IND 405 - Introduction to Data Science and Analytics

Assignment # 1

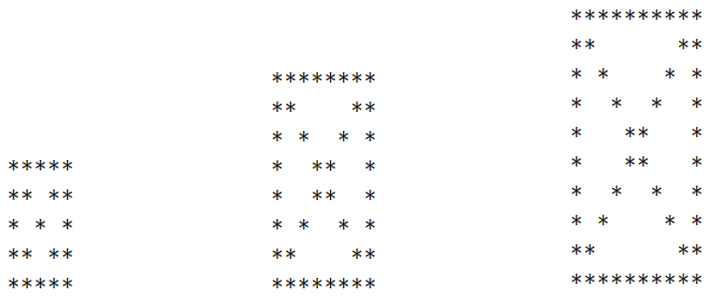
Out September 23

Due October 3, **8:00AM**

Late submissions will not be accepted!

Solve the following problems. You may only work in *groups* of 2-3 students. You *must* cite any references (texts, papers or websites) you have used to help you solve these problems. Submit a *jupyter* notebook that contains your solutions in the respective assignment submission folder in D2L. Assignment submission folders are under Assignments in the Assessment tab.

1. (20) Write a python function that takes the dimension (n) of a square as an input and output a corresponding shape that consists of '$*$', which has the shape given in Figure 1 (a-b-c) depending on the dimension parameter.



(a) $n = 5$ (b) $n = 8$ (c) $n = 10$

Figure 1: Sample shapes

2. (20) Write a python program that uses random number generation to create sentences. The program should use four lists called **article**, **noun**, **verb**, and **preposition**. The program should create a sentence by selecting a word at random from each array in the following order: **article**, **noun**, **verb**, **preposition**, **article**, and **noun**. As each word is picked, it should be concatenated to the previous words. The words should be separated by spaces. When the final sentence is output, it should start with a capital letter and end with a period. The program should generate 20 such sentences.

   The lists should be filled as follows:

```
article = ['the', 'a', 'one', 'some', 'any']
noun = ['boy', 'girl', 'dog', 'town', 'car']
verb = ['drove', 'jumped', 'ran', 'walked', 'skipped']
preposition = ['to', 'from', 'over', 'under', 'on']
```

   After the preceding program is written and working, modify the program to produce a short story consisting of several of these sentences.

3. (30) Dictionary of dictionaries can be used as a simple database. For instance, dictionary in Figure 2 can be used for tracking information about scientists.

```
db_dic = {
  'jgoodall'  : {'surname'  : 'Goodall',
                 'forename' : 'Jane',
                 'born'     : 1934,
                 'died'     : None,
                 'notes'    : 'primate researcher',
                 'author'   : ['In the Shadow of Man', 'The Chimpanzees of Gombe']},
  'rfranklin' : {'surname'  : 'Franklin',
                 'forename' : 'Rosalind',
                 'born'     : 1920,
                 'died'     : 1957,
                 'notes'    : 'contributed to discovery of DNA'},
   'rcarson'  : {'surname'  : 'Carson',
                 'forename' : 'Rachel',
                 'born'     : 1907,
                 'died'     : 1964,
                 'notes'    : 'raised awareness of effects of DDT',
                 'author'   : ['Silent Spring']}
}
```

Figure 2: A dictionary of dictionary example

Assume that the data is provided as a text file (see *dic_db_scientists.txt*). First, you need to read and save the data as a dictionary of dictionaries as shown in Figure 2. Note that first value in each line is a key, and the rest is the value. If there are multiple values to be stored, these values are separated by comma (as in the case for author key of jgoodall). Next, answer the following questions:

(a) Write a python function called db_headings that returns the set of keys used in any of the inner dictionaries. In the example in Figure 2, the function should return

$$set('author','forename','surname','notes','born','died')$$

(b) Write another function called db_consistent that takes a dictionary of dictionaries in the format described in the previous question and returns True if and only if every one of the inner dictionaries has exactly the same keys. (This function would return False for the previous example, since Rosalind Franklin's entry does not contain the 'author' key.)

4. (30) Consider a chain of stores that ship products to customers that are residing in either houses or apartments. The distances between the stores and the residences are given in file *q4_distance.txt*. Shipping costs are dependent on the distance values and each given product have a different cost multiplier for shipping cost calculation. Let p_k be the cost multiplier for product $k \in P$, d_{ij} be the distance between store $i \in I$ (for a set of stores I) and residence $j \in S_H \cup S_A$ (for a set of houses S_H and apartments S_A). Also, u_r^{LB} and u_r^{UB} represent the min and max costs for shipment for residence type $r \in \{H, A\}$. Then, for product $k \in P$, the cost of shipment c_{ij} between store $i \in I$ and residence j of type $r \in \{H, A\}$ can be calculated as

$$c_{ij} = p_k \times \left(u_r^{LB} + (u_r^{UB} - u_r^{LB}) \times \frac{(d_{ij} - \min\limits_{i \in I, b \in S_r} \{d_{ib}\})}{\max\limits_{i \in I,\ b \in S_r} \{d_{ib}\} - \min\limits_{i \in I, b \in S_r} \{d_{ib}\}} \right) \tag{1}$$

Following values in the problem data are fixed:

$$u_H^{LB} = 5, \qquad u_H^{UB} = 10, \qquad u_A^{LB} = 3, \qquad u_A^{UB} = 7$$

Your task is to create the cost matrices for a given set of products (it can be any number of products) and their respective cost multipliers. These cost matrices should be written to a file in a specific format. For instance, for the following input (which will be provided as a dictionary of values), e.g.,

```
p = {1: 1.0, 2: 1.2}
```

your program is expected to generate the cost matrices that are stored in *q*4_*answer*_*product*1.*txt* and *q*4_*answer*_*product*2.*txt* for each product. Note that each file has separate matrices for each residence type. Calculations over equation (1) is provided in *q*4_*calculations*.*xlsx*

P.S. In order to get full credit from this question, your output files should have the same format as the sample answers provided in *q*4_*answer*_*product*1.*txt* and *q*4_*answer*_*product*2.*txt*.