USS: Introduction to mathematical cryptography
Tuesday August 2 problems

A (relatively) simple decryption circuit

Recall the simple cipher we studied in class last week on Tuesday. The truth is that it is only somewhat homomorphic and unable to evaluate its own decryption circuit. In the article where the cipher is introduced, the authors must first “squash” the decryption circuit by modifying the key generation, encryption, and decryption algorithms. I recommend reading this but we will not cover it in our course.

Instead, we will think deeply of the decryption circuit for the original simple cipher to see what is involved in creating a circuit. Recall that the decryption algorithm is

$$m \equiv r_p(c) \pmod 2.$$

Throughout assume that you are given the ciphertext c as a string of bits, and the prime p as a string of bits. For simplicity you can assume that p is 3 bits long and c is 6 bits long.

1. Suppose that you knew the value of $r_p(c)$, given in bits. Show/convince yourself that the value of m is the least significant bit of $r_p(c)$.

2. We will compute $r_p(c)$ in the following manner:

   - First compute the value $\frac{1}{p}$ in bits to some amount of accuracy. For simplicity you can assume that you must compute $\frac{1}{p}$ up to 10 bits past the decimal point.
   - Then compute the fractional part of $c \times \frac{1}{p}$, in bits. Record the most significant bit for later.
   - Take the fractional part and multiply by p. Round to the nearest integer. This is the remainder with $0 \leq r < p$.
   - To get $r_p(c)$, add the most significant bit of the fractional part of $c \times \frac{1}{p}$ to the least significant bit of the remainder you just obtained.

   To begin, perform this algorithm, using first digits if you need, and then bits with $p = 7$ and $c = 13$.

3. Now **imagine** how you would describe each operation above in more detail explaining what to do to each bit of p and c to compute the various quantities, using only addition and multiplication. These are the operations you would perform on the **ciphertexts** (which are integers!) to perform the decryption circuit homomorphically. Imagine what happens!

Practicing PLWE

For these problems, we will use the field $K = \mathbb{Q}(\gamma)$, for γ a root of the irreducible polynomial $x^3 - x^2 + 1$. $\mathcal{O}_K$ is monogenic and generated by γ. Let $q = 17$.

1. Generate, by hand, a private and public key for PLWE as described in class.
2. Encrypt a message with 3 bits.
3. Decrypt the ciphertext you produced.
4. Notice what happens to the polynomial coefficients when you multiply two elements of $\mathcal{O}_K$: they grow. The rate at which they grow is called the "expansion factor" of the polynomial $x^3 - x^2 + 1$.