

Assignment #9 – Traverse the graph using DFS, BFS, Dijkstra's Algorithm

Graph is an important data structure. There are many questions can be solved with graph related algorithms. These algorithms also utilize other data structures that we have studies previously, such as array list, doubly linked list, map, queue, priority queue, iterator, set etc. In this assignment, you will use many data structures to traverse a city map.

Objectives:

The main objectives of this assignment are:

- Use graph to represent data
- Implement DFS/BFS algorithm
- Implement Shortest Distance algorithm

The Assignment:

Given some direct routes between cities, find the list of cities starting from one city using the three methods **DFS()**, **BFS()**, **Dijkstra ()** in Travel.java

1. Download and import the code starter zip file.
2. This assignment depends on your implementation of
 1. Doubly Linked List
 2. Queue (Alternative: doubly linked list)
 3. Adaptable Priority Queue
 4. Sorting algorithm (Cannot use directly, has to be modified)
 5. Map (HashMap is the best choice)
 6. Graph

If you didn't do well with the related previous assignments, **please meet with the TA/instructor to get approval and instructions to import library cs2321util at least 3 days before the project due.**

3. Please note that you have been given some test cases in TravelTest.java. Feel free to create more test cases if you will.

Submission:

Look at the following checklist:

1. Do you ever `import` from `java.util`. If so, be sure you only import allowed components (like `Iterator`, `Exceptions`, etc.). Unless the assignment specifically

mentions it is permissible, you should never include any of java's native data structures.

2. Does the program meet all required interfaces? Did you test them?
3. Is the indentation easily readable? You can have Eclipse correct indentation by highlighting all code and select "Source → Correct Indentation".

If you have reviewed your code and all the criteria on the checklist are acceptable, follow the submission procedure, export to **prog9.zip** and upload to canvas.

Grading Criteria:

1. DFS (): 25 points
2. BFS (): 25 points
3. Dijkstra (): 35 points
4. Comments and concise code: 15 points total. 5 points for each algorithm.