/***

* Project Report Template

* Project 3 (Map Routing), ECE368

***/

Name: Matthew Bolda

Login: MBolda

/***

* Explain your overall approach to the problem and a short

* general summary of your solution and code.

***/

I started at the first file, made a helper function to read the numbers one by one. This helped me get the vertices into a list and create an array for them. Then I would create an array of the connected vertices. From there I moved to the second file, the first would be the source and the second would be the destination. I would put the source into the heap and make the distance 0, then I would put all its neighbors in the heap and go until the heap is empty, updating the distance. Once the distances are done I would go to the destination vertex and print the previous vertex until I reached the source. If the distance is still infinite, then the vertices are not connected and you should print “INF”. Continue this process for every query.

NOTE: I used the math and limits libraries for the pow function and MAX_INT. I used the pow function to find the distance between two vertices. Used MAX_INT to be “infinite”.

/***

* Known bugs / limitations of your program / assumptions made.

***/

I do not completely free all allocated data, I cannot find where I lose this memory

/***

* List whatever help (if any) that you received.

***/

A youtube video I used for learning more about Dijkstra’s Algorithm helped me create my strategy for this project. The video had no code or anything other than visualizations to help guide me.

LINK: https://www.youtube.com/watch?v=pVfj6mxhdMw

/***

* Describe any serious problems you encountered.

***/

I have had a very difficult time trying to change the queue to a priority queue.

/**

 * List any other comments/feedback here (e.g., whether you

 * enjoyed doing the exercise, it was too easy/tough, etc.).

 **/

The compile line I used for this project was:

gcc -Werror -Wall adjacent.c -o adjacent -lm

./adjacent (insert map.txt) (insert query.txt) will start the program.

I enjoyed the project, originally I thought it was too easy but it was very difficult to make faster and guarantee the shortest path and not just a short path.

NOTE (again): I used the math and limits libraries for the pow function and MAX_INT. I used the pow function to find the distance between two vertices. Used MAX_INT to be “infinite”.