

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

Факультет Радиотехнический
Кафедра РТ5

Курс «Парадигмы и конструкции языков программирования»

Отчет по лабораторной работе №2

«Объектно-ориентированные возможности языка Python»

Выполнил:

студент группы РТ5-31Б:

Бабасанова Н. С.

Руководитель:

преподаватель каф. ИУ5

Гапанюк Ю. Е.

Москва, 2023г.

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. Если они не заданы, то вводятся с клавиатуры в соответствии с пунктом 2.
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.
5. Дополнительное задание 1. Разработайте две программы на языке Python - одну с применением процедурной парадигмы, а другую с применением объектно-ориентированной парадигмы.
6. Дополнительное задание 2. Разработайте две программы - одну на языке Python, а другую на любом другом языке программирования (кроме C++).

Текст программы

Python с использованием процедурной парадигмы:

```
import sys
from math import sqrt
```

```
def get_coef(index, prompt):
    coef_str = None
    if len(sys.argv) > 1:
```

```

try:
    coef_str = float(sys.argv[index])
except ValueError:
    print("Некорректный аргумент командной строки.")
    sys.exit()
else: # случай, когда sys.argv не имеет аргументов
    # Вводим с клавиатуры, проверяем на ввод чисел
    print(prompt)
    while coef_str is None:
        try:
            coef_str = float(input("Введите число: "))
        except ValueError:
            print("Некорректный ввод!")
            continue
        break
    # Переводим строку в действительное число
    coef = float(coef_str)
    return coef

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

def main():
    a = get_coef(1, 'Коэффициент A:')
    b = get_coef(2, 'Коэффициент B:')
    c = get_coef(3, '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('Два корня: +/-{}'.format(roots[0]))
    elif len_roots == 2:
        print('Четыре корня: +/-{}, +/-{}'.format(roots[0], roots[1]))

```

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

Python с использованием объектно-ориентированной парадигмы:

```
import sys
from math import sqrt
```

```
class BiquadraticEquation:
```

```
    def __init__(self):
        self.coef_A = 0.0
        self.coef_B = 0.0
        self.coef_C = 0.0
        self.num_roots = 0 # Количество корней
        self.roots_list = [] # Список корней
```

```
    @staticmethod
```

```
    def get_coef(index, prompt):
        coef_str = None
        if len(sys.argv) > 1:
            try:
                coef_str = float(sys.argv[index])
            except ValueError:
                print("Некорректный аргумент командной строки.")
                sys.exit()
        else: # случай, когда sys.argv не имеет аргументов
            print(prompt)
            while coef_str is None:
                try:
                    coef_str = float(input("Введите число: "))
                except ValueError:
                    print("Некорректный ввод!")
                    continue
                break
        coef = float(coef_str)
        return coef
```

```
    def get_coefs(self):
        self.coef_A = self.get_coef(1, 'Введите коэффициент A:')
        self.coef_B = self.get_coef(2, 'Введите коэффициент B:')
        self.coef_C = self.get_coef(3, 'Введите коэффициент C:')
```

```
    def calculate_roots(self):
        a = self.coef_A
        b = self.coef_B
        c = self.coef_C
```

```

discriminant = b * b - 4 * a * c
if discriminant == 0.0:
    root = sqrt(-b / (2.0 * a))
    self.num_roots = 1
    self.roots_list.append(root)
elif discriminant > 0.0:
    sqd = sqrt(discriminant)
    root1 = sqrt((-b + sqd) / (2.0 * a))
    root2 = sqrt((-b - sqd) / (2.0 * a))
    self.num_roots = 2
    self.roots_list.append(root1)
    self.roots_list.append(root2)

def print_roots(self):
    # Проверка отсутствия ошибок при вычислении корней
    if self.num_roots != len(self.roots_list):
        print(('Ошибка. Уравнение содержит {} действительных корней, ' +
              'но было вычислено {} корней.').format(self.num_roots, len(self.roots_list)))
    else:
        if self.num_roots == 0:
            print('Нет корней')
        elif self.num_roots == 1:
            print('Два корня: +/-{}'.format(self.roots_list[0]))
        elif self.num_roots == 2:
            print('Четыре корня: +/-{}, +/-{}'.format(self.roots_list[0], self.roots_list[1]))

def main():
    biquad = BiquadraticEquation()
    biquad.get_coefs()
    biquad.calculate_roots()
    biquad.print_roots()

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

```

C#:

```

using System;

namespace ConsoleApp1
{
    public class BiquadraticEquation
    {
        private double _coefA, _coefB, _coefC;

        public BiquadraticEquation()
        {

