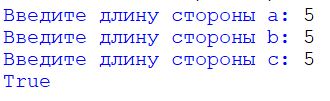
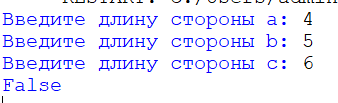
Акимкина Анна Лабораторная работа 2  
**Вариант 2**

**Задание 1. Логические выражения, условные операторы:**

Составить программу, которая выведет TRUE если заданное логическое выражение истинно и FALSE, если ложно, нельзя использовать условный оператор. Логическое выражение «Треугольник со сторонами а, b, с является равнобедренным».





КОД:

a = float(input("Введите длину стороны a: "))

b = float(input("Введите длину стороны b: "))

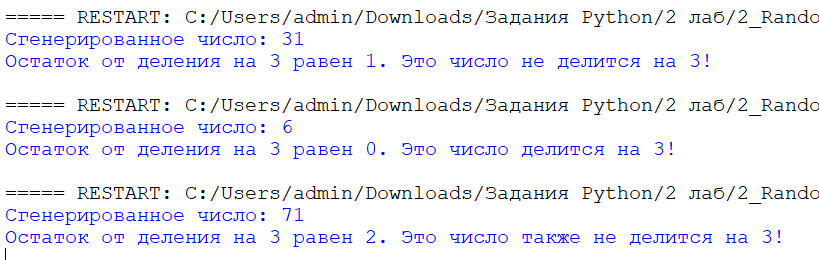
c = float(input("Введите длину стороны c: "))

is\_isosceles = (a == b) or (a == c) or (b == c)

print(is\_isosceles)

**Задание 2. Случайные числа + условный оператор**

Напишите программу, которая по полученному случайному числу, определит остаток от деления данного числа на 3 и в зависимости от значения остатка 0, 1, 2 вывести сообщение разные сообщения.



a = float(input("Введите длину стороны a: "))

b = float(input("Введите длину стороны b: "))

c = float(input("Введите длину стороны c: "))

is\_isosceles = (a == b) or (a == c) or (b == c)

print(is\_isosceles)

**Задание 3. Работа с операторами range и for**

Подсчитайте сумму значений S=-1+2 -3+4-…+(-1)n\*n



def calculate\_sum(n):

S = 0 # Начальная сумма

for i in range(1, n + 1):

if i % 2 == 0: # Проверяем, четное ли число

S += i # Если четное, добавляем его

else:

S -= i # Если нечетное, вычитаем его

return S

# Ввод значения n

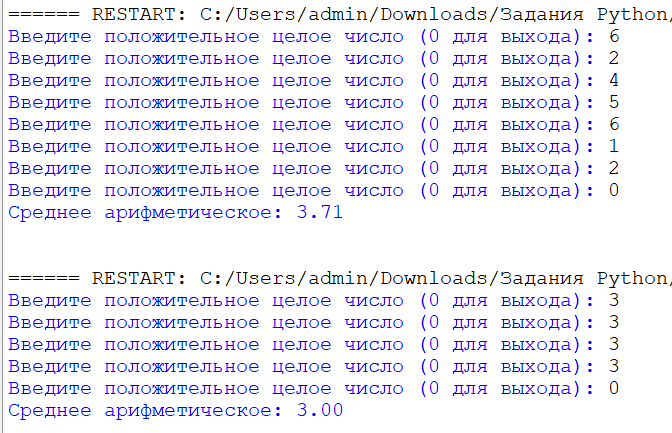
n = int(input("Введите значение n: "))

result = calculate\_sum(n)

print("Сумма S =", result)

**Задание 4. Принудительный выход из цикла, продолжение итерации**

Дана последовательность положительных целых чисел, за которыми следует 0, найти их среднее арифметическое чисел.



def calculate\_average():

total = 0 # Сумма всех положительных чисел

count = 0 # Счетчик положительных чисел

while True:

try:

number = int(input("Введите положительное целое число (0 для выхода): "))

if number == 0: # Проверяем на выход

break

if number < 0: # Игнорируем отрицательные числа

print("Пожалуйста, введите положительное целое число.")

continue

total += number # Накапливаем сумму

count += 1 # Увеличиваем счетчик

except ValueError: # Обработка некорректного ввода

print("Это не целое число. Попробуйте снова.")

# Проверка, чтобы избежать деления на ноль

if count > 0:

average = total / count

print(f"Среднее арифметическое: {average:.2f}")

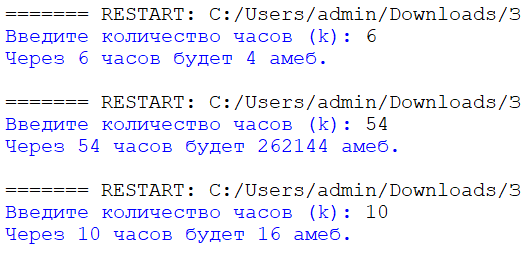
else:

print("Не было введено ни одного положительного числа.")

calculate\_average()

**Задание 5. Решите задание с применением цикла while**

Одноклеточная амеба каждые 3 часа делится на 2 клетки. Определить, сколько амеб будет через k часов



def ameba\_count(k):

# Начальное количество амеб

count = 1

# Количество делений

divisions = 0

while divisions \* 3 < k:

count \*= 2

divisions += 1

return count

k = int(input("Введите количество часов (k): "))

result = ameba\_count(k)

print(f"Через {k} часов будет {result} амеб.")

**Задание 6. Списки одномерные**

Найти среднее арифметическое элементов списка X, лежащих в интервале [1, 50].



X = [23, 45, 67, 2, 15, 51, 34, 8, 0, 49, 30]

filtered\_elements = [x for x in X if 1 <= x <= 50]

if filtered\_elements:

average = sum(filtered\_elements) / len(filtered\_elements)

print(f"Среднее арифметическое элементов списка X, лежащих в интервале [1, 50]: {average:.2f}")

else:

print("Нет элементов в интервале [1, 50]")