

ЛАБОРАТОРНАЯ РАБОТА 2

ТЕКСТОВЫЕ ФАЙЛЫ, СИМВОЛЫ, СТРОКИ

Задание

Создать текстовый файл, содержащий текст.

Программа должна выводить текст из файла на экран, сохранять измененный текст, позволять задавать два слова, как первое и второе, которые нужно найти и выделить в тексте, а затем там же поменять их местами.

Проектирование приложения.


Для работы с файлами воспользуемся возможностями пространства имен System.IO, который содержит типы, позволяющие осуществлять чтение и запись в файлы и потоки данных, а также типы для базовой поддержки файлов и папок. Так же для работы необходимы компоненты openFileDialog и saveFileDialog предоставляемые средой.

Выводить текст, имея в виду последующую работу с текстом, целесообразно в многострочное окно редактирования – компонент richTextBox, который работает с текстом в обогащенном формате RTF.

Для регистрации хода выполнения программы воспользуемся компонентом textBox, имеющим свойство Multiline, позволяющее выводить многострочные данные.

Контекстное меню будет создаваться программно, посредством функционала класса ContextMenu. Всплывающее меню должно содержать команды Сохранить и Сохранить как и Открыть.

Первое и второе слова, которые необходимо вписать в окнах textBox, а затем там же поменять их местами.

1. Запустите VS.
2. Создайте новый проект.
3. Выделите форму, щелкнув на ней левой кнопкой мыши, и в свойство **Text** впишите *Текстовые файлы, символы и строки*.
4. Перенесите на форму компоненты openFileDialog1 и saveFileDialog1. Все диалоги являются невизуальными компонентами. При обращении к этим компонентам вызываются стандартные диалоги.
5. Перенесите на форму 3 компонента textBox и один richTextBox. Для одного из компонентов textBox, который будет отвечать за вывод информации о ходе работы программы, установите свойство Multiline в позицию true.
6. Для реализации контекстного меню, необходимо создать обработчик события Form1_Load. Для этого перейдите на вкладку События () . Найдите событие Load и создайте функцию-обработчик, выбрав из выпадающего списка Form1_Load.
7. Добавьте 4 компонента button. Первая кнопка будет отвечать за очистку всех полей, вторая за поиск первого слова, третья за поиск второго, четвертая осуществит замену слов.

8. По окончании проектирования, форма примет вид, представленный на рисунке 2.1.

