**AMRITA VISHWA VIDYAPEETHAM, ETTIMADAI**



**19AIE111**

**DATA STRUCTURES AND ALGORITHM - 1**

**PROJECT REPORT of B. TECH 2nd Sem CSE-AI**

**TOPIC: SHORTEST PATH**

**GROUP – 15**

**TEAM MEMBERS:**

1. SRIKAKOLAPU VENKATA SAI DHANUSH [CB.EN.U4AIE20068]
2. SWARNAMUHI K [CB.EN.U4AIE20069]
3. A SWETHA TIRUMALAI [CB.EN.U4AIE20070]
4. S TARUN [CB.EN.U4AIE20071]

SHORTEST PATH ALGORITHM

* The shortest path algorithm is given a weighted graph or diagram G = (V,E,W) and two specified vertices v and w; the algorithm finds a shortest path from v to w.
* The distance from a vertex v to a vertex w (denoted d(v, w)) is the weight of a shortest path from v to w.
* Shortest path algorithms have various uses, most notable being Route planning software such as Google Maps, etc.

WHY DIFFERENT ALGORITHMS?

* Most people are aware of the shortest path problem, but their familiarity with it begins and ends with considering the shortest path between two points, A and B.
* However, for computer scientists this problem takes a different turn, as different algorithms may be needed to solve the different problems.

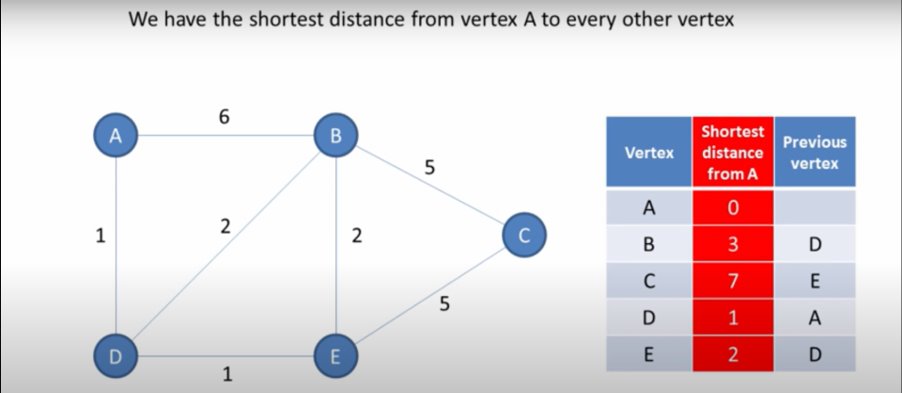
SHORTEST PATH ALGORITHMs

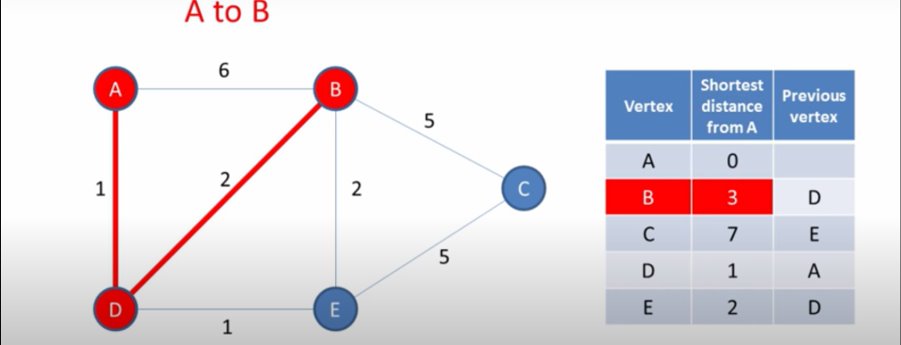
* Dijkstra’s Algorithm
* Bellman-Ford Algorithm
* Floyd-Warshall Algorithm
* Johnson’s Algorithm

