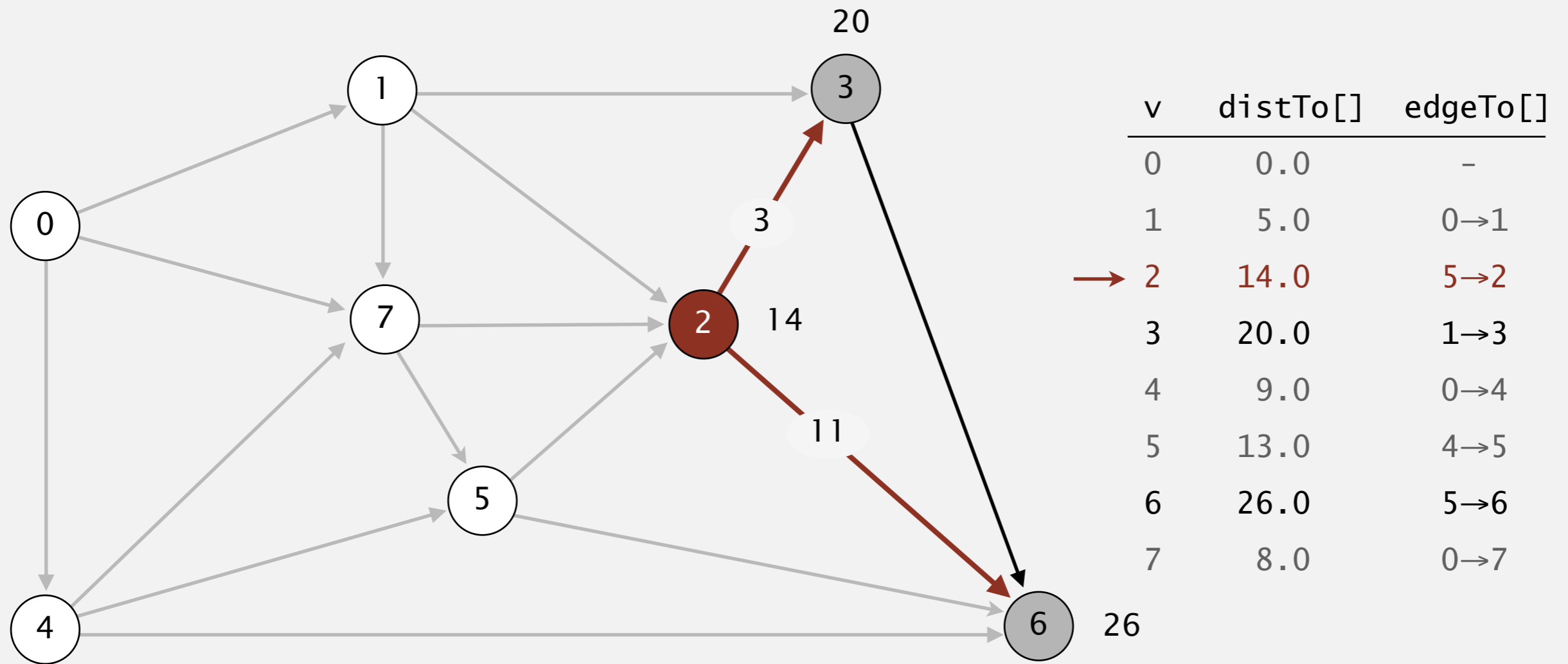


<u>v</u>	<u>distTo[]</u>	<u>edgeTo[]</u>
0	0.0	-
1	5.0	0→1
→ 2	14.0	5→2
3	20.0	1→3
4	9.0	0→4
5	13.0	4→5
6	26.0	5→6
7	8.0	0→7

**select vertex 2**

# Dijkstra's algorithm demo

- Consider vertices in increasing order of distance from  $s$  (non-tree vertex with the lowest  $\text{distTo}[]$  value).
- Add vertex to tree and relax all edges adjacent from that vertex.

