

Linear Cyclic Codes

♠ **Polynomial and Words.** A polynomial of degree n over \mathbb{K} is a polynomial

$$p(x) = a_0 + a_1x + \cdots + a_{n-1}x^{n-1} + a_nx^n,$$

where the coefficients $a_0, a_1, a_2, \dots, a_n$ are elements of \mathbb{K} with $a_n = 1$. The set of all polynomials $p(x)$ over \mathbb{K} is denoted by $\mathbb{K}[x]$. Elements of $\mathbb{K}[x]$ are added and multiplied in the usual fashion except that since $1 + 1 = 0$, we have $x^k + x^k = 0$. This implies that

$$p(x)^2 = (a_0 + a_1x + \cdots + a_{n-1}x^{n-1} + a_nx^n)^2 = a_0 + a_1x^2 + \cdots + a_{n-1}x^{2n-2} + a_nx^{2n}.$$

The polynomial $p(x) = a_0 + a_1x + \cdots + a_{n-1}x^{n-1}$ of degree at most $n - 1$ over \mathbb{K} may be regarded in general as the word $a_0a_1 \dots a_{n-1}$ of length n in \mathbb{K}^n . For example in $n = 7$,

Polynomial	Word
$1 + x + x^4$	1100100
$1 + x^4 + x^5 + x^6$	1000111
$1 + x + x^3$	1101000

If $v \in \mathbb{K}^n$, then we denote by $v(x)$ its corresponding polynomial. For example, if $v = 00101111$, then $v(x) = x^2 + x^4 + x^5 + x^6 + x^7$.

♠ **Division Algorithm.** Let $f(x)$ and $h(x)$ be in $\mathbb{K}[x]$ with $h(x) \neq 0$. Then there exists unique polynomials $q(x)$ and $r(x)$ in $\mathbb{K}[x]$ such that

$$f(x) = q(x)h(x) + r(x),$$

where $r(x) = 0$ or $\text{degree } r(x) \leq \text{degree } h(x)$.

The polynomial $q(x)$ is called the quotient, and $r(x)$ is called the remainder. The procedure for finding the quotient and the remainder when $f(x)$ is divided into $h(x)$ is the familiar long division process, but with the arithmetic in \mathbb{K} among the coefficients. Also note that since $p(x) = -p(x)$, if $f(x) = q(x)h(x) + r(x)$, then $r(x) = q(x)h(x) + f(x)$.

We say that $f(x)$ modulo $h(x)$ is $r(x)$ if $r(x)$ is the remainder when $f(x)$ is divided by $h(x)$; we shall write $f(x) \equiv g(x) \pmod{h(x)}$ if and only if they have the same remainder when divided by $h(x)$; that is if $[f(x) - g(x)]$ is divisible by $h(x)$.

Theorem 1. If $f(x) \equiv g(x) \pmod{h(x)}$ then for any $p(x)$,

$$f(x) + p(x) \equiv g(x) + p(x) \pmod{h(x)}$$

and

$$f(x)p(x) \equiv g(x)p(x) \pmod{h(x)}.$$

The proof follows from the fact that $f(x) - g(x) \equiv 0 \pmod{h(x)}$.

A polynomial $p(x)$ of degree greater than one is called reducible, if it is divisible by a polynomial of degree at least one; otherwise we say $p(x)$ is irreducible. In general $1+x$ is a factor of $f(x)$ if and only if 1 is a root of $f(x)$; that is, if and only if $f(1) = 0$. Similarly x is a factor of $g(x)$ if and only if $g(0) = 0$. However finding other irreducible factors of a polynomial is more difficult and at this point is simply a matter of trial and error. For $n \geq 2$, the polynomials $p_n(x) = 1+x^n$ are reducible. These polynomials are important tools for our investigation. Since $1+x^{2n} = (1+x^n)^2$, we only need to find the factors of $1+x^n$, where n is odd. Here is a list of all the factors of $x^n + 1$ for $3 \leq n \leq 31$, n odd.

Factorization of $1 + x^n$	
n	Factorization
3	$(1+x)(1+x+x^2)$
5	$(1+x)(1+x+x^2+x^3+x^4)$
7	$(1+x)(1+x+x^3)(1+x^2+x^3)$
9	$(1+x)(1+x+x^2)(1+x^3+x^6)$
11	$(1+x)(1+x+\dots+x^9+x^{10})$
13	$(1+x)(1+x+\dots+x^{11}+x^{12})$
15	$(1+x)(1+x+x^2)(1+x+x^4)(1+x+x^2+x^3+x^4)(1+x^3+x^4)$
17	$(1+x)(1+x+x^2+x^4+x^6+x^7+x^8)(1+x^3+x^4+x^5+x^8)$
19	$(1+x)(1+x+\dots+x^{17}+x^{18})$
21	$(1+x)(1+x+x^2)(1+x^2+x^3)(1+x+x^3)$
	$(1+x^2+x^4+x^5+x^6)(1+x+x^2+x^4+x^6)$
23	$(1+x)(1+x+x^5+x^6+x^7+x^9+x^{11})$
	$(1+x^2+x^4+x^5+x^6+x^{10}+x^{11})$
25	$(1+x)(1+x+x^2+x^3+x^4)(1+x^5+x^{10}+x^{15}+x^{20})$
27	$(1+x)(1+x+x^2)(1+x^3+x^6)(1+x^9+x^{18})$
29	$(1+x)(1+x+\dots+x^{27}+x^{28})$
31	$(1+x)(1+x^2+x^5)(1+x^3+x^5)(1+x+x^2+x^3+x^5)$
	$(1+x+x^2+x^4+x^5)(1+x+x^3+x^4+x^5)(1+x^2+x^3+x^4+x^5)$

Idempotent Polynomials. We denote by R_n , the ring of all polynomials modulo $1+x^n$. A polynomial $I(x) \in R_n$ is called idempotent, if $I(x)^2 = I(x) \pmod{1+x^n}$. For example, $x^3 + x^6 \pmod{1+x^9}$ is an idempotent. Note that for any n , $I(x) = 1$ is an idempotent in R_n .

It is easy to show that if $u(x)$ and $v(x)$ are idempotents, so is their sum $u(x) + v(x)$ and product $u(x)v(x) \bmod 1 + x^n$. Thus we need to construct only a “basic” set of idempotent polynomial. To obtain this basic set for an odd n , we need to partition $Z_n = \{0, 1, \dots, n-1\}$ into classes called cyclotomic cosets.

