

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний аерокосмічний університет ім. М.Є. Жуковського
“Харківський авіаційний інститут”

Кафедра комп’ютерних систем та мереж

Лабораторна робота № 7

“Разработка и исследование программы с использованием
LINQ-запросов”

По дисциплине “Технологии программирования”

XAI.503.525B.123. 1705067.180

Виконав студент гр. 525B Пеналоза Г.
(№ групи) (П.І.Б.)

05-05-19
(підпис, дата)

Перевірив ст. викладач каф. 503
(науковий ступінь, вчене звання, посада)

(підпис, дата) (П.І.Б.)

2019

Задание:

Постановка задачи: разработать программу для обработки последовательностей с использованием LINQ-запросов.

Цель: изучение, использование и исследование языка интегрированных запросов LINQ to Objects.

Проект должен содержать четыре формы:

1-я форма – для задания и редактирования значений и параметров последовательности объектов,

2-я форма – для визуализации результатов обработки последовательностей,

3-я форма – для задания и редактирования значений и параметров последовательности объектов и сравнительной оценки быстродействия методов,

4-я форма – для сведений об авторе проекта (модальная форма, которая может вызываться из первых двух форм).

Вариант 20

14 - 20	массив	Программист	- уникальный идентификатор, - фамилия, - имя, - отчество, - дата рождения, - язык программирования, - должность, - стаж работы, - зарплата
19 - 20	коллекция	Double	Поиск дисперсии

Текст программы

```
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;
```

```

namespace Lab2._2._7
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();

            private void label7_Click(object sender, EventArgs e)
            {

            }
            //Comparison of methods.
            public Visualization visualization = new Visualization();
            /// <summary>
            /// для визуализации результатов обработки последовательностей
            /// </summary>
            public Form2 form2 = new Form2();
            /// <summary>
            /// about
            /// </summary>
            /// <param name="sender"></param>
            /// <param name="e"></param>
            private void button3_Click(object sender, EventArgs e)
            {
                About about = new About();
                about.ShowDialog();
            }
            /// <summary>
            /// для визуализации результатов обработки последовательностей
            /// </summary>
            /// <param name="sender"></param>
            /// <param name="e"></param>
            private void button2_Click(object sender, EventArgs e)
            {
                if (MetodosAUX.programador.Length!=0)
                {
                    form2.Refresh();
                    form2.Show();
                    form2.Focus();
                }
                else
                {
                    MessageBox.Show("There are not entered data!");
                }
            }
        }
    }
}

```

```

    }
}

private void button4_Click(object sender, EventArgs e)
{
    visualization.Show();
}

private void dataGridView1_CellContentClick(object sender,
DataGridViewCellEventArgs e)
{
}

private void dataGridView1_KeyDown(object sender, KeyEventArgs e)
{
}

private void dataGridView1_CellContentDoubleClick(object sender,
DataGridViewCellEventArgs e)
{
}

/// <summary>
/// Edit data
/// </summary>
/// <param name="sender"></param>
/// <param name="e"></param>
private void dataGridView1_CellMouseDoubleClick(object sender,
DataGridViewCellMouseEventArgs e)
{
    String ap, nom, snom, fn, len, pos, el, sal;
    ap= Microsoft.VisualBasic.Interaction.InputBox("фамилия", "Apellido!",
dataGridView1[1, e.RowIndex].Value.ToString());
    if (ap=="")
    {
        ap = dataGridView1[1, e.RowIndex].Value.ToString();
    }
    nom= Microsoft.VisualBasic.Interaction.InputBox("имя", "Nombre!",
dataGridView1[2, e.RowIndex].Value.ToString());
    if (nom=="")
    {
        nom= dataGridView1[2, e.RowIndex].Value.ToString();
    }
}

```

```

        snom= Microsoft.VisualBasic.Interaction.InputBox("отчество", "Segundo
nombre!", dataGridView1[3, e.RowIndex].Value.ToString());
        if (snom=="")
        {
            snom= dataGridView1[3, e.RowIndex].Value.ToString();
        }
        fn= Microsoft.VisualBasic.Interaction.InputBox("дата рождения", "Fecha
de nacimiento!", dataGridView1[4, e.RowIndex].Value.ToString());
        if (fn=="")
        {
            fn= dataGridView1[4, e.RowIndex].Value.ToString();
        }
        len= Microsoft.VisualBasic.Interaction.InputBox("язык
программирования", "Lenguaje de programacion!", dataGridView1[5,
e.RowIndex].Value.ToString());
        if (len=="")
        {
            len= dataGridView1[5, e.RowIndex].Value.ToString();
        }
        pos= Microsoft.VisualBasic.Interaction.InputBox("должность",
"Posicion!", dataGridView1[6, e.RowIndex].Value.ToString());
        if (pos=="")
        {
            pos = dataGridView1[6, e.RowIndex].Value.ToString();
        }
        el= Microsoft.VisualBasic.Interaction.InputBox("стаж работы",
"Experiencia Laboral!", dataGridView1[7, e.RowIndex].Value.ToString());
        if (el=="")
        {
            el= dataGridView1[7, e.RowIndex].Value.ToString();
        }
        sal = Microsoft.VisualBasic.Interaction.InputBox("зарплата", "Salario!",
dataGridView1[8, e.RowIndex].Value.ToString());
        if (sal=="")
        {
            sal= dataGridView1[8, e.RowIndex].Value.ToString();
        }
        dataGridView1[1, e.RowIndex].Value = ap;
        dataGridView1[2, e.RowIndex].Value = nom;
        dataGridView1[3, e.RowIndex].Value = snom;
        dataGridView1[4, e.RowIndex].Value = fn;
        dataGridView1[5, e.RowIndex].Value = len;
        dataGridView1[6, e.RowIndex].Value = pos;
        dataGridView1[7, e.RowIndex].Value = el;
        dataGridView1[8, e.RowIndex].Value = sal;

```

```

        Programador progra = new Programador(e.RowIndex, dataGridView1[1,
e.RowIndex].Value.ToString(), dataGridView1[2, e.RowIndex].Value.ToString(),
        dataGridView1[3, e.RowIndex].Value.ToString(), dataGridView1[4,
e.RowIndex].Value.ToString(), dataGridView1[5, e.RowIndex].Value.ToString(),
        dataGridView1[6,
e.RowIndex].Value.ToString(),Int32.Parse(dataGridView1[7,
e.RowIndex].Value.ToString()),double.Parse(dataGridView1[8,
e.RowIndex].Value.ToString()));
        MetodosAUX.programador[e.RowIndex] =progra;
    }

    private void maskedTextBox1_Validating(object sender, CancelEventArgs e)
    {

    }
    /// <summary>
    /// Verify data
    /// </summary>
    /// <param name="sender"></param>
    /// <param name="e"></param>
    private void maskedTextBox1_TypeValidationCompleted(object sender,
TypeValidationEventArgs e)
    {
        if (!e.IsValidInput)
        {
            MessageBox.Show("The data you supplied must be a valid date in the
format mm/dd/yyyy.");
            maskedTextBox1.Text = "";
        }
        else
        {
            DateTime userDate = (DateTime)e.ReturnValue;
            if (userDate > DateTime.Now)
            {
                MessageBox.Show("Invalid Date");
                //MessageBox.Show("The date in this field must be less than today's
date.");
                maskedTextBox1.Text = "";
                e.Cancel = true;
            }
        }
    }
    /// <summary>
    /// Add data
    /// </summary>

