# ПРАКТИЧЕСКАЯ РАБОТА № 10.

# ОБРАБОТКА ОДНОМЕРНЫХ МАССИВОВ НА ЯЗЫКЕ C#.

**Цель работы**: овладение практическими навыками работы с одномерными массивами, особенностями их ввода и вывода и обработке данных в них.

|  |  |  |
| --- | --- | --- |
| №  п/п | Ф.И.О. | НОМЕР ВАРИАНТА |
|  | Балицкая Анастасия |  |
|  | Вартик Максим |  |
|  | Вдовиченко Ярослав |  |
|  | Веряскин Иван |  |
|  | Гоцалюк Арзу |  |
|  | Градинар Илья |  |
|  | Гребенщиков Данил |  |
|  | Завтонев Роман |  |
|  | Захаров Станислав |  |
|  | Иванов Богдан |  |
|  | Ильинский Никита |  |
|  | Каплин Егор |  |
|  | Коваленко Михаил |  |
|  | Колак Михаил |  |
|  | Левицкая Мария |  |
|  | Милюкова Екатерина |  |
|  | Никифорова Елизавета |  |
|  | Подолян Евгений |  |
|  | Попов Максим |  |
|  | Родичев Максим |  |
|  | Рознован Роман |  |
|  | Тизул Игорь |  |
|  | Тихончук София |  |
|  | Толоченко Павел |  |
|  | Фокша Кирилл |  |
|  | Чебан Евгений | 5 |
|  | Чебан Илья | 7 |
|  | Юраш Ренат | 10 |

**Теоретическая часть**

*Массив* - набор элементов одного и того же типа, объединенных общим именем. Массивы в С# можно использовать по аналогии с тем, как они используются в других языках программирования. Однако С#-массивы имеют существенные отличия: они относятся к ссылочным типам данных, более того - реализованы как объекты. Фактически имя массива является ссылкой на область динамической памяти, в которой последовательно размещается набор элементов определенного типа. Выделение памяти под элементы происходит на этапе инициализации массива. А за освобождением памяти следит система сборки мусора - неиспользуемые массивы автоматически утилизируются данной системой.

*Одномерные массивы*

*Одномерный массив* - это фиксированное количество элементов одного и того же типа, объединенных общим именем, где каждый элемент имеет свой номер. Нумерация элементов массива в С# начинается с нуля, то есть, если массив состоит из 10 элементов, то его элементы будут иметь следующие номера: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9.

Одномерный массив в С# реализуется как объект, поэтому его создание представляет собой двухступенчатый процесс. Сначала объявляется ссылочная переменная на массив, затем выделяется память под требуемое количество элементов базового типа, и ссылочной переменной присваивается адрес нулевого элемента в массиве. Базовый тип определяет тип данных каждого элемента массива. Количество элементов, которые будут храниться в массиве, определяется размер массива.

В общем случае процесс объявления переменной типа массив, и выделение необходимого объема памяти может быть разделено. Кроме того на этапе объявления массива можно произвести его инициализацию. Поэтому для объявления одномерного массива может использоваться одна из следующих форм записи:

|  |  |
| --- | --- |
| *Форма записи* | *Пояснения* |
| базовый\_тип [ ] имя\_\_массива;  *Например*:  int [ ] a; | Описана ссылка на одномерный массив, которая в дальнейшем может быть использована:  - для адресации на существующий массив;  - передачи массива в метод в качестве параметра  - отсроченного выделения памяти под элементы массива |
| базовый\_тип [ ] имя\_\_массива = new базовый\_тип [размер];  *Например*:  int [ ]a=new int [10]; | Объявлен одномерный массив заданного типа и выделена память под одномерный массив указанной размерности. Адрес данной области памяти записан в ссылочную переменную. Элементы массива равны нулю. **Замечание**. В C# элементам массива присваиваются начальные значения по умолчанию в зависимости от базового типа: для арифметических типов - нули, для ссылочных типов - null, для символов - пробел. |
| базовый\_тип [ ] имя\_\_массива={список инициализации};  *Например*:  int [ ]a={0, 1, 2, 3}; | Выделена память под одномерный массив, размерность которого соответствует количеству элементов в списке инициализации. Адрес этой области памяти записан в ссылочную переменную. Значение элементов массива соответствует списку инициализации. |

Обращения к элементам массива происходи с помощью индекса, для этого нужно указать имя массива и в квадратных скобках его номер. Например, a[0], b[10], c[i].

