Franco Canales

COP2800

Dr. Hien Nguyen

December 12, 2019

Reflection of COP2800 and Final Project

I barely passed the intro to C++ and was very pessimist about starting COP2800. I began to suffer from imposter syndrome. I understood some of the theory behind Object Oriented Programming in the C++ class but for some reason I would freeze when given a simple assignment. I just couldn’t connect theory into practice. So, I turned to the internet for more resources such as Code Academy and Udemy courses. Went through Tim Buchalka’s course on Java fundamentals with the objective of providing a framework for me to connect this class’s theory to practical use. And so, with that I went from very pessimistic, even before the semester started, to a very optimistic programming student. From the very first assignment of displaying a string to the last chapter on recursions, I was filling some of the holes from my earlier class and connecting theory to practice by analyzing the many code examples in the book.

Although this was an intro class, COP2800 was very in depth in introducing logic, reasoning, structure, and mathematic concepts that’s vital in one’s development to become a computer scientist. I was initially disappointed that Miami Dade College North Campus wasn’t offering this course, and so I had to take it through their online class. I was thinking that I would struggle with the concepts of this course because I had no easy access to my peers or my teacher, that I would be left behind. However, your syllabus and assignment structure helped force me to stay structured and study to meet the deadlines. In addition, the programming assignments was a great way to put what you learned from that unit into practice. Wasn’t too difficult nor too easy, overall a good tool to help prevent the common frustration one will go through when learning Java. One of my favorite topics in the book was the different types of loops and recursion functions because I’m very interested into becoming a Data Scientist or researching in A.I., and these fields require a logical mind to understand complex algorithms. So, understanding loops and recursive functions was a nice stepping stone to be able to read algorithms.

When the final project was announced, I glanced over the assignment and I initially thought it was going to be a difficult project. I was only focusing at the fact that there was a lot of variables and strict requirements for each function. However, as we got closer to the deadline, I began to really think about the assignment and it just all clicked. The way I understood this assignment was that it just needed an algorithm that can accept input and, through loops, can check the parameters as stated from the assignment. Basically, similar to how an ATM works but with the functionality to calculate the interest rate. 70% of the final project was pen and paper determining the public/private classes and initiating variables and creating a simple algorithm describing how this program will behave. Then the last 30% of the work was typing the syntax and testing/debugging. I would admit that the JavaFX is my weak point, but with all things, practice makes perfect and I intend to take with me the things I learned in this course and from the final project to my professional career.

In the beginning, I struggled with imposter syndrome and was struggling with remembering how to display a string. At the end of this course, I came out confident and optimistic about my programming career. If I could suggest a recommendation on improving this course, I would recommend expanding your peer-to-peer system.