



**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»**

**Якимова Татьяна ИУ5-35Б
Парадигмы и конструкции языков программирования**

ОТЧЁТ ПО Лабораторной работе №2

Москва
2023

Задание.

Написать консольное приложение на Python для решения биквадратного уравнения. Программа должна выполнять следующие шаги:

- Получить коэффициенты А, В и С от пользователя через ввод с клавиатуры или параметры командной строки.
- Проверить, что введенные коэффициенты могут быть корректно преобразованы в действительные числа. Если какой-то коэффициент введен некорректно, программа должна запросить ввод повторно, пока значения не будут корректными.
- Вычислить дискриминант и вывести корни уравнения в зависимости от значения дискриминанта.

Дополнительное задание: реализовать две версии программы на Python - одну с использованием процедурной парадигмы и другую с использованием объектно-ориентированной парадигмы.

Текст программы.

Процедурный подход:

```
import sys
import math

def get_koeff(index):
    if index == 1:
        letter = 'A'
    elif index == 2:
        letter = 'B'
    else:
        letter = 'C'
    try:
        koeff_str = sys.argv[index]
    except:
        while True:
            print('Введите коэффициент {}: '.format(letter))
            koeff_str = input()
            try:
                koeff = float(koeff_str)
                return koeff
            except ValueError:
                print('Введено неверное значение. Попробуйте снова.\n')

def get_roots(a, b, c):
    result = []
    D = b * b - 4 * a * c
    if D == 0.0:
        root = math.sqrt(-b / (2.0 * a))
        if (root == 0.0):
            result.append(abs(root))
        else:
            result.append(root)
            result.append(-root)
    elif D > 0.0:
        sqD = math.sqrt(D)
        r1 = (-b + sqD) / (2.0 * a)
        r2 = (-b - sqD) / (2.0 * a)
```

```

        if r1 == 0.0:
            result.append(r1)
        if r2 == 0.0 and r1 != 0.0:
            result.append(r2)
        if r1 > 0.0:
            root1 = math.sqrt(r1)
            result.append(root1)
            result.append(-root1)
        if r2 > 0.0:
            root2 = math.sqrt(r2)
            result.append(root2)
            result.append(-root2)
    return result

def main():
    a = get_koeff(1)
    b = get_koeff(2)
    c = get_koeff(3)

    roots = get_roots(a, b, c)

    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()

```

Объектно-ориентированный подход:

```

import sys
import math

class Equation:
    def __init__(self):
        self.A = 0.0
        self.B = 0.0
        self.C = 0.0
        self.solutions = []

    def get_koeff(self, index):
        if index == 1:
            letter = 'A'
        elif index == 2:
            letter = 'B'

```

```

else:
    letter = 'C'
try:
    coeff_str = sys.argv[index]
except:
    while True:
        print('Введите коэффициент {}: '.format(letter))
        coeff_str = input()
        try:
            coeff = float(coeff_str)
            return coeff
        except ValueError:
            print('Введено неверное значение. Попробуйте снова.\n')
def get_koeffs(self):
    self.A = self.get_koeff(1)
    self.B = self.get_koeff(2)
    self.C = self.get_koeff(3)

def get_roots(self):
    a = self.A
    b = self.B
    c = self.C
    result = []
    D = b*b - 4 * a * c
    if D == 0.0:
        root = math.sqrt(-b / (2.0 * a))
        if (root == 0.0):
            result.append(abs(root))
        else:
            result.append(root)
            result.append(-root)
    elif D > 0.0:
        sqD = math.sqrt(D)
        r1 = (-b + sqD) / (2.0 * a)
        r2 = (-b - sqD) / (2.0 * a)
        if r1 == 0.0:
            result.append(r1)
        if r2 == 0.0 and r1 != 0.0:
            result.append(r2)
        if r1 > 0.0:
            root1 = math.sqrt(r1)
            result.append(root1)
            result.append(-root1)
        if r2 > 0.0:
            root2 = math.sqrt(r2)
            result.append(root2)
            result.append(-root2)
    return result

def main():
    eq = Equation()
    eq.get_koeffs()

    roots = eq.get_roots()

    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()

```

Экранные формы:

Четыре корня:

```

Введите коэффициент A:
2.5
Введите коэффициент B:
-15
Введите коэффициент C:
1.6
Четыре корня: 2.427213723379174, -2.427213723379174, 0.3295960270388705 и -0.3295960270388705

```

Два корня:

```

Введите коэффициент A:
1
Введите коэффициент B:
-2
Введите коэффициент C:
1
Два корня: 1.0 и -1.0

```

Один корень:

```

Введите коэффициент A:
1
Введите коэффициент B:
0
Введите коэффициент C:
0
Один корень: 0.0

```

Нет корней:

```
Введите коэффициент A:  
1  
Введите коэффициент B:  
2  
Введите коэффициент C:  
5  
Нет корней
```

Некорректный ввод из командной строки:

```
Введите коэффициент A:  
=  
Введено неверное значение. Попробуйте снова.  
  
Введите коэффициент A:  
2  
Введите коэффициент B:
```