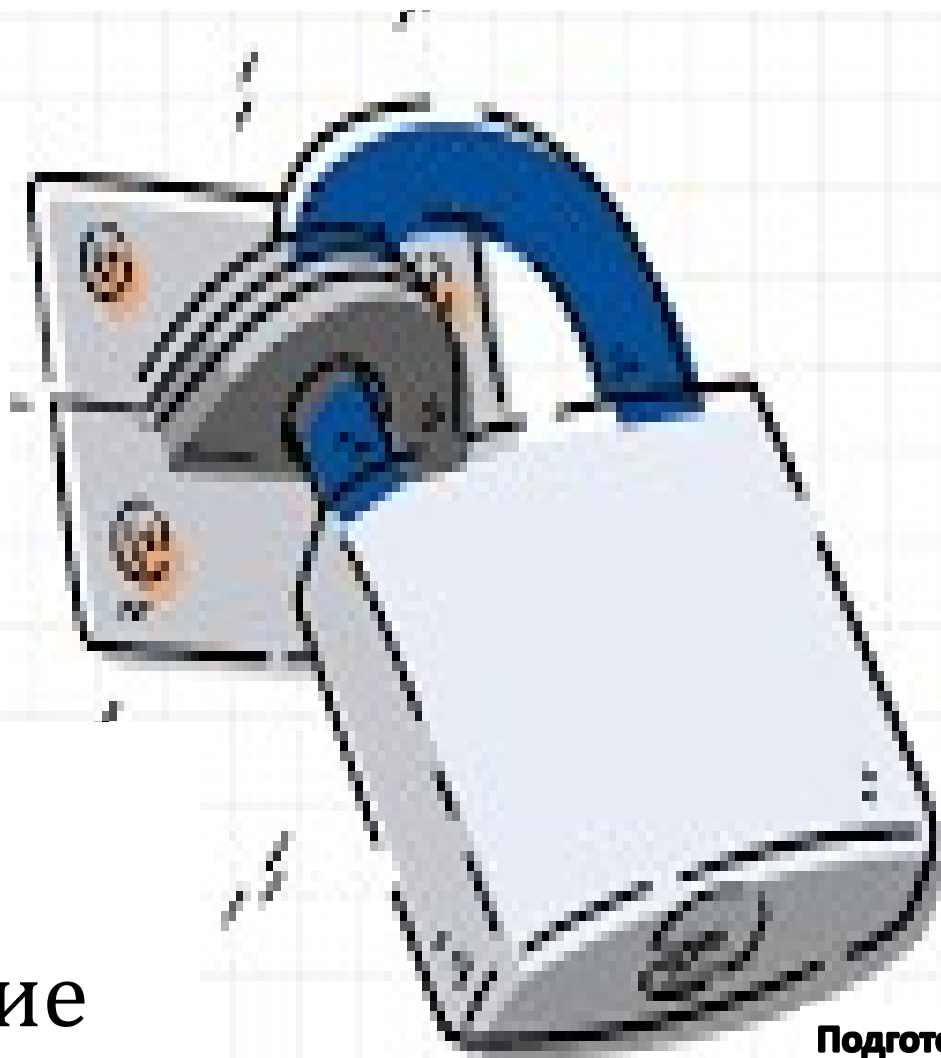


# Российский университет дружбы народов Научный факультет

Математические  
основы защиты  
информации и  
информационной  
безопасности

Гамма-  
шифрование



Подготовлено студентом:  
Елиенис Санчес Родригес.  
Преподаватель: Дмитрий Сергеевич

# Гамма-шифрование

симметричный метод шифрования, заключающийся в "наложении" последовательности, образованной случайными числами, на текст. Последовательность случайных чисел называется гамма-последовательностью и используется для зашифровывания и расшифровки данных. Суммирование обычно выполняется в одном из целевое поле.

