

**Московский Государственный Технический Университет имени Н. Э. Баумана**

**Базовые компоненты интернет-технологий**

**Лабораторная работа №4**

**ИСПОЛНИТЕЛЬ:**

**ФИО**

**Болотов Н.А.**

**Группа РТ5-31**

**ПРЕПОДАВАТЕЛЬ:**

**ФИО**

**Гапанюк Ю. Е.**

**Москва 2017**

## Программа лабораторной работы:

Разработать программу, реализующую работу с файлами.

1. Программа должна быть разработана в виде приложения Windows Forms на языке C#. По желанию вместо Windows Forms возможно использование WPF.
2. Добавить кнопку, реализующую функцию чтения файла в список слов `List<string>`.
3. Для выбора имени файла используется класс `OpenFileDialog`, который открывает диалоговое окно с выбором файла. Ограничить выбор только файлами с расширением «.txt».
4. Для чтения из файла рекомендуется использовать статический метод `ReadAllText()` класса `File` (пространство имен `System.IO`). Содержимое файла считывается методом `ReadAllText()` в виде одной строки, далее делится на слова с использованием метода `Split()` класса `string`. Слова сохраняются в список `List<string>`.
5. При сохранении слов в список `List<string>` дубликаты слов не записываются. Для проверки наличия слова в списке используется метод `Contains()`.
6. Вычислить время загрузки и сохранения в список с использованием класса `Stopwatch` (пространство имен `System.Diagnostics`). Вычисленное время вывести на форму в поле ввода (`TextBox`) или надпись (`Label`).
7. Добавить на форму поле ввода для поиска слова и кнопку поиска. При нажатии на кнопку поиска осуществлять поиск введенного слова в списке. Слово считается найденным, если оно входит в элемент списка как подстрока (метод `Contains()` класса `string`).
8. Добавить на форму список (`ListBox`). Найденные слова выводить в список с использованием метода «название\_списка.Items.Add()». Вызовы метода «название\_списка.Items.Add()» должны находиться между вызовами методов «название\_списка.BeginUpdate()» и «название\_списка.EndUpdate()».
9. Вычислить время поиска с использованием класса `Stopwatch`. Вычисленное время вывести на форму в поле ввода (`TextBox`) или надпись (`Label`).

## Код 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;
using System.IO;
using System.Diagnostics;
namespace WindowsFormsApplication3
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }
        List<string> list = new List<string>();
        private void button1_Click(object sender, EventArgs e)
        {
            OpenFileDialog FileDownload = new OpenFileDialog();
            FileDownload.Filter = "текстовые файлы |*.txt";
            if (FileDownload.ShowDialog() == DialogResult.OK)
            {
                Stopwatch t = new Stopwatch();
                t.Start();
                string text = File.ReadAllText(FileDownload.FileName);
                char[] separators = new char[] { ' ', '.', ',', '!', '?', '/', '\\t', '\\n' };
                string[] textArray = text.Split(separators);
                foreach (string strTemp in textArray)
                {
                    string str = strTemp.Trim();
                    if (!list.Contains(str)) list.Add(str);
                }
                t.Stop();
                this.textBoxFileReadTime.Text = t.Elapsed.ToString();
                this.textBoxFileReadCount.Text = list.Count.ToString();
            }
            else
            {
                MessageBox.Show("Необходимо выбрать файл");
            }
        }

        private void button2_Click(object sender, EventArgs e)
        {
            string word = this.textBoxFind.Text.Trim();

            //Если слово для поиска не пусто
            if (!string.IsNullOrEmpty(word) && list.Count > 0)
            {
                //Слово для поиска в верхнем регистре
            }
        }
    }
}
```

```

        string wordUpper = word.ToUpper();
        //Временные результаты поиска
        List<string> tempList = new List<string>();
        Stopwatch t = new Stopwatch();
        t.Start();
        foreach (string str in list)
        {
            if (str.ToUpper().Contains(wordUpper))
            {
                tempList.Add(str);
            }
        }
        t.Stop();
        this.textBoxExactTime.Text = t.Elapsed.ToString();
        this.listBoxResult.BeginUpdate();
        //Очистка списка
        this.listBoxResult.Items.Clear();
        //Вывод результатов поиска
        foreach (string str in tempList)
        {
            this.listBoxResult.Items.Add(str);
        }
        this.listBoxResult.EndUpdate();
    }
    else
    {
        MessageBox.Show("Необходимо выбрать файл и ввести слово для поиска");
    }
}
}
}

```

## Program.cs

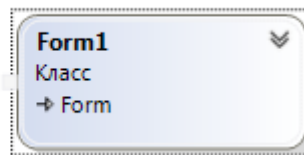
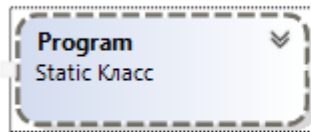
```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Threading.Tasks;
using System.Windows.Forms;

