**Министерство науки и высшего образования Российской Федерации Федеральное государственное бюджетное образовательное учреждение высшего образования** **«Новгородский государственный университет имени Ярослава Мудрого»**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Кафедра ИТиС

Лабораторная работа №2

По дисциплине: Защита информации

Отчет

Выполнил: студент гр. 9091

\_\_\_\_\_\_\_\_\_\_\_Чалый.С.М

Проверил: Преподаватель

\_\_\_\_\_\_\_\_\_\_\_Жгун.Т.В

Великий Новгород

2022

1. **Формулировка цели и задач**

Целью данной работы является шифровка текста объемом около 500 символов. Открытый текст выбирается из текстового документа. Используются шифрование Вижинера.

С помощью ключа (пароля), где в качестве ключа используется:

1.1 Константа, равную номеру в списке(19);

1.2 Поговорка из таблицы «РОДНОЙ КУСТ И ЗАЙЦУ ДОРОГ».

» используется алфавит Z26 = (А….z);

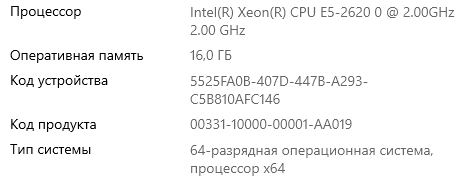
Задание2:

Построить гистограммы для полученных зашифрованных текстов, проанализировав результат

устройства.

1. **Текст программы**

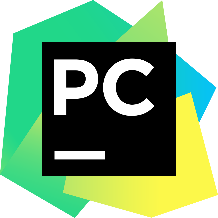
**2.1 Тестирование и запуск.**



Необходимые средства применяемые мною при выполнении данной работы:

IntelliJ IDEA Educational Edition 2021.2.2

* Version: 2021.3.2
* Build: 213.6777.50
* 31 January 2022
* Community
* Python 3.10



"**2.1 Тестирование и запуск**".

import math  
import numpy as np  
import matplotlib.pyplot as plt  
  
  
def h\_line0(): return "---------------------------------------------------------------"  
  
  
def h\_line1(): return "---------------------Cryptography\_by\_#571----------------------"  
  
  
# Tests  
Name = []  
Values = []  
  
  
def makeGraph(str):  
 Name.clear()  
 Values.clear()  
 fig, ax = plt.subplots()  
 for i in str:  
 counter = 0  
 for g in str:  
 if i == g:  
 counter += 1  
 Name.append(i)  
 Values.append(counter)  
  
 plt.bar(Name, Values)  
 plt.show()  
  
  
def c\_vishener():  
 # 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25  
 arr = ['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']  
 print()  
  
 def switch(n):  
 while n < 26:  
 for i in arr:  
 if i == arr[0]:  
 print(end="|")  
 print(i, end="|")  
 arr.append(arr[0])  
 arr.remove(arr[0])  
 print()  
 n += 1  
  
 switch(0)  
 print()  
 print(h\_line0() + "\n" + h\_line1() + "\n" + h\_line0())  
 print("1) Create the Crypt and Key files")  
 print("2) Create the Crypt and Key files (Disposable)")  
 print("3) Read the Crypt-file with Key-file")  
 print("4) Read the Crypt-text with Key\n")  
 try:  
 answer = int(input(">> Choose the Number: "))  
 print()  
 if answer == 1:  
 with open("key.txt", "w") as file\_key:  
 m = input("Write the text: ")  
 k = input("Write the key: ")  
 k \*= len(m) // len(k) + 1  
 file\_key.write(k)  
 with open("crypt.txt", "w") as file\_crypt:  
 c = ""  
 for i, j in enumerate(m):  
 gg = (ord(j) + ord(k[i]))  
 c += chr(gg % 26 + ord('A'))  
 print()  
 print("Encrypted message: " + str(c))  
 makeGraph(str(c))  
 file\_crypt.write(c)  
 v = ""  
 for i, j in enumerate(c):  
 gg = (ord(j) - ord(k[i]))  
 v += chr(gg % 26 + ord('A'))  
 print("Decrypted message: " + str(v) + "\n")  
 print("Encrypted file: crypt.txt")  
 print("Key-file: key.txt\n")  
 elif answer == 2:  
 with open("key.txt", "w") as file\_key:  
 m = input("Write the text: ")  
 length = len(m)  
 import random  
 k = ""  
 for i in range(length):  
 rand = random.randint(0, 25)  
 k += arr[rand]  
 print("Key: " + k)  
 file\_key.write(k)  
 with open("crypt.txt", "w") as file\_crypt:  
 c = ""  
 for i, j in enumerate(m):  
 gg = (ord(j) + ord(k[i]))  
 c += chr(gg % 26 + ord('A'))  
 print()  
 print("Ecrypted message: " + str(c))  
  
