

Контрольные вопросы

1. Массив на языке C# объявляется с помощью

типа переменной, `[]` (где `square` — квадрат) и `[,]` (где `square` — квадрат), название массива

Например

```
int[] array = new int[9] (записанный вручную)  
int[] array = new int[] {1, 3, 2, -1, -3, -2, 1, -2, 3}  
int[] array = new[] {1, 4, 5, 2, 3}  
int[] array = {1, 4, 5, 2, 3}
```

2. Длина массива можно узнать с помощью метода `.Length`, а длину строки или массива `.GetLength()` (где `1` — индекс) и `.GetLength(1)` (где `2` — индекс)

3. `foreach (int element in array)`

`element` принимает элементы массива как переменные цикла, соответствующие номеру итерации.

int[,] array;

int[,] array = new int[2,3];

int[,] array = new int[5,5] { {1,2,3,4,5}, {5,4,3,2,1} }

int[,] array = new int[2,2] { {1,2,3}, {4,5,6} }

int[,] array = new [2,2] { {1,2,3}, {4,5,6} }

int[,] array = { {1,2,3}, {4,5,6} }

[,] - означает анонимно двумерное массив. [7,5] - у этого массива 7 строк и 5 столбцов.

5. Параметр по умолчанию - это параметр, имеющий значение по умолчанию. Если позволить передать значение, то это значение используется вместо default.

Пользовательские ср-ва - шаблоны, использующие где всегда один строк. Возвращает значение, может использоваться как выражение.

Пример: static float sum (float a, float b) {

static - модификатор float - ман возвращаемого

значения sum - название (float a, float b) - аргументы

{ } - тело.

Если функция не возвращает значение,
то имеет void, вместо типа.

6. ^{параметр} in - указывает, что ^{параметр} переменная может принимать
переменные кем-то определены.

ref указывает, что этот параметр передается
по ссылке и может быть считан или записан
вызывающим местом.

- out указывает, что этот параметр передается
по ссылке и записывается вызывающим местом.

Параметры, без ^{ссылки} этих слов, передаются в
вызывающее место по значению. Это значение
может быть изменено в месте, откуда не будет
сохранено.

7. Рекурсивная ф-ция - это ф-ция, вызывающая
саму себя. В себе она должна содержать
оператор if, чтобы было возможным выйти
из рекурсивной функции. Все рекурсивные вызовы
связаны с подпрограммой, сходящие к изнач. вызову.