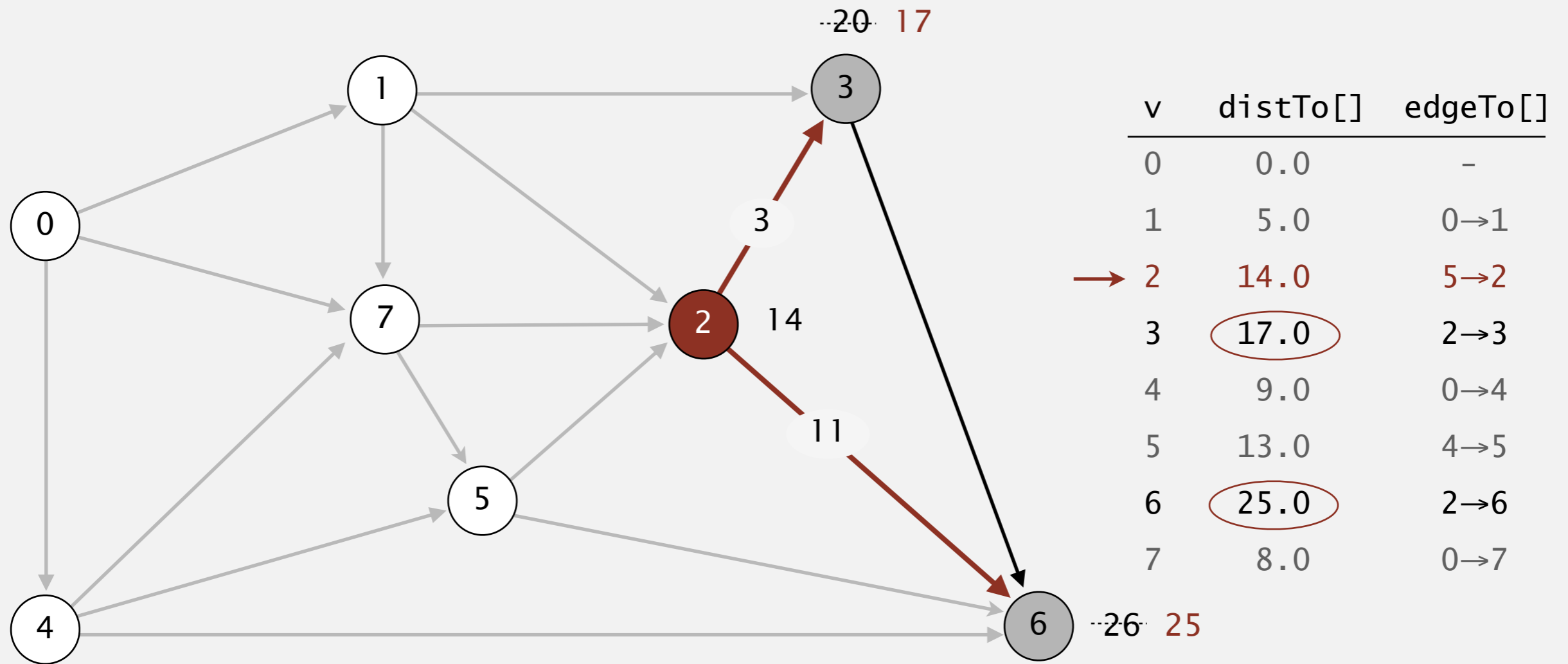


relax all edges adjacent from 2



# Dijkstra's algorithm demo

- Consider vertices in increasing order of distance from  $s$  (non-tree vertex with the lowest  $\text{distTo}[]$  value).
- Add vertex to tree and relax all edges adjacent from that vertex.

