Technical Description of the Process of Dijkstra's Algorithm

By: Joseph Hudson, Stamatios Morellas, Atharva Kushare

Introduction

Dijkstra's Algorithm is a notorious computer science algorithm used for finding the shortest, most cost-efficient path between nodes connected by edges in an undirected, weighted graph. This algorithm will start the search at an initial node, search through the graph, and return the most cost-efficient path to an end node. This algorithm is used in real-life applications in the mapping and telecommunication services.

Discussion

Step I: Initializing properties and values

To start the process of Dijkstra's algorithm every node will possess two values which are a distance integer value and a visited truth value. To initialize all these values we will set all distances to infinite and all visited values to false.

Step II: Select initial node in the path

Once all values are set the Dijkstra's algorithm can begin. With the design of this algorithm the first node is manually picked by the user. In figure 1 it is displayed that node A has been picked as the initial node. With the decided node being picked its distance value is set to zero and can be considered visited and requires its neighbors, or in programming terms edges, need to be analyzed.

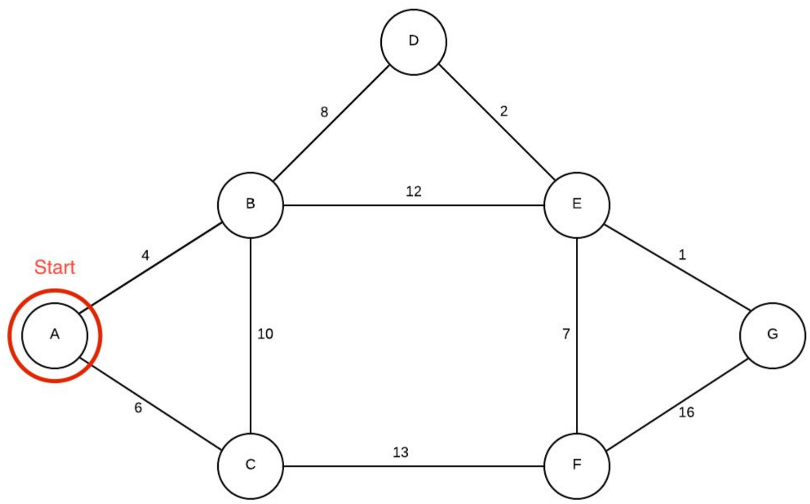

Figure 1: Start at node A

Step III: Analyze current node's connections to other nodes

With the current node that is currently being focused on it is now time to look at its edges. The order in which the edges are analyzed does not matter as the conditions which determine whether or not values change will have the end result be the same. The first condition when determining whether or not values can be altered is that the other node connected to the must not have been visited yet and if this is the case the next condition can be used to determine if any values change. The second and last condition that must be met is that the sum of the current node's distance and the edge's cost must be less than the other node's on the edge distance value. If both of these conditions are met the node on the edge can have its distance value changed to the sum of the distance of the current node and the cost of the edge. In figure 2, with node A's distance being 0, node B's distance being infinite, and the edge between A and B has a cost of 4, the distance value for node B can be set to 4 as B has yet to be visited, and 4 is less than infinite. Once this is repeated with all the edges the next node can now be decided.

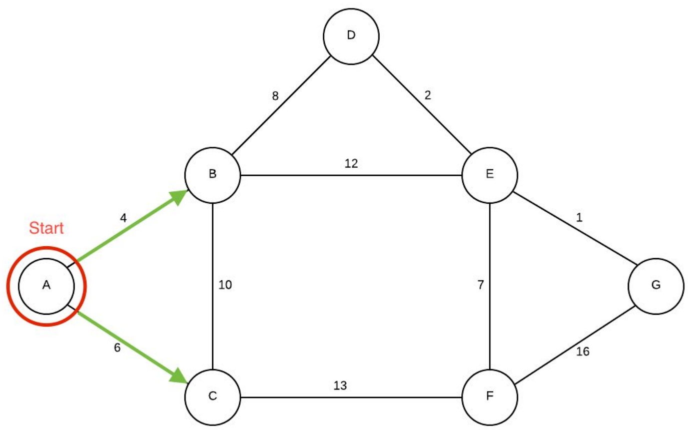

Figure 2: Analyze the outgoing edges

Step IV: Decide the next node in the path to analyze

Upon the updating of the values of the edges the current node can now have its visited truth value to be set to true, and the next node can now be decided. With the desired output being the shortest path, the node with the smallest distance value and the visited truth value being false shall be chosen as the next node to analyze. If there is no node that has a false visited truth value then there shall be no next node to analyze.