```

```

/// <param name="sender"></param>
/// <param name="e"></param>
private void button1_Click(object sender, EventArgs e)
{
    if (textBox2.Text.Trim()!=""&&textBox3.Text.Trim()!
="""&&textBox4.Text.Trim()!="""&&textBox6.Text.Trim()!
="""&&textBox7.Text.Trim()!="""&&textBox8.Text.Trim()!
="""&&textBox9.Text.Trim()!="""&&maskedTextBox1.Text!= " / /")
    {
        Array.Resize<Programador>(ref MetodosAUX.programador,
MetodosAUX.programador.Length + 1);
        Programador _programador = new
Programador(MetodosAUX.id+1,textBox2.Text.Trim(),textBox3.Text.Trim(),text
Box4.Text.Trim(),maskedTextBox1.Text,textBox6.Text,textBox7.Text,Int32.Parse
(textBox8.Text.Trim()),double.Parse(textBox9.Text.Trim()));
        MetodosAUX.programador[MetodosAUX.id] = _programador;
        dataGridView1.Rows.Add();
        dataGridView1[0, MetodosAUX.id].Value = _programador.id;
        dataGridView1[1, MetodosAUX.id].Value = _programador.familia;
        dataGridView1[2, MetodosAUX.id].Value = _programador.nombre;
        dataGridView1[3, MetodosAUX.id].Value = _programador.snombre;
        dataGridView1[4, MetodosAUX.id].Value = _programador.fnacimiento;
        dataGridView1[5, MetodosAUX.id].Value = _programador.lenguajeP;
        dataGridView1[6, MetodosAUX.id].Value = _programador.posicion;
        dataGridView1[7, MetodosAUX.id].Value = _programador.exLaboral;
        dataGridView1[8, MetodosAUX.id].Value = _programador.salario;
        MessageBox.Show("Information added!");
        MetodosAUX.id = MetodosAUX.id + 1;
    }
    else
    {
        MessageBox.Show("Enter all the data!");
    }
}

private void maskedTextBox1_Click(object sender, EventArgs e)
{
}

//Validate the last textboxes
private void textBox8_Validated(object sender, EventArgs e)
{
    try
    {
        int tmp = Int32.Parse(textBox8.Text);
    }
}

```

```

        catch
        {
            MessageBox.Show("Insert only numbers!");
            textBox8.Text = "";
        }
    }

    private void textBox9_Validated(object sender, EventArgs e)
    {
        try
        {
            int tmp = Int32.Parse(textBox9.Text);
        }
        catch
        {
            MessageBox.Show("Insert only decimal numbers!");
            textBox9.Text = "";
        }
    }

    private void Form1_Load(object sender, EventArgs e)
    {
    }
}
}
Form2.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;

namespace Lab2._2._7
{
    public partial class Form2 : Form
    {
        public Form2()
        {
            InitializeComponent();
        }
    }
}

```



```

private void exitToolStripMenuItem_Click(object sender, EventArgs e)
{
    this.Visible = false;
}

private void aboutToolStripMenuItem_Click(object sender, EventArgs e)
{
    About about = new About();
    about.ShowDialog();
}

        private void dataGridView1_CellContentClick(object sender,
DataGridViewCellEventArgs e)
    {

    }

private void Form2_Load(object sender, EventArgs e)
{

}
/// <summary>
/// Set data
/// </summary>
public void CargarData()
{
    dataGridView1.Rows.Clear();
    for (int i=0;i<MetodosAUX.programador.Length;i++)
    {
        dataGridView1.Rows.Add();
        dataGridView1[0, i].Value = MetodosAUX.programador[i].id;
        dataGridView1[1, i].Value = MetodosAUX.programador[i].familia;
        dataGridView1[2, i].Value = MetodosAUX.programador[i].nombre;
        dataGridView1[3, i].Value = MetodosAUX.programador[i].snombre;
        dataGridView1[4, i].Value = MetodosAUX.programador[i].fnacimiento;
        dataGridView1[5, i].Value = MetodosAUX.programador[i].lenguajeP;
        dataGridView1[6, i].Value = MetodosAUX.programador[i].posicion;
        dataGridView1[7, i].Value = MetodosAUX.programador[i].exLaboral;
        dataGridView1[8, i].Value = MetodosAUX.programador[i].salario;
    }
}

private void Form2_Activated(object sender, EventArgs e)
{
    CargarData();
}

```

```

    }

    private void radioButton1_CheckedChanged(object sender, EventArgs e)
    {

    }

    /////Radiobuttons with the filters, groups,sort*****
    private void radioButton1_Click(object sender, EventArgs e)
    {
        dataGridview1.Rows.Clear();
        int i = 0;
        var s1 = from ap in MetodosAUX.programador orderby ap.familia select
ap;
        foreach (var s in s1)
        {
            dataGridview1.Rows.Add();
            dataGridview1[0, i].Value = s.id;
            dataGridview1[1, i].Value = s.familia;
            dataGridview1[2, i].Value = s.nombre;
            dataGridview1[3, i].Value = s.snombre;
            dataGridview1[4, i].Value = s.fnacimiento;
            dataGridview1[5, i].Value = s.lenguajeP;
            dataGridview1[6, i].Value = s.posicion;
            dataGridview1[7, i].Value = s.exLaboral;
            dataGridview1[8, i].Value = s.salario;
            i = i + 1;
        }
    }

    private void radioButton2_Click(object sender, EventArgs e)
    {
        dataGridview1.Rows.Clear();
        int i = 0;
        var s2 = from ap in MetodosAUX.programador orderby ap.nombre select
ap;
        foreach (var s in s2)
        {
            dataGridview1.Rows.Add();
            dataGridview1[0, i].Value = s.id;
            dataGridview1[1, i].Value = s.familia;
            dataGridview1[2, i].Value = s.nombre;
            dataGridview1[3, i].Value = s.snombre;
            dataGridview1[4, i].Value = s.fnacimiento;
            dataGridview1[5, i].Value = s.lenguajeP;
            dataGridview1[6, i].Value = s.posicion;

```

```

        dataGridView1[7, i].Value = s.exLaboral;
        dataGridView1[8, i].Value = s.salario;
        i = i + 1;
    }
}

```

```

private void radioButton3_Click(object sender, EventArgs e)
{
    dataGridView1.Rows.Clear();
    int i = 0;
    var s3 = from ap in MetodosAUX.programador orderby ap.lenguajeP select
ap;
    foreach (var s in s3)
    {
        dataGridView1.Rows.Add();
        dataGridView1[0, i].Value = s.id;
        dataGridView1[1, i].Value = s.familia;
        dataGridView1[2, i].Value = s.nombre;
        dataGridView1[3, i].Value = s.snombre;
        dataGridView1[4, i].Value = s.fnacimiento;
        dataGridView1[5, i].Value = s.lenguajeP;
        dataGridView1[6, i].Value = s.posicion;
        dataGridView1[7, i].Value = s.exLaboral;
        dataGridView1[8, i].Value = s.salario;
        i = i + 1;
    }
}