Так как массив представляет собой набор элементов, объединенных общим именем, то обработка массива обычно производится в цикле.

*Пример*:

static void Main()

{

int[ ] myArray = { 0, 1, 2, 3, 4, 5, 6, 7, 8, 9 };

int i;

for (i = 0; i < 10; ++i)

Console.WriteLine(myArray[i]);

}

*Массив как параметр*

Так как имя массива фактически является ссылкой, то он передается в метод по ссылке и, следовательно, все изменения элементов массива, являющегося формальным параметром, отразятся на элементах соответствующего массива, являющимся фактическим параметром.

*Пример*: передача массива как параметра:

class Program

{

static void Print(int n, int[ ] a) //n - размерность массива, а - ссылка на массив

{

for (int i = 0; i < n; i++) Console.Write("{0} ", a[i]);

Console.WriteLine();

}

static void Change(int n, int[ ] a)

{

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

if (a[i] > 0) a[i] = 0; // изменяются элементы массива

}

static void Main()

{

int[ ] myArray = { 0, -1, -2, 3, 4, 5, -6, -7, 8, -9 };

Print(10, myArray);

Change(10, myArray);

Print(10, myArray);

}

}

*Массив как объект*

Так как массивы в С# реализованы как объекты и они реализованы на основе базового класса Array, определенного в пространстве имен System. Данный класс содержит различные свойства и методы. Например, свойство Length позволяет определять количество элементов в массиве. Преобразуем предыдущий пример:

class Program

{

static void Print(int[ ] a) // передаем только ссылку на массив

{

for (int i = 0; i < a.Length; i++) Console.Write("{0} ", a[i]);

Console.WriteLine();

}

static void Change(int[ ] a)

{

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

if (a[i] > 0) a[i] = 0;

}

static void Main()

{

int[] myArray = { 0, -1, -2, 3, 4, 5, -6, -7, 8, -9 };

Print(myArray);

Change(myArray);

Print(myArray);

}

}

Некоторые свойства и методы класса Array:

|  |  |  |
| --- | --- | --- |
| **Элемент** | **Вид** | **Описание** |
| Length | Свойство | Количество элементов массива (по всем размерностям) |
| BinarySearch | Статический метод | Двоичный поиск в отсортированном массиве |
| Clear | Статический метод | Присваивание элементам массива значений по умолчанию |
| Copy | Статический метод | Копирование заданного диапазона элементов одного массива в другой |
| CopyTo | экземплярный метод | Копирование всех элементов текущего одномерного массива в другой массив |
| GetValue | экземплярный метод | Получение значения элемента массива |
| IndexOf | статический метод | Поиск первого вхождения элемента в одномерный массив |
| LastIndexOf | статический метод | Поиск последнего вхождения элемента в одномерный массив |
| Reverse | статический метод | Изменение порядка следования элементов на обратный |
| SetValue | экземплярный метод | Установка значения элемента массива |
| Sort | статический метод | Упорядочивание элементов одномерного массива |

Вызов статических методов происходит через обращение к имени класса, например, Array.Sort(myArray). В данном случае происходит обращение к статическому методу Sort класса Array и происходит передача данному методу в качестве параметра объект myArray - экземпляр класса Array.

Обращение к свойству или вызов экземплярного метода производится через обращение к экземпляру класса, например, myArray.Length или myArray.GetValue(i).

