1.

Theorem: Let A be the adjacency matrix of a graph. Then $(A^\ell)_{ij}$ is the number of walks of length ℓ starting at v_i and ending at v_j.

According to the theorem, for every integer $\ell \geq 1$, $(A^\ell)_{ij} = \#\{\text{walks of length } \ell \text{ from } v_i \text{ to } v_j\}$.

Definition: The distance $d(v_i, v_j)$ between nodes v_i and v_j in a graph is the smallest length of any walk connecting v_i and v_j.

According to the definition and theorem,

$$d(v_i, v_j) \;=\; \begin{cases} 0, & \text{if } i = j, \\ \min\left\{\ell \geq 1 : (A^\ell)_{ij} > 0\right\} & \text{if } i \neq j \end{cases}$$

Therefore, to find $d(v_i, v_j)$, we compute A^ℓ for $\ell \geq 1$ and check the (i, j)-entry, continuing to increment ℓ until the condition is met which is the smallest ℓ for which $(A^\ell)_{ij} > 0$.

Example: Consider the graph on vertices $\{1, 2, 3, 4\}$ with adjacency matrix

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

For $(i, j) = (1, 3)$ $A_{13} = 0$ (no walk of length $\ell = 1$ exists) then increment ℓ

$$(A^{\ell+1})_{ij} = \sum_{k=1}^{n} (A^\ell)_{ik}\, A_{kj}, \qquad A^1 = A, \quad n = 4,$$

$$A^2 = A \cdot A, \qquad (A^2)_{ij} = \sum_{k=1}^{4} A_{ik}\, A_{kj}$$

$$A^2 = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 2 & 0 & 1 \\ 1 & 0 & 2 & 0 \\ 0 & 1 & 0 & 1 \end{bmatrix}$$

$$d(v_1, v_3) \;=\; \min\left\{\ell \geq 1 : (A^\ell)_{13} > 0\right\}.$$

Therefore, since $(A^2)_{13} = 1 > 0$, there exists a walk of length $\ell = 2$ between v_1 and v_3. Hence, $d(1, 3) = 2$. This matches the path $1 \to 2 \to 3$, which has length 2 (two edges).

2.

Definition: The degree of v_i is the number of edges connected to v_i.

The degree of v_i is the number of distinct neighbors that v_i is connected to. Thus, counting the edges adjacent to v_i is the same as counting the number of neighbors of v_i.

A graph G of order n with nodes $V_G = \{v_1, v_2, \ldots, v_n\}$ corresponds to a unique symmetric $n \times n$ adjacency matrix A_G.

$$(A_G^2)_{ii} \;=\; \sum_{k=1}^{n} (A_G)_{ik}\,(A_G)_{ki}$$

Assuming G is a simple, undirected, loopless graph, A_G is symmetric, so $(A_G)_{ik} = (A_G)_{ki} \in \{0, 1\}$. Hence

$$(A_G^2)_{ii} \;=\; \sum_{k=1}^{n} (A_G)_{ik}^2$$

As $x^2 = x$ for $x \in \{0, 1\}$. Therefore

$$(A_G^2)_{ii} \;=\; \sum_{k=1}^{n} (A_G)_{ik}$$

As $(A_G)_{ik} = 1$ when v_i is adjacent to v_k, this sum counts the number of neighbors of v_i (adjacent nodes). By definition, this equals the degree of v_i, i.e.

$$\sum_{k=1}^{n} (A_G)_{ik} = \deg(v_i)$$

Therefore,

$$(A_G^2)_{ii} \;=\; \deg(v_i)$$

3.

Assuming G is a simple, undirected, loopless graph.

a.

Definition: A complete graph on n nodes is a graph with edges joining every pair of nodes.

