Proceeding Konferensi Nasional Matematika (KNM) XXI
Universitas Mataram, 18-19 Oktober 2022
Dipublikasikan online pada:
PATTIMURA PROCEEDING: Conference of Science and Technology
e-ISSN: 2829-3770, volume 4 issue 1, pp. 9-16, Maret 2023

Application of MDS Codes in Solving Problem of Distributing Exam Questions

Ricky Aditya[1, a)]

Department of Mathematics, Sanata Dharma University
Kampus III Paingan, Maguwoharjo, Depok, Sleman, Yogyakarta 55282

[a)]Corresponding author: rickyaditya@usd.ac.id

Abstract. In online learning, giving an objective assessment is quite tricky task. Most of exams in online learning are done as take-home exams. Unlike in face-to-face onsite class, in which the students can be observed directly when doing the exam, in online class the students have chance to cheat and work together to gain unfair advantage. To tackle this, the teachers need to modify the exam format. One possible solution is to create some variations of exam questions and distribute them to the students such that they do not get the same set of questions. However, we cannot create too many variations since it would make the grading process more difficult for the teacher. Thus, we need to find an optimal way to do so. In this article, we will discuss how Maximum Separable Distance (MDS) Codes in coding theory can be applied to provide a solution for this problem. Moreover, distribution patterns for class of size 27, 64 and 125 students will also be presented.

***Keywords**:* teaching strategies, cheating prevention, coding theory, MDS codes
2020 *Mathematical Subject Classification*: 11T71, 94B05, 94B25, 97D60

1. INTRODUCTION

Basically, cheating is an old problem in our education system. Many students with narrow-minded and pragmatic way of thinking tend to do everything to achieve good grades and not using their own abilities. Even in "classical" examination system by doing test onsite, under observation from the teacher, some students often find such a way to cheat. This problem already existed at various level of education, as explained in [1], [2] and [3], and it becomes worse in online learning because take-home exams format gives them more room to cheat and/or work together with other students. All exams should be done independently, but not many students doing so in take-home exams.

These situations make more difficulties for the teacher to give accurate and objective assessment to the students. Therefore, the teacher needs to be more creative in creating the exam questions to tackle any chance of cheating. Preventing students to cheat is not practically possible, but at least the unfair advantage they gain by cheating must be minimized. In other side, the teacher must make sure that the modified exam format is still relevant to assess specific skills and learning outcomes of the course.

Starting from this concern, the author tries to create a strategy to deal with this situation. The initial idea is creating variations of exam questions such that the set of questions done by the students are different one to each other. However, creating too many variations means creating too many exam questions and that just makes the teacher's work more difficult. The teacher will put too much extra effort in creating the exam questions and the grading will be some nightmare. Thus, an optimal solution is needed: how the teacher distributes the variations of exam questions in such way without having to create too many variations.

In solving this problem, the author tries to apply concept from coding theory. There are some terminologies in coding theory which can be associated with this exam questions distribution problem. The main concept used in this approach is the linear block codes, and more specifically to reach the optimality, the maximum distance separable (MDS) codes. Details of this approach and how it works will be discussed in later sections of this article.

2. THE PROBLEM OF DISTRIBUTING EXAM QUESTIONS

First, we need to describe the exam questions distribution problem in previous section more precisely and define the criteria of the expected solution. In college level, questions in mathematics exams are usually essays. We need to assess the students' conceptual understanding and they are expected to write the answer in detailed, structured, and systematic manner. Multiple choice and short answer questions are rarely found in college exams since they can be done by guessing and using some slices of luck.

Usually, the number of essay questions is either 4 or 5, done in two hours, depend on the difficulty level of the course material. Typically, those exam questions represent different topics, and the teachers want to assess how good the students in understanding all topics. For example, in Differential Calculus exam we want to create four questions: first question is about limits, second question is about continuous functions, third question is about derivatives and fourth question is about extreme value problem. Therefore, we cannot just pick 4 or 5 questions randomly from a "question bank".

