Q 1.

(a) To prove that if $1 < m \leq n$, then there cannot exist a set of n mutually orthogonal $m \times n$ Latin rectangles, $(MOLR)_{m\times n}$, consider the following (which follows the argument given in **H Theorem 10.18**, p122.).

Assume we have a set of $(MOLR)_{m\times n}$ where $2 \leq m \leq n$ and with symbols from the alphabet $F_n = \{\lambda_1, \lambda_2, \ldots, \lambda_n\}$. Each of the Latin rectangles in the set of $(MOLR)_{m\times n}$ may have their symbols renamed while still maintaining the orthogonality of the set. For example, assume that we have a set of two $(MOLR)_{2\times 3}$, namely:

Glossary: Alphabet p5.

$$\left\{ \begin{array}{cccccc} \lambda_1 & \lambda_2 & \lambda_3 & \lambda_3 & \lambda_2 & \lambda_1 \\ \lambda_2 & \lambda_3 & \lambda_1 & \lambda_1 & \lambda_3 & \lambda_2 \end{array} \right\} \text{ then } \begin{array}{ccc} (\lambda_1,\lambda_3) & (\lambda_2,\lambda_2) & (\lambda_3,\lambda_1) \\ (\lambda_2,\lambda_1) & (\lambda_3,\lambda_3) & (\lambda_1,\lambda_2) \end{array}.$$

Now, by renaming the symbols of each of the Latin rectangles in the set so that the symbols in the first row of each rectangle are $0, 1, 2$ in natural order, we obtain

$$\left\{ \begin{array}{cccccc} 0 & 1 & 2 & 0 & 1 & 2 \\ 1 & 2 & 0 & 2 & 0 & 1 \end{array} \right\} \text{ then } \begin{array}{ccc} (0,0) & (1,1) & (2,2) \\ (1,2) & (2,0) & (0,1) \end{array},$$

showing that the two mutually orthogonal Latin rectangles in the set still remain mutually orthogonal after renaming the symbols in the way shown. So, in general, a set of $(MOLR)_{m\times n}$ can have the symbols in their first rows renamed so that each of the first rows of each rectangles are $0, 1, \ldots, n-1$ in natural order and still maintain the orthogonality between pairs of rectangles in the set. Having established this, now consider the symbol in the first index position of the second row (i.e. row 1, column 0 using the indexing scheme given in the question) of each of the Latin rectangles. From the definition of a Latin rectangle given in the question this symbol cannot be 0 as the first row of each rectangle is in the natural order of $0, 1, \ldots, n-1$ so it must be one of the symbols in the set $\{1, 2, \ldots, n-1\}$. Also, none of the symbols in the first column of the second row of each rectangle in the set can be the same, for if they were, then when superimposing one rectangle upon another a duplicate of $(0,0), (1,1), \ldots (n-1, n-1)$ would appear in the first row of the superimposed rectangles. This constrains the cardinality of the set of $(MOLR)_{m\times n}$ to a maximum of $n-1$. Consequently, it has been proved that if $1 < m \leq n$, then there cannot exist a set of n mutually orthogonal $m \times n$ Latin rectangles as the maximum value of the cardinality of such a set is $n-1$.

To determine the maximum possible number of mutually orthogonal $1 \times n$ Latin rectangles consider the following cases for $n = 1$, $n = 2$ and $n = 3$.

For the case where $n = 1$ then we have $1! = 1$ Latin rectangle so there is no other rectangle for it to be mutually orthogonal to apart from itself.

For the case where $n = 2$ then we have $2! = 2$ mutually orthogonal Latin rectangles: $[a\ b]$ and $[b\ a]$.

For the case where $n = 3$ then we have $3! = 6$ mutually orthogonal Latin rectangles: $[a\ b\ c]$, $[a\ c\ b]$, $[b\ a\ c]$, $[b\ c\ a]$, $[c\ a\ b]$ and $[c\ b\ c]$.

For the general case where we have a $1 \times n$ Latin rectangles then we have the permutation of n objects taken n at a time:

$$^nP_n = \frac{n!}{(n-n)!} = n!$$

In view of this the maximum possible number of mutually orthogonal $1 \times n$ Latin rectangles is $n!$.

(b)

(i) From the question preamble $p > 1$ is prime and $q > 1$ is an integer such that each of the prime factors of q are greater or equal to p. Let $n = pq$. Then for $\lambda = 1, 2, \ldots, n-1$, define A_λ to be the $p \times n$ array with entries $a_{i,j}^{(\lambda)} \equiv \lambda i + j \pmod{n}$ for $i = 0, 1, \ldots, p-1$, $j = 0, 1, \ldots, n-1$. Now, in order to prove that A_λ is a $p \times n$ Latin rectangle consider the following.

The entries in A_λ, $a_{i,j}^{(\lambda)}$, are those in the $\mathbb{Z}_n = \{0, 1, 2, \ldots, n-1\}$ as $a_{i,j}^{(\lambda)} \equiv \lambda i + j \pmod{n}$, where λ is non-zero. The rows are indexed by $0, 1, \ldots, n-1$ and columns are indexed by $0, 1, \ldots, p-1$ where i refers to the row index and j to that of the column index in the array element $a_{i,j}^{(\lambda)}$. Therefore, the array A_λ is a $p \times n$ array, i.e. a rectangle with the number of columns greater than the number of rows.

Now, assume that two elements in a given row, row i say, are the same at index locations j and j'. In this case we then have $a_{i,j}^{(\lambda)} = a_{i,j'}^{(\lambda)}$ in which case $\lambda i + j = \lambda i + j'$ with arithmetic in $\mathbb{Z}_n$. As $\lambda \neq 0$ this implies that $j = j'$ and therefore each element of the i-th row appears exactly once.

