**Лабораторная работа №2. Массивы. Класс Random, класс Array**

***Цель работы –*** получить навыки работы с массивами: правильно описывать массивы, вводить и выводить данные массива, различными способами обрабатывать элементы массива.

***Ключевые понятия:*** ссылочные типы данных, одномерные и многомерные массивы, размерность, диапазон.

**Теоретический материал**

**Массив — упорядоченная совокупность элементов одного типа.**

Тип «массив» относится к ссылочным типам данных, применяется в тех случаях, когда приходится иметь дело с наборами однотипных и однородных данных. Элементы массива пронумерованы, и обратиться к каждому из них можно, указав имя массива и индекс элемента в квадратных скобках. Индекс может быть буквальной константой или выражением.

Массив задает способ организации данных. Количество индексов характеризует размерность массива. Каждый индекс изменяется в некотором диапазоне от нуля до N. Индексы задаются целочисленным типом.

В программировании количество индексов массива называется его *размерностью* (рангом), количество допустимых значений каждого индекса – его *диапазоном*, а совокупность размерности и диапазонов – *формой* массива.

В математических вычислениях часто приходится иметь дело с векторами. Вектор – это пример массива, в котором элементы нумеруются одним индексом. Например, a (a0, a1, a3)

Если речь идёт о хранении в массиве таблицы значений (матрицы), его элементы нумеруются двумя индексами.

В роли базового класса для всех массивов в среде CLR выступает класс Array, он предоставляет методы для создания, изменения, поиска и сортировки массивов. Например, свойство **length** — позволяет определить длину массива, а метод **Sort** – отсортировать массив.

Заполнять массив можно «вручную» или случайными числами. Для этого нужно воспользоваться классом **Random.**

Класс**Random** имеет конструктор класса: для того, чтобы **вызывать методы класса**, нужно вначале **создавать экземпляр класса**.

Этим**Random** отличается от класса**Math,** у которого все поля и методы статические, что позволяет обойтись без создания экземпляров класса**Math**.

Методы класса**Random,** необходимые для генерирования последовательностей случайных чисел:

**конструкторы класса** (две реализации)‏:

* + **public Random()** позволяет генерировать неповторяющиеся при каждом запуске серии случайных чисел.
* **public Random(int)** обеспечивает возможность генерирования повторяющейся серии случайных чисел. Параметр используется для построения начального элемента серии, поэтому при задании одного и того же значения параметра серия будет повторяться.

**public int Next()** - выдает целые положительные числа во всем положительном диапазоне типа int;

**public int Next(int max)** — выдает целые положительные числа в диапазоне [0, max-1];

**public int Next(int min, int max)** — выдает целые положительные числа в диапазоне [min,max-1].

**public double NextDouble()** — при каждом вызове метода выдается новое случайное число, равномерно распределенное в интервале [0..1).

**public voidNextBytes (byte[ ] buffer)** — позволяет при одном обращении получать целую серию случайных чисел.

Параметр - массив, который и будет заполнен случайными числами в диапазоне [0, 255].

***Объявление одномерного массива.***

Общий вид: **<тип>[] <объявление>;**

*Варианты объявления:*

**1). тип[] имя;** объявление с отложенной инициализацией (инициализация должна произойти позже)

пример, int [] m1; (описана ссылка на массив с элементами целого типа, память не выделена)

null

**m1**

**стек**

**стек**

**стек**

**а**

Рисунок 19. Переменная m1 хранит значение null

**2). тип[] имя = new тип [ размерность ];** (типы должны совпадать)

пример, double [] m2 = newdouble [5] (объявлен и размещён в динамической памяти массив размера 5 с элементами вещественного типа)

**куча**

**стек**

адрес

**m2**

0

0

0

0

0

**0**

**1**

**2**

**3**

**4**

**стек**

**а**

Рисунок 20. Переменная m2 хранит ссылку на массив в куче

**3). тип[] имя = { список\_инициализаторов };**

string[] weekDays = {"Mon", "Tue", "Wed", "Thu", "Fri", "Sat","Sun",} (объявлен и размещён в динамической памяти массив размера 7 с заданными элементами строкового типа)

**куча**

адрес

**weekDays**

"Thu"

"Fri"

"Sun"

"Sat"

"Wed"

"Tue"

"Mon"