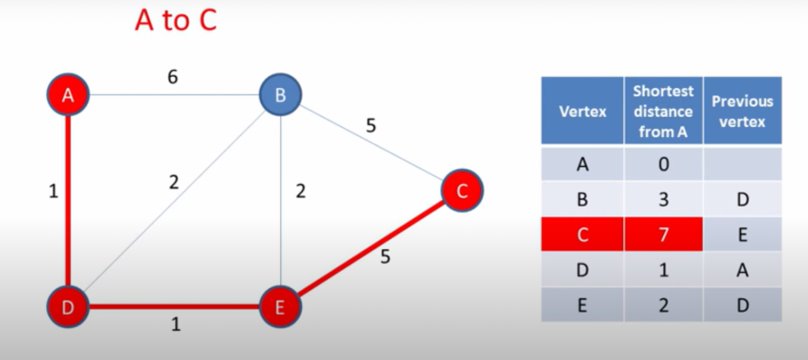
Shortest path algorithms operate on a graph, which is made up of vertices and edges that connect them. A graph may be directed, undirected, weighted, and more. It’s these distinctions that determine which algorithm will work better than another for certain graph types.

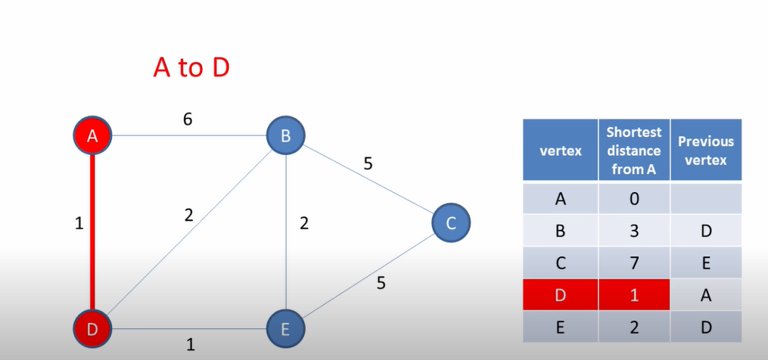
In our case, we will use Dijkstra's shortest path algorithm to find shortest paths.

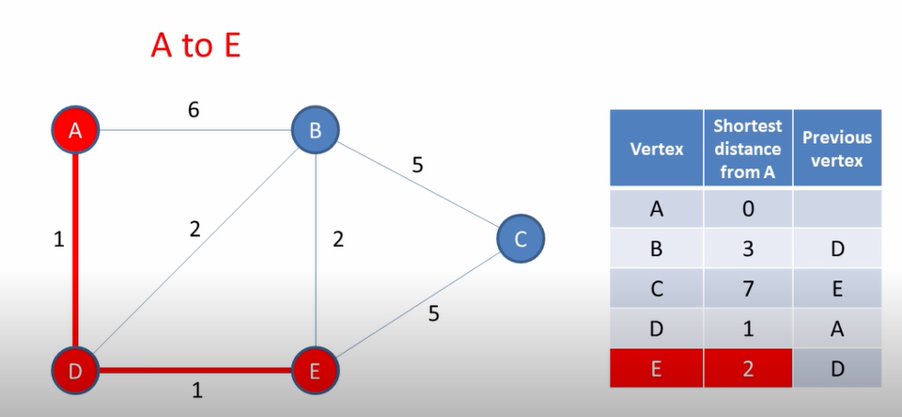
FINDING SHORTEST DISTANCE FROM VERTEX A (BEGINNING)



