



Министерство науки и высшего образования Российской Федерации  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Московский государственный технический университет  
имени Н.Э. Баумана  
(национальный исследовательский университет)»  
(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ

РАДИОТЕХНИЧЕСКИЙ

КАФЕДРА

СИСТЕМЫ ОБРАБОТКИ ИНФОРМАЦИИ И УПРАВЛЕНИЯ

## **ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №1**

### **Основные конструкции языка Python.**

Студент Столярова Ольга Денисовна  
*фамилия, имя, отчество*

Группа РТ5-51Б

Название предприятия МГТУ им. Н. Э. Баумана

Студент

Столярова О.Д.

Преподаватель

Гапанюк Ю.Е.

2021 г.

## Цель работы

Изучение основных конструкций языка Python.

## Задание

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.

## Текст программы

```
import sys
import math

def get_coef(index, prompt):
    try:
        # Пробуем прочитать коэффициент из командной строки
        coef_str = sys.argv[index]
    except:
        # Вводим с клавиатуры
        print(prompt)
        coef_str = input()
    # Переводим строку в действительное число
    coef = float(coef_str)
    return coef
```

```

def get_roots(a, b, c):
    result = []
    D = b*b - 4*a*c
    if D == 0.0:
        root = -b / (2.0*a)
        if root > 0:
            result.append(math.sqrt(root))
            result.append(-math.sqrt(root))
        if root == 0:
            result.append(root)

    elif D > 0.0:
        sqD = math.sqrt(D)
        root1 = (-b + sqD) / (2.0*a)
        if root1 > 0:
            result.append(math.sqrt(root1))
            result.append(-math.sqrt(root1))
        if root1 == 0:
            result.append(root1)

        root2 = (-b - sqD) / (2.0*a)
        if root2 > 0:
            result.append(math.sqrt(root2))
            result.append(-math.sqrt(root2))
        if root2 == 0:
            result.append(root2)
    return result

def main():
    """
    Основная функция
    """
    a = get_coef(1, 'Введите коэффициент А:')
    b = get_coef(2, 'Введите коэффициент В:')
    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: {}'.format(roots[0]))
    elif len_roots == 2:
        print('Два корня: {} и {}'.format(roots[0], roots[1]))
    elif len_roots == 3:
        print('Три корня: {}, {} и {}'.format(roots[0], roots[1], roots[2]))
    elif len_roots == 4:
        print('Четыре корня: {}, {}, {} и {}'.format(roots[0], roots[1], roots[2], roots[3]))

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

```

## Пример выполнения программы

```
C:\Program Files (x86)\Microsoft Visual Studio\Shared\Python36_64\python.exe
Введите коэффициент A:
1
Введите коэффициент B:
-5
Введите коэффициент C:
-36
Два корня: 3.0 и -3.0
Press any key to continue . . .
```

```
C:\Program Files (x86)\Microsoft Visual Studio\Shared\Python36_64\python.exe
Введите коэффициент A:
1
Введите коэффициент B:
-5
Введите коэффициент C:
4
Четыре корня: 2.0, -2.0, 1.0 и -1.0
Press any key to continue . . .
```

Ссылка на GitHub  
<https://github.com/OlyaSto/Olyabmstu>