Similarly assume two elements are the same in two columns, column j say. In this case we then have $a_{i,j}^{(\lambda)} = a_{i',j}^{(\lambda)}$ in which case $\lambda i + j = \lambda i' + j$ with, as before, arithmetic in $\mathbb{Z}_n$. As $\lambda \neq 0$ this implies $i = i'$ and therefore each element of the j-th column are distinct.

Thus, the conditions for the array A_λ to be that of a Latin rectangle have been met.

(ii) To prove that A_λ and A_μ are mutually orthogonal given that $\lambda \neq \mu$, thus forming a set $\{A_\lambda : 1 \leq \lambda \leq n-1\}$ of $n-1$ mutually orthogonal Latin rectangles consider the following.

First we prove that the elements of the first row of each rectangle A_λ and A_μ are equal as follows:

Let the elements of the first rectangle be $a_{i,j} = \lambda i + j$ and those of the second rectangle be $b_{i,j} = \mu i + j$. When $i = 0$, that is the first row of each rectangle, then

$$a_{0,j} = 0 \times i + j = j; \quad \text{and} \quad b_{0,j} = 0 \times i + j = j,$$

and thus, elements of the first rows of each of the rectangles are equal corresponding to the column index, j, for $j = 0, 1, \ldots, n-1$.

Now consider a row, i, which is not the first and as such $i \neq 0$ and assume that the element $a_{i,j} = b_{i,j}$ in which case the rectangles A_λ and A_μ are *not* mutually orthogonal as the pair $(a_{i,j}, b_{i,j})$ will have occurred in the first row of the superposition of A_λ on A_μ. Then,

$$a_{i,j} = \lambda \times i + j \quad \text{and} \quad b_{i,j} = \mu \times i + j, \quad i \neq 0, \quad j = 0, 1, \ldots, n-1,$$

which implies that $\lambda = \mu$ which is a contradiction. As such each of the Latin rectangles in the set $\{A_\lambda : 1 \leq \lambda \leq n-1\}$ are mutually orthogonal.

(c)

(i)

0	0	0	0	0	0	0
0	1	1	1	1	1	1
0	2	2	2	2	2	2
0	3	3	3	3	3	3
0	4	4	4	4	4	4
0	5	5	5	5	5	5
1	0	1	2	3	4	5
1	1	2	3	4	5	0
1	2	3	4	5	0	1
1	3	4	5	0	1	2
1	4	5	0	1	2	3
1	5	0	1	2	3	4

Table 1: The codewords of $C_{2,3}$.

(ii) To determine the minimum distance of $C_{p,q}$ in terms of p and q consider the following.

A code $C_{p,q}$ is formed from Latin rectangles A_λ by taking as codewords all vectors of the form:

$$(i, j, a_{i,j}^{(1)}, a_{i,j}^{(2)}, \ldots, a_{i,j}^{(n-1)})$$

for $i = 0, 1, \ldots, p-1$ and $j = 0, 1, \ldots, n-1$ where $n = pq$. Now, $a_{i,j}^{(\lambda)} = \lambda i + j$ for $\lambda = 1, 2, \ldots, n-1$ and the codewords can be generated using the generator matrix:

$$G = \begin{pmatrix} 1 & 0 & 1 & 2 & \cdots & n-1 \\ 0 & 1 & 1 & 1 & \cdots & 1 \end{pmatrix}.$$

Thus, for example, each of the codewords shown in Table 1 can be generated from the generator matrix, G, as follows.

$$\begin{pmatrix} i & j \end{pmatrix} \begin{pmatrix} 1 & 0 & 1 & 2 & 3 & 4 & 5 \\ 0 & 1 & 1 & 1 & 1 & 1 & 1 \end{pmatrix},$$

$$i = 0, 1, \ldots, p-1, \quad j = 0, 1, \ldots, n-1.$$

As an example consider the case for $i = 1$ and $j = 5$; the codeword generated is thus:

$$\begin{pmatrix} 1 & 5 \end{pmatrix} \begin{pmatrix} 1 & 0 & 1 & 2 & 3 & 4 & 5 \\ 0 & 1 & 1 & 1 & 1 & 1 & 1 \end{pmatrix} = \begin{pmatrix} 1 & 5 & 0 & 1 & 2 & 3 & 4 \end{pmatrix}.$$

In order to determine the minimum distance between codewords for the general case consider what happens when i and j are both zero; i is zero but j is non-zero and visa-versa; and when both are non-zero.

$i = 0, 1, \ldots, p-1,$
$j = 0, 1, \ldots, pq-1.$

1. When $i = 0$ and $j = 0$ then the codeword is the zero vector of length $pq + 1$.

2. When $i = 0$ and $j \neq 0$ then the codewords are of the form:

   $$\begin{pmatrix} 0 & j & j & \cdots & j \end{pmatrix}$$

   where the length of the vector is $pq+1$ so that the symbol represented by j appears exactly pq times. That is, the weight of the vector is pq.

3. When $i \neq 0$ and $j = 0$ the the codewords are of the form:

   $$\begin{pmatrix} i & 0 & i & 2i & \cdots & (pq-1)i \end{pmatrix} \pmod{pq},$$

   $\lambda = 1, 2, \ldots, pq-1.$

   where the length of the vector is $pq + 1$ and the symbol zero appears precisely once in the codewords generated in this fashion. This can easily been seen by considering the case where $i = p - 1$ and $\lambda = pq - 1$ (giving the maximum value of λi), then the last symbol of the codeword will be $\lambda i = (pq - 1)(p - 1) < pq$ and therefore, $\lambda i \not\equiv 0 \pmod{pq}$. Thus, the weights of these codewords are also pq.

4. Finally, when $i \neq 0$ and $j \neq 0$ the codewords are of the form

   $$\begin{pmatrix} i & j & i+j & 2i+j & \cdots & (pq-1)i+j \end{pmatrix} \pmod{pq},$$

   where the length of the vector is $pq + 1$ and the symbol zero appears precisely once in the codewords generated in this fashion. This can be seen by considering how and when the symbol zero is generated in the codeword. It is generated precisely when

   $$pq = \lambda i + j, \quad \lambda = 1, 2, \ldots, pq-1,$$

   which can only occur once in a codeword. So, again the weight of the codewords is equal to pq when $i \neq 0$ and $j \neq 0$.