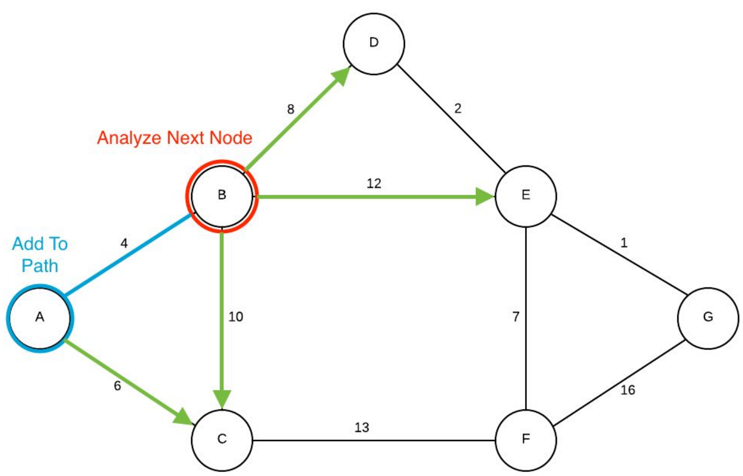

Figure 3: Move along the shortest path to the next node and scan the outgoing edges

Part V: Repeat steps III and IV with the next node until no more nodes are left in the path

With the previous node's analysis being completed, the next node to be analyzed being decided, and this being a recursive algorithm, the process needs to repeat until the desired outcome is found. Once all nodes have been analyzed the algorithm is complete.

