**CSE 231** **Fall 2020**

# Computer Project #6

This assignment focuses on the design, implementation and testing of a Python program that uses lists and tuples.

It is worth 50 points (5.0% of course grade) and must be completed no later than **11:59 PM on Monday, October 26, 2020).**

## Assignment Overview

In this assignment, you will practice with lists and tuples.

## Assignment Background

Nuclear power plants are still widespread in America, as most states have at least one operating plant. Commercial nuclear power plants in Michigan are in Bridgman, Newport, and Covert. To provide data to the public, the Nuclear Regulatory Commission provides a spreadsheet containing common information on these plants such as location, parent company, and how much power they are licensed to produce. The goal of this project is to gain insights into the data using lists and tuples.

The data was retrieved from https://catalog.data.gov/dataset/us-commercial-nuclear-power-reactors and modified for clarity.

## Assignment Specifications

You will develop a Python program that extracts and displays information from a data file. The basic concept is that you will read *all* the data into a list of lists and then extract data of interest from that master_list. We provide a file which is in the comma-separated-value format (file extension is `.csv`). You can open the file to look at it using a spreadsheet program such as Excel.

### `open_file () → file pointer:`

a. This function takes no arguments, prompts for a file name and returns the file pointer to that file. If no file name is given, the program will open `reactors-operating.csv`. The function will keep asking until a file is found. Use `try-except` and `FileNotFoundError`.
   The file has some non-standard characters so use the `encoding` argument as follows:
   `fp = open(file_name, 'r', encoding ="windows-1252")`
b. Parameters: `None`
c. Returns : file pointer
d. The function displays a prompt and possibly an error message.

## read_file (fp) → list of lists:

a. This function accepts a file pointer and returns a master list of lists of all the data in the file. Each line of the file will be a list in the master list. The master list will be in the same order that the data is in the file. The file is a comma-separated-value file (`.csv`). However, some data fields have commas within the field so you must use the `csv` module:

```
import csv   # place this line at the top of the program file

#place these lines in this function
reader = csv.reader(fp)
for line_list in reader:  # note that using csv.reader you get a list!
```

Do not include headers or blank lines in the final list of lists. Not having header lines makes later functions easier to write. These lines can be skipped by using the line:

```
next(reader, None)
```

b. Parameters: file pointer
c. Returns : `master_list (list of lists)`
d. The function displays nothing.

## get_reactor_location (location_string)→ string:

a. Extract and return a string with only the city name and state abbreviation from the long location string located at the column labeled "`Location`" in the data file. For example, if the input string `location_string` is `London, AR (6 MI WNW of Russellville, AR)`, then the string returned will be `London, AR`
Note that some cities have county information in parentheses so that county information needs to be removed. For example, if the `location_string` is
`Burlington (Coffey County), KS (28 MI SE of Emporia,†KS)`
return only the following, i.e. without the county information:
`Burlington, KS`
b. Parameters: `location_string (string)`
c. Returns : `string`
The format of the returned string is the city name followed by a comma and a space and then the two-character state abbreviation. Note that the city name may have multiple strings, e.g. `Bay City`
d. The function displays nothing.

## reactors_per_region (master_list)---→ [int,int,int,int]

a. This function takes as input the master list and returns a list of four integers representing the number of reactors in each **NRC region** (1,2,3,4). Return the data in this order. The region number for each plant is located in the column labeled "`NRC Region`". Make sure you convert each value to type **int**.
b. Parameters: `master_list (list of lists)`
c. Returns : list [`int, int, int, int`]
d. The function displays nothing.

### `top_react_by_mwt (master_list)---→ list of tuples`

a. This function accepts the master list and returns a sorted list of the top ten tuples where each tuple has the form:
`(MWt, plant name, location)`

   The list will be sorted based on the value of MWt from highest to lowest. Only the top ten tuples will be returned. The MWt for each plant is found in the column "`Licensed MWt`", plant name from the column "`Plant Name, Unit Number`", and the location from the return value obtained by calling `get_reactor_location`. **Remember to convert MWt to an int before sorting.** You can use slicing to extract the first (top) ten from the sorted list. Sorting should be done on the first index of the tuple, which Python will do by default.

b. Parameters: `master_list (list of lists)`
c. Returns : `list of tuples`
   sorted largest to smallest, and only the top ten are returned
d. The function displays nothing.

### `bot_react_by_mwt (master_list)---→ list of tuples`

a. This function is identical to the previous function except for sorting the list of tuples from lowest to highest MWt.
b. Parameters: `master_list (list of lists)`
c. Returns : `list of tuples`
   sorted smallest to largest, and only the ten smallest are returned
d. The function displays nothing.

