

1. How many cyclic codes of length 4 are there over  $GF(3)$ ? Write down the generator polynomial for each of these cyclic codes.
2. Let  $C$  be a binary cyclic code of length  $n$  (note that  $n$  is odd). Prove that exactly one of the following holds:
  - Every codeword in  $C$  has even weight.
  - The word  $11\dots 11$  is a codeword.
3. In this exercise, you are asked to construct BCH-codes of length 15 and designed distance 5 over certain fields. Try to find an example with maximal dimension. For each code, you must identify  $b$  and the dimension.
  - (a) over  $GF(4)$ .
  - (b) over  $GF(16)$ .
4. Construct a BCH-code of length 17 and dimension 9 over  $GF(4)$  that can correct at least three errors.
5. Construct a binary BCH-code of length 31 of designed distance 5 with minimum distance at least 11.
6. This exercise is about the relation between certain Hamming codes and cyclic codes.

Let  $r \geq 1$ . Let  $\alpha$  be a primitive element in  $GF(2^r)$  (so  $\alpha$  is an element of order  $n := 2^r - 1$ ). Then every element of  $GF(2^r)$  can be written as a polynomial over  $GF(2)$  in  $\alpha$  of degree less than  $r$ :  $a_0 + a_1\alpha + a_2\alpha^2 + \dots + a_{r-1}\alpha^{r-1}$  where  $a_0, a_1, \dots, a_{r-1} \in GF(2)$ .

Consider the matrix

$$H = \begin{bmatrix} 1 & \alpha & \alpha^2 & \dots & \alpha^{n-1} \end{bmatrix}$$

where we write  $\alpha^j$  in polynomial form and turn it into a column. Let  $C$  be the binary code with  $H$  as a parity check matrix.

- (a)  $H$  is parity check matrix if the rows of  $H$  are linearly independent. What are the dimensions of  $H$ ? Are the rows of  $H$  linearly independent?
- (b) Show that  $C$  is a binary Hamming code.
- (c) Let  $\vec{w} = (w_0, w_1, \dots, w_{n-1}) \in GF(2)^n$ . Prove that  $\vec{w} \in C$  if and only if  $\vec{w}(\alpha) = 0$ .
- (d) Deduce that  $C$  is a cyclic code. What is the generator polynomial of  $C$ ? What is the dimension of  $C$ ?
- (e) Prove that the following decoding algorithm is Nearest Neighbor Decoding:
  - When receiving a words  $\vec{w}$ , evaluate  $\vec{w}(\alpha)$ .
  - If  $\vec{w}(\alpha) = 0$  then we decode  $\vec{w}$  as  $\vec{w}$ .
  - If  $\vec{w}(\alpha) \neq 0$  then  $\vec{w}(\alpha) = \alpha^j$  for a unique  $0 \leq j \leq n - 1$  (why?) and we decode  $\vec{w}$  as  $\vec{w} + \vec{e}_j$  (the word corresponding to  $\vec{w}(x) + x^j$ ).