Task 1. «WaterMarking Cipher»

Special Prize from the Program Committee!

Let X, Y and K be the sets of plaintexts, ciphertexts and keys respectively, where $X = Y = \{0,1\}^n$ and $K = \{0,1\}^m$ for some integer n and m. Recall that two functions $E : X \times K \to Y$ and $D : Y \times K \to X$ are called *an encryption algorithm* and *a decryption algorithm* respectively if for any $x \in X$, $k \in K$ it holds $D(E(x,k),k) = x$. Together E and D form *a cipher*.

Let us call a cipher *watermarking* if for any key $k \in K$ and any subset $I \subseteq \{1,2,\ldots,n\}$ there exists a key k_I such that for any $x \in X$ it holds

$$D(E(x,k),k_I) = x',$$

where x' is obtained from x by changing all bits with coordinates from I.

A simple example of such a cipher. Let $m = n$ and encryption and decryption algorithms be the following:

$$E(x,k) = x \oplus k \quad \text{and} \quad D(y,k) = y \oplus k.$$

For any set I and any key k we can easily get the key k_I that is obtained from k by changing all bits with coordinates from I. The main disadvantage of such a cipher that every key should be used only once.

How can we use a watermarking cipher? Suppose you own some digital products (for example, videos), which you want to sell. Let x represent a binary code of a product. For each customer of x you choose the unique set I of coordinates and send to him the encrypted with the key k copy y and the correspondent key k_I. Then after receiving y and k_I the customer decrypts y and gets x'. The difference between the original x and x' is not significant; thus the customer does not know about it. If someone illegally spreads on the Internet bought by him product, you can easily understand who do it because you choose the unique set I for each customer!

Summarize the ideas we need to construct a cipher that has to put into the video something like a «watermark». Lets try! So, the task is to construct a watermarking cipher. Please think about easy usage of it for an owner and a customer.

Task 2. «An APN Permutation»

Special Prize from the Program Committee!

Suppose we have a mapping F from $\mathbb{F}_2^n$ to itself (recall that $\mathbb{F}_2^n$ is the vector space of all binary vectors of length n). This mapping is called a **vectorial Boolean function in n variables**. Such functions are used, for example, as S-boxes in block ciphers and should have special cryptographic properties. In this task we consider the following two properties and the problem of combining them.

- A function F in n variables is a **permutation** if for all distinct vectors $x, y \in \mathbb{F}_2^n$ it has distinct images, i. e. $F(x) \neq F(y)$.
- A function F in n variables is called **Almost Perfect Nonlinear** (APN) if for any nonzero vector $a \in \mathbb{F}_2^n$ and any vector $b \in \mathbb{F}_2^n$ an equation $F(x) \oplus F(x \oplus a) = b$ has at most 2 solutions. Here $\oplus$ is the coordinate-wise sum of vectors modulo 2.

Try to find an APN permutation in 8 variables or prove that it doesn't exist.

History of the problem. The question «Does there exist an APN permutation in even number of variables?» has been studied for more that 20 years. If the number of variables is odd, APN permutations exist as it was proved by K. Nyberg (1994). It is known that for 2 and 4 variables the answer is «No». But for 6 variables J.F. Dillon and K. Browning, M. McQuistan, A.J Wolfe have found such a function in 2009! You can see it bellow:

$$G = (\ 0\ \ 54\ 48\ 13\ 15\ 18\ 53\ 35\ 25\ 63\ 45\ 52\ 3\ \ 20\ 41\ 33$$
$$59\ 36\ 2\ \ 34\ 10\ 8\ \ 57\ 37\ 60\ 19\ 42\ 14\ 50\ 26\ 58\ 24$$
$$39\ 27\ 21\ 17\ 16\ 29\ 1\ \ 62\ 47\ 40\ 51\ 56\ 7\ \ 43\ 44\ 38$$
$$31\ 11\ 4\ \ 28\ 61\ 46\ 5\ \ 49\ 9\ \ 6\ \ 23\ 32\ 30\ 12\ 55\ 22\ ).$$

This function is presented as the list of its values, i. e. $G(0) = 0, G(4) = 15, G(16) = 59$ and so on. For brevity we use integers instead of binary vectors. A binary vector $x = (x_1, \ldots, x_n)$ corresponds to an integer $k_x = x_1 \cdot 2^{n-1} + x_2 \cdot 2^{n-2} + \ldots + x_{n-1} \cdot 2 + x_n$.

