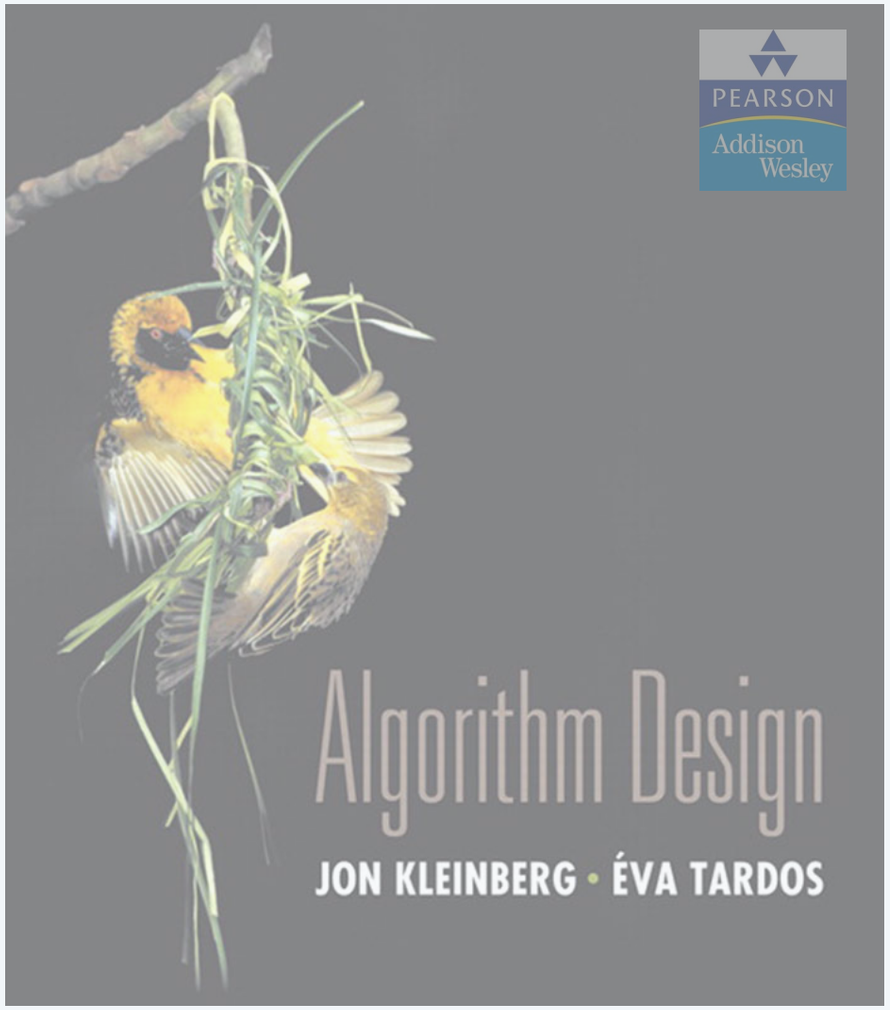


SECTION 4.5

4. GREEDY ALGORITHMS II

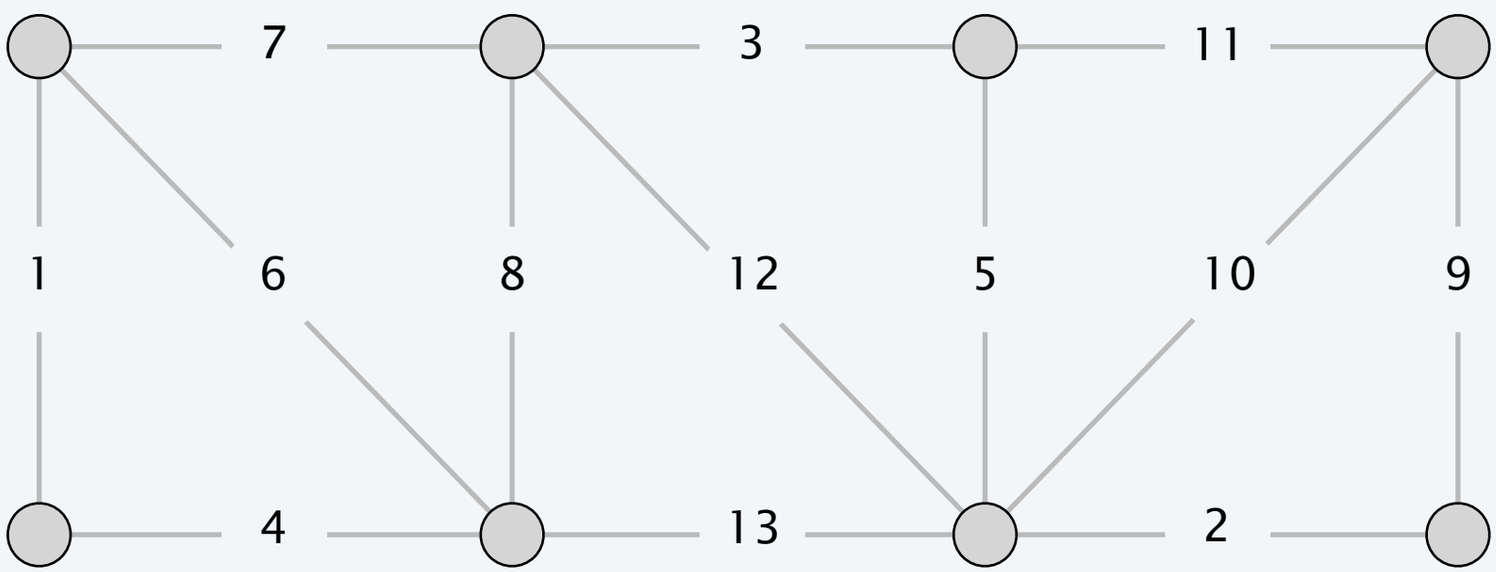
- *red-rule blue-rule demo*
- *Prim's algorithm demo*
- *Kruskal's algorithm demo*
- *Boruvka's algorithm demo*

Prim's algorithm demo

Initialize S = any node, $T = \emptyset$.

Repeat $n - 1$ times:

- Add to T a min-weight edge with one endpoint in S.
- Add new node to S.

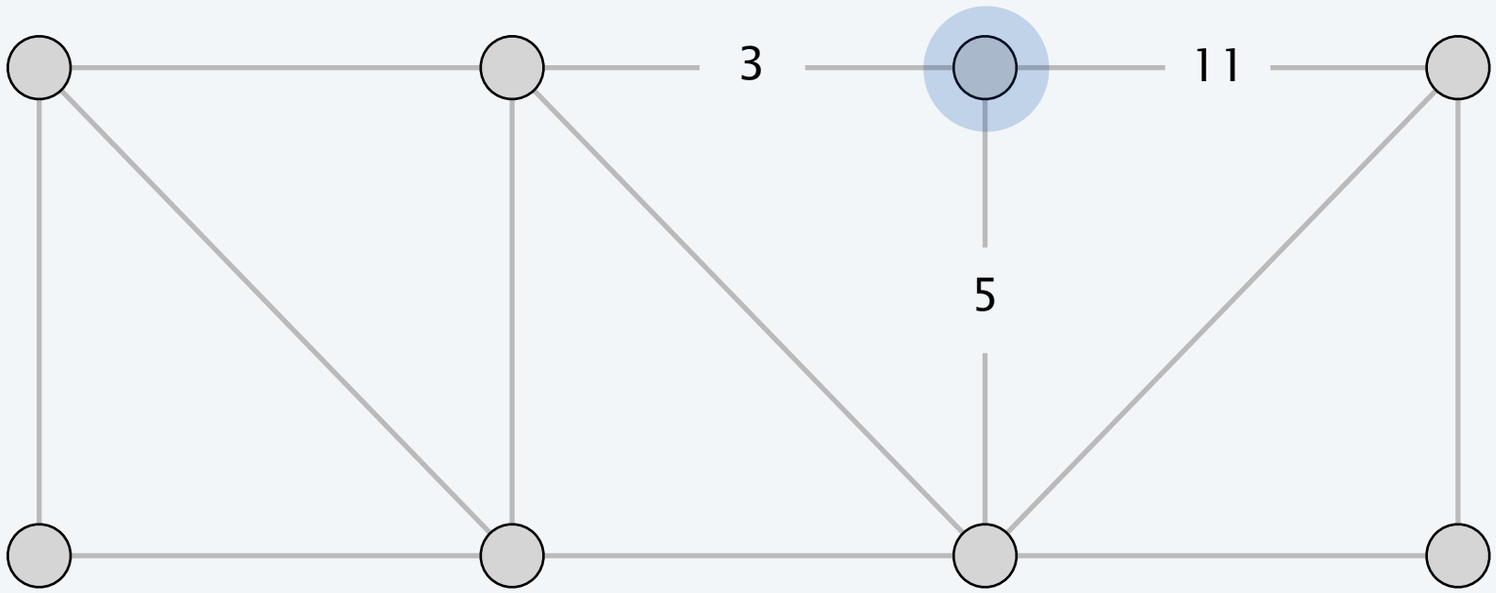


Prim's algorithm demo

Initialize S = any node, $T = \emptyset$.

Repeat $n - 1$ times:

- Add to T a min-weight edge with one endpoint in S.
- Add new node to S.

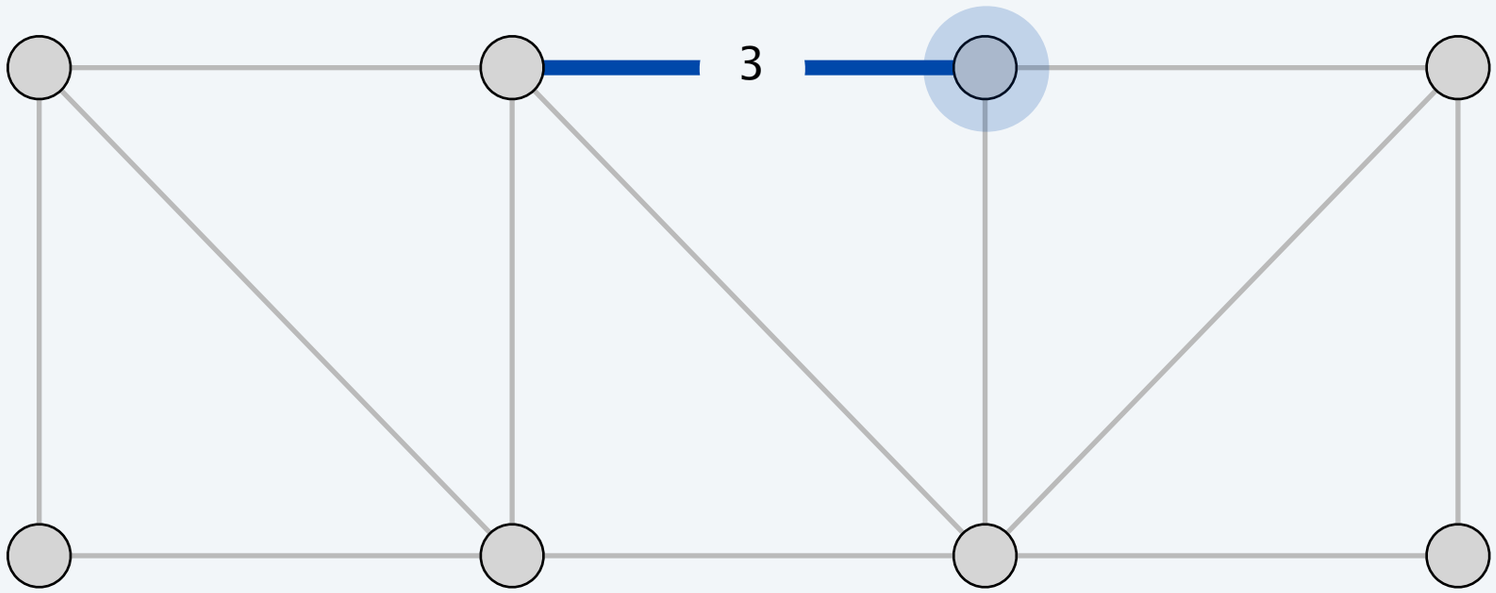


Prim's algorithm demo

Initialize S = any node, $T = \emptyset$.

Repeat $n - 1$ times:

- Add to T a min-weight edge with one endpoint in S.
- Add new node to S.

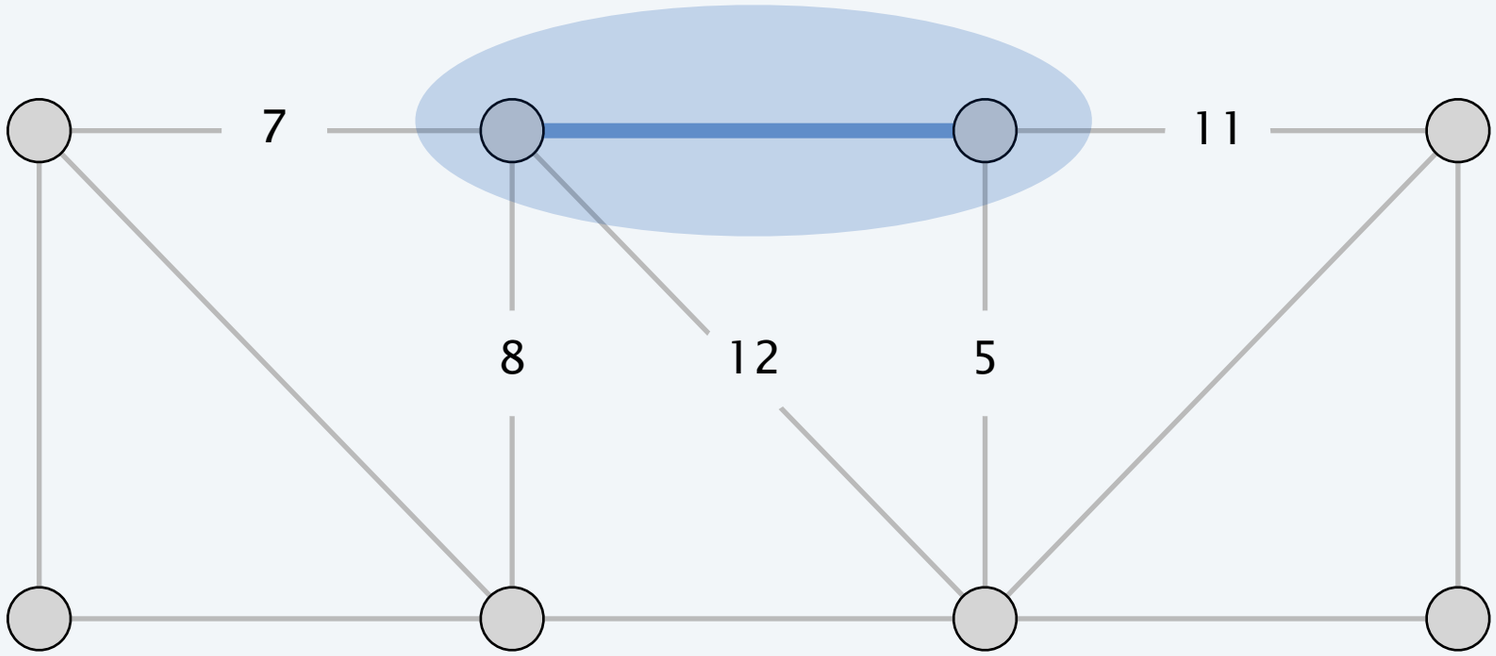


Prim's algorithm demo

Initialize S = any node, $T = \emptyset$.

