Assignment 2 - ACT1

Nirjhar Nath
nirjhar@cmi.ac.in
BMC202239

Solution 1:

a) We have, $\mathbb{F}_{q^m} = \mathbb{F}_q[x]/(p(x))$, where $p(x)$ is an irreducible polynomial of degree m over $\mathbb{F}_q$. This setup implies an isomorphism between $\mathbb{F}_{q^m}$ and the vector space $\mathbb{F}_q^m$. As a result, each element of $\mathbb{F}_{q^m}$ can be represented as an m-tuple (or vector) over $\mathbb{F}_q$.

Given the original $(n, k, d)_{q^m}$ code, we can replace each symbol from $\mathbb{F}_{q^m}$ with its corresponding m-tuple over $\mathbb{F}_q$. Thus, each codeword in the original code, which has length n over $\mathbb{F}_{q^m}$, is transformed into a codeword of length $n \cdot m$ over $\mathbb{F}_q$. The number of codewords remains the same, so the code size $|C|$ is unchanged. Since each codeword can be described by k symbols over $\mathbb{F}_{q^m}$, the total number of possible codewords is

$$|C| = (q^m)^k = q^{km}.$$

Therefore, the dimension of the new code over $\mathbb{F}_q$ is km.

Now, we analyze the minimum distance of the new code. If two codewords differ in at least d positions in the original $(n, k, d)_{q^m}$ code, their transformed versions will differ in at least $d \cdot m$ positions in the new $(nm, km, d')_q$ code (since each differing symbol in $\mathbb{F}_{q^m}$ leads to m differing entries in $\mathbb{F}_q$). Therefore, $d' \geq d$. Thus, given an $(n, k, d)_{q^m}$ code, it is possible to construct an $(nm, km, d')_q$ code with $d' \geq d$.

b) Let G be the generator matrix for the original $[n, k, d]_{q^m}$ code. To construct a code over $\mathbb{F}_q$, we first identify $\mathbb{F}_{q^m}$ with $\mathbb{F}_q[x]/(p(x))$, where $p(x)$ is an irreducible polynomial of degree m. This lets us view elements of $\mathbb{F}_{q^m}$ as m-dimensional vectors over $\mathbb{F}_q$ (from part a)). Using this identification, we define an isomorphism ϕ from $\mathbb{F}_{q^m}$ to $\mathbb{F}_q^m$, where each element $a \in \mathbb{F}_{q^m}$ is mapped to an m-dimensional vector over $\mathbb{F}_q$. Now we construct a new matrix G' by replacing each entry g_{ij} in G with a column vector in $\mathbb{F}_q^m$, structured as follows:

$$G'_{ij} = \begin{bmatrix} \phi(g_{ij}) \\ x\phi(g_{ij}) \\ \vdots \\ x^{m-1}\phi(g_{ij}) \end{bmatrix},$$

where x is interpreted as a power in the field representation. This replacement transforms each entry of G into an $m \times m$ matrix over $\mathbb{F}_q$, effectively expanding G into an $[nm, km]$ matrix G' over $\mathbb{F}_q$.

The rows of G' remain linearly independent because the rows of G were linearly independent over $\mathbb{F}_{q^m}$, and ϕ is an isomorphism. Thus, the rows of G' form a basis for a new $[nm, km, d']_q$ code, where $d' \geq d$. Hence, G' is the generator matrix for the new code over $\mathbb{F}_q$.

c) Consider the new code defined by

$$C = \{(u, u + v) \mid u \in C_1, v \in C_2\}$$

where C_1 and C_2 are the original $[n, k_1, d_1]_q$ and $[n, k_2, d_2]_q$ codes, respectively.

This new code is a $[2n, k_1 + k_2, d']_q$ code, with $d' = \min(2d_1, d_2)$. To see why this holds, note that if $u = 0$, the minimum distance is $\min\{\mathrm{wt}(v)\} = d_2$, and if $v = 0$, the minimum distance is $2 \min_{u \in C_1} \mathrm{wt}(u) = 2d_1$.

When both u and v are non-zero, we have

$$\mathrm{wt}(u, u+v) = \mathrm{wt}(u) + \mathrm{wt}(u+v) \geq \mathrm{wt}(u) + \mathrm{wt}(v) - \mathrm{wt}(u) = \mathrm{wt}(v) \geq \min(2d_1, d_2).$$

This holds because if v has d non-zero coordinates, at most $\mathrm{wt}(u)$ out of them can be made 0 by adding u. So $d' = \min(2d_1, d_2)$.

