Ministerul Educației, Culturii și Cercetării al Republicii Moldova
Universitatea Tehnică a Moldovei
Facultatea Calculatoare, Informatică și Microelectronică
Departamentul Ingineria Software și Automatică

Raport

pentru lucrare de laborator Nr. 3
la cursul "*Cifruri polialfabetice*"

A efectuat: Arteom KALAMAGHIN, FAF-211
A verificat: Aureliu ZGUREANU

Chișinău - 2023

Subject: Cryptanalysis of polyalphabetic ciphers

Tasks:

De implementat algoritmul Vigenere în unul din limbajele de programare pentru mesaje în limba română (31 de litere), acestea fiind codificate cu numerele 0, 1, ... 30. Valorile caracterelor textului sunt cuprinse între 'A' și 'Z', 'a' și 'z' și nu sunt premise alte valori. În cazul în care utilizatorul introduce alte valori - i se va sugera diapazonul corect al caracterelor. Lungimea cheii nu trebuie să fie mai mică de 7. Criptarea și decriptarea se va realiza în conformitate cu formulele din modelul matematic prezentat mai sus. În mesaj mai întâi trebuie eliminate spațiile, apoi toate literele se vor transforma în majuscule. Utilizatorul Va putea alege operația - criptare sau decriptare, va putea introduce cheia, mesajul sau criptograma și va obține criptograma sau mesajul decriptat.

Theoretical notes:

The Vigenère cipher is attributed to Blaise de Vigenère, a French diplomat, who described the method in the 16th century. The cipher became widely known after its usage by the court of Henry III of France for secret communication. To use the Vigenère cipher, you need a keyword, which is typically a short word or phrase. The keyword is repeated to match the length of the message. Then it is required to create a Vigenère table (or Vigenère square) where the letters of the alphabet are shifted in a different manner for each row. Each row represents a Caesar cipher with a different shift value. To encrypt a message, you align the plaintext message with the keyword. Then, for each letter in the plaintext, you find the corresponding row in the Vigenère table, based on the letter of the keyword, and locate the column that corresponds to the letter in the plaintext. The letter at the intersection of the row and column is the ciphertext letter. To decrypt the message, you use the same keyword and Vigenère table. You align the ciphertext with the keyword and find the row associated with the keyword letter. Then, you locate the column that contains the ciphertext letter and identify the plaintext letter at the intersection.

Implementation:

The biggest challenge for me was to setup the Vigenere table properly:

```
def set_map(self):
        map = []
        n = len(self.lang_alphabet)

        for i in range(n):
            row = []
            for j in range(n):
                idx = (i + j) % n
                row.append(self.lang_alphabet[idx])

            map.append(row)

        self.map = map
```

Then it is a piece of cake to find out which letters to substitute to get the cryptogram:

```
def encrypt(self):
        for l in range(len(self.msg)):
            j = self.find_letter_id(self.msg[l])
            i = self.find_letter_id(self.keys[1][l])

            self.msg = self.msg[:l] + self.map[i][j] +
self.msg[l+1:]
```

The decryption process works in the same manner.

Conclusion:

Throughout the laboratory work, I've delved into the mechanics of the cipher, utilizing a keyword and a Vigenère table to encode and decode messages. It's important to note that the Vigenère cipher, once considered secure, has shown significant vulnerabilities in the face of modern cryptanalysis methods. The security of the Vigenère cipher primarily rests on the length and randomness of the keyword. Short and predictable keywords make the cipher susceptible to attacks, including frequency analysis. However, with a long and truly random keyword, the Vigenère cipher can still offer a degree of security. Nevertheless, in today's era of powerful computers and advanced cryptographic techniques, it's clear that the Vigenère cipher is no longer suitable for secure communications and should be used primarily for educational purposes or as part of more complex encryption methods.