Let $\widehat{C}_i = \{s = 2^j \cdot i \bmod n : j = 0, 1, \dots, r\}$. For example, for $n = 9$,

$$\widehat{C}_0 = \{0\}, \widehat{C}_1 = \{1, 2, 4, 8, 7, 5\}, \widehat{C}_3 = \{3, 6\}.$$

Next for each class \widehat{C}_i we form a polynomial

$$c_i(x) = \sum_{j \in \widehat{C}_i} x^j.$$

We claim that $c_i(x)$ is an idempotent in R_n . To see this note that,

$$c_i(x)^2 = c_i(x^2) = \sum_{j \in \widehat{C}_i} x^{2j} = \sum_{k \in \widehat{C}_i} x^k \bmod (1 + x^n)$$

since if $j \in \widehat{C}_i$, then so is $2j \bmod n$.

In R_9 all the idempotents are generated by

$$I_1(x) = 1, I_2(x) = x^3 + x^6, \text{ and } I_3(x) = x + x^2 + x^4 + x^5 + x^7 + x^8.$$

There are exactly 7 different nonzero idempotents:

$$I_1(x), I_2(x), I_3(x), I_1(x) + I_2(x), I_1(x) + I_3(x), I_2(x) + I_3(x), \text{ and } I_1(x) + I_2(x) + I_3(x).$$

Note that $I_2(x)I_3(x) = 0$ (the zero polynomial).

To obtain an idempotent in R_n for even $n = 2m$, where m is odd; first we find an idempotent $I(x)$ of R_m , then $J(x) = I(x)^2 \bmod n$ will be an idempotent in R_n . For example, $I(x) = x + x^2 + x^3 + x^4$ is an idempotent in R_5 , hence $J(x) = I(x)^2 = x^2 + x^4 + x^6 + x^8$ is an idempotent in R_{10} . If $n = 2^j$, then the only nonzero idempotent of R_n is $I(x) = 1$.

Exercises. Find a basic set of idempotents in (a) : R_7 , (b) : R_{11} , (c) : R_{12} .

♠ **The Euclidean Algorithm.** The greatest common divisor (or g.c.d.) of two polynomials $f(x), g(x) \in \mathbb{K}[x]$ is the polynomial $d(x) \in \mathbb{K}[x]$ of largest degree such that $f(x) = q_1(x)d(x)$ and $g(x) = q_2(x)d(x)$. In which case we will denote this by $\text{g.c.d.}(f(x), g(x)) = d(x)$.

Given $f(x), g(x) \in \mathbb{K}[x]$ with $\text{degree } f(x) \geq \text{degree } g(x)$ and $g(x) \neq 0$.

Step 1. Initialize $r_0(x) = f(x)$, $r_1(x) = g(x)$, $i = 1$.

Step 2. While $r_i(x) > 0$, divide $r_i(x) > 0$ into $r_{i-1}(x)$ and let $r_{i+1}(x)$ be the remainder. That is $r_{i+1} = r_{i-1}(x) \bmod r_i(x)$. Increment i and repeat.

Step 3. $r_i(x) = 0$. Then $\text{g.c.d.}(f(x), g(x)) = r_{i-1}(x)$.

Using induction one can prove the following:

Theorem 2. If $\text{g.c.d.}(f(x), g(x)) = d(x)$, then there exist polynomials $t(x), s(x) \in \mathbb{K}[x]$ such that

$$t(x)f(x) + s(x)g(x) = d(x).$$

Example 1. If $f(x) = x^2 + x^3 + x^6 + x^7$ and $g(x) = 1 + x^3 + x^4 + x^5$, then find $\text{g.c.d.}(f(x), g(x))$.

Set $i = 1$, dividing $f_1(x) = f(x)$ by $g_1(x) = g(x)$; we find $q_1(x)$ and $r_1(x)$ yields,

$$x^2 + x^3 + x^6 + x^7 = (1 + x^3 + x^4 + x^5)(1 + x^2) + (1 + x^4).$$

Set $i = 2$, dividing $f_2(x) = g_1(x)$ by $g_2(x) = r_1(x)$; we find $q_2(x)$ and $r_2(x)$

$$1 + x^3 + x^4 + x^5 = (1 + x^4)(1 + x) + (x + x^3).$$

Set $i = 3$, dividing $f_3(x) = g_2(x)$ by $g_3(x) = r_2(x)$; we find $q_3(x)$ and $r_3(x)$

$$1 + x^4 = (x + x^3)(x) + (1 + x^2).$$

Set $i = 4$, dividing $f_4(x) = g_3(x)$ by $g_4(x) = r_3(x)$; we find $q_4(x)$ and $r_4(x)$

$$x + x^3 = (1 + x^2)(x) + (0).$$

Steps	$f_k(x)$	$g_k(x)$	$q_k(x)$	$r_k(x)$
Step 1	$x^2 + x^3 + x^6 + x^7$	$1 + x^3 + x^4 + x^5$	$1 + x^2$	$1 + x^4$
Step 2	$1 + x^3 + x^4 + x^5$	$1 + x^4$	$1 + x$	$x + x^3$
Step 3	$1 + x^4$	$x + x^3$	x	$1 + x^2$
Step 4	$x + x^3$	$1 + x^2$	x	0

Since $r_4(x) = 0$, $r_3(x) = 1 + x^2$ is the greatest common divisor of $f(x)$ and $g(x)$:

$$1 + x^2 = g.c.d.(x^2 + x^3 + x^6 + x^7, 1 + x^3 + x^4 + x^5).$$

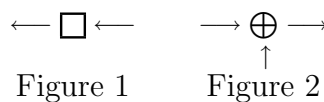
Exercises. Find the greatest common divisor of each of the following pairs of polynomials.

(a) $f(x) = 1 + x + x^5 + x^6 + x^7$, $g(x) = 1 + x + x^3 + x^5$.

(b) $f(x) = 1 + x + x^2 + x^3 + x^4$, $g(x) = x + x^3 + x^4$.

♠ The Origin and Design of Cyclic Codes. We now begin the study of a class of codes, called cyclic codes. These codes have a slick representation in terms of polynomials. Cyclic codes form an important class of codes for many reasons. One is that they can be efficiently encoded by means of shift registers. There are also decoding schemes utilizing shift registers. Many important codes such as the Golay codes, the Hamming codes, and BCH codes can be represented as cyclic codes. Furthermore, much is known theoretically about cyclic codes, which enhances their practical applications.