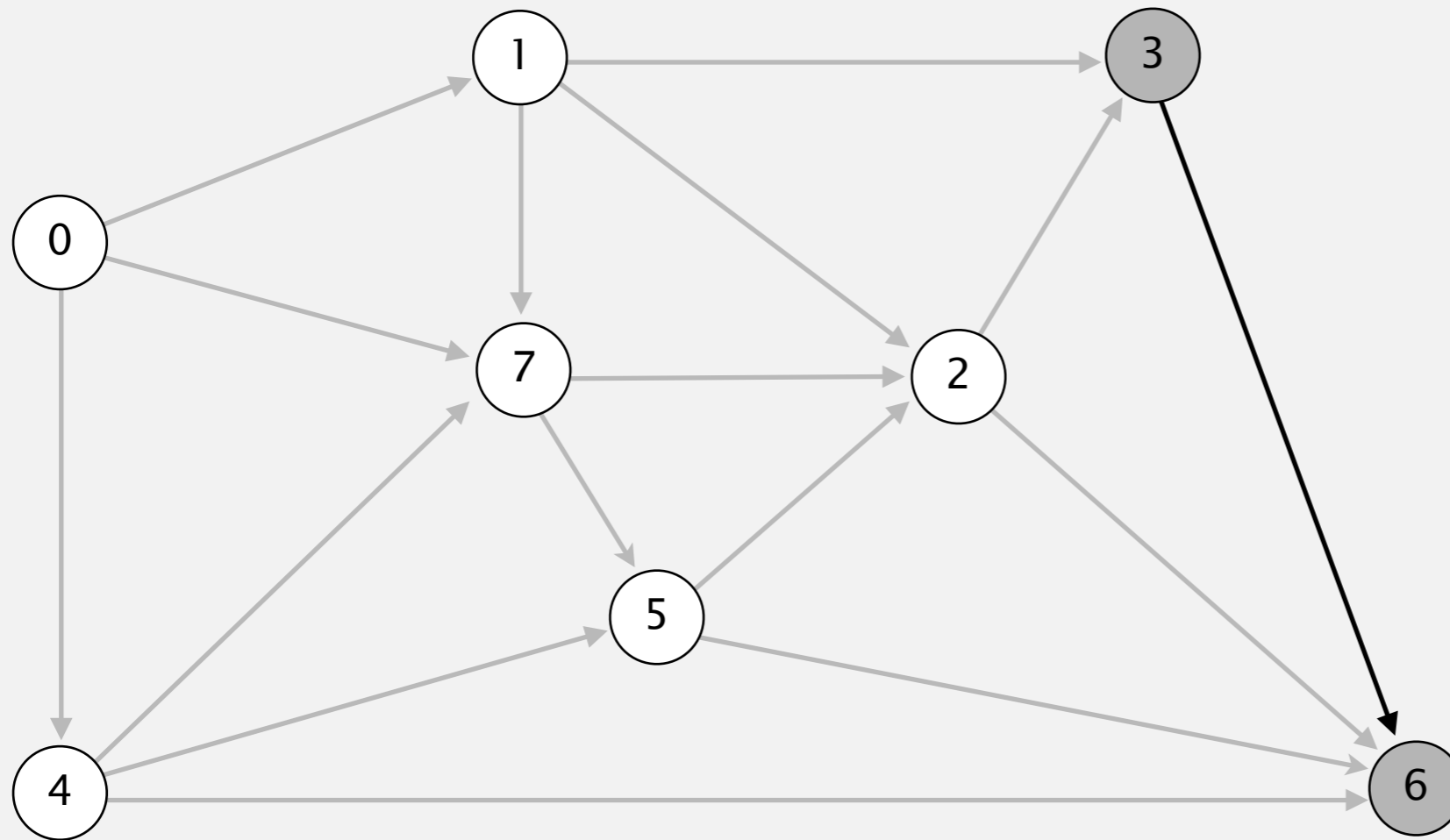


relax all edges adjacent from 2

# Dijkstra's algorithm demo

---

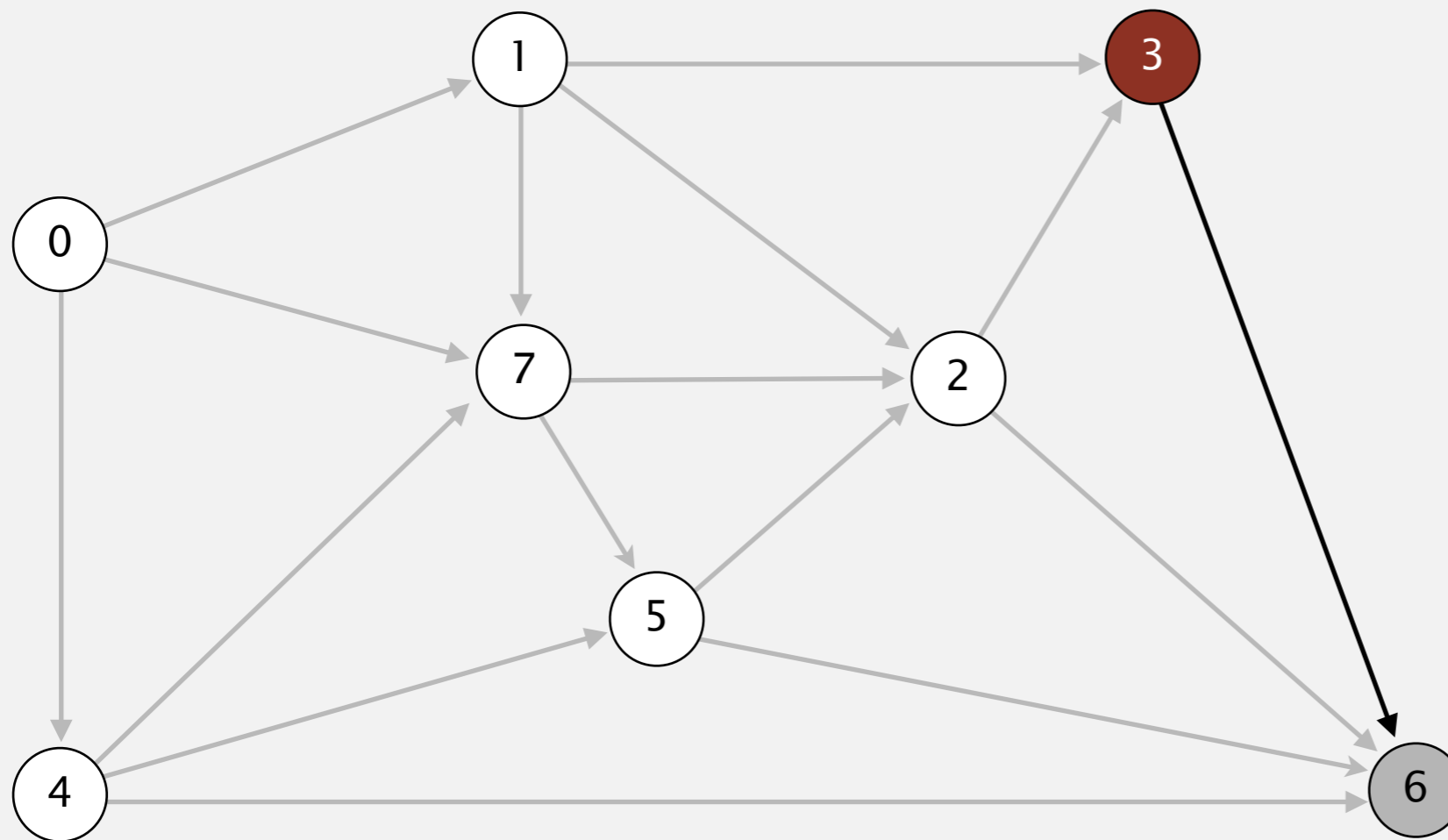
- Consider vertices in increasing order of distance from  $s$  (non-tree vertex with the lowest `distTo[]` value).
- Add vertex to tree and relax all edges adjacent from that vertex.



<code>v</code>	<code>distTo[]</code>	<code>edgeTo[]</code>
0	0.0	-
1	5.0	0→1
2	14.0	5→2
3	17.0	2→3
4	9.0	0→4
5	13.0	4→5
6	25.0	2→6
7	8.0	0→7

# Dijkstra's algorithm demo

- Consider vertices in increasing order of distance from  $s$  (non-tree vertex with the lowest `distTo[]` value).
- Add vertex to tree and relax all edges adjacent from that vertex.

