RSA

import random

import math

def is\_prime(n):

    if n <= 1:

        return False

    if n <= 3:

        return True

    if n % 2 == 0 or n % 3 == 0:

        return False

    i = 5

    while i \* i <= n:

        if n % i == 0 or n % (i + 2) == 0:

            return False

        i += 6

    return True

def pred(s):

    llst = ['а','б','в','г','д','е','ж','з','и','й','к','л','м','н','о','п','р','с','т','у','ф','х','ц','ч','ш','щ','ъ','ы','ь','э','ю','я']

    s = s.lower().replace(' ', '')

    for sim in s:

        if sim not in llst:

            if sim == '.':

                s = s.replace('.', 'тчк')

            elif sim == ',':

                s = s.replace(',', 'зпт')

            elif sim == '-':

                s = s.replace('-', 'тире')

            elif sim == 'ё':

                s = s.replace('ё', 'е')

            elif sim == '0':

                s = s.replace('0', 'ноль')

            elif sim == '1':

                s = s.replace('1', 'один')

            elif sim == '2':

                s = s.replace('2', 'два')

            elif sim == '3':

                s = s.replace('3', 'три')

            elif sim == '4':

                s = s.replace('4', 'четыре')

            elif sim == '5':

                s = s.replace('5', 'пять')

            elif sim == '1':

                s = s.replace('6', 'шесть')

            elif sim == '2':

                s = s.replace('7', 'семь')

            elif sim == '3':

                s = s.replace('8', 'восемь')

            elif sim == '4':

                s = s.replace('9', 'девять')

            else:

                s = s.replace(sim, '')

    return s

def hesh(str,p,i):

    alp = " АБВГДЕЖЗИЙКЛМНОПРСТУФХЦЧШЩЪЫЬЭЮЯ"

    if i==0:

        q=(alp.index(str[i])\*\*2)%p

        return q

    else:

        q=((hesh(str,p,i-1)+alp.index(str[i]))\*\*2)%p

        return q

### RSA ###

def simple(l, b):

    k = random.randint(l, 500)

    while k != 0 and b != 0:

        if k > b:

            k %= b

        else:

            b %= k

    gcd = k + b

    if gcd != 1:

        simple(l, b)

    return k

def encrsa(s, P, Q):

    llst = ['а','б','в','г','д','е','ж','з','и','й','к','л','м','н','о','п','р','с','т','у','ф','х','ц','ч','ш','щ','ъ','ы','ь','э','ю','я']

    s = pred(s)

    res = []

    D = 1

    N = P \* Q

    F = (P - 1) \* (Q - 1)

    arr = [i for i in range(2,F) if math.gcd(i,F) == 1]

    E = random.choice(arr)

    print("N = ", N)

    print("F = ", F)

    print("E = ", E)

    for i in range(10000):

        if i \* E % F == 1 and i != D:

            D = i

            break

    print("D = ", D)

    for x in s:

        x = llst.index(x)

        res.append((x \*\* E) % N)

    return res, D, N

def decrsa(s, d, n):

    llst = ['а','б','в','г','д','е','ж','з','и','й','к','л','м','н','о','п','р','с','т','у','ф','х','ц','ч','ш','щ','ъ','ы','ь','э','ю','я']

    res = ""

    for i in s:

        ind = (i \*\* d) % n

        res = res + llst[ind]

    return res

# для RSA

print("RSA")

P = int(input("P = "))

if is\_prime(P):

    Q = int(input("Q = "))

    if is\_prime(Q):

        if P \* Q >= 32:

            text = str(input("Введите текст: "))

            res = encrsa(text, P, Q)

            print("Зашифрованный текст: ",res[0])

            print("Расшифрованный текст: ",decrsa(res[0],res[1], res[2]))

        else:

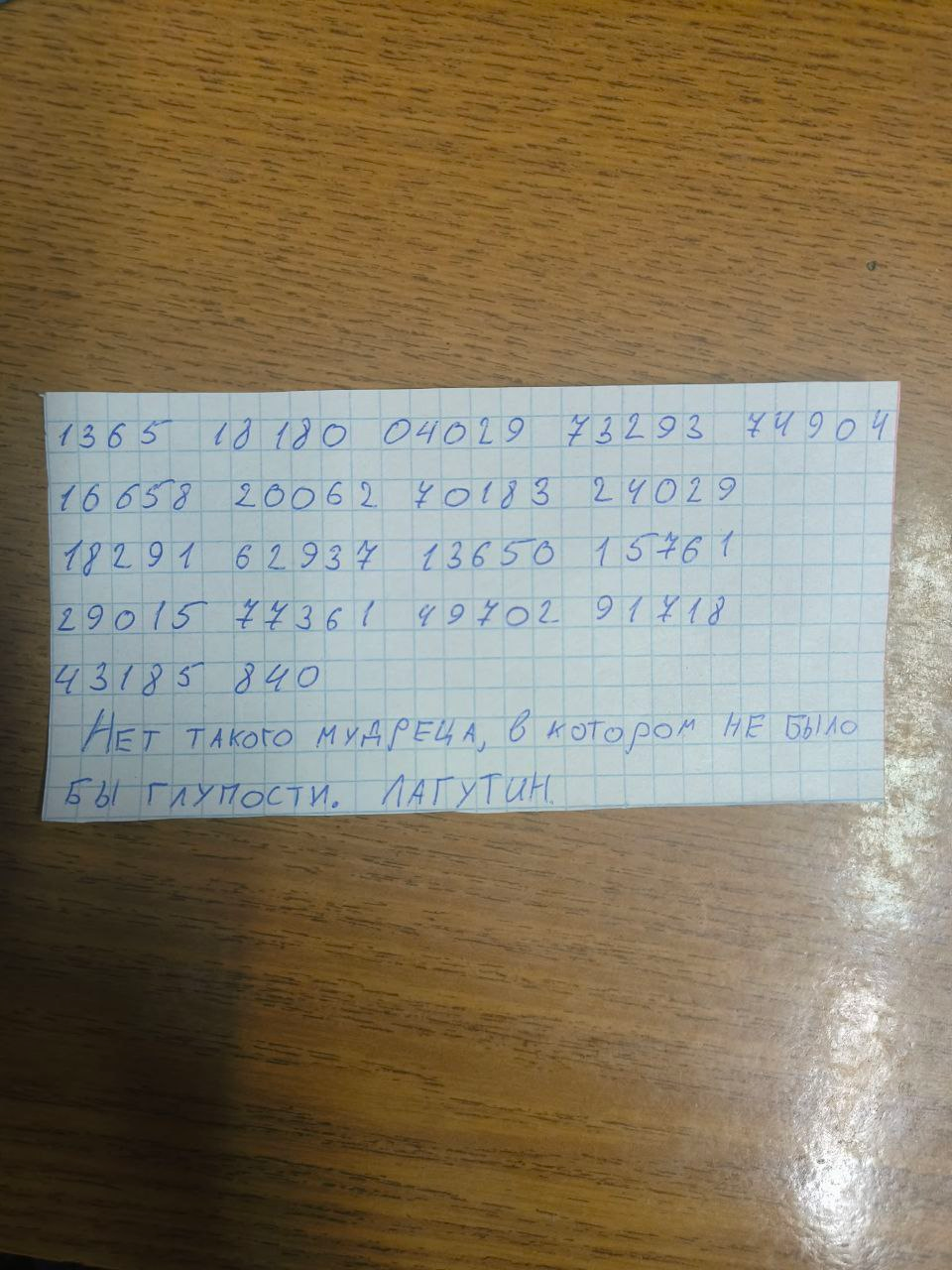
            print("Произведение p и q меньше длинны алфавиты (32)")