According to the definition, in a complete graph each node is adjacent to all other $n-1$ nodes. This essentially means that no self-loops are present: there are edges between each node v_i and all other adjacent $n-1$ nodes, but no edges to itself. Therefore, the adjacency matrix of the complete graph is

$$A_G = J - I,$$

where J is the $n \times n$ all-ones matrix and I is the $n \times n$ identity matrix. Subtracting I ensures that the diagonal entries are zero, i.e. $(A_G)_{ii} = 0$, so that self-loops are not present in the complete graph adjacency matrix.

$$A_G = \begin{bmatrix} 0 & 1 & 1 & \cdots & 1 \\ 1 & 0 & 1 & \cdots & 1 \\ 1 & 1 & 0 & \cdots & 1 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & 1 & 1 & \cdots & 0 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ 1 & 1 & 1 & \cdots & 1 \\ 1 & 1 & 1 & \cdots & 1 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & 1 & 1 & \cdots & 1 \end{bmatrix} - \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \end{bmatrix}$$

Therefore,

$$(A_G)_{ij} = (A_G)_{ji} = \begin{cases} 1, & \text{if } i \neq j \quad \text{(every distinct pair of nodes is adjacent)} \\ 0, & \text{if } i = j \quad \text{(no self-loops)} \end{cases}$$

b.

To find the number of walks of length $\ell = 2$ between any two nodes i and j in a graph of n nodes, we use

$$(A^2)_{ij} = \sum_{k=1}^{n} (A)_{ik}\, A_{kj}$$

This sum counts the number of walks of the form $i \to k \to j$, since a walk of length 2 must pass through an intermediate node k with $k \in \{1, 2, \ldots, n\}$.

In a complete graph without self-loops, each node v_i is adjacent to all other $n-1$ nodes but not to itself, so a valid 2-walk $i \to k \to j$ must satisfy

$$i \neq j, \qquad k \neq i, \qquad k \neq j$$

Since self-loops are not allowed, we must have $k \neq i, j$. Thus the intermediate node k can be chosen from $k \in \{1, 2, \ldots, n\} \setminus \{i, j\}$. Therefore, the number of walks of length $\ell = 2$ between any two distinct vertices i and j in a complete graph is $n - 2$.

c.

To find the number of walks of length $\ell = 3$ between any two vertices i and j in a graph of n nodes, we use

$$(A^3)_{ij} = \sum_{a=1}^{n} \sum_{b=1}^{n} (A)_{ia}\, (A)_{ab}\, (A)_{bj}$$

This double sum counts the number of walks of the form $i \to a \to b \to j$ since a walk of length 3 must pass through two intermediate nodes a and b, with $a, b \in \{1, 2, \ldots, n\}$.

In a complete graph without self-loops, each node v_i is adjacent to all other $n-1$ vertices, but not to itself. For a walk of the form $i \to a \to b \to j$, this means that $a \neq i$, $b \neq j$, and $a \neq b$. Since the graph is complete, any two distinct vertices are adjacent, so the condition that a and b are connected is automatically satisfied whenever $a \neq b$. Therefore, a valid 3-walk $i \to a \to b \to j$ must satisfy

$$i \neq j, \quad a \neq i, \quad b \neq j, \quad a \neq b$$

Case 1: Let $a = j$. The walk from i to j has the form $i \to j \to b \to j$

Since self-loops are not allowed, we must have $b \neq j$. Thus the intermediate node b can be chosen from $b \in \{1, 2, \ldots, n\} \setminus \{j\}$ which gives $n - 1$ possible choices for b.

Case 2: Let $a \neq j$. The walk from i to j has the form $i \to a \to b \to j$

Since self-loops are not allowed, we must have $a \neq i$. As $a \neq j$, therefore the intermediate node a can be chosen from $a \in \{1, 2, \ldots, n\} \setminus \{i, j\}$ which gives $n - 2$ possible choices for a.

For each a, the adjacent node b must satisfy $b \neq a$ and $b \neq j$, therefore the intermediate node b can be chosen from $b \in \{1, 2, \ldots, n\} \setminus \{a, j\}$ which gives $n - 2$ possible choices for b.

