Министерство образования Республики Беларусь

Учреждение образования

БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

КАФЕДРА ИНФОРМАТИКИ

Лабораторная работа № 1

**Компьютерная реализация шифра Цезаря и шифра Виженера**

Выполнил студент гр. 853505

Лазарева Е.В.

Проверил

Протько М.И.

Минск, 2021

1. **Введение**

Целью данной лабораторной работы было изучить алгоритмы шифра Цезаря и шифра Виженера и реализовать программные средства шифрования и дешифрования текстовых файлов при их помощи.

1. **Блок-схемы алгоритмов**

Схема алгоритма шифра Цезаря:

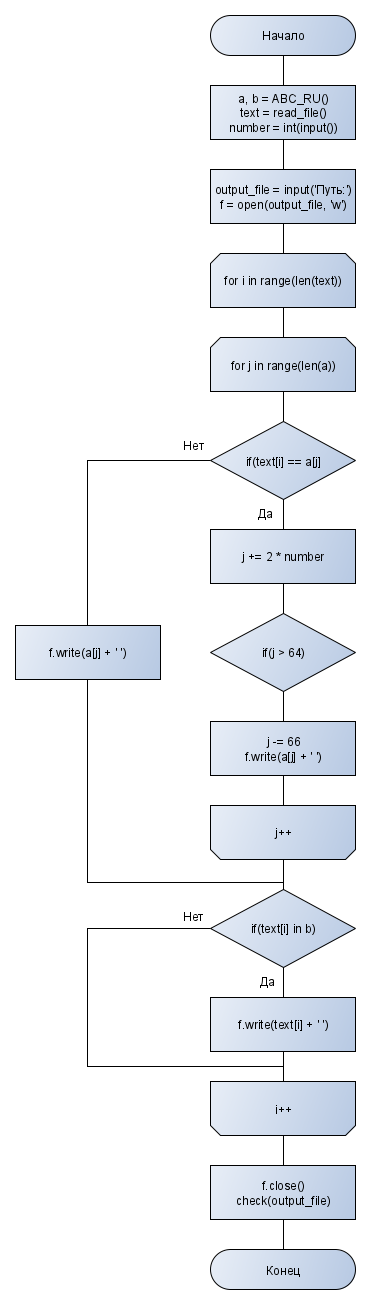
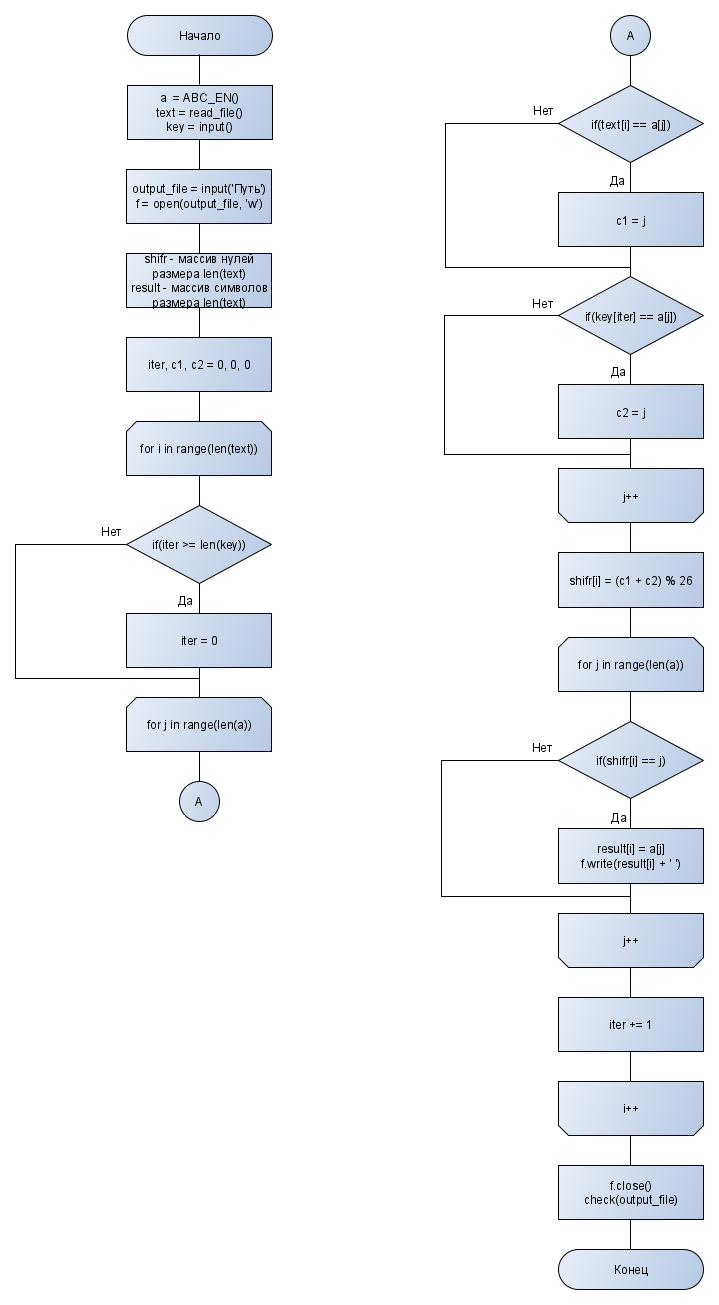


