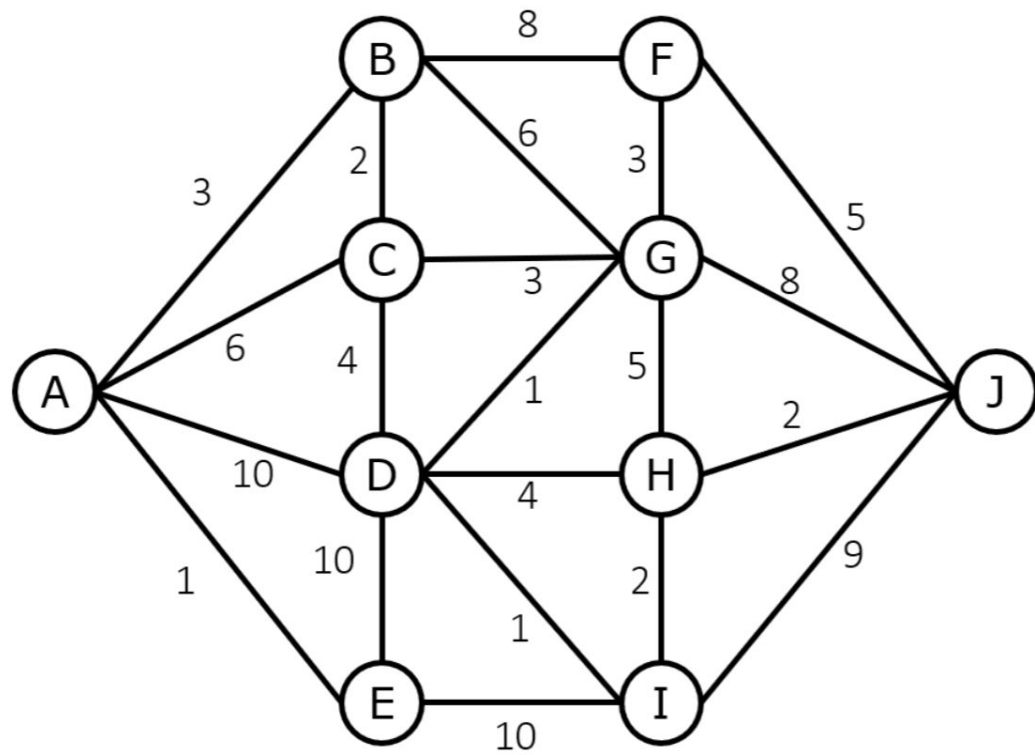


# dijkstra Algorithmus

## händisch



Wir fangen bei Vertex A an. A ist fertig/selected, hat Distanz 0 zu sich selbst und als Vorgänger sich selbst.

Vertex	done / selected	distance	predecessor
-> A	✓	0	A
B	✗	$\infty$	x
C	✗	$\infty$	x
D	✗	$\infty$	x
E	✗	$\infty$	x
F	✗	$\infty$	x
G	✗	$\infty$	x
H	✗	$\infty$	x
I	✗	$\infty$	x
J	✗	$\infty$	x

Wir suchen nun alle von A aus erreichbaren Nodes.

Vertex	done / selected	distance	predecessor
-> A	✓	0	A
x B	✗	$\infty$	x
x C	✗	$\infty$	x
x D	✗	$\infty$	x
x E	✗	$\infty$	x
F	✗	$\infty$	x
G	✗	$\infty$	x
H	✗	$\infty$	x
I	✗	$\infty$	x
J	✗	$\infty$	x

Wir tragen das Gewicht ("Distanz") zu allen Vertices ein, die von A aus erreichbar sind. Zusätzlich setzen wir den predecessor von ihnen auf A.

Vertex	done / selected	distance	predecessor
-> A	✓	0	A
x B	✗	$\infty$ , 3	✗, A
x C	✗	$\infty$ , 6	✗, A
x D	✗	$\infty$ , 10	✗, A
x E	✗	$\infty$ , 1	✗, A
F	✗	$\infty$	x
G	✗	$\infty$	x
H	✗	$\infty$	x
I	✗	$\infty$	x
J	✗	$\infty$	x

Die alten Distanzen und Vorgänger (predecessors) können durchgestrichen werden.

