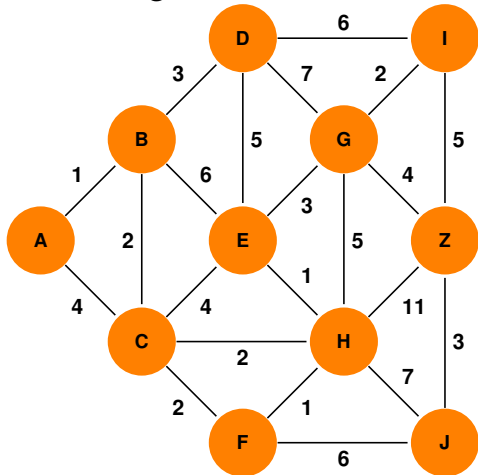


Kürzeste Wege



Wie lang ist der kürzeste Weg von A nach Z?



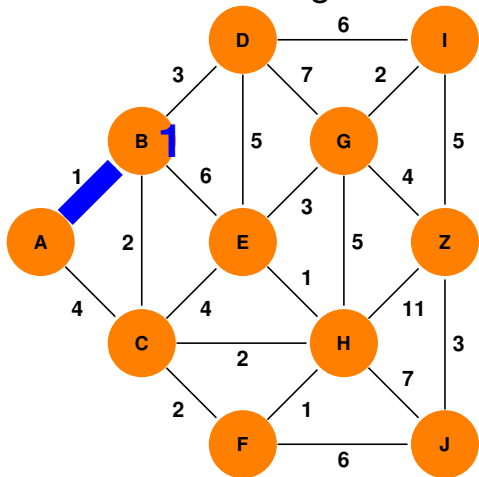
Zahlen neben den Kanten geben die jeweilige Weglängen an.

- 1 4
- 2 9
- 3 11
- 4 12
- 5 13
- 6 14
- 7 15
- 8 16

Kürzeste Wege



Alle kürzesten Wege von A \Rightarrow kürzester Weg $A \rightarrow Z$



Zahlen neben den Kanten geben die jeweilige Weglängen an.

Dijkstra's Algorithmus:

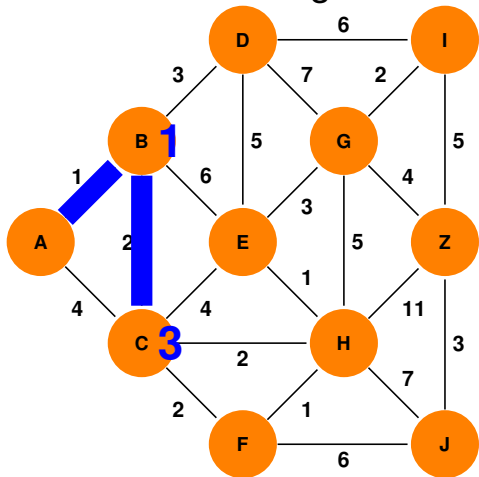
In jedem Schritt, füge den Knoten hinzu, der über die bisherigen Knoten auf kürzestem Weg erreichbar ist!

1	4
2	9
3	11
4	12
5	13
6	14
7	15
8	16

Kürzeste Wege



Alle kürzesten Wege von A \Rightarrow kürzester Weg $A \rightarrow Z$



Zahlen neben den Kanten geben die jeweilige Weglängen an.

Dijkstra's Algorithmus:

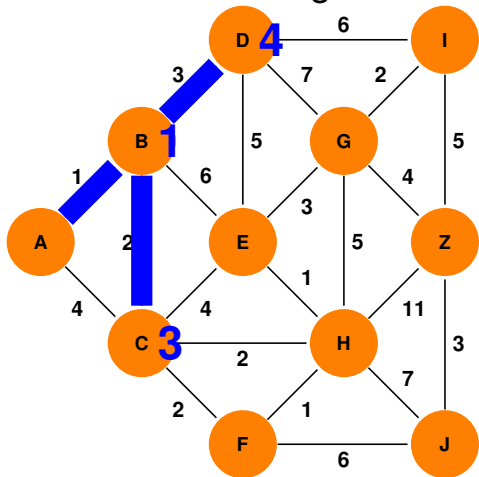
In jedem Schritt, füge den Knoten hinzu, der über die bisherigen Knoten auf kürzestem Weg erreichbar ist!

1	4
2	9
3	11
4	12
5	13
6	14
7	15
8	16

Kürzeste Wege



Alle kürzesten Wege von A \Rightarrow kürzester Weg $A \rightarrow Z$



Zahlen neben den Kanten geben die jeweilige Weglängen an.