The basic idea is to create some variations of each exam question with are assumed to have equal difficulty level. We may create few variations for each of those questions, namely type A, type B, etc. We want that question set for one student is different with other students, so they cannot just copy-paste their friend's work. For some questions, two different students might get to do the same variants, but for the other questions, they will do different variants. For example, in Differential Calculus exam, it is possible that two students get same variant for question about limits, but they get different variant for question about derivatives.

We do not expect to create too many variants for each question since it will be very difficult to grade. The most extreme case is when the teacher creates question variants as many as the number of students. This is both the best and the most stupid solution since the number of different exam questions will be increased extremely, and this will backfire the teacher and create a disaster when it comes to grading. Thus, we expect that the number of variants is low (let's say at least two and at most five), and then the problem is switched to how to distribute the variants such that any two students will not get too similar set of questions.

Therefore, based on previous explanation about the problem, we can describe some criteria for our expected solution, which are:

1. The exam consists of 4 or 5 different types of questions.
2. For each type of questions, at most 5 different variants are created.
3. Two sets of questions which have at least one question with different variants are considered as different ones.
4. Any two different sets of questions can only have at most two questions in common of the same variants.

To make it clear, let us identify the variants of first question as 1A, 1B and 1C, variants of second questions as 2A, 2B and 2C, and so on. Suppose that a student gets this set of questions: 1A-2B-3A-4C. Another student can get 1A-2A-3C-4A (only has one question in common, that is 1A), but should not get 1A-2A-3A-4C (has three questions in common: 1A, 3A and 4C). This condition must be satisfied for any possible pair of students, and this is the main challenge.

3. MAXIMUM DISTANCE SEPARABLE (MDS) CODES AND THEIR PROPERTIES

The main idea of this research is that properties of maximum distance separable (MDS) codes in coding theory have equivalent association with our expected solution. In this section we discuss MDS codes and their properties, which also explain why MDS codes can be a solution for the problem. Then we can construct some specific MDS codes and apply them to create an exam questions distribution patterns as the main result.

Coding theory is a branch of mathematics which is concerned with how to successfully transmit data through a noisy channel and correct errors in possible corrupted messages. In [4], concepts of coding theory can be based on various branches of mathematics such as linear algebra, abstract algebra, and combinatorics. In coding theory, a message is represented as a sequence of code symbols. One of the most familiar examples is the binary strings that is a sequence of binary digits (the code symbols is either 0 or 1).

Coding theory has two main processes: encoding and decoding. Encoding is the process of representing messages into codewords before they are sent through a communication channel. Decoding is the process of detecting and correcting errors in the received words. Errors might occur in the data transmission such that the received word is not same with the sent codeword. An encoding-decoding process is expected to be able to detect and correct as many as possible number of errors. Here are some definitions and terminologies which are used in coding theory:

Definition 1. ([4]) Let $A = \{a_1, a_2, \dots, a_q\}$ be a set of size q. A *word* of length n over A is a sequence $w = w_1 w_2 \dots w_n$ with each $w_i \in A$ for all i, which may also be written as a vector $(w_1, w_2, \dots, w_n)$. In this context, the set A is the *code alphabet* and its elements are called *code symbols*.

(i) A *block code* of length n over A is a nonempty set C of words of the same length n.
(ii) An element of C is called a *codeword* in C.
(iii) The *size* of C is the number of codewords in C, denoted by $|C|$.
(iv) Let x and y be words of length n over A. The *Hamming distance* between x and y, denoted by $d(x, y)$, is defined to be the number of different corresponding digits between x and y.
(v) If a block code C containing at least two codewords, then the *minimum distance* of C, denoted by $d(C)$, is the smallest possible Hamming distance between two different codewords in C.

