

CS205 – Lab Assignment 2

You will in this lab build on what was done in Lab 1 (computation of distances) and write a program that is easier to use. This program isn't very simple if written well and you'll have several weeks to write it.

As entering latitude and longitude is boring, you are provided with **a text file** that contains this information for many cities in the world (You can add your hometown if you wish). This file is named **world_cities.csv** and contains one line per city with the following information:

- City name
- Province or state (may be absent)
- Country
- Latitude
- Longitude

These values are **separated by commas** (.csv means "Comma Separated Values", it's a commonly used format for exchanging data). When a piece of data is missing (province or state), you get two commas in succession, for instance (first row in the file):

Aarhus, ,Denmark,56.150,10.217

The cities are in alphabetical order.

Part 1

All constant values must be set as #define so that they can be easily changed.

We aren't in this lab interested in the "province or state" information. You are asked to define a **structure** to hold city and country name, latitude and longitude.

You must **create an array containing structures** as you have defined them, read the file, and load data into the array.

1. Initially set **the maximum length for names** (city and country name) to **25**, and **the array size to 800**. Your program should **issue a warning when data is truncated or not loaded**, but it mustn't crash.
2. Set **the maximum length for names to 35**, and **the array size to 1000**. Check that your program runs without any problem.
3. **Rename the file** to **world_cities.tmp**. Run your program. It mustn't crash and should display a warning about the missing file.
4. Rename the file to **world_cities.csv** and continue with part 2.

Part 2

Once all data is loaded, your program should enter a loop from which it will exit when the user types "bye" (case-insensitive). For each iteration, it will prompt for the names of two cities.

The program must search the cities in the table by their names and retrieve the latitudes and longitudes. If the city isn't found or if the length of the name is shorter than three letters, a message must be displayed and the user must be prompted for another name. The city names should be case-insensitive.

Please note that in the file New York appears as "New York City". If people type "New York", then "New York City" must be retrieved. However, if users only type "New" (minimum acceptable length), it can match several cities. The list of the matched cities must be displayed, and the user prompted for the right one.

Finally, the names of the cities (as stored in the memory) and the distance between them must be displayed.

As before, your program should ignore all whitespaces on both ends of user inputs.