

## Brief Description How I tested my Structures

### **Stack Class**

- Starting with the stack class I had a pretty good understanding of what needed to be done because of the CS 199 class.
- I started with actually creating class Stack and then implementing the files
- I knew that I had to make an outline for the main function and decided to start creating that.
  - o Once that was finished I tested the add function to see if the values being added, would in fact, be added to the list.
    - I would add 3-4 values and then display the list. Everything was working properly.
      - At this point I knew the add function was the same for the Queue structure so I went ahead and added that there.
- The remove function threw me for some loops however. I was returning the values of the nodes being deleted but for some reason that was throwing off my output and not actually removing the node.
  - o Once I implemented an integer value at the beginning of the program that would hold the value of the node being deleted, everything worked.
    - This did take me about 45 – 1 hour of testing however.
      - At first I thought it could be my print function as well.

### **Queue Class**

- Since I had everything pretty well set up with the stack class all I had to really test was the Queue class.
  - o I knew that I had a pointer to the first Node (head) and that I did not need to loop to any sort of new value.
    - I created a new Node that would hold the place of the head variable and then set the head to the next variable.
    - When testing this, I was pleasantly surprised to have it working almost instantly after implementation.

### **Overall Testing Thoughts**

The most trouble that I had with was the remove for each function. I was definitely overthinking the Queue remove function for the longest time and thinking that it was overly complicated. Once I understood that I had a pointer at the head, I knew that the beginning node could be accessed very easily.

The biggest problems I think was finding where my errors actually took place. Once I knew where the error was, it was easy to fix the problems.