Вариативная самостоятельная работа №3

**Тема:** Строки. Байты. Файлы

**Оборудование:** VS code, python3

**Задача 3.1**

**Постановка задачи:** Реализовать программу-игру «Угадай число», в которой для вывода на экран информации использовать метод format.

**Код программы:**

#Угадай число

import random

random.seed(version=2)

flag = False

start = int(input("Введите начало интервала "))

end = int(input("Введите конец интервала "))

guess\_num = random.randint(start,end)

while flag != True:

num = int(input("Введите предполагаемое число"))

s = "Ваше число {}.".format(num)

print(s)

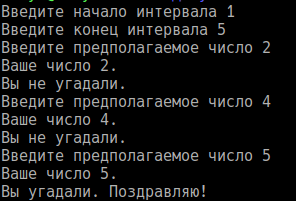
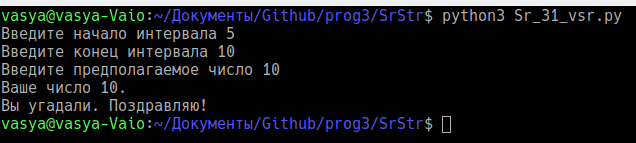
if guess\_num == num:

print("Вы угадали. Поздравляю!")

flag = True

else:

print("Вы не угадали.")

**Результат работы программы:**

**Задача 2.3**

**Постановка задачи:** Реализовать программу шифрующую строку, задаваемую пользователем, с помощью алгоритма шифрования, использующего сдвиг на определенное количество знаков (шифр Цезаря). Сдвиг задается пользователем.

**Код программы:**

# Cesar

import string

alphabet\_low = "абвгдеёжзийклмнопрстуфхцчшщъыьэюя"

alphabet\_up = "АБВГДЕЁЖЗИЙКДМНОПРСТУФХЦЧШЩЪЫЬЭЮЯ"

def cesar(s, step):

new\_s = ""

for char in s:

if char in string.ascii\_lowercase:

if string.ascii\_lowercase.find(char)+step < len(string.ascii\_lowercase):

new\_s += string.ascii\_lowercase[string.ascii\_lowercase.find(char)+step]

else:

add = abs(len(string.ascii\_lowercase)-string.ascii\_lowercase.find(char)-step)

new\_s += string.ascii\_lowercase[add]

elif char in string.ascii\_uppercase:

if string.ascii\_uppercase.find(char)+step < len(string.ascii\_uppercase):

new\_s += string.ascii\_uppercase[string.ascii\_uppercase.find(char)+step]

else:

add = abs(len(string.ascii\_uppercase)-string.ascii\_uppercase.find(char)-step)

new\_s += string.ascii\_uppercase[add]

elif char in alphabet\_low:

if alphabet\_low.find(char)+step < len(alphabet\_low):

new\_s += alphabet\_low[alphabet\_low.find(char)+step]

else:

add = abs(len(alphabet\_low)-alphabet\_low.find(char)-step)

new\_s += alphabet\_low[add]

elif char in alphabet\_up:

if alphabet\_up.find(char)+step < len(alphabet\_up):

new\_s += alphabet\_up[alphabet\_up.find(char)+step]

else:

add = abs(len(alphabet\_up)-alphabet\_up.find(char)-step)

new\_s += alphabet\_up[add]

else:

new\_s += char

print(new\_s)

def main():

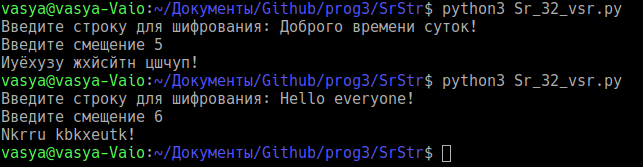
s = str(input("Введите строку для шифрования: "))

step = int(input("Введите смещение "))

cesar(s, step)

main()

**Результат работы программы:**

****