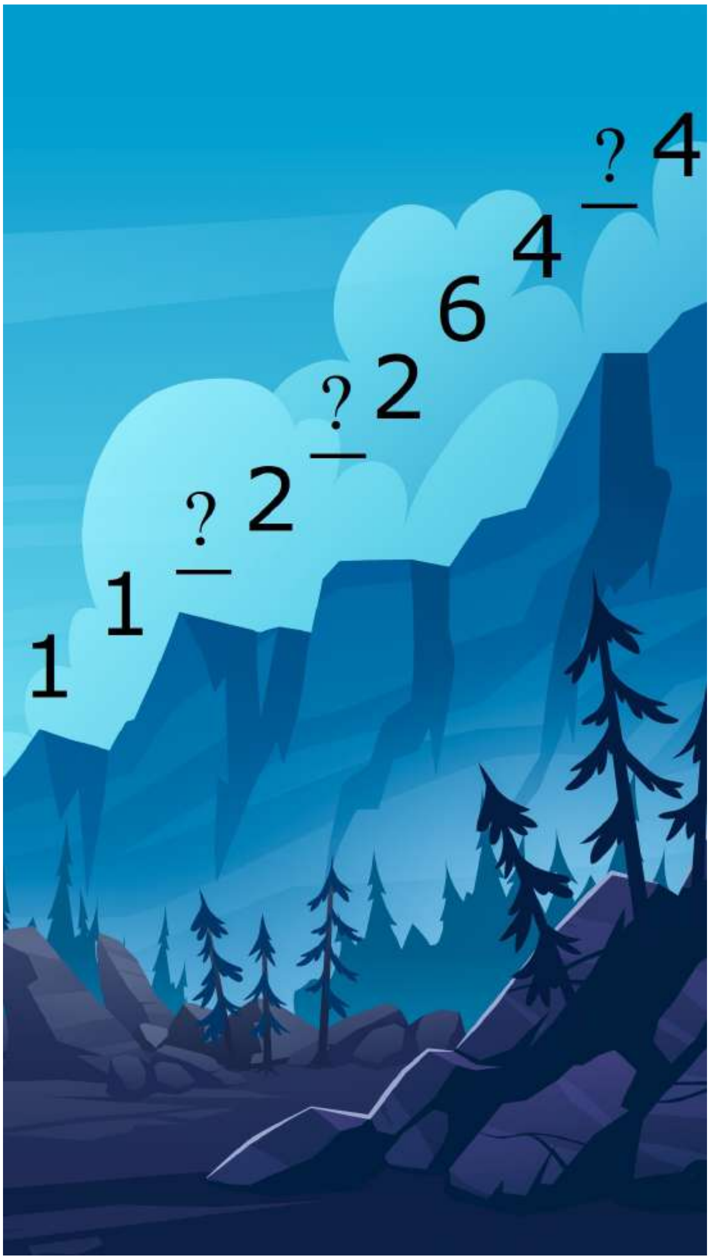
Problem 1. «Crypto growth»

One old, experienced cryptographer told his young student: «You can't live without this function. Really, I believe you'll recognize it anywhere.»

Do you?



Problem 2. «Understand the Oracle»

Bob challenges Alice by providing a matrix $\mathbf{B} \in \mathbb{Z}^{2n\times 2n}$ and asks her to find an integer vector $\mathbf{g} \in \mathbb{Z}^{2n}$ that is short in the Euclidean norm and satisfies $\mathbf{Bf} = \mathbf{g}$, for some $\mathbf{f} \in \mathbb{Z}^{2n}$.

Alice possesses a magical oracle capable of finding short vectors in dimension n, and it can be invoked any number of times. Bob further guarantees to Alice that there exists a way to reduce the dimension of $\mathbf{B}$ to n, and that adding any two coefficients of the vector $\mathbf{g}$ yields a small value. The first time, Bob challenged Alice by giving the matrix

$$\mathbf{B} = \begin{pmatrix} \mathbf{H}_0 & \mathbf{H}_1 \\ \mathbf{H}_1 & \mathbf{H}_0 \end{pmatrix}.$$

Alice quickly realized that if she splits $\mathbf{g}$ as $(\mathbf{g}_0, \mathbf{g}_1)$, then she can provide the oracle with the matrix $\mathbf{B}_0 = \mathbf{H}_0 + \mathbf{H}_1$ of dimension n, such that $\mathbf{B}_0(\mathbf{f}_0 + \mathbf{f}_1) = (\mathbf{g}_0 + \mathbf{g}_1)$ holds. Since $\mathbf{g}_0 + \mathbf{g}_1$ is still a short vector (by Bob's guarantee), the oracle can find $\mathbf{g}_0 + \mathbf{g}_1$. Similarly, she can invoke the oracle with $\mathbf{B}_1 = \mathbf{H}_0 - \mathbf{H}_1$ to obtain the vector $\mathbf{g}_0 - \mathbf{g}_1$, and ultimately recover $\mathbf{g}$.

