МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ

РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ

УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«МОСКОВСКИЙ АВИАЦИОННЫЙ ИНСТИТУТ

(национальный исследовательский университет)»

Институт №3 «Системы управления, информатика и электроэнергетика»

Кафедра 311 «Прикладные программные средства и математические методы»

**Лабораторная работа №4**

по дисциплине:

«Информационная безопасность»

на тему:

**«**Взломшифра Виженера»

Работу выполнил:

студент гр. М3О-318Бк-19

Артиков Темур Улугбекович

Научный руководитель:

ассистент, Кос О.И.

Дата выдачи задания: 18.09.2021 .

Дата сдачи: 27.11.2021 .

.

Москва 2021 г.

Оглавление:

Глава 1: Введение………………………………………………………………….

Глава 2: Алгоритм для реализации……………………………………………….

Глава 3: Код программы…………………………………………………………..

Глава 4: Демонстрация работы……………………………………………….......

Выводы………...……………………………………………………………………

Список литературы…………………………………………………………………

**Глава 1: Введение**

**Цель работы:** программирование взлом шифра Виженера (шифрование, дешифрование)

Данная лабораторная работа была выполнена с помощью языка программирования Python в среде программирования

**Глава 2: Алгоритм для реализации**

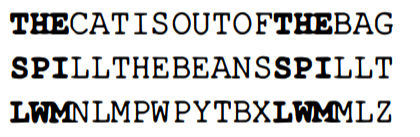
Существуют два метода взлома шифра Виженера. Один из них - перебор по словарю, или словарная атака (dictionary attack), предполагающий полный перебор всех слов из файла словаря в качестве ключа. Это сработает только в том случае, если искомый ключ является английским словом, например "RAVEN" или "DESK". Второй, более сложный метод, открытый в XIX веке математиком Чарльзом Бэббиджем, дает результат, даже если ключ представляет собой случайный набор букв, такой как "VUWFE" или "PNFJ". Для реализации взлома Виженера необходимо знать длину ключа.

**Использование метода Касиски для определения длины ключа**

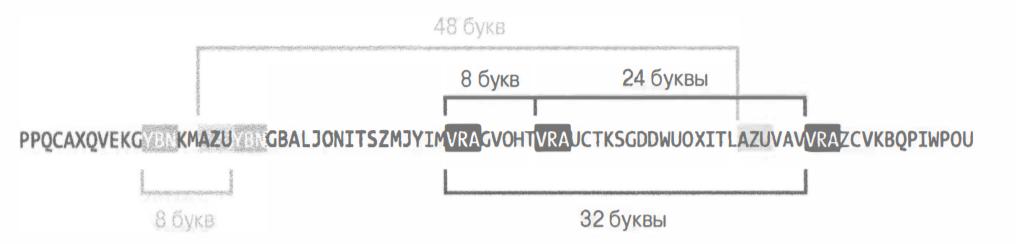
Метод Касиски предназначен для определения длины ключа шифра Виженера. Это дает возможность применить частотный анализ для независимого взлома каждого из подключей. Первым человеком, которому удалось взломать шифр Виженера, был Чарльз Бэббидж, но об этом никто не узнал, так как он не опубликовал свои результаты. Позже аналогичное открытие сделал немецкий математик Фридрих Касиски, в честь которого и был назван метод. Рассмотрим этапы алгоритма, которые будут реализованы нашей программой взлома шифра Виженера.

**Нахождение повторяющихся сегментов**

В соответствии с методом Касиски первое, что необходимо сделать, - это найти в шифротексте повторяющиеся сегменты, содержащие по крайней мере три буквы. Такие последовательности могут быть повторениями одних и тех же наборов символов в исходном тексте, зашифрованных с помощью одних и тех же подключей. Например, если зашифровать открытый текст "ТНЕ САТ IS OUT OF ТНЕ BAG" с помощью ключа "SPILLTHEBEANS", то получим следующее:



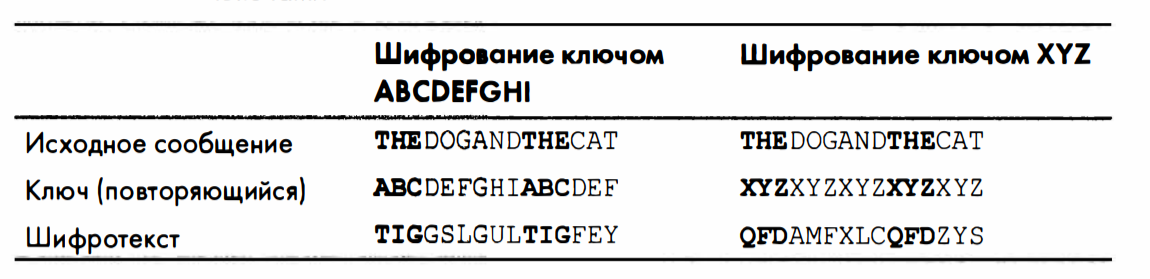
Сочетание "LWМ" повторяется дважды. Причина заключается в том, что группа букв "LWМ" в шифротексте — это слово "ТНЕ", зашифрованное с помощью одних и тех же букв ключа, "SPI", поскольку так получилось, что ключ повторяется на втором слове "ТНЕ". Количество букв от начала первого буквосочетания "LWM" до начала второго такого же буквосочетания, которое мы будем называть интервалом повторения, равно 1 3. Это дает основания полагать, что длина ключа, применяемого в данном шифротексте, составляет 13 букв. Таким образом, мы выявили длину ключа путем простого анализа повторяющихся последовательностей. Однако в большинстве шифротекстов ключ не будет удобным образом выравниваться по повторяющимся последовательностям букв, или же ключ может встречаться между повторяющимися последовательностями более одного раза, т.е. интервал повторения будет кратным длине ключа, а не равным ей. Чтобы попытаться найти пути решения этой проблемы, рассмотрим более длинный пример, в котором нам не известно, что собой представляет ключ. Если удалить из шифротекста "PPQCA XQVEKGYBNKMAZU YВNGBAL JON 1 TSZM JYIM. VRA (; VOHT VRAU С TКSG.DDWUO XITLAZU VAW RAZ С VКВ QP IWPOU" все небуквенные символы, то он примет вид строки. На этом рисунке также выделены повторяющиеся последовательности - "VM' , "AZU" и "YBN" - и указано количество букв между каждой парой последовательностей.



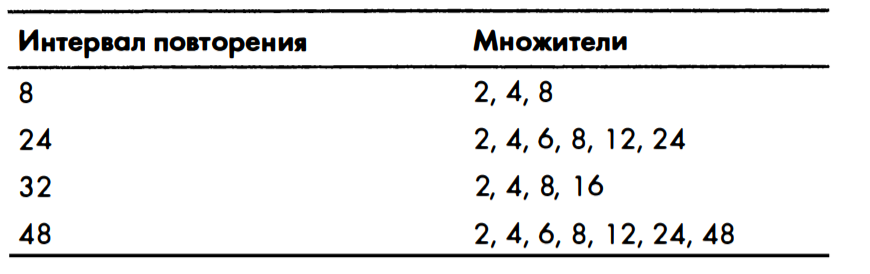
В этом примере существует несколько возможных вариантов длины ключа. Следующий шаг метода Касиски заключается в вычислении множителей всех интервалов повторения для сужения круга возможных вариантов.

**Определение множителей в интервалах повторения**

В нашем случае промежутки между последовательностями, т.е. интервалы повторения, равны 8, 8, 24, 32 и 48. Но для нас важны не они, а их множители. Чтобы понять, почему это так, рассмотрим сообщение "THEDOGANDTHECAT" и попытаемся зашифровать его с помощью девяти-буквенного ключа ABCПEFGHI, а также трехбуквенного ключа XYZ. Каждый из этих ключей повторяется на протяжении всей длины сообщения.