In view of the foregoing and given each codeword is distinct then the minimum distance of $C_{p,q}$ is pq as each codeword differs from another in exactly pq coordinate positions.

Q 2.

(a) Supposing that C is a perfect binary $(n, M, 5)$-code it can be proved that $n^2 + n + 2$ is necessarily of the form 2^m, where m is an integer, and that if C is non-trivial (i.e. if $n > 5$), then $m > 5$ in the following way.

From **Theorem 2.16** a q-ary $(n, M, 2t + 1)$-code satisfies **H** p20.

$$M\left(\binom{n}{0} + \binom{n}{1}(q-1) + \ldots + \binom{n}{t}(q-1)^t\right) \leq q^n. \tag{2.1}$$

Now as the code under consideration is *perfect* the code achieves the sphere-packing bound such that equality occurs in 2.1. For the case of the given binary code $q = 2$ and $2t + 1 = 5$ so that $t = 2$. Thus, (2.1) becomes **H** p97.

$$\begin{aligned} 2^n =& M\left(1 + n + \frac{n(n-1)}{2}\right), \\ =& M\left(1 + n + \frac{n(n-1)}{2}\right), \\ =& M\left(1 + n + \frac{n^2}{2} - \frac{n}{2}\right), \\ =& M\left(1 + \frac{n}{2} + \frac{n^2}{2}\right). \end{aligned}$$

Then, multiplying both sides of the last expression by 2 gives

$$\begin{aligned} 2^{n+1} &= 2M\left(1 + \frac{n}{2} + \frac{n^2}{2}\right), \\ &= M\left(n^2 + n + 2\right). \end{aligned} \tag{2.2}$$

Now,

$\log_2(2^{n+1}) = n + 1 = \log_2(M) + \log_2(n^2 + n + 2),\ n > 5$ if C is nontrival.

Hence, both M and $n^2 + n + 2$ are positive integer powers of 2 and as such $n^2 + n + 2$ is necessarily of the form 2^m where m is a positive integer.

Let $n = 5$ then $2^m = 5^2 + 5 + 2 = 32 = 2^5$ and so when $n > 5$ then $m > 5$ also.

(b) The Lloyd polynomial $L_2(x)$ will be shown to be

$$2L_2(x) = 4x^2 - 4(n+1)x + (n^2 + n + 2)$$

as follows.

From **Theorem 9.6** if there exists a perfect $(n, M, 2t+1)$-code over $GF(q)$ then the Lloyd polynomial is given by **H** p103.

$$L_t(x) = \sum_{j=0}^{t} (-1)^j (q-1)^{t-j} \binom{x-1}{j} \binom{n-x}{t-j}$$

and has t distinct roots in the interval $1 \leq x \leq n$.

In our case, $q = 2$ and $t = 2$ so we have

$$L_2(x) = \sum_{j=0}^{2} (-1)^j \binom{x-1}{j} \binom{n-x}{2-j},$$

which becomes

$$L_2(x) = \binom{x-1}{0}\binom{n-x}{2} - \binom{x-1}{1}\binom{n-x}{1} + \binom{x-1}{2}\binom{n-x}{0}.$$

Expanding the binomial coefficients we obtain

$$L_2(x) = \frac{(n-x)(n-x-1)}{2} - (x-1)(n-x) + \frac{(x-1)(x-2)}{2}.$$

Then,

$$\begin{aligned} 2L_2(x) &= (n-x)(n-x-1) - 2(x-1)(n-x) + (x-1)(x-2), \\ &= n^2 - 2nx - n + x^2 + x - 2\left(nx - x^2 - n + x\right) + x^2 - 3x + 2, \\ &= n^2 - 4nx + n + 4x^2 - 2x + 2, \\ &= 4x^2 - 4x(n+1) + (n^2 + n + 2) \text{ as required.} \end{aligned}$$

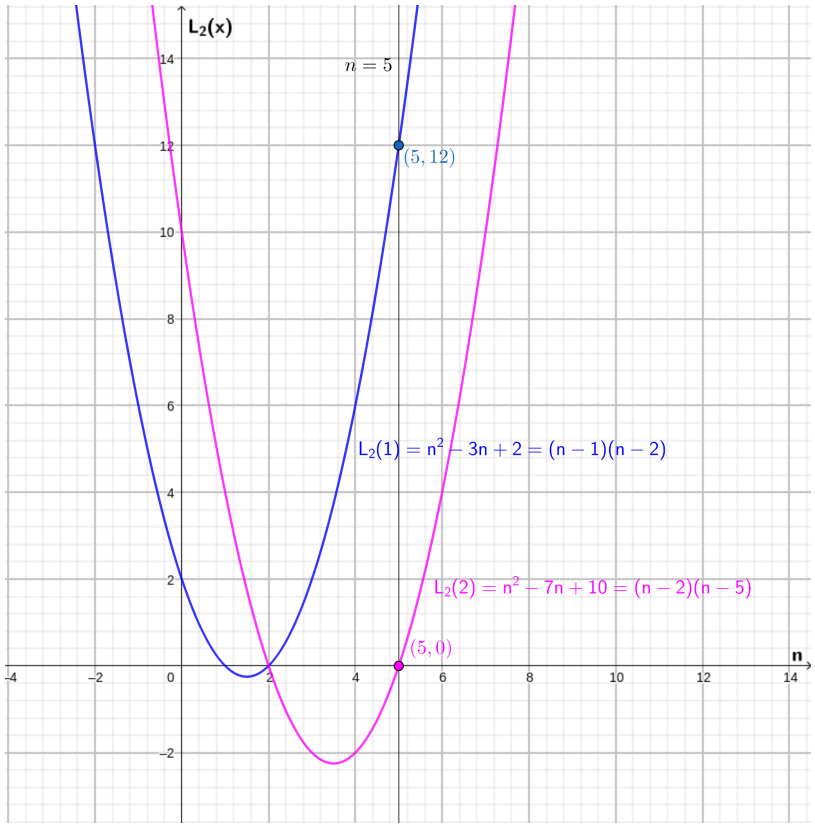

Figure 1: Plots of $L_2(1)$ and $L_2(2)$ against n showing that when $n > 5$ neither $L_2(1)$ nor $L_2(2)$ is zero.