Vertex	done / selected	distance	predecessor
-> A	✓	0	A
x B	✗	<del>∞</del> , 3	<del>∞</del> , A
x C	✗	<del>∞</del> , 6	<del>∞</del> , A
x D	✗	<del>∞</del> , 10	<del>∞</del> , A
x E	✗	<del>∞</del> , 1	<del>∞</del> , A
F	✗	∞	x
G	✗	∞	x
H	✗	∞	x
I	✗	∞	x
J	✗	∞	x

Jetzt wählen wir den Vertex mit der kleinsten Distanz aus, das ist E.  
 E ist jetzt fertig / selected. Wir markieren E als letzten Vertex.

Vertex	done / selected	distance	predecessor
A	✓	0	A
B	✗	$\infty$ , 3	✗, A
C	✗	$\infty$ , 6	✗, A
D	✗	$\infty$ , 10	✗, A
-> E	✓	$\infty$ , 1	✗, A
F	✗	$\infty$	x
G	✗	$\infty$	x
H	✗	$\infty$	x
I	✗	$\infty$	x
J	✗	$\infty$	x

Wir suchen nun alle Vertices die von E aus erreicht werden können, außer denen die schon fertig / selected sind.

Vertex	done / selected	distance	predecessor
A	✓	0	A
B	✗	∞, 3	✗, A
C	✗	∞, 6	✗, A
x D	✗	∞, 10	✗, A
-> E	✓	∞, 1	✗, A
F	✗	∞	x
G	✗	∞	x
H	✗	∞	x
x I	✗	∞	x
J	✗	∞	x



Wir addieren nun die optimale Distanz zu E (1) zur Distanz von E zum jeweiligen Zielvertex (D und I). Ist der neue Wert kleiner, so tragen wir ihn ein und streichen die alten Werte. Der Vorgänger dieser Vertices wird auf E gesetzt.

Vertex	done / selected	distance	predecessor
A	✓	0	A
B	✗	∞, 3	✗, A
C	✗	∞, 6	✗, A
x D	✗	∞, 10	✗, A
-> E	✓	∞, 1	✗, A
F	✗	∞	x
G	✗	∞	x
H	✗	∞	x
x I	✗	∞, 11	✗, E
J	✗	∞	x

Wir selektieren nun wieder den kleinsten Wert, also B.

Vertex	done / selected	distance	predecessor
A	✓	0	A
-> B	✓	∞, 3	x, A
C	✗	∞, 6	x, A
D	✗	∞, 10	x, A
E	✓	∞, 1	x, A
F	✗	∞	x
G	✗	∞	x
H	✗	∞	x
I	✗	∞, 11	x, E
J	✗	∞	x

Wir suchen Vertices die von B aus erreicht werden können.

Vertex	done / selected	distance	predecessor
A	✓	0	A
-> B	✓	$\infty$ , 3	x, A
x C	✗	$\infty$ , 6	x, A
D	✗	$\infty$ , 10	x, A
E	✓	$\infty$ , 1	x, A
x F	✗	$\infty$	x
x G	✗	$\infty$	x
H	✗	$\infty$	x
I	✗	$\infty$ , 11	x, E
J	✗	$\infty$	x

Wir berechnen die neuen Abstände und ersetzen falls wir einen kleineren Wert entdecken, auch der Vorgänger der Zielvertices wird ersetzt.

Vertex	done / selected	distance	predecessor
A	✓	0	A
-> B	✓	$\infty$ , 3	x, A
x C	✗	$\infty$ , 6, 5	A, B
D	✗	$\infty$ , 10	x, A
E	✓	$\infty$ , 1	x, A
x F	✗	$\infty$ , 11	x, B
x G	✗	$\infty$ , 9	x, B
H	✗	$\infty$	x
I	✗	$\infty$ , 11	x, E
J	✗	$\infty$	x

Wir selektieren wieder den kleinsten Wert, C.

Vertex	done / selected	distance	predecessor
A	✓	0	A
B	✓	∞, 3	x, A
-> C	✓	∞, 6, 5	A, B
D	✗	∞, 10	x, A
E	✓	∞, 1	x, A
F	✗	∞, 11	x, B
G	✗	∞, 9	x, B
H	✗	∞	x
I	✗	∞, 11	x, E
J	✗	∞	x

Wir suchen die von C aus erreichbaren Vertices.

