

# ОСНОВЫ ПРОГРАММИРОВАНИЯ

Самойлов Михаил Юрьевич

# Задачи

- ▶ Задать массив из  $N$  случайных чисел, вывести на экран элементы массива через пробел.
- ▶ Задать массив из  $N$  чисел с клавиатуры, переставить местами первый и последний элемент и вывести элементы массива на экран.
- ▶ Задать массив из  $N$  случайных чисел, вывести на экран в первой строке положительные элементы массива через пробел, во второй отрицательные.
- ▶ Задать массив из  $N$  случайных чисел, вывести на экран элементы массива через пробел, найти максимальный элемент и сколько раз он встречается.
- ▶ Задать массив из  $N$  случайных чисел, вывести на экран элементы массива через пробел, найти количество положительных и отрицательных чисел.

# Многомерный массив

Массивы бывают многомерными. В предыдущих примерах мы создавали одномерные массивы, а теперь также создадим двухмерный:

```
int[] nums1 = new int[] { 0, 1, 2, 3, 4, 5 };
```

```
int[,] nums2 = { { 0, 1, 2 }, { 3, 4, 5 } };
```

# Многомерный массив

Визуально оба массива можно представить следующим образом:

Одномерный массив `nums1`

0 1 2 3 4 5

Двухмерный массив `nums2`

0 1 2

3 4 5

# Многомерный массив

Поскольку массив `nums2` двумерный, он представляет собой простую таблицу. Объявление трехмерного массива могло бы выглядеть так:

```
int[, ,] nums3 = new int[2, 3, 4];
```

# Многомерный массив

Все возможные способы определения двумерных массивов:

- ▶ `int[,] nums1;`
- ▶ `int[,] nums2 = new int[2, 3];`
- ▶ `int[,] nums3 = new int[2, 3] { { 0, 1, 2 }, { 3, 4, 5 } };`
- ▶ `int[,] nums4 = new int[,] { { 0, 1, 2 }, { 3, 4, 5 } };`
- ▶ `int[,] nums5 = new [,] { { 0, 1, 2 }, { 3, 4, 5 } };`
- ▶ `int[,] nums6 = { { 0, 1, 2 }, { 3, 4, 5 } };`

# Многомерный массив

Определенную сложность может представлять перебор многомерного массива. Прежде всего надо учитывать, что длина такого массива - это совокупное количество элементов.

```
int[,] mas = { { 1, 2, 3 }, { 4, 5, 6 }, { 7, 8, 9 }, { 10, 11, 12 } };
```

```
foreach (int i in mas)
```

```
    Console.Write(i + " ");
```

```
Console.WriteLine();
```

# Многомерный массив

Определенную сложность может представлять перебор многомерного массива. Прежде всего надо учитывать, что длина такого массива - это совокупное количество элементов.

```
int[,] mas = { { 1, 2, 3 }, { 4, 5, 6 }, { 7, 8, 9 }, { 10, 11, 12 } };
```

```
foreach (int i in mas)
```

```
    Console.Write(i + " ");
```

```
Console.WriteLine();
```

В данном случае длина массива mas равна 12. И цикл foreach выводит все элементы массива в строку.



# Многомерный массив

Если нужно отдельно пройти по каждой строке в таблице, то надо получить количество элементов в размерности. У каждого массива есть метод `GetUpperBound(dimension)`, который возвращает индекс последнего элемента в определенной размерности.

Если мы говорить о двухмерном массиве, то первая размерность (с индексом 0) по сути это и есть таблица. И с помощью выражения `mas.GetUpperBound(0) + 1` можно получить количество строк таблицы, представленной двухмерным массивом. А через `mas.Length / rows` можно получить количество элементов в каждой строке

# Многомерный массив

```
int[,] mas = { { 1, 2, 3 }, { 4, 5, 6 }, { 7, 8, 9 }, { 10, 11, 12 } };
```

```
int rows = mas.GetUpperBound(0) + 1;
```

```
int columns = mas.Length / rows;
```

```
for (int i = 0; i < rows; i++)
```

```
{
```

```
    for (int j = 0; j < columns; j++)
```

```
    {
```

```
        Console.Write(mas[i, j] + "\t");
```

```
    }
```

```
    Console.WriteLine();
```

```
}
```

# Массив массивов

От многомерных массивов надо отличать массив массивов:

```
int[][] nums = new int[3][];  
nums[0] = new int[2] { 1, 2 };      // выделяем память для первого подмассива  
nums[1] = new int[3] { 1, 2, 3 };   // выделяем память для второго подмассива  
nums[2] = new int[5] { 1, 2, 3, 4, 5 }; // выделяем память для третьего подмассива
```

# Массив массивов

Здесь две группы квадратных скобок указывают, что это массив массивов, то есть такой массив, который в свою очередь содержит в себе другие массивы. В данном случае у нас массив `pims` содержит три массива. Причем размерность каждого из этих массивов может не совпадать.

1 2

1 2 3

1 2 3 4 5

# Массив массивов

Здесь две группы квадратных скобок указывают, что это массив массивов, то есть такой массив, который в свою очередь содержит в себе другие массивы. В данном случае у нас массив `nums` содержит три массива. Причем размерность каждого из этих массивов может не совпадать.

1 2

1 2 3

1 2 3 4 5