We start with a device called a linear shift register. This is commonly used to encode cyclic codes. As we shall see, the encoding process is efficient because no storage is required as the codewords are generated by shifting and adding. There are two basic building blocks for shift registers. One is the storage element, which can be a flip-flop, in which a field element is stored, and which has one input and one output (Figure 1). The arrows indicate the input and output. The other building block is the binary adder, which has two inputs and one output (Figure 2), which is the binary sum of the inputs.



We give an example of a shift register, Figure 3, with four storage elements and two binary adders.

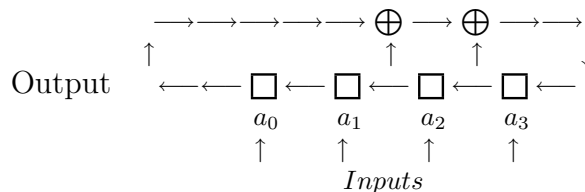


Figure 3

At time zero, four binary elements are placed in a_0, a_1, a_2 , and a_3 . After one time interval a_0 is output, a_1 is shifted into a_0 , a_2 into a_1 , and a_3 into a_2 ; and the new element

is entered into a_3 . In our example this element is the sum $a_0 + a_2 + a_3$ corresponding to the polynomial $g(x) = 1 + x^2 + x^3 \pmod{(1 + x^7)}$. We suppose that the digits 1101 are placed in a_0, a_1, a_2 , and a_3 and follow the outputs and inputs for seven time intervals.

Outputs	a_0 a_1 a_2 a_3	Time
	1 1 0 1	t_0
	1 1 0 1 0	t_1
	1 1 0 1 0 0	t_2
	1 1 0 1 0 0 0	t_3
1 1 0 1 0 0 0 1		t_4
1 1 0 1 0 0 0 1 1		t_5
1 1 0 1 0 0 0 1 1 0		t_6
1 1 0 1 0 0 0 1 1 0 1		t_7

If this process is continued, the vector $(1, 1, 0, 1, 0, 0, 0)$ will be repeated. This shift register will repeat any vector of length 7 that it has generated from four initial entries a_0, a_1, a_2 , and a_3 . This code has the property that whenever $(a_0, a_1, a_2, a_3, a_4, a_5, a_6)$ is a codeword so is $(a_1, a_2, a_3, a_4, a_5, a_6, a_0)$, that is, whenever a word is in the code so are all of its cyclic shifts.

Exercises. First draw the diagram for the shift registers corresponding to generator polynomials $g(x)$, then compute $a(x)g(x)$:

(a) $g(x) = 1 + x + x^3$, $a = 1011$.

(b) $g(x) = x + x^2 + x^3 + x^4$, $a = 10110$.

♠ **Polynomial Construction of Cyclic Codes.** Let v be a word of length n and let P_n be a full cycle permutation matrix obtained from the identity matrix I_n by moving its first row to the last row. The cyclic shift $\sigma(v)$, of v is the word vP_n ; that is the word obtained from v by moving its last digit to the beginning of the word, all other digits moving one position to the right. We can also define $\mu(v) = vP_n^t$; which moves its first digit to the end of the word, all other digits move one position to the left. For example

v	01101	000111	1101
$\sigma(v)$	10110	100011	1110
$\mu(v)$	11010	001110	1011

A code C is said to be cyclic code, if the cyclic shift of each codeword is also a codeword. For example, the linear code $C = \{000, 110, 011, 101\}$ is cyclic. To see that we compute $\sigma(v)$ for all $v \in C$.

$$\sigma(000) = 000, \sigma(110) = 011, \sigma(011) = 101, \text{ and } \sigma(101) = 110.$$

The linear code $C_1 = \{0000, 1100, 0011, 1111\}$ is not cyclic but $C_2 = \{0000, 1010, 0101, 1111\}$ which is equivalent to C_1 is cyclic. The codes \mathbb{K}^n and $\{\theta\}$ are also cyclic; they are called improper cyclic codes. Otherwise, the code is a proper cyclic code. The proper cyclic code $C = \{000\dots 000, 111\dots 111\}$ is of no use to us.

Note that the cyclic shift $\sigma(v) = vP_n$ is a *linear transformation*. Hence to construct a cyclic code, we pick a word, v , form a set S consisting of v and all of its cyclic shifts,

$$S = \{v, \sigma(v), \sigma^2(v), \dots, \sigma^{n-1}(v)\} = \{v, vP_n, vP_n^2, \dots, vP_n^{n-1}\}$$

and define C to be the linear span of S ; that is $C = \langle S \rangle$. C is called the smallest linear cyclic code containing v . For example, if $v = 0101$, then $S = \{0101, 1010, \dots\}$ and $C = \langle S \rangle = \{0000, 0101, 1010, 1111\}$. We say v is a generator of the linear cyclic code C . Note that $\sigma(v)$ is also a generator of C . In general we choose the generator to be the codeword, where its corresponding polynomial $v(x)$ has the least degree. In all that follows, we assume that the corresponding polynomial $v(x)$ to a generator word has the least degree. We shall prove that this polynomial is unique.

If v is a codeword of a linear cyclic code of length n , then $\sigma(v)$ corresponds to the polynomial $xv(x) \bmod (1+x^n)$. For cyclic codes we refer to the codewords in C and polynomials in $C(x)$. We can now restate the definition of cyclic codes in terms of polynomials:

Given a word v of length n , the cyclic shifts of v corresponds to the polynomials

$$x^i v(x) \bmod (1+x^n) \quad \text{for } i = 1, 2, \dots, n-1.$$

To construct a linear cyclic code of length n and dimension k , one must find a factor of $1+x^n$ having degree $n-k$. Of course there may be several choices or none for given n and k . There is also the question of minimum distance; a question which is not settled in general.

Example 2. Let $n = 7$ and $v = 1101000$. Then $v(x) = 1 + x + x^3$

Word	Polynomial (mod (1 + x ⁷))
1101000	$v(x) = 1 + x + x^3$
0110100	$xv(x) = x + x^2 + x^4$
0011000	$x^2v(x) = x^2 + x^3 + x^5$
0001101	$x^3v(x) = x^3 + x^4 + x^6$
1000110	$x^4v(x) = x^4 + x^5 + x^7 \equiv 1 + x^4 + x^5 \bmod (1 + x^7)$
0100011	$x^5v(x) = x^5 + x^6 + x^8 \equiv x + x^5 + x^6 \bmod (1 + x^7)$
1010001	$x^6v(x) = x^6 + x^7 + x^9 \equiv 1 + x^2 + x^6 \bmod (1 + x^7)$

Lemma 1. Let C be a cyclic code of length n and let $v \in C$. Then for any polynomial $p(x)$ of degree $m < n$,

$$c(x) \equiv p(x)v(x) \pmod{(1+x^n)}$$

is a codeword of $C(x)$.

