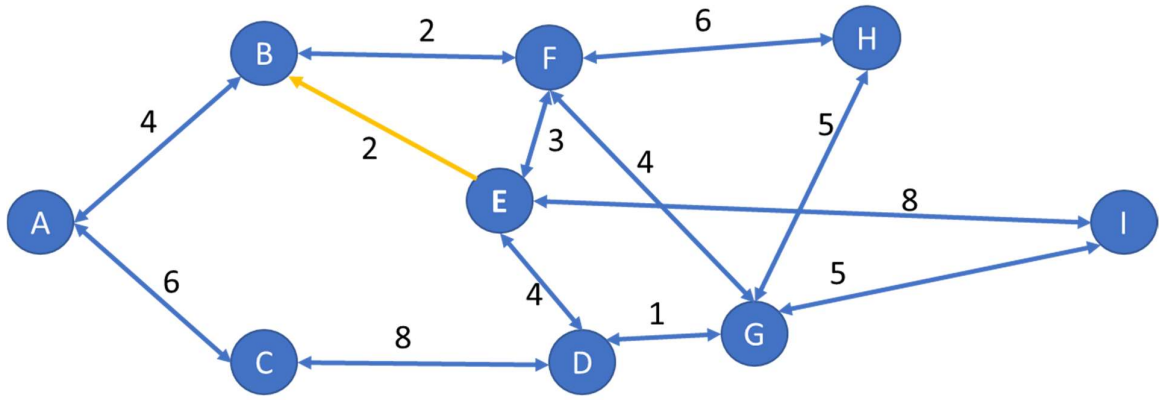
TransVirtual

Shortest Route Optimizer

Design an application to calculate the shortest path from one location to another location within connected nodes.



Hints – Use Dijkstra's Algorithm, quick video about the algorithm:

https://www.youtube.com/watch?v=ba4YGd7S-TY

Requirements:

1. Develop an ASP.NET MVC (.NET Framework 4.8) application.
2. Take input from the user for the selected FROM and TO nodes.
3. Display the calculated list of traversed nodes between FROM and TO.
4. Display the aggregate distance travelled.
5. You will need to be able to demonstrate the application running on your system.
6. Develop an equivalent console app using C# (.NET Framework 4.8) reusing the logic from the MVC app. This will need to runnable on our systems.
7. The source node graph should be contained within a data model.
8. Not all nodes are bidirectional. E.g. from B we can't directly go to E; but from E we can directly go to B.
9. A project/assembly should expose a method to calculate the shortest route, which accepts three parameters:
    a. From Node
    b. To Node
    c. Node Source for the list of nodes to use

E.g. public ShortestPathData ShortestPath(string fromNodeName, string toNodeName, List<Nodes> graphNode)
{ your code here }

Continued on next page…

10. The exposed method should return a DTO containing the results.

E.g. Class ShortestPathData
{
List<string> NodeNames { get; set; }
Int Distance { get; }
}

11. The results should include a List of node names in the order they are traversed between FROM and TO.
12. The list of traversed nodes should be displayed in comma separated format.
13. Display the results to the user.

E.g.
> fromNodeName = "A", toNodeName = "D": A, B,C, D
> Total Distance: 10

Bonus Points:

Create unit test(s) for the solution.