Project #: 1
Semester: 2
Name: Sam DeCook

I. Requirements: Restate the problem specification and any detailed requirements in your own words.

Take in a string and output it with the characters between the specified front and rear index reversed.

II. Design: How did you attack the problem? What choices did you make in your design, and why? Show class diagrams for more complex designs.

This was a simple project, so I first wrote the code to check the indices inputted then created pointers to the characters at those indices. Finally, I wrote the reverse function and finished the project.

III. Security Analysis: State the potential security vulnerabilities of your design. How could these vulnerabilities be exploited by an adversary? What would be the impact if the vulnerability was exploited?

There wasn't anything checking the input before running the code, so someone could potentially inject malicious code and compromise the computer.

IV. Implementation: Outline any interesting implementation details in your solution.

The implementation was mostly straightforward. It was interesting to me how we were able to move the pointers and change the characters without having the string as one of the parameters., I had never done that in code before.

V. Testing: Explain how you tested your program, enumerating the tests if possible. Explain why your test set was sufficient to believe that the software is working properly, i.e., what were the range of errors for which you were testing.

I wasn't able to satisfactorily test my program. In my terminal, I was able to check for warnings and errors, but never truly run it. The only way I could run it was through ZyBooks, which was really slow. However, I do believe my code is working properly; the indices where checked to be valid, and the reverse function is robust, at least to my eyes.

VI. Summary/Conclusion: Present your results. Did it work properly? Are there any limitations? NOTE: If it is an analysis-type project, this section may be significantly longer than for a simple implementation-type project.

The program correctly reverses the characters specified and works properly.

VII. Lessons Learned: List any lessons learned. For example, what might you have done differently if you were going to solve this problem again?

Don’t overestimate the difficulty of a problem, that can lead to overcomplicating the code.