

Тел: +7 (707) 900 92 67  
Почта: saken.yan@yandex.com

### 3 ЛЕКЦИЯ

## C-SHARP АҚПАРАТТЫҚ БАҒДАРЛАМАЛАУ ТІЛІ

### §3. C-Sharp тіліндегі массивтер.

#### Массивтер.

C# тілінде массив дегеніміз барлығына ортақ аты бар, бір типті айнымалылар жиынтығын айтады.

C# тілінде массивті қолдану үшін екі сатылы процедура жасау қажет, себебі C# массивті объект ретінде қарастырады. Алдымен массивтің атын жариялау керек. Содан соң, «new» операторының көмегімен массивтің объектісін құру керек.

Бір өлшемді массивті с # тілінде жариялаудың жалпы синтаксисін келесі түрде жазуға болады:

```
Тип [] Имя массива = new Тип [количество содержимого элементов];
```

Бір өлшемді массивті с # тілінде жариялаудың мысалы:

```
// int типті, аты array болатын, ұзындығы 5 элементтен тұратын массив жариялаймыз.  
int[] array = new int[5];  
  
// Массивтің әр элементінің қолмен инициализация жасаймыз.  
array[0] = 100;  
array[1] = 23;  
array[2] = 25;  
array[3] = 31;  
array[4] = 1;  
  
// Цикл ішінде массивтің әр элементінің қолданамыз.  
for (int i = 0; i < 5; i++)  
{  
    Console.WriteLine("Array[" + i + "] = " + array[i]);  
}
```

Сондай ақ, массив элементтерін нөмері бойынша инициализация жасаумен қатар арнайы инициализациялау синтаксистері арқылы толтыруға болады. Ол үшін фигурлық жақша ішінде әр элементтерді үгір арқылы толтырады. Бұл жол қолайлы болып табылады егер массив аз ғана элементтерден құралған болса және массивті тез арада толтыру қажет болса.

## Мысал

```
// Массивті new операторы арқылы толтырудың арнайы инициализациялау синтаксисі
int[] array = new int[] {10,20,30,40,50};

// Массивті new операторын қолданбай толтырудың арнайы инициализациялау синтаксисі
string[] info = { "Фамилия", "Имя", "Отчество" };

// Массивті new операторы арқылы және ұзындығын көрсету арқылы толтыру
char[] symbols = new char[4] { 'X', 'Y', 'Z', 'M' };
```

## Length қасиеті

Әр массивтің **Length** деген қасиеті бар. Бұл қасиет арқылы массивтің элементтер санын (ұзындығы) анықтай аламыз. **Length** қасиетіне келесі мысалды қарастырайық:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace ConsoleApplication1
{
    class Program
    {
        static void Main(string[] args)
        {
            int[] array = { 1, 2, 3, 4 };

            for (int i = 0; i < array.Length; i++)
                Console.WriteLine(array[i]);

            Console.ReadLine();
        }
    }
}
```

Массивтерді біз практикада математикалық функцияларды файлдан оқуғанда, файлға фазғанда, матрицалармен жұмыс жасағанда және де т.б. көптеген жағдайларда қолданатын боламыз. Енді массивті қолдануға мысал қарастырайық:

$y = x^3 - x^2$  деген функцияның мәндерін 0 ден бастап 1-ге дейін 0.1 деген қадаммен анықтап шығып, мәндерін массивке жазайық.

```
static void Main(string[] args)
{
    double[] y = new double[11];
    double[] x = {0.0,0.1,0.2,0.3,0.4,0.5,0.6,0.7,0.8,0.9,1.0};

    for (int i = 0; i < x.Length; i++)
        y[i] = Math.Pow(x[i],3) - Math.Pow(x[i],2);

    Console.WriteLine("Our defined function:");

    for (int i = 0; i < x.Length; i++)
        Console.WriteLine("x[" + i + "]=" + x[i] + ";y[" + i + "]=" + y[i]);

    Console.ReadLine();
}
```

Программаны қолданушы геометриялық қатардың бірінші мүшесін  $b_1$ , 0 мен 1 аралығында жататын қатардың өсімшесін  $q$ , және қатардың соңғы мүшесінің реттік нөмерін  $n$  еңгізеді. Программа қатардың барлық мүшелерін және барлық  $n$  мүшелерінің қосындысын консолға шығарсын.

```
static void Main(string[] args)
{
    Console.WriteLine("Please, Enter the first term of the series:");
    double b1 = 0; bool Checker = true;

    while (Checker){
        try{
            b1 = double.Parse(Console.ReadLine());
            Checker = false;
        } catch (Exception e){
            Console.WriteLine("Wrong value! Please, Enter again");
            Console.WriteLine(e.Message);
        } //catch
    } //while

    double q = 0; Checker = true;
    Console.WriteLine("Please, Enter the q");

    while (Checker){
        try{
            q = double.Parse(Console.ReadLine());
            if (q >= 1 || q <= 0) throw new Exception("the of range q (0,1)");
            Checker = false;
        } catch (Exception e){
            Console.WriteLine("Wrong value! Please, Enter again");
            Console.WriteLine(e.Message);
        } //catch
    } //while

    uint n = 0; Checker = true;
    Console.WriteLine("Please, Enter the n");

    while (Checker){
        try{
            n = uint.Parse(Console.ReadLine());
            Checker = false;
        } catch (Exception){
            Console.WriteLine("Wrong value! Please, Enter again");
        } //catch
    } //while

    double[] b = new double[n+1];
    b[0] = 1;

    for (int i = 1; i < b.Length; i++){
        b[i] = b[i-1]*q;
        Console.WriteLine("b[" + i + "]=" + b[i]);
    } //for

    double sum = 0;
    for (int i = 1; i < b.Length; i++) sum = sum + b[i];

    Console.WriteLine("s = " + sum);
    Console.ReadLine();
}
```

Студент өз бетімен орындайтын тапсырма. Есеп №1.

1.1. Аты *RandArray* деп аталатын `int` типтес 100 элементке арналған массивті жариялау.

1.2. Осы *RandArray* массивін мәндері 0 мен 1000 аралығында жататын кездейсоқ шамалармен толтыру. (с# тілінде кездейсоқ шамалар **Random** деген класс арқылы құрылады.)

**Нұсқау:** **Random** класы туралы және оның қасиеттері мен әдістері (методы класста) туралы ақпаратты интернетті қолданып, тауып алыңыз.

1.3. *RandArray* массивін кездейсоқ шамалармен толтырғаннан соң, осы элементтердің орташа мәнін есептеп табыңыз.

**СОӨЖ№1.**

1. `List<T>` - бір байланысты тізімдер (односвязный список, Linked List) туралы және олардың қасиеттері туралы оқып, қысқаша конспект жазыңыздар.

2. Үшінші лекциядағы **Есеп №1**-ді (жоғарыда келтірілген) массивтердің орнына `List<T>` қолданып программаны қайта құрыңыздар.