EECS 10: Assignment 6

October 28, 2005

Due Monday 11/14/05 at 12:00pm
Note: This is a two-week assignment.

1 Calculations with Complex Numbers [80 points + 10 bonus points]

Recall that a complex number is one that has a real part and an imaginary part, often written as ***x*** + ***y***i. *i* is defined as *i* * *i* = -1. **x** represents the real part, while ***y*** represents the imaginary part.

Any complex number can also be represented in polar coordinates, as a magnitude and an angle. To convert the regular format ***x*** + ***y****i* into polar coordinates, one can use the following equations:

magnitude = **|z| = sqrt($x^2 + y^2$)**
angle = **arctan(y/x)**

Note that we will use degrees as the units for the angles (not radians).

To refresh your memory on complex numbers, consult your math books and/or use one of the following links:

http://www.clarku.edu/~djoyce/complex/
http://mathworld.wolfram.com/ComplexNumber.html

The formulas required for the assignment have been provided below, but you may prefer to have more background information.

Your program should be a menu driven program where the current complex number is always displayed in both regular and polar format and the user then selects operations from the following menu:

```
Current number: 9 + 4i
Polar notation: 9.849 @ 23.962 degrees

( 1) Set the complex number in regular format
( 2) Set the complex number in polar format
( 3) Compute the absolute value of the number
( 4) Compute the reciprocal of the number
( 5) Add another complex number
( 6) Subtract another complex number
```

```
( 7) Multiply by another complex number
( 8) Divide by another complex number
( 9) "Rotate" the number
(10) Raise the current number to an integer power
(11) Exit

Please enter your choice:
```

Your program should perform the following operations for the options:

(1)Set the complex number in regular format

This function should allow the user to set the value of the complex number by entering the real and imaginary parts separately.

```
Please enter the real part: 3.0
Please enter the imaginary part: 4.0
```

(2)Set the complex number in polar format

This function should allow the user to set the value of the complex number by entering the magnitude and phase (angle).

```
Please enter the magnitude: 10.5
Please enter the phase: 30.0
```

Hint: To convert the polar coordinates into regular format, you may use:

$x + yi = r \cos \Theta + i\, r \sin \Theta$, where r is the magnitude and Θ is the phase.

(3)Compute the absolute value of the number

Your program should compute the absolute value of the current number and display it on the screen. The current value should not change due to this operation.

```
The absolute value of 3.0 + 4.0i is 5.0
```

Hint: The absolute value , **|z|,** can be expressed as**:**

$$|z| = \mathbf{sqrt}(x^2 + y^2)$$

(4)Compute the reciprocal of the number

This time your program will instead calculate the reciprocal of the current number and store the result back as the new current number.

Hint: The reciprocal is computed as follows:

$$1 / (x + yi) = x / (x^2 + y^2) - yi / (x^2 + y^2)$$

Example:
The reciprocal of $3.0 + 4.0i$ is $0.12 - 0.16i$

(5)Add another complex number

Here, a second complex number should be entered by the user. Then, the program should compute the addition of the two numbers and assign the result back to the current number. (If you prefer, you may store the result of the addition first in a temporary value and then copy to the current number).

```
Current number: 0 + 0i
Polar notation: 0.0 @ 0.00 degrees
. . .

Please enter the real part: 3.0
Please enter the imaginary part: 4.0

Current number: 3.0 + 4.0i
Polar notation: 5.0 @ 166.937 degrees

. . .
```

Hint: To add or subtract two complex numbers, just add or subtract the corresponding real and imaginary parts.

Example:
$(3.4 + 7.8\ i) + (2.3 + 34\ i) = (5.7 + 41.8\ i)$

(6)Subtract another complex number

This should follow the very same procedure as for option (2), except that the 2nd number is subtracted from the current number.

(7)Multiply by another complex number

Again, this is very similar to option (2). Here, however, the 2nd number should be multiplied by the current number.
Hint: $(x + yi)(u + vi) = (xu - yv) + (xv + yu)i$.

(8)Divide by another complex number

Again, this is very similar to option (2). Here, however, this time the current number should be divided by the 2nd number.

Hint: You can reuse the reciprocal (step 4) and multiplication (step 7) functions.

(9)"Rotate" the number

(Bonus feature). Since this is a bonus feature you should investigate what it takes to perform a rotation operation and then implement this feature.

(10)Raise the current number to an integer power

(Bonus feature). Since this is a bonus feature you should investigate what it takes to perform a power operation and then implement this feature.

(11)Exit

This option will terminate your program.

Implementation

To implement your program, you should use separate functions to perform the operations required by the menu options. Name your functions appropriately. The function names should detail what the function does (e.g. **DisplayComplexNumber**() would display the current complex number, **AddComplex**() would perform the addition, etc.).

Define the current and 2nd complex numbers as global variables so that all functions have easy access to them. For any temporary results, use local variables.

More specifically, there should be four global variables representing the current and second complex numbers used in your program:

- `Real` and `Imag`: used as the current number and the one on which all operations are performed. The result of each operation should also be stored in `Real` and `Imag` for use in the next operation.
- `Real2` and `Imag2`: used as the operand complex number. That is, when performing addition, subtraction, etc., `Real`, `Imag` should be the first, and `Real2`, `Imag2` should be the second operand. This should be consistent throughout the program.

Script file

To demonstrate that your program works correctly, perform the following steps and submit them as your script file:

1. Set the current complex number to 2.0 + 5.0*i*
2. Add the following complex number to the current value: 3.4 + 2.44*i*
3. Subtract the following complex number from the current value: 9.0 + 9.99*i*
4. Reset the current value with the following number in polar notation: 4.0 at angle of 30 degrees
5. Multiply by 6.0 + 3.5*i*
6. Divide by 7 + 60*i*
7. Compute the absolute value of the current complex number
8. Compute the reciprocal of the current complex number

Bonus sections:

9. Rotate the current number by 90 degrees
10. Raise the current number to the 5^{th} power.

What to submit

Use the standard submission procedure to submit the following files stored in the hw6 directory:

- complex.c
- complex.script
- complex.txt