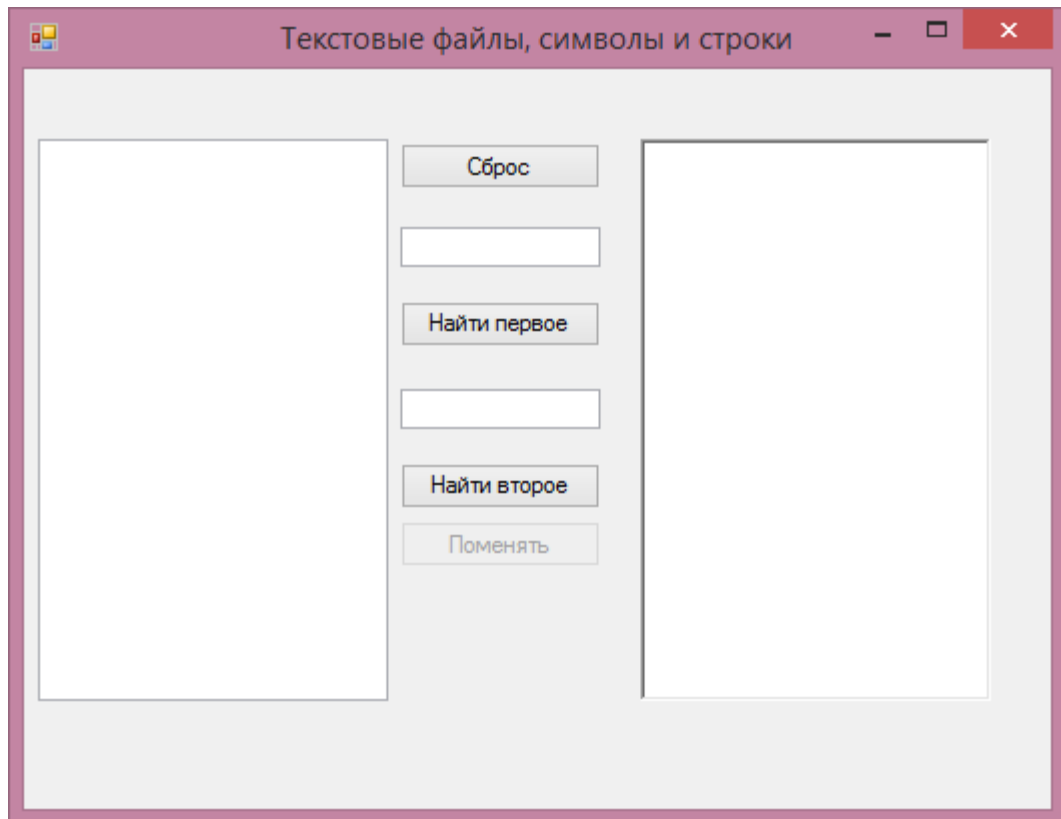


Рисунок 2.1 – Форма по окончании проектирования

9. В `public Form1()` введите следующий код:

```
button4.Enabled = false;
```

10. В обработчике события `Form1_Load`, отвечающего за создание контекстного меню при загрузке формы введите следующий код:

```
private void Form1_Load(object sender, EventArgs e)
{
    //создание контекстного меню
    System.Windows.Forms.ContextMenu contextMenu1;
    contextMenu1 = new System.Windows.Forms.ContextMenu();

    System.Windows.Forms.MenuItem menuItem1;
    menuItem1 = new System.Windows.Forms.MenuItem();
    System.Windows.Forms.MenuItem menuItem2;
    menuItem2 = new System.Windows.Forms.MenuItem();
    System.Windows.Forms.MenuItem menuItem3;
    menuItem3 = new System.Windows.Forms.MenuItem();

    contextMenu1.MenuItems.AddRange(new System.Windows.Forms.MenuItem[] {
menuItem1, menuItem2, menuItem3 });
    menuItem1.Index = 0;
    menuItem1.Text = "Открыть";
    menuItem2.Index = 1;
    menuItem2.Text = "Сохранить";
    menuItem3.Index = 2;
    menuItem3.Text = "Сохранить как";

    richTextBox1.ContextMenu = contextMenu1;
    menuItem1.Click += new System.EventHandler(this.menuItem1_Click);
    menuItem2.Click += new System.EventHandler(this.menuItem2_Click);
}
```

```
menuItem3.Click += new System.EventHandler(this.menuItem3_Click);
```

```
}
```

11. Введите код, отвечающий за обработку событий выбора позиций в контекстном меню. Так же необходимо вне функций, объявить переменную MyFName и добавить пространство имен `using System.IO`.

```
string MyFName = "";
private void menuItem1_Click(object sender, System.EventArgs e)
{
    openFileDialog1.Filter = "Текстовые файлы (*.rtf; *.txt; *.dat) | *.rtf;
*.txt; *.dat";
    if (openFileDialog1.ShowDialog() == DialogResult.OK)
    {
        MyFName = openFileDialog1.FileName;
        richTextBox1.LoadFile(MyFName);
    }
}

private void menuItem2_Click(object sender, EventArgs e)
{
    if (MyFName != "")
    {
        richTextBox1.SaveFile(MyFName);
    }
    else
    {
        saveFileDialog1.Filter = "Текстовые файлы (*.rtf; *.txt; *.dat) | *.rtf;
*.txt; *.dat";
        if (saveFileDialog1.ShowDialog() == DialogResult.OK)
        {
            MyFName = saveFileDialog1.FileName;
            richTextBox1.SaveFile(MyFName);
        }
    }
}

private void menuItem3_Click(object sender, System.EventArgs e)
{
    saveFileDialog1.Filter = "Текстовые файлы (*.rtf; *.txt; *.dat) | *.rtf;
*.txt; *.dat";
    if (saveFileDialog1.ShowDialog() == DialogResult.OK)
    {
        MyFName = saveFileDialog1.FileName;
        richTextBox1.SaveFile(MyFName);
    }
}
```

12. Код события, отвечающего за очистку всех полей формы:

```
private void button1_Click(object sender, EventArgs e)
{
    textBox1.Clear();
    textBox2.Clear();
    textBox3.Clear();
    richTextBox1.Clear();
    button2.Enabled = true;
    button3.Enabled = true;
    button4.Enabled = true;
}
```

