

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

**Факультет «Радиотехнический»  
Кафедра «Системы обработки информации и управления»**

**Курс «Парадигмы и конструкции языков программирования»**

**Отчет по лабораторной работе №1  
«Основные конструкции языка C#»**

Выполнил:  
студент группы РТ5-31Б:  
Чиканчи В. А.

Подпись и дата:

Проверил:  
преподаватель кафедры ИУ5  
Гапанюк Ю. Е.

Подпись и дата:

Москва, 2024 г.

# Постановка задачи

Разработать программу для решения биквадратного уравнения.

1. Программа должна быть разработана в виде консольного приложения на языке C#.
2. Программа осуществляет ввод с клавиатуры коэффициентов A, B, C, вычисляет дискриминант и действительные корни уравнения (в зависимости от дискриминанта).
3. Если коэффициент A, B, C введен некорректно, то необходимо проигнорировать некорректное значение и вводить коэффициент повторно.
4. Корни уравнения выводятся зелёным цветом. Если корней нет, то сообщение выводится красным цветом.
5. 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. Проверка из пункта 3 в этом случае производится для параметров командной строки без повторного ввода с клавиатуры.

## Текст программы

Файл «main.py»:

```
from python_oop.shape import Rectangle, Circle, Square
import numpy as np

if __name__ == "__main__":
    N = 5

    rectangle = Rectangle(N, N, "blue")
    circle = Circle(N, "green")
    square = Square(N, "red")

    print(rectangle)
    print(circle)
    print(square)

    print("Сумма чисел от 1 до N:", np.sum(range(1, N + 1)))
```

Файл «shape.py»:

```
from abc import ABC, abstractmethod
import math

class Shape(ABC):
    @abstractmethod
```

```

    def area(self):
        pass

class Color:
    def __init__(self, color_name):
        self.color_name = color_name

class Rectangle(Shape):
    def __init__(self, width, height, color):
        self.width = width
        self.height = height
        self.color = Color(color)

    def area(self):
        return self.width * self.height

    def __repr__(self):
        return "Rectangle(width={}, height={}, color={}, area={})".format(
            self.width, self.height, self.color.color_name, self.area()
        )

class Circle(Shape):
    def __init__(self, radius, color):
        self.radius = radius
        self.color = Color(color)

    def area(self):
        return math.pi * (self.radius ** 2)

    def __repr__(self):
        return "Circle(radius={}, color={}, area={:.2f})".format(
            self.radius, self.color.color_name, self.area()
        )

class Square(Rectangle):
    def __init__(self, side, color):
        super().__init__(side, side, color)

    def __repr__(self):
        return "Square(side={}, color={}, area={})".format(
            self.width, self.color.color_name, self.area()
        )

```

**Экранные формы с примерами выполнения  
программы**

```
Rectangle(width=5, height=5, color=blue, area=25)  
Circle(radius=5, color=green, area=78.54)  
Square(side=5, color=red, area=25)  
Сумма чисел от 1 до N: 15
```