Therefore, the number of walks of length $\ell = 3$ between any two distinct vertices i and j in a complete graph is $(n-1) + (n-2)(n-2) = n^2 - 3n + 3$.

4.

The degree of each vertex in G:

$$\deg_G(v_1) = 2\ (v_2, v_3), \quad \deg_G(v_2) = 3\ (v_1, v_3, v_4), \quad \deg_G(v_3) = 3\ (v_1, v_2, v_6),$$

$$\deg_G(v_4) = 2\ (v_2, v_6), \quad \deg_G(v_5) = 2\ (v_6, v_8), \quad \deg_G(v_6) = 5\ (v_3, v_4, v_5, v_7, v_8),$$

$\deg_G(v_7) = 2\ (v_6, v_8), \quad \deg_G(v_8) = 3\ (v_5, v_6, v_7).$

The degree of each vertex in H is:

$\deg_H(u_1) = 3\ (u_3, u_6, u_7), \quad \deg_H(u_2) = 3\ (u_4, u_5, u_8), \quad \deg_H(u_3) = 2\ (u_1, u_6),$

$\deg_H(u_4) = 2\ (u_2, u_6), \quad \deg_H(u_5) = 3\ (u_2, u_6, u_8), \quad \deg_H(u_6) = 5\ (u_1, u_3, u_4, u_5, u_7),$

$\deg_H(u_7) = 2\ (u_1, u_6), \quad \deg_H(u_8) = 2\ (u_2, u_5).$

Thus,

$$\text{degree multiset}(G) = \{2, 2, 2, 2, 3, 3, 3, 5\}, \qquad \text{degree multiset}(H) = \{2, 2, 2, 2, 3, 3, 3, 5\}.$$

Since G and H have identical degree multisets, the degree test confirms they can be isomorphic.

$$V_G = \{v_1, v_2, v_3, v_4, v_5, v_6, v_7, v_8\}, \qquad V_H = \{u_1, u_2, u_3, u_4, u_5, u_6, u_7, u_8\},$$

Claim The map $\phi : V_G \to V_H$ given by

$$\phi(v_1) = u_8, \quad \phi(v_2) = u_2, \quad \phi(v_3) = u_5, \quad \phi(v_4) = u_4, \quad \phi(v_5) = u_7, \quad \phi(v_6) = u_6, \quad \phi(v_7) = u_3, \quad \phi(v_8) = u_1$$

is an isomorphism

Proof. ϕ is a bijection. Checking edge preservation under ϕ:

$\{v_1, v_2\} \mapsto \{\phi(v_1), \phi(v_2)\} = \{u_8, u_2\} \in E_H$

$\{v_1, v_3\} \mapsto \{\phi(v_1), \phi(v_3)\} = \{u_8, u_5\} \in E_H$

$\{v_2, v_3\} \mapsto \{\phi(v_2), \phi(v_3)\} = \{u_2, u_5\} \in E_H$

$\{v_2, v_4\} \mapsto \{\phi(v_2), \phi(v_4)\} = \{u_2, u_4\} \in E_H$

$\{v_3, v_6\} \mapsto \{\phi(v_3), \phi(v_6)\} = \{u_5, u_6\} \in E_H$

$\{v_4, v_6\} \mapsto \{\phi(v_4), \phi(v_6)\} = \{u_4, u_6\} \in E_H$

$\{v_5, v_6\} \mapsto \{\phi(v_5), \phi(v_6)\} = \{u_7, u_6\} \in E_H$

$\{v_5, v_8\} \mapsto \{\phi(v_5), \phi(v_8)\} = \{u_7, u_1\} \in E_H$

$\{v_6, v_7\} \mapsto \{\phi(v_6), \phi(v_7)\} = \{u_6, u_3\} \in E_H$

$\{v_6, v_8\} \mapsto \{\phi(v_6), \phi(v_8)\} = \{u_6, u_1\} \in E_H$

$\{v_7, v_8\} \mapsto \{\phi(v_7), \phi(v_8)\} = \{u_3, u_1\} \in E_H$

Thus, under the mapping ϕ, every edge of G corresponds to an edge of H, so adjacency is preserved:

$$\{v_i, v_j\} \in E_G \iff \{\phi(v_i), \phi(v_j)\} \in E_H,$$

showing that ϕ is an isomorphism between G and H.

