Министерство образования и молодежной политики Свердловской области



ГАПОУ СО «Екатеринбургский колледж транспортного строительства»

Отчёт по программе «**Решение задач используя методы LINQ**»

Выполнил: Федоров Д.А.

Группа: ПР-23

Преподаватель: Мирошниченко Г.В

2025

**Задание 3:**

Дан массив чисел с плавающей запятой.

Напишите программу:

a) Для вывода на экран чисел и частоты (ch), с которой встречается число в заданном массиве.

b) Для вывода на экран нового массива, в котором a[i]=a[i]\*частоту

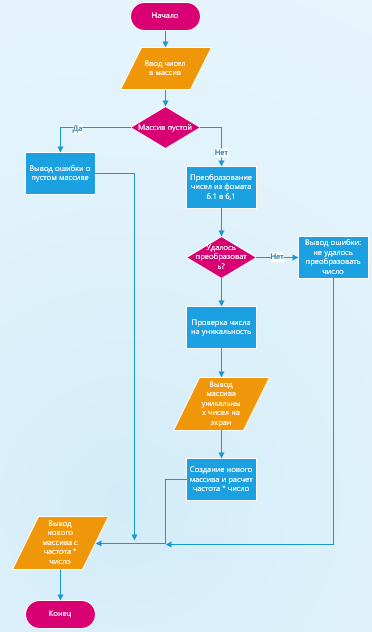
**Входные данные:**

Массив из некоторого количества чисел.

**Выходные данные:**

2 массива: 1 массив показывает число и его частоту, 2 массив умножает число на его частоту

**Блок-схема:**

****

**Код программы:**

Program.cs:

static void Main(string[] args)

{

Console.WriteLine("Введите элементы массива (через пробел):");

string input = Console.ReadLine();

double[] mas = ConvertInputToDoubleMas(input);

if (mas.Length == 0)

{

Console.WriteLine("Массив пустой. Проверьте ввод.");

return;

}

double[] uniqueMas = mas.Distinct().ToArray();

int[] ch = new int[uniqueMas.Length];

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

{

double current = uniqueMas[i];

ch[i] = mas.Count(x => Math.Abs(x - current) < 1e-10);

}

Console.WriteLine("\nЧисло\t\tЧастота");

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

{

Console.WriteLine($"{uniqueMas[i]:F1}\t\t{ch[i]}");

}

double[] newMas = new double[mas.Length];

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

{

double current = mas[i];

for (int j = 0; j < uniqueMas.Length; j++)

{

if (Math.Abs(mas[i] - uniqueMas[j]) < 1e-10)

{

newMas[i] = current \* ch[j];

break;

}

}

}

Console.WriteLine("\nНовый массив:");

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

{

Console.WriteLine($"{newMas[i]:F1} ");

}

Console.WriteLine();

}

static double[] ConvertInputToDoubleMas(string input)

{

string[] parts = input.Trim().Split(new[] { ' ' }, StringSplitOptions.RemoveEmptyEntries);

List<double> result = new List<double>();

foreach (string part in parts)

{

string normalized = part.Replace('.', ',');

if (double.TryParse(normalized, out double number))

{

result.Add(number);

}

else

{

Console.WriteLine($"Не удалось преобразовать {part}");

}

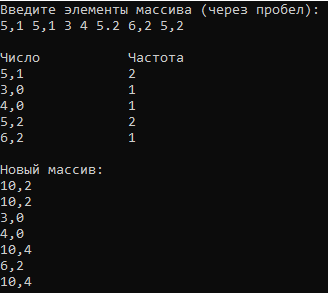
}

return result.ToArray();

}

}

**Тестовые ситуации:**

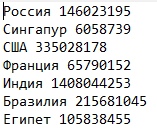
****

**Задание 4**:

Напишите программу для отображения отсортированного списка стран в соответствии с длиной строки (названия страны), затем по имени в порядке возрастания, у которых численность населения больше n=104 000 000 человек (численность вводится с клавиатуры пользователем). Тип данных – коллекция. В текстовом файле записана информация о странах (название страны, численность населения).

**Входные данные:**

Файл countries.txt с странами и их численностью населения

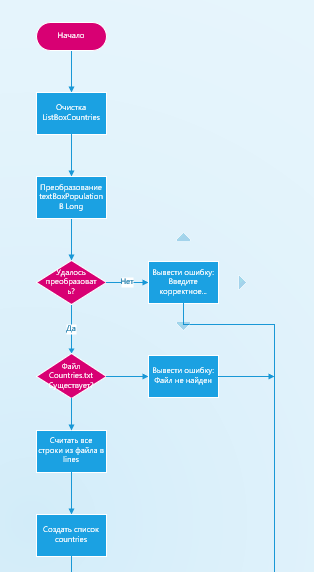


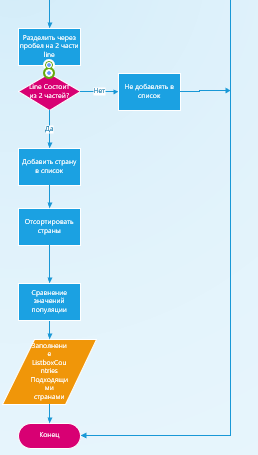
Число n население

**Выходные данные:**

Упорядоченный список стран, у которых численность населения больше n

**Блок-схема:**

****

****

**Код программы:**

**Country.cs:**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace pr16\_4\_Fedorov\_PR\_23

{

public class Country

{

public string Name { get; set; }

public long Population { get; set; }

}

}

**Form1.cs:**

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using System.IO;

namespace pr16\_4\_Fedorov\_PR\_23

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

}

private void buttonLoadCountries\_Click(object sender, EventArgs e)

