

```

/*****
* Project Report Template
* Project 3 (Map Routing), ECE368
*****/
Name: Karthik Bokka
Login: kbokka
/*****
* Explain your overall approach to the problem and a short
* general summary of your solution and code.
*****/
    • Created an undirected graph using an adjacency list.
    • Each vertex had a field for its: x and y coordinates, index number,
      vertex distance, visited flag, and a pointer to the head of the linked
      list of edges.
    • The fields for the edges are: distance, search_distance, a link to the
      previous node, pointer to the node on the edge, and a pointer to the
      next node on a separate edge.
    • The distance on the edge was calculated using the Euclidean formula
      given in the project description.
    • I created a min heap to get the node with the lowest distance on its
      vertex distance field.
    • I implemented Dijkstra's algorithm where I would push the source node
      and store its edges and retrieve the node with the lowest edge. Edge
      relaxation was applied to nodes as needed. This process continued until
      the flag on the destination node was 1 and the min heap was not empty.
/*****
* Known bugs / limitations of your program / assumptions made.
*****/
    • There are some rounding errors for some of the final distances when the
      USA map is loaded.
    • The path is not printed to the stdout because some of the paths that
      are written to stdout take up too much space to properly look at.
    • Only the final search distance is printed to stdout.
    • Utilized math.h library.
    • Compile Flags = "lm", "o3"
/*****
* List whatever help (if any) that you received.
*****/
    • Used Lecture notes on Graphs and Heaps
    • Received help from Professor Lin during his office hours for debugging
      parts of the code.
/*****
* Describe any serious problems you encountered.
*****/
    • I ran into invalid memory write problems with the fscanf
/*****
* List any other comments/feedback here (e.g., whether you
* enjoyed doing the exercise, it was too easy/tough, etc.).
*****/
    • Files Used: adjacent.c, adjacent.h

```