Proof. Let $p(x) = a_0 + a_1x + \cdots + x^m$ and p be its corresponding word in \mathbb{K}^n , then from $p(x)v(x) = a_0v(x) + a_1xv(x) + \cdots + x^mv(x)$, we obtain the word $c = pv = a_0v + a_1\sigma(v) + \cdots + a_m\sigma^m(v)$ of length n . This word c must be a codeword of the linear cyclic code C . Thus we conclude that $c(x) \equiv p(x)v(x) \pmod{(1+x^n)}$ is a codeword of $C(x)$.

Theorem 3. Let C be a cyclic code of length n , then:

- (i) Among all the nonzero codewords of C , there is a unique codeword g such that $g(x)$ has minimum degree.
- (ii) $g(x)$ divides any polynomial in $C(x)$.
- (iii) If degree $g(x) = n - k$, then the codewords $g, \sigma(g), \dots, \sigma^{k-1}(g)$ corresponding to the polynomials $g(x), xg(x), \dots, x^{k-1}g(x)$ form a basis for C .
- (iv) C has dimension k .

Proof. (i) : Suppose there are two polynomials $g(x)$ and $g'(x)$ of minimum degree. Now suppose that this minimum degree is $n - k$, then a contradiction follows from the facts that $h(x) = g(x) + g'(x) \in C(x)$ and $x^{n-k} + x^{n-k} = 0$.

(ii) : Let $c(x) \in C(x)$, then by the Division Algorithm, we have $c(x) = g(x)q(x) + r(x)$ which implies that $r(x) = g(x)q(x) + c(x)$. Lemma 1 and the fact that C is linear imply that $r(x) \in C(x)$. Since all the members of $C(x)$ have higher degrees than $g(x)$, we conclude that $r(x) = 0$.

(iii) : First we show the linear independence. Suppose for some $\alpha_0, \alpha_1, \dots, \alpha_{k-1}$,

$$\alpha_0g + \alpha_1\sigma(g) + \cdots + \alpha_{k-1}\sigma^{k-1}g = \theta.$$

Then
$$\alpha_0g(x) + \alpha_1xg(x) + \cdots + \alpha_{k-1}x^{k-1}g(x) \equiv 0 \pmod{(1+x^n)}.$$

Since the word associated with the polynomial $1 + x^n$ has length $n + 1$, we conclude that $\alpha_0 = \alpha_1 = \cdots = \alpha_{k-1} = 0$. And since $g(x)$ divides any polynomial in $c(x) \in C(x)$, we have:

$$c(x) = a(x)g(x), \text{ where } a(x) = a_0 + a_1x + \cdots + a_{k-1}x^{k-1}.$$

Thus $c(x) = a_0g(x) + a_1xg(x) + \dots + a_{k-1}x^{k-1}g(x)$ and $g(x), xg(x), \dots, x^{k-1}g(x)$ is a basis for $C(x)$. Obviously (iv) follows from (iii).

Theorem 4. Let C be a linear cyclic code of length n .

- (i) $g(x)$ is the generator polynomial for C of length n , if and only if $g(x)$ divides $1 + x^n$.
- (ii) There is a unique idempotent polynomial which generates the code.

Proof. (i) : By the Division Algorithm, $1 + x^n = g(x)q(x) + r(x)$ with $r(x) = 0$ or the degree of $r(x)$ is less than the degree of $g(x)$. Equivalently $r(x) = g(x)q(x) + (1 + x^n)$. But we have

$$r(x) = g(x)q(x) + (1 + x^n) \equiv g(x)q(x) \pmod{1 + x^n}.$$

Thus $r(x) \in C(x)$. Since $g(x)$ has minimum degree, we conclude that $r(x) = 0$.

(ii) : We prove it for odd n . Suppose $g(x)h(x) = 1 + x^n$. Then $\text{g.c.d.}(g(x), h(x)) = 1$ and by Euclidean Algorithm, there exists polynomials $t(x), s(x) \in \mathbb{K}[x]$ such that

$$1 = t(x)g(x) + s(x)h(x).$$

Multiplying both sides by $t(x)g(x)$ gives,

$$t(x)g(x) = [t(x)g(x)]^2 + [t(x)s(x)](1 + x^n) \equiv [t(x)g(x)]^2 \pmod{1 + x^n}.$$

Thus $t(x)g(x)$ is an idempotent in R_n and

$$g(x) = \text{g.c.d.}(t(x)g(x), 1 + x^n).$$

According to this theorem, in order to construct a linear cyclic code of length n , we either factorize $1 + x^n$ and choose one of its factors as a generator polynomial $g(x)$ for the code or obtain an idempotent polynomial. Since there are seven different idempotents in R_9 , and $I(x) = 1$ generates the improper cyclic code \mathbb{K}^9 , we conclude that there are exactly 6 proper linear cyclic codes of length 9. One of the idempotent polynomials in R_9 is $g(x) = 1 + x^3 + x^6$ of degree 6 with 3 nonzero coefficients. It generates a $(9, 9 - 6, 3) = (9, 3, 3)$ cyclic code which can correct at least one error. Its generator matrix is:

$$G = \begin{pmatrix} 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 \end{pmatrix}.$$

The fact that $I(x) = 1$ is the only nonzero idempotent of R_8 , forces us to use the factors of $1 + x^8$, to generate cyclic codes.

Corollary 4.1. *The generator polynomial $g(x)$ for the smallest cyclic code of length n containing the word v (polynomial $v(x)$) is the greatest common divisor of $v(x)$ and $1 + x^n$.*

Proof. If $g(x)$ is the generator polynomial then $g(x)$ divides both $v(x)$ and $1 + x^n$. Since C is the smallest linear cyclic code containing v , $g(x)$ must be in $\langle \{v(x), xv(x), \dots, x^{n-1}v(x)\} \rangle$, thus we have

$$g(x) \equiv p(x)v(x) \pmod{1 + x^n}$$

or equivalently

$$g(x) = p(x)v(x) + q(x)(1 + x^n).$$

Theorem 2 then completes the proof.

