

Результат выполнения лабораторной работы №1

Цель выполнения лабораторной работы

Цель выполнения лабораторной работы

- Цель данной лабораторной работы заключается в ознакомлении с основами алгоритмов шифрования текста, а именно шифрования Цезаря и Атбаш, а также в их реализации. Целью является изучение принципов работы данных методов шифрования и их практическое применение

Теоретические сведения

Теоретические сведения. Шифр Цезаря

Шифр Цезаря — это один из простейших и наиболее известных методов шифрования текста. Он назван в честь римского полководца Гая Юлия Цезаря, который, согласно легенде, использовал этот метод для секретной переписки.

Принцип работы шифра Цезаря основан на сдвиге букв алфавита на определенное число позиций. Каждая буква заменяется на букву, идущую в алфавите после сдвига на заданное число позиций.

Пример шифрования: сдвиг алфавита на 3 позиции (shift = 3): Оригинальное сообщение: HELLO Зашифрованное сообщение: KNOOR

а б в г д е ж з и й к л м н о п р с т у ф х ц ч ш щ ъ ы ь э ю я
я ю э ь ы ъ щ ш ч ц х ф у т с р п о н м л к й и з ж е д г в б а

{width=70%}

Теоретические сведения. Шифр Атбаш

Шифр Атбаш — это шифр замены, в котором каждая буква заменяется на свою "зеркальную" букву в алфавите. Алфавит разбивается пополам, и каждая буква заменяется на соответствующую букву из противоположной половины алфавита.

Пример шифрования: Оригинальное сообщение: HELLO Зашифрованное сообщение: SV00L

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
D E F G H I J K L M N O P Q R S T U V W X Y Z A B C

{width=70%}

Результат выполнения лабораторной работы

Результат выполнения лабораторной работы

Постановка задачи:

1. Реализовать шифр Цезаря с произвольным ключом \$k\$
2. Реализовать шифр Атбаш

Результат выполнения лабораторной работы. Шифр Цезаря

Алгоритм поиска новых символов и вывода полученного текста на основе принципа формирования шифра Цезаря:

```
fun encryptCaesar(text: String, shift: Int): String {  
    val encryptedText = StringBuilder()  
    for (char in text) {  
        if (char.isLetter()) {  
            val startChar = if (char.isLowerCase()) 'a' else 'A'  
            val shiftedChar = ((char - startChar + shift) % 26 + startChar.toInt()).toChar()  
            encryptedText.append(shiftedChar)  
        } else {  
            encryptedText.append(char)  
        }  
    }  
    return encryptedText.toString()  
}
```

{width=80%}

Результат выполнения лабораторной работы. Шифр Атбаш

Принцип формирования нового алфавита для зашифровки сообщения из введенных данных:

```
fun encryptAtbash(text: String): String {  
    val encryptedText = StringBuilder()  
    for (char in text) {  
        if (char.isLetter()) {  
            val startChar = if (char.isLowerCase()) 'a' else 'A'  
            val mirroredChar = (25 - (char - startChar) + startChar.toInt()).toChar()  
            encryptedText.append(mirroredChar)  
        } else {  
            encryptedText.append(char)  
        }  
    }  
    return encryptedText.toString()  
}
```

{width=80%}

Результат выполнения лабораторной работы

Примеры работы функции по реализации шифров Цезаря и Атбаш:

A screenshot of an IDE window titled 'MainKt'. The code editor shows a Kotlin function 'main' with two print statements. The first prints the result of 'encryptCaesar' with text 'zzzz' and shift 1. The second prints the result of 'encryptAtbash' with text 'bbbbbbB'. Below the code, the output console shows the path to the IDE, followed by 'aaaa' and 'yyyyyyY'.

```
fun main() {  
    println(encryptCaesar(text: "zzzz", shift: 1))  
  
    println(encryptAtbash(text: "bbbbbbB"))  
}
```

"D:\Intelli J\IntelliJ IDEA Community Edition 2022.1.
aaaa
yyyyyyY

{width=80%}

Выводы

Выводы

1. Изучили шифры простой замены
2. Реализовали шифр Цезаря с произвольным ключом \$k\$
3. Реализовали шифр Атбаш