Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение высшего образования

«Рязанский государственный радиотехнический

университет имени В.Ф. Уткина»

Кафедра «ЭВМ»

Отчет о лабораторной работе №1

«Криптоанализ шифра простой замены»

по дисциплине

«Защита информации»

Выполнили:

Студенты группы 045

Анохин В.А.

Вашкулатов Н.А.

Проверили:

доц. Крошилина С.В.

доц. Тишкина В.В.

**Цель работы**: выполнить криптоанализ текста, зашифрованного шифром простой замены с применением непереборного метода вскрытия шифротекста.

**Ход работы**

На рисунках 1 – 4 показаны схемы алгоритмов данной программы.

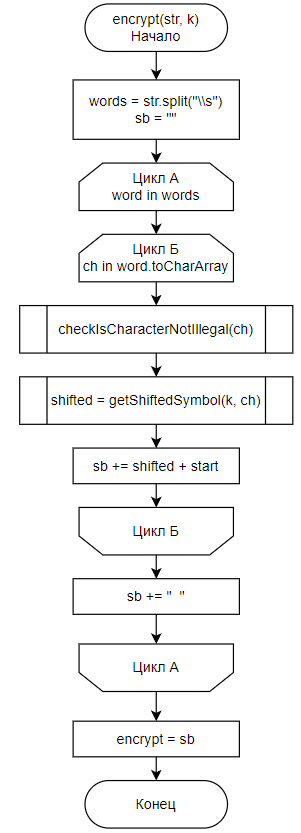


Рисунок 1 – Подпрограмма шифрации текста

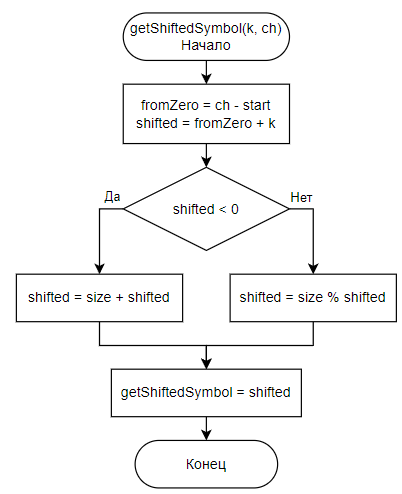


Рисунок 2 – Получение сдвинутого символа

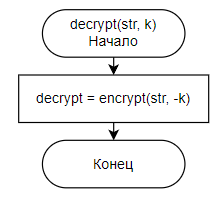


Рисунок 3 – Подпрограмма дешифрации текста

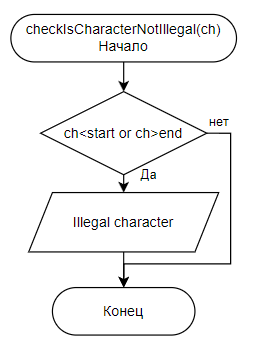


Рисунок 4 – Проверка корректности символа

**Код программы:**

public final class Caesar {

private final char start;

private final char end;

private final int size;

// Алфавит задается начальным и конечным символом таблицы unicode

public Caesar(char start, char end) {

this.start = start;

this.end = end;

// Количество символов в массиве

this.size = (char) (end - start + 1);

}

public String encrypt(String str, int k) {

// Разбивка на слова по пробелу т.к. пробел не сдвигается

var words = str.split("\\s");

var sb = new StringBuilder();

//Обход всех слов

for (String word : words) {

// Каждого символа в строке

for (Character ch : word.toCharArray()) {

//Проверка на то, что символ в алфавите

checkIsCharacterNotIllegal(ch);

//Получаем сдвинутый символ

var shifted = getShiftedSymbol(k, ch);

//Добавляем символ в результат

sb.append((char) (shifted + start));

}

//Пробел между словами

sb.append(" ");

}

return sb.toString().trim();

}

private int getShiftedSymbol(int k, Character ch) {

//Сдвигаем алфавит в таблице, чтобы отсчет алфавита был с 0

var fromZero = ch - start;

var shifted = fromZero + k;

//Нормализуем, чтобы оставить символ в пределах алфавита (циклически)

if (shifted < 0)

shifted = size + shifted;

else

shifted = shifted % size; //mod

return shifted;

}

private void checkIsCharacterNotIllegal(Character ch) {

if (ch < start || ch > end)

throw new IllegalArgumentException("String cotains illegal character " + ch);

}

public String decrypt(String encrypted, int k) {

// Для расшифровки - шифровка с обратным сдвигом

return encrypt(encrypted, -k);

}

public List<String> tryDecrypt(String encrypted, int minShift, int maxShift) {

var result = new ArrayList<String>();

//Перебор всех вариантов в пределах от minShift до maxShift

for (int i = minShift; i <= maxShift; i++) {

result.add(decrypt(encrypted, i));

}

return result;

}

public List<String> tryDecrypt(String encrypted) {

return tryDecrypt(encrypted, 0, size - 1);

}

public char getStart() {

return start;

}

public char getEnd() {

return end;

}

public int getSize() {

return size;

}

}

**Вывод**: в ходе выполнения работы был выполнен криптоанализ текста, зашифрованного шифром простой замены с применением непереборного метода вскрытия шифротекста.