Лабораторная работа №1

Математические основы защиты информации и информационной безопасности

Минов Кирилл Вячеславович| НПМмд-02-23

Содержание

1 Цель работы....2
2 Теоретическое введение....3
3 Выполнение лабораторной работы....4
4 Выводы....7
Список литературы....7

1 Цель работы

Реализовать на языке программирования шифр Цезаря с произвольным ключом k и шифр Атбаш.

2 Теоретическое введение

Шифр Цезаря (шифр простой замены) - это моноалфавитная подстановка, т.е. каждой букве открытого текста ставится в соответствие одна буква шифртекста. На практике при создании шифра простой замены в качестве шифроалфавита берется исходный алфавит, но с нарушенным порядком букв (алфавитная перестановка).

Математически процедуру шифрования можно описать следующим образом: $T_m = \{T^j\}, j = 0,1,\dots,m-1, T^j(a) = (a+j)mod(m)$, где $(a+j)mod(m)$ - операция нахождения остатка от целочисленного деления $a+j$ на m, а T_m - циклическая группа. Обобщение шифра Цезаря на случай произвольного ключа k для латинского алфавита: $(i+k)mod(26)$.

Шифр Атбаш является шифром сдвига на всю длину алфавита. Для реализации шифра целесообразно пользоваться таблицей ASCII и функциями работы с ней: ord и char.

3 Выполнение лабораторной работы

Первым заданием был шифр Цезаря. Переменные message и offset соответсвуют введеными с клавиатуры тексту и ключу, нужному для шифрования. result - конечное представления введенного текста. originalAlphabetPosition - переменная, в которой лежит изначальное положение элемента, newAlphabetPosition - позиция элемента после примнения шифрования. newCharacter - переменная отвечающая за хранения полученного элемента после шифровки. Весь алгоритм представляет из себя цикл в котором мы вычисляем текущее положение, новое положение и новую букву.

```java
public class CaesarCipher {
    public StringBuilder cipher(String message, int offset) {
        StringBuilder result = new StringBuilder();
        for (char character : message.toCharArray()) {
            if (character != ' ') {
                int originalAlphabetPosition = character - 'a';
                int newAlphabetPosition = (originalAlphabetPosition + offset) % 26;
                char newCharacter = (char) ('a' + newAlphabetPosition);
                result.append(newCharacter);
            } else {
                result.append(character);
            }
        }
        return result;
    }
```

Рис.1 (Реализация шифра Цезаря)

```java
    public static void main(String[] args) {
        CaesarCipher caesarCipher = new CaesarCipher();
        StringBuilder a = caesarCipher.cipher( message: "abc",  offset: 3);
        System.out.println(a);
    }
}
```

```
Run: CaesarCipher
"C:\Program Files\Java\jdk-17.0.1\bin\java.exe" "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA 2021.3\lib\idea_rt.jar=55333:C:\Program Files\Je
def

Process finished with exit code 0
```

Рис.2 (Пример работы)

Второе задание - реализация шифра Атбаш. Создание пустой строки ciphertext для хранения зашифрованного текста.

Проход по каждому символу в исходном тексте plaintext с помощью цикла for. Для каждого символа в plaintext:

Проверка, является ли символ буквой с помощью Character.isLetter(currentChar). Если символ не является буквой, он остается без изменений, и мы добавляем его к ciphertext. Если символ - буква, мы определяем, является ли она заглавной или строчной буквой, сравнивая её с символом 'A'. Если символ в верхнем регистре, baseChar устанавливается как 'A', иначе как 'a'. Мы вычисляем смещение offset текущей буквы относительно baseChar. Например, для буквы 'c' смещение составит 2 (поскольку 'c' идет после 'a' и 'b' в алфавите). Мы находим зашифрованную букву, вычитая offset из индекса текущей буквы относительно baseChar и добавляя это значение к baseChar. Например, если текущая буква - 'c', то зашифрованная буква будет 'x'. Зашифрованная буква добавляется к строке ciphertext. По завершении цикла, строка ciphertext содержит зашифрованный текст, и он возвращается как результат работы метода encrypt.

```java
public class AtbashCipher {
    public static String encrypt(String plaintext) {
        StringBuilder ciphertext = new StringBuilder();
        for (int i = 0; i < plaintext.length(); i++) {
            char currentChar = plaintext.charAt(i);
            if (Character.isLetter(currentChar)) {
                char baseChar = Character.isUpperCase(currentChar) ? 'A' : 'a';
                int offset = currentChar - baseChar;
                char encryptedChar = (char) (baseChar + (25 - offset));
                ciphertext.append(encryptedChar);
            } else {
                // Если символ не является буквой, оставляем его без изменений
                ciphertext.append(currentChar);
            }
        }
        return ciphertext.toString();
    }
```

Рис.3(Реализация шифра)

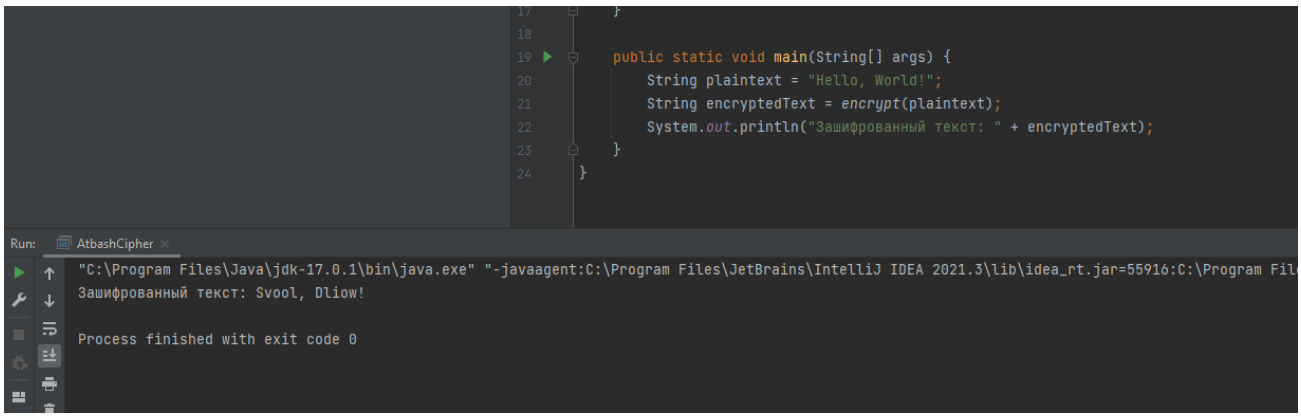

Рис.4(Пример работы)

4 Выводы

В ходе выполнения данной лабораторной работы были реализованы шифры Цезаря и Атбаш на языке программирования Java.

Список литературы

1. Шифр Цезаря [Электронный ресурс]. URL: https://ru.wikipedia.org/wiki/%D0%A8%D0%B8%D1%84%D1%80_%D0%A6%D0%B5%D0%B7%D0%B0%D1%80%D1%8F.

2. Шифр Атбаш [Электронный ресурс]. URL: https://ru.wikipedia.org/wiki/%D0%90%D1%82%D0%B1%D0%B0%D1%88