

Часть 1. Создание простого приложения

Цель работы: изучение среды разработки для языка программирования C#.

Выполнение: диаграмма классов (класса Main) изображена на рисунке 1

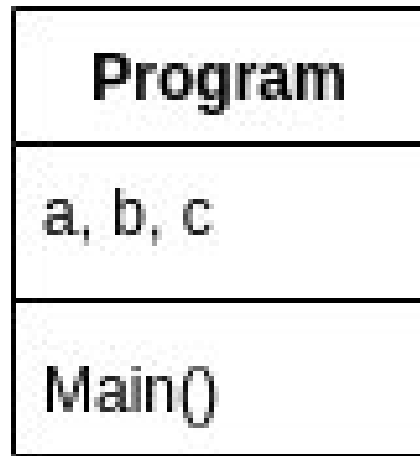


Рисунок 1 — Диаграмма класса Main

Схема алгоритма метода Main() изображена на рисунке 2

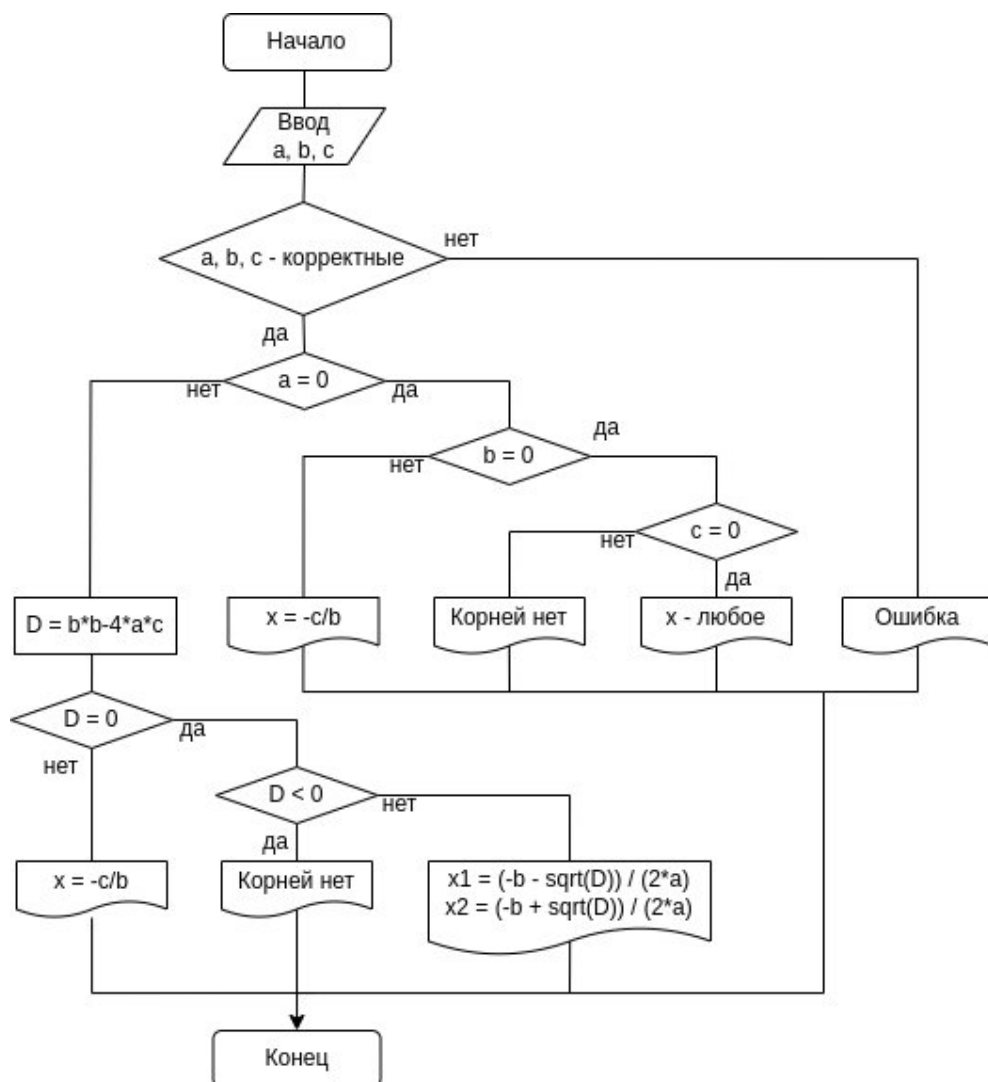


Рисунок 2 — Схема алгоритма Main

Основной код программы изображён на рисунке 3

```
public class Program
{
    private static double a, b, c;

    private static void Main(string[] args)
    {
        Console.WriteLine("Эта программа решает уравнение Ax^2 + Bx + C = 0");
        Console.WriteLine("Введите аргументы A, B, C: ");
        if (!double.TryParse(Console.ReadLine(), out a) || !double.TryParse(Console.ReadLine(), out b) ||
            !double.TryParse(Console.ReadLine(), out c))
        {
            Console.WriteLine("Ошибка в введённых данных!");
            return;
        }

        Console.WriteLine("{0}x^2 + {1}x + {2} = 0", a, b, c);

        if (a == 0)
        {
            if (b == 0)
            {
                if (c == 0)
                {
                    Console.WriteLine("x - любое действительное число!");
                }
                else
                {
                    Console.WriteLine("Корней среди действительных чисел нет!");