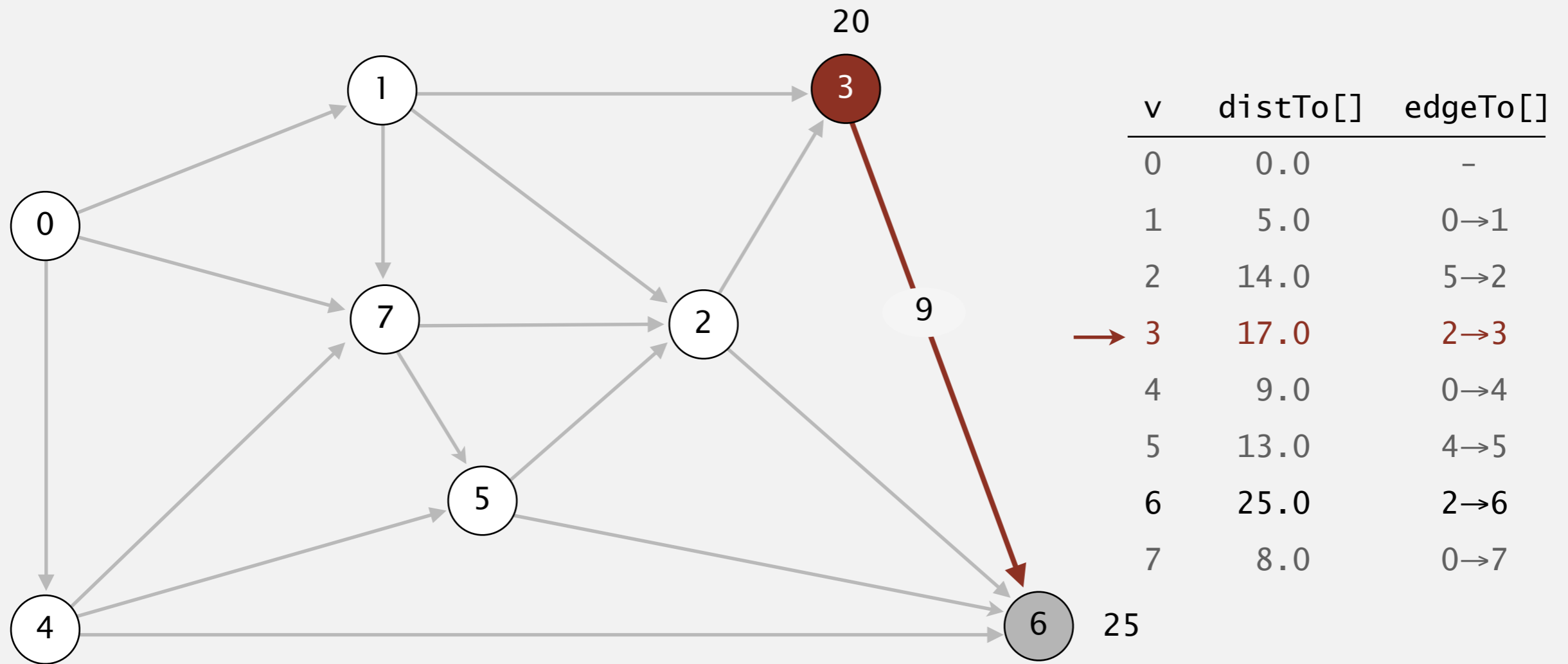


<u>v</u>	<u>distTo[]</u>	<u>edgeTo[]</u>
0	0.0	-
1	5.0	0→1
2	14.0	5→2
→ 3	17.0	2→3
4	9.0	0→4
5	13.0	4→5
6	25.0	2→6
7	8.0	0→7

**select vertex 3**

# Dijkstra's algorithm demo

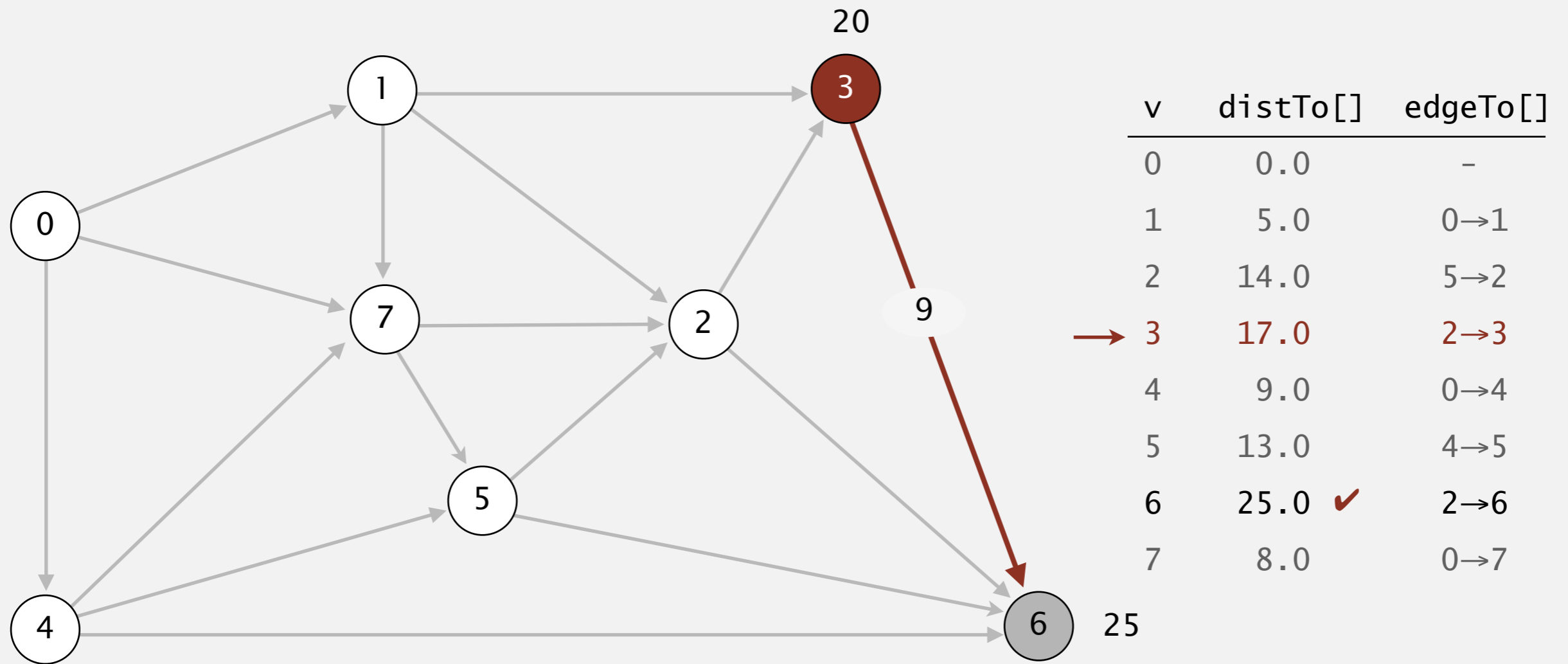
- Consider vertices in increasing order of distance from  $s$  (non-tree vertex with the lowest  $\text{distTo}[]$  value).
- Add vertex to tree and relax all edges adjacent from that vertex.



relax all edges adjacent from 3

# Dijkstra's algorithm demo

- Consider vertices in increasing order of distance from  $s$  (non-tree vertex with the lowest  $\text{distTo}[]$  value).
- Add vertex to tree and relax all edges adjacent from that vertex.