Example 1. Let the code alphabet is $A = \{0,1\}$. Both $x = 01010$ and $y = 11111$ are words over A of length 5. Their Hamming distance is $d(x, y) = 3$, because they differ at the first, third and last digits.
The set $C = \{00000, 01010, 10101, 11111\}$ is a block code over A of size 4. Calculating the Hamming distance between two different codewords in C, we obtain: $d(00000, 01010) = 2$, $d(00000, 10101) = 3$, $d(00000, 11111) = 5$, $d(01010, 10101) = 5$, $d(01010, 11111) = 3$, and $d(10101, 11111) = 2$. Therefore, the minimum distance of C is $d(C) = 2$.

Minimum distance of a block code determines its error-detecting and error-correcting abilities when being used in data transmission. The basic principle of error-detection is that if the received word is not in the block code, then it is not a valid codeword and there must be some errors in transmission. For error-correction, we usually use nearest neighbor decoding principle, where the received word (containing some errors) is corrected to the codeword which has the smallest Hamming distance (i.e., has minimum number of digit differences) to the received word. The number of errors is the Hamming distance between the sent codeword and the received word. Based on these principles, we have this theorem:

Theorem 1. ([4]) A linear block code with minimum distance d can detect up to $d - 1$ errors and correct up to $\left\lfloor \frac{d-1}{2} \right\rfloor$ errors.

Since any words of length n over code alphabet A can be regarded as n-tuple vectors over A, naturally we can use the concept of linear algebra in coding theory. The block code can be constructed as a vector space over a finite field, and here the finite field be the code alphabet. For example, a binary block code can be constructed as a vector space over $\mathbb{Z}_2 = \{0,1\}$. The set of all binary sequences of length n is actually $(\mathbb{Z}_2)^n$ and any subspace of $(\mathbb{Z}_2)^n$ can be a binary block code of length n. Similarly, a ternary block code can be constructed as a vector space over $\mathbb{Z}_3 = \{0,1,2\}$, which is also a finite field.

Definition 2. ([4]) Let F_q denotes the finite field of order q. The set $(F_q)^n$ is a vector space over F_q with respect to standard vector addition and scalar multiplication (similar with in $\mathbb{R}^n$, but the operations are done in F_q).

(i) A *linear code* C of length n over F_q is a subspace of $(F_q)^n$.
(ii) The *dimension* of a linear code C over F_q is the dimension of C as a vector space over F_q, denoted by $\dim(C)$.
(iii) A linear code over F_q of length n, dimension k and minimum distance d is called as a *q-ary* $[n, k, d]$-*code*.

Example 2. In previous example, $C = \{00000, 01010, 10101, 11111\}$ is a linear code of length 5 over $F_2 = \mathbb{Z}_2$, or simply binary linear code of length 5. Its dimension is $\dim(C) = 2$, since it has $\{01010, 10101\}$ as basis. Therefore, C is a binary $[5,2,2]$-linear code.

Note that the context of basis and dimension here is the generalization from basis and dimension of real vector space to vector space over finite field. A linear code over F_q is the set of all possible linear combinations of its basis, where the scalars are in F_q. By counting the number of possible combinations, we have this theorem:

Theorem 2. ([4]) Let C be a linear code over F_q. If $\dim(C) = k$, then C contains q^k different codewords, i.e., $|C| = q^k$.

It is clear from Theorem 1 that codes with larger minimum distance will be better in error-detection and error-correction. Therefore, in coding theory, codes with large distance will be more preferred. However, construction methods for linear codes becomes trickier because of some constraints which come from the linear algebra itself. One of them is called the *Singleton bound*, as explained in the following theorem:

Theorem 3 ([4], Singleton bound). For any $[n, k, d]$-linear code over a finite field F_q, the parameters n, k and d satisfy the inequality:

$$d \leq n - k + 1$$

From Theorem 3, we find that the minimum distance of a linear code cannot exceed $n - k + 1$, where n is its length and k is its dimension. If k is bigger, then $n - k + 1$, an upper bound of d, is smaller, and vice versa. From Theorem 2, if k is bigger, then the linear code contains more codewords and moreover, can encode more different messages. However, it also implies that d is potentially smaller and therefore the error-detection and correction abilities are worse. If we want the linear code to have better error-detection and correction, then d must be bigger and k is potentially must be smaller. This will decrease the number of codewords and consequently unable to encode many different messages.

