



**Министерство науки и высшего образования
Российской Федерации Федеральное государственное
бюджетное образовательное учреждение высшего
образования
«Московский государственный технический
университет имени Н.Э. Баумана»**

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

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

Выполнила:
студентка группы ИУ5-35Б
Сухова Мария Андреевна

Подпись:

Дата:

Проверил:
Преподаватель кафедры ИУ5
Гапанюк Юрий Евгеньевич

Подпись:

Дата:

2022 г.

В ходе лабораторной работы №1 было необходимо выполнить следующее задание:

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 А, В, С, вычисляет дискриминант и **ДЕЙСТВИТЕЛЬНЫЕ** корни уравнения (в зависимости от дискриминанта).
3. Коэффициенты А, В, С 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.
4. Если коэффициент А, В, С 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 программы:

```
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:
        # Пробуем прочитать коэффициент из командной строки
        coef_str = sys.argv[index]
    except:
        # Вводим с клавиатуры
        print(prompt)
        coef_str = input()
    # Переводим строку в действительное число
    coef = 'coefficient'

    while not isinstance(coef, float):
        try:
            coef = float(coef_str)
        except:
            print('Проверьте корректность вводимых данных')
            print(prompt)
            coef_str = input()
    return coef

def get_roots(a, b, c):
    """
    Вычисление корней квадратного уравнения
    Args:
        a (float): коэффициент А
        b (float): коэффициент В
        c (float): коэффициент С
    Returns:
        list[float]: Список корней
    """
    answer = []
```

```

D = b * b - 4 * a * c
if D == 0.0:
    root = -b / (2.0 * a)
    if root > 0:
        t1 = math.sqrt(root)
        t2 = -math.sqrt(root)
        answer.append(t1)
        answer.append(t2)
    if root == 0:
        t = 0
        answer.append(t)
elif D > 0.0:
    sqD = math.sqrt(D)
    root1 = (-b + sqD) / (2.0 * a)
    if root1 > 0:
        t1 = math.sqrt(root1)
        t2 = -math.sqrt(root1)
        answer.append(t1)
        answer.append(t2)
    if root1 == 0:
        t = 0
        answer.append(t)
    root2 = (-b - sqD) / (2.0 * a)
    if root2 > 0:
        t3 = math.sqrt(root2)
        t4 = -math.sqrt(root2)
        answer.append(t3)
        answer.append(t4)
    if root2 == 0:
        t = 0
        answer.append(t)
return answer

def main():
    '''
    Основная функция
    '''
    a = 0
    while a == 0:
        print('Значение коэффициента А не может равняться ')
        a = get_coef(1, 'Введите коэффициент А:')
    b = get_coef(2, 'Введите коэффициент В:')
    c = get_coef(3, 'Введите коэффициент С:')
    # 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: {} и {}'.format(roots[0], roots[1]))
    elif len_roots == 3:
        print('Три корня: {}, {} и {}'.format(roots[0], roots[1], roots[2]))
    elif len_roots == 4:
        print('Четыре корня: {}, {}, {} и {}'.format(roots[0], roots[1],
roots[2], roots[3]))

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

```

```
# Пример запуска
# qr.py 1 0 -4
```

Экранные формы с примерами выполнения программы:

Пример 1:

```
C:\Users\suhov\AppData\Local\Programs\Python\Python310\python.exe C:/Users/suhov/Documents/3sem/bkit/lab1/main.py
Значение коэффициента A не может равняться 0
Введите коэффициент A:
1
Введите коэффициент B:
2
Введите коэффициент C:
3
Два корня: 1.0 и -1.0

Process finished with exit code 0
```

Пример 2:

```
C:\Users\suhov\AppData\Local\Programs\Python\Python310\python.exe C:/Users/suhov/Documents/3sem/bkit/lab1/main.py
Значение коэффициента A не может равняться 0
Введите коэффициент A:
0
Значение коэффициента A не может равняться 0
Введите коэффициент A:
1
Введите коэффициент B:
2
Введите коэффициент C:
3
Нет корней

Process finished with exit code 0
```

Пример 3:

```
C:\Users\suhov\AppData\Local\Programs\Python\Python310\python.exe C:/Users/suhov/Documents/3sem/bkit/lab1/main.py
Значение коэффициента A не может равняться 0
Введите коэффициент A:
1
Введите коэффициент B:
0
Введите коэффициент C:
0
Один корень: 0

Process finished with exit code 0
```

Пример 4:

```
C:\Users\suhov\AppData\Local\Programs\Python\Python310\python.exe C:/Users/suhov/Documents/3sem/bkit/lab1/main.py
Значение коэффициента A не может равняться 0
Введите коэффициент A:
1
Введите коэффициент B:
-3
Введите коэффициент C:
4
Четыре корня: 1.7320508075688772, -1.7320508075688772, 1.4142135623730951 и -1.4142135623730951

Process finished with exit code 0
```