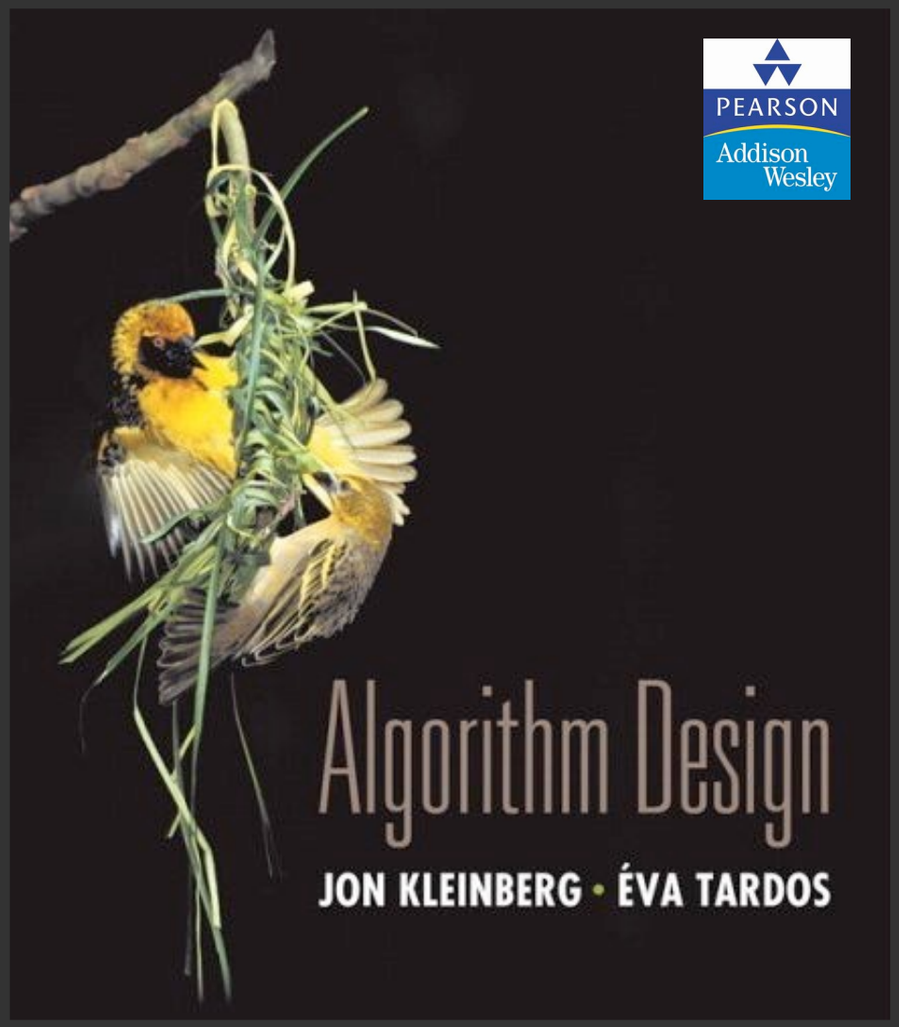


4. Greedy Algorithms II

- *Prim's algorithm demo*

http://www.cs.princeton.edu/~wayne/kleinberg-tardos

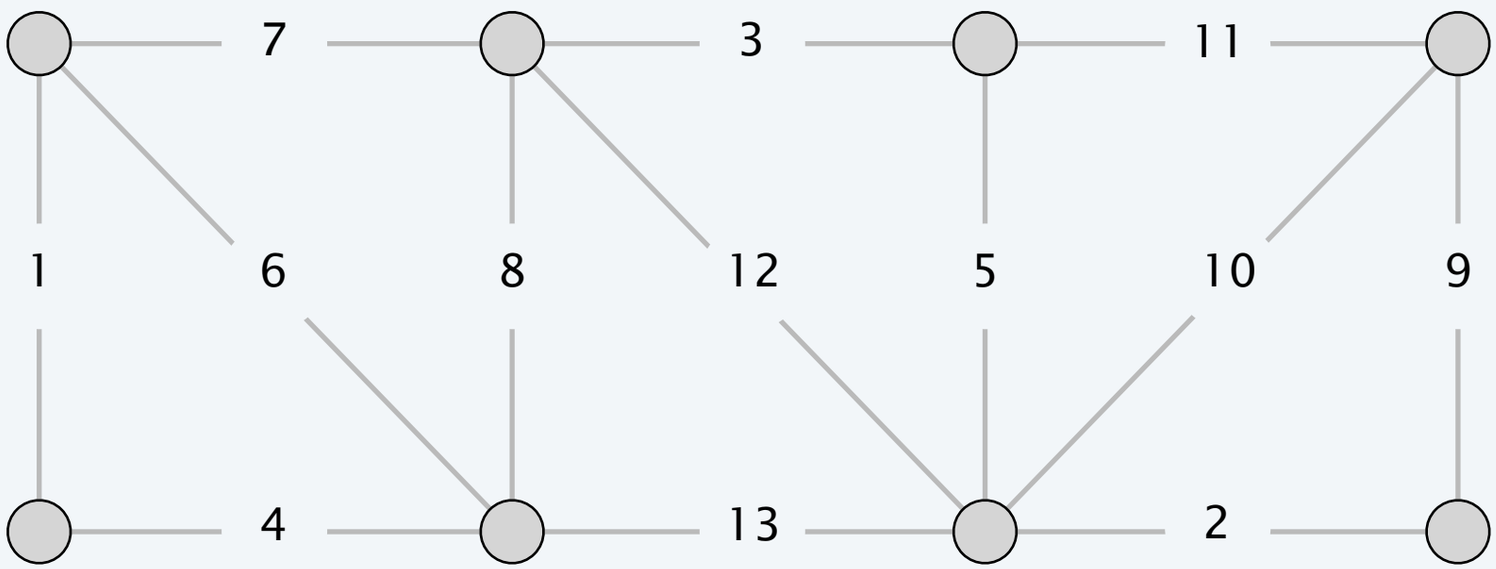
Last updated on 7/25/17 10:58 AM

Prim's algorithm demo

Initialize S = any node, $T = (V, \varnothing)$.

