**Практическая работа № 2**

**ИСПОЛЬЗОВАНИЕ НЕКОТОРЫХ КЛАССОВ C# В ПРИЛОЖЕНИЯХМетоды класса System.Array**

Массивы в C# основаны на классе System.Array. У всех классов, являющихся массивами, много общего, поскольку все они являются потомками класса System.Array. Класс Array имеет довольно большое число собственных методов и свойств.

Пример:

using System;

namespace test

{

 class Test

 {

 static void Main(string[] args)

 {

 //Объявление массива.

 int[] num = { 4, -5, 2, 0, 23 };

 //Выводим массив.

 foreach (int i in num)

 {

 Console.WriteLine(i.ToString());

 }

 //Переворачиваем массив.

 Console.WriteLine("Перевернутый массив");

 Array.Reverse(num);

 foreach (int i in num)

 {

 Console.WriteLine(i.ToString());

 }

 //Сортируем массив.

 Array.Sort(num);

 Console.WriteLine("Отсортированный массив");

 foreach (int i in num)

 {

 Console.WriteLine(i.ToString());

 }

 //Обнуляем массив.

 Array.Clear(num, 0, 5);

 Console.WriteLine("Обнуленный массив");

 foreach (int i in num)

 {

 Console.WriteLine(i.ToString());

 }

 }

 }

}

Тут используются статические методы класса Array для сортировки, переворачивания и очистки массива. Разумеется, эти методы не единственные.

В классе Array есть, например, еще встроенный статический метод IndexOf предназначенный для поиска элемента в массиве. Пример использования:

int k = 5;

Console.WriteLine("Число {0} находится на {1} месте.", k,

Array.IndexOf(num, k));

Этот метод возвращает индекс искомого элемента (нумерация с нуля). Если такого элемента нет, то возвращается -1.

Обратите внимание, что эти методы действуют для встроенных типов (в нашем примере массив был типа int). Для пользовательских типов данных их применение тоже возможно. Некоторые свойства и методы класса Array приведены в таблицах 2-4.







Пример, использования выше приведенных методов, для работы с массивами:

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

namespace Array1

{

 class Program

 {

 static void Main(string[] args)

 {

 //Объявление массива.

 int[] Arr1 = { 4, -5, 2, 0, 23 };

 //Выводим массив.

 foreach (int i in Arr1)

 {

 Console.WriteLine(i.ToString());

 }

 //Число элементов массива (Arr1.Length)

 Console.WriteLine("Количество элементов массива = " + Arr1.Length);

 //Размерность массива (Rank)

 Console.WriteLine("Размерность массива = " + Arr1.Rank);

 // Верхняя граница массива (GetUpperBound)

 Console.WriteLine("Верхняя граница массива = " + Arr1.GetUpperBound(0));

 // Нижняя граница массива (GetLowerBound)

 Console.WriteLine("Нижняя граница массива = {0}", Arr1.GetLowerBound(0));

 //Прочесть значение 2-го элемента

 Console.WriteLine("Значение 2-го элемента = {0}", Arr1.GetValue(1));

 //Изменить значение 2-го элемента на -11

 Arr1.SetValue(-11, 1);

 Console.WriteLine("Измененное значение 2-го элемента = {0}", Arr1.GetValue(1));

 //Объявление массива Arr2

 //Количество элементов массива определяется как

 //верхняя граница массива Arr1 + 1 (Arr1.GetUpperBound(0) + 1)

 int[] Arr2 = new int[Arr1.GetUpperBound(0) + 1];

 Console.WriteLine("Скопированный массив");

 //Копирование массива Arr1 в Arr2 начиная с 0 элемента

 Arr1.CopyTo(Arr2, 0);

 foreach (int i in Arr2)

 {

 Console.WriteLine(i.ToString());

 }

 //Переворачиваем массив

 Console.WriteLine("Перевернутый массив");

 Array.Reverse(Arr1);

 foreach (int i in Arr1)

 {

 Console.WriteLine(i.ToString());

 }

 // Поиск индекса в несортированном массиве элемента "23"

 Console.WriteLine("Индекс элемента 23 =" + Array.IndexOf(Arr1, 23));

 //Сортируем массив.

 Array.Sort(Arr1);

 Console.WriteLine("Отсортированный массив");

 foreach (int i in Arr1)

 {

 Console.WriteLine(i.ToString());

 }

 // Двоичный поиск индекса в отсортированном массиве элемента "23"

 Console.WriteLine("Индекс элемента 23 =" + Array.BinarySearch(Arr1, 23));

 Console.WriteLine("Индекс элемента 23 после сортировки =" + Array.IndexOf(Arr1, 23));

 //Обнуляем массив.

