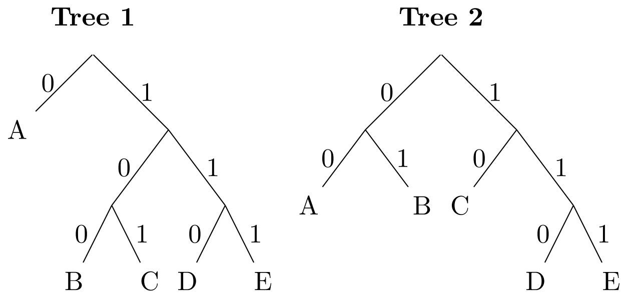
InC Assignment 2

Sricharan Vinoth Kumar, Roll No: 2024112022

1 Huffman Coding

1.1

Given Huffman Trees:



(a) Message: 0100011101

In Tree 1, the different code words are:

A: 0

B: 100

C: 101

D: 110

E: 111

Using these to decode the message, we get,

0-A, 100-B, 0-A, 111-E, 110-C

$\implies$ Original Message: ABAEC

(b) Given:

$p(A) = 0.5, p(B) = p(C) = p(D) = p(E) = 0.125$

To figure out which tree would yield the shortest encoded message, averaged over many messages, we need to compare their Expected Lengths.

The Expected Length of a Codebook is given by,

$$L(C) = \sum_{x \in X} l(x) p_X(x) \tag{1.1.1}$$

Where,

$$\begin{aligned} L(C) &- \text{Expected Length of Codebook C} \\ l(x) &- \text{Length of codeword of } x \\ p_X(x) &- \text{Probability of } x \text{ appearing in the original message} \end{aligned}$$

The expected length of the Codebook formed by Tree 1 would be (using the codewords obtained in Q1.1),

$$\begin{aligned} L(C_1) &= \sum_{x \in X} l(x)p_X(x) \\ &= 1 \times 0.5 + 4(3 \times 0.125) \\ &= 0.5 + 1.5 \\ \therefore L(C_1) &= 2 \end{aligned}$$

In Tree 2, the different codewords are:
A: 00
B: 01
C: 10
D: 110
E: 111
Using these, the expected length of the Codebook formed by Tree 2 would be,

$$\begin{aligned} L(C_2) &= \sum_{x \in X} l(x)p_X(x) \\ &= 2 \times 0.5 + 2(2 \times 0.125) + 2(3 \times 0.125) \\ &= 1 + 0.5 + 0.75 \\ \therefore L(C_2) &= 2.25 \end{aligned}$$

Clearly, $L(C_1) < L(C_2)$
$\therefore$ The Codebook formed by **Tree 1** would have shorted encoded messages, averaged over many messages

(c) The Expected Length of the n length extension (not nth order extension since we are using the same Codebook) of a Codebook C, is given by

$$L(C^n) = nL(C) \tag{1.1.2}$$

This equation is valid since our code is memoryless, i.e, the occurence of each symbol in the message is independent of other symbols, so the expected length of each symbol within the extension would be the same, so,

$$\begin{aligned}
l(x^n) &= l(x_1) + l(x_2) + l(x_3) + \cdots + l(x_n) \\
E(l(x^n)) &= E(l(x_1) + l(x_2) + l(x_3) + \cdots + l(x_n)) \\
E(l(x^n)) &= E(l(x_1)) + E(l(x_2)) + E(l(x_3)) + \cdots + E(l(x_n)) \\
\implies L(C^n) &= L(C) + L(C) + \cdots n\ times \\
\implies L(C^n) &= nL(C)
\end{aligned}$$

Where,

$$\begin{aligned}
x^n &- \text{Message in the n length extension} \\
x_i &- \text{ith character of the message} \\
l(x) &- \text{Length of codeword of x} \\
E(x) &- \text{Expectation of x}
\end{aligned}$$

Therefore, for the 100 length extension of the Codebook formed by Tree 2,

$$\begin{aligned}
L(C^n) &= nL(C) \\
&= 100 \times 2.25, \text{ (from 1.1(b))} \\
\therefore L(C^{100}) &= 225
\end{aligned}$$

$\therefore$ The average encoded length of 100-symbol messages is **225**.