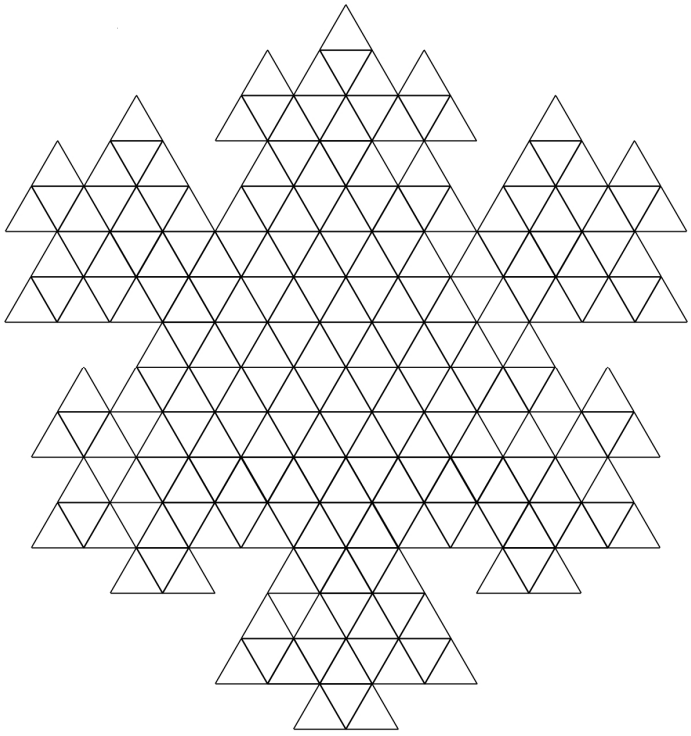
Thus, you are welcome to study the next case, $n = 8$.

Task 3. «The Snowflake cipher»

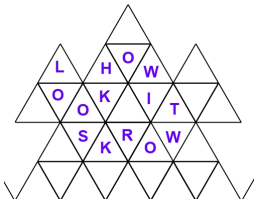
Alice wants to encrypt some text using the Snowflake cipher. Encryption is described by the following algorithm:

Step 1. Choose an arbitrary small triangle in the snowflake (see below);
Step 2. Put the first letter of your message into this triangle;
Step 3. Write the next letter of the message (without spaces) into an arbitrary empty neighbouring triangle. Neighbouring means having a common edge. Repeat this step until the end of message.
Step 4. After inserting of all the letters, write down the text from snowflake in horizontal order from top to bottom and left to right.

Determine what is the maximal possible length of a message that can be encrypted with the Snowflake cipher?



An example. We want to encrypt the message: `LOOK HOW IT WORKS`. As a result we can get the ciphertext: `LHOWOOKITSKROW`.



Task 4. «The number of solutions»

Let $\mathbb{F}_{256}$ be the finite field of characteristic 2 with 256 elements. Consider the function

$$F : \mathbb{F}_{256} \to \mathbb{F}_{256} \text{ such that } F(x) = x^{254}.$$

Since $x^{255} = 1$ for all nonzero $x \in \mathbb{F}_{256}$, we have $F(x) = x^{-1}$ for all nonzero elements of $\mathbb{F}_{256}$. Further, we have $F(0) = 0$.

Alice is going to use the function F as an S-box (that maps 8 bits to 8 bits) in a new block cipher. But before she wants to find answers to the following questions.

- How many solutions may the equation

$$F(x + a) = F(x) + b \tag{1}$$

  have for all different pairs of nonzero parameters a and b, where $a, b \in \mathbb{F}_{256}$?

- How many solutions does the equation (1) have for the function $F(x) = x^{2^n-2}$ over the finite field $\mathbb{F}_{2^n}$ for an arbitrary n?

Please, help to Alice!

Task 5. «Super-Sboxes for AES: differential characteristics»

Special Prize from the Program Committee!

Let $\mathbb{F}_{256}$ be the finite field of 256 elements and α be a primitive element (it means that for any nonzero $x \in \mathbb{F}_{256}$ there exists $i \in \mathbb{N}$ such that $x = \alpha^i$). Let $\mathbb{F}_{256}^4$ be the vector space of dimension 4 over $\mathbb{F}_{256}$. Thus, any element $x \in \mathbb{F}_{256}^4$ is $x = (x_1, x_2, x_3, x_4)$, where $x_i \in \mathbb{F}_{256}$. An arbitrary function from $\mathbb{F}_{256}^4$ to $\mathbb{F}_{256}^4$ can be considered as the set of 4 coordinate functions from $\mathbb{F}_{256}^4$ to $\mathbb{F}_{256}$. Define the following auxiliary functions $F_4, M : \mathbb{F}_{256}^4 \to \mathbb{F}_{256}^4$:

$$F_4(x_1, x_2, x_3, x_4) = (x_1^{254}, x_2^{254}, x_3^{254}, x_4^{254});$$

$$M(x_1, x_2, x_3, x_4) = (x_1, x_2, x_3, x_4) \times \begin{bmatrix} \alpha+1 & 1 & 1 & \alpha \\ \alpha & \alpha+1 & 1 & 1 \\ 1 & \alpha & \alpha+1 & 1 \\ 1 & 1 & \alpha & \alpha+1 \end{bmatrix}.$$

