SOFTWARE ENGINEERING ASSIGNMENT 1 REFLECTION

By UMEIGBO, Kosidinna Kingsley; CID: 01201479

I wrote a program that can convert between phasors and waveforms and vice versa. In our Analysis of Circuits course, we cover phasors which is basically trying to represent a sinusoidal graph into a complex number representing its parameters at time t=0. The phasor can be in the rectangular form of complex number or the exponential form. The waveform can also appear into forms: One containing just ‘Xcos (wt)’ and ‘Ysin (wt)’ and the other containing the magnitude of the waveform, A, and the phase angle, ɸ, in the form containing the both of them as ‘Acos (wt + ɸ)’. I wrote a program that can represent all of these forms and then you can select whichever one you feel is convenient for you. My program contains a function, arg(x, y), that finds an argument of a phasor taking into account the various quadrants of a unit circle and also the fact that argument of a complex number falls in the range of –π < ɸ ≤ π. The value of A is the modulus of the complex phasor. One other rule I had to take into consideration was that a phasor of ‘X + jY’ reflects to a waveform of the expression: ‘Xcos (wt) – Ysin (wt)’.

I first thought about this idea during one of the lectures for this course. My initial idea was to convert from just the Rectangular complex number form of the phasor form to waveform. I still thought I could do more so I added the Exponential form of the phasor. I felt confident with my plan but I thought: “Why not extend the program to be able to convert from either of the forms to the other including all forms of each?” And so that was how the program I wrote turned out to be. I drew inspiration from the interest I had in this topic in the sense that you can represent a waveform in the form of a complex number at time t=0. Thinking about it, it made some sense as complex numbers are also involved with cosine and sine of angles too and the waveform treated at the beginning of this topic was sinusoidal.

If I wanted someone else to write this program, I would tell them first of all to consider the various quadrants of a unit circle and write a function that finds the argument of any phasor taking into account its quadrant location. Then the modulus can be found from the expression: , where X is the real part of the phasor and Y is the imaginary part of the phasor. The rest of the program would be a series of inputs and outputs.

The topics involved here are the Basics I/O, Conditionals and Functions. The ‘cin >>’ and ‘cout <<’ are part of the Basics I/O topic. My program contained if, else and else if statements when the user had to input a number depending on what they wanted to achieve from the program. The topic on Functions was created and used to find the argument of any phasor. I wish there was a way I could have incorporated the use of loops but from the look of things, I didn’t think they were needed here.

In terms of assessment, one of the criteria will be the question: “Does the program work?” Apart from that part, it would be quite hard to assess a program as people write different things. If there is another important one of these criteria to assess, it would be creativity and effort. In terms of assessing my program, first of all, it works. Second of all, it is quite innovative. I did take into account all possibilities of the phasors. The only downside of the program is that when converting from waveform to phasor, the input amplitude has to be positive or all else it wouldn’t produce the right result. I couldn’t really think of a solution to this problem because most waveforms treated have positive valued amplitude, A.

If someone showed me this program, I would first of all check if it works by trying various different phasors and waveforms. The general feedback would be positive.

This term has been a very new experience for me in terms of my courses and the whole Software Engineering course itself. This was my first time doing programming and so it was a bit of a challenge at first because I didn’t put in enough time into practising what I read and study but I later learnt and put in more effort and although I still have quite a way to go, I’m more committed. I did receive enough support from the lab assistants and the GTA’s. If anything, I have myself to blame because I didn’t put in enough time.

Hopefully if possible, I could extend my program to be able to add these phasors and waveforms. In general, this program on its own could be very useful in circuits when it comes to treating electrical components like capacitors and inductors. Since these components depend on frequency, they do not have just resistance but have a complex impedance which is like resistance for these components but vary with frequency of waveform. Therefore the program could also be used to output the complex impedance by using the formula to find out the values.

I enjoyed the fact that I was able to use knowledge from one course and transfer it to this course to write a program. For this course in general, I really enjoyed it. The fact that I had never done programming before really scared me but by reading the textbook and doing the exercise, I gained more knowledge and confidence on the basic aspects and was able to apply them in some of the exercises set. I am confident that with more practice and studying, I will be a more knowledgeable programmer.