Example 3. Let $n = 8$ and $v = 11011000$, i.e. $v(x) = 1 + x + x^3 + x^4$. The greatest common divisor of $1 + x^8$ and $v(x)$ is $g(x) = 1 + x^2$. Thus $g = 10100000$ generates the smallest cyclic code C containing v . Note that C has dimension $8 - 2 = 6$. The following matrix

$$G = \begin{bmatrix} g \\ \sigma(g) \\ \sigma^2(g) \\ \sigma^3(g) \\ \sigma^4(g) \\ \sigma^5(g) \end{bmatrix} = \begin{pmatrix} 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \end{pmatrix}$$

is a generator matrix for C but not in $\mathcal{RRE}\mathcal{F}$ form. An alternative way of finding a generator matrix is to create a matrix M using v and $\sigma^i(v)$, $i = 1, 2, \dots, 7$,

$$M = \begin{bmatrix} v \\ \sigma(v) \\ \sigma^2(v) \\ \sigma^3(v) \\ \sigma^4(v) \\ \sigma^5(v) \\ \sigma^6(v) \\ \sigma^7(v) \end{bmatrix} = \begin{pmatrix} 1 & 1 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 & 1 & 1 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 1 & 0 & 0 & 0 & 1 \end{pmatrix}.$$

Then some simple row reduction operations will produce a 6×8 generator matrix.

Exercises. 1) Find the generator of the smallest linear cyclic code containing the word;

(a) $v = 0110110$;

(b) $v = 0101100$.

2) Find all the proper cyclic codes of \mathbb{K}^5 and \mathbb{K}^{11} .

♠ **Polynomial Encoding and Decoding.** Let C be a linear cyclic code of length n and dimension k (so its generator polynomial $g(x)$ has degree $n - k$). The k information digits

$$a = (a_0, a_1, \dots, a_{k-1})$$

to be encoded can be thought of as a polynomial

$$a(x) = a_0 + a_1x + \dots + a_{k-1}x^{k-1}$$

called information or message polynomial. Encoding consists of simply multiplying $a(x)$ by $g(x)$; that is, $a(x)$ is encoded as $v(x) \equiv a(x)g(x) \pmod{1 + x^n}$. So instead of storing the entire $k \times n$ generator matrix, one only has to store the generator polynomial, which is a significant improvement in terms of complexity of encoding. The inverse operation to polynomial multiplication is polynomial division. Hence finding the message corresponding to the closest codeword to the received code $w(x) = v(x) + u(x)$ consists of dividing $w(x)$ by $g(x)$, thus recovering the message $a(x)$.

The *syndrome polynomial*, $s(x)$, is defined by $s(x) \equiv w(x) \pmod{g(x)}$. Assuming $g(x)$ has degree $n - k$, then $s(x)$ will have degree less than $n - k$ and will correspond to a binary word s of length $n - k$. Since $w(x) = v(x) + u(x)$ and $v(x) = a(x)g(x)$ we have that $s(x) \equiv u(x) \pmod{g(x)}$. We see that the syndrome polynomial depends only on the error.

Example 4. Let $n = 7$ and $g(x) = 1 + x + x^3$, then $k = 4$, and $n - k = 7 - 4 = 3$. The polynomial $g(x)$ will encode any message $a = a_0 a_1 a_2 a_3$ into a codeword $v = v_0 v_1 v_2 v_3 v_4 v_5 v_6$.

To encode $a = 1011$ into a codeword v , we define $a(x) = 1 + x^2 + x^3$ and find

$$\begin{aligned} v(x) &= a(x)g(x) = (1 + x^2 + x^3)(1 + x + x^3) \pmod{1 + x^7} \\ &= 1 + x + x^2 + x^3 + x^4 + x^5 + x^6 \pmod{1 + x^7}. \end{aligned}$$

Thus the codeword is $v = 1111111$.

To decode the received word $w = w_0 w_1 w_2 w_3 w_4 w_5 w_6$, we first define the polynomial $w(x) = w_0 + w_1x + w_2x^2 + w_3x^3 + w_4x^4 + w_5x^5 + w_6x^6$ and then divide $w(x)$ by $g(x)$; the quotient polynomial will simply produce the message.

Suppose the word $w = 1011001$ is received, then

$$\begin{aligned} w(x) &= g(x)q(x) + r(x) \\ 1 + x^2 + x^3 + x^6 &= (1 + x + x^3)(x + x^3) + (1 + x). \end{aligned}$$

The polynomial

$$w(x) + r(x) = (1 + x^2 + x^3 + x^6) + (1 + x) = x + x^2 + x^3 + x^6$$

will produce the codeword $v = 0111001$ and $q(x) = x + x^3$ will give us the message $a = 0101$.

The facts that $r(x) = 1 + x \longleftrightarrow r = 1100000$ and the weight of r is larger than $t = 1$, our decoded word could be wrong. We shall see a decoding algorithm later on.

Parity-check matrix. We can define an $n \times (n - k)$ matrix H , where the i^{th} row H_i is the word of length $n - k$ corresponding to $H_i(x) \equiv x^i \pmod{g(x)}$ $i = 0, 1, \dots, n - 1$. It turns out that this matrix is a parity-check matrix for the code. For, if w is a received word then

$$\begin{aligned} w(x) = v(x) + u(x) &\leftrightarrow wH = (v + u)H = \sum_{i=0}^{n-1} (v_i + u_i)r_i \\ &\leftrightarrow \sum_{i=0}^{n-1} (v_i + u_i)H_i(x) \equiv \sum_{i=0}^{n-1} v_i x^i \pmod{g(x)} + \sum_{i=0}^{n-1} u_i x^i \pmod{g(x)} \\ &\equiv 0 \pmod{g(x)} + u(x) \pmod{g(x)} = s(x) \end{aligned}$$

Then $s(x) = 0$ if and only if $w(x) \in C(x)$, so H is a parity-check matrix.

