Decrypting Book 1

To decipher the encoded book took a few different steps. For starters, we had to determine the cipher that was applied to the document. In our case, this was an LCG cipher with UTF-8 encoding added. We also needed to determine a crib for the document, " Project Gutenburg" in this instance.

Since it is know that LCG was used to encoded the document, we would need to find an a, b, and m such that **x1 = (a*x0) + b (mod m).**

From this formula, we can deduce *(x2-x1)^2 – (x3-x2)(x1-x0)* ≅ 0 *mod M*, where *m* is a divisor of *M*, using a system of equations.

With this *M*, we can find all the possible divisors that could possibly be *m* in our equation. In my solution, I iterate through all possible values of *m*, trying to find a corresponding *a, b* that will generate encoded values – (UTF 8 encoding).

To find a possible *a,* first check to make sure *a* is invertible. If *a* is invertible, a possible value can be found by taking the extended Euclid of *(x2-x1)* and our possible *m*.

After finding a possible *a,* we can use the equation *(x2-a*x1) % m.* We then compare this value with our lcgEncoded values. If these values match, we have found an *a, b, m* that suffice for our LCG.

To decrypt the document, we take our corresponding *a, b, m* and generate the lcg values. We subtract our encoded values from this to get the raw UTF 8 characters, which can be outputted to their corresponding ASCII characters.

Substitution Transposition Cipher

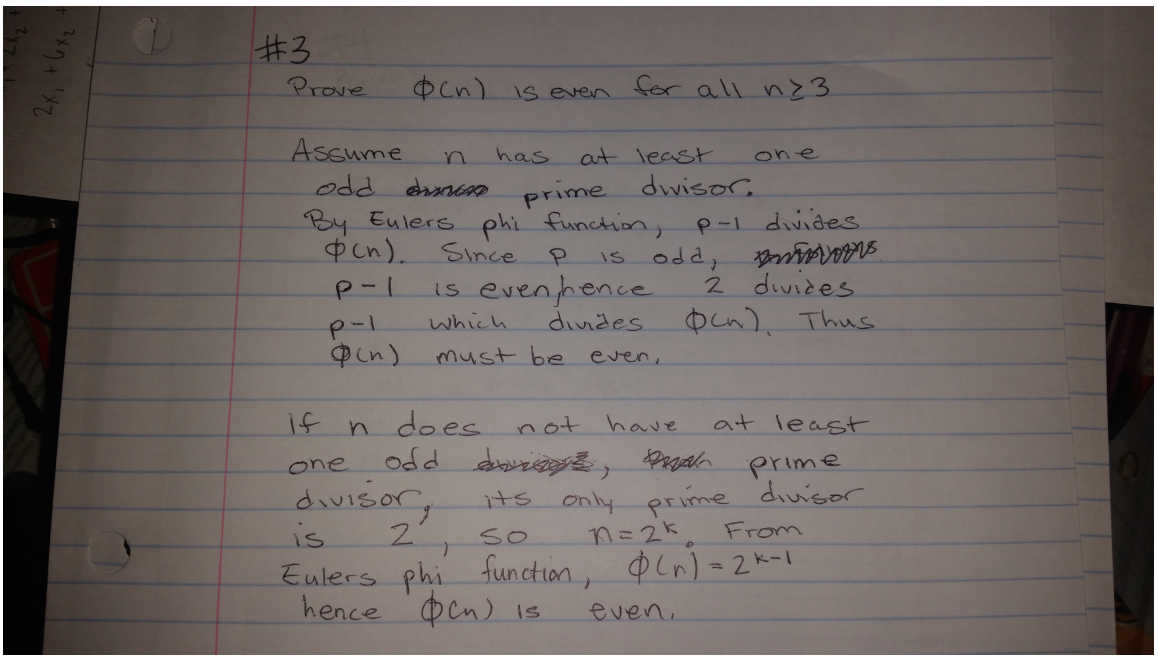
To decipher the substitution columnar transposition cipher, one needs to first understand the two different ciphers separately.

With a mono-alphabetic cipher, we know each letter maps one-to-one to another character. With a columnar transposition cipher, we know the column length can be found by dividing the message length by the guessed keyword length.

