Problem 1:

(a) It can be indeed observed that

\[ wt(x + y) = (\text{# of 1's in } x) + (\text{# of 1's in } y) + 2(\text{# of places where both } x \text{ and } y \text{ have a 1}). \]

Thus

\[ wt(x + y) = wt(x) + wt(y) - 2wt(xy). \]

A formal proof goes as follows. Letting \( I \triangleq \{1, 2, \ldots, n\} \), define the sets \( A \) and \( B \) as

\[ A \triangleq \{i \in I : x_i = 1\}, \]

and

\[ B \triangleq \{i \in I : y_i = 1\}. \]

Then \( |A| = wt(x) \) and \( |B| = wt(y) \). Furthermore,

\[ A \cap B = \{i \in I : x_i = 1 \text{ and } y_i = 1\}, \]

\[ A \setminus B = \{i \in I : x_i = 1 \text{ and } y_i = 0\}, \]

\[ B \setminus A = \{i \in I : y_i = 1 \text{ and } x_i = 0\}, \]

and

\[ (A \setminus B) \cup (B \setminus A) = \{i \in I : x_i + y_i = 1\}. \]

Thus

\[ |A \cap B| = wt(xy), \]

and

\[ |(A \setminus B) \cup (B \setminus A)| = |A \setminus B| + |B \setminus A| = wt(x + y). \]

Now, observe that \( A \) and \( B \) can be written as the following disjoint unions:

\[ A = (A \setminus B) \cup (A \cap B), \]

and

\[ B = (B \setminus A) \cup (A \cap B). \]
and

\[ B = (B \setminus A) \cup (A \cap B). \]

Hence

\[ |A| = |A \setminus B| + |A \cap B| \iff wt(x) = |A \setminus B| + wt(xy) \quad (1) \]
\[ |B| = |B \setminus A| + |A \cap B| \iff wt(y) = |B \setminus A| + wt(xy). \quad (2) \]

Adding equations (1) and (2) yields

\[ wt(x) + wt(y) = |A \setminus B| + |B \setminus A| + 2wt(xy), \]

or

\[ wt(x) + wt(y) = wt(x + y) + 2wt(xy). \]

Therefore

\[ wt(x + y) = wt(x) + wt(y) - 2wt(xy). \]

\[ \square \]

(b) From (a) we have

\[ wt(x) + wt(y) = wt(x + y) + 2wt(xy). \]

But \( wt(xy) \geq 0 \), thus

\[ wt(x) + wt(y) \geq wt(x + y). \]

\[ \square \]

We have equality iff \( wt(xy) = 0 \). This is equivalent to \( |A \cap B| = 0 \); that is \( A \cap B = \emptyset \).

Therefore to show that a necessary and sufficient condition for equality is that \( y_i = 0 \) whenever \( x_i = 1 \), we need to prove that

\[ A \cap B = \emptyset \iff x_i = 1 \iff y_i = 0. \]

If \( A \cap B = \emptyset \), then clearly \( x_i = 1 \implies y_i = 0 \). Conversely, if we have that \( x_i = 1 \implies y_i = 0 \), then \( i \in A \cap B \) is impossible (since we cannot have \( x_i = y_i = 1 \)); so \( A \cap B = \emptyset \).

\[ \square \]

(d) We have:

\[ d(x, z) = wt(x + z) = wt(x + y + y + z) \quad (\text{since } y + y = 0^a \text{ in } B^n) = wt((x + y) + (y + z)) \]
\[
\leq \ wt(x + y) + wt(y + z) \quad \text{(by (b))}
\]
\[
= \ d(x, y) + d(y, z).
\]

(c) Given \(x\) and \(y\), let \(u = x + y\). Then the triangle inequality of (b) states that
\[
wt(y + u) \leq wt(y) + wt(u).
\]
But \(y + u = y + (x + y) = x\); thus \(wt(y + u) = wt(x)\). Therefore, the above expression becomes
\[
wt(x) \leq wt(y) + wt(x + y),
\]
or
\[
wt(x + y) \geq wt(x) - wt(y).
\]

From part (b), we know that we have equality iff \(y_i = 1 \implies u_i = 0\); that is \(y_i = 1 \implies x_i + y_i = 0\); that is \(y_i = 1 \implies x_i = 1\).

Problem 2:
First note that \(d_{\text{min}} = 3\) for \(C\); so the code can correct one error per codeword. The distance computations \((d(w, c), c \in C)\) for the nearest neighbor decoding rule yield the following:

<table>
<thead>
<tr>
<th>codewords (c \in C)</th>
<th>((a))</th>
<th>((b))</th>
<th>((c))</th>
<th>((d))</th>
</tr>
</thead>
<tbody>
<tr>
<td>(w = 0110101)</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>(w = 0101110)</td>
<td>5</td>
<td>1</td>
<td>5</td>
<td>3</td>
</tr>
<tr>
<td>(w = 1011001)</td>
<td>2</td>
<td>6</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>(w = 1100110)</td>
<td>3</td>
<td>5</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>(w = 1011010)</td>
<td>6</td>
<td>4</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>(w = 0100101)</td>
<td>1</td>
<td>3</td>
<td>5</td>
<td>3</td>
</tr>
<tr>
<td>(w = 0001111)</td>
<td>4</td>
<td>2</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>(w = 1111111)</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
</tbody>
</table>

(a) Decode \(w = 0110101\) into \(0100101\).

(b) Decode \(w = 0101110\) into \(0101010\).
(c) In this case, there is a tie. There are two nearest neighbors to \( w \): 1010101 and 1011010 (at distance 2). Decode \( w \) into one of them at random. (In this case, at least two errors occurred; so there is no guarantee that the code will yield a correct decision, as indicated by the tie).