13. Код события, отвечающего за нахождение и выделения первого слова:

```
int result1, result2;
private void button2_Click(object sender, EventArgs e)
{
    int LenText;
    textBox3.Text += "Поиск первого слова" + Environment.NewLine;
    String FWord = textBox1.Text.ToString();
    LenText = richTextBox1.Text.Length;
    result1 = FindWord(FWord, LenText);
    if(result1 != -1)
    {
        textBox3.Text += "Позиция первого слова: " + (result1+1) +
Environment.NewLine + Environment.NewLine;
        richTextBox1.SelectionStart = result1;
        richTextBox1.SelectionLength = FWord.Length;
        richTextBox1.SelectionBackColor = Color.Red;
        button2.Enabled = false;
        if (button3.Enabled == false)
        { button4.Enabled = true; }
    }
    else
    {
        textBox3.Text += "Слово не найдено " + Environment.NewLine
+ Environment.NewLine;
    }
}
```

14. Код события, отвечающего за нахождение и выделения второго слова:

```
private void button3_Click(object sender, EventArgs e)
{
    int LenText;

    textBox3.Text += "Поиск второго слова" + Environment.NewLine;
    String FWord = textBox2.Text.ToString();
    LenText = richTextBox1.Text.Length;
    result2 = FindWord(FWord, LenText);
    if (result2 != -1)
    {
        textBox3.Text += "Позиция второго слова: " + (result2+1) +
Environment.NewLine + Environment.NewLine;
    }
}
```

```

        richTextBox1.SelectionStart = result2;
        richTextBox1.SelectionLength = FWord.Length;
        richTextBox1.SelectionBackColor = Color.Green;
        button3.Enabled = false;
        if (button2.Enabled == false )
        { button4.Enabled = true; }
    }
    else
    {
        textBox3.Text += "Слово не найдено " + Environment.NewLine +
Environment.NewLine;
    }
}

```

15. Код события, отвечающего за замену слов местами:

```

private void button4_Click(object sender, EventArgs e)
{
    if (result1 < result2)
    {
        richTextBox1.Select(result2, textBox2.Text.Length);
        richTextBox1.SelectedText = textBox1.Text.ToString();
        richTextBox1.Select(result1, textBox1.Text.Length);
        richTextBox1.SelectedText = textBox2.Text.ToString();
        textBox3.Text += "Произошла замена слов";
        button4.Enabled = false;
    }
    else
    {
        richTextBox1.Select(result1, textBox1.Text.Length);
        richTextBox1.SelectedText = textBox2.Text.ToString();
        richTextBox1.Select(result2, textBox2.Text.Length);
        richTextBox1.SelectedText = textBox1.Text.ToString();
        textBox3.Text += "Произошла замена слов";
        button4.Enabled = false;
    }
}

```

16. Код функции FindWord, которая осуществляет поиск слова и возвращает номер позиции начала слова, в случае неудачи, функция вернет -1. Аргументами, передаваемыми в функцию, является слово для поиска и длина текста.

```

int FindWord(String FWord, int n)
{
    int LenWord;
    String ComparText;
    LenWord = FWord.Length;
    for (int i = 0; i <= n - LenWord; i++)
    {
        ComparText = richTextBox1.Text.Substring(i, LenWord);
        if (ComparText == FWord)
        {
            return i;
        }
    }
    return -1;
}

```

17. По окончании проектирования код программы будет выглядеть следующим образом:

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.IO;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace TextFiles
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
            button4.Enabled = false;
        }

        private void Form1_Load(object sender, EventArgs e)
        {
            //создание контекстного меню
            System.Windows.Forms.ContextMenu contextMenu1;
            contextMenu1 = new System.Windows.Forms.ContextMenu();