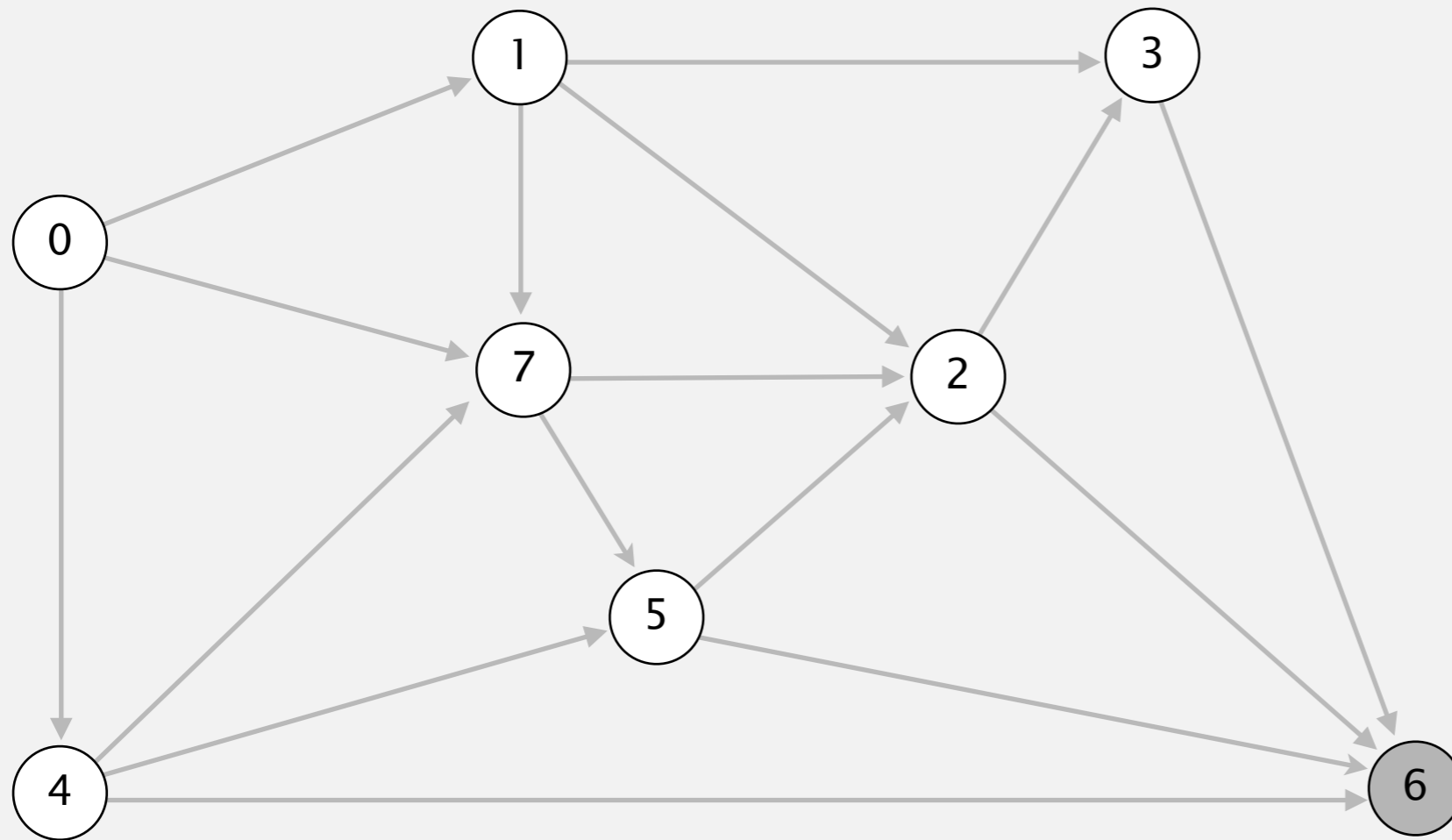


relax all edges adjacent from 3

# Dijkstra's algorithm demo

---

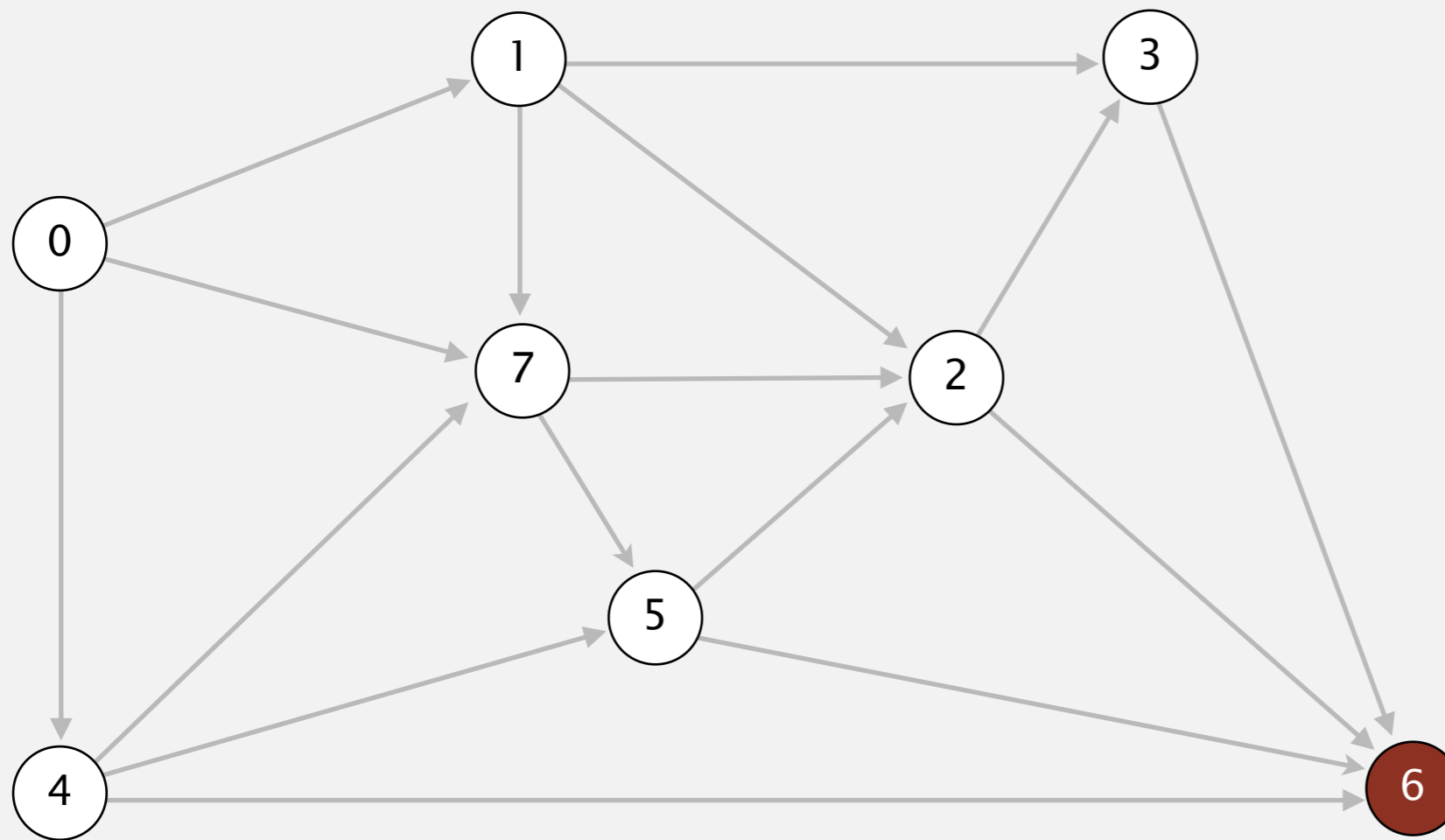
- Consider vertices in increasing order of distance from  $s$  (non-tree vertex with the lowest  $\text{distTo}[]$  value).
- Add vertex to tree and relax all edges adjacent from that vertex.



$v$	$\text{distTo}[]$	$\text{edgeTo}[]$
0	0.0	-
1	5.0	0→1
2	14.0	5→2
3	17.0	2→3
4	9.0	0→4
5	13.0	4→5
6	25.0	2→6
7	8.0	0→7

# Dijkstra's algorithm demo

- Consider vertices in increasing order of distance from  $s$  (non-tree vertex with the lowest  $\text{distTo}[]$  value).
- Add vertex to tree and relax all edges adjacent from that vertex.

