

ЛАБОРАТОРНАЯ РАБОТА № 3. ПРОГРАММИРОВАНИЕ С ИСПОЛЬЗОВАНИЕМ ОДНОМЕРНЫХ МАССИВОВ

Цель лабораторной работы: Изучить способы получения случайных чисел. Написать программу для работы с одномерными массивами.

2.1. Работа с массивами

Массив - набор элементов одного и того же типа, объединенных общим именем. Массивы в С# можно использовать по аналогии с тем, как они используются в других языках программирования. Однако С#-массивы имеют существенные отличия: они относятся к ссылочным типам данных, более того - реализованы как объекты. Фактически имя массива является ссылкой на область кучи (динамической памяти), в которой последовательно размещается набор элементов определенного типа. Выделение памяти под элементы происходит на этапе инициализации массива. А за освобождением памяти следит система сборки мусора - неиспользуемые массивы автоматически утилизируются данной системой.

Рассмотрим в данной лабораторной работе одномерные массивы. *Одномерный массив* - это фиксированное количество элементов одного и того же типа, объединенных общим именем, где каждый элемент имеет свой номер. Нумерация элементов массива в С# начинается с нуля, то есть, если массив состоит из 10 элементов, то его элементы будут иметь следующие номера: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9.

Одномерный массив в С# реализуется как объект, поэтому его создание представляет собой двухступенчатый процесс. Сначала объявляется ссылочная переменная на массив, затем выделяется память под требуемое количество элементов базового типа, и ссылочной переменной присваивается адрес нулевого элемента в массиве. Базовый тип определяет тип данных каждого элемента массива. Количество элементов, которые будут храниться в массиве, определяется размер массива.

В общем случае процесс объявления переменной типа массив, и выделение необходимого объема памяти может быть разделено. Кроме того на этапе объявления массива можно произвести его инициализацию. Поэтому для объявления одномерного массива может использоваться одна из следующих форм записи:

базовый_тип [] имя_массива;

Описана ссылка на одномерный массив, которая в дальнейшем может быть использована для адресации на уже существующий массив. Например:
int [] a;

базовый_тип [] имя_массива = new базовый_тип [размер];

Объявлен одномерный массив заданного типа и выделена память под одномерный массив указанной размерности. Адрес данной области памяти записан в ссылочную переменную. Элементы массива равны нулю (В С# элементам массива присваиваются начальные значения по умолчанию в

зависимости от базового типа. Для арифметических типов - нули, для ссылочных типов - null, для символов - пробел). Например:

```
int []a=new int [10];
```

базовый_тип [] имя__массива={список инициализации};

Выделена память под одномерный массив, размерность которого соответствует количеству элементов в списке инициализации. Адрес этой области памяти записан в ссылочную переменную. Значение элементов массива соответствует списку инициализации. Например:

```
int []a={0, 1, 2, 3};
```

Обращения к элементам массива происходит с помощью индекса, для этого нужно указать имя массива и в квадратных скобках его номер. Например,

```
a[0], b[10], c[i].
```

Так как массив представляет собой набор элементов, объединенных общим именем, то обработка массива обычно производится в цикле. Например:

...

```
int[] myArray = { 0, 1, 2, 3, 4, 5, 6, 7, 8, 9 };  
int i;  
for (i = 0; i < 10; ++i)  
    Console.WriteLine(myArray[i]);
```

...

2.2. Индексаторы

Индексаторы являются синтаксическим удобством, позволяющим создавать класс, структуру или интерфейс, доступ к которому клиентские приложения получают, как к массиву. Чаще всего индексаторы реализуются для доступа к закрытой внутренней коллекции или закрытому массиву. Вместе с модификаторами доступа индексаторы реализуют механизм инкапсуляции для полей-массивов и являются аналогами свойств, определяемых для обычных полей.

Пример объявления индексатора

```
public class AClass1 {  
    private int[] imyArray = new int[20];  
    public int this[int ind1] //индексатор  
    {  
        get  
            { return imyArray[ind1]; }  
        set  
            { imyArray[ind1] = value; }  
    }  
}
```

2.3. Случайные числа

Одним из способов задания массива является задание определению элементов через случайные числа. Для работы со случайными числами

используются в основном два метода класса **Random**: **Random** и **Next**. Метод **Random** подготавливает работу со случайными числами, обеспечивая, надежный способ создания непредсказуемой последовательности чисел.

