



Программирование на C#

Семинар №12

Модуль №1

Тема:

Работа с массивами массивов
(зубчатыми массивами).



Задания преподавателя к семинару

Выполняем задания категорий ToDo и Self.

Уровень сложности заданий для реализации и последовательность проектов выбирайте сами.



Полезные материалы к семинару

1. Массивы. Многомерные массивы. Массивы массивов.

<https://learn.microsoft.com/ru-ru/dotnet/csharp/language-reference/builtin-types/arrays#multidimensional-arrays>

2. Класс Array. Методы <https://learn.microsoft.com/ru-ru/dotnet/api/system.array?view=net-7.0>

3. Класс Array [<https://docs.microsoft.com/ru-ru/dotnet/api/system.array?view=net-5.0>

4. Массивы массивов <https://learn.microsoft.com/ru-ru/dotnet/csharp/language-reference/builtin-types/arrays#jagged-arrays>

Демо 01. Переброска строк массива



//Условие соответствует задаче предыдущего семинара: переброска строк
//попарно (первая с последней, вторая с предпоследней и т.д.)
//Отличие: матрица реализована как массив массивов.

```
int  n = 4,      //Количество строк.  
     m = 5,      //Количество столбцов.  
     mp,         //Количество пар строк.  
     ном1,       //Номер первой строки в паре.  
     ном2,       //Номер второй строки в паре.  
     i, j;
```

Формируем ссылку на
массив массивов

```
double[][] a;    //Исходная таблица.  
double[] b;      //Буферная ссылка на строку таблицы.
```

```
a = new double[n][];  
for (i = 0; i < n; i++)  
    {a[i] = new double[m];} //Выделение памяти под строки.
```

Под каждую ссылку на массив память
выделяется отдельно

```
for (i = 0; i < a.Length; i++) //Ввод ступенчатого массива.
```

```
    for (j = 0; j < a[i].Length; j++)  
    {
```

```
        Console.WriteLine("Элемент[{0}][{1}]: ", i, j);  
        a[i][j] = double.Parse(Console.ReadLine());  
    }
```

```
Console.WriteLine("\nИсходная таблица"); //Вывод массива
```

```
for (i = 0; i < a.Length; i++, Console.WriteLine())  
    for (j = 0; j < a[i].Length; j++)  
        { Console.WriteLine("{0,8:f2}", a[i][j]); }
```

```
mp = n / 2; //Количество перебрасываемых пар строк.
```

```
for (ном1 = 0, ном2 = n - 1; ном1 < mp; ном1++, ном2--)  
{
```

```
    b = a[ном1];  
    a[ном1] = a[ном2];  
    a[ном2] = b;  
}
```

Подумаем, каково
количество
перебрасываемых
пар при нечетном
количестве
ссылок?

Переброска строк (ссылок
на них). В качестве буфера
используется одномерный
массив

```
Console.WriteLine("\nТаблица после перестановки строк");
```

```
for (i = 0; i < a.Length; i++, Console.WriteLine())  
    for (j = 0; j < a[i].Length; j++)  
        Console.WriteLine("{0,8:f2}", a[i][j]);
```



Юникод в действии,
никогда не используйте
символы национальных
алфавитов в
идентификаторах
программ на C#



ToDo 01 к Demo 01.

Сформируйте методы:

- Заполнения массива массивов
- Вывода массива массивов
- Переброски строк

ToDo 02 к Demo 01.

Сформируйте метод упорядочивания нечетных строк массива массивов в обратном порядке без использования методов Sort Revers

Демо 02. Определить массив из трех элементов – ссылок на массивы разной длины.



1-й элемент - ссылка на массив из 3-х элементов-ссылок (на массивы из 2-х, 3-х и 4-х элементов типа char.)

2-й элемент – ссылка на массив из 2-х элементов- ссылок (на массивы, соответственно, из 2-х и 3-х элементов типа char).

3-й элемент - массив из ОДНОГО элемента – ссылки (на массив из 4-х элементов типа char).

Используя свойства и методы класса Array вывести:

- ранг массива,
- общее число его элементов,
- число элементов по разным измерениям,
- предельные значения всех индексов.

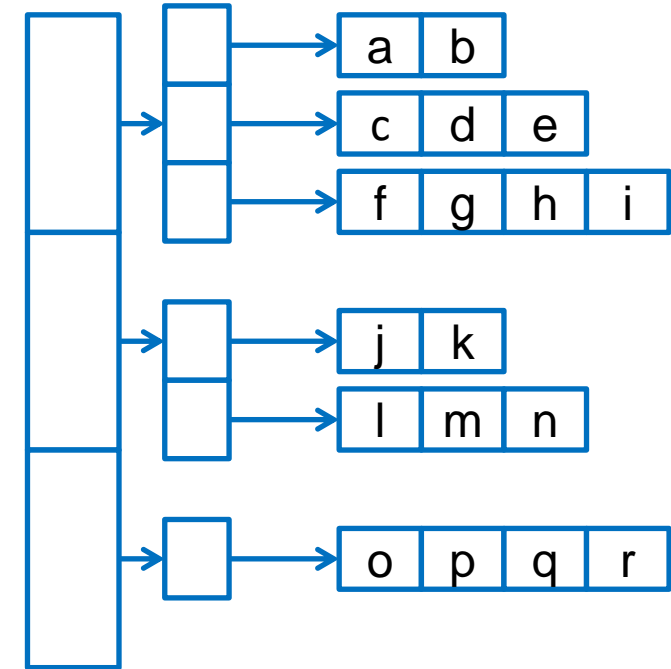
ToDo 03. Вывести элементы массива с помощью циклов foreach, размещая значения элементов каждого массива нижнего уровня по строкам.

