Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение высшего образования

«Рязанский государственный радиотехнический университет

имени В.Ф.Уткина»

Кафедра «Электронные вычислительные машины»

Отчет по лабораторной работе №6

на тему

«Делегаты»

по дисциплине  
**«Визуальное программирование»**

Выполнили:

Студенты группы №140

Бригада 5

Сафаров Д. А.

Тимохин Е. С.

Проверили:  
ст. преп. Хизриева Н.И.

ст. преп. Бастрычкин А.С.

**Цель работы**

Изучить работу с делегатами в языке C#.

**Задание**

1) Расширить класс DynamicArray методом фильтрации Filter элементов коллекции, который принимает входным параметром делегат типа   
Func<T, bool>, где T - тип элементов коллекции. Выходное значение вызываемого делегата определяет, остается ли элемент в коллекции (true), или должен быть отфильтрован (false). Вызывая этот делегат для каждого элемента, в коллекции должны остаться только элементы, подходящие заданному в делегате условию.

2) Расширить класс DynamicArray методом Sort сортировки элементов коллекции, который принимает входным параметром делегат-”компаратор” типа Func<T, T, int>, где T - тип элементов коллекции.

**Практическая часть**

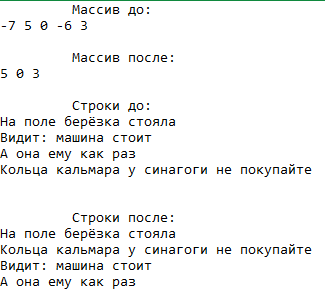
Код программы представлен в приложении А. Результат её работы представлен на рисунке 1.

Рисунок 1 – Результаты работы программы

Приложение А. Текст программы

**DAExpansions.cs**

namespace DAExpansion

{

public static class DAExpansions

{

public static void Filter<T>(this DinamicArrayNS.DinamicArray<T> dinamicArray,

Func<T, bool> elementRemains)

{

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

if (elementRemains(dinamicArray.ElementAt(i)) == false)

dinamicArray.RemoveAt(i);

}

public static void Sort<T>(this DinamicArrayNS.DinamicArray<T> dinamicArray,

Func<T, T, int> compare)

{

for (int i = 0; i < dinamicArray.Count - 1; i++)

for (int j = 0; j < dinamicArray.Count - i - 1; j++)

if (compare(dinamicArray.ElementAt(j),

dinamicArray.ElementAt(j+1))>0)

{

T temp = dinamicArray.ElementAt(j + 1);

dinamicArray.RemoveAt(j+1);

dinamicArray.Insert(temp, j);

}

}

}

}

**Program.cs**

using DinamicArrayNS;

using DAExpansion;

/\* 1 \*/

DinamicArray<int> numbers = [-7, 5, 0, -6, 3];

Console.WriteLine("\t Массив до:");

foreach (var number in numbers)

Console.Write($"{number.ToString()} ");

Console.WriteLine("\n");

numbers.Filter((x) => x >= 0);

Console.WriteLine("\t Массив после:");

foreach (var number in numbers)

Console.Write($"{number.ToString()} ");

Console.WriteLine("\n");

/\* 2 \*/

DinamicArray<string> strings =

[

"На поле берёзка стояла",

"Видит: машина стоит",

"А она ему как раз",

"Кольца кальмара у синагоги не покупайте"

];

Console.WriteLine("\t Строки до:");

foreach (var str in strings)

Console.WriteLine(str);

Console.WriteLine("\n");

strings.Sort((str1, str2) => -str1.CompareTo(str2));

Console.WriteLine("\t Строки после:");

foreach (var str in strings)

Console.WriteLine(str);

Console.WriteLine("\n");

массива ПОСЛЕ: {dinamicArray.Capacity}");