Repeat $n - 1$ times:

- Add to T a min-weight edge with one endpoint in S.
- Add new node to S.

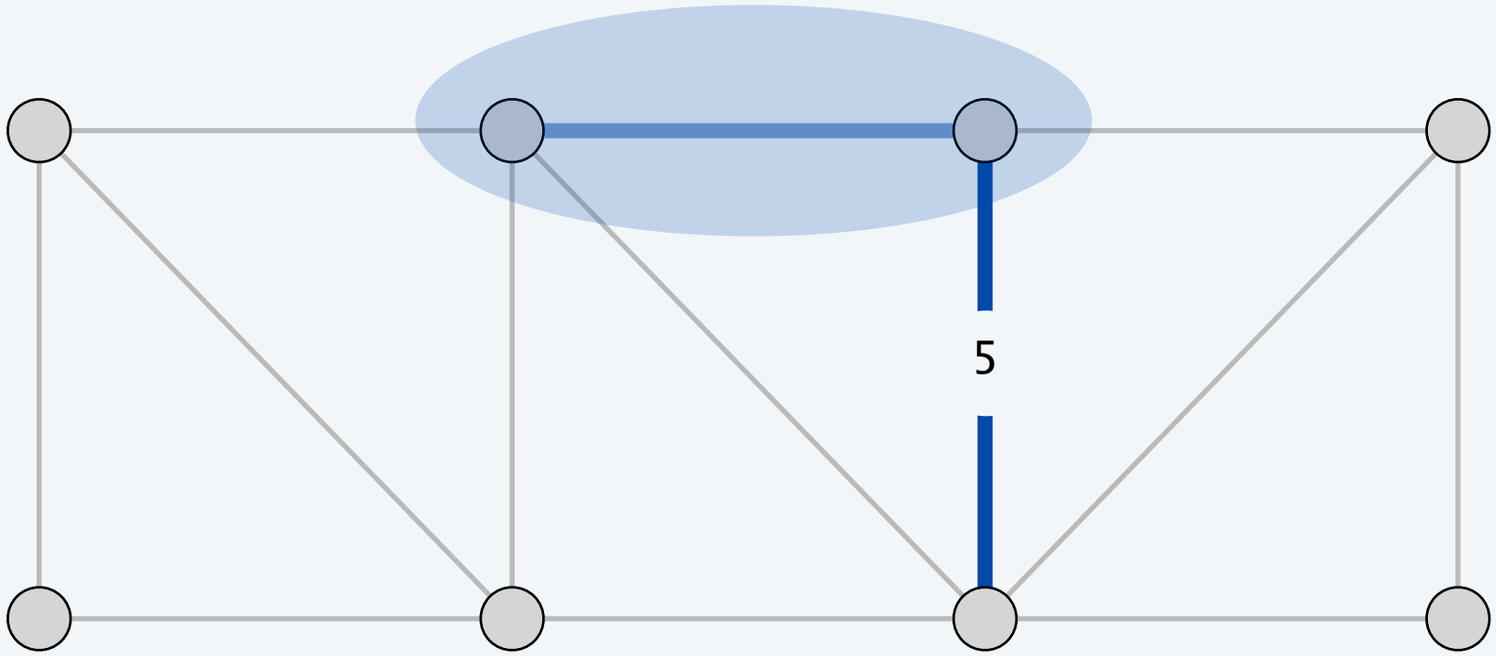


Prim's algorithm demo

Initialize S = any node, $T = \emptyset$.

Repeat $n - 1$ times:

- Add to T a min-weight edge with one endpoint in S.
- Add new node to S.

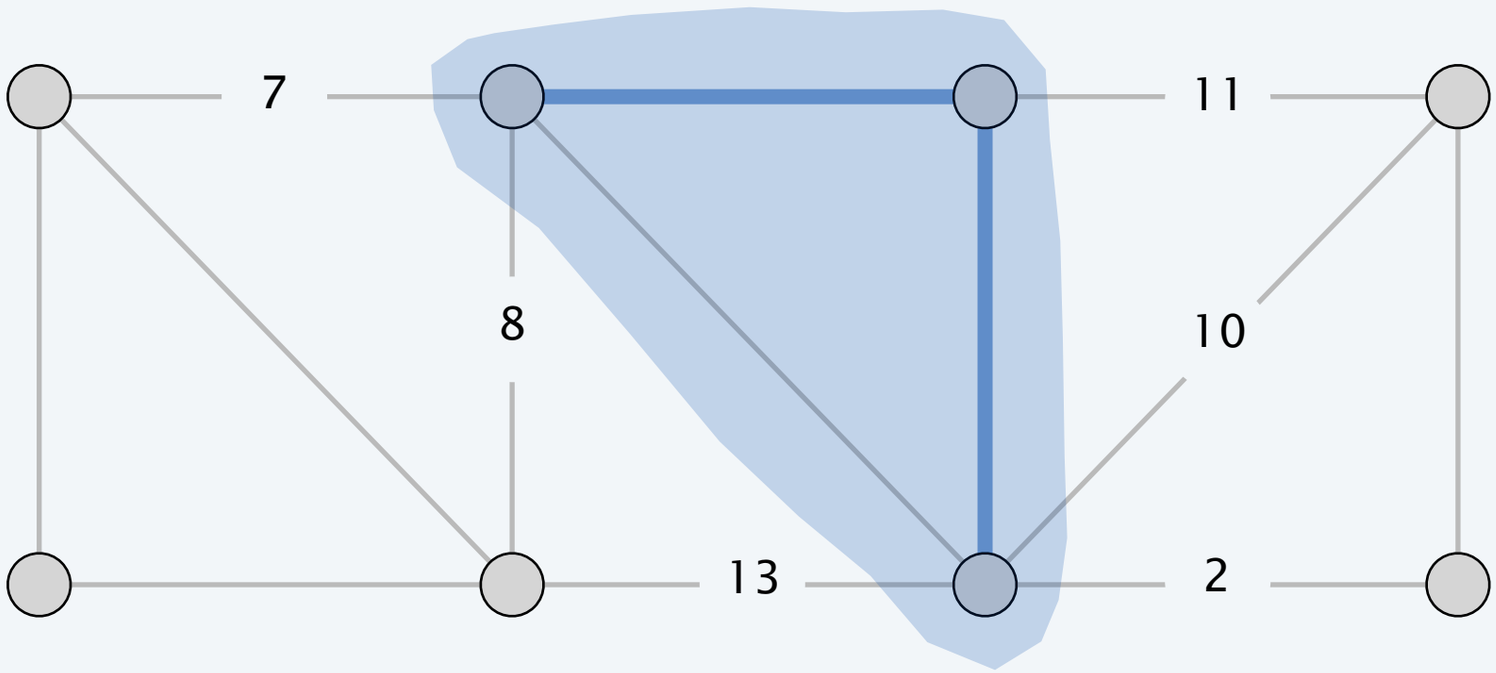


Prim's algorithm demo

Initialize S = any node, $T = \emptyset$.

Repeat $n - 1$ times:

- Add to T a min-weight edge with one endpoint in S.
- Add new node to S.

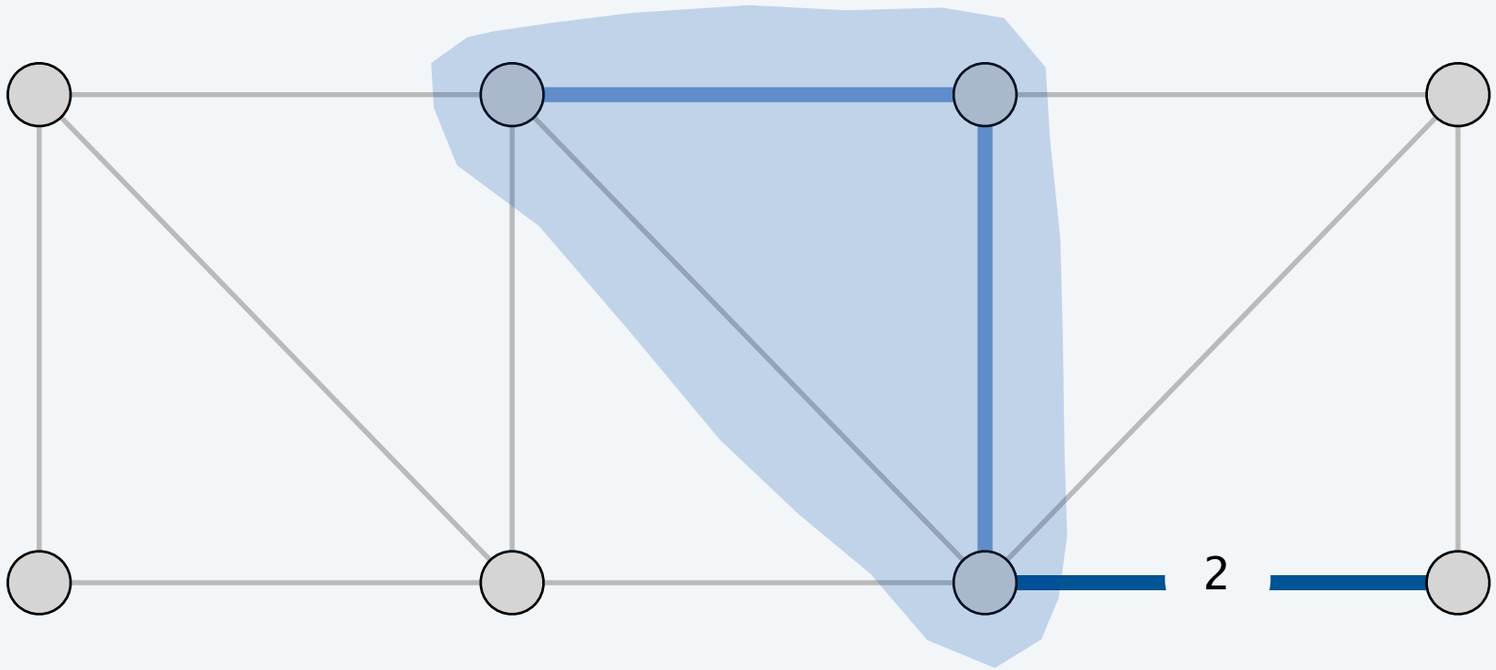


Prim's algorithm demo

Initialize S = any node, $T = \emptyset$.

Repeat $n - 1$ times:

- Add to T a min-weight edge with one endpoint in S.
- Add new node to S.

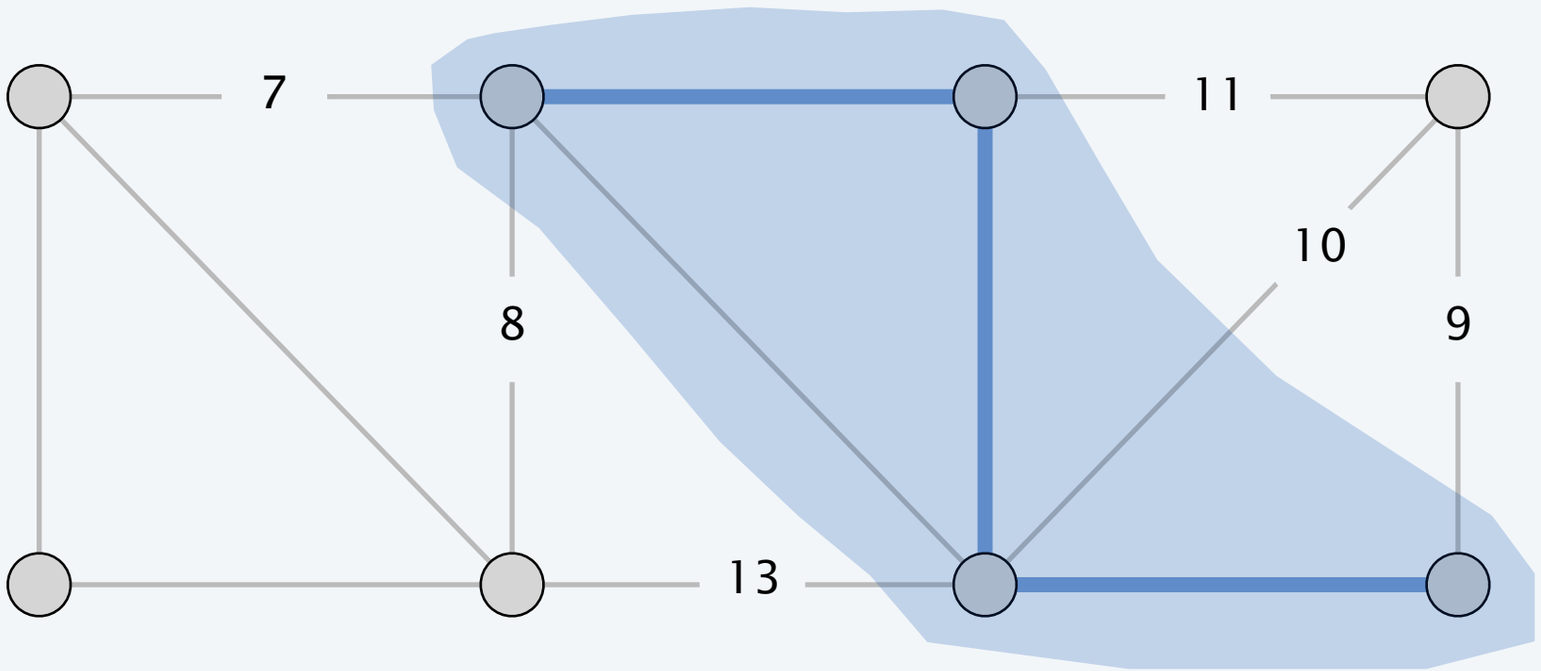


Prim's algorithm demo

Initialize S = any node, $T = \emptyset$.

