Отчёт по лабораторной работе №8

Элементы криптографии. Шифрование (кодирование) различных исходных текстов одним ключом

Венчаков Никита НБИбд-01-19"

Содержание

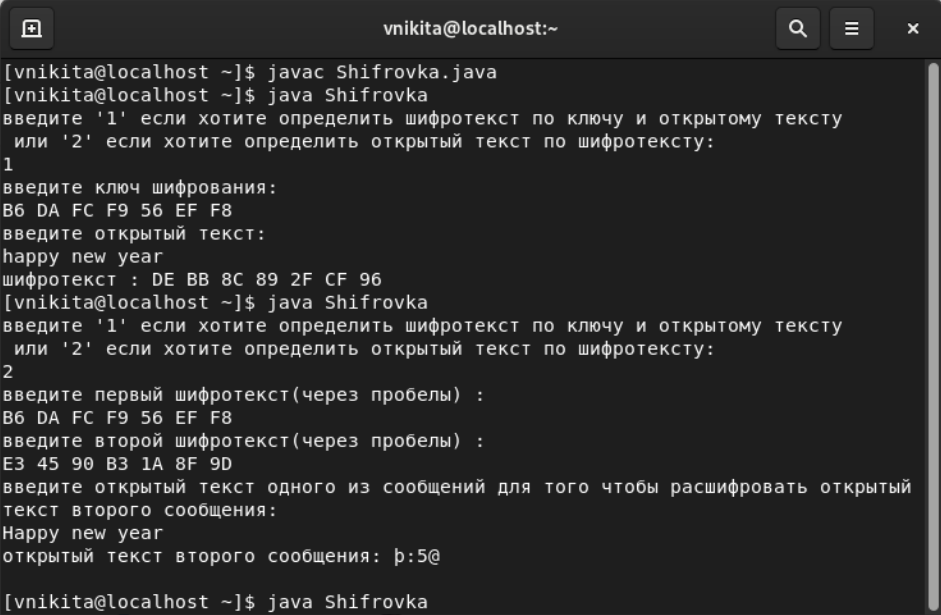
# Цель работы

Освоить на практике применение режима однократного гаммирования на примере кодирования различных исходных текстов одним ключом.

# Выполнение лабораторной работы

1. Создал программу на Java для гаммирования текста. Текст программы:

import java.util.HashMap;  
import java.util.Iterator;  
import java.util.Map;  
import java.util.Scanner;  
public class Shifrovka {  
public static void main(String [] args) {  
 HashMap<Character, String> map = new HashMap<Character ,String>();  
 map.put('0', "0000");  
 map.put('1',"0001");  
 map.put('2',"0010");  
 map.put('3', "0011");  
 map.put('4', "0100");  
 map.put('5',"0101");  
 map.put('6',"0110");  
 map.put('7',"0111");  
 map.put('8',"1000");  
 map.put('9', "1001");  
 map.put('A', "1010");  
 map.put('B',"1011" );  
 map.put('C', "1100");  
 map.put('D', "1101");  
 map.put('E',"1110" );  
 map.put('F', "1111");  
 //System.out.println(shifrovanie("14 15 15 ","41 43 42",map));  
 String text="";  
 String cipher;  
 String cipher2;   
 Scanner in = new Scanner(System.in);  
 System.out.println("введите '1' если хотите определить шифротекст по ключу и открытому тексту \n или '2' если хотите определить открытый текст по шифротексту: ");  
 int input = in.nextInt();  
 if(input==1) {  
 Scanner in2 = new Scanner(System.in);  
 System.out.println("введите ключ шифрования: ");   
 cipher= in2.nextLine();  
 System.out.println("введите открытый текст: ");  
 cipher2 = in2.nextLine();  
 cipher2= characterto16(cipher2,map);  
 String shifr = shifrovanie(cipher,cipher2,map);  
 System.out.println("шифротекст : "+shifr);  
   