To begin deciphering this, we first should create a frequency distribution of the characters within the document. From this, we found *s* and *q* to be the most occurring letter. With this, we can compare with a frequency distribution of characters occurring in the English language.

After completing the possible substitution, we must place the message into a column like structure and begin guessing at the length and the arrangement of the columns within the transposition matrix.

To better understand the columnar transposition, I created a program to simulate the encryption process. With this, I was able to see how the cipher transforms text into outputs similar to those found within our document.

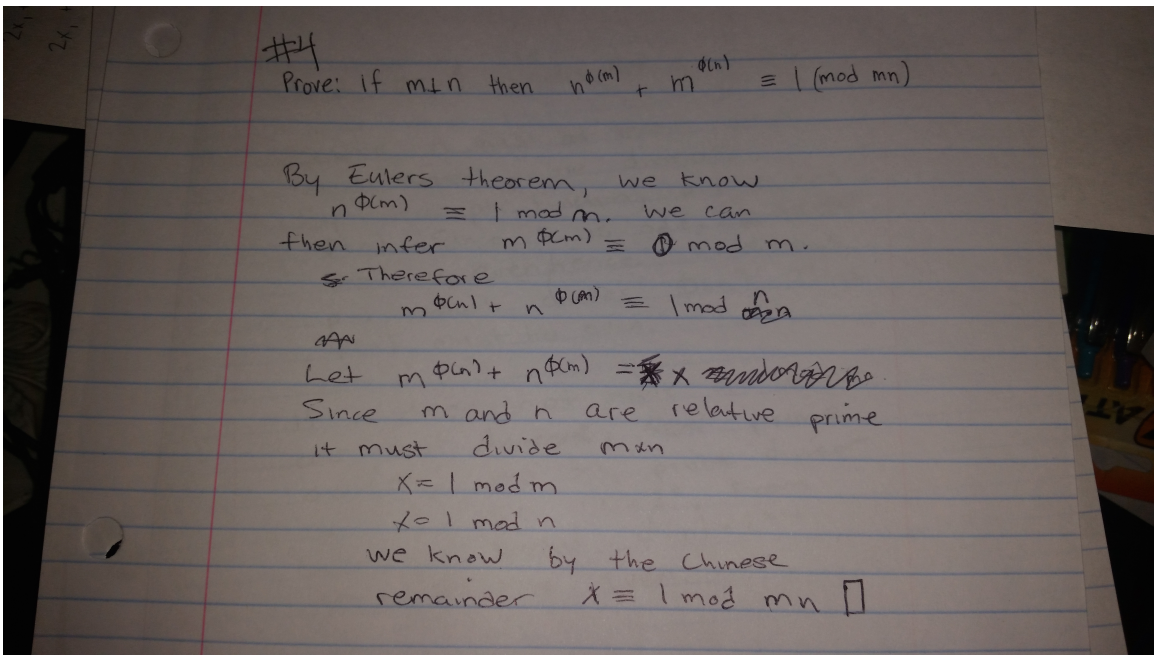


#3

Prove $\phi(n)$ is even for all $n \geq 3$

Assume n has at least one odd prime divisor. By Eulers phi function, $p-1$ divides $\phi(n)$. Since p is odd, $p-1$ is even, hence 2 divides $p-1$ which divides $\phi(n)$. Thus $\phi(n)$ must be even.

if n does not have at least one odd prime divisor, its only prime divisor is 2, so $n = 2^k$. From Eulers phi function, $\phi(n) = 2^{k-1}$ hence $\phi(n)$ is even.



#4
Prove: if $m \perp n$ then $n^{\phi(m)} + m^{\phi(n)} \equiv 1 \pmod{mn}$

By Eulers theorem, we know $n^{\phi(m)} \equiv 1 \bmod m$. we can then infer $m^{\phi(m)} \equiv 0 \bmod m$.

∴ Therefore

$$m^{\phi(n)} + n^{\phi(m)} \equiv 1 \bmod n$$

Let $m^{\phi(n)} + n^{\phi(m)} = x$

Since m and n are relative prime it must divide $m \cdot n$

$x = 1 \bmod m$

$x = 1 \bmod n$

we know by the Chinese remainder $x \equiv 1 \bmod mn$ □