Repeat $n - 1$ times:

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

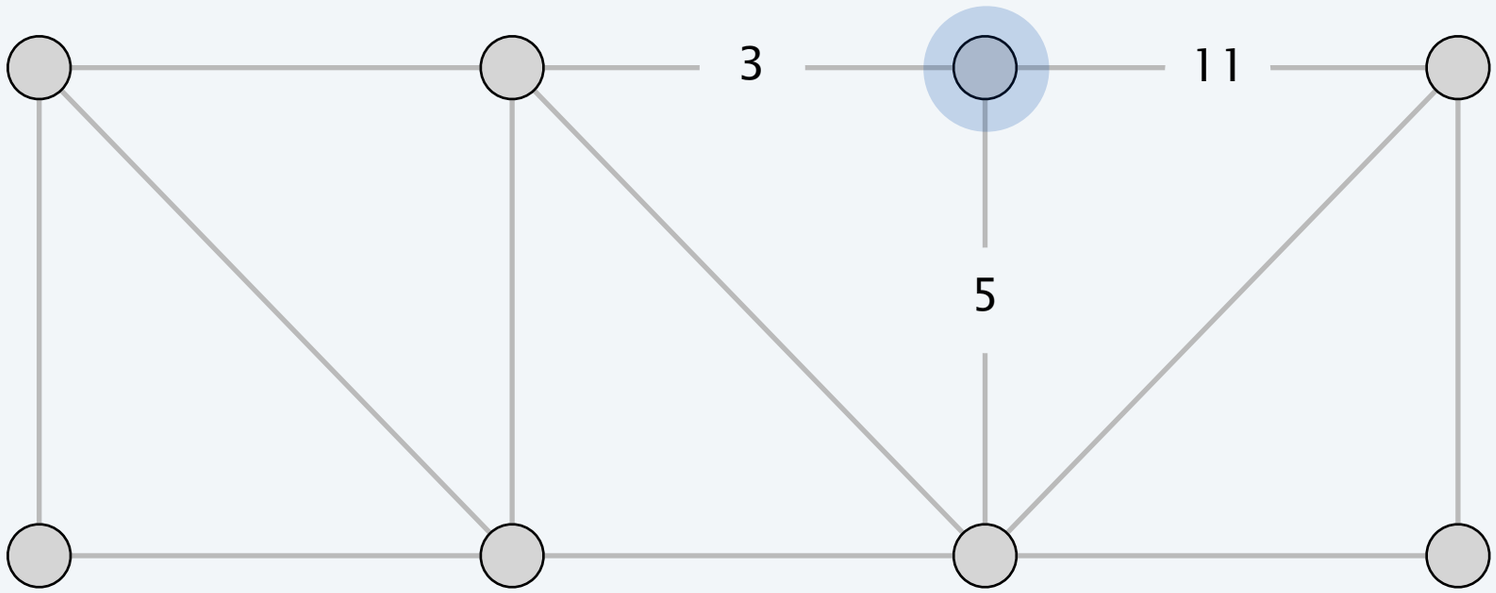


Prim's algorithm demo

Initialize S = any node, $T = (V, \varnothing)$.

Repeat $n - 1$ times:

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

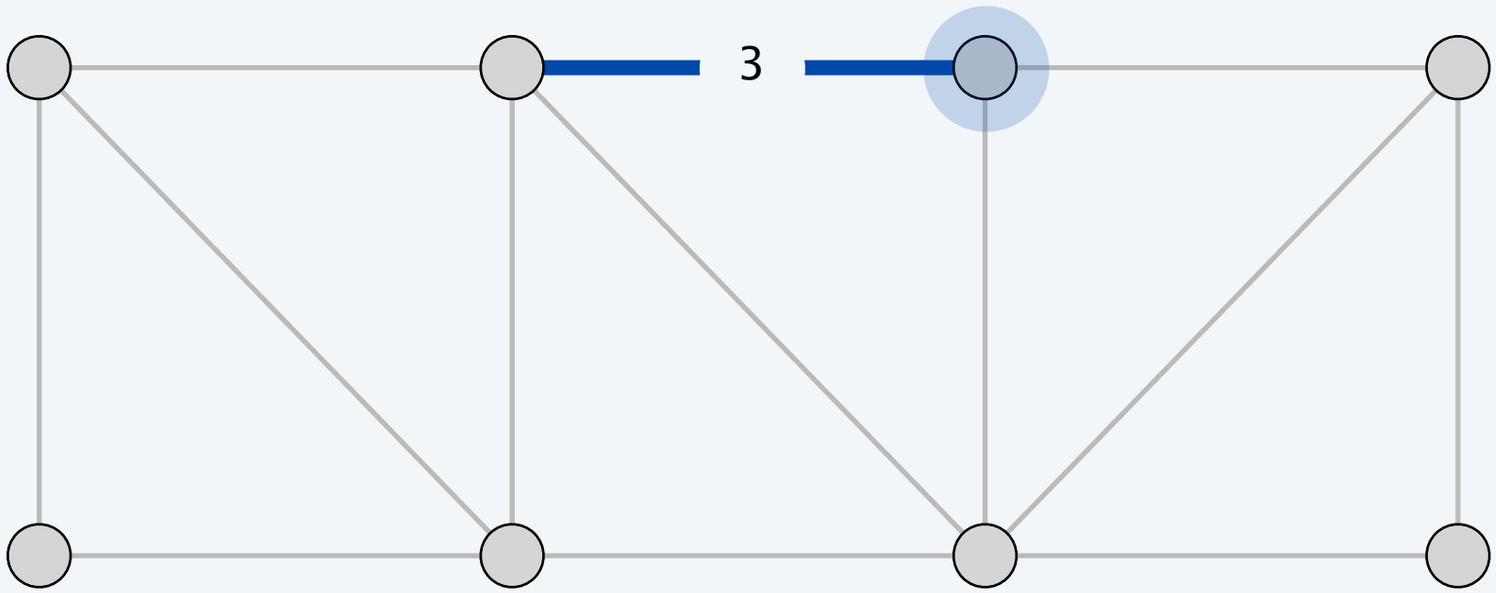


Prim's algorithm demo

Initialize S = any node, $T = (V, \varnothing)$.

Repeat $n - 1$ times:

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

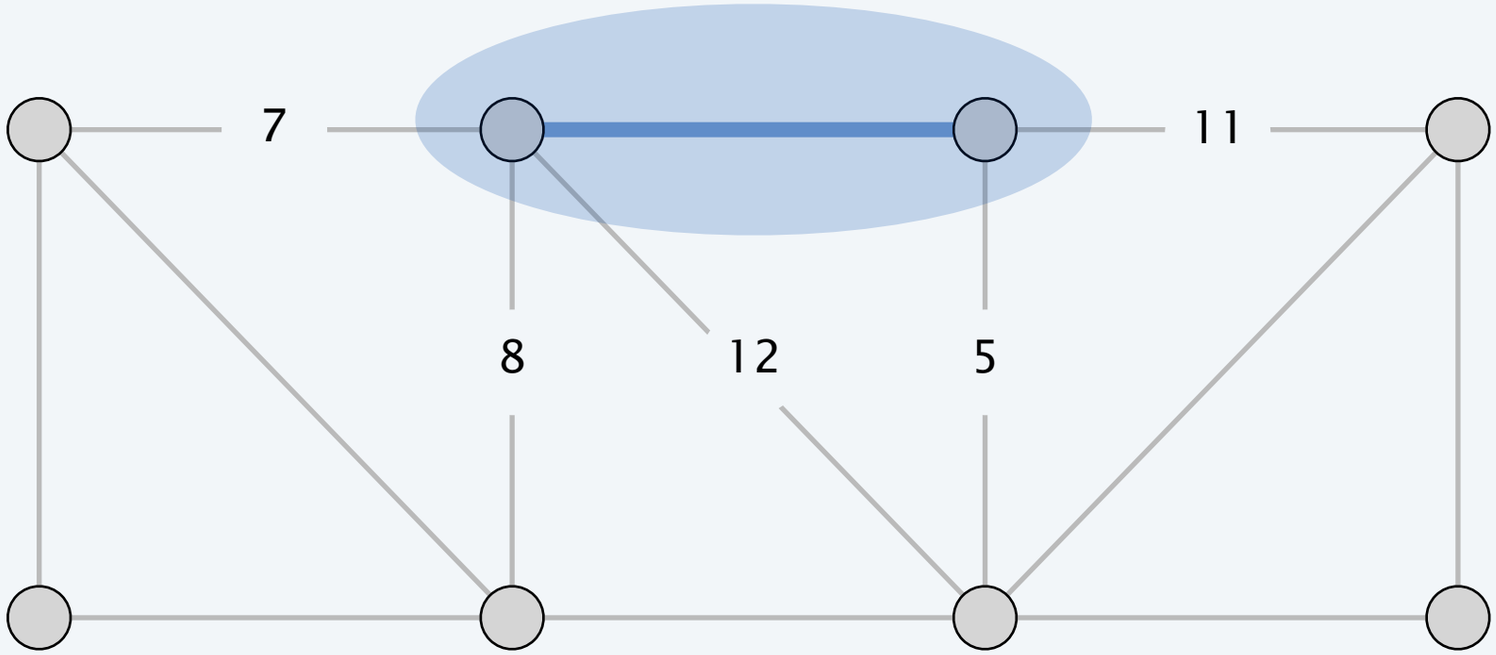


Prim's algorithm demo

Initialize S = any node, $T = (V, \varnothing)$.

Repeat $n - 1$ times:

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

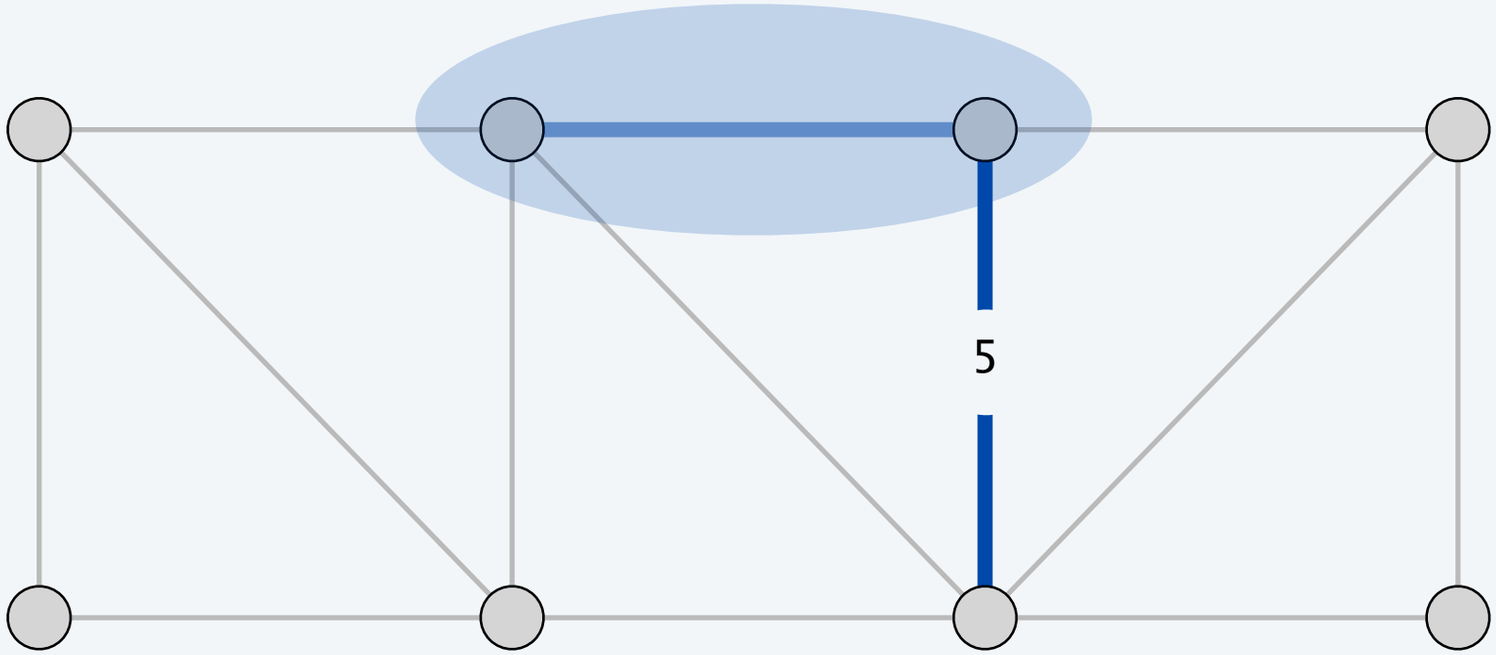


Prim's algorithm demo

Initialize S = any node, $T = (V, \varnothing)$.

Repeat $n - 1$ times:

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

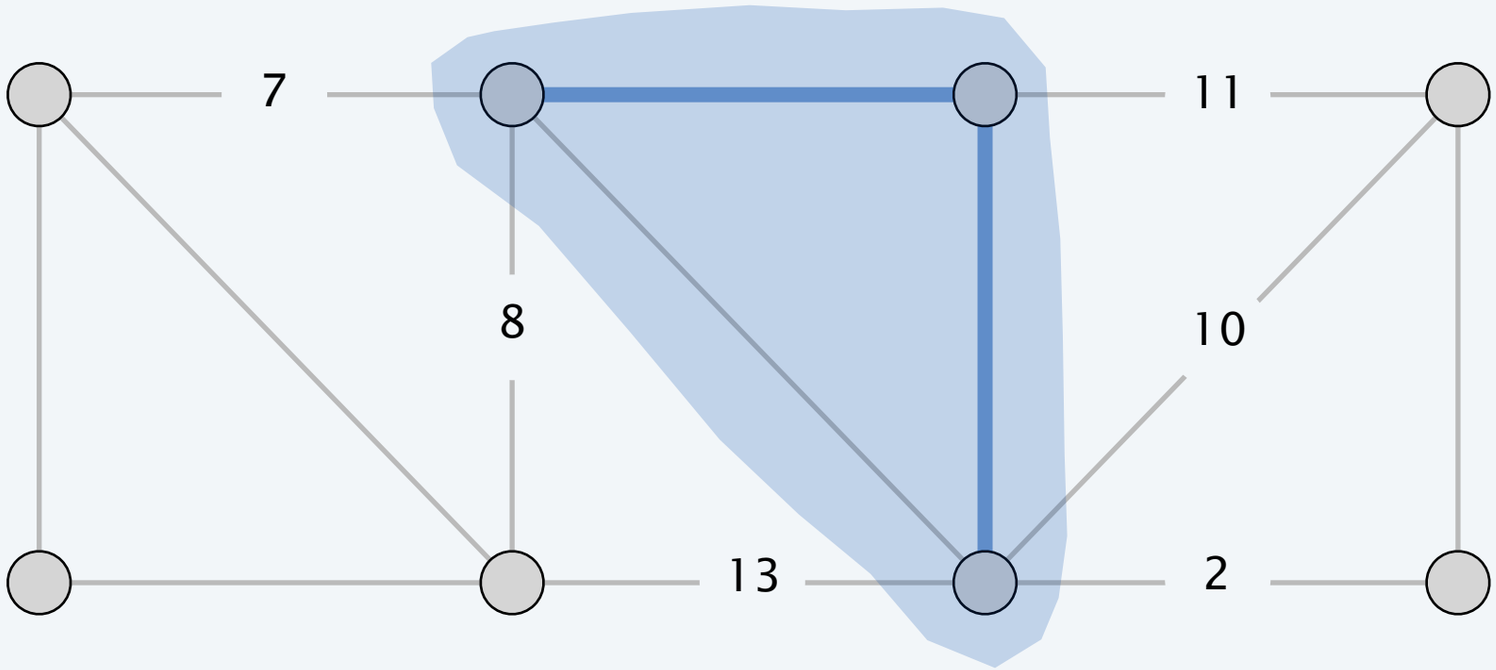


Prim's algorithm demo

Initialize S = any node, $T = (V, \varnothing)$.

Repeat $n - 1$ times:

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

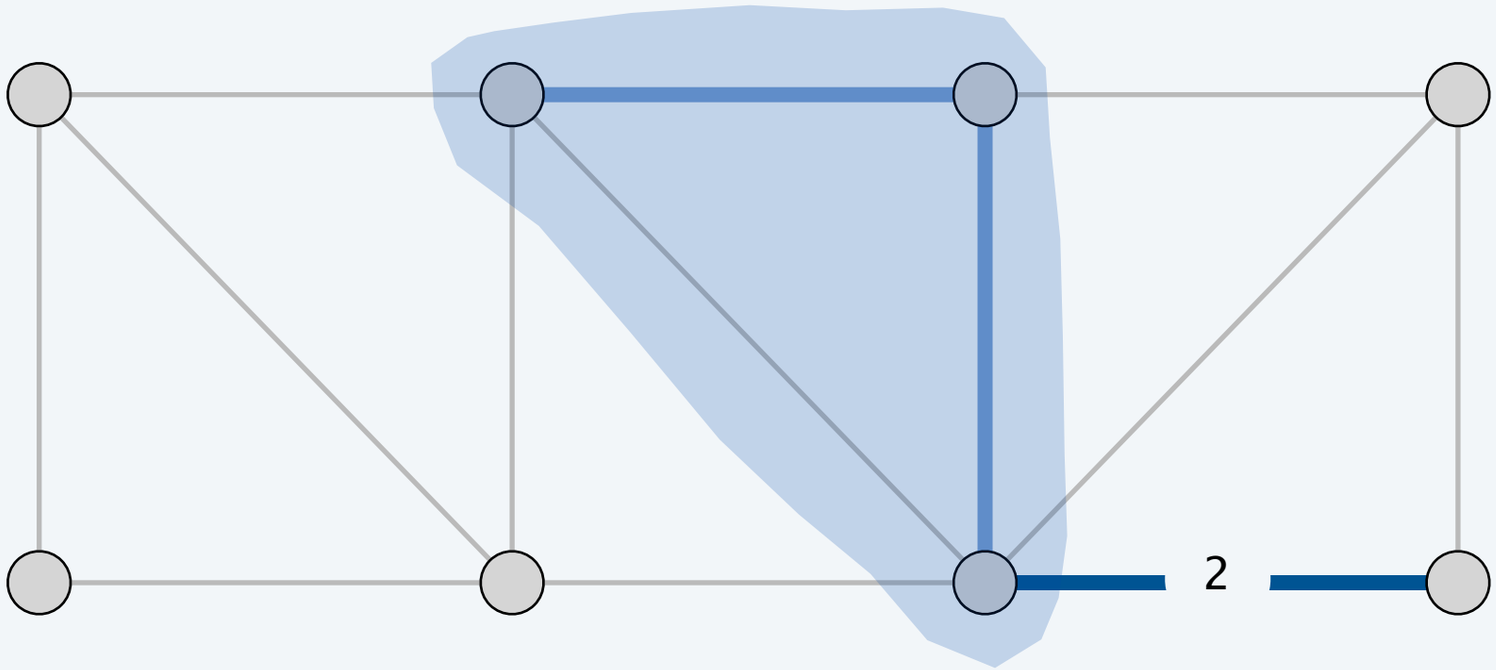


Prim's algorithm demo

Initialize S = any node, $T = (V, \varnothing)$.

Repeat $n - 1$ times:

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

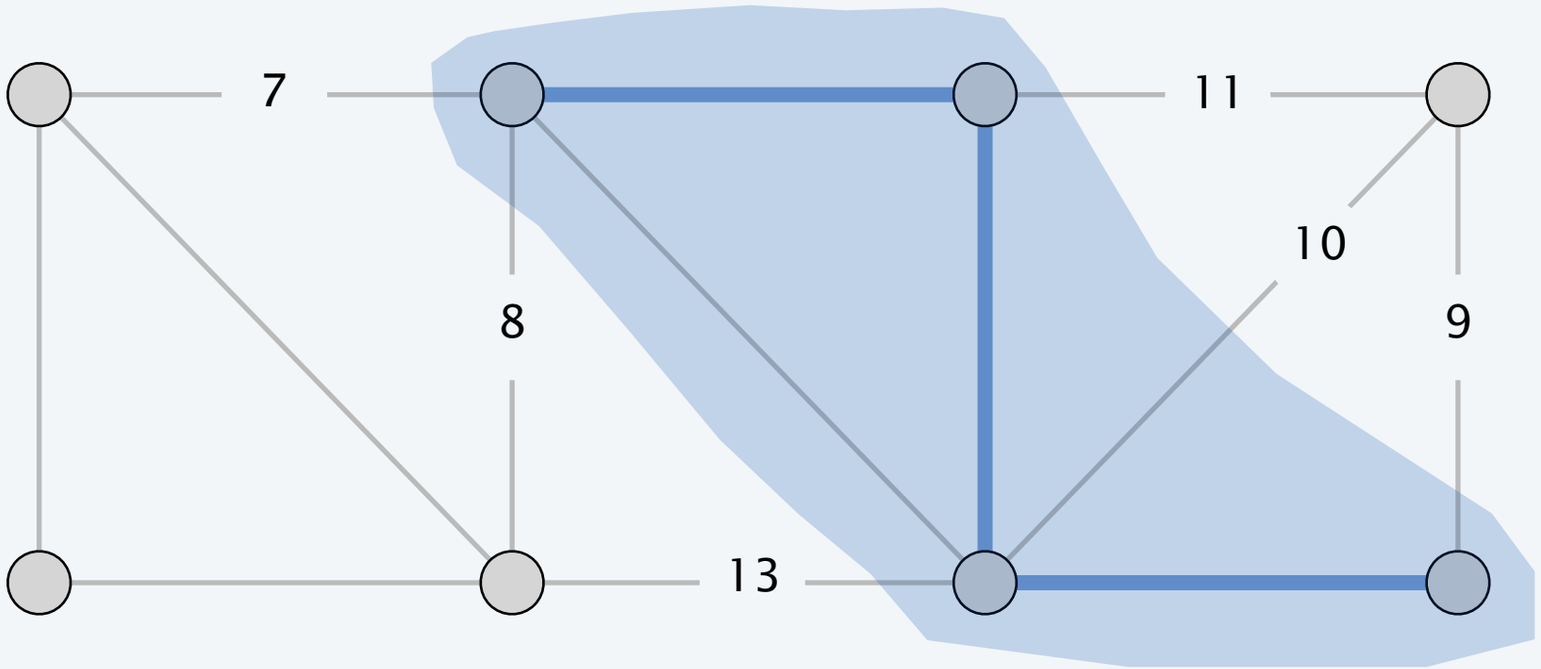


Prim's algorithm demo

Initialize S = any node, $T = (V, \varnothing)$.

