

# Программирование на языке Си#

## **Тема 9. Циклы и массивы**

# Операторы организации циклов <sup>2</sup>

Под *циклом* понимается многократное выполнение одних и тех же операторов при различных значениях промежуточных данных. Число повторений может быть задано в явной или неявной форме.

К операторам цикла относятся:

- *цикл с предусловием* while
- *цикл с параметром* for
- *цикл с постусловием* do while
- *цикл перебора* foreach

## Свойства элементов управления:

Items	С помощью этого свойства можно получить ссылку на список элементов, хранящихся в настоящее время в элементе управления (например, <code>ListBox</code> ). С помощью этой ссылки можно добавлять и удалять элементы, а также определять число элементов в коллекции.
-------	---

## Методы для работы с массивами

Length	Свойство, которое возвращает целое число, представляющее общее число элементов во всех измерениях массива.
Max()	Возвращает максимальное значение, содержащееся в массиве.
Min()	Возвращает минимальное значение, содержащееся в массиве.
Reverse()	Изменяет порядок элементов массива на противоположный.
Sort()	сортировка элементов

## Методы добавления элементов:

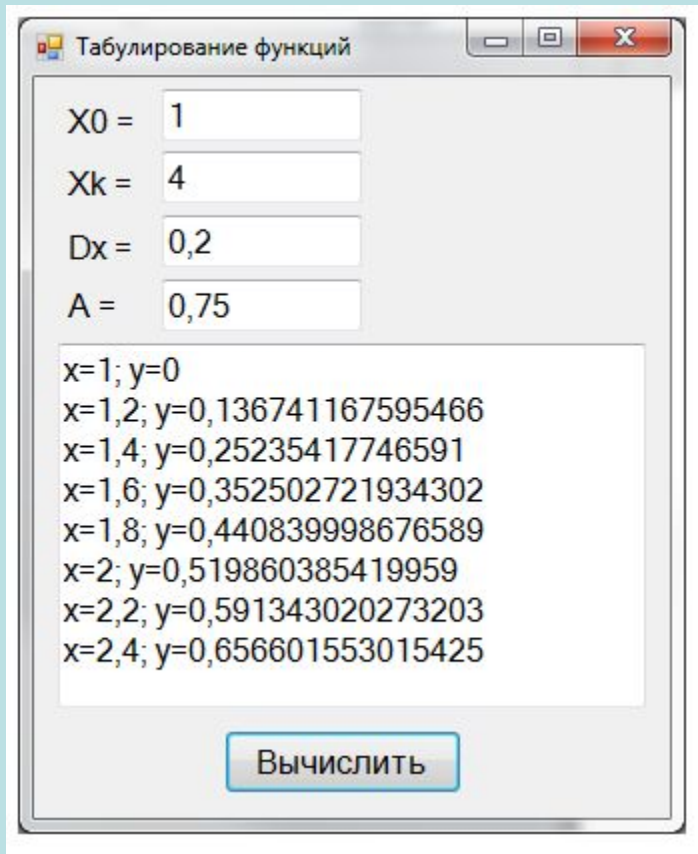
Метод	Описание
Add([элемент])	добавляет элемент в конец списка
AddRange([список элементов])	добавляет в конец списка элементы указанного списка
Insert([индекс],[элемент])	вставляет элемент на позицию соответствующую индексу, все элементы «правее» будут сдвинуты на одну позицию
InsertRange([индекс], [список элементов])	то же самое, только вставляется множество элементов

## Методы удаления элементов

Remove([элемент])	удаляет первое вхождение указанного элемента из списка
RemoveRange([индекс], [количество])	удаляет указанное количество элементов, начиная с указанной позиции
RemoveAt([индекс])	удаляет элемент, который находится на указанной позиции
Clear()	удаляет все элементы списка

# Цикл с предусловием while

Задание 1: Вычислить и вывести на экран таблицу значений функции  $y=a*\ln(x)$  при  $x$ , изменяющемся от  $x_0$  до  $x_k$  с шагом  $dx$ ,  $a$ – константа.



Табулирование функций

X0 = 1  
Xk = 4  
Dx = 0,2  
A = 0,75

```
x=1; y=0  
x=1,2; y=0,136741167595466  
x=1,4; y=0,25235417746591  
x=1,6; y=0,352502721934302  
x=1,8; y=0,440839998676589  
x=2; y=0,519860385419959  
x=2,2; y=0,591343020273203  
x=2,4; y=0,656601553015425
```

Вычислить

```
private void button1_Click(object sender, EventArgs e)
```

```
{  
    // Считывание начальных данных  
    double x0 = Convert.ToDouble(textBox1.Text);  
    double xk = Convert.ToDouble(textBox2.Text);  
    double dx = Convert.ToDouble(textBox3.Text);  
    double a = Convert.ToDouble(textBox4.Text);
```

```
// Цикл
```

```
double x = x0;
```

```
while (x <= xk )
```

```
{
```

```
    double y = a * Math.Log(x);
```

```
    textBox5.Text = textBox5.Text + "x=" + Convert.ToString(x) + "; y=" + Convert.ToString(y) + "\r\n";
```

```
    x = x + dx;
```

```
}
```

```
}
```

