The Fifth International Students’ Olympiad in Cryptography — NSUCRYPTO:
problems and their solutions

A. Gorodilova¹,², S. Agievich³, C. Carlet⁴, X. Hou⁵, V. Idrisova¹,², N. Kolomeec¹,², A. Kutsenko¹, L. Mariot⁶, A. Oblaukhov¹,², S. Picek⁷, B. Preneel⁸, R. Rosie⁹, N. Tokareva¹,²

¹Novosibirsk State University, Novosibirsk, Russia
²Sobolev Institute of Mathematics, Novosibirsk, Russia
³Belarusian State University, Minsk, Belarus
⁴University of Paris 8, Paris, France
⁵University of South Florida, Tampa, FL 33620, United States of America
⁶DISCo, Università degli Studi di Milano-Bicocca, Viale Sarca 336/14, 20126 Milano, Italy
⁷Cyber Security Research Group, Delft University of Technology, Mekelweg 2, Delft, The Netherlands
⁸ESAT-COSIC, KU Leuven, Leuven, Belgium
⁹University of Luxembourg, Luxembourg, Luxembourg

E-mail: nsucrypto@nsu.ru

Abstract. Problems and their solutions of the Fifth International Students’ Olympiad in cryptography NSUCRYPTO’2018 are presented. We consider problems related to attacks on ciphers and hash functions, Boolean functions, quantum circuits, Enigma, etc. We discuss several open problems on orthogonal arrays, Sylvester matrices and disjunct matrices. The problem of existing an invertible Sylvester matrix whose inverse is again a Sylvester matrix was completely solved during the Olympiad.

Keywords. cryptography, ciphers, hash functions, Enigma, quantum circuits, metrically regular sets, irreducible polynomials, orthogonal arrays, Sylvester matrices, disjunct matrices, Olympiad, NSUCRYPTO.

*The paper was supported by the Russian Ministry of Science and Education (the 5-100 Excellence Programme and the Project no. 1.13559.2019/13.1), by the Russian Foundation for Basic Research (projects no. 18-07-01394, 18-31-00479, 18-31-00374), by the program of fundamental scientific researches of the SB RAS no. 1.5.1, project no. 0314-2019-0017.
Introduction

NSUCRYPTO — The International Students’ Olympiad in cryptography — celebrated its 5-year anniversary in 2018. The geography of the Olympiad is significant: there were more than 1600 participants from 52 countries in the first five Olympiads from 2014 to 2018! The Olympiad program committee includes specialists from Belgium, France, The Netherlands, USA, Norway, India, Belarus’, Russia.

Let us shortly formulate the format of the Olympiad. One of the Olympiad main ideas is that everyone can participate: school students, university students and professionals! Each participant chooses his/her category when registering on the Olympiad website nsucrypto.nsu.ru. The Olympiad consists of two independent Internet rounds: the first one is individual while the second round is team. The first round is divided into two sections: A — for school students, B — for university students and professionals. Participants read the Olympiad problems and send their solutions using the Olympiad website. The language of the Olympiad is English.

The Olympiad participants are always interested in solving different problems of various complexity at the intersection of mathematics and cryptography. They show their knowledge, creativity and professionalism. That is why the Olympiad not only includes interesting tasks with known solutions but also offers unsolved problems in this area. This year, one of such open problems, “Sylvester matrices”, was completely solved by three teams! All the open problems stated during the Olympiad history can be found at nsucrypto.nsu.ru/unsolved-problems. On the website we also mark the current status of each problem. For example, as well as “Sylvester matrices” was solved in 2018, the problem “Algebraic immunity” was completely solved during the Olympiad in 2016. And what is important for us, some participants are trying to find solutions after Olympians. For example, a partial solution for the problem “A secret sharing” (2014) was proposed in [7]. We invite everybody who has ideas how to solve the problems to send your solutions to us!

The paper is organized as follows. We start with problem structure of the Olympiad in section 1. Then we present formulations of all the problems stated during the Olympiad and give their detailed solutions (section 2). Finally, we publish the lists of NSUCRYPTO’2018 winners in section 3.

Mathematical problems of the previous International Olympiads NSUCRYPTO’2014, NSUCRYPTO’2015, NSUCRYPTO’2016, and NSUCRYPTO’2017 can be found in [2], [1], [11], and [8] respectively.

1 Problem structure of the Olympiad

There were 16 problems stated during the Olympiad, some of them were included in both rounds (Tables 1, 2). The school section of the first round consisted of six problems, whereas the student section contained seven problems. Three problems were common for both sections. The second round was composed of eleven problems; they were common for all the participants. Three problems of the second round were marked as unsolved (awarded special prizes from the Program Committee).

2 Problems and their solutions

In this section we formulate all the problems of NSUCRYPTO’2018 and present their detailed solutions with paying attention to solutions proposed by the participants.
2.1 Problem “A digital signature”

2.1.1 Formulation

Alice uses a new digital signature algorithm, that turns a text message $M$ into a pair $(M, s)$, where $s$ is an integer and generated in the following way:

1. The special function $h$ transforms $M$ into a big positive integer $r = h(M)$.
2. The number $t = r^2$ is calculated, where $t = t_1t_2 \ldots t_n$.
3. The signature $s$ is calculated as $s = t_1 + t_2 + \ldots + t_n$.

Bob obtained the signed message

(Congratulations on the fifth year anniversary of NSUCRYPTO!, 2018)

from Alice and immediately recognized that something was wrong with the signature! How did he discover it?

**Remarks.** By $t = t_1t_2 \ldots t_n$ we mean that $t_1, t_2, \ldots, t_n$ are decimal digits and all digits over the bar form decimal number $t$.

2.1.2 Solution

It is widely known that every integer is congruent to the sum of its digits modulo 3. So, we have that $t \equiv 3 2018 \equiv 2$. But $t$ is equal to $r^2$ and a square can not be equal to 2 modulo 3. Thus, we have a contradiction.

We got a lot correct solutions. The most accurate and detailed solutions were sent by Ruxandra Icleanu (Tudor Vianu National College of Computer Science, Romania), by Petr Ionov (Yaroslavl State University, Russia), by Henning Seidler and Katja Stumpp (TU Berlin, Germany).
2.2 Problem “Jack and the Beanstalk”

2.2.1 Formulation

Little Jack is only 7 years old and likes solving riddles involving the powers of 2. Recently, his uncle Bitoshi gave him sixteen BeanCoin seeds and promised that Jack can collect all BeanCoins which will grow from these seeds. But in order for BeanCoins to grow big and fruitful, Jack must plant the seeds in the garden in a special way. He have to draw eight lines on the ground and plant all sixteen seeds on these lines in such a way that each of the lines contains exactly four seeds.

Can you help Jack to achieve his goal and suggest how to plant the seeds?

2.2.2 Solution

The seeds can be place on the corners and intersection points of an octagram, as depicted in Figure 1. Please note that the main requirement was to show that no other line contained exactly four seeds.

Many school students found interesting solutions how to draw these lines. The most interesting ones were given by Gorazd Dimitrov (Yahya Kemal College, Macedonia), Artem Ismagilov (The Specialized Educational and Scientific Center UrFU, Russia) and Igor Pastushenko (The Specialized Educational Scientific Center of Novosibirsk State University, Russia).

2.3 Problem “Key matrices”

2.3.1 Formulation

Let $n$ be an odd positive integer. In some cipher, a key is a binary $n \times n$ matrix

\[
A = \begin{pmatrix}
a_{1,1} & a_{1,2} & \ldots & a_{1,n} \\
a_{2,1} & a_{2,2} & \ldots & a_{2,n} \\
\vdots & \vdots & \ddots & \vdots \\
a_{n,1} & a_{n,2} & \ldots & a_{n,n}
\end{pmatrix},
\]

where $a_{i,j}$ is either 0 or 1, such that each diagonal of any length 1, 2, $\ldots$, $n-1, n$ contains an odd number of 1s.

What is the minimal and the maximal number of 1s that can be placed in a key matrix $A$?

**Remarks.** For example, for $n = 3$, diagonals are the following six lines:

\[
A = \begin{pmatrix}
\alpha_{1,1} & \alpha_{1,2} & \alpha_{1,3} \\
\alpha_{2,1} & \alpha_{2,2} & \alpha_{2,3} \\
\alpha_{3,1} & \alpha_{3,2} & \alpha_{3,3}
\end{pmatrix}
\]

2.3.2 Solution

The correct solution of this problem must consist of two steps. The first step is to find theoretical lower and upper bounds for the number of 1s, and the second step is to prove that these bounds are tight. The best solution was proposed by Aleksei Udovenko (University of Luxembourg), we provide it below.

1. **Minimum.** Consider the $n \times n$ matrix $A$ ($n$ is odd) with both the top row filled with 1s, the bottom row filled with 1s and the central cell equal to 1; all other elements are 0:
\[
\begin{align*}
\begin{cases}
a_{1,i} = 1, & 1 \leq i \leq n; \\
a_{n,i} = 1, & 1 \leq i \leq n; \\
a_{(n+1)/2,(n+1)/2} = 1; \\
a_{i,j} = 0, & \text{otherwise.}
\end{cases}
\end{align*}
\]

Any diagonal of length less than \( n - 1 \) includes exactly single 1 (either from the top row or from the bottom row). The two diagonals of length \( n \) include three 1s (one from the top row, one from the bottom row and one from the center). Therefore, this matrix satisfies the condition. It has \( 2n + 1 \) ones.