Repeat $n - 1$ times:

- Add to T a min-weight edge with one endpoint in S.
- Add new node to S.

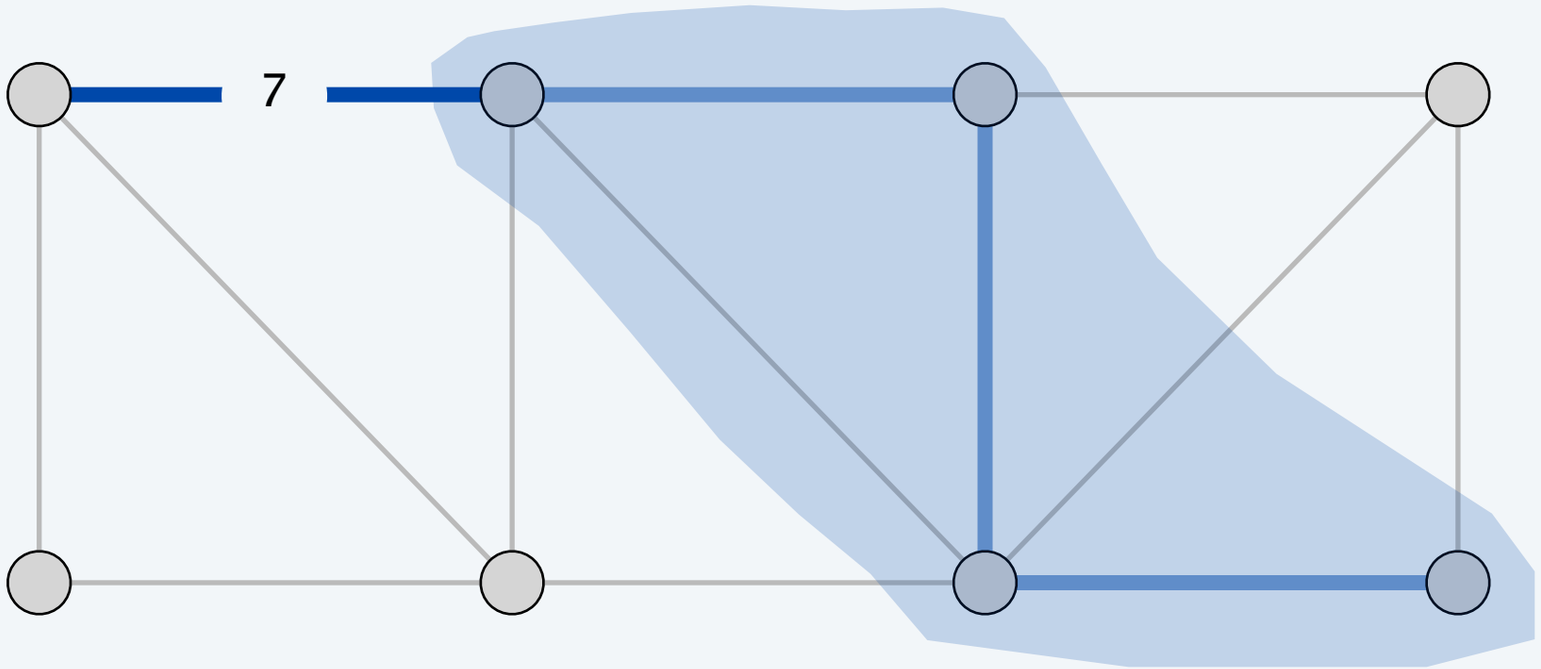


Prim's algorithm demo

Initialize S = any node, $T = \emptyset$.

Repeat $n - 1$ times:

- Add to T a min-weight edge with one endpoint in S.
- Add new node to S.

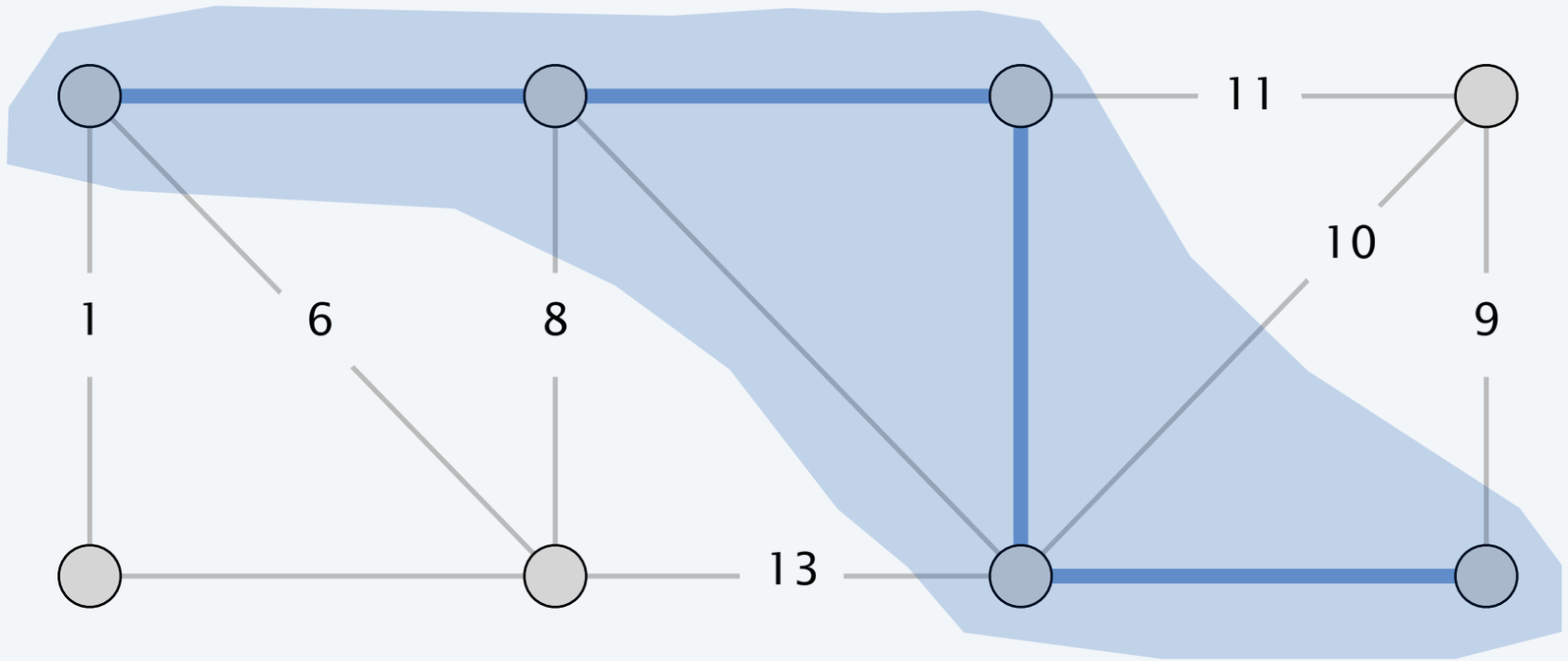


Prim's algorithm demo

Initialize S = any node, $T = \emptyset$.

Repeat $n - 1$ times:

- Add to T a min-weight edge with one endpoint in S.
- Add new node to S.

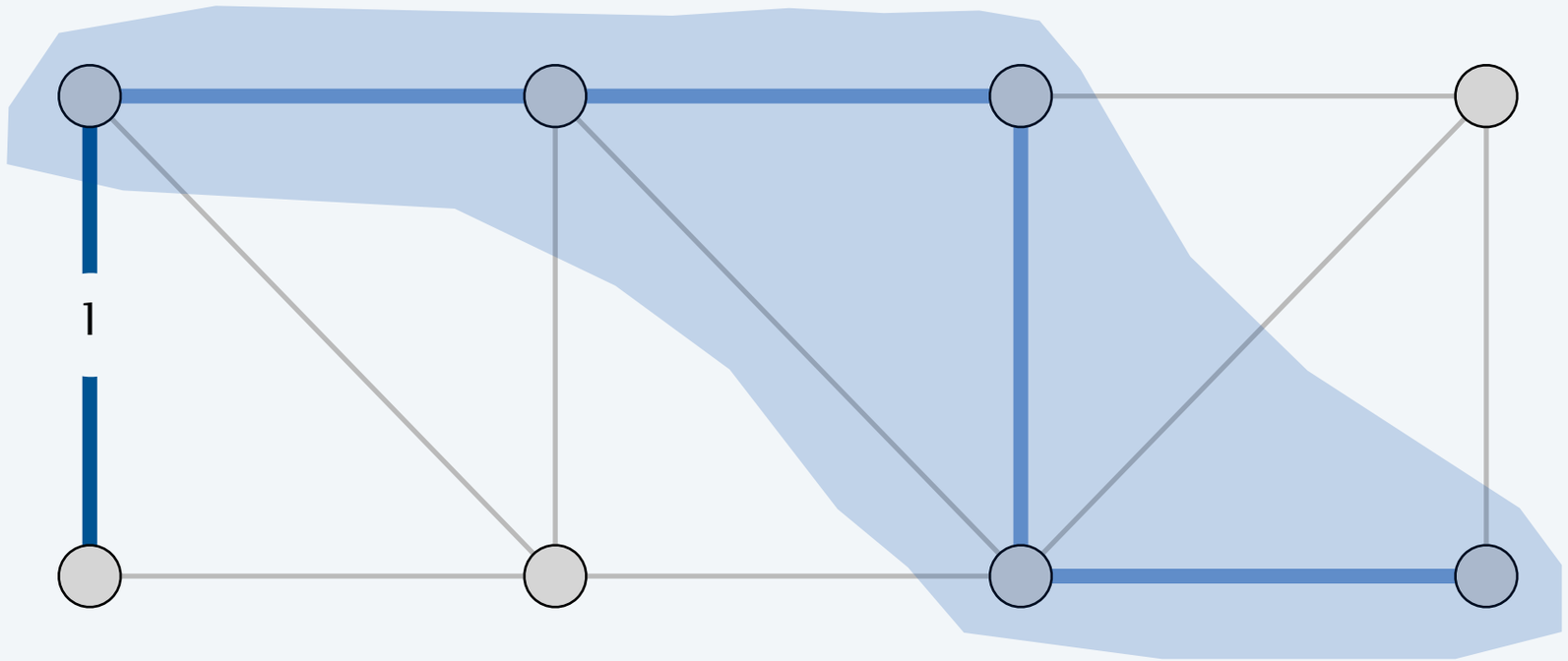


Prim's algorithm demo

Initialize S = any node, $T = \emptyset$.

Repeat $n - 1$ times:

- Add to T a min-weight edge with one endpoint in S.
- Add new node to S.

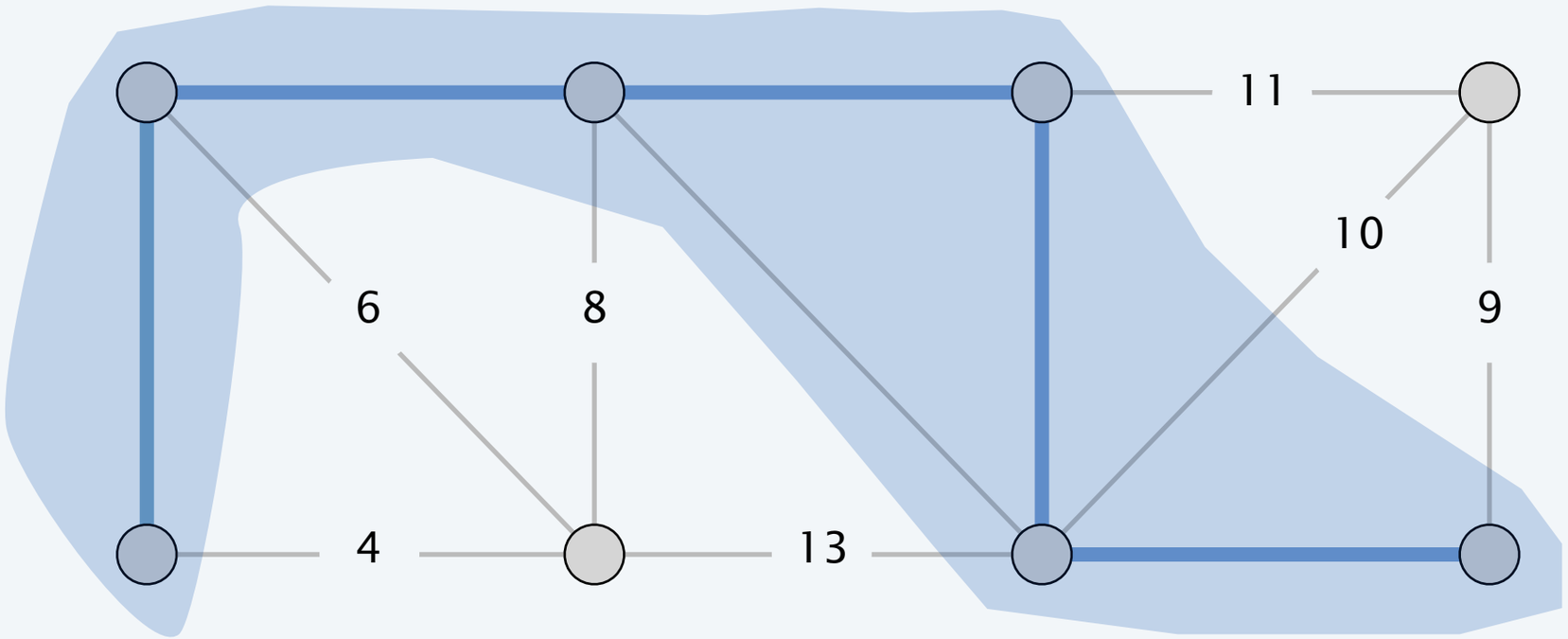


Prim's algorithm demo

Initialize S = any node, $T = \emptyset$.

