23.04.2020 **Sebastian Grzelak**

ECRYPT

IDEA Algorithm

Introduction and algorithm explanation

In cryptography, block cyphers are very important in the designing of many cryptographic algorithms and are widely used to encrypt the bulk of data in chunks. By chunks, it means that the cypher takes a fixed size of the plaintext in the encryption process and generates a fixed size ciphertext using a fixed-length key. An algorithm's strength is determined by its key length.

The International Data Encryption Algorithm (IDEA) is a symmetric key block cypher that:

- uses a fixed-length plaintext of 64 bits and
- encrypts them in 4 chunks of 16 bits each
- to produce 64 bits ciphertext
- The length of the key used is 128 bits
- The key is also divided into 52 subkeys of 4 bits each

This algorithm involves a series of 4 identical complete rounds and 1 half-round. Each complete round involves a series of 14 steps that includes operations like:

- Bitwise XOR
- Addition modulo (2^4)
- Multiplication modulo ($2^4 + 1$)

After 8 complete rounds, the final “half-round” consists of only first 4 out of the 14 steps previously used in the full-rounds. To perform these rounds, each binary notation must be converted to its equivalent decimal notation, perform the operation and the result obtained should be converted back to the binary representation for the final result of that particular step.

Key Schedule: 6 subkeys of 4 bits are used in each complete round, while 4 are used in the half-round. So, 8.5 rounds require 52 subkeys. The given key, ‘K’, directly gives the first 8 subkeys. By rotating the main key left by 24 bits between each group of 8, further groups of 8 subkeys are created.

Example keys table:

	K1	K2	K3	K4	K5	K6
Round 1	1111	0000	0011	1111	1000	1001
Round 2	1010	1100	1111	0011	0100	1100
Round 3	1111	0000	0011	1111	1000	1001
Round 4	1010	1100	1111	0011	0100	1100
Round 5	1111	0000	0011	1111	1000	1001
Round 6	1010	1100	1111	0011	0100	1100
Round 7	1111	0000	0011	1111	1000	1001
Round 8	1010	1100	1111	0011	0100	1100
Round 8.5	1111	0000	0011	1111		

The 64-bit plaintext is represented as **X1 || X2 || X3 || X4**, each of size 16 bits. The 128-bit key is broken into 8 subkeys denoted as **K1 || K2 || K3 || K4 || K5 || K6 || K7 || K8**, again of size 4 bits each. Each round of 14 steps uses the three algebraic operation:

- **Addition modulo (2^4),**
- **Multiplication modulo (2^4)+1**
- **Bitwise XOR**

The steps involved are as follows:

STEP	FORMULA
1	X1 * K1
2	X2 + K2
3	X3 + K3
4	X4 + K4
5	Step 1 ^ Step 3
6	Step 2 ^ Step 4
7	Step 5 * K5
8	Step 6 + Step 7
9	Step 8 * K6
10	Step 7 + Step 9
11	Step 1 ^ Step 9
12	Step 3 ^ Step 9
13	Step 2 ^ Step 10
14	Step 4 ^ Step 10

The input for the next round is **Step 11 || Step 13 || Step 12 || Step 14**, which becomes **X1 || X2 || X3 || X**4. This swap between 12 and 13 takes place after each complete round, except the last complete round (8th round), where the input to the final half round is **Step 11 || Step 12 || Step 13 || Step 14**.

After last complete round, the half-round is as follows:

STEP	FORMULA
1	X1 * K1
2	X2 + K2
3	X3 + K3
4	X4 + K4

Decryption

IDEA decrypts using the same steps as encryption, but new keys must be generated for decryption. K^i_j denotes the j-th decryption key of decryption round i. Z^i_j denotes the j-th encryption key of encryption round i. For the first decryption round: $K^1_1 = (Z^9_1)^{-1}$, where $(Z^9_1)^{-1}$ denotes the multiplicative inverse of the first encryption key of encryption round 9 – the "half round" final transformation – modulo 17; $K^1_2 = -Z^9_2$, where $-Z^9_2$ denotes the additive inverse of the second encryption key of encryption round 9 modulo 16; $K^1_3 = -Z^9_3$; $K^1_4 = (Z^9_4)^{-1}$; $K^1_5 = Z^9_5$; and $K^1_6 = Z^9_6$. The decryption keys are similarly generated in the remaining complete decryption rounds. The decryption keys for the final transformation "half round" are: $K^9_1 = (Z^1_1)^{-1}$, $K^9_2 = -(Z^1_2)$, $K^9_3 = -(Z^1_3)$, and $K^9_4 = (Z^1_4)^{-1}$