In short, the Singleton bound tells us that dimension and minimum distance have contradictive impact in terms of encoding and decoding performance. This leads to an idea to construct a such optimal code that has the best error-detection and correction abilities among any q-ary codes of same length and size.

Definition 3 ([4], MDS codes). A linear code with parameters $[n, k, d]$ such that $d = n - k + 1$ is called a *maximum distance separable* (MDS) code.

Example 3. These are examples of MDS codes:

(i) The binary linear code $C_1 = \{00000, 11111\}$ is a binary MDS code because its length is $n = 5$, its dimension is $k = 1$, and its distance is $d = 5 = n - k + 1$.

(ii) The ternary linear code $C_2 = \{0000, 0112, 0221, 1011, 1120, 1202, 2022, 2101, 2210\}$ is a ternary MDS code because it has length $n = 4$, dimension $k = 2$, and distance $d = 3 = n - k + 1$.

MDS codes reach the upper bound described in the Singleton bound. Thus, MDS codes are simply the best in error-detection and correction among any linear codes with the same parameters q, n and k. However, constructing MDS codes is a quite tricky task. In some cases, MDS codes with such choice of parameters might not exist. For big enough parameters, constructing MDS codes also becomes an interesting open problem. Many works on constructing MDS codes and their special classes have been done like in [5], [6], and [7], for examples. Sometimes any trial-and-error approach might be needed in constructing MDS codes.

4. MAIN RESULTS

Now we will apply some concepts of MDS codes for getting a solution for the exam questions distribution problem. The key here is that in an $[n, k, d]$-linear code, any two different codewords have at least d digit differences. This condition satisfies the fourth criteria: any two different sets of questions have a minimum number of different questions. We have learned that MDS codes are the most optimal codes among codes of the same length and dimension. This property makes the work of the teacher easier because the number of variants can be minimized. In this section we will see some examples of MDS codes and their associated questions distribution pattern, which can be implemented in class of 27, 64 and 125 students.

4.1. Exam Questions Distribution Pattern for Class of 27 Students

Let's say we want to create an exam of 4 essay questions with 3 variants each. This is equivalent with creating a set of 4-digit ternary words, i.e., words over $F_3 = \{0,1,2\}$. If we want to make two different sets of questions can only have at most two questions in common of the same variants, then any two different 4-digit ternary words must have at least two differences, or in other words, their Hamming distance is at least 2. If we want to create a ternary MDS code with length $n = 4$ and minimum distance $d = 2$, then its dimension must be $k = n - d + 1 = 3$. Furthermore, the code will contain $3^3 = 27$ codewords. An example of $[4,3,2]$-ternary linear code is the code generated by the set $\{1002,0102,0012\}$, which is:

$$\begin{Bmatrix} 0000,0012,0021,0102,0111,0120,0201,0210,0222, \\ 1002,1011,1020,1101,1110,1122,1200,1212,1221, \\ 2001,2010,2022,2100,2112,2121,2202,2211,2220 \end{Bmatrix}$$

We can show that in above set, the Hamming distance of any two different codewords is always ≥ 2.