            System.Windows.Forms.MenuItem menuItem1;
            menuItem1 = new System.Windows.Forms.MenuItem();
            System.Windows.Forms.MenuItem menuItem2;
            menuItem2 = new System.Windows.Forms.MenuItem();
            System.Windows.Forms.MenuItem menuItem3;
            menuItem3 = new System.Windows.Forms.MenuItem();

            contextMenu1.MenuItems.AddRange(new System.Windows.Forms.MenuItem[] {
menuItem1, menuItem2, menuItem3 });
            menuItem1.Index = 0;
            menuItem1.Text = "Открыть";
            menuItem2.Index = 1;
            menuItem2.Text = "Сохранить";
            menuItem3.Index = 2;
            menuItem3.Text = "Сохранить как";

            richTextBox1.ContextMenu = contextMenu1;
            menuItem1.Click += new System.EventHandler(this.menuItem1_Click);
            menuItem2.Click += new System.EventHandler(this.menuItem2_Click);
            menuItem3.Click += new System.EventHandler(this.menuItem3_Click);

        }

        string MyFName = "";

        private void menuItem1_Click(object sender, System.EventArgs e)
        {

```

```

        openFileDialog1.Filter = "Текстовые файлы (*.rtf; *.txt; *.dat) | *.rtf;
*.txt; *.dat";
        if (openFileDialog1.ShowDialog() == DialogResult.OK)
        {
            MyFName = openFileDialog1.FileName;
            richTextBox1.LoadFile(MyFName);
        }
    }

    private void menuItem2_Click(object sender, EventArgs e)
    {
        if (MyFName != "")
        {
            richTextBox1.SaveFile(MyFName);
        }
        else
        {
            saveFileDialog1.Filter = "Текстовые файлы (*.rtf; *.txt; *.dat) | *.rtf;
*.txt; *.dat";
            if (saveFileDialog1.ShowDialog() == DialogResult.OK)
            {
                MyFName = saveFileDialog1.FileName;
                richTextBox1.SaveFile(MyFName);
            }
        }
    }

    private void menuItem3_Click(object sender, System.EventArgs e)
    {
        saveFileDialog1.Filter = "Текстовые файлы (*.rtf; *.txt; *.dat) | *.rtf;
*.txt; *.dat";
        if (saveFileDialog1.ShowDialog() == DialogResult.OK)
        {
            MyFName = saveFileDialog1.FileName;
            richTextBox1.SaveFile(MyFName);
        }
    }

    private void button1_Click(object sender, EventArgs e)
    {
        textBox1.Clear();
        textBox2.Clear();
        textBox3.Clear();
        richTextBox1.Clear();
        button2.Enabled = true;
        button3.Enabled = true;
        button4.Enabled = true;
    }

    int result1, result2;
    private void button2_Click(object sender, EventArgs e)
    {
        int LenText;
        textBox3.Text += "Поиск первого слова" + Environment.NewLine;
        String FWord = textBox1.Text.ToString();
        LenText = richTextBox1.Text.Length;
    }

```

```

        result1 = FindWord(FWord, LenText);
        if(result1 != -1)
        {
            textBox3.Text += "Позиция первого слова: " + (result1+1) +
Environment.NewLine + Environment.NewLine;
            richTextBox1.SelectionStart = result1;
            richTextBox1.SelectionLength = FWord.Length;
            richTextBox1.SelectionBackColor = Color.Red;
            button2.Enabled = false;
            if (button3.Enabled == false)
            { button4.Enabled = true; }
        }
        else
        {
            textBox3.Text += "Слово не найдено " + Environment.NewLine +
Environment.NewLine;
        }
    }

    private void button3_Click(object sender, EventArgs e)
    {
        int LenText;

        textBox3.Text += "Поиск второго слова" + Environment.NewLine;
        String FWord = textBox2.Text.ToString();
        LenText = richTextBox1.Text.Length;
        result2 = FindWord(FWord, LenText);
        if (result2 != -1)
        {
            textBox3.Text += "Позиция второго слова: " + (result2+1) +
Environment.NewLine + Environment.NewLine;
            richTextBox1.SelectionStart = result2;
            richTextBox1.SelectionLength = FWord.Length;
            richTextBox1.SelectionBackColor = Color.Green;
            button3.Enabled = false;
            if (button2.Enabled == false )
            { button4.Enabled = true; }
        }
        else
        {
            textBox3.Text += "Слово не найдено " + Environment.NewLine +
Environment.NewLine;
        }
    }

    int FindWord(String FWord, int n)
    {
        int LenWord;
        String ComparText;
        LenWord = FWord.Length;
        for (int i = 0; i <= n - LenWord; i++)
        {
            ComparText = richTextBox1.Text.Substring(i, LenWord);
            if (ComparText == FWord)
            {

```