1.2

There are 8 possibilities for 3-bit numbers and they all have equal probability of occuring.

The original Entropy of this scenario, given no clues is,

$$\begin{aligned}
H(X) &= \sum_{x \in X} p_X(x) \log_2 \left(\frac{1}{p_X(x)} \right) \\
H(X) &= 8 \left(\frac{1}{8} \right) \log_2 (8) \quad \text{,(8 possibilities and all of them equal)} \\
\implies H(X) &= 3
\end{aligned}$$

Assume the Entropy of the scenario after getting the clue to be $H'(X)$, then the Information given by the clue will be,

$$I = H(X) - H'(X) \tag{1.2.1}$$

Since, $H'(X) \leqslant H(X)$, as the clue will reduce the uncertainity of the scenario, and that reduction is purely due to the Information contained within the clue (denoted by I). Therefore,

$$I = 3 - H'(X) \tag{1.2.2}$$

We will use Equation 1.2.2 in the following questions

(a) Given Clue: The number is odd
Given the clue, the possibilities can be: 001 (1), 011(3), 101(5), 111(7). So there are 4 possibilities with each having an equal probability of occuring. Therefore, the new Entropy is,

$$\begin{array}{rcl} H'(X) & = & 4\left(\frac{1}{4}\right)\log_2(4) \\ \implies H'(X) & = & 2 \end{array}$$

Using Equation 1.2.2,

$$\begin{array}{rcl} I & = & 3 - H'(X) \\ \implies I & = & 1\ bit \end{array}$$

Therefore, the information given by this clue is 1 bit.

(b) Given Clue: The number is not a multiple of 3 (0, 3, 6).
Given the clue, the possibilities can 1, 2, 4, 5, 7, 8. So, the new Entropy is,

$$\begin{array}{rcl} H'(X) & = & 6\left(\frac{1}{6}\right)\log_2(6) \\ \implies H'(X) & = & \log_2(6) \approx 2.585 \end{array}$$

Using Equation 1.2.2,

$$\begin{array}{rcl} I & = & 3 - 2.585 \\ \implies I & = & 0.415 \end{array}$$

Therefore, the information given by this clue is 0.415 bits.

(c) Given Clue: The number contains two 1's
Given the clue, the possibilities will be 011(3), 101(5), 110(6). So the new Entropy is,

$$\begin{aligned} H'(X) &= 3\left(\frac{1}{3}\right)\log_2(3) \\ \implies H'(X) &= \log_2(3) \approx 1.585 \end{aligned}$$

Using Equation 1.2.2,

$$\begin{aligned} I &= 3 - 1.585 \\ \implies I &= 1.415 \end{aligned}$$

Therefore, the information given by this clue is 1.415 bits.

(d) If we are given all the clues, the possibilities will be,
$\{1,3,5,7\} \cap \{1,2,4,5,7,8\} \cap \{3,5,6\} = \{5\}$ Since there is only one possibility,

$$\begin{aligned} H'(X) &= 1\log_2(1) \\ \implies H'(X) &= 0 \end{aligned}$$

Therefore, the information gained from the clues is,

$$\begin{aligned} I &= 3 - 0 \\ \implies I &= 3 \end{aligned}$$

The information gained from the clues is 3 bits.

1.3

Given symbols and their probabilities:

I	$p(I)$	$\log_2(1/p(I))$	$p(I) \cdot \log_2(1/p(I))$
A	0.22	2.18	0.48
E	0.34	1.55	0.53
I	0.17	2.57	0.43
O	0.19	2.40	0.46
U	0.08	3.64	0.29
Totals	**1.00**	**12.34**	**2.19**

Table 1: Table of probabilities and entropy calculations

From the table we see that $H(X) = 2.19$ bits

(a) Given: The symbol is either a I or U

The new Entropy due to this fact is:

