**Report**

My thoughts on this program was that this was a tough program. I worked and worked on trying to figure out how to decode this. I searched Google every night and even tried online decrypts and they didn’t even work. I went to the tutor and worked on it for two hours and not even the tutors knew how to do the program. I asked a friend that had already taken Python and he couldn’t help me either. I don’t know what else I can do to figure out how to get this program to run. I did, however, get something to work so the decrypt executed English letters, but it was only one letter per decrypt, and I couldn’t figure out why it was executing that way. I did this by looking up how to brute force a Caesar Cipher and went from there. This was by far the hardest, most frustrating program I’ve ever attempted. It was nice to have something that difficult to work on, but it was so problematic that not even the tutors knew what was going on.

**Algorithm**

Define program module

Open encrypted txt file

Write new decrypted txt file

Set iFile to userString and read lines

Import from sys library

Create distance equal to integer of nothing

Create plaintext value set to an empty string

Make a For loop to find characters in userString

Create ordValue equal to the ordinal of the character in userString

Create cipherValue equal to the difference of ordValue and distance

Make If statement to find if the cipherValue is less than the ordinal of ‘a’

Sets cipherValue to the ordinal of ‘z’ minus the distance and ordinal of ‘a’, then minus the ordValue plus 1 to shift the character

Make plaintext add/equal to character in the cipherValue

Make If statement to find if the cipherValue is greater than the ordinal of ‘z’

Sets cipherValue to the ordinal of ‘a’ plus the distance minus ordinal of ‘a’, then minus the ordValue plus 1 to shift the character

Set userString equal to the string-character of ordValue

Use the sys we imported to put every character from userString onto one line instead of all on separate lines

Create offsetValue

Once the count reaches -26 stop, hammer time

Set decryptIndex value to 0

Create While loop to run if decryptIndex is less than the length of userString

Create c equal to the ordinal of userString in decryptIndex plus the offsetValue

Create If statement to find if c variable is less than 0, then c is added/equal to 128

Set decryptChr equal to character of c

Set decryptString and decryptChr added/equal to each other

Add/equal decryptIndex and 1

Subtract/equal offsetValue and 1

Print decryptedString plus newline

Write decryptedString to new file

Close new file

Define main module

Run program() inside of main module

Run main()

**TROUBLESHOOTING:**

Pretty much the entire thing