

ТЕМА 3 ОДНОМЕРНЫЕ МАССИВЫ В ЯЗЫКЕ C#

3.1 Цель третьей темы

Изучение способов организации данных в виде одномерных массивов языка C#. Приобретение практических навыков в программировании на языке C# задач, использующих одномерные массивы.

3.2 Теоретические сведения

3.2.1 Понятие массива

Массив задает способ организации переменных одного типа. Иногда массивом называют упорядоченную совокупность переменных одного типа. Каждая переменная массива имеет индекс – номер переменной в массиве. В языке C#, как и во многих других языках, индексы задаются целочисленным типом. Первая переменная массива имеет 0 индекс, N-я переменная – N-1 индекс.

В некоторых языках программирования при объявлении массива задается число переменных массива. Если число элементов массива известно в момент его объявления и ему может быть выделена память еще на этапе трансляции, то такие массивы называются статическими.

Работа с массивами в языке C# выполняется в два этапа. На первом этапе объявляется тип переменных, объединяемых в массив. На втором этапе – во время работы программы выполняется инициализация массива, т.е. определяется число элементов массива. Поэтому все массивы в языке C# являются динамическими. При инициализации массива ему, как правило, выделяется непрерывная область памяти в куче.

3.2.2 Объявление массива

Объявление одномерных массивов имеет следующий формат:

`<тип>[]<объявители>;`

где `<тип>` – тип переменных, объединяемых в массив;

`[]` – признак одномерного массива;

`<объявители>` – перечень переменных, объявляемых как переменные массива. Каждый объявитель может быть именем или именем с инициализацией. В первом случае речь идет об отложенной инициализации. Нужно понимать, что при объявлении с отложенной инициализацией сам массив не формируется, а создается только ссылка на массив, имеющая неопределенное значение. Поэтому пока массив не будет реально создан и его элементы инициализированы, использовать его в вычислениях программы нельзя.

Например, объявление трех массивов с отложенной инициализацией имеет следующий вид:

`int[] a, b, c;`

Можно каждый массив с отложенной инициализацией объявлять отдельно, например:

```
int[] a;
int[] b;
int[] c;
double[] d;
```

Квадратные скобки приписаны не к имени переменной, а к типу. Они являются неотъемлемой частью определения типа и запись `int[]` следует понимать как тип, задающий одномерный массив с элементами целого типа.

Что же касается границ изменения индексов, то эта характеристика не является принадлежностью типа, она является характеристикой переменных данного типа — экземпляров, каждый из которых является одномерным массивом со своим числом элементов, задаваемых в объявителе переменной.

3.2.2 Инициализация массива

Если массив объявляется без инициализации, то создается только ссылка на массив со значением `void`.

Если инициализация выполняется конструктором, то в динамической памяти создается сам массив (выделяется место под массив), элементы которого инициализируются константами соответствующего типа (ноль для арифметики, пустая строка для строковых массивов), и ссылка связывается с этим массивом.

Например:

//объявление массивов с отложенной инициализацией

```
int[] u;
u = newint[3];
```

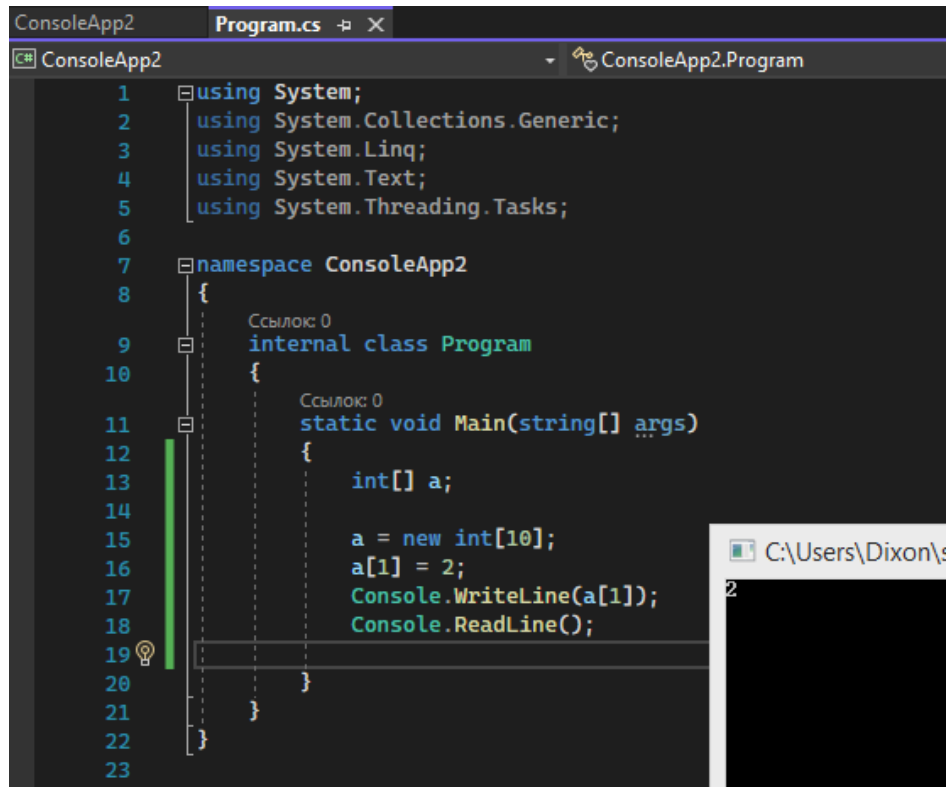
В приведенном примере первоначально объявляется массив целого типа, а затем (во время работы конструктора массива — `new`) в куче выделяется место под массив на три значения целого типа, в которые записываются нули.