**стек**

Рисунок 21. Переменная weekDays хранит ссылку на массив в куче

**4). тип[] имя = new тип [] { список\_инициализаторов };**

пример, int [] m4 = new int [] {2, -5, 7, 1 -20} (объявлен и размещён в динамической памяти массив размера 5 с заданными элементами)

**5). тип[] имя = new тип [ размерность ] { список\_инициализаторов };**

пример, int [] m5 = new int [5] {2, -5, 7, 1 -20} (объявлен и размещён в динамической памяти массив размера 5 с заданными элементами)

Обращение к элементам массива возможно только после инициализации, поэтому к элементам массива m1 обратиться нельзя.

m2 [1] = -100.25; m2 [15 % 4] = 10;

weekDays [0] = “Monday”;

m4 [2+2] = -23;

Console.WriteLine(m5[3]);

***Объявление двумерного массива.***

int[ , ] m\_4x2 = new int[4, 2]; (объявлен и размещён в динамической памяти массив размера 4x2, то есть четыре строки и два столбца)

int[,] m\_3x4 = {

{1,10,15,-2},

{2,20,10,11},

{3,30,34,-67,}

};

(объявлен и размещён в динамической памяти массив размера 3x4 c заранее заданными элементами)

***Варианты заполнения элементов массива:***

* при объявлении (рассмотрено ранее)
* элементы вычисляются
* ввод элементов с клавиатуры
* заполнение массива случайными числами

**Элементы массива вычисляются:**

**Пример 1.**

int[] a = new int[5];

for (int i = 0; i < 5; i++)

a[i] = (i+1)\*2;

**Пример 2.**

for (int i=0; i < a.**Length**; i++) //вместо константы использовано свойство

a[i] = (i+1)\*2;

В результате в массиве будут храниться числа 2 4 6 8 10

**Ввод элементов с клавиатуры**

Console.WriteLine("введите размерность массива");

**int n =**  **int.Parse(Console.ReadLine());**

int[] a = new int[n];

for (int i = 0; i < n; ++i)‏

{ Console.Write("a[{0}]= ", i);

**a[i] = int.Parse(Console.ReadLine());**

}

**Заполнение массива случайными числами**

Random rnd = new Random();

// случайные числа в диапазоне [0,1)

double[] d = new double[4];

for (int i = 0; i < d.Length; i++)

**d[i] = rnd.NextDouble();**

// случайные числа в диапазоне[min,max)

int min = -100, max = 100

Random rnd = new Random();

int[] a = new int[5];

for (int i = 0; i < a.Length; i++)

**a[i] = rnd.Next(min, max)**;

// случайный массив байтов

const int initRnd = 77;

Random arrRnd = new Random(initRnd);

byte[] b = new byte[10];