Example 5. Let $n = 7$ and $g(x) = 1 + x + x^3$, then $n - k = 7 - 4 = 3$. We define H as follows:

$H_0(x) = 1$	$\pmod{g(x)} = 1$	\longleftrightarrow	100
$H_1(x) = x$	$\pmod{g(x)} = x$	\longleftrightarrow	010
$H_2(x) = x^2$	$\pmod{g(x)} = x^2$	\longleftrightarrow	001
$H_3(x) = x^3$	$\pmod{g(x)} = 1 + x$	\longleftrightarrow	110
$H_4(x) = x^4$	$\pmod{g(x)} = x + x^2$	\longleftrightarrow	011
$H_5(x) = x^5$	$\pmod{g(x)} = 1 + x + x^2$	\longleftrightarrow	111
$H_6(x) = x^6$	$\pmod{g(x)} = 1 + x^2$	\longleftrightarrow	101

$$H = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 0 & 1 \end{pmatrix} \text{ with } G = \begin{pmatrix} 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 \end{pmatrix} \text{ and } GH = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}.$$

Note that H is a parity-check matrix of a Hamming code. If $w = 1000011 \leftrightarrow w(x) = 1 + x^5 + x^6$ is received, then $wH = s = 110$ or $s(x) \equiv 1 + x^5 + x^6 \pmod{1 + x + x^3}$.

To construct a Hamming code of length 15, we use $g(x) = 1 + x + x^4$. For $n = 23$, the polynomial $g(x) = 1 + x + x^5 + x^6 + x^7 + x^9 + x^{11}$ generates a Golay code. Finally, if $n = 15$, then $g(x) = 1 + x^4 + x^6 + x^7 + x^8$ generates a 2-error correcting code called BCH code.

Exercises. Find a parity-check matrix for a linear cyclic code of length n with generator $g(x)$:

(a) $n = 6, g(x) = 1 + x^3$.

(b) $n = 9, g(x) = 1 + x^3 + x^6$.

Definition. A cyclic run of 0 of length m of a word of length n is a succession of m cyclically consecutive zero components. For example $e_1 = (1, 1, 0, 0, 0, 0, 0, 1, 0)$ has a cyclic run of 0 of length 5. The error pattern $e_2 = (0, 0, 1, 0, 1, 1, 0, 0, 0)$ with two zeros at the beginning and three zeros at the end, also has a cyclic run of 0 of length 5

♣ **Algorithm for Decoding Linear Cyclic Codes.** Rather than constructing a standard decoding array (SDA) for a cyclic code of length n and distance $d = 2t+1$ or $2t+2$ generated by $g(x)$; we use an algorithm that utilizes polynomial multiplication and division.

Algorithm.

Let C be a $[n, k, d]$ -cyclic code with generator polynomial $g(x)$. Let $w(x)$ be a received word with an error pattern $e(x)$, where $e(x)$ has a cyclic run of 0 of length at least k and

$$\text{weight} \{e(x)\} \leq \lfloor (d-1)/2 \rfloor.$$

The syndrome polynomial, $s(x)$, is defined by $s(x) \equiv w(x) \pmod{g(x)}$. The goal is to determine the error pattern $e(x)$.

Step 1: For $i = 0, 1, 2, \dots$, compute the syndromes

$$s_i(x) \equiv x^i s(x) \equiv x^i w(x) \pmod{g(x)}.$$

Step 2: Find the first $i = j$ such that

$$\text{weight} \{s_j(x)\} = \text{weight} \{x^j s(x)\} \leq \lfloor (d-1)/2 \rfloor.$$

Step 3: Compute the remainder $e(x)$ of $x^{n-j} s_j(x)$ divided by $1 + x^n$, i.e.,

$$e(x) \equiv x^{n-j} s_j(x) \pmod{1 + x^n}.$$

Decode $w(x)$ to $w(x) + e(x)$.

Remark. This decoding algorithm will only correct error pattern $e(x)$ with at least k consecutive zeros. It is quite possible that there are error patterns of weight at most

$\lfloor (d-1)/2 \rfloor$ that do not satisfy this property. Such error patterns are correctable by the codes, but the closest codeword is not found by this algorithm. However, by slightly modifying this algorithm, we can use it to correct burst error patterns. Clearly the above algorithm could be used to correct a 1-error pattern in a Hamming code.

Example 6. Let $n = 7$ and $g(x) = 1+x+x^3$ be the generator polynomial for 1 error-correcting Hamming code. If $w = 0011000 \leftrightarrow w(x) = x^2 + x^3$ is received, then

$$s(x) \equiv w(x) \bmod g(x) \equiv x^2 + x^3 \bmod (1 + x + x^3) \equiv 1 + x + x^2$$

is the syndrome polynomial. We next compute

$$s_1(x) \equiv xs(x) \bmod g(x) \equiv x(1 + x + x^2) \bmod g(x) \equiv 1 + x^2$$

$$s_2(x) \equiv x^2s(x) \bmod g(x) \equiv x(1 + x^2) \bmod g(x) \equiv 1$$

which has weight $t = 1$. So $j = 2$ and therefore

$$u(x) = x^{7-2}s_2(x) \bmod (1 + x^7) \equiv x^5.$$

Thus $v(x) = w(x) + u(x) = (x^2 + x^3) + x^5 \leftrightarrow 0011010$ is the most likely codeword.

Exercises. The polynomial $g(x) = 1 + x^4 + x^6 + x^7 + x^8$ generates a 2-error-correcting linear cyclic code of length 15. Decode the following received words:

(a) $w = 110001101000101$.

(b) $w = 001111101001001$.

Definition A burst of length $\mathcal{L} > 1$ is a binary vector whose nonzero components are confined to \mathcal{L} cyclically consecutive positions, with the first and last positions being nonzero. For example 0011010000 is a burst of length 4, while 010000000100 is a burst of length 5.

A code is called an \mathcal{L} -burst-error-correcting code if it can correct all burst errors of length \mathcal{L} or less; i.e., error patterns that are bursts of length \mathcal{L} or less.

Theorem 5. A linear code C is an \mathcal{L} -burst-error-correcting code if and only if all the burst errors of length \mathcal{L} or less lie in distinct cosets of C .

Proof. If all the burst errors of length \mathcal{L} or less lie in distinct cosets, then each burst error is determined by its syndrome. The error can then be corrected through its syndrome.

On the other hand, suppose that two distinct burst errors e_1 and e_2 of length \mathcal{L} or less lie in the same coset of C . The difference $d = e_1 - e_2$ is a codeword. Thus, if e_1 is received, then e_1 could be decoded to both 0 and d .

Corollary 5.1. *Let C be an $[n, k]$ -linear \mathcal{L} -burst-error-correcting code. Then*

- (i) *No nonzero burst of length $2\mathcal{L}$ or less can be a codeword;*
- (ii) *$n - k \geq 2\mathcal{L}$.*

Proof. (i) Suppose that there exists a codeword v which is a burst of length $\leq 2\mathcal{L}$. Then, v is of the form $(0, 1, u_1, u_2, 1, 0)$, where u_1 and u_2 two words of length $\leq \mathcal{L} - 1$. Hence, the words $w = (0, 1, u_1, 0, 0, 0)$ and $v + w = (0, 0, 0, u_2, 1, 0)$ are two bursts of length $\leq \mathcal{L}$ in the same coset. This is a contradiction to Theorem 5.