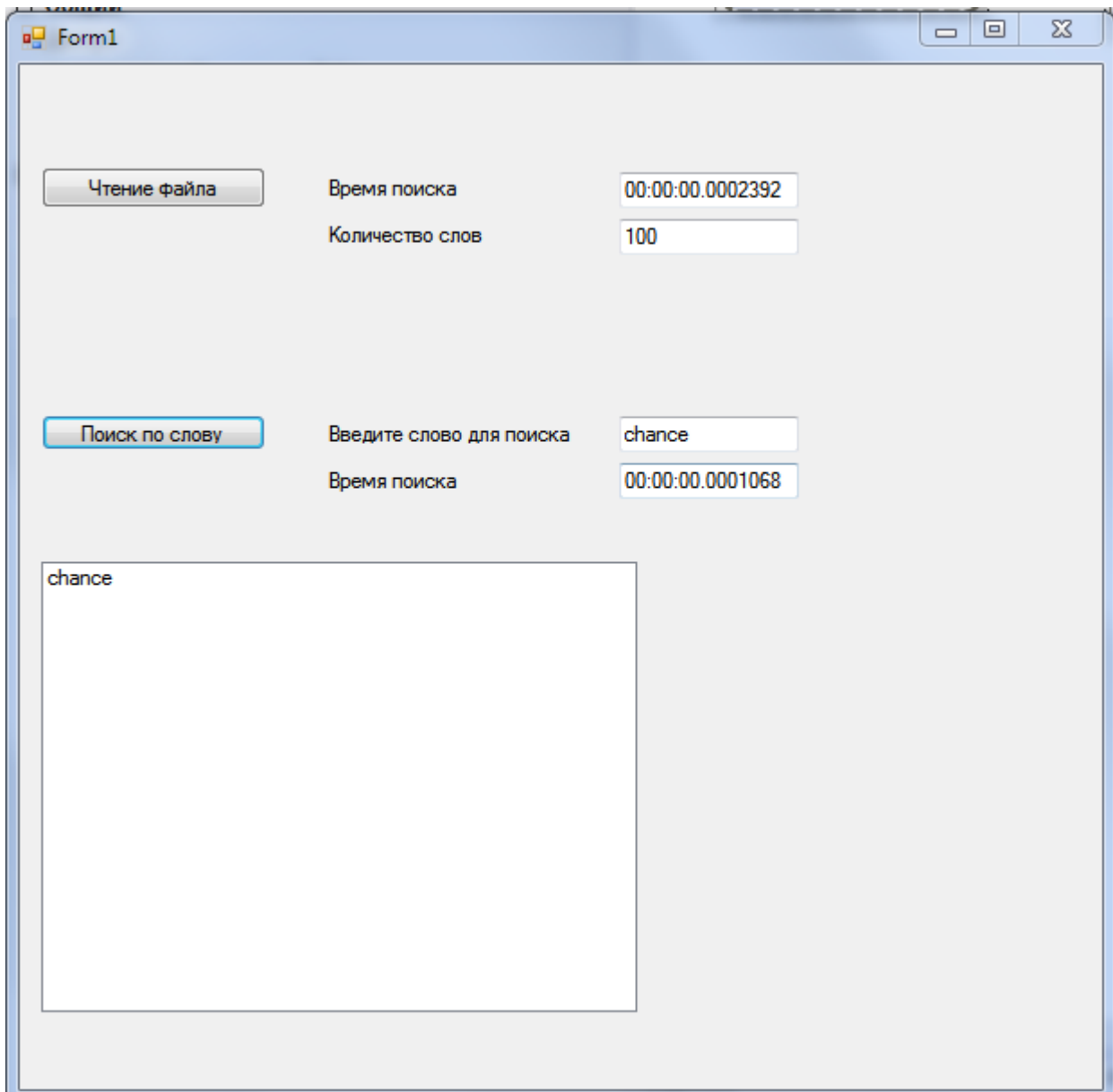
namespace WindowsFormsApplication3
{
    static class Program
    {
        /// <summary>
        /// Главная точка входа для приложения.
        /// </summary>
        [STAThread]
        static void Main()
        {
            Application.EnableVisualStyles();
            Application.SetCompatibleTextRenderingDefault(false);
            Application.Run(new Form1());
        }
    }
}

```

## Диаграмма классов:



## Пример консольного вывода:



The screenshot shows a Windows application window titled "Form1". The interface contains two main sections. The top section has a button "Чтение файла" (File Reading) and two labels: "Время поиска" (Search Time) with a value of "00:00:00.0002392" and "Количество слов" (Number of words) with a value of "100". The bottom section has a button "Поиск по слову" (Search by word) and two labels: "Введите слово для поиска" (Enter word for search) with a value of "chance" and "Время поиска" (Search Time) with a value of "00:00:00.0001068". Below these labels is a text box containing the word "chance".