



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

PRIM'S ALGORITHM DEMO

- *Prim's algorithm*
- *lazy implementation*
- *eager implementation*



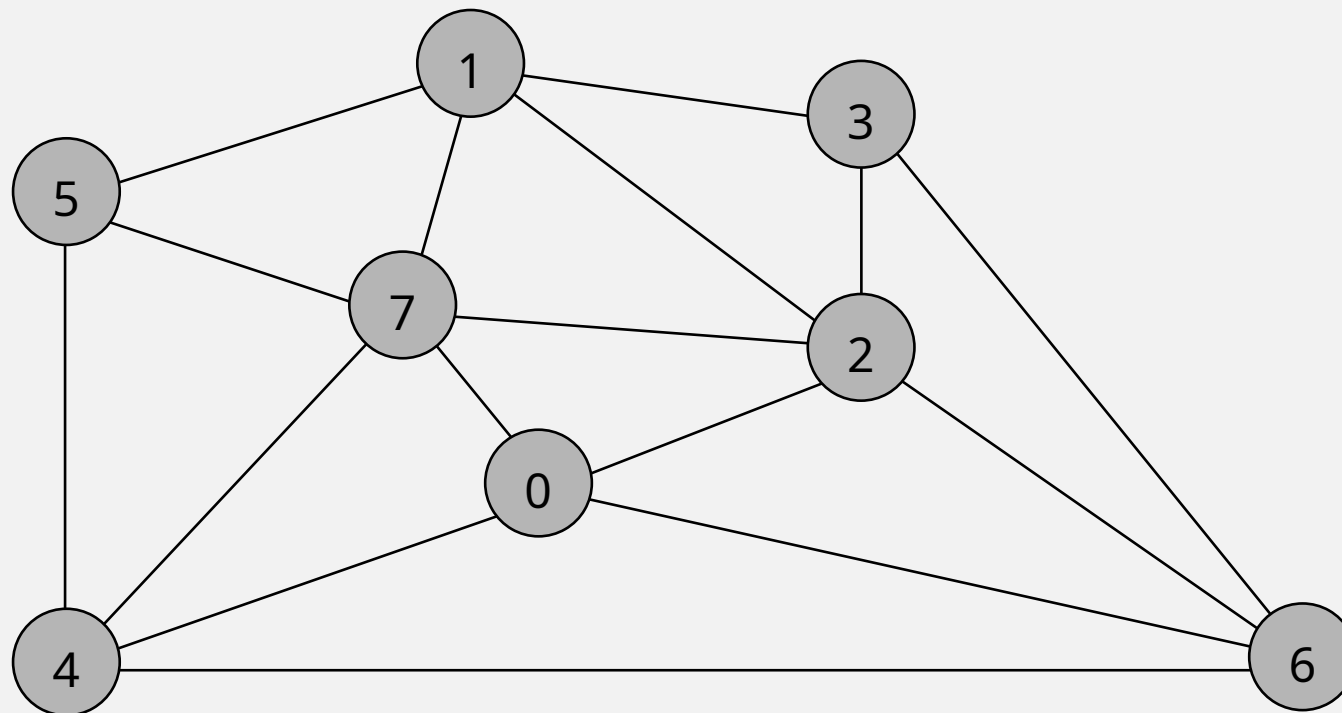
<http://algs4.cs.princeton.edu>

PRIM'S ALGORITHM DEMO

- *Prim's algorithm*
- *lazy implementation*
- *eager implementation*

Prim's algorithm demo

- Start with vertex 0 and greedily grow tree T .
- Add to T the min weight edge with exactly one endpoint in T .
- Repeat until $V - 1$ edges.



an edge-weighted graph

0-7 0.16

2-3 0.17

1-7 0.19

0-2 0.26

5-7 0.28

1-3 0.29

1-5 0.32

2-7 0.34

4-5 0.35

1-2 0.36

4-7 0.37

0-4 0.38

6-2 0.40

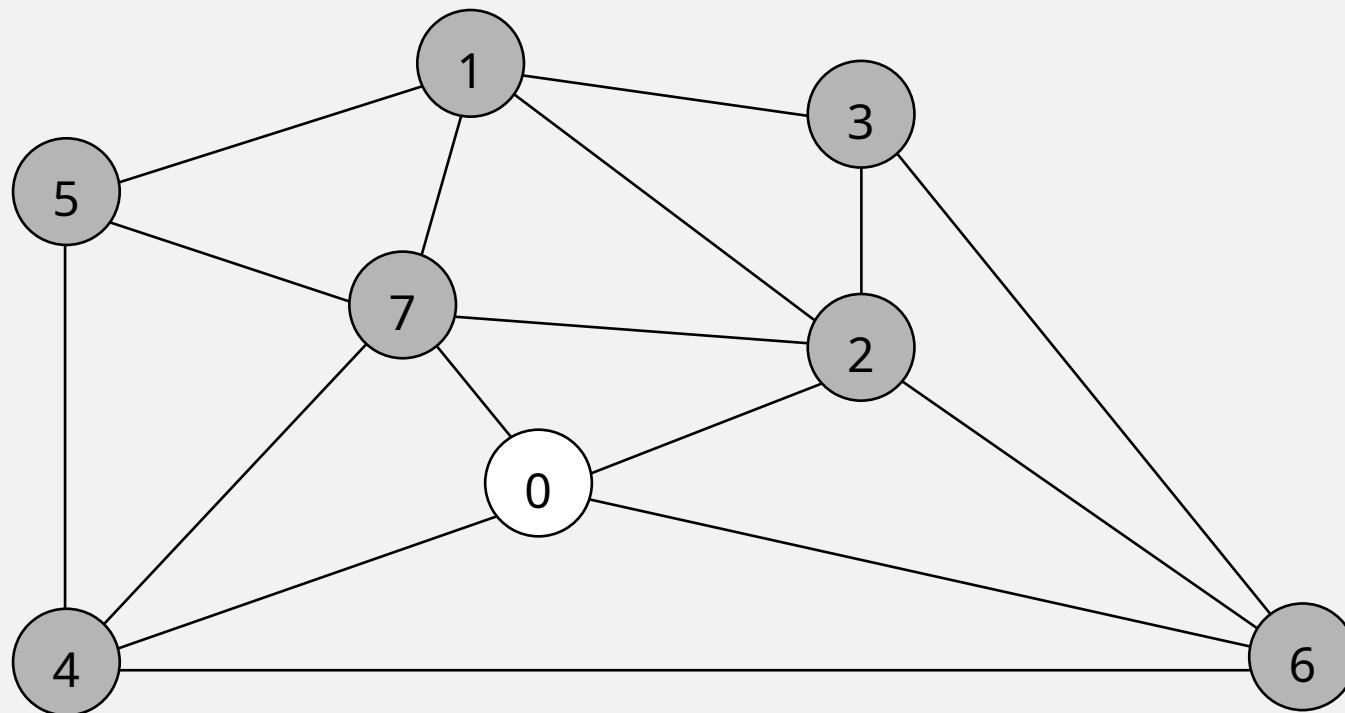
3-6 0.52

6-0 0.58

0-1 0.60

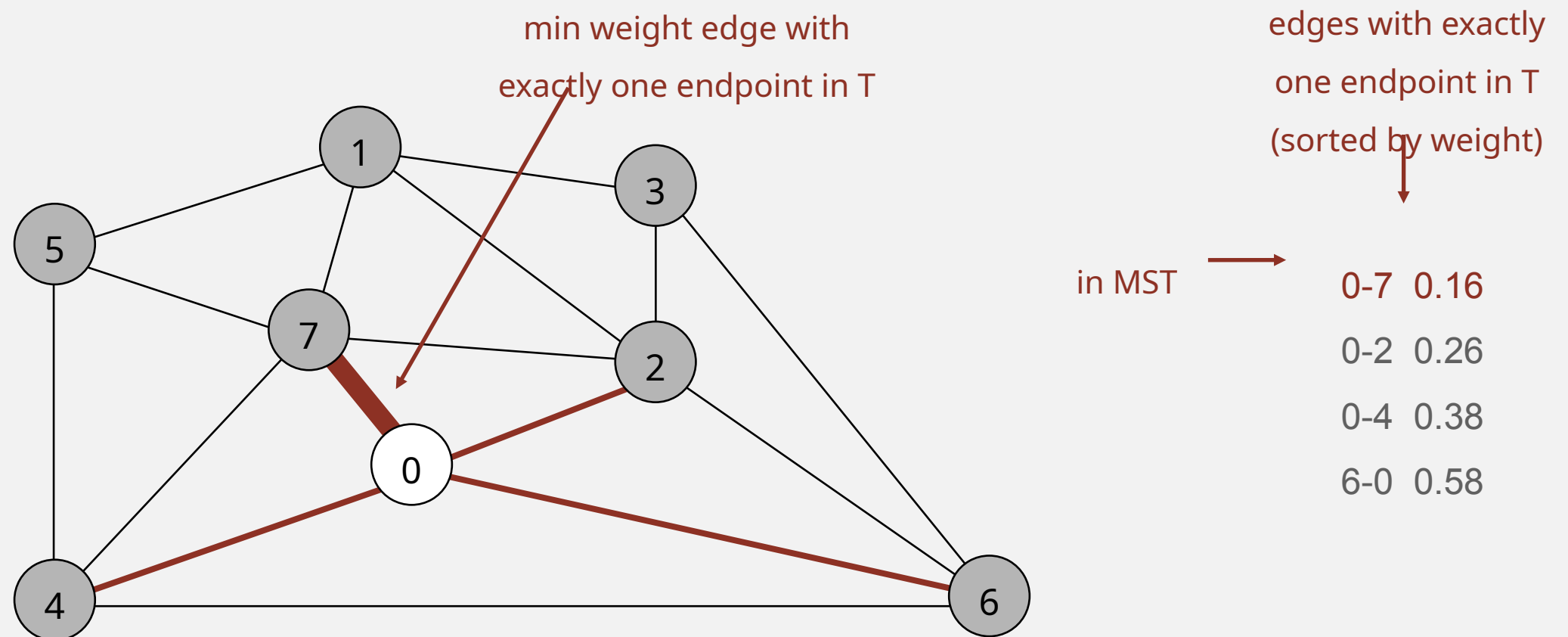
Prim's algorithm demo

- Start with vertex 0 and greedily grow tree T .
- Add to T the min weight edge with exactly one endpoint in T .
- Repeat until $V - 1$ edges.



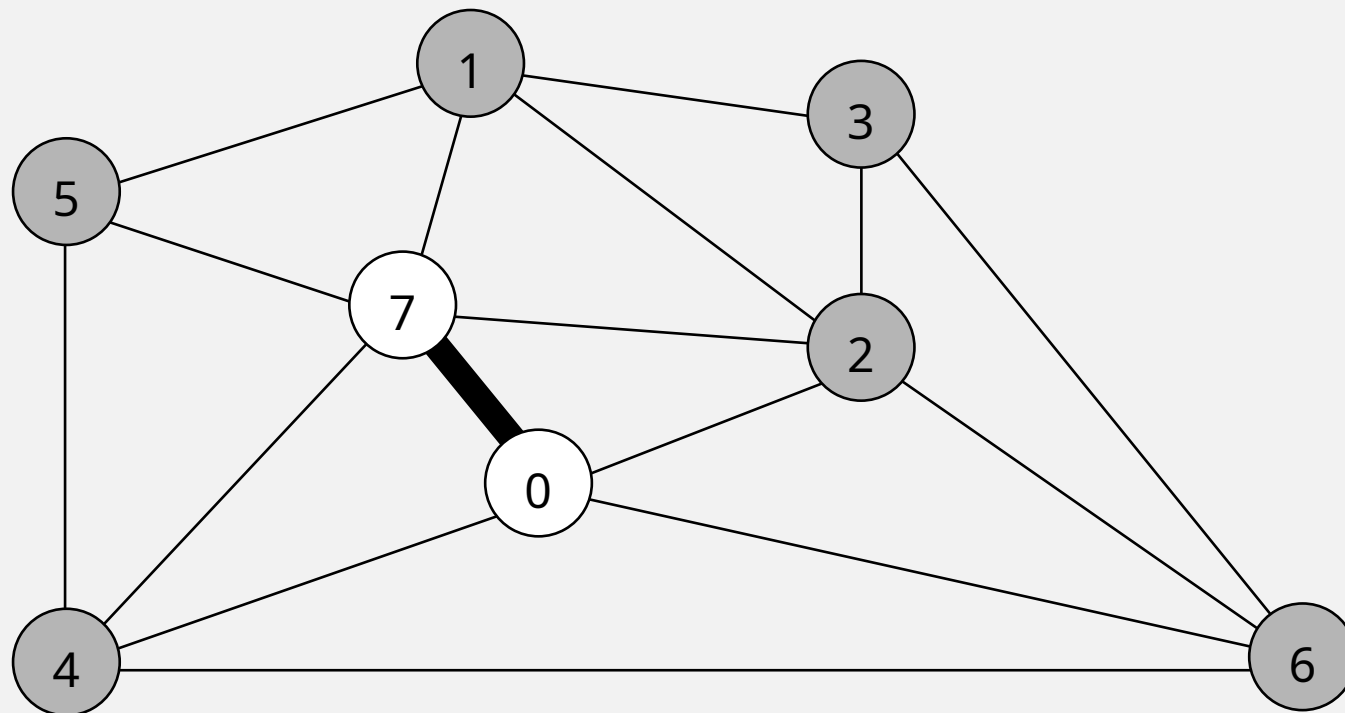
Prim's algorithm demo

- Start with vertex 0 and greedily grow tree T .
- Add to T the min weight edge with exactly one endpoint in T .
- Repeat until $V - 1$ edges.



Prim's algorithm demo

- Start with vertex 0 and greedily grow tree T .
- Add to T the min weight edge with exactly one endpoint in T .
- Repeat until $V - 1$ edges.

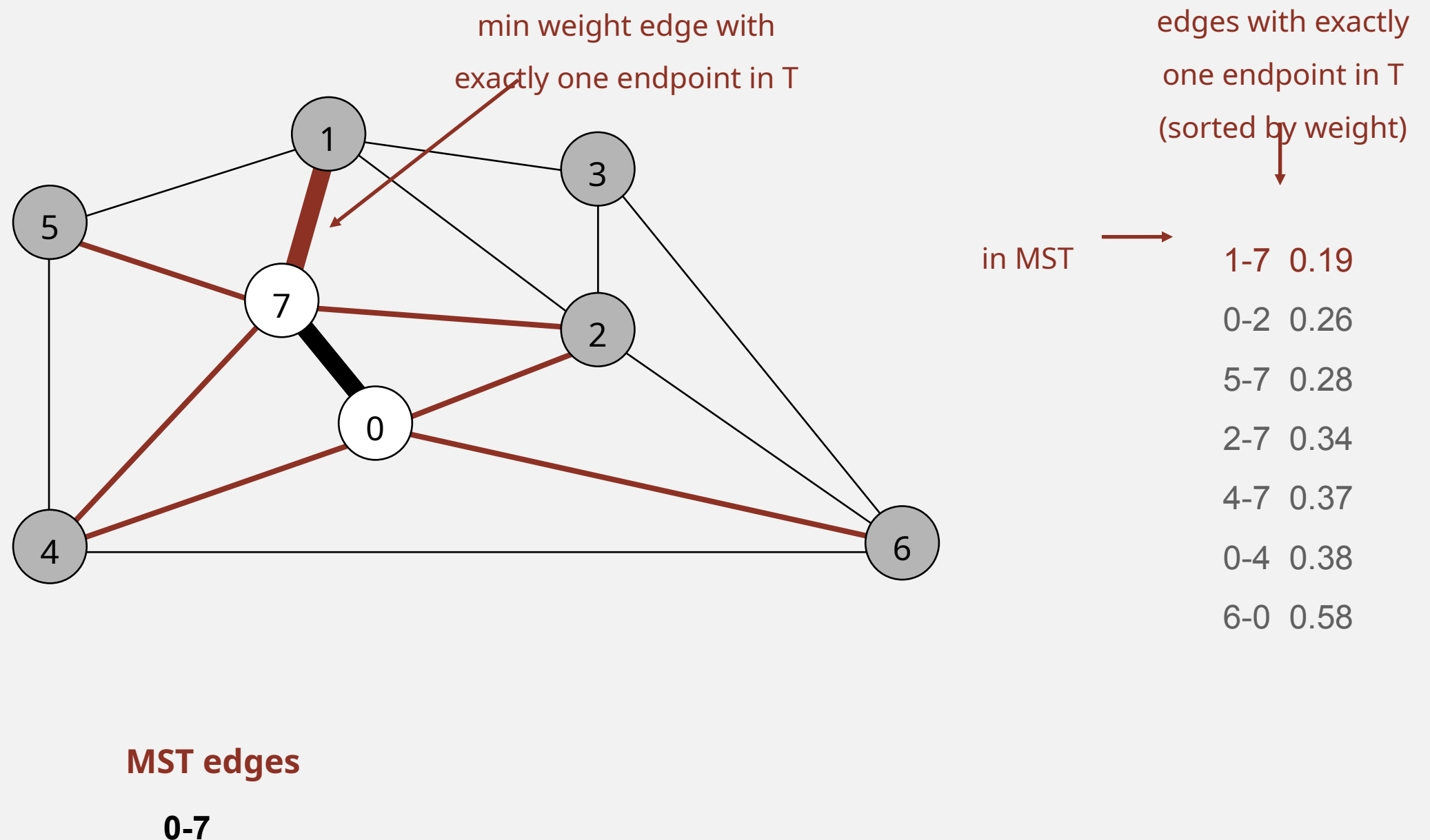


MST edges

0-7

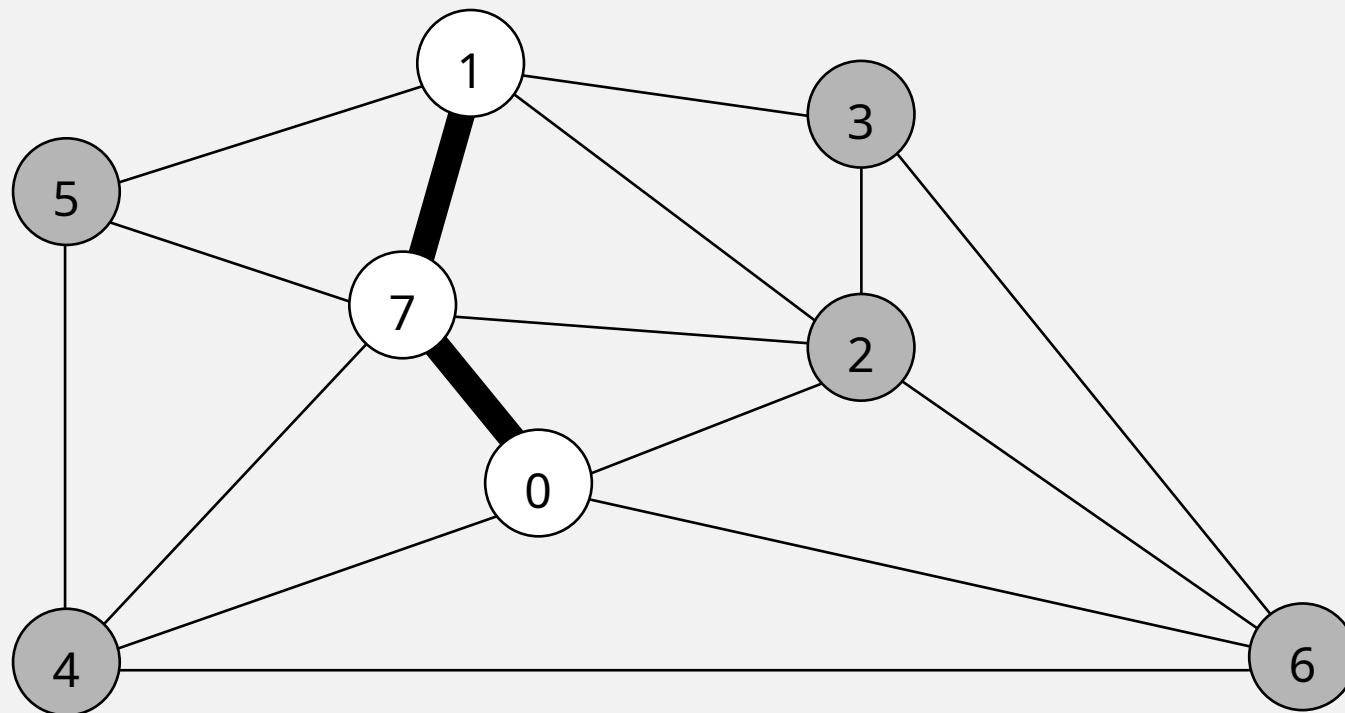
Prim's algorithm demo

- Start with vertex 0 and greedily grow tree T .
- Add to T the min weight edge with exactly one endpoint in T .
- Repeat until $V - 1$ edges.



Prim's algorithm demo

- Start with vertex 0 and greedily grow tree T .
- Add to T the min weight edge with exactly one endpoint in T .
- Repeat until $V - 1$ edges.

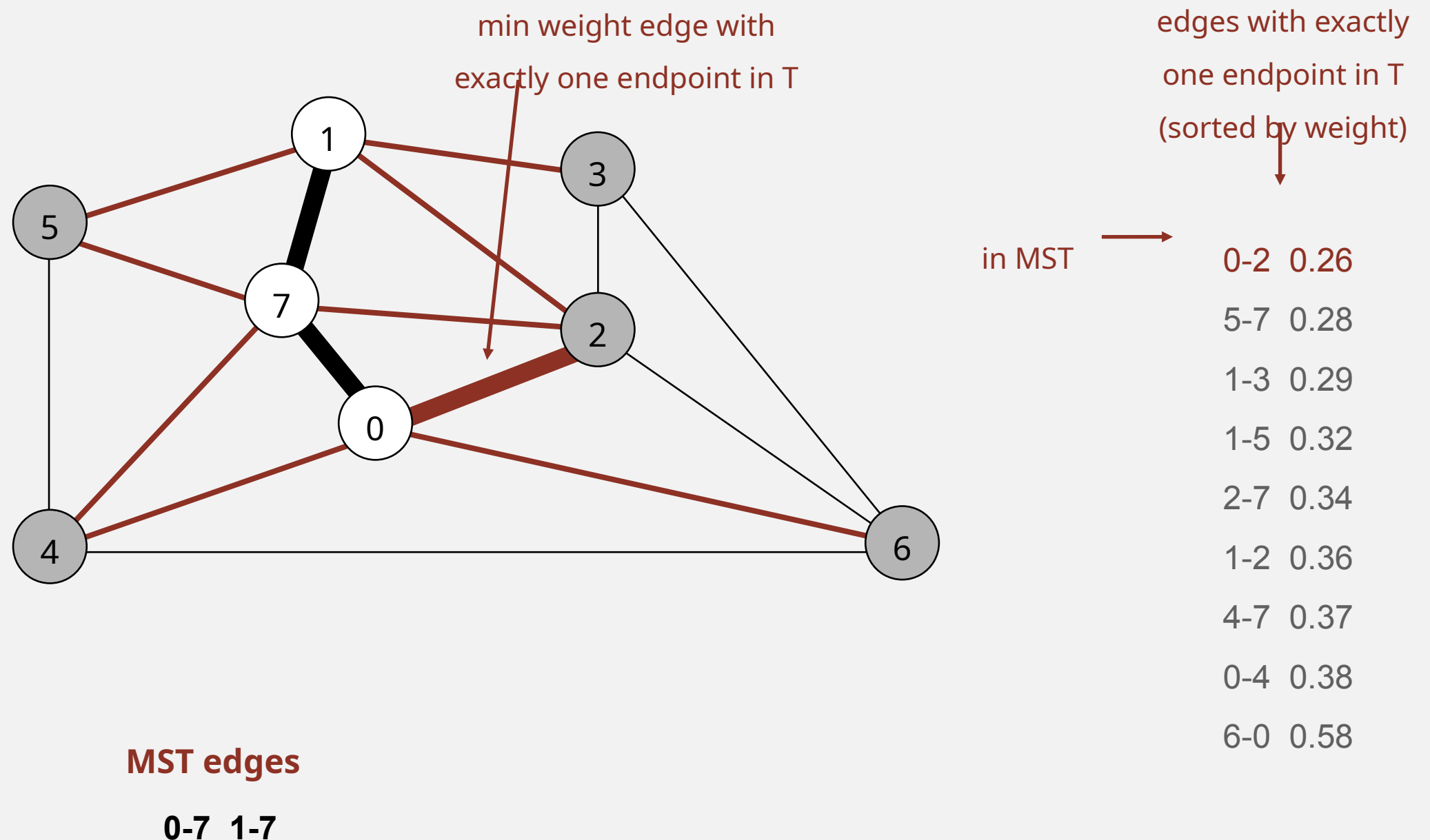


MST edges

0-7 1-7

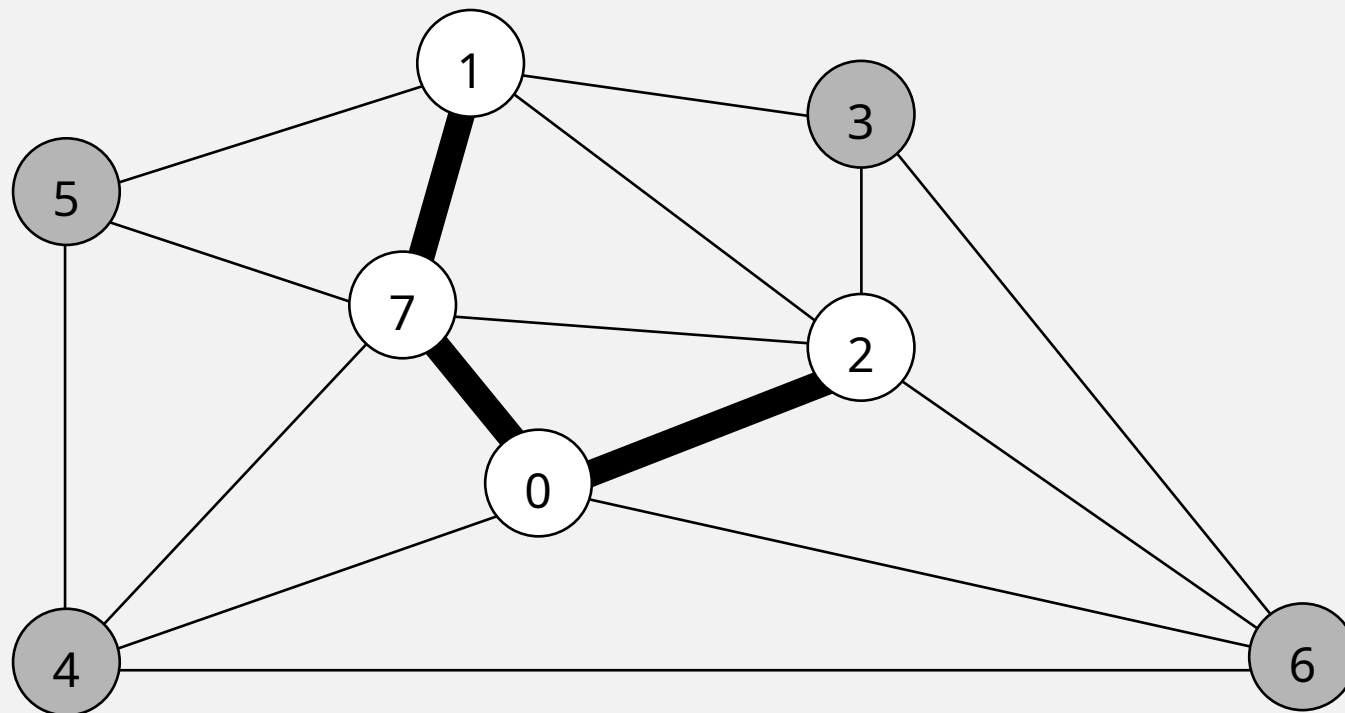
Prim's algorithm demo

- Start with vertex 0 and greedily grow tree T .
- Add to T the min weight edge with exactly one endpoint in T .
- Repeat until $V - 1$ edges.



Prim's algorithm demo

- Start with vertex 0 and greedily grow tree T .
- Add to T the min weight edge with exactly one endpoint in T .
- Repeat until $V - 1$ edges.