Repeat $n - 1$ times:

- Add to T a min-weight edge with one endpoint in S.
- Add new node to S.

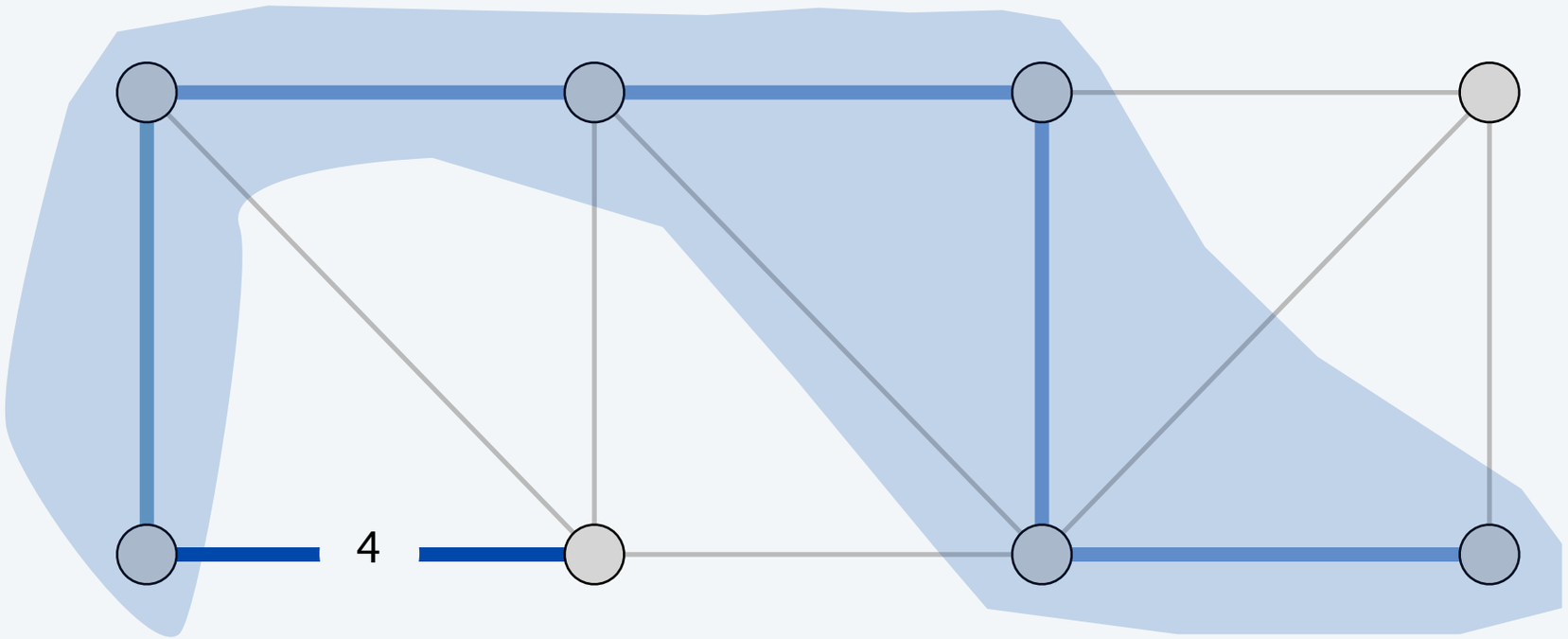


Prim's algorithm demo

Initialize S = any node, $T = \emptyset$.

Repeat $n - 1$ times:

- Add to T a min-weight edge with one endpoint in S.
- Add new node to S.

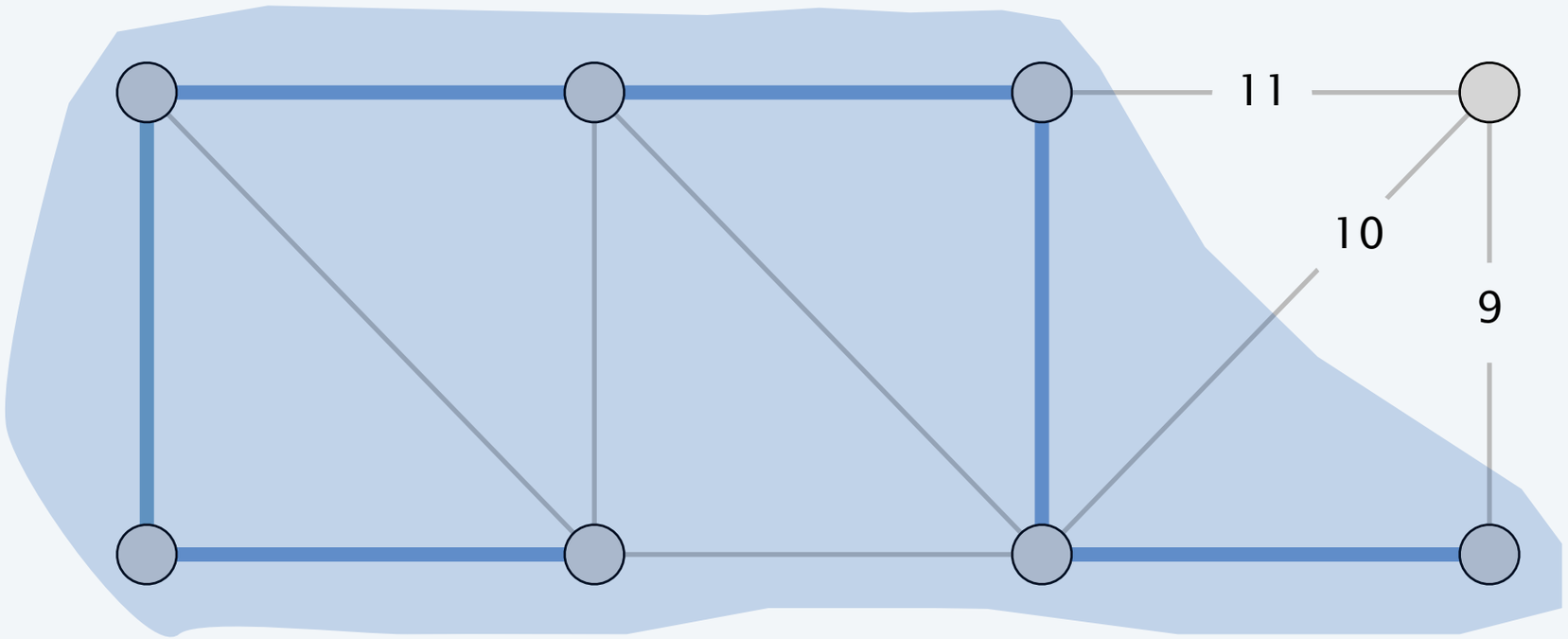


Prim's algorithm demo

Initialize S = any node, $T = \emptyset$.

Repeat $n - 1$ times:

- Add to T a min-weight edge with one endpoint in S.
- Add new node to S.

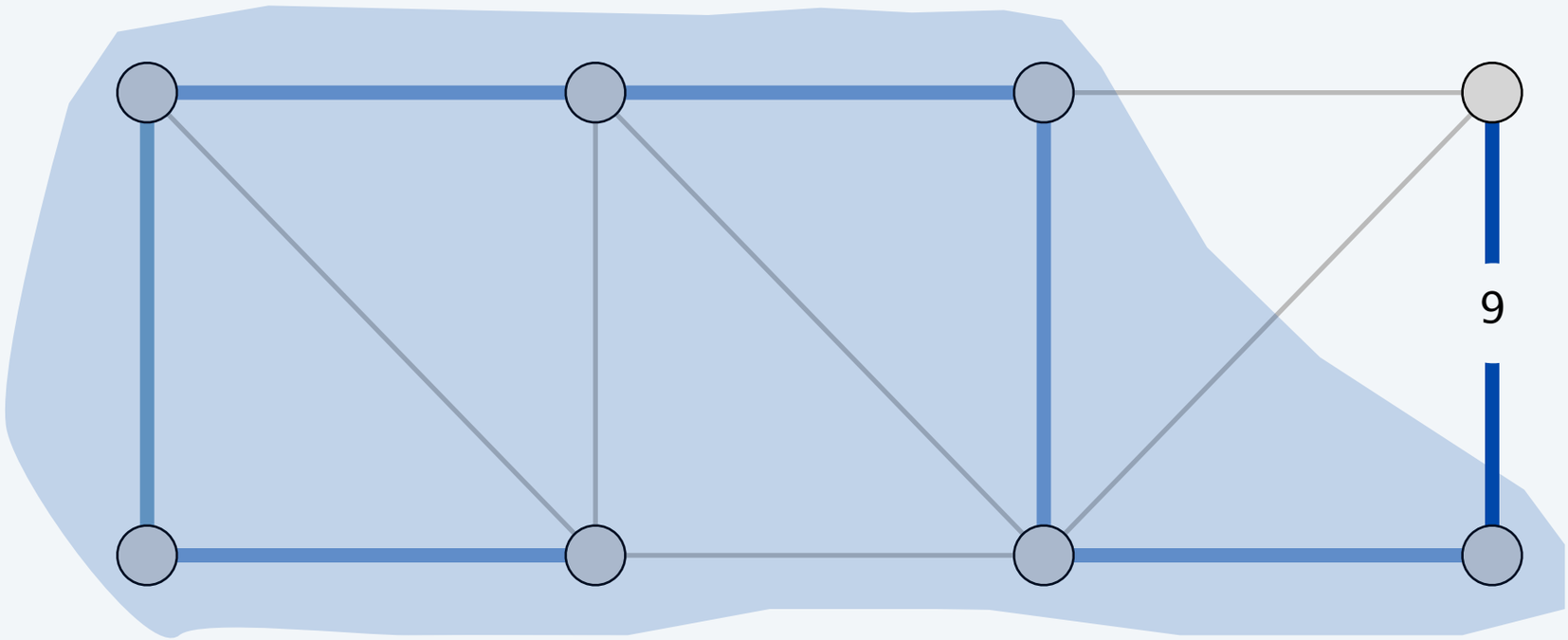


Prim's algorithm demo

Initialize S = any node, $T = \emptyset$.

Repeat $n - 1$ times:

- Add to T a min-weight edge with one endpoint in S.
- Add new node to S.

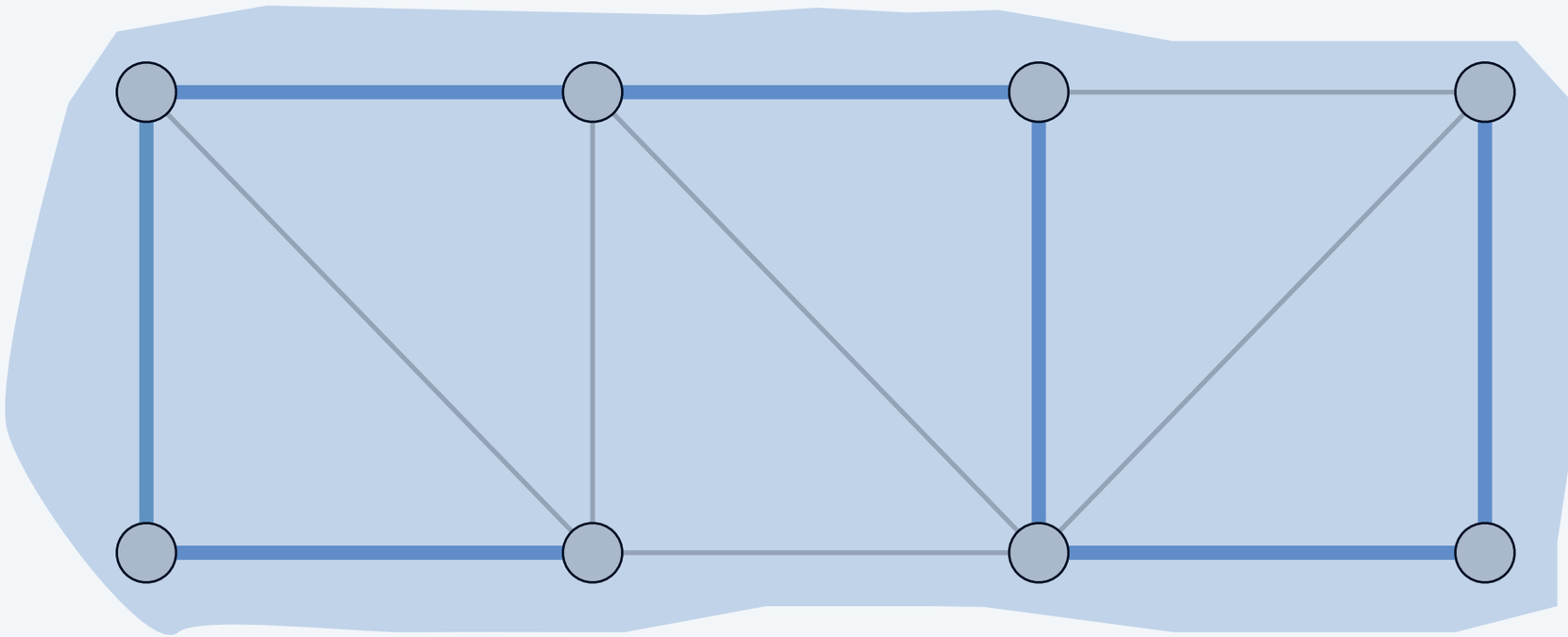


Prim's algorithm demo

Initialize S = any node, $T = \emptyset$.

Repeat $n - 1$ times:

- Add to T a min-weight edge with one endpoint in S.
- Add new node to S.