```

```

private void radioButton4_Click(object sender, EventArgs e)
{
    this.Refresh();
    var fil = from ap in MetodosAUX.programador group ap by ap.lenguajeP;
    List<String> len = new List<string>();
    foreach (var i in fil)
    {
        len.Add(i.Key);
    }
    Filtro f = new Filtro(len.ToArray());
    f.ShowDialog();
    int ii = 0;
    var newdata = from j in MetodosAUX.programador where
j.lenguajeP==f.fil_selec select j;
    dataGridView1.Rows.Clear();
    foreach (var jr in newdata)
    {

```

```

        dataGridView1.Rows.Add();
        dataGridView1[0, ii].Value = jr.id;
        dataGridView1[1, ii].Value = jr.familia;
        dataGridView1[2, ii].Value = jr.nombre;
        dataGridView1[3, ii].Value = jr.snombre;
        dataGridView1[4, ii].Value = jr.fnacimiento;
        dataGridView1[5, ii].Value = jr.lenguajeP;
        dataGridView1[6, ii].Value = jr.posicion;
        dataGridView1[7, ii].Value = jr.exLaboral;
        dataGridView1[8, ii].Value = jr.salario;
        ii = ii + 1;
    }
}

```

```

private void radioButton8_Click(object sender, EventArgs e)
{
    dataGridView1.Rows.Clear();
    int i = 0;
    var s3 = from ap in MetodosAUX.programador group ap by ap.familia;
    foreach (var ss in s3)
    {
        foreach (Programador s in ss) {
            dataGridView1.Rows.Add();
            dataGridView1[0, i].Value = s.id;
            dataGridView1[1, i].Value = s.familia;
            dataGridView1[2, i].Value = s.nombre;
            dataGridView1[3, i].Value = s.snombre;
            dataGridView1[4, i].Value = s.fnacimiento;
            dataGridView1[5, i].Value = s.lenguajeP;
            dataGridView1[6, i].Value = s.posicion;
            dataGridView1[7, i].Value = s.exLaboral;
            dataGridView1[8, i].Value = s.salario;
            i = i + 1;
        }
    }
}

```

```

private void radioButton9_Click(object sender, EventArgs e)
{
    dataGridView1.Rows.Clear();
    int i = 0;
    var s3 = from ap in MetodosAUX.programador group ap by ap.salario;
    foreach (var ss in s3)
    {
        foreach (Programador s in ss)

```

```

    {
        dataGridView1.Rows.Add();
        dataGridView1[0, i].Value = s.id;
        dataGridView1[1, i].Value = s.familia;
        dataGridView1[2, i].Value = s.nombre;
        dataGridView1[3, i].Value = s.snombre;
        dataGridView1[4, i].Value = s.fnacimiento;
        dataGridView1[5, i].Value = s.lenguajeP;
        dataGridView1[6, i].Value = s.posicion;
        dataGridView1[7, i].Value = s.exLaboral;
        dataGridView1[8, i].Value = s.salario;
        i = i + 1;
    }
}

private void radioButton10_Click(object sender, EventArgs e)
{
    dataGridView1.Rows.Clear();
    int i = 0;
    var s3 = from ap in MetodosAUX.programador group ap by ap.lenguajeP;
    foreach (var ss in s3)
    {
        foreach (Programador s in ss)
        {
            dataGridView1.Rows.Add();
            dataGridView1[0, i].Value = s.id;
            dataGridView1[1, i].Value = s.familia;
            dataGridView1[2, i].Value = s.nombre;
            dataGridView1[3, i].Value = s.snombre;
            dataGridView1[4, i].Value = s.fnacimiento;
            dataGridView1[5, i].Value = s.lenguajeP;
            dataGridView1[6, i].Value = s.posicion;
            dataGridView1[7, i].Value = s.exLaboral;
            dataGridView1[8, i].Value = s.salario;
            i = i + 1;
        }
    }
}

private void radioButton11_Click(object sender, EventArgs e)
{
    var s3 = (from st in MetodosAUX.programador select st.salario).Max();
    MessageBox.Show("MAX зарплата: "+s3);
}

```

```

private void radioButton12_CheckedChanged(object sender, EventArgs e)
{

}

private void radioButton13_CheckedChanged(object sender, EventArgs e)
{

}

private void radioButton12_Click(object sender, EventArgs e)
{
    var s3 = (from st in MetodosAUX.programador select st.exLaboral).Max();
    MessageBox.Show("MAX стаж работы: " + s3);
}

private void radioButton13_Click(object sender, EventArgs e)
{
    var s3 = MetodosAUX.programador.GroupBy(s =>
s.lenguajeP).OrderByDescending(s => s.Count()).First().Key;
    MessageBox.Show("MAX язык программирования: " + s3);
}

private void radioButton14_Click(object sender, EventArgs e)
{
    var s3 = from d in MetodosAUX.programador where (DateTime.Now -
Convert.ToDateTime(d.fnacimiento)).Days / 365 >= 20 select d.nombre;
    foreach (var i in s3)
    {
        MessageBox.Show("Name: "+i);
    }
}

private void radioButton14_CheckedChanged(object sender, EventArgs e)
{
}

private void radioButton4_CheckedChanged(object sender, EventArgs e)
{

}

private void radioButton5_Click(object sender, EventArgs e)

```

```

{
    this.Refresh();
    var fil = from ap in MetodosAUX.programador group ap by ap.posicion;
    List<String> len = new List<string>();
    foreach (var i in fil)
    {
        len.Add(i.Key);
    }
    Filtro f = new Filtro(len.ToArray());
    f.ShowDialog();
    int ii = 0;
    var newdata = from j in MetodosAUX.programador where j.posicion ==
f.fil_selec select j;
    dataGridView1.Rows.Clear();
    foreach (var jr in newdata)
    {
        dataGridView1.Rows.Add();
        dataGridView1[0, ii].Value = jr.id;
        dataGridView1[1, ii].Value = jr.familia;
        dataGridView1[2, ii].Value = jr.nombre;
        dataGridView1[3, ii].Value = jr.snombre;
        dataGridView1[4, ii].Value = jr.fnacimiento;
        dataGridView1[5, ii].Value = jr.lenguajeP;
        dataGridView1[6, ii].Value = jr.posicion;
        dataGridView1[7, ii].Value = jr.exLaboral;
        dataGridView1[8, ii].Value = jr.salario;
        ii = ii + 1;
    }
}

private void radioButton6_Click(object sender, EventArgs e)
{
    this.Refresh();
    var fil = from ap in MetodosAUX.programador group ap by ap.familia;
    List<String> len = new List<string>();
    foreach (var i in fil)
    {
        len.Add(i.Key);
    }
    Filtro f = new Filtro(len.ToArray());
    f.ShowDialog();
    int ii = 0;
    var newdata = from j in MetodosAUX.programador where j.familia ==
f.fil_selec select j;
    dataGridView1.Rows.Clear();