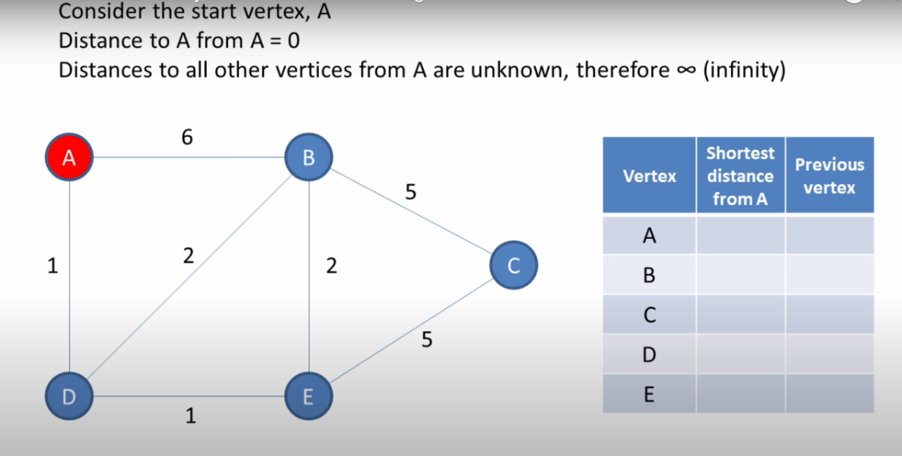


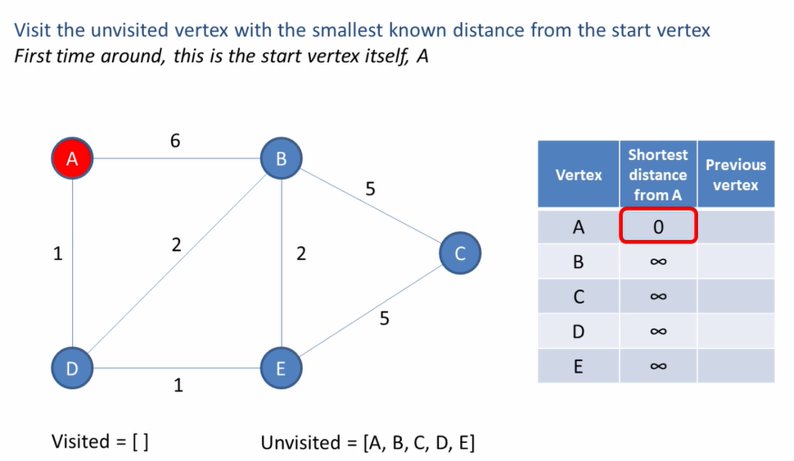


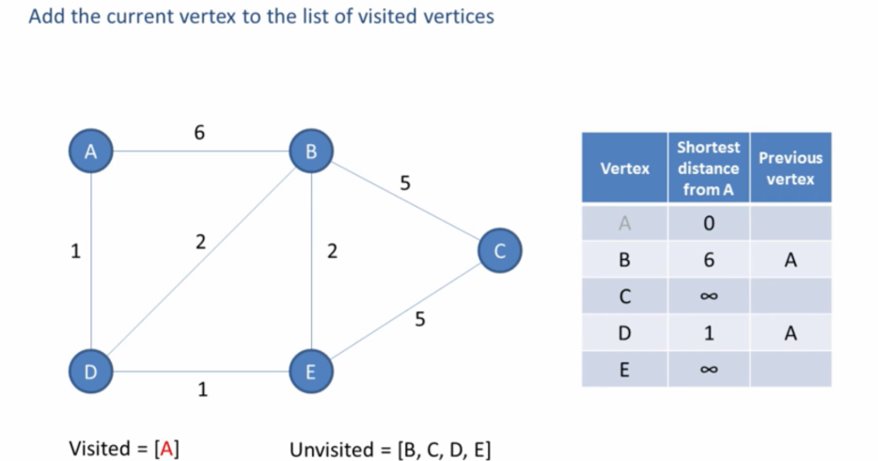