(d) Here also, there is a tie. There are four nearest neighbors to \( w \) (at distance 3): 0101010, 1110000, 0100101 and 1111111. Decode \( w \) into one of them at random.

Problem 3:

• \( C_1 \) is a (5,2) group code with

\[
G_1 = \begin{bmatrix}
1 & 1 & 1 & 1 & 0 \\
0 & 0 & 1 & 1 & 1
\end{bmatrix}.
\]

Its codewords \( c \) can be obtained from the operation \( c = uG \), for each message word \( u \in B^2 \). This yields

\[
C_1 = \{00000, 00111, 11110, 11001\}.
\]

The minimum distance \( d_{\text{min}} = 3 \). So \( C_1 \) can detect up to two errors per codeword. It also can correct one error per codeword. (Note that \( C_1 \) is not a systematic group code; however it can be shown that we can always obtain a systematic group code \( C \) from \( C_1 \) by performing elementary row operations and column permutations on \( G_1 \). The systematic code \( C \) will have the same minimum distance as \( C_1 \), and thus yield a similar performance).

• \( C_2 \) is a (7,3) systematic group code with

\[
G_2 = \begin{bmatrix}
1 & 0 & 0 & 1 & 1 & 0 & 1 \\
0 & 1 & 0 & 1 & 0 & 1 & 1 \\
0 & 0 & 1 & 0 & 1 & 1 & 1
\end{bmatrix}.
\]

Thus

\[
C_2 = \{0000000, 0010111, 0101011, 0111000, 1001101, 1011010, 1100110, 1110001\}.
\]

The minimum distance \( d_{\text{min}} = 4 \). So \( C_2 \) can detect three errors per codeword; it also can correct one error per codeword.
Problem 4:

The (6,3) group code with

\[ G = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 \end{bmatrix} \]

has the following codebook:

\[ C = \{000000, 001001, 010110, 011111, 100101, 101100, 110011, 111010\}. \]

The code has \(2^6 - 3 = 8\) cosets. The coset decoding table is as follows.

<table>
<thead>
<tr>
<th>(C)</th>
<th>000000</th>
<th>001001</th>
<th>010110</th>
<th>011111</th>
<th>100101</th>
<th>101100</th>
<th>110011</th>
<th>111010</th>
</tr>
</thead>
<tbody>
<tr>
<td>(C + z_1)</td>
<td>000001</td>
<td>001000</td>
<td>010111</td>
<td>011110</td>
<td>100100</td>
<td>101101</td>
<td>110010</td>
<td>111011</td>
</tr>
<tr>
<td>(C + z_2)</td>
<td>000010</td>
<td>001011</td>
<td>010100</td>
<td>011101</td>
<td>100111</td>
<td>101110</td>
<td>110001</td>
<td>111000</td>
</tr>
<tr>
<td>(C + z_3)</td>
<td>000100</td>
<td>001101</td>
<td>010010</td>
<td>011011</td>
<td>100001</td>
<td>101000</td>
<td>110111</td>
<td>111110</td>
</tr>
<tr>
<td>(C + z_4)</td>
<td>010000</td>
<td>011001</td>
<td>000110</td>
<td>001111</td>
<td>110101</td>
<td>111100</td>
<td>100011</td>
<td>101010</td>
</tr>
<tr>
<td>(C + z_5)</td>
<td>100000</td>
<td>101001</td>
<td>110110</td>
<td>111111</td>
<td>000101</td>
<td>001100</td>
<td>010011</td>
<td>011010</td>
</tr>
<tr>
<td>(C + z_6)</td>
<td>000011</td>
<td>001010</td>
<td>010101</td>
<td>011100</td>
<td>100110</td>
<td>101111</td>
<td>110000</td>
<td>111001</td>
</tr>
<tr>
<td>(C + z_7)</td>
<td>011000</td>
<td>010001</td>
<td>001110</td>
<td>000111</td>
<td>111101</td>
<td>110100</td>
<td>101011</td>
<td>100010</td>
</tr>
</tbody>
</table>

The words in the second column are the coset leaders.

Problem 5:

(a) The generator matrix \(G\) of this (6,3) code is:

\[ G = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 \end{bmatrix} \]

(b)

<table>
<thead>
<tr>
<th>Message (u \in B^3)</th>
<th>000</th>
<th>001</th>
<th>010</th>
<th>011</th>
<th>100</th>
<th>101</th>
<th>110</th>
<th>111</th>
</tr>
</thead>
<tbody>
<tr>
<td>Codeword (c = uG)</td>
<td>000000</td>
<td>001011</td>
<td>010110</td>
<td>011101</td>
<td>100101</td>
<td>101100</td>
<td>110011</td>
<td>111000</td>
</tr>
</tbody>
</table>

(c) The code’s minimum distance is \(d = 3\). Thus the code can detect up to two errors per codeword, and it can correct one error per codeword.
(d) The syndrome decoding table is as follows.

<table>
<thead>
<tr>
<th>Syndrome $s = zH$</th>
<th>000</th>
<th>001</th>
<th>010</th>
<th>100</th>
<th>011</th>
<th>110</th>
<th>101</th>
<th>111</th>
</tr>
</thead>
<tbody>
<tr>
<td>Coset leader $z$</td>
<td>000000</td>
<td>000001</td>
<td>000010</td>
<td>000100</td>
<td>001000</td>
<td>010000</td>
<td>100000</td>
<td>010001</td>
</tr>
</tbody>
</table>

Note that other permissible candidates for the coset leader of the last syndrome ($s = 111$) are: 100010 or 001100.

Using the above decoding table, the decoded sequence will be:

111000, 110011, 011101, 011101, 101110, 000000, 010110, 100101.