Consider the function $G : \mathbb{F}_{256}^4 \to \mathbb{F}_{256}^4$ that is a combination of F_4 and M:

$$G(x_1, x_2, x_3, x_4) = F_4(M(F_4(x_1, x_2, x_3, x_4))).$$

Find the number of solutions of the equation $G(x + a) = G(x) + b$, where parameters a and b run all nonzero values from $\mathbb{F}_{256}^4$.

Foundation of the problem. J. Daemen and V. Rijmen, the designers of AES (Rijndael), have introduced the Super-Sbox representation of two rounds of AES in order to study differential properties. The function G can be considered as a simplified Super-Sbox model of two rounds of AES. To study resistance of AES to differential cryptanalysis, we welcome you to start with differential characteristics of the function G.

Task 6. «Boolean cubes»

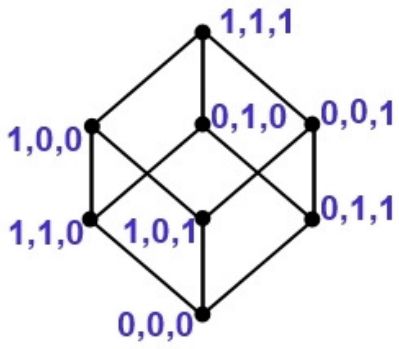
Alice has two cubes E_1 and E_2 of dimension 3 (see the picture below). Their vertices have labels consisting of three integers; for example, (1,0,1) consists of integers 1, 0, 1. Consider an operation A that can be applied for a cube. The operation A contains three steps:

Step 1. Take an arbitrary edge of the cube;

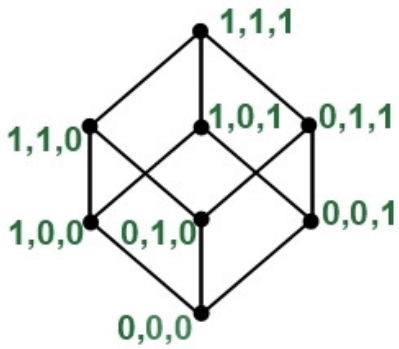
Step 2. Take the number a equal to 1 or -1;

Step 3. Add a to an arbitrary position of the first vertex of the chosen edge. Add a to an arbitrary position of the second vertex of the edge.

Is it possible to get the cube E_2 from the cube E_1 by applying the operation A as many times as necessary? Give your arguments.

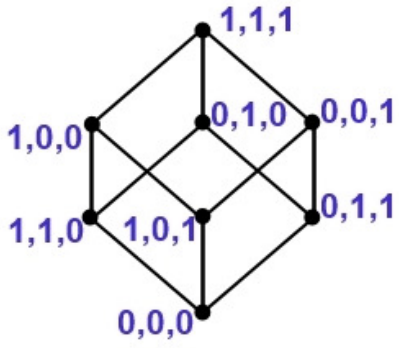


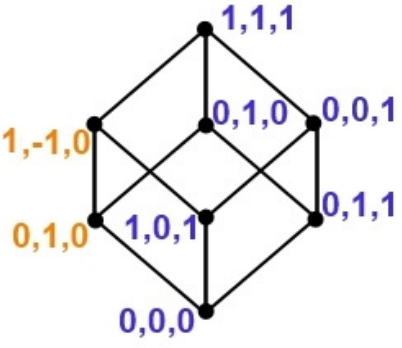
The cube E_1



The cube E_2

An example of applying an operation. Step 1. Take the edge $((1,0,0);(1,1,0))$. Step 2. Let $a=-1$. Step 3. For the vertex $(1,0,0)$ we choose position 2 and for the vertex $(1,1,0)$ we choose position 1; after adding the edge $((1,0,0);(1,1,0))$ becomes $((1,-1,0);(0,1,0))$.





Task 7. «A special parameter»

In differential cryptanalysis of block ciphers a special parameter P is used to measure the diffusion strength. In this task we study its properties.