Shown in Figure 1 are plots of $L_2(1)$ and $L_2(2)$ against n which shows for $n = 5$ that $L_2(1) = 12$ and $L_2(2) = 0$. As both plots are quadratic with roots shown, neither plot crosses the x-axis again so as n increases beyond 5 both $L_2(1)$ and $L_2(2)$ also increase. So neither $L_2(1)$ nor $L_2(2)$ is zero for $n > 5$.

(c) Recall that the Lloyd polynomial $L_2(x)$ is given by

$$2L_2(x) = 4x^2 - 4(n+1)x + (n^2 + n + 2),$$

or

$$L_2(x) = 2x^2 - 2(n+1)x + \frac{n^2 + n + 2}{2}.$$

Supposing that $L_2(x)$ has two integer roots satisfying $3 \leq x_1 < x_2 \leq n$; the roots of $L_2(x)$ are given by

$$\begin{aligned} x_{1,2} &= \frac{2(n+1) \pm \sqrt{2^2(n+1)^2 - 4(2)(n^2+n+2)/2}}{4}, \\ &= \frac{2(n+1) \pm 2\sqrt{n^2 + 2n + 1 - n^2 - n - 2}}{4}, \\ &= \frac{(n+1) \pm \sqrt{n-1)}}{2}. \end{aligned}$$

So,

$$x_1 = \frac{(n+1) - \sqrt{n-1)}}{2} \text{ and } x_2 = \frac{(n+1) + \sqrt{n-1)}}{2}.$$

Now, consider the product of the two roots

$$\begin{aligned} x_1 x_2 &= \frac{(n+1) - \sqrt{n-1)}}{2} \times \frac{(n+1) + \sqrt{n-1)}}{2}, \\ &= \frac{(n+1)^2 - (n-1)}{4}, \\ &= \frac{n^2 + 2n + 1 - n + 1}{4}, \\ &= \frac{n^2 + n + 2}{4} = \frac{n^2 + n + 2}{2^2}. \end{aligned}$$

From part (a) it was shown that $2^m = n^2 + n + 2$ so we have,

$$x_1 x_2 = \frac{2^m}{2^2} = 2^{m-2}, \quad m > 5.$$

C is a nontrivial code so $n > 5$ and $m > 5$.

Thus, both x_1 and x_2 are integer powers of 2.

If $x_1 = 2^a$ and $x_2 = 2^b$ it can be shown that $(2^{a+1} + 2^{b+1} - 1)^2 = 2^{m+2} - 7$ as follows.

$$\begin{aligned} x_1 + x_2 &= \frac{(n+1) - \sqrt{n-1)}}{2} + \frac{(n+1) + \sqrt{n-1)}}{2} = n + 1, \\ 2^a + 2^b &= n + 1, \\ n &= 2^a + 2^b - 1. \end{aligned}$$

So,

$$\begin{aligned} 2n &= 2^{a+1} + 2^{b+1} - 2 = (2^{a+1} + 2^{b+1} - 1) - 1, \\ 2n + 1 &= 2^{a+1} + 2^{b+1} - 1, \\ (2n+1)^2 &= (2^{a+1} + 2^{b+1} - 1)^2, \\ (2^{a+1} + 2^{b+1} - 1)^2 &= 4n^2 + 4n + 1. \end{aligned}$$

Now, from above $2^m = n^2 + n + 2$, thus $2^{m+2} = 4n^2 + 4n + 8$ and so $2^{m+2} - 7 = 4n^2 + 4n + 1$. Consequently, we have

$$(2^{a+1} + 2^{b+1} - 1)^2 = 2^{m+2} - 7, \text{ as required.}$$

Considering this last equation modulo 16 a contradiction can be obtained thus proving that $L_2(x)$ cannot have two distinct integer roots for $1 \leq x \leq n$ in the following way.

First expand the left-hand side of the last equation; let $\alpha = 2^{a+1} + 2^{b+1}$ to obtain

$$(\alpha - 1)^2 = \alpha^2 - 2\alpha + 1.$$

Next substitute back for α and expand α^2 to give

$$\begin{aligned} (2^{a+1} + 2^{b+1})(2^{a+1} + 2^{b+1}) &= 2^{2a+2} + 2 \cdot 2^{a+b+2} + 2^{2b+2}, \\ &= 2^{2a+2} + 2^{a+b+3} + 2^{2b+2}. \end{aligned}$$

So, substituting for the relevant parts in $\alpha^2 - 2\alpha + 1$ we get

$$\begin{aligned} \alpha^2 - 2\alpha + 1 &= 2^{2a+2} + 2^{a+b+3} + 2^{2b+2} - 2(2^{a+1} + 2^{b+1}) + 1, \\ &= 2^{2a+2} + 2^{a+b+3} + 2^{2b+2} - (2^{a+2} + 2^{b+2}) + 1. \end{aligned}$$

Given that $3 \leq 2^a < 2^b \leq n$ the minimum values on a and b are 2 and 3 respectively. Therefore, the last expression is divisible by 2^4 with remainder 1, thus

$$\begin{aligned} (2^{a+1} + 2^{b+1} - 1)^2 &= 2^{2a+2} + 2^{a+b+3} + 2^{2b+2} - (2^{a+2} + 2^{b+2}) + 1, \\ &\equiv 1 \pmod{16}. \end{aligned}$$

Now considering the right-hand side of $(2^{a+1} + 2^{b+1} - 1)^2 = 2^{m+2} - 7$. It was established earlier that if C is nontrivial then $m > 5$. As such $2^{m+2} - 7$ is divisible by 2^4 with remainder -7. That is

$$2^{m+2} - 7 \equiv -7 \equiv 9 \pmod{16}.$$

We have now established a contradiction; $1 \not\equiv 9 \pmod{16}$ therefore proving that $L_2(x)$ cannot have two distinct integer roots for $1 \leq x \leq n$.