# Гамма-шифрование

```
from pyfiglet import figlet_format
```

```
#importamos la funcion para colocar  
letras en la portada
```

```
#импортируем функцию размещения  
букв на обложке
```

```
print(figlet_format("Cifrado de gama by  
Elienis", font = "cybermedium"))
```

```
#crearemos un archivo que se llama  
Source.txt y comenzaremos a escribir en  
el
```

```
#мы создадим файл Source.txt и  
начнем в него писать
```

```
archivo = open('Source.txt','w')  
n= 0
```

```
while n < 1:
```

```
    texto=input('ingresa la frase a  
    encriptar')
```

```
    # вводим в файл фразу для  
    шифрования
```

```
    #introducimos en el archivo la frase a  
    encriptar
```

```
    archivo.write(texto+'\n')  
    n=n+1
```

```
archivo.close()
```

```
#cerramos el archivo
```

```
# закрыть файл
```

```
# cadena aleatoria de bits como clave y  
la combina con el texto sin formato
```

```
# случайная строка битов в качестве  
ключа и комбинирует ее с обычным  
текстом
```

```
A = 15
```

```
B = 17
```

```
M = 4096
```

```
Y0 = 4003
```

```
#definimos gamma y denotamos las  
variables aleatorias
```

```
#определить гамму и обозначить  
случайные величины
```

```
def Gamma(y):
```

```
    gamma_list = []
```

```
    for _ in range(8):
```

```
        y = (A * y + B) % M
```

```
        gamma_list.append(y)
```

```
    return gamma_list
```

```
#definimos la funcion encriptar, abrimos  
el txt de entrada Source y escribimos en  
Result
```

```
#определить функцию шифрования,  
открыть исходный входной txt и  
записать в результат
```

```
def Crypt():
```

```
    gamma = Gamma(Y0)
```

```
    res = open("Result.txt", "w",  
    encoding="utf-8")
```

```
    with open('Source.txt', 'r',  
    encoding="utf-8") as f:
```

```
        r_int = ""
```

```
        r = ""
```

```
        while True:
```

```
            temp = f.read(8)
```

```
            if temp:
```

```
                for i, item in enumerate(temp):
```

```
                    r_int = r_int + " " +
```

```
                    str(ord(item) ^ gamma[i])
```

```
                    # объединить гамма-
```

```
                    переменную с введенным текстом
```

```
                    #combinamos la variable  
                    gama con el texto intriducido
```

```
                    r = r + " " + chr(ord(item) ^
```

```
                    gamma[i])
```

```
                    res.write(chr(ord(item) ^
```

```
                    gamma[i]))
```

```
    else:
```

```
        break
```

```
    print(r_int)
```

```
    res.close()
```

```
#guardamos en resultado en el archivo
```

```
Result.txt
```

```
# сохранить результат в файле
```

```
Result.txt
```

```
Crypt()
```

```
# definimos la funcion para
```

```
desencriptar y generar un archivo
```

```
llamado NewResult
```

```
# определить функцию для
```

```
расшифровки и создания файла с  
именем NewResult
```

```
def DeCrypt():
```

```
    gamma = Gamma(Y0)
```

```
    #crearemos un nuevo archivo que  
    contendra la frase decifrada
```

```
    # создаст новый файл, который  
    будет содержать расшифрованную  
    фразу
```

```
    res = open("NewResult.txt", "w",  
    encoding="utf-8")
```

```
    # abrimos el archivo con la frase
```

```
    cifrada
```

```
    with open('Result.txt', 'r',
```

```
    encoding="utf-8") as f:
```

```
        r_int = ""
```

```
        r = ""
```

```
        while True:
```

```
            temp = f.read(8)
```

```
            if temp:
```

```
                for i, item in enumerate(temp):
```

```
# realizamos el decifrado de gamma
```

```
    # выполнить гамма-  
    расшифровку
```

```
        r_int = r_int + " " + str(ord(item) ^  
    gamma[i])
```

```
        r = r + chr(ord(item) ^ gamma[i])
```

```
        res.write(chr(ord(item) ^
```

```
        gamma[i]))
```

```
    else:
```

```
        break
```

```
    print(r_int)
```

```
#импримимос el resultado y lo grabamos
```

```
# распечатываем результат и сохраняем  
res.close()
```

```
DeCrypt()
```

```
print("a continuacion se mostrara el texto  
cifrado\n") # распечатать результаты
```

```
шифрования
```

```
with
```

```
open("C:/Users/kami/Documents/matematica/I  
ab03/Result.txt", "r") as archivo:
```

```
    for linea in archivo:
```

```
        print(linea)
```

```
print("a continuacion se mostrara el texto  
decifrado\n") # вывести расшифрованные
```

```
результаты
```

```
with
```

