

# Лекция 9. Массивы

Козьминых Н.М.

дисциплина «Программирование»

# Понятие массива

2

- Массив - структура данных, позволяющая хранить под одним именем совокупность данных любого, но только одного какого-то типа
- С массивом связаны следующие его свойства:
  - имя;
  - тип;
  - размерность;
  - размер.

# Виды массивов

3

- Массивы в C#:
  - одномерные массивы;
  - многомерные массивы;
  - массивы массивов (ступенчатые, зубчатые массивы).

# Одномерные массивы

4

- Объявление:

```
<тип>[ ] <объявители>;
```

```
int[ ] a, b, c;
```

- Объявление с инициализацией:

- Явная инициализация:

```
double[ ] x = {5.5, 6.6, 7.7};
```

- Инициализация с вызовом конструктора массива (ноль для арифметики, пустая строка для строковых массивов):

```
int[ ] d = new int[5];
```

# Одномерные массивы

## Пример

5

- Инициализация массива:

- с использованием ключевого слова **new**:

```
int[] myArr = new int[] {10,20,30,40,50};
```

- без использования ключевого слова **new**:

```
string[] info = { "Фамилия", "Имя", "Отчество" };
```

- с использованием **new** и указанием **размера**:

```
char[] symbol = new char[4] { 'X','Y','Z','M' };
```

# Одномерные массивы

## Пример

6

```
// Инициализация массив
```

```
int[] arr = new int[3];
```

```
// Определение значений элементов массива вручную
```

```
arr[0] = 100;
```

```
arr[1] = 23;
```

```
arr[2] = -6;
```

<b>№</b> (номер элемента массива)	<b>1</b>	<b>2</b>	<b>3</b>
<b>i</b> (индекс элемента массива)	<b>0</b>	<b>1</b>	<b>2</b>
<b>arr[i]</b> (значение элемента массива)	100	23	-6

# СВОЙСТВО Length

7

- Свойство Length – содержит значение количества элементов, из которых состоит массив

```
int[] arr = new int[4];
```

```
for (int i = 0; i < arr.Length; i++)
```

```
    a[i] = (i + 1) * 2;
```

<b>i</b>	<b>0</b>	<b>1</b>	<b>2</b>	<b>3</b>
<b>arr[i]</b>	2	4	6	8

# Класс Random

8

- Класс Random содержит средства для генерации случайных чисел
- Две реализации конструктора:
  - генерация **неповторяющейся** серии случайных чисел – начальный элемент строится на основе текущей даты и времени  
**public Random()**
  - генерация **повторяющейся** серии случайных чисел  
**public Random (int)**