(d) To prove that there is a binary linear $(90, 2^{73}, 5)$-code, that is a $[90, 73, 5]$-code, use will be made of **Theorem 8.10** as follows. **H** p91.

From this theorem which assumes that q is a prime power, then there exists a q-ary $[n, k]$-code with minimum distance at least d provided the following inequality holds:

$$\sum_{i=0}^{d-2}(q-1)^i\binom{n-1}{i} < q^{n-k}.$$

For the code under consideration $q = 2$ (a prime power), $n = 90$, $k = 73$ and $d = 5$, therefore

$$\sum_{i=0}^{5-2}(2-1)^i\binom{90-1}{i} < 2^{90-73},$$

$$\sum_{i=0}^{3}(1)^i\binom{89}{i} < 2^{17},$$

$$1 + 89 + 3916 + 113564 = 117570 < 131072 = 2^{17},$$

showing that there is a binary linear $(90, 2^{73}, 5)$-code.

Q 3.

(a) Using **Definition 7.2** the codewords of $RM(1,1)$, $RM(1,2)$ and $RM(1,3)$ are: Block **2** Course Notes p23.

$$RM(1,1) = \begin{matrix} 00 \\ 01 \\ 10 \\ 11 \end{matrix}$$

$$RM(1,2) = \begin{matrix} 0000 \\ 0101 \\ 1010 \\ 1111 \\ 0011 \\ 0110 \\ 1001 \\ 1100 \end{matrix}$$

$$RM(1,3) = \begin{matrix} 00000000 \\ 00001111 \\ 00110011 \\ 00111100 \\ 01010101 \\ 01011010 \\ 01100110 \\ 01101001 \\ 11111111 \\ 11110000 \\ 11001100 \\ 11000011 \\ 10101010 \\ 10100101 \\ 10011001 \\ 10010110 \end{matrix}$$

The weight figures for these codes are summarised below.

$w(RM(1,m))$	$RM(1,1)$	$RM(1,2)$	$RM(1,3)$
0	1	1	1
1	2	0	0
2	1	6	0
4	0	1	14
8	0	0	1

(b) From **Theorem 7.2** if $r > 0$ then $RM(r-1,m)$ is contained in $RM(r,m)$. Block **2** Course Notes p27.

We are considering the cases for $r = 1$ and for $m \geq 1$.

Now from **Definition 7.3** the generator matrix $G(0, m) = [11 \cdots 1]$ is of length 2^m bits for $m > 0$. Block **2** Course Notes p24.

$RM(0, m)$ is a binary repetition code and therefore has precisely two codeword which are of length 2^m; namely the all zero codeword and the all ones codeword as $G(0, m)$ is multiplied on the left by [0] or [1] to generate the codewords.

This establishes that each of the vectors in $\{\mathbf{0}, \mathbf{1}\}$ of lengths 2^m is contained in $RM(1, m)$. These respectively have weights of 0 and 2^m and as such $A_0 = 1$ and $A_{2m} = 1$. These are the respective coefficients for z^0 and z^{2m} in the polynomial $1 + (2^{m+1} - 2)z^{2^{m-1}} + z^{2m}$.

Now, looking at the codewords of $RM(1, 1)$, $RM(1, 2)$ and $RM(1, 3)$ and considering the weights of the codewords other than the all zero codeword and the all ones codeword of each of these codes.

From part (a) these are 2, 6 and 14 for m equal to $1, 2, 3$ respectively and are the values associated with the middle coefficient of the weight enumerator, namely $2^{m+1} - 2$. Thus,

$$\sum_{i=1}^{m} 2^i = 2^{m+1} - 2.$$

Note firstly that when $m = 1$ we have $2 = 2^{1+1} - 2$. Assume inductively, that when $m = k$ the result is true i.e. assume that

$$\sum_{i=1}^{k} 2^i = 2^{k+1} - 2 \text{ is true.}$$

Then the sum for $m = k + 1$ may be written as

$$\sum_{i=1}^{k+1} 2^i = \sum_{i=1}^{k} 2^i + 2^{k+1}.$$

Using the above assumption this becomes

$$\sum_{i=1}^{k+1} 2^i = 2^{k+1} - 2 + 2^{k+1}.$$

$$= 2^{k+2} - 2 \text{ which is the desired result for } m = k + 1.$$

It therefore follows that by induction the middle coefficient of weight enumerator is $2^{m+1} - 2$.

Next we need to establish that the exponent of the indeterminate variable, z, is 2^{m-1}.

Now 2^{m-1} is just d, the minimum distance of the codewords of the first order Reed-Muller codes. From **Theorem 7.1**, $RM(r,m)$ is a binary linear $[2^m, \sum_{i=0}^{r}\binom{m}{i}, 2^{m-r}]$-code. Thus, for first order Reed-Muller codes $RM(1,m)$ the minimum distance is 2^{m-1} which is the exponent of the middle indeterminate variable, z, of the weight enumerator $1+(2^{m+1}-2)z^{2^{m-1}}+z^{2^m}$.

Block **2** Course Notes p25.

Thus, we have established that for $m \geq 1$, $RM(1,m)$ has weight enumerator

$$W_C(z) = 1 + (2^{m+1} - 2)z^{2^{m-1}} + z^{2^m}$$

.

