**Hakeem S Buchanan**

**Problem Statement:**

The problem put forth by the project designer was to create a four-function calculator that did operations using complex numbers.

The code I have developed has solved the problem put forth and follows the specifications.

**Requirements Analysis:**

The resulting design had to have the following specifications:

* The user interface had to be menu driven(giving users a choice with how to proceed. Add, subtract, etc.). However should there be an incorrect input, there needed to be an error message and the menu would have to be redisplayed.
* The user must be able to enter two complex numbers “a and b” . Complex numbers had to be entered in either polar or rectangular form. Once entered the program had to output both forms of the same value.
* The calculations should be completed inside the class.
* The user interface had to be the main() function
* Finally the code must utilize a class called Complex Number(No Space).

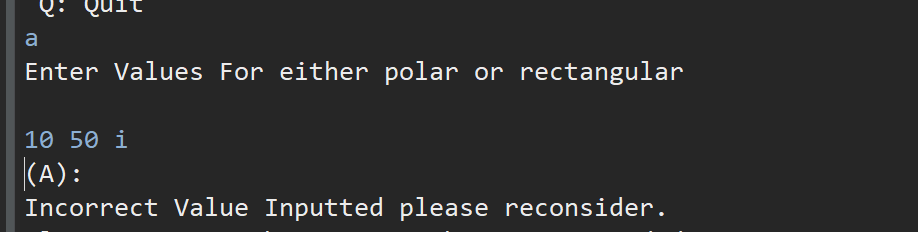
**Problems Encountered:**

1. Form recognition
2. Move to Ans
3. Swap
4. Member function not declared
5. One of the main problems I faced while designing was figuring out to get the form to be recognized as the user inputs it. I tried to do an if statement initially, or at least have the user input it separately. However, that didn’t really fit with the rest of the design. What I ended up doing was creating a member function that received the values and the form all together as a one “cin input”. This worked way better that my other tries.
6. The move to ans function of the calculator refused to work. I tried taking the value and putting it into a temporary object and then sending it into the direction I needed it to go. In the end, I just sent it straight to (‘a’) or (‘b’) object overwriting what was already there.
7. The swap was similar to the move ans function. It just would not work and the values kept telling me they weren’t numbers.(“NAN”). MY solution was the problem of the move ans. I had to create temp values that transferred the values for me. And they needed to be in order apparently.
8. For a lot the operator overloading member functions I received an error that said Member function not declared. This hindered me from compiling. The fix was an easy one. I needed to make sure that I added declarations (void, int, double) before the class name.

**Testing:**

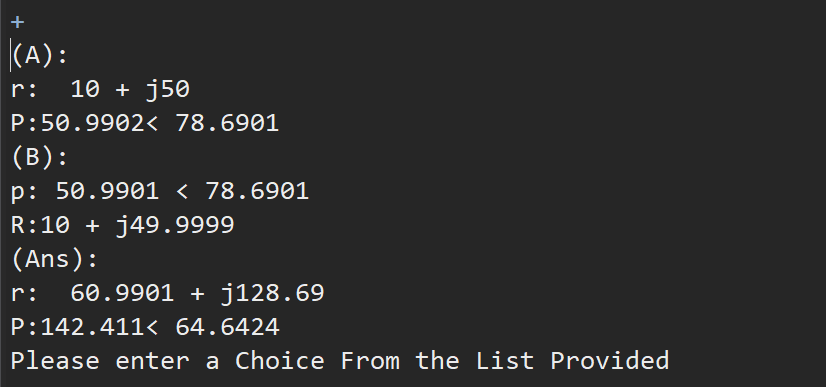
To test to my code, I inputted complex numbers that I could test the from input and conversions. In the screenshot below I tested the complex value 10+50ifor a and 50.999<79.6901 p for b. Then I added them to test the addition operator.

This value was inputted in the incorrect format so the calculator outputs an error message.



Correct input A





Discussion:

When writing this code, I learned that while writing to have a running list of mistakes made. This way if you make the same mistake again you can fix it fast. I also learned to split up large projects into parts. Doing this ensures that you at least have working parts so that if the final product doesn’t work you can dissect it and see the final product.