Due date
Monday, September 16, 2013

Program objectives
The objectives of this assignment are as follows.
An ability to analyze a problem, and identify and define the computing requirements appropriate to its solution (ABET b).

Point value
This program is worth 15 points. The distribution will be as follows.

Criterion	**Value**
Globals	1
Style	3
Class	4
Output	7

Delivery method
Please archive your files using the tar command (see below). Use the name hw2.tar (all lowercase) for your archived file and submit it to the class account at csc2421@orion.ucdenver.edu. Attach this file to the email that you send to the class account. Put HW2 in the subject field and your name in the body of the email.
tar –cvf hw2.tar hw2.cpp hw2functions.h hw2functions.cpp complex.h complex.cpp

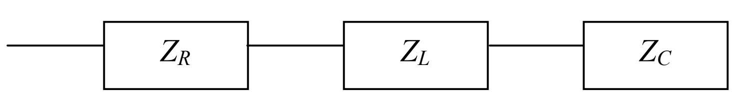
Background
Analysis of electronic circuits in the sinusoidal steady state can become quite complex. Using *phasors* to represent voltages and currents and *complex impedance* to represent circuit elements, this analysis can be greatly simplified. This assignment will consider complex impedance of series and parallel circuits consisting of resistors, inductors, and capacitors, where the value of a resistor R is expressed in *ohms*, the value of an inductor L is expressed in *henries*, and the value of a capacitor C is expressed in *farads*. The complex impedance Z of a circuit consisting of RLC components can be expressed as follows.

$Z = Z_R + Z_L + Z_C$ for a series circuit (1)

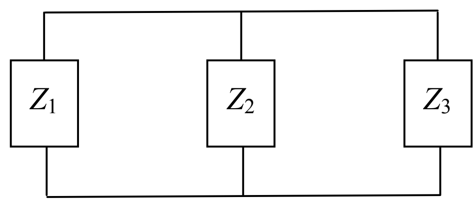
$1/Z = 1/Z_R + 1/Z_L + 1/Z_C$ for a parallel circuit (2)

In (1) and (2), $Z_R = R$, $Z_L = j\omega L$, and $Z_C = -j/\omega C$, and ω = sinsusoidal steady-state frequency in rad/sec of the driving force (voltage or current). In (2), branch 1 consists of resistance R only, branch 2 inductance L only, and branch 3 capacitance C only. Also notice that a series circuit consists of components that are connected in a sequence, whereas, a parallel circuit represents components connected in parallel as shown in the following diagrams. In the parallel circuit shown below, Z_1 may consist of a combination of series connected Z_R, Z_L, and Z_C components.

Series Circuit



3-Branch Parallel Circuit



To compute complex impedance, we first need to transform the *RLC* components to complex values. First, however, define a complex number z as follows.
$z = x + jy$, where x and y are real numbers and j is the imaginary operator defined as $\sqrt{-1}$. $z = x + jy$ is known as the rectangular form of a complex number. We call x the real part of z and y the imaginary part of z. Complex numbers support many operations including addition, subtraction, multiplication and division. Furthermore, there are several identities associated with complex numbers. For example,

$z = x + jy = r(\cos\theta + j\sin\theta)$, where $r = \sqrt{x^2 + y^2}$ and $\theta = \tan^{-1} y/x$. (3)

In addition, by *Euler's* formula, we have the *polar* form of a complex number z.

$z = re^{j\theta}$ where r and θ are defined above. (4)

In circuit analysis, this is often written as follows. $z = r\angle\theta$ (5)

Other identities are as follows.

$x - jy = r\angle\text{-}\theta$ (6)

$z_1z_2 = (r_1\angle\theta_1)(r_2\angle\theta_2) = r_1r_1\angle(\theta_1+\theta_2)$ (7)

$z_1/z_2 = (r_1\angle\theta_1)/(r_2\angle\theta_2) = (r_1/r_2)\angle(\theta_1\text{-}\theta_2)$ (8)

$z_1 + z_2 = (x_1+x_2) + j(y_1+y_2)$ (9)

$z_1 - z_2 = (x_1\text{-}x_2) + j(y_1\text{-}y_2)$ (10)

$1/j = -j$ (11)

Notice that the polar form is more convenient when performing multiplication and division, while the rectangular form is more convenient when performing addition and substraction.