If the first digit represents the variant of first question, the second digit represents the variant of second question, and so on, then we can "translate" above set into a distribution pattern. If 0 denotes variant A, 1 denotes variant B, and 2 denotes variant C, then we will have this distribution pattern:

TABLE 1. Exam Questions Distribution Pattern Based on A Ternary $[4,3,2]$-MDS Code

ID	Q1	Q2	Q3	Q4
1	1A	2A	3A	4A
2	1A	2A	3B	4C
3	1A	2A	3C	4B
4	1A	2B	3A	4C
5	1A	2B	3B	4B
6	1A	2B	3C	4A
7	1A	2C	3A	4B
8	1A	2C	3B	4A
9	1A	2C	3C	4C
10	1B	2A	3A	4C
11	1B	2A	3B	4B
12	1B	2A	3C	4A
13	1B	2B	3A	4B
14	1B	2B	3B	4A

ID	Q1	Q2	Q3	Q4
15	1B	2B	3C	4C
16	1B	2C	3A	4A
17	1B	2C	3B	4C
18	1B	2C	3C	4B
19	1C	2A	3A	4B
20	1C	2A	3B	4A
21	1C	2A	3C	4C
22	1C	2B	3A	4A
23	1C	2B	3B	4C
24	1C	2B	3C	4B
25	1C	2C	3A	4C
26	1C	2C	3B	4B
27	1C	2C	3C	4A

Distribution pattern described on Table 1 can be implemented in a class of 27 or less students. A class of 27 students is usually considered as a small one, while a class of more than 60 students is considered as a big one. If we work on larger class, we need to create more set of questions and consequently also need an MDS code of larger size. It will be possible if we increase the number of exam questions or the number of variations for each question.

4.2. Exam Questions Distribution Pattern for Class of 64 Students

Like in previous result, we can create a distribution pattern for a class of 64 students by constructing an MDS code over F_4 of length $n = 5$, dimension $k = 3$, and distance $d = n - k + 1 = 3$. In this case, the exam will consist of 5 questions and each question has 4 different variants. Furthermore, any two different students will get at least 3 questions of different variants, or equivalently, only have at most 2 questions in common of the same variants.

Remember that F_4, the finite field of order 4, is the $\mathbb{Z}_2[x]/< 1 + x + x^2 > = \{0, 1, x, 1 + x\}$, i.e., the set of binary polynomials modulo $1 + x + x^2$. Here we need a [5,3,3]-linear code over F_4, and one of its examples is the linear code with standard generator matrix:

$$\begin{bmatrix} 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & x \\ 0 & 0 & 1 & 1 & 1 + x \end{bmatrix}$$

which is explicitly the set:

$$\{(0,0,0,0,0), (0,0,1,1,1+x), (0,0,x,x,1), (0,0,1+x,1+x,x), (0,1,0,1,x), (0,1,1,0,1), (0,1,x,1+x,1+x),$$
$$(0,1,1+x,x,0), (0,x,0,x,1+x), (0,x,1,1+x,0), (0,x,x,0,x), (0,x,1+x,1,1), (0,1+x,0,1+x,1),$$
$$(0,1+x,1,x,x), (0,1+x,x,1,0), (0,1+x,1+x,0,1+x), (1,0,0,1,1), (1,0,1,0,x), (1,0,x,1+x,0),$$
$$(1,0,1+x,x,1+x), (1,1,0,0,1+x), (1,1,1,1,0), (1,1,x,x,x), (1,1,1+x,1+x,1), (1,x,0,1+x,x),$$
$$(1,x,1,x,1), (1,x,x,1,1+x), (1,x,1+x,0,0), (1,1+x,0,x,0), (1,1+x,1,1+x,1+x), (1,1+x,x,0,1),$$
$$(1,1+x,1+x,1,x), (x,0,0,x,x), (x,0,1,1+x,1), (x,0,x,0,1+x), (x,0,1+x,1,0), (x,1,0,1+x,0),$$
$$(x,1,1,x,1+x), (x,1,x,1,1), (x,1,1+x,0,x), (x,x,0,0,1), (x,x,1,1,x), (x,x,x,x,0), (x,x,1+x,1+x,1+x),$$
$$(x,1+x,0,1,1+x), (x,1+x,1,0,0), (x,1+x,x,1+x,x), (x,1+x,1+x,x,1), (1+x,0,0,1+x,1+x),$$
$$(1+x,0,1,x,0), (1+x,0,x,1,x), (1+x,0,1+x,0,1), (1+x,1,0,x,1), (1+x,1,1,1+x,x), (1+x,1,x,0,0),$$
$$(1+x,1,1+x,1,1+x), (1+x,x,0,1,0), (1+x,x,1,0,1+x), (1+x,x,x,1+x,1), (1+x,x,1+x,x,x),$$
$$(1+x,1+x,0,0,x), (1+x,1+x,1,1,1), (1+x,1+x,x,x,1+x), (1+x,1+x,1+x,1+x,0)\}$$

