



Домашнее задание

# Тайминг	5
≡ Формат	Объясняет преподаватель.

Пример идеального решения:

Задача 1:

```
// Пользователь вводит с клавиатуры М чисел. Посчитайте, сколько чисел  
// больше 0 ввёл пользователь.  
// 0, 7, 8, -2, -2 -> 2  
// 1, -7, 567, 89, 223-> 3  
  
//Считать число с консоли  
int Prompt(string message)  
{  
    System.Console.Write(message); //вывести сообщение  
    string value = Console.ReadLine(); //считывает с консоли строку  
    int result = Convert.ToInt32(value); // преобразует строку в целое  
    число  
    return result; //возвращает результат  
}
```

```

// Ввести массив
int[] InputArray(int length)
{
    int[] array = new int[length];
    for (int i = 0; i < array.Length; i++)
    {
        array[i] = Prompt($"Введите {i + 1}-й элемент");
    }
    return array;
}

void PrintArray(int[] array)
{
    for (int i = 0; i < array.Length; i++)
    {
        Console.WriteLine($"a[{i}] = {array[i]}");
    }
}

int CountPositiveNumbers(int[] array)
{
    int count = 0;
    for (int i = 0; i < array.Length; i++)
    {
        if (array[i] > 0)
        {
            count++;
        }
    }
    return count;
}

int lenght = Prompt("Введите количество элементов >");
int[] array;
array = InputArray(lenght);
PrintArray(array);
Console.WriteLine($"Количество чисел больше 0 -
{CountPositiveNumbers(array)}");

```

Задача 2:

```

// Напишите программу, которая найдёт точку пересечения двух прямых,
// заданных уравнениями  $y = k_1 * x + b_1$ ,  $y = k_2 * x + b_2$ ; значения  $b_1$ ,  $k_1$ ,  $b_2$ 
// и  $k_2$  задаются пользователем.
//  $b_1 = 2$ ,  $k_1 = 5$ ,  $b_2 = 4$ ,  $k_2 = 9 \rightarrow (-0,5; -0,5)$ 

const int COEFFICIENT = 0;
const int CONSTANT = 1;
const int X_COORD = 0;
const int Y_COORD = 1;
const int LINE1 = 1;
const int LINE2 = 2;

double[] lineData1 = InputLineData(LINE1);
double[] lineData2 = InputLineData(LINE2);

if (ValidateLines(lineData1, lineData2))
{
    double[] coord = FindCoords(lineData1, lineData2);
    Console.WriteLine($"Точка пересечения уравнений
 $y = \{lineData1[COEFFICIENT]\} * x + \{lineData1[CONSTANT]\}$  и
 $y = \{lineData2[COEFFICIENT]\} * x + \{lineData2[CONSTANT]\}$  ");
    Console.WriteLine($" имеет координаты ({coord[X_COORD]},
{coord[Y_COORD]})");
}

// Ввод числа
double Prompt(string message)
{
    System.Console.Write(message); // вывести сообщение
    string value = Console.ReadLine(); // считывает с консоли строку
    double result = Convert.ToDouble(value); // преобразует строку в целое
    число
    return result; // возвращает результат
}

// Ввод данных по прямой
double[] InputLineData(int numberOfLine)
{
    double[] lineData = new double[2];
    lineData[COEFFICIENT] = Prompt($"Введите коэффициент для {numberOfLine}
прямой >");
    lineData[CONSTANT] = Prompt($"Введите константу для {numberOfLine}
прямой >");
    return lineData;
}

```

```

// Поиск координат
double[] FindCoords(double[] lineData1, double[] lineData2)
{
    double[] coord = new double[2];
    coord[X_COORD] = (lineData1[CONSTANT] - lineData2[CONSTANT]) /
(lineData2[COEFFICIENT] - lineData1[COEFFICIENT]);
    coord[Y_COORD] = lineData1[CONSTANT] * coord[X_COORD] +
lineData1[CONSTANT];
    return coord;
}

bool ValidateLines(double[] lineData1, double[] lineData2)
{
    if (lineData1[COEFFICIENT] == lineData2[COEFFICIENT])
    {
        if (lineData1[CONSTANT] == lineData2[CONSTANT])
        {
            Console.WriteLine("Прямые совпадают");
            return false;
        }
        else
        {
            Console.WriteLine("Прямые параллельны");
            return false;
        }
    }
    return true;
}

```