Vertex	done / selected	distance	predecessor
A	✓	0	A
B	✓	$\infty$ , 3	x, A
-> C	✓	$\infty$ , 6, 5	A, B
x D	✗	$\infty$ , 10, 9	x, A, C
E	✓	$\infty$ , 1	x, A
F	✗	$\infty$ , 11	x, B
x G	✗	$\infty$ , 9, 8	x, B, C
H	✗	$\infty$	x
I	✗	$\infty$ , 11	x, E
J	✗	$\infty$	x

Wir selektieren den kleinsten Wert und ersetzen die Vorgänger.

Vertex	done / selected	distance	predecessor
A	✓	0	A
B	✓	∞, 3	x, A
C	✓	∞, 6, 5	A, B
D	✗	∞, 10, 9	x, A, C
E	✓	∞, 1	x, A
F	✗	∞, 11	x, B
-> G	✓	∞, 9, 8	x, B, C
H	✗	∞	x
I	✗	∞, 11	x, E
J	✗	∞	x

Wir suchen von G aus erreichbare Vertices und die tragen die neue Distanz ein, falls kleiner auch die Vorgänger werden ersetzt.

Vertex	done / selected	distance	predecessor
A	✓	0	A
B	✓	$\infty$ , 3	$\times$ , A
C	✓	$\infty$ , 6, 5	A, B
x D	✗	$\infty$ , <del>10</del> , 9	$\times$ , A, C
E	✓	$\infty$ , 1	$\times$ , A
x F	✗	$\infty$ , 11	$\times$ , B
-> G	✓	$\infty$ , 9, 8	$\times$ , B, C
x H	✗	$\infty$ , 13	$\times$ , G
I	✗	$\infty$ , 11	$\times$ , E
x J	✗	$\infty$ , 16	$\times$ , G



Wir wählen D aus.

Vertex	done / selected	distance	predecessor
A	✓	0	A
B	✓	∞, 3	*, A
C	✓	∞, 6, 5	A, B
-> D	✓	∞, <del>10</del> , 9	*, A, C
E	✓	∞, 1	*, A
F	✗	∞, 11	*, B
G	✓	∞, 9, 8	*, B, C
H	✗	∞, 13	*, G
I	✗	∞, 11	*, E
J	✗	∞, 16	*, G

...

Vertex	done / selected	distance	predecessor
A	✓	0	A
B	✓	$\infty$ , 3	$\times$ , A
C	✓	$\infty$ , 6, 5	A, B
-> D	✓	$\infty$ , <del>10</del> , 9	$\times$ , A, C
E	✓	$\infty$ , 1	$\times$ , A
F	✗	$\infty$ , 11	$\times$ , B
G	✓	$\infty$ , 9, 8	$\times$ , B, C
x H	✗	$\infty$ , 13	$\times$ , G
x I	✗	$\infty$ , <del>11</del> , 10	$\times$ , E, D
J	✗	$\infty$ , 16	$\times$ , G

...

Vertex	done / selected	distance	predecessor
A	✓	0	A
B	✓	$\infty$ , 3	$\times$ , A
C	✓	$\infty$ , 6, 5	A, B
D	✓	$\infty$ , <del>10</del> , 9	$\times$ , A, C
E	✓	$\infty$ , 1	$\times$ , A
F	✗	$\infty$ , 11	$\times$ , B
G	✓	$\infty$ , 9, 8	$\times$ , B, C
H	✗	$\infty$ , 13	$\times$ , G
-> I	✓	$\infty$ , <del>11</del> , 10	$\times$ , E, D
J	✗	$\infty$ , 16	$\times$ , G

...

Vertex	done / selected	distance	predecessor
A	✓	0	A
B	✓	$\infty$ , 3	$\times$ , A
C	✓	$\infty$ , 6, 5	A, B
D	✓	$\infty$ , <del>10</del> , 9	$\times$ , A, C
E	✓	$\infty$ , 1	$\times$ , A
F	✗	$\infty$ , 11	$\times$ , B
G	✓	$\infty$ , 9, 8	$\times$ , B, C
$\times$ H	✗	$\infty$ , <del>13</del> , 12	$\times$ , G, I
-> I	✓	$\infty$ , <del>11</del> , 10	$\times$ , E, D
$\times$ J	✗	$\infty$ , 16	$\times$ , G

...