 file\_crypt.write(c)  
 v = ""  
 for i, j in enumerate(c):  
 gg = (ord(j) - ord(k[i]))  
 v += chr(gg % 26 + ord('A'))  
 print("Decrypted message: " + str(v) + "\n")  
 print("Encrypted file: crypt.txt")  
 print("Key-file: key.txt\n")  
 elif answer == 3:  
 with open("crypt.txt", "r") as file\_crypt:  
 c = file\_crypt.read()  
 with open("key.txt", "r") as file\_key:  
 k = file\_key.read()  
 v = ""  
 for i, j in enumerate(c):  
 gg = (ord(j) - ord(k[i]))  
 v += chr(gg % 26 + ord('A'))  
 print("Encrypted file: crypt.txt")  
 print("Key-file: key.txt\n")  
 print("Decrypted message: " + str(v) + "\n")  
 elif answer == 4:  
 c = input("Write the Crypt-text: ")  
 k = input("Write the Key: ")  
 v = ""  
 for i, j in enumerate(c):  
 gg = (ord(j) - ord(k[i]))  
 v += chr(gg % 26 + ord('A'))  
 print("\nDecrypted message: " + str(v) + "\n")  
 else:  
 print("Number is not Defined!")  
 except ValueError:  
 print("[x] Error!"); print("Write only Integer Numbers!\n")  
 except FileNotFoundError:  
 print("[x] Error!"); print("Crypt/Key Files is not Defined!\n")  
  
  
c\_vishener()

Изображение выглядит как текст

Автоматически созданное описание

import math  
import numpy as np  
import matplotlib.pyplot as plt  
  
  
def h\_line0(): return "---------------------------------------------------------------"  
  
  
def h\_line1(): return "---------------------Cryptography\_by\_#571----------------------"  
  
  
# Tests  
Name = []  
Values = []  
  
  
def makeGraph(str):  
 Name.clear()  
 Values.clear()  
 fig, ax = plt.subplots()  
 for i in str:  
 counter = 0  
 for g in str:  
 if i == g:  
 counter += 1  
 Name.append(i)  
 Values.append(counter)  
  
 plt.bar(Name, Values)  
 plt.show()  
  
  
def c\_vishener():  
 # 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25  
 arr = ['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']  
 print()  
  
 def switch(n):  
 while n < 26:  
 for i in arr:  
 if i == arr[0]:  
 print(end="|")  
 print(i, end="|")  
 arr.append(arr[0])  
 arr.remove(arr[0])  
 print()  
 n += 1  
  
 switch(0)  
 print()  
 print(h\_line0() + "\n" + h\_line1() + "\n" + h\_line0())  
 print("1) Create the Crypt and Key files")  
 print("2) Create the Crypt and Key files (Disposable)")  
 print("3) Read the Crypt-file with Key-file")  
 print("4) Read the Crypt-text with Key\n")  
 try:  
 answer = int(input(">> Choose the Number: "))  
 print()  
 if answer == 1:  
 with open("key.txt", "w") as file\_key:  
 m = input("Write the text: ")  
 k = input("Write the key: ")  
 k \*= len(m) // len(k) + 1  
 file\_key.write(k)  
 with open("crypt.txt", "w") as file\_crypt:  
 c = ""  
 for i, j in enumerate(m):  
 gg = (ord(j) + ord(k[i]))  
 c += chr(gg % 26 + ord('A'))  
 print()  
 print("Encrypted message: " + str(c))  
 makeGraph(str(c))  
 file\_crypt.write(c)  
 v = ""  
 for i, j in enumerate(c):  
 gg = (ord(j) - ord(k[i]))  
 v += chr(gg % 26 + ord('A'))  
 print("Decrypted message: " + str(v) + "\n")  
 print("Encrypted file: crypt.txt")  
 print("Key-file: key.txt\n")  
 elif answer == 2:  
 with open("key.txt", "w") as file\_key:  
 m = input("Write the text: ")  
 length = len(m)  
 import random  
 k = ""  
 for i in range(length):  
 rand = random.randint(0, 25)  
 k += arr[rand]  
 print("Key: " + k)  
 file\_key.write(k)  
 with open("crypt.txt", "w") as file\_crypt:  
 c = ""  
 for i, j in enumerate(m):  
 gg = (ord(j) + ord(k[i]))  
 c += chr(gg % 26 + ord('A'))  
 print()  
 print("Ecrypted message: " + str(c))  
  
 file\_crypt.write(c)  
 v = ""  
 for i, j in enumerate(c):  
 gg = (ord(j) - ord(k[i]))  
 v += chr(gg % 26 + ord('A'))  
 print("Decrypted message: " + str(v) + "\n")  
 print("Encrypted file: crypt.txt")  
 print("Key-file: key.txt\n")  
 elif answer == 3:  
 with open("crypt.txt", "r") as file\_crypt:  
 c = file\_crypt.read()  
 with open("key.txt", "r") as file\_key:  
 k = file\_key.read()  
 v = ""  
 for i, j in enumerate(c):  
 gg = (ord(j) - ord(k[i]))  
 v += chr(gg % 26 + ord('A'))  
 print("Encrypted file: crypt.txt")  
 print("Key-file: key.txt\n")  
 print("Decrypted message: " + str(v) + "\n")  
 elif answer == 4:  
 c = input("Write the Crypt-text: ")  
 k = input("Write the Key: ")  
 v = ""  
 for i, j in enumerate(c):  
 gg = (ord(j) - ord(k[i]))  
 v += chr(gg % 26 + ord('A'))  
 print("\nDecrypted message: " + str(v) + "\n")  
 else:  
 print("Number is not Defined!")  
 except ValueError:  
 print("[x] Error!"); print("Write only Integer Numbers!\n")  
 except FileNotFoundError:  
 print("[x] Error!"); print("Crypt/Key Files is not Defined!\n")  
  
  
c\_vishener()