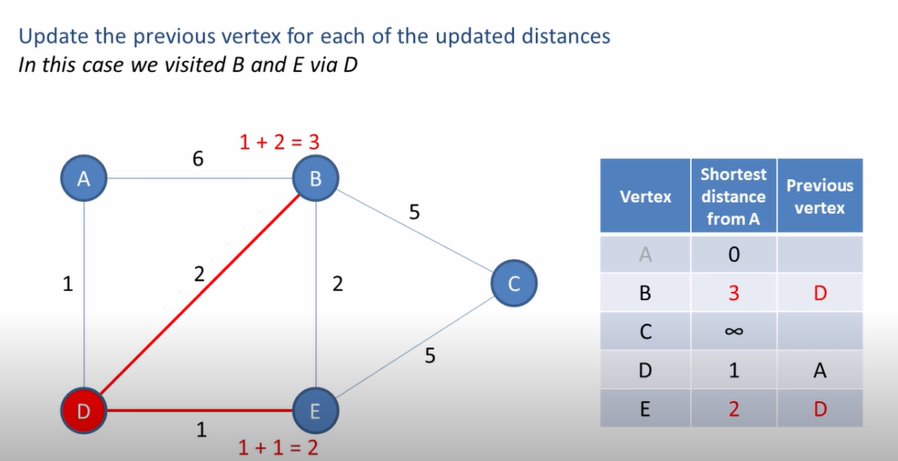


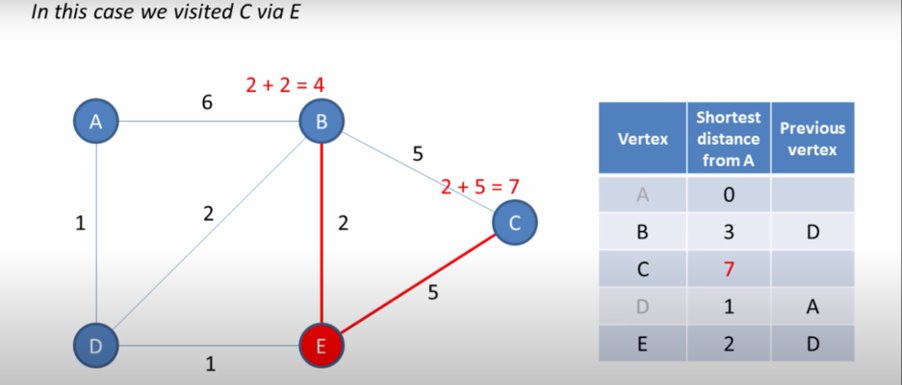
EXPLAINING THE ALGORITHM WITH AN EXAMPLE