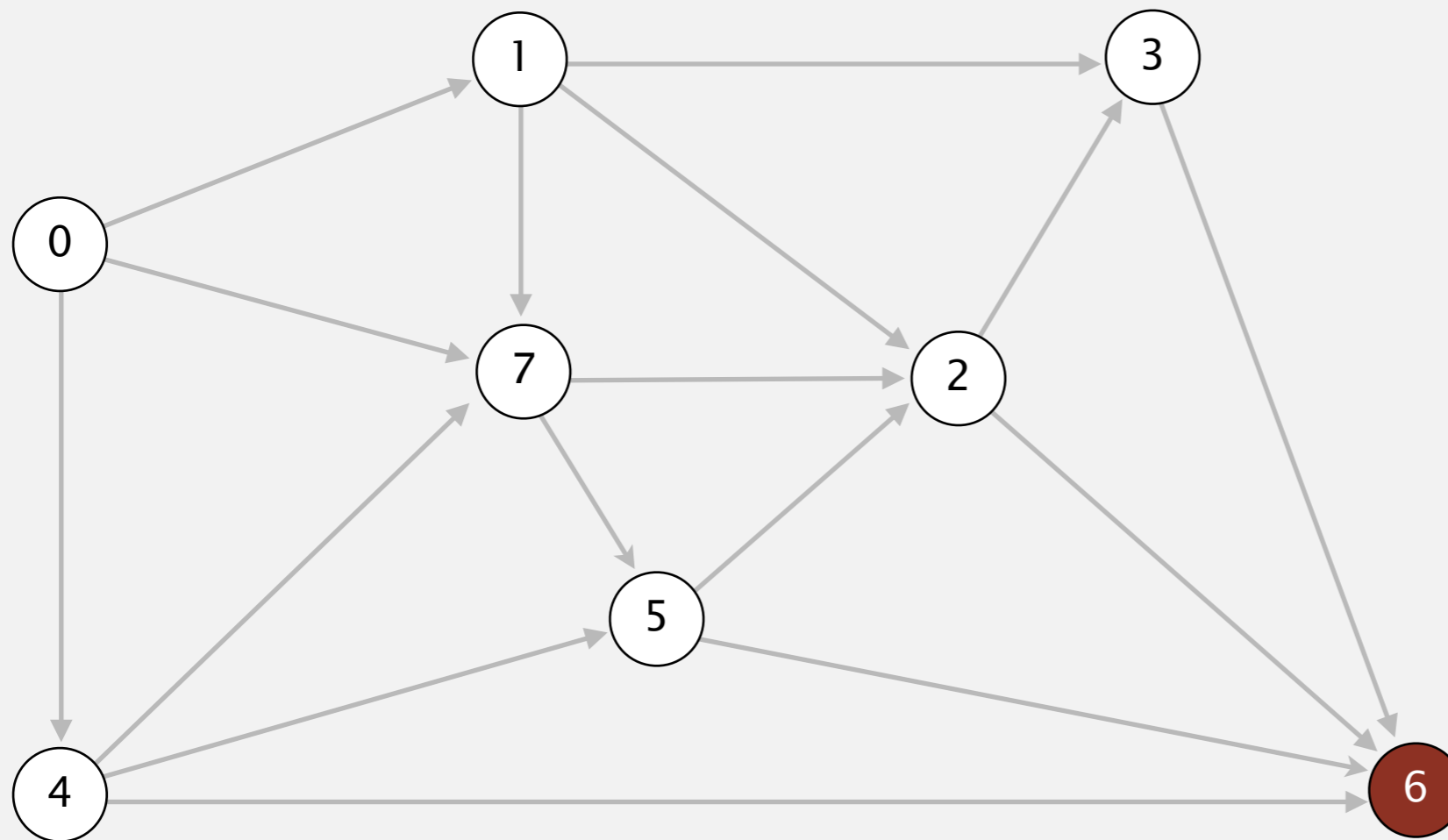


<u>v</u>	<u>distTo[]</u>	<u>edgeTo[]</u>
0	0.0	-
1	5.0	0→1
2	14.0	5→2
3	17.0	2→3
4	9.0	0→4
5	13.0	4→5
→ 6	25.0	2→6
7	8.0	0→7

**select vertex 6**

# Dijkstra's algorithm demo

- Consider vertices in increasing order of distance from  $s$  (non-tree vertex with the lowest `distTo[]` value).
- Add vertex to tree and relax all edges adjacent from that vertex.



<u>v</u>	<u>distTo[]</u>	<u>edgeTo[]</u>
0	0.0	-
1	5.0	0→1
2	14.0	5→2
3	17.0	2→3
4	9.0	0→4
5	13.0	4→5
→ 6	25.0	2→6
7	8.0	0→7