```

        return i;
    }
}
return -1;
}

private void button4_Click(object sender, EventArgs e)
{
    if (result1 < result2)
    {
        richTextBox1.Select(result2, textBox2.Text.Length);
        richTextBox1.SelectedText = textBox1.Text.ToString();
        richTextBox1.Select(result1, textBox1.Text.Length);
        richTextBox1.SelectedText = textBox2.Text.ToString();
        textBox3.Text += "Произошла замена слов";
        button4.Enabled = false;
    }
    else
    {
        richTextBox1.Select(result1, textBox1.Text.Length);
        richTextBox1.SelectedText = textBox2.Text.ToString();
        richTextBox1.Select(result2, textBox2.Text.Length);
        richTextBox1.SelectedText = textBox1.Text.ToString();
        textBox3.Text += "Произошла замена слов";
        button4.Enabled = false;
    }
}
}
}

```

Тестирование и использование приложения

Пример выполнения приложения представлен на рисунках 2.2, 2.3.

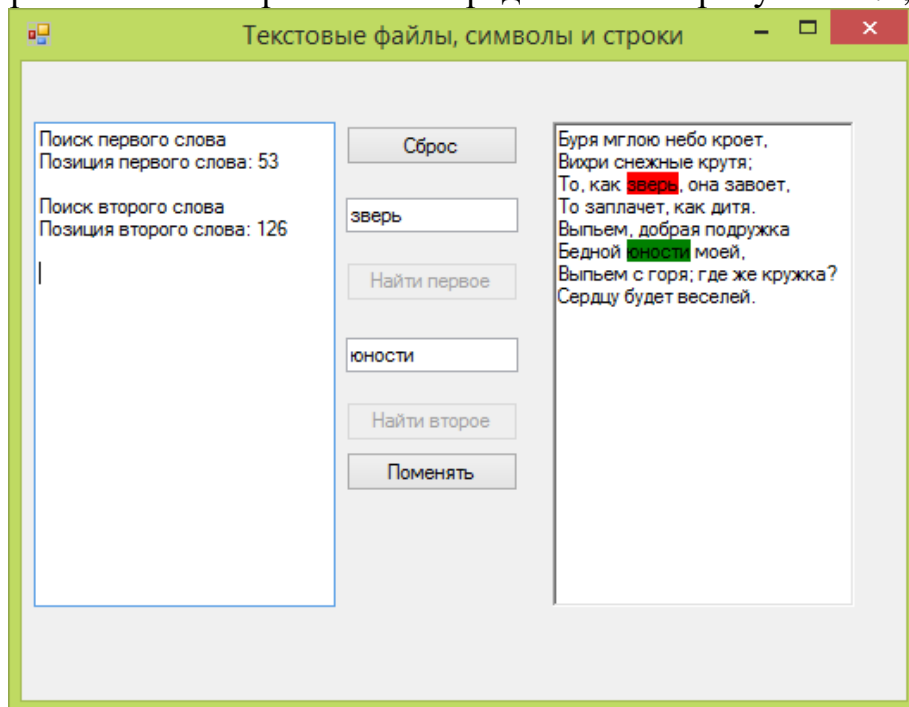


Рис.2.2 – Заданные первое и второе слова найдены и выделены в тексте

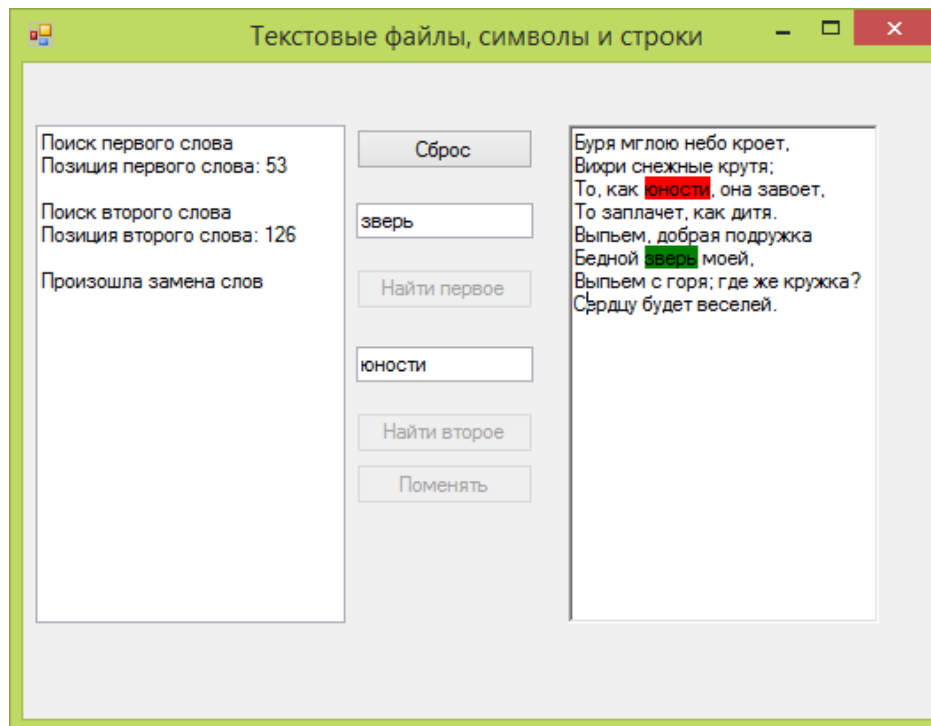


Рис.2.3 – Найденные слова переставлены местами

1. Запустите приложение на выполнение, нажав быстрые кнопки Сохранить все и Запуск. Нажмите кнопку СБРОС.
2. Наберите текст в окне richTextBox для вывода в файл. Правой кнопкой мыши щелкните на окне и во всплывшем меню выберите нужную команду (сохранить или сохранить как). Сохраните текст в файле с расширением .dat, .txt, .rtf.
3. Сотрите текст, щелкнув на кнопке СБРОС. Щелкните правой кнопкой мыши на окне richTextBox и прочитайте файл.
4. В окно первое слово введите слово из текста и нажмите кнопку найти первое слово.
5. В окно второе слово введите слово из текста и нажмите кнопку найти второе слово. Результат представлен на рис.2.2.
6. Нажатием кнопки поменять завершаем выполнение задания (рис.2.3).

Контрольные вопросы

1. Поясните назначение и использование компонентов “Сохранить файл” и “Открыть файл”.
2. Поясните реализацию контекстного всплывающее меню. Как осуществляется связь этого компонента с окном richTextBox?
3. Как используются компоненты richTextBox и textBox при выполнении задания?
4. Представьте блок-схему алгоритма, реализованного функцией FindWord(). Расскажите по алгоритму, как осуществляется поиск слова.
5. Как, когда и где вызывается функция FindWord()?

6. Объясните, как переставляются местами найденные в тексте слова. Какие методы при этом используются?
7. Поясните назначение глобальных переменных.
8. Как сохранить в файле текст, представленный в richTextBox?
9. Как изменить размер шрифта и цвет выделенных в тексте слов?

Задания

Написать программу, которая:

- а) выводит текст на экран дисплея и сохранять его;
- б) произвольная клавиша, это любая клавиша на клавиатуре.
- в) далее – по варианту.