Dijkstra's Algorithmus:

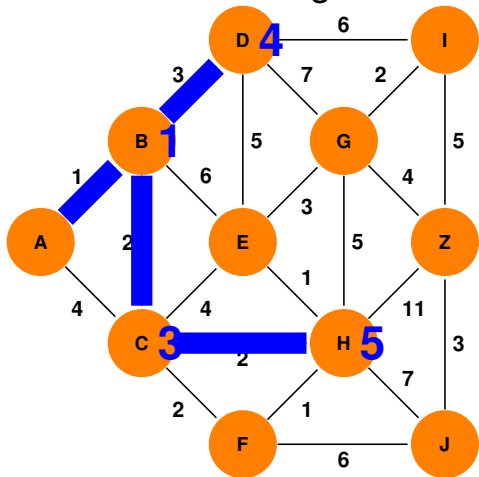
In jedem Schritt, füge den Knoten hinzu, der über die bisherigen Knoten auf kürzestem Weg erreichbar ist!

1	4
2	9
3	11
4	12
5	13
6	14
7	15
8	16

Kürzeste Wege



Alle kürzesten Wege von A \Rightarrow kürzester Weg $A \rightarrow Z$



Zahlen neben den Kanten geben die jeweilige Weglängen an.

Dijkstra's Algorithmus:

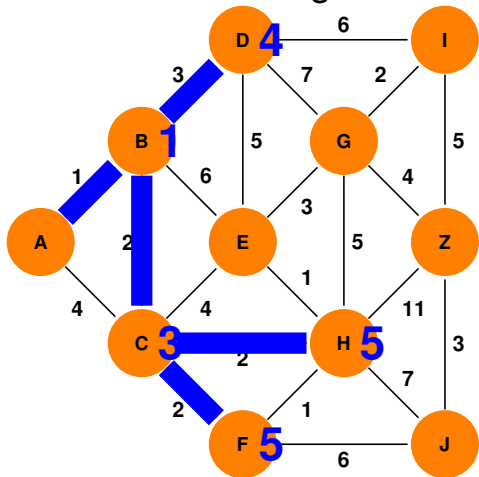
In jedem Schritt, füge den Knoten hinzu, der über die bisherigen Knoten auf kürzestem Weg erreichbar ist!

1	4
2	9
3	11
4	12
5	13
6	14
7	15
8	16

Kürzeste Wege



Alle kürzesten Wege von A \Rightarrow kürzester Weg $A \rightarrow Z$



Zahlen neben den Kanten geben die jeweilige Weglängen an.

Dijkstra's Algorithmus:

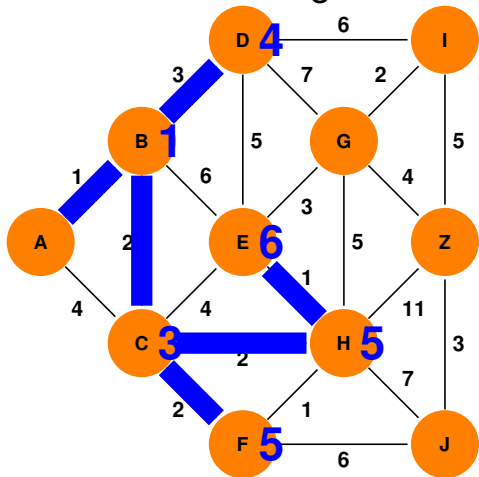
In jedem Schritt, füge den Knoten hinzu, der über die bisherigen Knoten auf kürzestem Weg erreichbar ist!

1	4
2	9
3	11
4	12
5	13
6	14
7	15
8	16

Kürzeste Wege



Alle kürzesten Wege von A \Rightarrow kürzester Weg $A \rightarrow Z$



Zahlen neben den Kanten geben die jeweilige Weglängen an.

Dijkstra's Algorithmus:

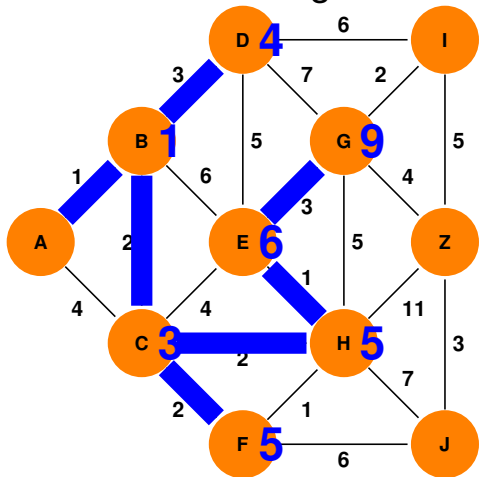
In jedem Schritt, füge den Knoten hinzu, der über die bisherigen Knoten auf kürzestem Weg erreichbar ist!

1	4
2	9
3	11
4	12
5	13
6	14
7	15
8	16

Kürzeste Wege



Alle kürzesten Wege von A \Rightarrow kürzester Weg $A \rightarrow Z$



Zahlen neben den Kanten geben die jeweilige Weglängen an.