Examples

Let $z_1 = 2+j3$, $z_2 = 4 - j2 \rightarrow z_1 = \sqrt{2^2 + 3^2}\ \angle\tan^{-1}3/2 = \sqrt{13}\angle 56.3^0$, and $z_2 = \sqrt{4^2 + (-2)^2}\ \tan^{-1}\text{-}2/4 = 2\sqrt{5}\ \angle\text{-}26.57^0$

$z_1 + z_2 = (2+4) + j(3+(-2)) = 6+j$

$z_1\text{-}z_2 = (2\text{-}4) + j(3\text{-}(-2)) = -2 + j5$

$z_1z_2 = 2\sqrt{13}\sqrt{5}\ \angle 29.74^0$

$z_1/z_2 = ((\sqrt{13}/2\sqrt{5})\angle 82.88^0$

Now, using these identities and transformations, we represent Z_R as R, Z_L as $j\omega L$ and Z_C as $-j/\omega C$, where ω is the sinsusoidal steady state frequency of the phasor current or voltage given in rad/sec.

Example 1

Let R = 75 ohms, L=2×10^{-3} henries, C=5×10^{-6} farads, and ω=5000 rad/sec.
Z_R=R=75, Z_L=$j\omega L$ = j(5000×2×10^{-3}) = j10 ohms, Z_C = $-j/\omega C$ = $-j$/(5000×5×10^{-6}) = $-j$40

Example 2

Let R, L, C, and ω be as above, and let R, L, and C be in a series circuit. Then,
$Z = Z_R + Z_L + Z_C = 75 + j10 - j40 = 75 - j30$ ohms.

Example 3

Let R, L, C, and ω be as above, and let R, L, and C be in a parallel circuit. Then,
$1/Z = 1/Z_R + 1/Z_L + 1/Z_C = 1/75 + 1/j10 - 1/j40 = 1/75 + j3/40 \rightarrow Z = 1/(1/75 + j3/40)$.

Example 4

Let R, L, C, and ω be as above, and let R and L be in series with each other and this combination be in parallel with C.
$1/Z = 1/(Z_R+Z_L) + 1/Z_C$ (You do the math).

In a two-branch parallel circuit such as in example 4, the equivalent impedance Z can be computed as follows.
$Z = Z_1Z_2/(Z_1+Z_2)$, where Z_1 is the total series impedance in branch 1 and Z_2 is the total series impedance in branch 2. In example 4, $Z_1 = Z_R + Z_L$ and $Z_2 = Z_C$.

In multiple parallel branches, we have,
$1/Z = 1/Z_1 + 1/Z_2 + \ldots + 1/Z_n$, where each of the Z_i for i=1, 2, … , n is the series impedance of branch i.

Problem

Design a class named **complex** that represents a complex number $z = x + jy$, then write global functions to compute and return the complex impedance Z of series and parallel circuits consisting of R, L, and C components for a given frequency ω.

Class requirements

Here are the minumum class requirements.

1. Class complex must have at least one constructor. The required constructor will have the following prototype.
   complex(double, double);
2. Get and set functions for getting and setting the real and imaginary parts of the complex number. Notice that there are two functions for the real part (get and set) and two functions for the imaginary part (get and set).
3. A display function that displays a complex number in the following format.
   (x, y), where x is the real part and y is the imaginary part.

4. A function named mag that returns the magnitude of a complex number; that is, the function computes and returns $\sqrt{x^2 + y^2}$.
5. A function named arg that returns the polar angle of a complex number; that is, the function computes and returns $\tan^{-1} y/x$.
6. Overloaded operators +, -, *, and /.

Minimum global function requirements
The minimum requirements for globals are as follows. However, feel free to write other globals as you deem necessary.
1. A function named s_impedance that computes and returns the complex series impedance of its arguments which are Z_R, Z_L and Z_C.
2. A function named p_impedance that computes and returns the complex parallel impedance of its arguments which are Z_R, Z_L and Z_C.

Program requirements
Your driver (hw2.cpp) must do the following.
1. Short main function mostly consisting of top-level function calls.
2. Error checking where appropriate.
3. Greet the user with a brief description of the program.
4. Prompt for and get the number of parallel branches in the circuit. Notice that the minimum value of this number is 1 which would represent a series circuit.
5. Prompt for and get the R, L, and C values for each branch.
6. Prompt for and get the sinusoidal steady state frequency ω.
7. Compute and display the rectangular form of the total complex impedance Z for the circuit; that is, in the form $Z = (x, y)$ where x is the real part of the complex impedance Z, and y is the imaginary part of the complex impedance Z.
8. Program style that includes (but not limited to) the following.
    a. Pre and post conditions.
    b. Header comments in all files that include the author, course and term, file name and brief description of the file. The driver (hw2.cpp) should describe the nature of the programs.
    c. Whitespace and indentation.
    d. Self-describing names for functions, variables, constants, etc.
    e. Macro guard around the class definition.

Program notes
Your grade will be computed on the basis of correctness and style. The grader reserves the right to deduct for issues not explicitly stated in the problem specification. For example, if the grader deems that your main function is too long or has code that could logically be converted to a function call, a deduction may ensue. You may use libraries, such as <cmath> to help you compute some of the mathematical expressions.