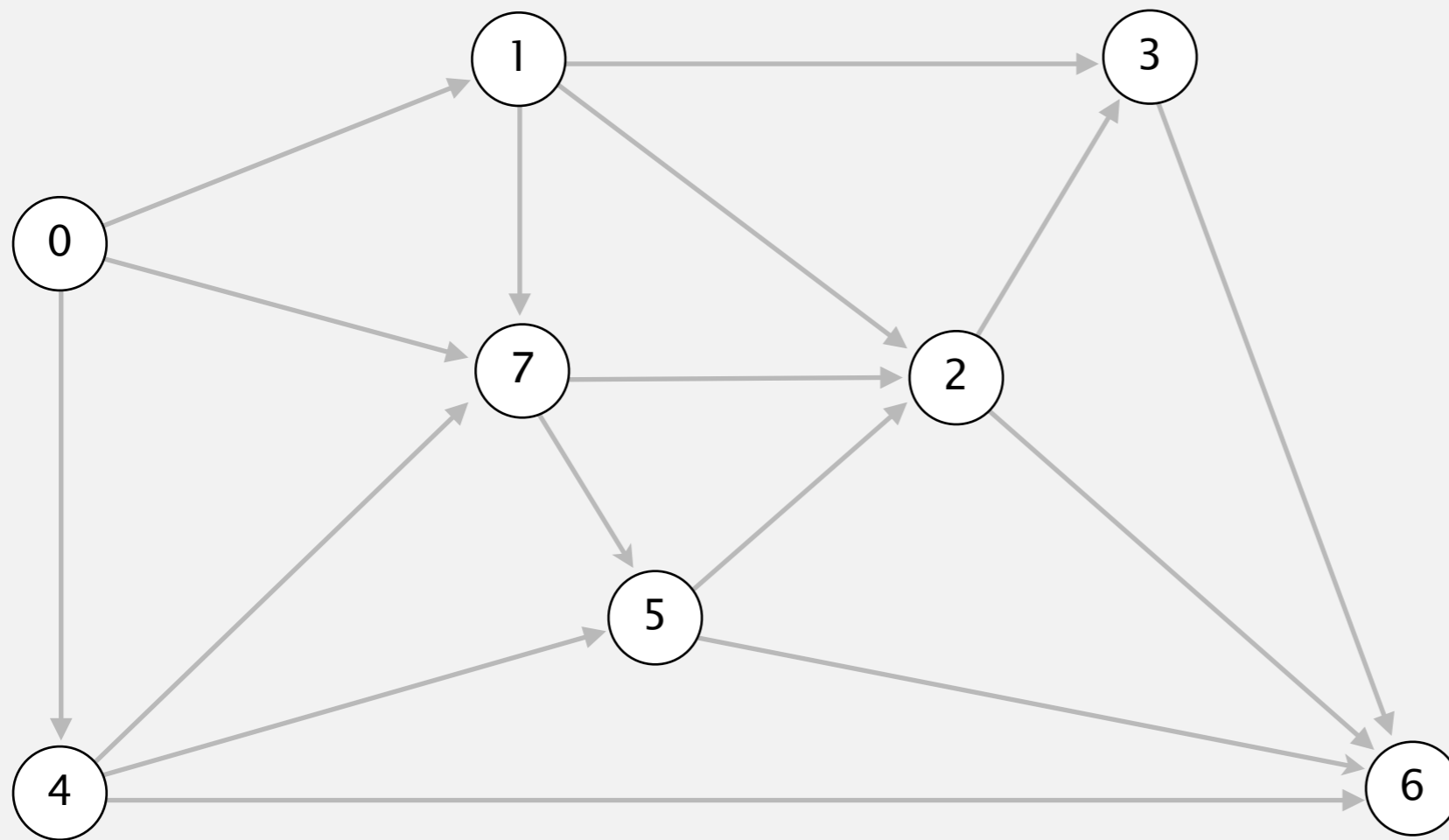
relax all edges adjacent from 6



# Dijkstra's algorithm demo

---

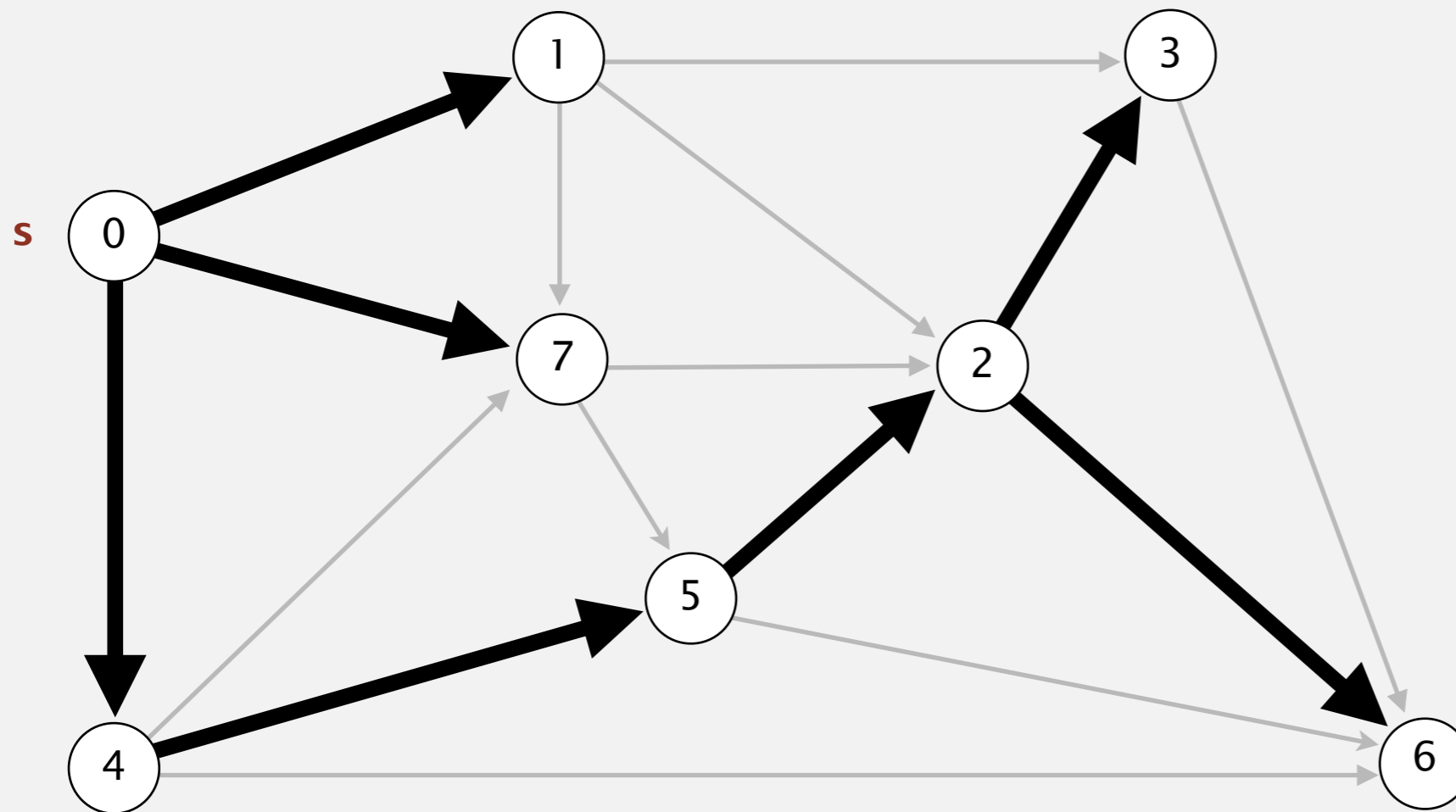
- Consider vertices in increasing order of distance from  $s$  (non-tree vertex with the lowest  $\text{distTo}[]$  value).
- Add vertex to tree and relax all edges adjacent from that vertex.



$v$	$\text{distTo}[]$	$\text{edgeTo}[]$
0	0.0	-
1	5.0	0→1
2	14.0	5→2
3	17.0	2→3
4	9.0	0→4
5	13.0	4→5
6	25.0	2→6
7	8.0	0→7

# Dijkstra's algorithm demo

- Consider vertices in increasing order of distance from  $s$  (non-tree vertex with the lowest `distTo[]` value).
- Add vertex to tree and relax all edges adjacent from that vertex.



<code>v</code>	<code>distTo[]</code>	<code>edgeTo[]</code>
0	0.0	-
1	5.0	0→1
2	14.0	5→2
3	17.0	2→3
4	9.0	0→4
5	13.0	4→5
6	25.0	2→6
7	8.0	0→7

shortest-paths tree from vertex  $s$