**15 ОБОБЩЕННИЯ**

Задание 1. Используя Visual Studio, создайте проект по шаблону Console Application. Создайте класс MyList&lt;T&gt;. Реализуйте возможность использования его экземпляра аналогично экземпляру класса List&lt;T&gt.

Листинг программы:

using System;

using static System.Console;

namespace task1

{

class MyList<T>

{

protected int index = 0;

protected T[] Values = new T[0];

public void Add(T value)

{

Array.Resize(ref Values, index+1);

Values[index] = value;

index++;

}

public T this[int index]

{

get => Values[index];

}

public int Length

{

get => Values.Length ;

}

}

class Program

{

static void Main(string[] args)

{

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

list.Add(2121);

list.Add(21933434);

list.Add(21121);

WriteLine($"Элемент с индексом 1: {list[1]}");

WriteLine($"Длина: {list.Length}");

}

}

}

Таблица 1.1 – Входные и выходные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
|  | Элемент с индексом 1: 21933434  Длина: 3 |

Анализ результатов:



Рисунок 1.1 – Результаты работы программы

Источник: собственная разработка

Задание 2. Используя Visual Studio, создайте проект по шаблону Console Application. Создайте класс MyDictionary &lt;TKey, TValue&gt;. Реализуйте возможность использования его экземпляра аналогично экземпляру класса Dictionary.

Листинг программы:

using System;

using static System.Console;

namespace task2

{

class MyDictionary<TKey,TValue>

{

protected int index = 0;

protected TKey[] Keys = new TKey[0];

protected TValue[] Values = new TValue[0];

public void Add(TKey key, TValue value)

{

if (Array.IndexOf(Keys, key) == -1)

{

Array.Resize(ref Keys, index + 1);

Keys[index] = key;

Array.Resize(ref Values, index + 1);

Values[index] = value;

index++;

}

else

{

WriteLine("Значение с таким ключем уже существует");

}

}

public TValue this[TKey index]

{

get => Values[Array.IndexOf(Keys,index)];

}

public int Length

{

get => Values.Length;

}

}

class Program

{

static void Main(string[] args)

{

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

list.Add("key1", 32994);

list.Add("key12?", 200002);

list.Add("key13!", 101);

list.Add("key14", 5345353);

WriteLine("Элемент с ключем 'key14': " + list["key14"]);

WriteLine("Количество пар: " + list.Length);

}

}

}

Таблица 2.1 – Входные и выходные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
|  | Элемент с ключем 'key14': 5345353  Количество пар: 4 |

Анализ результатов:



Рисунок 2.1 – Результаты работы программы

Источник: собственная разработка