**arrRnd.NextBytes(b);**

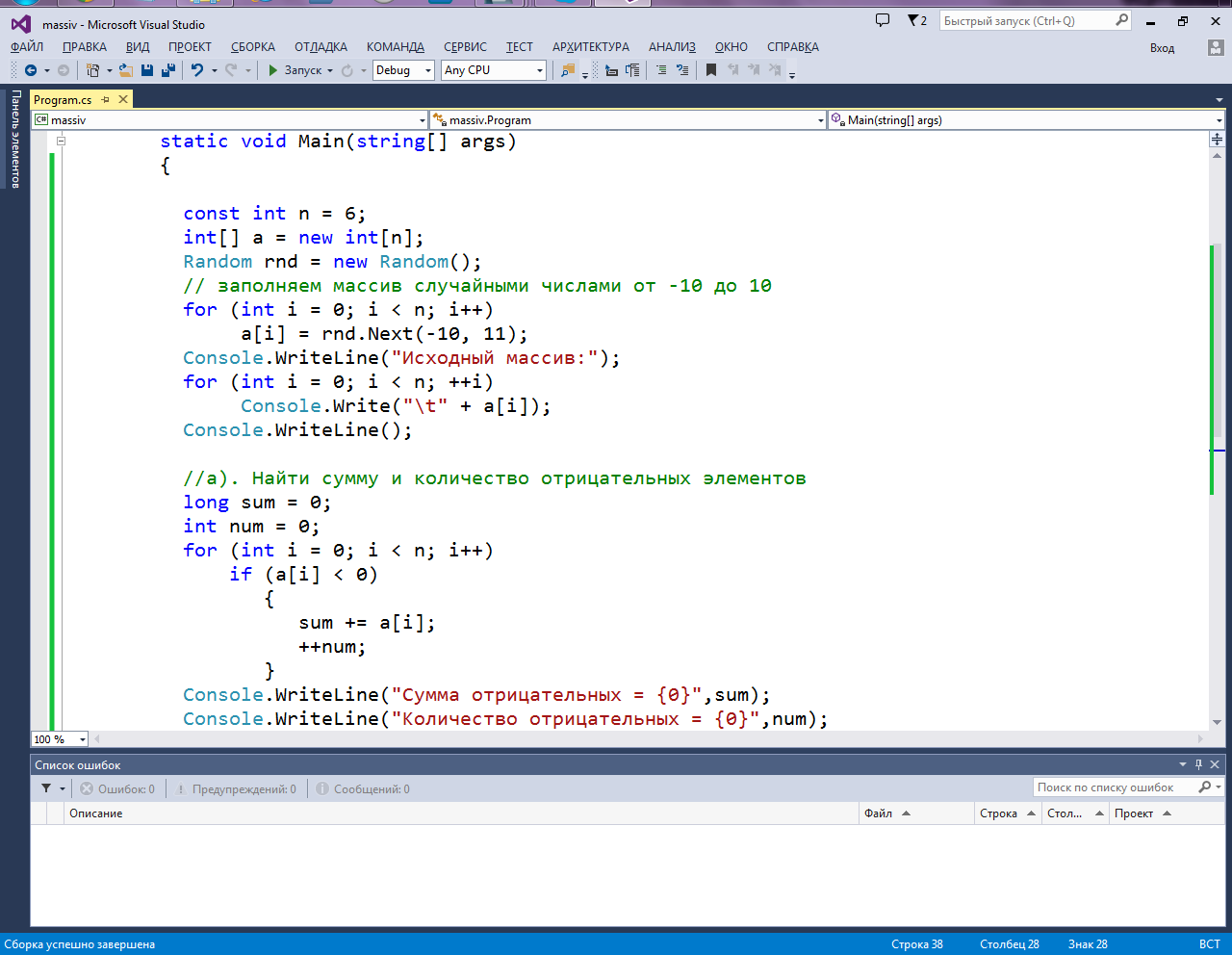
Console.WriteLine("Массив случайных чисел в диапазоне [0, 255]");