Как и ожидалось, два ключа приводят к двум разным шифротекстам. Разумеется, ни исходное сообщение, ни ключ не будут известны взломщику шифра, но он заметит, что в шифротексте "TIGGSLGULПGFEY" последовательность "TIG" встречается при значениях индекса О и 9. Поскольку 9 – О = 9, интервал повторения для данной последовательности равен 9, а это может свидетельствовать о том, что истинный ключ является девятибуквенным, и в рассматриваемом случае он именно такой. Однако и в шифротексте "QFDAМFXLCQFDZYS" имеется своя повторяющаяся последовательность, "QFD", которая встречается при значениях индекса О и 9. Интервал повторения здесь тоже равен 9, и это, на первый взгляд, указывает на то, что ключ, использованный для получения данного шифротекста, насчитывает девять букв. Однако мы знаем, что в действительности ключ содержит всего три буквы: "XYZ". Повторяющиеся последовательности встречаются тогда, когда одни и те же буквы в исходном сообщении ("ТНЕ" в нашем примере) шифруются одними и теми же буквами ключа (в данном примере это "АВС" и "XYZ"), что происходит в том случае, если аналогичные группы в сообщении и ключе "выравниваются" и шифруются в одну и ту же последовательность. Подобное выравнивание может встречаться в позициях, кратных длине реального ключа (таких, как 3, 6, 9, 12 и т.д.), и именно поэтому трехбуквенный ключ может порождать последовательности с интервалом повторения, равным 9. Таким образом, возможная длина ключа может быть обусловлена не только интервалом повторения, но и его множителями. Множителями числа 9 являются числа 9, 3 и 1. Поэтому, если вы нашли последовательности с интервалом повторения 9, то должны рассмотреть два возможных варианта длины ключа: 9 и 3. Ключ длиной 1 можно игнорировать, поскольку шифр Виженера с ключом такой длины сводится к шифру Цезаря. Второй этап метода Касиски включает нахождение множителей каждого из интервалов повторения.

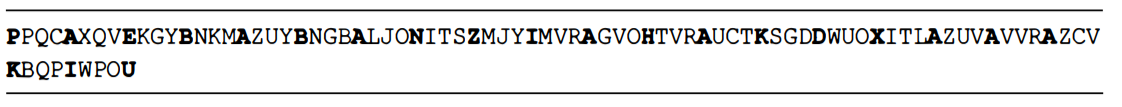


Числа 8, 8, 24, 32 и 48 в совокупности имеют следующие множители: 2, 2, 2, 2, 4, 4, 4, 4, 6, 6, 8, 8, 8, 8, 12, 12, 1 6, 24, 24 и 48.

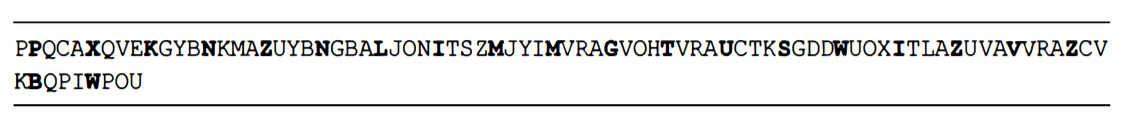
С наибольшей вероятностью ключами окажутся чаще всего встречающиеся множители. Поскольку в данном примере чаще остальных встречаются множители 2, 4 и 8, то именно они являются наиболее вероятными вариантами длины ключа Виженера.

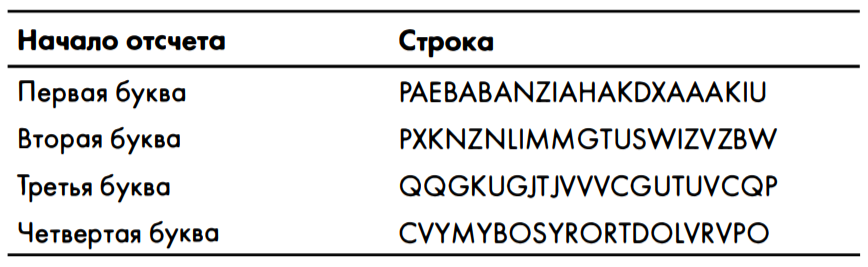
**Получение каждой n-буквы строки**

Теперь, когда нам известны возможные значения длины ключа, мы можем воспользоваться этой информацией для дешифрования сообщения по одному подключу за раз. Предположим, что длина ключа в нашем примере равна 4. Если нам не удастся взломать шифротекст, мы попытаемся использовать ключи длиной 2 или 8. Поскольку ключ применяется циклично, использование ключа длиной 4 означает, что каждая четвертая буква открытого текста шифруется первым подключом, каждая четвертая буква исходного текста, начиная со второй, шифруется вторым подключом и т.д. Мы сформируем строки из букв шифротекста, шифруемых одним и тем же подключом. Сначала мы определим, какой будет каждая четвертая буква строки, если начинать с разных букв, а затем объединим все такие буквы в одну строку. В рассматриваемых примерах каждая четвертая буква выделяется полужирным шрифтом. Идентифицируем каждую четвертую букву, начиная с первой:



Далее находим каждую четвертую букву, начиная со второй:



Затем проделываем то же самое, начиная с третьей и четвертой буквы, пока не исчерпаем всю проверяемую длину ключа. Здесь приведены объединенные строки, составленные из букв, выделенных полужирным шрифтом на каждой итерации. 

