

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

Факультет «Информатика и управление»

Кафедра «Системы обработки информации и управления»

Курс «Базовые компоненты интернет-технологий»

Отчет по лабораторной работе №1
«Основные конструкции языка Python. Решение биквадратного уравнения»

Выполнил:

студент группы ИУ5-33Б

Семенов Вячеслав

Подпись и дата:

Проверил:

преподаватель каф. ИУ5

Канев Антон

Подпись и дата:

1. Описание задания

Разработайте программу для решения биквадратичного уравнения.

1. Программа должна быть разработана как консольное приложение на Python.
2. Программа вводит коэффициенты АВС с клавиатуры, вычисляет дискриминант и ДЕЙСТВИТЕЛЬНЫЕ корни уравнения (в зависимости от дискриминанта).
3. Коэффициенты А, В и С могут быть установлены в качестве параметров командной строки (вариант установки параметров приведен в конце файла с примером кода). Если они не указаны, они вводятся с клавиатуры в соответствии с пунктом 2. Описание работы с параметрами командной строки.
4. Если коэффициент А, В, С введен или установлен неправильно в командной строке, то необходимо игнорировать неверное значение и вводить коэффициент снова, пока коэффициент не будет введен правильно. Правильно установленный коэффициент-это коэффициент, значение которого может быть преобразовано в действительное число без ошибок.

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]
    except:
        # Вводим с клавиатуры
        print(prompt)
        while True:
            try:
                coef_str=input()
                coef = float(coef_str)
            except ValueError:
                print("Введите верное число: ")
                continue
            if index == 1 and coef == 0.0:
                print("Введите число, отличное от 0: ")
            else:
                break
        # Переводим строку в действительное число
    return coef
```

```

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:
        root = -b / (2.0*a)
        if root < 0:
            root == 0
        else:
            x1 = math.sqrt(root)
            x2 = -x1
            if x2 == x1:
                result.append(x2)
            else:
                result.append(x1)
                result.append(x2)
    elif D > 0.0:
        sqD = math.sqrt(D)
        root1 = (-b + sqD) / (2.0*a)
        root2 = (-b - sqD) / (2.0*a)
        if root1 < 0:
            root1==0
        else:
            x1 = math.sqrt(root1)
            x2 = -x1
            if x2 == x1:
                result.append(x1)
            else:
                result.append(x1)
                result.append(x2)
        if root2 < 0:
            root2==0
        else:
            x3 = math.sqrt(root2)
            x4 = -x3
            if x3 == x4:
                result.append(x3)
            else:
                result.append(x3)
                result.append(x4)
    return result

def main():
    """

```

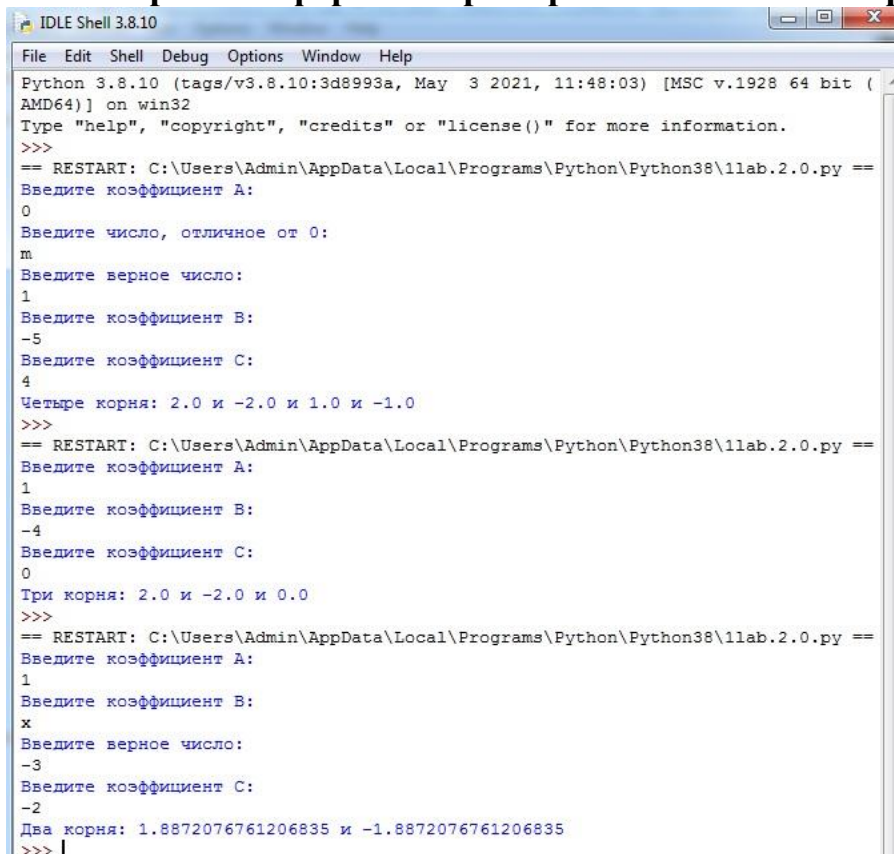
Основная функция

```
"""
a = get_coef(1, "Введите коэффициент A: ")
b = get_coef(2, "Введите коэффициент B: ")
c = get_coef(3, "Введите коэффициент C: ")
# 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: {}'.format(roots[0]))
elif len_roots == 2:
    print('Два корня: {} и {}'.format(roots[0], roots[1]))
elif len_roots == 3:
    print('Три корня: {} и {} и {}'.format(roots[0], roots[1], roots[2]))
elif len_roots == 4:
    print('Четыре корня: {} и {} и {} и {}'.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



```
IDLE Shell 3.8.10
File Edit Shell Debug Options Window Help
Python 3.8.10 (tags/v3.8.10:3d8993a, May 3 2021, 11:48:03) [MSC v.1928 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
== RESTART: C:\Users\Admin\AppData\Local\Programs\Python\Python38\11ab.2.0.py ==
Введите коэффициент A:
0
Введите число, отличное от 0:
m
Введите верное число:
1
Введите коэффициент B:
-5
Введите коэффициент C:
4
Четыре корня: 2.0 и -2.0 и 1.0 и -1.0
>>>
== RESTART: C:\Users\Admin\AppData\Local\Programs\Python\Python38\11ab.2.0.py ==
Введите коэффициент A:
1
Введите коэффициент B:
-4
Введите коэффициент C:
0
Три корня: 2.0 и -2.0 и 0.0
>>>
== RESTART: C:\Users\Admin\AppData\Local\Programs\Python\Python38\11ab.2.0.py ==
Введите коэффициент A:
1
Введите коэффициент B:
-3
Введите верное число:
-3
Введите коэффициент C:
-2
Два корня: 1.8872076761206835 и -1.8872076761206835
>>> |
```