Vertex	done / selected	distance	predecessor
A	✓	0	A
B	✓	$\infty$ , 3	$\times$ , A
C	✓	$\infty$ , 6, 5	A, B
D	✓	$\infty$ , <del>10</del> , 9	$\times$ , A, C
E	✓	$\infty$ , 1	$\times$ , A
-> F	✓	$\infty$ , 11	$\times$ , B
G	✓	$\infty$ , 9, 8	$\times$ , B, C
H	✗	$\infty$ , <del>13</del> , 12	$\times$ , G, I
I	✓	$\infty$ , <del>11</del> , 10	$\times$ , E, D
J	✗	$\infty$ , 16	$\times$ , G

...

Vertex	done / selected	distance	predecessor
A	✓	0	A
B	✓	$\infty$ , 3	$\times$ , A
C	✓	$\infty$ , 6, 5	A, B
D	✓	$\infty$ , <del>10</del> , 9	$\times$ , A, C
E	✓	$\infty$ , 1	$\times$ , A
-> F	✓	$\infty$ , 11	$\times$ , B
G	✓	$\infty$ , 9, 8	$\times$ , B, C
H	✗	$\infty$ , <del>13</del> , 12	$\times$ , G, I
I	✓	$\infty$ , <del>11</del> , 10	$\times$ , E, D
x J	✗	$\infty$ , 16	$\times$ , G

...

Vertex	done / selected	distance	predecessor
A	✓	0	A
B	✓	$\infty$ , 3	$\times$ , A
C	✓	$\infty$ , 6, 5	A, B
D	✓	$\infty$ , <del>10</del> , 9	$\times$ , A, C
E	✓	$\infty$ , 1	$\times$ , A
F	✓	$\infty$ , 11	$\times$ , B
G	✓	$\infty$ , 9, 8	$\times$ , B, C
-> H	✓	$\infty$ , <del>13</del> , 12	$\times$ , G, I
I	✓	$\infty$ , <del>11</del> , 10	$\times$ , E, D
J	✗	$\infty$ , 16	$\times$ , G

...

Vertex	done / selected	distance	predecessor
A	✓	0	A
B	✓	$\infty$ , 3	$\times$ , A
C	✓	$\infty$ , 6, 5	A, B
D	✓	$\infty$ , <del>10</del> , 9	$\times$ , A, C
E	✓	$\infty$ , 1	$\times$ , A
F	✓	$\infty$ , 11	$\times$ , B
G	✓	$\infty$ , 9, 8	$\times$ , B, C
-> H	✓	$\infty$ , <del>13</del> , 12	$\times$ , G, I
I	✓	$\infty$ , <del>11</del> , 10	$\times$ , E, D
x J	✗	$\infty$ , 16	$\times$ , G



...

Vertex	done / selected	distance	predecessor
A	✓	0	A
B	✓	$\infty$ , 3	x, A
C	✓	$\infty$ , 6, 5	A, B
D	✓	$\infty$ , <del>10</del> , 9	x, A, C
E	✓	$\infty$ , 1	x, A
F	✓	$\infty$ , 11	x, B
G	✓	$\infty$ , 9, 8	x, B, C
-> H	✓	$\infty$ , <del>13</del> , 12	x, G, I
I	✓	$\infty$ , <del>11</del> , 10	x, E, D
x J	✗	$\infty$ , <del>16</del> , 14	x, G, H

Fertig! = alle Vertices done/selected!  
(Die predecessor und distance "history" hat keinerlei Bedeutung mehr)

Vertex	done / selected	distance	predecessor
A	✓	0	A
B	✓	∞, 3	✕, A
C	✓	∞, 6, 5	A, B
D	✓	∞, 10, 9	✕, A, C
E	✓	∞, 1	✕, A
F	✓	∞, 11	✕, B
G	✓	∞, 9, 8	✕, B, C
H	✓	∞, 13, 12	✕, G, I
I	✓	∞, 11, 10	✕, E, D
-> J	✓	∞, 16, 14	✕, G, H