(c) From **Theorem 7.3** if $r < m$ then $RM(r,m)^{\perp} = RM(m-r-1,m)$.

Block **2** Course Notes p27.

r	m	$m-2$	$RM(r,m)^{\perp}$	$RM(m-2,m)^{\perp}$	$RM(m-r-1,m)$
1	3	1	$RM(1,3)$	$RM(1,3)$	$RM(1,3)$
2	4	2	$RM(2,4)$	$RM(2,4)$	$RM(1,4)$
3	5	3	$RM(3,5)$	$RM(3,5)$	$RM(1,5)$
4	6	4	$RM(4,6)$	$RM(4,6)$	$RM(1,6)$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$

Table 2: The dual code of $RM(m-2,m)$ is $RM(1,m)$ for $m > 2$.

From Table 2 we can see that the dual of the $RM(1,m)$ code is the $RM(m-2,m)$ code for $m > 2$. From **Theorem 13.6** we have the MacWilliams identity we obtain

H p168.

$$W_{C^{\perp}}(z) = \frac{1}{q^k}\left[1 + (q-1)z\right]^n W_C\left(\frac{1-z}{1+(q-1)z}\right) \tag{3.1}$$

Now, from **Theorem 7.1**, $RM(r,m)$ is a binary linear $[n,k]$-code where $n = 2^m$ and $k = \sum_{i=0}^{r}\binom{m}{i}$. From Table 2 it is seen that in the right most column that $r = 1$ in all the rows. Thus,

Block **2** Course Notes p25.

$$k = \sum_{i=0}^{1}\binom{m}{i} = \binom{m}{0} + \binom{m}{1} = 1 + m.$$

In 3.1 $q = 2$ and $k = m + 1$ so 3.1 becomes

$$W_{C^{\perp}}(z) = \frac{1}{2^k}\left[1 + z\right]^n W_C\left(\frac{1-z}{1+z}\right). \tag{3.2}$$

From part (b) we have the weight enumerator for $RM(1,m)$, $m \geq 1$

$$W_C(z) = 1 + (2^{m+1} - 2)z^{2^{m-1}} + z^{2m}.$$

Thus,

$$W_C\left(\frac{1-z}{1+z}\right) = 1 + (2^{m+1} - 2)\left(\frac{1-z}{1+z}\right)^{2^{m-1}} + \left(\frac{1-z}{1+z}\right)^{2m}.$$

Substituting this last expression into the right-hand side of (3.2) we obtain

$$W_{C^\perp}(z) = \frac{1}{2^k}\,[1+z]^n \left(1 + (2^{m+1}-2)\left(\frac{1-z}{1+z}\right)^{2^{m-1}} + \left(\frac{1-z}{1+z}\right)^{2^m}\right),$$
$$= \frac{1}{2^k}\left((1+z)^n + (1+z)^n\,(2^{m+1}-2)\left(\frac{1-z}{1+z}\right)^{2^{m-1}} + (1+z)^n\left(\frac{1-z}{1+z}\right)^{2^m}\right),$$

It was established above that $n = 2^m$ and $k = m+1$, thus we obtain

$$W_{C^\perp}(z) = \frac{1}{2^{m+1}}\left((1+z)^{2^m} + (1+z)^{2^m}\,(2^{m+1}-2)\left(\frac{1-z}{1+z}\right)^{2^{m-1}} + (1+z)^n\left(\frac{1-z}{1+z}\right)^{2^m}\right),$$
$$= \frac{1}{2^{m+1}}\left((1+z)^{2^m} + (2^{m+1}-2)(1-z^2)^{2^{m-1}} + (1-z)^{2^m}\right), \text{ as required.}$$

(d) Given $C = RM(2,4)$, which we note is of the form $RM(m-2,m)$, we can use the result of part (c) above in determining an expression for $P_{undetec}(C)$.

Now, for $C = RM(2,4)$ and using the result of part (c) we obtain the weight enumerator:

$$W_{C^\perp}(z) = z^{16} + 140\,z^{12} + 448\,z^{10} + 870\,z^8 + 448\,z^6 + 140\,z^4 + 1,$$

from which we obtain an expression (**Theorem 6.14**) for the probability of an incorrect message being received undetected, $P_{undetec}(C)$: **H** p64.

$$140\,(1-p)^{12}p^4 + 448\,(1-p)^{10}p^6 + 870\,(1-p)^8p^8 + 448\,(1-p)^6p^{10} + 140\,(1-p)^4p^{12} + p^{16}.$$

We see from this last expression that when p is small then $P_{undetec}(C) \approx 140p^4$. For the case when $p = 0.01$ we obtain the rough estimate for the average number of vectors per undetected error is 0.000 001 4. So only fourteen words in about 10 000 000 will be accepted with errors undetected.

Q 4.

(a) Given that C is the cyclic code $R_{15} = F_2[x]/(x^{15}-1)$ with generator polynomial $g(x) = 1 + x + x^2 + x^3 + x^6$ it can be shown that the check polynomial $h(x) = 1 + x + x^4 + x^5 + x^6 + x^9$ *does* corresponding to $g(x)$ by considering the following.

By **Theorem 12.9 (iii)**, $g(x)$ is a factor of $x^{15} - 1$ so that **H** p147.

$$x^{15} - 1 = g(x)h(x).$$

Thus, multiplying $g(x)$ by $h(x) = 1 + x + x^4 + x^5 + x^6 + x^9$ gives

$$1+2x+2x^2+2x^3+2x^4+2x^5+4x^6+4x^7+2x^8+2x^9+2x^{10}+2x^{11}+2x^{12}+x^{15},$$

which is just equal to $1 + x^{15}$ and which in turn is equal to $x^{15} - 1$ over $GF(2)$.