 Array.Clear(Arr1, 0, 5);

 Console.WriteLine("Обнуленный массив");

 foreach (int i in Arr1)

 {

 Console.WriteLine(i.ToString());

 }

 //Объявление массива float

 //Элементы считываются с клавиатуры

 Console.WriteLine("Введите размер массива");

 int Zise\_Arr3 = int.Parse(Console.ReadLine());

 float[] Arr3 = new float[Zise\_Arr3];

 Console.WriteLine("Введите элементы массива");

 for (int ii = 0; ii < Zise\_Arr3; ii++)

 {

 Arr3[ii] = float.Parse(Console.ReadLine());

 }

 Console.WriteLine("Введенный отсортированный массив");

 Array.Sort(Arr3);

 for (int ii = 0; ii < Zise\_Arr3; ii++)

 {

 Console.WriteLine(Arr3[ii].ToString());

 }

 //Поиск максимума

 float k = Arr3.Max();

 Console.WriteLine("Max=" + k.ToString());

 //Поиск минимума

 k = Arr3.Min();

 Console.WriteLine("Min=" + k.ToString());

 //Сравнение двух массивов (Equals)

 bool l = Array.Equals(Arr1, Arr2);

 Console.WriteLine("BOOL=" + l.ToString());

 l = Array.Equals(Arr1, Arr1);

 Console.WriteLine("BOOL=" + l.ToString());

 }

 }

**ЗАДАНИЕ**

Приложить решение заданий из практического занятия №2.

Изучить материал, приведенный выше, и в соответствии с заданным вариантом написать программу для решения Задачи с использованием методов класса System.Array.

**Задача**

Создать массив с элементами $z\_{k,n}=n∙f\left(k\right)+\sin((k)∙f(n))$, , где n,k = 1 … N, N– целое число, вводимое пользователем с клавиатуры. Значение функции f(n) выбираются из таблицы 1. Вывести на экран значения массива. Выполнить задание в соответствии с вариантом

******

**Варианты индивидуальных заданий.**

**Задание 1**

1. Определить индексы минимального элемента массива. Вычислить произведение его отрицательных элементов.

2. Вычислить количество элементов массива, для которых выполняется неравенство 1 < Zk,n < 6.

3. Вычислить произведение значений тех элементов, для которых справедливо неравенство Zk,n < – 1 или Zk,n > 1.

4. Вычислить количество положительных элементов и их произведение.

5. Вычислить сумму квадратов элементов, значения которых больше 1.

6. Вычислить произведение квадратов тех элементов, для которых выполняется неравенство |Zk,n|< 3.

7. Вычислить количество элементов массива, для которых выполняется неравенство и Zk,n > 3 и сумму элементов, значение которых меньше 9.

8. Вычислить произведение отрицательных элементов. Определить индексы максимального элемента.

9. Вычислить сумму диагональных элементов массива и количество отрицательных элементов.

10. Вычислить произведение элементов массива, для которых справедливо неравенство 2 < Zk,n < 10.

11. Определить индексы максимального элемента массива. Вычислить произведение элементов выше главной диагонали.

12. Вычислить произведение элементов первых двух строк.

13. Вычислить сумму элементов массива выше главной диагонали. Определить индексы минимального элемента.

**Задание 2**

1. Вычислить сумму отрицательных элементов. Найти максимальный элемент.

2. Вычислить произведение минимального и максимального члена массива.

3. Определить индексы минимального и максимального элементов массива.

4. Значения элементов массива равные нулю заменить на 1. Определить сумму элементов ниже главной диагонали.

5. Определить количество отрицательных и сумму положительных элементов массива.

6. Вычислить произведение элементов, для которых выполняется неравенство Zk,n < – 5.или Zk,n > 3.

7. Определить индексы максимального и минимального элементов массива. Вычислить их произведение.

8. Вычислить произведение элементов выше главной диагонали матрицы и определить их количество.

9. Вычислить среднее арифметическое положительных элементов массива.

10. Вычислить сумму элементов массива, для которых выполняется неравенство 1 < Zk,n < 5.

11. Вычислить сумму диагональных элементов матрицы и количество элементов, значения которых меньше 3.

12. Вычислить произведение элементов ниже главной диагонали на сумму элементов, которые расположены ниже главной диагонали.