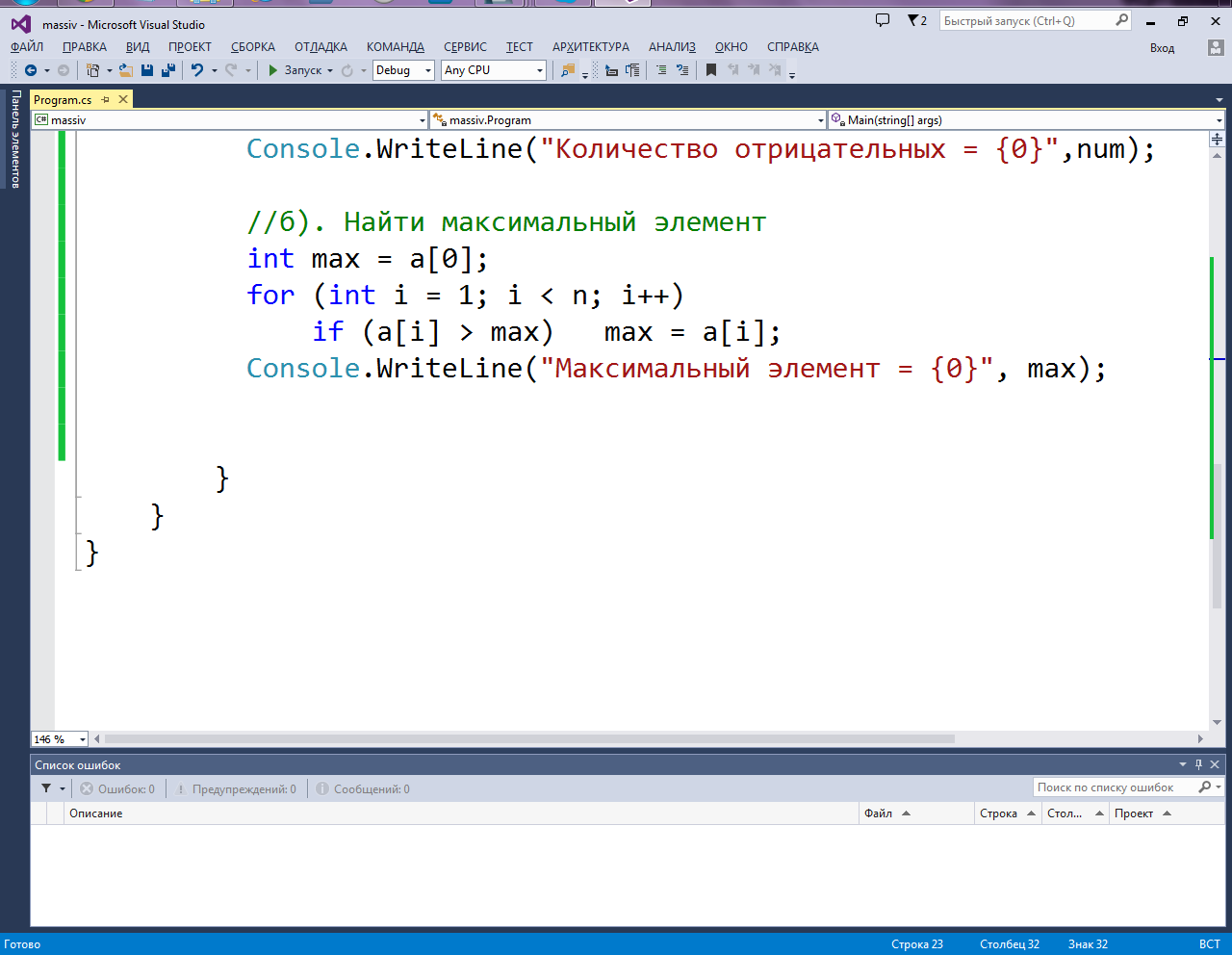
for (int i = 0; i < b.Length; i++)

Console.WriteLine("b[{0}] = {1}", i, b[i]);

***Упражнения***

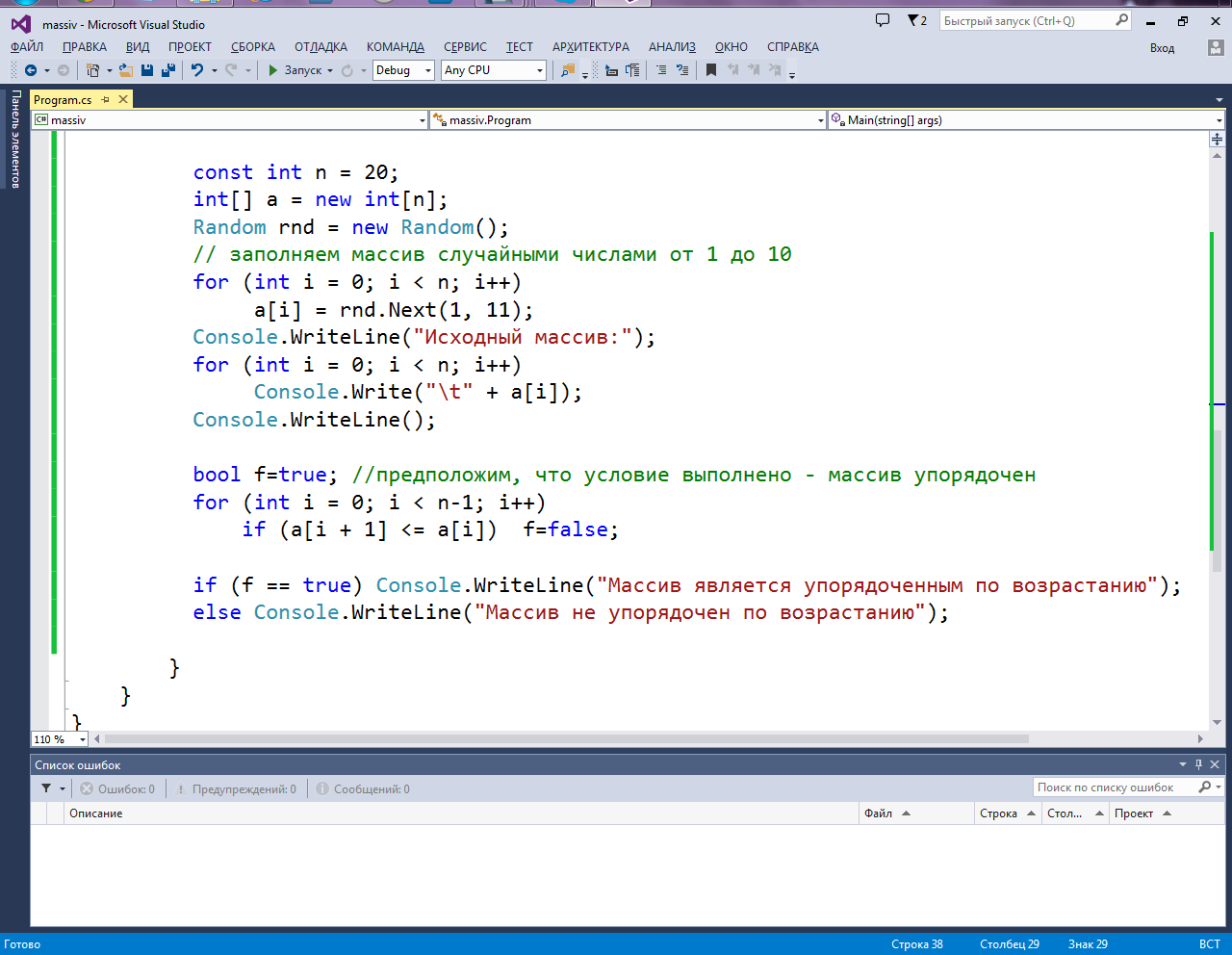
1. В заданном массиве найти а). сумму и количество отрицательных элементов; б). максимальный элемент.