Repeat $n - 1$ times:

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

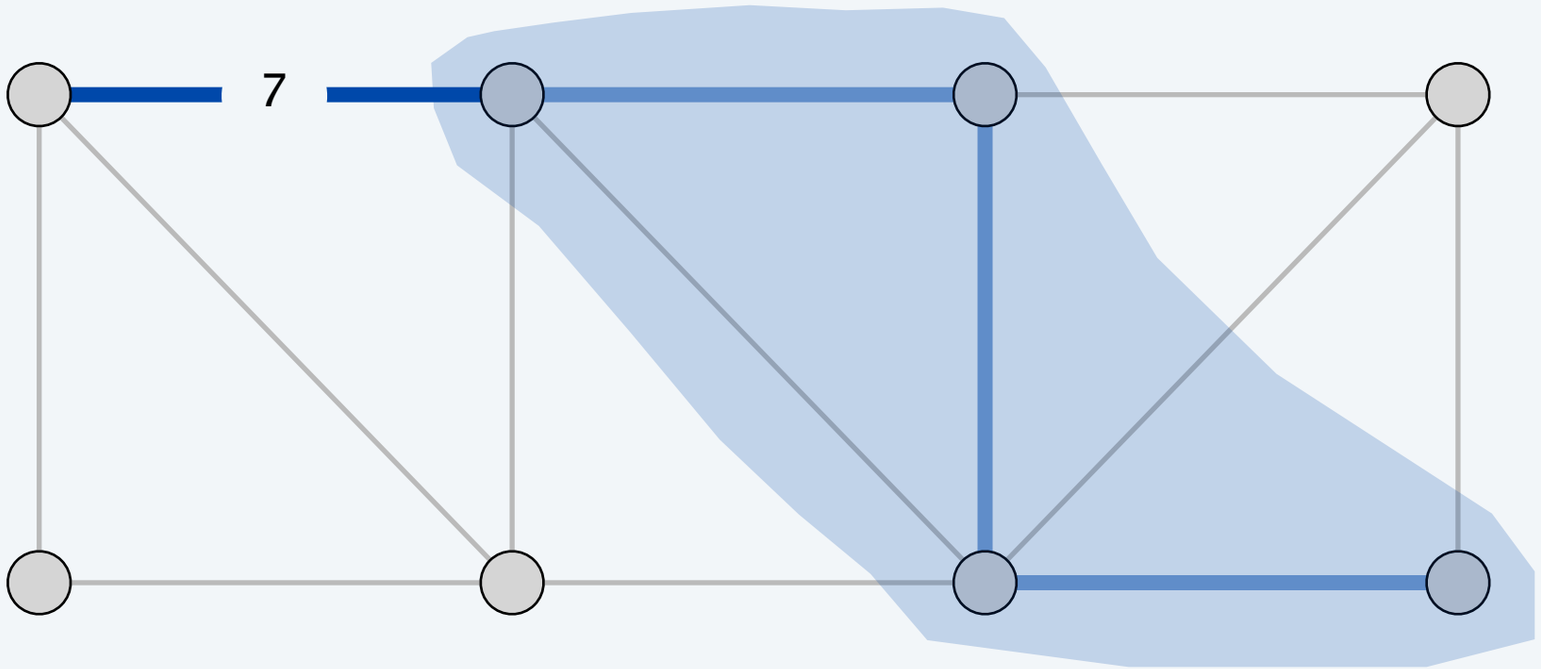


Prim's algorithm demo

Initialize S = any node, $T = (V, \varnothing)$.

Repeat $n - 1$ times:

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

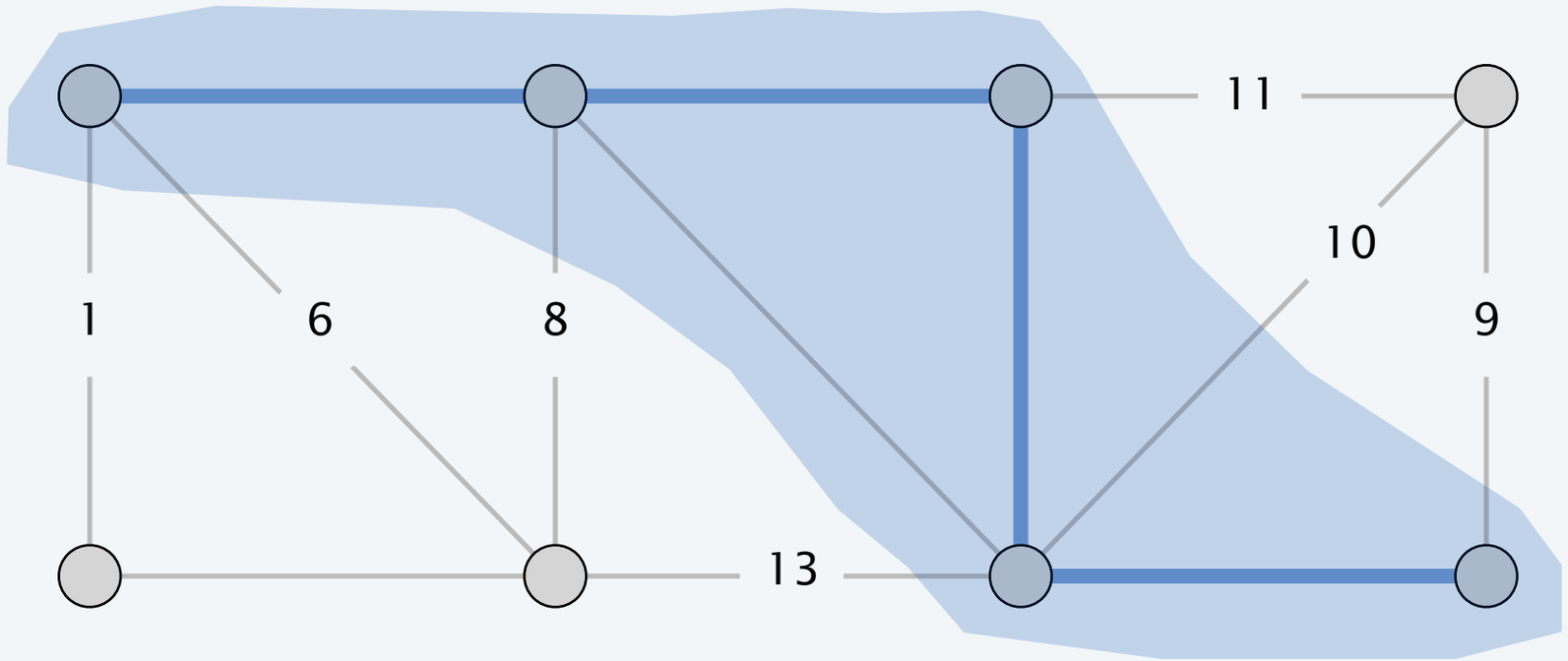


Prim's algorithm demo

Initialize S = any node, $T = (V, \varnothing)$.