Если массив инициализируется константным массивом, то в памяти создается константный массив, с которым и связывается ссылка. Элементы константного массива необходимо заключать в фигурные скобки, например:

```
int[] x = {5,5,6,6,7,7}; //размерность вычисляется, new подразумевается
```

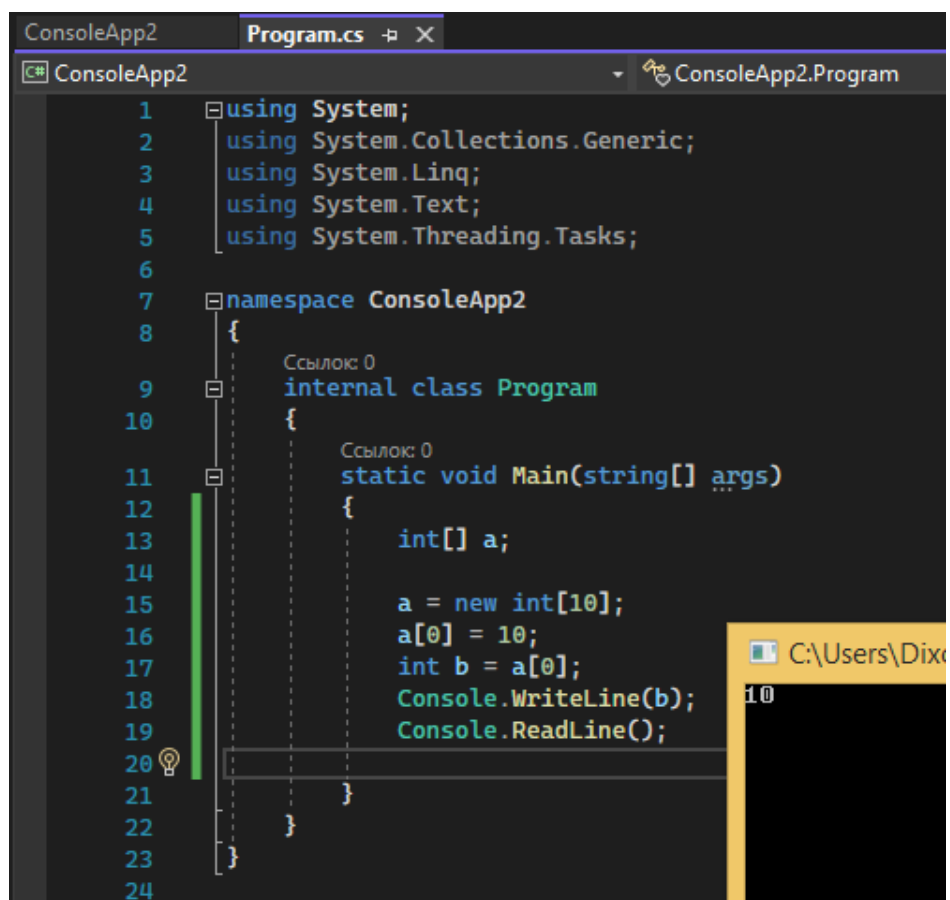
```
int[] x = newint[] {5,5,6,6,7,7}; // размерность вычисляется
```

```
int[] x = newint[6] {5,5,6,6,7,7}; // считается избыточным описанием без ошибки
```



```
1 using System;
2 using System.Collections.Generic;
3 using System.Linq;
4 using System.Text;
5 using System.Threading.Tasks;
6
7 namespace ConsoleApp2
8 {
9     internal class Program
10     {
11         static void Main(string[] args)
12         {
13             int[] a;
14
15             a = new int[10];
16             a[1] = 2;
17             Console.WriteLine(a[1]);
18             Console.ReadLine();
19         }
20     }
21 }
22
23
```

C:\Users\Dixon\... 2



```
1 using System;
2 using System.Collections.Generic;
3 using System.Linq;
4 using System.Text;
5 using System.Threading.Tasks;
6
7 namespace ConsoleApp2
8 {
9     internal class Program
10     {
11         static void Main(string[] args)
12         {
13             int[] a;
14
15             a = new int[10];
16             a[0] = 10;
17             int b = a[0];
18             Console.WriteLine(b);
19             Console.ReadLine();
20         }
21     }
22 }
23
24
```