Bob then made the challenge harder and gave Alice a matrix of the form

$$\mathbf{B} = \begin{pmatrix} \mathbf{H}_0 & \mathbf{H}_1 \\ \mathbf{H}_1{}^T & \mathbf{H}_0{}^T \end{pmatrix},$$

where $\mathbf{H}_0$ is a right circulant matrix and $\mathbf{H}_1$ is a left-circulant matrix, and $\mathbf{H}_0^T, \mathbf{H}_1^T$ denote the transposes of $\mathbf{H}_0$ and $\mathbf{H}_1$, respectively. Can you help Alice to reduce this matrix to dimension n so that she can use the oracle?

Note 1: We say that $\mathbf{H}_0$ is a *right circulant matrix* if it has the form $\begin{pmatrix} a_0 & a_1 & \dots & a_{n-1} \\ a_{n-1} & a_0 & \dots & a_{n-2} \\ \vdots & \vdots & \ddots & \vdots \\ a_1 & a_2 & \dots & a_0 \end{pmatrix}$,

and similarly $\mathbf{H}_1$ is a *left circulant matrix* if it has the form $\begin{pmatrix} b_0 & b_1 & \dots & b_{n-1} \\ b_1 & b_2 & \dots & b_0 \\ \vdots & \vdots & \ddots & \vdots \\ b_{n-1} & b_0 & \dots & b_{n-2} \end{pmatrix}$.

Note 2: This problem is related to lattice-based cryptography, as finding the short vector $\mathbf{g}$ corresponds to solving the Shortest Vector Problem (SVP) in the lattice generated by $\mathbf{B}$.

Problem 3. «Key for the 2025»

The cipher key is defined by the positive integers a, b, c, d, e, f, g, h, i, such that the following relation holds:

$$a^3 + b^3 + c^3 + d^3 + e^3 + f^3 + g^3 + h^3 + i^3 = 2025^{2026}.$$

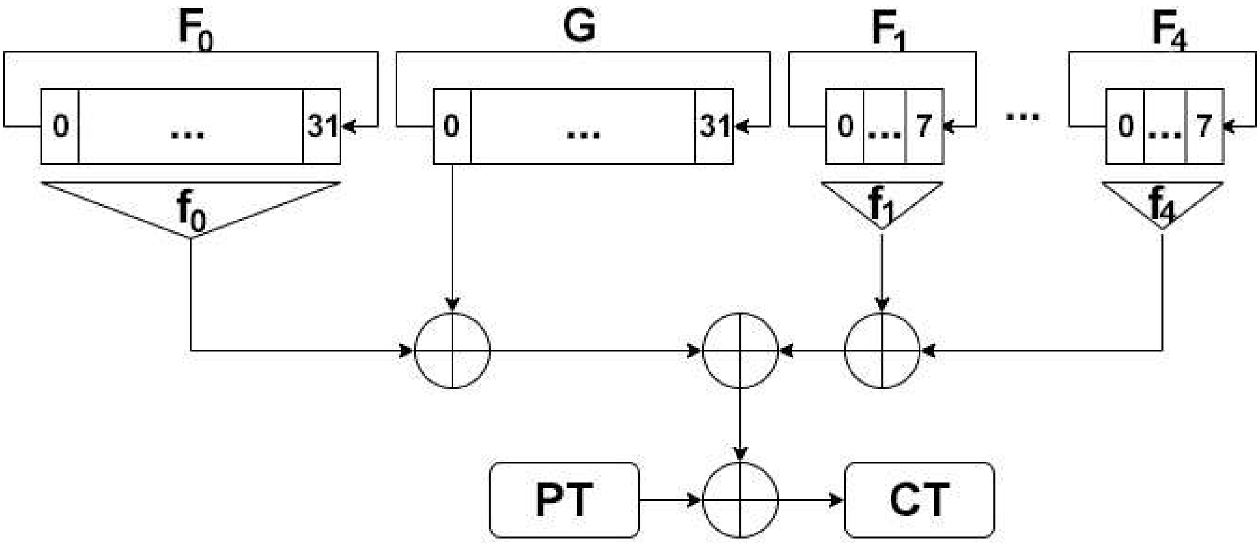
Please, find the key!