We now prove that this number of ones is minimal. Note that the corner cells \( a_{1,1}, a_{1,n}, a_{n,1}, a_{n,n} \) make a single element diagonals. Therefore, these cells must contain 1s. There are \( 2(n - 2) \) diagonals going in the down-right direction and not touching the corners (starting from the cells of the leftmost column and from the cells for the topmost row). Furthermore, the main diagonal without the corner cells must have odd number of ones too. Therefore, \( 2n - 3 \) disjoint diagonals must contain at least one 1, in addition to 4 corner 1s. Therefore, there should be at least \( 2(n - 2) + 1 + 4 = 2n + 1 \) ones in the matrix.

2. **Maximum.** Consider the \( n \times n \) matrix \( A \) (\( n \) is odd) filled with 1s except cells in the leftmost and the rightmost columns which have an even row index:

\[
\begin{align*}
\begin{cases}
a_{2i,1} = 0, & 1 \leq i \leq (n - 1)/2; \\
a_{2i,n} = 0, & 1 \leq i \leq (n - 1)/2; \\
a_{i,j} = 1, & \text{otherwise.}
\end{cases}
\end{align*}
\]

It is easy to check that all diagonals that contain an even number of elements contain a single zero either from the leftmost or from the rightmost column. Therefore, these diagonals have an odd number of 1s. Also, all diagonals that contain an odd number of elements contain no zeroes and thus have an odd number of 1s too. Therefore, this matrix satisfies the condition. It has \( n^2 - 2(n - 1)/2 = n^2 - n + 1 \) ones.

We now prove that this number is maximal. Consider diagonals going in the down-right direction that have an even number of elements. There are \( 2(n - 1)/2 = (n - 1) \) such diagonals and they are disjoint. Each of them must contain at least a single zero. Therefore, the maximum number of ones is \( n^2 - n + 1 \).

2.4 **Problem “A sequence”**

2.4.1 **Formulation**

Two friends, Roman and Anton, are very interested in sequences and ciphers. Their new cryptosystem encrypts binary messages of length \( n \), \( X = (x_1, x_2, \ldots, x_n) \), where each \( x_i \) is either 0 or 1. A key \( K \) of the cipher is a set of \( n \) integers \( a_1, a_2, \ldots, a_n \). The ciphertext \( Y \) for the message \( X \) encrypted with the key \( K \) is the integer

\[
Y = x_1 \cdot a_1 + x_2 \cdot a_2 + \ldots + x_n \cdot a_n.
\]

Roman and Anton change their key regularly. Today, the key \( K \) is defined by

\[
a_i = 2^i + (-1)^i \text{ for all } i = 1, \ldots, n.
\]

The friends can easily decipher any message using the key defined by this sequence for any \( n! \)
1. Prove that the encryption is correct for this key $K$ for any $n$: there are no two distinct input messages $X^1$ and $X^2$ such that their ciphertexts $Y^1$ and $Y^2$ are equal, i.e., $Y^1 = Y^2$.

2. Describe an algorithm which can be used to easily decipher any ciphertext $Y$ encrypted with today’s key $K$. Here “easily” means that the algorithm should work much faster than checking all possible variants for an input message $X$.

2.4.2 Solution

Let us firstly show that the sequence $\{a_i\}$ is superincreasing, i.e. $a_{i+1} > \sum_{k=1}^{i} a_k$ for any $i > 0$. Indeed,

$$\sum_{k=1}^{i} a_k = \sum_{k=1}^{i} (2^k + (-1)^k) = 2^{i+1} - 2 + \sum_{k=1}^{i} (-1)^k = \begin{cases} 2^{i+1} - 2, & \text{if } i \text{ is even} \\ 2^{i+1} - 3, & \text{if } i \text{ is odd} \end{cases} < 2^{i+1} + (-1)^i = a_{i+1}.$$ 

1. Let us show that the encryption is correct. Let $X^1 = (x_1^1, \ldots, x_n^1)$ and $X^2 = (x_1^2, \ldots, x_n^2)$ be two distinct messages, and $i$ is the largest position such that $x_i^1 \neq x_i^2$. Without loss of generality, suppose that $x_i^1 = 1$. Then

$$Y^1 - Y^2 = (x_1^1 \cdot a_1 + \ldots + x_i^1 \cdot a_i + \ldots + x_n^1 \cdot a_n) - (x_1^2 \cdot a_1 + \ldots + x_i^2 \cdot a_i + \ldots + x_n^2 \cdot a_n)$$

$$= (x_1^1 - x_i^2) \cdot a_1 + \ldots + (x_{i-1}^1 - x_{i-1}^2) \cdot a_{i-1} + a_i > 0$$

since $\{a_i\}$ is a superincreasing sequence.

2. The correctness of the decryption algorithm (Algorithm 1) is also based on the superincreasing property of $\{a_i\}$. The complexity of the algorithm consists of $n$ integer comparisons.

\begin{algorithm}
\caption{The decryption algorithm}
\begin{algorithmic}
\STATE Input: $Y$, $n$.
\STATE Output: $X = (x_1, \ldots, x_n)$.
\STATE Step 0. $T := Y$, $i := n$.
\STATE Step 1. If $T > a_i$, then $x_i = 1$; else $x_i = 0$.
\STATE Step 2. $T := T - x_i \cdot a_i$, $i := i - 1$. If $i > 0$, go to step 1; else return $X$.
\end{algorithmic}
\end{algorithm}

The problem was solved by the majority of participants including eight school students.

2.5 Problem “Solutions of the equation”

2.5.1 Formulation

Alice is studying special functions that are used in symmetric ciphers. Let $E^n$ be the set of all binary vectors $x = (x_1, x_2, \ldots, x_n)$ of length $n$, where $x_i$ is either 0 or 1. Given two vectors $x$ and $y$ from $E^n$ consider their sum $x \oplus y = (x_1 \oplus y_1, \ldots, x_n \oplus y_n)$, where $\oplus$ is addition modulo 2.

Example. If $n = 3$, then $E^3 = \{(000), (001), (010), (011), (100), (101), (110), (111)\}$. Let $x = (010)$ and $y = (011)$, then vector $x \oplus y$ is equal to $(010) \oplus (011) = (0 \oplus 0, 1 \oplus 1, 0 \oplus 1) = (011)$.

We will say that a function $F$ maps $E^n$ to $E^n$ if it transforms any vector $x$ from $E^n$ into some vector $F(x)$ from $E^n$.

Example. Let $n = 2$. For instance, we can define $F$ that maps $E^2$ to $E^2$ as follows: $F(00) = (00)$, $F(01) = (10)$, $F(10) = (11)$ and $F(11) = (10)$.
Alice found a function $S$ that maps $E^6$ to $E^6$ in such a way that the vectors $S(x)$ and $S(y)$ are not equal for any nonequal vectors $x$ and $y$. Also, $S$ has another curious property: the equation

$$S(x) \oplus S(x \oplus a) = b$$

has either 0 or 2 solutions for any nonzero vector $a$ from $E^6$ and any vector $b$ from $E^6$.

Find the number of pairs $(a, b)$ such that this equation has exactly 2 solutions!

### 2.5.2 Solution

Consider a function $S$ that satisfies the conditions of the problem. Let us fix an arbitrary value $a$ that is nonzero. Consider the set of all possible values of $S(x) \oplus S(x \oplus a)$:

$$B_a = \{S(x) \oplus S(x \oplus a) \mid x \in E^6\} = 2^5,$$

since $S(x) \oplus S(x \oplus a) = S(x \oplus a) \oplus S(x \oplus a \oplus a)$. Then for every nonzero $a$ there exist $2^5$ values of $b$, such that $S(x) \oplus S(x \oplus a) = b$ has 2 solutions. Then the number of pairs is equal to $63 \times 32 = 2016$.

Correct answers were sent by only three school students: Alexey Lvov (Gymnasium 6 of Novosibirsk, Russia), Borislav Kirilov (The First Private Mathematical Gymnasium of Sofia, Bulgaria) and Razvan Andrei Draghici (National College Fratii Buzesti, Romania).

### 2.6 Problem “Quantum circuits”

#### 2.6.1 Formulation

Alice and Bob are interested in quantum circuits. They studied quantum operations and would like to use them for their simple cipher. Let an input plaintext be $P = (p_1, p_2, \ldots, p_{16}) \in \mathbb{F}_2^{16}$. The ciphertext $C \in \mathbb{F}_2^{16}$ is calculated as

$$C = K \oplus (F(p_1, \ldots, p_4), F(p_5, \ldots, p_8), F(p_9, \ldots, p_{12}), F(p_{13}, \ldots, p_{16})),$$

where $K \in \mathbb{F}_2^{16}$ is a secret key and $F$ is a function from $\mathbb{F}_2^4$ to $\mathbb{F}_2^4$; $\oplus$ is bitwise XOR.

