Ministry of Education, Culture and Research of the Republic of Moldova

Technical University of Moldova

Department of Software and Automation Engineering

REPORT

Laboratory Work Nr.3
Discipline: Cryptographic methods of information protection

Realised by: st.gr. FAF-213
Iațco Sorin

Checked by : asist.univ.
Mîțu Cătălin

Chișinău 2023

Subject: Polyalphabetic ciphers.

Tasks:

Implement the Playfair algorithm in one of the programming languages for Messages in Romanian language (31 letters). Text character values range from 'A' and 'Z', 'a' and 'z' and no other values are allowed. If the user enters other values - the correct range of characters will be suggested. The length of the key should not be less of 7. The user will be able to choose the operation - encryption or decryption, enter the key, the message or cryptogram and will get the cryptogram or decrypted message. The final phase of adding new spaces, depending on the language used and the logic of the message – will be done manually.

PlayFair cipher:

The Playfair cipher is a symmetric encryption system based on a symmetric key: the same key is used for both encryption and decryption. This cipher is part of the category of polygraphic substitution cryptographic primitives, which involve replacing (substituting) a pair of characters from the plaintext with the ciphertext. It provides the necessary confusion for encryption but does not provide diffusion. The Playfair cipher involves the following steps:

1. Preparing the text to be encrypted.
2. Constructing the encryption matrix.
3. Constructing the encrypted message.

Preparing the text

First setp on preparing the text to be encrypted This first step involves writing all uppercase letters in pairs, without spaces and punctuation. All ‘J’ letters in the text will be replaced by ‘I’.

The next step in preparing text for encryption is to insert a letter 'Q', 'X' or 'Z' (which are the rarest letters in English vocabulary) between each double couple letters. For example, the word "FR EE DO M" in the example above will become "FR EX ED OM". Because of the three-fold repetition of the letter S between the first 2 words of the example ("CO NG RE SS SH AL") they will be rewritten as "CO NG RE SX SZ SH AL L".

The final step of preparing the text to be encrypted is to add an additional letter chosen by the person encrypting the message if there is an odd number of letters in the previous step.

Constructing the matrix

At this step, first, the repeated letters will be removed from the encryption key, starting from their second occurrence. For example, if the key is k = “dublura”, it will be transformed into “dublra”. For encrypting our message, we will use the key k = “First Amendment”, which will become “FIRST AMEND” after processing. After this, the encryption matrix is constructed. For the English alphabet with 26 letters, it can be a 5x5 matrix. In general, if the alphabet of the language in which the message to be encrypted is written has a different number of letters, the matrix can be different, for example, 6x5, 5x6, 6x4, 4x6, etc., in which all (or almost all) letters of the alphabet can be placed. If there are not enough spaces to place all the letters of the alphabet - we can proceed as shown above, that is, replace the letter J with I or eliminate the least common letter(s) in that language. During decryption, it will be intuitively restored. If the matrix has more cells than letters, free cells can be filled with any symbols. Then the matrix will be completed according to the algorithm:

- Starting from the top left corner, complete each row with the key from the previous step.
- Complete the matrix with other letters of the alphabet in alphabetical order except for letter J.

Constructing the message

Pairs of letters from the original text will be encrypted according to the following algorithm:

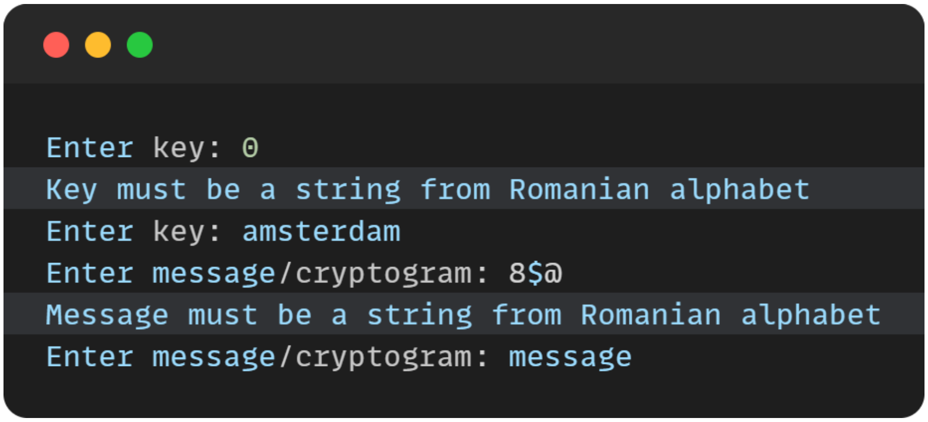
1. If the two letters are in different rows and columns, each letter will be replaced by the letter on the same row but on the column of the other letter in the current pair. For example, the pair NP will be encrypted as EQ.
2. If the two letters are on the same row of the matrix, each will be replaced by the next one on the current row; the last letter in the row will be replaced with the first letter in the same row. For example, the pair IT will be encoded as RF.
3. Similarly, if the letters are on the same column, each will be replaced by the one immediately below it on the same column; the last letter in the column will be replaced with the first letter in that column. For example, pair CW will be encoded as OI (since W is at the bottom of its column and has no other letters below it, it will be encrypted to the first letter in that column).

Decryption

To decrypt a message using the Playfair algorithm, we will reverse all the steps followed at encryption. We divide the encrypted text into pairs (we do not need the preparatory steps at the stage of encryption).

Results:

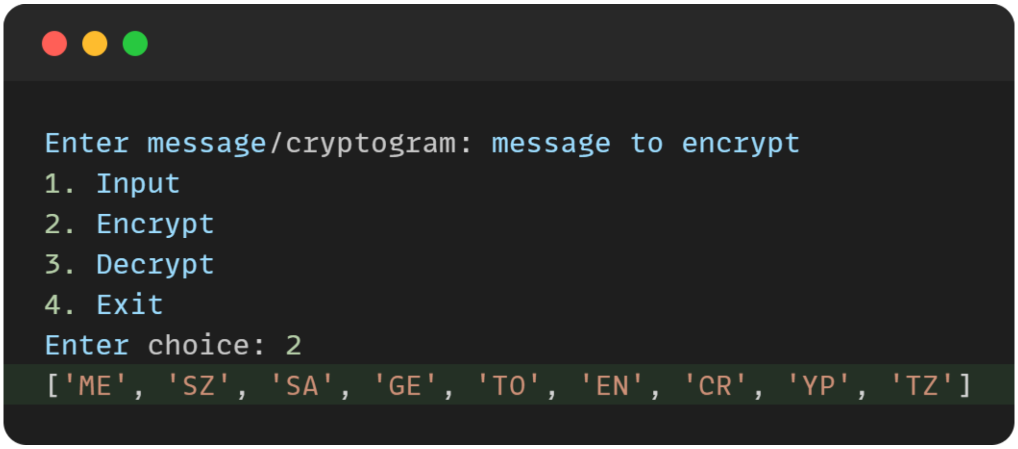
Firstly the inputs of the user are checked to be in the range nedeed. In (Figure 1) is represented how the input of the user is verified to match the given parameters that I need for my laboratory work.



```
Enter key: 0
Key must be a string from Romanian alphabet
Enter key: amsterdam
Enter message/cryptogram: 8$@
Message must be a string from Romanian alphabet
Enter message/cryptogram: message
```