Let n, m be positive integer numbers. Let $a = (a_1, \ldots, a_m)$ be a vector, where a_i are elements of the finite field $\mathbb{F}_{2^n}$. Denote by $\mathrm{wt}(a)$ the number of nonzero coordinates a_i, $i = 1, \ldots, m$, and call this number the *weight* of a.

We say that $a, b \in \mathbb{F}_{2^n}^m$ represent *states*. The sum of two states a, b is defined as $a + b = (a_1 + b_1, \ldots, a_n + b_n)$.

Thus, the *special parameter* P of a function $\varphi : \mathbb{F}_{2^n}^m \to \mathbb{F}_{2^n}^m$ is given by

$$P(\varphi) = \min_{a,\, b,\, \text{such that } a \neq b} \{\mathrm{wt}(a + b) + \mathrm{wt}(\varphi(a) + \varphi(b))\}.$$

- Rewrite (simplify) the definition of $P(\varphi)$ when the function φ is linear (recall that a function ℓ is linear if for any x, y it holds $\ell(x + y) = \ell(x) + \ell(y)$).
- Rewrite the definition of $P(\varphi)$ in terms of linear codes, when the linear transformation φ is given by a $m \times m$ matrix M over $\mathbb{F}_{2^n}$, i. e. $\varphi(x) = M \cdot x$.
- Let φ be an arbitrary function. Find a tight upper bound for $P(\varphi)$ as a function of m.
- Can you give an example of the function φ with the maximal possible value of P?

Task 8. «A pseudo-random generator»

Alice and Bob communicate in Russia through the Internet using some protocol. In the process of communication Bob sends random numbers to Alice. It is known, that Bob's pseudo-random generator works in the following way:

1. it generates the binary sequence $u_0, u_1, u_2, \ldots$, where $u_i \in \mathbb{F}_2 = \{0, 1\}$, such that for some secret $c_0, \ldots, c_{15} \in \mathbb{F}_2$ it holds

$$u_{i+16} = c_{15}u_{i+15} \oplus c_{14}u_{i+14} \oplus \ldots \oplus c_0 u_i \text{ for all integer } i \geqslant 0;$$

2. i-th random number r_i, $i \geqslant 1$, is calculated as

$$r_i = u_{16i} + u_{16i+1}2 + u_{16i+2}2^2 + \ldots + u_{16i+15}2^{15};$$

3. Bob initializes $u_0, u_1, \ldots, u_{15}$ using some integer number IV (initial value), where $0 < IV < 2^{16}$, by the same way, i. e.

$$IV = u_0 + u_1 2 + u_2 2^2 + \ldots + u_{15}2^{15};$$

4. it is known that as IV Bob uses the number of seconds from January 1, 1970, 00:00 (in his time zone) to his current time (in his time zone too) modulo 2^{16}.

Eva has intercepted the third and the fourth random numbers ($r_3 = 9\,731$ and $r_4 = 57\,586$). She lives in Novosibirsk and knows that Bob has initialized the generator at November 17, 2014, at about 12:05 UTC+6 up to several minutes. The number of seconds from January 1, 1970, 00:00 UTC+6 to November 17, 2014, 12:05 UTC+6 is equal to $1\,416\,225\,900$.

Help Eva to detect Bob's time zone.

Task 9. «Add-Rotate-Xor»

Let $\mathbb{F}_2^n$ be the vector space of dimension n over $\mathbb{F}_2 = \{0, 1\}$. A vector $x \in \mathbb{F}_2^n$ has the form $x = (x_1, x_2, \ldots, x_n)$, where $x_i \in \mathbb{F}_2$. This vector can be interpreted as the integer $x_1 \cdot 2^{n-1} + x_2 \cdot 2^{n-2} + \ldots + x_{n-1} \cdot 2 + x_n$.

Alice can implement by hardware the following functions from $\mathbb{F}_2^n$ to $\mathbb{F}_2^n$ for all vectors $a, b \in \mathbb{F}_2^n$ and all integers r, $0 < r < n$:

1) $f_a(x) = x \boxplus a$ — addition of vectors x and a as integers modulo 2^n for a fixed a;

2) $g_r(x) = x \lll r$ — cyclic rotation of a vector x to the left by r positions for a fixed positive integer r;

3) $h_b(x) = x \oplus b$ — coordinate-wise sum of vectors x and b modulo 2 for a fixed b.