**Применение частотного анализа для взлома каждого подключа**

Если мы угадали длину ключа, то каждая из четырех строк, созданных нами в предыдущем разделе, должна быть зашифрована с помощью одного ключа. Это означает, что если строка, зашифрованная корректным ключом, подвергается частотному анализу, то дешифрованные буквы, вероятнее всего, будут иметь высокую оценку частотного соответствия.

**Перебор возможных подключей методом грубой силы**

Метод грубой силы предполагает полный перебор всех возможных комбинаций подключей. Все возможных комбинаций приведены ниже.

Завершающим этапом алгоритма, реализуемого нашей программой взлома шифра Виженера, будет тестирование всех возможных вариантов дешифрования на полном шифротексте, чтобы увидеть, какие из них приводят к получению осмысленного текста на английском языке.

**Глава 3: Код программы**

Исходный код программы:

import detectEnglish, vigenereCipher, pyperclip

def main():

ciphertext = input("""Введите текст: """)

hackedMessage = hackVigenereDictionary(ciphertext)

if hackedMessage != None:

print('Копирование взломанного сообщения в буфер обмена:')

print(hackedMessage)

pyperclip.copy(hackedMessage)

else:

print('Не удалось взломать шифрование')

def hackVigenereDictionary(ciphertext):

fo = open('dictionary.txt')

words = fo.readlines()

fo.close()

for word in words:

word = word.strip() # Remove the newline at the end.

decryptedText = vigenereCipher.decryptMessage(word, ciphertext)

if detectEnglish.isEnglish(decryptedText, wordPercentage=40):

# Check with user to see if the decrypted key has been found:

print()

print('Возможный ключ шифрования:')

print('Ключ ' + str(word) + ': ' + decryptedText[:100])

print()

print('Нажмите D при нахождении ключа, либо Enter для продолжения перебора:')

response = input('> ')

if response.upper().startswith('D'):

return decryptedText

if \_\_name\_\_ == '\_\_main\_\_':

main()

**Глава 4: Демонстрация работы**

1. Для начала работы требуется создать шифрованный текст методом Виженера. Для этого требуется открыть файл vigenereCipher.py и ввести текст для создания шифра. Выбираю myMode == ‘encrypt’ для шифрования. Выбранный мною текст я оставлю в папке с программой.

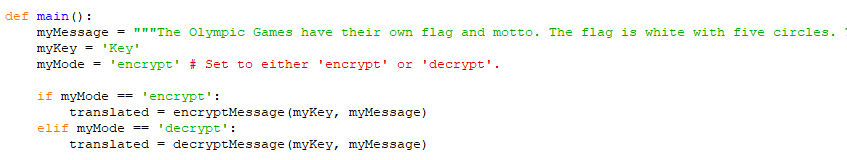


Рисунок 1 – Создание шифра

1. При запуске программы, я получаю шифрованный текст, который находится в файле cipher.py.