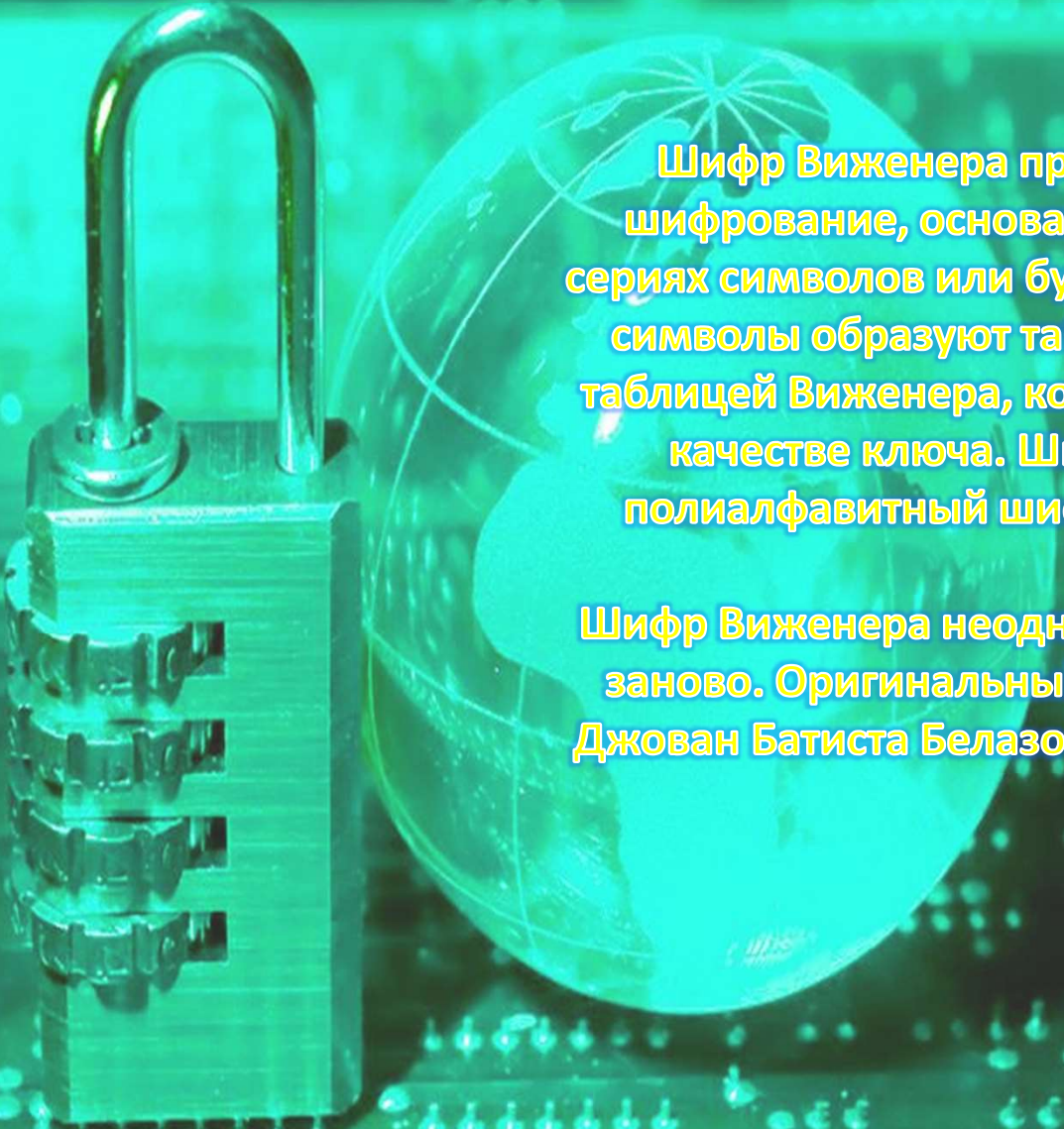
```
open("C:/Users/kami/Documents/matematica/I  
ab03/NewResult.txt", "r") as archivo:
```

```
    for linea in archivo:
```

```
        print(linea)
```



