Лабораторная работа №2

«Построение лексического блока»

**Задание:**

Разработать программу, реализующую лексический блок для заданного языка.

Длина идентификатора - не менее 10 литер

Лексический блок реализуется как подпрограмма, вызываемая синтаксическим анализатором.

Лексический блок должен строить таблицы идентификаторов и констант

**Оператор присваивания:**

<ид> := <ар. выр.>;

**Условный оператор:**

if (<лог. выр.>) <оператор> [ else <оператор> ]

**Оператор цикла:**

ЦИКЛ <совокупность операторов> ПОКА\_НЕ <лог. выр.>;

**Арифметическое выражение:**

<E> := <T> <E-список>

<E-список> := -<T> <E-список>

<E-список>:= !

<T> := <F> <T-список>

<T-список> := / <F><T-список>

<T-список> := !

<F> := <Id>

<F> := <Int>

**Логическое выражение**

<лог. выр.> := <F><лог. опер.><F>

< лог. опер.> := >

< лог. опер.> := <

**Пример программы:**

d := 23 - 1;

a := 3 - 254 / d;

c := 200 / d -367 / a;

if (d > c) a := d - 2; else a: = d / 3;

h = 10;

ЦИКЛ

h := h - 1;

ПОКА\_НЕ h<0;

Возможны следующие виды лексем:

- идентификаторы

- ключевые слова

- целые числа

- разделители

**Описание работы программы:**

private void button2\_Click(object sender, EventArgs e) - Функция, вызываемая при нажатии на кнопку “Открыть”. Считывает файл в строку

private void button1\_Click(object sender, EventArgs e) - Функция, вызываемая при нажатии на кнопку ”Ввод“. Очищает таблицы лексем. И заполняет их после выполнения программы.

public void Analiz(string s) - функция вызываемая из private void button1\_Click. Анализирует полученную строку и выделяет из неё лексемы. После того как лексема найдена вызывается public void Add1(int el1, string el2) в любом случае и public void Add2(int el1, string el2) в случае если лексема оказалось идентификатором или константой

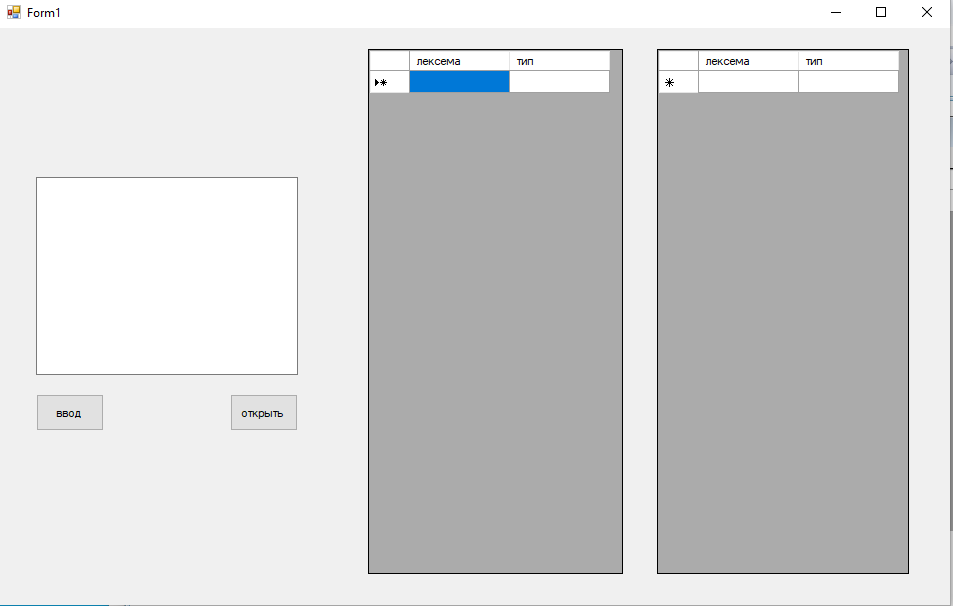
public void Add1(int el1, string el2) - Функция, получающая на вход лексему и ёё тип и добавляющая её в массив, или не добавляющая если в лексеме обнаружена ошибка.

public void Add2(int el1, string el2) - Функция, получающая на вход лексему и ёё тип(в случае если лексема является идентификатором или константой) и добавляющая её в массив, сортируя все добавленные лексемы по алфавиту.

