CMPSC-132: Programming and Computation II
Fall 2019

Lab #5

Due Date: 09/27/2019, 11:59PM EST

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 files from the LAB5 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. As a reminder, Gradescope should not be used to debug and test code!
- 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 *Complex* that supports the basic complex number operations. Such operations are addition (+), subtraction (-) and multiplication (*) of complex numbers, and multiplication (*) of a complex by a scalar (float or int). All methods must **return** (not print) the result. Class also supports the rich comparison for equality (= =)

- Check the doctest for object behavior examples.
- 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**.
- Multiplication by scalar and by another complex number use the same operator, so you must check the type of the object in order to decide which operation you have to perform
- **Rich comparison returns a Boolean value**
- **<u>The rest of the methods must return a new Complex object</u>**, this means the original objects should not change after the operation is performed.
- 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’
- You are not allowed to modify the constructor or any given code.
- Hint: Section 3.3.1. Basic customization (__*eq*__) and Full Section 3.3.8. Emulating numeric types in https://docs.python.org/3/reference/datamodel.html#emulating-numeric-types

Write the property method conjugate into the Complex class. The complex conjugate of the complex number *z* = *x* + *y*i is given by *x* – *y*i

Remember, **you are not allowed to modify the given constructor**. The method *conjugate* **must be a property method, otherwise, no credit will be given**. Property methods are discussed in the video lectures!

You can visit https://en.wikipedia.org/wiki/Complex_number for complex number formulas

Deliverables:

- Submit your code in a file name LAB5.py to the Lab5 GradeScope assignment before the due date