**Project 9 Documentation**

The overall purpose of the program was to help build my understanding of Queues for both arrays and nodes and to work through how these memory storages act when given memory. ArrayQueue is no different from any regular class function, no dynamic memory allocation needed for arrays. Constructors functioned exactly how any class constructors would work, the only difference would be the iteration through the array. The purpose of the arrayqueue class is to take a large array with a large amount of data and be able to read and edit certain parts of the array. For example, in this class, the array has a size 1000, which means it could hold 1000 different memory addresses. The ArrayQueue allows us to edit and view the array only from certain positions of the array, be able to push, pop, and set data to those specific areas of the array in which we want to edit.

NodeQueue works extremely similarly to the arrayqueue, but we are iterating through data with nodes instead of directly through an array. Instead of having data members of the class to directly edit, we have to edit the data through the helper node class and iterate between each data members through the dynamically allocated nodes which store the data given. NodeQueue has the same purpose of ArrayQueue, being able to edit and read certain parts of the data, but through nodes rather than array, because we are iterating through the data with nodes, there are no size limitations of the data we can store. So we allocate enough nodes the store and work with the data that we need, instead of editing parts of an already large array, we only edit parts of memory in which we need.

If I had more time on the project, I would definitely try to make the test driver able to test multiple things than the tests than I have on there, my program should be able to handle wrap around queue and maybe better implementation on the linear lists.