Схема алгоритма шифра Виженера:



1. **Теоретические сведения**

**Шифр Цезаря**

Шифр Цезаря, также известный, как шифр сдвига, код Цезаря или сдвиг Цезаря – один из самых простых и наиболее широко известных методов шифрования.

Шифр Цезаря – это вид шифра подстановки, в котором каждый символ в открытом тексте заменяется символом находящимся на некотором постоянном числе позиций левее или правее него в алфавите. Например, в шифре со сдвигом 4 А была бы заменена на Г, Б станет Д, и так далее.

Шифр назван в честь римского императора Гая Юлия Цезаря, использовавшего его для секретной переписки со своими генералами.

Шаг шифрования, выполняемый шифром Цезаря, часто включается как часть более сложных схем, таких как шифр Виженера, и все ещё имеет современное приложение в системе ROT13. Как и все моноалфавитные шифры, шифр Цезаря легко взламывается и не имеет практически никакого применения на практике.



Рисунок 1. Шифр Цезаря

Если сопоставить каждому символу алфавита его порядковый номер (нумеруя с 0), то шифрование и дешифрование можно выразить формулами модульной арифметики:

y=(x+k)\ \mod\ n

x=(y-k+n)\ \mod\ n,

где ~x – символ открытого текста, ~y – символ шифрованного текста, ~n – мощность алфавита, а ~k – ключ.

С точки зрения математики шифр Цезаря является частным случаем аффинного шифра.

**Шифр Виженера**

Шифр Виженера состоит из последовательности нескольких шифров Цезаря с различными значениями сдвига. Для зашифровывания может использоваться таблица алфавитов, называемая tabula recta или квадрат (таблица) Виженера. Применительно к латинскому алфавиту таблица Виженера составляется из строк по 26 символов, причём каждая следующая строка сдвигается на несколько позиций. Таким образом, в таблице получается 26 различных шифров Цезаря. На каждом этапе шифрования используются различные алфавиты, выбираемые в зависимости от символа ключевого слова. Например, предположим, что исходный текст имеет такой вид:

ATTACKATDAWN

Человек, посылающий сообщение, записывает ключевое слово («LEMON») циклически до тех пор, пока его длина не будет соответствовать длине исходного текста:

LEMONLEMONLE

Первый символ исходного текста A зашифрован последовательностью L, которая является первым символом ключа. Первый символ L шифрованного текста находится на пересечении строки L и столбца A в таблице Виженера. Точно так же для второго символа исходного текста используется второй символ ключа; то есть второй символ шифрованного текста X получается на пересечении строки E и столбца T. Остальная часть исходного текста шифруется подобным способом.