 }else {  
 Scanner in2 = new Scanner(System.in);  
 System.out.println("введите первый шифротекст(через пробелы) : ");  
 cipher= in2.nextLine();  
 System.out.println("введите второй шифротекст(через пробелы) : ");  
 cipher2= in2.nextLine();  
 System.out.println("введите открытый текст одного из сообщений для того чтобы расшифровать открытый текст второго сообщения:");  
 text =in2.nextLine();  
 String C1xorC2= shifrovanie(cipher,cipher2,map);  
 String cipher16=characterto16(text,map);  
 String result = shifrovanie(C1xorC2,cipher16,map);  
 System.out.println("открытый текст второго сообщения: "+tocharacter(result,map));   
 }  
}  
public static String characterto16 (String cipher,HashMap<Character, String> map) {  
 char[] chararray = cipher.toCharArray();  
 String finalcode="";  
 for(int i=0;i<chararray.length;i++) {  
 char character = chararray[i];  
 int ascii = (int) character;  
 String code = Integer.toString(ascii,2);  
 String curcode=code;  
 for(int j=0;j<8-code.length();j++) {  
 curcode="0"+curcode;  
 }  
 code= curcode;  
 String val = code.substring(0, 4);  
 String val2= code.substring(4);  
 char nval=' ';  
 char nval2=' ';  
 Iterator it = map.entrySet().iterator();  
 while (it.hasNext()) {  
 Map.Entry pair = (Map.Entry)it.next();  
 if(pair.getValue().equals(val)) {  
 nval=(char)pair.getKey();  
 }  
 if(pair.getValue().equals(val2)) {  
 nval2=(char)pair.getKey();  
 }  
 }  
 String v = String.valueOf(nval)+String.valueOf(nval2);  
 finalcode=finalcode+v+" ";   
 }  
 return finalcode;  
}  
public static String tocharacter(String cipher, HashMap<Character, String> map) {  
 String[] splt = cipher.split("\\s+");  
 String finalcode="";  
 for(int i=0;i<splt.length;i++) {  
 char[] symbols = splt[i].toCharArray();  
 String symbol = map.get(symbols[0])+map.get(symbols[1]);  
 int number = Integer.parseInt(symbol, 2);  
 finalcode+=Character.toString ((char) number);  
}  
 return finalcode;  
 }  
public static String shifrovanie(String cipher, String cipher2,HashMap<Character, String> map) {  
   
   
 String[] splt = cipher.split("\\s+");  
 String[] splt2 = cipher2.split("\\s+");  
 String finalcode="";  
 for(int i=0;i<splt.length;i++) {   
 char[] symbols = splt[i].toCharArray();  
 String symbol = map.get(symbols[0])+map.get(symbols[1]);  
 char[] symbols2 = splt2[i].toCharArray();  
 String symbol2 = map.get(symbols2[0])+map.get(symbols2[1]);  
 String newsymbol="";  
 for(int j=0;j<symbol2.length();j++) {  
 int number= Character.digit(symbol2.charAt(j), 10);  
 int number2 = Character.digit(symbol.charAt(j), 10);  
 newsymbol+=number^number2;  
 }  
 String val = newsymbol.substring(0, 4);  
 String val2= newsymbol.substring(4);  
 char nval=' ';  
 char nval2=' ';  
 Iterator it = map.entrySet().iterator();  
 while (it.hasNext()) {  
 Map.Entry pair = (Map.Entry)it.next();  
 if(pair.getValue().equals(val)) {  
 nval=(char)pair.getKey();  
 }  
 if(pair.getValue().equals(val2)) {  
 nval2=(char)pair.getKey();  
 }  
 }  
 String v = String.valueOf(nval)+String.valueOf(nval2);  
 finalcode=finalcode+v+" ";   
 }  
 return finalcode;  
}  
}

1. Скомпилировал программу с помощью команды javac Shivrofka.java
2. Запустил программу с помощью команды java Shifrovka
3. Написал первый шифротекст: B6 DA FC F9 56 EF F8  
   Написал второй шифротекст: E3 45 90 B3 1A 8F 9D  
   Написал текст одного из сообщений: Happy new year  
   В итоге получил текст второго зашифрованного сообщения([рис.№1](Скриншоты выполнения\Пункт 1.png))  
   

# Вывод

Освоил на практике применение режима однократного гаммирования на примере кодирования различных исходных текстов одним ключом.