GIS 5578
LAB 1: Introduction to Python scripting

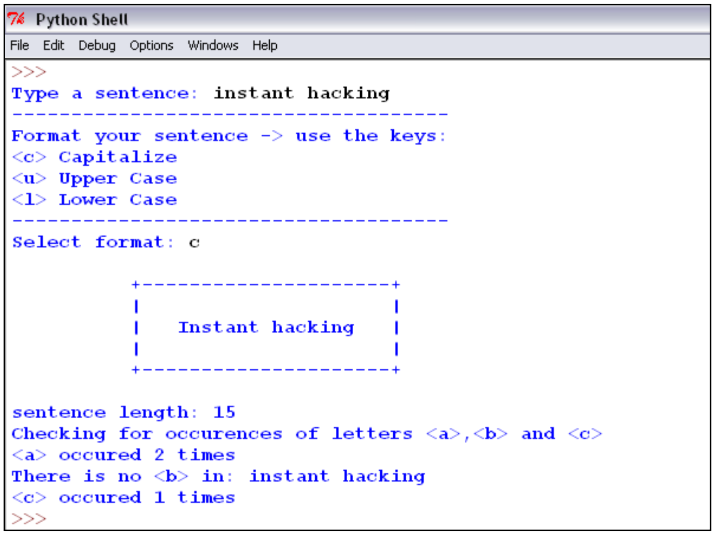
Write and test two Python scripts following the instructions below. You may use the two given script templates to help with your coding. These scripts also include detailed instructions about how to comment the code. A possible approach to accomplish the task is to start from scratch by testing the calculations one at a time rather than trying to modify the whole template at once. Also, there are some questions about the existing code. Please address these questions by using comments within your code.

The following two scripts need to be created:

1. frameInput.py
   Your goal is to display user input (sentence) in a simple frame. You should also apply some basic string formatting: capitalize, upper case all letters, lower case all letters, replace all x with y. Finally, you will calculate and display simple stats for the input sentence: its length and the number of times letter ‘a’, ‘b’, and ‘c’ occur in the sentence.

   Figure 1 depicts, the run time of the script. The program implements the following programming skills: decision making (if-elif-else), handling user input, sequence operations (adding, multiplying), and importing and using modules.
2. grid.py
   This time you will work on a simple 2 dimensional matrix (grid). The objective is to calculate its stats like size, sum of values, average, min and max value, and the range of values. Additionally, you will display the values for a selected row, selected column. The value and row/column to calculate on will be obtained from the user. Finally, you will use the ‘enumerate’ function to display row id and its values for all rows in the grid. See figure 2 for sample script run.
   The script covers the following programming skills: loops, the list object, and built-in functions, importing the numpy module.

Upload your completed (and tested!) scripts to Moodle.

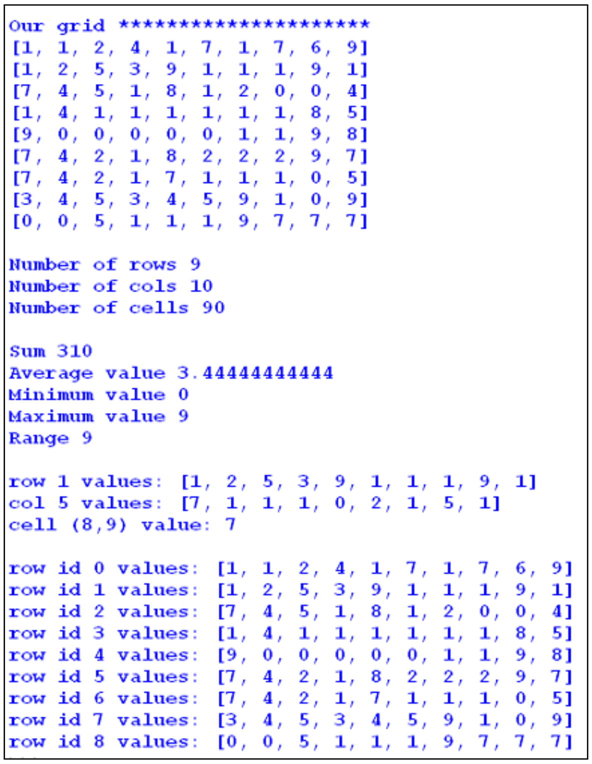


```
>>>
Type a sentence: instant hacking
-------------------------------------
Format your sentence -> use the keys:
<c> Capitalize
<u> Upper Case
<l> Lower Case
-------------------------------------
Select format: c

          +---------------------+
          |                     |
          |   Instant hacking   |
          |                     |
          +---------------------+

sentence length: 15
Checking for occurences of letters <a>,<b> and <c>
<a> occured 2 times
There is no <b> in: instant hacking
<c> occured 1 times
>>>
```

Figure 1 Sample run of *frameInput.py*



```
Our grid *********************
[1, 1, 2, 4, 1, 7, 1, 7, 6, 9]
[1, 2, 5, 3, 9, 1, 1, 1, 9, 1]
[7, 4, 5, 1, 8, 1, 2, 0, 0, 4]
[1, 4, 1, 1, 1, 1, 1, 1, 8, 5]
[9, 0, 0, 0, 0, 0, 1, 1, 9, 8]
[7, 4, 2, 1, 8, 2, 2, 2, 9, 7]
[7, 4, 2, 1, 7, 1, 1, 1, 0, 5]
[3, 4, 5, 3, 4, 5, 9, 1, 0, 9]
[0, 0, 5, 1, 1, 1, 9, 7, 7, 7]

Number of rows 9
Number of cols 10
Number of cells 90

Sum 310
Average value 3.44444444444
Minimum value 0
Maximum value 9
Range 9

row 1 values:  [1, 2, 5, 3, 9, 1, 1, 1, 9, 1]
col 5 values:  [7, 1, 1, 1, 0, 2, 1, 5, 1]
cell (8,9) value: 7

row id 0 values:  [1, 1, 2, 4, 1, 7, 1, 7, 6, 9]
row id 1 values:  [1, 2, 5, 3, 9, 1, 1, 1, 9, 1]
row id 2 values:  [7, 4, 5, 1, 8, 1, 2, 0, 0, 4]
row id 3 values:  [1, 4, 1, 1, 1, 1, 1, 1, 8, 5]
row id 4 values:  [9, 0, 0, 0, 0, 0, 1, 1, 9, 8]
row id 5 values:  [7, 4, 2, 1, 8, 2, 2, 2, 9, 7]
row id 6 values:  [7, 4, 2, 1, 7, 1, 1, 1, 0, 5]
row id 7 values:  [3, 4, 5, 3, 4, 5, 9, 1, 0, 9]
row id 8 values:  [0, 0, 5, 1, 1, 1, 9, 7, 7, 7]
```

Figure 2 Sample run of *grid.py*