Figure 1. *The user inputs validation*

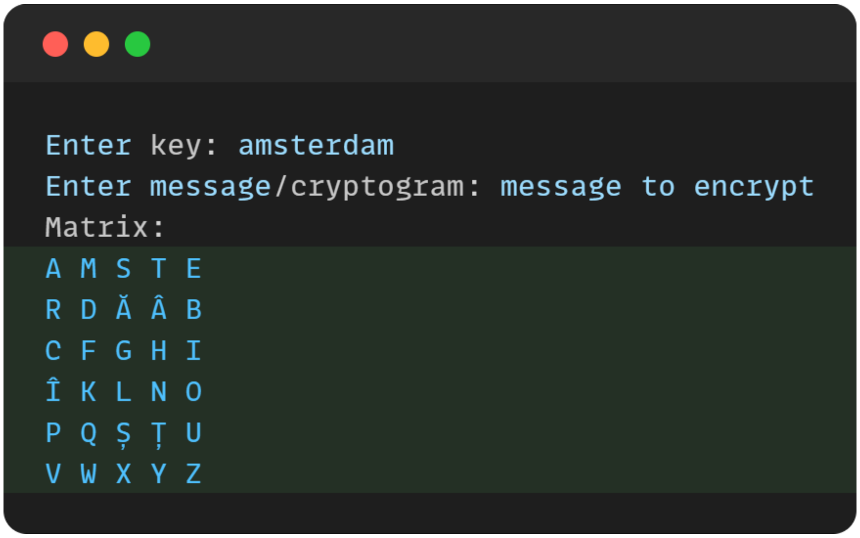
Next the text is prepared for encryption, the letter “J” is replaced with “I”, a random letter from “Q, X, Z” is inserted between each couple of repeating letters. Next the message is sliced into pairs of two, the text is made upparcase and all the spaces are removed. The final step in the preparation is adding a letter to the end of the text if it has an odd nr of characters, and this letter I chose is “Z”. In (Figure 2) is represented how the original message is transformed for encryption.



```
Enter message/cryptogram: message to encrypt
1. Input
2. Encrypt
3. Decrypt
4. Exit
Enter choice: 2
['ME', 'SZ', 'SA', 'GE', 'TO', 'EN', 'CR', 'YP', 'TZ']
```

Figure 2. *Message after preparation*

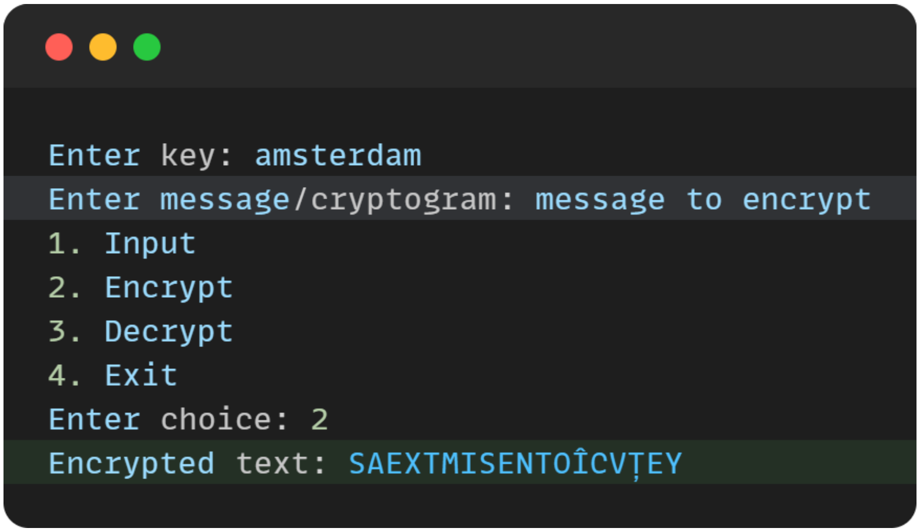
Matrix creation starts with eliminating the repeating letters in the key, next because I use Romanian alphabet I will use a 5x6 matrix, because the Romanian alphabet has 31 letters and I replaced letter “J” with “I” so I have 30 letter to work with. Next the matrix is populated first with the modified key in the first row starting from the left and continuing on the next row if necessary, the rest of the matrix is filled up with the rest of the remaining letters in the alphabet. In (Figure 3) is represented how the created matrix looks like.



```
Enter key: amsterdam
Enter message/cryptogram: message to encrypt
Matrix:
A M S T E
R D Ă Â B
C F G H I
Î K L N O
P Q Ș Ț U
V W X Y Z
```

