**15. Обобщение**

Задание №1. Используя Visual Studio, создайте проект по шаблону Console Application.  Создайте класс MyList<T>. Реализуйте возможность использования его экземпляра аналогично экземпляру класса List<T>. Минимально требуемый интерфейс взаимодействия с экземпляром, должен включать метод добавления элемента, индексатор для получения значения элемента по указанному индексу и свойство только для чтения для получения общего количества элементов.

Листинг программы:

MyList<int> myList = new MyList<int>();

myList.Add(1);

myList.Add(2);

myList.Add(3);

myList.Add(4);

Console.WriteLine("\nEnter index elementa, that you interested");

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

Console.WriteLine();

Console.WriteLine($"Element at index {index} - {myList[index]} " );

Console.WriteLine($"Count element {myList.Count}");

Console.WriteLine();

Console.WriteLine("Array:");

for (int i = 0; i < myList.Count; i++)

{Console.Write(myList[i] + " ");}

protected T[] arr =Array.Empty<T>();

public void Add(T value)

{ var newArray = new T[arr.Length+1];

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

{ newArray[i] = arr[i]; }

newArray[arr.Length] = value;

arr = newArray; }

public T this[int index]

{ get => arr[index];

set => arr[index] = value;}

public int Count

{ get { return arr.Length; }}

Таблица 15.1 – Входные и выходные данных задание №1

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
|  | Enter index elementa, that you interested  1  Element at index 1 - 2  Count element 4  Array:  1 2 3 4 |

Анализ результатов:

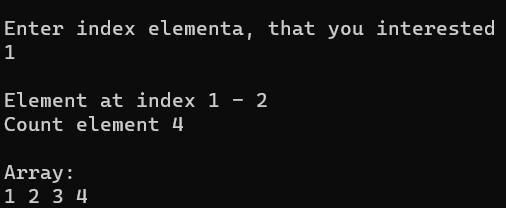


Рисунок 15.1 – Результат работы программы из задания №1

Задание №2. Используя Visual Studio, создайте проект по шаблону Console Application.  Создайте класс MyDictionary <TKey, TValue>. Реализуйте возможность использования его экземпляра аналогично экземпляру класса Dictionary. Минимально требуемый интерфейс взаимодействия с экземпляром, должен включать метод добавления пар элементов, индексатор для получения значения элемента по указанному индексу и свойство только для чтения для получения общего количества пар элементов.

Листинг программы:

protected K[] keyArr ;

protected V[] valueArr ;

public void Add(K key , V val)

{ if (keyArr != null)

{ if (Array.IndexOf(keyArr, key) != -1)

{ return; }

Array.Resize(ref keyArr, keyArr.Length + 1);

Array.Resize(ref valueArr, valueArr.Length + 1);}

else

{ keyArr = new K[1];

valueArr = new V[1];}

keyArr[keyArr.Length - 1] = key;

valueArr[valueArr.Length - 1] = val;}

public V this[int index]

{ get

{ if (valueArr != null)

{if (0 <= index && index < keyArr.Length)

{return valueArr[index]; }}

return default(V); }}

public int Count

{ get

{if (keyArr != null)

{return keyArr.Length;}

return 0;}}}

MyDictionary<int, string> myDictionary = new MyDictionary<int, string>();

myDictionary.Add(0, "Obj 1");

myDictionary.Add(1, "Obj 2");

myDictionary.Add(2, "Obj 3");

myDictionary.Add(1, "Obj 4"); //Не добавится, так как есть уже такой ключ

for (int i = 0; i < myDictionary.Count; i++)

{ Console.WriteLine(myDictionary[i]);

Console.WriteLine($"Size myDictionary {myDictionary.Count}");

Таблица 15.2 – Входные и выходные данных задание №2

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
|  | Obj 1 Obj 2 Obj 3  Enter index elementa, that you interested  2  Element at index 2 - Obj 3  Size myDictionary 3 |

Анализ результатов:

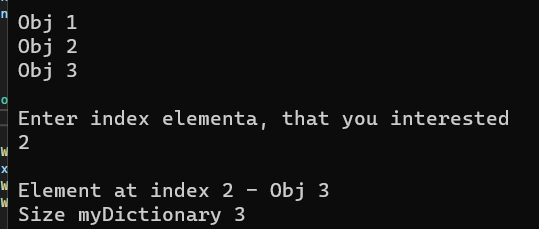


Рисунок 15.2 – Результат работы программы из задания №2

Задание №3. Используя Visual Studio, создайте проект по шаблону Console Application.  Создайте расширяющий метод: public static T[] GetArray<T>(this MyList<T> list) Примените расширяющий метод к экземпляру типа MyList<T>, разработанному в задании 2 для  данного урока. Выведите на экран значения элементов массива, который вернул расширяющий метод GetArray().

Листинг программы:

internal static T[] GetArray(MyList<T> myList)

{ if (myList.arr != null)

{ return myList.arr; }

return default(T[]); }

Console.WriteLine();

Console.WriteLine("Challenge a static method: ");

WriteArray(MyList<int>.GetArray(myList));

void WriteArray<T>(T[] arr)

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

{ Console.Write(arr[i] + " ");}}

Таблица 15.3 – Входные и выходные данных задание №2

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
|  | Статический метод: 1234 |

Анализ результатов:



Рисунок 15.3 – Результат работы программы из задания №3