## VII. Results Summary of Deliverables

Record all defects found in the specification inspection, the design process, the design walkthrough, the coding process, and testing. Also, write a short (half-page or less) analysis of what you can learn about your development and testing strategy.

There were many defects when I started doing this assignment, I originally started thinking of it in C++ format using strings rather than character arrays, which forced me to shift to thinking of it in character arrays to try and be like how lisp handles io, with cells and atoms.

The design process/walkthrough, coding process and testing was fine, most of the issues I had were with syntax, making sure I was writing the logic correctly for lisp and not the way it’s written in C/C++. I learned many things about development and testing strategies.

Regarding testing strategy, it is helpful to create different test cases, such as using racecar as a test in palindrome, to prove the logic works normally, and test cases that try to exploit decisions, if possible, for example, if testing palindrome, writing an input that will test all of the decisions.

What I learned about my development is that I find the most success when creating pseudocode in C/C++ that is loosely like how it would be implemented in the programming language that it is going to be converted to.

Creating picture diagrams to represent the input quote/numbers/expression(s) and the manipulation of those through the pseudocode written in C/C++ to test if the logic works as intended is another way, I found success through the development.

One other way I found success in my development is bouncing ideas off of others on how to create the logic for problem(s) and debating on how edge cases would be handled and translating that debate into pseudocode regardless of if it’s in C/C++ or plain English.