- Bob asks Alice to construct two devices that compute the functions S_1 and S_2 from $\mathbb{F}_2^2$ to $\mathbb{F}_2^2$ given by their truth table:

x	(00)	(01)	(10)	(11)
$S_1(x)$	(01)	(00)	(10)	(11)
$S_2(x)$	(01)	(11)	(00)	(01)

  Can Alice do it? If «yes», show how it can be done; if «no», give an explanation!

- Generalizing the problem above: can we construct any function from $\mathbb{F}_2^n$ to $\mathbb{F}_2^n$ using only a finite number of compositions of functions f_a, g_r and h_b?

  And what about any permutation over $\mathbb{F}_2^n$?

  Consider at least the cases $n = 2, 3, 4$.

- Is it possible to compute every function h_b using only functions f_a and g_r?

Task 10. «Linear subspaces»

For constructing a new secret sharing scheme Mary has to solve the following task on binary vectors. Let n be an integer number, $n \geqslant 2$. Let $\mathbb{F}_2^{2n}$ be a $2n$-dimensional vector space over $\mathbb{F}_2$, where $\mathbb{F}_2 = \{0, 1\}$ is a prime field of characteristic 2.

Do there exist subsets $L_1, \ldots, L_{2^n+1}$ of $\mathbb{F}_2^{2n}$ such that the following conditions hold

1. L_i is a linear subspace of dimension n for every $i \in \{1, \ldots, 2^n + 1\}$;
2. $L_i \cap L_j = \{\mathbf{0}\}$ for all $i, j \in \{1, \ldots, 2^n + 1\}$, $i \neq j$;
3. $L_1 \cup \ldots \cup L_{2^n+1} = \mathbb{F}_2^{2n}$?

If «yes», show how to construct these subspaces for an arbitrary integer n.

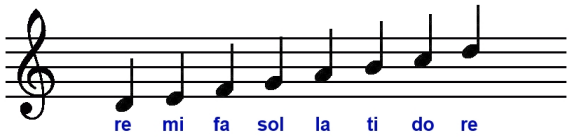
Remark I. Recall several definitions and notions. Each element $x \in \mathbb{F}_2^k$ is a binary vector of length k, i. e. $x = (x_1, \ldots, x_k)$, where $x_1, \ldots, x_k \in \mathbb{F}_2$. For two vectors x and y of length k their sum is $x \oplus y = (x_1 \oplus y_1, \ldots, x_k \oplus y_k)$, where $\oplus$ stands for XOR operation. Let $\mathbf{0}$ be the zero element of the vector space, i. e. vector with all-zero coordinates. A nonempty subset $L \subseteq \mathbb{F}_2^k$ is called a *linear subspace* if for any $x, y \in L$ it holds $x \oplus y \in L$. It is easy to see that zero vector belongs to every linear subspace. A linear subspace L of $\mathbb{F}_2^k$ has *dimension* n if it contains exactly 2^n elements.

Remark II. For example the case $n = 2$ we consider together. In the vector space $\mathbb{F}_2^4$ we can choose the following 5 required subspaces:

$$L_1 = \{(0000), (0001), (1110), (1111)\};$$
$$L_2 = \{(0000), (0010), (1001), (1011)\};$$
$$L_3 = \{(0000), (0011), (0100), (0111)\};$$
$$L_4 = \{(0000), (0101), (1000), (1101)\};$$
$$L_5 = \{(0000), (0110), (1010), (1100)\}.$$

Task 11. «The musical notation»

Alice and Bob invented a new way for encrypting messages based on musical notations of melodies. They are not very good in musical notations but they know the basic notes «do», «re», «mi», «fa», «sol», «la», «ti», and their places in the staff:



To encrypt a message of length n in English alphabet Alice chooses a melody consisting of n notes. She writes a message under the musical notation of the melody in such a way that each letter of the message corresponds to exactly one note's position in the musical notation. Then for each note («do», «re», ..., «ti») Alice forms the ordered group of corresponding letters. Further she takes a random integer number k_i, $i = 1, \ldots, 7$, and cyclically shifts letters in the i-th group on k_i positions to the right. After that Alice forms the ciphertext by writing letters of the shifted groups under the musical notation again.

An example. Suppose that Alice wants to send the message `H E L L O`.

The group for «re» is (`E`, `L`); for «mi» — (`H`, `L`, `O`). Alice takes random numbers 2 and 1 for «re» and «mi» respectively. After shifting she gets groups (`E`, `L`) and (`O`, `H`, `L`).
Hence the ciphertext for the message is `O E H L L`.

Decrypt the following ciphertext sent to Bob by Alice:

`R O L E L I S E O E E E H T O M V C P B D E F S O N`

It is known that Alice used the musical notation below.