The friends found a representation of $F$ from wires and elementary quantum gates which form a quantum circuit. They use Dirac notation and denote computational basis states by $|0\rangle$ and $|1\rangle$. Further, quantum bits (qubits) are considered only in quantum states $|0\rangle$ and $|1\rangle$. Alice and Bob used the following quantum gates and circuit symbols which are given in Table 3.

| Table 3: Quantum gates and circuit symbols |
|-------------------------------------------|
| **Pauli-X gate** | $|x\rangle \rightarrow X \rightarrow |x\rangle \oplus 1\rangle$ | acts on a single qubit in the state $|x\rangle$, $x \in \{0, 1\}$. |
| **controlled-NOT gate (CNOT gate)** | $|x\rangle \rightarrow \bigotimes x \rightarrow |x\rangle \oplus y\rangle$ | acts on two qubits in the states $|x\rangle$, $|y\rangle$, $x, y \in \{0, 1\}$; it flips the second qubit if and only if the first qubit is in the state $|1\rangle$. |
| **Toffoli gate (CCNOT gate)** | $|x\rangle \rightarrow \bigotimes x \rightarrow |x\rangle \oplus y\rangle \oplus z\rangle$ | acts on three qubits in the states $|x\rangle$, $|y\rangle$, $|z\rangle$, $x, y, z \in \{0, 1\}$; it flips the third qubit if and only if the states of the first and the second qubits are both equal to $|1\rangle$. |
| $|x\rangle \rightarrow \bigotimes x$ | a measurement of a qubit in the state $|x\rangle$, $x \in \{0, 1\}$, in the computational basis $\{|0\rangle, |1\rangle\}$. |
| | a wire carrying a single qubit (time goes left to right). |
| | a wire carrying a single classical bit. |

A quantum circuit which describes action of $F$ on $x = (x_1, x_2, x_3, x_4) \in \mathbb{F}_2^4$, where $F = (f_1, f_2, f_3, f_4)$ and $f_i, i = 1, 2, 3, 4$, are Boolean functions in 4 variables, is the following:
The problem. The friends encrypted the plaintext $P = (0011010111110010)$ and got the ciphertext $C = (1001101010010010)$. Find the secret key $K$!

2.6.2 Solution

One can notice that the given circuit can be simplified by observing that the following evolutions

actually swap two states $|x\rangle, |y\rangle$, $x, y \in \{0, 1\}$:

Both of the evolutions

have form

for $|x\rangle, |y\rangle$, $x, y \in \{0, 1\}$.

The algebraic normal forms of coordinate Boolean functions of $F$ are

$$f_1(x) = x_1 \oplus x_2 x_3,$$
$$f_2(x) = x_2 \oplus x_1 x_4 \oplus x_2 x_3 x_4 \oplus 1,$$
$$f_3(x) = x_3 \oplus x_4 \oplus x_1 x_2 \oplus x_1 x_3,$$
$$f_4(x) = x_4 \oplus 1,$$
where \( x \in \mathbb{F}^4_2 \). Then

\[
K_{1,...,4} = C_{1,...,4} \oplus F(p_1, \ldots, p_4) = C_{1,...,4} \oplus (0100) = (1101),
\]
\[
K_{5,...,8} = C_{5,...,8} \oplus F(p_5, \ldots, p_8) = C_{5,...,8} \oplus (0010) = (1000),
\]
\[
K_{9,...,12} = C_{9,...,12} \oplus F(p_9, \ldots, p_{12}) = C_{9,...,12} \oplus (0000) = (1001),
\]
\[
K_{13,...,16} = C_{13,...,16} \oplus F(p_{13}, \ldots, p_{16}) = C_{13,...,16} \oplus (0111) = (0101),
\]

and finally, the key is the following:

\[
K = (1101100010010101).
\]

Many participants coped with this problem and correctly found the key.

### 2.7 Problem “Stickers”

#### 2.7.1 Formulation

Bob always takes into account all the recommendations of security experts. He switched from short passwords to long passphrases and changes them every month. Bob usually chooses passphrases from the books he is reading. Passphrases are so lengthy and are changed so often! In order to not forget them, Bob decided to use stickers with hints. He places them on his monitors (ooh, experts...). The only hope is that Bob’s hint system is reliable because it uses encryption. But is that true? Could you recover Bob’s current passphrase from the photo of his workspace (Figure 2)?

![Figure 2: Workspace](image)
2.7.2 Solution

At the picture we see three stickers. One of them is “A Discourse of Fire and Salt” that represents a title of a book written by Blaise de Vigenère. This is the first hint that probably the Vigenère cipher was used. Then we have a sticker with the ciphertext AJKTUWLZABQYRSLS that consists of 19 letters. And finally, we see the sticker with arrows, where we can find 19 angles that probably correspond to the ciphertext letters.

There is a keyboard at the picture. So, we can guess that these arrows could be related to the letters from the keyboard. Let us look at the first two keyboard rows (Figure 3). We can recover the secret key ESWAQRTFOGYHJUKOLP. By deciphering the ciphertext using this key and the Vigenère cipher, we get WROTEFIRSTATTHEHEAD. Thus, Bob’s current passphrase is “Wrote first at the head”.

Surprisingly, nobody solved this problem in the first round, while five teams coped with it in the second round.

2.8 Problem “Bash-S3”

2.8.1 Formulation

The sponge function Bash-f uses the permutation $S_3$ that transforms a triple of 64-bit binary words $a, b, c$ in the following way:

$$S_3(a, b, c) = (b \lor \neg c \oplus a, a \lor c \oplus b, a \land b \oplus c).$$

Here $\neg$, $\land$, $\lor$, $\oplus$ denote the binary bitwise operations “NOT”, “AND”, “OR”, “XOR” respectively. The operations are listed in descending order of priority. Let $w^k$ also denote the cyclic shift of a 64-bit word $w$ to the left by $k \in \{1, 2, \ldots, 63\}$ positions.

Alice wants to strengthen $S_3$. She can add by XOR any input $a, b, c$ or its cyclic shift to any output. She must use at least one cyclic shift and she cannot add two identical terms to the same output.

Help Alice to change $S_3$ in such a way that a modified $S_3$ will be still a permutation!

Remarks. 1. For example, in the expression $b \lor \neg c \oplus a$, we firstly calculate $\neg c$, then calculate $b \lor \neg c$, and after that the final result (according to descending order of operations priority).

2. The modification

$$(b \lor \neg c \oplus a \oplus a^{11}, a \lor c \oplus a^7 \oplus c, a \land b \oplus b^{32})$$

is allowed but it does not satisfy the permutation condition.

2.8.2 Solution

It is allowed to add to the outputs of $S_3$ the outputs of the following linear transformation:

$$L(a, b, c) = (L_0(a, b, c), L_1(a, b, c), L_2(a, b, c))$$

that is defined by bitwise XOR operations and cyclic shifts.

The permutation property of a modified $S_3$ will be broken if for some distinct $(a, b, c)$ and $(a', b', c')$
Let \( S3(a, b, c) \oplus S3(a', b', c') = L(a, b, c) \oplus L(a', b', c') \). (1)

We will call the expressions from both sides of equality (1) and the sum \((a, b, c) \oplus (a', b', c')\) by differences. Let

\[
(w_0, w_1, w_2) = (a, b, c) \oplus (a', b', c'), \\
(W_0, W_1, W_2) = S3(a, b, c) \oplus S3(a', b', c').
\]

On the one hand, input and output differences of \(S3\) satisfy (for instance, see [3]) the equality

\[
w_0 \land W_0 \oplus w_1 \land W_1 \oplus w_2 \land W_2 = 11\ldots1.
\]

On the other hand, by (1) the permutation property of a modified \(S3\) will be broken if

\[
(W_0, W_1, W_2) = L(a, b, c) \oplus L(a', b', c') = L(w_0, w_1, w_2).
\]

As a result, a modified \(S3\) will be still a permutation if the following equality

\[
w_0 \land L_0(w_0, w_1, w_2) \oplus w_1 \land L_1(w_0, w_1, w_2) \oplus w_2 \land L_2(w_0, w_1, w_2) = 11\ldots1
\]

(2)
does not hold for any nonzero \((w_0, w_1, w_2)\). For example, if

\[
L(a, b, c) = (a \oplus a^d \oplus b, a \oplus c, b), \quad d \in \{1, 2, \ldots, 63\},
\]

then (2) becomes

\[
w_0 \land (w_0 \oplus w_0^d \oplus w_1) \oplus w_1 \land (w_0 \oplus w_2) \oplus w_2 \land (w_1) = w_0 \land (w_0 \oplus w_0^d) \neq 11\ldots1.
\]

Thus, we found the following solution for the problem:

\[
S3(a, b, c) \oplus L(a, b, c) = (b \lor c \oplus a^d \oplus b, a \lor c \oplus a \oplus b \oplus c, a \land b \oplus b \oplus c).
\]

Note that there are many other possible solutions.

This problem was completely solved by three participants in the first round and by nine teams in the second round. Many of these solutions were interesting and compact.

2.9 Problem “Metrical cryptosystem – 2”

2.9.1 Formulation

Let \( F_2^n \) be an \( n \)-dimensional vector space over the field \( F_2 = \{0, 1\} \). Alice and Bob exchange messages using the following cryptosystem.

1. First, they use a supercomputer to calculate two special large secret sets \( A, B \subseteq F_2^n \) which have the following property: there exists a constant \( \ell (\ell \geq 26) \), such that for any \( x \in F_2^n \) it holds

\[
d(x, A) + d(x, B) = \ell,
\]

where \( d(x, A) \) denotes Hamming distance from the vector \( x \) to the set \( A \).

2. Alice then saves the number \( \ell \), the set \( A \) and a set of vectors \( a_1, a_2, \ldots, a_r \) such that for any \( k : 0 \leq k \leq \ell \), there is a vector \( a_i \) at distance \( k \) from \( A \). Similarly, Bob saves the number \( \ell \), the set \( B \) and a set of vectors \( b_1, b_2, \ldots, b_s \) such that for any \( k : 0 \leq k \leq \ell \), there is a vector \( b_i \) at distance \( k \) from \( B \).
3. Text messages are encrypted letter by letter. In order to encrypt a letter Alice replaces it with its number in the alphabet, say \( k \). Then she chooses some vector \( a_i \) at distance \( k \) from the set \( A \) and sends this vector over to Bob. Bob then calculates the distance \( d(a_i, B) \) and using the property of the sets \( A, B \), calculates \( k = \ell - d(a_i, B) \). So, he gets the letter Alice sent. If Bob wants to send an encrypted message to Alice, he does the same but using his saved vectors and the set \( B \).

