

Lab 3 Report

Chad Lape and Colton Murray

February 12th, 2020

1 Objectives

This lab explored the concepts within object oriented programming of creating classes and then testing instances of those classes within objects. This is currently how a majority of code in the world is written due to how versatile it is this way. Thus the objectives of this lab is useful for if one were to follow a career in CS.

2 Class Design

2.1 Task 1

At this version we had created the complex number with a pair of ints as the coefficients of the class. This was due as to ease of use and no need for anything more complex than an int. Then, in addition to setters and getters, created a print and to string function. The 3 constructors were also created and implemented to take from cartesian and polar coordinate systems.

2.2 Task 2

No changes to the base .h file, although the implementation for the setters and getters remained here

2.3 Task 3

During the creation of this task there was a minor issue thought up due to the division. We had felt it would be better to divide doubles instead of ints, this was somewhat last minute and thus had to turn member arguments and variables into doubles instead of ints. This was deemed better due to their accuracy when dealing with division of numbers. This was the only change to the code.

2.4 Task 4

```
Enter Cartisian(1) or polar(2) complex number?
1: List Paragraph Times New Roman 12
Enter x value of complex number ( x + yi ):
3
Enter y value of complex number ( x + yi ):
5
what operation would you like to perform?
1: + 2: - 3: * 4: / 5: == 6: Show Polar Coordinates
1
Enter Cartisian(1) or polar(2) complex number?
1
Enter x value of complex number ( x + yi ):
2
Enter y value of complex number ( x + yi ):
3
Answer: 5 + 8i

Continue? yes(1) no(2):
1
Enter Cartisian(1) or polar(2) complex number?
1
Enter x value of complex number ( x + yi ):
1
Enter y value of complex number ( x + yi ):
2
what operation would you like to perform?
1: + 2: - 3: * 4: / 5: == 6: Show Polar Coordinates
2
Enter Cartisian(1) or polar(2) complex number?
2
Enter Radius:
5
Enter Angle (Deg):
30
Answer: -3.3 + -0.5i

Continue? yes(1) no(2):
```

```
Continue? yes(1) no(2):
1
Enter Cartisian(1) or polar(2) complex number?
2
Enter Radius:
4
Enter Angle (Deg):
30
what operation would you like to perform?
1: + 2: - 3: * 4: / 5: == 6: Show Polar Coordinates
3
Enter Scalar: 2
Answer: 6.9 + 4i

Continue? yes(1) no(2):
1
Enter Cartisian(1) or polar(2) complex number?
1
Enter x value of complex number ( x + yi ):
4
Enter y value of complex number ( x + yi ):
6
what operation would you like to perform?
1: + 2: - 3: * 4: / 5: == 6: Show Polar Coordinates
4
Enter Scalar: 2
Answer: 2 + 3i

Continue? yes(1) no(2):
```

The tester required no change to the original class other than some miscellaneous debugging.

3 Compilation

Compiled version included within the base file system. Source folder is included under ./src and can be compiled with mingw g++ or other compilation system. Otherwise the Lab3 folder should be importable to visual studio community. kindly email if any issues with compilation for immediate resolution.

4 Contributions

4.1 Chad

- Created complex.h
- Implemented complex.cpp excluding constructors
- Helped debug test.cpp
- Created Lab report

4.2 Colton

- Created Test.cpp and implemented the tester
- Figured out conversion between polar and cartaesian
- Tested class