Permutation matrix P corresponding to ϕ:

$P =$

	u_1	u_2	u_3	u_4	u_5	u_6	u_7	u_8
v_1	0	0	0	0	0	0	0	1
v_2	0	1	0	0	0	0	0	0
v_3	0	0	0	0	1	0	0	0
v_4	0	0	0	1	0	0	0	0
v_5	0	0	0	0	0	0	1	0
v_6	0	0	0	0	0	1	0	0
v_7	0	0	1	0	0	0	0	0
v_8	1	0	0	0	0	0	0	0

The adjacency matrix for H:

$$A_H = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}$$

$P\,A_H\,P^\top$ is divided into $R = P\,A_H$ and $S = R\,P^\top$

P is a permutation matrix with $P_{i,\phi(i)} = 1$ and all other entries 0, we have $R_{ij} = (A_H)_{\phi(i),j}$for each i:

$$\begin{aligned}
&P_{1,\phi(1)} = 1, \quad \phi(1) = 8 \quad \Rightarrow R_{1j} = (A_H)_{8j} \\
&P_{2,\phi(2)} = 1, \quad \phi(2) = 2 \quad \Rightarrow R_{2j} = (A_H)_{2j} \\
&P_{3,\phi(3)} = 1, \quad \phi(3) = 5 \quad \Rightarrow R_{3j} = (A_H)_{5j} \\
&P_{4,\phi(4)} = 1, \quad \phi(4) = 4 \quad \Rightarrow R_{4j} = (A_H)_{4j} \\
&P_{5,\phi(5)} = 1, \quad \phi(5) = 7 \quad \Rightarrow R_{5j} = (A_H)_{7j} \\
&P_{6,\phi(6)} = 1, \quad \phi(6) = 6 \quad \Rightarrow R_{6j} = (A_H)_{6j} \\
&P_{7,\phi(7)} = 1, \quad \phi(7) = 3 \quad \Rightarrow R_{7j} = (A_H)_{3j} \\
&P_{8,\phi(8)} = 1, \quad \phi(8) = 1 \quad \Rightarrow R_{8j} = (A_H)_{1j}
\end{aligned}$$

$$R = P\,A_H = \begin{bmatrix} 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 \end{bmatrix}$$

$P^\top$ is a permutation matrix with $(P^\top)_{\phi(j),j} = 1$, we have $S_{ij} = R_{i,\phi(j)}$ for each j:

$$\begin{aligned}
&(P^\top)_{\phi(1),1} = 1, \quad \phi(1) = 8 \quad \Rightarrow S_{i1} = R_{i,8} \\
&(P^\top)_{\phi(2),2} = 1, \quad \phi(2) = 2 \quad \Rightarrow S_{i2} = R_{i,2} \\
&(P^\top)_{\phi(3),3} = 1, \quad \phi(3) = 5 \quad \Rightarrow S_{i3} = R_{i,5} \\
&(P^\top)_{\phi(4),4} = 1, \quad \phi(4) = 4 \quad \Rightarrow S_{i4} = R_{i,4} \\
&(P^\top)_{\phi(5),5} = 1, \quad \phi(5) = 7 \quad \Rightarrow S_{i5} = R_{i,7} \\
&(P^\top)_{\phi(6),6} = 1, \quad \phi(6) = 6 \quad \Rightarrow S_{i6} = R_{i,6} \\
&(P^\top)_{\phi(7),7} = 1, \quad \phi(7) = 3 \quad \Rightarrow S_{i7} = R_{i,3} \\
&(P^\top)_{\phi(8),8} = 1, \quad \phi(8) = 1 \quad \Rightarrow S_{i8} = R_{i,1}
\end{aligned}$$

$$S = R\,P^\top = \begin{bmatrix} 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 \end{bmatrix} \qquad A_G = \begin{bmatrix} 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 \end{bmatrix}$$

Therefore as $S = A_G$,

$$A_G = P\,A_H\,P^\top$$

5.