A generator matrix for this code is

$$G = \begin{bmatrix} G_1 & G_1 \\ 0 & G_2 \end{bmatrix}$$

where G_1 and G_2 are the generator matrices for C_1 and C_2, respectively.

d) Given an $(n, k, d)_q$ code C where $d = \delta n$, we aim to show that for any positive integer m, there exists an $\left(n^m, \frac{k}{m}, (1-(1-\delta)^m) \cdot n^m\right)_{q^m}$ code. To construct this code, we use the tensor power $C^{\otimes m} = C \otimes C \otimes \cdots \otimes C$ (with m copies of C).

In the original code C, the alphabet size is q, so each symbol in C is an element of the set $\Sigma = \mathbb{F}_q$. For the new code $C^{\otimes m}$, we define an alphabet of size q^m, denoted by Σ', where Σ' consists of all possible m-tuples over $\mathbb{F}_q$: $\Sigma' = \{(q_1, q_2, \ldots, q_m) : q_i \in \mathbb{F}_q\}$. Thus, $|\Sigma'| = q^m$.

Each codeword $c = (p_1, p_2, \ldots, p_n) \in C$ is expanded in $C^{\otimes m}$ by taking all possible products of elements from $\{p_1, p_2, \ldots, p_n\}$ across m copies of C, resulting in codewords of length n^m over Σ'. Each element in a codeword of $C^{\otimes m}$ is represented as an m-tuple, so the length of each codeword in the new code is n^m.

Since each codeword in C is expanded to an alphabet of size q^m, the number of codewords in $C^{\otimes m}$ remains the same as in C, giving us $|C| = (q^m)^{\frac{k}{m}}$. This implies that the dimension of the new code over $\mathbb{F}_{q^m}$ is $\frac{k}{m}$.

To determine the minimum distance of $C^{\otimes m}$, we note that the minimum distance of C is $d = \delta n$, meaning that any two distinct codewords in C differ in at least d positions. Thus, in C, at most $n - d$ positions can match between any two codewords. When we expand each codeword in C to form $C^{\otimes m}$, each position in the original codeword (which could match in at most $n-d$ positions) is represented by m symbols in $C^{\otimes m}$, resulting in a maximum of $(n-d)^m$ matching positions in the new code.

Therefore, the minimum number of differing positions between any two codewords in $C^{\otimes m}$ is at least $n^m - (n-d)^m$. Substituting $d = \delta n$, we find that the minimum distance d' of $C^{\otimes m}$ is:

$$d' = n^m - (n - \delta n)^m = n^m - n^m \cdot (1-\delta)^m = n^m \cdot (1 - (1-\delta)^m)\,.$$

Thus, the minimum distance of $C^{\otimes m}$ is $d' = n^m(1-(1-\delta)^m)$.

Thus, given an $(n, k, \delta n)_q$ code, for any positive integer m, we can construct an $\left(n^m, \frac{k}{m}, (1-(1-\delta)^m) \cdot n^m\right)_{q^m}$ code.

e) If there exists an $[n, k, \delta n]_2$ code, then for every positive integer m, there exists an $[n^m, k, \frac{1}{2}(1-(1-2\delta)^m) \cdot n^m]_2$ code.

Let the initial code be an $[n, k, d]_2$ code, where $d = \delta n$. Let G be the $k \times n$ generator matrix for this code. We create a new generator matrix G' of dimension $n^m \times k$ as follows: each column in G' is represented by an m-tuple $(i_1, i_2, \ldots, i_m)$, where $1 \leq i_p \leq n$. We represent the i^{th} column of G by c_i, and a bit $b \in \{0, 1\}$ is represented as $(-1)^b \in \{1, -1\}$.

In G', the $(i_1, \ldots, i_m)^{\text{th}}$ column is defined as the sum of the columns $i_1, i_2, \ldots, i_m$ in G. Thus, for any vector x, xG' is an n^m-length code. This construction ensures that the new code has code length n^m, and since the number of codewords remains the same, the dimension remains k.