To show that the 56 polynomials in R_{15}: $\mathbf{0}$, $\boldsymbol{a_k}h(x)$, $\boldsymbol{b_k}h(x)$, $\boldsymbol{c_k}h(x)$ and $\boldsymbol{d_k}h(x)$ where

- $\mathbf{0}$, the all zero vector;
- $\boldsymbol{a_k} = x^k$, $0 \leq k \leq 14$;
- $\boldsymbol{b_k} = x^k(1 + x) = x^k + x^{k+1}$, $0 \leq k \leq 13$;
- $\boldsymbol{c_k} = x^k(1 + x^2) = x^k + x^{k+2}$, $0 \leq k \leq 13$; and
- $\boldsymbol{d_k} = x^k(1 + x + x^2) = x^k + x^{k+1} + x^{k+2}$, $0 \leq k \leq 13$,

are all distinct consider the following.

The total number of such vectors is $1 + 15 + 14 + 13 + 13 = 56$.

Take as an example the vector $x^k(1 + x)h(x)$ when k takes a specific value, $k = 13$ say. Then, for this example, first write down the polynomial $h(x) = 1 + x + x^4 + x^5 + x^6 + x^9$ in vector form which is a vector of length 15:

$$h(x) = [110011100100000].$$

Multiplying $h(x)$ by x^{13} rotates $h(x)$ by 13 places, thus **H** p.146

$$x^{13}h(x) = [001110010000011].$$

Similarly, multiplying $h(x)$ by x^{14} rotates $h(x)$ by 14 places, thus

$$x^{14}h(x) = [100111001000001].$$

Adding these two vectors element by element and reducing modulo 2 gives the vector:

$$x^{13}(1 + x)h(x) = x^{13}h(x) + x^{14}h(x) \pmod 2 = [101001011000010].$$

To convert this last vector into a base 10 integer we perform the element by element multiplication of it by $[2^{14}\, 2^{13}, \cdots, 2^0]$. Summing the elements the vector formed in this way gives $16384+8192+1024+512+256+32 = 26400$. This is the value shown in the seventh row, sixth column of Table 3.

All other values in the table can be obtained in a similar fashion. Inspection of Table 3 shows all 56 entries to be distinct.

0	4199	9207	14598	18483	22612	27588
825	4958	9605	14911	19210	23839	28303
1355	5420	9916	15437	19832	24102	28981
1650	5653	10840	16244	20033	24914	29196
2479	6600	11306	16796	20987	25195	29822
2710	7299	12051	17061	21186	25625	30535
3300	8122	12457	17623	21680	26400	30874
4061	8398	13200	18414	22409	26877	32488

Table 3: Distinct representations in base 10 of the 56 vectors generated from polynomials in R_{15} as described in the text.

The vectors of burst length at most 3 are those formed from the polynomials that premultiply $h(x)$ as given given above. Namely, x^k is a vector with a 1 in the k^{th} position and other entries 0; $x^k(1+x)$ is the vector with a 1 in the k^{th} and $(k+1)^{th}$ positions and other entries 0; $x^k(1+x^2)$ is the vector with a 1 in the k^{th} and $(k+2)^{th}$ positions and 0 in other positions; and $x^k(1+x+x^2)$ is the vector with a 1 in the k^{th}, $(k+1)^{th}$ and $(k+2)^{th}$ positions and 0 in other positions.

Now, recall from TMA02 Q4(c)(iii) that:

> Given a polynomial $p(x)$ of degree at most 8, the syndrome of p(x) is defined to be $p(x)h(x)$ (reduced modulo x^9-1), where $h(x)$ is the check polynomial corresponding to $g(x)$.

Consequently, the the 56 polynomials: $\mathbf{0}$, $\boldsymbol{a_k}h(x)$, $\boldsymbol{b_k}h(x)$, $\boldsymbol{c_k}h(x)$ and $\boldsymbol{d_k}h(x)$ represent syndromes which because of the way they were constructed have been reduced modulo $x^{15}-1$. As these 56 syndromes are all distinct then each one is in a distinct coset of C.

To prove that C is *not* 3 error-correcting consider the following which follows the argument given in the solution of exercise 12.1:

Block **3** Solutions to Unit 12 p85.

C is a cyclic code in R_{15} so in this case $n = 15$. The degree of $g(x)$ is $r = 6$ and therefore the dimension of the code is $k = n - r = 9$. As such, C is a $[15, 9]$-code (see **Theorem 12.12**) and the number of cosets of C is $2^{n-k} = 2^6 = 64$. Compare this value with the number of vectors in $V(15, 2)$

H p149.

having weight no more than 3, i.e.

$$\binom{15}{0} + \binom{15}{1} + \binom{15}{2} + \binom{15}{3} = 1 + 15 + 105 + 455 = 576.$$

So, C cannot be a 3 error-correcting code because if it were then each of these 576 vectors would have to lie in a distinct coset (See solution to 12.1). This is impossible as the total number of cosets is 64 which is considerably less than 576 required for C to be a 3 error-correcting code.

Block **3** Solutions to Unit 12 p85.

(b)

(i) The following codewords are to be transmitted using interleaving to depth 3.

$$\begin{aligned} \boldsymbol{c_1} &= 1100110 \quad \boldsymbol{c_4} = 0111100 \\ \boldsymbol{c_2} &= 1001100 \quad \boldsymbol{c_5} = 1111111 \\ \boldsymbol{c_3} &= 0010110 \quad \boldsymbol{c_6} = 0001111 \end{aligned}$$

We take as the first set of codewords: $\{\boldsymbol{c_1}, \boldsymbol{c_2}, \boldsymbol{c_3}\}$. The sequence of bits to be transmitted is formed from the columns of the listing of $\boldsymbol{c_1}$, $\boldsymbol{c_2}$ and $\boldsymbol{c_3}$ given above. Thus the transmission is 110 100 001 010 111 101 000 where spaces have been placed between each set of three bits for clarity. In a similar way, $\boldsymbol{c_4}$, $\boldsymbol{c_5}$ and $\boldsymbol{c_6}$ are transmitted as 010 110 110 111 111 011 011.