Repeat $n - 1$ times:

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

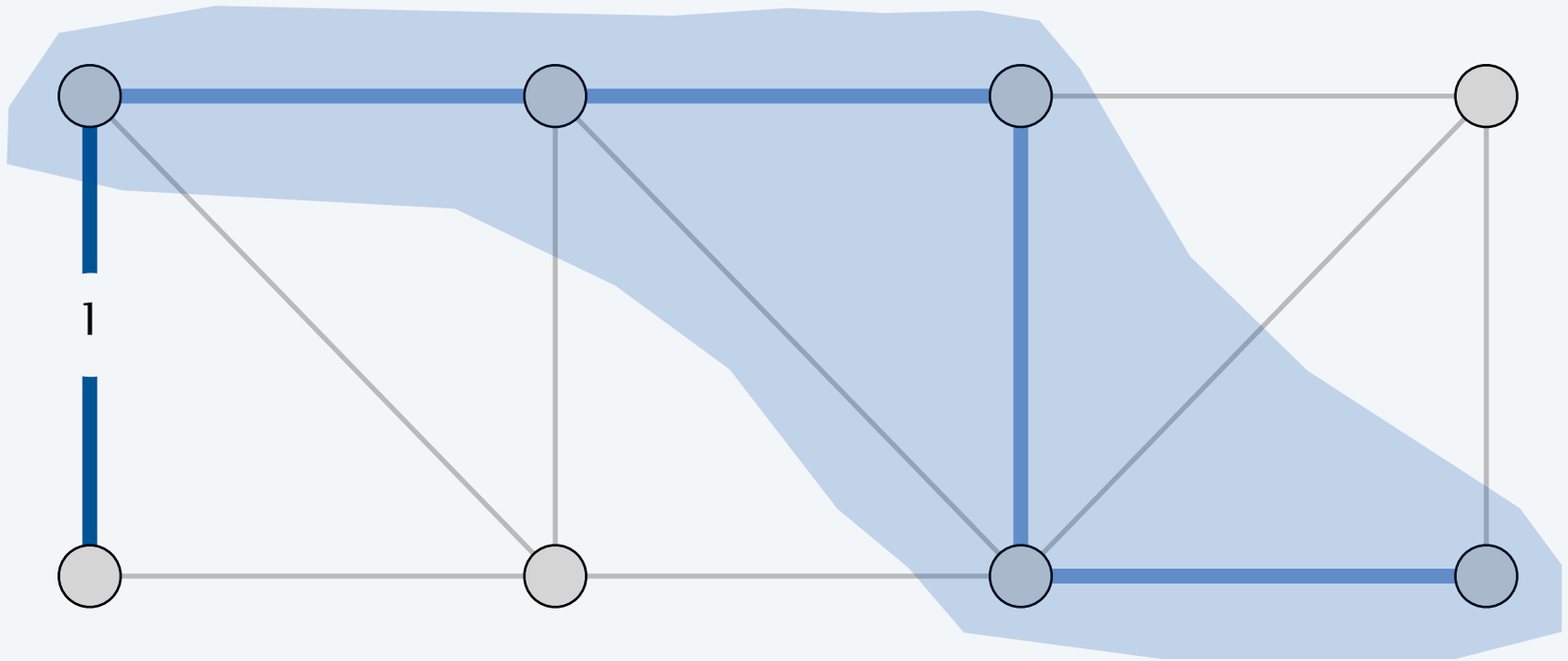


Prim's algorithm demo

Initialize S = any node, $T = (V, \varnothing)$.

Repeat $n - 1$ times:

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

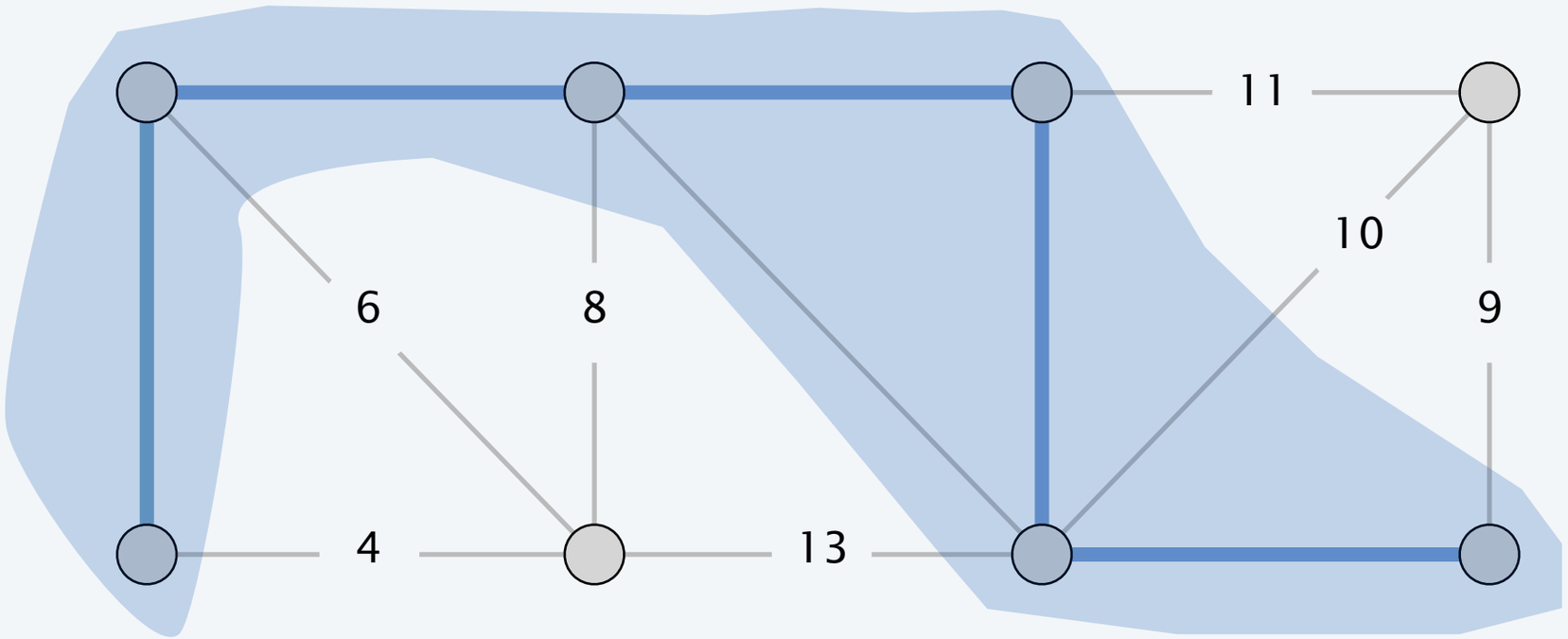


Prim's algorithm demo

Initialize S = any node, $T = (V, \varnothing)$.

