

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

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

Курс «Базовые компоненты интернет-технологий»

Отчет по лабораторной работе №1  
**«Основные конструкции языка Python.»**

Выполнил: Тянутов  
Александр Дмитриевич  
студент группы : ИУ5-31Б

Проверил:  
  
преподаватель каф.  
ИУ5

Подпись и дата:

Подпись и дата:

Москва, 2022 г.

## Постановка задачи

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 ввести 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
import argparse

from math import sqrt

def get_sqrtroot(root):
    """
    Get square of root without errors
    Args:
        root(float)
    Returns:
        float/None
    """
    if root < 0.0:
        return None
    else:
        return sqrt(root)
```

```

def get_roots(a, b, c):
    """
    Calculating roots for equation  $a*x^4+b*x^2+c=0$ 

    Args:
        a(float): coeff of A
        b(float): coeff of B
        c(float): coeff of C

    Returns:
        list[float]: list of roots
    """
    res = []
    d = b*b-4*a*c
    if d == 0.0:
        root = -b / (2*a)
        sqroot = get_sqroot(root)
        if sqroot is not None:
            res.append(sqroot)
            res.append(-sqroot)
    elif d > 0.0:
        sqd = sqrt(d)
        root1 = (-b-sqd)/(2*a)
        root2 = (-b+sqd)/(2*a)
        sqroot1 = get_sqroot(root1)
        sqroot2 = get_sqroot(root2)
        if sqroot1 is not None:
            res.append(sqroot1)
            res.append(-sqroot1)
        if sqroot2 is not None and sqroot2 != sqroot1:
            res.append(sqroot2)
            res.append(-sqroot2)
    res = list(map(lambda x: round(x, 2), res))
    return res

def parses_args():
    """
    Parser arguments
    """
    parser = argparse.ArgumentParser(description='Solves  $ax^4+bx^2+c=0$  where  $a \neq 0$ ')

```

```

    parser.add_argument('num', nargs=3, type=float, help='Three int/float numbers ex: 1 3.5 -8.3')
    parser.parse_args()

def main():
    """
    main function
    """
    args = [float(x) for x in sys.argv[1:4]]
    ans = get_roots(args[0], args[1], args[2])
    len_ans = len(ans)
    if len_ans == 0:
        print('No roots')
    else:
        print('{} root '.format(len_ans), str(ans)[1:-1])

if __name__ == '__main__':
    parses_args()
    main()

```

## Анализ результатов

```

PS D:\Projects\BKIT\lab_1> python .\lab1.py
usage: lab1.py [-h] num num num
lab1.py: error: the following arguments are required: num
PS D:\Projects\BKIT\lab_1> python .\lab1.py 1 2 3
No roots
PS D:\Projects\BKIT\lab_1> python .\lab1.py 5 -2 3
No roots
PS D:\Projects\BKIT\lab_1> python .\lab1.py 5 -2
usage: lab1.py [-h] num num num
lab1.py: error: the following arguments are required: num
PS D:\Projects\BKIT\lab_1> python .\lab1.py 1 -10 9
4 root  1.0, -1.0, 3.0, -3.0

```

## Вывод

Освоил основные конструкции языка python(переменные, создание функций, работа с методами, list)