From **Theorem 12.3** if D is a burst l error-correcting binary linear code interleaved to depth s, then all bursts of length at most sl will be corrected, provided that each codeword is affected by at most one burst error and by no other errors. Now, D is a Ham$(3, 2)$-code and is equivalent to the perfect $[7, 4, 3]$-code. Using **Definition 12.2** we determine that code D is a 1 burst error-correcting code (i.e. $l = 1$). So interleaving a Ham$(3, 2)$-code to depth 3 (i.e. $s = 3$) will ensure that all bursts of length no greater than 3 will be corrected subject to the proviso outlined above.

Block **3** Course Notes p42.

Block **3** Course Notes p40.

(ii) Using 2-frame interleaving we generate the following array from the six codewords $\boldsymbol{c_1}, \boldsymbol{c_2}, \ldots, \boldsymbol{c_6}$.

1	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	1	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	1	0	1	1	1	0	0	0	0	0	0
0	0	0	0	0	0	0	0	1	1	1	1	1	1	0	0	0	0
0	0	0	0	0	0	0	0	0	0	1	0	1	0	1	1	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1

The first row of the array contains all the first coordinate position bits; the second row contains all the second coordinate bits, and so on. The number of rows is the length of a codeword (i.e. $n = 7$). The offset of each row

from the previous row is 2 (i.e. $f = 2$). The bits are transmitted column by column, so that the transmitted vector in this case will be:

1000000 1000000 0100000 0000000 1000000 0100000 0110000 0011000 0010100
0001100 0001110 0001100 0000110 0000100 0000010 0000010 0000001 0000001

where spaces have again been added for clarity.

D is a 1 burst error-correcting code. When D is 2-frame interleaved bursts of length at most $l(fn + 1) = 1 \times (2 \times 7 + 1) = 15$ will be corrected (see **Theorem 12.4**). This is provided that each codeword is affected by at most one burst error and by no other errors.

Block **3** Course Notes p43.

(c) Given that the received vector is

111 111 000 000 000 000 111 111
111 111 000 000 111 111 000 111
100 111 011 000 000 011 111 100
000 111 111 000 000 111 111 000
000 110 110 000 000 110 110 000 ...

unscrambling the interleaving, gives the received vectors

$$\boldsymbol{d_1} = 11000011$$
$$\boldsymbol{d_2} = 11000011$$
$$\boldsymbol{d_3} = 11000011$$

$$\boldsymbol{d_4} = 11001101$$
$$\boldsymbol{d_5} = 11001101$$
$$\boldsymbol{d_6} = 11001101$$

$$\boldsymbol{d_7} = 11000011$$
$$\boldsymbol{d_8} = 01100110$$
$$\boldsymbol{d_9} = 01100110$$

$$\boldsymbol{d_{10}} = 01100110$$
$$\boldsymbol{d_{11}} = 01100110$$
$$\boldsymbol{d_{12}} = 01100110$$

$$\boldsymbol{d_{13}} = 01100110$$
$$\boldsymbol{d_{14}} = 01100110$$
$$\boldsymbol{d_{15}} = 00000000$$

The sixteen codewords of C_2 are

$$
\begin{aligned}
[0\ 0\ 0\ 0]\,G_2 &= 00000000\\
[0\ 0\ 0\ 1]\,G_2 &= 00001111\\
[0\ 0\ 1\ 0]\,G_2 &= 00110011\\
[0\ 0\ 1\ 1]\,G_2 &= 00111100\\
[0\ 1\ 0\ 0]\,G_2 &= 01010101\\
[0\ 1\ 0\ 1]\,G_2 &= 01011010\\
[0\ 1\ 1\ 0]\,G_2 &= 01100110\\
[0\ 1\ 1\ 1]\,G_2 &= 01101001\\
[1\ 0\ 0\ 0]\,G_2 &= 11111111\\
[1\ 0\ 0\ 1]\,G_2 &= 11110000\\
[1\ 0\ 1\ 0]\,G_2 &= 11001100\\
[1\ 0\ 1\ 1]\,G_2 &= 11000011\\
[1\ 1\ 0\ 0]\,G_2 &= 10101010\\
[1\ 1\ 0\ 1]\,G_2 &= 10100101\\
[1\ 1\ 1\ 0]\,G_2 &= 10011001\\
[1\ 1\ 1\ 1]G_2 &= 10010110
\end{aligned}
$$

Thus, it is seen that $\boldsymbol{d_4}$, $\boldsymbol{d_5}$ and $\boldsymbol{d_6}$ are invalid codewords. So the interleaved vector produced by C_1 must have had the form

1011 1011 1011 ???? ???? ???? 1011 0110 0110 0110 0110 0110 0110 0110

where ? denotes a bit which may be incorrect. Hence we have

$$
\begin{aligned}
\boldsymbol{c_1} &= 111???1\\
\boldsymbol{c_2} &= 000???0\\
\boldsymbol{c_3} &= 111???1\\
\boldsymbol{c_4} &= 111???1\\
\boldsymbol{c_5} &= 0000000\\
\boldsymbol{c_6} &= 1111111\\
\boldsymbol{c_7} &= 1111111\\
\boldsymbol{c_8} &= 0000000
\end{aligned}
$$

Since C_1 has a minimum distance of 7 and the majority of symbols in the codewords $\boldsymbol{c_1}$, $\boldsymbol{c_2}$, $\boldsymbol{c_3}$ and $\boldsymbol{c_4}$ are either 1's or 0's, then we must have $\boldsymbol{c_1} = 1111111$, $\boldsymbol{c_2} = 0000000$, $\boldsymbol{c_3} = 1111111$ and $\boldsymbol{c_4} = 1111111$.

Finally, the first 8 messages bits, assuming that a single burst of length at most 9 had affected the transmission, will have been 10110110.