George C. Wilborn || vvf555 || Aug: 2017-08-08

Report for Assignment 5:

I chose to try and create a modular design, as I have been taught to do, for the assigned program. It should have (given more time) would create a more optimal solution to our program. Aid and possible help from mentors and friends was very difficult to come by due to my schedule. My results are not complete as I did not have enough time to implement it to completion.

Initially I set out to build the frame work. Diagrammed the desired output and crafted UMLs for the assignment. I feel like my planning and actual design work is what pushed my assignment so far back. If I could re-do the project I would probably take less time to plan, doing no to little UML analysis and cutting back on the developer side of it. Probably to make a better use of my time I would ask my friends and colleagues for support and aid; however, being that I work late currently, scheduling meet-ups were not possible if at least extremely inconvenient for my friends in general. The UML and various mock-ups were in vain as my time limit came too soon.

When writing my code I always tried to create my functions to work with one another and always call to a common include file. I wanted the threads to function seamlessly and never fault. I went to *StackOverflow,* various universities, *YouTube, softpixel.com, geeksforgeeks.com, randu.org, Linux Academy,* and many other sights to try and understand thread more clearly. Semaphores and the actual multi-threading that I needed to understand was also sought after using google and the various sites in question. The most difficult to understand was the use of semaphores as they are very difficult to place correctly without just using trial and error. All in all reading of the book and paying closer attention to my notes and slides from class would have probably benefited me the same outcome. This was again a fault for why my program is not complete.

If completed I would have used semaphores to complete the threaded functions. Posting at the end of a thread and waiting at the beginning I would need about 3-4 to successfully function correctly. I projected optimal CPU and IO usage and would create an optimum result using these semaphores. I understand the usage of semaphores much better now because of the studies I needed to do in order to figure out a starting port for this project. I was also going to use the suggested doubly linked lists originally but could not figure out where to implement such a structure and by the time I had any idea of where it could possibly fit my time was close to up.

My skeletal structure of a program did not complete successfully and thus my results are NIL.