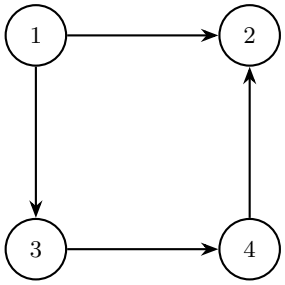
$$Q = \begin{bmatrix} -1 & 1 & 0 & 0 \\ -1 & 0 & 1 & 0 \\ 0 & 1 & 0 & -1 \\ 0 & 0 & -1 & 1 \end{bmatrix}$$

$e_1 : 1 \to 2$ contributes -1 in column 1 and $+1$ in column 2

$e_2 : 1 \to 3$ contributes -1 in column 1 and $+1$ in column 3

$e_3 : 4 \to 2$ contributes -1 in column 4 and $+1$ in column 2

$e_4 : 3 \to 4$ contributes -1 in column 3 and $+1$ in column 4



6.

Q is the edge–node incidence matrix of a graph G with n vertices and k connected components. Each row of Q corresponds to an edge, and each column corresponds to a vertex. Each row of Q has exactly one entry -1 (for the source of the edge), one entry $+1$ (for the destination of the edge), and zeros elsewhere.

If Q is multiplied by a column vector where every entry is the same constant value, for example:

$$x = \begin{bmatrix} c \\ c \\ \vdots \\ c \end{bmatrix}$$

then, in every row of Q,

$$(\text{row}) \cdot x = (+1) \cdot c + (-1) \cdot c = 0.$$

So every row gives 0, which means $Qx = 0$. Hence, any constant vector $x = (c, c, c, c)^\top \text{or} [1, 1, \ldots, 1]^\top$ satisfies $Q[1, 1, \ldots, 1]^\top = 0$.

Let

$$x = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}$$

represent a vector of vertex potentials.

If $Qx = 0$, then for each edge (i, j), $x_j - x_i = 0$. This means that the potential values at the two endpoints of every edge are equal. Therefore, all vertices within the same connected component share the same potential value.

Vertices in different connected components may have different constant values. Therefore, x is constant on each connected component.

The nullspace (or kernel) of a matrix Q is the set of all vectors x such that $Qx = 0$. That is,

$$\text{Null}(Q) = \{\, x \in \mathbb{R}^n \mid Qx = 0 \,\}.$$

Generalizing for k connected components: since x is constant on each component, the nullspace consists of k independent vectors, one for each connected component.

For the incidence matrix Q,

$$\text{Null}(Q) = \{\, x \in \mathbb{R}^n \mid x \text{ is constant on each connected component of the graph} \,\}$$

Thus,

$$\text{Nullity}(Q) = k, \qquad \text{Rank}(Q) = n - k$$

7.

a.

For the Hamming $(7, 4)$ code:

$$(n, k) = (7, 4)$$

$$G = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix}$$

A k-bit message $m \in \mathbb{F}_2^{1\times 4}$ is encoded as $c = mG \in \mathbb{F}_2^{1\times 7}$ (all arithmetic mod over $\mathbb{F}_2$).

Received message with blocks:

$$\{\, 0100110,\ 1100110,\ 0100110,\ 1111111,\ 0100110,\ 1110100,\ 0100110,\ 0101101 \,\}$$

