

**Московский государственный технический
университет им. Н.Э. Баумана**

**Факультет «Информатика и системы управления»
Кафедра ИУ5 «Системы обработки информации и управления»**

Курс «Базовые компоненты интернет-технологий»

**Отчет по лабораторной работе №1
«Основные конструкции языка Python»**

Выполнил:
студент группы ИУ5-32Б:
Панов Герман Дмитриевич
Подпись и дата:

Проверил:
преподаватель каф. ИУ5
Гапанюк Ю.Е.
Подпись и дата:

Москва, 2021 г.

Цель лабораторной работы:

Изучение основных конструкций языка Python.

Постановка задачи:

Разработать программу для решения биквадратного уравнения.

1. Программа должна быть разработана в виде консольного приложения на языке Python.
2. Программа осуществляет ввод с клавиатуры коэффициентов A , B , C , вычисляет дискриминант и **ДЕЙСТВИТЕЛЬНЫЕ** корни уравнения (в зависимости от дискриминанта).
3. Коэффициенты A , B , C могут быть заданы в виде параметров командной строки (вариант задания параметров приведен в конце файла с примером кода). Если они не заданы, то вводятся с клавиатуры в соответствии с пунктом 2. Описание работы с параметрами командной строки.
4. Если коэффициент A , B , C введен или задан в командной строке некорректно, то необходимо проигнорировать некорректное значение и вводить коэффициент повторно пока коэффициент не будет введен корректно. Корректно заданный коэффициент - это коэффициент, значение которого может быть без ошибок преобразовано в действительное число

Текст кода:

```
import sys
import math

def get_coef(index, prompt):
    """
    Читаем коэффициент из командной строки или вводим с клавиатуры
    Args:
        index (int): Номер параметра в командной строке
        prompt (str): Приглашение для ввода коэффициента
    Returns:
        float: Коэффициент квадратного уравнения
    """

    try:
        # Попробуем прочесть коэффициент из командной строки
        coef_str = sys.argv[index]
        if index == 1:
            1 / float(sys.argv[1])

    except ZeroDivisionError:
        print('Коэффициент A не должен быть равен нулю, иначе это будет не биквадратное уравнение')
        print(prompt)
        coef_str = input()
        #coef = float(coef_str)

    # Пытаемся перевести строку в действительное число
    try:
```

```

        coef = float(coef_str)

    except (ValueError, TypeError):
        print('Введены некорректные данные!')
        print(prompt)
        coef_str = input()
        coef = float(coef_str)

    # coef = float(coef_str)
    return coef

def get_roots(a, b, c):
    """
    Вычисление корней биквадратного уравнения
    Args:
        a (float): коэффициент A
        b (float): коэффициент B
        c (float): коэффициент C
    Returns:
        list[float]: Список корней
    """
    result = []
    D = b * b - 4 * a * c

    if D == 0.0:
        root1 = math.sqrt(-b / (2.0 * a))
        if root1 == 0:
            result.append(abs(root1))
        else:
            root2 = -root1
            result.append(root1)
            result.append(root1)
    elif D > 0.0:
        sqD = math.sqrt(D)
        if ((-b + sqD) / (2.0 * a)) >= 0:
            root3 = round(math.sqrt((-b + sqD) / (2.0 * a)), 2)
            if root3 == 0:
                result.append(abs(root3))
            else:
                root4 = -root3
                result.append(root3)
                result.append(root4)
        if ((-b - sqD) / (2.0 * a)) >= 0:
            root5 = round(math.sqrt((-b - sqD) / (2.0 * a)), 2)
            if root5 == 0:
                result.append(abs(root5))
            else:
                root6 = -root5
                result.append(root5)
                result.append(root6)

    return result

def main():
    """
    Основная функция
    """

    a = get_coef(1, 'Введите коэффициент A:')
    b = get_coef(2, 'Введите коэффициент B:')
    c = get_coef(3, 'Введите коэффициент C:')

```

```

# Вычисление корней
roots = get_roots(a, b, c)

# Вывод корней
len_roots = len(roots)

if len_roots == 0:
    print('Нет корней')
elif len_roots == 1:
    print('Один корень: {}'.format(roots[0]))
elif len_roots == 2:
    print('Два корня: {}, {}'.format(roots[0], roots[1]))
elif len_roots == 3:
    print('Три корня: {}, {}, {}'.format(roots[0], roots[1], roots[2]))
elif len_roots == 4:
    print('Четыре корня: {}, {}, {}, {}'.format(roots[0], roots[1],
roots[2], roots[3]))

# Если сценарий запущен из командной строки
if __name__ == "__main__":
    main()

# Пример запуска
# python3 main.py 1 0 -4

```

Тестирование:

Ввод/ожидание:	Результат:
0 0 0 - Бесконечное число корней	Бесконечное число корней
0 0 1 – Корней нет	Корней нет
1 0 -4 - Два корня: 1.4142135623730951 и -1.4142135623730951	Два корня: 1.4142135623730951 и -1.4142135623730951
1 -4 0 – Три корня: 2.0 и -2.0 и 0	Три корня: 2.0 и -2.0 и 0
1 1 1 – Корней нет	Корней нет
0 1 -4 – Два корня: 2.0 и -2.0	Два корня: 2.0 и -2.0

Вывод:

В процессе выполнения данной лабораторной работы я смог познакомиться с синтаксисом языка Python, разобраться в принципе функционирования базовых функций , применить в действии конструкцию обработки исключений try – except.