# шифр Виженера

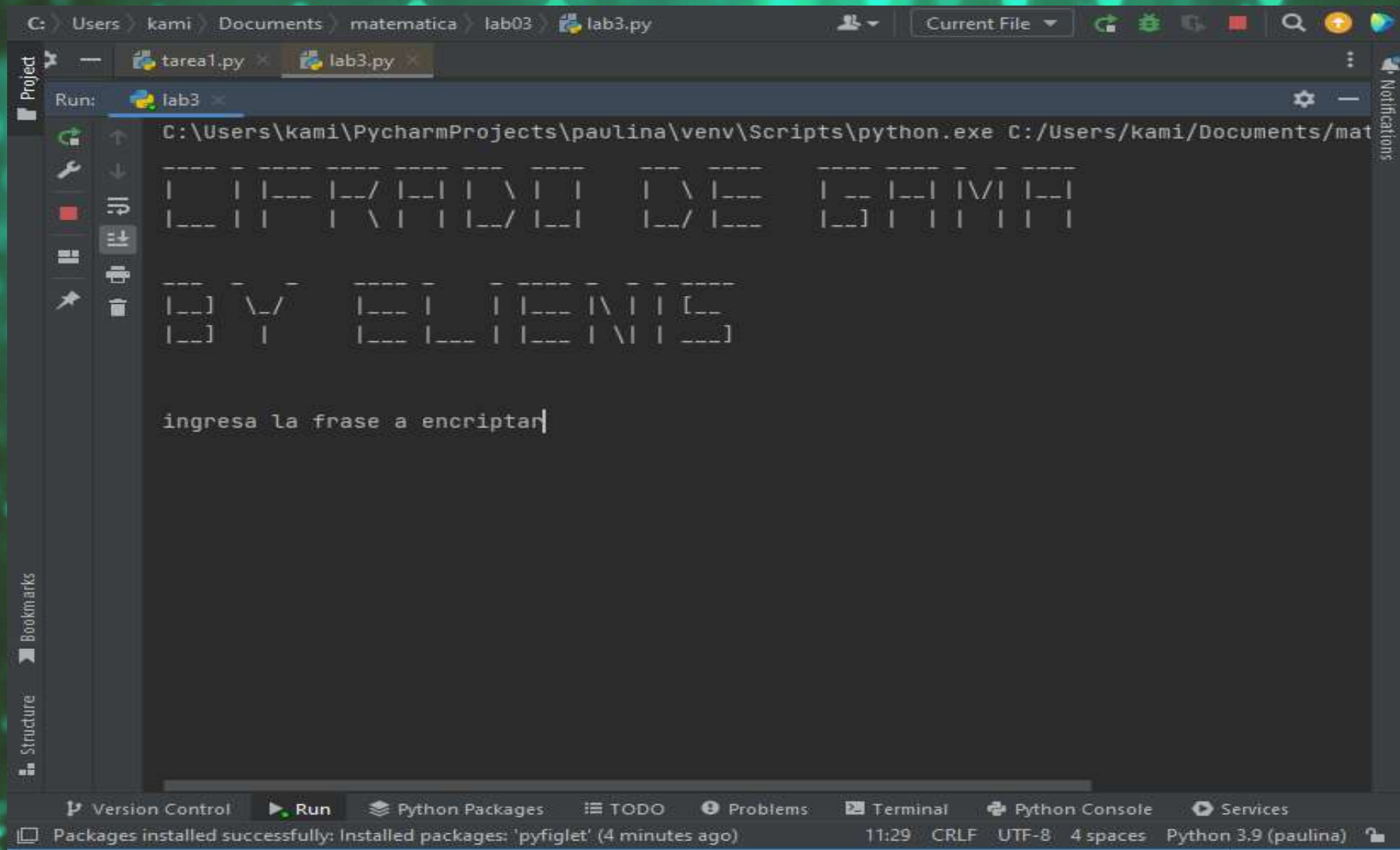
A padlock and a globe are positioned in the center of the image. The padlock is on the left, and the globe is on the right. They are set against a background of a green circuit board with glowing blue dots. The text is overlaid on the right side of the image.

