

CSI 11

In-class Project: Statistical data analysis

We received a file with a collection of test grades from a class and were asked to run some analysis on the data. In particular, we would like to find the lowest test score, the greatest test score, the median test score and the average test score. In addition, we would like to be able to check if anybody scored a particular score and to display all the scores ordered from smallest to greatest.

We expect that the data will change from class to class, from semester to semester. Moreover, we would like to give our program to people who do not know Python programming language, but who would like to run this statistical data analysis program themselves. Therefore, we need a menu in our program.

So it should work something like that upon running the program:

Step 1: the user is asked for a file name

Step 2: the user is provided with a menu:

```
Choose which report you would like to see
(1) the smallest and the greatest test scores
(2) the average of the test scores
(3) the median of the test scores
(4) check if a test score is present
(5) display all the test scores, ordered from smallest to greatest
(6) quit the program
```

Enter your choice:

Upon choosing an option, the user will be provided with the corresponding report (unless option (6) is chosen) and will be able to make more selections (so the menu is displayed again).

I already started drafting the program. Let's go over what I have. See the file [statisticalDataAnalysis.py](#) In addition, grab several input files: [input1.txt](#), [input2.txt](#), and [input3.txt](#); put them in the same folder where the program is.

Some comments:

[Def] Median of the sequence of ordered numbers (from smallest to greatest) is the “middle number” or the average of two “middle numbers”

Examples:

(1) the **median** of the sequence [1,4,7,12,45,67,89] is 12

(2) the **median** of the sequence [9,11,34,78,90,123] is $(34+78) \div 2 = 56$

To **sort** the numbers in a Python list, say `myList`, use function `sort`.

Example: `myList.sort()`

Note that the function `sort()` sorts the list in place, i.e. modifies the original list.

CSI 11

In-class Project: States

Let's develop a program that will allow the user to enter the state (using two letters, for example, NY) and get a brief information about it, namely the state's population (with reference to the date stamp of the information), the capital of the state and the population of the state capital (also with reference to the date stamp of this information).

Here is an example of the input/output of one run of the program:

```
This program provides brief information about a US state.
```

```
Enter the two letter of the state: NY
```

```
You chose New York State.
```

```
Here is what we know about it:
```

```
Albany is the capital of New York State.
```

```
The state's population is 19850000 people (as of 2017).
```

```
The capital's population is 98251 people (as of 2017).
```

This project requires two tasks to be fulfilled:

- 1) build the database and store it in a file named **states.py**,
- 2) write the program

Database:

see the file **states.py** on our web-site.

It has only two states so far in the dictionary **statesInfo**: New York (NY) and Connecticut (CT).

You need to add at least 10 other states to it.

The program:

See the file **statesProject.py**. It already has some code in it.

Finish it.