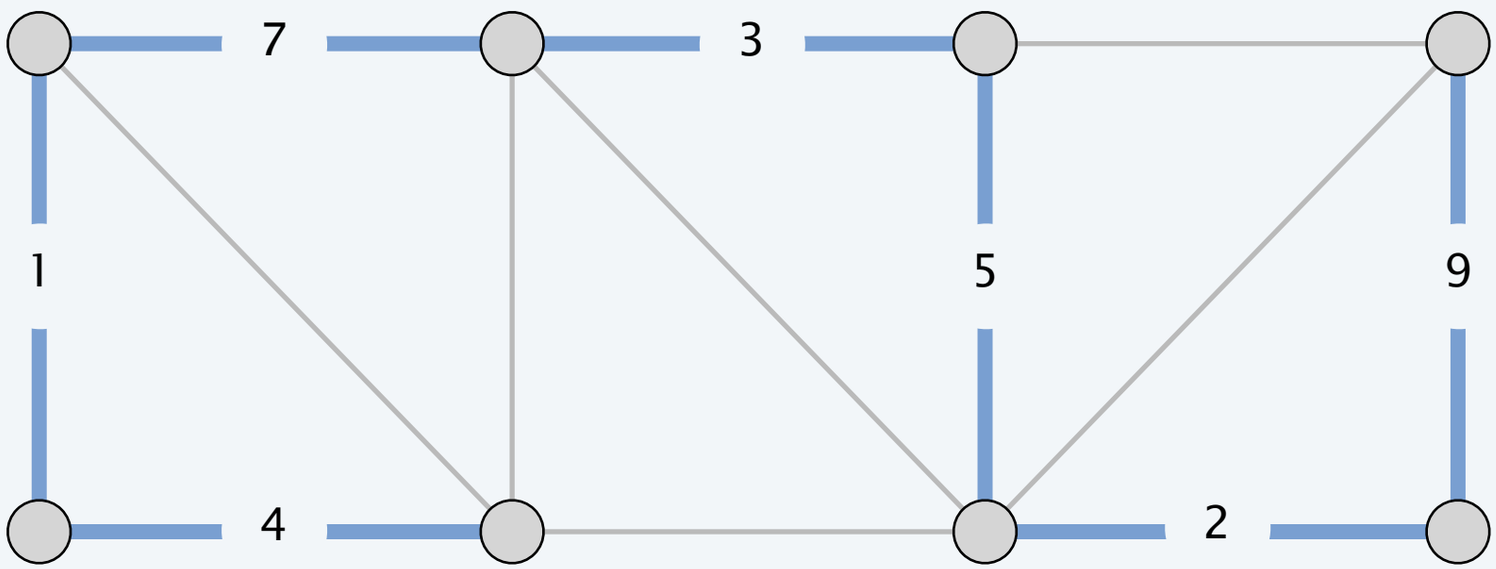


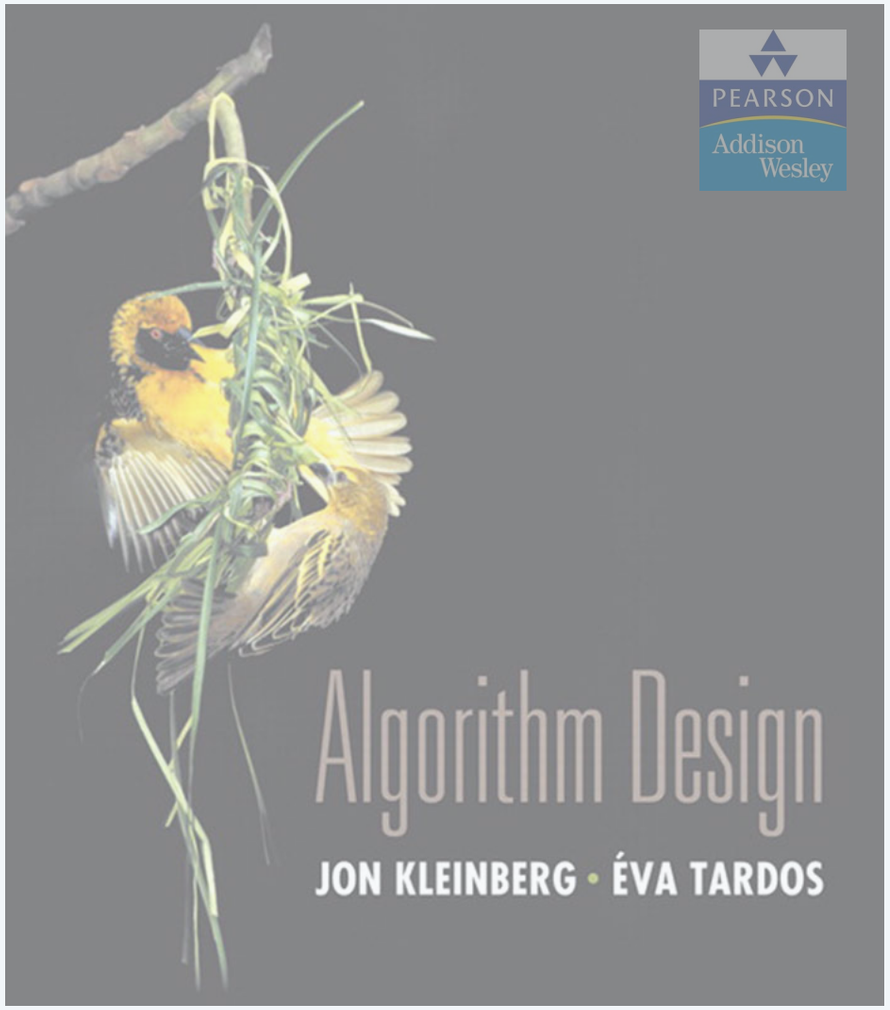
Prim's algorithm demo

Initialize S = any node, $T = \emptyset$.

Repeat $n - 1$ times:

- Add to T a min-weight edge with one endpoint in S.
- Add new node to S.





SECTION 4.5

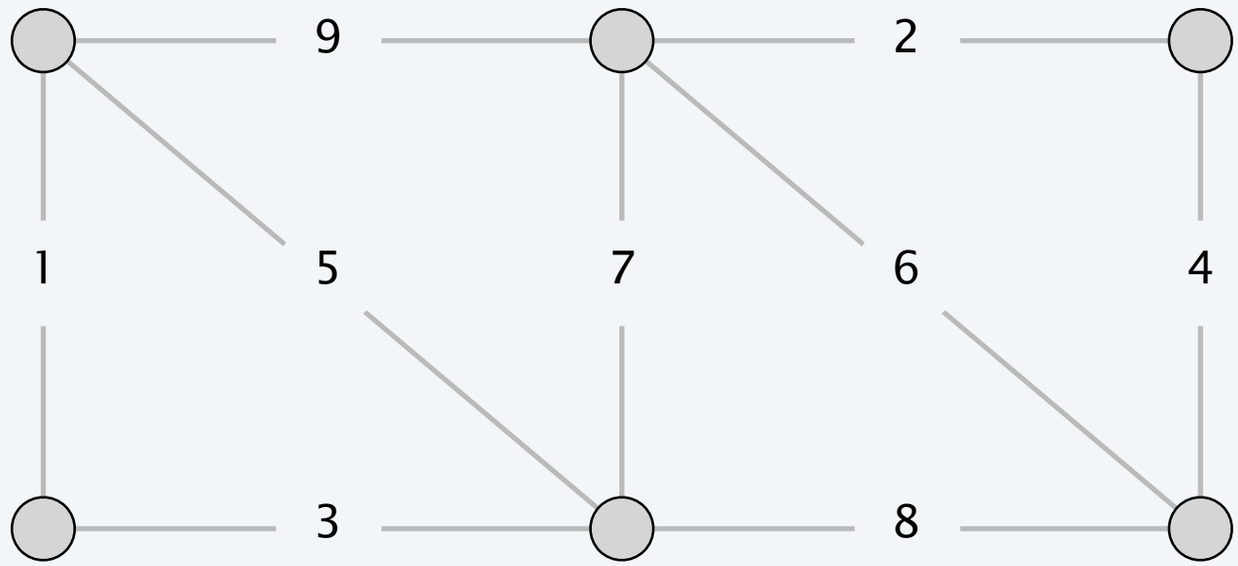
4. GREEDY ALGORITHMS II

- *red-rule blue-rule demo*
- *Prim's algorithm demo*
- *Kruskal's algorithm demo*
- *Boruvka's algorithm demo*

Kruskal's algorithm demo

Consider edges in ascending order of weight:

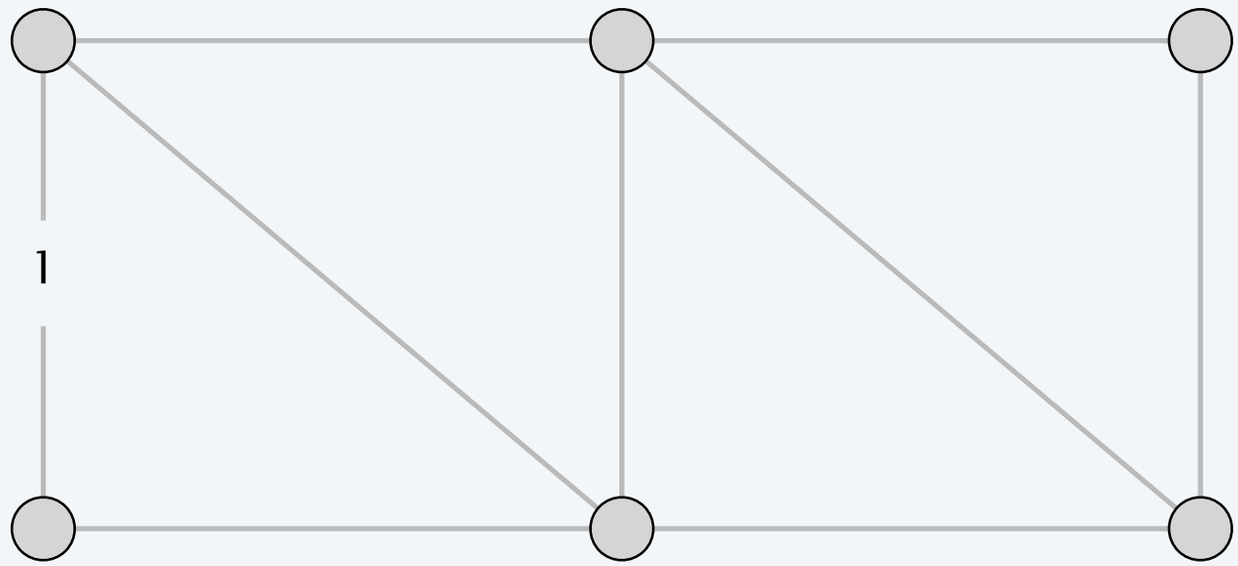
- Add to T unless it would create a cycle.



Kruskal's algorithm demo

Consider edges in ascending order of weight:

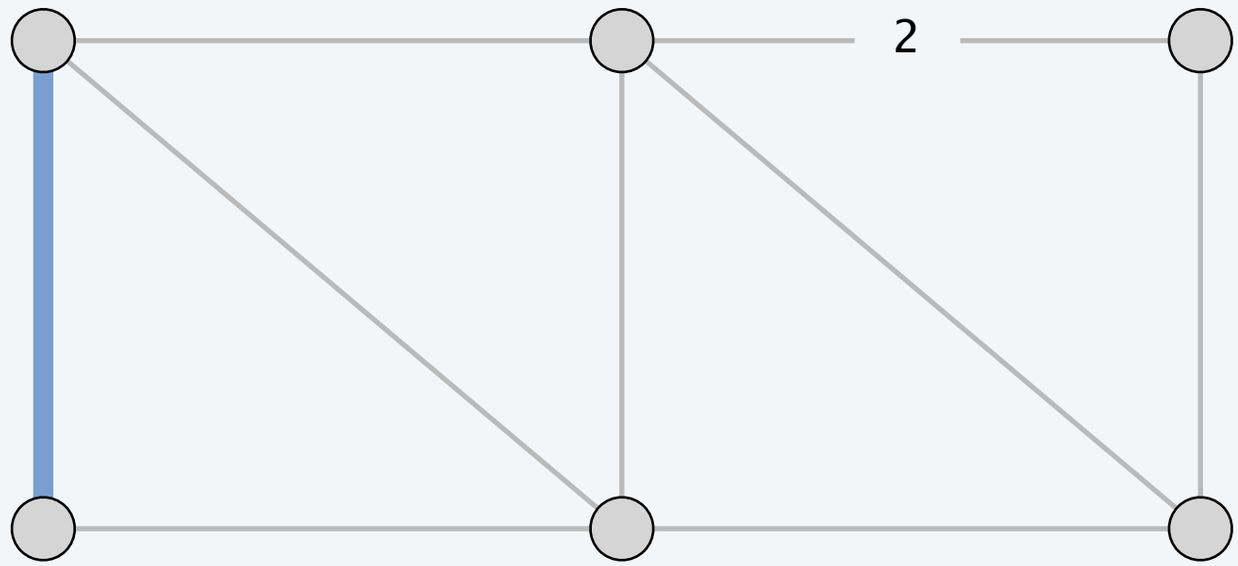
- Add to T unless it would create a cycle.



Kruskal's algorithm demo

Consider edges in ascending order of weight:

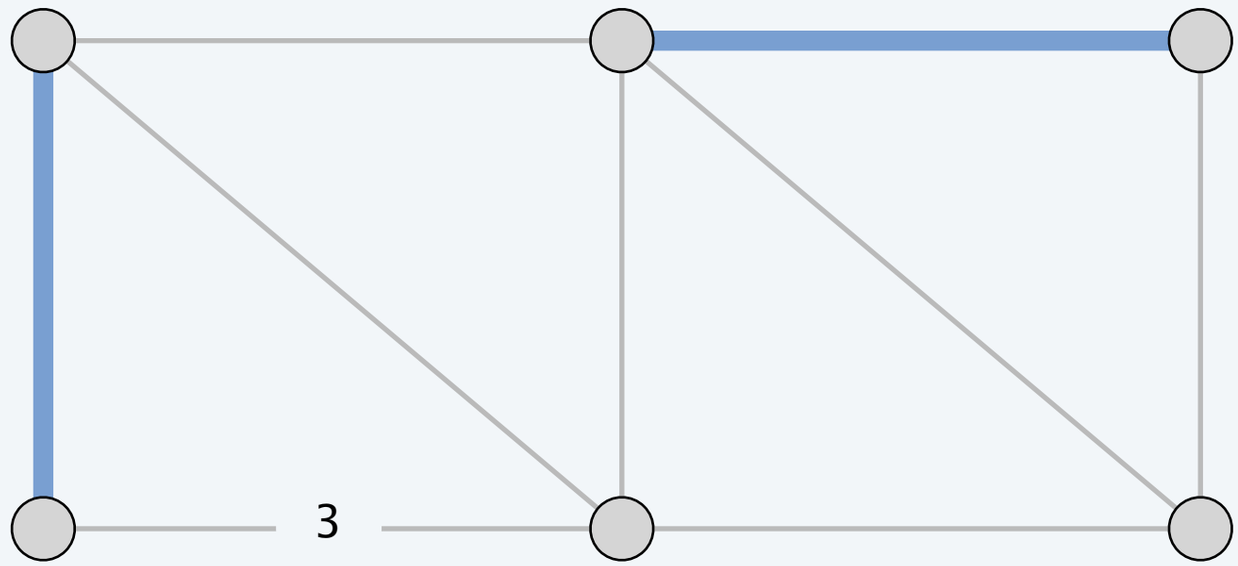
- Add to T unless it would create a cycle.