Dijkstra's Algorithmus:

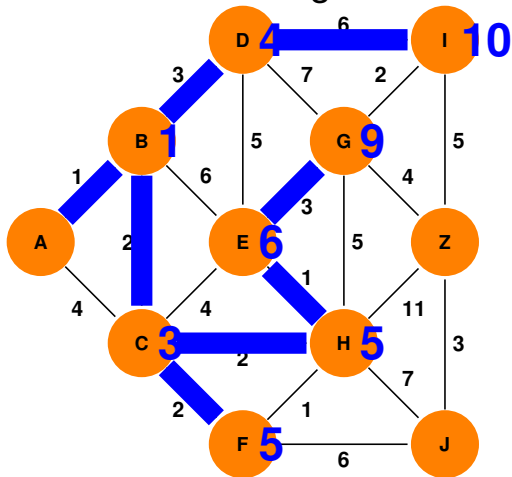
In jedem Schritt, füge den Knoten hinzu, der über die bisherigen Knoten auf kürzestem Weg erreichbar ist!

1	4
2	9
3	11
4	12
5	13
6	14
7	15
8	16

Kürzeste Wege



Alle kürzesten Wege von A \Rightarrow kürzester Weg $A \rightarrow Z$



Zahlen neben den Kanten geben die jeweilige Weglängen an.

Dijkstra's Algorithmus:

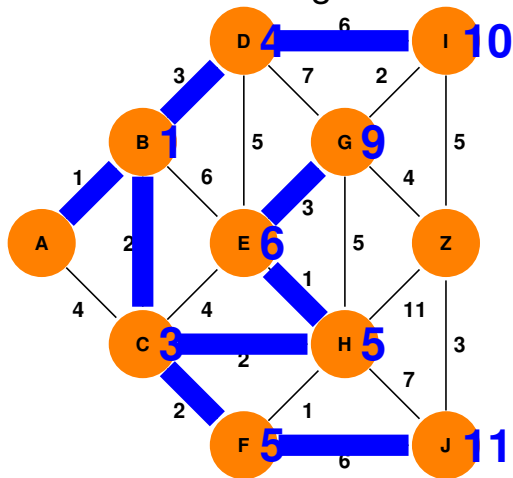
In jedem Schritt, füge den Knoten hinzu, der über die bisherigen Knoten auf kürzestem Weg erreichbar ist!

1	4
2	9
3	11
4	12
5	13
6	14
7	15
8	16

Kürzeste Wege



Alle kürzesten Wege von A \Rightarrow kürzester Weg $A \rightarrow Z$



Zahlen neben den Kanten geben die jeweilige Weglängen an.

Dijkstra's Algorithmus:

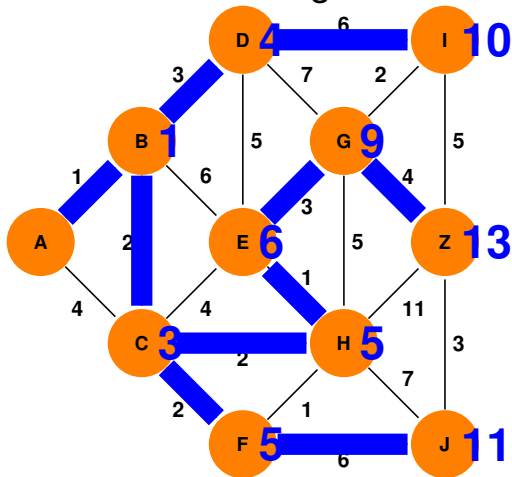
In jedem Schritt, füge den Knoten hinzu, der über die bisherigen Knoten auf kürzestem Weg erreichbar ist!

1	4
2	9
3	11
4	12
5	13
6	14
7	15
8	16

Kürzeste Wege



Alle kürzesten Wege von A \Rightarrow kürzester Weg $A \rightarrow Z$



Zahlen neben den Kanten geben die jeweilige Weglängen an.

Dijkstra's Algorithmus:

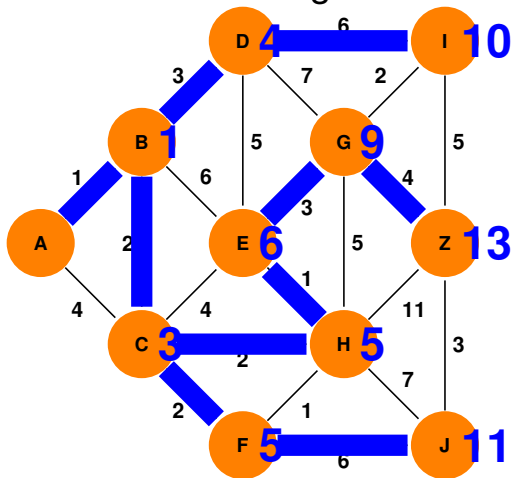
In jedem Schritt, füge den Knoten hinzu, der über die bisherigen Knoten auf kürzestem Weg erreichbar ist!

1	4
2	9
3	11
4	12
5	13
6	14
7	15
8	16

Kürzeste Wege



Alle kürzesten Wege von A \Rightarrow kürzester Weg $A \rightarrow Z$



Zahlen neben den Kanten
geben die jeweilige Weglängen
an.

1	4
2	9
3	11
4	12
5	13
6	14
7	15
8	16