Repeat $n - 1$ times:

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

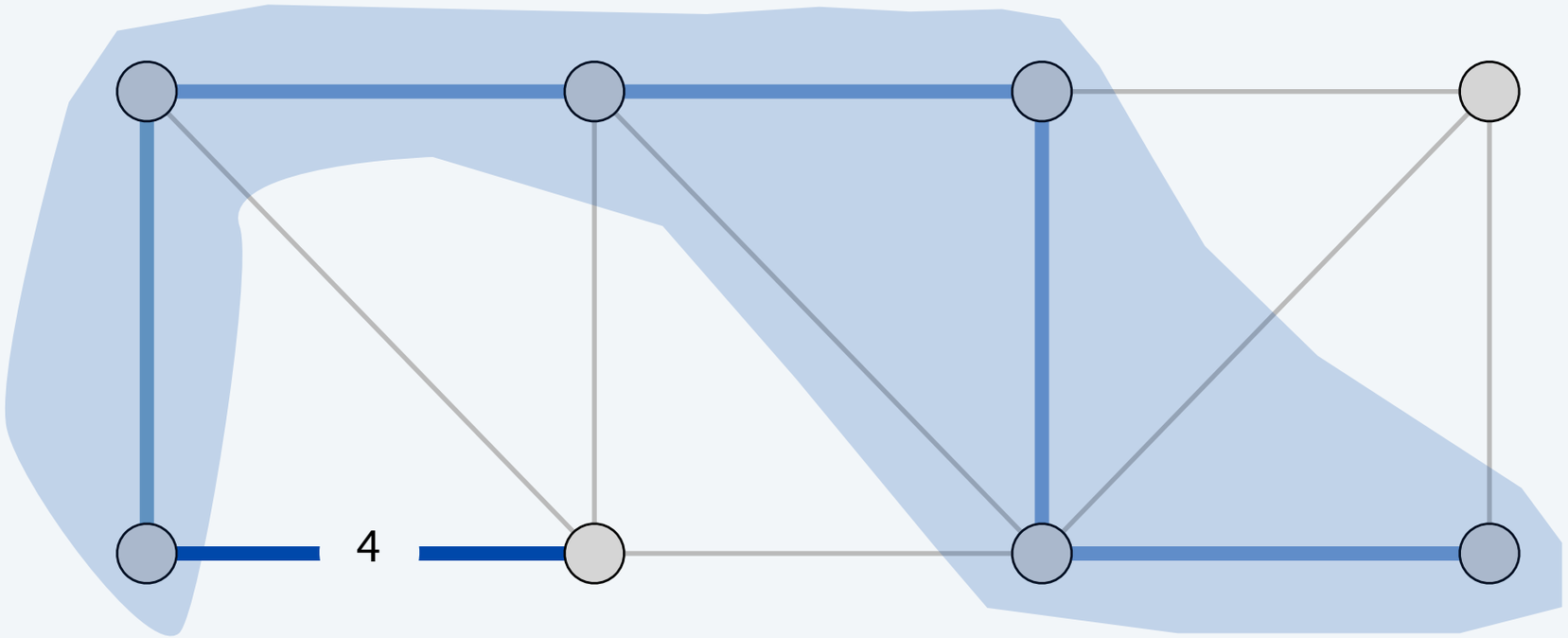


Prim's algorithm demo

Initialize S = any node, $T = (V, \varnothing)$.

Repeat $n - 1$ times:

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

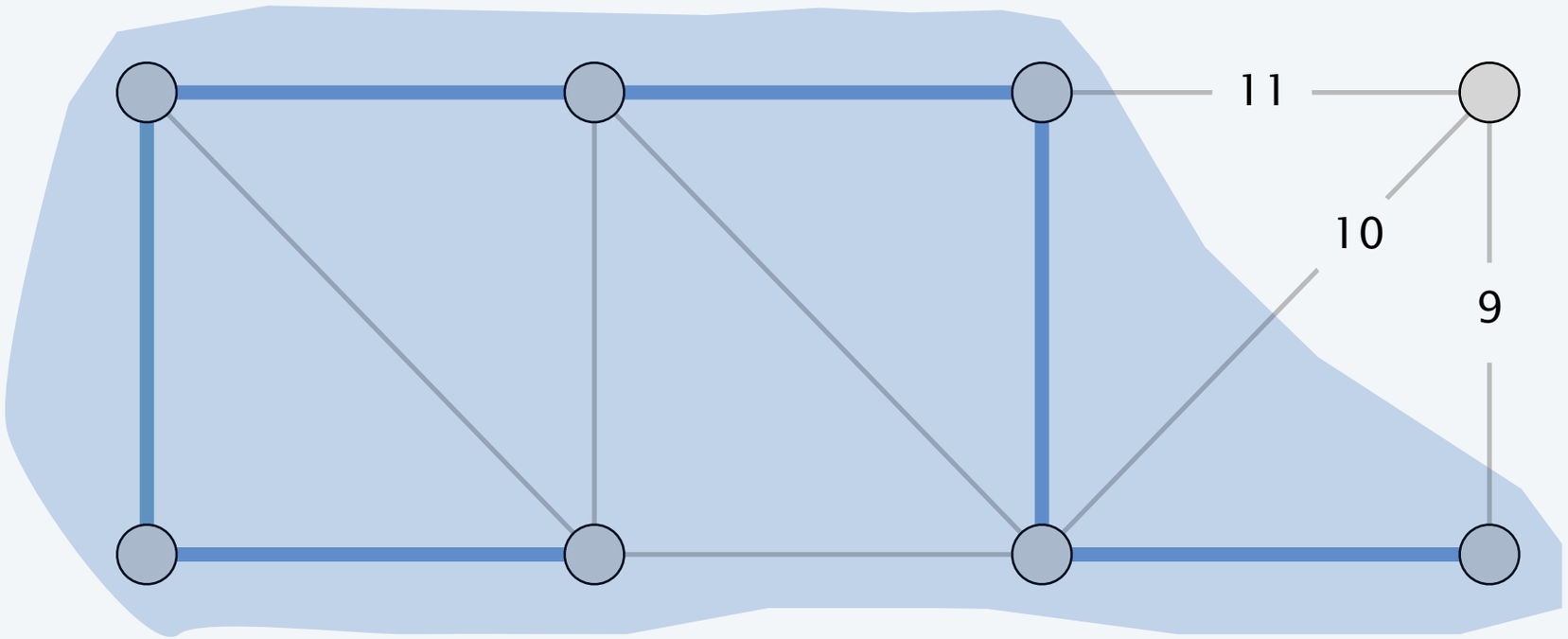


Prim's algorithm demo

Initialize S = any node, $T = (V, \varnothing)$.