For any $i \in [n]$, $\langle c_i, x \rangle$ represents the i^{th} coordinate of xG. For an n^m-tuple w, we define

$$[w] := \frac{(\text{number of 0's in } w) - (\text{number of 1's in } w)}{n^m}.$$

Now, consider $w = xG'$. Then

$$[w] = \frac{1}{n^m} \sum_{i_1, \ldots, i_m \in [n]} (-1)^{\sum_{j=1}^{m} \langle c_{i_j}, x \rangle}.$$

Expanding this summation and simplifying, we have:

$$\begin{aligned}
[w] &= \frac{1}{n^m} \sum_{i_1, \ldots, i_m \in [n]} \prod_{j=1}^{m} (-1)^{\langle c_{i_j}, x \rangle} \\
&= \prod_{j=1}^{m} \left(\frac{1}{n} \sum_{i \in [n]} (-1)^{\langle c_i, x \rangle} \right) \\
&= \prod_{j=1}^{m} [xG] \\
&= ([xG])^m.
\end{aligned}$$

To find a bound for $[xG]$, let

$$[xG] = \frac{(\text{number of 0's in } xG) - (\text{number of 1's in } xG)}{n}.$$

Since the weight wt$c) = d$, we have

$$[xG] \geq \frac{n - d - d}{n} = \frac{n - 2d}{n}.$$

Substituting this bound into the expression for $[w]$, we get

$$[w] \geq \left(\frac{n - 2d}{n} \right)^m.$$

Therefore, this bound implies

$$[w] \geq \left(\frac{n - 2d}{n} \right)^m,$$

which means that the difference between the number of 0's and 1's in w is at least $n^m \cdot \left(\frac{n-2d}{n}\right)^m$, or equivalently,

$$(\text{no. of 0's in } w) - (\text{no. of 1's in } w) \geq (n-2d)^m.$$

Since the number of 1's in w satisfies

$$\text{no. of 1's in } w \geq \frac{n^m - (n-2d)^m}{2},$$

we find that the effective minimum distance of the new code is at least

$$\frac{1}{2} \cdot n^m(1-(1-2\delta)^m).$$

Therefore, the new code is described by the parameters

$$\left[n^m, k, \frac{1}{2}(1-(1-2\delta)^m) \cdot n^m\right]_2.$$

Thus, given an $[n, k, \delta n]_2$ code, it is possible to construct a new code with parameters $[n^m, k, \frac{1}{2}(1-(1-2\delta)^m) \cdot n^m]_2$ for any positive integer m.

■

Solution 2:

a) i. For $x, y \in F_q$, define

$$\Delta(x,y) = \begin{cases} 0, & \text{if } x = y \\ 1, & \text{if } x \neq y \end{cases}$$

Let $\mathcal{C}$ be the set of all columns for the code C. We express S as follows:

$$S = \sum_{c_1, c_2 \in C} \Delta(c_1, c_2) = \sum_{x \in \mathcal{C}} \sum_{i \neq j} \Delta(x_i, x_j).$$

Let $\text{freq}_x(\alpha)$ denote the number of times $\alpha \in F_q$ appears in column x. Then the sum S becomes:

$$S = \sum_{x \in \mathcal{C}} \sum_{\alpha \in F_q} (\text{freq}_x(\alpha))(\alpha)(|C| - (\text{freq}_x(\alpha))(\alpha)).$$

The inner summation simplifies to

$$\sum_{\alpha \in F_q} (\text{freq}_x(\alpha))^2(|C| - (\text{freq}_x(\alpha))),$$

because for each α, there are $|C| - (\text{freq}_x(\alpha))$ elements α' such that $\alpha' \neq \alpha$ and $\Delta(\alpha, \alpha') = 1$, occurring $\text{freq}_x(\alpha)$ times within x.
Since $\sum_{\alpha \in F_q} (\text{freq}_x(\alpha))^2 = |C|$, we can rearrange:

$$S = \sum_{x \in \mathcal{C}} (|C|^2 - \sum_{\alpha \in F_q} (\text{freq}_x(\alpha))^2)$$

Applying the Cauchy-Schwarz inequality:

$$\sum_{\alpha \in F_q} ((\text{freq}_x(\alpha))^2) \geq \left(\sum_{\alpha \in F_q} (\text{freq}_x(\alpha)) \right)^2 \cdot \frac{1}{q} = |C|^2 \cdot \frac{1}{q}$$

