Jia Li

Project 9: Stack and Queue

For this project, I was able to learn a lot about Stack and Queue principles. In addition, I was able to use the “cout” approach taught by Sonu Jose to further enhance my debugging skills dealing with bugs that involve minor details regarding tightly-coupled functions and member functions of classes. Finally, I learn providing accurate test cases are extremely important in order to ensure that all your classes and their objects are working correctly.

However, I should have test my code as I go along to ensure that the time I spent on debugging is minimal as possible as other bugs have been abstracted with the program being constant debugged at all times. When I don’t do so, I run into a much more variety range of errors that cause me spend a majority of my time debugging the program rather than completing the task at hand. Yet, I was able to have a good picture of program by reading the project specification more than once. In the end, I learn that for a complex program to be able to run, it is necessary that the program to be robust and free of error.

To do so, I had to thoroughly write explicit function calls in my test driver to ensure that all functions are performing their right tasks to prevent further mistakes. In addition, I must ensure that the test driver I created is modular, meaning that one fault in parts of the program does not affect another part of the program. To do all that, I must ensure that minor bugs in other classes are abstracted with repetitive debugging. Finally, the “cout” approach taught by Sonu Jose was especially useful when it comes to debugging errors and incorrect results arise from the copy constructors in both the stack and the queue. Overall, this project is a good learning experience and I hope to experience more of this in the future.