Исходный текст: ATTACKATDAWN

Ключ: LEMONLEMONLE

Зашифрованный текст: LXFOPVEFRNHR

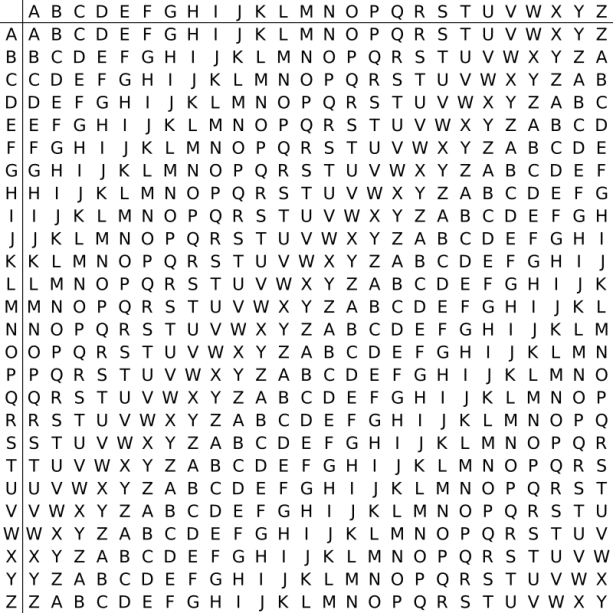


Рисунок 2. Квадрат Виженера

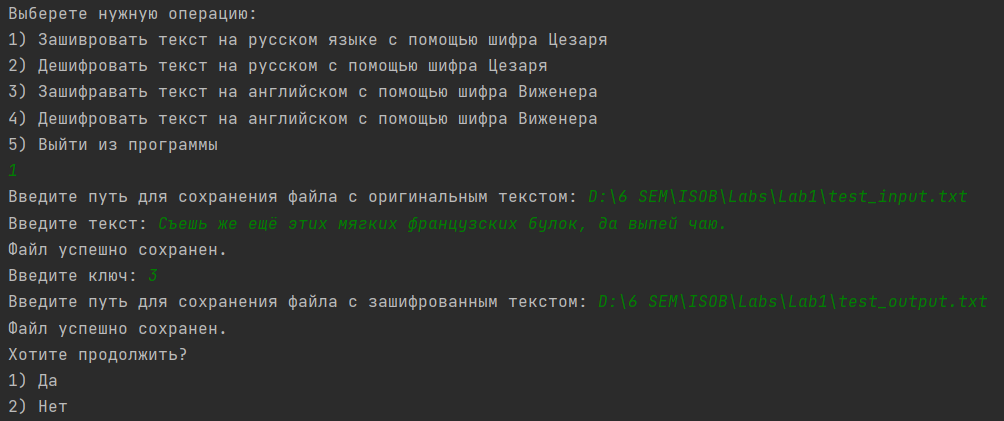
Если *n* – количество букв в алфавите, *mj* – буквы открытого текста, *kj* – буквы ключа, то шифрование Виженера можно записать следующим образом:

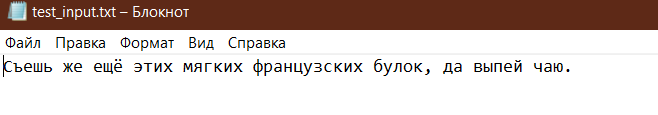
*cj* = (*mj* + *kj*) mod *n*

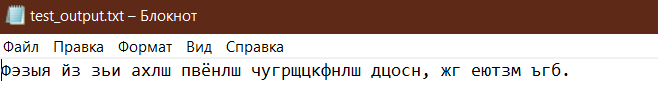
И расшифрование:

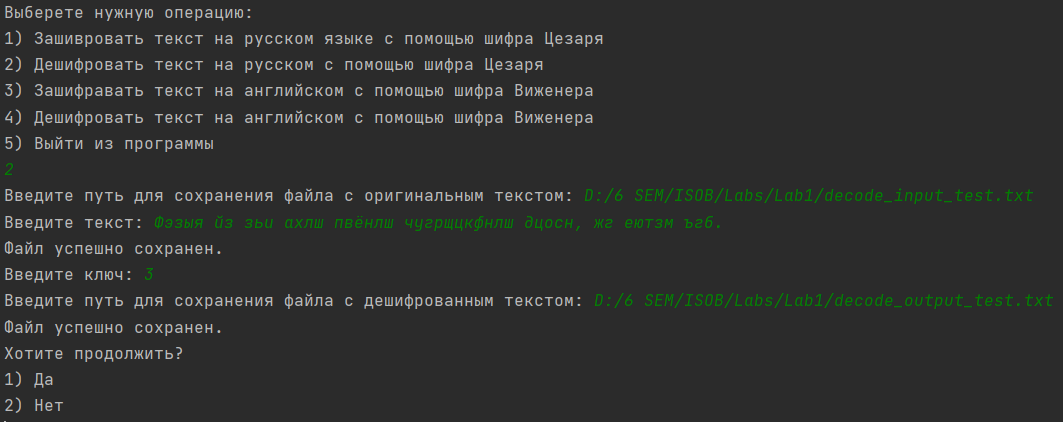
*mj* = (*cj* + *n* - *kj*) mod *n*

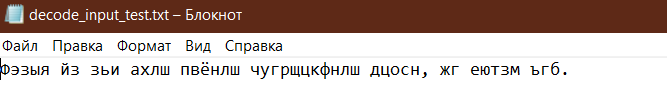
1. **Результат работы программы**

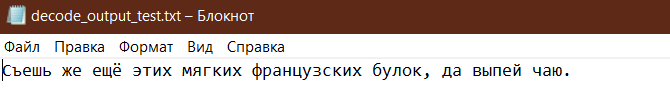


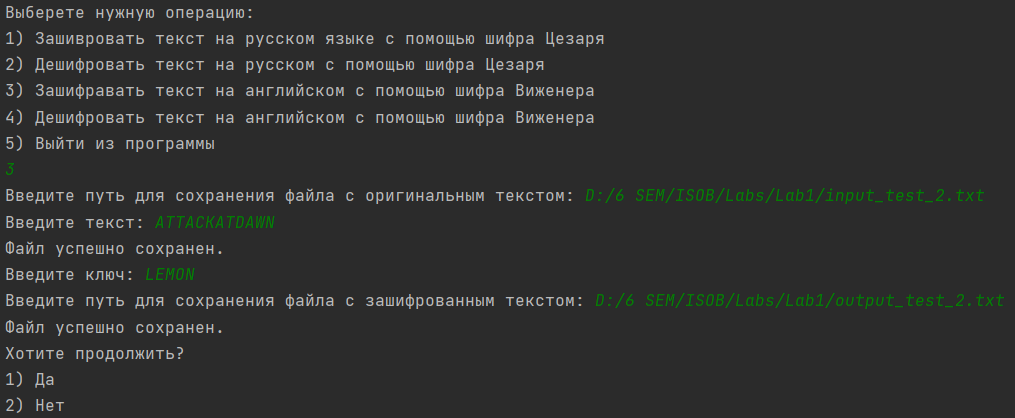






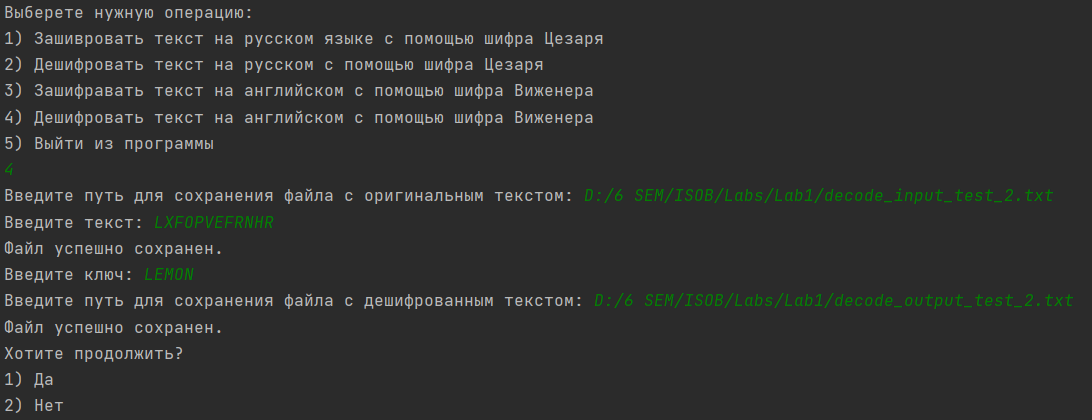
















1. **Код программы**

import os  
import numpy as np  
  
def check (name):  
 if (os.path.exists(name)):  
 print('Файл успешно сохранен.')  
 else:  
 print('ОШИБКА! Не удалось сохранить файл.')  
 return 0  
  
def ABC\_RU():  
 a = 'АаБбВвГгДдЕеЁёЖжЗзИиЙйКкЛлМмНнОоПпРрСсТтУуФфХхЦцЧчШшЩщЪъЫыЬьЭэЮюЯя'  
 b = '1234567890@#–—$\_&-+()/\*":;!?,.~`|•√π÷×¶∆£¢€¥^°={}\%©®™✓[]<> '  
 return a, b  
  
def ABC\_EN():  
 a = ['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']  
 return a  
  
def simple\_choice():  
 flag = True  
 while (flag == True):  
 print('Хотите продолжить?\n'  
 '1) Да\n'  
 '2) Нет')  
 choice = int(input())  
 if (choice == 1):  
 flag = False  
 menu()  
 elif (choice == 2):  
 exit(0)  
 else:  
 print('Выбрана неверная операция! Повторите попытку!\n')  
  