Метод **Random.Next** создает случайное число в диапазоне значений от нуля до **Int32.MaxValue**. Для создания случайного числа в диапазоне от нуля до какого-либо другого положительного числа используется перегрузка метода **Random.Next(Int32)**. Для создания случайного числа в другом диапазоне используется перегрузка метода **Random.Next(Int32, Int32)**.

2.4. Порядок выполнения индивидуального задания

Создать форму с элементами управления как приведено на рис. 2.1. Описать одномерный массив. Создать обработчики события для кнопок (пример кода приведен ниже). Данная программа заменяет все отрицательные числа нулями. Протестировать правильность выполнения программы. Модифицировать программу в соответствии с индивидуальными задачами. При решении каждой задачи должны быть определены классы, «оборачивающие» обрабатываемые массивы, в этих классах должны быть реализованы индексаторы.

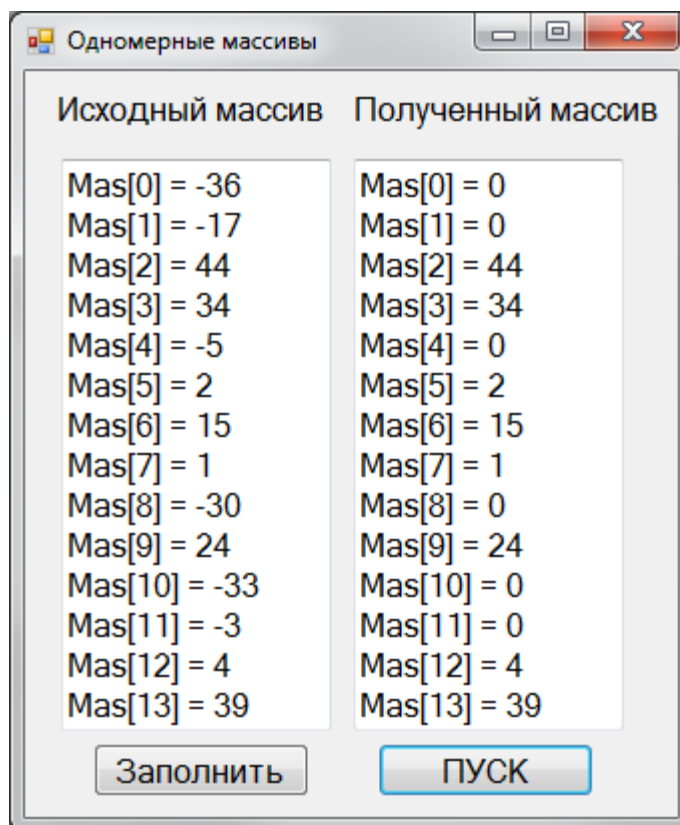


Рис. 2.1. Окно программы для работы с одномерными массивами

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;

