CMPSC-132: Programming and Computation II
Spring 2019

Lab #6

Due Date: 02/08/2019, 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 LAB6 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:

[10pts] Write the class *Vector* that supports the basic vector operations. Such operations are addition (+) and subtraction (-) of vectors of the same length, dot product (*) and multiplication (*) of a vector by a scalar. All methods must **return** (not print) the result. Your class should also support the rich comparison for equality (==)

- You must use the special methods for those 4 operators in order to override their behavior
- You will need other special methods to achieve a **legible object representation**.
- Dot product and multiplication by scalar use the same operator, so you must check the type of the object in order to decide which operation you have to perform
- For addition and subtraction, you must check that vectors have the same length
- **The dot product results in a scalar, not a Vector**
- **The rest of the methods must return a Vector object** (not a string with the word Vector)
- Test your code, this is how you ensure you get the most credit out of your work!!
- When returning error messages, make sure your string contains the word 'error'
- Check the doctest for object behavior examples. Vector size is variable
- Hint: Section 3.3.8. Emulating numeric types in https://docs.python.org/3/reference/datamodel.html#emulating-numeric-types

Deliverables:

- Submit your code in a file name LAB6.py to the Lab6 GradeScope assignment before the due date