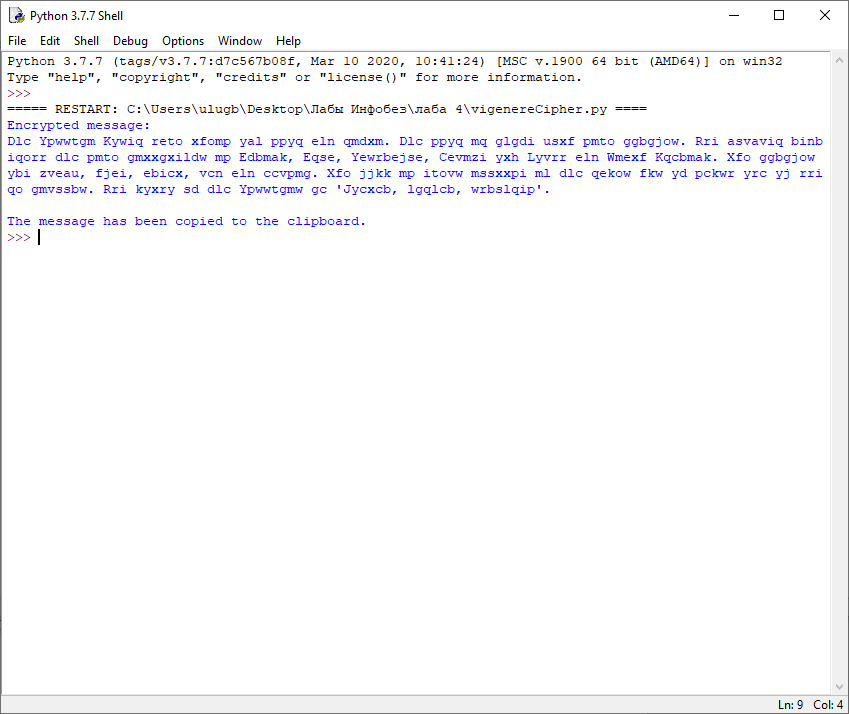
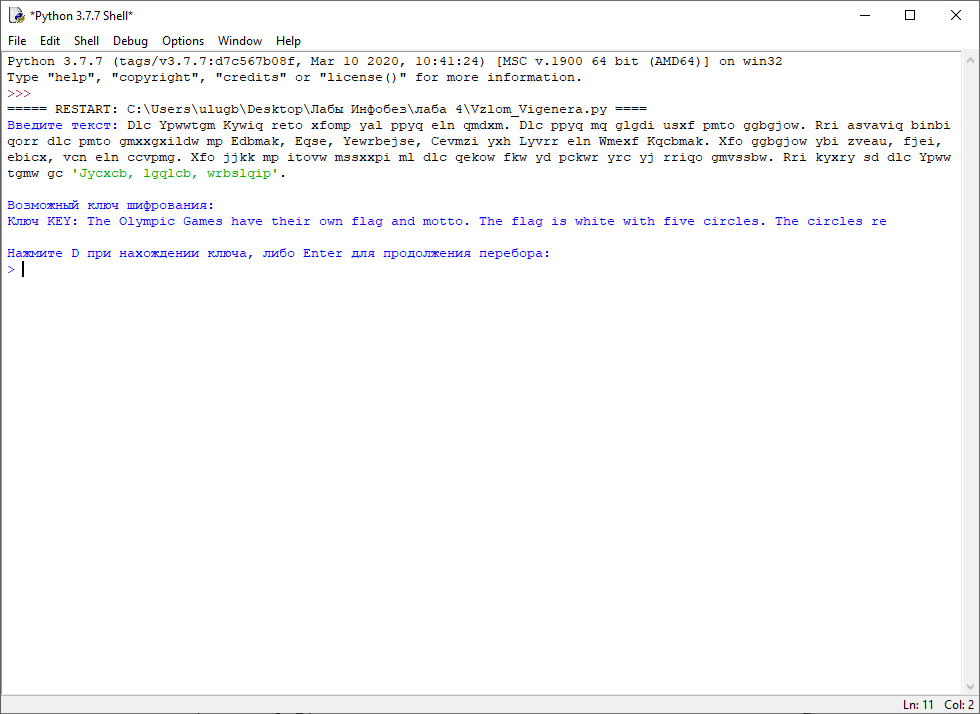
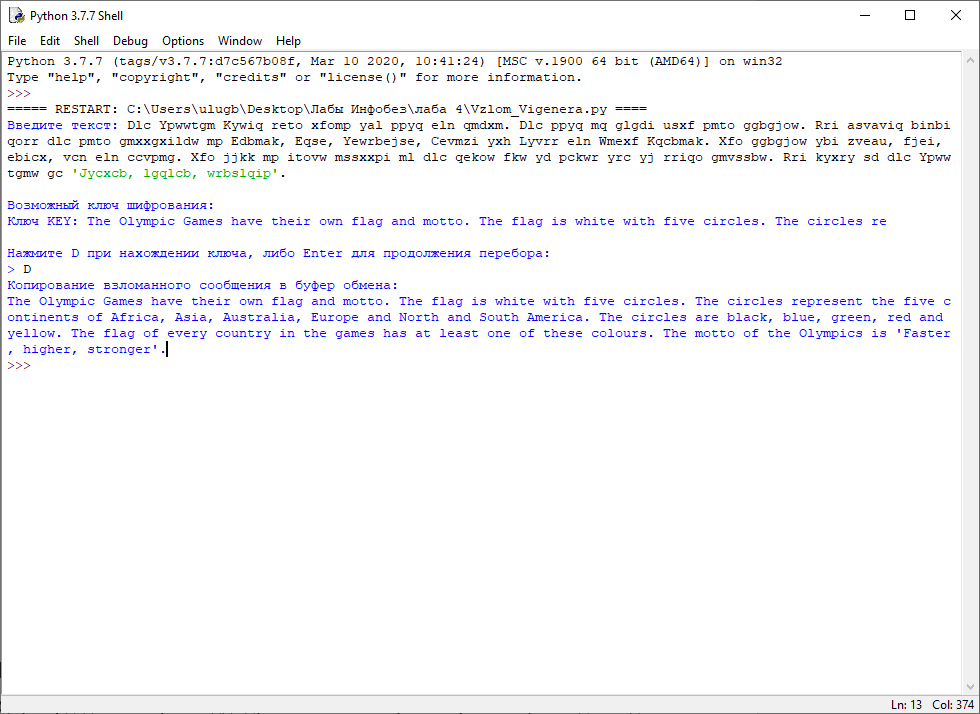