Problem 4. «Negotiations»

Before the upcoming negotiations, our intelligence managed to intercept an encrypted telegram sent from the enemy headquarters to its negotiation team:

```
289e1fa87b606203a790f93e3a3e4c1a
7b60474a1df415badf49b12b6b6e16ef
7e559f015c.
```

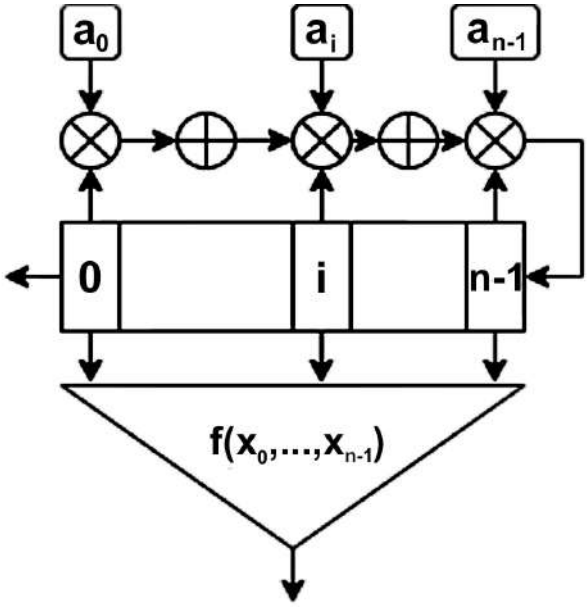
Undercover operative Jack Beam managed to recover the schematics of the cipher device in use. It is given below:



The cipher device is a stream cipher consisting of six binary shift registers, where for each clock cycle a bit of the output keystream is XORed with a bit of plaintext.

Another undercover operative, Jim Daniels, managed to provide a more detailed description of the shift registers.

For each clock cycle, the register output is formed by feeding its state into a filter function f (least significant bits are on the left). The state is then bitwise multiplied with the coefficients $a_0, \ldots, a_{n-1} \in GF(2)$ of a feedback polynomial $F(x) = x^n + a_{n-1}x^{n-1} + \ldots + a_1 x + a_0$. The results of the multiplications are summed modulo 2, the register state is shifted one cell towards the least significant bits, and the result of the sum is written into the most significant cell, see the next figure.



Jim Daniels wrote down the feedback polynomials:

$F_0(x) = x^{32} + x^{15} + x^9 + x^7 + x^4 + x^3 + 1,$
$G(x) = x^{32} + x^{31} + x^{29} + x^{25} + x^{19} + x^{18} + x^{17} + x^{16} + x^9 + x^8 + x^7 + x^3 + x^2 + x + 1,$
$F_1(x) = x^8 + x^7 + x^6 + x^5 + x^2 + x + 1,$
$F_2(x) = x^8 + x^5 + x^3 + x + 1,$
$F_3(x) = x^8 + x^6 + x^5 + x^2 + 1,$
$F_4(x) = x^8 + x^6 + x^5 + x + 1.$

He found the truth tables in hexadecimal format for the filter functions $f_1, \dots, f_4$:

f_1 : 431DCF02529EF04356F2B59E90860BD22FBCEFDB380F8838767DD716E9712A04,
f_2 : 4617142FE4475B3FF3D50CBB58E7A0F40CAC85B8C278A0C131FFD85413BE9E4A,
f_3 : 0138CCC35F889EAA5D8F1EE0442FB54D2AEAEFB96288DB3303FD1FBF65860A20,
f_4 : 095F8FFC8CA0C53C5D3782243D36EE575186DBD244CB9DE1E0ED1730CACF2053.

Also he got the function f_0. Thus, $f_0(x_0, \dots, x_{31}) = f(x_5, x_{21}, x_{29}, x_1, x_0, x_{27}, x_{12})$, where it holds $f(x_0, \dots, x_7) = x_0x_2x_5x_6 + x_0x_3x_5x_6 + x_0x_1x_5x_6 + x_1x_2x_5x_6 + x_0x_2x_3x_6 + x_1x_3x_4x_6 + x_1x_3x_5x_6 + x_0x_2x_4 + x_0x_2x_3 + x_0x_1x_3 + x_0x_2x_6 + x_0x_1x_4 + x_0x_1x_6 + x_1x_2x_6 + x_2x_5x_6 + x_0x_3x_5 + x_1x_4x_6 + x_1x_2x_5 + x_0x_3 + x_0x_5 + x_1x_3 + x_1x_5 + x_1x_6 + x_0x_2 + x_1 + x_2x_3 + x_2x_5 + x_2x_6 + x_4x_5 + x_5x_6 + x_2 + x_3 + x_5$.

