

Защищено:
Гапанюк Ю.Е.

"__" 2025 г.

Демонстрация:
Гапанюк Ю.Е.

"__" 2025 г.

**Отчет по лабораторной работе № 1 по курсу
Парадигмы и конструкции языков программирования
ГУИМЦ**

Тема работы: "Основные конструкции языка Python."

5
(количество листов)

ИСПОЛНИТЕЛЬ:

студент группы ИУ5Ц-54Б

Щетинин Д.С.

(подпись)

"__" 2025 г.

1. Тема и задание для выполнения лабораторной работы.

Тема работы: "Изучение основных конструкций языка Python."

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.

$$Ax^4 + Bx^2 + C = 0$$

2. Листинг программы

```
# Импорт необходимых модулей
import sys # Для работы с параметрами командной строки
import math # Для математических операций (например, извлечения корня)

def get_coef(index, prompt):
    """
    Функция для безопасного ввода коэффициентов
    Пытается получить коэффициент из командной строки, если не получается - запрашивает ввод с
    клавиатуры
    Повторяет ввод до тех пор, пока не будет введено корректное число
    """

    Аргументы:
        index (int): индекс параметра в sys.argv (1 для A, 2 для B, 3 для C)
        prompt (str): приглашение для ввода

    Возвращает:
        float: введенный коэффициент
```

Возможные результаты:

- Возвращает число float при успешном вводе
- При ошибке ввода выводит сообщение и повторяет запрос

while True: # Бесконечный цикл, пока не получим корректный ввод

```
try:
    # Пробуем получить коэффициент из командной строки
    if len(sys.argv) > index: # Проверяем, есть ли параметр с таким индексом
        coef_str = sys.argv[index]
        print(prompt + ' ' + coef_str) # Выводим приглашение и значение из командной строки
    else:
        # Если параметра нет, запрашиваем ввод с клавиатуры
        print(prompt)
        coef_str = input()
    # Пробуем преобразовать строку в число
    coef = float(coef_str)
    return coef # Возвращаем коэффициент, если преобразование успешно
except ValueError: # Ошибка при преобразовании строки в число
```

```

print("Ошибка: введите корректное число")
continue # Повторяем цикл
except IndexError: # Ошибка при обращении к несуществующему индексу в sys.argv
    print(prompt)
    coef_str = input()
    try:
        coef = float(coef_str)
        return coef
    except ValueError:
        print("Ошибка: введите корректное число")
        continue

```

def get_biquadratic_roots(a, b, c):
 """
 Функция для вычисления действительных корней биквадратного уравнения $ax^4 + bx^2 + c = 0$
 """

Аргументы:

a (float): коэффициент при x^4
b (float): коэффициент при x^2
c (float): свободный член

Возращает:

list[float]: список действительных корней уравнения

Возможные результаты:

1. При $a = b = c = 0$: ["бесконечное множество решений"] (любое x - решение)
2. При $a = b \neq 0, c \neq 0$: [] (нет решений)
3. При $a = 0$: решает квадратное уравнение $bx^2 + c = 0$
4. При $D < 0$: [] (нет действительных корней)
5. При $D \geq 0$: находит все возможные действительные корни

"""

result = [] # Список для хранения корней

```

# Случай, когда a = 0 (уравнение становится квадратным относительно x^2)
if a == 0:
    # Если и b = 0
    if b == 0:
        if c == 0:
            # Уравнение 0 = 0 - бесконечное множество решений
            return ["бесконечное множество решений"]
        else:
            # Уравнение c = 0, где c ≠ 0 - нет решений
            return []
    else:
        # Решаем линейное уравнение bx^2 + c = 0
        x_squared = -c / b
        if x_squared >= 0: # Проверяем, чтобы выражение под корнем было неотрицательным
            root = math.sqrt(x_squared)
            result.append(root)
            result.append(-root) # Добавляем оба корня (±)
return result

# Вычисляем дискриминант для квадратного уравнения относительно y = x^2
D = b * b - 4 * a * c

# Если дискриминант отрицательный - нет действительных корней
if D < 0:
    return result

```

```

# Вычисляем корни уравнения относительно y = x^2
y1 = (-b + math.sqrt(D)) / (2 * a)
y2 = (-b - math.sqrt(D)) / (2 * a)

# Проверяем первый корень y1 на неотрицательность
if y1 >= 0:
    root1 = math.sqrt(y1)
    root2 = -math.sqrt(y1)
    result.append(root1)
    result.append(root2)

# Проверяем второй корень y2 на неотрицательность и уникальность
if y2 >= 0 and not math.isclose(y1, y2, rel_tol=1e-9):
    root3 = math.sqrt(y2)
    root4 = -math.sqrt(y2)
    result.append(root3)
    result.append(root4)

return result

def main():
    """
    Основная функция программы
    Обрабатывает ввод коэффициентов, вычисляет корни и выводит результат
    """

    print("Решение биквадратного уравнения вида Ax^4 + Bx^2 + C = 0")

    # Получаем коэффициенты
    a = get_coef(1, 'Введите коэффициент A:')
    b = get_coef(2, 'Введите коэффициент B:')
    c = get_coef(3, 'Введите коэффициент C:')

    # Вычисляем корни уравнения
    roots = get_biquadratic_roots(a, b, c)

    # Обрабатываем и выводим результаты
    if len(roots) == 0:
        print('Действительных корней нет')
    elif roots[0] == "бесконечное множество решений":
        print('Бесконечное множество решений (любое x является решением)')
    else:
        # Удаляем дубликаты (если y1 = y2) и сортируем корни по абсолютному значению
        unique_roots = sorted(list(set(roots)), key=lambda x: abs(x))
        print('Найдены следующие действительные корни:')
        for root in unique_roots:
            print('x = {:.6f}'.format(root)) # Выводим с точностью до 6 знаков после запятой

    # Стандартная проверка для запуска из командной строки
if __name__ == "__main__":
    main()

```

3. Результаты работы программы

C:\Program Files\Python313\python.exe" C:\Users\Денис\Desktop\Python-1lab\lab1.py

Решение биквадратного уравнения вида $Ax^4 + Bx^2 + C = 0$

Введите коэффициент A:

-5

Введите коэффициент B:

2

Введите коэффициент C:

15

Найдены следующие действительные корни:

x = 1.394116

x = -1.394116

Process finished with exit code 0