Массивы массивов <https://learn.microsoft.com/ru-ru/dotnet/csharp/language-reference/builtin-types/arrays#jagged-arrays>

Demo 02. Пример Массива Массивов



```
char[ ][ ][ ] ch = { // Элементы массива – массивы ссылок на массивы.  
new char[ ][ ] { new char[ ] {'a', 'b'},  
new char[ ] {'c', 'd', 'e'},  
new char[ ] {'f', 'g', 'h', 'i'}  
},  
new char[ ][ ] { new char[ ] {'j', 'k'},  
new char[ ] {'l', 'm', 'n'}  
},  
new char[ ][ ] { new char[ ] {'o', 'p', 'q', 'r'}, }  
};
```



Demo 02. Пример Массива МассивОВ Продолжение



```
Console.WriteLine("ch.Rank = " + ch.Rank);  
Console.WriteLine("ch[0].Rank = " + ch[0].Rank);  
Console.WriteLine("ch[0][0].Rank = " + ch[0][0].Rank);  
Console.WriteLine("ch.GetType() = " + ch.GetType());  
Console.WriteLine("ch[1][1][2] = " + ch[1][1][2]);  
Console.WriteLine("ch.Length = " + ch.Length);  
Console.WriteLine("ch.GetLength(0) = " + ch.GetLength(0));  
Console.WriteLine("ch[1].GetLength(0) = " +  
ch[1].GetLength(0));
```

Self. Задания для самостоятельной работы



Self 01. Создать «зубчатый» массив вида:

```
1 1 1 1 1 1 1
 2 2 2 2 2 2
  3 3 3 3 3
   4 4 4 4
    5 5 5
     6 6
      7
```

Self 02. В массиве массивов найти только те одномерные массивы, все модули элементов которых различны (значения элементов варьируются от -9 до 9). Размерность массива массивов задаёт пользователь. На экран вывести начальный массив массивов и итоговый.

Self 03. Задания для самостоятельной работы



Сформировать таблицу целых чисел размером N строк. Строки имеют разное количество элементов. Правило формирования длины строки и значений элементов строки показано на рисунке. Полученную таблицу построчно выдать на экран. Значение N вводится с клавиатуры и должно быть нечетным.

1				
1	2			
...		
1	2	3	...	$N/2 + 1$
...
1	2			
1				

Self 04 Треугольник Паскаля



Определить массив массивов для представления треугольника Паскаля.

0-й элемент - массив из одного элемента со значением $C(0,0)=1$,

1-й элемент - массив из 2-х элементов $C(1,0)=C(1,1)=1$.

2-й элемент - массив из 3-х элементов $C(2,0)=C(2,2)=1$, $C(2,1)=2...$

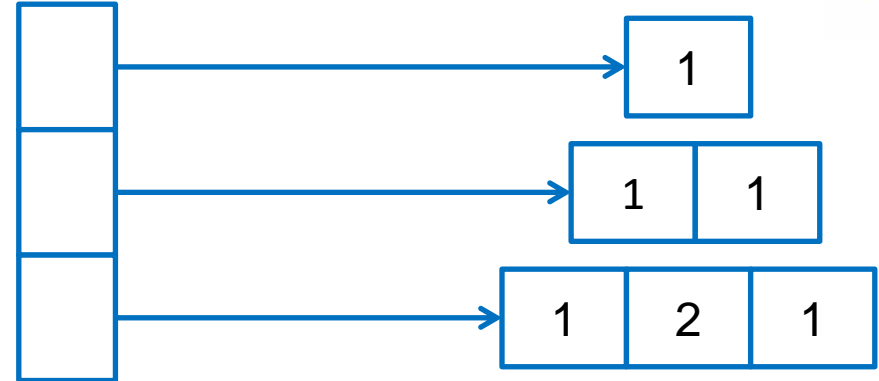
n-й элемент - массив из n+1 элементов: $C(n,0)=C(n,n)=1$,

$C(n,k)=C(n-1,k-1)+C(n-1,k)$.

Вводя неотрицательные значение n, построить массив-массивов со значениями биномиальных коэффициентов и вывести его на экран с помощью циклов `foreach`, размещая значения элементов каждого массива нижнего уровня на отдельной строке...



Разработать код, формирующий массив с элементами треугольника Паскаля. Оформить его в отдельный метод. Обеспечить вывод треугольника Паскаля в виде, соответствующем рисунку (вершина “1” должна располагаться по центру экрана или середины “нижней” строки).



...