```

```

foreach (var jr in newdata)
{
    dataGridview1.Rows.Add();
    dataGridview1[0, ii].Value = jr.id;
    dataGridview1[1, ii].Value = jr.familia;
    dataGridview1[2, ii].Value = jr.nombre;
    dataGridview1[3, ii].Value = jr.snombre;
    dataGridview1[4, ii].Value = jr.fnacimiento;
    dataGridview1[5, ii].Value = jr.lenguajeP;
    dataGridview1[6, ii].Value = jr.posicion;
    dataGridview1[7, ii].Value = jr.exLaboral;
    dataGridview1[8, ii].Value = jr.salario;
    ii = ii + 1;
}
}

private void radioButton7_CheckedChanged(object sender, EventArgs e)
{

}

private void radioButton7_Click(object sender, EventArgs e)
{
    this.Refresh();
    var fil = from ap in MetodosAUX.programador group ap by ap.salario;
    List<double> len = new List<double>();
    foreach (var i in fil)
    {
        len.Add(i.Key);
    }
    Filtro f = new Filtro(len.ToArray());
    f.ShowDialog();
    int ii = 0;
    var newdata = from j in MetodosAUX.programador where j.salario ==
double.Parse(f.fil_selec) select j;
    dataGridview1.Rows.Clear();
    foreach (var jr in newdata)
    {
        dataGridview1.Rows.Add();
        dataGridview1[0, ii].Value = jr.id;
        dataGridview1[1, ii].Value = jr.familia;
        dataGridview1[2, ii].Value = jr.nombre;
        dataGridview1[3, ii].Value = jr.snombre;
        dataGridview1[4, ii].Value = jr.fnacimiento;
        dataGridview1[5, ii].Value = jr.lenguajeP;
    }
}

```



```

        dataGridView1[6, ii].Value = jr.posicion;
        dataGridView1[7, ii].Value = jr.exLaboral;
        dataGridView1[8, ii].Value = jr.salario;
        ii = ii + 1;
    }
}
////////*****
}
}
Filtro.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;

namespace Lab2._2._7
{
    /// <summary>
    /// Filters for radiobuttons
    /// </summary>
    public partial class Filtro : Form
    {
        public Array a;
        public String fil_selec;
        public Filtro(Array array)
        {
            a = array;
            InitializeComponent();
        }

        private void Filtro_Load(object sender, EventArgs e)
        {
            comboBox1.Items.Clear();
            foreach (var i in a)
            {
                comboBox1.Items.Add(i);
            }
            comboBox1.SelectedIndex = 0;
        }
    }
}

```

```

        private void comboBox1_SelectedIndexChanged(object sender, EventArgs e)
        {
            fil_selec = comboBox1.SelectedItem.ToString();
        }
    }
}
About.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;

namespace Lab2._2._7
{
    public partial class About : Form
    {
        public About()
        {
            InitializeComponent();
        }
    }
}
MetodosAUX.cs
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace Lab2._2._7
{
    public class MetodosAUX
    {
        public static int id = 0;
        public static Programador[] programador = new Programador[0];
    }
}
Program.cs
using System;
using System.Collections.Generic;

```

```
using System.Linq;
using System.Threading.Tasks;
using System.Windows.Forms;
```

```
namespace Lab2._2._7
{
    static class Program
    {
        /// <summary>
        /// Punto de entrada principal para la aplicación.
        /// </summary>
        [STAThread]
        static void Main()
        {
            Application.EnableVisualStyles();
            Application.SetCompatibleTextRenderingDefault(false);
            Application.Run(new Form1());
        }
    }
}
```

Programador.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
```

```
namespace Lab2._2._7
{
    public class Programador
    {
        public int id;
        public String familia;
        public String nombre;
        public String snombre;
        public String fnacimiento;
        public String lenguajeP;
        public String posicion;
        public int exLaboral;
        public double salario;
        public Programador(int _id,String _familia,String _nombre, String
        _snombre,String _fnacimiento,String _lenguajeP,
        String _posicion,int _exLaboral,double _salario)
        {
            id = _id;
```

```

        familia = _familia;
        nombre = _nombre;
        snombre = _snombre;
        fnacimiento = _fnacimiento;
        lenguajeP = _lenguajeP;
        posicion = _posicion;
        exLaboral = _exLaboral;
        salario = _salario;
    }
}
}
Visualization.cs
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Diagnostics;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace Lab2._2._7
{
    public partial class Visualization : Form
    {
        public Visualization()
        {
            InitializeComponent();
        }

        private void exitToolStripMenuItem_Click(object sender, EventArgs e)
        {
            this.Hide();
        }

        public Random random = new Random();
        public List<int> DATA = new List<int>();
        /// <summary>
        /// Generate and upload information
        /// </summary>
        /// <param name="sender"></param>
        /// <param name="e"></param>
        private void Visualization_Activated(object sender, EventArgs e)
        {

```

```

dataGridView1.Rows.Clear();
foreach (var s in chart1.Series)
{
    s.Points.Clear();
}
Stopwatch sw = new Stopwatch();
for (int i=0;i< random.Next(5, 7); i++) {
    for (int c = 0; c < 1000*(i+1); c++)
    {
        int val = random.Next(-1000, 1000);
        DATA.Add(val);
    }
    double c1, c2, l1, l2, timeSpan;
    sw.Reset();
    dataGridView1.Rows.Add();
    sw.Start();
    c1 = VarianzaCS1(DATA);
    sw.Stop();
    timeSpan = sw.ElapsedMilliseconds;
    dataGridView1[0, i*4].Value = "C#";
    dataGridView1[1, i*4].Value = c1;
    dataGridView1[2, i*4].Value = "дисперсии";
    dataGridView1[3, i*4].Value = timeSpan;
    chart1.Series[0].Points.AddXY(1000*(i+1), timeSpan);

    sw.Reset();
    dataGridView1.Rows.Add();
    sw.Start();
    c2 = VarianzaCS2(DATA);
    sw.Stop();
    timeSpan = sw.ElapsedMilliseconds;
    dataGridView1[0, i * 4+1].Value = "C# 2";
    dataGridView1[1, i * 4+1].Value = c2;
    dataGridView1[2, i * 4+1].Value = "дисперсии";
    dataGridView1[3, i * 4+1].Value = timeSpan;
    chart1.Series[1].Points.AddXY(1000 * (i + 1), timeSpan);

    sw.Reset();
    dataGridView1.Rows.Add();
    sw.Start();
    l1 = VarianzaLINQ(DATA);
    sw.Stop();
    timeSpan = sw.ElapsedMilliseconds;
    dataGridView1[0, i * 4+2].Value = "LINQ";
    dataGridView1[1, i * 4+2].Value = l1;

```

```

dataGridView1[2, i * 4+2].Value = "дисперсии";
dataGridView1[3, i * 4+2].Value = timeSpan;
chart1.Series[2].Points.AddXY(1000 * (i + 1), timeSpan);

```

