**ОТЧЕТ К ЛАБОРАТОРНОЙ РАБОТЕ № 1**

**Лабораторная работа № 1**

**Шифр Цезаря**

**Вариант № 9**

Ф. И. О студента: Круглов Богдан Евгеньевич

Группа: ФИТ-222

Проверил: Белим С. Ю. Дата: 26.09.2024

**Основные сведения**

Прямое преобразование шифра Цезаря:

Обратное преобразование шифра Цезаря:

Таблица кодировки символов:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| а | 0 | и | 8 | р | 16 | ш | 24 |
| б | 1 | й | 9 | с | 17 | щ | 25 |
| в | 2 | к | 10 | т | 18 | ъ | 26 |
| г | 3 | л | 11 | у | 19 | ы | 27 |
| д | 4 | м | 12 | ф | 20 | ь | 28 |
| е\ё | 5 | н | 13 | х | 21 | э | 29 |
| ж | 6 | о | 14 | ц | 22 | ю | 30 |
| з | 7 | п | 15 | ч | 23 | я | 31 |

**Результаты**

ШИФР-ТЕКСТ (ШТ): пътрьяшигзхуюцхсюыхзвюпьюугхйхбърчрвм

РАСШИФРОВАННЫЙ ТЕКСТ (ОТ): яквaмпишучегожеболечтоямогуещескaзaть

КЛЮЧ: 16

АВТОР И ПРОИЗВЕДЕНИЕ (ОТ): пушкиневгенийонегин

ЗАШИФРОВАННЫЕ ФАМИЛИИ И НАЗВАНИЕ (ШТ): ягиъшэхтухэшщюэхушэ

Варианты расшифрования исходного ШТ при различных значениях ключа:

k = 1: пътрьяшигзхуюцхсюыхзвюпьюугхйхбърчрвм

k = 2: ощспыючзвжфтэхфрэъфжбэоыэтвфифaщпцпбл

k = 3: ншроъэцжбеусьфупьщуеaьнъьсбузуяшохоaк

k = 4: мчпнщьхеaдтрыутоыштдяымщырaтжтючнфняй

k = 5: лцомшыфдягспътснъчсгюълшъпясесэцмумюи

k = 6: кхнлчъугюврощсрмщцрвэщкчщоюрдрьхлтлэз

k = 7: йфмкцщтвэбпншрплшхпбьшйцшнэпгпыфкскьж

k = 8: иулйхшсбьaомчпокчфоaычихчмьовоъуйрйые

k = 9: зткифчрaыянлцонйцуняъцзфцлынбнщтипиъд

k = 10: жсйзуцпяъюмкхнмихтмющхжухкъмaмшсзозщг

k = 11: ерижтхоющэлйфмлзфслэшфетфйщлялчржнжшв

k = 12: дпзесфнэшькиулкжуркьчудсуишкюкцпемечб

k = 13: гождрумьчыйзткйетпйыцтгртзчйэйходлдцa

k = 14: внегптлыцъижсйидсоиъхсвпсжциьифнгкгхя

k = 15: бмдвоскъхщзеризгрнзщфрборехзызумвйвфю

k = 16: aлгбнрйщфшждпзжвпмжшупaнпдфжъжтлбибуэ

k = 17: яквaмпишучегожеболечтоямогуещескaзaть

k = 18: юйбялозчтцдвнедaнкдцснюлнвтдшдрйяжясы

k = 19: эиaюкнжцсхгбмдгямйгхрмэкмбсгчгпиюеюръ

k = 20: ьзяэймехрфвaлгвюливфпльйлaрвцвозэдэпщ

k = 21: ыжюьилдфпубяквбэкзбуокыикяпбхбнжьгьош

k = 22: ъеэызкгуотaюйбaьйжaтнйъзйюоaфaмеывынч

k = 23: щдьъжйвтнсяэиaяыиеясмищжиэняуялдъбъмц

k = 24: шгыщеибсмрюьзяюъздюрлзшезьмютюкгщaщлх

k = 25: чвъшдзaрлпэыжюэщжгэпкжчджылэсэйвшяшкф

k = 26: цбщчгжяпкоьъеэьшевьойецгеъкьрьибчючйу

k = 27: хaшцвеюойныщдьычдбынидхвдщйыпызaцэцит

k = 28: фячхбдэнимъшгыъцгaъмзгфбгшиъоъжяхьхзс

k = 29: уюцфaгьмзлщчвъщхвящлжвуaвчзщнщеюфыфжр

k = 30: тэхуявылжкшцбщшфбюшкебтябцжшмшдэуъуеп

k = 31: сьфтюбъкейчхaшчуaэчйдaсюaхечлчгьтщтдо

k = 32: рыусэaщйдицфячцтяьцигярэяфдцкцвысшсгн

**Код программы**

dict = {

'а':0,

'б' : 1,

'в' : 2,

'г' : 3,

'д' : 4,

'е' : 5,

'ж' : 6,

'з' : 7,

'и' : 8,

'й' : 9,

'к' : 10,

'л' : 11,

'м' : 12,

'н' : 13,

'о' : 14,

'п' : 15,

'р' : 16,

'с' : 17,

'т' : 18,

'у' : 19,

'ф' : 20,

'х' : 21,

'ц' : 22,

'ч' : 23,

'ш' : 24,

'щ' : 25,

'ъ' : 26,

'ы' : 27,

'ь' : 28,

'э' : 29,

'ю' : 30,

'я' : 31,

}

dict\_reverse = {

0:'a',

1:'б',

2:'в',

3:'г',

4:'д',

5:'е',

6:'ж',

7:'з',

8:'и',

9:'й',

10:'к',

11:'л',

12:'м',

13:'н',

14:'о',

15:'п',

16:'р',

17:'с',

18:'т',

19:'у',

20:'ф',

21:'х',

22:'ц',

23:'ч',

24:'ш',

25:'щ',

26:'ъ',

27:'ы',

28:'ь',

29:'э',

30:'ю',

31:'я'

}

def add\_in\_ring(a, b, n):

"""

Складывает два числа a и b в кольце вычетов по модулю n.

:param a: Первое число

:param b: Второе число

:param n: Модуль (размер кольца)

:return: Сумма по модулю n

"""

return (a + b) % n

def min\_in\_ring(a, b, n):

"""

Складывает два числа a и b в кольце вычетов по модулю n.

:param a: Первое число

:param b: Второе число

:param n: Модуль (размер кольца)

:return: Сумма по модулю n

"""

return (a - b) % n

def is\_russian\_letter(char):

return 'а' <= char <= 'я' or 'А' <= char <= 'Я'

def cesar\_decipher(text):

syms = [sym for sym in text]

b = ""

for i in range(32):

a = []

out = ''

for j in range(len(syms)):

if is\_russian\_letter(syms[j]):

a.append(dict\_reverse[min\_in\_ring(dict[syms[j]], i, 32)])

else:

a.append(syms[j])

for h in range(len(a)):

out += a[h]

b += f"k = {i}: {out}\n"

print(f'k = {i}: {out}')

return b

def cesar\_decipher\_with\_key(text, key):

syms = [sym for sym in text]

a = []

out = ''

for j in range(len(syms)):

if is\_russian\_letter(syms[j]):

a.append(dict\_reverse[min\_in\_ring(dict[syms[j]], key, 32)])

else:

a.append(syms[j])

for h in range(len(a)):

out += a[h]

print(f'{out}')

return out

def cesar\_encrypt(text, key):

syms = [sym for sym in text]

a = []

out = ''

for j in range(len(syms)):

if is\_russian\_letter(syms[j]):

a.append(dict\_reverse[add\_in\_ring(dict[syms[j]], key, 32)])

else:

a.append(syms[j])

for h in range(len(a)):

out += a[h]

print(f'{out}')

return out

def main\_menu():

while True:

print("Выберите опцию:")

print("1: Дешифровать текст с перебором ключа")

print("2: Дешифровать текст с заданным ключом")

print("3: Зашифровать текст с заданным ключом")

print("4: Выход")

choice = input("Введите номер опции: ")

if choice == '1':

text = input("Введите текст для дешифровки: ")

cd = cesar\_decipher(text)

with open('output\_many.txt', 'w', encoding='utf-8') as file:

file.write(cd + '\n')

elif choice == '2':

text = input("Введите текст для дешифровки: ")

key = int(input("Введите ключ: "))

cesar\_decipher\_with\_key(text, key)

elif choice == '3':

text = input("Введите текст для шифровки: ")

key = int(input("Введите ключ: "))

ce = cesar\_encrypt(text, key)

with open('output\_one.txt', 'w', encoding='utf-8') as file:

file.write(ce + '\n')

elif choice == '4':

print("Выход из программы.")

break

else:

print("Неверный ввод. Пожалуйста, попробуйте снова.")

main\_menu()