 Hand -- in

* Please collect your answers in a single .py file called **lab6\_yourname.py**
* Submit the zipped file to the assignment dropbox called “**lab6**”.
* Include appropriate comments to explain what each line or block of code accomplishes. **You must comment your code for full credit**.

 Getting Started

* You need to deal with bad user inputs in this lab.
* Under no circumstances should your program bomb (i.e. it either restarts or ends gracefully).

We will be using the same file as we did in previous lab.

[CityPop.csv](https://eipd.dcs.wisc.edu/for-credit/GEOG/FA17/geog378_labs_fa17/CityPop.csv)

 Tasks

For the following tasks, you can assume all coordinates are given in valid formats.

Task 1: (4 pts.)

In last lab, we created python scripts that read in the content of [**CityPop.csv**](https://eipd.dcs.wisc.edu/for-credit/GEOG/FA17/geog378_labs_fa17/CityPop.csv) and store the data in certain containers. Now, let’s try to replace your containers with **classes** and create **instances** to store the data.

* The name of your class should be City.
* Your class should have an \_init method to assign values to the following attributes: city name, city label, latitude, longitude, population values from 1970 to 2010.
* Create a list called Cities to store the city instances based on reading the entries in CityPop.csv.
* Print out the attributes of all cities at the end.
* Like last lab, you need to deal with bad inputs when trying to read the file.

Task 2: (6 pts.)

Add the following **methods**to your City class to make it more useful

* printDistance(self, othercity): calculate and output the distance between this city and a given city.
* printPopChange(self, year1, year2): calculate and output the population change over two years. You may assume the input values are valid.