

Ministerul Educatiei, Culturii și Cercetarii al Republicii Moldova Universitatea Tehnică a Moldovei

Facultatea Calculatoare, Informatică şi Microelectronică Departamentul Ingineria Software și Automatica

# Raport

# pentru lucrarea de laborator Nr. 4

# la cursul „Metode criptografice de protecție a informației”

A efectuat:

Konjevic Alexandra, gr. FAF-213

A verificat: Mîțu Cătălin

Chișinău – 2023

**Subject**: Block cyphers. DES Algorithm.

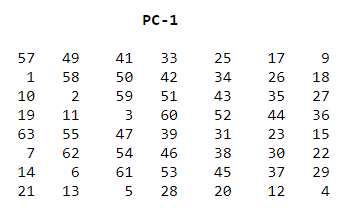
**Objectives:**

Implement one step of the DES algorithm in a chosen programing language. Given my variant (1), I had to determine the key K+, given an 8 symbol DES key.

**Theory:**

The Data Encryption Standard (DES) is a symmetric-key block cipher algorithm that was widely used for secure data encryption in the past. Developed in the early 1970s, DES was the de facto standard for encryption and data security for many years. Although it has been largely replaced by more advanced encryption algorithms like the Advanced Encryption Standard (AES), DES remains a historically significant encryption algorithm.

DES algorithm uses a key converted in binary format, and encrypts every block of 64 bits of the key. Given an initial block, there is performed a permutation, according to the following table:



*Figure 1.* Table for initial permutation of the key

Since the first entry in the table is "57", this means that the 57th bit of the original key becomes the first bit of the permuted key. The 49th bit of the original key becomes the second bit of the permuted key. The 4th bit of the original key is the last bit of the permuted key. Note only 56 bits of the original key appear in the permuted key (bits numbered 8, 16, 24, 32, 40, 48, 56, and 64 are skipped from the initial binary key).

**Task Completion (Variant 1):**

I implemented the task in JavaScript. First of all, the program gets the key from the user, and the input is validated (function `*getInputKey*`). The key needs to be 8 characters long. The key is converted to binary using the function ` *stringToBinary* `:

const stringToBinary = (str) => {

let binaryString = "";

for (let i = 0; i < str.length; i++) {

const charCode = str.charCodeAt(i);

const binaryRepresentation = charCode.toString(2);

binaryString +=

"0".repeat(8 - binaryRepresentation.length) + binaryRepresentation;

}

return binaryString;

};

After that, I declared the PC1 talbe:

const PC1 = [

57, 49, 41, 33, 25, 17, 9, 1, 58, 50, 42, 34, 26, 18, 10, 2, 59, 51, 43, 35,

27, 19, 11, 3, 60, 52, 44, 36, 63, 55, 47, 39, 31, 23, 15, 7, 62, 54, 46, 38,

30, 22, 14, 6, 61, 53, 45, 37, 29, 21, 13, 5, 28, 20, 12, 4,

];

And I wrote a function to display this table in console (`*printTablePC1*`).

After that, I declared the function `*permuteKey*`, which returns a new string, composed of elements of the original key, but placed by the indexes which are the elements of PC1:

const permuteKey = (key) => {

let permutedKey = PC1.map((index) => key[index - 1]).join("");

return permutedKey;

};

**Results:**

Example 1:

Enter key: 64bitkey

key binary: 0011011000110100011000100110100101110100011010110110010101111001

PC1 table:

57 49 41 33 25 17 9

1 58 50 42 34 26 18

10 2 59 51 43 35 27

19 11 3 60 52 44 36

63 55 47 39 31 23 15

7 62 54 46 38 30 22

14 6 61 53 45 37 29

21 13 5 28 20 12 4

permuted key: 00000000111111001111111110010010010101010011101010000011

Example 2:

Enter key: hellokey

key binary: 0110100001100101011011000110110001101111011010110110010101111001

PC1 table:

57 49 41 33 25 17 9

1 58 50 42 34 26 18

10 2 59 51 43 35 27

19 11 3 60 52 44 36

63 55 47 39 31 23 15

7 62 54 46 38 30 22

14 6 61 53 45 37 29

21 13 5 28 20 12 4

permuted key: 00000000111111111111111110000011000001011110101111010000

**Conclusion:**

In conclusion, the Data Encryption Standard (DES) was a pioneering cryptographic algorithm that played a significant role in the history of data security. Developed in the early 1970s, DES introduced the concept of symmetric-key encryption and set the stage for many subsequent encryption standards. Its use of a Feistel network structure, key expansion, permutations, and substitutions provided valuable insights into encryption techniques. However, DES's primary limitation was its 56-bit key length, which made it vulnerable to brute-force attacks as computing power advanced. Recognizing this weakness, the encryption community transitioned to more robust standards like the Advanced Encryption Standard (AES), which offered longer key lengths and enhanced security.

Git repository: https://github.com/alya1007/Labs-semester-5/tree/master/CS