This allows us to bound S by:

$$S \leq \sum_{x \in \mathcal{C}} \left(|C|^2 - |C|^2 \cdot \frac{1}{q} \right) = n|C|^2 \left(1 - \frac{1}{q} \right)$$

ii. Let $\mathcal{R}$ be the set of all rows for the code C. Since $\Delta(c, x) \geq d$ for all $c \neq x$ in C, the sum S can be described as:

$$S = \sum_{x \in \mathcal{R}} \sum_{c \neq x \in C} \Delta(c, x).$$

Given the minimum distance d, the inequality $\Delta(c, x) \geq d$ holds for all $c \neq x$, and therefore,

$$S \geq |C|(|C| - 1)d.$$

iii. From the results from parts a) and b), we have,

$$\begin{aligned}
&|C|(|C| - 1)d \leq S \leq \left(1 - \frac{1}{q} \right) |C|^2 n \\
\implies &(|C| - 1)d \leq \left(1 - \frac{1}{q} \right) |C| n \\
\implies &|C|qd - qd \leq |C|qn - |C|n \\
\implies &|C|(qd - qn + n) \leq qd \\
\implies &|C| \leq \frac{qd}{qd - qn + n} = \frac{qd}{qd - (q-1)n}.
\end{aligned}$$

b) i. The Griesmer Bound states that for an $[n, k, d]_q$ code, the length n must satisfy the following inequality:

$$n \geq \sum_{i=0}^{k-1} \left\lceil \frac{d}{q^i} \right\rceil.$$

We start separating the sum into its first term and the remaining terms, as follows:

$$n \geq \left\lceil \frac{d}{q^0} \right\rceil + \sum_{i=1}^{k-1} \left\lceil \frac{d}{q^i} \right\rceil = d + \sum_{i=1}^{k-1} \left\lceil \frac{d}{q^i} \right\rceil.$$

Since each term in the sum $\left\lceil \frac{d}{q^i} \right\rceil$ for $i \geq 1$ is at least 1 (assuming d and q are positive), the sum of $k - 1$ such terms is at least $k - 1$. Thus,

$$n \geq d + (k - 1).$$

Rearranging the inequality gives:

$$k \leq n - d + 1.$$

ii. The Griesmer Bound for an $[n, k, d]_q$ code states that

$$n \geq \sum_{i=0}^{k-1} \left\lceil \frac{d}{q^i} \right\rceil .$$

Since each term in the series is the ceiling of a fraction, the sum can be bounded from below by the sum of the actual fractions, as follows:

$$\sum_{i=0}^{k-1} \left\lceil \frac{d}{q^i} \right\rceil \geq \sum_{i=0}^{k-1} \frac{d}{q^i} = d \sum_{i=0}^{k-1} \left(\frac{1}{q}\right)^i .$$

This series sums to:

$$d \left(\frac{1 - \left(\frac{1}{q}\right)^k}{1 - \frac{1}{q}} \right) = d \frac{\left(1 - \frac{1}{q^k}\right)}{1 - \frac{1}{q}}.$$

Since $|C| = q^k$, we therefore have,

$$n \geq d \left(\frac{1 - \frac{1}{|C|}}{1 - \frac{1}{q}} \right) .$$

This implies:

$$\begin{aligned} 1 - \frac{1}{|C|} &\leq \frac{n}{d} \left(1 - \frac{1}{q}\right) \\ \implies \frac{1}{|C|} &\geq 1 - \frac{n}{d} + \frac{n}{qd} \\ \implies |C| &\leq \frac{1}{1 - \frac{n}{d} + \frac{n}{qd}} = \frac{qd}{qd - qn + n}. \end{aligned}$$

This finally gives:

$$|C| \leq \frac{qd}{qd - (q-1)n}.$$

■

Solution 3:

a) Suppose C is a cyclic code. Given the linearity of C, it follows that $(C, +)$ constitutes a subgroup of the additive group formed by $\mathbb{F}_q[x]/(x^n - 1)$. Define R to be the quotient ring $\mathbb{F}_q[x]/(x^n - 1)$, where the set $\{1, x, x^2, \ldots, x^{n-1}\}$ is a basis for R. Consider a codeword $(c_0, c_1, \ldots, c_{n-1}) \in C$, For $i = 0$ to $n - 1$,

$$x^i(c_0 + c_1 x + \cdots + c_{n-1}x^{n-1}) = c_0 x^i + c_1 x^{i+1} + \cdots + c_{n-1}x^{i+n-1}.$$

Simplifying modulo $x^n - 1$ results in:

$$(c_{n-i}, \ldots, c_{n-1}, c_0, \ldots, c_{n-i-1}) \in C.$$

Thus, C is an ideal in R.