Figure 3. *Matrix created from key*

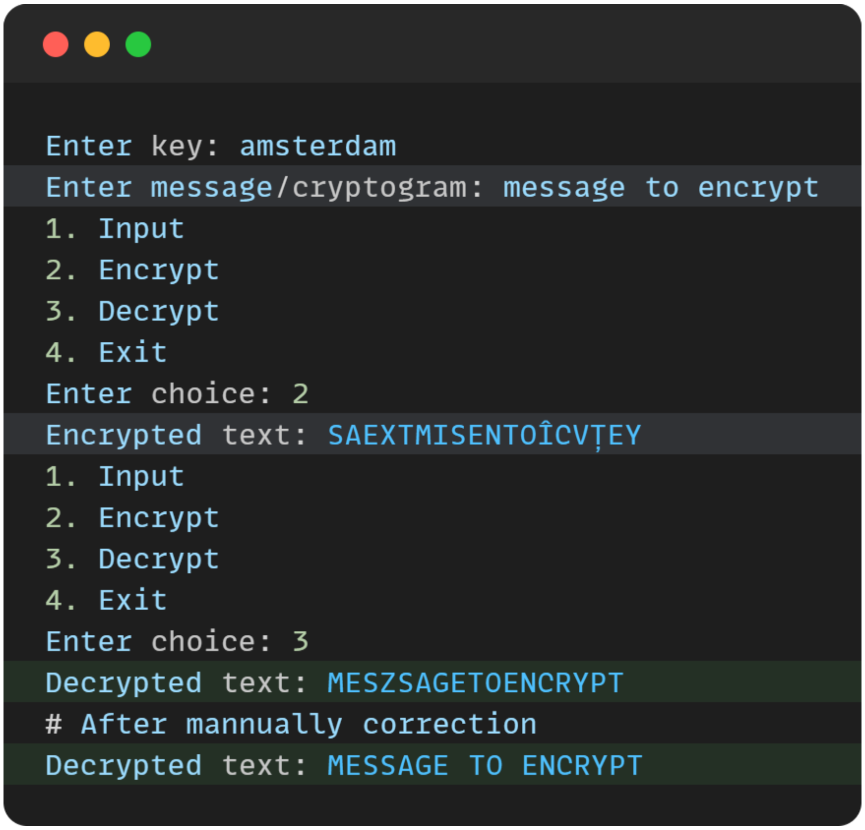
Finally the message encryption, according to the rules the pairs are replaced respectively, so each pair of words are tested by the rules and replaced with the specific letters. In (Figure 4) is demostrated the final encrypted message.



```
Enter key: amsterdam
Enter message/cryptogram: message to encrypt
1. Input
2. Encrypt
3. Decrypt
4. Exit
Enter choice: 2
Encrypted text: SAEXTMISENTOÎCVȚEY
```

Figure 4. *Encrypted message*

Now for the decryption proccess are followed the same rules as in encryption but in reverse, so in the (Figure 5) is the decrypted text.



```
Enter key: amsterdam
Enter message/cryptogram: message to encrypt
1. Input
2. Encrypt
3. Decrypt
4. Exit
Enter choice: 2
Encrypted text: SAEXTMISENTOÎCVȚEY
1. Input
2. Encrypt
3. Decrypt
4. Exit
Enter choice: 3
Decrypted text: MESZSAGETOENCRYPT
# After mannually correction
Decrypted text: MESSAGE TO ENCRYPT
```

Figure 5. *Decrypted message*

Conclusion:

In conclusion, the implementation of the Playfair cipher for the Romanian alphabet in Python has been a rewarding and educational laboratory experience. Throughout this project, we delved into the fascinating world of cryptography and encryption techniques, gaining valuable insights into the fundamental principles behind secure communication.

In addition to the technical aspects, this laboratory work reinforced the importance of cybersecurity in today's digital age. Cryptographic techniques like the Playfair cipher serve as building blocks for securing sensitive information in various applications, from online communication to data protection. Understanding and being able to implement such techniques is a valuable skill in the field of information security.

Lastly, the implementation of the Playfair cipher for the Romanian alphabet in Python was a challenging yet rewarding laboratory experience. It not only deepened our understanding of encryption and language-specific cryptography but also enhanced our programming and problem-solving capabilities. This project serves as a reminder of the crucial role that cryptography plays in safeguarding sensitive information, making us better-equipped individuals in the realm of cybersecurity.

Resources:

CS-Labs/Lab2 at main · Syn4z/CS-Labs · GitHub