

**Ministerul Educației și Cercetării al Republicii Moldova**  
**Universitatea Tehnică a Moldovei**  
**Facultatea Calculatoare, Informatică și Microelectronică**



## Laboratory work 1: Cryptography and Security

Elaborated:  
st. gr. FAF-211

Echim Mihail

Verified:  
asist. univ.

Aureliu ZGUREANU

Chișinău - 2023

# CRYPTOGRAPHY AND SECURITY

## Subject:

Caesar's Cipher

## Tasks:

1. De implementat algoritmul Cezar pentru alfabetul limbii engleze în unul din limbajele de programare. Utilizați doar codificarea literelor cum este arătat în tabelul 1 (nu se ține seama de folosirea codificărilor specificate în limbajul de programare, de ex. ASCII sau UTF-8 Unicode). Valorile cheii vor fi cuprinse între 1 și 25 inclusiv și nu se permit alte valori. Valorile caracterelor textului sunt cuprinse între 'A' și 'Z', 'a' și 'z' și nu sunt permise alte caractere. În cazul în care utilizatorul introduce alte valori - i se va sugera diapazonul corect. Înainte de criptare textul va fi transformat în majuscule și vor fi eliminate spațiile. Utilizatorul va putea alege operația - criptare sau decriptare, va putea introduce cheia, textul mesajului sau criptograma și va obține respectiv criptograma sau mesajul decriptat.

2. De implementat algoritmul Cezar cu 2 chei, cu posibilitatea condițiilor exprimate în sarcina 1. În plus, cheia 2 trebuie să conțină doar litere ale alfabetului latin, și să aibă o lungime nu mai mică de 7.

## Caesar's Cipher:

Caesar's Cipher, also known as the Caesar Shift or Caesar's Code, is one of the simplest and earliest encryption techniques used in cryptography. It's a substitution cipher that involves shifting the letters of the alphabet by a fixed number of positions. This fixed number is the "key" or "shift value." For example, if you use a Caesar Cipher with a shift value of 3, "A" would be replaced by "D," "B" would become "E," and so on.

A Caesar Cipher with alphabet permutation based on a keyword is a more advanced form of the classic Caesar Cipher that introduces additional security through the use of a keyword to determine the letter permutation. This variation is sometimes called a "Keyword Cipher" or "Caesar Cipher with Keyword."

1. **Keyword Selection:** You start by selecting a keyword. This keyword can be any word or phrase you choose. Let's use the example keyword "SECRET" for illustration.
2. **Permutation Key:** Next, you create a permutation key from the keyword. To do this, you first convert the keyword to lowercase (or uppercase, depending on your preference). Then, you remove any duplicate letters from the keyword. In this case, after removing duplicates, the keyword becomes "SECR." Finally, you arrange these unique letters in alphabetical order to create the permutation key, which becomes "CERS."

3. Encryption: Now, you're ready to encrypt your message. Suppose your plaintext message is "HELLO, WORLD!" Here's how you would encrypt it using the permutation key "CERS":
  - You iterate through each character in the plaintext.
  - For each letter in the plaintext, you find its corresponding position in the standard alphabet. For example, "H" corresponds to position 7 in the alphabet (counting from 0).
  - You then substitute the letter with the letter from the permutation key at the same position. So, in this case, "H" becomes "C" because "C" is at position 7 in the permutation key.
  - You repeat this process for each letter, always referencing the permutation key. Non-letter characters (like punctuation) remain unchanged.
4. Result: After encrypting the entire message, you get the ciphertext. In our example, "HELLO, WORLD!" would become "CERSR, VJNRY!" based on the keyword "SECRET" and the permutation key "CERS."

### Results of the implementation:

I wrote the logic in javascript and made html forms to make the process simple

#### 1.1

In order to encrypt the message, the user needs to input the message and the key and leave the cryptogram field empty, otherwise, an alert will pop up asking to leave the cryptogram field empty. Then just press the Encrypt button.

<p>Cryptogram <input type="text"/></p> <p>Key <input type="text" value="4"/></p> <p>Message <input type="text" value="e"/></p> <p><input type="button" value="Encrypt"/> <input type="button" value="Decrypt"/> <input type="button" value="Clear"/></p>	<p>Cryptogram <input type="text" value="l"/></p> <p>Key <input type="text" value="4"/></p> <p>Message <input type="text" value="e"/></p> <p><input type="button" value="Encrypt"/> <input type="button" value="Decrypt"/> <input type="button" value="Clear"/></p>
--	--

fig.1

To decrypt a message you need to do the same but the other way around and press the Decrypt button.

