**COMP1001**

**Problem-Solving Methodology in Information Technology**

**Class Project Report**

**ZHENG Hongyi**

**13104036d**

1. **Problem**

This program aims to save the MTR travel cost for a family.

Assuming that there are n>1 family members. Each one of them will travel starting from a given station to another station. For each member, he or she have specific ticket type, namely Adult ticket, Student ticket, and Child ticket. Besides, each member could have different round trip per day. This program will compute the location (in terms of the station) where they should live to minimize the total travel cost.

1. **Program Design**
   1. **Data Preprocessing**

The data from the [www.mtr.com.hk](http://www.mtr.com.hk) is in .pdf format. First, exact data from .pdf file and arrange them in .csv file. Each row is corresponding to one start station and each column is corresponding to one ending station. As the is 108 station available, the list is a 108 \* 108 matrix.

* 1. **User Information Storing**

For each member, there are 3 information need to be stored, namely destination station, ticket type, and number of round trip per day. However, if more than one member has the same 3 choices, they can be recorded in one group to simplify the record. Hence for each entry, we record following 4 items, namely destination station, ticket type, number of members, and number of round trip per day. Each entry is corresponding to one tuple and all the tuples consists a list.

* 1. **Opination Scheme**

To find the best home station, the Brute Force algorithm is used.

Firstly, we set the minimum cost as -1. We traversal all the stations as start destination. For each start destination, we calculate the cost for the whole family per day. To further elaborate:

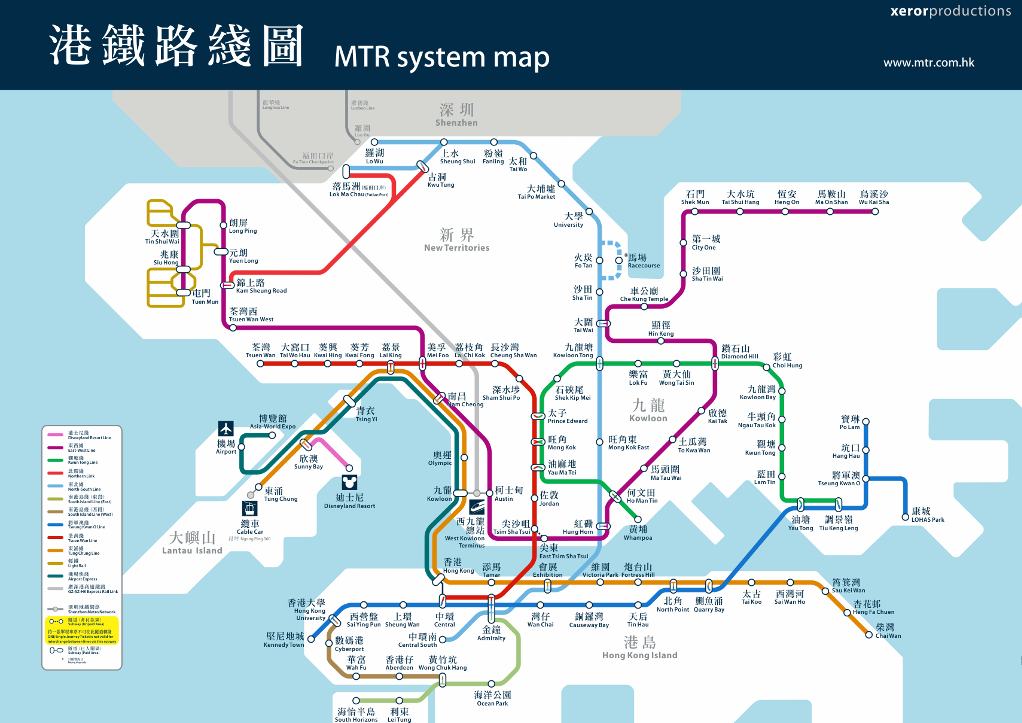
1. Traversal the record.
2. For each record, calculate the cost: cost between destination and start station \* number of member \* specific ticket price \* round trip per day \* 2

Then if current minimum cost is -1 or the current cost is less than current minimum cost, we set the current start station as best station and current cost and minimum cost.

* 1. **GUI Design**

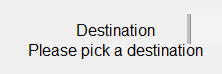
The graphic user interface is based on graphics.py.

The window is separated into two areas. The left area shows the mrt map. User can click on the destination to select the station.

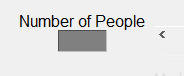


The right area is for user to input the member’s information.

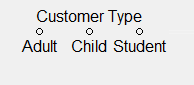
First one is the destination information The name of the chosen destination will be shown here.



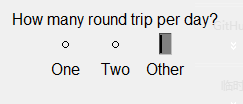
The second one is an entry box for user to input the number of members.



The third one is a radio button for user to choose the ticket type.



The fourth one is for user to input the number of round trip per day.

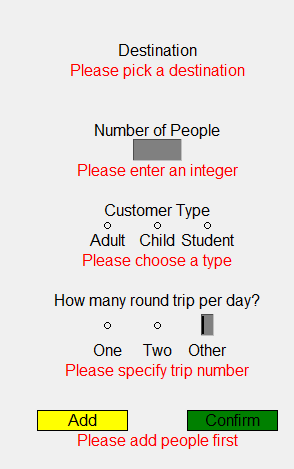


After that, two button are set for add record and calculate the minimum cost and best start station.



**2.5 Exception Handling Scheme**

Possible exceptions include: not selecting destination, not input number of people, input value is not integer, not select ticket type, and not input round trip per day. Each exception will be show in red warning in the window.



1. **Structure of Program**

The main body of the program is the class Interface. The main method would initialize a instance of Interface then run the interface.

* 1. **Initialization**

The initializer would prepare the necessary data container.



* 1. **readData()**

After initialization, readData() is called to read the data from .csv files.



* 1. **setUi()**

After reading data, setUi() method is call to initializa the GUI window.



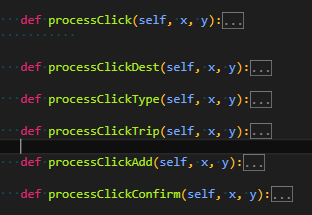
* 1. **run()**

After setting UI, run() is call to process each mouse click by calling processClick()



* 1. **processClick()**

InprocessClick(), it will judge the different click command according to the click position then call corresponding method to process the command. The methods include processClickDest(), processClickType(), processClickTrip(), processClickAdd(), and processClickConfirm.



* 1. **Helping functions**

Several tiny methods are used to help the processing. validClick() and validClickRange() are used to examine the click position

