DIJKSTRA'S ALGORITHM

DEFINITION

Dijkstra's algorithm allows us to find the shortest path between any two vertices of a graph.

It differs from the minimum spanning tree because the shortest distance between two vertices might not include all the vertices of the graph.

HOW IT WORKS

Starts at the node that we give as a parameter and it will return the shortest path between this node and all the other nodes (or vertexes) in the graph.

It calculates the shortest distance from each node to the source and saves this value if it finds a shorter path that the path that it had saved before. It calculates the distance between a node and the origin node, if this distance is less than it has been saved before, the new minimum distance will be the new distance.

HOW IT WORKS

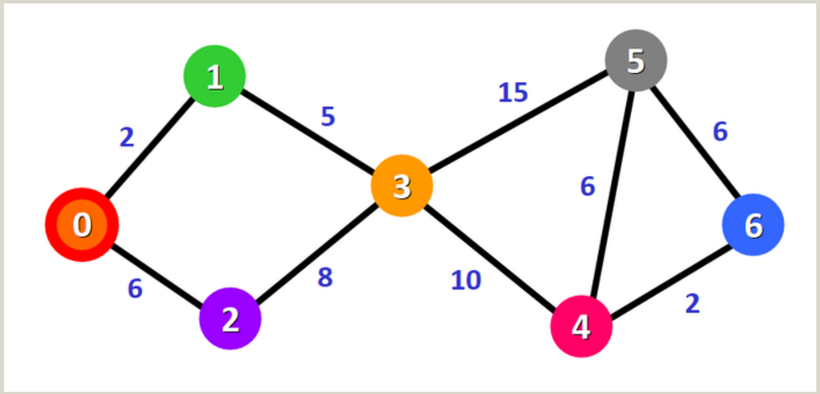
Once Dijkstra's algorithm has found the shortest path between the origin node and another node, it marks the node as visited (if it didn't do it the algorithm could enter into an infinite loop).

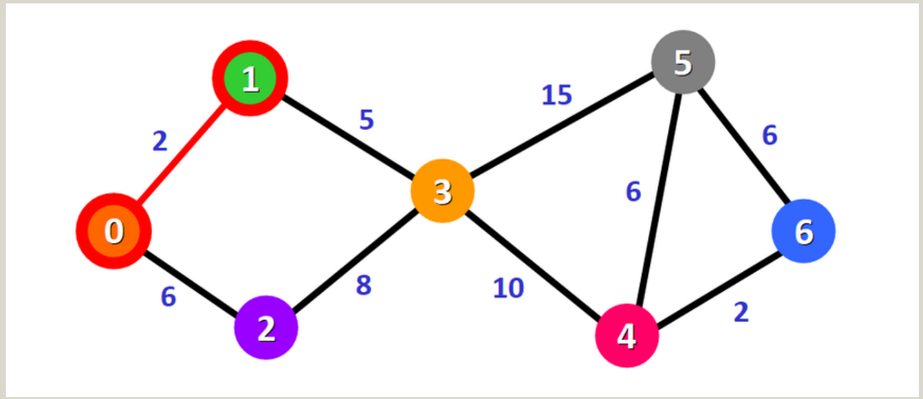
Steps 2 and 3 are repeated until all the nodes are visited. This way, we have visited all the nodes and we've saved the shortest path possible to reach each node.

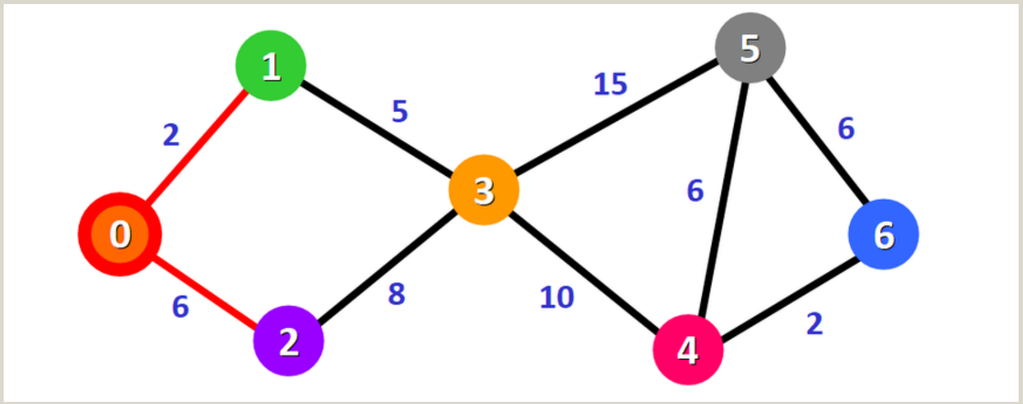
HISTORY

What is the shortest way to travel from Rotterdam to Groningen, in general: from given city to given city. It is the algorithm for the shortest path, which I designed in about twenty minutes. One morning I was shopping in Amsterdam with my young fiancée, and tired, we sat down on the café terrace to drink a cup of coffee and I was just thinking about whether I could do this, and I then designed the algorithm for the shortest path. As I said, it was a twenty-minute invention. In fact, it was published in '59, three years later. The publication is still readable, it is, in fact, quite nice. One of the reasons that it is so nice was that I designed it without pencil and paper. I learned later that one of the advantages of designing without pencil and paper is that you are almost forced to avoid all avoidable complexities. Eventually, that algorithm became to my great amazement, one of the cornerstones of my fame. -Edsger Dijkstra in an interview with Philip L. Frana, Communications of the ACM, 2001[

Example


0
1
2
3
4
5
6
2
5
15
6
6
6
8
10
2


0
1
2
3
4
5
6
2
5
15
6
6
6
8
10
2


0
1
2
3
4
5
6
2
5
15
6
6
6
8
10
2

PSEUDOCODE

To maintain the path distance of every vertex. We can store that in an array of size v, where v is the number of vertices.

We also want to be able to get the shortest path, not only know the length of the shortest path. For this, we map each vertex to the vertex that last updated its path length.

Once the algorithm is over, we can backtrack from the destination vertex to the source vertex to find the path.

THANK YOU FOR LISTENING