Conversely, if C is an ideal in $\mathbb{F}_q[x]/(x^n-1)$, then C is inherently a linear code by the properties of ring ideals. Considering any codeword $(c_0, \ldots, c_{n-1})$ in C and applying a single multiplication by x, we have:

$$x(c_0 + c_1x + \cdots + c_{n-1}x^{n-1}) = c_{n-1} + c_0x + c_1x^2 + \cdots + c_{n-2}x^{n-1}.$$

Under modulo $x^n - 1$, this gives the vector $(c_{n-1}, c_0, \ldots, c_{n-2}) \in C$. Thus, C is cyclic.

b) To show that the generating polynomial $g(x)$ of a cyclic code C of block length n divides $x^n - 1$ in $\mathbb{F}_q[x]$, we start by noting that C forms a principal ideal in $\mathbb{F}_q[x]/(x^n-1)$. Therefore, any polynomial $f(x)$ in $\mathbb{F}_q[x]$ can be written as:

$$x^n - 1 = q(x)g(x) + r(x).$$

where the degree of $r(x)$ is less than the degree of $g(x)$. Since $r(x)x \equiv -q(x)g(x)$ mod $x^n - 1$, therefore, $r(x) \in C$. However, because $r(x)$ has a lower degree than that of the generator $g(x)$, the only polynomial of such degree in C that satisfies these conditions is the zero polynomial. Thus, $r(x) = 0$, which implies that

$$x^n - 1 = q(x)g(x).$$

Thus, $g(x)$ divides $x^n - 1$.

c) To describe the set of all possible cyclic codes of block length n over $\mathbb{F}_q$, we first observe that each cyclic code is defined by a generating polynomial that divides $x^n - 1$. Because $x^n - 1$ can be factored into irreducible polynomials over $\mathbb{F}_q$, every generating polynomial of a cyclic code can be represented as a product of these irreducible factors.

Let $x^n - 1 = f_1(x)f_2(x)\ldots f_m(x)$ be the decomposition of $x^n - 1$ into irreducible polynomials over $\mathbb{F}_q$. The set of all generating polynomials $g(x)$ of cyclic codes can be given by:

$$\left\{ \prod_{i \in S} f_i(x) \mid S \subseteq \{1, 2, \ldots, m\} \right\}$$

where each subset S of the index set $\{1, 2, \ldots, m\}$ corresponds to a unique cyclic code whose generating polynomial is the product of the corresponding irreducible factors selected by S.

d) In the field $\mathbb{F}_q$, each cyclic code of block length n can be generated by a distinct factor of the polynomial $x^n - 1$. Since $x^n - 1$ in $\mathbb{F}_q[x]$ decomposes into irreducible factors, the number of different cyclic codes is determined by the number of ways we can select these factors. If $x^n - 1$ has t distinct irreducible factors, then there are 2^t different subsets of these factors, including the empty set. Each subset corresponds to a product of factors and thus a distinct generator polynomial, leading to a distinct cyclic code. Therefore, there exist 2^t many distinct cyclic codes of block length n.

e) For a cyclic code C with generating polynomial $g(x) = g_0 + g_1x + \cdots + g_dx^d$, where $\deg(g) = d$, the code's dimension is determined as $n - d$. This is because each codeword in C can be expressed as $c(x) = g(x)q(x)$ for some polynomial $q(x)$ in

$\mathbb{F}_q[x]$, where the degree of $q(x)$ is less than $n-d$. This implies that the basis for C consists of the polynomials $\{g(x), xg(x), \ldots, x^{n-d-1}g(x)\}$.

The corresponding generator matrix G for C is constructed as follows:

$$G = \begin{bmatrix} g_0 & g_1 & \cdots & g_d & 0 & \cdots & 0 \\ 0 & g_0 & g_1 & \cdots & g_d & \cdots & 0 \\ \vdots & \ddots & \ddots & & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & g_0 & g_1 & \cdots & g_d \end{bmatrix}_{(n-d)\times n}$$

f) To compute the generating polynomial $g(x)$ of a cyclic code C from its generator matrix G in $\mathbb{F}_q^n$, the following algorithm can be applied:

Input: Generator matrix G of cyclic code C in $\mathbb{F}_q^n$.