The second register lacks a filter function, its output is the least significant (leftmost) bit of the register state.

It is known that the first 22 bytes of the intercepted message contain direct output of the cipher device without XORing any plaintext, and the subsequent 15 bytes contain an encrypted message, capable of altering the course of the upcoming negotiations.

The encrypted message consists of letters of the Latin alphabet in a 5-bit encoding:

Symbol	Code	Symbol	Code	Symbol	Code	Symbol	Code
A	00001	H	01000	O	01111	V	10110
B	00010	I	01001	P	10000	W	10111
C	00011	J	01010	Q	10001	X	11000
D	00100	K	01011	R	10010	Y	11001
E	00101	L	01100	S	10011	Z	11010
F	00110	M	01101	T	10100		
G	00111	N	01110	U	10101		

Question Could you help your group to decrypt the message and in this way to strengthen its negotiation position?

Remark. Let us present here some test vectors for the problem.

Register 1.
Initial state: 00000000000000001000001010001000
LFSR output: 0000000000000000100000101000100001000000000011001100
Register output: 0101001100000001001011011101111011010110110100011100

Register 2.
Initial state: 11010001000010010000101011001011
Register output: 1101000100001001000010101100101100100000111011101101

Register 3.
Initial state: 11111111
LFSR output: 11111111011101010011
Register output: 00100011010010010110

Register 4.
Initial state: 00101110
LFSR output: 00101110110111100101
Register output: 10001101001001011010

Register 5.
Initial state: 00101110
LFSR output: 00101110111101100111
Register output: 00110100100101101000

Register 6.
Initial state: 00101100
LFSR output: 00101100110110000111
Register output: 00101100010011011000

Problem 5. «Password to coins»

A famous Roman found a wallet with coins during his morning walk. To open it, he needs to enter a password hidden in the following line:

```
5655555556f012346789abcde5f012346789abcde5f012346789abcde5f012346789
abcde5555555558d6ac5b6c8c8bec8b8c7cec5c9b4b3c8cab8c7cec5c9b47657f607
18293a4bcde56789abcde5f01234f60718293a4bcde56789abcde5f0123444444444
56f57a7b55555555556ecbfe699badb4c3aaff8e4529b553cd616e459a11057a82ddf
155555555
```

Help Julius to get the password and take 5 coins from the wallet.

Problem 6. «A Greek cipher»

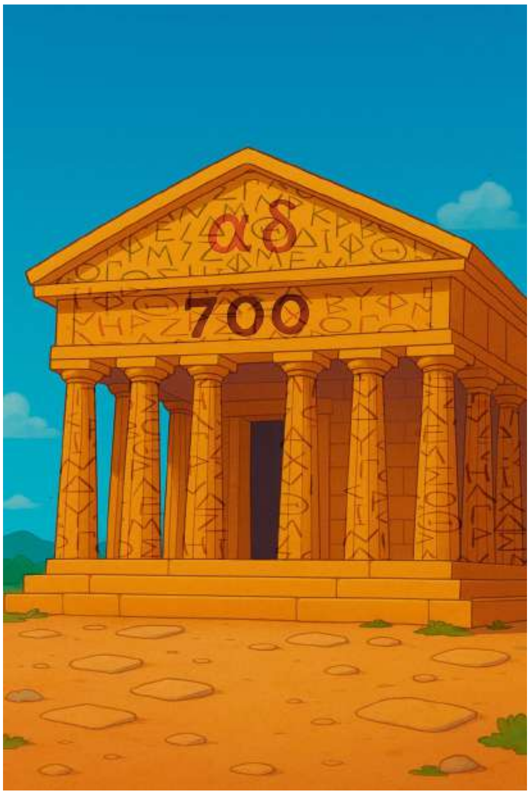
To encrypt the three-letter message we did the following. We matched each letter with it's numeric equivalent according to the table, and got p_1, p_2, p_3.

0	1	2	3	4	5	6	7	8	9
␣	A	B	C	D	E	F	G	H	I
10	11	12	13	14	15	16	17	18	19
J	K	L	M	N	O	P	Q	R	S
20	21	22	23	24	25	26			
T	U	V	W	X	Y	Z			

Then we chose secret natural number δ and formed $p_4 = p_1 + p_2 + p_3 + \delta$.
After that we chose another secret natural number α and calculated for $i = 1, 2, 3, 4$

$$c_i = p_i + 2p_{i+1} + (-1)^{\frac{i+1}{2}} \cdot \delta \bmod 27, \text{ if } i \text{ is odd},$$
$$c_i = p_{i-1} + p_i + (-1)^{\frac{i}{2}} \cdot \alpha \bmod 27, \text{ if } i \text{ is even}.$$

As a result we have got: «WGAD». Recover the secret message.



