



**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 Информатика и системы управления  
КАФЕДРА Системы обработки информации и управления

**Лабораторная работа №1  
По курсу «Разработка интернет приложений»**

Подготовил:  
Студент группы ИУ5-55Б.  
Турчин Д.С.  
30.09.2020

Проверил:  
Преподаватель кафедры ИУ5  
Гапанюк Ю.Е.

Москва, 2021 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 А, В, С, вычисляет дискриминант и **ДЕЙСТВИТЕЛЬНЫЕ** корни уравнения (в зависимости от дискриминанта).
3. Коэффициенты А, В, С могут быть заданы в виде параметров командной строки. Если они не заданы, то вводятся с клавиатуры в соответствии с пунктом 2.
4. Если коэффициент А, В, С 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:

Функция main состоит из 3 частей:

- 1) Ввод с клавиатуры коэффициентов А, В, С
- 2) Вычисление дискриминанта и действительных корней уравнения
- 3) Вывод корней

```
def main():
    a = get_coef(1, 'Введите коэффициент А:')
    b = get_coef(2, 'Введите коэффициент В:')
    c = get_coef(3, 'Введите коэффициент С:')

    roots = get_roots(a,b,c)
    len_roots = len(roots)
    if len_roots == 0:
        print('У уравнения {}x^2 {} {}x {} {} нет корней'.format(a, get_sign(b),
abs(b), get_sign(c), abs(c)))
    elif len_roots == 1:
        print('У уравнения {}x^2 {} {}x {} {} один корень: {}'.format(a,
get_sign(b), abs(b), get_sign(c), abs(c), roots[0]))
    elif len_roots == 2:
        print('У уравнения {}x^2 {} {}x {} {} два корня: {}, {}'.format(a,
get_sign(b), abs(b), get_sign(c), abs(c), roots[0], roots[1]))
        a = input('Нажмите ENTER для завершения')

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

## Используемые функции:

1) Получение коэффициента с командной строки или с Stdout:

```
def get_coef(index, prompt):  
    try:  
        coef_str = sys.argv[index]  
    except:  
        coef_str = input(prompt)  
    coef = float(coef_str)  
    return coef
```

2) Вычисление корней:

```
def get_roots(a, b, c):  
    result = []  
    D = b*b - 4*a*c  
    if D == 0.0:  
        root = -b / (2.0*a)  
        result.append(root)  
    elif D > 0.0:  
        sqD = math.sqrt(D)  
        root1 = (-b + sqD) / (2.0*a)  
        root2 = (-b - sqD) / (2.0*a)  
        result.append(root1)  
        result.append(root2)  
    return result
```

3) Выбор знака при красивом форматировании ответа.

```
def get_sign(number):  
    if number >= 0:  
        return '+'  
    return '-'
```

## Экранные формы с примерами работы программы

### 1 корень

```
Введите коэффициент A:4  
Введите коэффициент B:-4  
Введите коэффициент C:1  
У уравнения 4.0x^2 - 4.0x + 1.0 один корень: 0.5  
Нажмите ENTER для завершения
```

### 2 корня

```
Введите коэффициент A:3
Введите коэффициент B:-9
Введите коэффициент C:2
У уравнения  $3.0x^2 - 9.0x + 2.0$  два корня: 2.758305739211792, 0.2416942607882083
5
Нажмите ENTER для завершения
```

d

## Нет корней

```
Введите коэффициент A:1
Введите коэффициент B:1
Введите коэффициент C:1
У уравнения  $1.0x^2 + 1.0x + 1.0$  нет корней
Нажмите ENTER для завершения
```

## Один из параметров = 0

```
Input coef A:
2
Input coef B:
2
Input coef C:
0
Two roots: 0.0 , -1.0
```