Eve was able to hack the supercomputer when it was calculating the sets \( A \) and \( B \). She extracted the set \( C \) from its memory, which consists of all vectors of \( \mathbb{F}_2^n \) that are at distance 1 or less from either \( A \) or \( B \). She also learned that \( \ell \) is even.

Help Eve to crack the presented cryptosystem (to decrypt any short intercepted message)! You know that she has an (illegal) access to the supercomputer, which can calculate and output the list of distances from all vectors of \( \mathbb{F}_2^n \) to any input set \( D \) in reasonable (but not negligible) time.

**Remarks.** Recall several definitions and notions. The *Hamming distance* \( d(x, y) \) between vectors \( x \) and \( y \) is the number of coordinates in which these vectors differ. Distance from vector \( y \in \mathbb{F}_2^n \) to the set \( X \subseteq \mathbb{F}_2^n \) is defined as \( d(y, X) = \min_{x \in X} d(y, x) \).

### 2.9.2 Solution

Let us denote by \( A_i \) (\( B_i \) respectively) the set of all vectors at distance \( i \) from the set \( A \) (\( B \) respectively):

\[
A_i = \{ x \in \mathbb{F}_2^n : d(x, A) = i \}, \quad B_i = \{ x \in \mathbb{F}_2^n : d(x, B) = i \}.
\]

It is easy to see that

- \( A = A_0 = B_\ell \)
- \( B = B_0 = A_\ell \)
- \( A_i = B_{\ell-i} \) for any \( i \in \{0, \ldots, \ell\} \)
- \( C = A_1 \cup B_1 \)

From the definition of the Hamming distance it is easy to prove that if a vector \( x \) lies in the set \( A_i \), then it is at distance \( |i - j| \) from the set \( A_j \) for any \( i, j \).

**Proof.** Indeed, if \( i = j \), the statement is trivial.

Assume that \( i > j \). By definition, \( d(x, A) = i \), so there exists a shortest path of length \( i \) from \( A \) to \( x \), consisting of vectors \( x_0, x_1, \ldots, x_i = x \), where \( x_0 \in A \). Since consecutive vectors in the path differ in only one coordinate, and vectors from \( A_s \) and \( A_t \) can be neighbours only if \( |s - t| \leq 1 \), it follows that \( x_k \in A_k \) for every \( k = 0, \ldots, i \). So, vector \( x_j \) from the path belongs to \( A_j \) and is at distance \( i - j \) from vector \( x \). Therefore, \( d(x, A_j) \leq i - j \). Distance cannot be less than \( i - j \), because then \( d(x, A) \) would have been less that \( i \), which contradicts conditions of the statement. Thus, \( d(x, A_j) = i - j \).

If \( i < j \), then we can replace \( A_i \) with \( B_{\ell-i} \), \( A_j \) with \( B_{\ell-j} \) and use \( B \) instead of \( A \) for the same argument as in the previous case.

In particular, given \( x \) is in \( A_i \), it is at distance \( |i - 1| \) from the set \( A_1 \) and at distance \( |i - (\ell - 1)| \) from the set \( B_1 \).

Let us “feed” the set \( C \) to the supercomputer. We denote the maximal distance from vectors of \( \mathbb{F}_2^n \) to vectors of \( C \) as \( r \), and the set of all vectors achieving this distance as \( \hat{C} \). Taking into account the statement proven above (and the fact that \( \ell \) is even), we can see that the maximum is achieved for vectors of the set \( A_\frac{\ell}{2} \). Hence, \( r = \frac{\ell}{2} - 1 \) and \( \hat{C} = A_\frac{\ell}{2} \). Thus, we can calculate \( \ell \) as \( 2r + 2 \).
Assume now that Alice sends a message $a_{i_1}, a_{i_2}, \ldots, a_{i_k}$ to Bob. Eve intercepts it and (using the obtained table of distances from the set $C$) calculates that these vectors are at distances $s_1, s_2, \ldots, s_k$ from the set $C$. Therefore, they are at distances $s_1 + 1, s_2 + 1, \ldots, s_k + 1$ from the set $A \cup B$. Since $d(x, A \cup B) = \min(d(x, A), d(x, B))$, each encrypted letter could be either $s_i + 1$ or $\ell - (s_i + 1)$. If one of these two numbers is greater than 26, we can easily determine the encrypted letter, if not, we can consider both possibilities. In the worst case we would need to consider $2^N$ variants, where $N$ is the length of the message, but since messages are short and are written in natural language, we do not need to check all of them and the decryption should not be hard.

Note: Sets $A$ and $B$ satisfying condition from Step 1 of the problem (for an arbitrary constant $\ell$ not necessarily greater than 26) are called strongly metrically regular and are studied in [9].

Best solutions to the problem were submitted by Alexey Chilikov (Bauman Moscow State Technical University, Russia) and Saveliy Skresanov (Novosibirsk State University, Russia).

### 2.10 Problem “A fixed element”

#### 2.10.1 Formulation

A polynomial $f(X_1, \ldots, X_n) \in \mathbb{F}_2[X_1, \ldots, X_n]$ is called reduced if the degree of each $X_i$ in $f$ is at most 1. For $0 \leq r \leq n$, the $r$th order Reed—Muller code of length $2^n$, denoted by $R(r, n)$, is the $\mathbb{F}_2$-space of all reduced polynomials in $X_1, \ldots, X_n$ of total degree less or equal than $r$. We also define $R(-1, n) = \{0\}$.

The general linear group $\text{GL}(n, \mathbb{F}_2)$ acts on $R(r, n)$ naturally: Given $A \in \text{GL}(n, \mathbb{F}_2)$ and $f(X_1, \ldots, X_n) \in R(r, n)$, $Af$ is defined to be the reduced polynomial obtained from $f((X_1, \ldots, X_n)A)$ by replacing each power $X_i^k$ ($k \geq 2$) with $X_i$. Consequently, $\text{GL}(n, \mathbb{F}_2)$ acts on the quotient space $R(r, n)/R(r - 1, n)$.

Let $A \in \text{GL}(n, \mathbb{F}_2)$ be such that its characteristic polynomial is a primitive irreducible polynomial over $\mathbb{F}_2$. Prove that the only element in $R(r, n)/R(r - 1, n)$, where $0 < r < n$, fixed by the action of $A$ is 0.

#### 2.10.2 Solution

Let $\begin{pmatrix} 1, \ldots, n \end{pmatrix}_r$ denote the set of $r$-subsets of $\{1, \ldots, n\}$. When $A$ acts on $R(r, n)/R(r - 1, n)$, its matrix with respect to the basis $\prod_{i \in I} X_i, I \in \begin{pmatrix} 1, \ldots, n \end{pmatrix}_r$, is the $r$th compound matrix $C_r(A)$ of $A$. The eigenvalues of $A$ are $\gamma^{2^i}, 0 \leq i \leq n - 1$, where $\gamma$ is a primitive element of $\mathbb{F}_{2^n}$. The eigenvalues of $C_r(A)$ are all possible products of $r$ eigenvalues of $A$, i.e.,

$$
\gamma^{\sum_{i \in I} 2^i}, \quad I \in \begin{pmatrix} 0, \ldots, n - 1 \end{pmatrix}_r.
$$

Clearly, the above expression never equals 1. Hence 1 is not an eigenvalue of $C_r(A)$. Therefore, the action of $A$ does not fix any nonzero element in $R(r, n)/(r - 1, n)$.

The problem was solved by four teams in the second round: Aleksei Udovenko (University of Luxembourg), Dianthe Bose and Neha Rino (Chennai Mathematical Institute, India), Andrey Kalachev, Danil Cherepanov, and Alexey Radaev (Bauman Moscow State Technical University, Russia), Sergey Titov and Kristina Geut (Ural State University of Railway Transport, Russia).
2.11 Problem “Hash function FNV-1a”

2.11.1 Formulation

Hash function **FNV-1a** [13] processes a message \( x \) composed of bytes \( x_1, x_2, \ldots, x_n \in \{0, 1, \ldots, 255\} \) in the following way:

1. \( h \leftarrow h_0; \)
2. for \( i = 1, 2, \ldots, n \): \( h \leftarrow (h \oplus x_i)g \mod 2^{128}; \)
3. return \( h \).

Here \( h_0 = 144066263297769815596495629667062367629, \) \( g = 2^{88} + 315. \) The expression \( h \oplus x_i \) means that the least significant byte of \( h \) is added bitwise modulo 2 with the byte \( x_i. \)

Find a collision, that is, two different messages \( x \) and \( x' \) such that **FNV-1a**(\( x \)) = **FNV-1a**(\( x' \)). Collisions on short messages and collisions that are obtained without intensive calculations are welcomed. Supply your answer as a pair of two hexadecimal strings which encode bytes of colliding messages.

2.11.2 Solution

We will base on the solution for the problem “FNV2” (NSUCRYPTO’2017) [8], where it was required to find a collision for the similar hash function FNV2. **FNV-1a** differs from FNV2 in the following: instead of the \( \oplus \) operation for adding \( h \) and \( x_i \) it uses standard \( + \) operation.