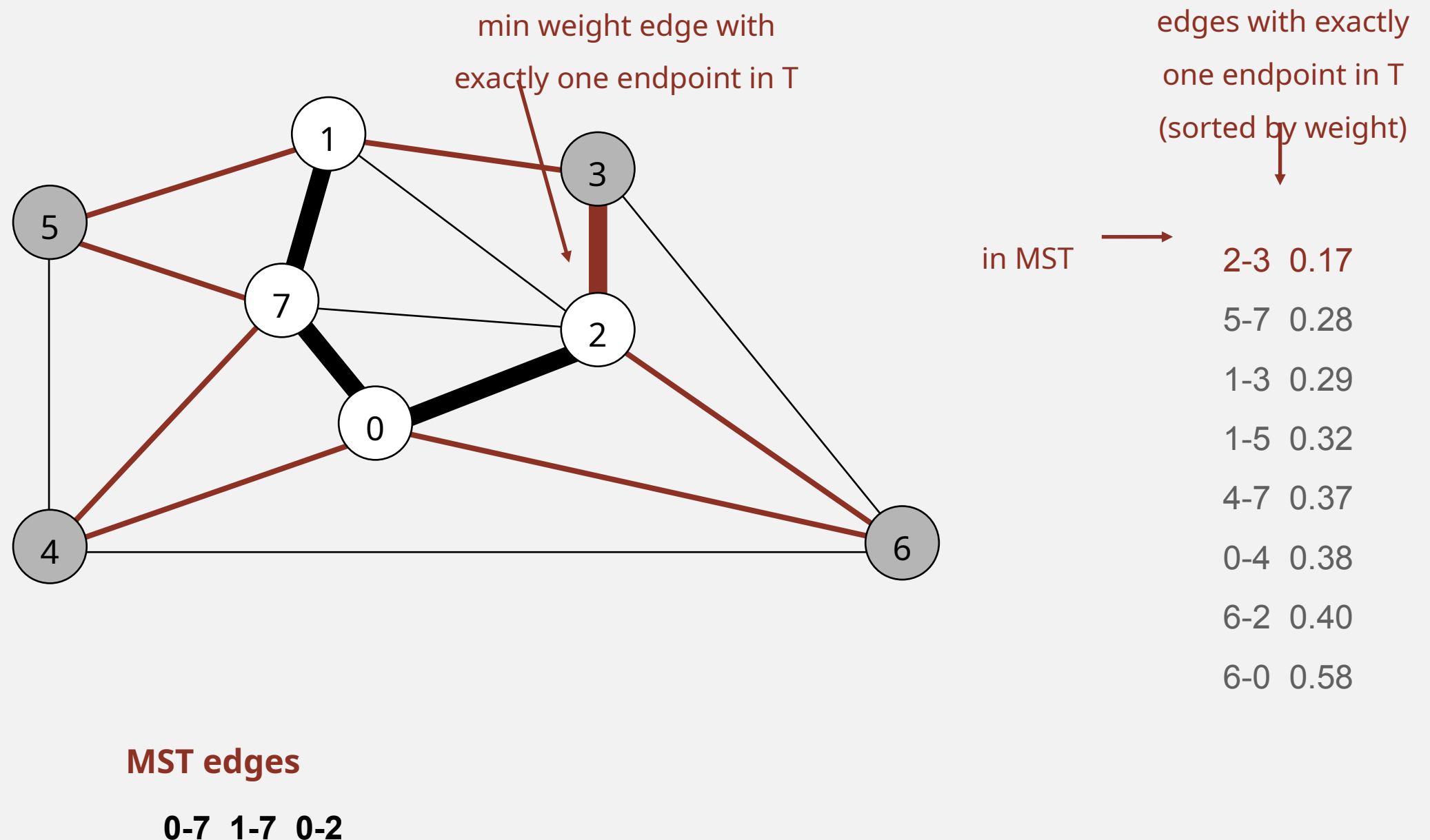


MST edges

0-7 1-7 0-2

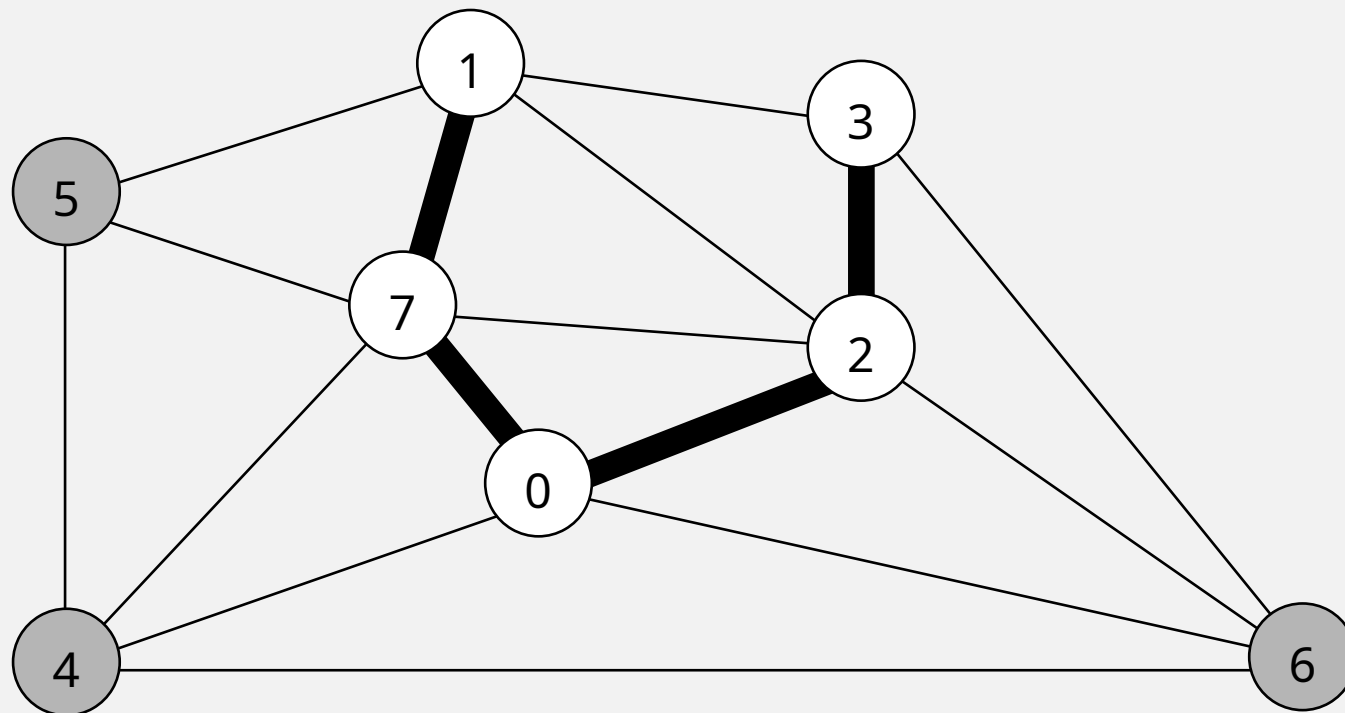
Prim's algorithm demo

- Start with vertex 0 and greedily grow tree T .
- Add to T the min weight edge with exactly one endpoint in T .
- Repeat until $V - 1$ edges.



Prim's algorithm demo

- Start with vertex 0 and greedily grow tree T .
- Add to T the min weight edge with exactly one endpoint in T .
- Repeat until $V - 1$ edges.

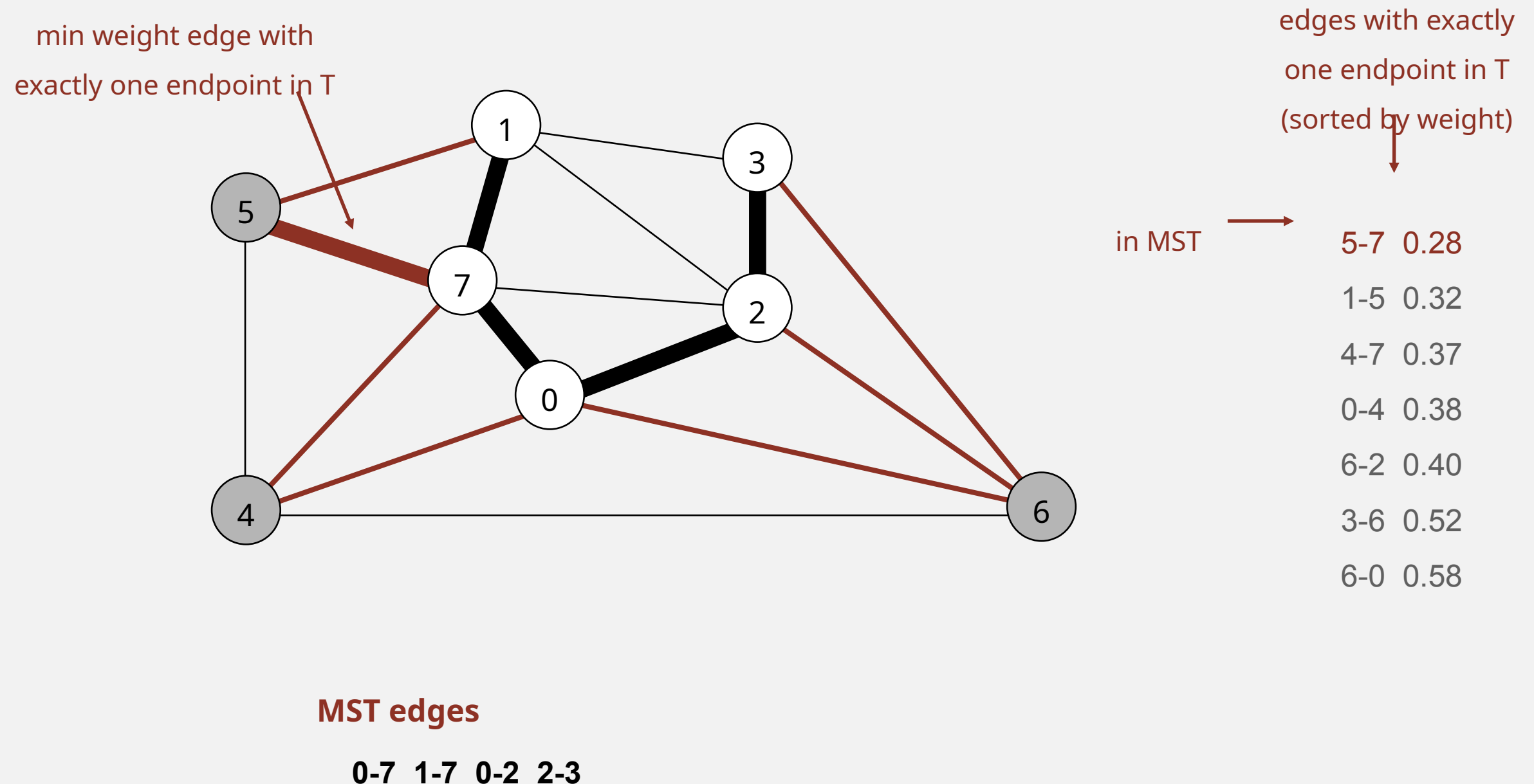


MST edges

0-7 1-7 0-2 2-3

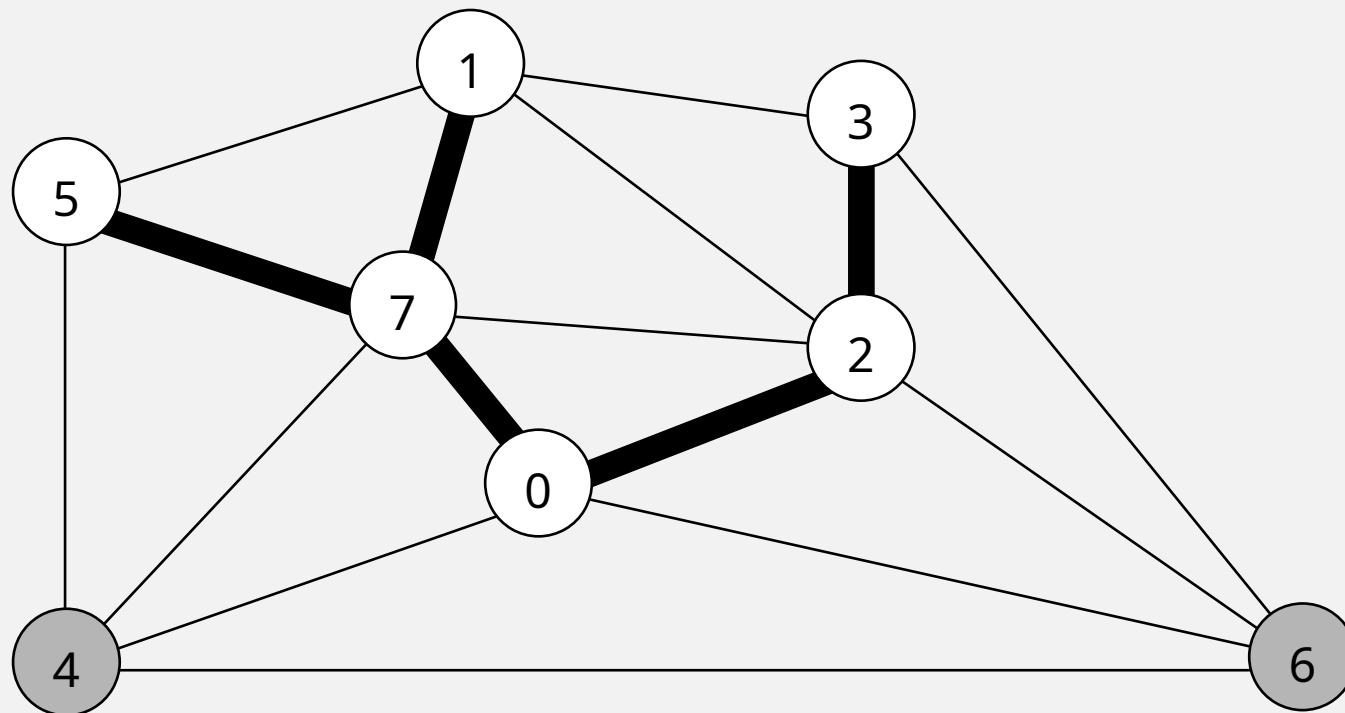
Prim's algorithm demo

- Start with vertex 0 and greedily grow tree T .
- Add to T the min weight edge with exactly one endpoint in T .
- Repeat until $V - 1$ edges.



Prim's algorithm demo

- Start with vertex 0 and greedily grow tree T .
- Add to T the min weight edge with exactly one endpoint in T .
- Repeat until $V - 1$ edges.

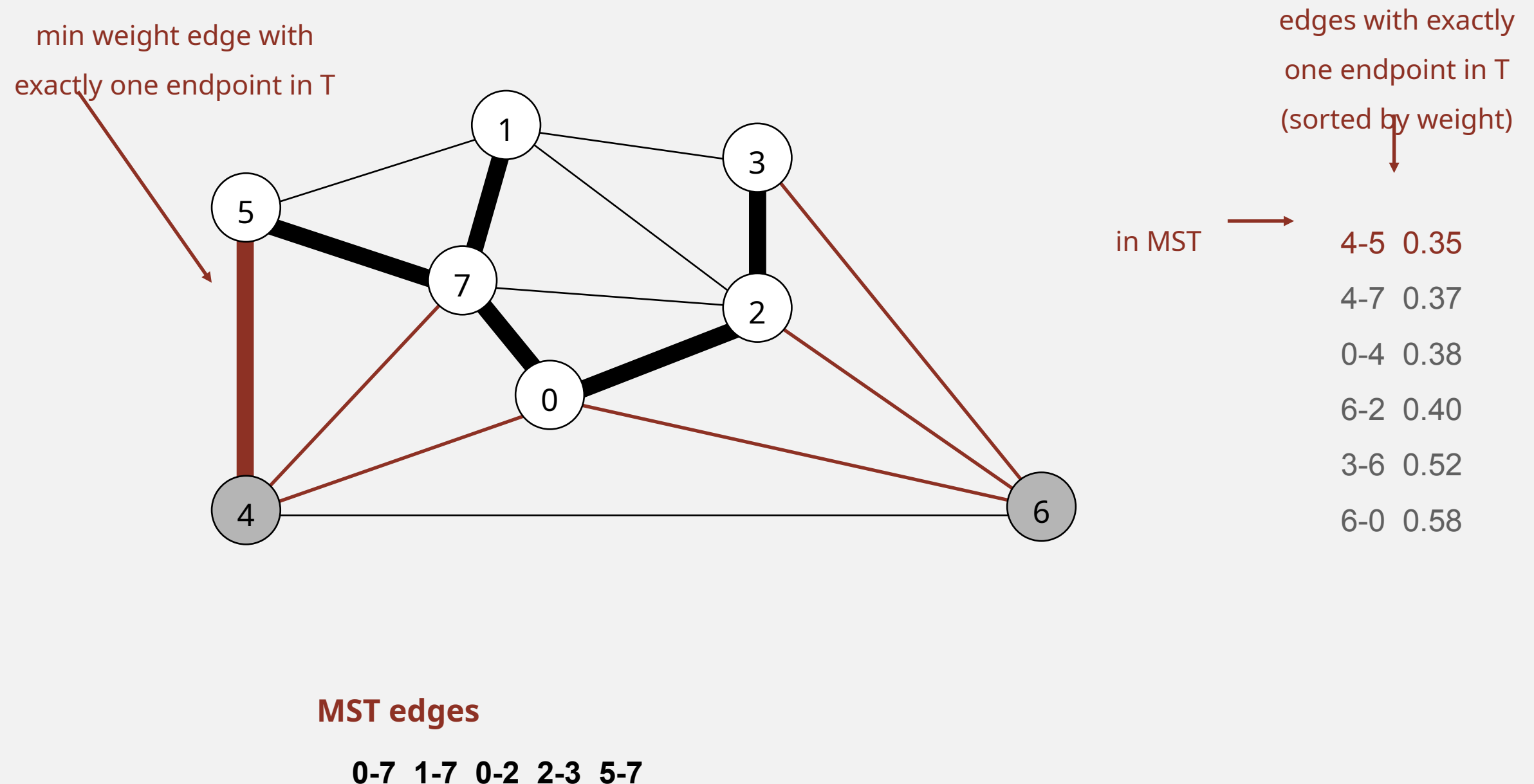


MST edges

0-7 1-7 0-2 2-3 5-7

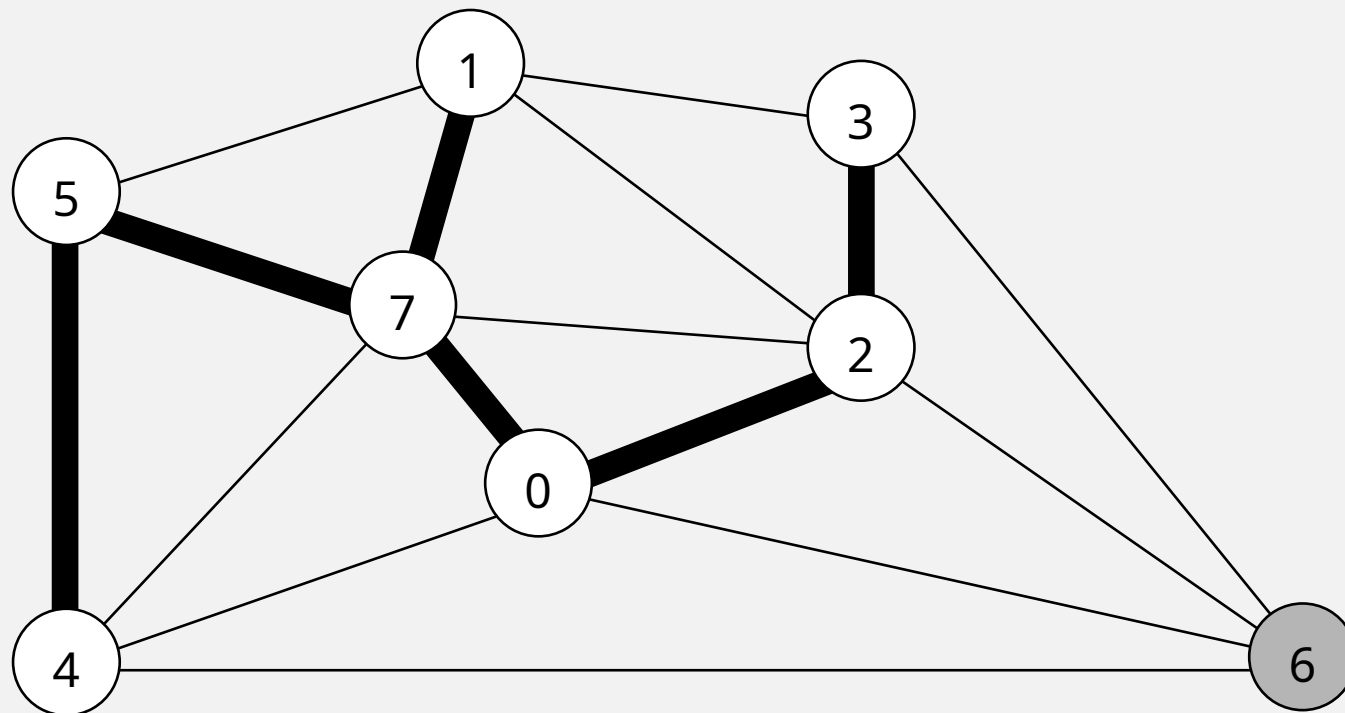
Prim's algorithm demo

- Start with vertex 0 and greedily grow tree T .
- Add to T the min weight edge with exactly one endpoint in T .
- Repeat until $V - 1$ edges.



Prim's algorithm demo

- Start with vertex 0 and greedily grow tree T .
- Add to T the min weight edge with exactly one endpoint in T .
- Repeat until $V - 1$ edges.

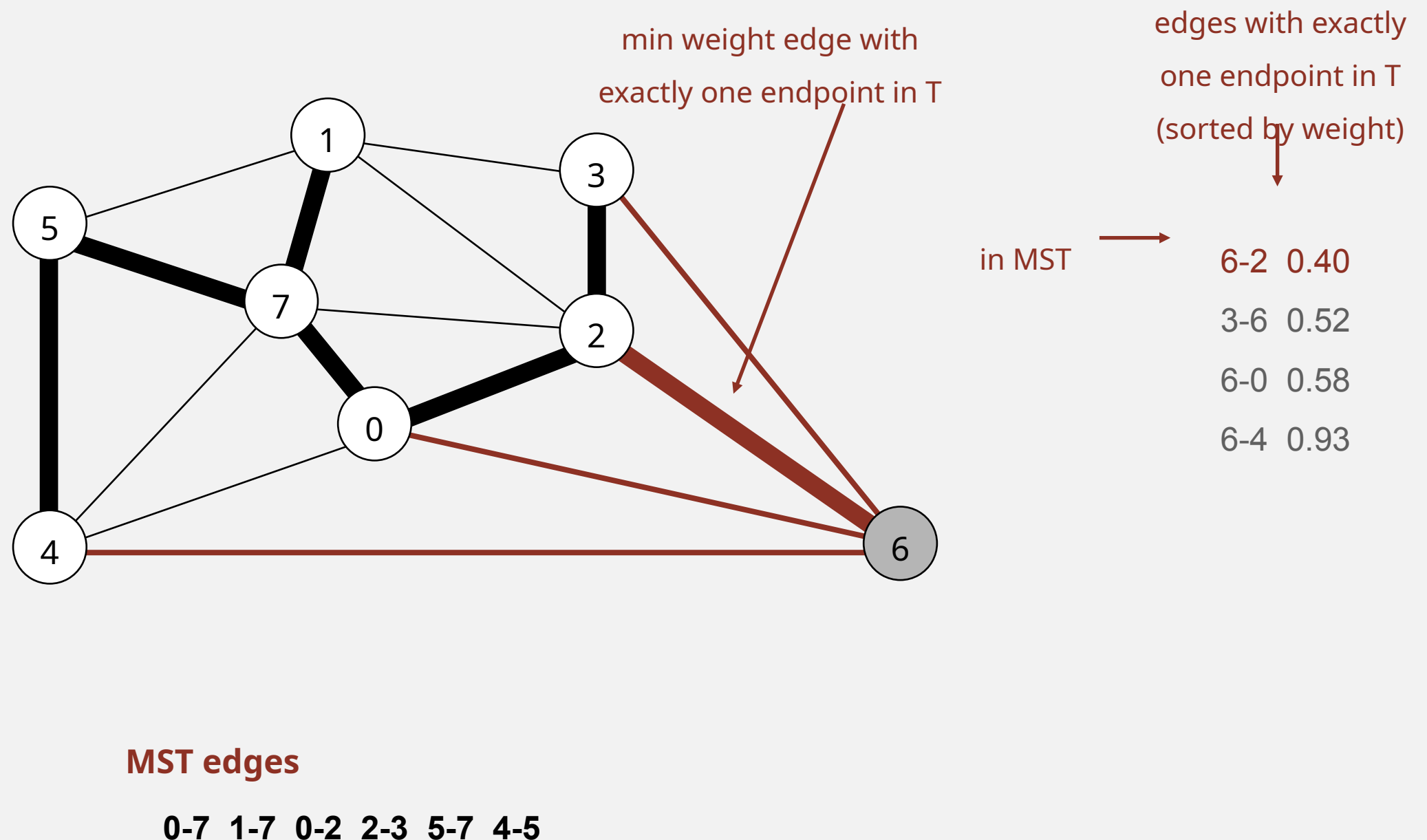


MST edges

0-7 1-7 0-2 2-3 5-7 4-5

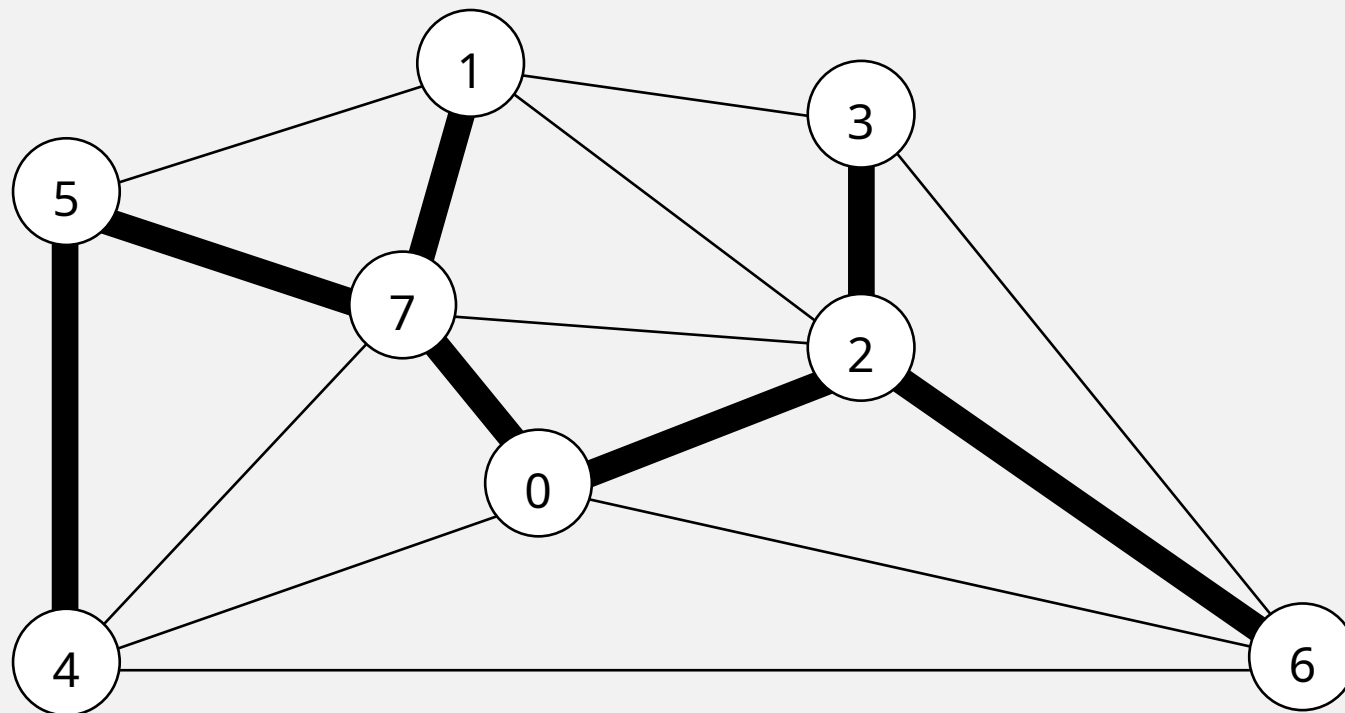
Prim's algorithm demo

- Start with vertex 0 and greedily grow tree T .
- Add to T the min weight edge with exactly one endpoint in T .
- Repeat until $V - 1$ edges.



Prim's algorithm demo

- Start with vertex 0 and greedily grow tree T .
- Add to T the min weight edge with exactly one endpoint in T .
- Repeat until $V - 1$ edges.



MST edges

0-7 1-7 0-2 2-3 5-7 4-5 6-2



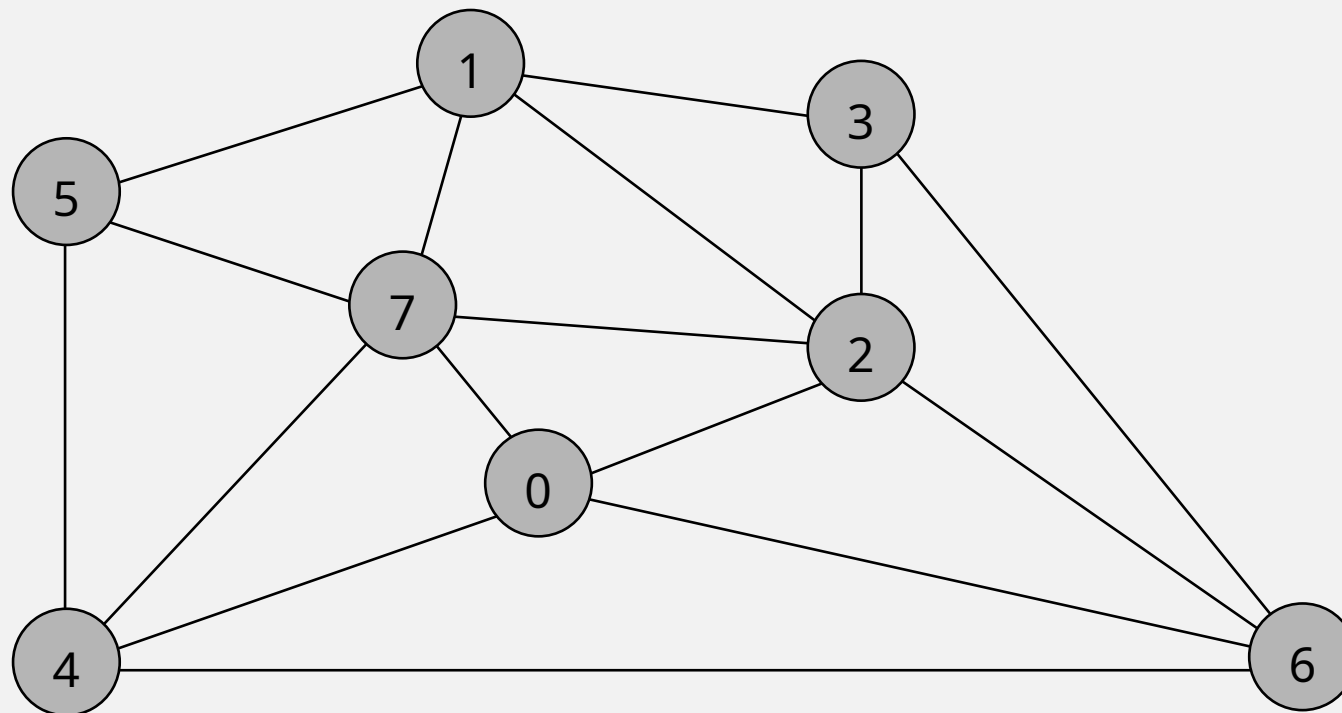
<http://algs4.cs.princeton.edu>

PRIM'S ALGORITHM DEMO

- *Prim's algorithm*
- *lazy implementation*
- *eager implementation*

Prim's algorithm: lazy implementation demo

- Start with vertex 0 and greedily grow tree T .
- Add to T the min weight edge with exactly one endpoint in T .
- Repeat until $V - 1$ edges.



an edge-weighted graph

0-7 0.16

2-3 0.17

1-7 0.19

0-2 0.26

5-7 0.28

1-3 0.29

1-5 0.32

2-7 0.34

4-5 0.35

1-2 0.36

4-7 0.37

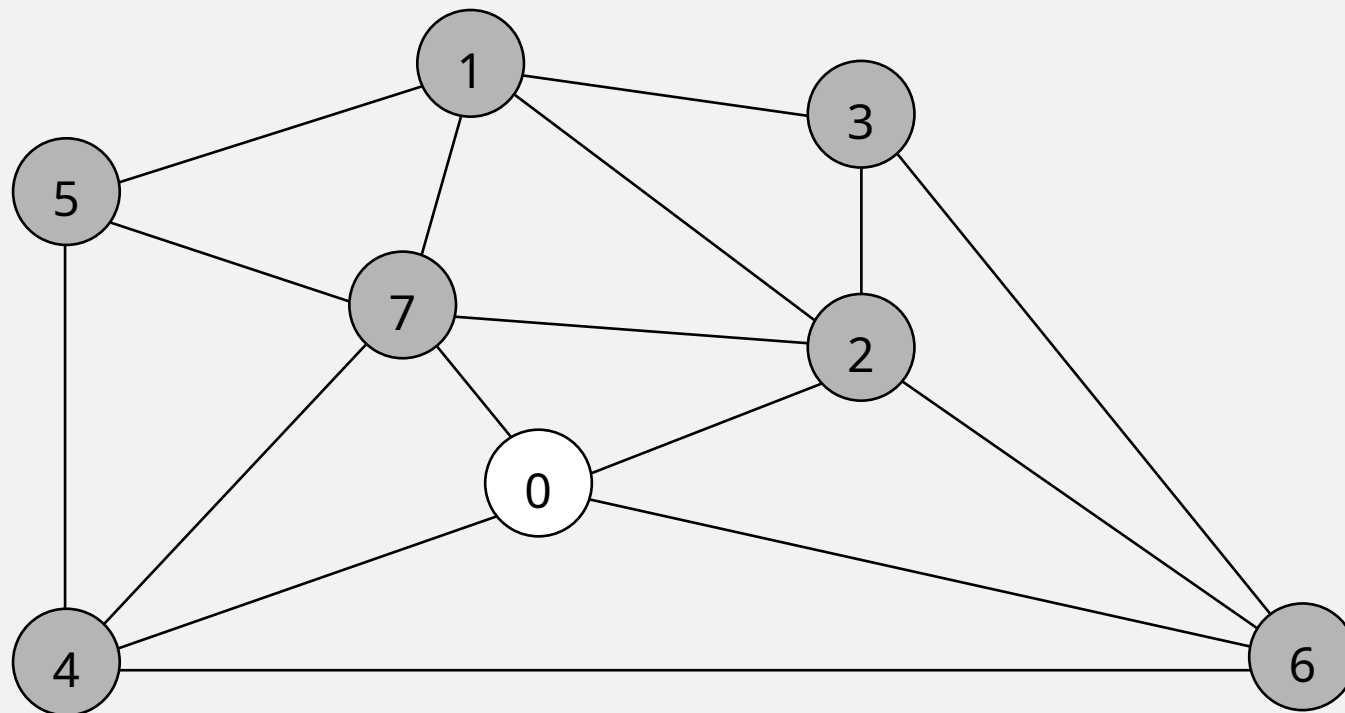
0-4 0.38

6-2 0.40

3-6 0.52

Prim's algorithm: lazy implementation demo

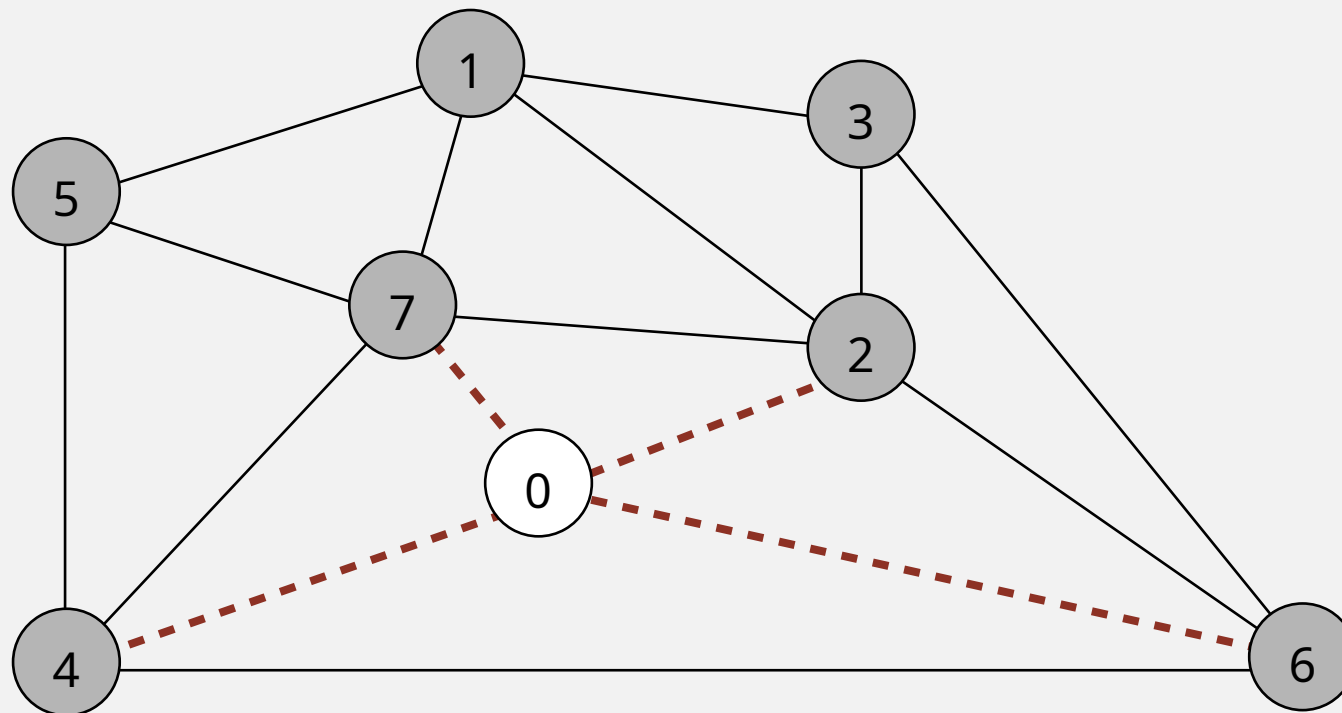
- Start with vertex 0 and greedily grow tree T .
- Add to T the min weight edge with exactly one endpoint in T .
- Repeat until $V - 1$ edges.