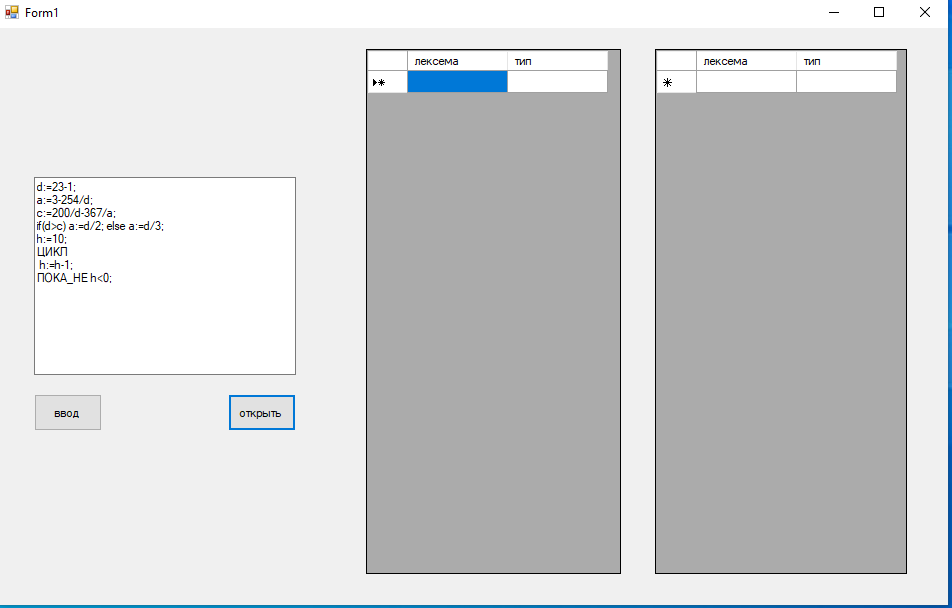
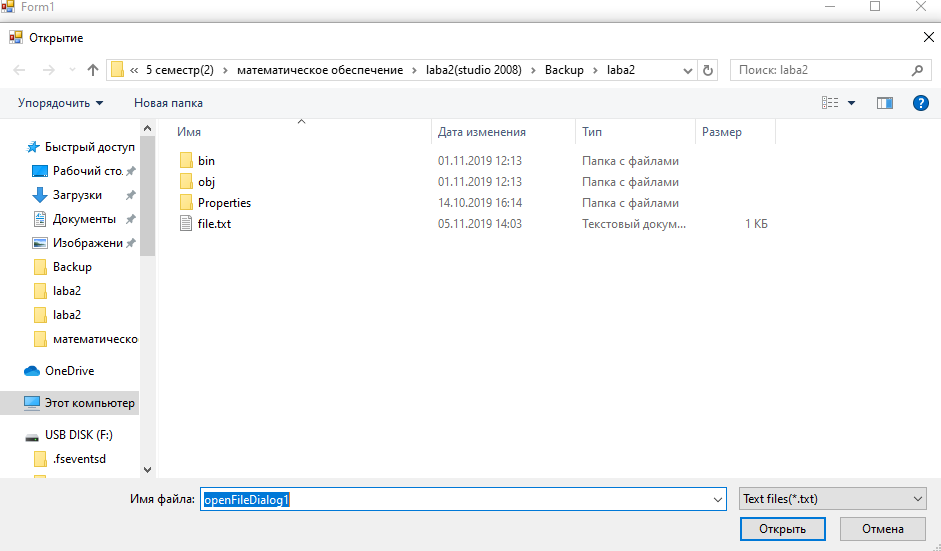
public int Inde(char nu1) - вспомогательная функция вызываемая из public void Add2(int el1, string el2) для сортировки. Она выдаёт номер входного символа в алфавите, который затем используется в Add2 для сортировки.