*Пример*:

class Program

{

static void Main()

{

try

{

int[ ] MyArray;

Console.Write("Введите размерность массива: ");

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

MyArray = new int[n];

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

{

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

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

}

PrintArray("исходный массив:", MyArray);

Array.Sort(MyArray);

PrintArray("массив отсортирован по возрастанию", MyArray);

Array.Reverse(MyArray);

PrintArray("массив отсортирован по убыванию", MyArray);

}

catch (FormatException)

{

Console.WriteLine("неверный формат ввода данных");

}

catch (OverflowException)

{

Console.WriteLine("переполнение");

}

catch (OutOfMemoryException)

{

Console.WriteLine("недостаточно памяти для создания нового объекта");

}

}

static void PrintArray(string a, int[ ] mas)

{

Console.WriteLine(a);

for (int i = 0; i < mas.Length; i++) Console.Write("{0} ", mas[i]);

Console.WriteLine();

}

}

}

**Вопросы для самоконтроля**

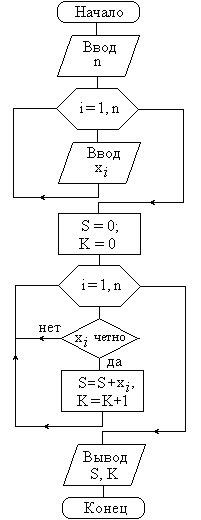
1. Дайте определение массиву.
2. Почему максимальный номер элемента массива на единицу меньше размерности массива?
3. Как задается размерность массива?
4. Что такое «индекс массива»?
5. Как происходит обращение к элементам массива?
6. Ограниченно ли количество измерений массивов в языке С#?
7. Какие данные могут выступать в качестве индексов и элементов массива?
8. В чем состоит особенность организации цикла при обработке массива?
9. Какие способы задания исходных значений элементов массива вам известны?
10. Как осуществляется доступ к каждому элементу массива?
11. Может ли индекс быть выражением вещественного типа?

**Практическая часть**

**Задание №1.** Работа с одномерными массивами

*Пример*: Найти сумму и количество четных чисел массива *x*1, *x*2, ... , *xn*, где *n* ≤ 30. Проверить правильность работы программы на тесте при *n*=3 и элементах массива 2, 5, 4.

Блок-схема алгоритма:



**Варианты задания №1**

1. В массиве из 15 целых чисел найти наибольший элемент и поменять его местами с первым элементом.
2. Вычислить сумму и количество элементов массива вещественных чисел.
3. Вывести на экран дисплея порядковые номера нечетных элементов массива целых чисел.
4. Вычислить среднее арифметическое массива вещественных чисел.
5. Найти минимальный элемент массива и его порядковый номер.
6. В массиве вещественных чисел найти количество отрицательных элементов.
7. Найти сумму чисел кратных трем в массиве целых чисел.
8. В массиве вещественных чисел найти произведение элементов, стоящих на четных позициях.
9. Найти количество отрицательных элементов массива А.
10. Найти порядковый номер максимального элемента массива вещественных чисел.
11. Вывести на экран дисплея положительные элементы массива и найти их количество.
12. Вывести на экран дисплея номера элементов массива, удовлетворяющих условию 0<*xi*<3,2.
13. Заменить четные элементы в массиве целых чисел нулями и вывести новый массив на экран.
14. Найти среднее арифметическое элементов массива целых чисел кратных пяти.
15. В массиве вещественных чисел найти минимальный элемент среди отрицательный элементов.
16. Найти сумму двузначных элементов в массиве целых чисел.
17. Найти количество и сумму положительных элементов в массиве вещественных чисел, удовлетворяющих условию 0<*xi*<5,7.
18. Заменить отрицательные элементы массива находящиеся на нечетных позициях единицей и вывести новый массив на экран дисплея.
19. Вывести на экран дисплея первый отрицательный элемент и его порядковый номер.
20. Найти сумму максимального и минимального элементов массива.
21. Найти произведение элементов находящихся на четных позициях в массиве.
22. Найти количество и порядковые номера элементов массива вещественных чисел, удовлетворяющих условию 2,1 ≤ *xi* ≤ 7.
23. Поменять местами в массиве вещественных чисел максимальный и минимальный элементы и вывести массив на экран дисплея.
24. Найти среднее геометрическое элементов массива вещественных чисел.
25. Найти порядковый номер максимального элемента среди четных значений массива целых чисел.

