Лабораторная работа №3 «Асимметричное шифрование»

Задание 1.

Открытый ключ для варианта 6: 761.

Секретный ключ: 29.

Задание 2.

Реализация шифрования с помощью RSA и Диффи-Хеллман.

Код программы:

import random

import sympy

#шифр диффи-хеллман

def diffie\_hellman(p, g, private\_a, private\_b):

    public\_a = pow(g, private\_a, p)

    public\_b = pow(g, private\_b, p)

    shared\_key\_a = pow(public\_b, private\_a, p)

    shared\_key\_b = pow(public\_a, private\_b, p)

    assert shared\_key\_a == shared\_key\_b

    return shared\_key\_a

#генерация rsa

def generate\_rsa\_keys():

    p = sympy.randprime(100, 500)

    q = sympy.randprime(100, 500)

    n = p \* q

    phi = (p - 1) \* (q - 1)

    e = 65537

    d = pow(e, -1, phi)

    return (e, n), (d, n)

#шифр rsa

def rsa\_encrypt(message, public\_key):

    e, n = public\_key

    encrypted\_message = [pow(ord(char), e, n) for char in message]

    return encrypted\_message

#дешифр rsa

def rsa\_decrypt(encrypted\_message, private\_key):

    d, n = private\_key

    decrypted\_message = ''.join(chr(pow(char, d, n)) for char in encrypted\_message)

    return decrypted\_message

#главная фукнция для вызова функций

def main():

    print("Выберите метод: 1 - Диффи-Хеллман, 2 - RSA")

    choice = input("Ваш выбор: ")

    if choice == "1":

        p = int(input("Введите простое число p: "))

        g = int(input("Введите первообразный корень g: "))

        private\_a = int(input("Введите приватный ключ А: "))

        private\_b = int(input("Введите приватный ключ B: "))

        shared\_key = diffie\_hellman(p, g, private\_a, private\_b)

        print(f"Общий ключ: {shared\_key}")

    elif choice == "2":

        public\_key, private\_key = generate\_rsa\_keys()

        print(f"Открытый ключ: {public\_key}")

        print(f"Закрытый ключ: {private\_key}")

        message = input("Введите фамилию и инициалы: ")

        encrypted = rsa\_encrypt(message, public\_key)

        print(f"Зашифрованное сообщение: {encrypted}")

        decrypted = rsa\_decrypt(encrypted, private\_key)

        print(f"Расшифрованное сообщение: {decrypted}")

    else:

        print("Некорректный выбор")

if \_\_name\_\_ == "\_\_main\_\_":

    main()