Problem 7. «Toy cipher cryptanalyst»

Bob is a beginner in cryptography and, for fun, he makes cryptanalysis for toy versions of various ciphers. One of his functions written in C++ is below.

```
uint32_t foo(uint32_t x) {
  uint32_t y = 0x20000000;
  for (uint32_t i = 0x40000000; i != 0x80000000; ++i) {
    if (x == y)
      return i;
    y = y + ((i << 2) >> 1) + (i >> 30) + 1;
    y = ((y << 1) >> 1) + (y >> 31);
  }
  return 0x80000000;
}
```

What is the purpose of this function? Is there anything that needs to be fixed in this function? Please provide as many details as possible in your answers.

Problem 8. «Bijections for ciphers»

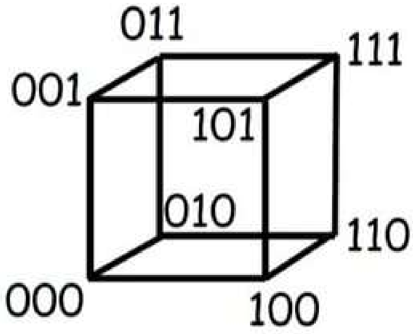
A mapping $F : \{0,1\}^n \to \{0,1\}^n$, $F = (f_1, \ldots, f_n)$, is such that all coordinate Boolean functions f_i, $i = 1, \ldots, n$, depend on k variables, $k \leqslant n$. Each function f_i is defined as follows: by a Boolean function g_i in k variables and an integer vector m_i of length k, containing the indices of the essential variables.

Example: Let $n = 3$, $k = 2$, $g_1 = g_2 = g_3 = x_1x_2$, $m_1 = (2,3)$, $m_2 = (1,3)$, $m_3 = (1,2)$; then $f_1 = x_2x_3$, $f_2 = x_1x_3$, $f_3 = x_1x_2$, and the mapping F is given by the table:

x_1	x_2	x_3	F
0	0	0	000
0	0	1	000
0	1	0	000
0	1	1	100
1	0	0	000
1	0	1	010
1	1	0	001
1	1	1	111

A problem. Formulate conditions (necessary; sufficient; both) on the functions g_i and vectors m_i under which the mapping F is a bijection (an one-to-one function).

A requirement: It should be not necessary to construct the truth table for F while checking the conditions.



Problem 9.«Crypto noise»

Bob obtained from Alice the ciphertext $c = (c_1, c_2, \dots, c_{20})$ that is a vector over $\mathbb{Z}_{16}$. He knows that the initial message is a vector $m = (m_1, m_2, m_3, m_4)$ over $\mathbb{Z}_{16}$. He also provided the information that for the ciphertext it holds $c = mA + e \pmod{16}$ where A is a 4×20 integer matrix

$$A = \left(\begin{array}{cccc|cccccccccccccccc}
2 & 2 & 1 & 1 & -1 & 0 & 3 & 7 & -2 & -2 & -1 & -1 & -2 & -2 & -1 & -1 & -2 & -2 & -1 & -1 \\
2 & 1 & 2 & 1 & -2 & -1 & -2 & -1 & -1 & 1 & 2 & 7 & -2 & -1 & -2 & -1 & -2 & -1 & -2 & -1 \\
1 & 2 & 1 & 2 & -1 & -2 & -1 & -2 & -1 & -2 & -1 & -2 & 0 & 0 & 3 & 6 & -1 & -2 & -1 & -2 \\
1 & 1 & 2 & 2 & -1 & -1 & -2 & -2 & -1 & -1 & -2 & -2 & -1 & -1 & -2 & -2 & 0 & 1 & 2 & 6
\end{array}\right)$$

and $e = (e_1, e_2, \dots, e_{20})$ is an unknown «noise» vector with elements from the set $\{-1, 0, +1\}$.

Let $c = (4, 2, 15, 11, 7, 4, 9, 5, 7, 2, 9, 4, 2, 14, 14, 13, 0, 8, 4, 12)$ and $\sum_{i=1}^{20} e_i^2 = 14$, provide the most efficient way to restore the initial message m (or any possible candidates for it).

Remark. The points for the solutions obtained via brute force or any computer algebra systems will be reduced.