Prim's algorithm: lazy implementation demo

- Start with vertex 0 and greedily grow tree T .
- Add to T the min weight edge with exactly one endpoint in T .
- Repeat until $V - 1$ edges.

add to PQ all edges incident to 0



edges on PQ
(sorted by weight)

* 0-7 0.16

* 0-2 0.26

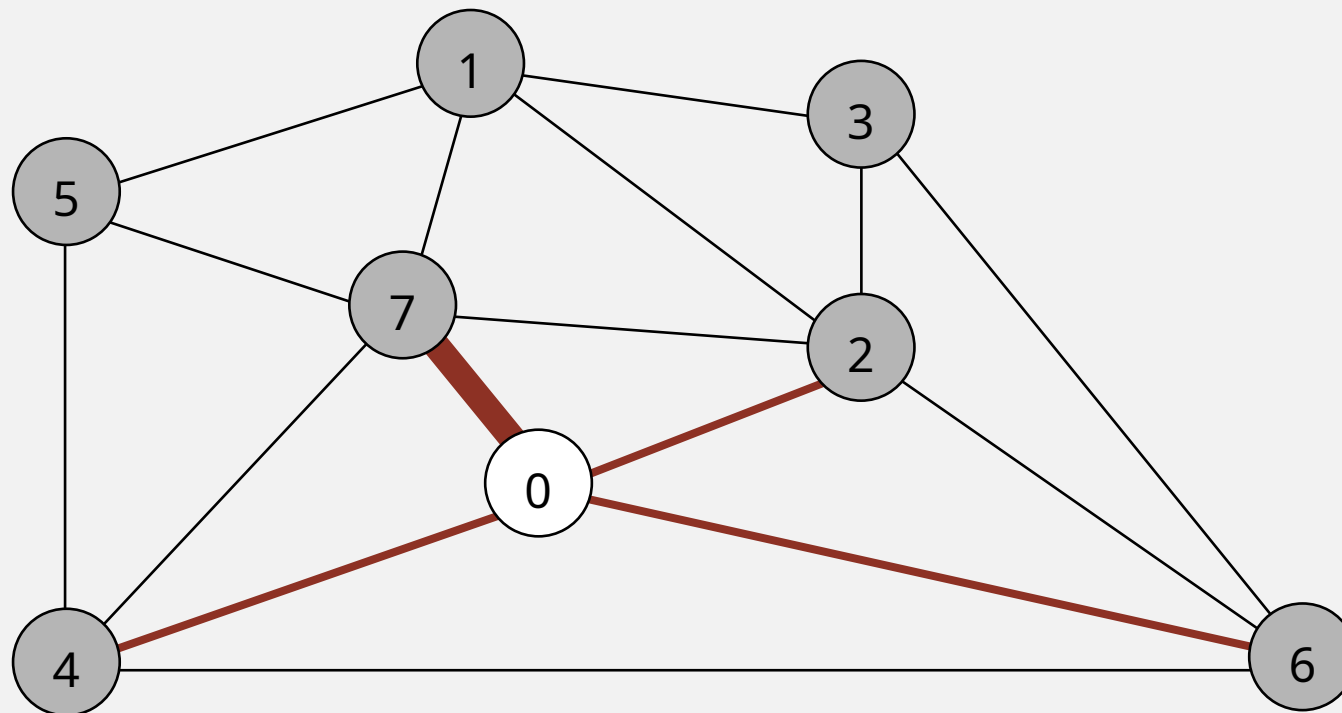
* 0-4 0.38

* 6-0 0.58

Prim's algorithm: lazy implementation demo

- Start with vertex 0 and greedily grow tree T .
- Add to T the min weight edge with exactly one endpoint in T .
- Repeat until $V - 1$ edges.

delete 0-7 and add to MST



edges on PQ
(sorted by weight)

0-7 0.16

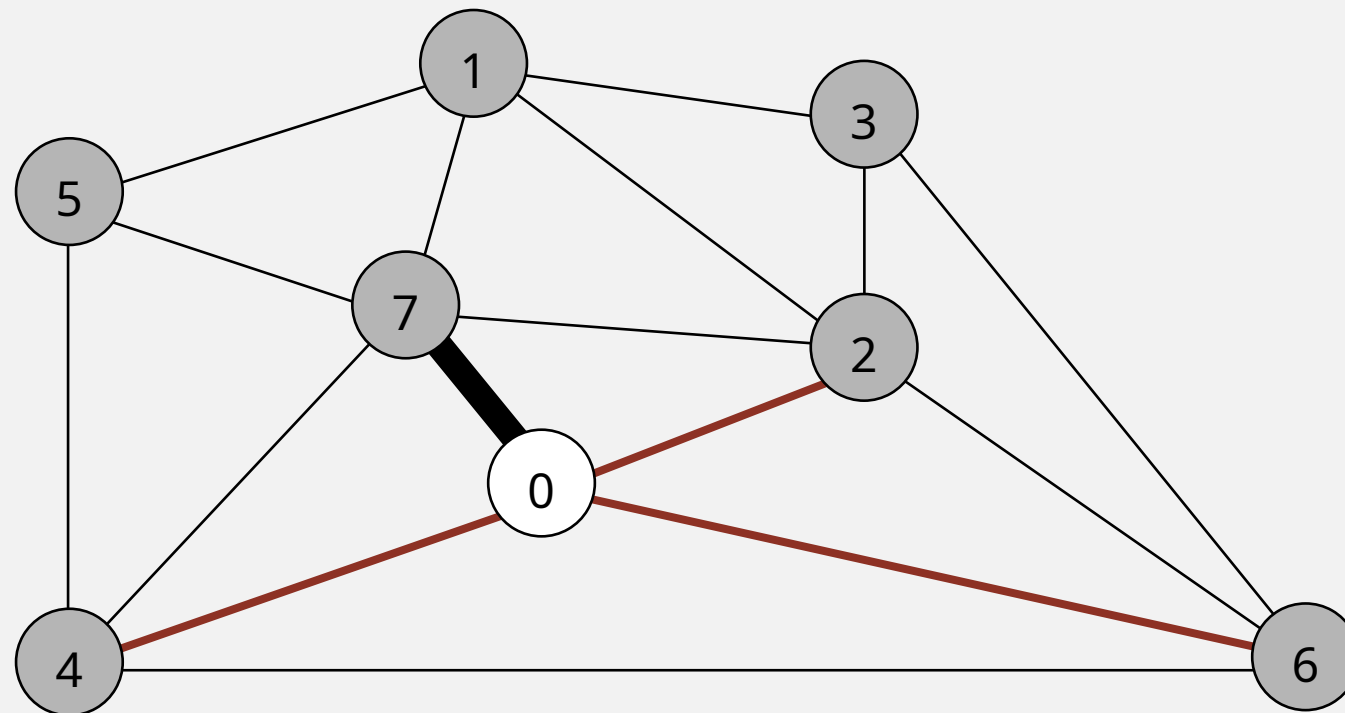
0-2 0.26

0-4 0.38

6-0 0.58

Prim's algorithm: lazy implementation demo

- Start with vertex 0 and greedily grow tree T .
- Add to T the min weight edge with exactly one endpoint in T .
- Repeat until $V - 1$ edges.



edges on PQ
(sorted by weight)

0-2 0.26

0-4 0.38

6-0 0.58

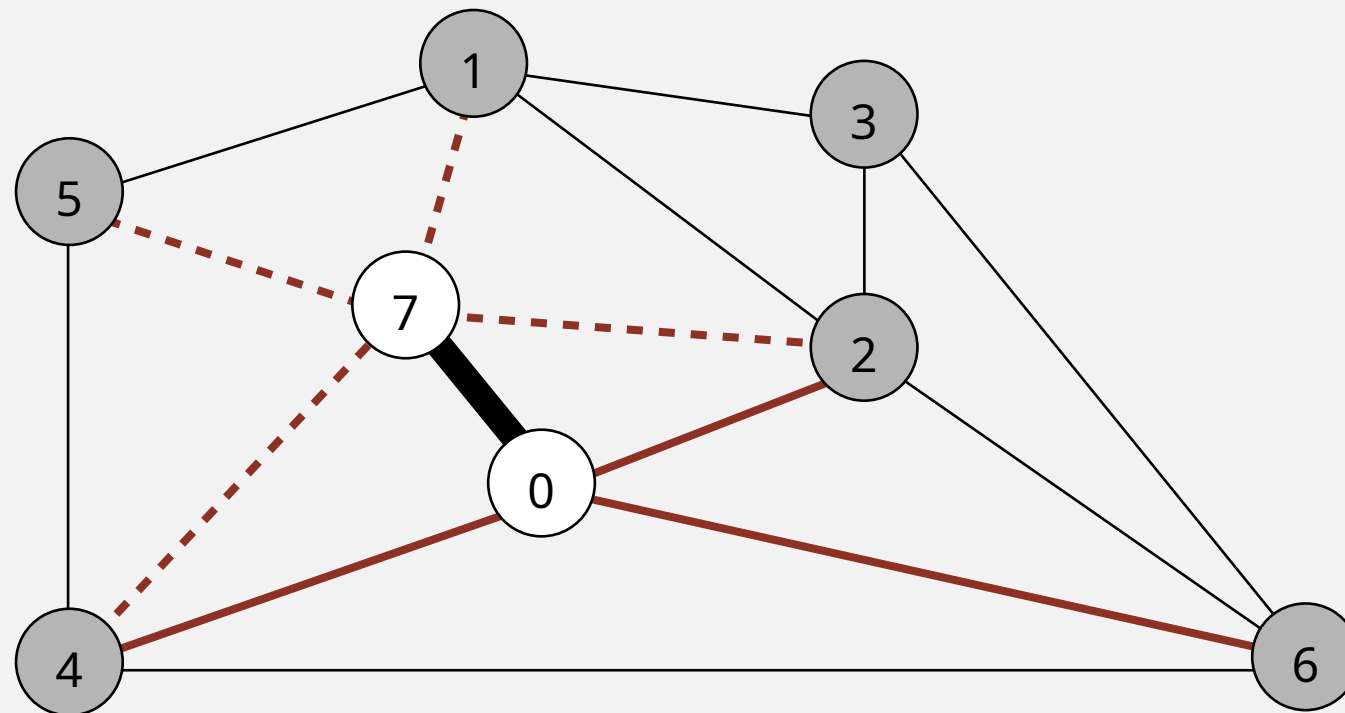
MST edges

0-7

Prim's algorithm: lazy implementation demo

- Start with vertex 0 and greedily grow tree T .
- Add to T the min weight edge with exactly one endpoint in T .
- Repeat until $V - 1$ edges.

add to PQ all edges incident to 7



edges on PQ
(sorted by weight)

* 1-7 0.19

0-2 0.26

* 5-7 0.28

* 2-7 0.34

* 4-7 0.37

0-4 0.38

6-0 0.58

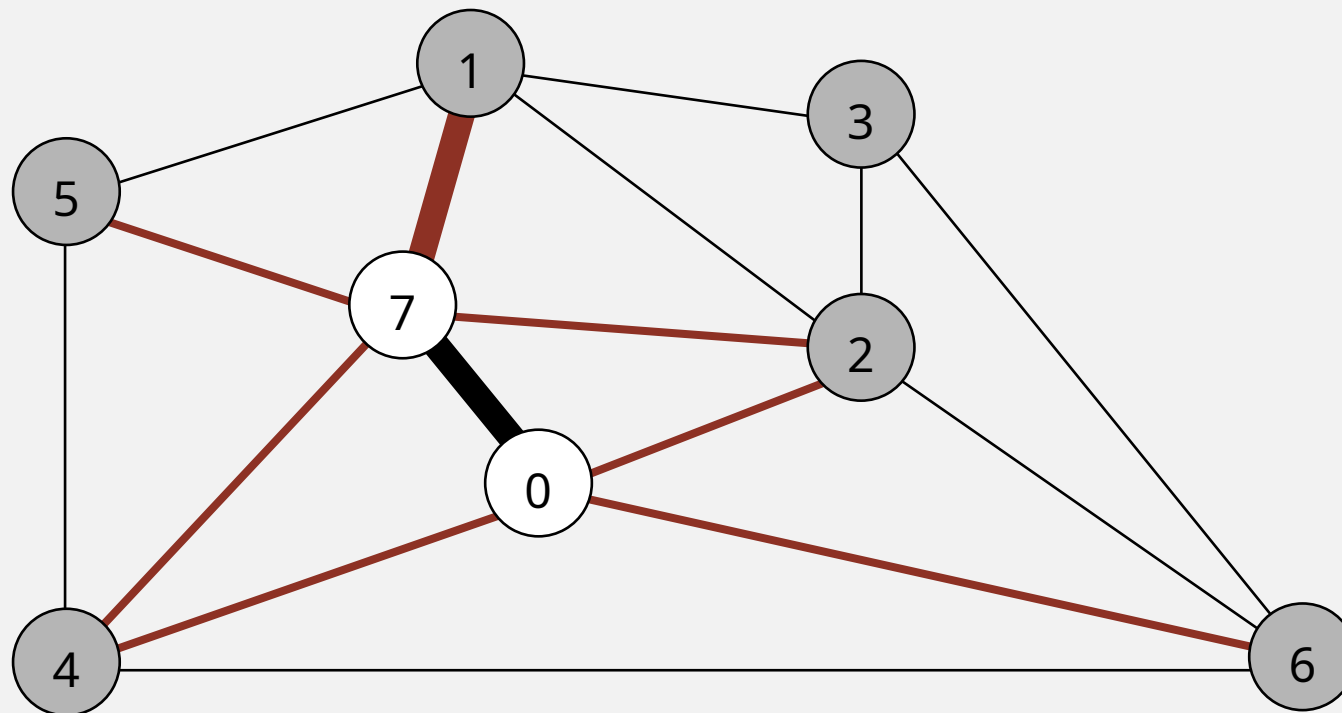
MST edges

0-7

Prim's algorithm: lazy implementation demo

- Start with vertex 0 and greedily grow tree T .
- Add to T the min weight edge with exactly one endpoint in T .
- Repeat until $V - 1$ edges.

delete 1-7 and add to MST



edges on PQ
(sorted by weight)

1-7 0.19

0-2 0.26

5-7 0.28

2-7 0.34

4-7 0.37

0-4 0.38

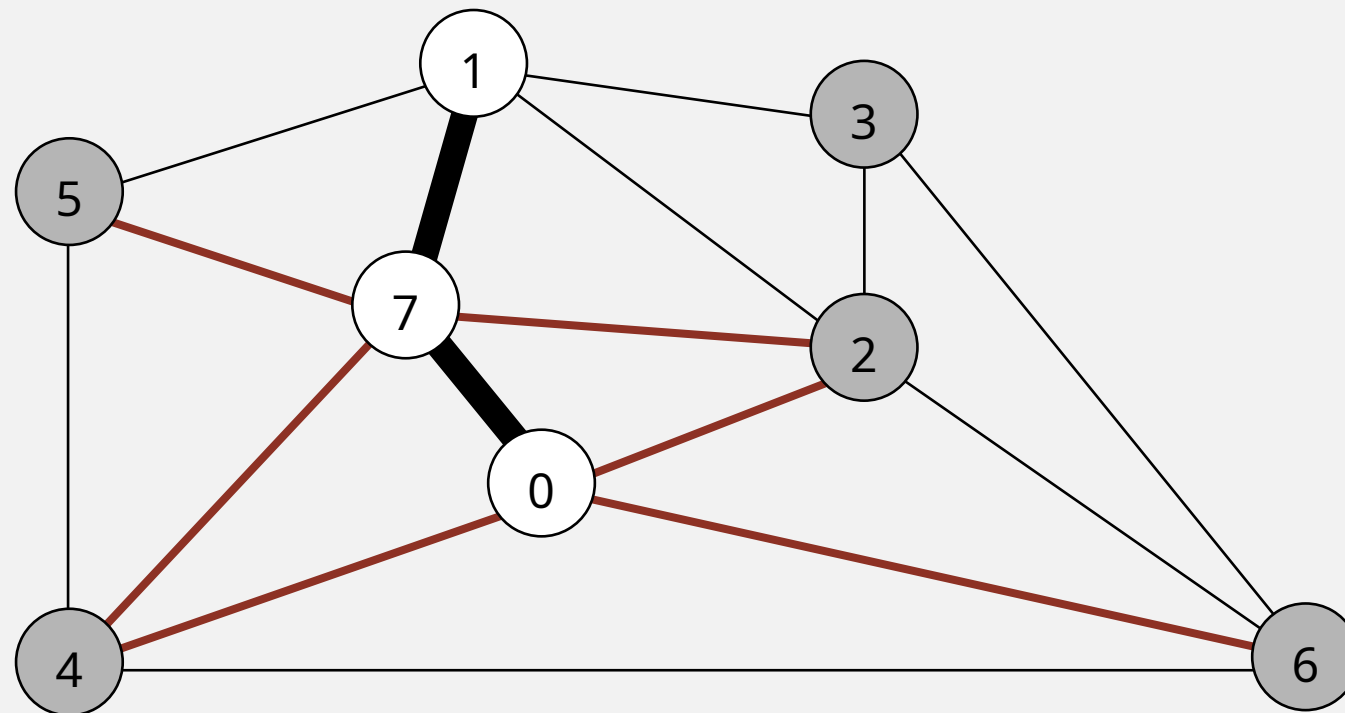
6-0 0.58

MST edges

0-7

Prim's algorithm: lazy implementation demo

- Start with vertex 0 and greedily grow tree T .
- Add to T the min weight edge with exactly one endpoint in T .
- Repeat until $V - 1$ edges.



edges on PQ
(sorted by weight)

0-2 0.26

5-7 0.28

2-7 0.34

4-7 0.37

0-4 0.38

6-0 0.58

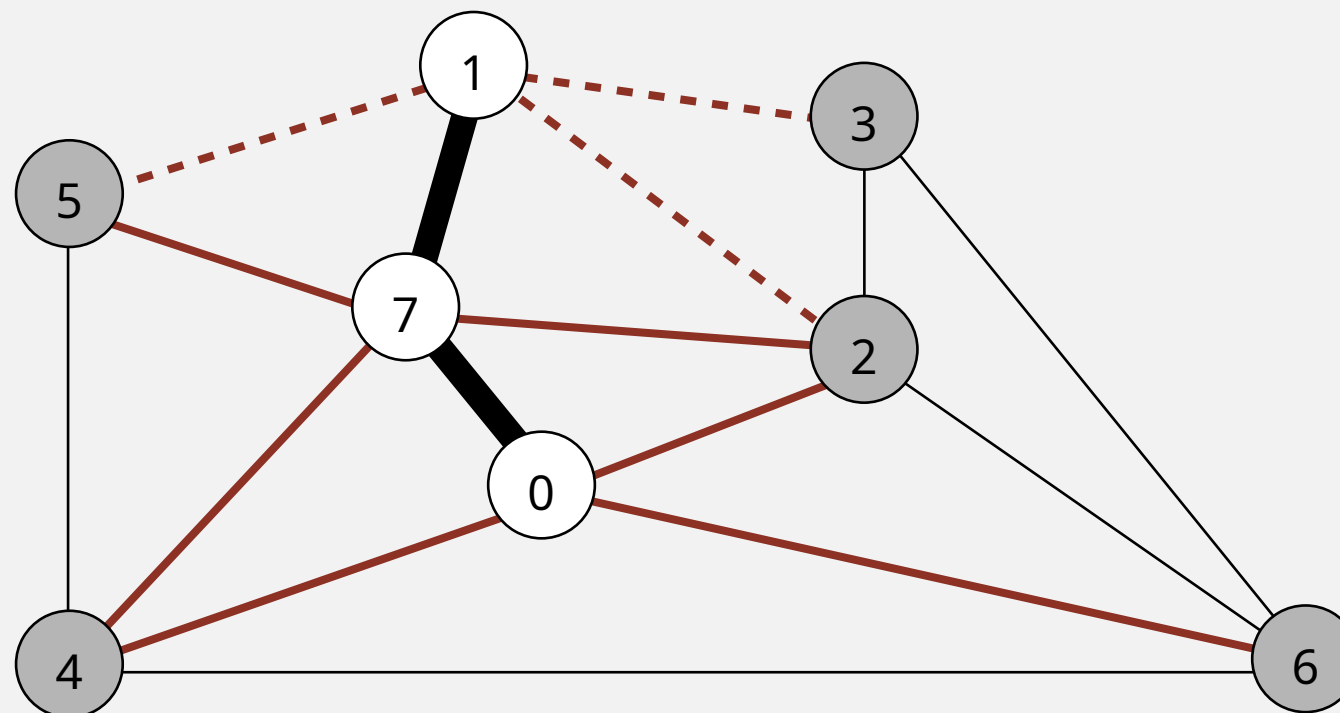
MST edges

0-7 1-7

Prim's algorithm: lazy implementation demo

- Start with vertex 0 and greedily grow tree T .
- Add to T the min weight edge with exactly one endpoint in T .
- Repeat until $V - 1$ edges.

add to PQ all edges incident to 1



MST edges

0-7 1-7

edges on PQ
(sorted by weight)

0-2 0.26

5-7 0.28

* 1-3 0.29

* 1-5 0.32

2-7 0.34

* 1-2 0.36

4-7 0.37

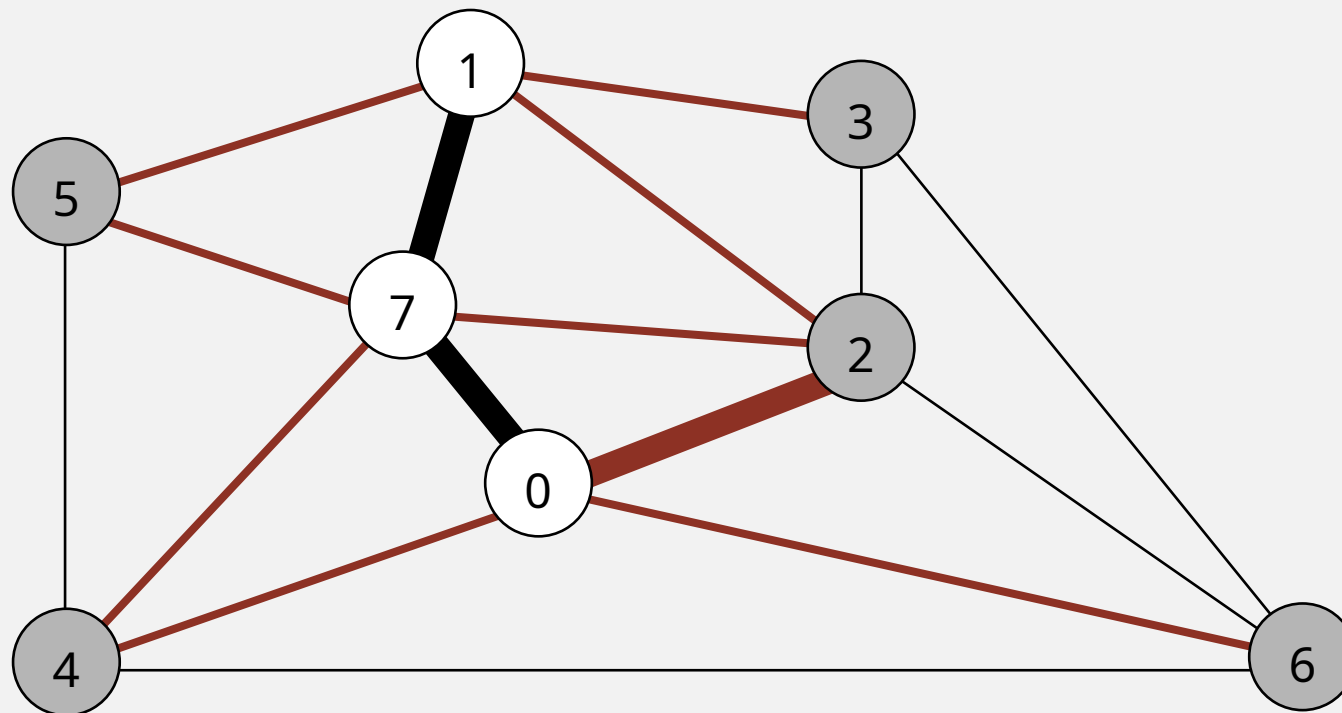
0-4 0.38

6-0 0.58

Prim's algorithm: lazy implementation demo

- Start with vertex 0 and greedily grow tree T .
- Add to T the min weight edge with exactly one endpoint in T .
- Repeat until $V - 1$ edges.

delete edge 0-2 and add to MST



MST edges

0-7 1-7

edges on PQ
(sorted by weight)

0-2 0.26

5-7 0.28

1-3 0.29

1-5 0.32

2-7 0.34

1-2 0.36

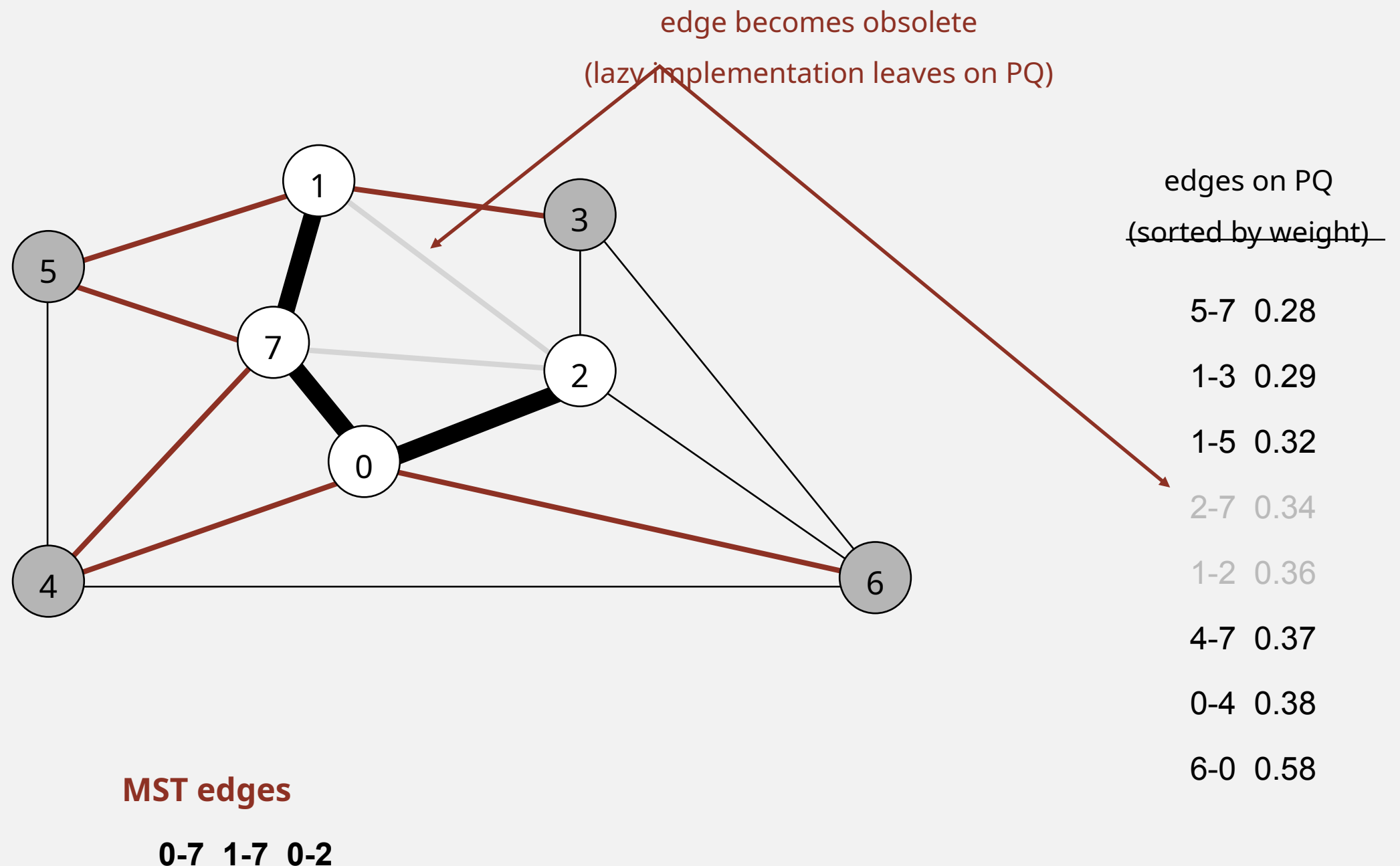
4-7 0.37

0-4 0.38

6-0 0.58

Prim's algorithm: lazy implementation demo

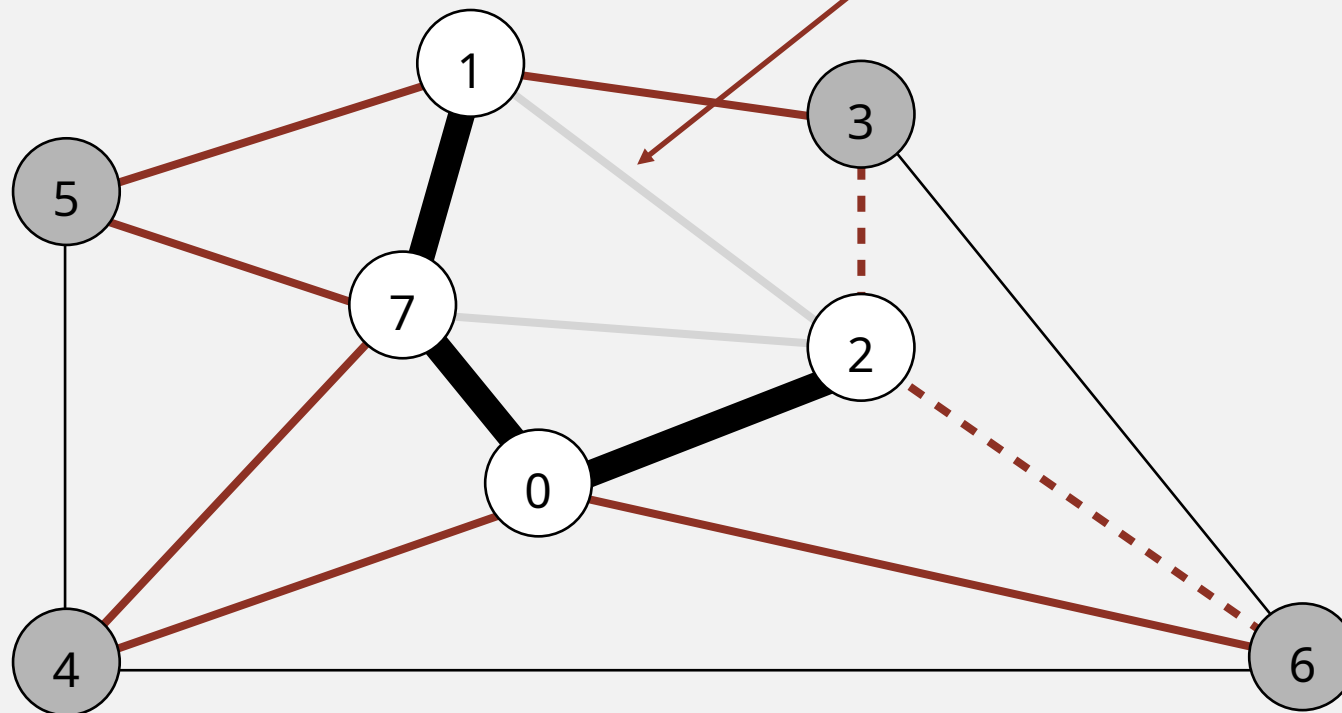
- Start with vertex 0 and greedily grow tree T .
- Add to T the min weight edge with exactly one endpoint in T .
- Repeat until $V - 1$ edges.



