Logan Maps: A Small-Scale Google Maps Clone Using Node Networks

For my final project, I decided to recreate Google Maps with Python. The project comes with a rudimentary user interface and uses the concept of networks learned in unit four of this course. The user can pick two points on the map provided via two drop-down menus. My program will calculate the shortest distance between the two nodes in the network, calculate the distance between the nodes, and provide an estimated travel time based on an average walking speed. While being quite a simple program, I think that it is very important that we have programs like Google Maps and other navigational tools.

The topic of networks is crucial in this project, as it is in any navigational program. Creating a network of nodes as landmarks for the program allowed me to weigh these edges using data from Google Maps. Using the Dijkstra algorithm to calculate the shortest path between these nodes is crucial in the project, as this algorithm does most of the heavy lifting. The reason I think that this approach is the best considering the real-world problem was scalability. In theory, Finding the shortest path between two nodes in a simple network by trial and error is easy. For example, if you had five nodes in a mesh network with varying weights, you could trial and error your way to the correct solution in under an hour. Let’s take that example and add 500 nodes, all with varying connections and weights. Trial and error on this new network would not be feasible. That’s where the systematic approach of the Dijkstra algorithm comes into play. While my program only has a few landmarks on campus, even expanding the map to the entire campus could prove to be a challenge without using Dijkstra’s algorithm and impossible without the use of networks.

After some reflection, implementation of the topic within this environment was quite easy. Most of the time the work revolved around the planning of the project and gathering of data. As far as coding, coding the topic itself was not a time-intensive part of the project. It was user interface development and trying to find python libraries to help aid in the efficiency of my program. Most of the limitations of demonstrating my topic were scaling up more nodes in the network. Some nodes would require multiple edges to be added to the network, which would further complicate constructing the network. One previous experience that helped tremendously was my Introduction to Computing II class where Dijkstra’s algorithm was taught. My biggest takeaway from this project was that without our idea of graph theory, navigation systems like Google Maps would cease to exist.