

**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
«Основные конструкции языка Python»**

Выполнил:
студент группы ИУ5-36Б
Ковалев Е.А.

Проверил:
Преподаватель каф. ИУ5
Нардид А.Н.

2024 г.

Описание задания

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.
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

def get_coef(prompt, zero=True):
    while True:
        c = input(prompt)
        try:
            res = float(c)
            if zero or res != 0:
                return res
            else:
                raise
        except:
            print("Invalid input!")

def get_coefs():
    A = get_coef("Enter coefficient A (non-zero): ", zero=False)
    B = get_coef("Enter coefficient B: ")
    C = get_coef("Enter coefficient C: ")
    return A, B, C

def calc_roots(A, B, C):
    D = B**2 - 4*A*C
    print("D = ", D)

    if D < 0:
        return []
    elif D == 0:
        x = -B / (2 * A)
        return [x**0.5, -x**0.5] if x > 0 else ([0] if x == 0 else [])
    else:
        x1 = (-B + D**0.5) / (2 * A)
        x2 = (-B - D**0.5) / (2 * A)

        roots = []
        if x1 >= 0:
            roots.extend([x1**0.5, -x1**0.5] if x1 > 0 else [0])
        if x2 >= 0:
            roots.extend([x2**0.5, -x2**0.5] if x2 > 0 else [0])

        return roots

def main():
    if len(sys.argv) == 4:
        try:
            A = float(sys.argv[1])
            if A == 0:
                raise
        except:
```

```
        B = float(sys.argv[2])
        C = float(sys.argv[3])
    except:
        A, B, C = get_coefs()
    else:
        A, B, C = get_coefs()

    res = calc_roots(A, B, C)

    if res:
        print("Real solutions of the equation:")
        for root in res:
            print(root)
    else:
        print("No real solutions")

if __name__ == "__main__":
    main()
```

Экранные формы с примерами выполнения программы

```
C:\Users\111\Desktop>python3 lab1.py 1 11 10
D = 81.0
No real solutions

C:\Users\111\Desktop>python3 lab1.py
Enter coefficient A (non-zero): 0
Invalid input!
Enter coefficient A (non-zero): qwerty
Invalid input!
Enter coefficient A (non-zero): 10
Enter coefficient B: 0
Enter coefficient C: 0
D = 0.0
Real solutions of the equation:
0

C:\Users\111\Desktop>python3 lab1.py
Enter coefficient A (non-zero): 1
Enter coefficient B: -2
Enter coefficient C: -8
D = 36.0
Real solutions of the equation:
2.0
-2.0

C:\Users\111\Desktop>python3 lab1.py -4 16 0
D = 256.0
Real solutions of the equation:
0
2.0
-2.0

C:\Users\111\Desktop>python3 lab1.py 1 -10 9
D = 64.0
Real solutions of the equation:
3.0
-3.0
1.0
-1.0
```