**Тесты:**

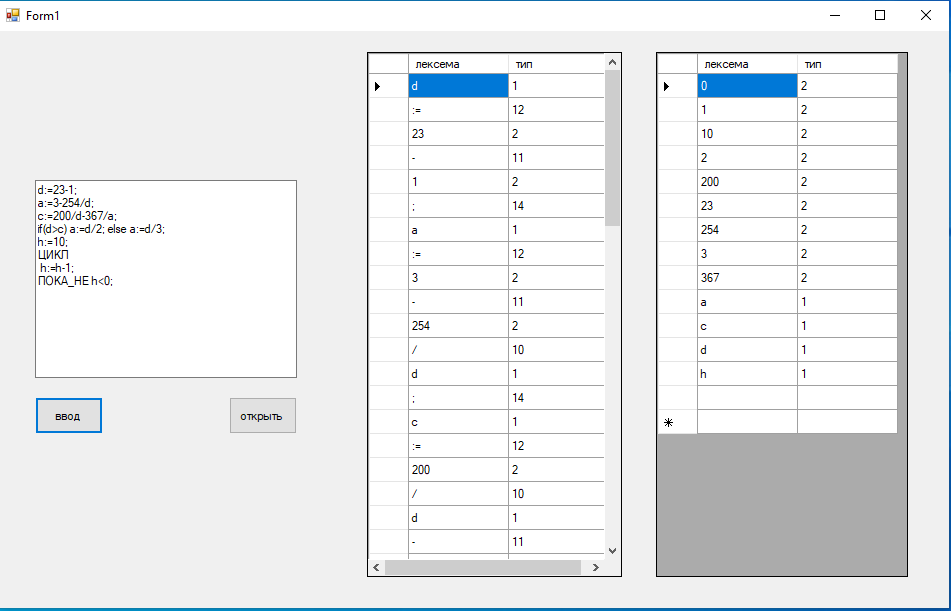
Главное меню программы



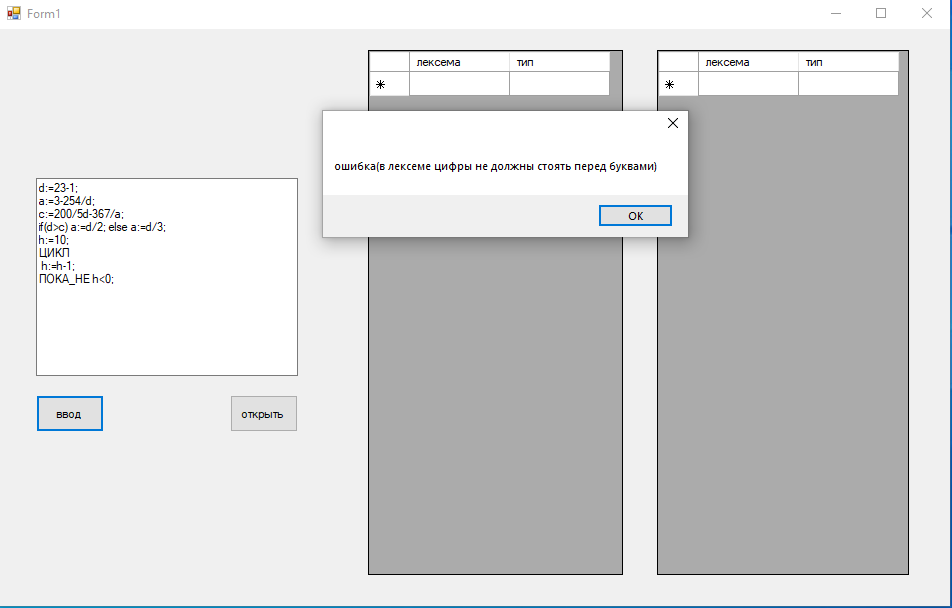
Открытие файла



Тест1



Тест2(с ошибкой в лексеме)



Тест 3

