Homework #5: Shortest Path (100 pts)

For this assignment, you must add another route to your web application. Specifically, your job is to implement a web application that calculates the shortest path between nodes given some input graph.

Requirements: Your app must:

- Use /hw5 route for GET and POST requests
- Use a textarea to input a weighted graph as a comma-delimited adjacency matrix
  - N nodes => N+1 x N+1 matrix
  - The first row will be a header row with N+1 comma-separated entries
    - N = 3, then header row => (-, A, B, C)
  - The first (left-most) column will be the same header
  - A number in row i, column j represents a weighted edge between nodes i and j
  - A dash (-) in row i, column j represents no edge between nodes i and j
- Have an input and Generate button that creates a random graph in the textarea
  - The input slider will determine the number of nodes, between 3 and 10
  - The nodes should be named A, B, C, ...
- Have two input fields to identify the “From” and “To” nodes
- Have a “ShortestPath” button that computes the shortest path
- Outputs the shortest path in the given graph from the From node to the To node
- Outputs the total weight of the shortest path

Example:
- Example from lecture:
  - Shortest path from A to G?

```plaintext
-A,B,C,D,E,F,G
A,-,2,4,7,-,-,5,-
B,2,-,-,6,3,-,8
C,4,-,-,-,-,6,-
D,7,6,-,-,-,1,6
E,-,3,-,-,-,-,7
F,5,-,6,1,-,-,6
G,-,8,-,-,6,7,6,-
```
Input graph..
After inputting “A” and “G” as From and To nodes, then clicked ShortestPath button:

**shortest path:**

A, B, G,

**distance:**

10