Output: Generating polynomial $g(x)$.

1. Transform G into its row-echelon form using Gaussian elimination.
2. Identify the first non-zero row in the row echelon form of G. Let this row be represented as $[r_0, r_1, r_2, \ldots, r_{n-1}]$.
3. Formulate the generating polynomial $g(x)$ from the coefficients of the first non-zero row:
$$g(x) = r_0 + r_1 x + r_2 x^2 + \cdots + r_{n-1}x^{n-1}$$
Here, $g(x)$ might often terminate at a degree less than $n-1$, depending on the position of the last non-zero coefficient.

Getting the cyclic generator matrix by Gaussian elimination takes poly(n) $\mathbb{F}_q$-operations.

g) Let's consider the degree $\deg(g)$ of the generating polynomial g to be d. Given that g is a divisor of $x^n - 1$, we express $x^n - 1$ as the product $g(x)h(x)$, where $h(x)$ is a polynomial in $\mathbb{F}_q[x]$ with $\deg(h) = n-d$. This polynomial h satisfies the conditions for being the check polynomial of the cyclic code C, and it is monic due to $g(x)$ and $x^n - 1$ both being monic.

For any codeword $c(x)$ belonging to C, it holds that $c(x) = q(x)g(x)$ for some polynomial $q(x)$ in $\mathbb{F}_q[x]$. Consequently, the product $c(x)h(x) = q(x)g(x)h(x) = q(x)(x^n - 1)$, which simplifies to 0 modulo $x^n - 1$.

Also, if there exists a codeword $c(x)$ such that $c(x)h(x)$ is zero modulo $x^n - 1$, and if we assume $c(x) = q(x)g(x) + r(x)$ with $\deg(r) < d$, then $c(x)h(x) = r(x)h(x)$. This results in 0 modulo $x^n - 1$ and since $\deg(r(x)h(x)) < d + n - d$ and $h \neq 0$, it must follow that $r = 0$, leading to $c(x) = q(x)g(x)$, and thus $c(x) \in C$.

Define $h(x)$ as $h_0 + h_1 x + \ldots + h_{n-d}x^{n-d}$. The parity check matrix H for C is constructed as follows:

$$H = \begin{bmatrix} h_{n-d} & \ldots & h_0 & 0 & \ldots & 0 \\ 0 & h_{n-d} & \ldots & h_0 & \ldots & 0 \\ \vdots & & \ddots & & \ddots & \vdots \\ 0 & \ldots & 0 & h_{n-d} & \ldots & h_0 \end{bmatrix}$$

To verify that the parity check matrix indeed functions correctly, consider a codeword $c(x) = c_0 + c_1x + \ldots + c_{n-1}x^{n-1}$ from the cyclic code C. Given that $c(x)h(x) = 0$ modulo $x^n - 1$, we need all coefficients from x^{n-1} down to x^0 in the expansion of $c(x)h(x)$ to be zero. This requirement can be expressed by the following system of linear equations:

$$\begin{aligned} c_0h_{n-d} + c_1h_{n-d-1} + \ldots + c_{n-d}h_0 &= 0, \\ c_1h_{n-d} + c_2h_{n-d-1} + \ldots + c_{n-d+1}h_0 &= 0, \\ &\vdots \\ c_{n-d}h_d + c_{n-d+1}h_{d-1} + \ldots + c_nh_0 &= 0. \end{aligned}$$

These equations demonstrate that all codewords c in C indeed satisfy $Hc = 0$. Thus, H is the appropriate parity matrix for C.

h) Given the generating polynomial $g(x)$ of a cyclic code C in $\mathbb{F}_q^n$ as input, to output the check polynomial of C in poly(n) $\mathbb{F}_q$-operations, we can follow the below algorithm:

Input: Generating polynomial of the cyclic code C in $\mathbb{F}_q^n$.

Output: Check polynomial of C.

Procedure:

1. Start with the polynomial $x^n - 1$.
2. Perform the polynomial division of $x^n - 1$ by $g(x)$.
3. Return the result of the division, which is the check polynomial $h(x)$ of C.

Here, the polynomial division process takes poly(n) $\mathbb{F}_q$-operations.

■