C:\Users\Dixon\... 10

```

1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Text;
5  using System.Threading.Tasks;
6
7  namespace ConsoleApp2
8  {
9      internal class Program
10     {
11         static void Main(string[] args)
12         {
13             int[] a;
14
15             a = new int[55];
16             Console.WriteLine(a.Length);
17             Console.ReadLine();
18         }
19     }
20 }

```

Console output: 55

3.3 Пример выполнения задания на лабораторную работу

В качестве примера выполнения задания на тему массивы языка C# выбрано решение простой задачи из списка индивидуальных заданий прошлых лет.

```

1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Text;
5  using System.Threading.Tasks;
6
7  namespace ConsoleApp2
8  {
9      internal class Program
10     {
11         static void Main(string[] args)
12         {
13             int[] numbers = { 1, 2, 3, 4, 5 };
14             for (int i = 0; i < numbers.Length; i++)
15             {
16                 Console.WriteLine(numbers[i]);
17             }
18             Console.ReadLine();
19         }
20     }
21 }

```

Console output: 1, 2, 3, 4, 5

Задача 3.1 Сформировать массив 10 целых случайных чисел. Напечатать его.

```

11 static void Main(string[] args)
12 {
13     int arraySize = 10;
14     int[] array = new int[arraySize];
15     Random random = new Random();
16     for (int i = 0; i < array.Length; i++)
17     {
18         array[i] = random.Next(10);
19     }
20     Console.WriteLine("Случайно заполненный массив:");
21     foreach (int number in array)
22     {
23         Console.Write(number + " ");
24     }
25     Console.ReadLine();
26 }
27
28
29
30

```

C:\Users\Dixon\source\repos\ConsoleApp

Случайно заполненный массив:
7 0 7 4 8 9 7 0 2 4

Задача 3.2 Напечатать одномерный массив из 10 элементов. Заполнить массив случайными числами от -40 до 40. Найти максимальное число.

```

static void Main(string[] args)
{
    int size = 10;
    Random random = new Random();
    int[] array = new int[size];
    for (int i = 0; i < size; i++)
    {
        array[i] = random.Next(-40, 41);
    }
    Console.WriteLine("Случайно заполненный массив:");
    for (int i = 0; i < size; i++)
    {
        Console.Write(array[i] + " ");
    }
    Console.ReadLine();

    int maxNumber = array[0];
    for (int i = 1; i < size; i++)
    {
        if (array[i] > maxNumber)
        {
            maxNumber = array[i];
        }
    }
    Console.WriteLine(maxNumber);
    Console.ReadKey();
}

```

C:\Users\Dixon\source\repos\ConsoleApp

Случайно заполненный массив:
-20 -22 0 30 -29 -32 10 -6 -25 39
39

3.5 Индивидуальные задания для СРС

Легкие задания

- 1 Создайте массив из 10 целых чисел и выведите его на экран.
- 2 Заполните массив из 10 элементов случайными числами и выведите его на экран.
- 3 Найдите сумму элементов массива из предыдущего задания.
- 4 Найдите среднее арифметическое элементов массива из предыдущего задания.
- 5 Найдите минимальный элемент в массиве из предыдущего задания.
- 6 Найдите максимальный элемент в массиве из предыдущего задания.
- 7 Отсортируйте массив из предыдущего задания в порядке возрастания и выведите его на экран.
- 8 Отсортируйте массив из предыдущего задания в порядке убывания и выведите его на экран.
- 9 Создайте новый массив, который содержит только четные элементы из массива из предыдущего задания.
- 10 Создайте новый массив, который содержит только нечетные элементы из массива из предыдущего задания.
- 11 Создайте новый массив, в котором элементы массива из предыдущего задания умножены на 2.
- 12 Создайте новый массив, в котором элементы массива из предыдущего задания возведены в квадрат.
- 13 Проверьте, есть ли в массиве из предыдущего задания заданное число. Выведите результат на экран.
- 14 Поменяйте местами максимальный и минимальный элементы в массиве из предыдущего задания.
- 15 Создайте новый массив, который содержит только уникальные элементы из массива из предыдущего задания.
- 16 Подсчитайте количество отрицательных элементов в массиве и выведите его на экран.
- 17 Подсчитайте количество положительных элементов в массиве и выведите его на экран.
- 18 Поменяйте порядок элементов массива на обратный и выведите его на экран.

Сложные задания

- 1 Сформировать массив 25 целых случайных чисел в диапазоне от минус 40 до 60. Напечатать его. Найти 3 maxчисла и вывести их.
- 2 Сформировать массив 30 целых случайных чисел в диапазоне от минус 40 до 40. Напечатать его. Все отрицательные числа поменять на 0 и вывести новый массив.
- 3 Сформировать массив 20 целых случайных чисел в диапазоне от минус 30 до 30. Напечатать его. Заменить все отрицательные элементы массива их квадратами и упорядочить элементы массива по возрастанию. Напечатать новый массив.
- 4 Сформировать массив 20 целых случайных чисел в диапазоне от минус 30 до 30. Напечатать его. Поменять местами максимальный и минимальный элементы массива. Напечатать новый массив.