It is easy to see that

\[
\text{FNV2}(x_1 x_2 \ldots x_n) = (h_0 g^n + x_1 g^n + x_2 g^{n-1} + \ldots + x_n g) \mod 2^{128}.
\]

For FNV2, we found a relation

\[
a_1 g^{n-1} + a_2 g^{n-2} + \ldots + a_n \equiv 0 \pmod{2^{128}},
\]

where \( a_i \in \{-255, \ldots, 255\}. \)

Then we represented \( a_i \) as the difference \( x_i - x'_i \) and found a collision

\[
\text{FNV2}(x_1 x_2 \ldots x_n) - \text{FNV2}(x'_1 x'_2 \ldots x'_n) = a_1 g^n + a_2 g^{n-1} + \ldots + a_n g \equiv 0 \pmod{2^{128}}.
\]

Let us call a representation \( a_i = x_i - x'_i \) as a splitting of \( a_i. \) There can be several splittings for a given \( a_i. \) Each of them induces two trajectories of intermediate values of \( h: \) the trajectory starting with a message \( x_1 x_2 \ldots x_n \) and the trajectory starting with a message \( x'_1 x'_2 \ldots x'_n. \)

Let \( h_i \) and \( h'_i \) be the low bytes of \( h \) for the first and second trajectories respectively before the additions \( h + x_i \) and \( h + x'_i. \) Let us call a splitting suitable if

\[
h_i + x_i < 256, \quad h'_i + x'_i < 256, \quad i = 1, 2, \ldots, n.
\]

Let us evaluate the probability of existing a suitable splitting for \( a_i. \) We will assume that \( h_i, h'_i \) are realizations of independent random variables with uniform distribution over \( \{0, 1, \ldots, 255\}. \)

Bytes \( x_i \) and \( x'_i \) can take any value from intervals \( \{0, \ldots, 255 - h_i\} \) and \( \{0, \ldots, 255 - h'_i\} \) respectively. At the same time, the difference \( x_i - x'_i \) takes value from the interval \( \{-255 + h'_i, \ldots, 255 - h_i\}. \)

Then \( a_i \) is in the interval \( \{-255 + h'_i, \ldots, 255 - h_i\} \) with the probability

\[
\Pr(-255 + h'_i \leq a_i \leq 255 - h_i) = \begin{cases} \Pr(h_i \leq 255 - a_i), & a_i \geq 0, \\ \Pr(h'_i \leq 255 - |a_i|), & a_i < 0, \end{cases}
\]
that is equal to \( 1 - |a_i|/256 \).

Thus, the probability of existing a suitable splitting for the whole sequence \( a_1a_2 \ldots a_n \) is the following:

\[
p = \prod_{i=1}^{n} \left( 1 - \frac{|a_i|}{256} \right).
\]

This probability can be rather high. For example, \( p \approx 1/25 \) for the following sequence for \( n = 18 \):

\((-64, 5, 73, 35, -53, 19, -10, -78, -44, 48, 61, -1, -80, 26, -22, 72, -31, 0)\).

Or, \( p \approx 1/13 \) for the following sequence for \( n = 19 \):

\((-37, 34, -74, -4, -17, 33, -18, 21, 54, 33, -1, 58, -71, -13, -10, 11, -88, -19, 0)\).

Moreover, the probability can be increased if we change a strategy of finding suitable splittings. We can allow to modify splittings \( a_1, \ldots, a_{i-1} \) that have been already built if it is impossible to find a splitting for \( a_i \).

After finding a suitable splitting, we determine the sequences \( (h_i), (h_i') \). Then we determine the bytes \( \tilde{x}_i, \tilde{x}'_i \) such that

\[
h_i \oplus \tilde{x}_i = h_i + x_i, \quad h_i' \oplus \tilde{x}'_i = h_i' + x'_i, \quad i = 1, 2, \ldots, n.
\]

It is important that there are no carries in high bytes in additions \( h_i + x_i, h_i' + x'_i \); and \( \tilde{x}_i, \tilde{x}'_i \) can be always found. Then a collision for FNV-1a is a pair of messages \( \tilde{x}_1 \tilde{x}_2 \ldots \tilde{x}_n \) and \( \tilde{x}'_1 \tilde{x}'_2 \ldots \tilde{x}'_n \).

It remains to say that the sequence \( (a_i) \) can be found using LLL algorithm. The algorithm is applied to the lattice defined by the basic vectors

\[
b_1 = (1, 0, \ldots, 0, g^{n-1} \mod 2^{128}),
\]
\[
b_2 = (0, 1, \ldots, 0, g^{n-2} \mod 2^{128}),
\]
\[
\quad \quad \vdots
\]
\[
b_n = (0, 0, \ldots, 1, g^{0} \mod 2^{128}),
\]
\[
b_{n+1} = (0, 0, \ldots, 0, t2^{128}),
\]

where \( t \) is a small integer. LLL finds a short basis of the lattice, i.e. vectors

\[
v = \sum_{i=1}^{n+1} a_i b_i
\]

with small coordinate values. Let the last coordinate \( v \) equal to 0. Then

\[
\sum_{i=1}^{n+1} a_i g^{n-i} \equiv 0 \pmod{2^{128}},
\]

i.e. \( (a_1, \ldots, a_n) \) is a required solution.

This problem was completely solved by fourteen teams (the most of them used a reduction to the problem FNV2). Some examples of collisions proposed by participants (in HEX format) are given in Table 4.
Table 4: Collisions of FNV-1a.

| message 1                  | message 2                  |
|---------------------------|---------------------------|
| f1dd5921afdf29cb3d3b357184e8c | 928ea41b7373792aae2bfa72ca64 |
| eb1811b160a95e0511357e158b58  | ab775b3a7c7c7c7c3a5dc94e   |
| f828e4070672220b195e0dd2114a4c008 | 3638fa655d1b61e21419134803222bbb35 |
| 3a7a3a7a3a4a5a5a5a5a5a5a5a5a5a5a5a5a | 51089c5e7fe7cc2d740b5f70b3cb5461824d |
| f4331cede51639057d05f80ff1d638b40b286f | eb270505187332116c611402081f1155326013 |
| 07160c2e0b700b1338ef6e63360419060507 | 10610bf23b0573e21419134803222bbb35 |
| 00ca0000cb00000000000000029092100d814 | 2d000158001b773a6364fc0905000000e90000 |

2.12 Problem "TwinPeaks2"

2.12.1 Formulation

Bob realized that his last year cipher TwinPeaks (NSUCRYPTO’2017) [8] is not secure enough and modified it. He considerably increased the number of rounds and made rounds more complicated. New Bob’s cipher works as follows.

A message \( X \) is represented as a binary word of length 128. It is divided into four 32-bit words \( a, b, c, d \) and then the following round transformation is applied 48 times:

\[
(a, b, c, d) \leftarrow (b, c, d, a \oplus S_3(S_1(b) \oplus S_2(b \land \neg c \oplus c \lor d) \oplus S_1(d))),
\]

Here \( S_1, S_2, S_3 \) are secret permutations over 32-bit words; \( \neg, \land, \lor, \oplus \) are binary bitwise “NOT’, “OR”, “AND”, “XOR” respectively (the operations are listed in descending order of priority). The concatenation of the final \( a, b, c, d \) is the resulting ciphertext \( Y \) for the message \( X \).

Agent Cooper again wants to read Bob’s messages! He caught the ciphertext \( Y = \text{DEB239852F1B47B005FB390120314478} \) and captured also Bob’s smartphone with the TwinPeaks2 implementation! Here it is [15]. Now Cooper (and you too) can encrypt any messages with TwinPeaks2 but still can not decrypt any one. Help Cooper to decrypt \( Y \).

Remarks. The ciphertext is given in hexadecimal notation, the first byte is \( \text{DE} \).

2.12.2 Solution

Let \( F \) be the round transformation of TwinPeaks2:

\[
F(a, b, c, d) = (b, c, d, a \oplus f(b, c, d)).
\]

The encryption transformation is the composition of 48 copies of \( F \), i.e. it can be written as \( F^{48} \). Consequently, \( F^{-48} \) is the decryption transformation. Let

\[
\tau(a, b, c, d) = (d, c, b, a).
\]

Let us note that \( f(b, c, d) = f(d, c, b) \). Then the composition of \( F, \tau \) and \( F \) gives us \( \tau \):

\[
F \circ \tau \circ F(a, b, c, d) = F(a \oplus f(b, c, d), d, c, b) = F(a \oplus f(d, c, b), d, c, b) = (d, c, b, a).
\]

Hence,

\[
F^{48} \tau F^{48} = \tau
\]
Thus, in order to decrypt $Y$ one should write its 32-bit blocks in reverse order, encrypt the result and then reverse the order of the blocks again. The result will be a hexadecimal word, which gives us the desired message

```
attacksgetbetter.
```

The best solution to the problem has been submitted by Carl Löndahl (Sweden), which not only provides a clean theoretical solution, but also proposes a slide attack on the cipher.

## 2.13 Problem “An Enigmatic Challenge”

### 2.13.1 Formulation

The Enigma machine is a symmetric cipher famous for being during the Second World War by the German military. Its internal structure comprises an 26-letter Latin alphabetic permutation, implemented as rotors. The machine used for this problem consists of 3 rotors and a reflector.

Figure 4 shows how a simplified Enigma machine works. The key components are the set of input switches (2) – which are reduced to 4 in the example but could have been 26 for the Latin alphabet – an input plugboard (3,7,8), three rotors (5), the reflector (6) and the output board (9).

The components have the following functionality:

- **Rotors**: a rotor (5) is a wheel with the upper-case alphabet in order on the rim and a hole for an axle. On both sides of a rotor are 26 electrical contacts each under a letter. Each contact on one side is wired to a contact on the other side at a different position. The rotor implements an one-to-one and onto function between the upper-case letters, where each letter is mapped to a different one (an irreflexive permutation).

- **Reflector**: the reflector (6) is positioned after the rotors and has contacts for each letter of the alphabet on one side only. The letters are wired up in pairs, so that an input current on a letter is reflected back to a different letter.