Prim's algorithm: lazy implementation demo

- Start with vertex 0 and greedily grow tree T .
- Add to T the min weight edge with exactly one endpoint in T .
- Repeat until $V - 1$ edges.

add to PQ all edges incident to 2



MST edges

0-7 1-7 0-2

edges on PQ
(sorted by weight)

* 2-3 0.17

5-7 0.28

1-3 0.29

1-5 0.32

2-7 0.34

1-2 0.36

4-7 0.37

0-4 0.38

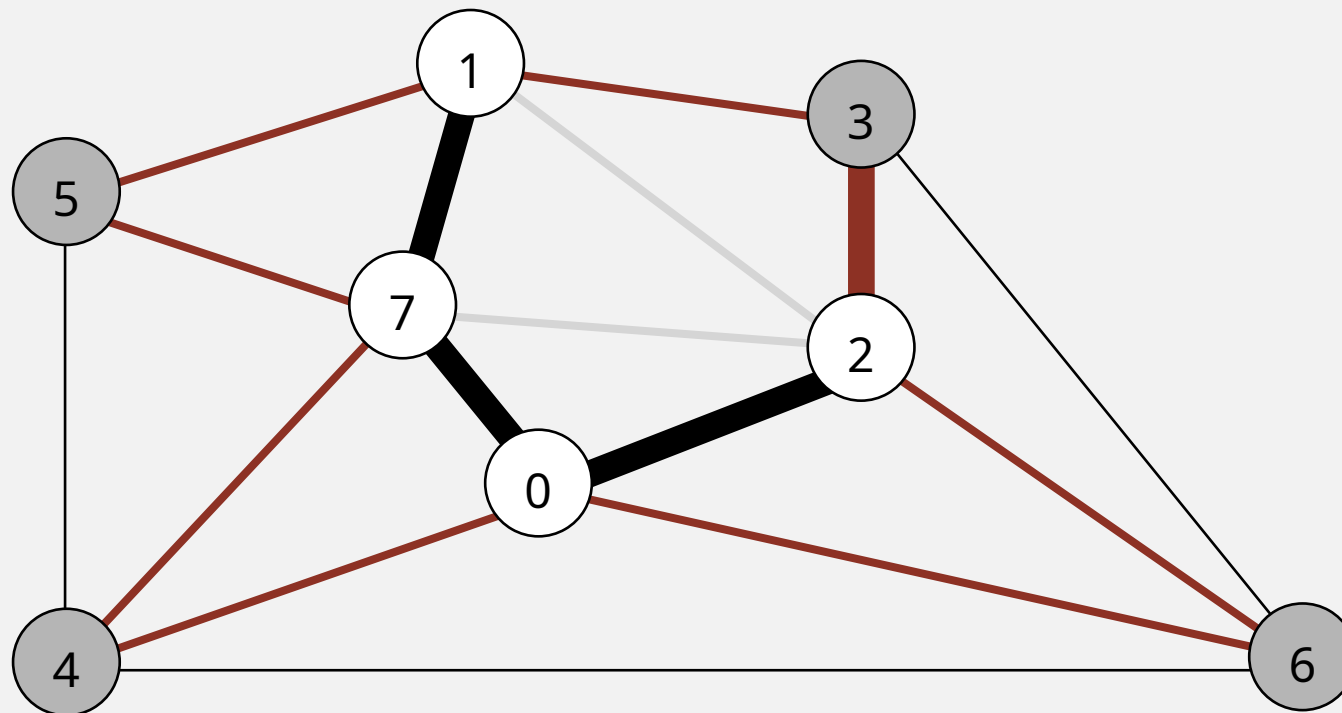
* 6-2 0.40

6-0 0.58

Prim's algorithm: lazy implementation demo

- Start with vertex 0 and greedily grow tree T .
- Add to T the min weight edge with exactly one endpoint in T .
- Repeat until $V - 1$ edges.

delete 2-3 and add to MST



MST edges

0-7 1-7 0-2

edges on PQ
(sorted by weight)

* 2-3 0.17

5-7 0.28

1-3 0.29

1-5 0.32

2-7 0.34

1-2 0.36

4-7 0.37

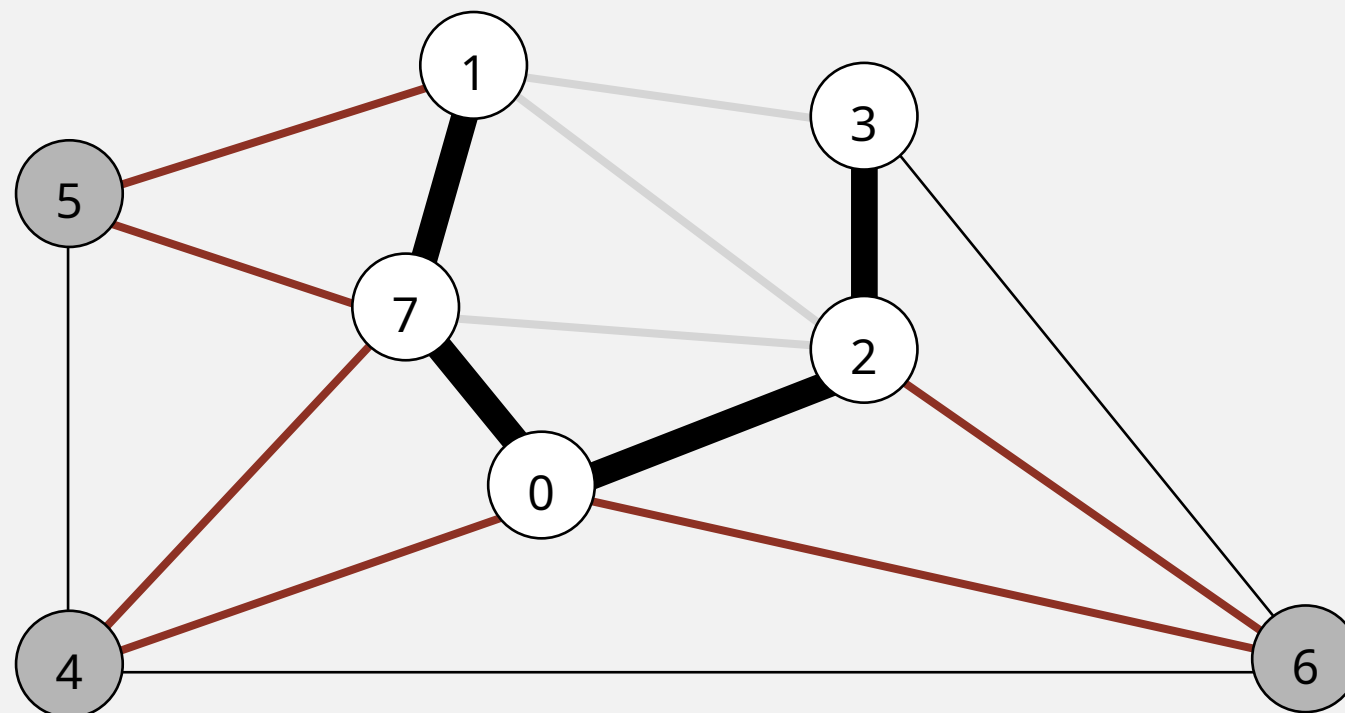
0-4 0.38

* 6-2 0.40

6-0 0.58

Prim's algorithm: lazy implementation demo

- Start with vertex 0 and greedily grow tree T .
- Add to T the min weight edge with exactly one endpoint in T .
- Repeat until $V - 1$ edges.



MST edges

0-7 1-7 0-2 2-3

edges on PQ
(sorted by weight)

5-7 0.28

1-3 0.29

1-5 0.32

2-7 0.34

1-2 0.36

4-7 0.37

0-4 0.38

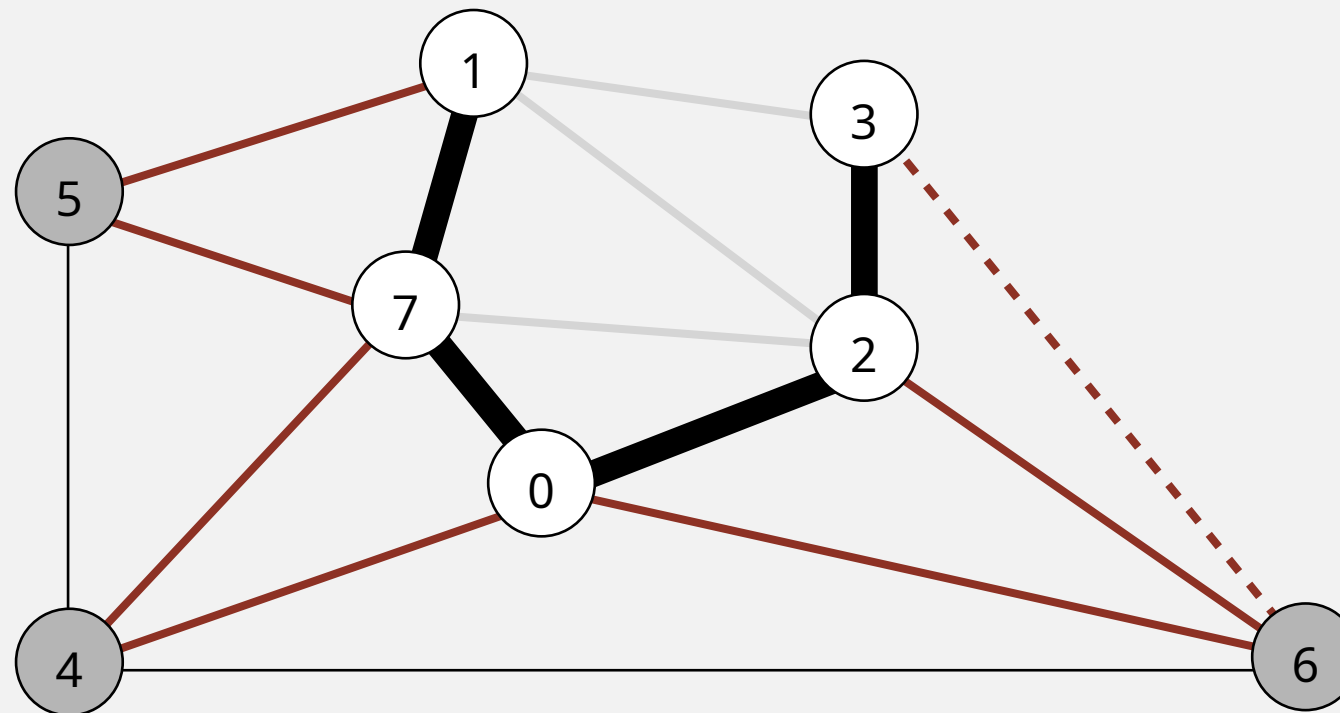
6-2 0.40

6-0 0.58

Prim's algorithm: lazy implementation demo

- Start with vertex 0 and greedily grow tree T .
- Add to T the min weight edge with exactly one endpoint in T .
- Repeat until $V - 1$ edges.

add to PQ all edges incident to 3



MST edges

0-7 1-7 0-2 2-3

edges on PQ
(sorted by weight)

5-7 0.28

1-3 0.29

1-5 0.32

2-7 0.34

1-2 0.36

4-7 0.37

0-4 0.38

6-2 0.40

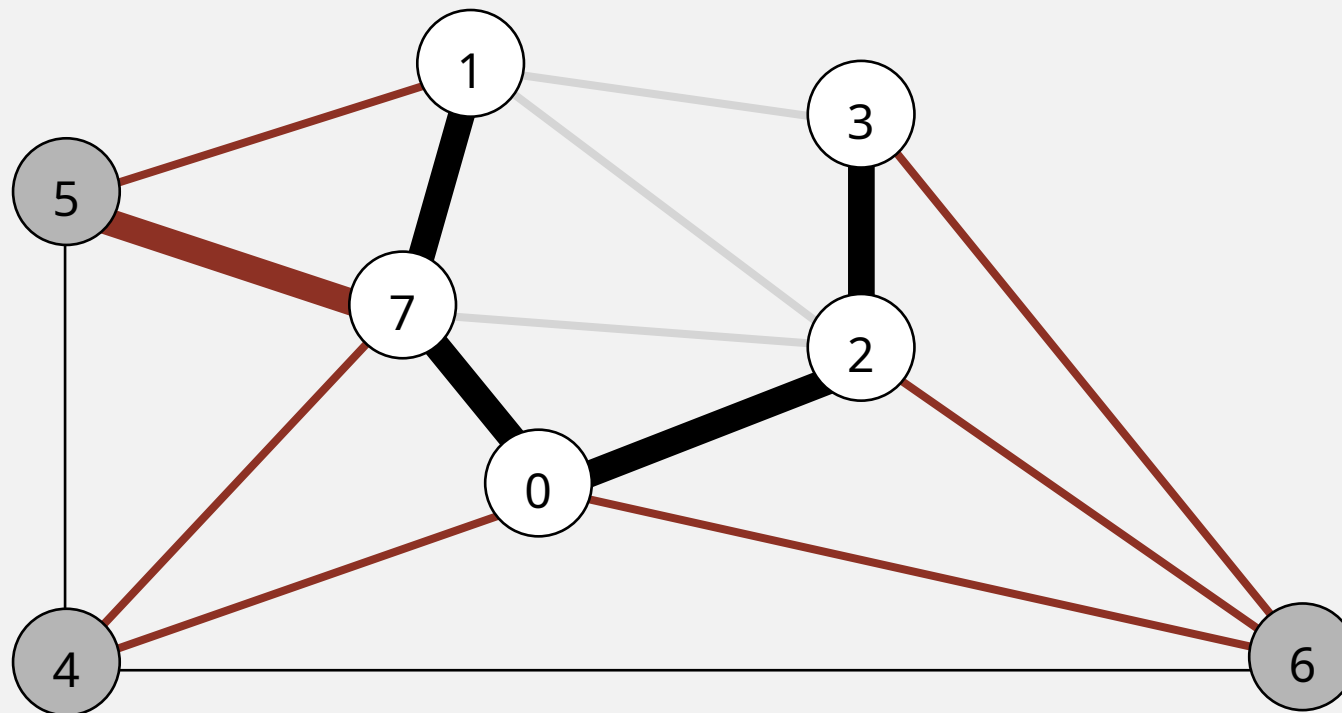
* 3-6 0.52

6-0 0.58

Prim's algorithm: lazy implementation demo

- Start with vertex 0 and greedily grow tree T .
- Add to T the min weight edge with exactly one endpoint in T .
- Repeat until $V - 1$ edges.

delete 5-7 and add to MST



MST edges

0-7 1-7 0-2 2-3

edges on PQ
(sorted by weight)

5-7 0.28

1-3 0.29

1-5 0.32

2-7 0.34

1-2 0.36

4-7 0.37

0-4 0.38

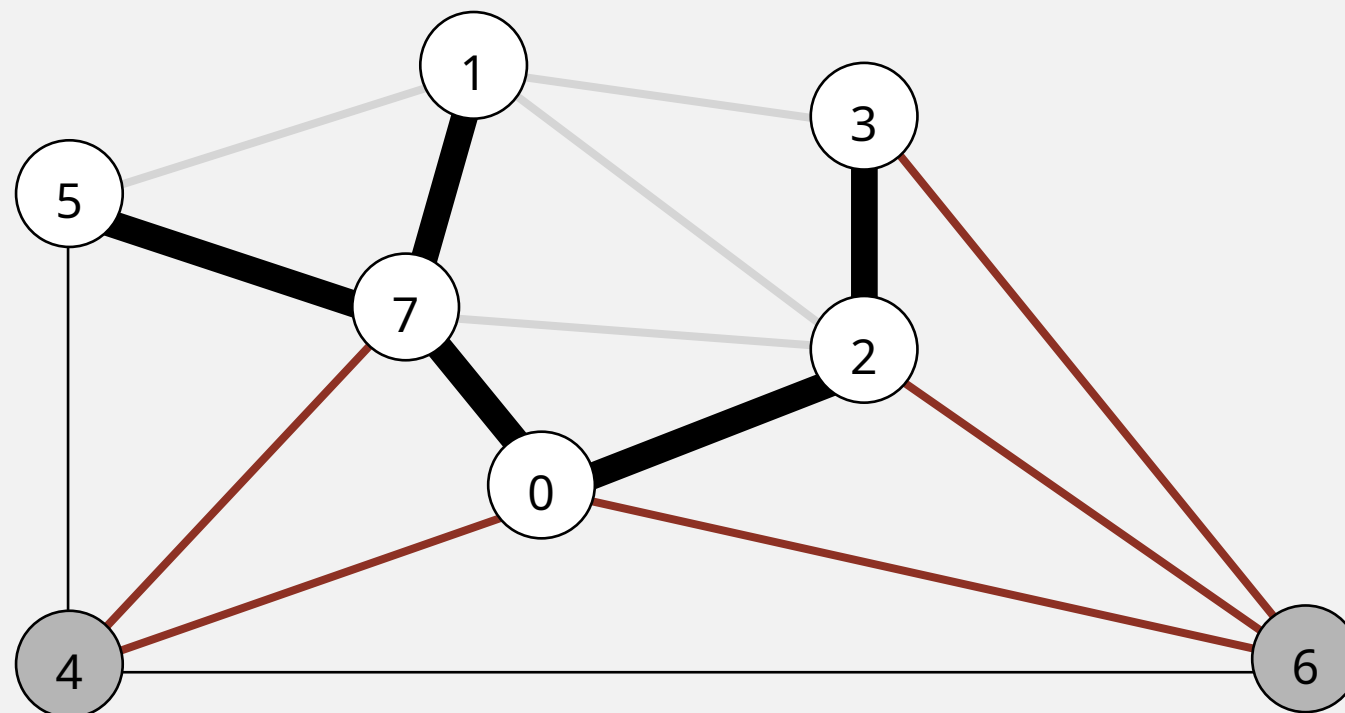
6-2 0.40

3-6 0.52

6-0 0.58

Prim's algorithm: lazy implementation demo

- Start with vertex 0 and greedily grow tree T .
- Add to T the min weight edge with exactly one endpoint in T .
- Repeat until $V - 1$ edges.



MST edges

0-7 1-7 0-2 2-3 5-7

edges on PQ
(sorted by weight)

1-3 0.29

1-5 0.32

2-7 0.34

1-2 0.36

4-7 0.37

0-4 0.38

6-2 0.40

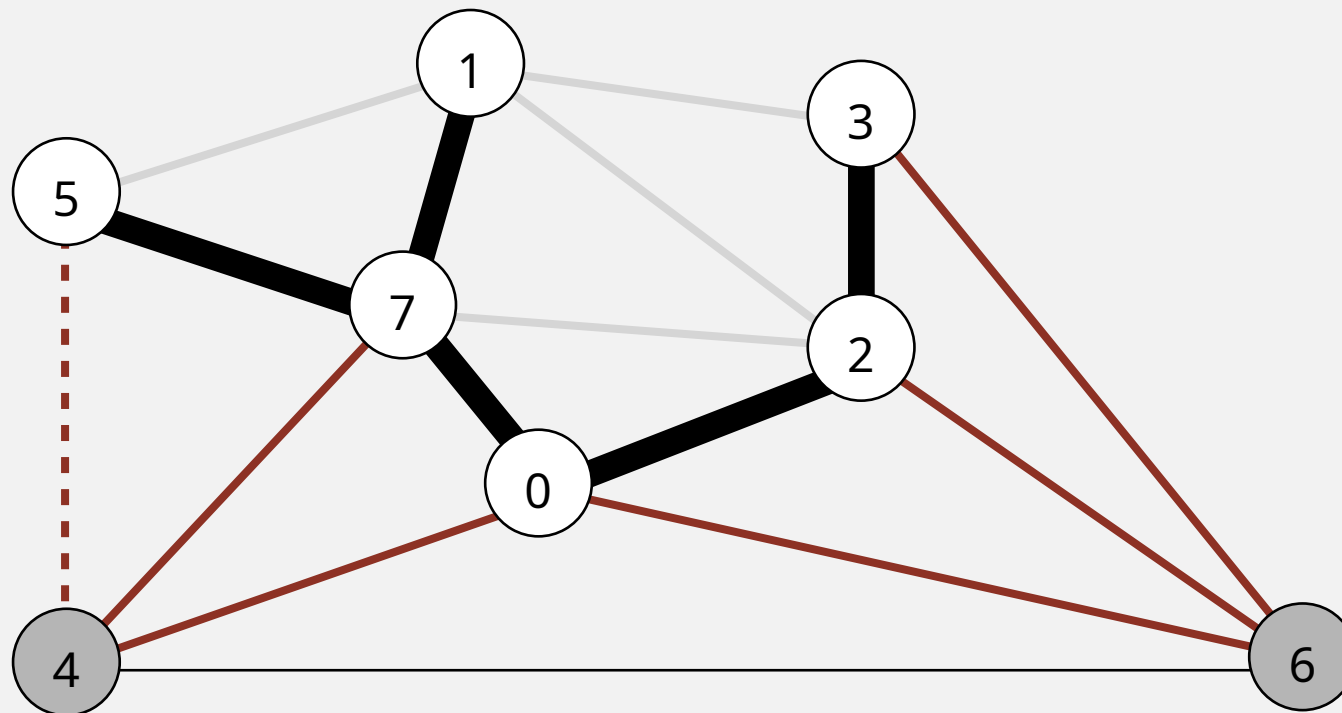
3-6 0.52

6-0 0.58

Prim's algorithm: lazy implementation demo

- Start with vertex 0 and greedily grow tree T .
- Add to T the min weight edge with exactly one endpoint in T .
- Repeat until $V - 1$ edges.

add to PQ all edges incident to 5



MST edges

0-7 1-7 0-2 2-3 5-7

edges on PQ
(sorted by weight)

1-3 0.29

1-5 0.32

2-7 0.34

* 4-5 0.35

1-2 0.36

4-7 0.37

0-4 0.38

6-2 0.40

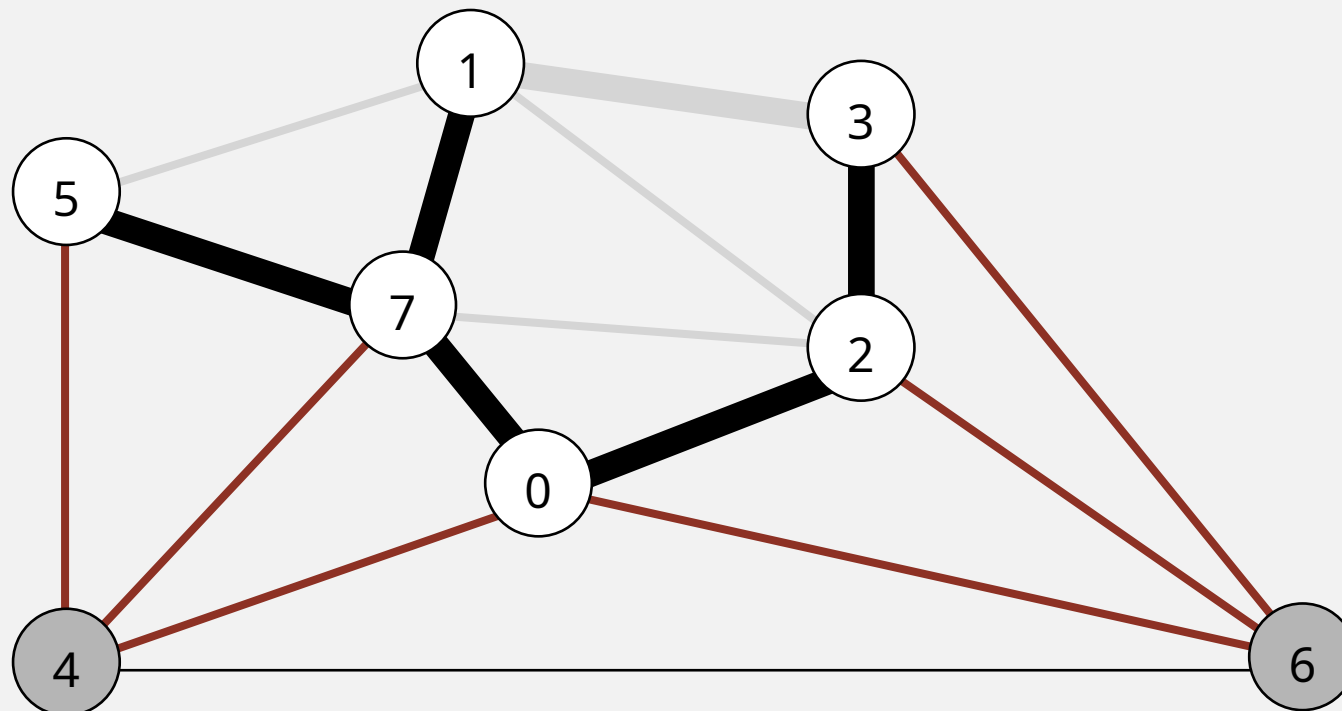
3-6 0.52

6-0 0.58

Prim's algorithm: lazy implementation demo

- Start with vertex 0 and greedily grow tree T .
- Add to T the min weight edge with exactly one endpoint in T .
- Repeat until $V - 1$ edges.

delete 1-3 and discard obsolete edge



MST edges

0-7 1-7 0-2 2-3 5-7

edges on PQ
(sorted by weight)

1-3 0.29

1-5 0.32

2-7 0.34

4-5 0.35

1-2 0.36

4-7 0.37

0-4 0.38

6-2 0.40

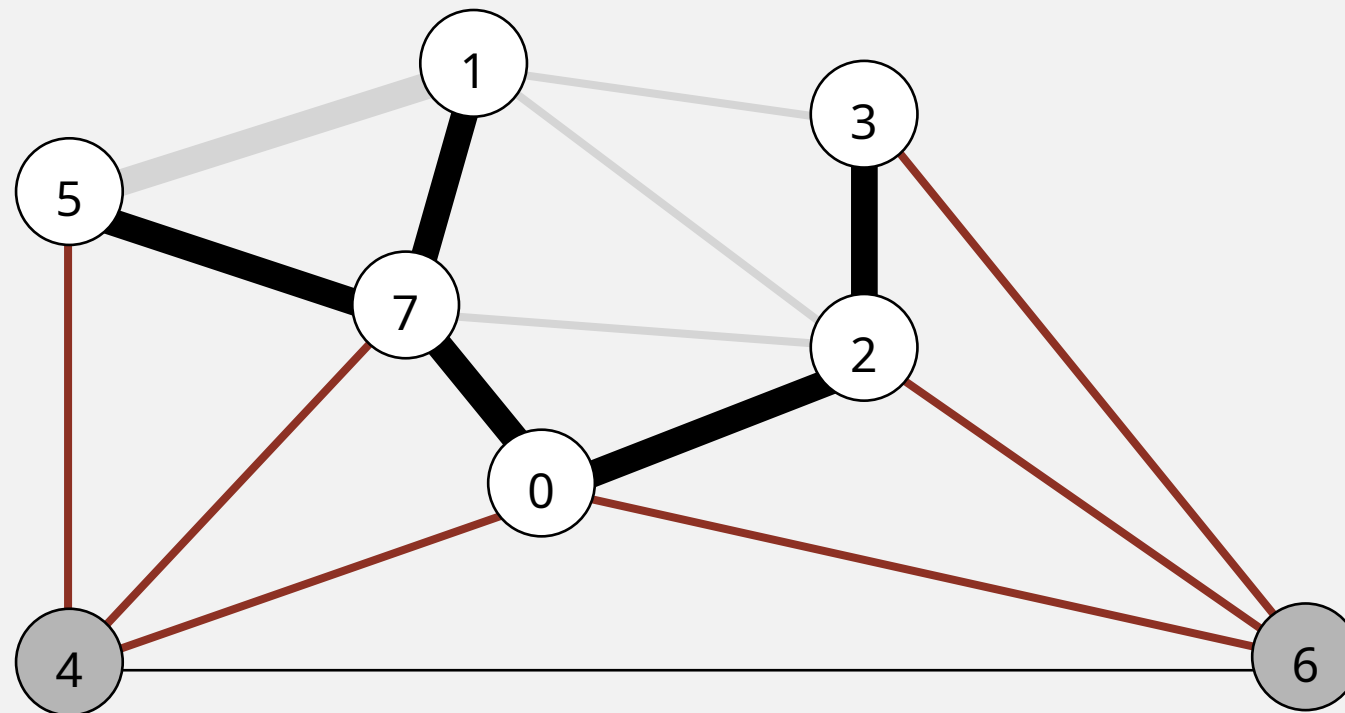
3-6 0.52

6-0 0.58

Prim's algorithm: lazy implementation demo

- Start with vertex 0 and greedily grow tree T .
- Add to T the min weight edge with exactly one endpoint in T .
- Repeat until $V - 1$ edges.

