Python In-class Programming Assignment
Substitution Cipher

This directory contains a template Python file (`subCipherDec.py`) and an encrypted version of the *Gettysburg Address* (`gettysburg.enc`). Here's how the file decryption works:

1. Start by building a string with all the ASCII characters from `0` to `127`.

2. Every three characters in the encrypted file represent the ordinal position of an ASCII character in the range from `0` to `127`.

3. Read three characters at a time from the file, cast that to an `int`, and use that as an index on the string you created in #1 to get an individual character.

4. Take the character you found in #3 and add (concatenate) that to another string that represents the plaintext of the *Gettysburg Address*.

5. Keep looping until you reach the end of the encrypted file. Once you're done, print the variable holding the plaintext *Gettysburg Address* to the screen.

Hints:

a. Every character in the plaintext is represented by exactly three characters in the encrypted file (padded with zeros). You're guaranteed that the total length of the encrypted file is a multiple of three.

b. Whenever you read from the encrypted file, you get a string datatype back. Even if it looks like a number ("`032`") it's actually a string. You need to cast that to an int to use it as an integer.

c. I recommend getting a hard-copy version of the ASCII chart and keeping it handy. There's one available here: `~/repo201/referencelib/asciiChart.pdf`