                }
            }
            else
            {
                Console.WriteLine("x = {0}", -c / b);
            }
        }
        else
        {
            var D = b * b - 4 * a * c;
            if (D == 0)
            {
                Console.WriteLine("x = {0}", -b / (2 * a));
            }
            else if (D < 0)
            {
                Console.WriteLine("Корней среди действительных чисел нет!");
            }
            else
            {
                Console.WriteLine("x1 = {0}", (-b - double.Sqrt(D)) / (2 * a));
                Console.WriteLine("x2 = {0}", (-b + double.Sqrt(D)) / (2 * a));
            }
        }
    }
}
```

Рисунок 3 — Основной код программы

Тестовые данные и результаты тестирования: тестовые данные и результаты тестирования изображены на рисунках 4-5

a	b	c	Вывод
0	0	0	x - любое действительное число!
0	0	1	Корней среди действительных чисел нет!
0	5	2,5	x = -0,5
1	5	-6	x1 = -6 x2 = 1
hello			Ошибка в введенных данных!

Рисунок 4 — Тестовые данные и результаты тестирования

```
"/home/sergey/МГТУ/АиП/2 sem/LW/03/03/bin/Debug/net9.0/03"
Эта программа решает уравнение Ax^2 + Bx + C = 0
Введите аргументы A, B, C:
hello
Ошибка в введенных данных!

Process finished with exit code 0.
```