Решение:



1. Дан массив из 20 чисел. Проверьте, является ли он упорядоченным по возрастанию.

Решение: В решении используется переменная-флаг типа boolean. Сначала её значение равно true, затем, если нарушится условие упорядоченности, её значение поменяется на false и больше не изменится.



1. Дан массив из 50 чисел. Определите:
   1. количество отличных от последнего элемента;
   2. число максимальных элементов;
   3. номера элементов, являющихся полными квадратами (1, 4, 9, 16, и т. д.);
   4. среднее геометрическое нечётных положительных чисел.
   5. Дан массив из 50 чисел. Используя сортировку массива, определите:
   6. значение третьего минимума;
   7. количество разных элементов массива;
   8. количество перестановок, необходимых для упорядочения массива по неубыванию (невозрастанию).
2. Дан фрагмент программы:

const int n = 365;

double [] t = new double[n];

По массиву t, где указана температура каждого дня некоторого невисокосного года, определите название месяца:

* 1. с максимальной среднемесячной температурой;
  2. с наибольшим количеством отрицательных температур;
  3. с минимальной годовой температурой;
  4. с максимальной температурой в третьей декаде.

1. Дан фрагмент программы:

const int n = 10;

string [] name = {"Ваня","Гена","Олег","Коля","Маша","Нина", "Оля", "Таня","Федя","Галя"};

bool [] pol = {true,true,true,true,false,false,false,false, true, false};

double [] ves = new double [n]; //вес в кг в диапазоне от 40 до 150

int [] rost = new int [n]; //рост в см в диапазоне от 140 до 200

* 1. определите имя самого высокого мужчины;
  2. определите средний вес женщин;
  3. определите индекс массы тела (ИМТ) для каждого;
  4. выведите на экран информацию (имя, рост, вес) о тех людях, чей ИМТ находится вне нормы.
  5. отсортируйте массив name по алфавиту (при этом соответствующие значения роста и веса тоже должны быть отсортированы)

*Примечание*. ИМТ вычисляется по формуле: . Например, при росте 180 см и весе 78 кг . Норма ИМТ от 18 до 25.

***Контрольные вопросы***

1. Дайте определение массива.
2. Где и как определяется общее число элементов массива?
3. Как осуществляется доступ к каждому элементу массива?
4. Какой массив называется одномерным, двумерным, n-мерным?
5. Существуют ли ограничения на размерность массива?