delete 1-5 and discard obsolete edge



MST edges

0-7 1-7 0-2 2-3 5-7

edges on PQ
(sorted by weight)

1-5 0.32

2-7 0.34

4-5 0.35

1-2 0.36

4-7 0.37

0-4 0.38

6-2 0.40

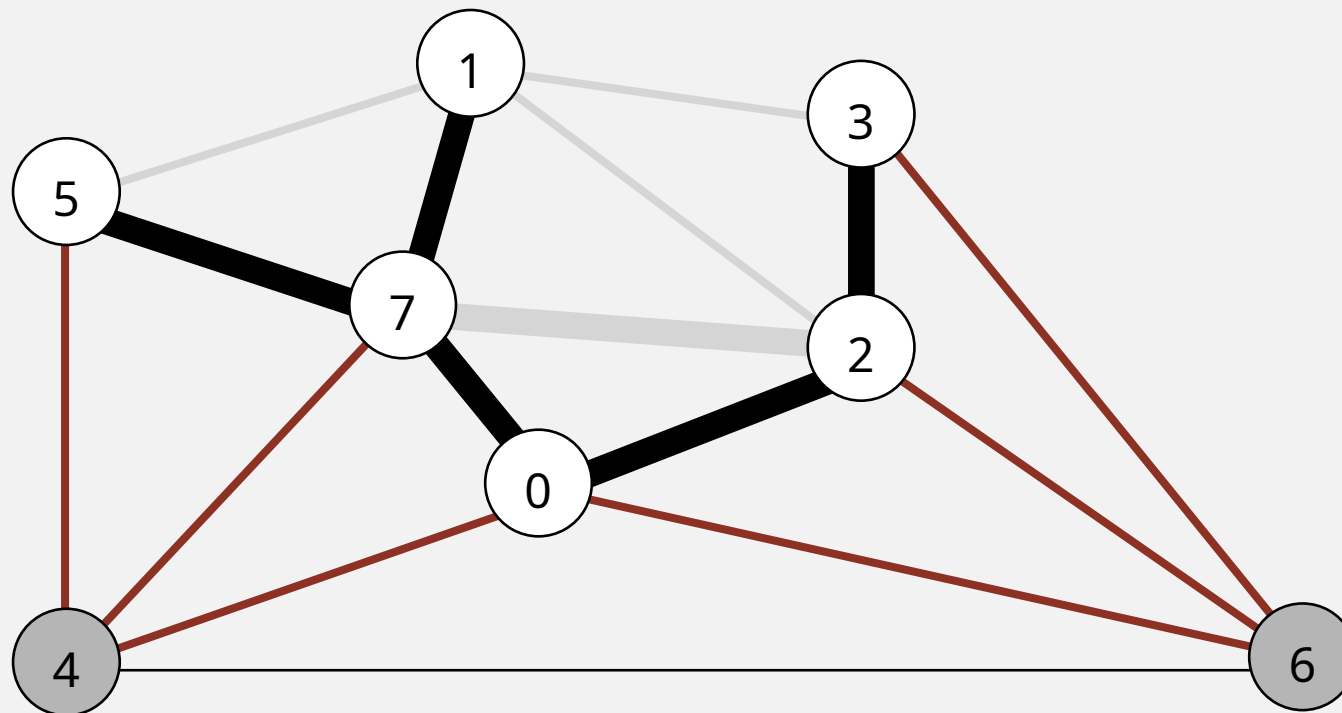
3-6 0.52

6-0 0.58

Prim's algorithm: lazy implementation demo

- Start with vertex 0 and greedily grow tree T .
- Add to T the min weight edge with exactly one endpoint in T .
- Repeat until $V - 1$ edges.

delete 2-7 and discard obsolete edge



edges on PQ
(sorted by weight)

2-7 0.34

4-5 0.35

1-2 0.36

4-7 0.37

0-4 0.38

6-2 0.40

3-6 0.52

6-0 0.58

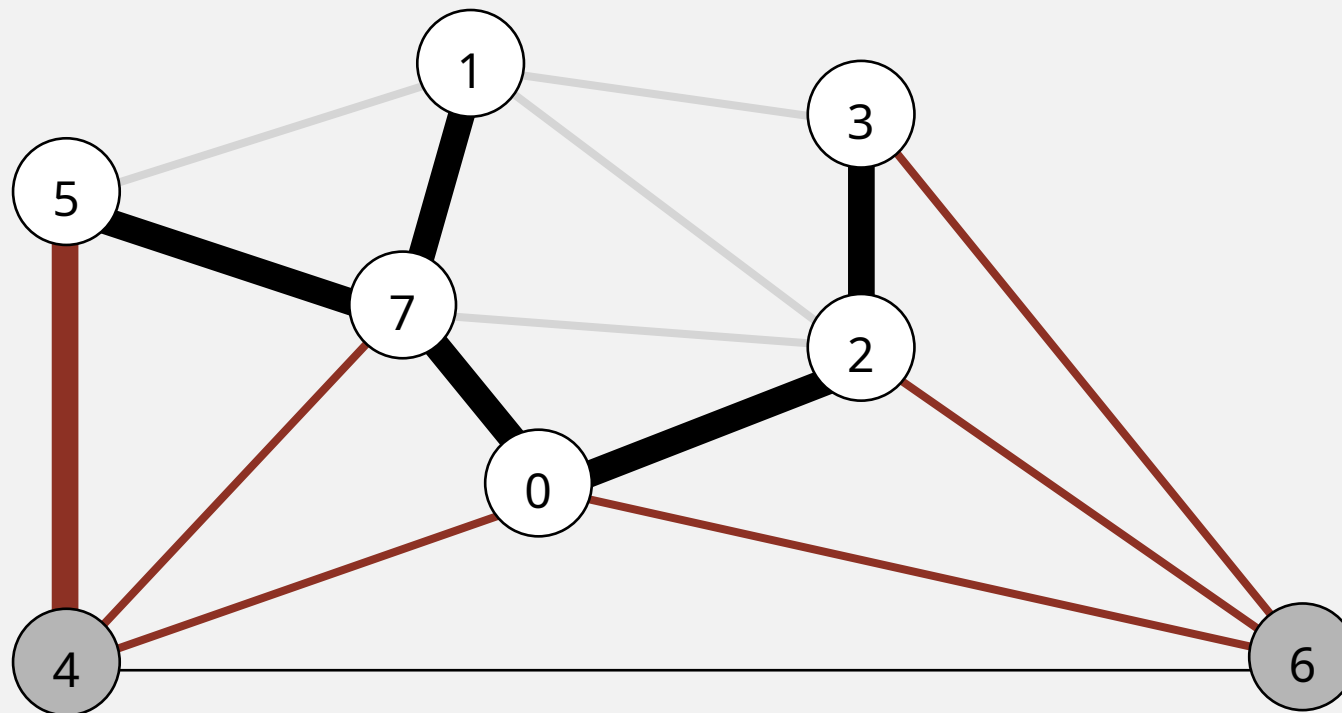
MST edges

0-7 1-7 0-2 2-3 5-7

Prim's algorithm: lazy implementation demo

- Start with vertex 0 and greedily grow tree T .
- Add to T the min weight edge with exactly one endpoint in T .
- Repeat until $V - 1$ edges.

delete 4-5 and add to MST



edges on PQ
(sorted by weight)

4-5 0.35

1-2 0.36

4-7 0.37

0-4 0.38

6-2 0.40

3-6 0.52

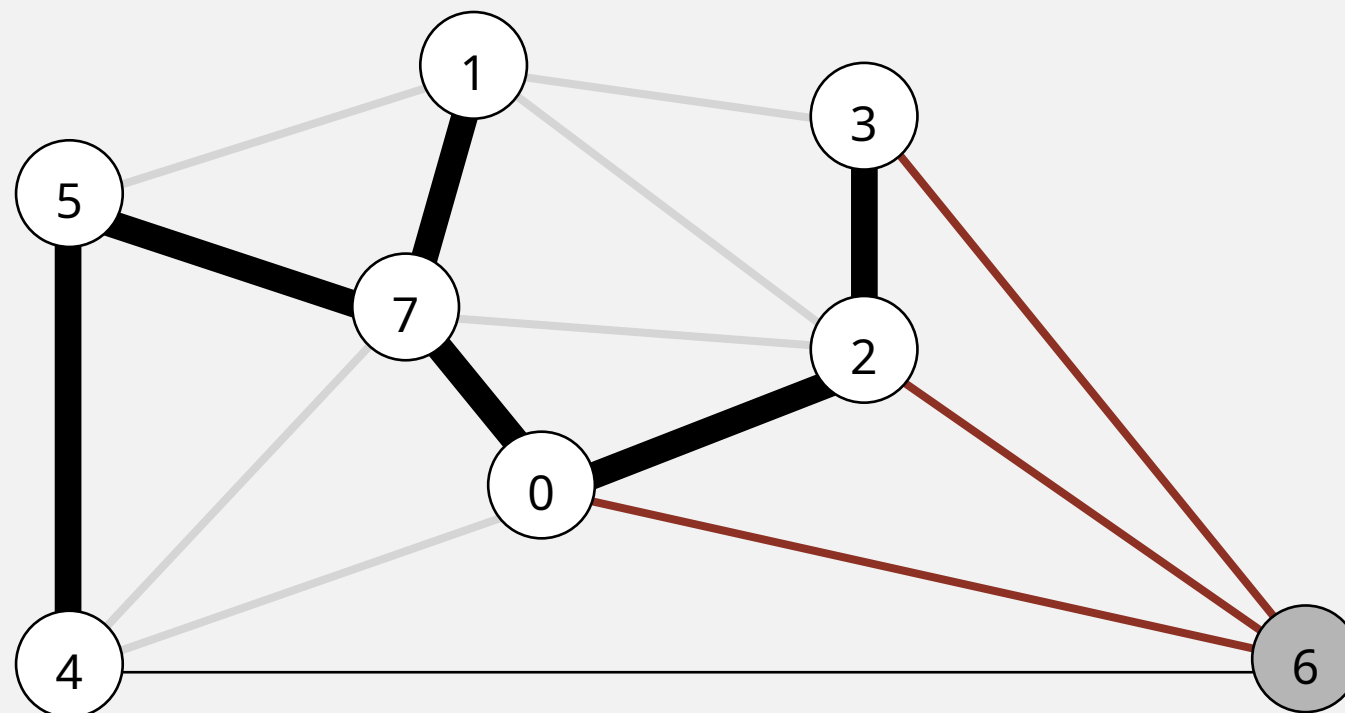
6-0 0.58

MST edges

0-7 1-7 0-2 2-3 5-7

Prim's algorithm: lazy implementation demo

- Start with vertex 0 and greedily grow tree T .
- Add to T the min weight edge with exactly one endpoint in T .
- Repeat until $V - 1$ edges.



edges on PQ
(sorted by weight)

1-2 0.36

4-7 0.37

0-4 0.38

6-2 0.40

3-6 0.52

6-0 0.58

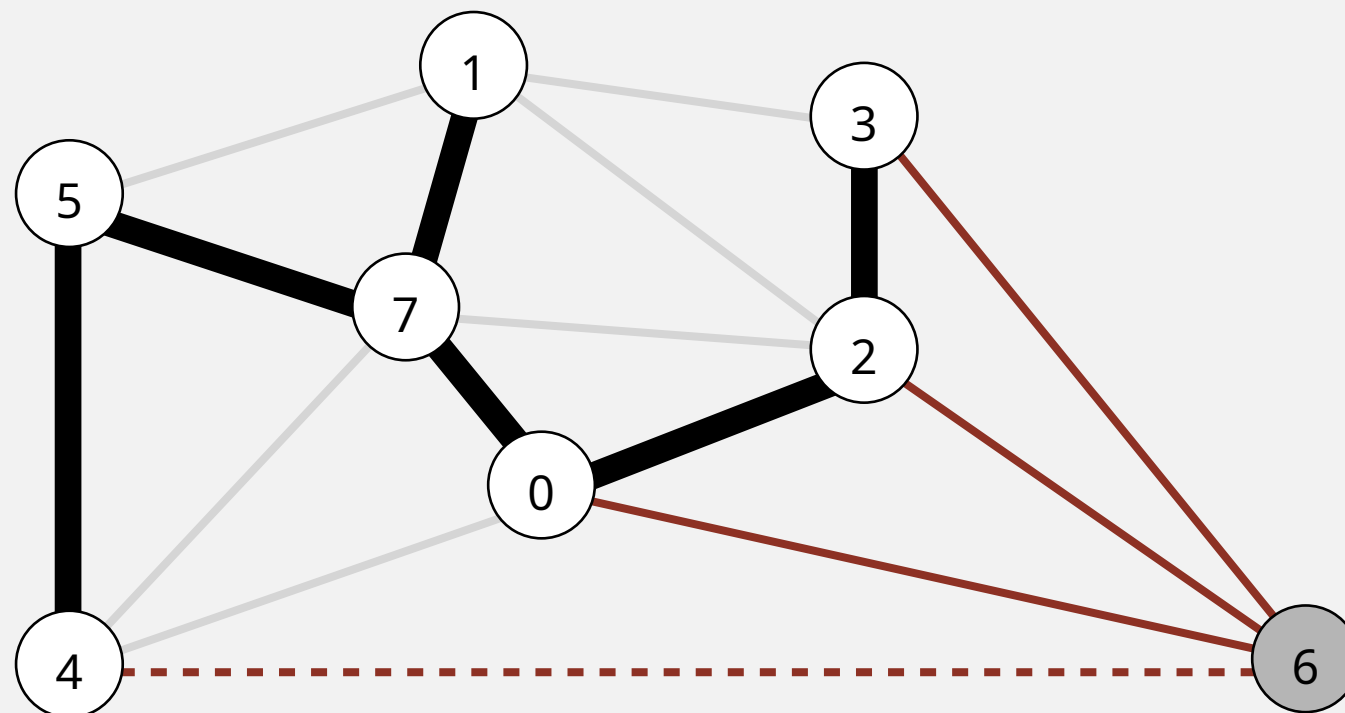
MST edges

0-7 1-7 0-2 2-3 5-7 4-5

Prim's algorithm: lazy implementation demo

- Start with vertex 0 and greedily grow tree T .
- Add to T the min weight edge with exactly one endpoint in T .
- Repeat until $V - 1$ edges.

add to PQ all edges incident to 4



edges on PQ
(sorted by weight)

1-2 0.36

4-7 0.37

0-4 0.38

6-2 0.40

3-6 0.52

6-0 0.58

* 6-4 0.93

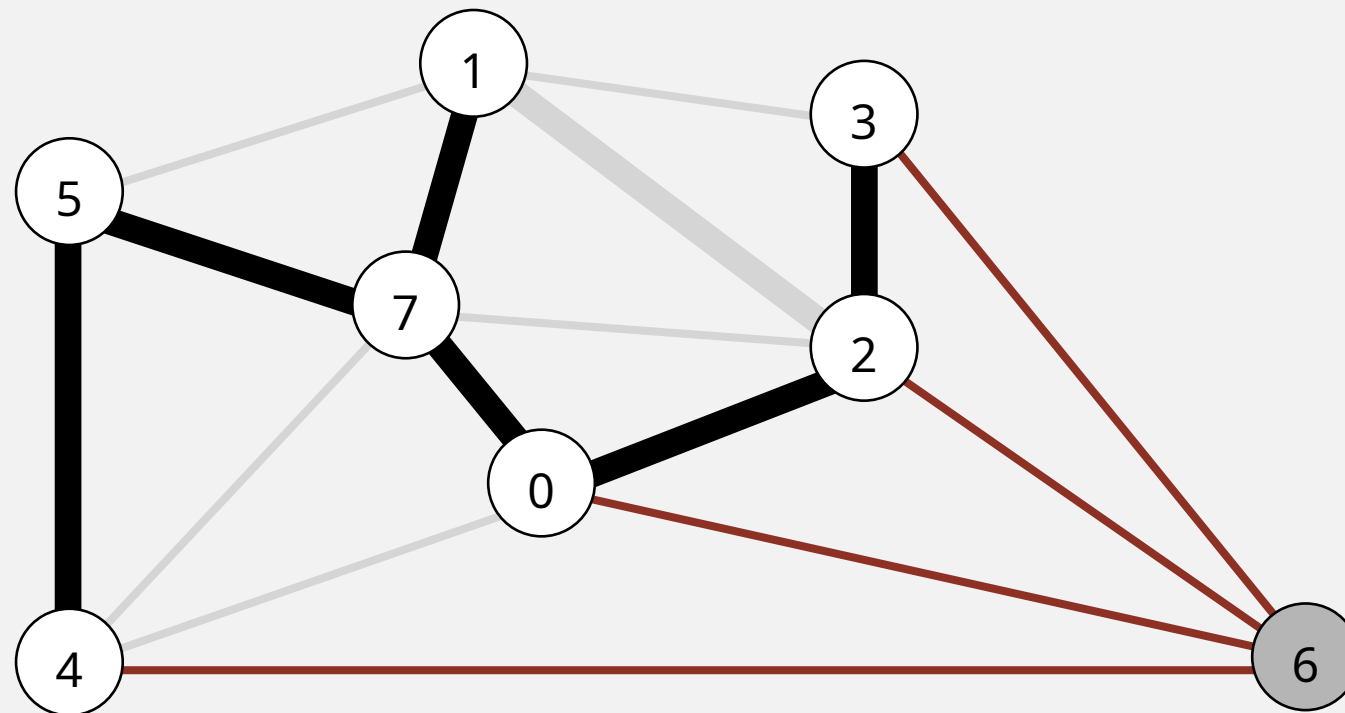
MST edges

0-7 1-7 0-2 2-3 5-7 4-5

Prim's algorithm: lazy implementation demo

- Start with vertex 0 and greedily grow tree T .
- Add to T the min weight edge with exactly one endpoint in T .
- Repeat until $V - 1$ edges.

delete 1-2 and discard obsolete edge



edges on PQ
(sorted by weight)

1-2 0.36

4-7 0.37

0-4 0.38

6-2 0.40

3-6 0.52

6-0 0.58

6-4 0.93

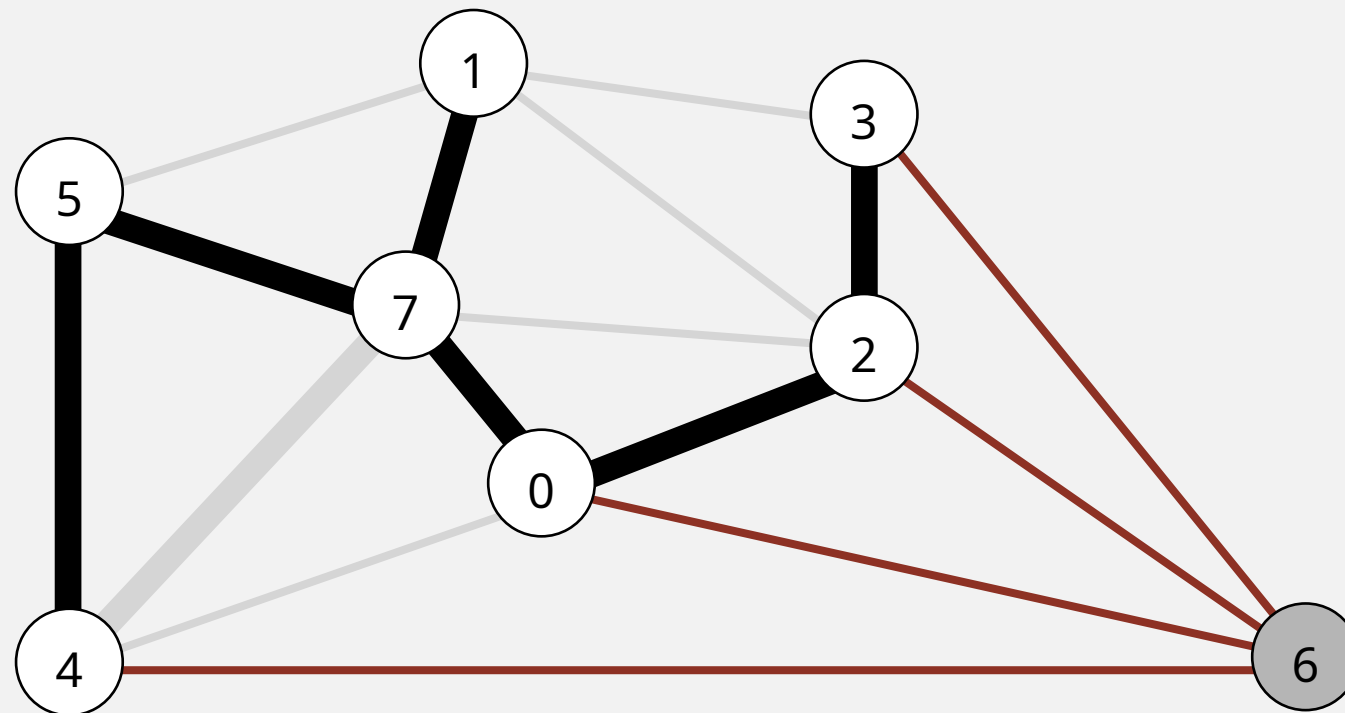
MST edges

0-7 1-7 0-2 2-3 5-7 4-5

Prim's algorithm: lazy implementation demo

- Start with vertex 0 and greedily grow tree T .
- Add to T the min weight edge with exactly one endpoint in T .
- Repeat until $V - 1$ edges.

delete 4-7 and discard obsolete edge



edges on PQ
(sorted by weight)

4-7 0.37

0-4 0.38

6-2 0.40

3-6 0.52

6-0 0.58

6-4 0.93

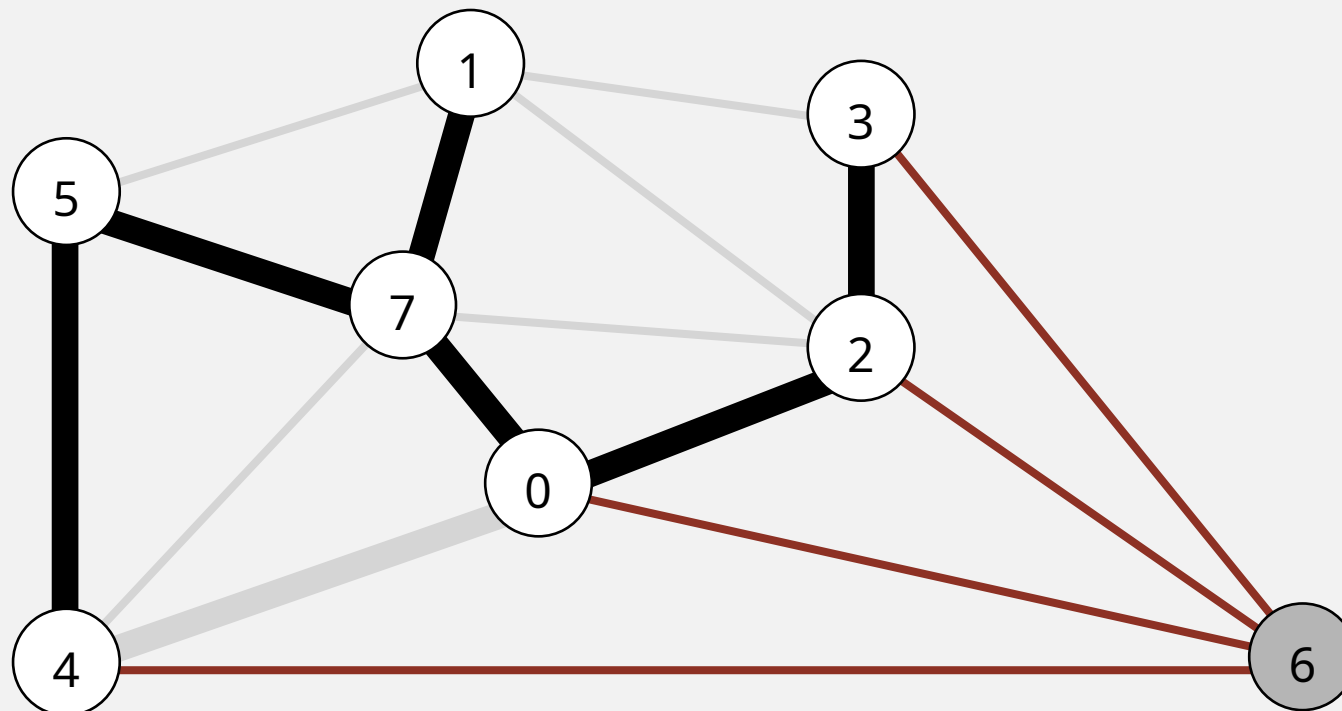
MST edges

0-7 1-7 0-2 2-3 5-7 4-5

Prim's algorithm: lazy implementation demo

- Start with vertex 0 and greedily grow tree T .
- Add to T the min weight edge with exactly one endpoint in T .
- Repeat until $V - 1$ edges.

delete 0-4 and discard obsolete edge



edges on PQ
(sorted by weight)

0-4 0.38

6-2 0.40

3-6 0.52

6-0 0.58

6-4 0.93

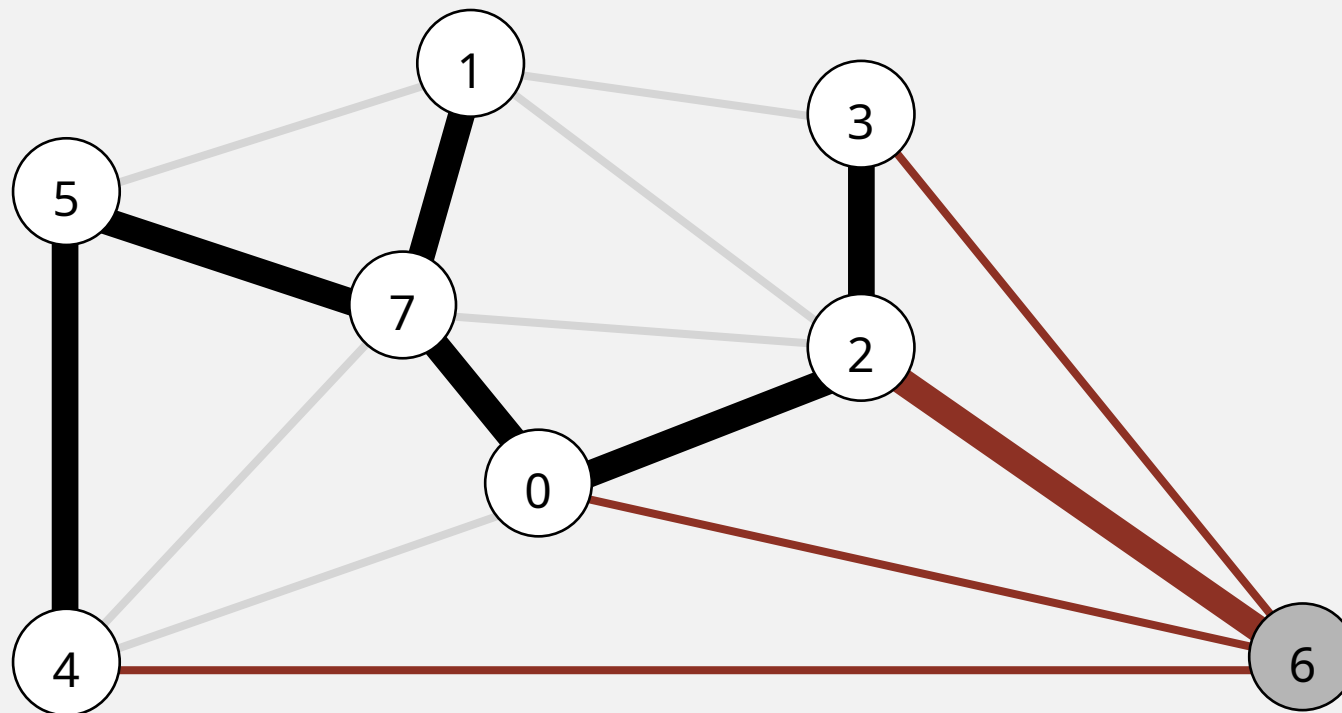
MST edges

0-7 1-7 0-2 2-3 5-7 4-5

Prim's algorithm: lazy implementation demo

- Start with vertex 0 and greedily grow tree T .
- Add to T the min weight edge with exactly one endpoint in T .
- Repeat until $V - 1$ edges.

delete 6-2 and add to MST



edges on PQ
(sorted by weight)

6-2 0.40

3-6 0.52

6-0 0.58

6-4 0.93

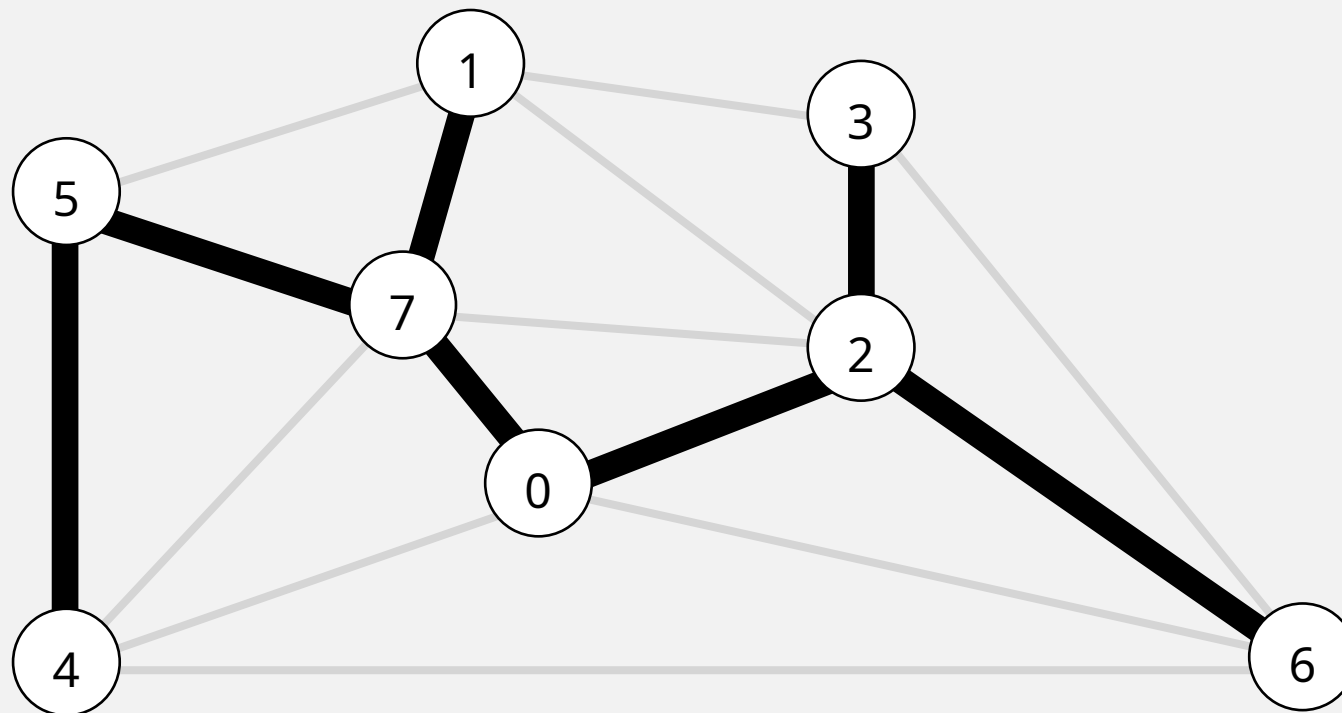
MST edges

0-7 1-7 0-2 2-3 5-7 4-5

Prim's algorithm: lazy implementation demo

- Start with vertex 0 and greedily grow tree T .
- Add to T the min weight edge with exactly one endpoint in T .
- Repeat until $V - 1$ edges.

