Affine Ciphers Problem Set

Questions 1-4 assume an alphabet size of m=26.

1. If plaintext is encoded into ciphertext as y=Ax + B, with A=19 and B=-5, what is the inverse transformation? **y=11x+3**
2. if plaintext is encoded into string1 with A=11 and B=13, then string1 is encoded into string2 with A=15 and B=19
   1. Is there one affine transformation that describes plantext🡪string2? **Yes**
   2. If so, write that transformation as y = Ax + B and its inverse as x = Cy + D **y=9x+6, inverse: x=3y+8**
   3. If not, write some equation that describes the compound transformation **N/A**
3. Based on your answer to number 2, are compound affine transformations more secure than single affine transformations? **No**
4. Using a 26 letter alphabet, how many values of A in y = Ax + B are allowed? How many values of B? How many total transformations are therefore possible?

**12 for A, 26 for B, 312 for total transformations**

1. Devise a new alphabet of your own choosing. Include at least numbers and digits, perhaps come punctutation, lowercase, etc.
   1. How many characters are in your alphabet **31 (,.’?/)**
   2. How many A values are allowed in transformations **30**
   3. How many B values? **31**
   4. And what is the total number of transformations possible? **930**
2. Using Python, encode a message in your new alphabet. Write the message and its encoded text here (or attach a printout), along with the transformation and its inverse.

**HOPEFULLY,THISWILLWORK.**

**OSX/ERDDGQMOTH.TDD.SC?V**

**Transformation: y=5x+10(mod31)**

**Inverse: y=25x+29(mod31)**

1. Write a short Python script using your library functions to print out a table with the following columns: {alphabet size, number of A values possible, number of B values possible, total number of transformations}

**(31, 30, 31, 930)**

1. Based on your results in question 7, discuss how you can change you alphabet in question 5 to make it more secure. Would you make it larger or smaller? Why? And how? **Yes, it would be harder, because there would be more possibilities to try to crack and less numbers relatively prime.**
2. Digraph encoding of alphabets begin by encoding pairs of letters into single integers in the same way you convert from another base to base 10. For example if H = 8 and I = 9, then the digraph HI would become 8\*26 + 9 = 213, in a 26 letter alphabet. Complete the methods in class Cryptoalphabet to perform digraph encoding and decoding of pairs of letters.

**Complete**

1. Add functions affine\_encode\_digraphs(plaintext, a, b) and affine\_decode\_digraphs(ciphertext, a, b) to your file affineCodes.py. Encode a message and include it and its encryption here. You may pad with a terminating X if the plaintext requires it.

**Message: Hopefully this will work**

**Code: BGCOOWCLRPZQCKHLFZB**

**y = 7x + 12 (mod676)**