

**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» Лабораторная работа №1

Выполнил:
Студент ИУ5-34Б

Беккиев Р.И.

Подпись и дата:

Проверил:
Преподаватель каф. ИУ5

Гапанюк Ю. Е.

Подпись и дата:

Москва, 2023 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.
5. Дополнительное задание 1 (*). Разработайте две программы на языке Python - одну с применением процедурной парадигмы, а другую с применением объектно-ориентированной парадигмы.
6. Дополнительное задание 2 (*). Разработайте две программы - одну на языке Python, а другую на любом другом языке программирования (кроме C++).

Текст программы:

```
import math

def get_coefficient(prompt):
    while True:
        try:
            coefficient = float(input(prompt))
            return coefficient
        except ValueError:
            print("Ошибка: Введите корректное значение.")

def solve_biquadratic_equation(a, b, c):
    discriminant = b ** 2 - 4 * a * c

    if discriminant > 0:
        x1 = (-b + math.sqrt(discriminant)) / (2 * a)
        x2 = (-b - math.sqrt(discriminant)) / (2 * a)
        return x1, x2
    elif discriminant == 0:
        x = -b / (2 * a)
        return x,
    else:
        return None

def main():
    a = get_coefficient("Введите коэффициент A: ")
    b = get_coefficient("Введите коэффициент B: ")
```

```

c = get_coefficient("Введите коэффициент C: ")

roots = solve_biquadratic_equation(a, b, c)

if roots is not None:
    print("Корни уравнения:", ", ".join(map(str, roots)))
else:
    print("Уравнение не имеет действительных корней.")

main()

```

Результаты выполнения:

```

/Users/razzle/PycharmProjects/lab.1/venv/bin/python /Users/razzle/PycharmPro
Введите коэффициент A: 2
Введите коэффициент B: 10
Введите коэффициент C: 3
Корни уравнения: -0.320550528229663, -4.6794494717703365

Process finished with exit code 0
|

```

```

main x
/Users/razzle/PycharmProjects/lab.1/venv/bin/python /Users/razzle/PycharmProjects/
Введите коэффициент A: 4
Введите коэффициент B: 7
Введите коэффициент C: 9
Уравнение не имеет действительных корней.

Process finished with exit code 0

```

```

/Users/razzle/PycharmProjects/lab.1/venv/bin/python /Users/razzle/PycharmProjects/lab
Введите коэффициент A: 10
Введите коэффициент B: 100
Введите коэффициент C: 1000
Уравнение не имеет действительных корней.

Process finished with exit code 0

```

/Users/razzle/PycharmProjects/lab.1/venv/bin/python /Users/razzle/PycharmProjects/lab.1

Введите коэффициент A: 10

Введите коэффициент B: 20

Введите коэффициент C: 6

Корни уравнения: -0.3675444679663241, -1.632455532033676

Process finished with exit code 0

|