

## Лабораторная работа № 16

**Тема:** Разработка программ с использованием обобщений и коллекций

**Цель работы:** Формирование умений и навыков использования обобщений и коллекций

```
using System;
using System.Collections;
using System.Collections.Generic;

namespace Collections
{
    class Program
    {
        static void Main(string[] args)
        {
            // необобщенная коллекция ArrayList
            ArrayList objectList = new ArrayList() { 1, 2,
"string", 'c', 2.0f };

            object obj = 45.8;

            objectList.Add(obj);
            objectList.Add("string2");
            objectList.RemoveAt(0); // удаление первого элемента
            foreach (object o in objectList)
            {
                Console.WriteLine(o);
            }
            Console.WriteLine("Общее число элементов коллекции:
" + objectList.Count);
            // обобщенная коллекция List
            List<string> countries = new List<string>() {
"Россия", "США", "Великобритания", "Китай" };
            countries.Add("Франция");
            countries.RemoveAt(1); // удаление второго элемента
            foreach (string s in countries)
            {
                Console.WriteLine(s);
            }

            Console.ReadLine();
        }
    }
}
```

Здесь используются две коллекции: необобщенная - ArrayList и обобщенная - List. Большинство коллекций поддерживают добавление элементов. Например, в данном случае добавление производится методом Add, но для других коллекций название метода может отличаться. Также большинство коллекций реализуют удаление (в данном примере производится с помощью метода RemoveAt).

С помощью свойства Count у коллекций можно посмотреть количество элементов.

И так как коллекции реализуют интерфейс IEnumerable/IEnumerable<T>, то все они поддерживают перебор в цикле foreach.

Конкретные методы и способы использования могут различаться от одного класса коллекции к другому, но общие принципы будут одни и те же для всех классов коллекций.

## Практическая часть

### Задание 1.

Вариант	Задание
1	Описать функцию, которая заменяет в списке все вхождения элемента E1, значение которого введено с клавиатуры, на элемент E2, значение которого также введено с клавиатуры.
2	Описать функцию, которая подсчитывает число вхождений элемента E, значение которого введено с клавиатуры, в списке Q.
3	Описать функцию, которая формирует список M1 - копию списка M и список M2, представляющий собой «перевернутый» список M.
4	Описать функцию, которая вставляет в список K новый элемент L1 за каждым вхождением элемента L. Значения элементов L и L1 ввести с клавиатуры.
5	Описать функцию, которая формирует список Common, включив в него элементы, которые входят одновременно в список M1 и M2.
6	Описать функцию, которая включает в упорядоченный по убыванию список новое значение, введенное с клавиатуры, таким образом, чтобы не нарушать упорядоченность. <b>(не используя сортировок!)</b>
7	Описать функцию, которая объединяет два упорядоченных по невозрастанию списка в один упорядоченный по невозрастанию список. <b>(не используя сортировок!)</b>
8	Описать функцию, которая формирует список L, включая в него элементы, которые входят в пересечения списков L1, L2.

9	Описать функцию, которая вставляет в список Long за первым вхождением элемента I, значение которого введено с клавиатуры, все элементы списка Short, если I входит в Long.
10	Описать функцию, которая удаляет из списка за каждым вхождением элемента E, значение которого введено с клавиатуры, один элемент, если такой есть и он отличен от E.

## Задание 2. Общее для всех

Реализовать, используя **обобщенный** и **необобщенный** список

При помощи коллекции Stack. Описать функции, которые будут заменять инфиксную на префиксную и постфиксную запись и наоборот, соответственно.

Инфиксная запись	Префиксная запись	Постфиксная запись
$A + B$	$+ A B$	$A B +$
$A + B * C$	$+ A * B C$	$A B C * +$