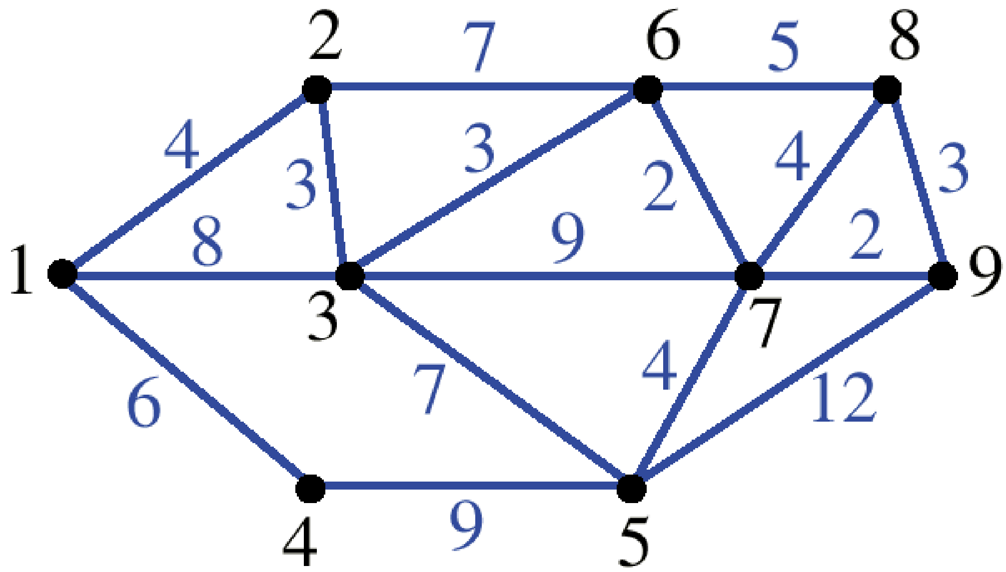
DIJKSTRA'S ALGORITHM

Dijkstra's algorithm involves finding a spanning tree that gives the minimum weight from the starting vertex to each vertex in the graph. Named after Professor Edsger Wybe Dijkstra (pronounced 'dikestraw') 1930-2002. Find the shortest path from vertex 1 to vertex 9.



1. Start with the shortest edge connected to the starting vertex, in this case edge 1-2 that has distance 4.

Consider each edge that we could connect to our tree without creating a cycle and calculate the total distance from the start to the vertex that would be added.

2. Our tree so far just consists of vertices 1 and 2 and edge 1-2. We could:
a. Add 1-3, this would create a path from 1 to 3 of length 8
b. Add 1-4, this would create a path from 1 to 4 of length 6*
c. Add 2-6, this would create a path from 1 to 6 via 2 of length 4+7 = 11
d. Add 2-3, this would create a path from 1 to 3 via 2 of length 4+3 = 7

Adding edge 1-4 gives us the lowest total length from our starting point, so we add 1-4 to our tree.

3. Our tree now consists of edges 1-2 and 1-4. We could:
a. Add 4-5, this would create a path 1-4-5 of length 6+9 = 15
b. Add 1-3, this would create a path 1-3 of length 8
c. Add 2-6, this would create a path 1-2-6 of length 4+7 = 11
d. Add 2-3, this would create a path 1-2-3 of length 4+3 = 7*

Therefore we add edge 2-3 to our tree.

4. Our tree now consists of edges 1-2, 1-4, and 2-3. Note that we cannot now add edge 1-3 to our tree because that would create a cycle. We could:
a. Add 4-5, this would create a path 1-4-5 of length 6+9 = 15
b. Add 3-5, this would create a path 1-2-3-5 of length 7+7 = 14
c. Add 3-7, this would create a path 1-2-3-7 of length 7+9 = 16
d. Add 3-6, this would create a path 1-2-3-6 of length 7+3 = 10*
e. Add 2-6, this would create a path 1-2-6 of length 4+7 = 11

So we add edge 3-6 to our tree.

5. We now have edges 1-2, 1-4, 2-3, and 3-6 in our tree. We cannot now add edge 2-6 to this because it would create a cycle. We could:
a. Add 4-5, this would create a path 1-4-5 of length 6+9 = 15
b. Add 3-5, this would create a path 1-2-3-5 of length 7+7 = 14
c. Add 3-7, this would create a path 1-2-3-7 of length 7+9 = 16
d. Add 6-8, this would create a path 1-2-3-6-8 of length 10+5 = 15
e. Add 6-7, this would create a path 1-2-3-6-7 of length 10+2 = 12*

6. We now have edges 1-2, 1-4, 2-3, 3-6, and 6-7 in our tree. We could:
a. Add 4-5, this would create a path 1-4-5 of length 6+9 = 15
b. Add 3-5, this would create a path 1-2-3-5 of length 7+7 = 14*
c. Add 7-5, this would create a path 1-2-3-6-7-5 of length 12+4 = 16
d. Add 6-8, this would create a path 1-2-3-6-8 of length 10+5 = 15
e. Add 7-8, this would create a path 1-2-3-6-7-8 of length 12+4 = 16
f. Add 7-9, this would create a path 1-2-3-6-7-9 of length 12+2 = 14*

We have a choice of either adding edge 3-5 to connect to vertex 5 or adding edge 7-9 to connect to vertex 9. In this case, because vertex 9 is our end point, we add the edge 7-9 to complete the path to our end point. Note that this path is a tree.

The final minimum distance path is 1-2-3-6-7-9 of total length 14.

Note that we can continue this algorithm to find the minimum distance from 1 to all the remaining vertices.

7. From the tree 1-2, 1-4, 2-3, 3-6, 6-7, and 7-9, we could:
a. Add 4-5, this would create a path 1-4-5 of length 6+9 = 15
b. Add 3-5, this would create a path 1-2-3-5 of length 7+7 = 14*
c. Add 7-5, this would create a path 1-2-3-5 of length 12+4 = 16
d. Add 9-5, this would create a path 1-2-3-6-7-9-5 of length 14+12 = 26
e. Add 6-8, this would create a path 1-2-3-6-8 of length 10+5 = 15
f. Add 7-8, this would create a path 1-2-3-6-7-8 of length 12+4 = 16
g. Add 9-8, this would create a path 1-2-3-6-7-9-8 of length 14+3 = 17

So we add edge 3-5 to complete the path to node 5 with total length 14. The final step is to add edge 6-8 to connect to vertex 8 with total length 15.