Министерство образования Республики Беларусь

Учреждение образования

«Белорусский государственный университет

информатики и радиоэлектроники»

Факультет информационных технологий и управления

Кафедра систем управления

Отчёт

по лабораторной работе №2

курс «Современные технологии программирования»

Выполнил Проверил

студент гр. 922403 В.А. Захарьев

Акимбек М.К

Минск 2022

**Лабораторная работа №2**

Тема: **Числа с плавающей точкой**

1.

Напишите функцию to\_float(num), которая преобразует любое число в число с плавающей точкой.

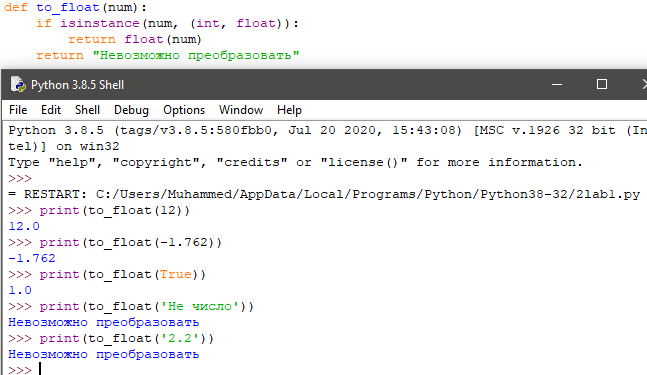
Если в качестве аргумента передан другой тип данных, она возвращает «Невозможно преобразовать».

**def** **to\_float**(num):

**if** isinstance(num, (int, float)):

**return** float(num)

**return** "Невозможно преобразовать"

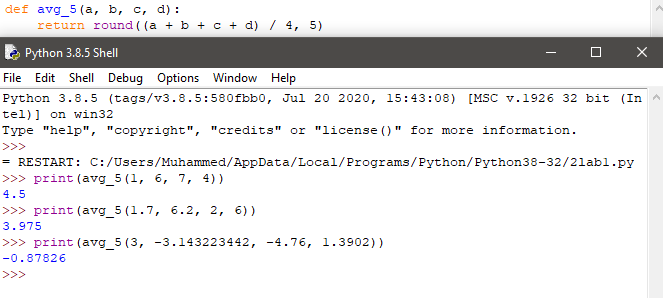


2.

Дано 4 числа.

Нужно написать функцию avg\_5(a, b, c, d), которая возвращает среднее арифметическое аргументов и округляет его до 5 знаков после запятой.

**def** **avg\_5**(a, b, c, d):

**return** round((a + b + c + d) / 4, 5) 

3.

Функция mul\_to\_int(a, b) может принимать целые или вещественные числа.

Если результат умножения аргументов не имеет значимых чисел после запятой, то она возвращает его в виде целого числа.

В противном случае – в виде float.

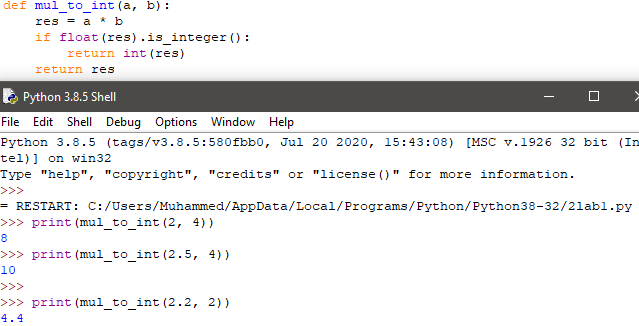
**def** **mul\_to\_int**(a, b):

res = a \* b

**if** float(res).is\_integer():

**return** int(res)

**return** res

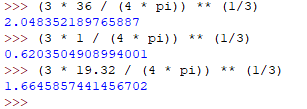


4.

Дан объем шара X куб. ед.

Найдите радиус фигуры.

Воспользуемся формулой объема шара:  
**V = 4/3 \* r \*\* 3 \* pi**  
Отсюда:  
**r = ((3 \* V / (4 \* pi)) \*\* (1/3))**



5. Напишите функцию округления round\_standard(num), принимающую число с плавающей точкой и округляющую его до целого числа в соответствии с правилами школьной математики.

**def** **round\_standard**(num):

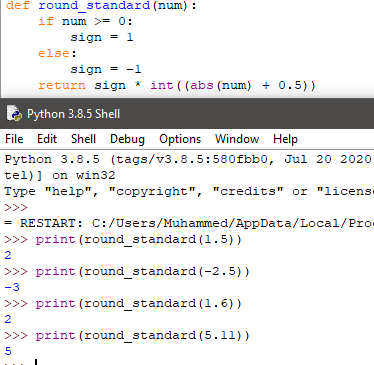
**if** num >= 0:

sign = 1

**else**:

sign = -1

**return** sign \* int((abs(num) + 0.5))



6. Так как в Python операции с вещественными числами могут давать неожиданные результаты (в частности, 0.1 + 0.2 не будет в точности равняться 0.3), стоит задача с этим как-то справляться.

Требуется написать функцию eqv(a, b, c), которая принимает 3 числа.

Числа a и b складываются.

Затем эта сумма сравнивается с числом “с” с определенной степенью точности.

Точность равняется 0.01 % от большего из чисел a и b.

Функция вернет **True**, если выполняется равенство, иначе **False**.

**def** **eqv**(a, b, c):

res = a + b

e = 0.01 / 100 *# Перевод процентов в долю*

tolerance = e \* max(abs(a), abs(b)) *# Находим значение отклонения*

**return** abs(res - c) <= tolerance *# Определяем, меньше ли разница, чем отклонение*

