Министерство образования Республики Беларусь

Учреждение образования
БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Факультет компьютерного проектирования

Кафедра инженерной психологии и эргономики

КРИПТОГРАФИЧЕСКИЕ ТЕХНОЛОГИИ

Практическая работа №2

Маршрутные и подстановочные шифры

Проверил:
Давыдович К. И.

Выполнили:
Мисевич А. С.

Минск 2023

Цель работы

Изучение и криптоанализ маршрутных и подстановочных шифров.

Задача

1. Задание.

Открытый текст:

Please note that spaces and punctuation characters have been removed before encryption

Шифротекст:

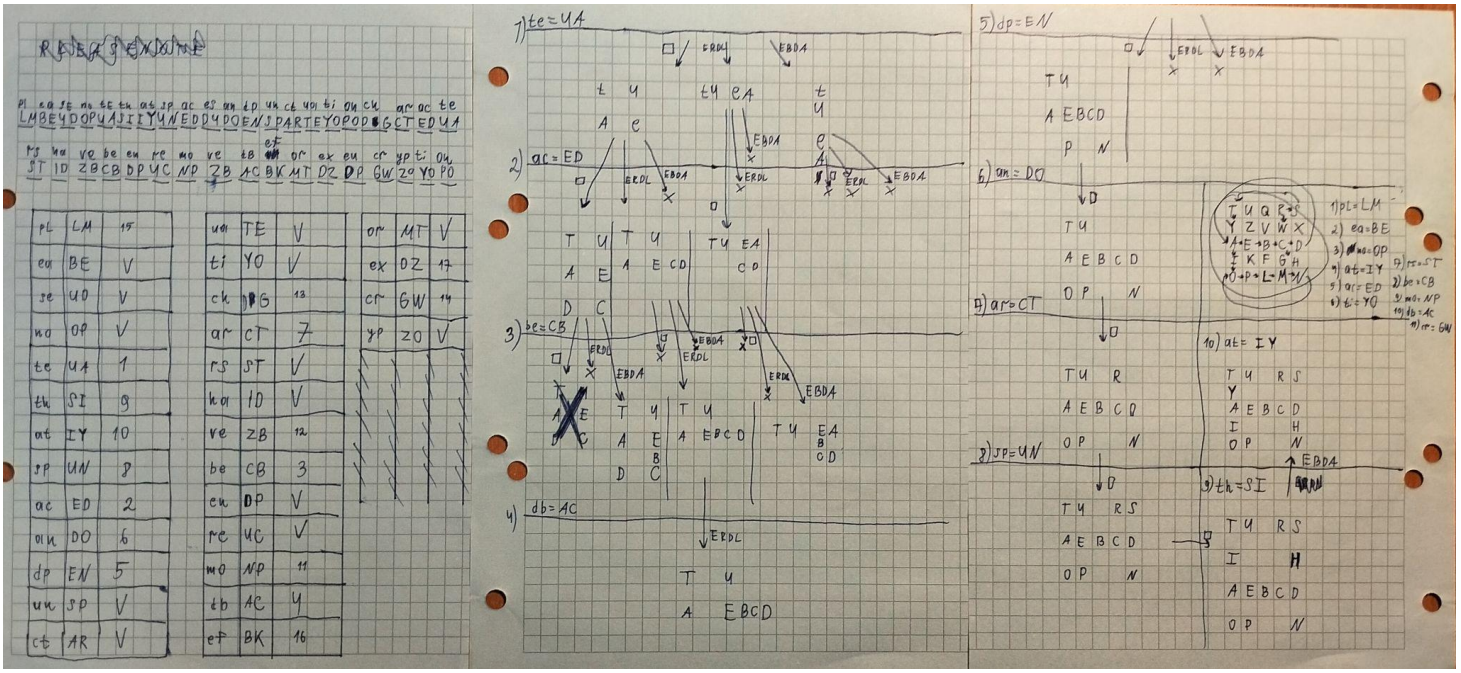
LMBEUDOPUASIIYUNEDDUDOENSPARTEYOPODGCTEDUA
STIDZBCBDPUCNPZBACBKMTDZDPGWZOYOPO

Найти ключ.

2. Задание.