Kruskal's algorithm demo

Consider edges in ascending order of weight:

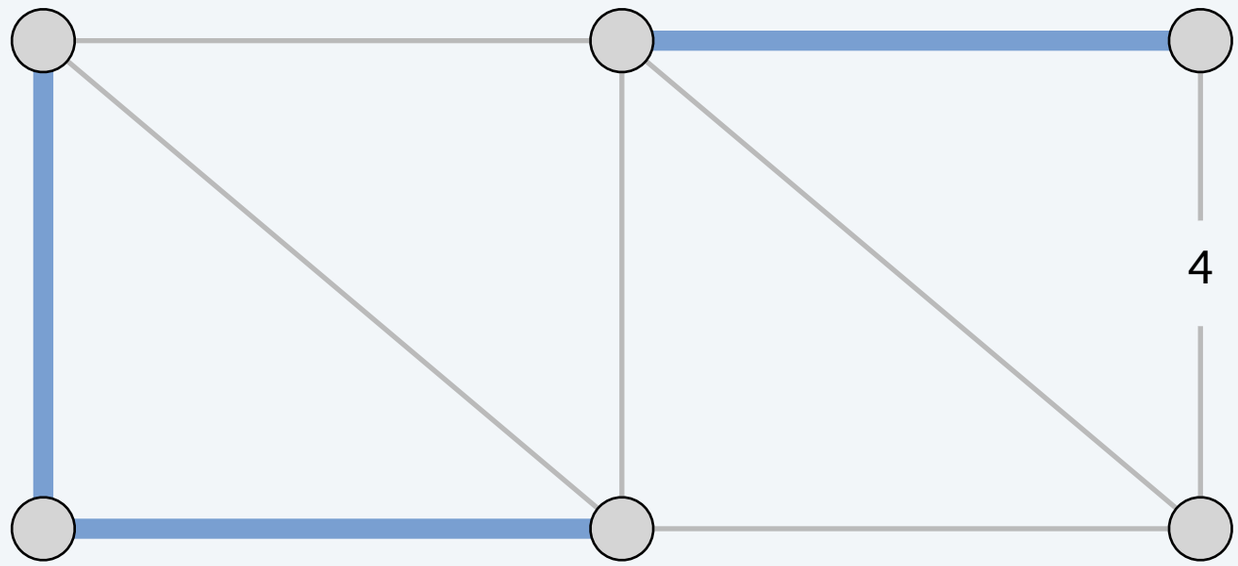
- Add to T unless it would create a cycle.



Kruskal's algorithm demo

Consider edges in ascending order of weight:

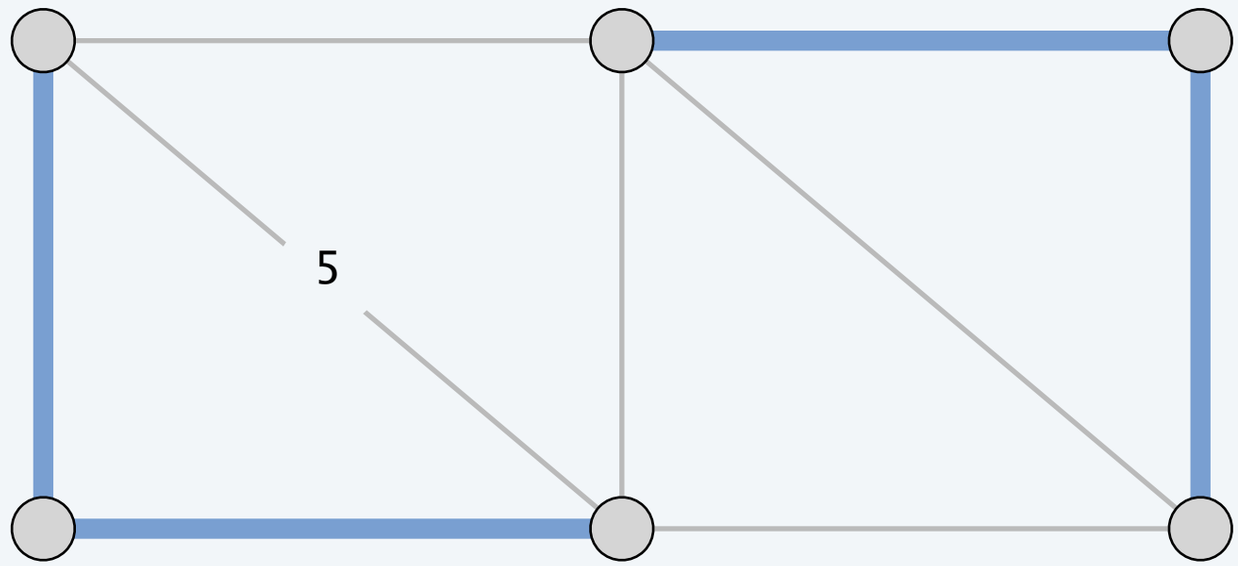
- Add to T unless it would create a cycle.



Kruskal's algorithm demo

Consider edges in ascending order of weight:

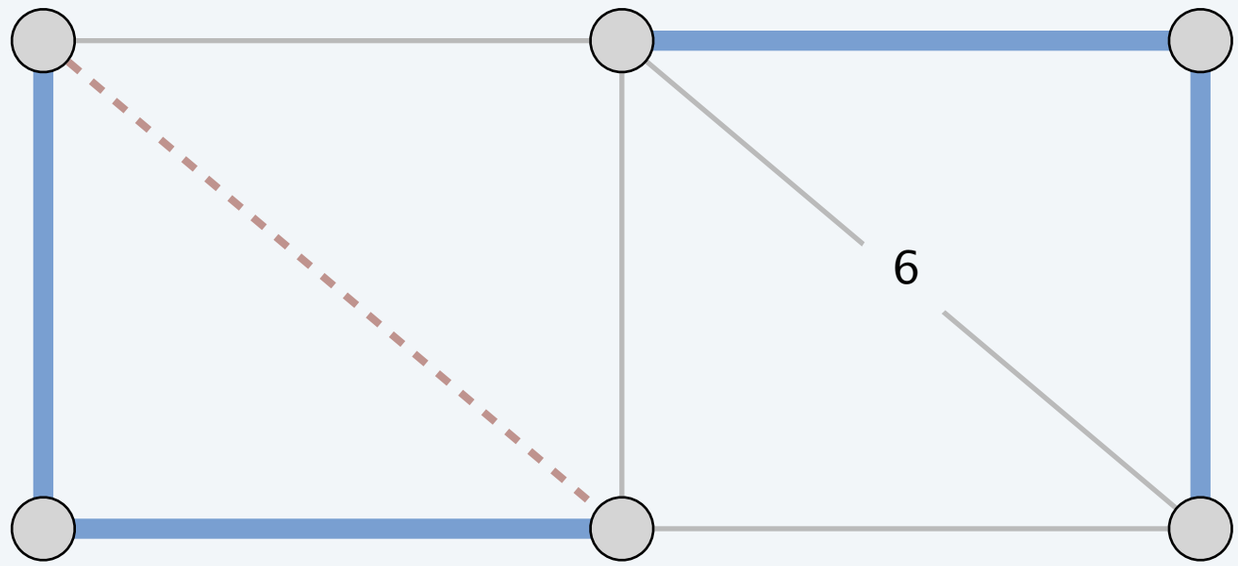
- Add to T unless it would create a cycle.



Kruskal's algorithm demo

Consider edges in ascending order of weight:

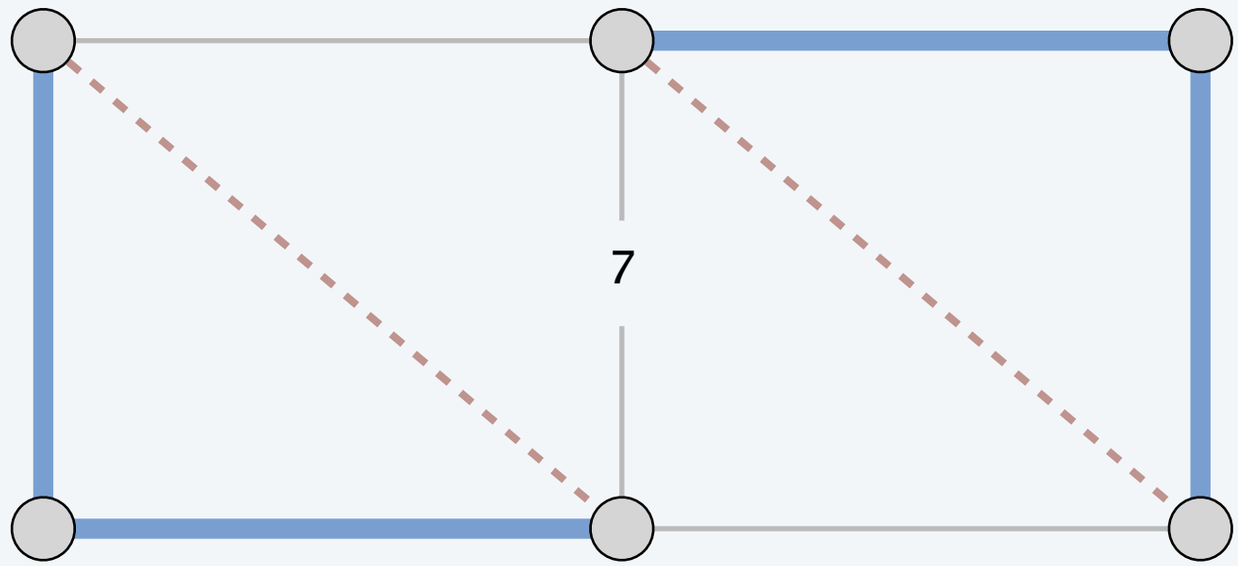
- Add to T unless it would create a cycle.



Kruskal's algorithm demo

Consider edges in ascending order of weight:

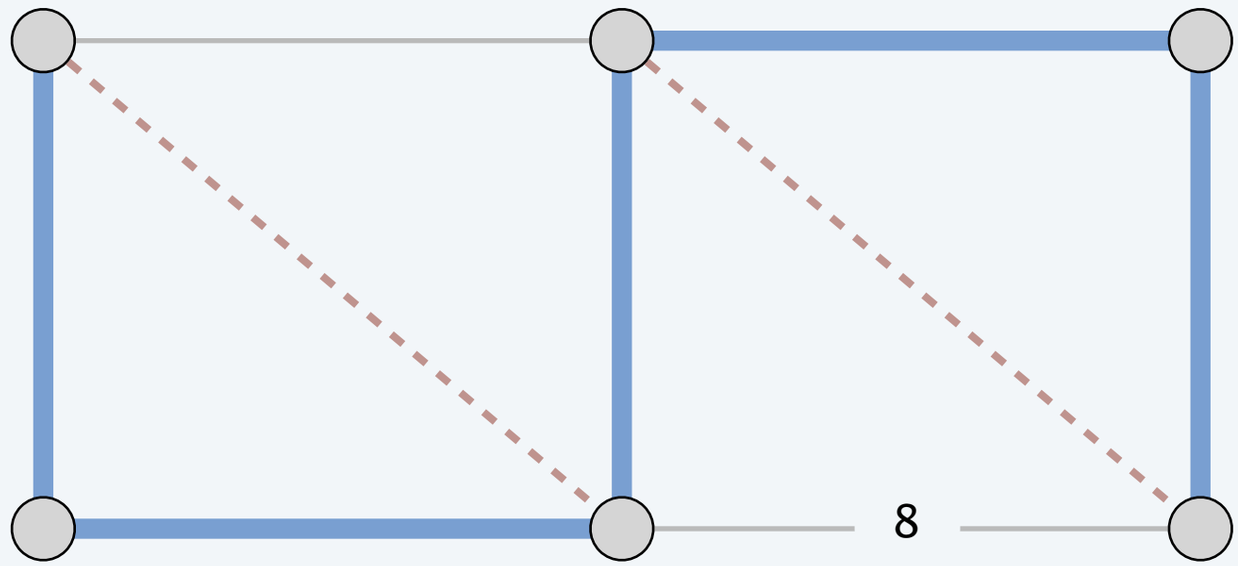
- Add to T unless it would create a cycle.



Kruskal's algorithm demo

Consider edges in ascending order of weight:

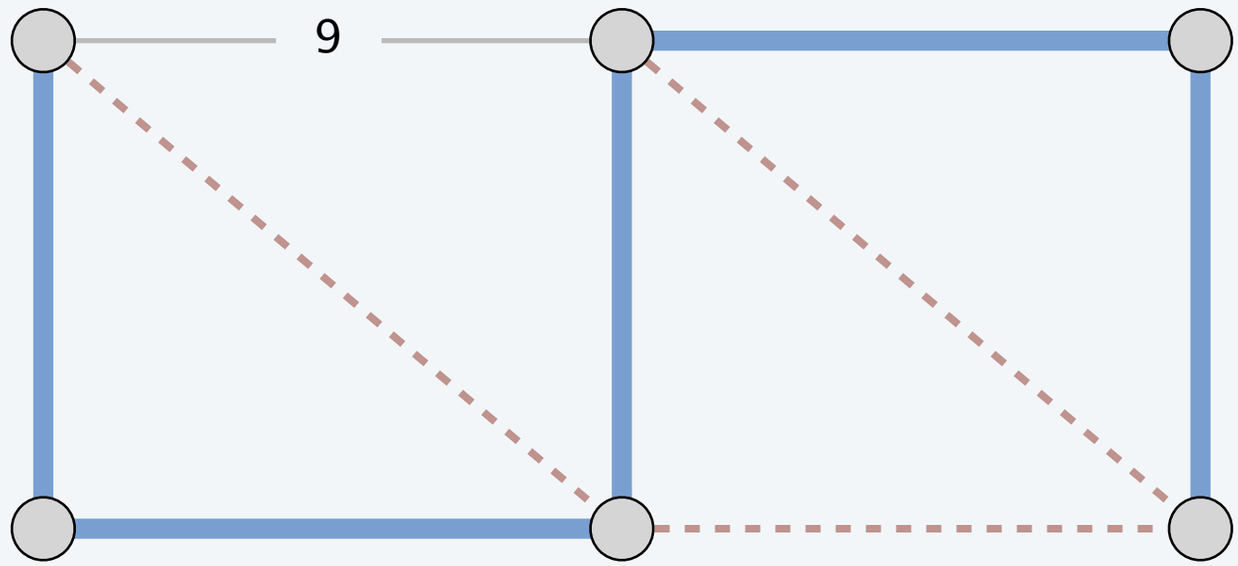
- Add to T unless it would create a cycle.