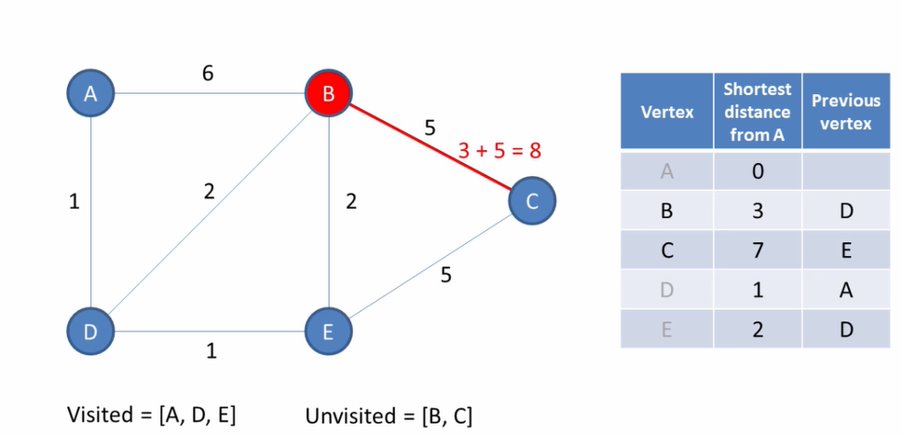


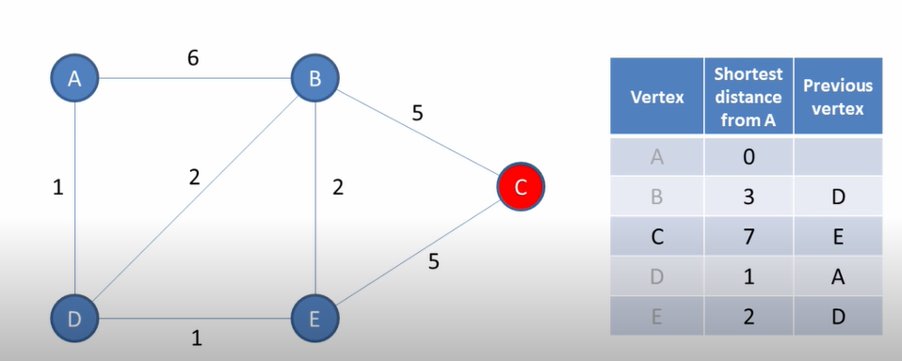












DJIKTRA’S ALGORITHM

* INITIALIZATION:
  + Let distance of start vertex from start vertex = 0
  + Let distance of all other vertices from start = ∞ (infinity)
* REPEATING STEPS:
  + Visit the unvisited vertex with the smallest known distance from the start vertex
  + For the current vertex, examine its unvisited Neighbours
  + For the current vertex, calculate distance of each neighbor from start vertex
  + If the calculated distance of a vertex is less than the known distance, update the shortest distance
  + Update the previous vertex for each of the updated distances
  + Add the current vertex to the list of visited vertices
* Repeat these steps until all vertices visited.

APPLICATIONS

Dijkstra’s Algorithm has several real-world use cases, some of which are as follows:

**Digital Mapping Services in Google Maps:** Many times, we have tried to find the distance in G-Maps, from one city to another, or from your location to the nearest desired location. There encounters the [Shortest Path Algorithm](https://www.geeksforgeeks.org/printing-paths-dijkstras-shortest-path-algorithm/), as there are various routes/paths connecting them but it has to show the minimum distance, so Dijkstra’s Algorithm is used to find the minimum distance between two locations along the path. Consider India as a graph and represent a city/place with a vertex and the route between two cities/places as an edge, then by using this algorithm, the shortest routes between any two cities/places or from one city/place to another city/place can be calculated.

**Social Networking Applications:** In many applications you might have seen the app suggests the list of friends that a particular user may know. How do you think many social media companies implement this feature efficiently, especially when the system has over a billion users. The standard Dijkstra algorithm can be applied using the shortest path between users measured through handshakes or connections among them. When the social networking graph is exceedingly small, it uses standard Dijkstra’s algorithm along with some other features to find the shortest paths, and however, when the graph is becoming bigger and bigger, the standard algorithm takes a few several seconds to count and alternate advanced algorithms are used.

**Telephone Network:** As we know, in a telephone network, each line has a bandwidth, ‘b.’ The bandwidth of the transmission line is the highest frequency that that line can support. If the frequency of the signal is higher in a certain line, the signal is reduced by that line. Bandwidth represents the amount of information that can be transmitted by the line. If we imagine a city to be a graph, the vertices represent the switching stations, and the edges represent the transmission lines and the weight of the edges represents ‘b.’ So as you can see it can fall into the category of shortest distance problem, for which the Dijkstra is can be used.

**IP routing to find Open shortest Path First:** [Open Shortest Path First (OSPF)](https://www.geeksforgeeks.org/open-shortest-path-first-ospf-protocol-states/) is a link-state [routing protocol](https://www.geeksforgeeks.org/classes-of-routing-protocols/) that is used to find the best path between the source and the destination router using its own Shortest Path First. Dijkstra’s algorithm is widely used in the routing protocols required by the routers to update their forwarding table. The algorithm provides the shortest cost path from the source router to other routers in the network.