**The input message**: is permuted by the rotors, passes through the reflector and then goes back through the rotors in the reverse order (as depicted in the figure). Finally, the light bulb indicates the encrypted letter. The plugboard plays no role in permuting the letter for this challenge, although it could have.

To prevent simple frequency analysis attack the right rotor rotates with every new input. After the right rotor completed a full rotation (after 26 letters were encrypted), the middle rotor rotates once. Similarly, after the middle rotor completes a full rotation (and the right rotor complete 676 rotations), the left rotor rotates once.

**Challenge**: you will play the role of an attacker that knows the source of the plaintext to be encrypted. You are given a ciphertext corresponding to a plaintext taken from this known source which happens to be “Moby Dick” by Herman Melville, and you are asked to recover the plaintext. The plaintext consists only of trimmed capital letters with no punctuation marks and spaces and is contiguous. All letters are from the Latin alphabet. Extra information on the settings of the rotors.
is provided: the configuration of the first rotor is very close to the one used in the 1930 commercial version: EKMFGLDQVZNTOWYHXUSPAIBRCJ.

Ciphertext:

RHS MZHXX AOWW ZTOW QQMB CRZA BARN MLAV MLSX SPBA ZTHG
YLGE VGZG KULJ FLOZ RQAW YGAA DCJB YWBB IYQQ FAAO RAGK
BGSW QARG EYSP IKYE LLUO YCNH HDBV AFKD HETA ONNR HXHE
BBRT R0ZD XJCC OMXR PNSW UAZB TNJY BANH FGCS GJWW YTBV
VGLX KUZW PARO NMXP LDLZ ICBEK XVSJ NNCF SOTA AQYS Y2FX
MZDH MSZI BAAH RFXT FTPU VWMC PEXQ NZVA LMFX BKHG QGYS
BIYE MEUE PJNR AVTL JSUZ PLHQ MOUI IQFD HVXI NOOJ YJAF
WAVU PVQA FMKP AHLK XJYD GITB QSPK CUZU XPRK MUJJ YRJ

Link to “Moby Dick” text file can be found in [14].

2.13.2 Solution

It is easy to observe that the Left and Middle Rotors will not change for each block of 26 characters of the plaintext. From this point of view, we can regard the composition of permutations induced by the Left and Middle rotors (as well as the reflector) as one, fixed permutation. After the next 26 letters are processed, the Middle rotor turns, and a distinct permutation is to be used for the incoming block of 26 letters. Due to the fact the challenge ciphertext is less than 676 characters, we do not bother with turning the Left rotor.

To fix some notations, let \( \rho_i, L_i, M_i \) denote permutations defined on the set \( \{A,\ldots,Z\} \), with \( i \) denoting each block of 26 letters to be processed. That is \( i \in \{1,\ldots,\lceil |C|/26 \rceil \} \), where \( |C| \) denotes the length of the ciphertext (its number of characters). We also write \( \pi_i = M_i^{-1} \circ L_i^{-1} \circ \rho \circ L_i \circ M_i, \quad i \in \{1,\ldots,\lceil |C|/26 \rceil \} \).

The next step is to split the challenge ciphertext into blocks of 26 characters, and use the fact that for each block \( i \), \( \pi_i \) acts as an oracle that returns the same value for the same input. We will correlate this with the information that is \emph{apriori} given on the first rotor. Although we do not have its exact configuration, we use the fact that it is close to some known rotor that (in fact, the real configuration permutes 4 elements and then applies a circular permutation of length four).

The permutation corresponding to the given Right rotor of the commercial Enigma (1930) is the following:

\[
\begin{array}{cccccccccccccccccccc}
A & B & C & D & E & F & G & H & I & J & K & L & M & N & O & P & Q & R & S & T & U & V & W & X & Y & Z \\
E & K & M & F & L & G & D & Q & V & Z & N & T & O & W & Y & H & X & U & S & P & A & I & B & R & C & J \\
\end{array}
\]

Then we take the first block of 26 letters and obtain their inverses, minding the fact that the rotor shifts with one place to the left after we read one letter. Hence, for the first block we obtain:

RHS MZHXX AOWW ZTOW QQMB CRZA BA
UVAH FOFL VRDQ TDNG DQLA BOIR JJ

Now we remark on a “distance-preserving” property: if the distance between identical characters returned by \( \pi_i \) (the input to the Right rotors) is \( \ell \), then it maintained in the original plaintext. As an example, the group ZHXX in the first block of ciphertext has been obtained for the group
FOFL and we note a distance of 2 ($F \rightarrow O \rightarrow F$) between $F$ and $F$. This means that an alphabetical distance of 2 exists between the corresponding letters of the plaintext. More precisely, if:
\[
\pi_1(R(x)) = \pi_1(R'(y)),
\]
where $R'$ is obtained by shifting $R$ with $\ell$ elements, then the character $y$ is at a distance of $\ell$ from the character $x$ (but in the opposite sense). Based on this observation, the solution is to identify such pairs inside a block and record the distance between them. As 4 elements are permuted in the real configuration of the rotor, false positives will appear.

After the colliding characters per block, say in position $i$ and $j$, have been identified and their distance recorded, say the distance is $\ell$, one will simply write a script that will pass through the given plaintext (after removing the non-alphabetic characters) and identify the sequence (matching the length of the ciphertext) where the distance between the characters in position $i$ and $j$ is $\ell$.

Finally, the plaintext that is to be recovered is:

\[
\text{ALREADY we are boldly launched upon the deep ; but soon we shall be lost in its unshored, harbourless immensities. Ere that come to pass ; ere the Pequod’a weedy hull rolls side by side with the barnacled hulls of the leviathan; at the outset it is but well to attend to a matter almost indispensable to a thorough appreciative understanding of the more special leviathanic revelations and allusions of all sorts which are to follow.}
\]

Finally, eight teams completely solved the problem. Note, that there was a simple way how to almost find a solution, which was used by many teams. It is based on the fact that no letter from the plaintext gets mapped to the same letter in the ciphertext using Enigma. But this approach gives two possible solutions and does not allow to prove that one of them is not correct.

2.14 Problem “Orthogonal arrays” (unsolved)

2.14.1 Formulation

Orthogonal arrays are closely connected with cryptographic Boolean functions. Namely, supports of correlation immune functions give orthogonal arrays when their elements are written as the rows of an array.

Given three positive integers $n$, $t$ and $\lambda$ such that $t < n$, we call $t-(2,n,\lambda)$ orthogonal array every $\lambda 2^t \times n$ binary array (i.e. matrix over the 2-element field) such that, in every subset of $t$ columns of the array, every (binary) $t$-tuple appears in exactly $\lambda$ rows. $t$ is called the strength of this orthogonal array.

Find a $4-(2,11,\lambda)$ orthogonal array with minimal value of $\lambda$.

2.14.2 Solution

The best known answer to this question is $\lambda = 8$ [10], but it is unknown whether there exists a $4-(2,11,\lambda)$ orthogonal array for $\lambda < 8$. This open problem remains unsolved. There were suggested several ideas by the participants.

The most interesting one was proposed by Aleksei Udovenko (University of Luxembourg). His study starts with the Nordstrom — Robinson code (that is, the Kerdock code of length 16 and size 256, whose dual distance is the minimum distance of the Preparata code, that is 6, which gives
a strength of the orthogonal array (OA) equal to 5). Only the codewords with the first element equal to zero are kept and their coordinate at 0 is deleted, which makes size 128, length 15 and strength 4. Then three columns are erased from the OA, which does not reduce the strength, and the resulting OA provides a solution to the problem with $\lambda = 8$. It is then shown (by using known results) that, for any solution to the problem, $\lambda$ is a least 6. This is interesting. The solution found is written in the form $(x, F(x))$ where $F$ is a quadratic $(7, 4)$-function. Its determination allows to determine the 4-th order correlation immune function whose support is this OA. This is a 11-variable Boolean function of algebraic degree 5. Then the annihilators of this function are studied. It is shown that the function has a linear annihilator (and has then algebraic immunity 1). After an observation on the impossibility of extending a solution which would have $\lambda \leq 7$, the Xiao — Massey characterization of OA is proved again in different terms. It is also shown that any affine annihilator of a $t$-th order correlation immune function must be $t$-resilient which is a nice observation. A computer search is made with Integer Linear Programming showing that any 4-th order correlation immune function having an affine annihilator should have weight at least 128, which is a nice observation. This nice work concludes with open questions.

Another one good solution was given by the team of Evgeniya Ishchukova, Vyacheslav Salmanov, and Oksana Shamiliyan (Southern Federal University, Russia). They firstly studied the maximum value of $n$ given $t$ for small values of $t$. Then an algorithm was designed which reduces the search to solutions presenting some symmetries observed in smaller values of $t$ and $n$. Finally, a solution was given with $\lambda = 8$ which is the coset of a linear code of length $n = 11$ and dimension $k = 7$ (and therefore 128 codewords), with dual distance 5 and the corresponding function is then indeed 4-th order correlation immune giving a $4 - (2, 11, \lambda)$ orthogonal array. Unfortunately, the question whether 128 is minimal was not addressed.

2.15 Problem “Sylvester matrices” (unsolved)

2.15.1 Formulation

Sylvester matrices play a role in security since they are connected with topics like secret sharing and MDS codes constructed with cellular automata.

Consider two univariate polynomials over a field, $P_1(x)$ of degree $n$ and $P_2(x)$ of degree $m$, where $P_1(x) = a_m x^m + \ldots + a_0$ and $P_2(x) = b_n x^n + \ldots + b_0$. The *Sylvester matrix* is an $(m + n) \times (m + n)$ matrix formed by filling the matrix beginning with the upper left corner with the coefficients of $P_1(x)$, then shifting down one row and one column to the right and filling in the coefficients starting there until they hit the right side. The process is then repeated for the coefficients of $P_2(x)$. All the other positions are filled with zero.

