**Ministerul Educaţiei și Cercetării al Republicii Moldova**

**Universitatea Tehnică a Moldovei**

**Facultatea Calculatoare, Informatică și Microelectronică**



*Laboratory work 4*

Subject: Algorithm DES

Done by: Sirețanu Andreia-Cristina

st. gr. FAF-211

Verified by: Mitu Cătălin

Chişinău - 2023

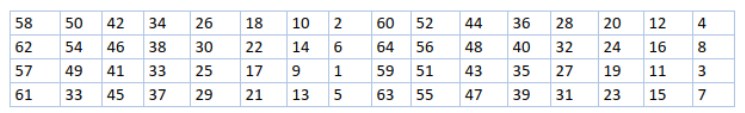
**Subject:** Algorithm DES

**Theory:**

Data Encryption Standard (DES) is a block cipher with a 56-bit key length that has played a significant role in data security. Data encryption standard (DES) has been found vulnerable to very powerful attacks therefore, the popularity of DES has been found slightly on the decline. DES is a block cipher and encrypts data in blocks of size of 64 bits each, which means 64 bits of plain text go as the input to DES, which produces 64 bits of ciphertext. The same algorithm and key are used for encryption and decryption, with minor differences. The key length is 56 bits.

**Work progress:**

1. In the first step, the 64-bit plain text block is handed over to an initial Permutation (IP) function.

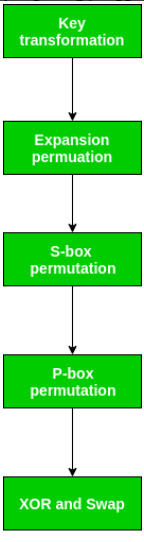


*Figure 1. Initial permutation table*

2. The initial permutation is performed on plain text.

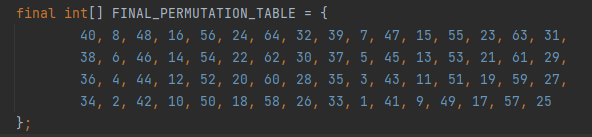
3. Next, the initial permutation (IP) produces two halves of the permuted block; saying Left Plain Text (LPT) and Right Plain Text (RPT).

4. Now each LPT and RPT go through 16 rounds of the encryption process.



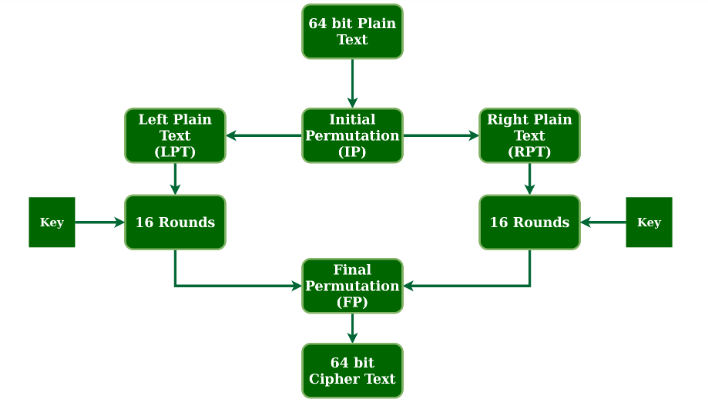
*Figure 2. Broad Level Steps in 16 rounds of permutations*

5. In the end, LPT and RPT are rejoined and a Final Permutation (FP) is performed on the combined block.



*Figure 3. Final permutation table*

6. The result of this process produces 64-bit ciphertext.



*Figure 1. Broad Level Steps in DES*

**Conclusion:** At the current laboratory work I studied how the DES algorithm works and how to implement this algorithm. It was not so hard for my task, but the real challenge was to understand the condition. It was harder to understand how key for round creation works and how to implement this thing.

**Source code:** https://github.com/AndyLy00/CS/tree/main/lab4