CSC 1300 – Problem Set 5

1. Prove or find a counterexample for each of the following statements about congruences. Assume a, b, c, d, and n are integers with n ≥ 2.
   a) If \( a \equiv b \pmod{n} \) and \( b \equiv c \pmod{n} \) then \( a \equiv c \pmod{n} \)
   b) If \( ac \equiv bc \pmod{n} \), then \( a \equiv b \pmod{n} \)
   c) If \( a \equiv b \pmod{n} \) and \( c \equiv d \pmod{n} \), then \( ab \equiv cd \pmod{n} \)
   d) If \( a \equiv b \pmod{n} \) and \( c \equiv d \pmod{n} \), then \( a - c \equiv b - d \pmod{n} \).
   e) If \( a \equiv b \pmod{n} \) and \( c \equiv d \pmod{n} \), then \( a^c \equiv b^d \pmod{n} \).

2. Encrypt "IFCATSCOULDTEXTYOUTHEYWOULDNT" using a shift cipher with shift 5.

3. An exercise in RSA coding
   In RSA one uses very large primes, but for the sake of this exercise, we will use small primes.
   Let \( p=9883 \) and \( q=7901 \). Then \( N = pq = 78085583 \). (Of course, in our example we will use decimal but the computer would use binary.)
   Your bank will use this \( N \), and it will choose an encoding key \( e = 12345677 \). It tells all its customers this \( N \) and \( e \). Since the bank knows \( p \) and \( q \), they could now calculate the decoding key \( d = 38273813 \).
   We now have a secret message we want to send the bank: **Send Alice $100**
   ASCII code is a standard way to translate letters into numbers. See the Wikipedia article or a variety of websites such as https://www.cs.drexel.edu/~introcs/Fa11/notes/10.1_Cryptography/ASCII.html
   Note that we must "pad" the numbers so each letter is given by three digits, so A is 065 and a blank space is 032 and $ is 036.
   a) Translate your message into standard ASCII.
   We break the message into chunks smaller than \( N \), in our case, into 8 digit chunks. For instance, from Exercise 1, you know the third chunk is \( M = 65108105 \).
   We now find the encoded message \( E \) which is \( M^e \pmod{N} \). To do the modular arithmetic, use a website such as http://comnuan.com/cmnn02/cmnn02008/cmnn02008.php or http://ptrow.com/perl/calculator_bigint.pl.
   b) Find \( E \) for our values of \( N \), \( e \) for each of the six chunks \( M \). For instance, the first chunk of the encoded message should be \( E1 = 76068910 \).
   The bank can calculate the original message using \( M = E^d \pmod{N} \).
   c) Find the original ASCII code for the six chunks of the message by decoding each \( E \).
   d) Convert these decoded chunks from ASCII back into regular text.
   It turns out that mod arithmetic "scrambles" numbers quite well, so even a tiny change will totally ruin the decoded message.
   e) Take your six chunks of encoded message, and add 1 to each chunk (so the first chunk is now 76068911, etc.) Now decode the six chunks and translate into ASCII.
   You should get total nonsense.

4. At the end of the class powerpoint, we briefly touched on ethical issues (ethical issues pervade computer science!) Write a three or four sentence reflection on the question “Is it desirable to have government able to crack all codes?”

Exercises (3,4) written by Dr. Styer.