- Придумать ключ.
- Написать код шифровки своих фамилии, имени, отчества
- Написать код дешифровки своих фамилии, имени, отчества

Поиск ключа



Листинг кода

```
import java.util.Scanner;

public class Main {
    static char[][] table = {{'T', 'U', 'Q', 'R', 'S'},
                             {'Y', 'Z', 'V', 'W', 'X'},
                             {'A', 'E', 'B', 'C', 'D'},
                             {'I', 'K', 'F', 'G', 'H'},
                             {'O', 'P', 'L', 'M', 'N'}};

    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        String text = "";
        String encryptedText = "";
        System.out.print("Введите текст: ");
        text = scan.nextLine();
        //Замена всех j на i, удаление всех пробелов, все
буквы подняты до верхнего регистра
        encryptedText = text.toUpperCase().replaceAll(" ",
"").replaceAll("J", "I");
        //Нахождение биграмм с одинаковыми буквами и вставка X
или Q между двумя буквами
        for (int i = 0; i < (encryptedText.length() % 2 == 0 ?
encryptedText.length() : encryptedText.length() - 1); i += 2) {
            if ((encryptedText.charAt(i) ==
encryptedText.charAt(i + 1))) {
                StringBuffer sb = new
StringBuffer(encryptedText);
                if (encryptedText.charAt(i) != 'X') {
                    encryptedText = sb.insert(i + 1,
'X').toString();
                } else {
                    encryptedText = sb.insert(i + 1,
'Q').toString();
                }
            }
        }
        //Если строка нечётной длины - вставка X или Q в конец
        if (encryptedText.length() % 2 != 0) {
            if (encryptedText.charAt(encryptedText.length() -
1) != 'X') {
                encryptedText = encryptedText + 'X';
            } else {
                encryptedText = encryptedText + 'Q';
            }
        }
        //Шифрование
        StringBuffer sb = new StringBuffer(encryptedText);
```

```
            for (int i = 0; i < encryptedText.length(); i += 2) {
                int[] index1 =
findInMatrix(encryptedText.charAt(i));
                int[] index2 = findInMatrix(encryptedText.charAt(i
+ 1));
                if (index1[0] == index2[0]) {
                    sb.setCharAt(i, table[index1[0]][(index1[1] ==
4 ? 0 : index1[1] + 1)]);
                    sb.setCharAt(i + 1,
table[index2[0]][(index2[1] == 4 ? 0 : index2[1] + 1)]);
                } else if (index1[1] == index2[1]) {
                    sb.setCharAt(i, table[(index1[0] == 4 ? 0 :
index1[0] + 1)][index1[1]]);
                    sb.setCharAt(i + 1, table[(index2[0] == 4 ? 0
: index2[0] + 1)][index2[1]]);
                } else {
                    sb.setCharAt(i, table[index1[0]][index2[1]]);
                    sb.setCharAt(i + 1,
table[index2[0]][index1[1]]);
                }
            }
            encryptedText = sb.toString();

            System.out.print("Текст после шифрования: ");
            System.out.println(encryptedText);

            //Дешифрование
            for (int i = 0; i < encryptedText.length(); i += 2) {
                int[] index1 =
findInMatrix(encryptedText.charAt(i));
                int[] index2 = findInMatrix(encryptedText.charAt(i
+ 1));
                if (index1[0] == index2[0]) {
                    sb.setCharAt(i, table[index1[0]][(index1[1] ==
0 ? 4 : index1[1] - 1)]);
                    sb.setCharAt(i + 1,
table[index2[0]][(index2[1] == 0 ? 4 : index2[1] - 1)]);
                } else if (index1[1] == index2[1]) {
                    sb.setCharAt(i, table[(index1[0] == 0 ? 4 :
index1[0] - 1)][index1[1]]);
                    sb.setCharAt(i + 1, table[(index2[0] == 0 ? 4
: index2[0] - 1)][index2[1]]);
                } else {
                    sb.setCharAt(i, table[index1[0]][index2[1]]);
                    sb.setCharAt(i + 1,
table[index2[0]][index1[1]]);
                }
            }
```

```
            encryptedText = sb.toString();

            System.out.print("Текст после дешифрования: ");
            System.out.println(encryptedText);
        }

        //Возвращает положение буквы в таблице. index[0] - строка,
index[1] - столбец
        public static int[] findInMatrix(char letter) {
            int[] index = {-1, -1};
            for (int i = 0; i < 5; i++) {
                for (int j = 0; j < 5; j++) {
                    if (table[i][j] == letter) {
                        index[0] = i;
                        index[1] = j;
                        return index;
                    }
                }
            }
            return index;
        }
    }
```

Результат работы программы

```
Введите текст: Misevich Arseniy Sergeevich
Текст после шифрования: OGUDYFDGCTUDOHXTCUKCBZGAND
Текст после дешифрования: MISEVICHARSENIYSERGEEVICHX

Process finished with exit code 0
```

Вывод

Были изучены маршрутные и подстановочные шифры.