delete 6-2 and add to MST



edges on PQ
(sorted by weight)

3-6 0.52

6-0 0.58

6-4 0.93

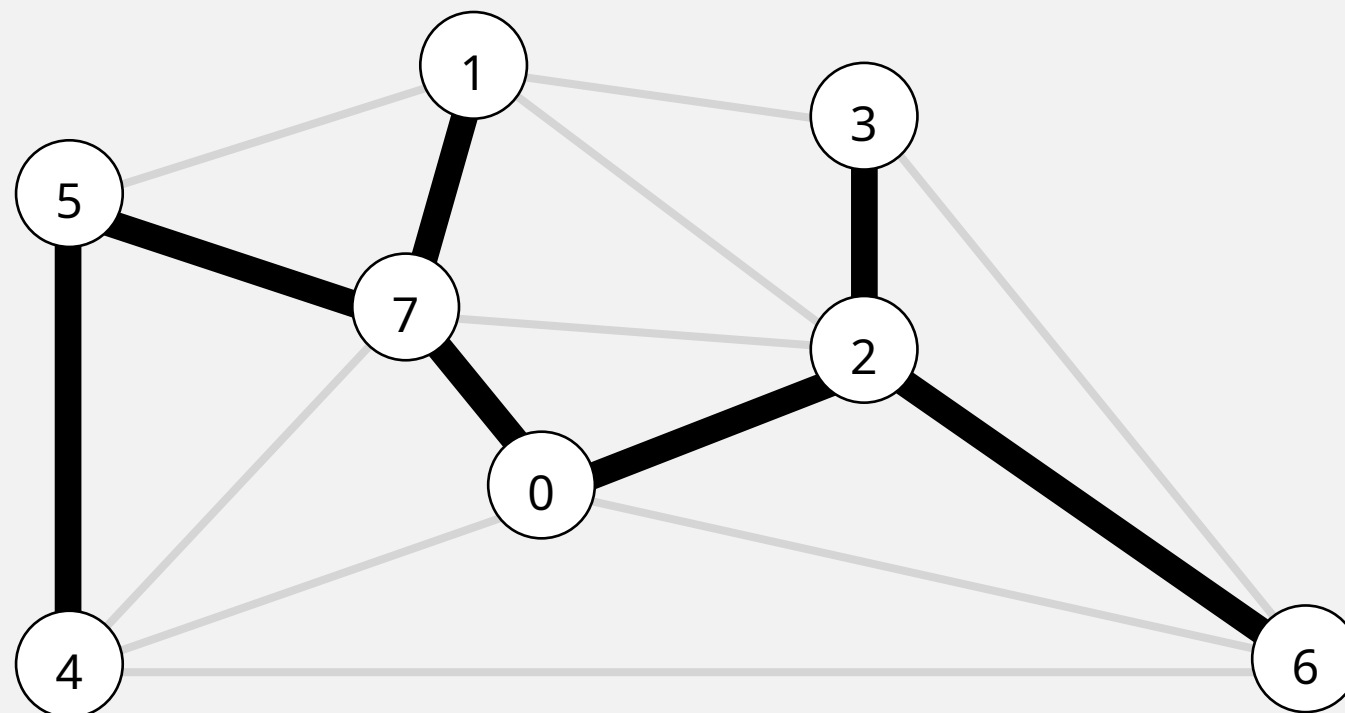
MST edges

0-7 1-7 0-2 2-3 5-7 4-5 6-2

Prim's algorithm: lazy implementation demo

- Start with vertex 0 and greedily grow tree T .
- Add to T the min weight edge with exactly one endpoint in T .
- Repeat until $V - 1$ edges.

stop since $V-1$ edges



edges on PQ
(sorted by weight)

3-6 0.52

6-0 0.58

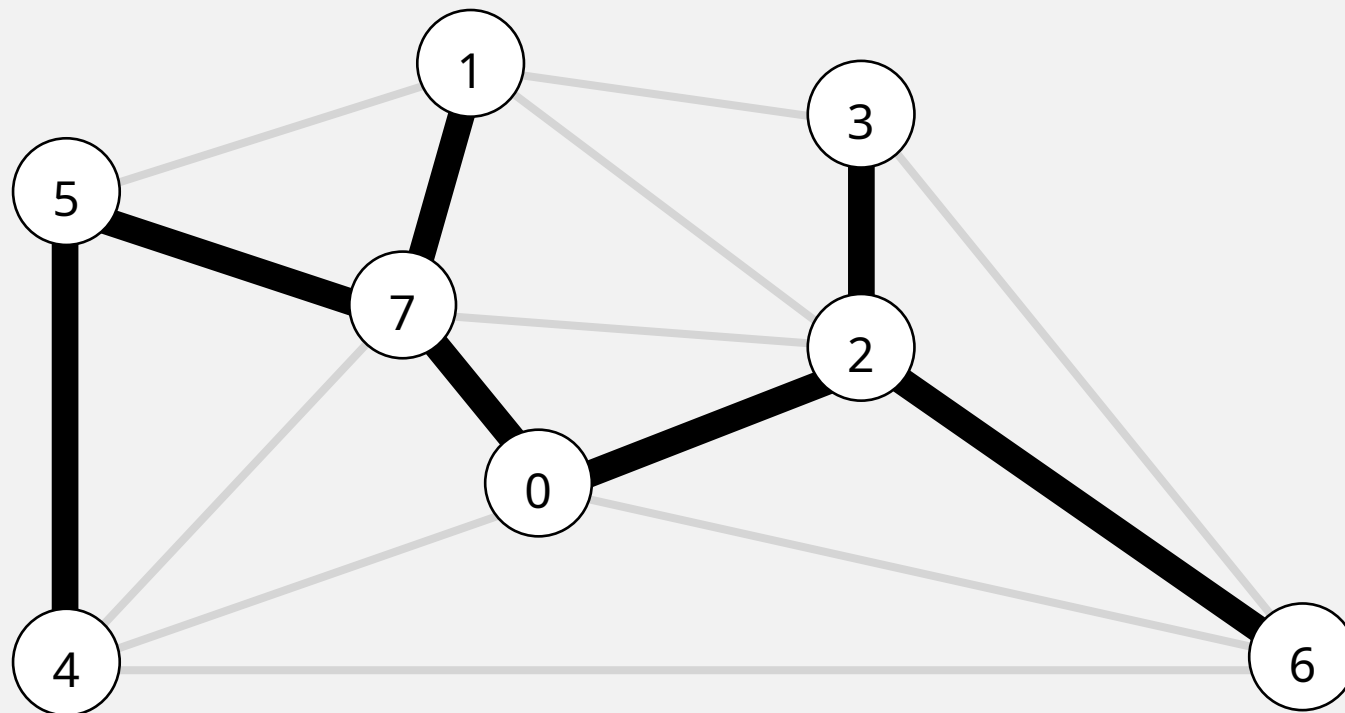
6-4 0.93

MST edges

0-7 1-7 0-2 2-3 5-7 4-5 6-2

Prim's algorithm: lazy implementation demo

- Start with vertex 0 and greedily grow tree T .
- Add to T the min weight edge with exactly one endpoint in T .
- Repeat until $V - 1$ edges.



MST edges

0-7 1-7 0-2 2-3 5-7 4-5 6-2



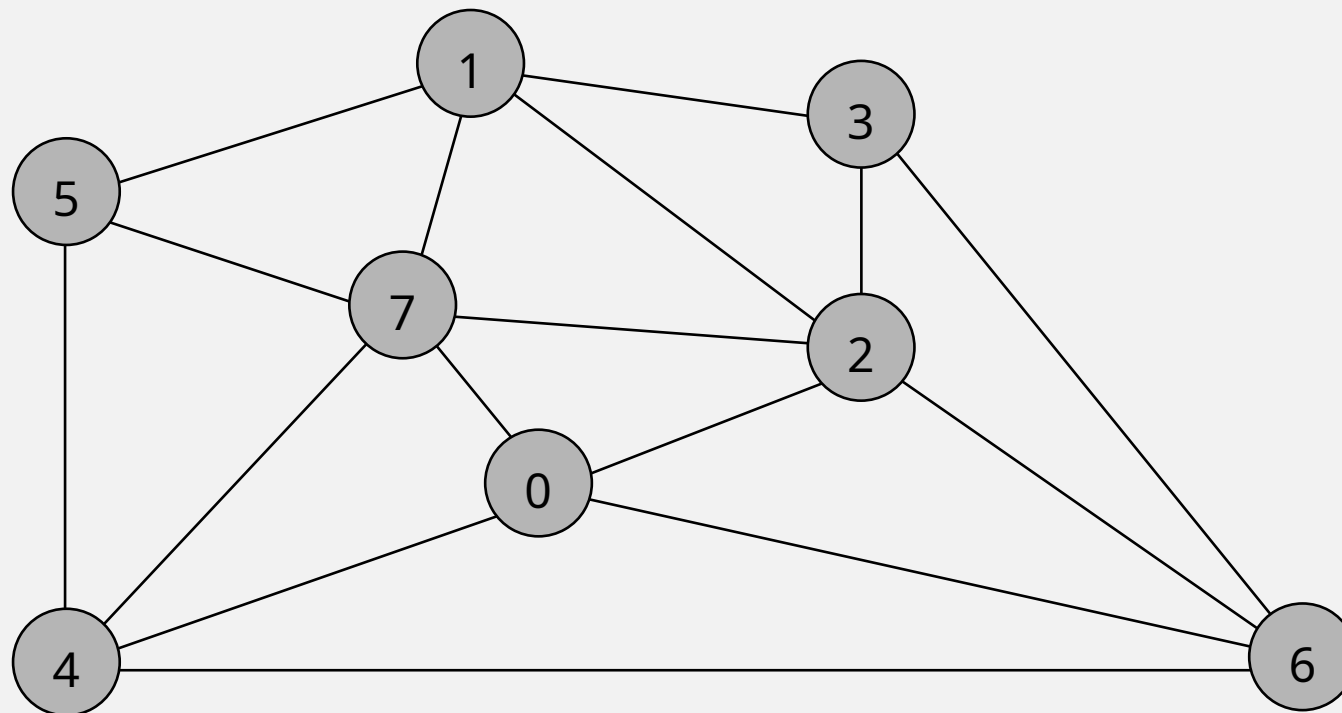
<http://algs4.cs.princeton.edu>

PRIM'S ALGORITHM DEMO

- *Prim's algorithm*
- *lazy implementation*
- *eager implementation*

Prim's algorithm: eager implementation demo

- Start with vertex 0 and greedily grow tree T .
- Add to T the min weight edge with exactly one endpoint in T .
- Repeat until $V - 1$ edges.

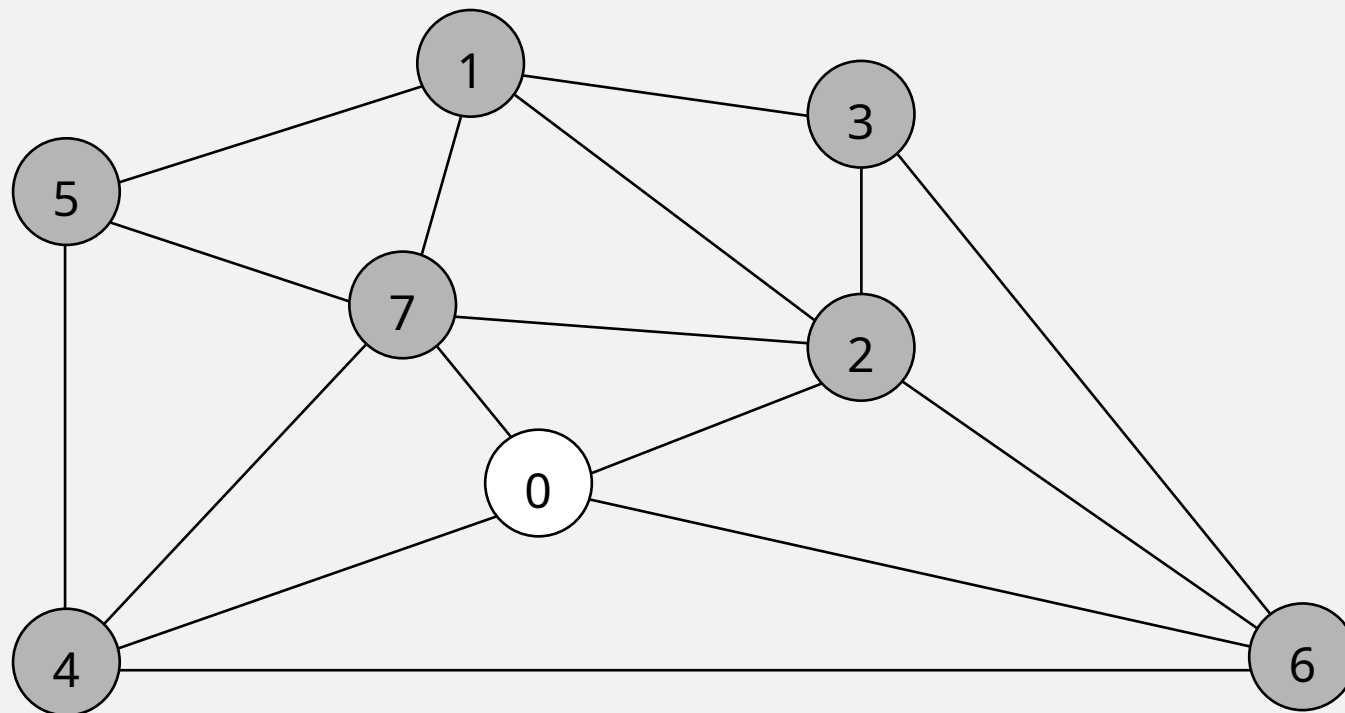


an edge-weighted graph

0-7	0.16
2-3	0.17
1-7	0.19
0-2	0.26
5-7	0.28
1-3	0.29
1-5	0.32
2-7	0.34
4-5	0.35
1-2	0.36
4-7	0.37
0-4	0.38
6-2	0.40
3-6	0.52

Prim's algorithm: eager implementation demo

- Start with vertex 0 and greedily grow tree T .
- Add to T the min weight edge with exactly one endpoint in T .
- Repeat until $V - 1$ edges.

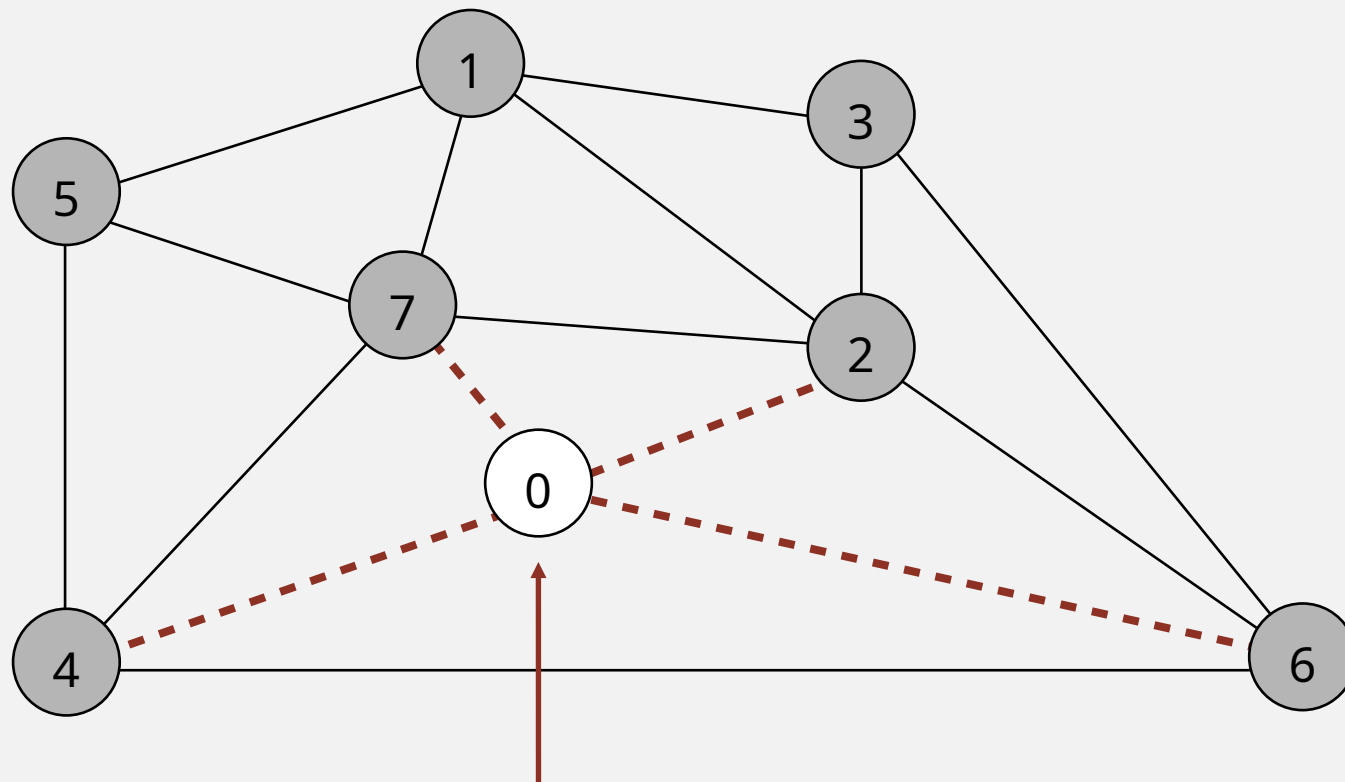


→

v	edgeTo[]	distTo[]
0	-	-

Prim's algorithm: eager implementation demo

- Start with vertex 0 and greedily grow tree T .
- Add to T the min weight edge with exactly one endpoint in T .
- Repeat until $V - 1$ edges.



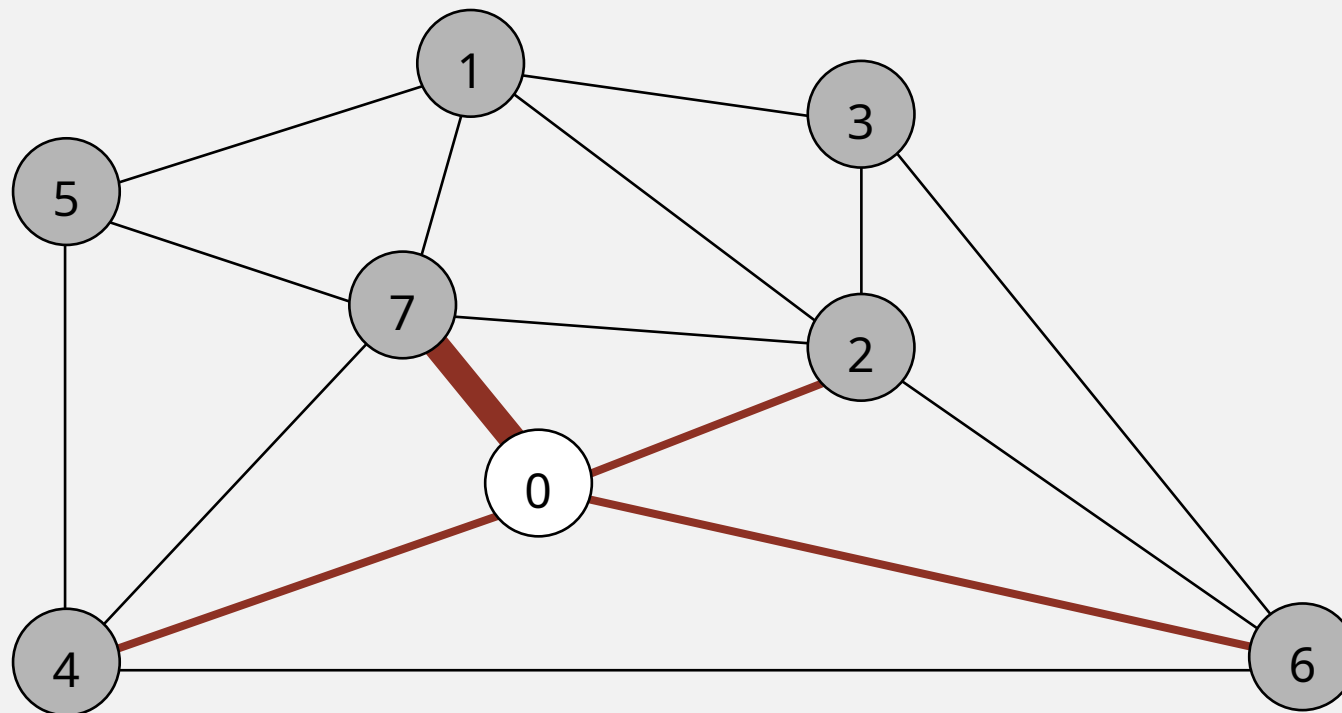
add vertices 7, 2, 4, and 6 to PQ

	v	edgeTo[]	distTo[]	
→	0	-	-	
	7	0-7	0.16	2 0-2
			0.26	
	4	0-4	0.38	
	6	6-0	0.58	

vertices on PQ
(sorted by weight)

Prim's algorithm: eager implementation demo

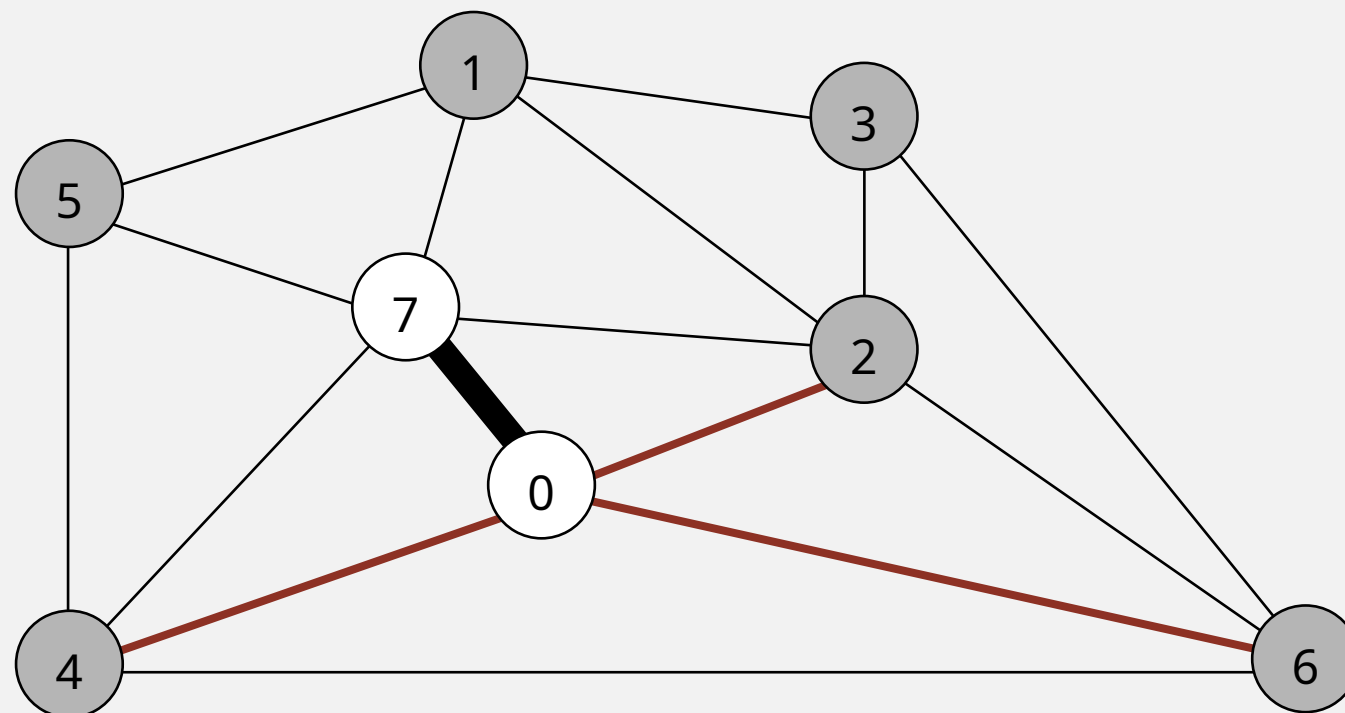
- Start with vertex 0 and greedily grow tree T .
- Add to T the min weight edge with exactly one endpoint in T .
- Repeat until $V - 1$ edges.



v	edgeTo[]	distTo[]		
0	-	-		
→				
7	0-7	0.16	2	0-2
		0.26		
4	0-4	0.38		
6	6-0	0.58		
vertices on PQ (sorted by weight)				

Prim's algorithm: eager implementation demo

- Start with vertex 0 and greedily grow tree T .
- Add to T the min weight edge with exactly one endpoint in T .
- Repeat until $V - 1$ edges.



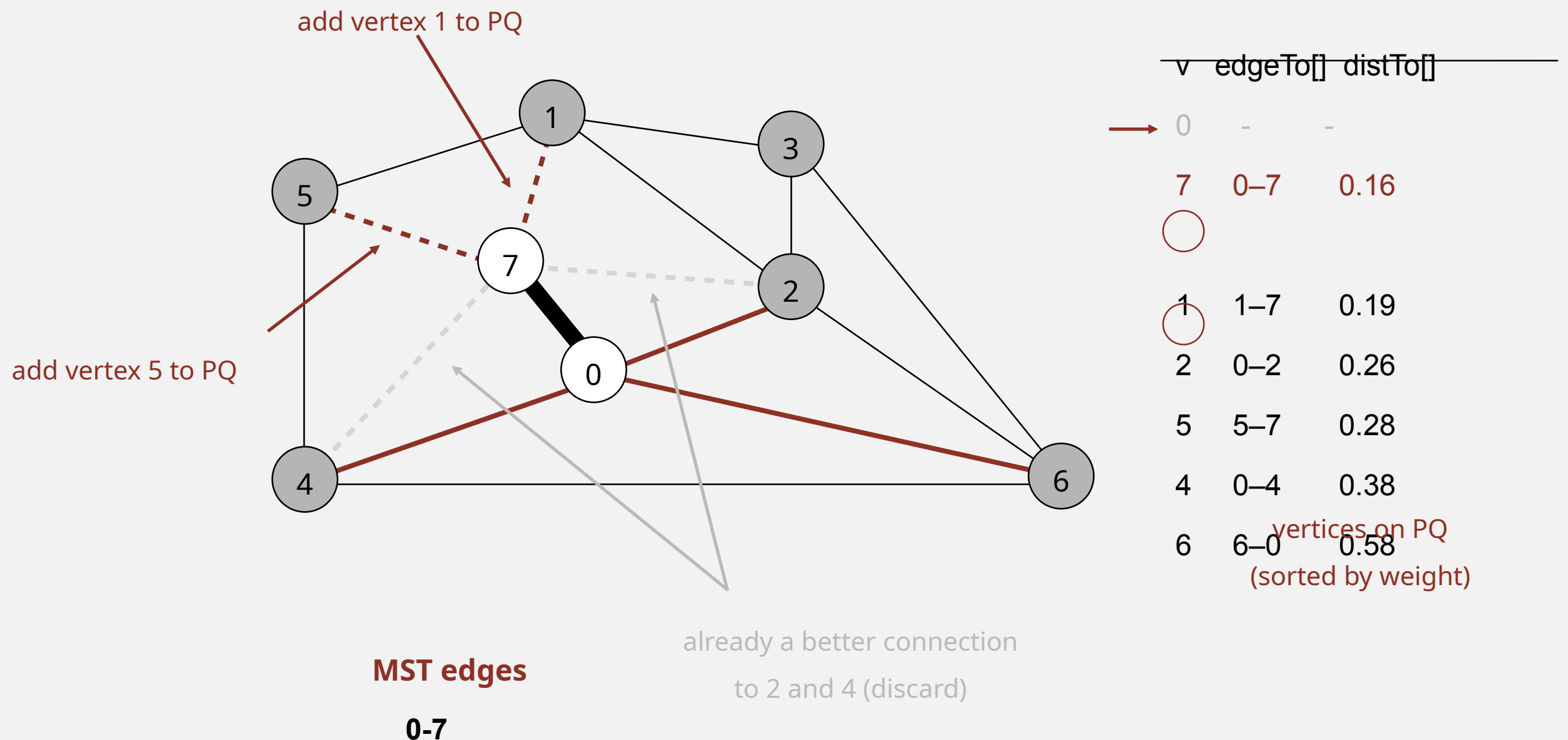
v	edgeTo[]	distTo[]
→ 0	-	-
7	0-7	0.16
2	0-2	0.26
4	0-4	0.38
6	6-0	0.58

MST edges

0-7

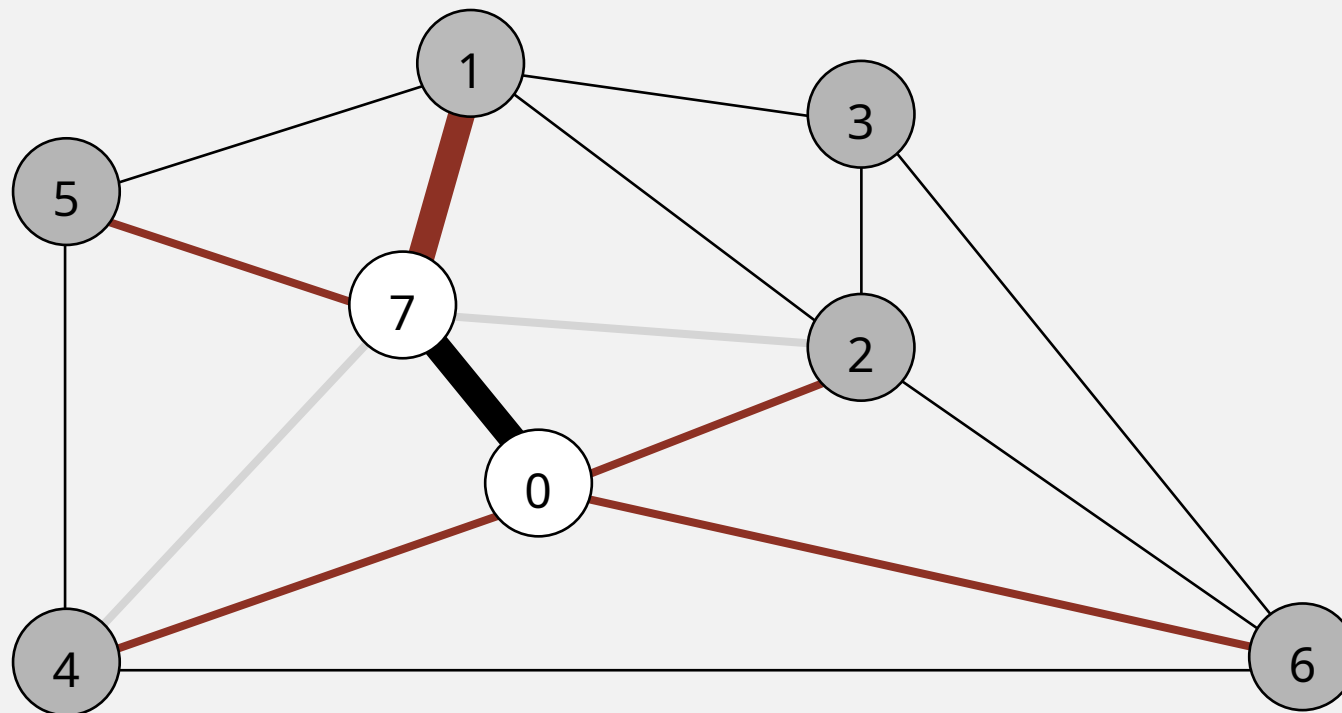
Prim's algorithm: eager implementation demo

- Start with vertex 0 and greedily grow tree T .
- Add to T the min weight edge with exactly one endpoint in T .
- Repeat until $V - 1$ edges.