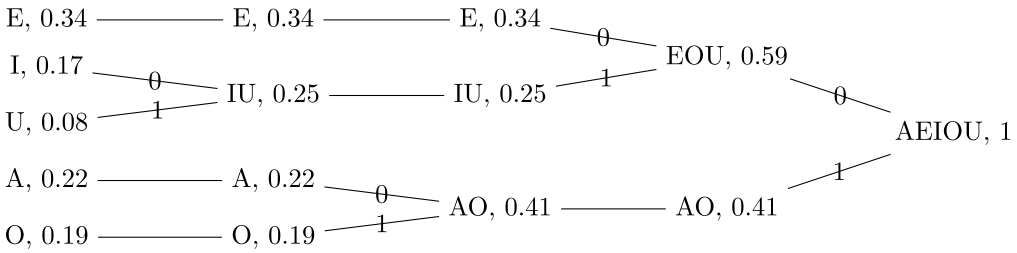
$$\begin{aligned} H'(X) &= p(I)\log_2\left(\frac{1}{p(I)}\right) + p(U)\log_2\left(\frac{1}{p(U)}\right) \\ &= 0.43 + 0.29 \\ \implies H'(X) &= 0.72 \end{aligned}$$

Using the Equation 1.2.1,

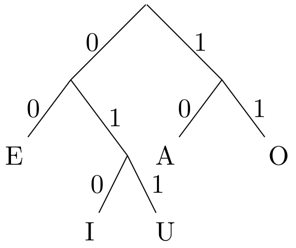
$$\begin{aligned} I &= H(X) - H'(X) \\ &= 2.19 - 0.72 \\ \implies I &= 1.47 \end{aligned}$$

The information gained by this fact is 1.47 bits.

(b) To get the Huffman Tree:



The Resultant Huffman Tree:



The Codewords will be:

A: 10

E: 00

I: 010

O: 11

U: 011

(c) We know that, from Equation 1.1.2,

$$L(C^n) = nL(C)$$

To find L(C),

$$\begin{aligned} L(C) &= \sum_{x \in X} l(x) p_X(x) \\ &= 2 \times 0.22 + 2 \times 0.34 + 2 \times 0.19 + 3 \times 0.17 + 3 \times 0.08 \\ \implies L(C) &= 2.25 \end{aligned}$$

Using Equation 1.1.2

$$\begin{aligned} L(C^{100}) &= 100L(C) \\ &= 100 \times 2.25 \\ \implies L(C^{100}) &= 225 \end{aligned}$$

(d) Ben's code encodes characters with an average length of 197 bits per 100 letters. That means,

$$\begin{aligned} L(C^{100}) &= 197 \\ \implies L(C^{100})/sym &= 1.97\ bits \end{aligned}$$

At first glance, this seems more efficient than the code obtained from our Huffman Tree. But, as per Source Coding Theorem.

$$L(C) < H(X) \implies P_e > 0, \text{(The converse statement of SCT)} \tag{1.3.1}$$

Where P_e is the probability of decoding error, Since we know H(X) to be 2.19 bits, clearly $L(C^{100})/sym > 2.19$. Therefore, **Ben's code will have some decoding error, hence it is not as robust as the Huffman code.**

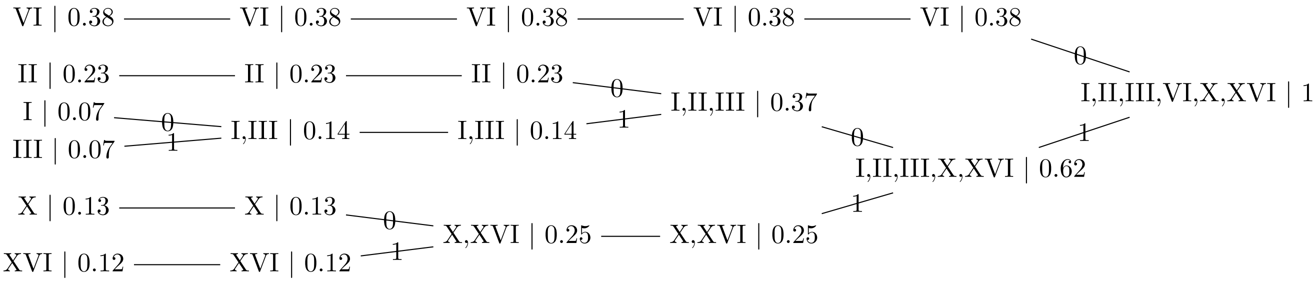
1.4

Course (Department)	**# of students**	**Prob.**
I (Civil & Env.)	121	0.07
II (Mech. Eng.)	389	0.23
III (Mat. Sci.)	127	0.07
VI (EECS)	645	0.38
X (Chem. Eng.)	237	0.13
XVI (Aero & Astro)	198	0.12
Total	**1717**	**1.0**