Removing duplicates from received message:

$$\{0100110,\ 1100110,\ 1111111,\ 1110100,\ 0101101\}$$

Encoding process for the specific blocks used in the received message:

$$\begin{aligned}
m &= [0\ 1\ 0\ 0] \quad \Rightarrow c = [0\ 1\ 0\ 0]G = [0\ 1\ 0\ 0\ 1\ 1\ 0] \\
m &= [1\ 1\ 0\ 0] \quad \Rightarrow c = [1\ 1\ 0\ 0]G = [1\ 1\ 0\ 0\ 0\ 1\ 1] \\
m &= [1\ 1\ 1\ 1] \quad \Rightarrow c = [1\ 1\ 1\ 1]G = [1\ 1\ 1\ 1\ 1\ 1\ 1] \\
m &= [1\ 1\ 1\ 0] \quad \Rightarrow c = [1\ 1\ 1\ 0]G = [1\ 1\ 1\ 0\ 1\ 0\ 0] \\
m &= [0\ 1\ 0\ 1] \quad \Rightarrow c = [0\ 1\ 0\ 1]G = [0\ 1\ 0\ 1\ 1\ 0\ 1]
\end{aligned}$$

The expected codeword for $m = [1\ 1\ 0\ 0]$ is $[1\ 1\ 0\ 0\ 0\ 1\ 1]$, but the codeword used in the received message is $[1\ 1\ 0\ 0\ 1\ 1\ 0]$. Thus, $[1\ 1\ 0\ 0\ 1\ 1\ 0]$ violates the constraints of $c = mG$, and is the corrupted block.

Removing corrupted block from received message:

$$\{0100110,\ 1111111,\ 1110100,\ 0101101\}$$

The Hamming distance $d(x, y)$ counts the number of coordinates where x and y differ. Computing all pairs:

$$\begin{aligned}
&d(0100110,\ 1111111) = 4 \\
&d(0100110,\ 1110100) = 3 \\
&d(0100110,\ 0101101) = 3 \\
&d(1111111,\ 1110100) = 3 \\
&d(1111111,\ 0101101) = 3 \\
&d(1110100,\ 0101101) = 4
\end{aligned}$$

Hence, the minimum distance is $d_{\min} = 3$. The $(7, 4)$ code can detect up to $d_{\min} - 1 = 2$ errors.
A code can correct up to t errors if $d_{\min} > 2t$. Equivalently, the error–correcting capability of a block code is

$$t = \left\lfloor \frac{d_{\min} - 1}{2} \right\rfloor$$

For the $(7, 4)$ code with $d_{\min} = 3 > 2 \cdot 1$, $t = 1$. Thus, the code can correct up to one error and detect up to two errors.

b.

Received message with blocks:

$$\{\, 0100110,\ 1100111,\ 0100110,\ 1111111,\ 0100110,\ 1110100,\ 0100110,\ 0101101 \,\}$$

Removing duplicates from received message:

$$\{0100110,\ 1100111,\ 1111111,\ 1110100,\ 0101101\}$$

Encoding process for the specific blocks used in the received message:

$$\begin{aligned}
m &= [0\ 1\ 0\ 0] \quad \Rightarrow c = [0\ 1\ 0\ 0]G = [0\ 1\ 0\ 0\ 1\ 1\ 0] \\
m &= [1\ 1\ 0\ 0] \quad \Rightarrow c = [1\ 1\ 0\ 0]G = [1\ 1\ 0\ 0\ 0\ 1\ 1] \\
m &= [1\ 1\ 1\ 1] \quad \Rightarrow c = [1\ 1\ 1\ 1]G = [1\ 1\ 1\ 1\ 1\ 1\ 1] \\
m &= [1\ 1\ 1\ 0] \quad \Rightarrow c = [1\ 1\ 1\ 0]G = [1\ 1\ 1\ 0\ 1\ 0\ 0] \\
m &= [0\ 1\ 0\ 1] \quad \Rightarrow c = [0\ 1\ 0\ 1]G = [0\ 1\ 0\ 1\ 1\ 0\ 1]
\end{aligned}$$

The expected codeword for $m = [1\ 1\ 0\ 0]$ is $[1\ 1\ 0\ 0\ 0\ 1\ 1]$, but the received codeword is $[1\ 1\ 0\ 0\ 1\ 1\ 1]$. Thus, $[1\ 1\ 0\ 0\ 1\ 1\ 1]$ violates the constraints of $c = mG$ and is the corrupted block.

