

Planning report

Project 4 – Route finding

Liang Minghao

2018110420

Summary:

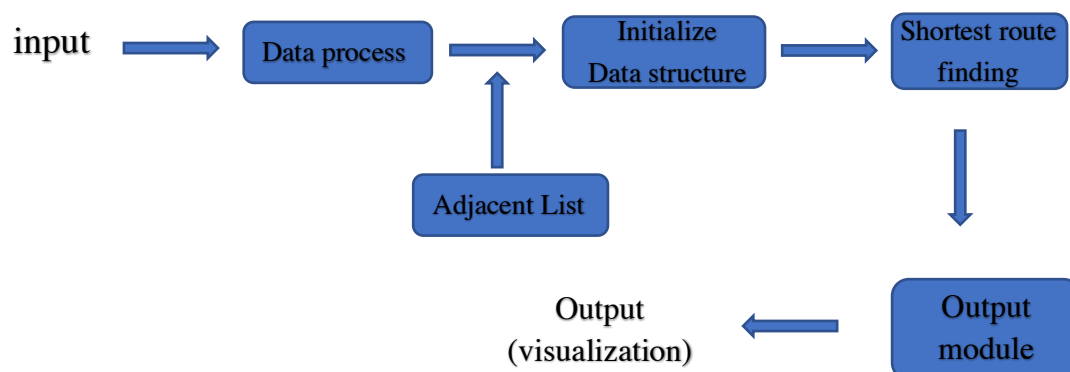
The project I choose is the project 3 – route finding project. This project aim to make a programme which will analyze and process large data that represents all the footpaths on the Leeds University Campus, then this programme will find the shortest path between two points on a map and visualize both map and the shortest path.

Design plan:

In the large-scale:

This project is aimed to develop a programme that can find the shortest path between two nodes in the map and visualize the map and the route. The input is a large data file, which contains thousands of lines of elements in the map including the link information, node information and way information, etc. However, the link information and the node information are linear and independent from each other, the usable information for the programme is the coordinates of each nodes and the relation of nodes, so the processing and concatenation of the elements in the information is necessary. The data transition between modules is based on the delivery of the adjacent list, all input

data all transferred into the inner data in the adjacent list. The output is some files which contains sets of combination of coordinates, the coordinates represent nodes right on the shortest route and the coordinates of nodes which are isolated with any other nodes. The command file which set the display format to be loaded by the gnuplot programme is also required. In the end, the visualized map and highlighted shortest route on the map should be displayed.



In the medium-scale:

Four modules supposed applied in this programme, they are data process module, create graph module, shortest path module and output module. For the data process module, it will process the raw data into proper format of data for initialize the data structure. For create graph module, it initializes the adjacent list by using the processed data. For the shortest path module, it will calculate all the path by using the data in adjacent list and implement the shortest route finding algorithm. The output module will output all nodes on specific path between two nodes

Iterations of design: the simplest application I will develop is the rough combination of functional modules, which can achieve the finding of the shortest path between two default nodes and output the nodes with relative information in the terminal. Next step is the optimize of the algorithm in modules and the integrity of the data format that transmit in modules. Then the user's interface should be considered and designed for the proper users and the

convenience of change, extend and test should also be taken into account. The most important requirement for iteration is that each iteration should be based on each module and repeat the cycle of design, develop and test, this method can speed up the development and can strengthen the connection between modules which can avoid the clash between modules.

Test plan

The test of the application is based on three aspects. The first aspect is the basic test, the evaluation on the compile, which requires no error or exception exists in compile process. The second aspects concern about the correctness of the output of the module, which requires the module output the correct content instead of messy code or a loop. The third aspects mainly focused on the integrity of module, the fully achievement of design, proper format of output, the way the data delivered between modules and the scalability should all be considered and evaluated. In detail, for the data process module, the correctness extraction of the original map data and the proper output of data especially the accessibility should be tested by visualizing with gnuplot programme and compare the display with the given map picture. For the create graph module, the randomly output the node information of the adjacent list can evaluate the integrity of the creation of adjacent list. For the shortest path finding module, it can be evaluated by verifying every node on the shortest route. For the output module, it can be evaluated by compare the map and shortest route generated by the programme with the evaluation picture.

Schedule:

Week 10:

Start to design the project in high level and consider the medium scale part of the project.

Week 11:

Finish the planning report and start to consider the algorithm and data structure used in the programme. Analyze the original map data file, consider the proper approach to transfer it into appropriate format.

Week 12:

Start to develop the whole project. The iteration in Data process module and create graph module can be finished in one day. Then the iteration in shortest path finding module takes one day to implement. The output module and the user's interface can be finished in one day. The rest of time will be used to finish the iteration over the whole project.

Week 13:

Focus on the optimize of the project by using the iteration. Start to write the report.

Week 14:

Review the whole project and modify the report. Prepare for submit the project.