```

```

        _coefA = 0.0;
        _coefB = 0.0;
        _coefC = 0.0;
    }

    public void SetCoefA(double coefA)
    {
        _coefA = coefA;
    }

    public void SetCoefB(double coefB)
    {
        _coefB = coefB;
    }
    public void SetCoefC(double coefC)
    {
        _coefC = coefC;
    }

    public static double GetCoef()
    {
        Console.Write("\nВведите число: ");
        string? inputStr = Console.ReadLine();
        while (!double.TryParse(inputStr, out _))
        {
            Console.Write("Некорректный ввод!\nВведите число: ");
            inputStr = Console.ReadLine();
        }
        return Convert.ToDouble(inputStr);
    }
    public void CalculateRoots()
    {
        double discriminant = this._coefB * this._coefB - 4 * this._coefA * this._coefC;
        if (discriminant == 0)
        {
            Console.Write("Два корня: +/-{0}\n", Math.Sqrt(-this._coefB / (2.0 * this._coefA)));
        }
        else if (discriminant > 0)
        {
            Console.Write("Четыре корня: +/-{0}, ", Math.Sqrt((-this._coefB + Math.Sqrt(discriminant)) /
(2.0 * this._coefA)));
            Console.Write("+/-{0}\n", Math.Sqrt((-this._coefB - Math.Sqrt(discriminant)) / (2.0 *
this._coefA)));
        }
        else
            Console.Write("Нет корней\n");
    }
}

public class Program
{

```

```

public static void Main(string[] args)
{
    // Создаем объект типа BiquadraticEquation
    BiquadraticEquation biquad = new BiquadraticEquation();
    if (args.Length == 0)
    {
        Console.Write("Коэффициент A:");
        biquad.SetCoefA(BiquadraticEquation.GetCoef());
        Console.Write("Коэффициент B:");
        biquad.SetCoefB(BiquadraticEquation.GetCoef());
        Console.Write("Коэффициент C:");
        biquad.SetCoefC(BiquadraticEquation.GetCoef());
    }
    else
    {
        try
        {
            biquad.SetCoefA(Convert.ToDouble(args[0]));
        }
        catch (FormatException)
        {
            Console.WriteLine("Некорректные аргументы командной строки.");
            return;
        }

        try
        {
            biquad.SetCoefB(Convert.ToDouble(args[1]));
        }
        catch (FormatException)
        {
            Console.WriteLine("Некорректные аргументы командной строки.");
            return;
        }

        try
        {
            biquad.SetCoefC(Convert.ToDouble(args[2]));
        }
        catch (FormatException)
        {
            Console.WriteLine("Некорректные аргументы командной строки.");
            return;
        }
    }
    biquad.CalculateRoots();
}
}

```

Результаты вывода:

Python с использованием процедурной парадигмы:

- Вывод корней:
 - Четыре корня:

```
Коэффициент А:  
Введите число: 1  
Коэффициент В:  
Введите число: -12  
Коэффициент С:  
Введите число: 4  
Четыре корня: +/-3.414213562373095, +/-0.5857864376269046
```

- Два корня:

```
Коэффициент А:  
Введите число: 1  
Коэффициент В:  
Введите число: -4  
Коэффициент С:  
Введите число: 4  
Два корня: +/-1.4142135623730951
```

- Нет корней:

```
Коэффициент А:  
Введите число: 5  
Коэффициент В:  
Введите число: 5  
Коэффициент С:  
Введите число: 5  
Нет корней
```

- Некорректный ввод:

```
Коэффициент А:  
Введите число: p  
Некорректный ввод!
```

Python с использованием объектно-ориентированной парадигмы:

- Вывод корней:
 - Четыре корня:

Введите коэффициент A:
Введите число: 1
Введите коэффициент B:
Введите число: -12
Введите коэффициент C:
Введите число: 4
Четыре корня: +/-3.414213562373095, +/-0.5857864376269046

- Два корня:

Введите коэффициент A:
Введите число: 1
Введите коэффициент B:
Введите число: -4
Введите коэффициент C:
Введите число: 4
Два корня: +/-1.4142135623730951

- Нет корней:

Введите коэффициент A:
Введите число: 5
Введите коэффициент B:
Введите число: 5
Введите коэффициент C:
Введите число: 5
Нет корней

- Некорректный ввод:

Введите коэффициент A:
Введите число: 3
Некорректный ввод!

C#:

- Вывод корней:

- Четыре корня:

Коэффициент A:
Введите число: 1
Коэффициент B:
Введите число: -12
Коэффициент C:
Введите число: 4
Четыре корня: +/-3,414213562373095, +/-0,5857864376269046

- Два корня:

Коэффициент А:
Введите число: 1
Коэффициент В:
Введите число: -4
Коэффициент С:
Введите число: 4
Два корня: $\pm 1,4142135623730951$

- Нет корней:

Коэффициент А:
Введите число: 5
Коэффициент В:
Введите число: 5
Коэффициент С:
Введите число: 5
Нет корней

- Некорректный ввод:

Коэффициент А:
Введите число: р
Некорректный ввод!