(ii) Let $u_1, u_2, \dots, u_{n-k+1}$ be the first $n - k + 1$ rows of a parity-check matrix H of C are linearly dependent.

Thus, there exist $c_1, c_2, \dots, c_{n-k+1}$, not all zero, such that

$$\sum_{i=1}^{n-k+1} c_i u_i = 0.$$

This implies that $(c_1, c_2, \dots, c_{n-k+1}, 0, 0 \dots 0)$ is a codeword, and it is clear that this codeword is a burst of length $\leq n - k + 1$. By part (i), we have $n - k + 1 > 2\mathcal{L}$; i.e., $n - k \geq 2\mathcal{L}$.

♣ Decoding Algorithm for Cyclic Burst-Error-Correcting Codes. First we need to see if a cyclic code is a candidate for the burst-error correcting algorithm.

An $[n, k]$ -linear \mathcal{L} -burst-error-correcting code, satisfies:

$$\mathcal{L} \leq \left\lfloor \frac{n - k}{2} \right\rfloor.$$

A linear burst-error-correcting code achieving the above *Reiger bound* is called an optimal burst-error-correcting code.

The previous algorithm can be directly employed to correct burst errors. The main difference is that, in the case of burst-error-correction, we do not require the weight of an error pattern to be less than or equal to $\lfloor (d(C) - 1)/2 \rfloor$. The modified decoding algorithm for burst-error-correction is as follows:

Let C be a $[n, k, d]$ -cyclic code with generator polynomial $g(x)$. Let $w(x)$ be a received word with an error pattern $e(x)$ that is a burst error of length \mathcal{L} or less.

Step 1: Compute the syndromes $s_i(x) = x^i w(x)$ for $i = 1, 2, \dots$

Step 2: Find the first $i = j$ such that the syndrome for $x^j w(x)$ is a burst of length $\leq \mathcal{L}$.

Step 3: Compute the remainder $e(x)$ of $x^{n-j} s_j(x)$ divided by $1 + x^n$, i.e.,

$$e(x) \equiv x^{n-j} s_j(x) \pmod{1 + x^n}.$$

Decode $w(x)$ to $w(x) + e(x)$.

Example 7. Let C be the binary cyclic code of length 15 generated by

$$g(x) = 1 + x + x^2 + x^3 + x^6.$$

It is a $[15, 9]$ -linear code with $\lfloor \frac{15-9}{2} \rfloor = 3$. Hence C is a 3-burst-error-correcting code.

Suppose $w(x) = 111011101100000 = 1 + x + x^2 + x^4 + x^5 + x^6 + x^8 + x^9$ is received. We need to compute the syndromes $s_i(x)$ of $x^i w(x)$ until $s_j(x)$ is a burst of length 3 or less is found:

i	Polynomial mod $g(x)$
0	$w(x) = 1 + x + x^4 + x^5$
1	$xw(x) = 1 + x^3 + x^5$
2	$x^2w(x) = 1 + x^2 + x^3 + x^4$
3	$x^3w(x) = x + x^3 + x^4 + x^5$
4	$x^4w(x) = 1 + x + x^3 + x^4 + x^5$
5	$x^5w(x) = 1 + x^3 + x^4 + x^5$
6	$x^6w(x) = 1 + x^2 + x^3 + x^4 + x^5$
7	$x^7w(x) = 1 + x^2 + x^4 + x^5$
8	$x^8w(x) = 1 + x^2 + x^5$
9	$x^9w(x) = 1 + x^2$

So we have $e(x) = x^{15-9}(1 + x^2) = x^6(1 + x^2)$. We decode $w = 111011101100000$ into

$$v(x) = w(x) + e(x) = w(x) + x^6 + x^8 = 1 + x + x^2 + x^4 + x^5 + x^9 \implies v = 111011000100000.$$

Here is a list of a few optimal burst-error-correcting cyclic codes:

Code	\mathcal{L}	Generator polynomials
$[7, 3]$	2	$1 + x + x^2 + x^4$
$[7, 3]$	2	$1 + x^2 + x^3 + x^4$
$[15, 9]$	3	$1 + x + x^2 + x^3 + x^6$
$[15, 7]$	4	$1 + x^4 + x^6 + x^7 + x^8$
$[15, 5]$	5	$1 + x + x^2 + x^3 + x^4 + x^5 + x^8 + x^{10}$

Example 8. Let $n = 7$ and let C be the linear cyclic code generated by the polynomial $g(x) = 1 + x + x^2 + x^4$.

(a) Find all the codewords in C .

(b) The word $v = 1110100$ was sent through three different channels and received as:

$$w_1 = 1110111, \quad w_2 = 1110001, \quad \text{and} \quad w_3 = 1110011.$$

(a)
$$C = \left\{ \begin{array}{cccc} 0000000 & 1110100 & 0111010 & 0011101 \\ 1001110 & 1101001 & 0100111 & 1010011 \end{array} \right\}.$$

(b) Notice that the weight of every non-zero codeword in C is exactly 4, so clearly w_1 with six ones and w_3 with five ones are not in C ; this tells us that there are at least two bits in w_1 which should be corrected. We hope that these two bits are next to each other (i.e., the error is a 2-burst-error); this way, we could correct w_1 by the algorithm and find the codeword v which was sent.

First we need to divide $w_1(x) = 1 + x + x^2 + x^4 + x^5 + x^6$ by $g(x)$; the remainder will be the syndrome $s(x)$. We have $s(x) = 1 + x^2$ which is a burst of length 3, so we need to find a j such that $s_j(x)$ is a burst of length 2. The syndrome $s_1(x) = x + x^3$ is a 3-burst syndrome, but $s_2(x) = 1 + x$ is a burst syndrome of length 2. Therefore the error would be

$$e_1(x) = x^{7-2}s_2(x) = x^5(1 + x) = x^5 + x^6 \quad \text{mod } 1 + x^7.$$

Now, we can decode w_1 into v_1 as follows:

$$v_1(x) = w_1(x) + e_1(x) = (1 + x + x^2 + x^4 + x^5 + x^6) + (x^5 + x^6) = 1 + x + x^2 + x^4 \implies v_1 = 1110100.$$

v_1 is exactly the same as the codeword v .