```

sw.Reset();
dataGridView1.Rows.Add();
sw.Start();
l2 = VarianzaPLINQ(DATA);
sw.Stop();
timeSpan = sw.ElapsedMilliseconds;
dataGridView1[0, i * 4+3].Value = "PLINQ";
dataGridView1[1, i * 4+3].Value = l2;
dataGridView1[2, i * 4+3].Value = "дисперсии";
dataGridView1[3, i * 4+3].Value = timeSpan;
chart1.Series[3].Points.AddXY(1000 * (i + 1), timeSpan);

```

```

}

```

```

}

```

```

/////Methodos to analyze

```

```

public double VarianzaCS1(IEnumerable<int> values)

```

```

{

```

```

    double ret = 0;

```

```

    for (int c=0;c<values.Count();c++)

```

```

    {

```

```

        ret = ret + Math.Pow((values.ElementAt(c)-values.Average()),2);

```

```

    }

```

```

    ret = ret / (values.Count()-1);

```

```

    return ret;

```

```

}

```

```

public double VarianzaCS2(IEnumerable<int> values)

```

```

{

```

```

    double ret = 0;

```

```

    double Sr = 0;

```

```

    int[] arr = values.ToArray<int>();

```

```

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

```

```

    {

```

```

        Sr += arr[i];

```

```

    }

```

```

    Sr = Sr / arr.Length;

```

```

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

```

```

    {

```

```

        ret = ret + Math.Pow((arr[c]-Sr),2);

```

```

    }

```

```

    ret = ret / (arr.Length-1);

