

Isaiah Avila

Professor Onwuka

COMPSCI-2

12 May 2025

Final Report

For my final project, I sought to try my hand at an encryption algorithm. I was inspired by our second discussion of the course ‘Discussion: Streams’, where the class was asked “What are encryption algorithms?” You could see from my response then that I was excited to discuss the topic. For my algorithm, I essentially took a very old (yet reliable) method of encryption, the Vigenere Cipher, and mimicked its properties through C++17, through the use of inheritance, smart pointers, classes, and streams. The premise of the Vigenere Cipher is that a ‘key’ is agreed upon by two (or more) parties. Each letter in a key corresponds to a numerical value between 0 and 25, representing the 26 letters in the English alphabet. So let’s say you and I coded our messages with the key ‘Apple.’ That would mean ‘A = 0’ (don’t alter the first letter), P = 15 (shift the second letter forward 15 letters in the alphabet), and so on. After ‘e’, you’d simply continue encrypting starting over from ‘A’. So ‘Hello World’ now becomes ‘Khoor Zruog.’ Decrypting is a simple process; you merely use the key, and *shift backwards* based on the letter’s value. I was able to accomplish this by opening an input file in binary mode, then streaming its entire contents into an ostringstream and then extracting a single string of text holding the whole file. Then, the program compares the first argument (mode) to decide whether to call `encrypt(text)` or `decrypt(text)` on the cipher object. Then the program loops over every byte in the input string, and iterates over every character in the text, one character at a time. Finally, we open the output file in binary mode, write the entire transformed string in one shot, and close the

file.

I think the hardest part of this project was seeking out how to use destructors, as I don't believe we've covered it before. Memory deallocation was also something that I struggled to implement correctly, and char pointers were also difficult to use here, despite being so vital for the project's feasibility. The number one thing that I learned here was the importance of inheritance. Virtual functions saved me from writing out several lines of code that would have either created issues with the programs logic, or at least drastically extend the amount of time I needed to complete this project. I had fun, professor! Thank you for everything and have a great summer!