1. По нажатию произвольной клавиши поочередно выделяет каждое предложение текста; определяет количество предложений в тексте.
2. По нажатию произвольной клавиши поочередно выделяет каждое слово текста; определяет количество слов в тексте.
3. По нажатию произвольной клавиши поочередно выделяет каждое слово текста, оканчивающееся на гласную букву; определяет количество таких слов в тексте.
4. По нажатию произвольной клавиши поочередно выделяет каждое предложение текста в последовательности 2, 1, 3.
5. По нажатию произвольной клавиши поочередно выделяет каждое из слов текста, у которых первый и последний символы совпадают; определяет количество таких слов в тексте.
6. По нажатию произвольной клавиши поочередно выделяет каждое слово текста, начинающееся на гласную букву; определяет количество таких слов в тексте.
7. Определяет количество символов в самом длинном слове; по нажатию произвольной клавиши поочередно выделяет каждое слово текста, содержащее максимальное количество символов.
8. Определяет количество символов в самом коротком слове; по нажатию произвольной клавиши поочередно выделяет каждое слово текста, содержащее минимальное количество символов.
9. Определяет в каждом предложении текста количество символов, отличных от букв и пробела; по нажатию произвольной клавиши поочередно выделяет каждое предложение текста, а в выделенном предложении – поочередно все символы, отличные от букв и пробела.
10. Определяет количество предложений текста и количество слов в каждом предложении; по нажатию произвольной клавиши поочередно выделяет каждое предложение текста, а в выделенном предложении – поочередно все слова.

11. Определяет количество букв 'а' в последнем слове текста; по нажатию произвольной клавиши выделяет последнее слово текста, а в выделенном слове – поочередно все буквы 'а'.

12. Определяет самую длинную последовательность цифр в тексте (считать, что любое количество пробелов между двумя цифрами не прерывает последовательности цифр); по нажатию произвольной клавиши поочередно выделяет каждую последовательность цифр, содержащую максимальное количество символов.

13. Определяет порядковый номер заданного слова в каждом предложении текста (заданное слово вводится с клавиатуры); по нажатию произвольной клавиши поочередно выделяет каждое предложение текста, а в выделенном предложении – заданное слово.

14. По нажатию произвольной клавиши поочередно выделяет в тексте заданное слово (заданное слово вводить с клавиатуры); выводит текст на экран дисплея ещё раз, выкидывая из него заданное слово и удаляя лишние пробелы.

15. По нажатию произвольной клавиши поочередно выделяет в тексте заданные слова, которые нужно поменять местами (заданные слова вводить с клавиатуры); выводит текст на экран дисплея ещё раз, меняя в нём местами заданные слова и удаляя лишние пробелы.

