

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

Факультет «Радиотехнический»

Кафедра ИУ5 «Системы обработки информации и управления»

Курс «Базовые компоненты интернет-технологий»

Отчет по лабораторной работе №1

«ОСНОВНЫЕ КОНСТРУКЦИИ ЯЗЫКА PYTHON»

Выполнил:

Студент группы РТ5-21Б

Фруктин А.Е.

Подпись и дата:

Проверил:

Преподаватель каф. ИУ5

Гапанюк Ю.Е.

Подпись и дата:

Москва, 2022 г.

## **Оглавление**

1. Описание задания.....	3
2. Текст программы.....	3
3. Экранные формы с примерами выполнения программы.....	6

## 1. 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. 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.

## 2. Текст программы

```
import sys
import math
def get_coef(index, prompt):
    """
    Читаем коэффициент из командной строки или вводим с клавиатуры
    Args:
        index (int): Номер параметра в командной строке
        prompt (str): Приглашение для ввода коэффициента
    Returns:
        float: Коэффициент квадратного уравнения
    """
    try:
        # Пробуем прочитать коэффициент из командной строки
        coef_str = sys.argv[index]
        coef = float(coef_str)

        return coef
    except:
```

```

    # Вводим с клавиатуры
    print(prompt)
    coef_str = input()
# Переводим строку в действительное число
    try:
        coef = float(coef_str)
        return coef
    except:
        return math.inf

def get_roots(a, b, c):
    """
    Вычисление корней квадратного уравнения
    Args:
        a (float): коэффициент А
        b (float): коэффициент В
        c (float): коэффициент С
    Returns:
        list[float]: Список корней
    """
    result = []
    D = b*b - 4*a*c
    if D == 0.0:
        timeroot = -b / (2.0*a)
        if timeroot==0.0:
            result.append(timeroot)
        elif timeroot>0.0:
            root1=math.sqrt(timeroot)
            root2=-math.sqrt(timeroot)
            result.append(root1)
            result.append(root2)
    elif D > 0.0:
        sqD = math.sqrt(D)
        timeroot1 = (-b + sqD) / (2.0*a)
        timeroot2 = (-b - sqD) / (2.0*a)
        if timeroot1==0.0:
            result.append(timeroot1)
        elif timeroot1>0.0:
            root1=math.sqrt(timeroot1)
            root2=-math.sqrt(timeroot1)
            result.append(root1)
            result.append(root2)
        if timeroot2==0.0:

```

```

        result.append(timeroot2)
    elif timeroot2>0.0:
        root1=math.sqrt(timeroot2)
        root2=-math.sqrt(timeroot2)
        result.append(root1)
        result.append(root2)
    return result

def main():
    """
    Основная функция
    """
    a=math.inf
    b=math.inf
    c=math.inf
    while a==math.inf or a==0:
        a = get_coef(1, 'Введите коэффициент А:')
    while b==math.inf:
        b = get_coef(2, 'Введите коэффициент В:')
    while c==math.inf:
        c = get_coef(3, 'Введите коэффициент С:')
    # Вычисление корней
    roots = get_roots(a,b,c)
    # Вывод корней
    len_roots = len(roots)
    if len_roots == 0:
        print('Нет корней')
    elif len_roots == 1:
        print('Один корень: x1={}'.format(roots[0]))
    elif len_roots == 2:
        print('Два корня: x1={ }; x2={}'.format(roots[0], roots[1]))
    elif len_roots == 3:
        print('Три корня: x1={ }; x2={ }; x3={}'.format(roots[0], roots[1], roots[2]))
    elif len_roots == 4:
        print('Четыре корня: x1={ }; x2={ }; x3={ }; x4={}'.format(roots[0], roots[1],
        roots[2], roots[3]))

# Если сценарий запущен из командной строки
if __name__ == "__main__":
    main()


# Пример запуска
# qr.py 1 0 -4

```

### 3. Экранные формы с примерами выполнения программы

```
Python 3.8.2 (tags/v3.8.2:7b3ab59, Feb 25 2020, 22:45:29) [MSC v.1916 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:\Users\Nekit\Desktop\Базовые компоненты\Lab1.py =====
Введите коэффициент A:
4
Введите коэффициент B:
3
Введите коэффициент C:
4
Нет корней
>>> |
```

```
Python 3.8.2 (tags/v3.8.2:7b3ab59, Feb 25 2020, 22:45:29) [MSC v.1916 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:\Users\Nekit\Desktop\Базовые компоненты\Lab1.py =====
Введите коэффициент A:
1
Введите коэффициент B:
-10
Введите коэффициент C:
илили
Введите коэффициент C:
9
Четыре корня: x1=3.0; x2=-3.0; x3=1.0; x4=-1.0
>>> |
```

 Командная строка

```
Microsoft Windows [Version 10.0.19044.2006]
(c) Корпорация Майкрософт (Microsoft Corporation). Все права защищены.

C:\Users\Nekit>cd Desktop\Базовые компоненты

C:\Users\Nekit\Desktop\Базовые компоненты>Lab1.py
Введите коэффициент A:
1
Введите коэффициент B:
-10
Введите коэффициент C:
9
Четыре корня: x1=3.0; x2=-3.0; x3=1.0; x4=-1.0

C:\Users\Nekit\Desktop\Базовые компоненты>Lab1.py 1 -10 9
Четыре корня: x1=3.0; x2=-3.0; x3=1.0; x4=-1.0

C:\Users\Nekit\Desktop\Базовые компоненты>
```