Here $0-1-x-(1+x)$ will then be replaced by A-B-C-D, respectively, to identify the variants. The associated distribution pattern is presented in the following table.

TABLE 2. Exam Questions Distribution Pattern Based on A $[5,3,3]$-MDS Code over F_4

ID	Q1	Q2	Q3	Q4	Q5	ID	Q1	Q2	Q3	Q4	Q5
1	A	A	A	A	A	33	C	A	A	C	C
2	A	A	B	B	D	34	C	A	B	D	B
3	A	A	C	C	B	35	C	A	C	A	D
4	A	A	D	D	C	36	C	A	D	B	A
5	A	B	A	B	C	37	C	B	A	D	A
6	A	B	B	A	B	38	C	B	B	C	D
7	A	B	C	D	D	39	C	B	C	B	B
8	A	B	D	C	A	40	C	B	D	A	C
9	A	C	A	C	D	41	C	C	A	A	B
10	A	C	B	D	A	42	C	C	B	B	C
11	A	C	C	A	C	43	C	C	C	C	A
12	A	C	D	B	B	44	C	C	D	D	D
13	A	D	A	D	B	45	C	D	A	B	D
14	A	D	B	C	C	46	C	D	B	A	A
15	A	D	C	B	A	47	C	D	C	D	C
16	A	D	D	A	D	48	C	D	D	C	B
17	B	A	A	B	B	49	D	A	A	D	D
18	B	A	B	A	C	50	D	A	B	C	A
19	B	A	C	D	A	51	D	A	C	B	C
20	B	A	D	C	D	52	D	A	D	A	B
21	B	B	A	A	D	53	D	B	A	C	B
22	B	B	B	B	A	54	D	B	B	D	C
23	B	B	C	C	C	55	D	B	C	A	A
24	B	B	D	D	B	56	D	B	D	B	D
25	B	C	A	D	C	57	D	C	A	B	A
26	B	C	B	C	B	58	D	C	B	A	D
27	B	C	C	B	D	59	D	C	C	D	B
28	B	C	D	A	A	60	D	C	D	C	C
29	B	D	A	C	A	61	D	D	A	A	C
30	B	D	B	D	D	62	D	D	B	B	B
31	B	D	C	A	B	63	D	D	C	C	D
32	B	D	D	B	C	64	D	D	D	D	A

Distribution pattern described on Table 2 can be implemented in a class of 64 or less students. In some cases, the number of students in a class can be more than 64. So, we might need another MDS code of bigger size. If increasing the number of questions is not always feasible for some courses, then we must increase the number of variants for each question, i.e., work on MDS codes over bigger finite field.

4.3. Exam Questions Distribution Pattern for Class of 125 Students

Now we will create a distribution pattern for a class of 125 students by constructing an MDS code over F_5 of length $n = 5$, dimension $k = 3$, and distance $d = n - k + 1 = 3$. Like the previous one, the exam will consist of 5 questions, but now each question has 5 different variants. Remember that F_5, the finite field of order 5, is the $\mathbb{Z}_5 = \{0,1,2,3,4\}$. Here we need a [5,3,3]-linear code over F_5, and one of its examples is the linear code with standard generator matrix:

$$\begin{bmatrix} 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 2 \\ 0 & 0 & 1 & 1 & 3 \end{bmatrix}$$

which is explicitly the set:

$\{00000,00113,00221,00334,00442,01012,01120,01233,01341,01404,02024,02132,02240,02303,02411,$
$03031,03144,03202,03310,03423,04043,04101,04214,04322,04430,10011,10124,10232,10340,10403,$
$11023,11131,11244,11302,11410,12030,12143,12201,12314,12422,13042,13100,13213,13321,13434,$
$14004,14112,14220,14333,14441,20022,20130,20243,20301,20414,21034,21142,21200,21313,21421,$
$22041,22104,22212,22320,22433,23003,23111,23224,23332,23440,24010,24123,24231,24344,24402,$
$30033,30141,30204,30312,30420,31040,31103,31211,31324,31432,32002,32110,32223,32331,32444,$
$33014,33122,33230,33343,33401,34021,34134,34242,34300,34413,40044,40102,40210,40323,40431,$
$41001,41114,41222,41330,41443,42013,42121,42234,42342,42400,$
$43020,43133,43241,43304,43412,44032,44140,44203,44311,44424\}$

Here 0-1-2-3-4 will then be replaced by A-B-C-D-E, respectively, to identify the variants. The associated distribution pattern is presented in the following table.

TABLE 3. Exam Questions Distribution Pattern Based on A [5,3,3]-MDS Code over F_5

ID	Q1	Q2	Q3	Q4	Q5
1	A	A	A	A	A
2	A	A	B	B	D
3	A	A	C	C	B
4	A	A	D	D	E
5	A	A	E	E	C
6	A	B	A	B	C
7	A	B	B	C	A
8	A	B	C	D	D
9	A	B	D	E	B
10	A	B	E	A	E
11	A	C	A	C	E
12	A	C	B	D	C
13	A	C	C	E	A
14	A	C	D	A	D
15	A	C	E	B	B
16	A	D	A	D	B
17	A	D	B	E	E
18	A	D	C	A	C
19	A	D	D	B	A
20	A	D	E	C	D
21	A	E	A	E	D
22	A	E	B	A	B
23	A	E	C	B	E
24	A	E	D	C	C
25	A	E	E	D	A
26	B	A	A	B	B
27	B	A	B	C	E
28	B	A	C	D	C
29	B	A	D	E	A
30	B	A	E	A	D
43	B	D	C	B	D
44	B	D	D	C	B
45	B	D	E	D	E
46	B	E	A	A	E
47	B	E	B	B	C
48	B	E	C	C	A
49	B	E	D	D	D
50	B	E	E	E	B
51	C	A	A	C	C
52	C	A	B	D	A
53	C	A	C	E	D
54	C	A	D	A	B
55	C	A	E	B	E
56	C	B	A	D	E
57	C	B	B	E	C
58	C	B	C	A	A
59	C	B	D	B	D
60	C	B	E	C	B
61	C	C	A	E	B
62	C	C	B	A	E
63	C	C	C	B	C
64	C	C	D	C	A
65	C	C	E	D	D
66	C	D	A	A	D
67	C	D	B	B	B
68	C	D	C	C	E
69	C	D	D	D	C
70	C	D	E	E	A
71	C	E	A	B	A
72	C	E	B	C	D
85	D	B	E	D	C
86	D	C	A	A	C
87	D	C	B	B	A
88	D	C	C	C	D
89	D	C	D	D	B
90	D	C	E	E	E
91	D	D	A	B	E
92	D	D	B	C	C
93	D	D	C	D	A
94	D	D	D	E	D
95	D	D	E	A	B
96	D	E	A	C	B
97	D	E	B	D	E
98	D	E	C	E	C
99	D	E	D	A	A
100	D	E	E	B	D
101	E	A	A	E	E
102	E	A	B	A	C
103	E	A	C	B	A
104	E	A	D	C	D
105	E	A	E	D	B
106	E	B	A	A	B
107	E	B	B	B	E
108	E	B	C	C	C
109	E	B	D	D	A
110	E	B	E	E	D
111	E	C	A	B	D
112	E	C	B	C	B
113	E	C	C	D	E
114	E	C	D	E	C

