

## User Guide: Dijkstra's Algorithm Simulation

### Introduction to Dijkstra's Algorithm

Dijkstra's algorithm is a popular algorithm for finding the shortest paths between nodes in a graph. It works by:

1. Initializing distances to all nodes as infinity except the source (which is set to 0)
2. Selecting the unvisited node with the smallest distance
3. Updating distances to neighbours if a shorter path is found
4. Marking the current node as visited
5. Repeating until all nodes are visited

This simulation visually demonstrates each step of the algorithm, making it easier to understand how it works.

### Getting Started

When you open the application, you'll see a clean interface with a header containing the title "Dijkstra's Algorithm Simulation" and several buttons:

- Add Edge
- Run Algorithm
- Reset
- Help

The main area consists of a graph canvas on the left and a results panel on the right.

When you first load the application, a "Create Graph" modal will appear, asking you to specify the number of nodes you want in your graph.

### Creating a Graph

1. In the "Create Graph" modal, enter the number of nodes you want (minimum 2).
2. Click the "Create" button.
3. The nodes will be created and arranged in a circular layout on the canvas.

Each node is represented by a blue circle with its ID number displayed inside.

### Adding Edges

After creating your graph, you'll need to add edges to connect the nodes:

1. Click the "Add Edge" button in the header.
2. In the "Add Edge" modal:
  - Select a "From Node" from the dropdown list.
  - Select a "To Node" from the dropdown list.
  - Enter a "Weight" value (must be greater than 0).
3. Click the "Add" button to create the edge.

The edge will appear on the graph as a line connecting the two nodes, with a small circle displaying the weight in the middle of the line.

### Important Notes:

- You cannot create an edge from a node to itself.
- You cannot create multiple edges between the same two nodes.
- Edge weights must be positive numbers.

## Running the Algorithm

Once you've created a graph with edges:

1. Click the "Run Algorithm" button in the header.
2. In the "Select Source Node" modal, choose the node you want to use as the starting point.
3. Click the "Start" button to run Dijkstra's algorithm.

## Understanding the Results

After the algorithm completes, the "Shortest Path Results" panel on the right will display:

- For each node (except the source), the shortest distance from the source
- The path taken to reach each node (as a sequence of node IDs)
- For nodes that cannot be reached from the source, " $\infty$  (No path)" will be displayed

The graph will show the shortest paths highlighted, making it easy to visually trace the path from the source to any node.

## Color Key:

- Blue: Unvisited node
- Orange: Source node
- Green: Visited node
- Orange lines: Shortest path edges

## Additional Features

### Reset

If you want to start over with a new graph:

1. Click the "Reset" button in the header.
2. The current graph will be cleared, and the "Create Graph" modal will appear again.

### Help

If you need information about how to use the application:

1. Click the "Help" button in the header.
2. A modal will appear with instructions on how to use the application and a brief explanation of Dijkstra's algorithm.

### Troubleshooting

If you encounter issues with the application, here are some common solutions:

Issue	Solution
Cannot add an edge	Make sure you're selecting different nodes for the "From" and "To" fields, and that an edge doesn't already exist between these nodes.
Cannot run the algorithm	Ensure you've created a graph with at least one edge before trying to run the algorithm.
No paths found	If the results show " $\infty$ (No path)" for all nodes, it means there are no valid paths from your source node to any other node. Check that your graph is connected (there is a path from the source to other nodes).
Canvas doesn't resize properly	If the canvas appears distorted after resizing your browser window, try refreshing the page.

### **Example Usage Scenario**

#### **Finding the Shortest Path in a Road Network**

Imagine a road network where:

- Nodes represent cities
  - Edges represent roads between cities
  - Weights represent distances in kilometers
1. Create a graph with 5 nodes (cities)
  2. Add edges (roads) between cities with appropriate weights
  3. Run Dijkstra's algorithm selecting your starting city
  4. The results will show the shortest routes to all other cities from your starting point

This can help you understand how GPS navigation systems determine the shortest route between locations.

***Submitted by:***

***Shreeya Panchal – 16010124813***

***Pradnya Jadhav – 16010124808***

***Neil Bhanushali - 16010124804***