



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

**(МГТУ им. Н.Э. Баумана)**

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

**Лабораторная работа №1**

**«Основные конструкции языка Python»**

**по дисциплине «Базовые компоненты интернет-технологий»**

**Выполнил:  
студент группы ИУ5-35Б  
Ищенко А.С.**

**2021 г.**

**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  $A$ ,  $B$ ,  $C$ , 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 программы:

Main.py

```
import sys
import math

def get_coef(index, prompt, flag):
    '''
    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
        coefStr = sys.argv[index]

        if coefStr == '0' and index == 1:
            raise

        coef = float(coefStr)
        return coef
    except:
        # Вводим с клавиатуры
        while True:
            print(prompt)
            coefStr = input()
            for i in range(len(coefStr)):
                if not(coefStr[i].isdigit() or coefStr[i] == '-'):
                    break
            else:
                break
        # Переводим строку в действительное число
        coef = float(coefStr)
        return coef

def get_roots(a, b, c):
    '''
    Вычисление корней квадратного 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 = -b / (2.0 * a)
        if root1 > 0:
```

```

        root1 = math.sqrt(root1)
        root2 = -root1
        result.append(root1)
        result.append(root2)
    if root1 == -0.0:
        root1 = 0.0
        result.append(root1)
elif D > 0.0:
    sqD = math.sqrt(D)
    root1 = (-b + sqD) / (2.0 * a)
    if root1 > 0:
        root1 = math.sqrt(root1)
        root2 = -root1
        result.append(root1)
        result.append(root2)
    if root1 == -0.0:
        root1 = 0.0
        result.append(root1)
    root3 = (-b - sqD) / (2.0 * a)
    if root3 > 0 and math.sqrt(root3) != -root1:
        root3 = math.sqrt(root3)
        root4 = -root3
        result.append(root3)
        result.append(root4)
    if root3 == -0.0:
        root3 = 0.0
        result.append(root3)
return result

def main():
    print()
    '''
    Основная функция
    '''
    flag = False
    a = get_coef(1, 'Введите коэффициент A:', flag)

    while a == 0:
        print('Коэффициент A не может быть равен 0')
        a = get_coef(1, 'Введите коэффициент A:', flag)

    b = get_coef(2, 'Введите коэффициент B:', flag)
    c = get_coef(3, 'Введите коэффициент C:', flag)
    # 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

```

## Результаты выполнения программы:

```
(venv) MacBook-Pro-Sergej:lab1 sergejadowlevic$ python main.py 0 0 1
```

Введите коэффициент A:

2

Нет корней

```
(venv) MacBook-Pro-Sergej:lab1 sergejadowlevic$ python main.py 1 d a
```

Введите коэффициент B:

2

Введите коэффициент C:

3

Нет корней

```
(venv) MacBook-Pro-Sergej:lab1 sergejadowlevic$ python main.py 4 2 16
```

Нет корней

```
(venv) MacBook-Pro-Sergej:lab1 sergejadowlevic$ python main.py 1 0 0
```

Один корень: 0.0

```
(venv) MacBook-Pro-Sergej:lab1 sergejadowlevic$ python main.py 1 -4 4
```

Два корня: 1.4142135623730951 и -1.4142135623730951

```
(venv) MacBook-Pro-Sergej:lab1 sergejadowlevic$ python main.py 1 -5 0
```

Три корня: 2.23606797749979, -2.23606797749979 и 0.0

```
(venv) MacBook-Pro-Sergej:lab1 sergejadowlevic$ python main.py 1 -7 12
```

Четыре корня: 2.0, -2.0, 1.7320508075688772 и -1.7320508075688772