By dividing $w_2(x)$ by $g(x)$, we obtain $s(x) = x^2 + x^3 = s_0(x)$, so

$$e_2(x) = x^{7-0}s_0(x) = 1(x^2 + x^3) = x^2 + x^3 \quad \text{mod } (1 + x^7).$$

$$v_2(x) = w_2(x) + e_2(x) = (1 + x + x^2 + x^6) + (x^2 + x^3) = 1 + x + x^3 + x^6 \implies v_2 = 1101001.$$

The fact that the second channel produced a 2-bit error, the word v_2 must be in the 2burst-error-correcting code C , but the 3-burst error, made v_2 , the wrong codeword.

By dividing $w_3(x)$ by $g(x)$, we obtain $s(x) = x = s_0(x)$, so

$$e_3(x) = x^{7-0}s_0(x) = 1(x) = x \pmod{(1+x^7)}.$$

$$v_3(x) = w_3(x) + e_3(x) = (1+x+x^2+x^5+x^6) + (x) = 1+x^2+x^5+x^6 \implies v_3 = 1010011.$$

Exercises. Let $n = 15$ and let C be the linear cyclic code generated by the polynomial $g(x) = 1 + x^4 + x^6 + x^7 + x^8$. Decode $w = 111011101100000$.

♠ **Dual Cyclic Codes.** Another important fact about cyclic codes is that the dual codes are also cyclic. To find the generator of the Dual of the linear cyclic code C of length n generated by the polynomial $g(x)$, first we find a polynomial $h(x)$ of degree k such that $g(x)h(x) = 1 + x^n$, then we define the generator polynomial of C^\perp as

$$g^\perp(x) = x^k h(x^{-1}).$$

Note that $g^\perp(x)$ is a polynomial of degree k since every single factor of $1+x^n$ has a nonzero constant part and the degree of $h(x)$ is exactly k . To prove that $g^\perp(x) = x^k h(x^{-1})$ is the generator of the dual code C^\perp , we need to relate the product of polynomials and the inner product (dot product) of vectors.

Lemma 2. Consider the following polynomials $a(x) = a_0 + a_1x + a_2x^2 + \dots + a_{n-2}x^{n-2} + a_{n-1}x^{n-1}$ and $b(x) = b_0 + b_1x + b_2x^2 + \dots + b_{n-2}x^{n-2} + b_{n-1}x^{n-1}$. Then $c(x) = a(x)b(x) \equiv 0 \pmod{(1+x^n)}$, if and only if $a = a_0a_1 \dots a_{n-2}a_{n-1}$ is orthogonal to $b = b_0b_1 \dots b_{n-2}b_{n-1}$ and every cyclic shift of this word.

Proof. The proof is computational and for simplicity we give it for $n = 7$ although the general proof is similar. We can express the fact that $a = a_0a_1 \dots a_6$ is orthogonal to the vector $b = b_0b_1 \dots b_6$ and all its cyclic shifts by the following equations:

$$a_0b_6 + a_1b_5 + \dots + a_6b_0 = 0$$

$$a_0b_0 + a_1b_6 + \dots + a_6b_1 = 0$$

$$\vdots \quad \vdots \quad \vdots \quad \vdots \quad \vdots \quad \vdots \quad \vdots = \vdots$$

$$a_0b_5 + a_1b_4 + \dots + a_6b_6 = 0.$$

We compute now $c(x) = a(x)b(x) \equiv 0 \pmod{(1+x^7)}$. For the purpose of our proof, we write

the coefficients of the powers of x in the order $x^6, x^0 = 1, x, x^2, x^3, x^4, x^5$. Then

$$\begin{aligned} c(x) &= (a_0b_6 + a_1b_5 + \cdots + a_6b_0)x^6 \\ &\quad + (a_0b_0 + a_1b_6 + \cdots + a_6b_1) \\ &\quad + \begin{matrix} \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \end{matrix} \\ &\quad + (a_0b_5 + a_1b_4 + \cdots + a_6b_6)^5. \end{aligned}$$

The proof follows since these coefficients equal the dot product above.

Theorem 6. *If C is a linear cyclic code of length n generated by the polynomial $g(x)$, then C^\perp is generated by the polynomial $g^\perp(x) = x^k h(x^{-1})$, where $g(x)h(x) = 1 + x^n$.*

Proof. Since $g(x)h(x) = 1 + x^n$, we have $g(x^{-1})h(x^{-1}) = 1 + (x^{-1})^n$ and

$$1 + x^n = x^n[1 + (x^{-1})^n] = g(x^{-1})h(x^{-1}) = [x^{n-k}g(x^{-1})][x^k h(x^{-1})].$$

Thus $x^k h(x^{-1})$ is a factor of $1 + x^n$, having degree k and hence the generator polynomial for the linear cyclic code C^\perp .

Lemma 3. *The Linear cyclic code generated by $g(x) = 1 + x$ is the $n - 1$ dimensional code C of all even words of length n .*

Proof. Note that $1 + x^n = (1 + x)(1 + x + \cdots + x^{n-1})$. If $g(x) = 1 + x$, then since $h(x) = 1 + x + \cdots + x^{n-1} = xh(x^{-1})$, the dual code C^\perp consists of the zero vector and vector h with all-ones. Clearly any word is orthogonal to h if and only if it has even weight.

Corollary 6.1. *A linear cyclic code $C = \langle g(x) \rangle$ is self-dual, if $1 + x$ divides $g(x)$.*

Proof. In a self-dual code, all weights are even. Clearly the converse need not hold.

Example 9. Let $n = 7$ and $g(x) = 1 + x + x^3$ be the generator polynomial for 1 error-correcting Hamming code. By long division we obtain $h(x) = 1 + x + x^2 + x^4$. The generator for C^\perp is

$$g^\perp(x) = x^4 h(x^{-1}) = x^4(1 + x^{-1} + x^{-2} + x^{-4}) = 1 + x^2 + x^3 + x^4$$

which corresponds to $g^\perp = 1011100$. Clearly $g.g^\perp = (11010000).(1011100) = 0$ and $\sigma^k(g).g^\perp = 0$. Note that $g^\perp(x) \neq h(x)$.

Exercises. Find a generator polynomial for the dual of the linear cyclic code of length n with generator $g(x)$:

- (a) $n = 6$, $g(x) = 1 + x + x^2$.
- (b) $n = 9$, $g(x) = 1 + x^3 + x^6$.