This project was easily the hardest one out of all of them this semester, but that is only to be expected. That being said, I believe I managed to do everything correctly.

One of the more challenging aspects of this project was trying to develop the algorithms needed to implement each of the class methods. I found it hard to turn the ideas in my head into code, and it still takes me a little while to understand how my code fully functions.

The ArrayList class was relatively straightforward, but I did come across some issues regarding the reallocation of the array, especially in the context of the insert functions. When the array is reallocated, any pointers to the elements in the array become invalid, which includes the pointer passed in as a parameter. Thus, I had to convert everything to their respective index form in order to be compatible across different arrays.

The NodeList class was a lot more challenging in that it required a lot more attention to dynamic memory management than the ArrayList class did. One major problem I came across while working on the project was trying to figure out how I could indirectly save and update a pointer variable. I noticed that some of my code looked the same, but functioned differently depending on whether a certain condition was met. (eg. if nullptr was given as an argument to insertAfter) My solution to this was to use double pointers, as they could save a certain pointer and change it without needing to search for it or binding a reference to it.