Рисунок 2 – Моё шифрованное слово.

1. Далее, я запускаю программу Vzlom\_Vigenera.py и ввожу скопированный предыдущей программой в буфер обмена текст, как показано на рис. 3



Рис. 3 – Ввод шифра.

1. На выходе программы получаю исходный текст и ключ шифрования, как показано на рис. 4. Если текст не удаётся опоздать как исходный текст, то следует продолжить работу программы, нажав кнопку ENTER. Для завершения программы, соответственно, нажать на кнопку D.   
     
    Рис. 4 – Работа программы.
2. Завершение программы. Полученный текст на выходе программы автоматически копируется, что упрощает взаимодействие с текстом, как показано на рис. 5.  
     
    Рис. 5 – Завершение работы программы.

**Результаты тестирования программы:**

**Шифрованный текст:** Dlc Ypwwtgm Kywiq reto xfomp yal ppyq eln qmdxm. Dlc ppyq mq glgdi usxf pmto ggbgjow. Rri asvaviq binbiqorr dlc pmto gmxxgxildw mp Edbmak, Eqse, Yewrbejse, Cevmzi yxh Lyvrr eln Wmexf Kqcbmak. Xfo ggbgjow ybi zveau, fjei, ebicx, vcn eln ccvpmg. Xfo jjkk mp itovw mssxxpi ml dlc qekow fkw yd pckwr yrc yj rriqo gmvssbw. Rri kyxry sd dlc Ypwwtgmw gc 'Jycxcb, lgqlcb, wrbslqip'.

**Полученный текст:** The Olympic Games have their own flag and motto. The flag is white with five circles. The circles represent the five continents of Africa, Asia, Australia, Europe and North and South America. The circles are black, blue, green, red and yellow. The flag of every country in the games has at least one of these colors. The motto of the Olympics is 'Faster, higher, stronger'.

**Ключ:** KEY

**Выводы**

Взлом шифра Виженера требует соблюдения определенной последовательности действий. Кроме того, многие компоненты программы взлома могут не сработать. Например, ключ шифра Виженера, использованный для шифрования текста, может иметь длину, превышающую значение максимально указанное, или же функция, выдающая оценку частотного соответствия, может получить неточные результаты, поскольку частотность букв в открытом тексте отличается от типичной. Кроме того, открытый текст может содержать слишком много слов, которые отсутствуют в файле словаря, и функция isEnglish () не распознает его как английский текст. Проверяя различные причины, по которым программа взлома не сработала, вы сможете вносить в код соответствующие изменения для обработки подобных случаев. Тем не менее представленный в книге вариант программы взлома отлично справляется с задачей снижения количества возможных ключей, исчисляемых миллиардами или триллионами, всего лишь до нескольких тысяч. Вместе с тем существует одна методика, которая делает взлом шифра Виженера математически невозможным, каким бы мощным ни был компьютер и какой бы изощренной ни была программа взлома.

Список литературы:

1. Шифр Виженера. <https://ru.wikipedia.org/wiki/%D0%A8%D0%B8%D1%84%D1%80_%D0%92%D0%B8%D0%B6%D0%B5%D0%BD%D0%B5%D1%80%D0%B0>
2. THE VIGENÈRE CIPHER. (Cracking Codes with Python)  
   <http://inventwithpython.com/hacking/chapter19.html>