### `reactors_in_state (master_list, state)---→ list of tuples`

a. The master_list of lists and a two-letter state abbreviation are supplied as inputs. This function returns a list of tuples where each tuple consists of:
`(plant name, location)`

   Where the plant name is from the column "`Plant Name, Unit Number`" and location is the return value from calling `get_reactor_location`. The tuples are not sorted and should appear in the list of tuples in the same order as they are found in the data file.

b. Parameters: `master_list (list of lists)`
c. Returns : `list of tuples`
d. The function displays nothing

### `years_active (master_list)---→ list of int`

a. This function takes the master_list and returns a list of ints, where each int is the number of active years for that plant. This data can be found in the column labeled "`Years of`

`Operation through 12/31/2019`". The resulting list should be sorted from smallest to largest.
(Challenge: write this function in one line using list comprehension)

b. Parameters: `master_list (list of lists)`
c. Returns : `list of int`
d. The function displays nothing

## `plot_years_active (years_active_list)---→ None`

a. **DO NOT MODIFY**. Given list of int as input, this functions plots a histogram showing age of all commercial reactors in the US. For more information on plotting a histogram in python see https://matplotlib.org/3.2.1/api/_as_gen/matplotlib.pyplot.hist.html
b. Parameters: `years_active_list (list of int)`
c. Returns : `None`
d. The function displays a histogram

## `display_options ()---→ None`

a. **DO NOT MODIFY**. Display a menu of options for the program.
b. Parameters: `None`
c. Returns : `None`
d. The function displays the options for main program

## `main():`

This function provides a user interface for the functions above. You should first open the file, read the file into a master list, and display the options the user can choose. The user will then be prompted to choose one of the following options:

1. Print the number of reactors in each NRC region. Use the format string `'{:<8}{:<8}{:<8}{:<8}'`

2. Print the top 10 power producing plants by MWt. Use the format string `'{:<8}{:<30}{:<10}'`. The data should line up evenly, so you will only want to display the first 25 characters of the plant name.

3. Print the bottom 10 power producing plants by MWt. Use the format string `'{:<8}{:<30}{:<10}'`. The data should line up evenly, so you will only want to display the first 25 characters of the plant name.

4. The user is continuously prompted to input a valid 2-letter state abbreviation. This string can be lowercase, uppercase or a combination since it will be converted to uppercase in main. '`Mi`', '`mi`', '`MI`', and '`mI`' are all valid inputs for '`MI`'. Once a valid code is input, you will call reactors_in_state and print the data appropriately.

   If there are no reactors in that state, print 'There are 0 reactors in {}' where {} will be replaced by the state code.

Otherwise print a header and use the format string `'{} in {}'` to print each tuple in the resulting list of tuples. The first value will be the plant name and the second will be the plant location
(Hint: checking the length of the value returned from the function call may be useful here)

5. The user should call `years_active` and use the return value to print a histogram of the data

You should be able to enter 'q' or 'Q' to quit the program, i.e. case doesn't matter

If the input is not one of the above options, you should print `'Invalid choice, please try again'`

**Assignment Notes and Hints**

1. The coding standard for CSE 231 is posted on the course website:

   http://www.cse.msu.edu/~cse231/General/coding.standard.html

Items 1-9 of the Coding Standard will be enforced for this project.

2. The program will produce reasonable and readable output, with appropriate labels for all values displayed.

3. We provide a `proj06.py` program for you to start with.

4. If you "hard code" answers, you will receive a grade of zero for the whole project. An example of hard coding is to simply print the approximate value of e rather than calculating it and then printing the calculated average.

**Suggested Procedure**

*The last version of your solution is the program which will be graded by your TA.*

*You should use the **Mimir** system to back up your partial solutions, especially if you are working close to the project deadline. That is the easiest way to ensure that you won't lose significant portions of your work if your machine fails or there are other last-minute problems.*

**Assignment Deliverable**

The deliverable for this assignment is the following file:

`proj06.py` – the source code for your Python program

Be sure to use the specified file name and to submit it for grading via the **Mimir system** before the project deadline.