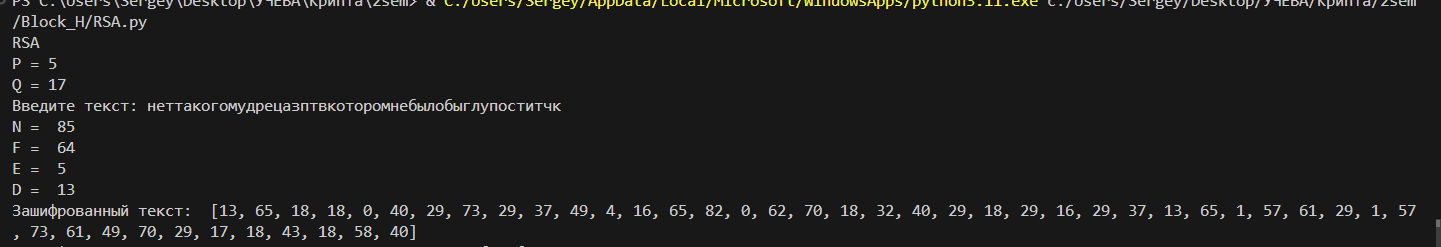
    else:

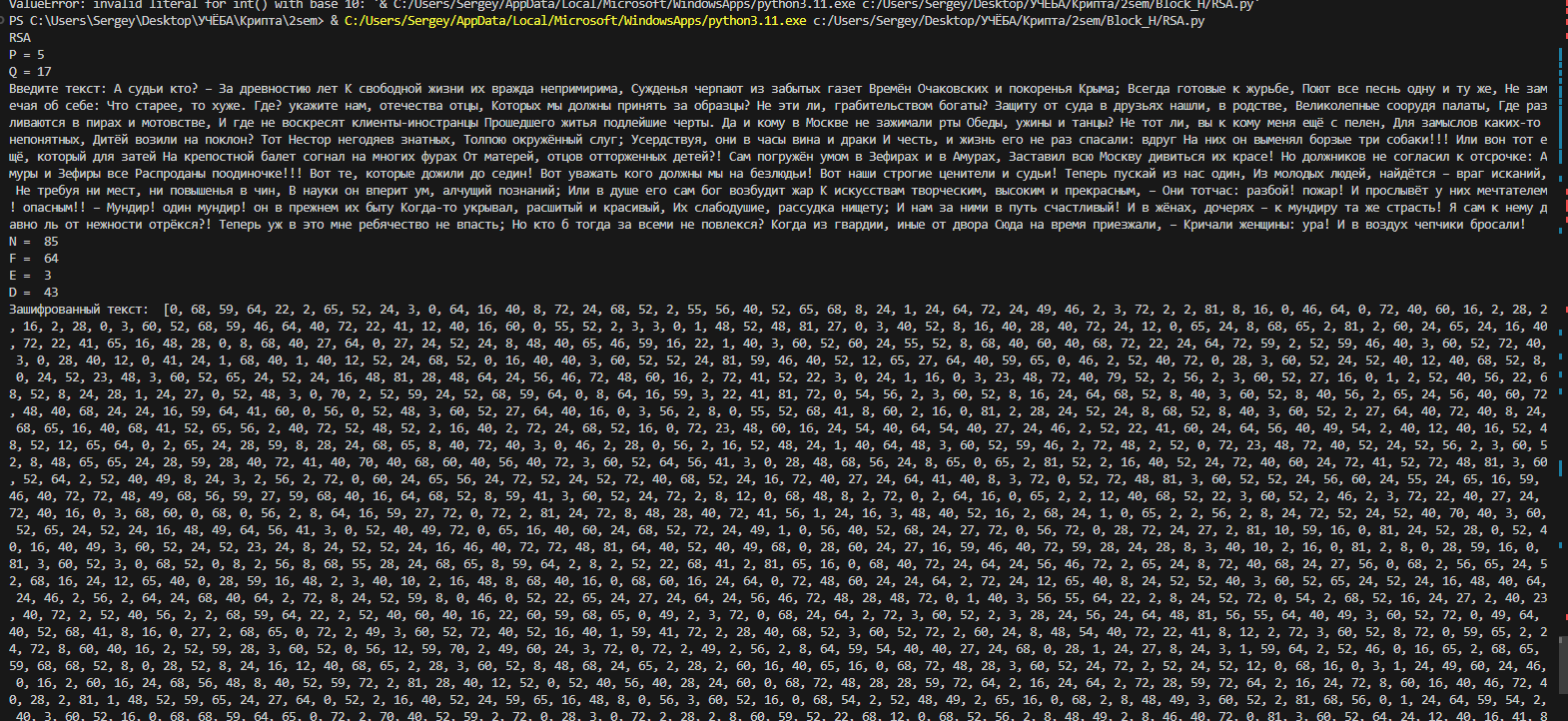
        print("Q не простое число")