Шифр Виженера представляет собой шифрование, основанное на различных сериях символов или букв шифра Цезаря, эти символы образуют таблицу, называемую таблицей Виженера, которая используется в качестве ключа. Шифр Виженера — полиалфавитный шифр с подстановкой.

Шифр Виженера неоднократно изобретался заново. Оригинальный метод был описан Джован Батиста Белазо в его книге 1553 года



# шифр Виженера

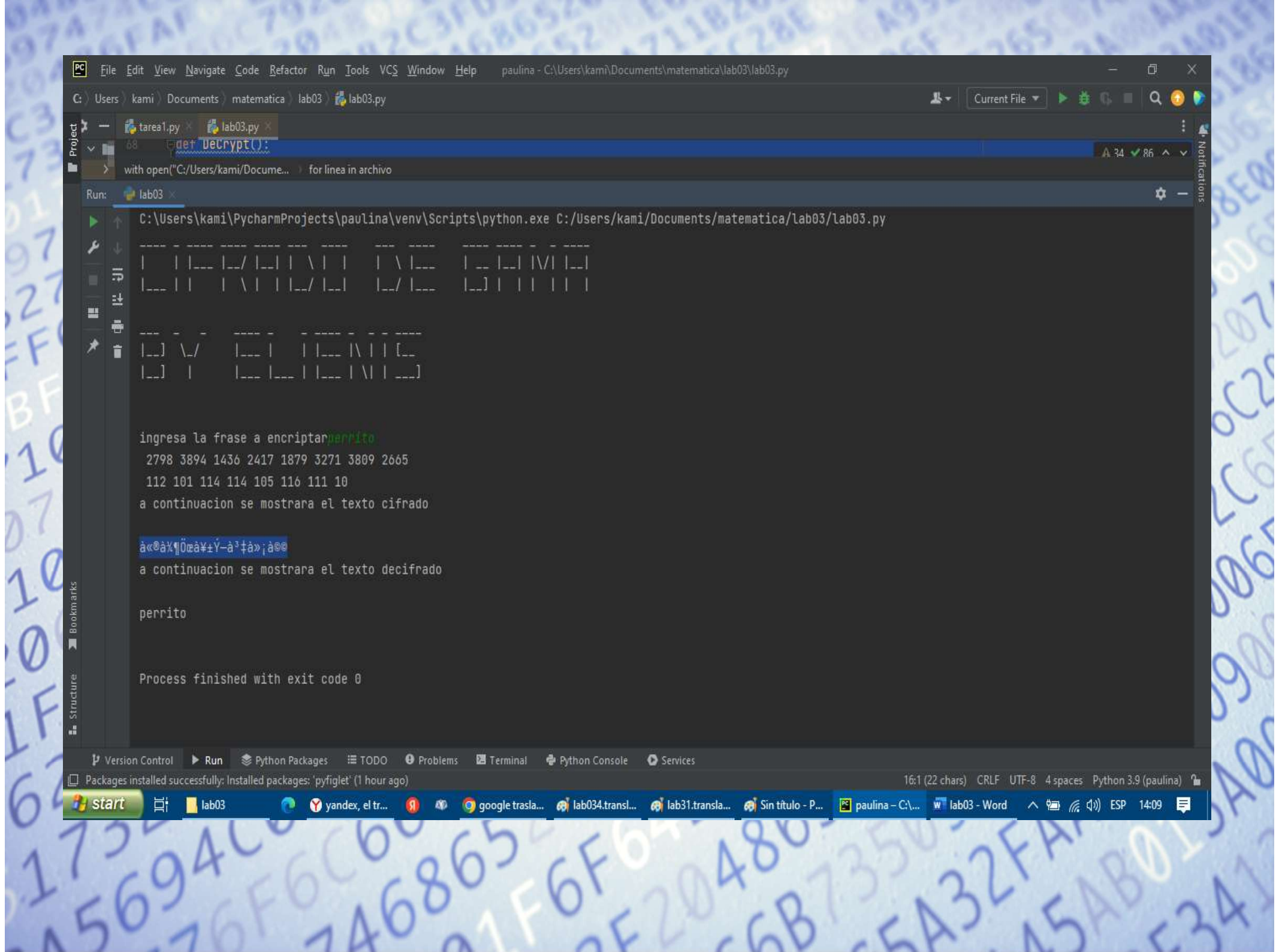


The screenshot shows a PyCharm IDE window with the following details:

- File Path:** C:\Users\kami\Documents\matematica\lab03\lab3.py
- Run Configuration:** Run: lab3
- Terminal Output:**

```
C:\Users\kami\PycharmProjects\paulina\venv\Scripts\python.exe C:/Users/kami/Documents/matematica/lab03/lab3.py
```
- Script Content:**

```
-----  
| | | | | | | | | | | | | | | | | | | | | | | | | | | |  
| | | | | | | | | | | | | | | | | | | | | | | | | | | |  
-----  
  
-----  
| | | | | | | | | | | | | | | | | | | | | | | | | | | |  
| | | | | | | | | | | | | | | | | | | | | | | | | | | |  
-----  
  
ingresa la frase a encriptar
```
- Status Bar:** Packages installed successfully: Installed packages: 'pyfiglet' (4 minutes ago) 11:29 CRLF UTF-8 4 spaces Python 3.9 (paulina)



Microsoft Windows [Versión 10.0.19042.1706]  
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Users\kami>

C:\Users\kami>