31	B	B	A	C	D
32	B	B	B	D	B
33	B	B	C	E	E
34	B	B	D	A	C
35	B	B	E	B	A
36	B	C	A	D	A
37	B	C	B	E	D
38	B	C	C	A	B
39	B	C	D	B	E
40	B	C	E	C	C
41	B	D	A	E	C
42	B	D	B	A	A

73	C	E	C	D	B
74	C	E	D	E	E
75	C	E	E	A	C
76	D	A	A	D	D
77	D	A	B	E	B
78	D	A	C	A	E
79	D	A	D	B	C
80	D	A	E	C	A
81	D	B	A	E	A
82	D	B	B	A	D
83	D	B	C	B	B
84	D	B	D	C	E

115	E	C	E	A	A
116	E	D	A	C	A
117	E	D	B	D	D
118	E	D	C	E	B
119	E	D	D	A	E
120	E	D	E	B	C
121	E	E	A	D	C
122	E	E	B	E	A
123	E	E	C	A	D
124	E	E	D	B	B
125	E	E	E	C	E

Distribution pattern described on Table 3 can be implemented in a class of 125 or less students. It can be shown that differences between any two different IDs in that distribution pattern is at least 3. We will not create another bigger MDS code since it is unlikely to have a class of more than 125 students. We can simply say that the three examples of MDS codes presented in this section have just solved our problems.

5. CONCLUSION

From the discussion in previous sections, we can summarize some important points as the conclusion of this article as follow:

1. In take-home exams, the students have bigger chance to cheat and/or work together to gain unfair advantage. One strategy to tackle it is by creating some variations for each type of exam questions. Those variations need to be distributed in such way to ensure that any two students will get different set of questions with significant number of differences.
2. Coding theory can be applied to solve this exam questions distribution problem since the distribution pattern can be associated with block codes. In the context of this problem, the distance between two codewords is associated with the number of differences between two different set of questions.
3. MDS codes have the maximum distance parameter among linear codes of same code alphabet, length, and dimension. This optimality can help the teacher in creating variations of exam questions, because the number of variants can be minimized.
4. An example of distribution pattern for small class of up to 27 students has been made using ternary MDS code with parameters $[4,3,2]$ and it satisfies all criteria of the expected solution. Another example based on an MDS code over F_4 with parameters $[5,3,3]$ can be implemented in class of up to 64 students. For a big class of up to 125 students, an MDS code over F_5 with parameters $[5,3,3]$ can be used.

6. REFERENCES

[1] G. Ednadita, S. Octavia, F. S. Khairunnisa, I. Rodhiyah and D. Hendraputra, "Effects of Self-Control on Cheating Among Indonesian College Students," Indonesian Psychological Research, vol. 2, no. 2, pp. 87-95, July 2020.

[2] K. A. D'Souza and D. V. Siegfeld, "A Conceptual Framework for Detecting Cheating in Online and Take-Home Exams," *Decision Sciences Journal of Innovative Education*, vol. 15 (4), pp. 370-391, October 2017.

[3] M. Valizadeh, "Cheating in Online Learning Programs: Learners' Perceptions and Solutions," *Turkish Online Journal of Distance Education*, Vol. 23 no. 1, pp. 195-209, January 2022.

[4] S. Ling and C. Xing, *Coding Theory: A First Course*. New York: Cambridge University Press, 2004.

[5] L. Jin and C. Xing, "New MDS Self-Dual Codes from Generalized Reed-Solomon Codes," *IEEE Transactions on Information Theory*, vol. 63 (3), March 2017.

[6] H. Yan, "A Note on the Construction of MDS Self-Dual Codes," *Cryptography and Communications,* vol. 11 (2), pp. 259-268, March 2019.

[7] A. Zhang and K. Feng, "A Unified Approach to Construct MDS Self-Dual Codes via Reed-Solomon Codes," *IEEE Transactions on Information Theory*, vol. 66 (6), June 2020.