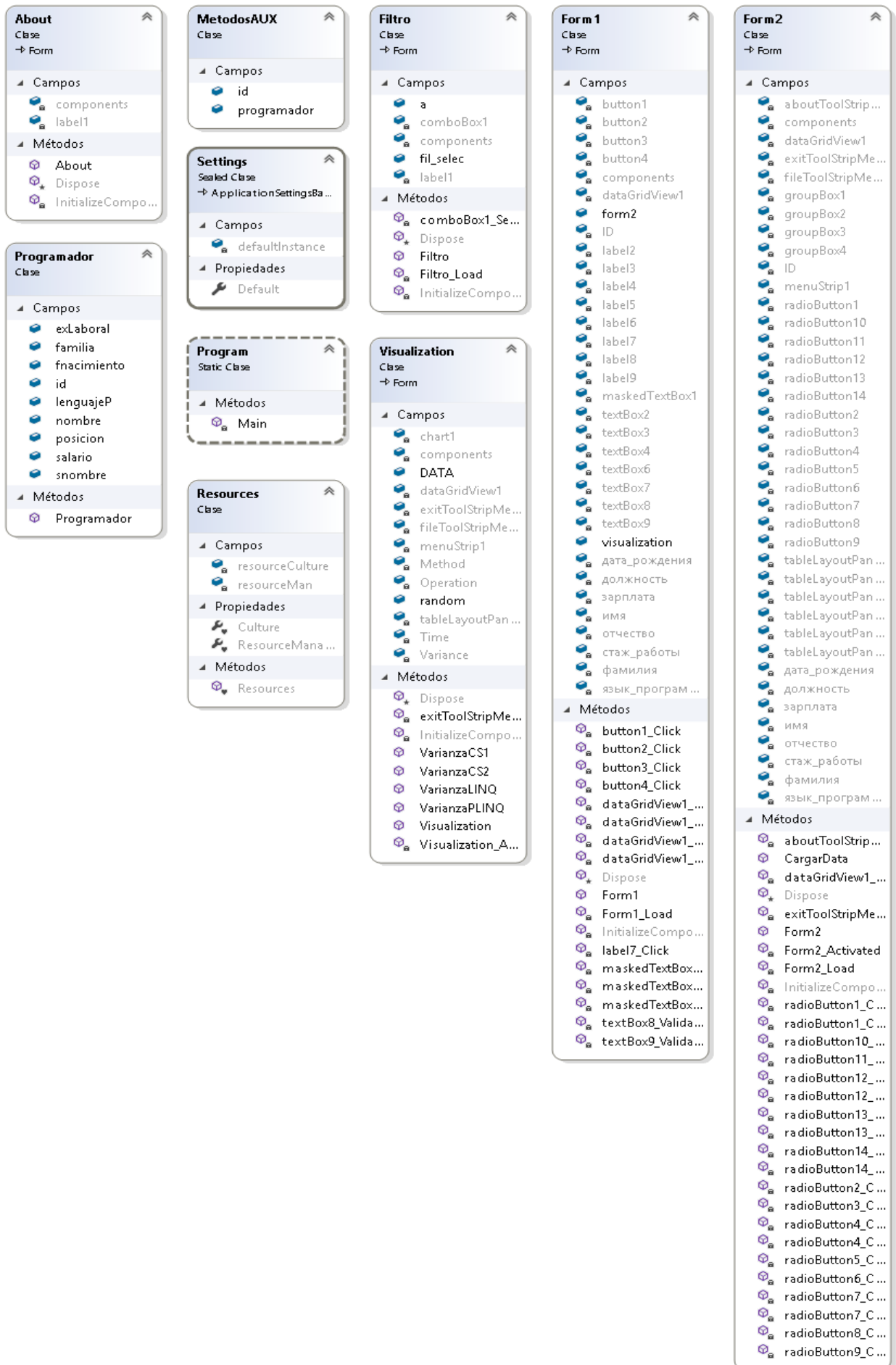
```

```

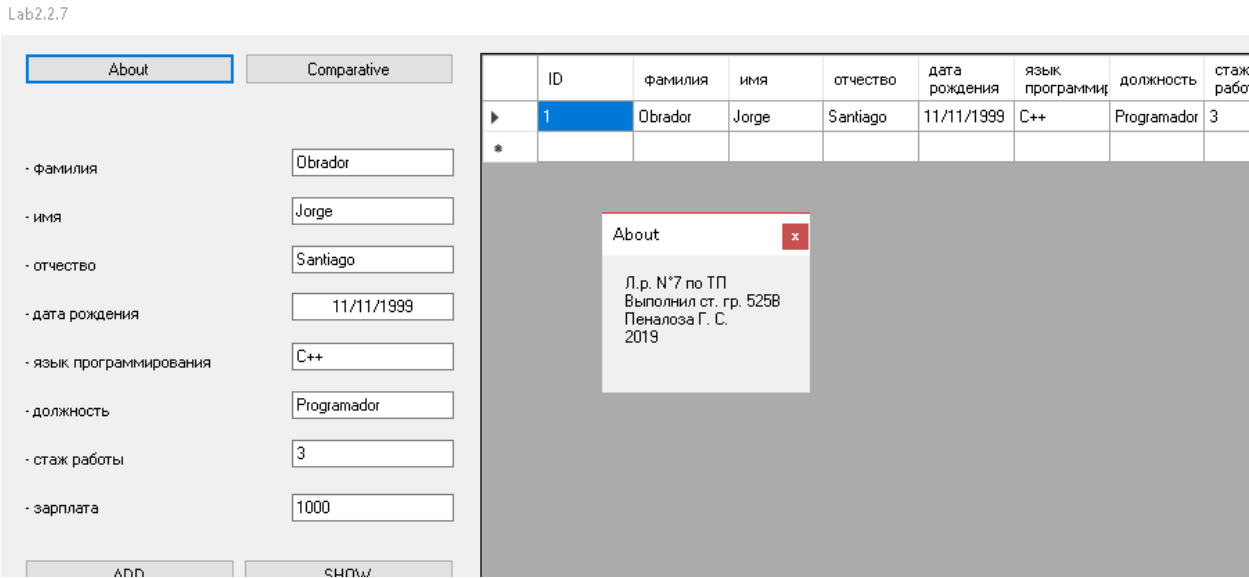
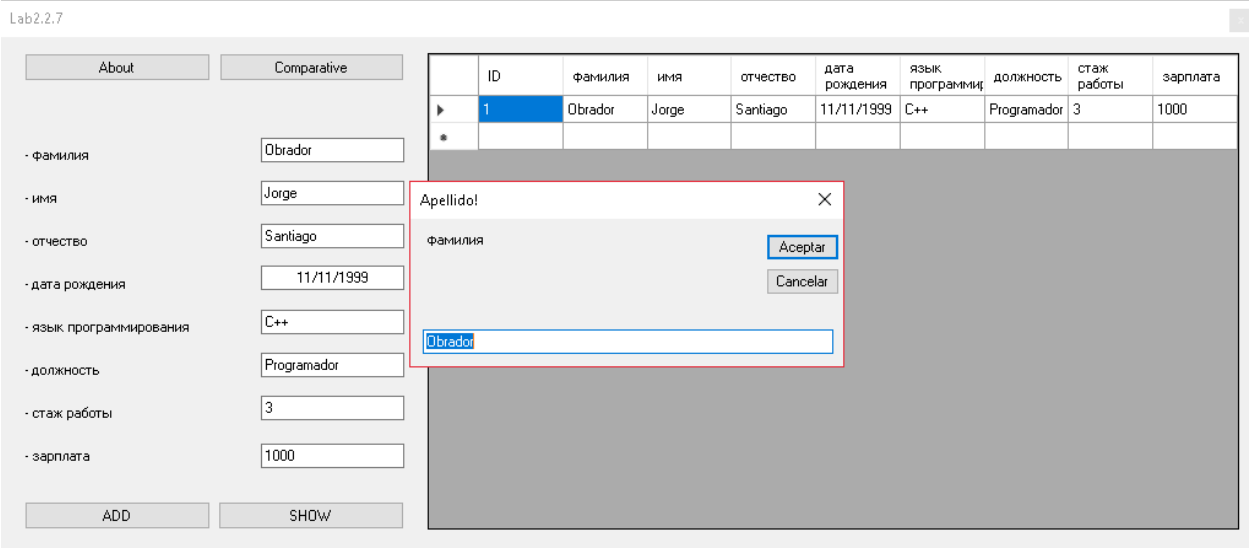
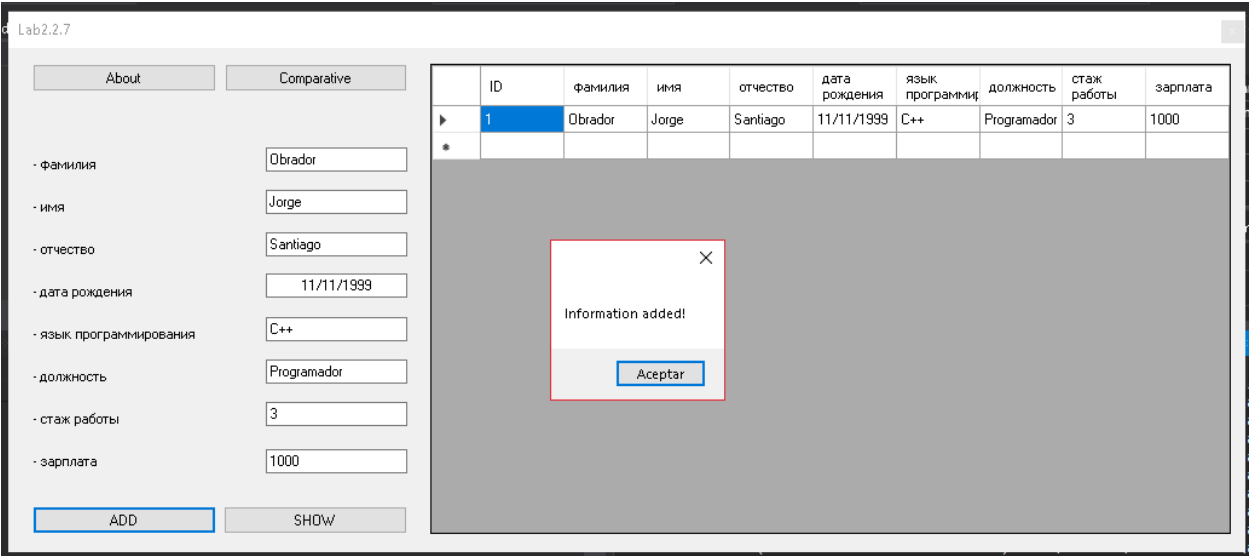
        return ret;
    }
    public double VarianzaPLINQ(IEnumerable<int> values)
    {
        double ret = 0;
        int count = values.Count();
        if (count > 1)
        {
            double avg = values.Average();
            double sum = values.AsParallel().Sum(d => Math.Pow((d - avg), 2));
            ret = sum / (count-1);
        }
        return ret;
    }
    public double VarianzaLINQ(IEnumerable<int> values)
    {
        double ret = 0;
        int count = values.Count();
        if (count > 1)
        {
            double avg = values.Average();
            double sum = values.Sum(d => Math.Pow((d - avg),2));
            ret = sum / (count - 1);
        }
        return ret;
    }
}
}
}

```

Диаграмму классов



Скриншоты



Lab2.2.7

File

	ID	фамилия	имя	отчество	дата рождения	язык программирования	должность	стаж работы	зарплата
▶	0	Obrador	Jorge	Santiago	11/11/1999	C++	Programador	3	1000
	2	Marquez	Pedro	Juan	11/11/1999	Golang	Programador	5	1500
*									

Выполнить сортировку

☒ фамилия
☐ имя
☐ язык программирования

Выполнить фильтрацию

☐ язык программирования
☐ должность
☐ фамилия
☐ зарплата

Выполнить группировку

☐ фамилия
☐ зарплата
☐ язык программирования

Выполнить агрегирования

☐ зарплата максимальная
☐ максимальный стаж
☐ наиболее используемый
☐ до 20 лет

ADDSHOW

	дата рождения	язык программирования	должность
0	11/11/1999	C++	Programador
2	11/11/1999	Golang	Programador

Lab2.2.7

File

	ID	фамилия	имя	отчество	дата рождения	язык программирования	должность	стаж работы	зарплата
▶	0	Obrador	Jorge	Santiago	11/11/1999	C++	Programador	3	1000
	2	Marquez	Pedro	Juan	11/11/1999	Golang	Programador	5	1500
*									

Выполнить сортировку

☐ фамилия
☐ имя

Выполнить фильтрацию

☐ язык программирования
☐ должность
☐ фамилия

Выполнить группировку

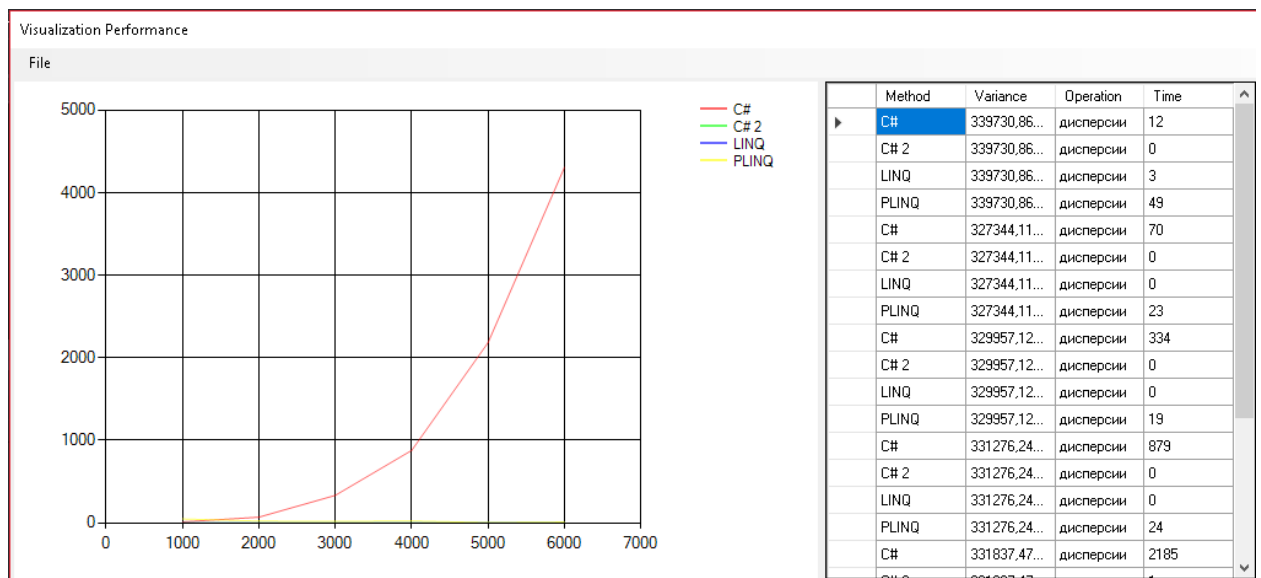
☐ фамилия
☐ зарплата

Выполнить агрегирования

☐ зарплата максимальная
☐ максимальный стаж
☒ наиболее используемый

MAX язык программирования: C++

Aceptar



Раздел “Сортировка”:

	ID	фамилия	имя	отчество	дата рождения	язык программирс	должность	стаж работы	зарплата
▶	0	Obrador	Jorge	Santiago	11/11/1999	C++	Programador	3	1000
	2	Marquez	Pedro	Juan	11/11/1999	Golang	Programador	5	1500
*									

Cod:

```
private void radioButton1_Click(object sender, EventArgs e)
{
    dataGridView1.Rows.Clear();
    int i = 0;
    var s1 = from ap in MetodosAUX.programador orderby ap.familia
select ap;
    foreach (var s in s1)
    {
        dataGridView1.Rows.Add();
        dataGridView1[0, i].Value = s.id;
        dataGridView1[1, i].Value = s.familia;
        dataGridView1[2, i].Value = s.nombre;
        dataGridView1[3, i].Value = s.snombre;
        dataGridView1[4, i].Value = s.fnacimiento;
        dataGridView1[5, i].Value = s.lenguajeP;
        dataGridView1[6, i].Value = s.posicion;
        dataGridView1[7, i].Value = s.exLaboral;
        dataGridView1[8, i].Value = s.salario;
        i = i + 1;
    }
}
```

```

    }
}

```

File									
	ID	фамилия	имя	отчество	дата рождения	язык программирс	должность	стаж работы	зарплата
▶	2	Marquez	Pedro	Juan	11/11/1999	Golang	Programador	5	1500
	0	Obrador	Jorge	Santiago	11/11/1999	C++	Programador	3	1000
*									

Раздел “Фильтрация”:

	ID	фамилия	имя	отчество	дата рождения	язык программирс	должность	стаж работы	зарплата
▶	1	Sanchez	Juan	Pedro	11/11/1998	C++	Programador	3	1000
	2	Martinez	Kevin	Josue	21/11/1999	C++	Programador	3	1000
	3	Marco	Mario	Euler	21/11/1999	Java	Programador	3	1000
*									

ID	фамилия	имя	отчество	дата рождения	язык программирс	должность
1	Sanchez	Juan	Pedro	11/11/1998	C++	Programad
2	Martinez	Kevin	Josue	21/11/1999	C++	Programad
3	Marco	Mario	Euler	21/11/1999	Java	Programad

Filtro

Выберите фильтр.

C++

	ID	фамилия	имя	отчество	дата рождения	язык программирс	должность	стаж работы	зарплата
▶	1	Sanchez	Juan	Pedro	11/11/1998	C++	Programador	3	1000
	2	Martinez	Kevin	Josue	21/11/1999	C++	Programador	3	1000
*									

Cod:

```

private void radioButton4_Click(object sender, EventArgs e)
{
    this.Refresh();
    var fil = from ap in MetodosAUX.programador group ap by
ap.lenguajeP;
    List<String> len = new List<string>();
    foreach (var i in fil)
    {

```


ID	фамилия	имя	отчество	дата рождения	язык программирования	должность	стаж работы	зарплата
1	Lapo	Juan	Pedro	12/11/1994	C++	Programador	3	600
2	Lapo	Santiago	Marco	02/11/1992	C++	Programador	3	600
5	Lapo	David	Lantira	02/01/1993	Golang	Programador	4	800
3	Juerez	Santiago	Marco	02/11/1992	Java	Programador	3	600
4	Martinez	Jose	Marco	02/11/1991	Golang	Programador	4	700
*								

Выполнить сортировку

☒ фамилия

Выполнить фильтрацию

☐ язык программирования

Выполнить группировку

☒ фамилия

Выполнить агрегирования

☐ зарплата максимальная.

Cod:

```

private void radioButton8_Click(object sender, EventArgs e)
{
    dataGridView1.Rows.Clear();
    int i = 0;
    var s3 = from ap in MetodosAUX.programador group ap by
ap.familia;
    foreach (var ss in s3)
    {
        foreach (Programador s in ss) {
            dataGridView1.Rows.Add();
            dataGridView1[0, i].Value = s.id;
            dataGridView1[1, i].Value = s.familia;
            dataGridView1[2, i].Value = s.nombre;
            dataGridView1[3, i].Value = s.snombre;
            dataGridView1[4, i].Value = s.fnacimiento;
            dataGridView1[5, i].Value = s.lenguajeP;
            dataGridView1[6, i].Value = s.posicion;
            dataGridView1[7, i].Value = s.exLaboral;
            dataGridView1[8, i].Value = s.salario;
            i = i + 1;
        }
    }
}

```

Раздел “Агрегирование”:

	ID	фамилия	имя	отчество	дата рождения	язык программирования	должность	стаж работы	зарплата
▶	1	111	111	1111	11/11/1111	111	111	1111	1111
	2	222	222	222	11/11/1111	222	222	222	2222
	3	6666	6666	6666	11/11/1111	6666	6666	6666	5555
	4	333	333	333	11/11/1111	333	333	333	3333
*									

	ID	фамилия	имя	отчество	дата рождения	язык программирования	должность	стаж работы	зарплата
▶	1	111	111	1111	11/11/1111	111	111	1111	1111
	2	222	222	222	11/11/1111	222	222	222	2222
	3	6666	6666	6666	11/11/1111	6666	6666	6666	5555
	4	333	333	333	11/11/1111	333	333	333	3333
*									

X
 MAX зарплата: 5555
 Аceptar

Выполнить сортировку: ☒ фамилия
 Выполнить фильтрацию: ☐ язык программирования
 Выполнить агрегирование: ☒ зарплата максимальная.

Cod:

```
private void radioButton11_Click(object sender, EventArgs e)
{
    var s3 = (from st in MetodosAUX.programador select
st.salario).Max();
    MessageBox.Show("MAX зарплата: "+s3);
}
```

Раздел “Сравнительная оценка быстродействия LINQ-методов”:

CODE M. C#

```
public double VarianzaCS1(IEnumerable<int> values)
{
    double ret = 0;
    for (int c=0;c<values.Count();c++)
    {
        ret = ret + Math.Pow((values.ElementAt(c)-values.Average()),2);
    }
    ret = ret / (values.Count()-1);
    return ret;
}
```

CODE M. C# 2

```

public double VarianzaCS2(IEnumerable<int> values)
{
    double ret = 0;
    double Sr = 0;
    int[] arr = values.ToArray<int>();
    for (int i=0;i<arr.Length;i++)
    {
        Sr += arr[i];
    }
    Sr = Sr / arr.Length;
    for (int c=0;c<arr.Length;c++)
    {
        ret = ret + Math.Pow((arr[c]-Sr),2);
    }
    ret = ret / (arr.Length-1);
    return ret;
}

```

CODE M. LINQ

```

public double VarianzaLINQ(IEnumerable<int> values)
{
    double ret = 0;
    int count = values.Count();
    if (count > 1)
    {
        double avg = values.Average();
        double sum = values.Sum(d => Math.Pow((d - avg),2));
        ret = sum / (count - 1);
    }
    return ret;
}

```

CODE M. PLINQ

```

public double VarianzaPLINQ(IEnumerable<int> values)
{
    double ret = 0;
    int count = values.Count();
    if (count > 1)
    {
        double avg = values.Average();
        double sum = values.AsParallel().Sum(d => Math.Pow((d - avg), 2));
        ret = sum / (count-1);
    }
}

```



```
return ret;  
}
```

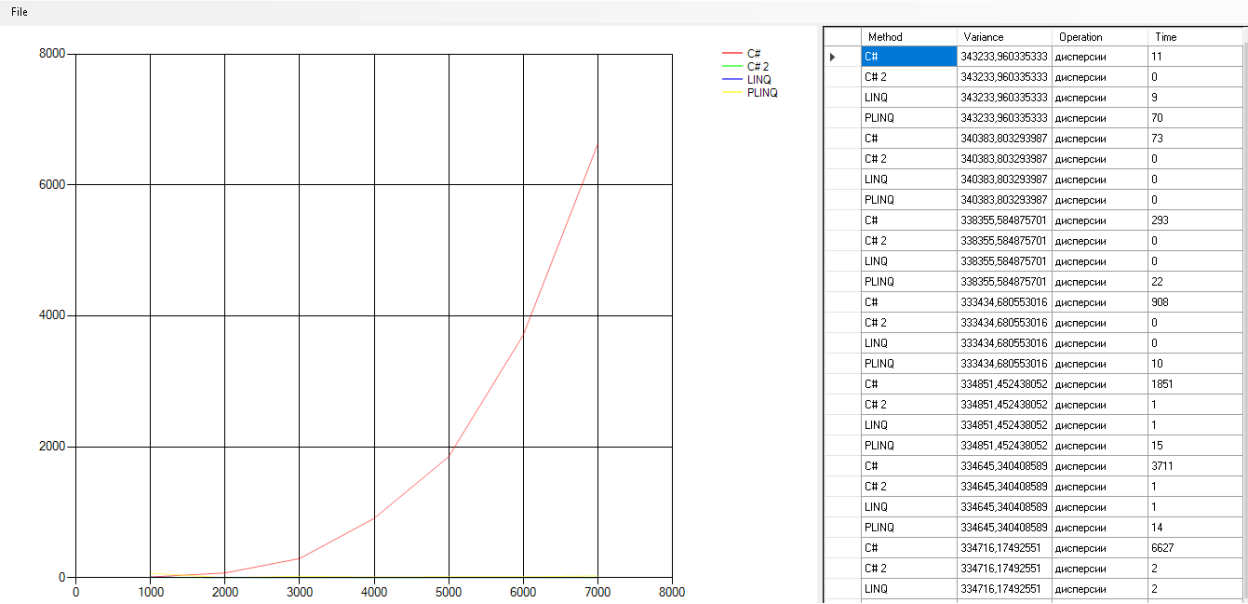


Таблица 1 - Поля и методы класса А и их назначение

№	Поле	Назначение
1	exLaboral	стаж работы
2	familia	фамилия,
3	fnacimiento	дата рождения,
4	id	уникальный идентификатор,
5	lenguajeP	язык программирования,
6	nombre	имя,
7	posicion	должность,
8	salario	зарплата
9	snombre	отчество,
10	id	Current index
11	programador	Array with programmers values
12	fil_selec	Values for the filter
13	DATA	Array with generated information
14
15	GUI objects in the forms	...
16

	Метод	Назначение
17	Programador	Set the data for the programmer array
18	Filtro	Get the array for the filtre
19	VarianzaCS1	1 C# Variance
20	VarianzaCS2	2 C# Variance
21	VarianzaLINQ	Variance with LINQ
22	VarianzaPLINQ	Variance with PLINQ
23	CargarData	Set the programmer array to 2 form

Таблица 2 - Обработчики событий проекта и их назначение

N°	Обработчик события	Назначение
1	button1_Click	Add new item
2	button2_Click	Show the 2 form
3	button3_Click	About program
4	button4_Click	Comparative form
5	dataGridView1_CellMouseDoubleClick	Edit information
6	maskedTextBox1_TypeValidationCompleted	Validation date
7	textBox8_Validated	Validate стаж работы
8	textBox9_Validated	Validate salary
9	aboutToolStripMenuItem_Click	About 2 form
10	radioButton(1-14)_Click	Operations for sort, filter, group and add

Выводы

Для выполнения анализа временной сложности используется «Секундомер» в 4 случаях. Требуется немного более мощное оборудование, потому что, поскольку существует несколько компиляций и с первым неэффективным методом C #, получение результатов происходит очень медленно.

В некоторых случаях работа с LINQ более эффективна, чем PLINQ.

Проект может быть улучшен визуально и с лучшей шкалой для «диаграммы».

Использованные источники

1. <https://docs.microsoft.com/es-es/dotnet/>