Kruskal's algorithm demo

Consider edges in ascending order of weight:

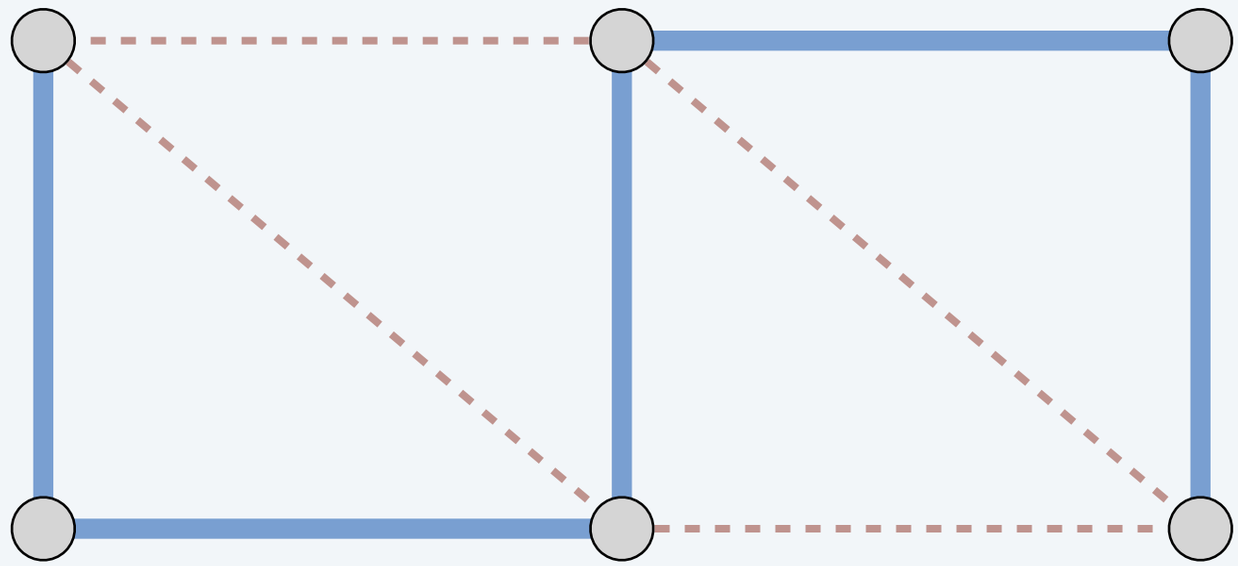
- Add to T unless it would create a cycle.



Kruskal's algorithm demo

Consider edges in ascending order of weight:

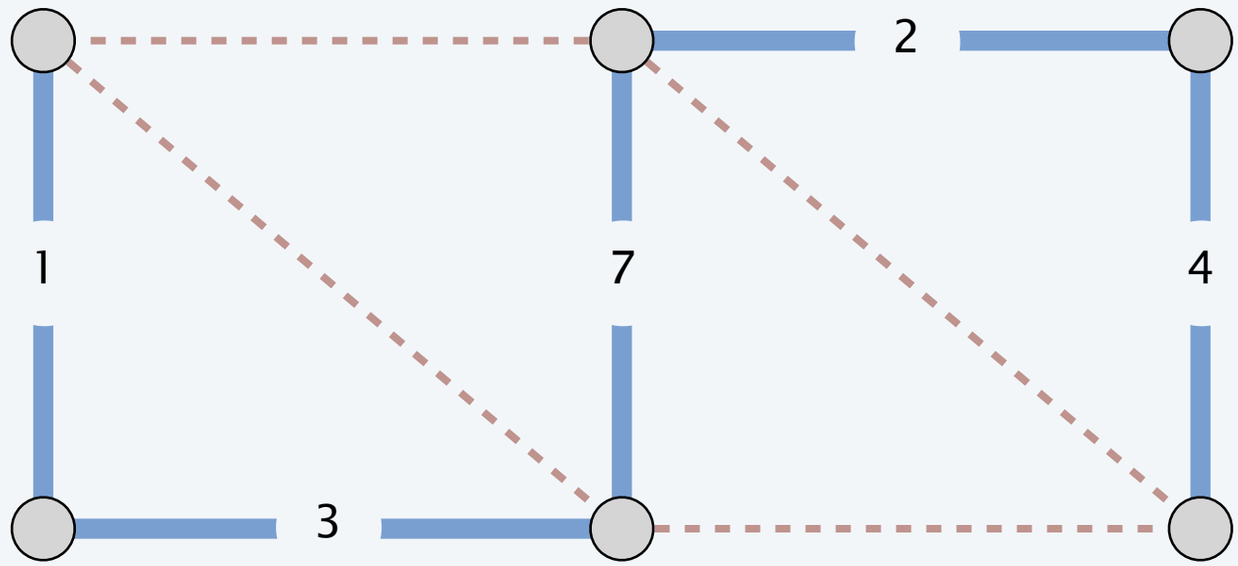
- Add to T unless it would create a cycle.

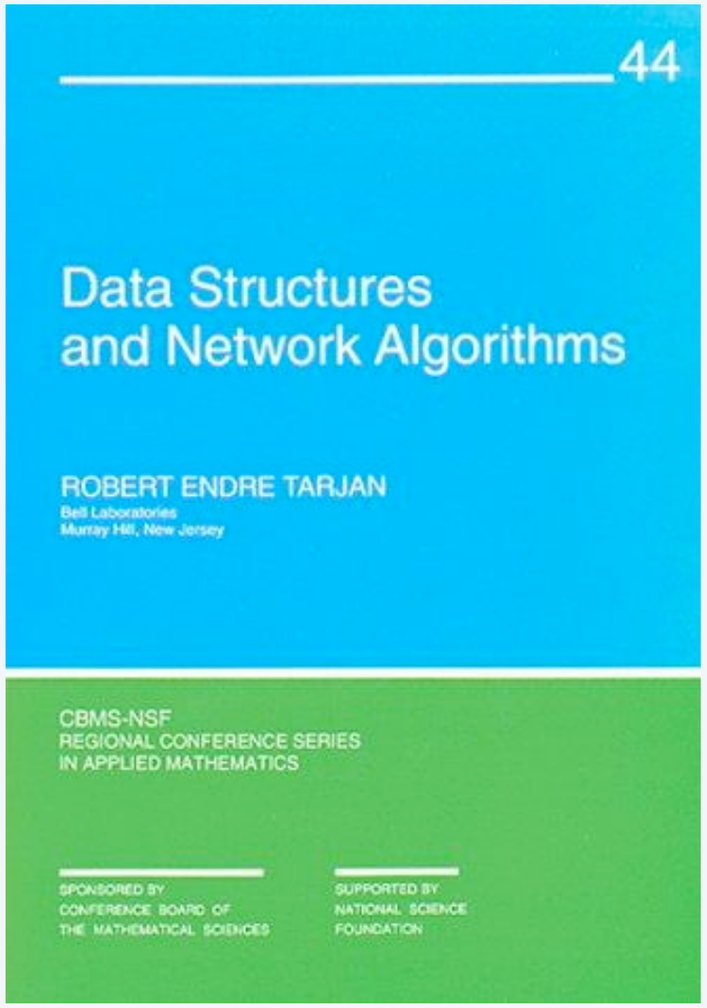


Kruskal's algorithm demo

Consider edges in ascending order of weight:

- Add to T unless it would create a cycle.





SECTION 6.2

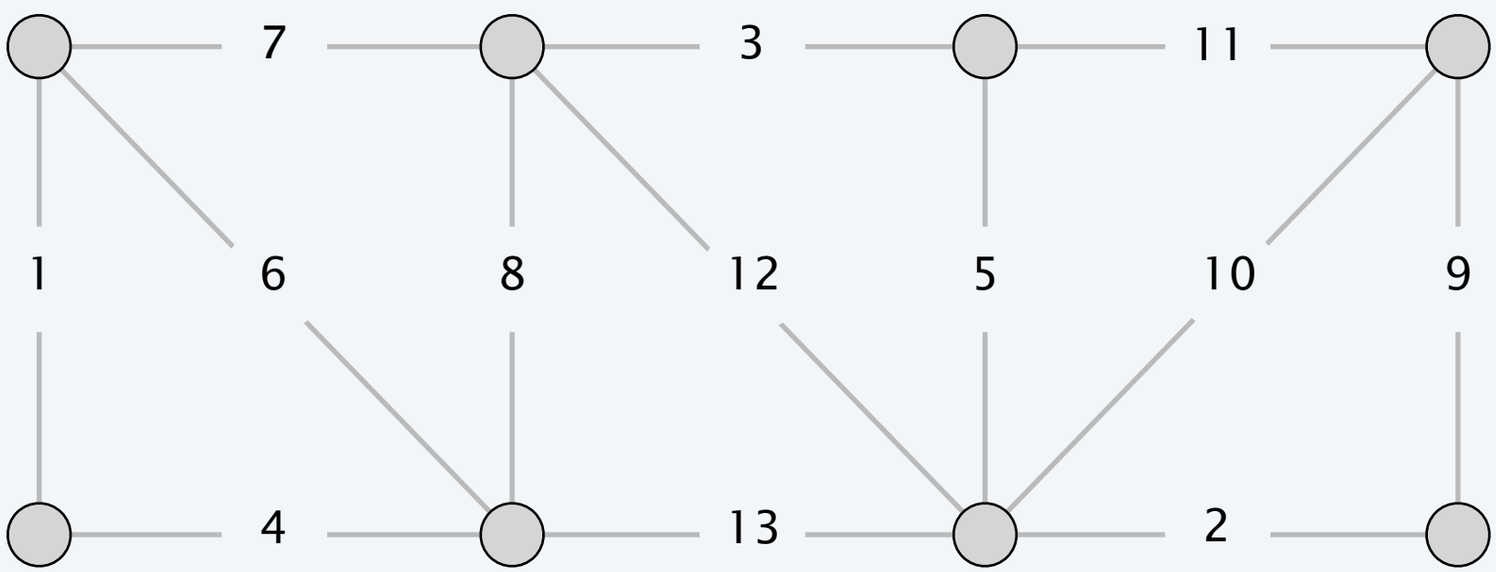
4. GREEDY ALGORITHMS II

- *red-rule blue-rule demo*
- *Prim's algorithm demo*
- *Kruskal's algorithm demo*
- *Boruvka's algorithm demo*

Borůvka's algorithm demo

Repeat until only one tree.

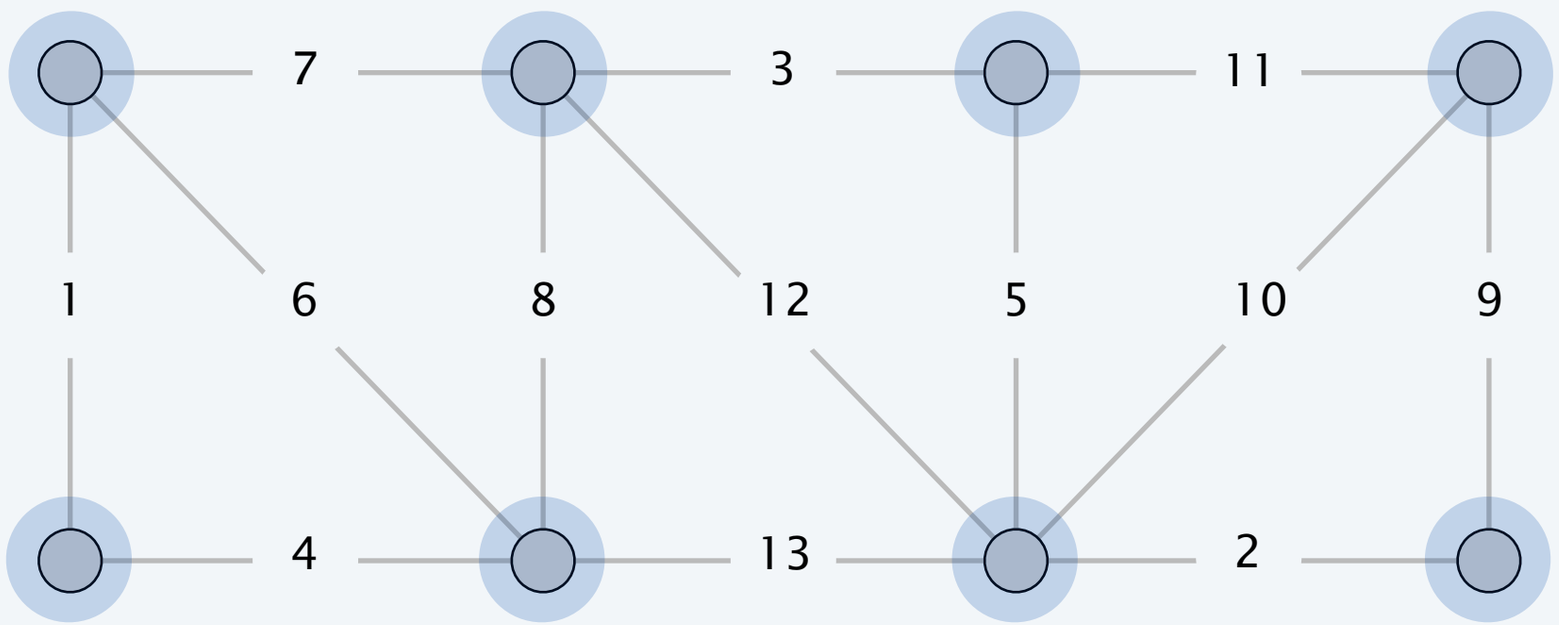
- Apply blue rule to cutset corresponding to each blue tree.
- Color all selected edges blue.