Prove whether there exist invertible Sylvester matrices whose inverse are Sylvester matrices as well.

**Example.** For $m = 4$ and $n = 3$, the Sylvester matrix is the following:

$$
\begin{pmatrix}
  a_4 & a_3 & a_2 & a_1 & a_0 & 0 & 0 \\
  0 & a_4 & a_3 & a_2 & a_1 & a_0 & 0 \\
  0 & 0 & a_4 & a_3 & a_2 & a_1 & a_0 \\
  b_3 & b_2 & b_1 & b_0 & 0 & 0 & 0 \\
  0 & b_3 & b_2 & b_1 & b_0 & 0 & 0 \\
  0 & 0 & b_3 & b_2 & b_1 & b_0 & 0 \\
  0 & 0 & 0 & b_3 & b_2 & b_1 & b_0 
\end{pmatrix}
$$
2.15.2 Solution

We are pleased to say that three teams completely solved this problem! They are Alexey Chilikov (Bauman Moscow State Technical University, Russia), Radu Caragea, Madalina Bolboceanu, and Miruna Rosca (Bitdefender, Romania), Samuel Tang and Harry Lee (Hong Kong). Here we present the main idea for the solution.

Let $m \leq n$. Let $P_1(x) = x^m$ and $P_2(x) = x^n + 1$. Then their Sylvester matrix is the following:

$$
\begin{pmatrix}
I_n & 0_{n \times m} \\
0_{m \times (n-m)} & I_m
\end{pmatrix},
$$

where $I_k$ denotes the $k \times k$ identity matrix; and $0_{k \times \ell}$ is the $k \times \ell$ zero matrix. Taking all operations over the the 2-element field, it is clear that

$$
\begin{pmatrix}
I_n & 0_{n \times m} \\
0_{m \times (n-m)} & I_m
\end{pmatrix} \cdot \begin{pmatrix}
I_n & 0_{n \times m} \\
0_{m \times (n-m)} & I_m
\end{pmatrix} = \begin{pmatrix}
I_n & 0_{n \times m} \\
0_{m \times n} & I_m
\end{pmatrix} = I_{m+n}.
$$

Thus, the considered Sylvester matrix is an involutory matrix. Therefore, its inverse is the Sylvester matrix as well.

Let $m > n$. Assume that the inverse of the Sylvester matrix of $P_1(x)$ and $P_2(x)$ is also the Sylvester matrix for two polynomials over the 2-element field, say $Q_1(x) = c_p x^p + c_{p-1} x^{p-1} + \cdots + c_0$, $Q_2(x) = d_q x^q + d_{q-1} x^{q-1} + \cdots + d_0$, of degrees $p > 0$ and $q > 0$ respectively, which satisfy $p + q = m + n$. The product of Sylvester matrices which correspond to $P_1(x), P_2(x)$ and $Q_1(x), Q_2(x)$ is equal to $I_{m+n}$, in particular

$$
\begin{pmatrix}
a_m & a_{m-1} & \cdots & a_0 \\
a_m & a_{m-1} & \cdots & a_0 \\
\vdots & \ddots & \ddots & \vdots \\
b_n & b_{n-1} & \cdots & b_0 \\
b_n & b_{n-1} & \cdots & b_0 \\
\vdots & \ddots & \ddots & \vdots \\
b_n & b_{n-1} & \cdots & b_0
\end{pmatrix} = \begin{pmatrix}
c_p \\
0 \\
\vdots \\
d_q \\
0 \\
\vdots \\
0
\end{pmatrix} \in \mathbb{F}_{2}^{m+n}.
$$

The condition $q > n$ implies $b_n c_p = 0$, but $b_n = c_p = 1$ since the polynomials $P_2(x)$ and $Q_1(x)$ have degrees $n$ and $p$ respectively. Therefore, it must hold $q \leq n$. Since $Q_2(x)$ has degree $q$, then $d_q = 1$ and $(1 + b_{n-q}) = b_{n-q+1} = b_{n-q+2} = \cdots = b_{n-q + \min\{q,m-1\}} = 0$. From $b_n = 1$ it follows that $\min\{q,m-1\} < q$, that is $m \leq q$. Finally, we get $m \leq q \leq n$ that is a contradiction.

Thus, in the case $m \leq n$ there exist invertible Sylvester matrices whose inverse are Sylvester matrices as well but for $m > n$ it does not hold.

2.16 Problem “Disjunct Matrices” (unsolved)

2.16.1 Formulation

Disjunct Matrices are used in some key distribution protocols for traitor tracing. Disjunct Matrices (DM) are a particular kind of binary matrices which have been especially applied to solve the Non-Adaptive Group Testing (NAGT) problem, where the task is to detect any configuration of $t$ defectives out of a population of $N$ items. Traditionally, the methods used to construct DM leverage on error-correcting codes and other related algebraic techniques.
Let $A = (x_1^\top, x_2^\top, \ldots, x_N^\top)$ be an $M \times N$ binary matrix. Then, $A$ is called $t$-disjunct if, for all subsets of $t$ columns $S = \{x_{i_1}, \ldots, x_{i_t}\}$, and for all remaining columns $x_j \notin S$, it holds that

$$\text{supp}(x_j) \nsubseteq \bigcup_{k=1}^{t} \text{supp}(x_{i_k}),$$

where $\text{supp}(x)$ denotes the set of coordinate positions of a binary vector $x$ with 1s.

In other words, a matrix $A$ is $t$-disjunct if for every subset $S$ of $t$ columns the support of any other column is not contained in the union of the supports of the columns in $S$.

Prove what is the minimum number of rows in a 5-disjunct matrix.

### 2.16.2 Solution

We find ourselves obliged to inform that the formulation of the problem did not include the condition which make this problem non-trivial. The condition is that the number of columns must be greater than the number of rows. This formulation comprises practical significance and has following equivalent form: given $t$, when does there exist a $t$-disjunct algorithm better than the trivial one that tests each item individually? Please find details regarding Non-Adaptive Group Testing (NAGT) problem together with known results and mentioned formulations in [12].

The solution of the initial problem is 6, an example is $6 \times 6$ identity matrix, it was discovered by several participants. However some participants (Alexey Chilikov from Bauman Moscow State Technical University, Aleksei Udovenko from University of Luxembourg, Henning Seidler and Katja Stumpp from Technical University of Berlin) obtained bounds for the number of rows depending the parameter $t$ and the number of columns.

### 3 Winners of the Olympiad

Here we list information about the winners of NSUCRYPTO’2018 in Tables 5, 6, 7, 8, 9, 10.

![Winners of NSUCRYPTO from 2014 to 2018](image)

Figure 5: Winners of NSUCRYPTO from 2014 to 2018
Table 5: **Winners of the first round in school section A ("School Student")**

| Place | Name               | Country, City   | School                                      | Scores |
|-------|--------------------|-----------------|---------------------------------------------|--------|
| 1     | Borislav Kirilov   | Bulgaria, Sofia | The First Private Mathematical Gymnasium    | 22     |
| 1     | Alexey Lvov        | Russia, Novosibirsk | Gymnasium 6                      | 21     |
| 1     | Razvan Andrei Draghici | Romania, Craiova | National College Fratii Buzesti           | 20     |
| 2     | Gorazd Dimitrov    | Macedonia, Skopje | Yahya Kemal College                  | 18     |
| 3     | Ivan Baksheev      | Russia, Novosibirsk | Gymnasium 6                      | 16     |
| 3     | Artem Ismagilov    | Russia, Yekaterinburg | The Specialized Educational and Scientific Center UrFU | 16 |
| 3     | Bogdan Cireanu      | Romania, Craiova | National College Fratii Buzesti           | 16     |
| 3     | Ruxandra Iliceanu   | Romania, Craiova | Tudor Vianu National College of Computer Science | 15 |
| Diploma| Sofya Gorbunova    | Russia, Yekaterinburg | The Specialized Educational and Scientific Center UrFU | 13 |
| Diploma| Kirill Poltoradnev | Russia, Yekaterinburg | The Specialized Educational and Scientific Center UrFU | 13 |
| Diploma| Tudor Moga         | Romania, Brasov | Grigore Moisil National College of Computer Science | 11     |
| Diploma| Markas Cerniauskas | Lithuania, Kaunas | Kaunas Technology University Gymnasium | 11     |
| Diploma| Mircea-Costin Preoteasa | Romania, Bucharest | Tudor Vianu National College of Computer Science | 11 |
| Diploma| Kirill Tugulukov    | Russia, Ulan-Ude | School 19, Olympiad training center ENTER | 11     |

