Overview:

This program is about navigating by people from source to destination. Program is implemented with graphs. A summary of the formatting requirements is in the CSCI 3901 course assignment #3 information in the course's bright space.

The Java program prints the shortest distance by sequence of coordinates from source to destination.

Program flow:-

- Will Read the vertices with the help of newIntersection() method.
- Will create edges to define road with the help of defineRoad().
- Will create a graph with the edges and storing the graph with AdjacencyList.
- Will print the shortest path with the sequence of coordinates from source to destination

Design of the code:

- Adjacency lists are used for implementing the graphs to store edges and vertices.
- HashMap is created to store the co-ordinates.
- Adjacencylistnode is created to define edge with source,destination and weight.
- Graph is created with adjacency lists.

Java files:

- AdjacencyListNode.java – class file is created to define edge with source,destination and weight.
- Graph.java –class file is created for creating graph with help of adjacency list and finding the shortest path
- HalifaxMap.java-class file is created for reading vertices, edges and printing the sequence of coordinates.

Modules :

- Implementation of a data structure to print shortest path from source to destination.
- Implementation of a data structure to store vertices or Intersections.
- Implementation of a data structure to store edges with source,destination and weight.
- Implementation of a graph with adjacencylist.
- Create a new level if random value is 1 and upperlist is null.
- Printing shortest path from source to destination.

Assumptions:

- Coordinates are integers.

- Implementing with own data structure without using library package related to graphs.