

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

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

Курс «Парадигмы и конструкции языков программирования»

Отчет по лабораторной работе № 1.1  
«Основные конструкции языка Python. Дополнительное задание № 1»

Выполнил:  
студент группы ИУ5-34Б  
Бромберг Е.А.

Проверил:  
преподаватель каф. ИУ5  
Нардид А.Н.

Москва, 2024 г.

## Постановка задачи

### 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.
5. **Дополнительное задание 1 (\*)**. Разработайте две программы на языке Python - одну с применением процедурной парадигмы, а другую с применением объектно-ориентированной парадигмы (см. лабораторную работу № 1).

## Текст программы

Main.py

```
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 = float(coef_str)
    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:
        if (-b / (2.0 * a)) > 0.0:
            root1 = -math.sqrt(-b / (2.0 * a))
            root2 = math.sqrt(-b / (2.0 * a))
            result.append(root1)
            result.append(root2)
        if (-b / (2.0 * a)) == 0.0:
            root1 = 0.0
            result.append(root1)
    elif D > 0.0:
        sqD = math.sqrt(D)
        if ((-b - sqD) / (2.0 * a)) > 0.0:
            root1 = -math.sqrt((-b - sqD) / (2.0 * a))
            root2 = math.sqrt((-b - sqD) / (2.0 * a))
            result.append(root1)
            result.append(root2)
```

```

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

def main():
    '''
    Основная функция
    '''
    while True:
        try:
            a = get_coef(1, 'Введите коэффициент А:')
            if a == 0:
                print("Коэффициент А не должен равняться 0!")
                sys.argv = []
                continue
            b = get_coef(2, 'Введите коэффициент В:')
            c = get_coef(3, 'Введите коэффициент С:')
            break
        except:
            print("Введите числа!")
            sys.argv = []

    # 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()

```

## Экранные формы с примерами выполнения программы

Командная строка

```
C:\Users\user\PycharmProjects\PCPL_lab1.1>python main.py hi hi r2
Введите числа!
Введите коэффициент A:
0
Коэффициент A не должен равняться 0!
Введите коэффициент A:
1
Введите коэффициент B:
0
Введите коэффициент C:
-4
Два корня: -1.4142135623730951 и 1.4142135623730951
```

Командная строка

```
C:\Users\user\PycharmProjects\PCPL_lab1.1>python main.py 0 1 2
Коэффициент A не должен равняться 0!
Введите коэффициент A:
1
Введите коэффициент B:
0
Введите коэффициент C:
4
Нет корней
```

Командная строка

```
C:\Users\user\PycharmProjects\PCPL_lab1.1>python main.py
Введите коэффициент A:
1
Введите коэффициент B:
-10
Введите коэффициент C:
9
Четыре корня: -1.0, 1.0, -3.0 и 3.0
C:\Users\user\PycharmProjects\PCPL_lab1.1>
```