<p>Cryptogram <input type="text" value="bloob"/></p> <p>Key <input type="text" value="4"/></p> <p>Message <input type="text"/></p> <p><input type="button" value="Encrypt"/> <input type="button" value="Decrypt"/> <input type="button" value="Clear"/></p>	<p>Cryptogram <input type="text" value="bloob"/></p> <p>Key <input type="text" value="4"/></p> <p>Message <input type="text" value="XHKKX"/></p> <p><input type="button" value="Encrypt"/> <input type="button" value="Decrypt"/> <input type="button" value="Clear"/></p>
--	--

fig.2

## 1.2

In order to encrypt with two keys, the user needs to correctly fill the message, the key and the keyword without any repeating letters. Then press the Encrypt button.

<div>Cryptogram <input type="text"/></div> <div>Key <input type="text" value="9"/></div> <div>KeyWord <input type="text" value="flop"/></div> <div>Message <input type="text" value="meme"/></div> <div><input type="button" value="Encrypt"/> <input type="button" value="Decrypt"/> <input type="button" value="Clear"/></div>	<div>Cryptogram <input type="text" value="XRXR"/></div> <div>Key <input type="text" value="9"/></div> <div>KeyWord <input type="text" value="flop"/></div> <div>Message <input type="text" value="meme"/></div> <div><input type="button" value="Encrypt"/> <input type="button" value="Decrypt"/> <input type="button" value="Clear"/></div>
--	---

fig.3

In order to decrypt, the user needs to do the same thing but the other way around.

<div>Cryptogram <input type="text" value="asdfsad"/></div> <div>Key <input type="text" value="4"/></div> <div>KeyWord <input type="text" value="bob"/></div> <div>Message <input type="text"/></div> <div><input type="button" value="Encrypt"/> <input type="button" value="Decrypt"/> <input type="button" value="Clear"/></div>	<div>Cryptogram <input type="text" value="asdfsad"/></div> <div>Key <input type="text" value="4"/></div> <div>KeyWord <input type="text" value="bop"/></div> <div>Message <input type="text" value="ZMOAMZO"/></div> <div><input type="button" value="Encrypt"/> <input type="button" value="Decrypt"/> <input type="button" value="Clear"/></div>
--	--

fig.4

## 1.3

I and a groupmate switched encrypted messages and keys and to our surprise, we gave each other the same message encrypted in different ways :)

He gave me this cryptogram:

cryptogram: DJBDUEVIYHJNNZ

shift: 7

keyword: MICHAEL

<div>Cryptogram <input type="text" value="DJBDUEVIYHJNNZ"/></div> <div>Key <input type="text" value="7"/></div> <div>KeyWord <input type="text" value="MICHAEL"/></div> <div>Message <input type="text"/></div> <div><input type="button" value="Encrypt"/> <input type="button" value="Decrypt"/> <input type="button" value="Clear"/></div>	<div>Cryptogram <input type="text" value="DJBDUEVIYHJNNZ"/></div> <div>Key <input type="text" value="7"/></div> <div>KeyWord <input type="text" value="MICHAEL"/></div> <div>Message <input type="text" value="IAMINYOURWALLS"/></div> <div><input type="button" value="Encrypt"/> <input type="button" value="Decrypt"/> <input type="button" value="Clear"/></div>
---	--

fig.5

Here is my cryptogram:  
cryptogram: MNSMBEYDXFNRRZ  
shift: 4  
keyword: aftonguy

Cryptogram	Cryptogram
<input type="text" value="MNSMBEYDXFNRRZ"/>	<input type="text" value="MNSMBEYDXFNRRZ"/>
Key	Key
<input type="text" value="4"/>	<input type="text" value="4"/>
KeyWord	KeyWord
<input type="text" value="aftonguy"/>	<input type="text" value="aftonguy"/>
Message	Message
<input type="text"/>	<input type="text" value="IAMINYOURWALLS"/>
<input type="button" value="Encrypt"/> <input type="button" value="Decrypt"/> <input type="button" value="Clear"/>	<input type="button" value="Encrypt"/> <input type="button" value="Decrypt"/> <input type="button" value="Clear"/>

fig.6

## Conclusion:

In conclusion, this lab work provided valuable insights into classical encryption techniques, specifically the Caesar Cipher and the Caesar Cipher with Keywords.

I began by implementing the traditional Caesar Cipher, which involved shifting characters by a fixed value. This simple encryption method helped me understand the fundamentals of substitution ciphers and their limitations in terms of security.

Subsequently, I explored an enhanced version of the Caesar Cipher with Keywords. By introducing a keyword-based permutation key, I added an additional layer of complexity to the encryption process. This improved the security of the cipher, making it more resistant to decryption without knowledge of the keyword.

Through these exercises, I gained hands-on experience in both encryption techniques and appreciated the importance of selecting strong keywords to enhance the security of classical ciphers. While these methods are educational and historically significant, it's crucial to acknowledge their vulnerability to modern cryptographic attacks, and they should not be relied upon for securing sensitive information in practical applications.