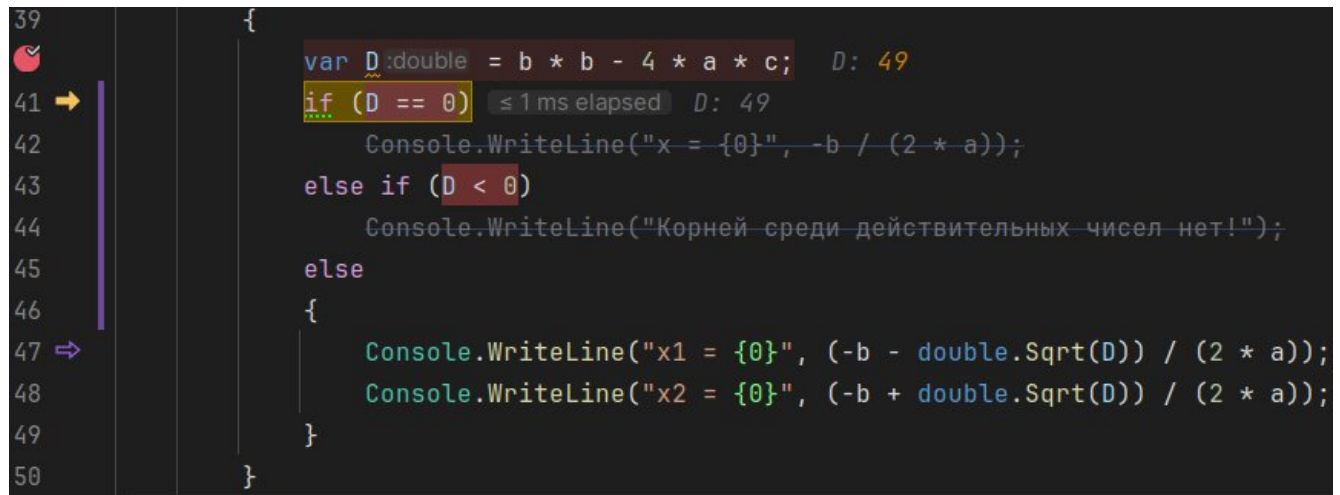
Рисунок 5 — Консоль программы

Вывод: в ходе лабораторной работы было изучено создание консольных приложений на языке C#, разработана программа для нахождения корней квадратного уравнения с обработкой исключений при вводе нечисловых данных и учтены случаи отсутствия корней. Программа была отлажена и протестирована на разных наборах данных.

Часть 2. Диагностические сообщения компилятора и средства отладки

Цель работы: изучить диагностические сообщения компилятора и средства отладки.

Выполнение: точка остановки и процесс отладки изображены на рисунке 6.



```
39 {  
40     var D:double = b * b - 4 * a * c; D: 49  
41     if (D == 0) ≤ 1 ms elapsed D: 49  
42         Console.WriteLine("x = {0}", -b / (2 * a));  
43     else if (D < 0)  
44         Console.WriteLine("Корней среди действительных чисел нет!");  
45     else  
46     {  
47         Console.WriteLine("x1 = {0}", (-b - double.Sqrt(D)) / (2 * a));  
48         Console.WriteLine("x2 = {0}", (-b + double.Sqrt(D)) / (2 * a));  
49     }  
50 }
```

Рисунок 6 — Точка остановки и процесс отладки

Также IDE позволяет смотреть ресурсы, потребляемые программой (рисунок 7)

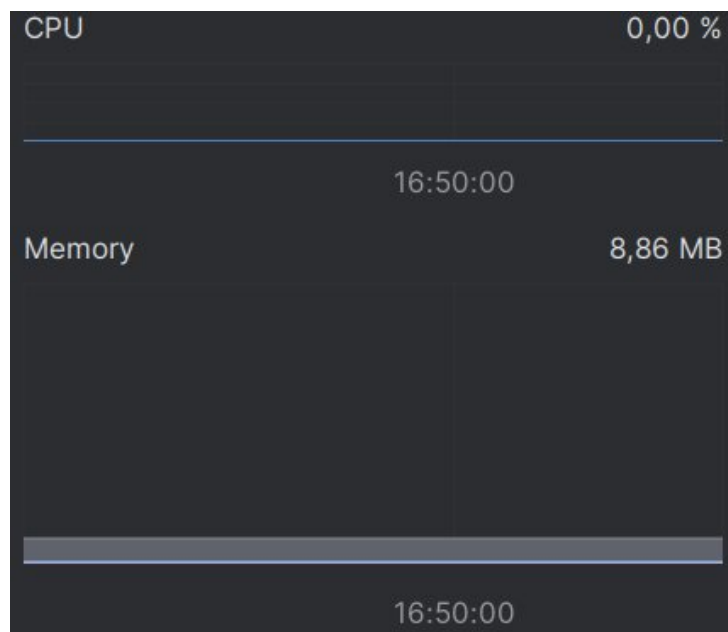


Рисунок 7 — Ресурсы, потребляемые программой

Вывод: в ходе лабораторной работы было изучено использование диагностических сообщений компилятора и средств отладки в IDE. На примере программы для решения квадратного уравнения были применены средства отладки для установки контрольных точек и анализа промежуточных результатов выполнения программы.