**Ввод данных**

**Цикл с условием**

**Вычисление**

**Вывод**

# Ввод данных в массив

1 способ: по одному (ввод в поле TextBox )

ссылка: 1

```
private void button1_Click(object sender, EventArgs e)
{
    if (x<3)
    {
        Mas[x]= Convert.ToInt32(textBox1.Text);
        x = x + 1;
        textBox1.Clear();
    }
}
```

# Ввод данных в массив

2 способ (генератор случайных чисел):

```
Random rand = new Random();// Инициализируем класс случайных чисел

// Генерируем и выводим 15 элементов
for (int i = 0; i < 15; i++)
{
    Mas[i] = rand.Next(-50, 50);
}
```



# Ввод данных в массив

3 способ (генератор случайных символов):

ссылка: 1

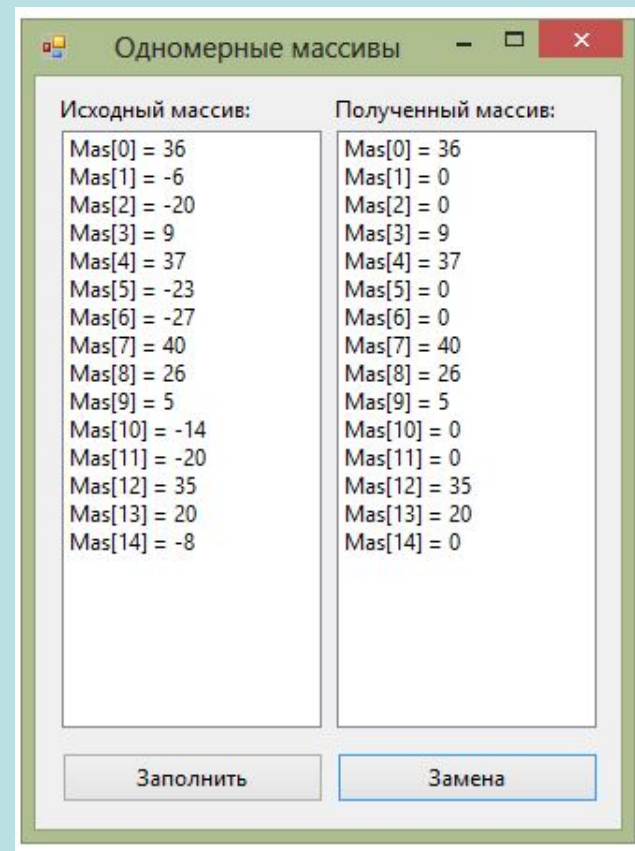
```
private void button1_Click(object sender, EventArgs e)
{
    string symbols = "abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ";
    Random r = new Random();
    while (x < 3)
    {
        k = r.Next(0, 52);
        Mas[x] = symbols.Substring(k,1);
    }
}
```

# Цикл с параметром for

Задание 2:

В одномерном массиве заполнить 15 элементов случайными числами от -50 до +50. Затем заменить все отрицательные числа нулями.

Создайте форму с элементами управления. Опишите одномерный массив. Создайте обработчики события для кнопок.



```
int[] Mas = new int[15]; // Глобальная переменная видна всем методам
```

ссылка: 1 | 0 изменений | 0 авторов, 0 изменений

```
private void button2_Click(object sender, EventArgs e) // Заполнение исходного массива
```

```
{  
    listBox1.Items.Clear(); // Очищаем элемент управления
```

```
    Random rand = new Random(); // Инициализируем класс случайных чисел
```

```
    // Генерируем и выводим 15 элементов
```

```
    for (int i = 0; i < 15; i++)
```

```
    {
```

```
        Mas[i] = rand.Next(-50, 50);
```

```
        listBox1.Items.Add("Mas[" + i.ToString() + "] = " + Mas[i].ToString());
```

```
    }
```

```
}
```

**Ввод данных**

**Вывод массива в listBox**

ссылка: 1 | 0 изменений | 0 авторов, 0 изменений

```
private void button3_Click_1(object sender, EventArgs e) // Замена отрицательных элементов нулями
{
    listBox2.Items.Clear(); // Очищаем элемент управления
    // Обрабатываем все элементы
    for (int i = 0; i < 15; i++)
    {
        if (Mas[i] < 0)
        {
            Mas[i] = 0;
        }
        listBox2.Items.Add("Mas[" + i.ToString() + "] = " + Mas[i].ToString());
    }
}
```

**Проверка <0  
и замена**

**Вывод массива в listBox**

## Задания:

3. В одномерном массиве R заполнить 25 элементов случайными числами от -100 до +250, заменить значения отрицательных элементов квадратами значений, значения положительных увеличить на 7, а нулевые значения оставить без изменения. Вывести получившийся массив R.

4. Дан массив A целых чисел, содержащий 30 элементов. Вычислить и вывести сумму тех элементов, которые нечетны и отрицательны. Способ заполнения массива выбрать самостоятельно.