



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

**Выполнил:
студент группы ИУ5-33Б
Слоква А. В.**

**Проверил:
Канев А.И.**

2021 г.

Оглавление

Постановка задачи	3
Текст программы.....	4
Экранные формы с примерами выполнения программы	6

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 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
    try:
        coef = float(coef_str)
    except:
        print("Invalid value")
        return get_coef(index, prompt)
    return coef

def roots_to_2sqr_roots(roots):
    res = []
    for root in roots:
        if root < 0:
            continue
            # 0 -0 fix
        if root == 0:
            res.append(0.0)
        else:
            res.append(abs(math.sqrt(root)))
            res.append(-abs(math.sqrt(root)))
    return res

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
    '''
    result = []
    if a == 0:
        if b == 0:
            return result
```

```

        result.append(-c / b)
    return result
D = b * b - 4 * a * c
if D == 0.0:
    root = -b / (2.0 * a)
    result.append(root)
elif D > 0.0:
    sqD = math.sqrt(D)
    root1 = (-b + sqD) / (2.0 * a)
    root2 = (-b - sqD) / (2.0 * a)
    result.append(root1)
    result.append(root2)
return result

def main():
    '''
    Основная функция
    '''
    a = get_coef(1, 'Введите коэффициент А:')
    b = get_coef(2, 'Введите коэффициент В:')
    c = get_coef(3, 'Введите коэффициент С:')
    if a == b == c == 0:
        print("Бесконечное число корней")
        return
    # Вычисление корней
    roots = get_roots(a, b, c)
    print(roots)
    roots = roots_to_2sqr_roots(roots)
    # 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 {} и {}'.format(roots[0], roots[1], roots[2]))
    elif len_roots == 4:
        print('Четыре корня: {} и {} и {} 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

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])) # формат - для подстановки
elif len_roots == 2:
    print('Два корня: {} и {}'.format(roots[0], roots[1])) # формат -

```

```

для подстановки
    elif len_roots == 3:
        print('Три корня: {}, {} и {}'.format(roots[0], roots[1], roots[2]))
# формат - для подстановки
    elif len_roots == 4:
        print('Четыре корня: {}, {}, {} и {}'.format(roots[0], roots[1],
roots[2], roots[3]))

    input("Press Enter to continue...")

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

# Пример запуска
# qr.py 1 0 -4

```

Экранные формы с примерами выполнения программы:

```

C:\Users\User\AppData\Local\Programs\Python\Python39\python.exe C:/Users/User/Desktop/LABS/BKIT_2021/code/lab1_code/qr.py
Введите коэффициент A:
invalid
Invalid value
Введите коэффициент A:
invalid
Invalid value
Введите коэффициент A:|
invalid
Invalid value
Введите коэффициент A:

```

```

C:\Users\User\AppData\Local\Programs\Python\Python39\python.exe C:/Users/User/Desktop/LABS/BKIT_2021/code/lab1_code/qr.py
Введите коэффициент A:
2
Введите коэффициент B:
-2
Введите коэффициент C:
-8
Два корня: 1.600485180440241 и -1.600485180440241

Process finished with exit code 0

```