# ВЫВОД

Этот тип шифрования довольно сложно взломать, поскольку ключ здесь является переменным. По сути, гамма шифрования должна изменяться случайным образом, чтобы каждый блок был зашифрован. Если период гамма превышает длину всего зашифрованного текста и злоумышленник не знает ни одной части исходного текста, такое шифрование может быть разрешено только путем прямого перечисления всех ключевых параметров. В этом случае надежность шифрования определяется длиной ключа.

Однако гамма-метод становится бессильным, если злоумышленник распознает фрагмент исходного текста и соответствующий шифр. Простым вычитанием по модулю получается отрезок псевдослучайной последовательности и из него восстанавливается вся последовательность.

Microsoft Windows [Versión 10.0.19042.1706]  
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Users\kami>

C:\Users\kami>

# Библиография

Perez P, B., & Acosta Velarde, R. (2019). Mejora en la seguridad python . Ciencia Digital, 2(3), 61-74 <https://retrolib.ru/realizacziya-algoritma-shifrovaniya-gammirovanie-pascalpaskal-python-piton/>

Cabrera R, Juan , python-transposition (2017). método transposición. Ciencia Digital, 8(5), <https://blog.finxter.com/python-transposition-algorithm/>



Microsoft Windows [Versión 10.0.19042.1706]  
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Users\kami>

C:\Users\kami>Большое спасибо