def menu ():  
 flag = True  
 while (flag == True):  
 print('Выберете нужную операцию:\n'  
 '1) Зашивровать текст на русском языке с помощью шифра Цезаря\n'  
 '2) Дешифровать текст на русском с помощью шифра Цезаря\n'  
 '3) Зашифравать текст на английском с помощью шифра Виженера\n'  
 '4) Дешифровать текст на английском с помощью шифра Виженера\n'  
 '5) Выйти из программы')  
 choice = int(input())  
 if (choice == 1):  
 flag = False  
 task1(choice)  
 elif (choice == 2):  
 task1(choice)  
 flag = False  
 elif (choice == 3):  
 task2(choice)  
 flag = False  
 elif (choice == 4):  
 task2(choice)  
 flag = False  
 elif (choice == 5):  
 exit(0)  
 else:  
 print('Выбрана неверная операция! Повторите попытку!\n')  
  
def read\_file ():  
 input\_file = input('Введите путь для сохранения файла с оригинальным текстом: ')  
 f = open(input\_file, 'w')  
 f.write(input('Введите текст: '))  
 f.close()  
 check(input\_file)  
 with open(input\_file) as f:  
 shifr = f.read()  
 f.close()  
 return shifr  
  
def Caesar (text, number, a, b):  
 output\_file = input('Введите путь для сохранения файла с зашифрованным текстом: ')  
 f = open(output\_file, 'w')  
 for i in range(len(text)):  
 for j in range(len(a)):  
 if (text[i] == a[j]):  
 j += 2 \* number  
 if (j > 64):  
 j -= 66  
 f.write(a[j] + '')  
 else:  
 f.write(a[j] + '')  
 if (text[i] in b):  
 f.write(text[i] + '')  
 f.close()  
 check(output\_file)  
  
def Caesar2 (text, number, a, b):  
 output\_file = input('Введите путь для сохранения файла с дешифрованным текстом: ')  
 f = open(output\_file, 'w')  
 for i in range(len(text)):  
 for j in range(len(a)):  
 if (text[i] == a[j]):  
 j -= 2 \* number  
 if (j > 64):  
 j = j - 66  
 f.write(a[j] + '')  
 else:  
 f.write(a[j] + '')  
 else:  
 if (text[i] in b):  
 f.write(text[i] + '')  
 f.close()  
 check(output\_file)  
  
def Vigener (text, key, choice, a):  
 if (choice == 3):  
 output\_file = input('Введите путь для сохранения файла с зашифрованным текстом: ')  
 elif (choice == 4):  
 output\_file = input('Введите путь для сохранения файла с дешифрованным текстом: ')  
 f = open(output\_file, 'w')  
 shifr = np.array([0 for i in range(len(text))])  
 result = np.array(['' for i in range(len(text))])  
 iter, c1, c2 = 0, 0, 0  
 for i in range(len(text)):  
 if (iter >= len(key)):  
 iter = 0  
 for j in range(len(a)):  
 if (text[i] == a[j]):  
 c1 = j  
 if (key[iter] == a[j]):  
 c2 = j  
 if (choice == 3):  
 shifr[i] = (c1 + c2) % 26  
 elif (choice == 4):  
 shifr[i] = (c1 - c2 + 26) % 26  
 for j in range(len(a)):  
 if (shifr[i] == j):  
 result[i] = a[j]  
 f.write(result[i] + '')  
 iter += 1  
 f.close()  
 check(output\_file)  
  
def task1 (choice):  
 a, b = ABC\_RU()  
 text = read\_file()  
 number = int(input('Введите ключ: '))  
 if (choice == 1):  
 Caesar(text, number, a, b)  
 elif (choice == 2):  
 Caesar2(text, number, a, b)  
 simple\_choice()  
  
def task2 (choice):  
 a = ABC\_EN()  
 text = read\_file()  
 key = input('Введите ключ: ')  
 Vigener(text, key, choice, a)  
 simple\_choice()  
if \_\_name\_\_ == '\_\_main\_\_':  
 menu()

1. **Вывод**

В этой лабораторной работе были изучены алгоритмы шифрования с помощью шифра Цезаря и шифра Виженера. Шифр Виженера является более сложным для взлома, поскольку ключом является не целое число, а комбинация символов.