|  |  |  |
| --- | --- | --- |
| Защищено:  Нардид А.Н.    "\_\_"\_\_\_\_\_\_\_\_\_\_\_\_\_2024 г. |  | Демонстрация:  Нардид А.Н.  "\_\_"\_\_\_\_\_\_\_\_\_\_\_\_\_2024 г. |

**Отчет по лабораторной работе № 1 по курсу**

**Парадигмы и конструкции языков программирования**

**ГУИМЦ**

#### Тема работы: "Основные конструкции языка Python."

3

(количество листов)

|  |  |
| --- | --- |
| ИСПОЛНИТЕЛЬ: |  |
| студент группы ИУ5Ц-51Б | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |
|  | (подпись) |
| Омарова З.Р. | "\_\_"\_\_\_\_\_\_\_\_\_\_\_\_\_2024 г. |

Москва, МГТУ - 2024 г.

# Тема и задание для выполнения лабораторной работы.

Тема работы: "Изучение основных конструкций языка Python.". Необходимо написать программу, которая решает квадратное уравнение вида:

# Листинг программы

import sys

def get\_coefficient(prompt):

    while True:

        try:

            value = float(input(prompt))

            return value

        except ValueError:

            print("Некорректное значение. Пожалуйста, введите действительное число.")

def solve\_biquadratic(A, B, C):

    # Вычисление дискриминанта для квадратного уравнения

    D = B\*\*2 - 4\*A\*C

    print(f"Дискриминант D = {D}")

    if D > 0:

        y1 = (-B + D\*\*0.5) / (2\*A)

        y2 = (-B - D\*\*0.5) / (2\*A)

        return [y1, y2]

    elif D == 0:

        y = -B / (2\*A)

        return [y]

    else:

        return []

def find\_x\_from\_y(y):

    if y < 0:

        return []  # Нет действительных корней

    else:

        return [y\*\*0.5, -y\*\*0.5]

def main():

    if len(sys.argv) == 4:

        try:

            A = float(sys.argv[1])

            B = float(sys.argv[2])

            C = float(sys.argv[3])

        except ValueError:

            print("Некорректные значения в командной строке. Пожалуйста, введите коэффициенты заново.")

            A = get\_coefficient("Введите коэффициент A: ")

            B = get\_coefficient("Введите коэффициент B: ")

            C = get\_coefficient("Введите коэффициент C: ")

    else:

        A = get\_coefficient("Введите коэффициент A: ")

        B = get\_coefficient("Введите коэффициент B: ")

        C = get\_coefficient("Введите коэффициент C: ")

    # Решение биквадратного уравнения

    y\_roots = solve\_biquadratic(A, B, C)

    all\_x\_roots = []

    for y in y\_roots:

        all\_x\_roots.extend(find\_x\_from\_y(y))

    if all\_x\_roots:

        print(f"Корни биквадратного уравнения: {all\_x\_roots}")

    else:

        print("Уравнение не имеет действительных корней.")

if \_\_name\_\_ == "\_\_main\_\_":

    main()

# Результаты работы программы

Введите коэффициент A: -2

Введите коэффициент B: 4

Введите коэффициент C: 5

Дискриминант D = 56.0

Корни биквадратного уравнения: [1.6943519980768371, -1.6943519980768371]