Compute Hamming distances $d(x, y)$ from the corrupted block $y = 1100111$ to each valid codeword:

$$\begin{aligned}
&d(1100111,\ 0100110) = 2 \\
&d(1100111,\ 1111111) = 2 \\
&d(1100111,\ 1110100) = 3 \\
&d(1100111,\ 0101101) = 3
\end{aligned}$$

Additionally, the distance to the valid encoding of $[1\ 1\ 0\ 0]$ is:

$$d(1100111,\ 1100011) = 1$$

Since the Hamming $(7, 4)$ code has a minimum distance of $d_{\min} = 3$, it can correct up to $t = 1$ error. The corrupted block 1100111 is only distance 1 from the valid codeword 1100011, which is within the error correction capability. But the corrupted block is also distance 2 from two other valid codewords (0100110 and 1111111). This creates ambiguity in the decoding process: a single-error correction would suggest 1100011 but multiple codewords at similar distances means the decoder cannot unambiguously determine which codeword was originally transmitted. Therefore, unique correction is not possible for this corrupted block.

8.

a.

Given the generator matrix

$$G = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 \end{bmatrix}, \qquad (n, k) = (6, 3)$$

A k-bit message $m \in \mathbb{F}_2^{1\times 3}$ is encoded as $c = mG \in \mathbb{F}_2^{1\times 6}$ (all arithmetic mod over $\mathbb{F}_2$).

Encoding:

$$
\begin{aligned}
m = [0\ 0\ 0]: &\quad c = [0\ 0\ 0] \cdot G = [0\ 0\ 0\ 0\ 0\ 0] \\
m = [1\ 0\ 0]: &\quad c = [1\ 0\ 0] \cdot G = [1\ 0\ 0\ 0\ 1\ 1] \\
m = [0\ 1\ 0]: &\quad c = [0\ 1\ 0] \cdot G = [0\ 1\ 0\ 1\ 1\ 0] \\
m = [0\ 0\ 1]: &\quad c = [0\ 0\ 1] \cdot G = [0\ 0\ 1\ 1\ 0\ 1] \\
m = [1\ 1\ 0]: &\quad c = [1\ 1\ 0] \cdot G = [1\ 1\ 0\ 1\ 0\ 1] \\
m = [1\ 0\ 1]: &\quad c = [1\ 0\ 1] \cdot G = [1\ 0\ 1\ 1\ 1\ 0] \\
m = [0\ 1\ 1]: &\quad c = [0\ 1\ 1] \cdot G = [0\ 1\ 1\ 0\ 1\ 1] \\
m = [1\ 1\ 1]: &\quad c = [1\ 1\ 1] \cdot G = [1\ 1\ 1\ 0\ 0\ 0]
\end{aligned}
$$

For the $(6, 3)$ code, the set of codewords is

$$C = \{000000,\ 100011,\ 010110,\ 001101,\ 110101,\ 101110,\ 011011,\ 111000\}\,.$$

b.

The Hamming distance $d(x, y)$ counts the number of coordinates where x and y differ. Computing all pairs:

$$
\begin{aligned}
&d(000000, 100011) = 3 \\
&d(000000, 010110) = 3 \\
&d(000000, 001101) = 3 \\
&d(000000, 110101) = 4 \\
&d(000000, 101110) = 4 \\
&d(000000, 011011) = 4 \\
&d(000000, 111000) = 3
\end{aligned}
$$

Hence, the minimum distance is $d_{\min} = 3$. The $(6, 3)$ code can detect up to $d_{\min} - 1 = 2$ errors.
A code can correct up to t errors if $d_{\min} > 2t$. Equivalently, the error–correcting capability of a block code is

$$t = \left\lfloor \frac{d_{\min} - 1}{2} \right\rfloor$$

For the $(6, 3)$ code with $d_{\min} = 3 > 2 \cdot 1$, $t = 1$. Thus, the code can correct up to one error and detect up to two errors.