**CMPSC-132: Programming and Computation II**
Spring 2019

# Lab #3

Due Date: 01/25/2019, 11:59PM

*Read the instructions carefully before starting the assignment. Make sure your code follows the stated guidelines to ensure full credit for your work.*

**Instructions:**

- The work in this lab must be completed alone and must be your own.
- **Download the starter code file from the LAB3 Assignment on Canvas. Do not change the function names or given started code on your script.**
- A doctest is provided as an example of code functionality. Getting the same result as the doctest does not guarantee full credit. You are responsible for debugging and testing your code with enough data, you can share ideas and testing code during your recitation class.
- Each function must return the output (Do not use print in your final submission, otherwise your submissions will receive a -1 point deduction)
- Do not include test code outside any function in the upload. Printing unwanted or ill-formatted data to output will cause the test cases to fail. Remove all your testing code before uploading your file (You can also remove the doctest). Do not include the input() function in your submission.

**Goal:**

**[5 pts]** Write the function *countWords*(*txt*), where *txt* is a string. This function **returns** a dictionary whose keys are included words, and values their word counts. Count contraction words as (can't), (don't), (isn't), etc. If a string is not provided as an input, the function must **return** a string with an error message (the string 'error' is enough).
**Hint**: Remove punctuation before counting the words, careful with contraction words

**[5 pts]** Write the function *studentGrades(gradeList)* that takes a nested list with the following structure:

- First list is always a descriptive header.
- Subsequent lists hold all the data.
- For lists that hold data, the first element is always a string, the rest of the elements are numeric values. Each list (except for the first one) represents the grades of the student and the first element of each list contains the name of the student.

```
grades = [

    ['Student', 'Quiz 1', 'Quiz 2', 'Quiz 3'],# List 1, header
    ['John', 100, 90, 80],
    ['McVay', 88, 99, 111],
    ['Rita', 45, 56, 67],
    ['Ketan', 59, 61, 67],
    ['Saranya', 73, 79, 83],
    ['Min', 89, 97, 101]]
```

and **returns** ONE list with the average score for each student in INTEGER format. If a list is not provided as an input, the function must **return** a string with an error message (the string 'error' is enough)
**Hint**: The method *sum* adds all the numeric elements of a list (sum([1,2,8.1]) returns 11.1).

**Deliverables:**

- Submit your code in a file name LAB3.py to the Lab3 GradeScope assignment before the due date

**Notes:**

- Verify your code is returning and not printing the output, otherwise, your code will return None and the test cases will fail
- To provide input to your functions, include –i as an argument in the command line to run an interactive Python session (`python –i LAB3.py`)
- A doctest is provided in the starter code. To run it, type `python -m doctest -v LAB3.py`
- Mac users type python3