Prim's algorithm: eager implementation demo

- Start with vertex 0 and greedily grow tree T .
- Add to T the min weight edge with exactly one endpoint in T .
- Repeat until $V - 1$ edges.



MST edges

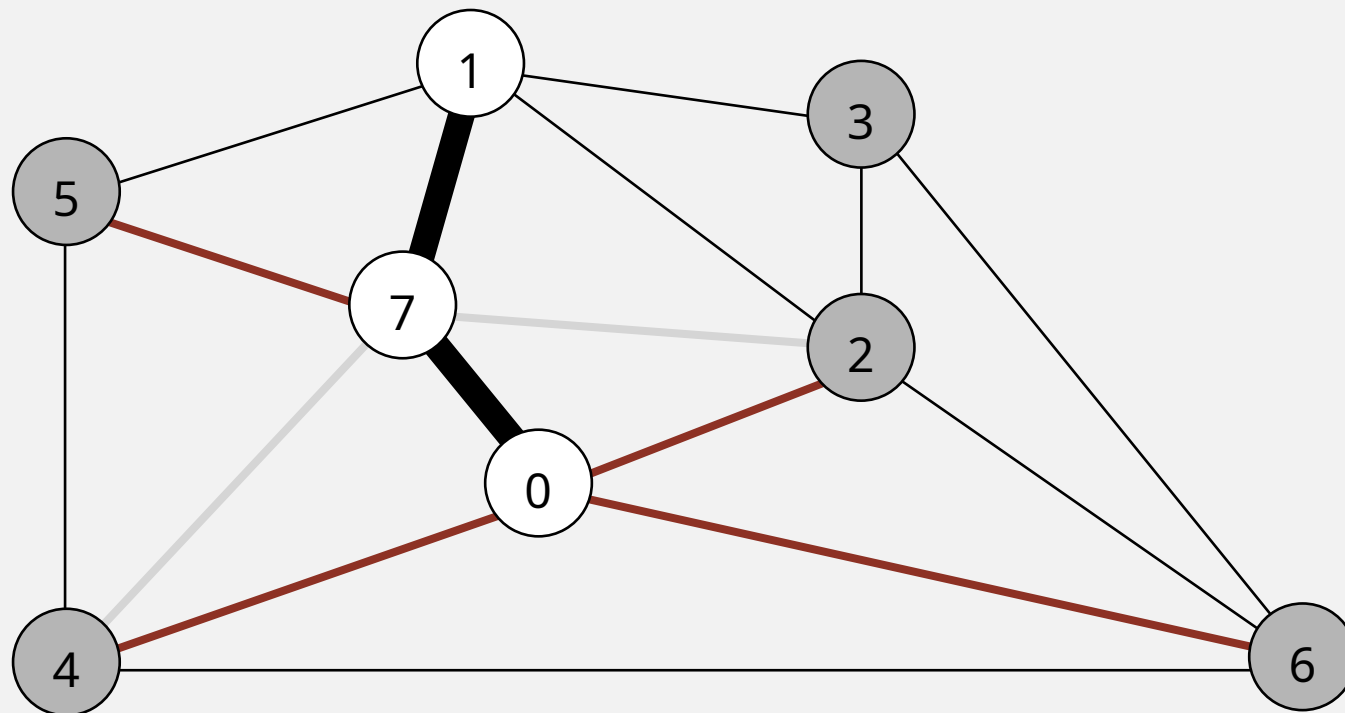
0-7 1-7

v	edgeTo[]	distTo[]
0	-	-
7	0-7	0.16
→		
1	1-7	0.19
2	0-2	0.26
5	5-7	0.28
4	0-4	0.38
6	6-0	0.58

vertices on PQ
(sorted by weight)

Prim's algorithm: eager implementation demo

- Start with vertex 0 and greedily grow tree T .
- Add to T the min weight edge with exactly one endpoint in T .
- Repeat until $V - 1$ edges.



MST edges

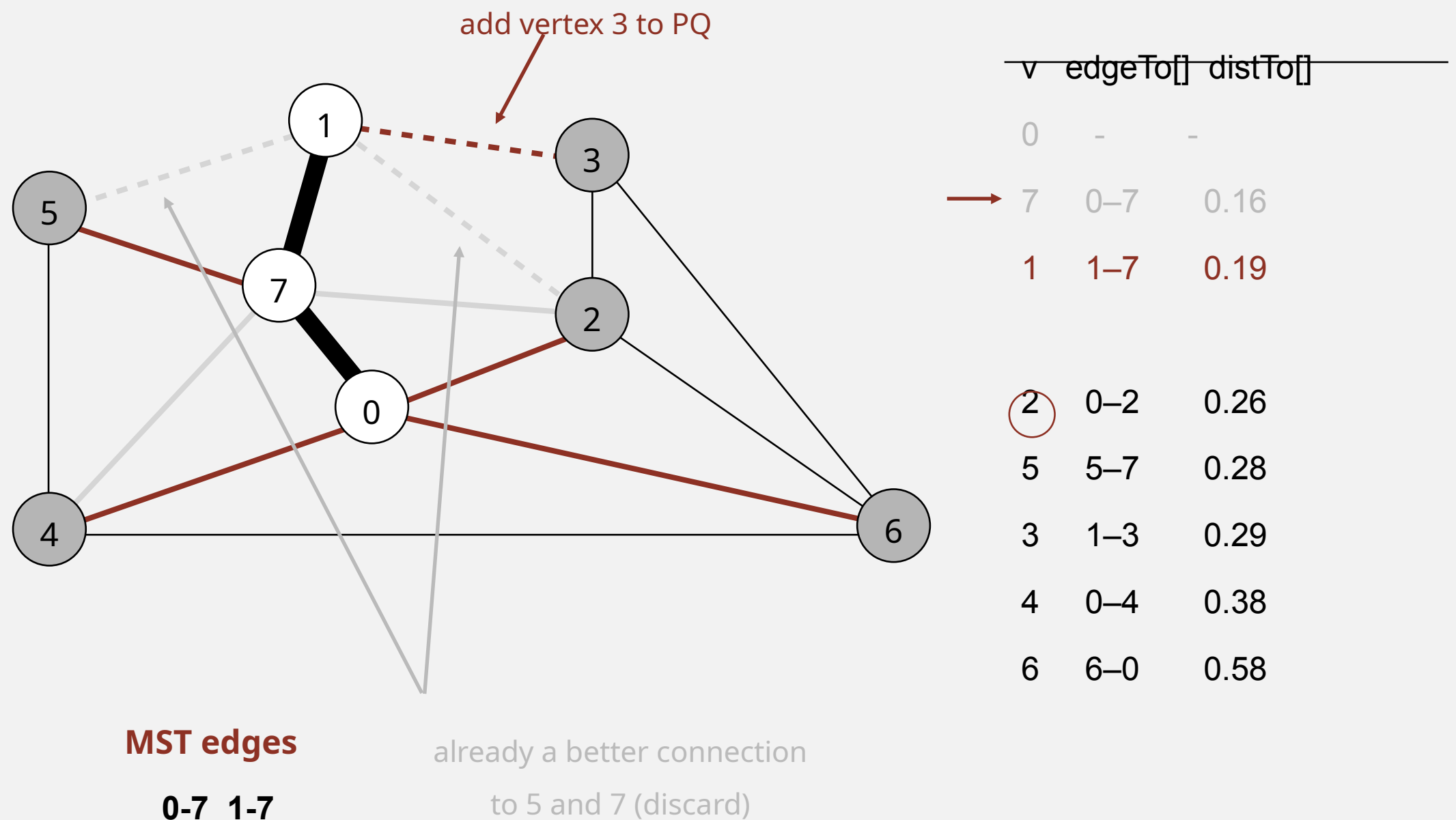
0-7 1-7

v	edgeTo[]	distTo[]
0	-	-
→ 7	0-7	0.16
1	1-7	0.19
2	0-2	0.26
5	5-7	0.28
4	0-4	0.38
6	6-0	0.58

vertices on PQ
(sorted by weight)

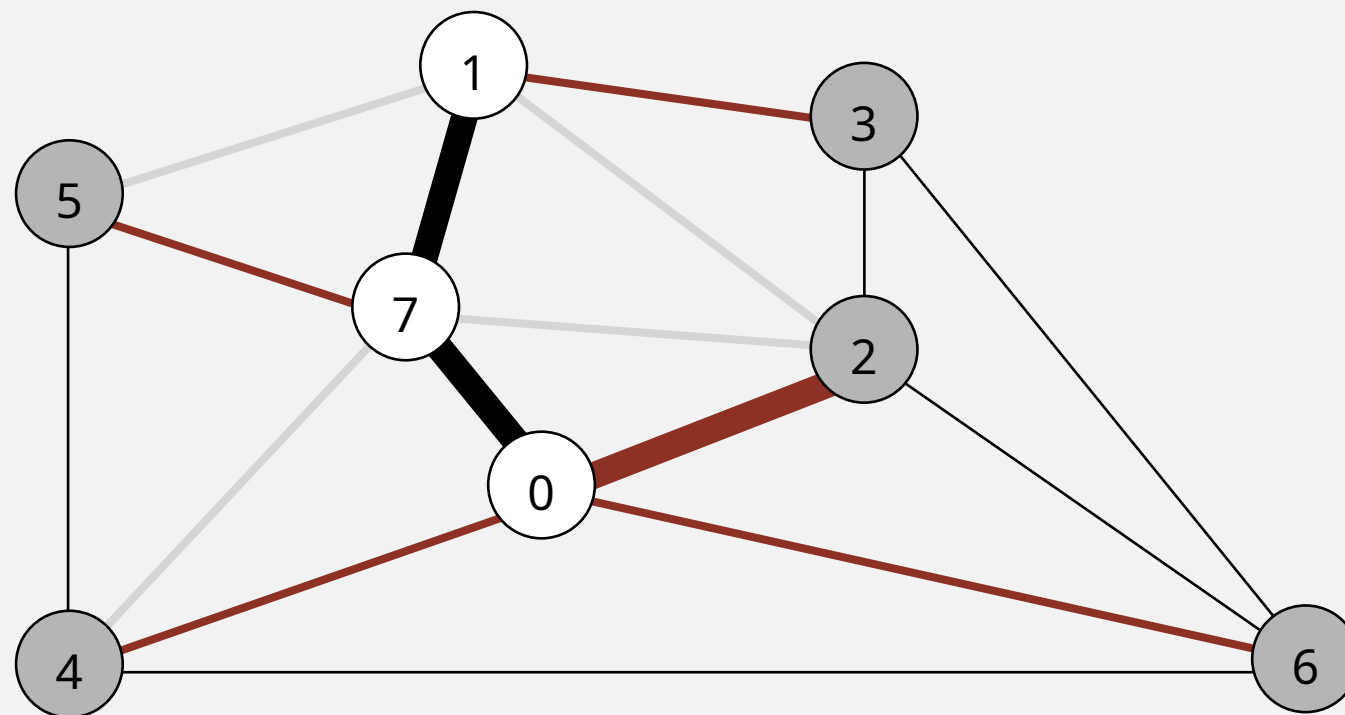
Prim's algorithm: eager implementation demo

- Start with vertex 0 and greedily grow tree T .
- Add to T the min weight edge with exactly one endpoint in T .
- Repeat until $V - 1$ edges.



Prim's algorithm: eager implementation demo

- Start with vertex 0 and greedily grow tree T .
- Add to T the min weight edge with exactly one endpoint in T .
- Repeat until $V - 1$ edges.



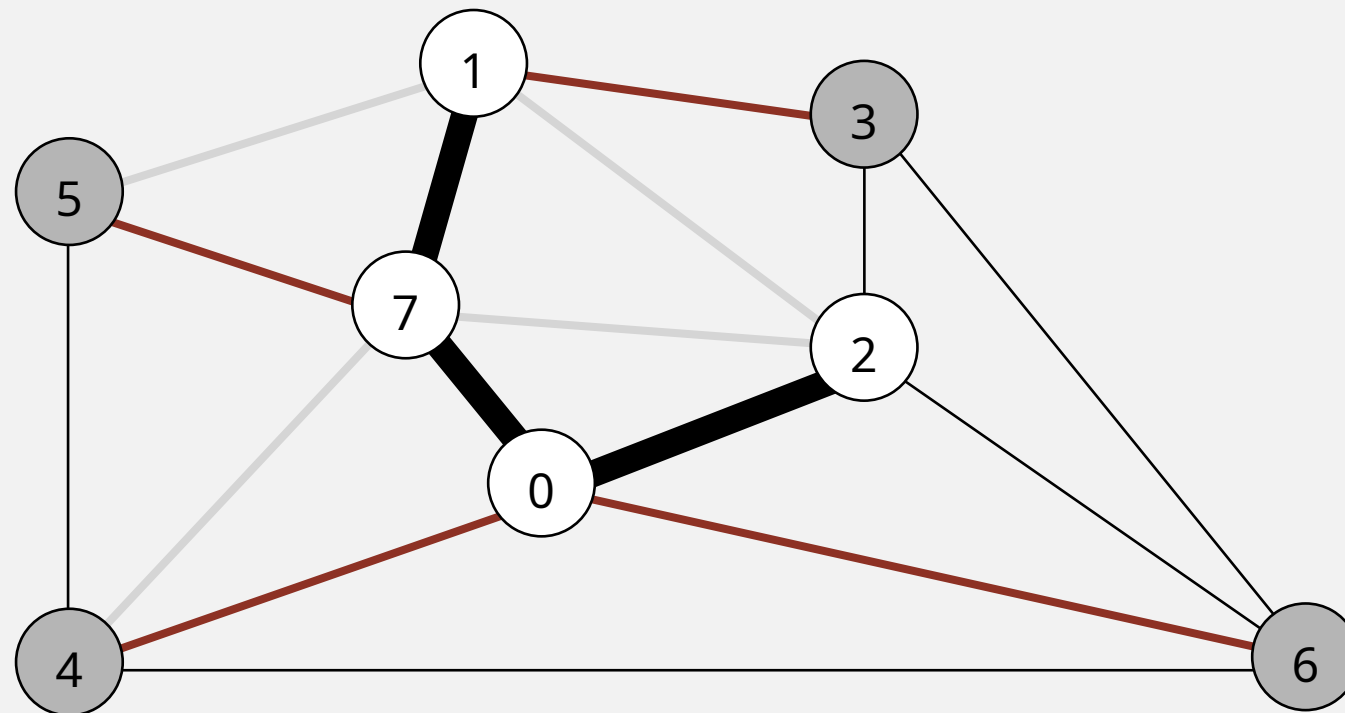
MST edges

0-7 1-7

v	edgeTo[]	distTo[]
0	-	-
7	0-7	0.16
1	1-7	0.19
2	0-2	0.26
5	5-7	0.28
3	1-3	0.29
4	0-4	0.38
6	6-0	0.58

Prim's algorithm: eager implementation demo

- Start with vertex 0 and greedily grow tree T .
- Add to T the min weight edge with exactly one endpoint in T .
- Repeat until $V - 1$ edges.



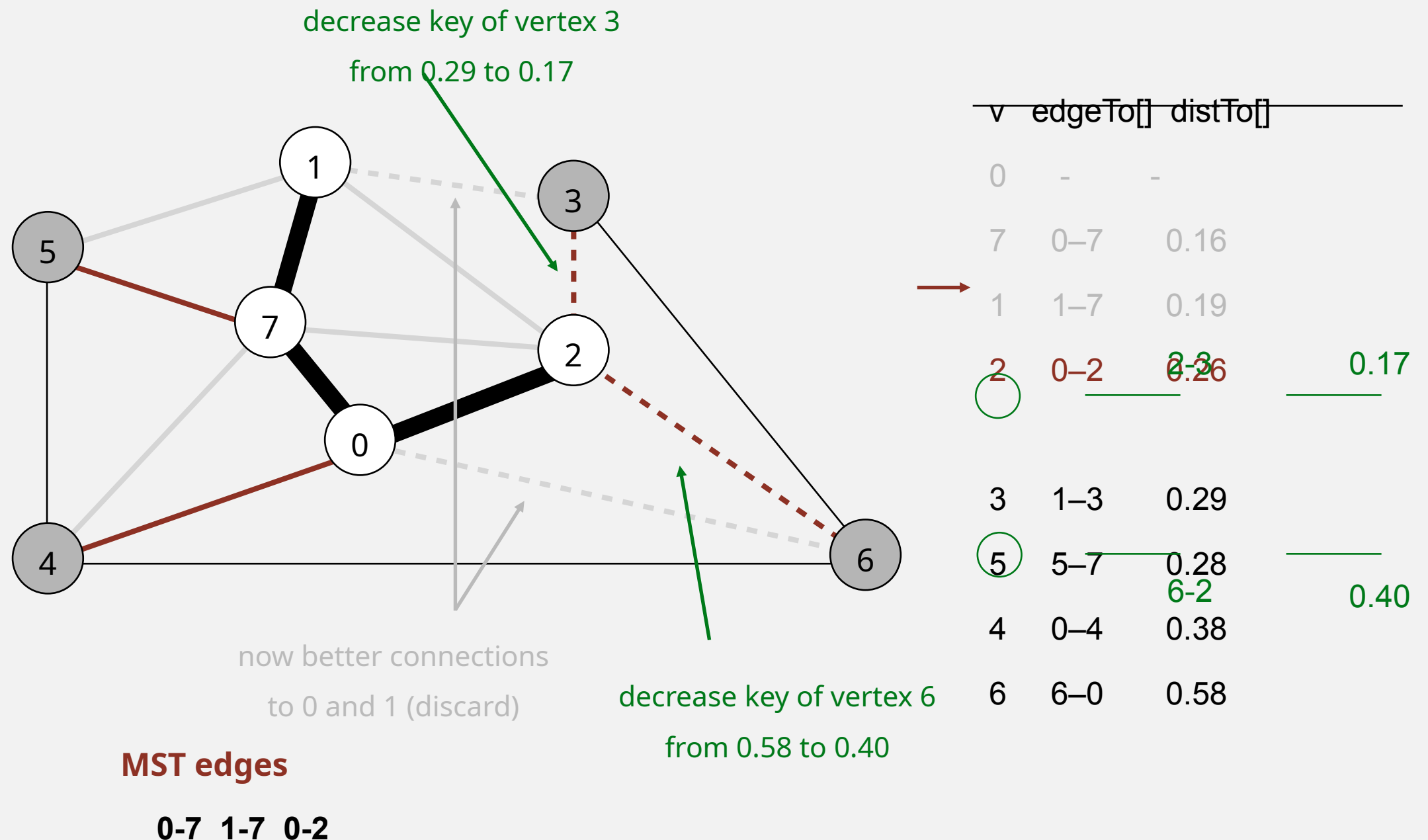
MST edges

0-7 1-7 0-2

v	edgeTo[]	distTo[]
0	-	-
7	0-7	0.16
1	1-7	0.19
2	0-2	0.26
5	5-7	0.28
3	1-3	0.29
4	0-4	0.38
6	6-0	0.58

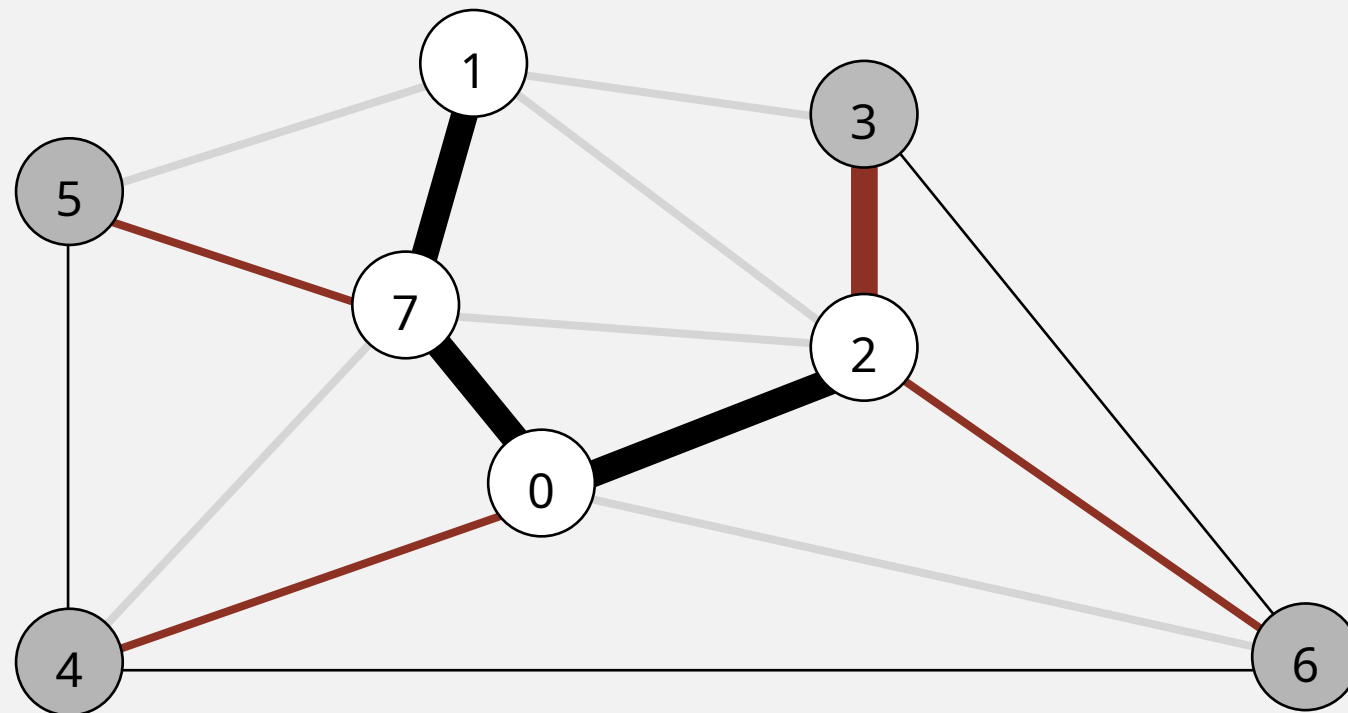
Prim's algorithm: eager implementation demo

- Start with vertex 0 and greedily grow tree T .
- Add to T the min weight edge with exactly one endpoint in T .
- Repeat until $V - 1$ edges.



Prim's algorithm: eager implementation demo

- Start with vertex 0 and greedily grow tree T .
- Add to T the min weight edge with exactly one endpoint in T .
- Repeat until $V - 1$ edges.



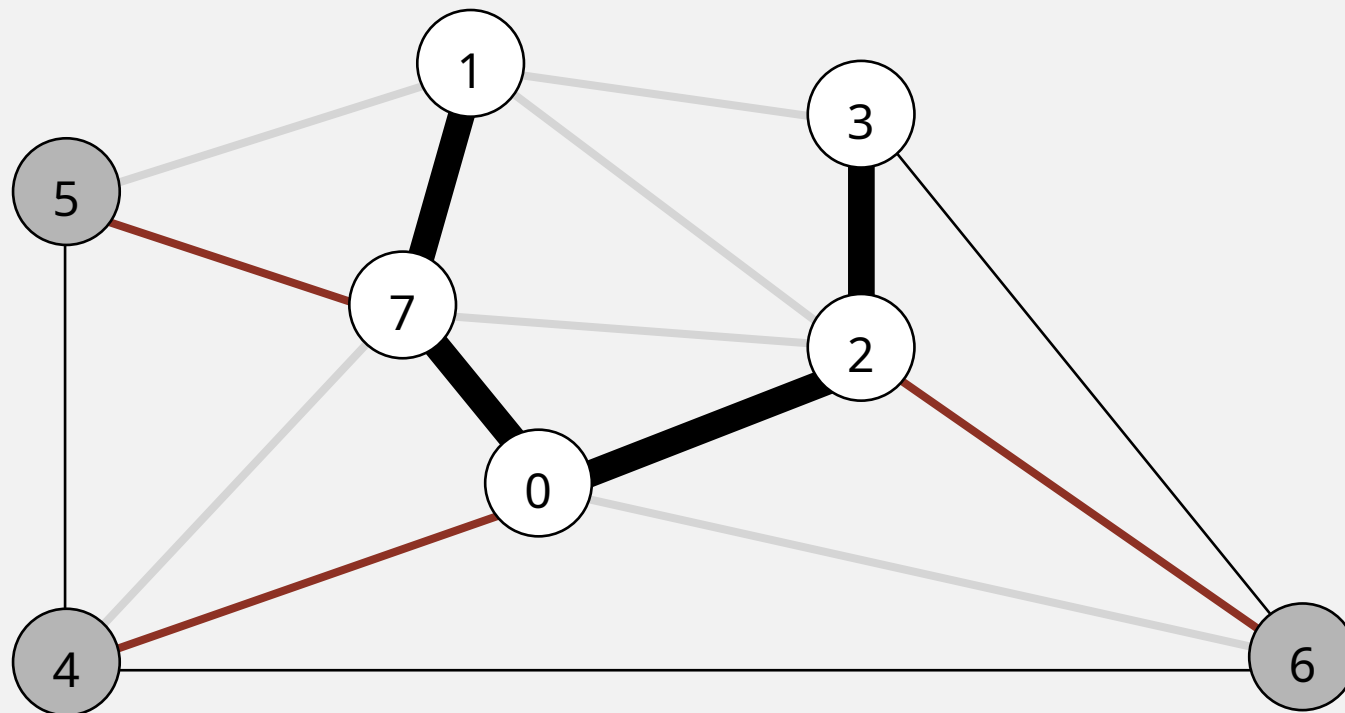
MST edges

0-7 1-7 0-2 2-3

v	edgeTo[]	distTo[]
0	-	-
7	0-7	0.16
1	1-7	0.19
2	0-2	0.26
3	2-3	0.17
5	5-7	0.28
4	0-4	0.38
6	6-2	0.40

Prim's algorithm: eager implementation demo

- Start with vertex 0 and greedily grow tree T .
- Add to T the min weight edge with exactly one endpoint in T .
- Repeat until $V - 1$ edges.



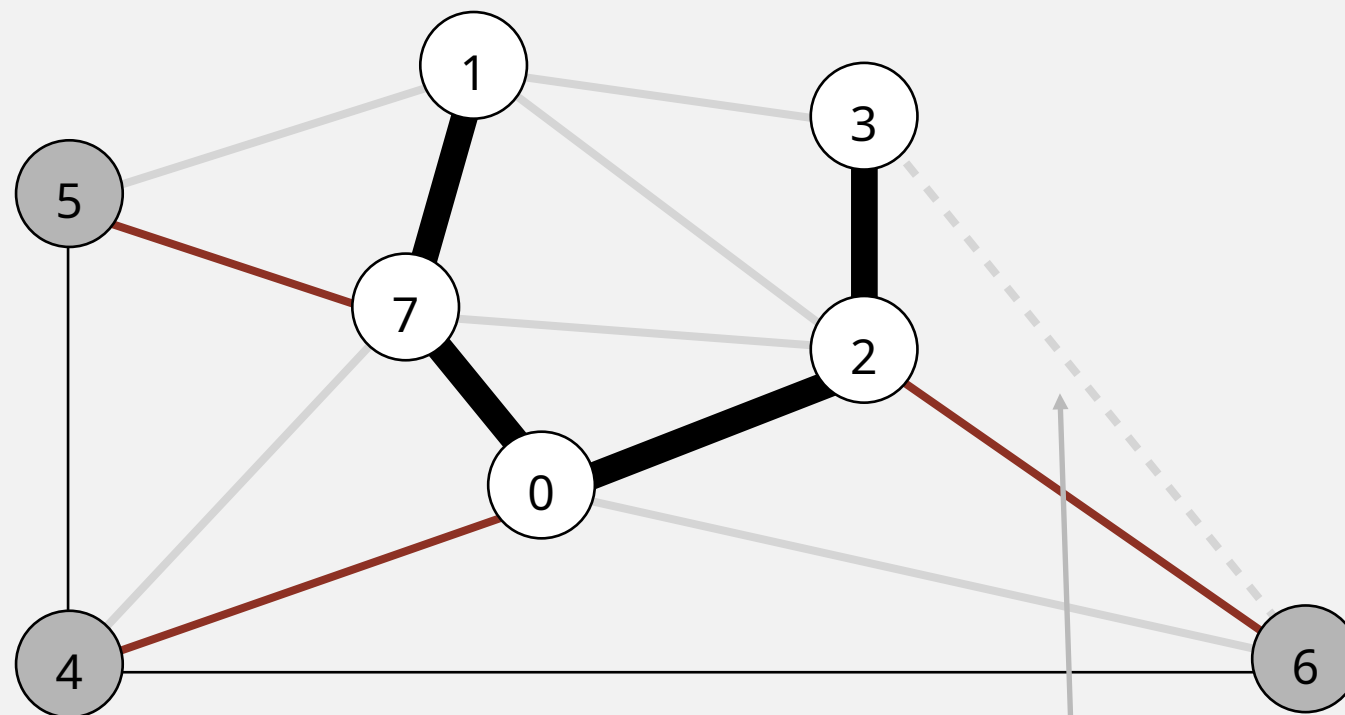
MST edges

0-7 1-7 0-2 2-3

v	edgeTo[]	distTo[]
0	-	-
7	0-7	0.16
1	1-7	0.19
2	0-2	0.26
3	2-3	0.17
5	5-7	0.28
4	0-4	0.38
6	6-2	0.40

Prim's algorithm: eager implementation demo

- Start with vertex 0 and greedily grow tree T .
- Add to T the min weight edge with exactly one endpoint in T .
- Repeat until $V - 1$ edges.