**Задание №2.** В одномерном массиве, состоящем из *n* вещественных элементов, вычислить:

1. сумму отрицательных элементов массива;
2. произведение элементов массива, расположенных между максимальным и минимальным элементами;
3. сумму положительных элементов массива;
4. произведение элементов массива, расположенных между максимальным по модулю и минимальным по модулю элементами;
5. произведение элементов массива с четными номерами;
6. сумму элементов массива, расположенных между первым и последним нулевыми элементами;
7. сумму элементов массива с нечетными номерами;
8. сумму элементов массива, расположенных между первым и последним отрицательными элементами;
9. максимальный элемент массива;
10. сумму элементов массива, расположенных до последнего положительного элемента;
11. минимальный элемент массива;
12. сумму элементов массива, расположенных между первым и последним положительными элементами;
13. произведение элементов массива, расположенных между первым и вторым нулевыми элементами;
14. номер минимального элемента массива;
15. сумму элементов массива, расположенных между первым и вторым отрицательными элементами;
16. максимальный по модулю элемент массива;
17. сумму элементов массива, расположенных между первым и вторым положительными элементами;
18. минимальный по модулю элемент массива;
19. сумму модулей элементов массива, расположенных после первого элемента, равного нулю;
20. номер минимального по модулю элемента массива;
21. сумму модулей элементов массива, расположенных после первого отрицательного элемента;
22. номер максимального по модулю элемента массива;
23. сумму элементов массива, расположенных после первого положительного элемента;
24. количество элементов массива, лежащих в отрезке от А до В;
25. сумму элементов массива, расположенных после максимального элемента.

**Задание №3.** В массиве содержатся результаты измерений температуры воздуха, которые проводились ежедневно в течение декабря месяца. Определите:

* 1. среднемесячную температуру декабря;
  2. сколько раз температура была выше 0°С;
  3. минимальную температуру первой декады декабря;
  4. сколько раз температура была ниже 0°С;
  5. день, когда температура была наибольшей;
  6. среднемесячную температуру первой и второй декады декабря;
  7. день, когда температура была наименьшей;
  8. день, когда первый раз температура поднялась выше нуля, и увеличьте эту температуру на 1°С;
  9. любой из самых холодных дней декабря;
  10. сколько дней в декабре температура была выше средней;
  11. день, когда температура была ближе всего к средней температуре в декабре;
  12. сколько дней в декабре температура была ниже средней;
  13. минимальную температуру второй декады декабря;
  14. минимальную температуру тех дней декабря, которые следуют после последнего из самых теплых дней в этом месяце;
  15. максимальную температуру первой декады декабря;
  16. температуры любого из самых холодных и самых теплых дней и поменяйте их местами;
  17. среднюю температуру тех дней, которые предшествуют первому из самых холодных дней в декабре;
  18. любые два самых холодных дня;
  19. температуры всех дней выше среднемесячной заменить на +10 С;
  20. сколько раз в декабре температура меняла знак;
  21. первую из минимальных температур перенесите на последнее место, сдвинув все остальные к началу массива;
  22. любые два самых теплых дня;
  23. температуры всех дней ниже среднемесячной заменить 0;
  24. максимальную температуру второй декады декабря;
  25. первую из максимальных температур перенесите на первое место, сдвинув все остальные к концу массива.