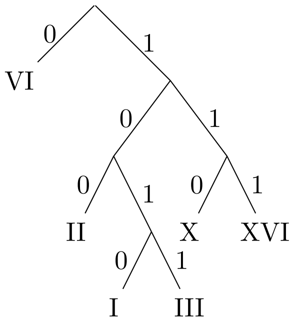
Table 2: Student Distribution by Course

(a) If we think of H(X) as the measure of uncertainity, then we can say that H(X) is **inversely related** to the amount of information provided. We know that the contribution of each component of H(X) is directly related to the probability of that component.
Therefore we can conclude that the amount of information contained in each component is inversely related to its probability. Using this result in the table, we can say that **Course VI(EECS) should contain the least information**, since it has the highest probability.

(b) Let I, II, III, VI, X, XVI be the representation for the courses (as per the given table). To design the Huffman Code,



The Resultant Huffman Tree:



The Codewords are,

- I: 1010
- II: 100
- III: 1011
- VI: 0
- X: 110
- XVI: 111

(c) We know that,

$$L(C^n) = nL(C)$$

From Equation 1.1.2. Also,

$$L(C) = \sum_{x \in X} l(x) p_X(x)$$

Using these, we can calculate $L(C^{100})$ as,

$$\begin{array}{rcl} L(C) & = & 0.07(4) + 0.23(3) + 0.07(4) + 0.38(1) + 0.13(3) + 0.12(3) \\ L(C) & = & 0.28 + 0.69 + 0.28 + 0.38 + 0.39 + 0.36 \\ \therefore L(C) & = & 2.38 \ bits \\ L(C^{100}) & = & 100L(C) \\ \therefore L(C^{100}) & = & 238 \ bits \end{array}$$

Therefore, the average encoded length of 100 length messages is **238 bits**.

2 Channel Coding

2.1

(a)

Channel Coding	**Source Coding**
Channel Coding is to encode binary strings for Error Handling in a noisy channel	Source Coding is to encode any string into binary strings for Data Compression with minimal loss of information
Usually occurs after Source Coding	Usually occurs before Channel Coding
Adds redundancy bits to the string, increasing the length.	Removes bits from the string, decreasing the length

(b) Generator Matrix:

Let $m_{1\times k}$ be any message vector and $v_{1\times n}$ be the corresponding codeword of the message in the Channel Code, st $dim(m) = k,\ dim(v) = n$, then the Generator Matrix is defined as the $k \times n$ matrix st,

$$m_{1\times k} \times G_{k\times n} = v_{1\times n},\ \forall\ m_{1\times k} \in F_2^k \tag{2.1.1}$$

Parity Matrix: Parity Matrix is defined as the matrix, whose rows correspond to the different parity check equations that define a Channel Code. Mathematically, H is termed as the Parity matrix of an LBC $C(n, k)$ if,

$$\forall\ \vec{v}_{1\times n} \in C,\ \vec{v} \cdot H_i^T = 0\ \forall\ i = 0, 1, 2, \dots, (n-k) - 1 \tag{2.1.2}$$

Where H_i represents the vector formed by the ith row of the matrix H.