MST edges

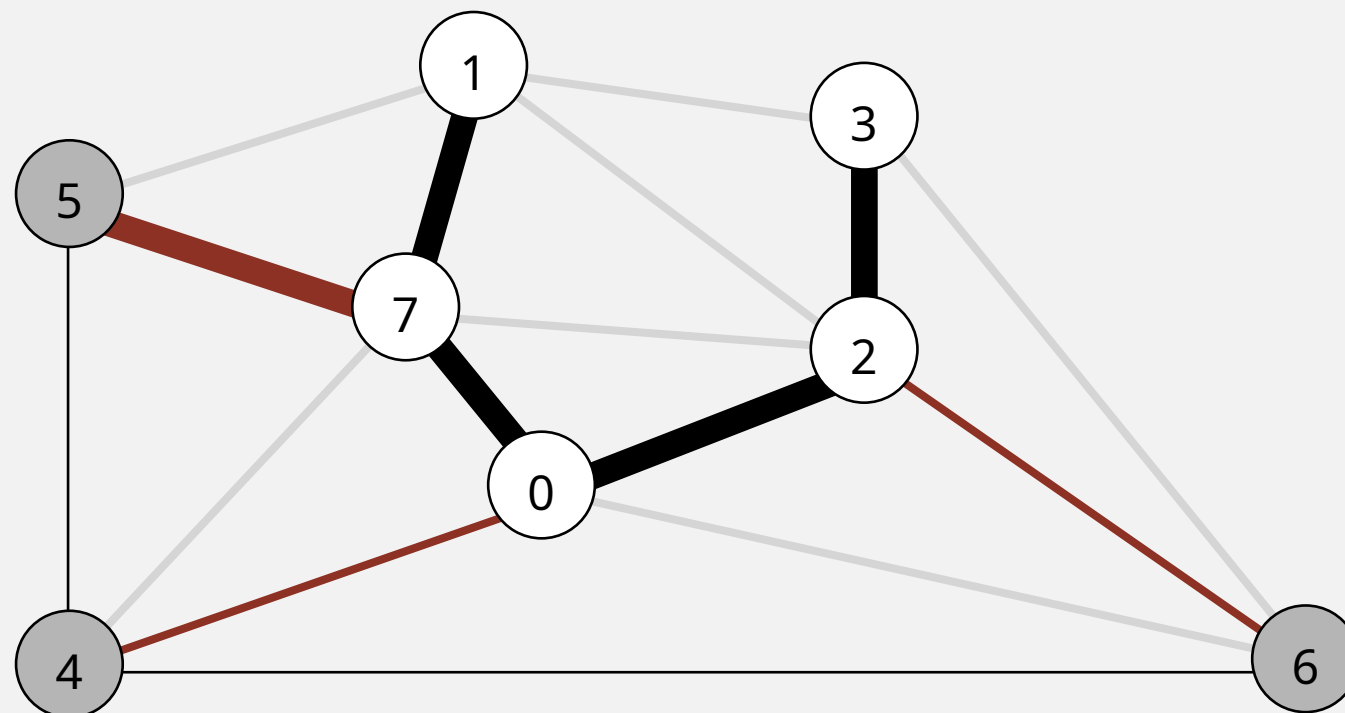
0-7 1-7 0-2 2-3

already a better connection
to 6 (discard)

v	edgeTo[]	distTo[]
0	-	-
7	0-7	0.16
1	1-7	0.19
2	0-2	0.26
3	2-3	0.17
5	5-7	0.28
4	0-4	0.38
6	6-2	0.40

Prim's algorithm: eager implementation demo

- Start with vertex 0 and greedily grow tree T .
- Add to T the min weight edge with exactly one endpoint in T .
- Repeat until $V - 1$ edges.



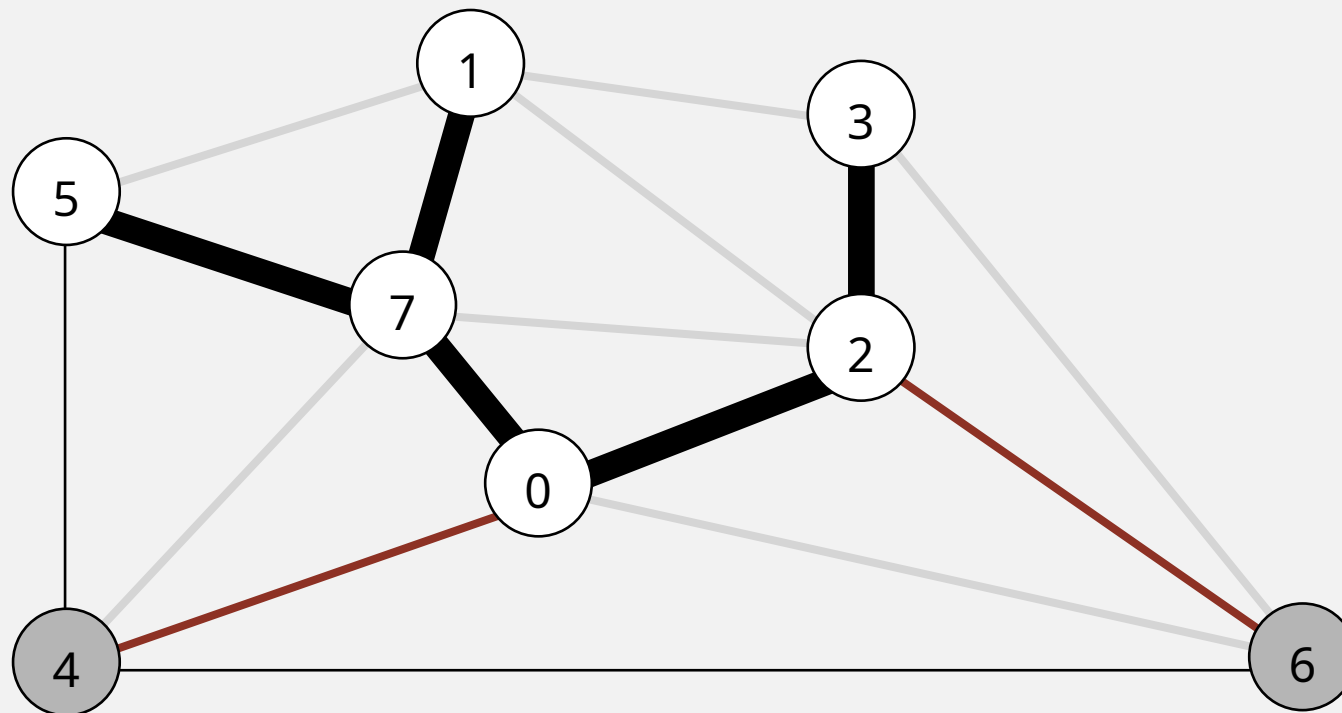
MST edges

0-7 1-7 0-2 2-3

v	edgeTo[]	distTo[]
0	-	-
7	0-7	0.16
1	1-7	0.19
2	0-2	0.26
3	2-3	0.17
5	5-7	0.28
4	0-4	0.38
6	6-2	0.40

Prim's algorithm: eager implementation demo

- Start with vertex 0 and greedily grow tree T .
- Add to T the min weight edge with exactly one endpoint in T .
- Repeat until $V - 1$ edges.



MST edges

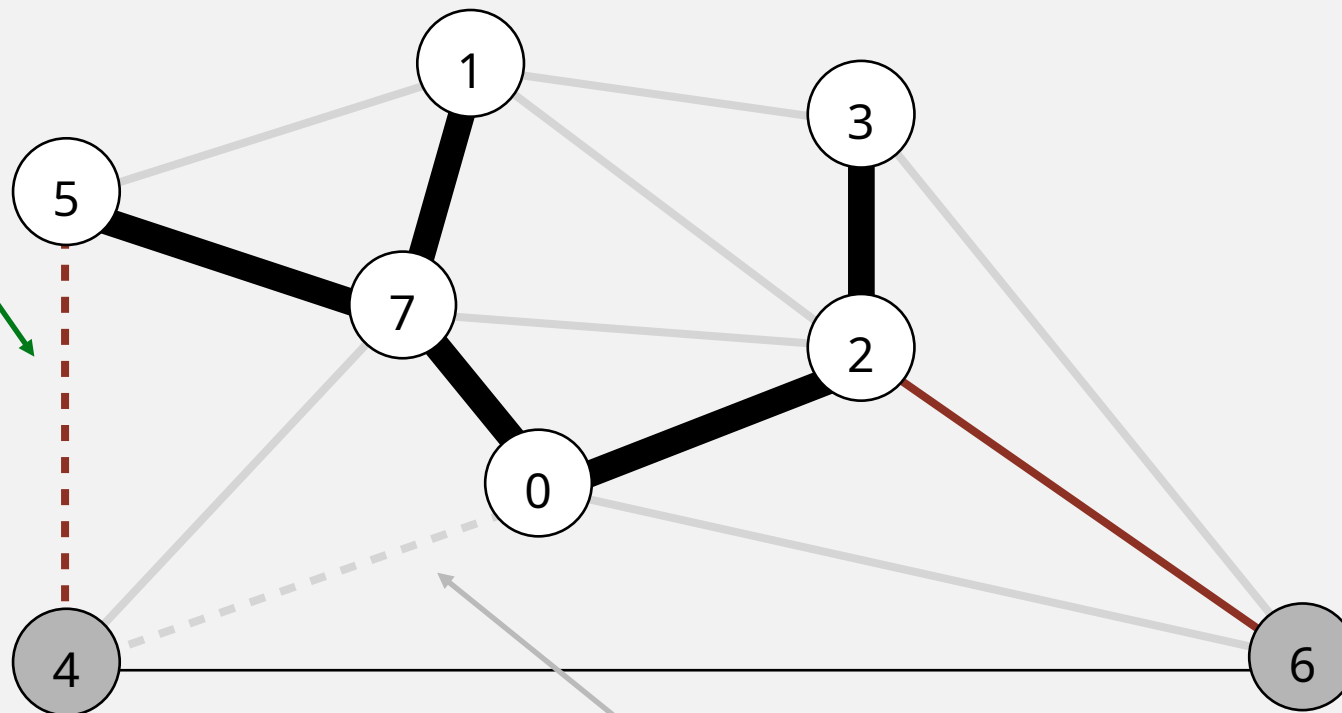
0-7 1-7 0-2 2-3 5-7

v	edgeTo[]	distTo[]
0	-	-
7	0-7	0.16
1	1-7	0.19
2	0-2	0.26
3	2-3	0.17
5	5-7	0.28
4	0-4	0.38
6	6-2	0.40

Prim's algorithm: eager implementation demo

- Start with vertex 0 and greedily grow tree T .
- Add to T the min weight edge with exactly one endpoint in T .
- Repeat until $V - 1$ edges.

decrease key of 4
from 0.38 to 0.35



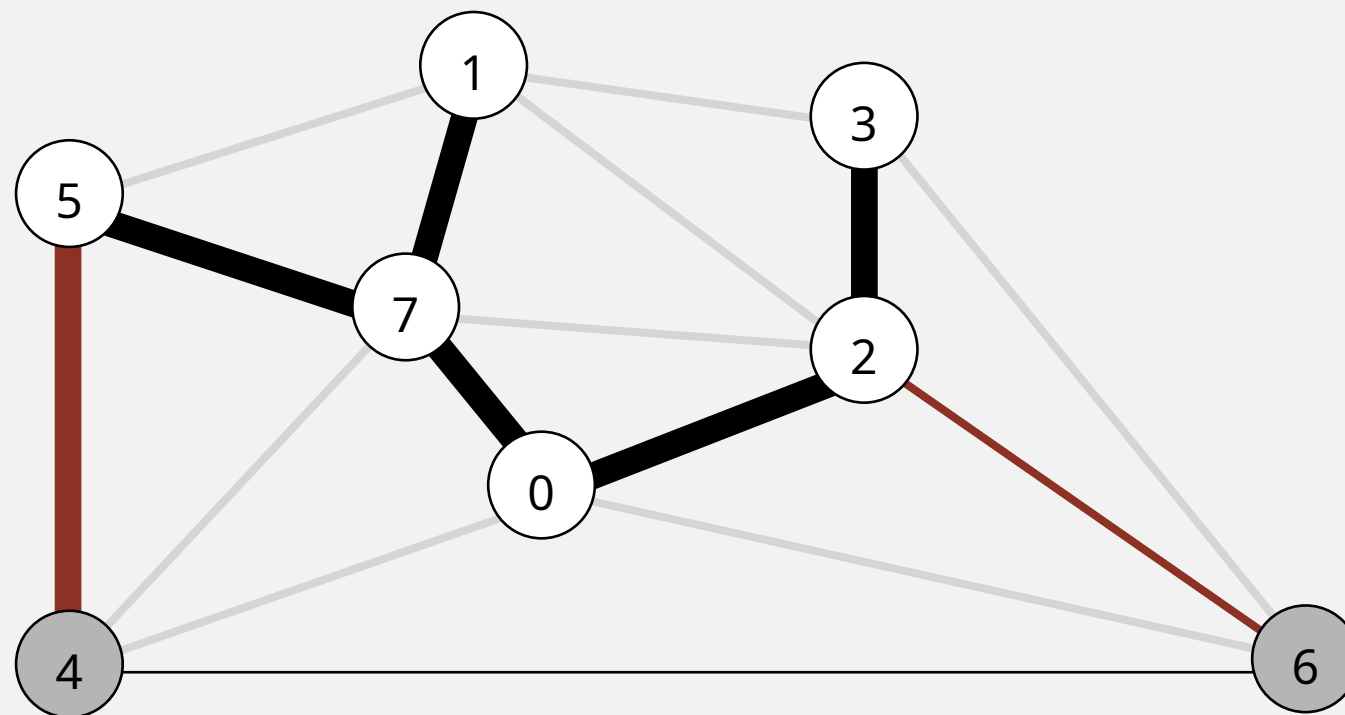
MST edges

0-7 1-7 0-2 2-3 5-7

v	edgeTo[]	distTo[]	
0	-	-	
7	0-7	0.16	
1	1-7	0.19	
2	0-2	0.26	
3	2-3	0.17	
5	5-7	0.28	0.35
4	0-4	0.38	
6	6-2	0.40	

Prim's algorithm: eager implementation demo

- Start with vertex 0 and greedily grow tree T .
- Add to T the min weight edge with exactly one endpoint in T .
- Repeat until $V - 1$ edges.



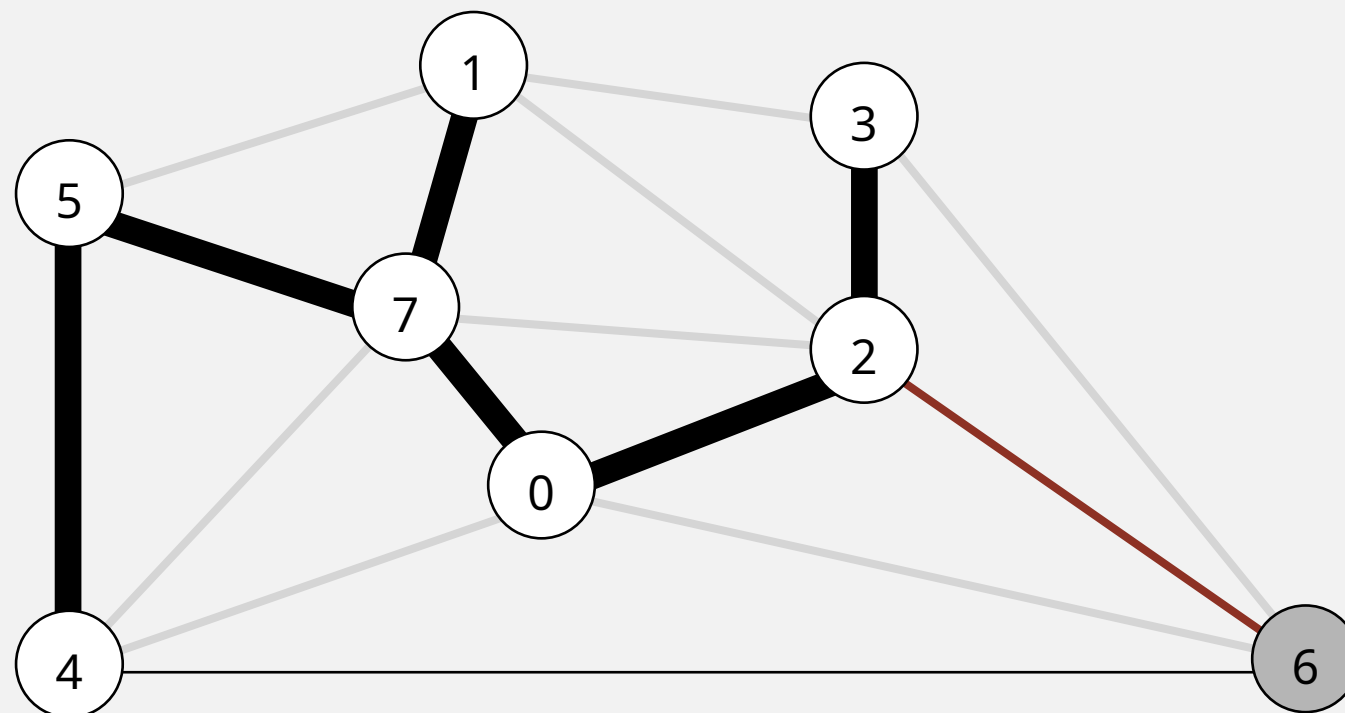
v	edgeTo[]	distTo[]
0	-	-
7	0-7	0.16
1	1-7	0.19
2	0-2	0.26
3	2-3	0.17
5	5-7	0.28
4	4-5	0.35
6	6-2	0.40

MST edges

0-7 1-7 0-2 2-3 5-7

Prim's algorithm: eager implementation demo

- Start with vertex 0 and greedily grow tree T .
- Add to T the min weight edge with exactly one endpoint in T .
- Repeat until $V - 1$ edges.



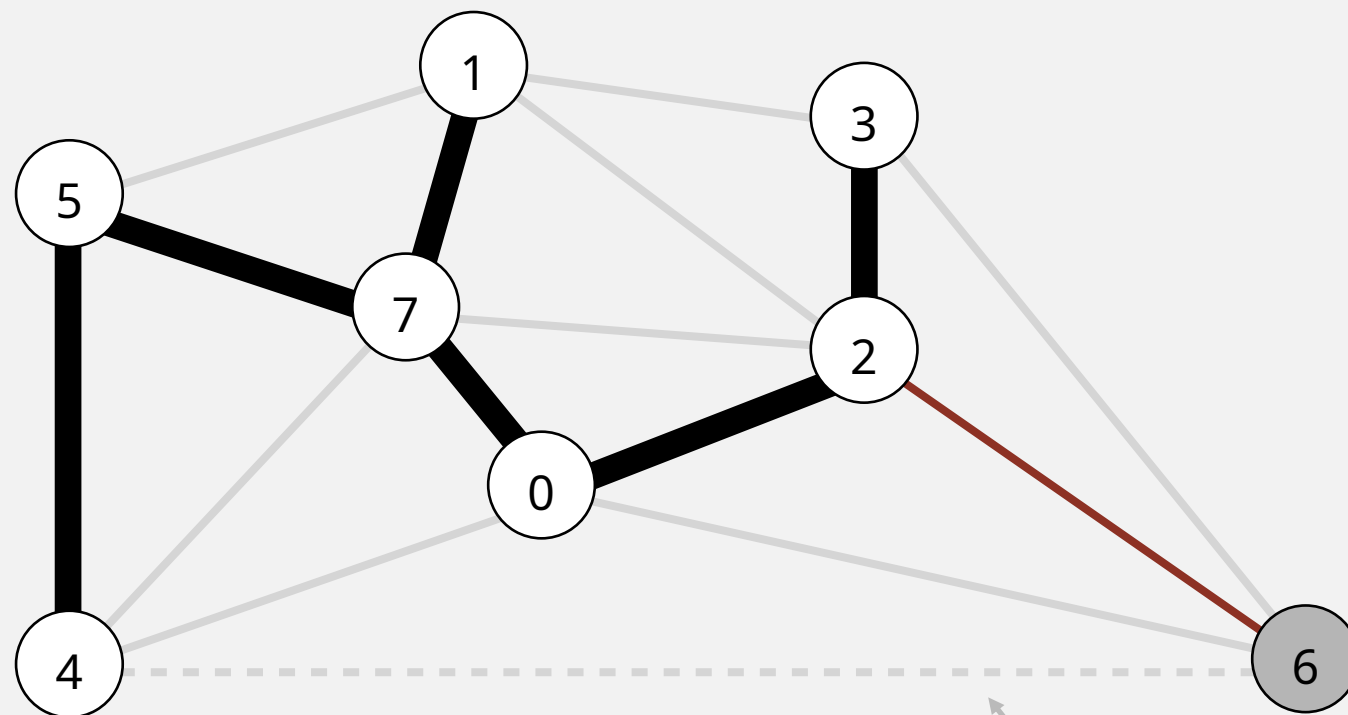
v	edgeTo[]	distTo[]
0	-	-
7	0-7	0.16
1	1-7	0.19
2	0-2	0.26
3	2-3	0.17
5	5-7	0.28
4	4-5	0.35
6	6-2	0.40

MST edges

0-7 1-7 0-2 2-3 5-7 4-5

Prim's algorithm: eager implementation demo

- Start with vertex 0 and greedily grow tree T .
- Add to T the min weight edge with exactly one endpoint in T .
- Repeat until $V - 1$ edges.



MST edges

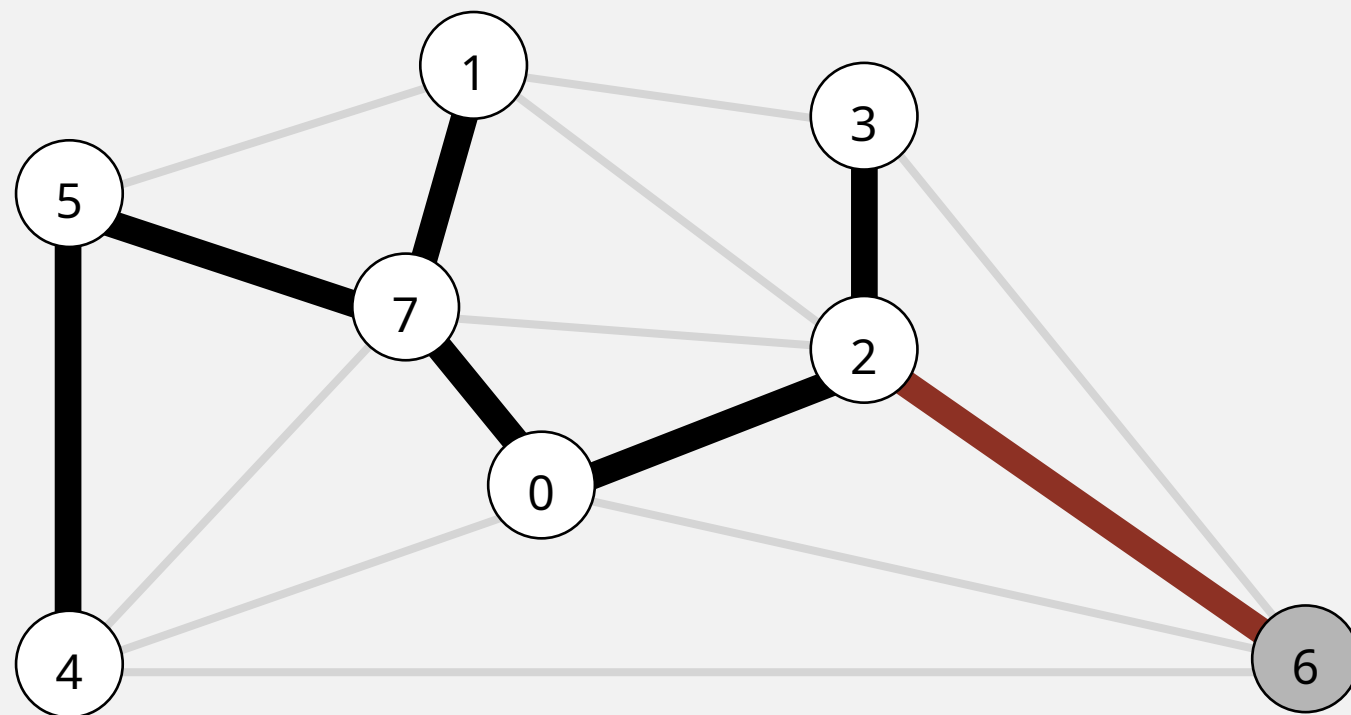
0-7 1-7 0-2 2-3 5-7 4-5

v	edgeTo[]	distTo[]
0	-	-
7	0-7	0.16
1	1-7	0.19
2	0-2	0.26
3	2-3	0.17
5	5-7	0.28
4	4-5	0.35
6	6-2	0.40

already a better connection
to 6 (discard)

Prim's algorithm: eager implementation demo

- Start with vertex 0 and greedily grow tree T .
- Add to T the min weight edge with exactly one endpoint in T .
- Repeat until $V - 1$ edges.



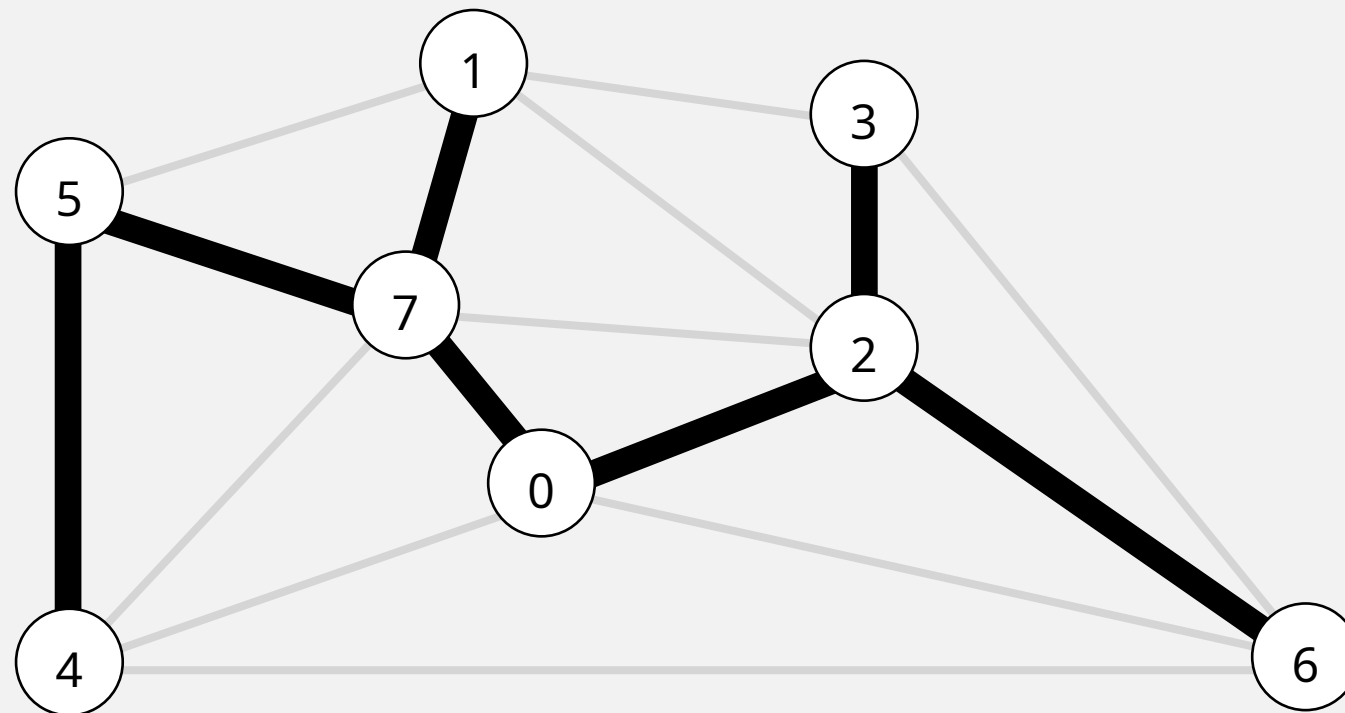
v	edgeTo[]	distTo[]
0	-	-
7	0-7	0.16
1	1-7	0.19
2	0-2	0.26
3	2-3	0.17
5	5-7	0.28
4	4-5	0.35
6	6-2	0.40

MST edges

0-7 1-7 0-2 2-3 5-7 4-5

Prim's algorithm: eager implementation demo

- Start with vertex 0 and greedily grow tree T .
- Add to T the min weight edge with exactly one endpoint in T .
- Repeat until $V - 1$ edges.



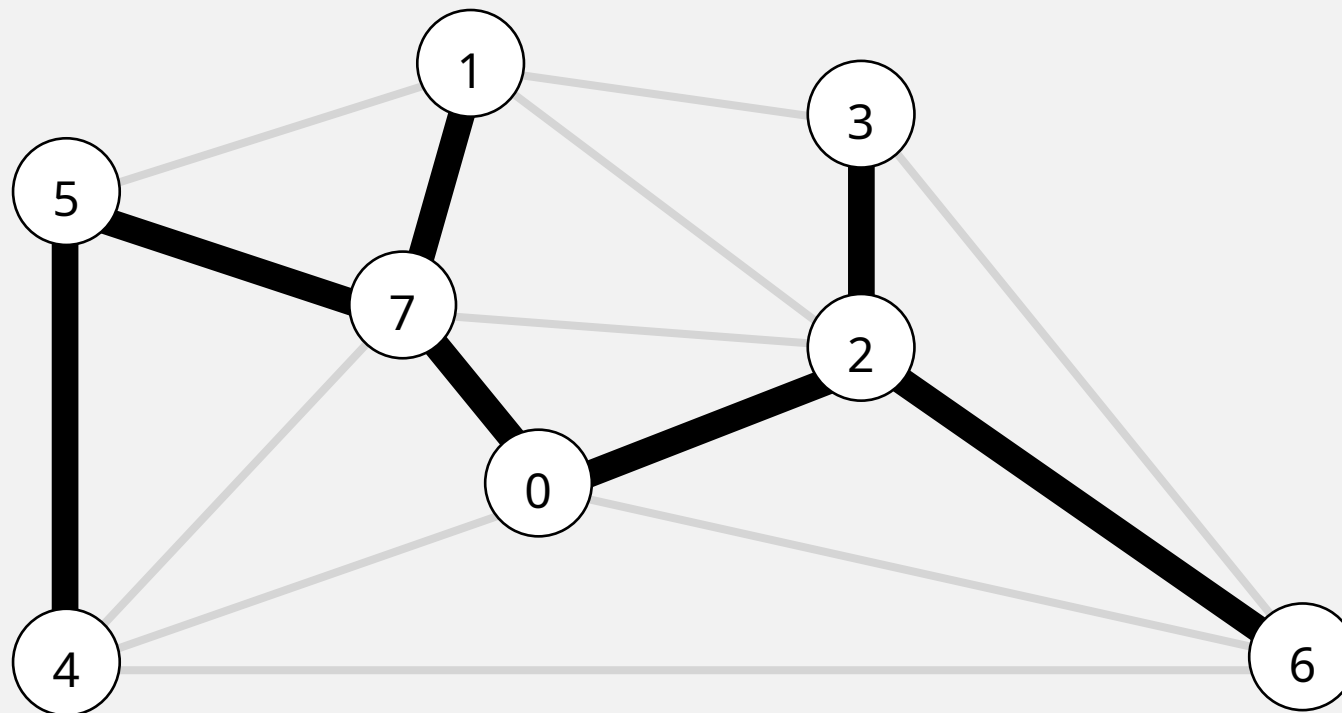
v	edgeTo[]	distTo[]
0	-	-
7	0-7	0.16
1	1-7	0.19
2	0-2	0.26
3	2-3	0.17
5	5-7	0.28
4	4-5	0.35
6	6-2	0.40

MST edges

0-7 1-7 0-2 2-3 5-7 4-5 6-2

Prim's algorithm: eager implementation demo

- Start with vertex 0 and greedily grow tree T .
- Add to T the min weight edge with exactly one endpoint in T .
- Repeat until $V - 1$ edges.



MST edges

0-7 1-7 0-2 2-3 5-7 4-5 6-2

v	edgeTo[]	distTo[]
0	-	-
7	0-7	0.16
1	1-7	0.19
2	0-2	0.26
3	2-3	0.17
5	5-7	0.28
4	4-5	0.35
6	6-2	0.40