# Методы класса Random

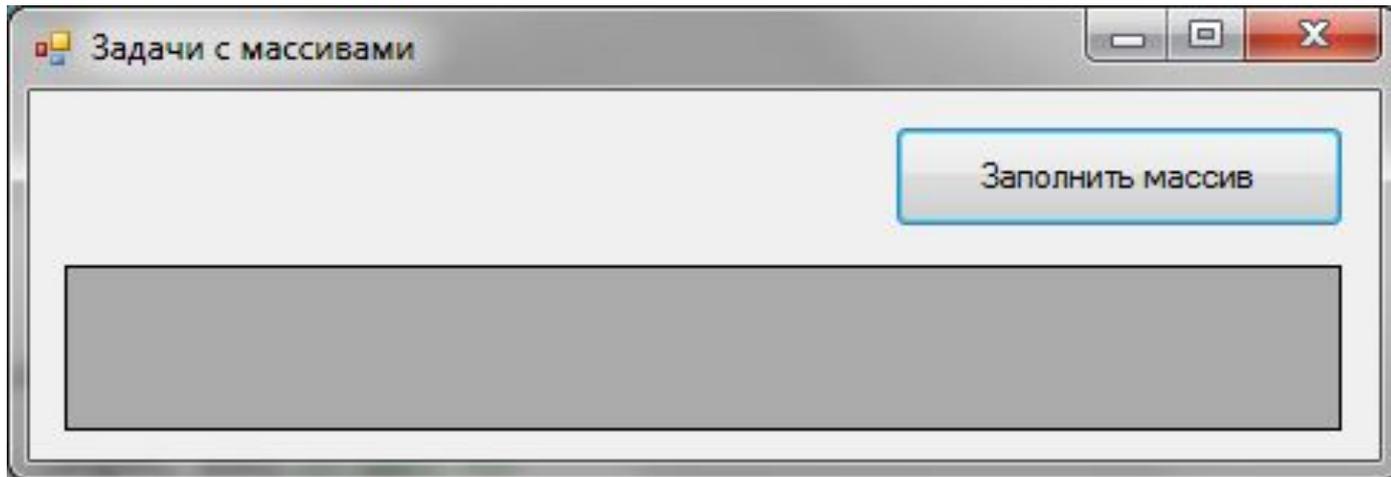
9

Метод	Описание
<b>public int Next()</b>	выдает целые положительные числа во всем положительном диапазоне типа <code>int</code>
<b>public int Next (int max)</b>	выдает целые положительные числа в диапазоне <code>[0,max]</code>
<b>public int Next (int min, int max)</b>	выдает целые положительные числа в диапазоне <code>[min,max]</code>
<b>public double NextDouble()</b>	выдается новое случайное число, равномерно распределенное в интервале <code>[0-1]</code>
<b>public void NextBytes (byte[ ] buffer)</b>	параметр <b>buffer</b> представляет массив байтов, то, естественно, генерированные случайные числа находятся в диапазоне <code>[0, 255]</code>

# Одномерные массивы

## Пример

10



```
private void btn_Click(object sender, EventArgs e){
    Random rand = new Random();
    int[] arr = new int[8];

    for (int i = 0; i < arr.Length; i++)
        arr[i] = rand.Next(-10, 10);

    grv.Rows.Clear();
    grv.RowCount = 1;
    grv.ColumnCount = arr.Length;

    for (int i = 0; i < arr.Length; i++)
    {
        grv.Columns[i].Name = (i + 1).ToString();
        grv.Columns[i].Width = 30;
        grv.Rows[0].Cells[i].Value = (arr[i]).ToString();
    }
}
```

# Многомерные массивы

12

- *Многомерным* называется такой массив, который отличается двумя или более измерениями, причем доступ к каждому элементу такого массива осуществляется с помощью определенной комбинации двух или более индексов
- Многомерный массив индексируется двумя и более целыми числами

# Двумерные массивы

13

- Простейшей формой многомерного массива является **двумерный массив**
- Местоположение любого элемента в двумерном массиве обозначается двумя индексами
- Двумерный массив можно представить в виде таблицы, на строки которой указывает один индекс, а на столбцы — другой

# Двумерные массивы

14

```
// Инициализация массива
```

```
int[,] myArr = new int[4, 5];
```

```
Random ran = new Random();
```

```
// Определение значений элементов массива  
случайным образом
```

```
for (int i = 0; i < 4; i++) {  
    for (int j = 0; j < 5; j++) {  
        myArr[i, j] = ran.Next(1, 15);  
    }  
}
```

# Двумерные массивы

15

	0	1	2	3	4
0	13	10	3	7	11
1	7	12	11	11	8
2	13	6	12	13	2
3	13	5	11	6	2

Левый индекс

Правый индекс

`myArr[2,3]`

# Массивы трех и более измерений

16

- Общая форма инициализации многомерного массива:

**тип[,... ] имя\_массива = new тип  
[размер1, размер2, ... размеры];**

```
int[, ,] myArr = new int[5,5,5];  
for (int i = 0; i < 5; i++)  
    for (int j = 0; j < 5; j++)  
        for (int k = 0; k < 5; k++)  
            myArr[i, j, k] = i + j + k;
```

# Инициализация многомерных массивов

17

```
тип[,] имя_массива = {  
    {val, val, val, ..., val},  
    { val, val, val, ..., val},  
    {val, val, val, ..., val}  
};
```

```
int[,] myArr = {  
    {1,10},  
    {2,20},  
    {3,30},  
    {4,40}  
};
```

# Массивы массивов (ступенчатые массивы)

18

- Ступенчатый массив представляет собой массив массивов, в котором длина каждого массива может быть разной

- Объявление:

**тип [][] имя массива = new тип[размер] [];**

, где размер обозначает число строк в массиве

# Массивы массивов (ступенчатые массивы)

19

```
int i = 0;
// Инициализация ступенчатого массива
int[][] myArr = new int[4][];
myArr[0] = new int[4];
myArr[1] = new int[6];
myArr[2] = new int[3];
myArr[3] = new int[4];
// Определение значений элементов массива
for (; i < 4; i++) myArr[0][i] = i;
for (i = 0; i < 6; i++) myArr[1][i] = i;
for (i = 0; i < 3; i++) myArr[2][i] = i;
for (i = 0; i < 4; i++) myArr[3][i] = i;
```

# Массивы массивов (ступенчатые массивы)

20

0	1	2	3		
0	1	2	3	4	5
0	1	2			
0	1	2	3		

`myArr[1][4]`

21

# Спасибо за внимание

Вопросы...