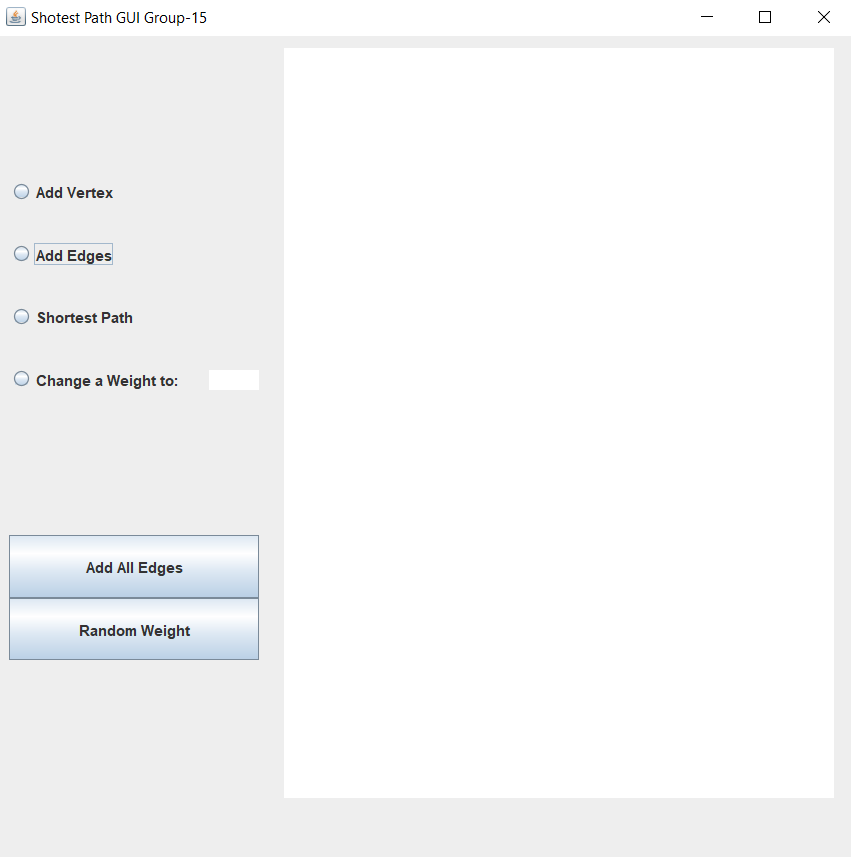
**Flighting Agenda:** For example, if a person needs software for making an agenda of flights for customers. The agent has access to a database with all airports and flights. Besides the flight number, origin airport, and destination, the flights have departure and arrival time. Specifically, the agent wants to determine the earliest arrival time for the destination given an origin airport and start time. There this algorithm comes into use.

**Designate file server:** To designate a file server in a [LAN(local area network)](https://www.geeksforgeeks.org/local-area-network-lan-technologies/), Dijkstra’s algorithm can be used. Consider that an infinite amount of time is required for transmitting files from one computer to another computer. Therefore, to minimize the number of “hops” from the file server to every other computer on the network the idea is to use Dijkstra’s algorithm to minimize the shortest path between the networks resulting in the minimum number of hops.