namespace WindowsFormsApplication1
{
    public partial class Form1 : Form
    {
        int[] Mas = new int[15];
        public Form1()
        {
            InitializeComponent();
        }
        private void button1_Click(object sender, EventArgs e)
        {
            Random rand = new Random();
            textBox1.Text = "";
            for (int i = 0; i < 15; i++)
            {
                Mas[i] = rand.Next(-50, 50);
                textBox1.Text += "Mas[" + Convert.ToString(i) + "] = "
                    + Convert.ToString(Mas[i]) + Environment.NewLine;
            }
        }
        private void button2_Click(object sender, EventArgs e)
        {
            textBox2.Text = "";
            for (int i = 0; i < 15; i++)
            {
                if (Mas[i] < 0)
                    Mas[i] = 0;
                textBox2.Text += "Mas[" + Convert.ToString(i) + "] = "
                    + Convert.ToString(Mas[i]) + Environment.NewLine;
            }
        }
    }
}

```

2.5. Варианты заданий

1. В массиве из 20 целых чисел найти наибольший элемент и поменять его местами с первым элементом.
2. В массиве из 10 целых чисел найти наименьший элемент и поменять его местами с последним элементом.
3. В массиве из 15 вещественных чисел найти наибольший элемент и поменять его местами с последним элементом.
4. В массиве из 25 вещественных чисел найти наименьший элемент и поменять его местами с первым элементом.
5. Дан массив X , содержащий 27 элементов. Вычислить и вывести элементы нового массива Y , где $y_i = 6.85x_i^2 - 1.52$. Если $y_i < 0$, то вычислить и вывести $a = x_i^3 - 0.62$ и продолжить вычисления; если $y_i \geq 0$, то вычислить и вывести $b = 1/x_i^2$ и продолжить вычисления.

6. Дан массив X , содержащий 16 элементов. Вычислить и вывести значения d_i , где $d_i = \frac{e^{x_i} + 2e^{-x_i}}{\sqrt{5 + \sin x_i}}$ и значения $d_i > 0.1$.

7. Дан массив Y , содержащий 25 элементов. Записать в массив R и вывести значения элементов, вычисляемые по формуле $r_i = \frac{5y_i + \cos^2 y_i}{2.35}$, $i=1,2,\dots,25$.

8. Дан массив F , содержащий 18 элементов. Вычислить и вывести элементы нового массива $p_i = 0.13f_i^3 - 2.5f_i + 8$. Вывести отрицательные элементы массива P .

9. Вычислить и вывести элементы массива Z , где $z_i = i^2 + 1$, если i – нечетное, и $z_i = 2i - 1$, если i – четное. Сформировать и вывести массив D : $d_i = 2.5z_i$, если $z_i < 2.5$ и $d_i = z_i / 2.5$, если $z_i \geq 2.5$.

10. Заданы массивы D и E . Вычислить и вывести значения $f_i = (2d_i + \sin e_i) / d_i$, где $i=1,2,\dots,16$; вывести $1 < f_i < 3$.

11. В массиве R , содержащем 25 элементов, заменить значения отрицательных элементов квадратами значений, значения положительных увеличить на 7, а нулевые значения оставить без изменения. Вывести массив R .

12. Дан массив A целых чисел, содержащий 30 элементов. Вычислить и вывести сумму тех элементов, которые кратны 5.

13. Дан массив A целых чисел, содержащий 30 элементов. Вычислить и вывести сумму тех элементов, которые нечетны и отрицательны.

14. Дан массив A целых чисел, содержащий 30 элементов. Вычислить и вывести сумму тех элементов, которые удовлетворяют условию $|a_i| < i^2$.

15. Дан массив A целых чисел, содержащий 30 элементов. Вычислить и вывести количество и сумму тех элементов, которые делятся на 5 и не делятся на 7.

16. Дан массив A вещественных чисел, содержащий 25 элементов. Вычислить и вывести число отрицательных элементов и число членов, принадлежащих отрезку $[1,2]$.

17. Дан массив C , содержащий 23 элемента. Вычислить и вывести среднее арифметическое всех значений $c_i > 3.5$.

18. Дан массив Z целых чисел, содержащий 35 элементов. Вычислить и вывести $R = S + P$, где S – сумма четных элементов, меньших 3, P – произведение нечетных элементов, больших 1.

19. Дан массив Q натуральных чисел, содержащий 20 элементов. Найти и вывести те элементы, которые при делении на 7 дают остаток 1, 2 или 5.

20. Дан массив Q натуральных чисел, содержащий 20 элементов. Найти и вывести те элементы, которые обладают тем свойством, что корни уравнения $q_i^2 + 3q_i - 5 = 0$ действительны и положительны.

21. Дан массив, содержащий 10 элементов. Вычислить произведение элементов, стоящих после первого отрицательного элемента. Вывести исходный массив и результат вычислений.

22. Дан массив, содержащий 14 элементов. Вычислить сумму элементов, стоящих до первого отрицательного элемента. Вывести исходный массив и результат вычислений.

23. Дан массив содержащий 12 элементов. Все четные элементы сложить, вывести массив и результат.

24. Дан массив, содержащий 15 элементов. Все положительные элементы возвести в квадрат, а отрицательные умножить на 2. Вывести исходный и полученный массив.

25. Дан массив, содержащий 14 элементов. Все отрицательные элементы заменить на 3. Вывести исходный и полученный массив.