Repeat $n - 1$ times:

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

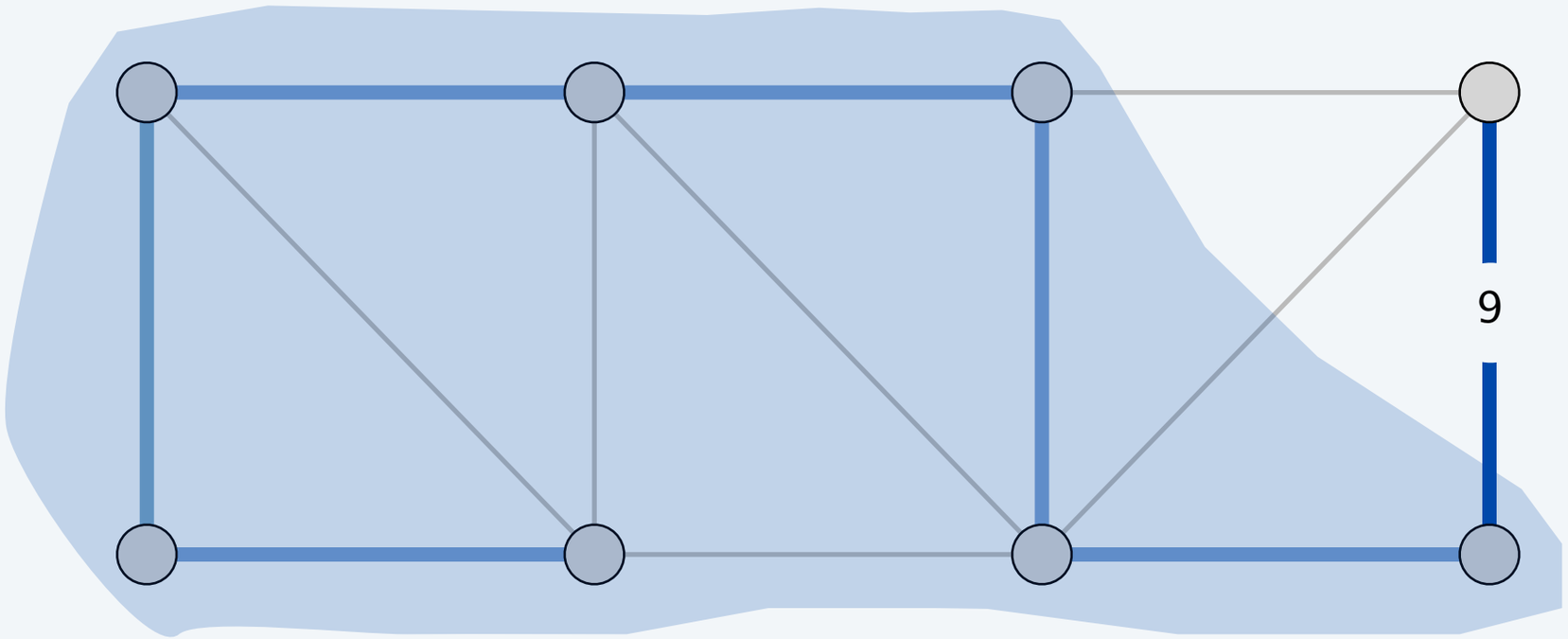


Prim's algorithm demo

Initialize S = any node, $T = (V, \varnothing)$.

Repeat $n - 1$ times:

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

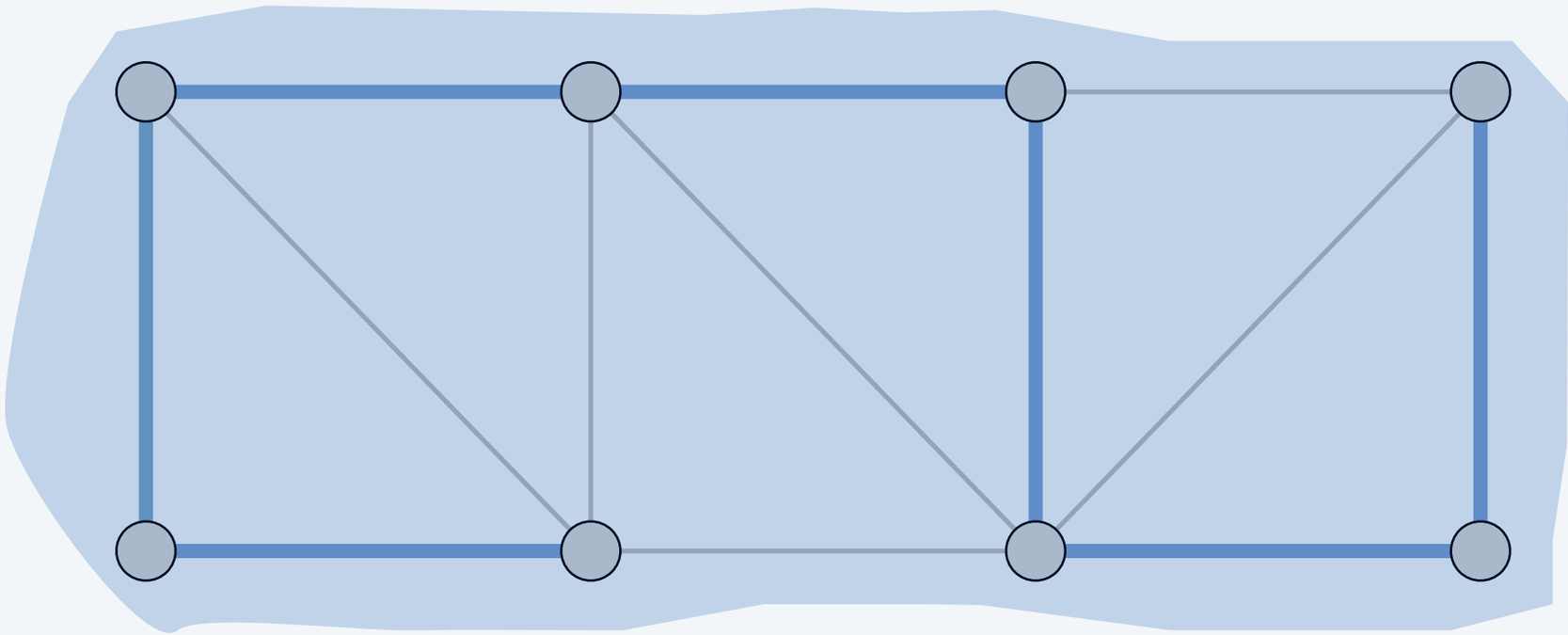


Prim's algorithm demo

Initialize S = any node, $T = (V, \varnothing)$.

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, $T = (V, \varnothing)$.

Repeat $n - 1$ times:

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