

GROUP HOMEWORK 2: GRAPH PROGRAMMING

1 Programming

You know Grab or GoJek (GoViet)? What if you make a tiny application (app) to help people travel around the world?

In this homework, you guys are required to make a simple app for path-finding. Your app should have a basic feature: Find the shortest path from A to B. That's all!

We assume the shortest path from location A to location B is an ordered list of streets with minimum total consumption time.

1.1 What should your app have?

Your app has a map, which contains:

- A set of named locations.
- A set of named streets (which connect two locations). Each street needs a consumption time to go through it. Consumption time is a positive integer.

In fact, some streets only allow certain vehicles to move:

- Some streets accept Motorbikes.
- Some streets accept Cars.
- Some streets accept both.

So your app must be designed to choose a transport type of Motorbike or Car to travel. Due to different transport types, you should take care of a valid path.

Your app must read input (database) from files:

- **street.txt**: The file contains information of an **Adjacency matrix** (Table 1), which stores the consumption time to move between locations (if the paths exist).
- **transport_type.txt**: The file has a format like an adjacency matrix, which stores the specified transport type to be transported (Table 1). Th. In which '0' represents no path, '1' represents MotorBike, '2' represents Car and '3' represents both.

<i>street.txt</i>	<i>transport_type.txt</i>
5	5
0 3 1 2 5	0 3 3 3 3
0 0 4 0 1	0 0 1 0 1
1 4 0 2 1	3 1 0 2 2
2 0 2 0 0	3 0 2 0 0
5 1 1 0 0	3 1 2 0 0

Table 1: Example files *street.txt* and *transport_type.txt*

- **query.txt**: The file contains 3 positive integers which are the transport type, the starting location, and the end location respectively. For example: 2 1 5.

Your app should output to file. This file contains:

- An integer denotes minimum total consumption time.
- An ordered list of names of streets.
- In the example above, we have vehicle type '2' \rightarrow transport type is Car, the starting location is 1, and the end location is 5. The minimum total consumption time is 2. Names of streets in turn passing are: (1, 3) \rightarrow (3, 5).

1.2 How should users use your app?

The following steps describe how users use your app:

1. Input the name of start location A.
2. Input the name of destination location B.
3. Choose a transport type: Motorbike or Car.
4. Your app displays the shortest path from A to B (in file).

1.3 Could your app have extra features?

Yes! You will get bonus score if your app has other extra features. One of the features you can install is coordinating the driver to receive the ride, similar to the Grab app. We have M drivers at M locations and N passengers at N locations. Your program should recommend coordinating drivers to pick up passengers so that total time consumption is optimal.

You will also get bonus score if your app has a graphical user interface (GUI).

2 Submission

- The submission file must be in the following format: **[SID1_SID2_SID3.zip]** (SID stands for Student ID), is the compression of the **[SID1_SID2_SID3]** folder. This folder contains:
 - The report file must be presented as a document **[SID1_SID2_SID3.pdf]**. The report must contains:
 - * Information (Names, Student IDs) must be declared clearly on the first page of your report.
 - * A table lists tasks of each member in group. Each member must have a percent of total work completed. Assume a group of 3 members, if student

A completed 40%, student B completed 20%, then student B completed 40% of the rest.

- * Describing the algorithm your group used.
- * Guiding how to use your app.
- * List of extra features (having Graphical User Interface, multiple destinations...).
- * Attaching some demo pictures.
- * All links and books related to your submission must be mentioned.
- * *The report file should be **structured, logical, clear and coherent**. The length of the submission should not exceed 20 pages for the document file.*
- The source code folder [`code_SID1_SID2_SID3`]. This folder contains all the source code (only `.cpp` and `.h` files). The code fragment must be clear, logical and commented.
- The release folder [`release_SID1_SID2_SID3`]. This folder contains:
 - * An executable file (in EXE format).
 - * Input data file(s). Note that input data must have at least 5 locations and 7 streets.
- Submission with wrong regulation will result in a "0" (zero).
- Plagiarism and Cheating will result in a "0" (zero) for the entire course and will be subject to appropriate referral to the Management Board of the CLC program for further action.

THE END