Introduction to programming: Final project

The topic of the final project is earthquakes! You will gather the data, analyze it, visualize it, and even try to predict the future. Your note for the project will be your note for the course, we will forward it to the official university contact you indicated.

Description of the project

Part 1: Gather the data

In this part, you must gather and clean the data. We will use the data collected by IRIS, which is a consortium of over 120 US universities dedicated to the operation of science facilities for the acquisition, management, and distribution of seismological data.

Task 1. Write a script download_data.py that for each region (Polynesia, Fiji/Tonga, Cascadia, Aleutian isles, Japan region, Southeast Asia, Central America, Central South America, E. Mediterranean, East African rift, Horn of Africa) downloads an HTML document containing the 25000 newest earthquakes from http://ds.iris.edu/ieb/index.html. Use the module requests. Then, the script should parse the data in the HTML document using the module BeautifulSoup and extract a list of the 25000 newest earthquakes, where for each earthquake we store the following fields: year, month, day, time, magnitude, latitude, longitude, depth, region. Finally, write the data for each region into a separate file in a .csv format, one earthquake per line (separate the fields with commas). Put the file into a folder data inside your main working directory.

Task 2. Download the file worldcitiespop.csv from https://www.kaggle.com/max-mind/world-cities-database (you can do it by hand, don't write scripts). Put the file into a folder ../data.

Task 3. Create a module read_data.py. It should contain two functions, read_earthquakes and read_cities. These two functions should read the csv files for earthquakes and return numpy arrays containing the data. For the earthquakes, the numpy array should contain the following columns: date as a datetime object, magnitude, latitude, longitude, depth, region. For cities, the numpy array should contain the following column: city, latitude, longitude.

Part 2: Statistics on the data

All functions that you will create in this part should be in a file statistics.py.

Task 1. Write a function print_array that takes a two-dimensional numpy array and its name and prints this information. Attention, do not simply use print my_array, the information you print out should be readable by an average user who knows nothing about numpy arrays.

Task 2. Write a function earthquake_frequency. It should take the following arguments: min_magnitude, max_magnitude, time_start, time_end, list_of_regions, and a Boolean variable show. You can assume that time_start and time_end are datetime objects. The function should return a (2-dimensional) numpy array that for each region in list_of_regions contains the total number of earthquakes of magnitude between min_magnitude and max_magnitude that happened between time_start and time_end. If the variable show is True, the function should additionally print out the array using the function print_array.

Introduction to programming: Final project

Task 3. Write a function strongest_earthquakes. It should take the following parameters: an integer k, time_start, time_end, list_of_regions, and a Boolean variable show. You can assume that time_start and time_end are datetime objects. The function should return a numpy array that for each region in list_of_regions contains the informations about k strongest (by magnitude) earthquakes that happened between time_start and time_end. If the variable show is True, the function should additionally print out the k strongest earthquakes for each region in the list using the function print_array.

Task 4. Write a function city_risk. It should take the following parameters: name of a city, radius R in kilometers, time_start, time_end. It should return the number of strong earthquakes (magnitude > 5, depth < 70 km) that happened between time_start and time_end at distance at most R from the city. To compute the distance d between two points given their longitude and latitude, use the harvesine formula:

$$a = \sin^2(\Delta\varphi/2) + \cos\varphi 1 \cdot \cos\varphi 2 \cdot \sin^2(\Delta\lambda/2)$$
$$c = 2 \cdot \operatorname{atan2}(\sqrt{a}, \sqrt{(1-a)})$$
$$d = R \cdot c$$

where $\varphi 1$ and $\varphi 2$ are the latitudes, $\Delta\varphi$ is the difference between latitudes, $\Delta\lambda$ is the difference between longitudes, R is earth's radius (mean radius = 6371km); note that angles need to be in radians to pass to trig functions!