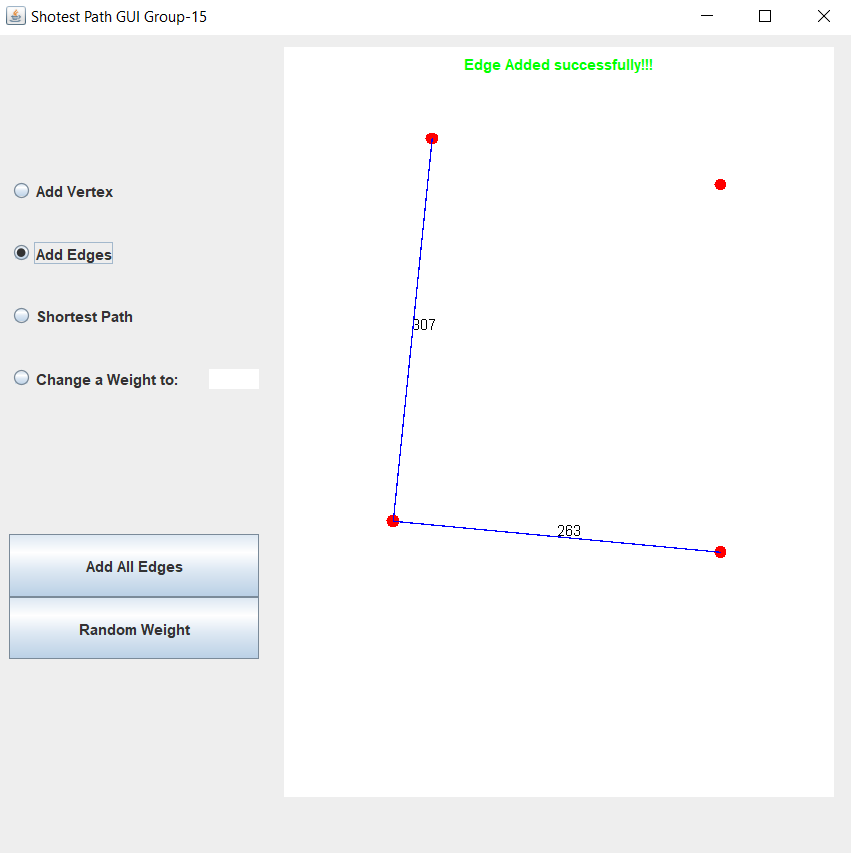
**Robotic Path:** Nowadays, drones and robots have come into existence, some of which are manual, some automated. The drones/robots which are automated and are used to deliver the packages to a specific location or used for a task are loaded with this algorithm module so that when the source and destination is known, the robot/drone moves in the ordered direction by following the shortest path to keep delivering the package in a minimum amount of time.

**RESULTS:**

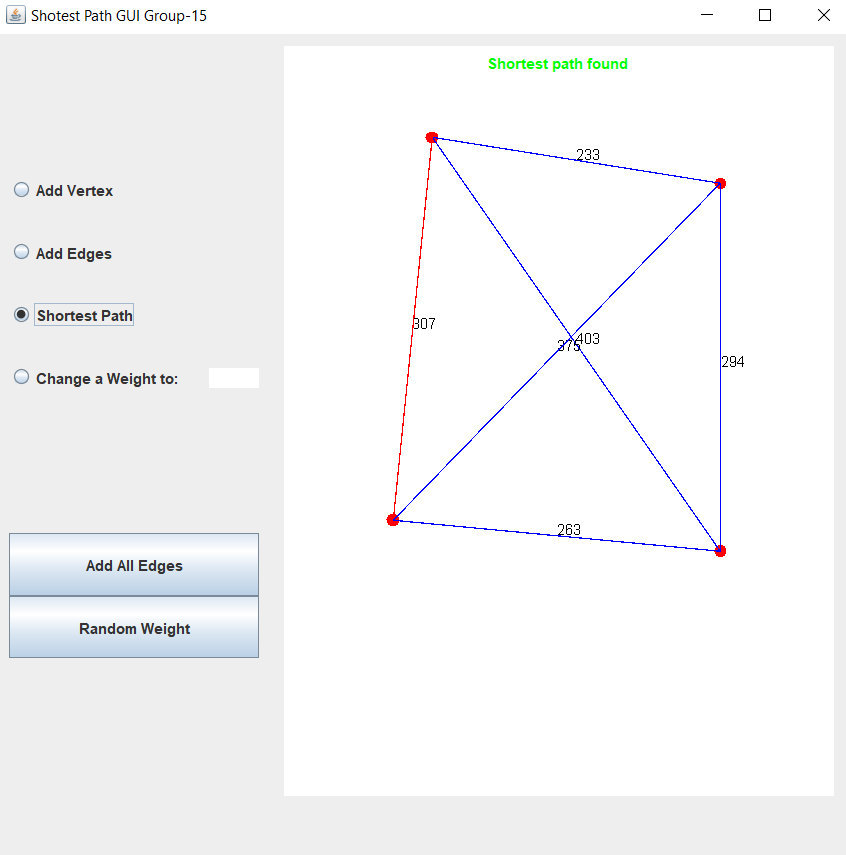
**GUI:**



**Adding Nodes and Edges:**



**Finding Shortest Path:**



A PRACTICAL APPLICATION