16. По нажатию произвольной клавиши поочередно выделяет в тексте заданное слово (заданное слово вводить с клавиатуры); выводит текст на экран дисплея ещё раз, заключая заданное слово в кавычки, и поочередно выделяет заданное слово вместе с кавычками.

17. Выводит текст на экран дисплея ещё раз, вставляя в каждое предложение в качестве последнего заданное слово, введенное с клавиатуры в качестве исходных данных; по нажатию произвольной клавиши поочередно выделяет в тексте вставленное слово.

18. По нажатию произвольной клавиши поочередно выделяет в тексте лишние пробелы между словами; выводит текст на экран дисплея ещё раз, удаляя лишние пробелы между словами и начиная каждое предложение с новой строки.

19. По нажатию произвольной клавиши поочередно выделяет в тексте заданное слово (заданное слово вводится с клавиатуры); выводит текст на экран дисплея ещё раз, заменяя в заданном слове строчные буквы прописными.

20. Определяет наибольшее количество подряд идущих пробелов в тексте; по нажатию произвольной клавиши поочередно выделяет каждую из последовательностей пробелов максимальной длины.

21. Определяет в каждой строке текста количество прописных букв; по нажатию произвольной клавиши поочередно выделяет каждое слово, начинающееся с прописной буквы, а в выделенном слове – прописные буквы.

22. По нажатию произвольной клавиши поочередно выделяет в тексте слово с заданной буквой; выводит на экран дисплея ещё раз те слова, в которых заданная буква встречается более одного раза.

23. По нажатию произвольной клавиши поочередно выводит фрагменты текста, отделенные знаками препинания; выводит на экран дисплея сведения о знаках препинания по строкам в виде: знак препинания – количество.

24. По нажатию произвольной клавиши поочередно выводит построчно фрагменты текста, разделенные символом горизонтальной табуляции; выводит на экран дисплея общее количество символов табуляции в тексте.

25. Выводит текст на экран дисплея ещё раз, разделяя знаками переноса каждое слово на слоги; по нажатию произвольной клавиши поочередно выделяет в каждой строке текста слово с наибольшим количеством слогов.

26. По нажатию произвольной клавиши поочередно выделяет в тексте слова, после которых стоит знак препинания; выводит текст на экран ещё раз, выделяя знаки препинания.

27. По нажатию произвольной клавиши выводит количество десятичных чисел по строкам; выводит текст на экран дисплея ещё раз, заменяя десятичные числа на шестнадцатеричные.

28. По нажатию произвольной клавиши поочередно выделяет каждое число в тексте; выводит текст на экран дисплея ещё раз, заменяя числа пробелами.

29. По нажатию произвольной клавиши поочередно выделяет в тексте слова с заданной буквой (вводится с клавиатуры); выводит на экран дисплея ещё раз те слова, в которых нет заданной буквы.

30. По нажатию произвольной клавиши поочередно выделяет в тексте каждые первое и второе слово с первыми строчными гласными буквами.