{

listBoxCountries.Items.Clear();

if (!long.TryParse(textBoxPopulation.Text, out long n))

{

MessageBox.Show("Введите корректное значение Численности населения.");

return;

}

string filePath = "countries.txt";

if (!File.Exists(filePath))

{

MessageBox.Show("Файл не найден.");

return;

}

string[] lines = File.ReadAllLines(filePath);

List<Country> countries = new List<Country>();

foreach (string line in lines)

{

string[] parts = line.Split(new char[] { ' ' }, StringSplitOptions.RemoveEmptyEntries);

if (parts.Length >= 2 && long.TryParse(parts[1], out long population))

{

countries.Add(new Country

{

Name = parts[0],

Population = population

});

}

}

List<Country> filteredCountries = countries

.Where(c => c.Population > n)

.OrderByDescending(c => c.Name.Length)

.ThenBy(c => c.Name)

.ToList();

if (filteredCountries.Count == 0)

{

MessageBox.Show("Нет стран, удовлетворяющих условиям фильтра.");

return;

}

foreach (Country country in filteredCountries)

{

listBoxCountries.Items.Add($"{country.Name} {country.Population}");

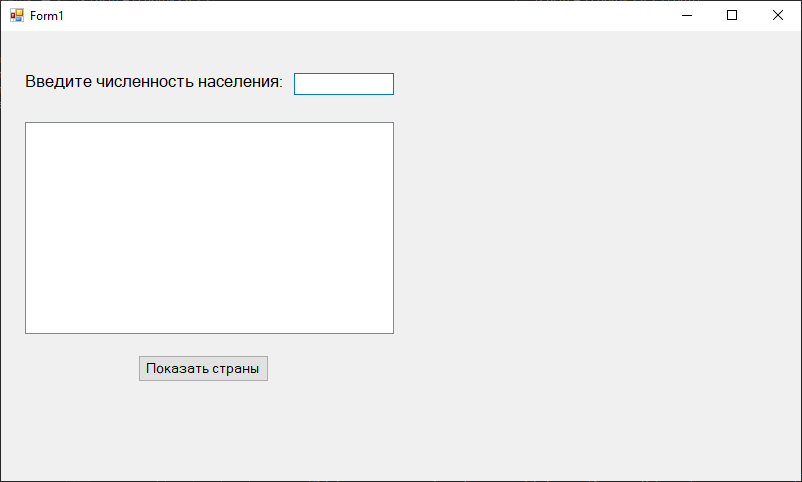
}

}

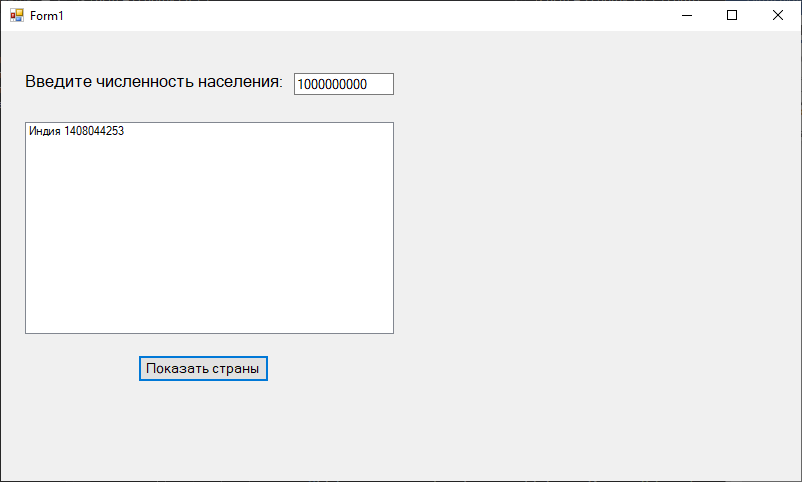
}

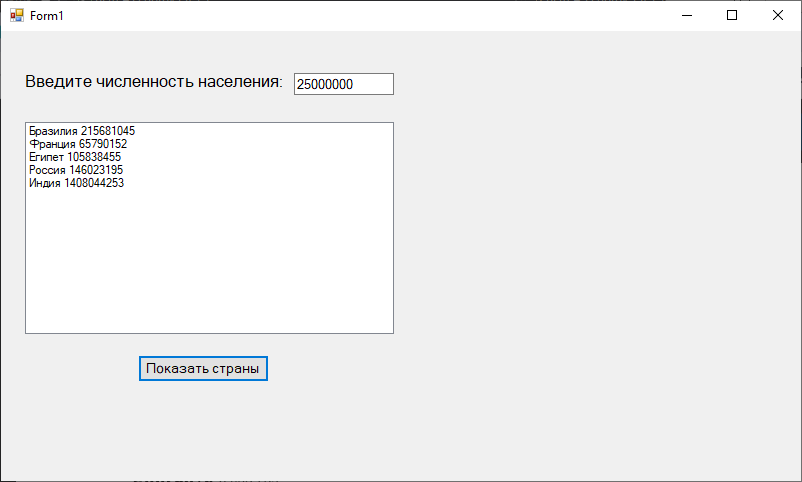
}

**Фото экрана:**

****

**Тестовые ситуации:**





Вывод:

В ходе выполнения данной практической работы были освоены возможности работы с LINQ