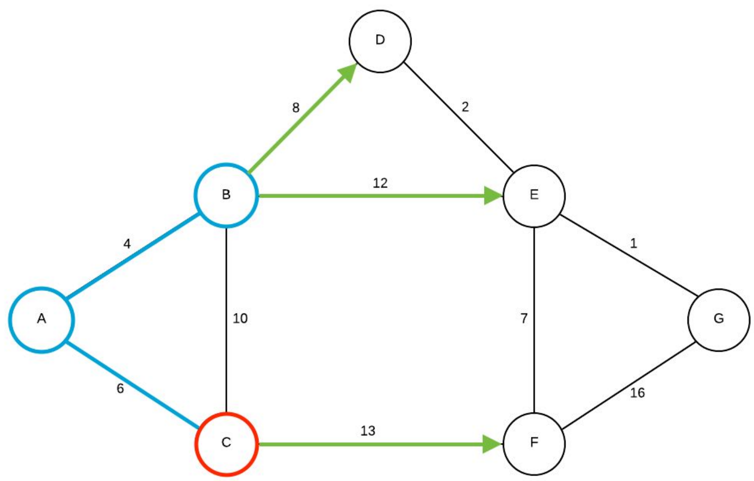

Figure 4: Node C is the current node and its edges are taken into account

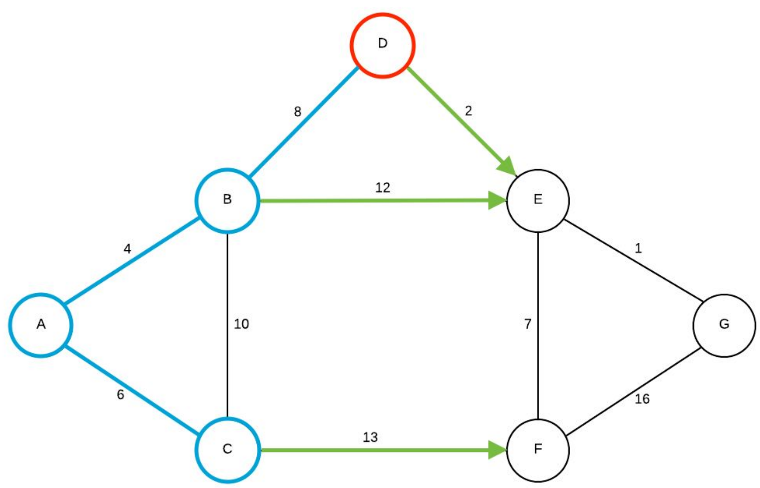

Figure 5:The process continues with D as the current node

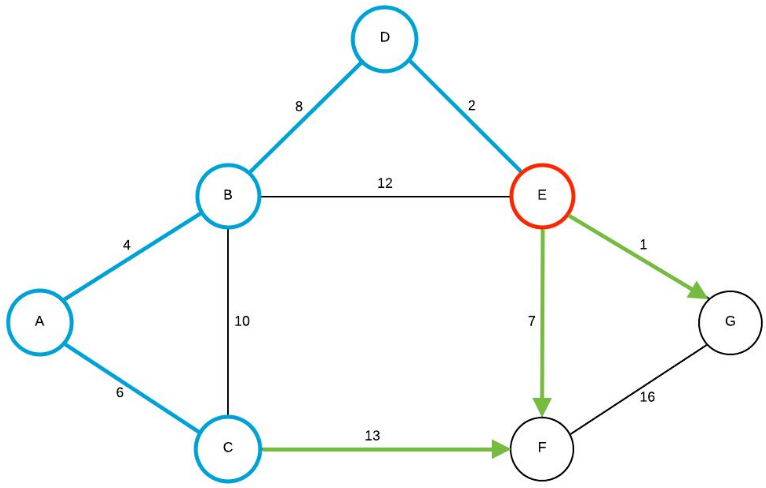

Figure 6: The paths continue with E as the current node

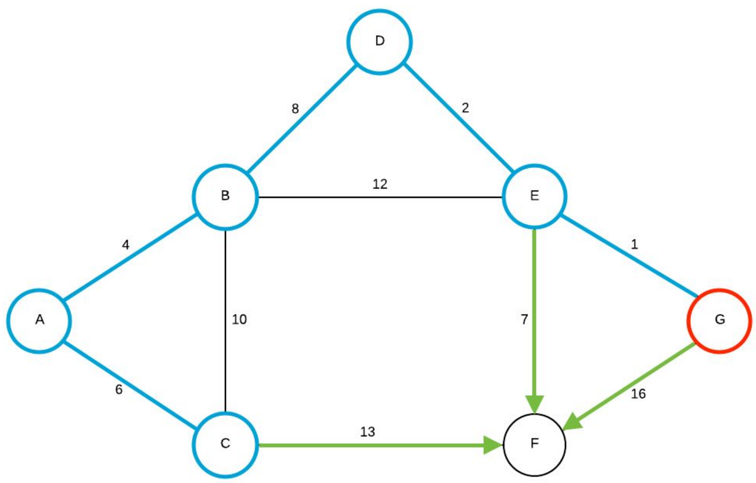

Figure 7: The process repeats itself with G as the current node

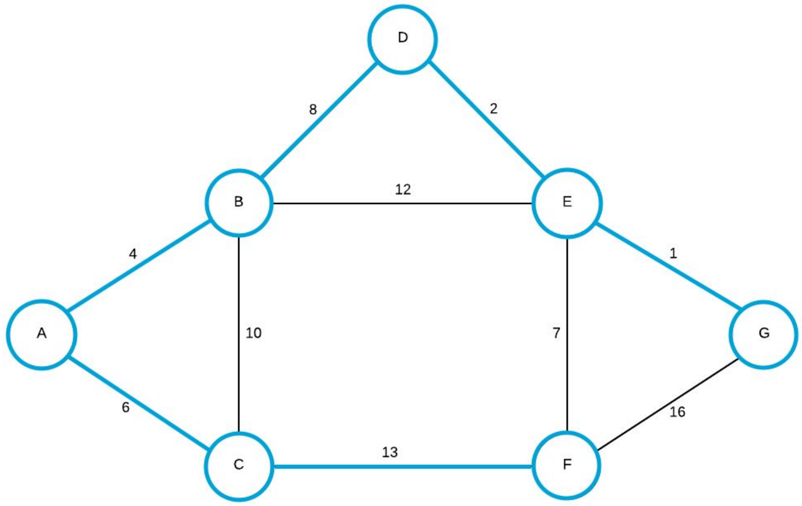

Figure 8: The algorithm is complete and all of the nodes in the graph have been covered

Conclusion

By using the logic in the process Dijkstra's can be used outside of graph theory and can be used to determine the shortest routes without an end goal. For the given example the shortest path that covers every node in the graph is determined by Dijkstra's algorithm is displayed in figure 8. With the application of Dijkstra's algorithm the means of finding the shortest path is possible.

Glossary

Integer Value – An assigned whole number.

Truth Value – A value that is either true or false.

Nodes – Also known as vertices; the fundamental unit that makes up a mathematical graph.

Edges – Connects two nodes (vertices) together through a relation.

Weights of Edges – The given values assigned to specific edges in a graph relation.

Outgoing Edges – Any edges that point *from* a particular node.

Incoming Edges – Any edges that point *to* a particular node.

Degree (of a Node) – The sum of all incoming and outgoing edges for a particular node.

Connected Graph – A graph in which every node can be reached by traversing at least one edge.

Undirected Graph – A graph in which all edges are bidirectional.

Weighted Graph – A graph in which each edge is given a weight represented by a numerical value.