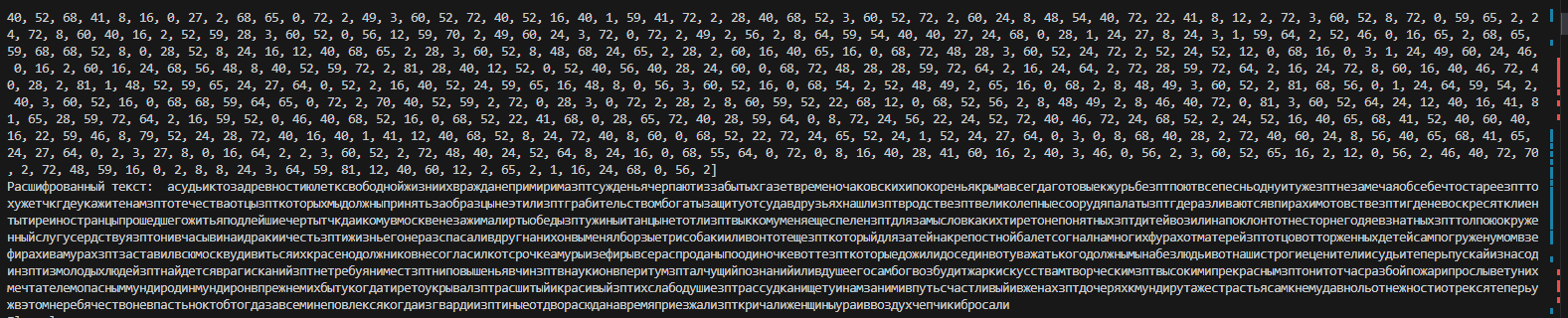
else:

    print("P не простое число")









Elgamal

### Elgamal ###

def encelga(s, P, mode):

    llst = ['а', 'б', 'в', 'г', 'д', 'е', 'ж', 'з', 'и', 'й', 'к', 'л', 'м', 'н', 'о', 'п', 'р', 'с', 'т', 'у', 'ф',

            'х', 'ц', 'ч', 'ш', 'щ', 'ъ', 'ы', 'ь', 'э', 'ю', 'я']

    s = pred(s)

    res = []

    X = random.randint(2, P - 1)

    G = random.randint(2, P - 1)

    print("X (секретный ключ)= ", X)

    print("G = ", G)

    Y = (G \*\* X) % P

    print("Y = ", Y)

    arr = [i for i in range(2,P-1) if math.gcd(i, P-1) == 1 ]

    mas = [3, 5, 7]

    count = 0

    for i in s:

        if mode == 1:

            k = mas[count%3]

        else:

            k = random.choice(arr)

        print(k)

        i = llst.index(i)

        a = G \*\* k % P

        res.append(a)

        b = Y \*\* k \* (i + 1) % P

        res.append(b)

        count += 1

    return res, P, X

def decelga(s, p, x):

    llst = ['а', 'б', 'в', 'г', 'д', 'е', 'ж', 'з', 'и', 'й', 'к', 'л', 'м', 'н', 'о', 'п', 'р', 'с', 'т', 'у', 'ф',

            'х', 'ц', 'ч', 'ш', 'щ', 'ъ', 'ы', 'ь', 'э', 'ю', 'я']

    res = ""

    for i in range(0, len(s), 2):

        a = s[i]

        b = s[i + 1]

        Mi = (b \* pow(a, p - 1 - x, p)) % p

        res += llst[Mi - 1]

    return res

# для Elgamal

print("Elgamal")

text = str(input("Введите текст : "))

mode = int(input("Проверка по карточке - 1, шифрование - 2: "))

P = int(input("P = "))

if P > 40 and is\_prime(P):

    res = encelga(text, P, mode)

    print("Зашифрованный текст = ", res[0])

    print("Расшифрованный текст = ", decelga(res[0], res[1], res[2]))

else:

    print("P должен быть больше 40 и простым числом")