2.2

We are given a binary repetition code of block length 5. The Codebook of the given code will be:

$$\begin{aligned} 0 &\rightarrow 000000 \\ 1 &\rightarrow 11111 \end{aligned}$$

This is an BLBC with $n = 5$ and $k = 1$. Therefore the order of the Generator matrix will be 1×5. The basis of the input vector space is 1. So the Generator matrix will be:

$$G_{1\times 5} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \end{bmatrix}$$

The no.of rows in the Parity matrix will be $n - k$, which is $5 - 1 = 4$ Using the Generator Matrix, we get the Parity check equations as:

$$\begin{aligned} v_0 \oplus v_1 &= 0 \\ v_0 \oplus v_2 &= 0 \\ v_0 \oplus v_3 &= 0 \\ v_0 \oplus v_4 &= 0 \end{aligned}$$

(The indices in this question are 0-indexed.)
Therefore the corresponding Parity Matrix is:

$$H_{1\times 5} = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 \end{bmatrix}$$

2.3

We are given a single parity check code of block length 6. The codewords of the basis vectors of the input vector space F_2^5, will be:

$$\begin{aligned} 10000 &\rightarrow 100001 \\ 01000 &\rightarrow 010001 \\ 00100 &\rightarrow 001001 \\ 00010 &\rightarrow 000101 \\ 00001 &\rightarrow 000011 \end{aligned}$$

Therefore, the Generator Matrix is,

$$G_{5\times 6} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 \end{bmatrix}$$

The no.of rows in the Parity matrix will be $n - k$, which is $6 - 5 = 1$ Using the Generator Matrix, we get the Parity check equations as:

$$v_0 \oplus v_1 \oplus v_2 \oplus v_3 \oplus v_4 \oplus v_5 = 0$$

Therefore, the Parity Check matrix is,

$$H = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix}$$

2.4

We are given a Hamming Code with m(number of parity bits) = 4, k(number of message bits) = 11.
In a Hamming Code, the ith parity bit covers the data bits whose ith binary digit is 1, that is,

- *The parity bit p_1 covers data bits with indices whose first bit (LSB) is 1.*
- *The parity bit p_2 covers data bits with indices whose second bit is 1.*
- *The parity bit p_3 covers data bits with indices whose third bit is 1.*
- *The parity bit p_4 covers data bits with indices whose fourth bit (MSB) is 1.*

(The indices in this and following questions are 1-indexed.)
Using the above points, we can construct the below parity equations that define the code:

$$\begin{aligned} p_1 &= v_1 \oplus v_3 \oplus v_5 \oplus v_7 \oplus v_9 \oplus v_{11} \\ p_2 &= v_2 \oplus v_3 \oplus v_6 \oplus v_7 \oplus v_{10} \oplus v_{11} \\ p_3 &= v_4 \oplus v_5 \oplus v_6 \oplus v_7 \\ p_4 &= v_8 \oplus v_9 \oplus v_{10} \oplus v_{11} \end{aligned}$$

Where

$$\begin{aligned} v_i &\quad - \quad \text{ith bit of message vector} \\ p_i &\quad - \quad \text{ith parity bit} \end{aligned}$$

Assign $v_{12}, v_{13}, v_{14}, v_{15}$ as the parity bits. Therefore, we get the Parity Check Equations of the Parity Matrix H as,

$$\begin{aligned} v_1 \oplus v_3 \oplus v_5 \oplus v_7 \oplus v_9 \oplus v_{11} \oplus v_{12} &= 0 \\ v_2 \oplus v_3 \oplus v_6 \oplus v_7 \oplus v_{10} \oplus v_{11} \oplus v_{13} &= 0 \\ v_4 \oplus v_5 \oplus v_6 \oplus v_7 \oplus v_{14} &= 0 \\ v_8 \oplus v_9 \oplus v_{10} \oplus v_{11} \oplus v_{15} &= 0 \end{aligned}$$

Where

$$v_i \quad - \quad \text{ith bit of the codeword}$$

The corresponding Parity Matrix will be,

$$H_{4\times15} = \left[\begin{array}{ccccccccccc|cccc} 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 \end{array}\right]$$

$$(H = \left[P^T | I_{4\times4}\right])$$

(P is the matrix containing the parity bits of the basis vectors of the input space. Will be defined in the Generator Matrix)

