

Report Lab 1 OOP

Marc Sirvent (240198) and Norbert Tomàs (242695)

1.

The objective of this program is to compute distances between cities which have some coordinates assigned. The cities will be stored in a matrix formed by a linked list.

To do this first we had to define 2 main classes:

1. The first class is named Point. This is the most basic class to structure our program. In this class we had to implement some methods to work with 2D geometric points such as getters, setters, distance between points and obviously the constructor which will assign values to 3 attributes: x and y (which are doubles) and name (which is a string). This class also has a method that calculates the distance between two points.
2. The second class is named DistanceMatrix. This class will be the one in charge of creating the matrix of the different cities with it's coordinates and compute the distances between them. We will also use this class to display the matrix with the code that has been given to us.
First we will create 2 attributes: the first one is a linked list of linked lists which store doubles. This will be the attribute in charge of storing all the values of the distances between cities. The second attribute will be a linked list of the class Point which we will store all the cities and it's coordinates.
Some of the methods we have to implement are: addCity, which will store in the linked list of points a new point with the values that are given by the user.
One of the other principal methods of this class is createDistanceMatrix. This method will compute the distances between points and store them in a matrix. To do this we have implemented an auxiliary LinkedList of doubles which will store the distances and, with 2 "for in" loops, we will pass it as a parameter to the matrix attribute.
Another important method that we will need in the createDistanceMatrix method is getDistance. Given 2 indexes of the array of points (cities) this method will call the distance between points.
Other methods that are more simple are one that returns the number of cities stored and another that returns the name of a city given a specific index.
3. We will also have to define some classes to test the previous ones. To display the matrix we have followed the instructions on the PDF.

2.

For doing this Lab we were doubting about the worthiness of using arrays in order to select the dynamic matrix because we didn't know the size of it. Finally we decided to do a linked list of linked lists because we thought it is easier to implement. In order to do the lab correctly we have taken into account all the concepts studied in class such as the design of the classes with their own attributes and methods and also trying to correctly follow the corresponding types of functions.

3.

We think that our way to do this lab with LinkedLists has worked correctly and was a pretty straightforward way to see how we would have to model our program to work as wished. Also in the test classes for checking the classes like Point or DistanceMatrix we have implemented correctly all the methods for it to work properly and we have shown that the classes they tested worked correctly. As a issue or setback we got while doing the program we were confused about how to implement correctly the function createDistanceMatrix because after making the computations of the distances between cities we had some problems with the way to store that values in a LinkedList which we finally solved with just 1 line of code but took us some time to realize it. For everything else we didn't have many issues, just that it's a new programming language and we need to adapt to it.