Table 6: **Winners of the first round, section B (in the category "University Student")**

| Place | Name               | Country, City   | University                                       | Scores |
|-------|--------------------|-----------------|--------------------------------------------------|--------|
| 1     | Maxim Plushkin     | Russia, Moscow  | Lomonosov Moscow State University                | 25     |
| 2     | Robert Koprinkov   | Netherlands, Nijmegen | Radboud University                | 20     |
| 2     | Irina Slonkina     | Russia, Moscow  | National Research Nuclear University MEPhI       | 18     |
| 2     | Marc Houben        | Belgium, Leuven | KU Leuven                                        | 18     |
| 3     | Dheeraj M Pai      | India, Chennai  | Indian Institute of Technology Madras            | 17     |
| 3     | Alexander Grebennikov | Russia, Saint Petersburg | Saint Petersburg State University | 16     |
| 3     | Roman Lebedev      | Russia, Novosibirsk | Novosibirsk State University               | 16     |
| 3     | Dianthe Bose       | India, Chennai  | Chennai Mathematical Institute                   | 15     |
| 3     | Ivan Sutormin      | Russia, Novosibirsk | Novosibirsk State University               | 15     |
| Diploma| Harikumar Krishnamurthy | India, Chennai    | Indian Institute of Technology Madras            | 14     |
| Diploma| Roman Tarasov      | Russia, Odintsovo | Higher School of Economics                      | 13     |
| Diploma| Sander Suverkropp  | Netherlands, Wageningen | Radboud University                | 13     |
| Diploma| Saeed Odak         | Iran, Tehran     | Khajeh Nasir Toosi University of Technology     | 13     |
| Diploma| Thijs van Loenhout | Netherlands, Nijmegen | Radboud University                | 12     |
| Diploma| Daniil Gurev       | Russia, Novosibirsk | Novosibirsk State University               | 12     |
| Diploma| Neha Rino          | India, Chennai  | Chennai Mathematical Institute                   | 12     |
| Diploma| Kristina Volyakova  | Russia, Yaroslavl | Yaroslavl State University                      | 12     |

Table 7: **Winners of the first round, section B (in the category "Professional")**

| Place | Name               | Country, City           | Organization                                        | Scores |
|-------|--------------------|-------------------------|-----------------------------------------------------|--------|
| 1     | Alexey Udovenko    | Luxembourg, Luxembourg | University of Luxembourg                           | 23     |
| 3     | Henning Seidler    | Germany, Berlin         | TU Berlin                                           | 16     |
| Diploma| Alexey Chilikov    | Russia, Moscow          | Bauman Moscow State Technical University            | 14     |
| Diploma| Samuel Tang        | Hong Kong, Hong Kong    | Blocksquare Limited                                  | 12     |
| Diploma| Amedeo Sgueglia    | United Kingdom, London   | London School of Economics and Political Science    | 12     |
| Diploma| Samad Alaamati     | Iran, Tehran            | American Society for Industrial Security             | 12     |
Table 8: Winners of the second round (in the category “School Student”)

| Place | Names                | Country, City    | School                  | Scores |
|-------|----------------------|------------------|-------------------------|--------|
| Diploma | Brian Ncube         | Zimbabwe, Hwange | Hwange High School      | 7      |

Table 9: Winners of the second round (in the category “University student”)

| Place | Name                          | Country, City                  | University                                      | Scores |
|-------|-------------------------------|--------------------------------|-------------------------------------------------|--------|
| 1     | Maxim Plushkin                | Russia, Moscow                 | Lomonosov Moscow State University               | 43     |
| 2     | Irina Slonkina                | Russia, Moscow                 | National Research Nuclear University MEPhI       | 38     |
| 2     | Dmitry Lavrenov, Egor Lavrenov, Uladzimir Paprotski | Belarus, Minsk | Belarusian State University                     | 37     |
| 3     | Thanh Nguyen Van, Tuong Nguyen Van, Dinh Ton | Vietnam, Ho Chi Minh City | Ho Chi Minh City University of Technology, University of Science | 37     |
| 3     | Ngoc Ky Nguyen, Phuoc Nguyen Ho Minh, Danh Nam Tran | Vietnam, Ho Chi Minh City; France, Paris | Ho Chi Minh City Pedagogical University, Ho Chi Minh City University of Technology, Ecole Normale Superieure | 34     |
| 3     | Diantthe Bose, Neha Rino      | India, Chennai                 | Chennai Mathematical Institute                  | 33     |
| 3     | Roman Lebedev, Vladimir Sitnov, Alexander Tkachev | Russia, Novosibirsk | Novosibirsk State University                     | 30     |
| 3     | Mikhail Sorokin, Darya Frolova, Vladimir Bobrov | Russia, Moscow | National Research Nuclear University MEPhI       | 29     |
| 3     | Andrey Kalachev, Danil Cherepanov, Alexey Radaev | Russia, Moscow | Bauman Moscow State Technical University         | 27     |
| Diploma | Saveliy Skresanov            | Russia, Novosibirsk            | Novosibirsk State University                     | 21     |
| Diploma | Harikumar Krishnamurthy, Aditya Pradeep, Dheeraj M Pai | India, Chennai | Indian Institute of Technology Madras           | 21     |

Table 10: Winners of the second round (in the category “Professional”)

| Place | Names                          | Country, City                  | Organization                                                   | Scores |
|-------|--------------------------------|--------------------------------|----------------------------------------------------------------|--------|
| 1     | Alexey Udovenko                | Luxembourg, Luxembourg         | University of Luxembourg                                       | 72     |
| 2     | Evgeniya Ishchukova, Vyacheslav Salmanov, Oksana Shamilyan | Russia, Taganrog | Southern Federal University                                     | 46     |
| 3     | Henning Seidler, Katja Stumpp  | Germany, Berlin                | Berlin Technical University                                     | 40     |
| 3     | Carl Londahl                   | Sweden, Karlskrona             | TrueSec AB                                                    | 38     |
| 3     | Alexey Chilikov                | Russia, Moscow                 | Bauman Moscow State Technical University                      | 38     |
| Diploma | Duc Tri Nguyen, Quan Doan, Quoc Bao Nguyen | Vietnam, Ho Chi Minh city | CERG at George Mason University, E-CURITY, Ho Chi Minh City University of Technology | 37     |
| Diploma | Radu Caragea, Miruna Rosca, Madalina Bolboceanu | Romania, Bucharest | Bitdefender                                                  | 32     |
| Diploma | Mikhail Polyakov, Mikhail Tsvetkov, Victoria Vlasova | Russia, Moscow | Bauman Moscow State Technical University                      | 30     |
| Diploma | Harry Lee, Samuel Tang         | Hong Kong, Hong Kong           | Blocksquare Limited, Hong Kong University of Science and Technology | 29     |
| Diploma | Lars Haulin                    | Sweden, Uppsala                | –                                                             | 28     |
| Diploma | Sergey Titov, Kristina Geut    | Russia, Yekaterinburg           | Ural State University of Railway Transport                  | 22     |
| Diploma | Khai Hanh Tang, Neng Zeng, Thu Hien Chu Thi | Singapore, Singapore | Ho Chi Minh City Pedagogical University, Nanyang Technological University | 21     |
References

[1] Agievich S., Gorodilova A., Idrisova V., Kolomeec N., Shushuev G., Tokareva N. Mathematical problems of the second international student’s Olympiad in cryptography // Cryptologia. 2017. V. 41. No. 6. P. 534–565.

[2] Agievich S., Gorodilova A., Kolomeec N., Nikova S., Preneel B., Rijmen V., Shushuev G., Tokareva N., Vitkup V. Problems, solutions and experience of the first international student’s Olympiad in cryptography // Prikladnaya Diskretnaya Matematika (Applied Discrete Mathematics). 2015. No. 3. P. 41–62.

[3] Agievich S., Marchuk V., Maslau A., Semenov V. Bash-f: another LRX sponge function // Cryptology ePrint Archive: Report 2016/587.

[4] Bhasin S., Carlet C., Guilley S. Theory of masking with codewords in hardware: low-weight dth-order correlation-immune Boolean functions // Cryptology ePrint Archive, Report 2013/303.

[5] Bierbrauer J. Bounds on orthogonal arrays and resilient functions // Journal of Combinatorial Designs. 1995. V. 3. P. 179–183.

[6] Friedman J. On the bit extraction problem // Proc. 33rd IEEE Symposium on Foundations of Computer Science. 1992. P. 314–319.

[7] Geut K., Kirienko K., Sadkov P., Taskin R., Titov S. On explicit constructions for solving the problem “A secret sharing” // Prikladnaya Diskretnaya Matematika. Prilozhenie. 2017. No. 10. P. 68–70. (in Russian)

[8] Gorodilova A., Agievich S., Carlet C., Gorkunov E., Idrisova V., Kolomeec N., Kutsenko A., Nikova S., Oblaukhov A., Picek S., Preneel B., Rijmen V., Tokareva N. Problems and solutions of the Fourth International Students Olympiad in Cryptography (NSUCRYPTO) // Cryptologia. 2019, Vol. 43, No. 2, pp. 138–174.

[9] Oblaukhov A. A lower bound on the size of the largest metrically regular subset of the Boolean cube // Cryptography and Communications. 2018. Published online.

[10] Picek S., Guilley S., Carlet C., Jakobovic D., Miller J.F. Evolutionary Approach for Finding Correlation Immune Boolean Functions of Order t with Minimal Hamming Weight. In: Dediu AH., Magdalena L., Martín-Vide C. (eds) Theory and Practice of Natural Computing. Lecture Notes in Computer Science, vol 9477. Springer, Cham.

[11] Tokareva N., Gorodilova A., Agievich S., Idrisova V., Kolomeec N., Kutsenko A., Oblaukhov A., Shushuev G. Mathematical methods in solutions of the problems from the Third International Students’ Olympiad in Cryptography // Prikladnaya Diskretnaya Matematika (Applied Discrete Mathematics). 2018. No. 40. P. 34–58.

[12] Shangguan C., Ge G. New bounds on the number of tests for disjunct matrices // IEEE Trans. Inf. Theory. 2016, vol. 62, no. 12, pp. 7518–7521.

[13] http://www.isthe.com/chongo/tech/comp/fnv/

[14] https://gist.github.com/StevenClontz/4445774/raw/1722a289b665d940495645a5eaaad4da8e3ad4c7/mobydick.txt

[15] https://nsucrypto.nsu.ru/olymp/2018/round/2/task/4