## Test 1

```
Please input a file to use:
Menu
1: # Reactors in each region
2: Top energy producing reactors
3: Bottom energy producing reactors
4: List reactors in specific state
5: Plot histogram for how long reactors have been active

Choose an option, q to quit: 1

NRC 1   NRC 2   NRC 3   NRC 4
23      33      23      18

Menu
1: # Reactors in each region
2: Top energy producing reactors
3: Bottom energy producing reactors
4: List reactors in specific state
5: Plot histogram for how long reactors have been active

Choose an option, q to quit: 2

Top 10 Reactors by MWt
MWt     Reactor Name                    Location
4408    Grand Gulf Nuclear Statio       Port Gibson, MS
4016    Peach Bottom Atomic Power       Delta, PA
4016    Peach Bottom Atomic Power       Delta, PA
3990    Palo Verde Nuclear Genera       Wintersburg, AZ
3990    Palo Verde Nuclear Genera       Wintersburg, AZ
3990    Palo Verde Nuclear Genera       Wintersburg, AZ
3988    Nine Mile Point Nuclear S       Scriba, NY
3952    Susquehanna Steam Electri       Salem Township, PA
3952    Susquehanna Steam Electri       Salem Township, PA
3952    Browns Ferry Nuclear Plan       Limestone County, AL

Menu
1: # Reactors in each region
2: Top energy producing reactors
3: Bottom energy producing reactors
4: List reactors in specific state
5: Plot histogram for how long reactors have been active

Choose an option, q to quit: 3

Bottom 10 Reactors by MWt
MWt     Reactor Name                    Location
1677    Prairie Island Nuclear Ge       Welch, MN
1677    Prairie Island Nuclear Ge       Welch, MN
1775    R.E. Ginna Nuclear Power        Ontario, NY
1800    Point Beach Nuclear Plant       Two Rivers, WI
1800    Point Beach Nuclear Plant       Two Rivers, WI
1850    Nine Mile Point Nuclear S       Scriba, NY
1912    Duane Arnold Energy Cente       Palo, IA
2004    Monticello Nuclear Genera       Monticello, MN
```

```
2339    H. B. Robinson Steam Elec     Hartsville, SC
2419    Cooper Nuclear Station        Brownville, NE

Menu
1: # Reactors in each region
2: Top energy producing reactors
3: Bottom energy producing reactors
4: List reactors in specific state
5: Plot histogram for how long reactors have been active

Choose an option, q to quit: q
```

## Test 2

```
Please input a file to use: reactors.csv
Invalid filename, please try again

Please input a file to use: reactors-operating.csv
Menu
1: # Reactors in each region
2: Top energy producing reactors
3: Bottom energy producing reactors
4: List reactors in specific state
5: Plot histogram for how long reactors have been active

Choose an option, q to quit: 4

Please enter a 2 letter state code: MI
There are 4 reactors in MI:
Donald C. Cook Nuclear Plant, Unit 1 in Bridgman, MI
Donald C. Cook Nuclear Plant, Unit 2 in Bridgman, MI
Fermi, Unit 2 in Newport, MI
Palisades Nuclear Plant in Covert, MI

Menu
1: # Reactors in each region
2: Top energy producing reactors
3: Bottom energy producing reactors
4: List reactors in specific state
5: Plot histogram for how long reactors have been active

Choose an option, q to quit: ab
Invalid choice, please try again

Menu
1: # Reactors in each region
2: Top energy producing reactors
3: Bottom energy producing reactors
4: List reactors in specific state
5: Plot histogram for how long reactors have been active

Choose an option, q to quit: 4

Please enter a 2 letter state code: we
Please input a valid state
```

```
Please enter a 2 letter state code: Wi
There are 2 reactors in WI:
Point Beach Nuclear Plant, Unit 1 in Two Rivers, WI
Point Beach Nuclear Plant, Unit 2 in Two Rivers, WI

Menu
1: # Reactors in each region
2: Top energy producing reactors
3: Bottom energy producing reactors
4: List reactors in specific state
5: Plot histogram for how long reactors have been active

Choose an option, q to quit: 4

Please enter a 2 letter state code: aK
There are 0 reactors in AK

Menu
1: # Reactors in each region
2: Top energy producing reactors
3: Bottom energy producing reactors
4: List reactors in specific state
5: Plot histogram for how long reactors have been active

Choose an option, q to quit: 4

Please enter a 2 letter state code: me
There are 0 reactors in ME

Menu
1: # Reactors in each region
2: Top energy producing reactors
3: Bottom energy producing reactors
4: List reactors in specific state
5: Plot histogram for how long reactors have been active

Choose an option, q to quit: 4

Please enter a 2 letter state code: ab
Please input a valid state

Please enter a 2 letter state code: MN
There are 3 reactors in MN:
Monticello Nuclear Generating Plant, Unit 1 in Monticello, MN
Prairie Island Nuclear Generating Plant, Unit 1 in Welch, MN
Prairie Island Nuclear Generating Plant, Unit 2 in Welch, MN

Menu
1: # Reactors in each region
2: Top energy producing reactors
3: Bottom energy producing reactors
4: List reactors in specific state
5: Plot histogram for how long reactors have been active

Choose an option, q to quit: q
```

**Grading Rubric**

```
Computer Project #06                                Scoring Summary
General Requirements:
  ( 4 pts) Coding Standard 1-9
      (descriptive comments, function headers, mnemonic identifiers,
      format, etc...)

Implementation:
 ( 4 pts)  open_file function (no Mimir test)

 ( 3 pts)  read_file function

 ( 5 pts)  get_reactor_location function

 ( 5 pts)  reactors_per_region function

 ( 5 pts)  top_react_by_mwt function

 ( 5 pts)  bot_react_by_mwt function

 ( 5 pts)  reactors_in_state function

 ( 4 pts)  years_active function

 ( 5 pts)  Test 1

 ( 5 pts)  Test 2


Note: hard coding an answer earns zero points for the whole project
-10 points for not using main()
```