A corollary of Bezout’s theorem

Last time we proved

**Bezout’s Theorem:** Let \( a, b \) and \( c \) be integers, \( (a, b) \neq (0, 0) \). The equation \( ax + by = c \) has an integer solution \( x, y \) if and only if \( c \) is divisible by \( \gcd(a, b) \).

The case, where \( \gcd(a, b) = 1 \) is particularly interesting. If this happens, we say that \( a \) and \( b \) are *relatively prime*. In this case Bezout’s theorem says that \( ax + by = c \) has an integer solution for every \( c \). Here is an important corollary.

**Corollary:** Suppose \( a \) and \( b \) are relatively prime. If \( a \) divides the product \( bk \) for some integer \( k \), then \( a \) divides \( k \).

**Proof:** Choose integers \( x \) and \( y \) suuch that \( ax + by = 1 \). Multiply both sides by \( k \):

\[
a(kx) + (bk)y = k.
\]

Both terms on the left are divisible by \( a \). Hence, their sum \( k \) is also divisible by \( a \). \( \square \)

**Multiplicative inverses mod \( n \)**

Recall that \( a \) has a multiplicative inverse in \( \mathbb{Z}_n \) if and only if \( \gcd(a, n) = 1 \). In particular, if \( n = p \) is a prime, then every non-zero element of \( \mathbb{Z}_n \) has an inverse. This means that \( \mathbb{Z}_p \) is a field with \( p \) elements. This is the field \( GF(p) \) discovered by Galois.

Note that \( \mathbb{Z}_{p^r} \) is not a field for any \( r \geq 2 \). There is a field \( GF(p^r) \) with \( p^r \) elements, but it is *not* \( \mathbb{Z}_{p^r} \).

The Euclidean algorithm allows us to compute the multiplicative inverse of any \( a \neq 0 \) in \( \mathbb{Z}_p \). To do this, we solve \( ax + nb = 1 \).
Example: What is $\frac{11}{21}$ in $\mathbb{Z}_{41}$? Note that 41 is a prime.

First let us compute $21^{-1}$ in $\mathbb{Z}_{41}$, then multiply by 11. To compute $21^{-1}$, we need to find an integer solution of $21x + 41y = 1$.

Apply the Euclidean algorithm to 41 and 21.

$41 = 1 \cdot 21 + 20$
$21 = 20 + 1$
$20 = 20 \cdot 1 + 0$.

Back substitution: $1 = 21 - 20 = 21 - (41 - 21) = 2 \cdot 21 + (-1) \cdot 41$.

Thus $21^{-1} \equiv 2 \pmod{41}$, and

$$\frac{11}{21} = 11 \cdot 21^{-1} \equiv 1 \cdot 2 \equiv 2 \pmod{41}.$$ 

Thus $\frac{11}{21} = 22$ in $\mathbb{Z}_{41}$.

As a first application of modular arithmetic to coding theory, we will now discuss two codes based on the idea of “parity check”.

Proposition: $A_q(n, 2) = q^{n-1}$.

Proof: By Singleton, $A_q(n, 2) \leq q^{n-1}$. To prove equality, we need to construct a $q$-ary $(n, q^{n-1}, 2)$-code. We have previously done this for $q = 2$, using the idea of “parity check”. Now we use a similar construction with $F_q = \mathbb{Z}_q$.

We define our code as follows. Start with a code $F_q^{n-1}$ consisting of every word of length $n - 1$. To each word, $(a_1, \ldots, a_{n-1})$, append $a_n$ in the last position, so that $a_1 + \cdots + a_n = 0$ in $\mathbb{Z}_q$. In other words,

$$a_n \equiv -a_1 - a_2 - \cdots - a_{n-1} \equiv 0 \pmod{q}.$$ 

The resulting $q$-ary code $C$ has $q^{n-1}$ words. It remains to check that $d(C) = 2$. 

2
Clearly $d(C) \leq 2$, since $(0, 0, 0, \ldots, 0)$ and $(1, n - 1, 0, \ldots, 0)$ both lie in $C$ and are at distance 2. On the other hand, since any two words in $C$ disagree somewhere in the first $n - 1$ positions, $d(C) \geq 1$. It thus remains that $d(C) \neq 1$. Indeed, suppose $\overline{a} = (a_1, \ldots, a_n)$ and $\overline{b} = (b_1, \ldots, b_n)$ are distinct codewords in $C$ such that $d(\overline{a}, \overline{b}) = 1$. Let us say, they disagree in position $i$ and agree in every other position. Subtracting

$$a_1 + \ldots + a_n \equiv 0 \pmod{q}$$

from

$$b_1 + \ldots + b_n \equiv 0, \quad (\text{mod } q)$$

and remembering that $a_j = b_j$ for any $j \neq i$, we obtain, $a_i - b_i \equiv 0 \pmod{q}$. In other words, $a_i = b_i$ in $\mathbb{Z}_q$, so that in fact, $\overline{a} = \overline{b}$, a contradiction. \hfill \Box

**The ISBN Code**

Here we take $q = 11$. Alphabet: $F_{11} = \mathbb{Z}_{11} = \{0, 1, \ldots, 9, X\}$ (write $X$ in place of 10). ISBN stands for “International Standard Book Number”.

The ISBN code is a code of length 10 consisting of words $(a_1, \ldots, a_{10})$ such that

$$a_1 + 2a_2 + \cdots + 10a_{10} \equiv 0 \pmod{11}.$$ 

This code has minimum distance 2.

The error detection algorithm for this code is as follows. If $\overline{a} = (a_1, \ldots, a_{10}$ is received, compute the sum

$$s \equiv a_1 + 2a_1 + \cdots + 10a_{10}.$$ 

If $s = 0$, then $\overline{a}$ is in $C$. Assume no error occurred. If $s \neq 0$, declare an error.
The ISBN can detect a single error in any position and any “transposition error”, where two digits are transposed. For proofs of these facts, see p. 37 in the book.

The ISBN code also has the property that any digit in a codeword can be recovered from the other nine. For example, if 

$$\overline{a} = (1 \: 2 \: 2, \: 0, \: x, \: 0, \: 3, \: 3, \: 3, \: X)$$

is an ISBN, then the missing digit \(x\) can be recovered by solving

$$1 \cdot 1 + 2 \cdot 2 + 3 \cdot 2 + 4 \cdot 0 + 5x + 6 \cdot 0 + 7 \cdot 3 + 8 \cdot 3 + 9 \cdot 3 + 10 \cdot 10 \equiv 0 \pmod{11}.$$ 

This simplifies to

$$1 + 4 + 6 + 5x + 21 + 24 + 27 + 100 \equiv 0 \pmod{11}$$

or \(5x + 7 \equiv 0 \pmod{11}\). Since \(7 \equiv -4 \pmod{11}\), we see that \(x = \frac{4}{5} \pmod{11}\). Using the Euclidean algorithm (or just finding a solution of \(5x + 11y = 1\) by trial and error), we see that \(5^{-1} \equiv 9 \pmod{11}\) and thus

$$x \equiv 4 \cdot 9 \equiv 3 \pmod{11}.$$ 

Thus the missing digit is 3.