Borůvka's algorithm demo

Repeat until only one tree.

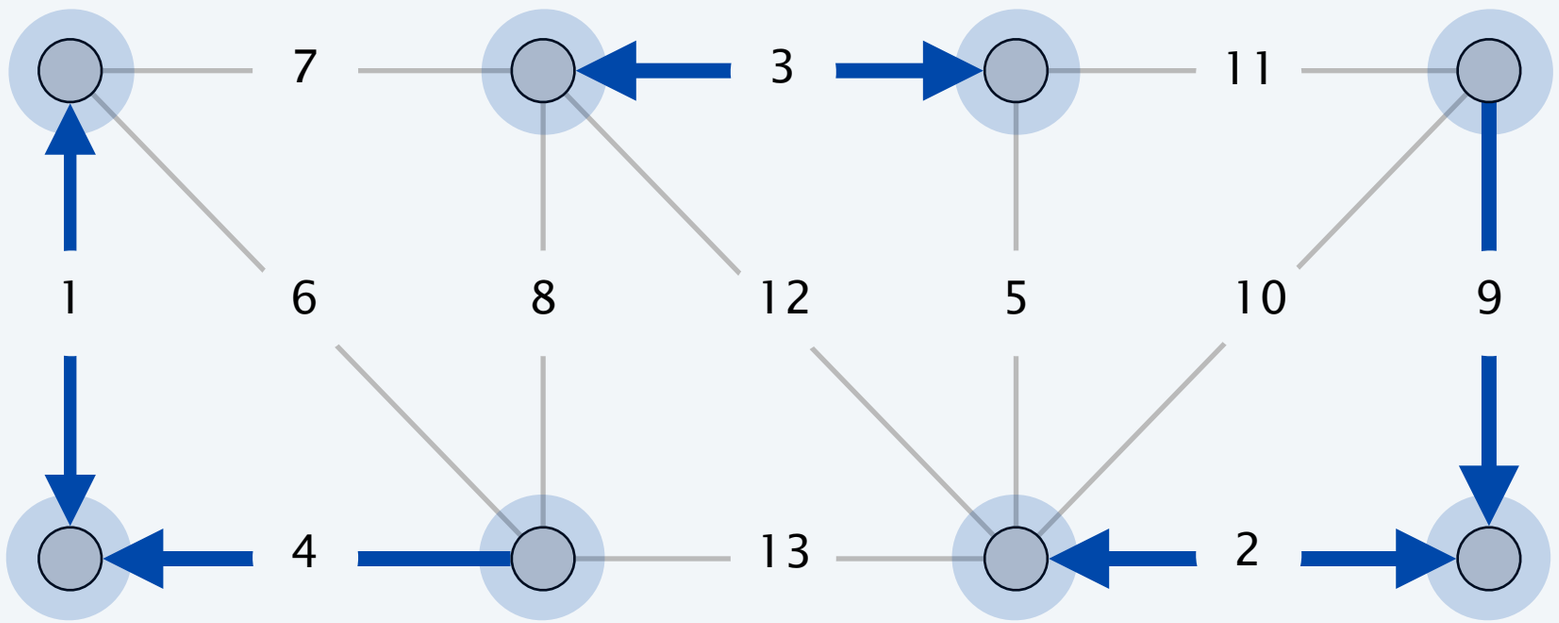
- Apply blue rule to cutset corresponding to each blue tree.
- Color all selected edges blue.



Borůvka's algorithm demo

Repeat until only one tree.

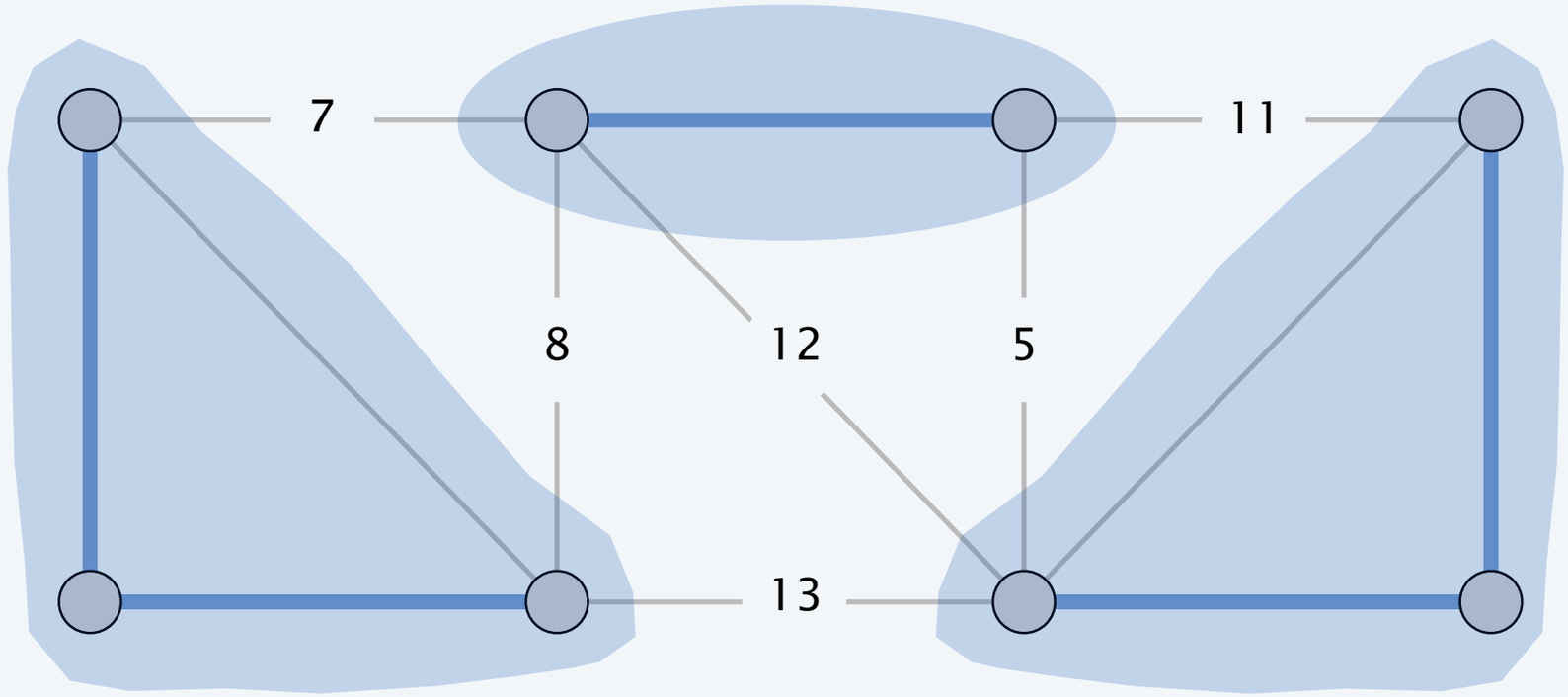
- Apply blue rule to cutset corresponding to each blue tree.
- Color all selected edges blue.



Borůvka's algorithm demo

Repeat until only one tree.

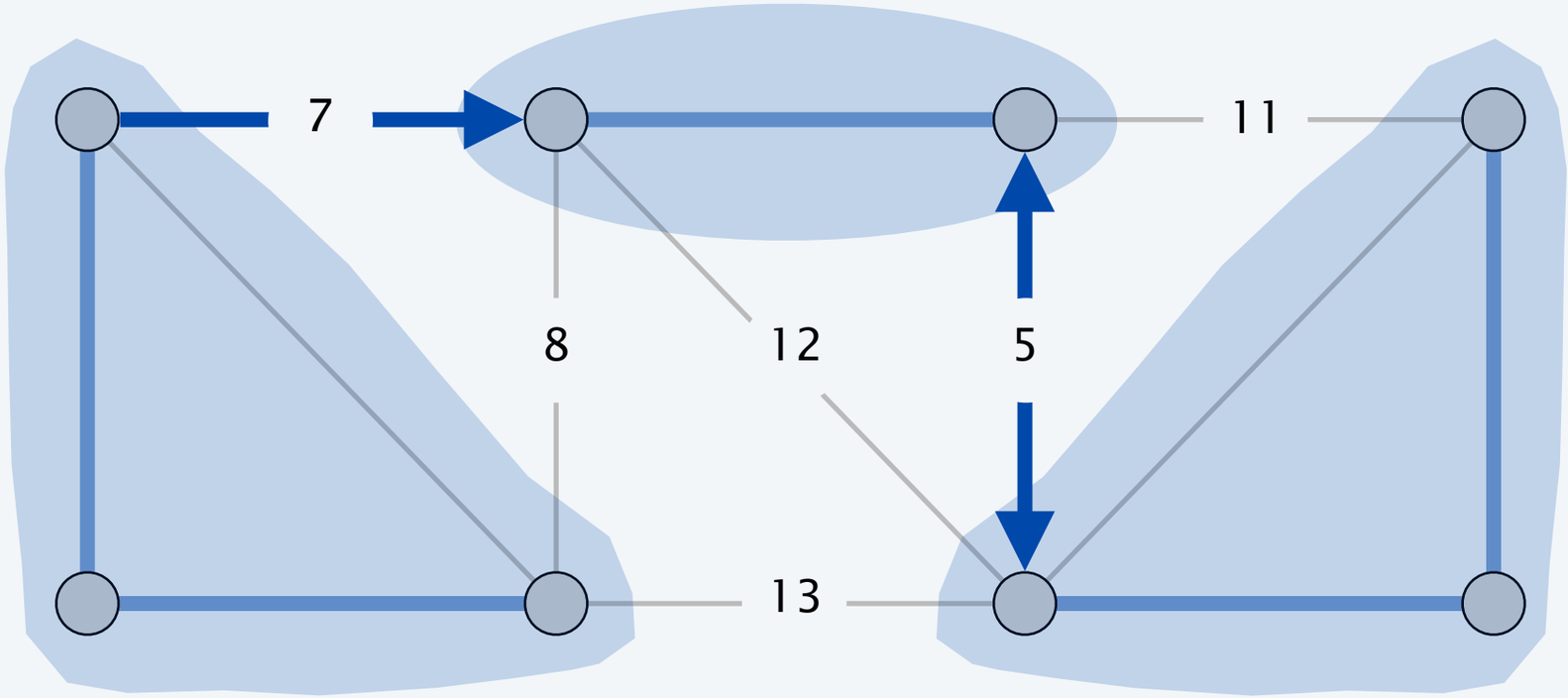
- Apply blue rule to cutset corresponding to each blue tree.
- Color all selected edges blue.



Borůvka's algorithm demo

Repeat until only one tree.

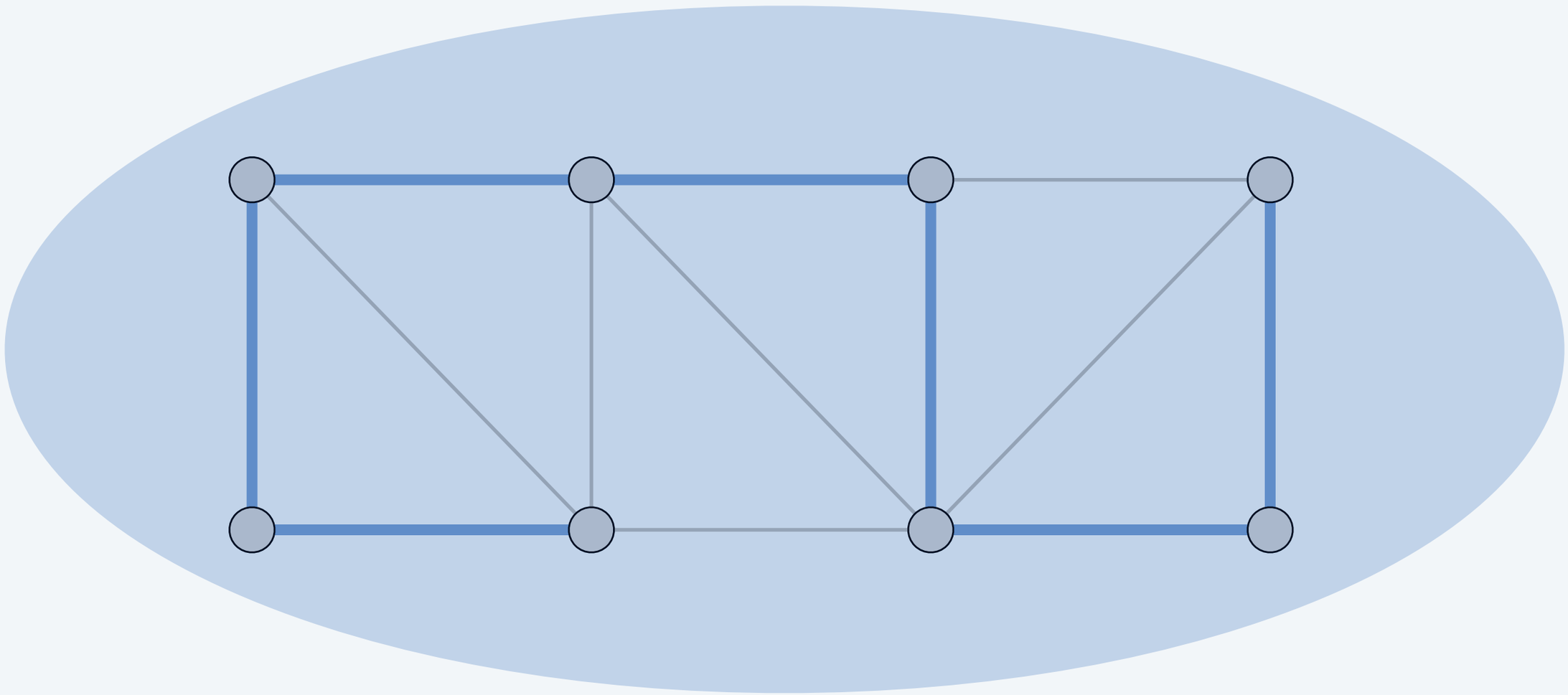
- Apply blue rule to cutset corresponding to each blue tree.
- Color all selected edges blue.



Borůvka's algorithm demo

Repeat until only one tree.

- Apply blue rule to cutset corresponding to each blue tree.
- Color all selected edges blue.



Borůvka's algorithm demo

Repeat until only one tree.

- Apply blue rule to cutset corresponding to each blue tree.
- Color all selected edges blue.