Applying the parity bits to the basis of the input vector space, F_2^{11}, we get,

$$\begin{array}{rcl}
10000000000 & \rightarrow & 100000000001000 \\
01000000000 & \rightarrow & 010000000000100 \\
00100000000 & \rightarrow & 001000000001100 \\
00010000000 & \rightarrow & 000100000000010 \\
00001000000 & \rightarrow & 000010000001010 \\
00000100000 & \rightarrow & 000001000000110 \\
00000010000 & \rightarrow & 000000100001110 \\
00000001000 & \rightarrow & 000000010000001 \\
00000000100 & \rightarrow & 000000001001001 \\
00000000010 & \rightarrow & 000000000100101 \\
00000000001 & \rightarrow & 000000000011101
\end{array}$$

Therefore the Generator Matrix will be,

$$G_{11\times15} = \left[\begin{array}{ccccccccccc|cccc}
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1
\end{array}\right]$$

$$(G = [I_{11\times11}|P])$$

2.5

The given Generator Matrix is:

$$
\begin{aligned}
G &= [I_5|1] \\
\implies G_{5\times 6} &= \left[\begin{array}{ccccc|c}
1 & 0 & 0 & 0 & 0 & 1 \\
0 & 1 & 0 & 0 & 0 & 1 \\
0 & 0 & 1 & 0 & 0 & 1 \\
0 & 0 & 0 & 1 & 0 & 1 \\
0 & 0 & 0 & 0 & 1 & 1
\end{array}\right]
\end{aligned}
$$

Since the rows of the Identity Matrix in G, are a basis set in the messages's vector space, the length of the original message must be 5 and the final length of the encoded message must be 6.

(a) The rate of a Channel Code is given by,

$$R = \frac{\text{Original Length}}{\text{Encoded Length}}$$

Using this equation we get,

$$
\begin{aligned}
R &= \frac{5}{6} \\
\implies R &= 0.833
\end{aligned}
$$

(b) The given Generator Matrix represents a SPC-6 Code, ie, the parity bit of a message $v_{1\times 5}$ is such that,

$$p = v_1 \oplus v_2 \oplus v_3 \oplus v_4 \oplus v_5$$

For the message 10010, we get the parity bit as,

$$p = 1 \oplus 0 \oplus 0 \oplus 1 \oplus 0 \ = \ 0$$

Therefore the Codeword is **100100**.

(c) The Parity Check Matrix is:

$$v_1 \oplus v_2 \oplus v_3 \oplus v_4 \oplus v_5 \oplus v_6 = 0 \ \forall \ v_{1\times 6} \in C$$

Where C represents the SPC-6 Codebook.

Therefore, the Parity Check Matrix is given as,

$$H_{1\times 6} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix} \tag{2.5.1}$$

2.6

The given Generator Matrix is,

$$G_{3\times 7} = \left[\begin{array}{ccc|cccc} 1 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 \end{array}\right]$$

From the Generator matrix, we can see that $k = 3$ and $n = 7$. Since for a given BLBC, there will be $n-k$ parity check equations, the given BLBC will have $7-3 = 4$ parity check equations. Since each column of P contains information about each parity bit, looking at the columns of P and the corresponding original message vector, we can derive the following Parity Check Equations,

$$\begin{aligned} m_1 \oplus m_2 = p_1 &\implies v_1 \oplus v_2 \oplus v_4 = 0 \\ m_2 \oplus m_3 = p_2 &\implies v_2 \oplus v_3 \oplus v_5 = 0 \\ m_1 \oplus m_3 = p_3 &\implies v_1 \oplus v_3 \oplus v_6 = 0 \\ m_1 \oplus m_2 = p_4 &\implies v_1 \oplus v_2 \oplus v_7 = 0 \end{aligned}$$

Where

$$\begin{aligned} m_i &\quad - \quad \text{ith bit of message vector} \\ p_i &\quad - \quad \text{ith parity bit} \\ v_i &\quad - \quad \text{ith bit of the codeword} \end{aligned}$$

This set of equations will give us the Parity Check Matrix,

$$H_{4\times 7} = \left[\begin{array}{ccc|cccc} 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 1 \end{array}\right]$$

Also, since the given Generator matrix is Systematic, (is of the form $[I|P]$), we can directly write the Parity Check Matrix as H = $[P^T|I]$. Therefore, H will be,

$$H_{4\times 7} = \left[\begin{array}{ccc|cccc} 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 1 \end{array}\right]$$

Which matches with our previous result.